



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Inclusión dinámica de las preferencias del decisor en un algoritmo genético multiobjetivo mediante un SID

Jaime Andrés Díaz Guerra

Universidad Nacional de Colombia
Facultad de Minas, Departamento de Ciencias de la Computación y de la Decisión
Medellín, Colombia

2024

Inclusión dinámica de las preferencias del decisor en un algoritmo genético multiobjetivo mediante un SID

Jaime Andrés Díaz Guerra

Tesis presentada como requisito parcial para optar al título de:

Magister en Ingeniería – Ingeniería de Sistemas

Director (a):

Ph.D. Gloria Patricia Jaramillo Álvarez

Línea de Investigación:

Optimización Multiobjetivo

Grupo de Investigación:

Ciencias de la Decisión

Universidad Nacional de Colombia

Facultad de Minas, Departamento de Ciencias de la Computación y de la Decisión

Medellín, Colombia

2024

“Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad”.

Albert Einstein

Agradecimientos

A Dios por todo lo que he sido, soy y seré.

A Jaime y a María Elena, quienes aun en la trascendencia me inspiran y nutren mi ser.

A Juan Felipe Díaz, Iván Darío Loaiza, Gloria Lucía Guerra, Juliana Guerra, Natalia Guerra y a María Camila Bedoya cuyo apoyo, compañía y amor incondicional hacen posible que mi vida sea feliz y que mi confianza se fortalezca para seguir creciendo cada día.

A mis amigos y amigas por creer permanentemente en mí.

A la Universidad Nacional de Colombia a quien debo el ser académico que habita en mí.

Finalmente, si tuviera que crear una palabra en un nuevo idioma, para definir un inmenso y desbordado sentido de generosidad, sin dudar lo crearía con las letras de PATRICIA.

Este trabajo es posible gracias a mi directora y asesora, Gloria Patricia Jaramillo Álvarez, quien de forma arriesgada se atrevió a confiar en el camino altamente no lineal, difuso y complejo de mi proceso académico.

Resumen

Inclusión dinámica de las preferencias del decisor en un algoritmo genético multiobjetivo mediante un SID

El presente trabajo propone una metodología que permite que un decisor incluya las preferencias que experimenta sobre un problema, utilizando un Sistema de Inferencia Difusa en un algoritmo genético de optimización multiobjetivo. Esta metodología se logra a través del diseño de un método denominado Algoritmo Genético Multiobjetivo con Sistema de Inferencia de Preferencias Difusas (AGMOSIPD). Este algoritmo es una técnica de incorporación de preferencias a priori que le ofrece al decisor una etapa de aprendizaje inicial donde, a través de la obtención de un conjunto de soluciones a través de simulación Monte Carlo, podrá construir su estructura de preferencias mediante el diseño de un Sistema de Inferencia Difusa (SID). Posteriormente, el SID se incorpora en un algoritmo genético a través de una restricción para dirigir los individuos hacia la zona de la Frontera de Pareto más preferida por el decisor y ofrecer un conjunto reducido de alternativas. Este trabajo se ejecuta en 60 casos de prueba que involucran 6 problemas y 2 algoritmos genéticos, se presentan los resultados gráficos, se verifica la obtención de soluciones eficientes y se comparan las soluciones obtenidas mediante AGMOSIPD con las soluciones obtenidas a través de la optimización de los problemas de prueba en un algoritmo genético sin preferencias. AGMOSIPD obtiene soluciones eficientes en la mayoría de los casos probados y presenta desafíos y oportunidades de mejora en otras circunstancias.

Palabras clave: Algoritmos Genéticos, Lógica Difusa, Optimización Multiobjetivo, Preferencias, Sistema de Inferencia Difusa.

Abstract

Dynamic inclusion of decision-maker preferences in a multi-objective genetic algorithm using a FIS

This work proposes a methodology that includes the decision maker preferences about a problem using a Fuzzy Inference System in a multiobjective genetic optimization algorithm. This methodology is achieved through the design of a method called Multiobjective Genetic Algorithm with Fuzzy Preference Inference System (AGMOSIPD). This algorithm is an a priori preference incorporation technique that offers an initial learning stage where, by obtaining a set of solutions through Monte Carlo simulation, the decision maker can build a preference structure through the design of a Fuzzy Inference System (FIS). Subsequently, the FIS is incorporated into a genetic algorithm through a constraint to direct the individuals towards the Pareto Frontier zone most preferred by the decision maker and offer a reduced set of alternatives. This work is run on 60 test cases involving 6 problems and 2 genetic algorithms, the graphical results are presented, the obtaining of efficient solutions is verified, and the solutions obtained through AGMOSIPD are compared with the solutions obtained through the optimization of test problems in a genetic algorithm without preferences. AGMOSIPD obtains efficient solutions in most of the tested cases and presents challenges and opportunities for improvement in other circumstances.

Keywords: Genetic Algorithms, Fuzzy Logic, Multiobjective Optimization, Preferences, Fuzzy Inference System.

Contenido

	Pág.
Resumen	VII
Lista de figuras.....	XI
Lista de tablas	XIII
Lista de Símbolos y abreviaturas	XV
Introducción	1
1. Marco Teórico	9
1.1 Optimización Multiobjetivo	9
1.2 Técnicas Metaheurísticas	12
1.3 Algoritmos Genéticos.....	13
1.4 Nondominated Sorting Genetic Algorithm II y III.....	16
1.5 Incorporación de Preferencias en MOEA	19
1.5.1 Obtención e Incorporación de Preferencias Precisas	20
1.5.2 Obtención e Incorporación de Preferencias Imprecisas	27
1.6 Conclusión.....	38
2. Propuesta de Metodología.....	41
2.1 Elección de Método de Asignación de Preferencias	42
2.2 Construcción de Estructura de Preferencias	44
2.3 Elección del Algoritmo Evolutivo.....	48
3. Implementación de la propuesta.....	51
3.1 Definición del Problema de Optimización Multiobjetivo	52
3.2 Generación de Soluciones mediante Simulación Monte Carlo.....	54
3.3 Construcción de la Estructura de Preferencias mediante un SID.....	54
3.3.1 Diseño de las Variables de Entrada.....	55
3.3.2 Diseño de la Variable de Salida	58
3.3.3 Diseño de la Base de Reglas de la Función de Razonamiento Aproximado..	59
3.4 Evaluación de las Soluciones simuladas en el SID	65
3.5 Selección de la Población Semilla.....	65
3.6 Definición de Restricción Incorporando el SID	66
3.7 Definición de Operadores Genéticos y Ejecución del Algoritmo	68
3.8 Obtención del Segmento de la FP y Elección de Solución Óptima	69
4. Resultados de la Implementación de la Propuesta.....	71
4.1 Premisas de Ejecución de la Propuesta.....	71

4.2	Resultados Problema 1	75
4.3	Resultados Problema ZDT1	77
4.4	Resultados Problema ZDT2	79
4.5	Resultados Problema ZDT3	80
4.6	Resultados Problema ZDT4	82
4.7	Resultados Problema ZDT6	84
4.8	Comportamiento de los Perfiles de Preferencia Grande y Muy Grande	85
4.9	Resumen de Resultados	87
4.10	Elección de Soluciones Óptimas	89
5.	Conclusiones, limitaciones y trabajos futuros	91
5.1	Conclusiones	91
5.2	Limitaciones	94
5.3	Trabajos Futuros	95
A.	Anexo: Figuras y Tablas Adicionales	97
B.	Anexo: Código de AGMOSIPD	117
	Bibliografía	119

Lista de figuras

	Pág.
Figura 1-1:	Representación gráfica de la Frontera de Pareto (FP)..... 11
Figura 1-2:	Representación gráfica de un conjunto de cadenas binarias..... 14
Figura 1-3:	Representación gráfica de las estrategias del NSGA-II. 17
Figura 1-4:	Representación gráfica de los puntos de referencia del NSGA-III. 18
Figura 1-5:	Representación gráfica del m. de las restricciones en dos objetivos. 26
Figura 1-6:	Representación gráfica de conjuntos difusos. 30
Figura 1-7:	Representación gráfica de una variable lingüística. 31
Figura 1-8:	Esquema que representa un Sistema de Inferencia Difusa. 32
Figura 3-1:	Diagrama de flujo de AGMOSIPD. 51
Figura 3-2:	Representación gráfica de la FP de los problemas de prueba..... 53
Figura 3-3:	Representación gráfica de las soluciones simuladas..... 55
Figura 3-4:	Representación gráfica para identificar zonas de interés..... 56
Figura 3-5:	Representación gráfica de las VL de entrada del SID..... 58
Figura 3-6:	Representación gráfica de la VL de salida del SID. 59
Figura 3-7:	Relación entre los CD de las VL de entrada y la simulación..... 61
Figura 3-8:	Comparación de \emptyset con etiquetas de \mathcal{Y} 62
Figura 3-9:	Representación gráfica de la superficie de reglas del SID..... 64
Figura 4-1:	Convenciones para interpretar los gráficos de los resultados. 74
Figura A-1:	Resultado de AGMOSIPD en el problema 1 con \mathcal{Y} = "Muy Pequeño" .. 97
Figura A-2:	Resultado de AGMOSIPD en el problema 1 con \mathcal{Y} = "Pequeño"..... 98
Figura A-3:	Resultado de AGMOSIPD en el problema 1 con \mathcal{Y} = "Grande". 98
Figura A-4:	Resultado de AGMOSIPD en el problema 1 con \mathcal{Y} = "Medio". 99
Figura A-5:	Resultado de AGMOSIPD en el problema 1 con \mathcal{Y} = "Muy Grande". ... 99
Figura A-6:	Resultado de AGMOSIPD en ZDT1 con \mathcal{Y} = "Muy Pequeño"..... 100
Figura A-7:	Resultado de AGMOSIPD en ZDT1 con \mathcal{Y} = "Grande". 101
Figura A-8:	Resultado de AGMOSIPD en ZDT1 con \mathcal{Y} = "Pequeño"..... 101
Figura A-9:	Resultado de AGMOSIPD en ZDT1 con \mathcal{Y} = "Muy Grande". 102
Figura A-10:	Resultado de AGMOSIPD en ZDT1 con \mathcal{Y} = "Medio". 102
Figura A-11:	Resultado de AGMOSIPD en ZDT2 para \mathcal{Y} = "Muy Pequeño". 103
Figura A-12:	Resultado de AGMOSIPD en ZDT2 para \mathcal{Y} = "Grande"..... 104
Figura A-13:	Resultado de AGMOSIPD en ZDT2 para \mathcal{Y} = "Pequeño". 104
Figura A-14:	Resultado de AGMOSIPD en ZDT2 para \mathcal{Y} = "Muy Grande"..... 105
Figura A-15:	Resultado de AGMOSIPD en ZDT2 para \mathcal{Y} = "Medio"..... 105
Figura A-16:	Resultado de AGMOSIPD en ZDT3 para \mathcal{Y} = "Muy Pequeño". 106

Figura A-17:	Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{"Pequeño"}$	107
Figura A-18:	Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{"Grande"}$	107
Figura A-19:	Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{"Medio"}$	108
Figura A-20:	Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{"Muy Grande"}$	108
Figura A-21:	Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{"Muy Pequeño"}$	109
Figura A-22:	Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{"Pequeño"}$	110
Figura A-23:	Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{"Grande"}$	110
Figura A-24:	Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{"Medio"}$	111
Figura A-25:	Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{"Muy Grande"}$	111
Figura A-26:	Resultado de AGMOSIPD en ZDT6 para $\gamma = \text{"Muy Pequeño"}$	112
Figura A-27:	Resultado de AGMOSIPD en ZDT6 para $\gamma = \text{"Pequeño"}$	113
Figura A-28:	Resultado de AGMOSIPD en ZDT6 para $\gamma = \text{"Grande"}$	113
Figura A-29:	Resultado de AGMOSIPD en ZDT6 para $\gamma = \text{"Medio"}$	114
Figura A-30:	Resultado de AGMOSIPD en ZDT6 para $\gamma = \text{"Muy Grande"}$	114

Lista de tablas

	Pág.
Tabla 1-1: Elementos que componen una variable lingüística.	31
Tabla 1-2: Matriz con base de reglas del SID.....	33
Tabla 1-3: Definición cualitativa para clasificar objetivos.	36
Tabla 3-1: Funciones de prueba.....	52
Tabla 3-2: Funciones del conjunto de problemas ZDT.....	52
Tabla 3-3: Definición de VL de entrada del SID.	57
Tabla 3-4: Funciones de pertenencia correspondientes a la Figura 3-5.	58
Tabla 3-5: Definición de VL de salida del SID.	59
Tabla 3-6: Funciones de pertenencia correspondientes a la Figura 3-6.	60
Tabla 3-7: Reglas del SID que representa preferencias – Parte I.....	63
Tabla 3-8: Reglas del SID que representa preferencias – Parte II.....	63
Tabla 3-9: Reglas del SID que representa preferencias – Parte III.....	63
Tabla 3-10: Orden de preferencia de valores de γ para poblar la semilla.	66
Tabla 4-1: Ejemplo de semilla para el problema 1 para $\gamma =$ "Muy Pequeño".	72
Tabla 4-2: Nro. de soluciones de AGMOSIPD no dominadas en el problema 1.....	76
Tabla 4-3: % de soluciones de AGMOSIPD no dominadas en el problema 1.	76
Tabla 4-4: Nro. de soluciones de AGMOSIPD no dominadas en ZDT1.....	78
Tabla 4-5: % de soluciones de AGMOSIPD no dominadas en ZDT1.	78
Tabla 4-6: Nro. de soluciones dominadas en AG sin preferencias en ZDT1.	78
Tabla 4-7: % de soluciones dominadas en AG sin preferencias en ZDT1.	79
Tabla 4-8: Nro. de soluciones de AGMOSIPD no dominadas en ZDT2.....	79
Tabla 4-9: % de soluciones de AGMOSIPD no dominadas en ZDT2.	79
Tabla 4-10: Nro. de soluciones dominadas en AG sin preferencias en ZDT2.....	80
Tabla 4-11: % de soluciones dominadas en AG sin preferencias en ZDT2.	80
Tabla 4-12: Nro. de soluciones de AGMOSIPD no dominadas en ZDT3.....	81
Tabla 4-13: % de soluciones de AGMOSIPD no dominadas en ZDT3.	81
Tabla 4-14: Nro. de soluciones dominadas en AG sin preferencias en ZDT3.....	82
Tabla 4-15: % de soluciones dominadas en AG sin preferencias en ZDT3.	82
Tabla 4-16: Nro. de soluciones de AGMOSIPD no dominadas en ZDT4.....	83
Tabla 4-17: % de soluciones de AGMOSIPD no dominadas en ZDT4.	83
Tabla 4-18: Nro. de soluciones dominadas en AG sin preferencias en ZDT4.....	83
Tabla 4-19: % de soluciones dominadas en AG sin preferencias en ZDT4.	83
Tabla 4-20: Nro. de soluciones de AGMOSIPD no dominadas en ZDT6.....	84

Tabla 4-21:	% de soluciones de AGMOSIPD no dominadas en ZDT6.	85
Tabla 4-22:	Nro. de soluciones dominadas en AG sin preferencias en ZDT6.	85
Tabla 4-23:	% de soluciones dominadas en AG sin preferencias en ZDT6.	85
Tabla 4-24:	Nro. de soluciones no dominadas en ZDT4 con perfil GG y MG.	86
Tabla 4-25:	% de soluciones no dominadas en ZDT4 con perfil GG y MG.	86
Tabla 4-26:	% de soluciones de AGMOSIPD no dominadas.	87
Tabla 4-27:	% de soluciones de la optimización sin preferencias dominadas.	88
Tabla A-1:	Soluciones del Problema 1 seleccionadas por AGMOSIPD.	100
Tabla A-2:	Soluciones del problema ZDT1 seleccionadas por AGMOSIPD.	103
Tabla A-3:	Soluciones del problema ZDT2 seleccionadas por AGMOSIPD.	106
Tabla A-4:	Soluciones del problema ZDT3 seleccionadas por AGMOSIPD.	109
Tabla A-5:	Soluciones del problema ZDT4 seleccionadas por AGMOSIPD.	112
Tabla A-6:	Soluciones del problema ZDT6 seleccionadas por AGMOSIPD.	115

Lista de Símbolos y abreviaturas

Símbolos con letras latinas

Símbolo	Término
n	Cantidad de variables de decisión
g	Restricción de desigualdad
h	Restricción de igualdad
w	Cantidad de soluciones que integran la población inicial del AG
x	Función objetivo
z	Cantidad de soluciones obtenidas mediante Simulación Monte Carlo
F	Función de razonamiento aproximado del SID
J	Cantidad de restricciones de desigualdad
K	Cantidad de restricciones de desigualdad
M	Cantidad de funciones objetivos
U	Universo del discurso de un conjunto difuso
V	Cantidad de soluciones obtenidas en AGMOSIPD
\mathcal{X}	VL de entrada en el SID que representa la función objetivo x
\mathcal{Y}	VL de salida en el SID

Símbolos con letras griegas

Símbolo	Término
θ	Variable de decisión
μ	Función de pertenencia de un conjunto difuso
ψ	Etiqueta preferida por el decisor

Subíndices

Subíndice	Término
i	Cantidad de variables de decisión
j	Cantidad de restricciones de desigualdad
k	Cantidad de restricciones de igualdad
m	Cantidad de funciones objetivos
v	Cantidad de soluciones obtenidas en AGMOSIPD

Superíndices

Superíndice Término

<i>L</i>	Valor inferior de la variable de decisión
<i>S</i>	valor superior de la variable de decisión

Abreviaturas

Abreviatura Término

AG	Algoritmos Genéticos
AGMOSIPD	Algoritmo Genético Multiobjetivo con Sistema de Inferencia de Preferencias Difusas
CD	Conjunto Difuso
DCI	Distancia a la Cota Ideal
DCN	Distancia a la Cota Nadir
DPM	Distancia al Punto Medio
FP	Frontera de Pareto
GG	Perfil de Preferencia Grande
LD	Lógica Difusa
MCDM	Multi Criteria Decision Making – Toma de Decisiones Multi Criterio
MG	Perfil de Preferencia Muy Grande
MM	Perfil de Preferencia Medio
MOEA	Multiobjective Optimization Evolutionary Algorithm - Algoritmo Evolutivo de Optimización Multiobjetivo
MP	Perfil de Preferencia Muy Pequeño
POM	Problema de Optimización Multiobjetivo
PP	Perfil de Preferencia Pequeño
PBR1	Problema 1
SID	Sistema de Inferencia Difusa
VL	Variable Lingüística
ZDT1	Funciones de optimización multiobjetivo de Zitzler, Deb y Thiele # 1
ZDT2	Funciones de optimización multiobjetivo de Zitzler, Deb y Thiele # 2
ZDT3	Funciones de optimización multiobjetivo de Zitzler, Deb y Thiele # 3
ZDT4	Funciones de optimización multiobjetivo de Zitzler, Deb y Thiele # 4
ZDT6	Funciones de optimización multiobjetivo de Zitzler, Deb y Thiele # 6

Introducción

"En un día muy helado, un grupo de erizos que se encuentran cerca sienten simultáneamente la necesidad de juntarse para darse calor y no morir congelados.

Cuando se aproximan mucho, sienten el dolor que les causan las púas de los otros erizos, lo que les impulsa a alejarse de nuevo.

Sin embargo, como el hecho de alejarse va acompañado de un frío insoportable, se ven en el dilema de elegir: herirse con la cercanía de los otros o morir. Por ello, van cambiando la distancia que les separa hasta que encuentran una óptima, en la que no se hacen demasiado daño ni mueren de frío".

Arthur Schopenhauer

Arthur Schopenhauer utilizaba la metáfora anterior, en su obra Parerga y Paralipómena (1851), para dar una idea sobre la manera como se producen las interacciones entre los seres humanos. Ahora, es mi intención alejarme de las razones filosóficas del autor de este texto y aprovecharlo para introducir el problema inmerso en la resolución de problemas en la vida real. Hay situaciones donde las personas o las organizaciones aspiran alcanzar un propósito, pero el logro de este se encuentra afectado por otras razones que le generan conflicto; aquí surge un dilema. En el caso de los erizos, el propósito de sobrevivir se ve afectado por el frío y el dolor que resulta al lastimarse con las púas de los demás cuando se aproximan buscando calor, el dilema sería sobrevivir a costa de sentir dolor.

Es posible suponer que en esta historia los erizos se enfrentan a dos propósitos:

- Sobrevivir disminuyendo la percepción de frío (obteniendo calor al reducir la distancia entre los erizos).
- Evitar el dolor al contacto con las púas (aumentando la distancia entre los erizos).

Como la distancia de los erizos afecta ambos propósitos, es posible concluir que la mínima percepción de frío posible equivaldrá a la mayor percepción posible de dolor al contacto

con las púas y, a su vez, se puede concluir lo contrario. El mejor escenario factible de cada propósito individual no es posible obtenerlo cuando se deben cumplir los dos propósitos en simultánea. La consecución de una solución ideal en el primer objetivo desmejorará el resultado que se obtiene en el segundo objetivo y puede notarse que ambos objetivos presentan conflicto.

Considerando lo anterior, podría preguntarse ¿cómo logran los erizos hallar la menor distancia que produzca menor riesgo de muerte por el frío y menor dolor, por el contacto con las púas, a la vez? Se podría sospechar que, para responder a esta pregunta, sería válido suponer que, a medida que los erizos van explorando diversas configuraciones de agrupamiento, van evaluando la satisfacción que ella les reporta en términos de la mínima percepción de frío y, a la vez, la mínima percepción de dolor. Además, se esperaría que la exploración cese hasta que los erizos encuentren la configuración más preferida donde los animales encuentran satisfacción aceptable para el nivel de frío y de dolor.

La metáfora anterior ejemplifica el proceso de búsqueda de una solución satisfactoria, de forma muy similar a cuando uno o varios actores desean encontrar la mejor solución preferida (solución óptima) a un problema que presenta varios objetivos (múltiples). En el caso de los erizos sería posible imaginar que cada uno de ellos podría tener un umbral de dolor distinto y, por tanto, según esa disposición, podría valorar de forma diferente la importancia que le reporta esa percepción de dolor con respecto a la percepción de frío; esto implica que cada erizo tendría una preferencia individual distinta y, por ende, una solución óptima igualmente diferente. Es posible incluso seguir imaginando que un erizo pueda tener varias soluciones óptimas para diferentes configuraciones de preferencias. Ante estas posibilidades, el dilema es ¿cuál solución óptima elegir?

Entender este proceso de búsqueda es objeto de interés de la **optimización multiobjetivo**. En este campo se han estudiado las técnicas que permiten encontrar las soluciones que mejor satisfacen una situación donde se presentan dos o más objetivos, teniendo en cuenta que el logro de cada objetivo se puede afectar por el comportamiento de los demás. El conjunto de soluciones que mejor satisface, de forma simultánea, el valor de las funciones objetivo se denomina frontera de Pareto, cuyos elementos se catalogan como eficientes, es decir son óptimos u Pareto óptimos, ya que ninguno de ellos es mejor

que otro que haga parte del mismo conjunto, ni existirán otros que ofrezcan mejores resultados que los que estos ya ofrecen.

Cuando en esta búsqueda se encuentran soluciones que hacen parte de la frontera de Pareto se debe considerar que la satisfacción que reportan una o varias posibles soluciones de un problema está determinada por los intereses que tienen los actores (decisores) involucrados en el mismo; es decir, por las preferencias, gustos, deseos, expectativas que estos poseen. Es por esto que no basta con que el problema ofrezca un conjunto de soluciones óptimas, sino que es necesario incorporar las preferencias que los decisores tienen para que el problema ofrezca la mejor solución preferida. Para efectos de acotar el alcance del estudio que se presenta en este documento se abordarán problemas de optimización multiobjetivo donde se involucran las preferencias de un solo decisor.

Los problemas de optimización multiobjetivo se han modelado mediante expresiones matemáticas y lógicas donde los objetivos se representan mediante funciones que, a su vez, contienen variables que las determinan (variables de decisión). En este campo se han desarrollado técnicas clásicas cuyos algoritmos ofrecen respuesta de forma eficiente a problemas sencillos, pero podrían ser insuficientes y, por ende, ineficientes para problemas complejos de resolver. Por ello, para combatir dificultades como las planteadas, se han diseñado técnicas metaheurísticas, las cuales son métodos computacionales que buscan descubrir las soluciones a través de reglas, estrategias, tanteos, silogismos y conclusiones a través de un procesamiento continuo mediante iteraciones. El modo de operación de estas técnicas intenta replicar o imitar fenómenos naturales o industriales que generalmente permiten obtener un resultado adecuado. Los **algoritmos genéticos**, por ejemplo, hacen parte de estas metaheurísticas y permiten encontrar soluciones a un problema de optimización emulando la manera como, en la biología, los genes transmiten material genético de generación en generación, permitiendo la evolución y el mejoramiento de los seres vivos. El desarrollo y la aplicación de diversas variantes de algoritmos genéticos para resolver problemas de optimización multiobjetivo ha mostrado efectividad a la hora de encontrar la frontera de Pareto del problema (Deb & Jain, 2014; Coello, 2019).

El fin último de un problema de optimización multiobjetivo será brindarle conclusiones claras al responsable del problema para que pueda tomar decisiones sobre el mismo, pero la obtención de la frontera de Pareto no necesariamente brinda al decisor información

suficiente para llevar a cabo este propósito. Por lo tanto, posterior al hallazgo de la frontera de Pareto, se hace necesario que el decisor elija la solución con la que finalmente apostará para abordar el problema que tiene a cargo. Esta elección no es sencilla puesto que todos los elementos que componen la frontera son eficientes (óptimos) y para tomar partido por uno de ellos deberá incorporar un criterio de preferencias con base en el comportamiento que tienen los objetivos en cada una de las soluciones que está analizando. La dificultad en la definición de las preferencias podría deberse a:

- La cantidad de soluciones que compone la frontera de Pareto puede ser infinita y compararlas podría ser imposible.
- El decisor no necesariamente tiene claras las expectativas frente a lo que espera de cada objetivo.
- El decisor podría no ser capaz de expresar de forma cuantitativa la magnitud con la que prefiere un objetivo con respecto a los demás.
- Podría existir incoherencia en el decisor al momento de expresar sus preferencias.

A medida que avanza el análisis multiobjetivo, no solamente se han definido estrategias para modelar las preferencias, sino que además se han establecido métodos para incorporarlas en el proceso de búsqueda de acuerdo a la información que el algoritmo va revelando sobre el comportamiento de los objetivos. Los métodos de incorporación de preferencias se distinguen de acuerdo al momento en el que participan en la ejecución de la técnica metaheurística:

- **Métodos a priori:** pretenden modelar las preferencias de los objetivos del problema antes de ejecutar un mecanismo de optimización multiobjetivo. Este modelamiento se incorpora en el problema y a partir de allí se ejecuta el algoritmo genético para obtener la mejor solución.
- **Métodos a posteriori:** requieren el hallazgo de varias soluciones de la frontera de Pareto para que, posteriormente, el decisor exprese las preferencias de los objetivos del problema, de tal manera que pueda encontrar el segmento de dicha frontera que más le satisface.
- **Métodos interactivos:** es un camino intermedio entre los métodos a priori y a posteriori, donde el decisor va interactuando con el algoritmo, de forma que el segundo le va informando al primero las soluciones que va encontrando, para que

el decisor a su vez indique sus preferencias con dicha información. Al final se obtiene la solución que se ajusta a las preferencias y que, a su vez, es eficiente.

Estas 3 formas de incorporación de preferencias presentan ventajas y desventajas y, a la fecha, se siguen diseñando e investigando estrategias que permitan despejar las diferentes limitaciones que estos métodos aun presentan, de las cuales se mencionan algunas a continuación:

- En el caso de los métodos a priori, hay una alta posibilidad de que el decisor deba especular el comportamiento de los objetivos y correr el riesgo de subestimar sus preferencias pues se asume que tiene poca o nula idea de las posibles soluciones que arroja el problema, dado que el algoritmo de optimización aún no se ha ejecutado y no ha brindado información de posibles soluciones.
- En los métodos a posteriori, sucede que el decisor podría experimentar saturación al analizar la infinidad de soluciones presentes en la frontera de Pareto.
- En los métodos interactivos, sucede que la alta dependencia que tiene el algoritmo con la interacción el decisor podría sobrecargar a este último, dada la cantidad de veces que debería expresar sus preferencias durante varias ejecuciones para llegar a la solución adecuada.

La asignación de preferencias en un problema multiobjetivo se ha realizado mediante diversas estrategias cuantitativas; por ejemplo, se han establecido pesos numéricos de importancia relativa, valoración de tasas de intercambio, funciones de utilidad, puntos de referencia (aproximación a puntos ideales), puntos de reserva (distanciamiento de puntos indeseados), umbrales de indeseabilidad, entre otros.

También, se han propuesto técnicas cualitativas que intentan traducir las preferencias en expresiones lingüísticas donde el decisor podría utilizar palabras como: mucho, poco, grande, pequeño, alto, bajo, entre otras. Dichas expresiones se representan matemáticamente estableciendo el rango de valores del objetivo que mejor explica dicho término lingüístico. En la sección 1 de este documento se explicará esto con mayor detalle.

En la literatura se pueden identificar estudios donde la expresión de preferencias a través de términos cualitativos (lingüísticos) se hace emitiendo comparaciones entre pares de

objetivos y traduciéndolos en pesos de importancia o en tasas de intercambio (Xin et al., 2018). Por ejemplo, si en un problema se tienen dos objetivos como: maximizar las utilidades monetarias de un proyecto y minimizar el tiempo de duración del mismo, el decisor podría indicar que tiene una preferencia “baja” por las utilidades monetarias y una preferencia “alta” por el tiempo de duración del proyecto. Esta forma de representación lingüística se estudia a profundidad en el campo de la Lógica Difusa, desarrollada por Lofti Zadeh en 1965, con aplicación directa en Sistemas de Inferencia Difusa (Duarte, 1999).

Tanto en las propuestas cuantitativas como cualitativas se ha encontrado que se asume que la forma como el decisor expresa sus preferencias es constante en todo el espacio de soluciones y, además, no consideran que estas preferencias podrían verse afectadas por la interacción entre los objetivos. En el caso de preferencias cuantitativas, no en todas las soluciones el decisor podría preferir un objetivo con un peso del 70% y el otro objetivo con un peso del 30%. En las preferencias cualitativas, no siempre podríamos suponer que el decisor tiene una preferencia “baja” por un objetivo A y una preferencia “alta” por un objetivo B, ya que no necesariamente estas preferencias permanecerán inamovibles en todo el espacio de soluciones: podría suceder que en otro lugar del espacio de búsqueda, otro conjunto de soluciones produzca en el decisor una configuración de preferencias distinta. Por ejemplo, si un conjunto de soluciones ofrece abundantes ganancias monetarias, el decisor estaría dispuesto a sacrificar más de estas a cambio de una menor duración del proyecto, con respecto a un conjunto de soluciones que ofrece bajas ganancias.

A partir de lo expuesto y teniendo en cuenta que aún existen limitaciones que dificultan la manipulación de preferencias, puesto que estas podrían ser relativas y cambiantes, y es difícil expresarlas de forma cuantitativa, indicarlas de forma cualitativa podría brindar una mejor aproximación al imaginario del decisor. Es muy probable que, antes del proceso de búsqueda, el decisor no tenga un conocimiento previo del problema y, además, incorporarlas en un método clásico de optimización multiobjetivo podría resultar complejo puesto que los modelos generalmente se expresan mediante formulaciones altamente no lineales o incluso se expresan mediante expresiones lineales tan complicadas que pueden ser difíciles de comprender para el decisor.

Se presentará una propuesta de aporte teórico para incorporar preferencias mediante un Sistema de Inferencia Difusa en un algoritmo genético multiobjetivo. La utilización de algoritmos genéticos presenta la ventaja de que, dada su implementación a través de la evaluación masiva de soluciones (poblaciones), el proceso de búsqueda brinda mucha más información sobre el problema, a lo ancho del espacio de búsqueda, con respecto a un método clásico que informa el comportamiento de soluciones evaluándolas de una en una.

Para abordar esta propuesta, se definen los siguientes objetivos:

- **Objetivo general:** Proponer una metodología que incorpore un Sistema de Inferencia Difusa para la asignación de las preferencias de un decisor en la resolución de un problema de optimización multi-objetivo con Algoritmos Genéticos.
- **Objetivos específicos:**
 1. Elegir el método de asignación de preferencias que permita modelar las expectativas del decisor.
 2. Determinar la manera como se implementará la lógica difusa al modelo de asignación de preferencias.
 3. Establecer los algoritmos evolutivos a los cuales se les incorporará el Sistema de Inferencia Difusa, y determinar el momento en el cual debe ser involucrado.
 4. Implementar la propuesta en un problema prueba (test) y validar su funcionamiento.
 5. Verificar que las soluciones presentadas por el modelo sean eficientes.

El presente trabajo se desarrollará mediante los siguientes apartados: en la sección 1 se detallará una revisión teórica en la cual se explicará la aproximación de los modelos de optimización multiobjetivo y la incorporación de preferencias en ellos. En la sección 2 se presentará una propuesta metodológica para dar respuesta a los objetivos de investigación, posteriormente en la sección 3 se presentará la implementación de la propuesta y en la sección 4 se abordarán los resultados. Este estudio finaliza con las conclusiones de la aplicación de la propuesta presentada, la presentación de las limitaciones de la investigación y una propuesta de futuros trabajos para la ampliación de este estudio.

1. Marco Teórico

En esta sección se introduce una aproximación teórica de los elementos que fundamentan el estudio que se desarrolla en este trabajo: en primer lugar, se explicarán algunos abordajes relacionados con la optimización multiobjetivo y, posteriormente, se hará referencia a los trabajos relacionados con la incorporación de preferencias en este tipo de problemas. Así mismo, se presentarán las ventajas y desventajas identificados en estos trabajos, y también se expondrá el funcionamiento de la lógica difusa y los sistemas difusos y la aplicación de estos en la asignación de preferencias.

1.1 Optimización Multiobjetivo

Según la Real Academia Española (2023) una de las definiciones del verbo decidir es: “Determinar el resultado de algo.” Este concepto permite deducir que el proceso de toma de decisiones implica poner en juego la obtención de un resultado y la manera como se logra podría ser variada.

Una camino que ayuda a dar respuesta a la manera como se pueden tomar decisiones en un problema que involucra diversos objetivos de forma simultánea es la optimización multiobjetivo, en la cual se representan los objetivos de forma matemática o mediante expresiones cuantitativas para encontrar las mejores soluciones posibles o una buena aproximación de estas, dado un conjunto de limitaciones o restricciones (Cuartas Torres, 2009).

Generalmente un problema de optimización multiobjetivo (en adelante POM) se define como (Taylor, 2022):

$$\begin{array}{lll}
\text{Minimizar} & x_m(\theta), & m = 1, 2, \dots, M; \\
\text{Sujeto a} & g_j(\theta) \geq 0, & j = 1, 2, \dots, J; \\
& h_k(\theta) = 0, & k = 1, 2, \dots, K; \\
& \theta_i^{(L)} \leq \theta_i \leq \theta_i^{(S)}, & i = 1, 2, \dots, n
\end{array}$$

Un vector solución θ comprende n variables de decisión tales que $\theta = [\theta_1, \theta_2, \dots, \theta_n]^T$, y está definido por la restricción $\theta_i^{(L)} \leq \theta_i \leq \theta_i^{(S)}$ que representa los límites que las variables de decisión pueden tomar. El conjunto de funciones objetivo $x = \{x_1, \dots, x_M\}$ donde M es el número de funciones objetivo. Las funciones $g_j(\theta)$ y $h_k(\theta)$ son respectivamente, restricciones de desigualdad e igualdad que deben satisfacerse (junto con los límites de las variables de decisión), para que una solución θ se considere factible.

Un problema de minimización puede transformarse en un problema de maximización y viceversa, para lograr dicha transformación basta con multiplicar las funciones objetivo por menos uno (-1).

En la literatura se establece la siguiente distinción entre problemas “multiobjetivos” y de “muchos objetivos” así (Blank & Deb, 2020):

- Un problema multiobjetivo (multi-objective problem) presenta entre 2 y 3 objetivos.
- Un problema de muchos objetivos (many-objective problem) presenta más de 3 objetivos.

En este estudio se abordarán los problemas multiobjetivo, los cuales ya se han denotado como POM.

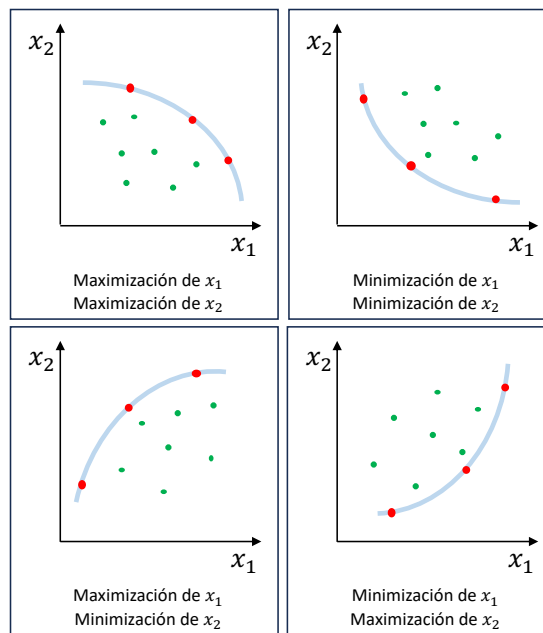
La finalidad de un POM consiste en encontrar un conjunto de vectores θ^* que pertenezcan al espacio de soluciones acotado por las restricciones y donde cada una de las funciones $x_m(\theta)$ sea minimizada. Adicionalmente, se espera que una solución θ^* sea eficiente, es decir, que sea la mejor en al menos uno de los objetivos con respecto a las demás soluciones. Todos los vectores θ^* aquí definidos hacen parte de un conjunto de soluciones eficientes que representan una zona en el espacio de los objetivos llamado Frontera de

Pareto (en adelante FP): estas soluciones tienen la particularidad de que son no dominadas pues ninguna de ellas es mejor que las demás en todos los objetivos (Shen et al., 2010).

Es importante tener en cuenta que la eficiencia de una solución se mide con respecto a la capacidad que tiene una alternativa de dominar otras soluciones. La dominancia se puede entender como la posibilidad de que los beneficios ofrecidos por una solución, sea mayor o igual en cada uno de sus objetivos con respecto a otras soluciones ya analizadas. En un POM se entiende que la FP es la curva (o la región) donde se encuentran todas aquellas soluciones eficientes y donde es indiferente para el decisor escoger una de ellas desde un punto de vista matemático.

La forma gráfica de la FP en un problema que involucra dos objetivos se aprecia en la Figura 1-1 mediante una curva azul, cuyo sentido cambia de acuerdo con el criterio de optimización (maximización o minimización) al cual esté orientado el problema. Los puntos rojos representan ejemplos de soluciones no dominadas que constituyen la frontera y por ende son eficientes, los puntos verdes representan soluciones que son dominadas, es decir, no ofrecen mejores resultados a los que se obtienen en la FP.

Figura 1-1: Representación gráfica de la Frontera de Pareto (FP).



Fuente: elaboración propia

1.2 Técnicas Metaheurísticas

Aplicar las técnicas clásicas para resolver un POM y, por ende, encontrar su FP podría resultar insuficiente e ineficiente dado que los problemas podrían ser complejos por aspectos como los siguientes:

- Gran cantidad de variables de decisión que generan un espacio de búsqueda igualmente grande.
- Planteamiento del problema a través de expresiones no lineales difíciles de manipular matemáticamente.
- Participación de un número extenso de funciones objetivo que dificulta comprender la incidencia de un objetivo sobre otro.
- Incorporación de restricciones excesivas que limitan la consecución de soluciones factibles.
- Complejidad inherente del problema de optimización difícil de comprender por parte del decisor.

Para lidiar con estas dificultades se han desarrollado técnicas en el área de la Inteligencia Artificial para descubrir buenas soluciones que en muchos casos se aproximan de forma adecuada a las soluciones óptimas del POM. Estas técnicas se conocen como Metaheurísticas y se diseñan mediante algoritmos que reproducen de forma similar las reglas de comportamiento de fenómenos que encuentran una mejora en su procesamiento. Estas reglas se plasman de forma secuencial en un algoritmo para que, a través de iteraciones, se obtenga una aproximación adecuada a la solución esperada. Algunos ejemplos de estas metodologías son:

- La técnica “colonia de hormigas” imita las acciones que ejecutan las hormigas para encontrar alimento de forma óptima.
- La técnica “enjambre de partículas” reproduce el comportamiento de un colectivo de individuos (por ejemplo, aves) para encontrar una trayectoria de desplazamiento óptima o alcanzar un objetivo deseado (por ejemplo, encontrar alimento).
- El método de “recocido simulado” se inspira en el proceso de calentamiento industrial de ciertos materiales para variar sus propiedades físicas hasta obtener un punto deseado (óptimo).

- En la biología la transmisión de caracteres, cualidades o características en los seres vivos se produce a través de la genética, generando la evolución y la preservación de las especies. Este comportamiento se reproduce en los algoritmos evolutivos, en ellos se va transmitiendo y modificando la información genética del problema con el fin de evolucionar las soluciones y aproximarse a los óptimos deseados. En los algoritmos evolutivos se encuentran los siguientes ejemplos: Programación Evolutiva, Estrategias Evolutivas, Programación Genética y Algoritmos Genéticos (Coello et al., 2007).

La técnica de los algoritmos genéticos es reconocida por su éxito para el hallazgo de soluciones que satisfacen los POM. Según Cortez et al. (2019) “la capacidad evolutiva de los algoritmos genéticos, permite optimizar funciones en espacios adaptativos complejos.” Adicionalmente, gozan de simplicidad conceptual, ofrecen diversas posibilidades de aplicabilidad, se hibridan fácilmente con otros algoritmos y generalmente son robustos a cambios (Santana & Coello, 2006). En la actualidad, existen diversos algoritmos diseñados bajo este paradigma y se ha comprobado su efectividad, por ello este estudio se centrará en la profundización de esta metaheurística.

1.3 Algoritmos Genéticos

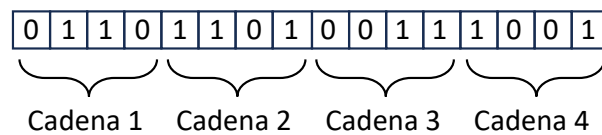
Entre las técnicas metaheurísticas más utilizadas se encuentran los Algoritmos Genéticos (en adelante AG) los cuales fueron concebidos por John H. Holland entre 1960 y 1970 motivado por su interés en resolver problemas de aprendizaje de máquina (Cuartas Torres, 2009). De forma resumida, los AG están inspirados en los mecanismos de selección natural de las especies, pues van analizando las alternativas existentes en el espacio de búsqueda y van seleccionando aquellas que son mejores.

Los AG hacen parte de los métodos de búsqueda estocástica, los cuales incorporan nociones como selección, cruce, reproducción y mutación. Un AG produce soluciones, conocidas como individuos, y las hace evolucionar durante un número establecido de generaciones (iteraciones). Los individuos pueden representarse como una cadena binaria y se evalúan mediante una función de aptitud, que determina lo "bien" que se comportan dichas soluciones en el espacio objetivo. Posteriormente, los individuos más aptos de cada

generación se combinan por pares conformando padres, para producir hijos (descendientes), que se evalúan en la siguiente generación (Taylor, 2022).

En un AG las variables de decisión se modelan, usualmente, mediante expresiones binarias denominadas cadenas. Un conjunto de cadenas conforma un cromosoma y este, a su vez, representa una potencial solución del POM, también es llamado individuo. Cada posición de la cadena se define como un gen y el valor contenido dentro de dicho gen se denomina alelo. En la Figura 1-2 se ilustra esta representación binaria (Coello, 2019).

Figura 1-2: Representación gráfica de un conjunto de cadenas binarias.



Fuente: (Coello, 2019)

La aplicación de un AG sigue los siguientes pasos:

1. Generar aleatoriamente una muestra de un conjunto de individuos, conocida también como población de soluciones potenciales del POM.
2. Calcular el valor que cada individuo obtiene en cada función objetivo del POM, conocido también como aptitud. Es posible calcular la aptitud con alguna otra función de evaluación que dé cuenta del desempeño que se espera que tenga la solución dentro del POM.
3. Aplicar un procedimiento de selección, el cual consiste en escoger los individuos de la población que presentan mejor valor de aptitud. Existen diferentes mecanismos para seleccionar estos individuos denominados padres.
4. Crear nuevos individuos a través de operaciones de cruzamiento, combinando la información genética entre pares de padres, y operaciones de mutación, alterando de forma probabilística los valores contenidos en los genes de los individuos. Estos individuos se denominan hijos y conformarán la descendencia que podrán ser parte de la nueva generación (iteración).

Las operaciones de selección, cruzamiento y mutación definidos en los pasos 3 y 4 se denominan operadores genéticos y trabajan con una probabilidad de selección, de cruzamiento y de mutación respectivamente. El cruzamiento ayuda a preservar la información genética presente en la población y la mutación ayuda a evitar la convergencia prematura del algoritmo, promoviendo la exploración de soluciones en zonas del espacio de búsqueda.

El AG requiere, además, que se defina el tamaño de la población que se evaluará en todo el proceso y la cantidad de generaciones que correrá el algoritmo. Esta última será el criterio de parada del AG.

Diferentes investigadores han desarrollado diversas herramientas para incorporar los AG a los POM, cuya variedad se debe al interés de encontrar una FP mejor delineada y con soluciones más satisfactorias, disminuir el costo computacional que traen consigo los algoritmos, reducir la complejidad de los POM ante el aumento de variables de decisión y funciones objetivo, aumentar la robustez de la calidad del proceso de búsqueda ante variaciones de los parámetros, entre otros. Entre los desarrollos más reconocidos en la literatura es posible mencionar los siguientes (Coello et al., 2007):

- Vector Evaluated Genetic Algorithm (VEGA)
- Lexicographic Ordering Genetic Algorithm
- Weight-Based Genetic Algorithm (WBGA)
- Multiobjective Genetic Algorithm (MOGA)
- Niche-Pareto Genetic Algorithm (NPGA, NPGA 2)
- Nondominated Sorting Genetic Algorithm (NSGA, NSGA-II, NSGA-III)
- Distance-based Pareto Genetic Algorithm
- Thermodynamical Genetic Algorithm (TDGA)
- Strength Pareto Evolutionary Algorithm (SPEA, SPEA2)
- Multiobjective Messy Genetic Algorithm (MOMGA-I, MOMGA-II, MOMGA-III)
- Micro GA-MOEA (μ GA, μ GA2)
- Multiobjective Struggle Genetic Algorithm (MOSGA)
- Multiobjective Evolutionary Algorithm Based on Decomposition (MOEA/D)

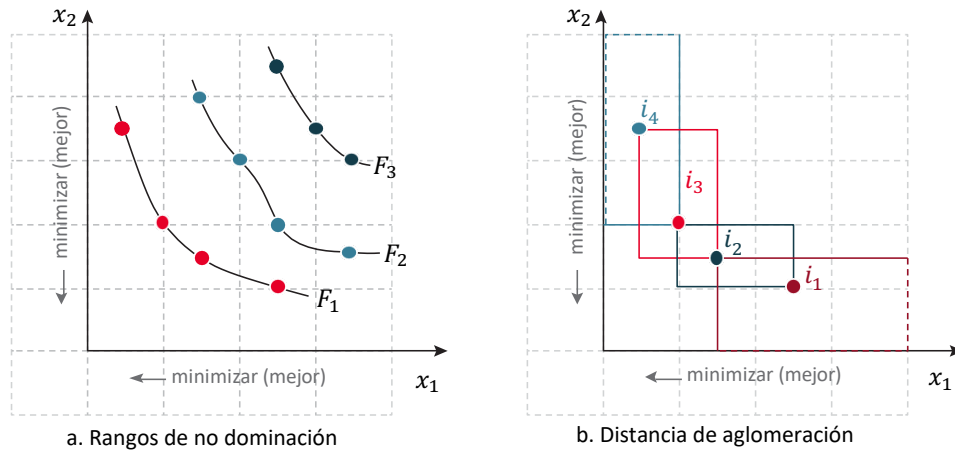
1.4 Nondominated Sorting Genetic Algorithm II y III

El NSGA-II es un algoritmo basado en la dominancia de Pareto, que considera las siguientes estrategias de funcionamiento (Deb et al., 2002):

- Durante las generaciones, el algoritmo encuentra posibles FP y va identificando en cuál frente se ubica cada individuo para luego clasificar, por este criterio, las mejores soluciones que va encontrando.
- Promueve la preservación de diversidad de soluciones, basándose en un enfoque de nicho, apiñamiento o agrupamiento, donde estima un indicador de densidad a partir del volumen del hiper-rectángulo definido por los vecinos más cercanos.
- Selecciona las soluciones que pasarán a la siguiente generación de acuerdo con la distancia de aglomeración entre los individuos en el espacio de las funciones objetivo, este criterio hace parte de la función de aptitud del algoritmo.

El NSGA-II es el algoritmo evolutivo de optimización multiobjetivo (en adelante MOEA por su expresión en inglés) que ha sido usado más frecuentemente en la literatura desde que fue propuesto en el año 2000; la mayoría de los nuevos MOEA presentados en la literatura casi siempre fueron comparados con el rendimiento del NSGA-II debido a su buen desempeño. Sin embargo, se ha encontrado que MOEA basados en dominancia de Pareto, como NSGA-II, no funcionan bien en problemas de prueba de muchos objetivos (Ishibuchi et al., 2016). Según (Taylor, 2022), se ha demostrado que el rendimiento del algoritmo se degrada cuando los objetivos aumentan a más de tres, donde una gran proporción de soluciones candidatas se vuelven no dominantes entre sí, y la medida de distancia de aglomeración se vuelve costosa computacionalmente.

La Figura 1-3 muestra, en la subfigura “a”, un ejemplo de un problema de minimización de dos objetivos donde se identifican cada una de las FP o rangos de no dominación F_1 , F_2 y F_3 que el NSGA-II identificaría. En la subfigura “b” muestra los individuos del frente F_1 y el tamaño de las regiones vacías en forma de rectángulo que rodea a cada uno de ellos, el tamaño de estas zonas se utiliza para encontrar la medida de distancia de aglomeración.

Figura 1-3: Representación gráfica de las estrategias del NSGA-II.

Fuente: (Taylor, 2022)

Posteriormente (Deb & Jain, 2014) proponen el algoritmo NSGA-III, el cual también se basa en el siguiente funcionamiento:

- Al igual que el NSGA-II, incorpora el análisis de dominancia de Pareto y, a su vez, asigna las soluciones candidatas a los distintos frentes que va encontrando.
- Sustituye la métrica de distancia de aglomeración usado en el NSGA-II por un operador de preservación de nicho que incorpora un conjunto de puntos de referencia bien distribuidos. Estos puntos de referencia pueden ser establecidos por el algoritmo o por el decisor.

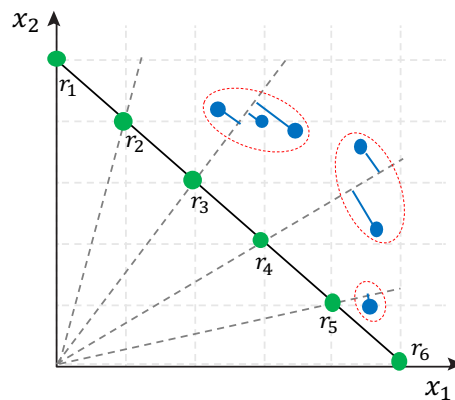
Para ejemplificar la manera como operan los puntos de referencia en NSGA-III se puede apreciar la Figura 1-4: en color verde se muestran 6 puntos de referencia denotados como r_1, r_2, r_3, r_4, r_5 y r_6 , los cuales están equidistantemente distribuidos; las soluciones candidatas se indican en color azul y ellas se asocian a la línea de proyección que conecta el punto ideal y cada punto de referencia, la asociación se realiza hacia la línea donde la solución tiene menor distancia.

El NSGA-III ha sido utilizado satisfactoriamente en problemas reales y de prueba con hasta 15 objetivos.

Para el desarrollo de este trabajo se elegirán los algoritmos NSGA-II y NSGA-III, dadas las pruebas satisfactorias que se han encontrado en la utilización de ambas propuestas. Es

importante recordar que estos algoritmos están basados en la dominancia de Pareto y su finalidad será encontrar la mayor cantidad de soluciones posibles que conforman la FP. Sin embargo, el número de soluciones posibles entre las que puede tener que elegir un decisor pueden ser muy grandes, y esta elección es una decisión difícil y exigente desde el punto de vista cognitivo (Taylor, 2022). Aunque estas metodologías hacen un trabajo adecuado por encontrar la FP, se hace necesario incorporar un proceso adicional al algoritmo que le permita al decisor escoger la alternativa que mejor satisfaga sus requerimientos y facilite el proceso de toma de decisiones.

Figura 1-4: Representación gráfica de los puntos de referencia del NSGA-III.



Fuente: (K. Li et al., 2019)

El fin último de los POM es ayudar al decisor a encontrar su solución más preferida, la cual, es una solución que pertenece a la FP y que el decisor prefiere a todas las demás (Xin et al., 2018). Introducir las preferencias del decisor en el proceso de optimización multiobjetivo es de vital importancia, ya que no sólo puede ayudar al decisor a obtener sus soluciones preferidas, sino también podría simplificar el proceso de búsqueda (Lai et al., 2021).

A continuación, se expondrán algunas aproximaciones teóricas que tratan la incorporación de preferencias en MOEA.

1.5 Incorporación de Preferencias en MOEA

Las preferencias brindan un aporte fundamental para tomar una decisión en un problema: cuando se expresa una preferencia, se espera que ella ayude a brindar criterios de satisfacción para realizar una comparación entre un conjunto de alternativas. Estos criterios pueden ser imprecisos, subjetivos, cuantitativos, cualitativos, racionales, cambiantes, y podrían depender de elementos que condicionan el modo de pensar del decisor. Estos elementos podrían ser el entorno en el que está inmerso el problema, la aversión al riesgo del decisor, la relación entre los objetivos, las emociones que experimenta el decisor, entre otros. En definitiva, las preferencias, con sus criterios, permiten hacer evaluaciones de un conjunto de opciones a través de comparaciones para finalmente elegir aquellas que sean más atractivas.

En la computación, particularmente en el análisis multiobjetivo y en la Toma de Decisiones Multicriterio (en adelante MCDM por su sigla en inglés), se ha estudiado la manera de obtener las preferencias que, según Kaci (2011), implica:

- Abordar un componente de aprendizaje por el que se infieren las preferencias de un decisor a partir de comportamientos anteriores o de las preferencias de decisores similares.
- Establecer una consulta o proceso interactivo para obtener las preferencias de un decisor.
- Involucrar algún tipo de representación o lenguaje para capturar y manipular las preferencias.
- Diseñar un modelo para incorporar relaciones y describir sus propiedades.

La manera como se modelen las preferencias depende de cómo se expresen. Esto puede darse de forma cuantitativa mediante expresiones como: "Valoro la alternativa A con un 7 sobre 10" o "Prefiero el durazno con un peso de 0,2", las cuales se representan mediante números reales; y también de forma cualitativas como "prefiero el color azul que el rojo" y "el rock es mi tipo de música favorito". El desafío de expresar y modelar las preferencias se presenta en el posible incremento del costo computacional al momento de incorporarlas en un POM, y también puede suceder que el modelo de preferencias que se proponga no refleje la intensidad interna (o configuración psicológica) del decisor o que carezca de

coherencia, por lo que tradicionalmente se asumen preferencias idealizadas o normativas (Coello et al., 2007).

1.5.1 Obtención e Incorporación de Preferencias Precisas

La obtención de preferencias puede realizarse a través de preguntas específicas al decisor de tal manera que directamente pueda expresarlas de forma cuantitativa a través de una valoración numérica precisa, o también puede expresarlas indirectamente, de forma imprecisa o cualitativa, mediante la observación del comportamiento, o también sometiendo al decisor a ejercicios de razonamiento sencillos que reduzcan el esfuerzo mental que debe hacer para indicar sus gustos y prioridades. En este apartado se mostrarán técnicas que requieren una expresión cuantitativa de preferencias.

Otro aspecto importante que aborda la MCDM, adicional al modelamiento de las preferencias, corresponde a la definición del momento en el cual se deben incorporar las preferencias en los POM (Li et al., 2020). En la literatura se han documentado tres métodos para hacerlo:

- **Métodos A Priori de Incorporación de Preferencias:**

Estos métodos articulan previamente las preferencias al problema antes de ejecutar el algoritmo de optimización. Generalmente los métodos clásicos para resolver un POM suelen agregar objetivos en conflicto en una sola función objetivo convirtiendo así un POM en un problema de optimización de un único objetivo: estos objetivos se agregan articulando las preferencias en la función objetivo resultante.

La incorporación de preferencias a priori se utiliza para evaluar la idoneidad de una solución y para dirigir a la población hacia una zona de interés del espacio de soluciones, a lo largo de una dirección "preferida" predefinida (Li et al., 2020). A continuación, se presentan algunos de estos métodos:

Suma ponderada: Bajo esta técnica se definen pesos numéricos en un intervalo de cero a uno, los cuales representan las preferencias del decisor frente a cada

objetivo individual. El nuevo objetivo se obtiene sumando todos los objetivos normalizados y cada uno multiplicado por su respectivo peso (Smith et al., 2000).

Programación por compromiso: Este método toma el objetivo resultante de la técnica de suma ponderada y elabora una función que representa la distancia que este objetivo tiene hacia un punto ideal: la menor distancia de esta función será la que dirija el proceso de optimización (Smith et al., 2000).

Punto de referencia (Programación por metas): Este mecanismo busca la mínima desviación entre una solución candidata y una solución aspiracional que el decisor debe elegir previamente. Las preferencias se indican expresando la meta o solución aspiracional que al decisor le parece atractiva alcanzar (Smith et al., 2000).

Función de utilidad: Corresponde a una expresión en la que el decisor especifica una función matemática que representa sus preferencias en condiciones de incertidumbre, sobre objetivos combinados con el fin de encontrar una solución que maximice dicha función (Rosenthal, 1984).

Función de valor: Basa su funcionamiento de manera análoga a las funciones de utilidad, con la diferencia de que se representan las preferencias bajo condiciones de certidumbre (Taylor, 2022).

Punto de reserva: Busca la máxima desviación entre una solución candidata y la peor solución que puede tener cada objetivo; el punto de reserva también puede definirse como la peor solución del problema pero que sigue siendo aceptable para el decisor. Las preferencias se indican expresando el peor punto aceptado (Taylor, 2022).

Tasas de compensación: En esta técnica el decisor debe expresar la cantidad numérica que sacrificaría en un objetivo para ganar en otro, de esta manera el algoritmo orienta la búsqueda ubicando las soluciones que mejor responden a las tasas de compensación preferidas (Branke et al., 2001).

Los métodos a priori aquí presentados presentan algunas desventajas que se enunciarán a continuación (Bechikh et al., 2015; Lichtenstein & Slovic, 2006):

- Las preferencias usualmente son desconocidas, no desarrolladas o mal definidas.
- Pueden requerir un esfuerzo cognitivo alto para expresarlas.
- La fatiga producida por el esfuerzo podría conducir a errores o contradicciones que podrían afectar el proceso de búsqueda.
- Estos métodos limitan el espacio de búsqueda y si el decisor no tiene conciencia de su alcance, podría ocurrir que exista una solución potencial que corresponda a una preferencia más satisfactoria y que el decisor no había considerado.
- La ausencia de información sobre el comportamiento del problema podría provocar que el decisor emita especulaciones al momento de expresar sus preferencias.
- Los métodos presentados generalmente asumen que las preferencias son estáticas en todo el espacio de soluciones y podría ocurrir que sean cambiantes.

▪ **Métodos Interactivos de Incorporación de Preferencias:**

Estas técnicas suelen comenzar con la ejecución del algoritmo de optimización durante un número pequeño de ejecuciones, de tal manera que la búsqueda ofrezca un conjunto de soluciones iniciales. Acto seguido, estas soluciones se presentan al decisor para que aprenda de ellas y determine cuáles son las mejores soluciones o las más preferidas, es decir, su estructura de preferencias. Posteriormente estas preferencias se incorporan en el algoritmo para orientar la búsqueda hacia soluciones que no solamente sean eficientes sino que también satisfagan las preferencias indicadas. Luego se ejecutan nuevas iteraciones y se detiene nuevamente el algoritmo para que el decisor pueda confirmar o modificar sus preferencias. El proceso culmina cuando el decisor así lo determine.

Las técnicas interactivas suelen usar los mismos mecanismos de asignación de preferencias que utilizan las técnicas a priori, con la diferencia de que las técnicas interactivas establecen el número de veces que el algoritmo se interrumpirá para

consultarle al decisor sus preferencias, a partir del aprendizaje que obtendrá con las soluciones que arroja la búsqueda.

Estos métodos presentan la ventaja de que el decisor puede calibrar sus preferencias durante el proceso, ya que muchas veces éstas no se tienen claramente definidas (Cuartas Torres, 2009). Incluso también el decisor puede corregir errores que detecte sobre preferencias previamente establecidas (K. Li et al., 2018).

Algunas propuestas interactivas se enuncian a continuación:

Método Stem: Las preferencias se expresan relajando un objetivo satisfactorio, con el fin de mejorar otro objetivo que aún no lo es. Una solución se obtiene al minimizar la distancia de Tchebychev respecto al punto ideal (Cuartas Torres, 2009).

Método Visual Interactive Approach – VIA: En esta técnica se agregan los objetivos normalizados en una función y se optimiza según un punto de referencia (meta) informado por el decisor (Cuartas Torres, 2009).

Método Ecuador: Este método le ofrece al decisor una solución inicial obtenida por alguno de los métodos a priori indicados y adicionalmente le muestra los mejores y peores valores que se pueden obtener en cada objetivo. Con esta información el decisor define qué objetivo desea mejorar a costa de desmejorar otro y hasta qué punto desea deteriorar este último; de esta manera se ejecuta el proceso de búsqueda y el decisor va cambiando sus preferencias (Smith et al., 2000).

Articulación progresiva de metas: Propone una metodología basada en MOGA incluyendo una meta de logro interactiva que puede ser modificada por el decisor a medida que avanza el proceso de búsqueda (Fonseca & Fleming, 1999).

Optimizador evolutivo interactivo: Se utiliza el algoritmo NSGA-II y se encuentra la frontera de Pareto articulando preferencias de forma interactiva. El decisor escoge

la mejor solución indicando sus preferencias (utilizando por ejemplo la suma ponderada) (Deb & Chaudhuri, 2005).

Modificación de parámetros: Mencionan que no necesariamente en un método interactivo el decisor debe estar indicando sus preferencias, sino que también pueden cambiar los parámetros de preferencias dependiendo del cumplimiento de las restricciones (Cvetković & Coello, 2005).

La incorporación interactiva de preferencias presenta las siguientes desventajas (Bechikh et al., 2015; Tomczyk & Kadziński, 2020):

- En algunos trabajos se ha identificado que estas técnicas no alcanzan a combatir plenamente el esfuerzo mental que implica definir cuantitativamente las preferencias.
- La continua interacción con el decisor puede generar fatiga y saturar la carga cognitiva para expresar las preferencias.
- La atención del decisor puede disminuir si se expone a un número elevado de consultas y por ende se puede perder la calidad en la definición de las preferencias.
- Estas técnicas no necesariamente combaten la aparición de preferencias imprecisas e incoherentes, pues asumen racionalidad perfecta en el decisor.
- Este enfoque puede caer en el riesgo de desperdiciar recursos al tener que reiniciar la búsqueda desde el principio después de cada consulta al decisor.

Algunas aproximaciones de estos métodos intentan combatir estas desventajas estableciendo un límite máximo de interacciones y consultas, le permiten al decisor elegir cómo quiere expresar sus preferencias o utilizan interfaces gráficas y visualizaciones para reducir el esfuerzo cognitivo. A pesar de estos esfuerzos aún se siguen desarrollando mecanismos que contribuyan a superar estas limitaciones que siguen estando vigentes (Taylor, 2022).

- **Métodos A Posteriori de Incorporación de Preferencias:**

Proponen la asignación de preferencias a partir del hallazgo de la FP y permiten que el decisor pueda encontrar el segmento de dicha frontera que mejor se ajusta a sus preferencias. A continuación, se presentan algunas técnicas desarrolladas:

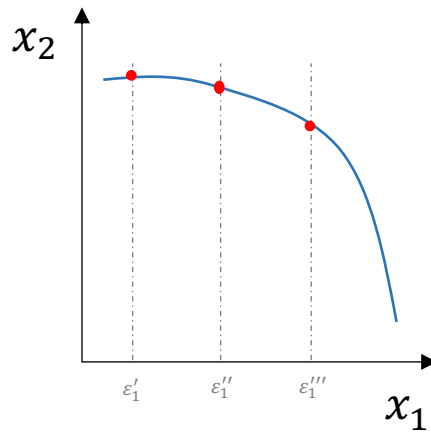
Método de las restricciones: Convierte un POM en un problema mono-objetivo, optimizando una función objetivo y representando las demás como restricciones. El problema se representa así (Smith et al., 2000):

$$\begin{aligned} & \text{Max } x_p(\theta) \\ \text{Sujeto a: } & x_m(\theta) \geq \varepsilon_m \quad m \neq p \quad m = 1, 2, \dots, M \end{aligned}$$

Es importante resaltar que M representa la cantidad de objetivos del problema, x_p es uno de ellos y es seleccionado arbitrariamente. El problema mono-objetivo se va optimizando para cada valor ε_m que se va definiendo. En definitiva, al final se obtendrán un conjunto de soluciones por cada ε_m . Esta técnica requiere que se conozca el rango de valores que puede tomar ε_m .

En la Figura 1-5 se representa un problema de dos objetivos, donde el objetivo x_2 se está maximizando y el objetivo x_1 se establece como restricción acotada por un coeficiente ε_1 . El problema se optimiza para ε_1' , ε_1'' y ε_1''' y en cada uno de ellos se encuentra una solución óptima representada en los puntos rojos de la Figura. La curva que se describe en azul representa la FP.

Método de la valoración sobre las tasas de intercambio: Esta técnica obtiene un conjunto discreto de soluciones no dominadas, tomando como referencia el método de las restricciones, y para cada una de ellas obtiene la tasa de intercambio entre cada par de objetivos respecto a su solución más próxima. El decisor valora las tasas de acuerdo con sus preferencias y de esta manera, forma una función de valoración sobre las tasas de intercambio. Al final, el decisor define cuál de las soluciones se aproxima a los intercambios más satisfactorios (Smith et al., 2000).

Figura 1-5: Representación gráfica del m. de las restricciones en dos objetivos.

Fuente: elaboración propia

Método de ponderación: Consiste en obtener un conjunto de alternativas no dominadas, sin establecer un orden de preferencias. A continuación, se divide cada objetivo con el valor de la alternativa y se suman mediante pesos de importancia relativa definidos por el decisor. El POM se convierte en un problema mono-objetivo y se realiza el respectivo proceso de búsqueda (Smith et al., 2000).

Algoritmos coevolutivos inspirados en preferencias - PICEAs: Las preferencias se modelan como un conjunto de soluciones que coevolucionan junto con la población (Li et al., 2015). En esta propuesta se utiliza la suma ponderada como estrategia para integrar las soluciones.

Algoritmo evolutivo de función de utilidad marginal: En este método se describe el rango preferido del usuario para cada objetivo, utilizando una función de utilidad marginal. De esta manera, el algoritmo puede suministrar al usuario un conjunto de soluciones en la zona de interés (Li et al., 2015).

Los algoritmos que se apoyan en esta estrategia presentan las siguientes debilidades:

- Estos métodos podrían entregar información excesiva que el decisor podría no saber cómo analizar para ajustarlas con sus preferencias.

- En problemas con más de tres objetivos y un gran conjunto de soluciones, la tarea de seleccionar una solución preferida es una tarea exigente.
- Su utilización puede suponer pérdida de recursos computacionales y de tiempo puesto que la mayoría de las soluciones se descartarán (Thiele et al., 2009).

1.5.2 Obtención e Incorporación de Preferencias Imprecisas

La obtención cuantitativa de preferencias depende de que el decisor tenga preferencias globales bien definidas y comprenda adecuadamente las limitaciones del problema. Este es un requisito difícil de cumplir y es especialmente complicado para los métodos que obtienen las preferencias de forma numérica (Thiele et al., 2009). También se ha encontrado que las personas generalmente no son buenas estimando frecuencias y probabilidades (Tversky & Kahneman, 1974), y esta dificultad podría vincularse a la complejidad de representar mediante expresiones cuantificables sus deseos a través de pesos, tasas de intercambio, valores de referencia, etc. Este inconveniente se presenta debido a las dificultades cognitivas que posee el decisor para llevar a cabo esta tarea.

Se ha encontrado que a las personas les resulta más fácil expresar sus preferencias en términos relativos que puntuar alternativas individuales utilizando una valoración directa (Kingsley, 2006). Cuando un decisor se enfrenta a un análisis de alternativas para establecer sus preferencias, recurre a estrategias de comparación sencillas que requieren poco esfuerzo cognitivo y que le ayudan a comprender mejor su estructura de preferencias. La mayoría de estas estrategias utilizan palabras o términos lingüísticos para describir la comparación y, por lo general, a estas expresiones se les asigna equivalencias numéricas para incorporarlas dentro de los MOEA. Algunas estrategias de comparación son (Taylor, 2022):

- **Priorización Top:** el decisor establece las n alternativas entre un conjunto de posibilidades con respecto a un criterio.
- **Agrupación:** el decisor clasifica las opciones con etiquetas de utilidad. Ejemplo: “preferida”, “no preferida”, “indiferente”.
- **Mejor y peor:** el decisor define un subconjunto de alternativas donde puede establecer las mejores y las peores según sus gustos.
- **Comparación por pares:** el decisor elige la opción que más le satisface.

Algunas técnicas de toma de decisiones multicriterio para alternativas discretas, como ELECTRE y PROMETHEE, basan su funcionamiento en estas estrategias para que el decisor pueda realizar comparaciones con menor carga cognitiva. Estos procedimientos se han usado tanto para clasificar soluciones como objetivos (por ejemplo, métodos de clasificación superior o outranking por su expresión en inglés) (Xin et al., 2018).

En la literatura se han encontrado una cantidad importante de modelos que establecen la imprecisión de forma numérica y no necesariamente de forma cualitativa, con palabras o comparando alternativas. Adicionalmente, es importante considerar que, aunque las preferencias puedan expresarse de forma imprecisa y/o cualitativa, también pueden incorporarse en un MOEA de forma a priori, interactiva o a posteriori.

Uno de los desarrollos que en los últimos tiempos se ha aceptado como adecuado para la modelación de preferencias inciertas es la Lógica Difusa. Su aceptación se basa en que esta disciplina da cabida al uso de palabras o expresiones lingüísticas y trata con verdades parciales que pueden ir desde completamente verdaderas a completamente falsas, y que se han acoplado notablemente con estructuras tradicionales de preferencias como pesos, puntos de referencia y métodos de clasificación (Taylor, 2022). A continuación, se presentan algunos conceptos base de la Lógica Difusa y se describen algunos aportes en la incorporación de preferencias imprecisas.

- **Lógica Difusa:**

La lógica difusa fue abordada por Lotfi Zadeh en 1965 y es una disciplina que permite modelar problemas cuando no se cuenta con información precisa. Términos como alto, medio, bajo, o poco, moderado, bastante, son expresiones que podrían explicar la magnitud o la dimensión de un fenómeno y que a su vez se hace difícil asignarle una única expresión numérica que la explique con precisión.

La lógica clásica se fundamenta en el uso de expresiones como falso y verdadero, que generalmente se traduce en “0” y “1”. La teoría de conjuntos, en la matemática tradicional, descansa en este fundamento y no considera puntos medios o ambiguos. Por ejemplo, al suponer que se quiere etiquetar con “1” si una persona

tiene una altura igual o superior a 1.8 metros y “0” si mide menos de dicho valor, en la lógica clásica si alguien mide 1.79 su etiqueta será “0” no importando que dicho valor esté muy próximo a 1.8. La lógica difusa (en adelante LD) es capaz de abordar estos puntos intermedios y de incorporar el manejo de la ambigüedad en estos problemas.

La LD funciona con la manipulación de conjuntos difusos y establece que la pertenencia de un elemento a un conjunto determinado está determinada por un valor cualquiera comprendido entre 0 y 1. Así mismo, bajo este paradigma es posible modelar variables difusas que tienen un comportamiento impreciso. Según Duarte et al. (2022), un conjunto difuso (en adelante CD) se define como:

Los conjuntos difusos son aquellos cuyos elementos pueden pertenecer parcialmente a dicho conjunto. Se utiliza el grado de pertenencia para medir qué tanto pertenece un elemento a un conjunto difuso. Formalmente, un conjunto difuso es un conjunto de parejas $\{(x, \mu_{A(x)}), x \in U, \mu_{A(x)} \in [0,1]\}$, donde x es un elemento del Universo de Discurso U , y $\mu_{A(x)}$ es el grado de pertenencia de x al conjunto A . Nótese que las fronteras del conjunto difuso no son precisas, y de allí su denominación de conjuntos difusos.

Si tenemos un conjunto discreto A , con la siguiente información:

$$x = \{0,1,2,3,4,5,6,7,8,9,10\}$$

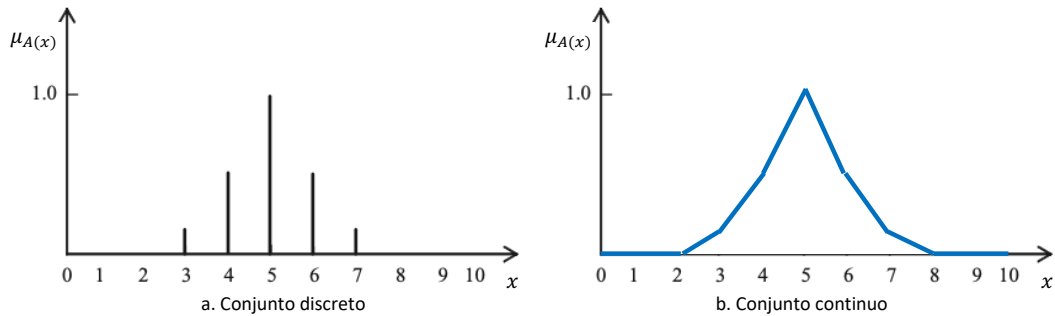
$$\mu_{A(0)} = \mu_{A(1)} = \mu_{A(2)} = \mu_{A(8)} = \mu_{A(9)} = \mu_{A(10)} = 0$$

$$\mu_{A(3)} = \mu_{A(7)} = 0.2$$

$$\mu_{A(4)} = \mu_{A(6)} = 0.5$$

$$\mu_{A(5)} = 1$$

Donde x son los elementos de un Universo U y $\mu_{A(x)}$ es el grado de pertenencia entre 0 y 1 de un conjunto A , entonces una representación gráfica de dicho conjunto A sería como la que se muestra en la Figura 1-6-a:

Figura 1-6: Representación gráfica de conjuntos difusos.

Fuente: elaboración propia inspirada en Pedrycz et al., (2011)

La función $\mu_A(x)$, que expresa los grados de pertenencia de un CD, se puede representar mediante una función de pertenencia cuyo rango está comprendido entre 0 y 1. La Figura 1-6-b ejemplifica lo que sería un conjunto continuo y la curva azul equivale a la gráfica de dicha función de pertenencia.

En la LD un número difuso se representa como un CD y según Duarte et al. (2022) es un intento de formalizar una expresión tal como “aproximadamente x ”. Un número difuso o un CD tienen diferentes formas de representación a partir del estilo de su función de pertenencia, pudiendo ser curvas, trapezoidales, triangulares, planas, entre otras.

A partir de la LD, con los CD es posible representar una variable lingüística (en adelante VL), una variable cuyo valor es una etiqueta, y compuesta de elementos que se enuncian en la Tabla 1-1.

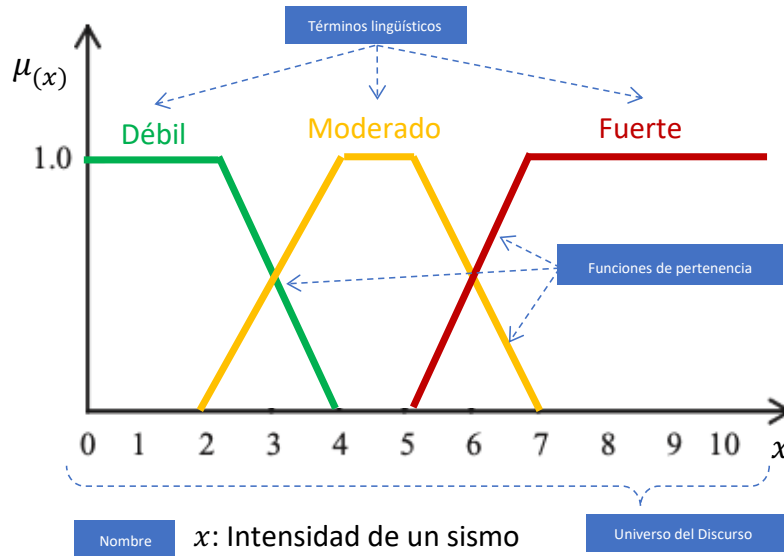
En la Figura 1-7 se ejemplifican gráficamente los elementos propuestos en la Tabla 1-1. Aquí se puede apreciar cómo una expresión lingüística se relaciona con elementos numéricos a través de las funciones de pertenencia.

Tabla 1-1: Elementos que componen una VL.

Elemento	Descripción	Ejemplo
Nombre	Expresión nominal que identifica la variable	Intensidad de un sismo
Términos lingüísticos	Etiqueta que puede tomar la VL y que describe su comportamiento. Son representadas como un CD.	Débil, Moderado, Fuerte
Universo de discurso	Valores numéricos a los cuales se puede asociar la VL.	$[0, +\infty)$ medido en Richter
Función de pertenencia	Corresponde a la función matemática que describe la pertenencia de cada CD.	Funciones trapezoidales

Fuente: elaboración propia

Figura 1-7: Representación gráfica de una VL.



Fuente: elaboración propia

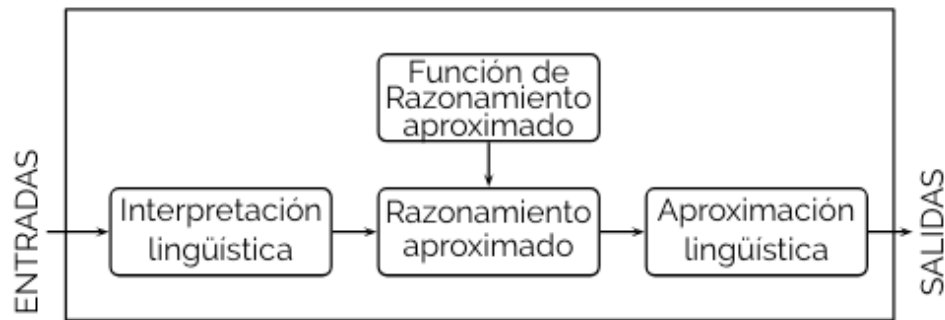
La manera como se pueden realizar cálculos matemáticos entre números y conjuntos difusos se estudia en la aritmética difusa y su aplicación se expone ampliamente en Pedrycz et al. (2011).

Así como en la lógica clásica las funciones permiten relacionar variables dependientes e independientes de la forma $y = f(x)$, de manera análoga en la lógica difusa es posible relacionar VL con otras a través de los Sistemas de Computación con Palabras, Sistemas Basados en Reglas o Sistemas de Inferencia

Difusa (en adelante SID). Según Duarte et al. (2022) los SID emplean aritmética difusa para calcular palabras a partir de otras palabras, están compuestos por unas VL de entrada, VL de salida y un bloque de razonamiento aproximado que calcula números difusos a partir de números difusos usando una función de razonamiento aproximado.

Un SID sigue la siguiente representación que se ilustra en la Figura 1-8. De acuerdo con ella, los valores que se toman como entradas en un SID sufren un proceso de interpretación lingüística donde se asocia cada valor numérico que toma la función con una expresión lingüística de las VL de entrada definida.

Figura 1-8: Esquema que representa un Sistema de Inferencia Difusa.



Fuente: Duarte et al., 2022

La función de razonamiento aproximado contiene un conjunto de reglas que relaciona los valores posibles de las VL de entrada con las etiquetas de la VL de salida. A continuación, se ejemplifica la manera como se establecen estas reglas:

Suponiendo 2 VL de entrada (x_1 y x_2) y 1 VL de salida (y) así:

x_1 con 3 etiquetas: “Bajo”, “Medio” y “Alto”

x_2 con 3 etiquetas: “Débil”, “Moderado” y “Fuerte”

y con 2 etiquetas: “Aceptable” e “Inaceptable”

Entonces las VL anteriores, se podrían construir reglas de la función de razonamiento aproximado de la forma:

Si x_1 es "A" y x_2 es "B" entonces y es "C"

Por tanto, para ejemplo en cuestión sería posible construir una matriz de reglas como se indica en la Tabla 1-2. Las celdas vacías de dicha tabla contendrían las respectivas etiquetas de la variable de salida.

La función de razonamiento aproximado podría expresarse de la siguiente forma:

Función de Razonamiento Aproximado: $F(x_1, x_2) = y$

Tabla 1-2: Matriz con base de reglas del SID.

Reglas de la función de razonamiento aproximado		x_2		
		Débil	Moderado	Fuerte
x_1	Bajo			
	Medio			
	Alto			

Fuente: elaboración propia

El bloque de razonamiento aproximado de la Figura 1-8 actúa de acuerdo con la base de reglas definidas y el resultado que emita lo llevará al bloque de aproximación lingüística para convertir el valor obtenido en una expresión lingüística de la variable de salida. En conclusión, cuando se evalúa un conjunto de valores de entrada (ejemplo x_1, x_2) en un SID, este entregará un valor numérico y una expresión lingüística para la variable de salida.

Las reglas del SID se construyen considerando la experiencia que tiene el decisor del problema, representando todo su conocimiento experto. Los SID, como sistemas expertos, dependen en gran medida del contexto de aplicación, de ahí que un sistema experto viable requiere un amplio conocimiento del dominio del problema a partir de una extensa exploración de instancias representativas del problema en cuestión (Rachmawati & Srinivasan, 2006).

Los SID tienen la ventaja de que utilizan una representación natural del conocimiento, el cual es fácil de obtener, representar y revisar. Así mismo, no requiere el uso de un número grande de reglas, ya que al computar los valores que se determinan en las entradas generan unos valores de salida a partir del mecanismo de interpolación que utiliza la función de razonamiento aproximado con las reglas difusas (Bonissone, 2001). Los SID se presentan como una solución potencial para modelar preferencias en un POM; ya que, usualmente, expresar una valoración subjetiva a través de un número exacto puede ser muy difícil y limitado; y en contraposición podría ser más fácil para el decisor expresar sus preferencias mediante una VL.

- **Preferencias con Lógica Difusa:**

Una de las ventajas más notables de la LD referente al modelamiento de preferencias imprecisas tiene que ver con la versatilidad que ofrece para incorporar otras estructuras de preferencia de corte cuantitativo. Así mismo, contribuye a disminuir el esfuerzo mental al que se expone el decisor cuando emite sus preferencias mediante palabras o realiza comparaciones simples. A continuación, se presentan algunos abordajes en este campo:

Pesos de importancia relativa difusos: En Choon & Tilahun (2011) cada función objetivo tiene asociado un peso de importancia relativo difuso, los objetivos se integran en uno solo a través de una suma ponderada, y los pesos se seleccionan aleatoriamente a través de una función de distribución de probabilidad tomando como referencia la función de pertenencia del peso. Otro trabajo que explora esta técnica es el de Jin & Sendhoff (2002).

Puntos de referencia difusos: Cuando el decisor no tiene claro un punto de referencia preciso, puede expresar puntos de referencia difusos mediante un CD con función de pertenencia triangular, cuyos vértices se expresan así: Sea r un punto de referencia difusa, cuyo CD sigue una función de pertenencia de forma triangular cuyos vértices se expresan así $r = (r^{min}, r^{mejor}, r^{max})$. Aquí el decisor debe expresar 3 valores que representan lo máximo y lo mínimo en el rango que

dicha solución puede moverse y el mejor valor que representa dicho punto. Esta aplicación puede apreciarse con mayor detalle en el trabajo de Xiong et al. (2013).

Tasas de intercambio difusas: Luo et al. (2018) abordan las preferencias a través de tasas de intercambio difusas establecidas en intervalos; estos valores se expresan en una serie de vectores de preferencias precisos y distribuidos uniformemente con respecto a los objetivos. La información de preferencias se incorpora en un algoritmo evolutivo MOEA/D para orientar el proceso de búsqueda.

Las técnicas anteriores, aunque incorporan imprecisión, aún le exigen al decisor expresar sus preferencias de forma numérica y no mediante expresiones cualitativas. Seguidamente se mencionarán algunos trabajos que han abordado el uso de expresiones cualitativas.

Cálculo de fuerza superior a través de clasificación (outranking): Es inspirado en los métodos de clasificación ELECTRE y PROMETHEE, se basa en la definición de la importancia relativa entre pares de objetivos i y j en etiquetas lingüísticas, y a cada etiqueta se le asigna un valor numérico de acuerdo con su importancia, denominado concordancia c_{ij} (Shen et al., 2009). En la Tabla 1-3, se muestra un ejemplo de etiquetas de 5 grados para un POM de objetivos x_i donde $i = 1, 2, \dots, m$. Con esta definición se calcula el factor de importancia P_i del objetivo x_i , así:

$$P_i = SL_i / \sum_{j=1}^m SL_j$$

$$SL_i = \sum_{j=1, j \neq i}^m c_{ij}$$

El factor de importancia P_i , se incorpora en el algoritmo evolutivo.

Tabla 1-3: Definición cualitativa para clasificar objetivos.

Importancia relativa del objetivo x_i con respecto al objetivo x_j	Valor de concordancia c_{ij}	Valor de concordancia c_{ji}
Igual	4+0	4-0
Entre igual y un poco importante	4+1	4-1
Entre un poco y más importante	4+2	4-2
Entre más y muy importante	4+3	4-3
Muy importante	4+4	4-4

Fuente: elaboración propia inspirada en Shen et al. (2009)

En dicho trabajo se construye un SID que considera dos entradas así:

- Una expresión de importancia relativa para cada objetivo definida como una VL que toma valores de “No importante”, “Medianamente importante” y “Muy importante”.
- Una variable que calcula el grado de mejora de una solución, que se calcula considerando la diferencia entre el valor máximo y el mínimo de todas las soluciones en un objetivo en particular. Esta VL toma valor de “Negativa grande”, “Negativa pequeña”, “Cero”, “Positiva Pequeña” y “Positiva Grande”.

La VL de salida del SID expresa la fuerza que una solución tiene en función de la importancia del objetivo y el grado de mejora de la solución. Esta VL toma valor de “Negativa grande”, “Negativa pequeña”, “Cero”, “Positiva Pequeña” y “Positiva Grande”.

El SID otorga información para ser incorporada en la función de evaluación del algoritmo. Este trabajo asume que el factor de importancia relativa cualitativa es constante en todo el espacio de búsqueda y ello no necesariamente es cierto. Una versión similar más reducida implementado en un generador de reluctancia conmutada puede apreciarse en (Li et al., 2021).

Clasificación de alternativas (outranking): La propuesta de Leyva-Lopez & Aguilera-Contreras (2005) se inspira en la técnica ELECTRE-III a través de la clasificación de alternativas discretas no dominadas. Aunque el procedimiento de clasificación

ayuda al decisor a realizar las comparaciones, la aplicación de la técnica podría inducir un incremento en la carga cognitiva del decisor y por ende proveer soluciones ineficientes.

Clasificación de objetivos y aplicación de un SID: Kim et al. (2012) se inspiran en el uso de la técnica PROMETHEE y la integral difusa para evaluar y clasificar soluciones no dominadas. En esta propuesta se utiliza un SID para generar soluciones potenciales para probar la técnica en un problema real.

Representación de tasas de intercambio en alternativas discretas: Ramakrishnan & Hasan (2013) proponen la implementación de un SID de 2 entradas y 1 salida en un problema biobjetivo, donde cada entrada contiene 4 etiquetas que representan los atributos de un objetivo (por ejemplo un objetivo es el tamaño de un computador: sus etiquetas serán Muy Pequeño, Pequeño, Medio y Grande; y la salida corresponde a 4 alternativas discretas, como 4 computadores diferentes, según el ejemplo de dicho trabajo). Las funciones de pertenencia de las variables de entrada son modelos gaussianos y la variable de salida está modelada en una función triangular. Este trabajo presenta la desventaja de que no modela una interrelación entre los objetivos al modelar las preferencias y solo trabaja con unas cuantas reglas difusas, por lo que no considera ciertas combinaciones entre las variables de entrada y la salida que podrían aportarle información valiosa al decisor.

Obtención de preferencias mediante un SID: En la propuesta de Bonissone (2001) se plantea el modelamiento de preferencias mediante un SID, estableciendo una base de reglas para 2 objetivos donde cada uno representa una VL de entrada y la VL de salida modela la preferencia. A diferencia del trabajo de Ramakrishnan & Hasan (2013), Bonissone (2001) sí define la totalidad de la base de reglas pero no las incorpora en un MOEA, únicamente las expresa y deja abierta la posibilidad de ser incorporada en un algoritmo evolutivo.

Objetivos y FP difusas: Jamwal et al. (2019) convierte las funciones objetivo en expresiones difusas y utiliza un SID para encontrar FP difusas, aunque este trabajo no incorpora preferencias y no resuelve la toma de decisiones. Su mención es importante para destacar que la incorporación de lógica difusa en optimización

multiobjetivo podría ayudar en exploraciones de mayor alcance, evitar la convergencia prematura y ofrecer resultados satisfactorios en problemas de optimización de muchos objetivos.

1.6 Conclusión

La búsqueda de soluciones óptimas en POM ha sido abordada ampliamente en diferentes técnicas metaheurísticas, particularmente mediante algoritmos evolutivos como los algoritmos genéticos. Técnicas como NSGA-II y NSGA-III, que se basan en encontrar un conjunto de soluciones óptimas a través de la dominancia de Pareto y la preservación de la dispersión de las soluciones, han mostrado eficiencia en la aplicación de diferentes problemas: el primero ha mostrado buen desempeño en situaciones que involucran hasta 3 objetivos, y el segundo ha revelado mayor éxito en casos donde participan más de 3 objetivos. En estas técnicas funcionan adecuadamente encontrando la FP en un número no muy grande de generaciones.

Sin embargo, no basta con conocer los elementos que componen una FP, ya que los decisores suelen interesarse sólo por una parte concreta de la FP. Es por ello que es necesario introducir las preferencias del decisor en el POM, esto no sólo puede ayudar al decisor a obtener sus soluciones preferidas, sino que también podría ayudar a simplificar el proceso de búsqueda.

De forma adicional los POM pueden tener FP desconocidas y grandes, y puede ser difícil para el decisor realizar una elección adecuada a partir de todas las alternativas que en ella se encuentran, dado que puede desconocer la dimensión del problema, no tener una idea concreta de sus gustos y deseos y puede expresar incoherencia y poca racionalidad al momento de emitir sus juicios. Para lidiar con las dificultades se han desarrollado diferentes mecanismos para expresar las preferencias, ya sea de forma directa (precisa) o indirecta (imprecisa). Así mismo se hace necesario establecer el momento propicio para incorporar las preferencias en los algoritmos de optimización; en la literatura se reportan 3 momentos: a priori, interactivo y a posteriori. Todos los diseños que se han elaborado para modelar e incorporar preferencias (en cualquiera de los 3 momentos) presentan ventajas y desventajas, no existe uno que reporte plena capacidad de respuesta para combatir las

debilidades encontradas y por ello sigue siendo un campo de trabajo la innovación en el diseño e incorporación de preferencias en POM.

Teniendo presente lo anterior, modelar las preferencias mediante palabras o expresiones lingüísticas es una aproximación interesante que permite traducir con mayor concordancia el proceso mental que un decisor utiliza para indicar sus gustos y deseos. Por ello, el uso de la lógica difusa como herramienta para lograrlo resulta prometedor, debido a que expresar las preferencias de forma cualitativa reduce las limitaciones que trae consigo la revelación cuantitativa de preferencias.

Dados los diferentes abordajes que se han realizado para expresar preferencias a través de LD, se identifica que aún no consideran que un objetivo no necesariamente es preferido por sus atributos en sí mismo, sino que su preferencia también puede alterarse por el comportamiento de otros objetivos. También se encuentra que los trabajos analizados, y que incorporan SID, no han incluido soluciones en escenarios continuos.

2. Propuesta de Metodología

Tomando como base que las personas, al momento de valorar cuantitativamente percepciones subjetivas, pueden cometer errores debido a las dificultades cognitivas que posee el decisor para llevar a cabo esta tarea (Kahneman, 2012) (lo cual representa una limitación para la estimación numérica de preferencias) y que los SID utilizan una representación natural del conocimiento del decisor (Bonissone, 2001), operan de forma similar al razonamiento humano, se adaptan fácilmente a casos con pocas variaciones de parámetros, permiten combinar en forma unificada expresiones lingüísticas con datos numéricos y no requiere algoritmos sofisticados para su implementación (Duarte, 1999), se presenta el siguiente como objetivo general de este trabajo el siguiente:

Proponer una metodología que incorpore un Sistema de Inferencia Difusa para la asignación de las preferencias de un decisor en la resolución de un problema de optimización multi-objetivo con Algoritmos Genéticos.

Para desarrollar este objetivo se proponen los siguientes objetivos específicos:

1. Elegir el método de asignación de preferencias que permita modelar las expectativas del decisor.
2. Determinar la manera cómo se implementará la lógica difusa al modelo de asignación de preferencias.
3. Establecer los algoritmos evolutivos a los cuales se les incorporará el Sistema de Inferencia Difusa y determinar el momento en el cual debe ser involucrado.
4. Implementar la propuesta en un problema prueba (test) y validar su funcionamiento.
5. Verificar que las soluciones presentadas por el modelo sean eficientes.

El trabajo que se desarrollará en este estudio se acotará a los problemas de optimización de dos objetivos y se sentarán las bases para promover una eventual ampliación a 3 o más objetivos que puedan ser abordados en trabajos futuros.

A continuación, se presenta una propuesta para plantear posibles soluciones al desarrollo de estos objetivos.

2.1 Elección de Método de Asignación de Preferencias

De acuerdo con el marco teórico de este estudio, la incorporación de preferencias puede realizarse antes (a priori), durante (interactiva) o después (a posteriori) del proceso de búsqueda de soluciones óptimas, y cada uno de estos momentos presenta desafíos distintos para el decisor. Uno de ellos tiene que ver con el conocimiento que el decisor tenga del comportamiento del problema, puesto que si el decisor no conoce las alternativas que ofrece el problema ni tiene una idea de las potenciales soluciones que podría entregar, entonces no tendría elementos suficientes para expresar sus preferencias.

Es posible suponer que las técnicas a posteriori tienen la ventaja de revelar al decisor un vasto conjunto de soluciones que conforman la FP. Sin embargo, el esfuerzo mental que le implicaría al decisor analizar estas opciones se convierte en una desventaja para este tipo de técnicas. Las técnicas interactivas ofrecen alternativas más acotadas e informadas para el decisor, pero lidian con el dilema de tener que establecer un número adecuado de interacciones con el decisor para evitar una posible fatiga por exceso en las consultas.

Sería interesante brindarle al decisor una idea del comportamiento del problema para que aprenda sobre él y pueda definir sus preferencias antes de iniciar el proceso de búsqueda, se le pueden proporcionar pocos datos al decisor de tal forma que sean suficientes para inferir el comportamiento general del problema. Así, no se ejecutaría en vano el algoritmo de optimización, sino que primero se propendería por un adecuado conocimiento del POM: se pretende, entonces, que el decisor primero aprenda del mismo antes de buscar posibles soluciones para no perder tiempo y recursos computacionales. Por esta razón en este estudio se propondrá una incorporación a priori de preferencias, apostando por un enfoque de aprendizaje previo, donde el decisor podrá elaborar una idea de sus preferencias con información real del problema a resolver. De esta manera se alcanza el cumplimiento del primer objetivo específico de este trabajo.

Seguidamente se presenta una estrategia de aprendizaje previo para incorporar preferencias de forma a priori al proceso de optimización.

Fase de Aprendizaje Inicial

Los métodos de simulación de problemas, particularmente los estocásticos, permiten encontrar posibles soluciones a un modelo siguiendo patrones probabilísticos explorando el espacio de soluciones. En específico, la simulación Monte Carlo utiliza una búsqueda aleatoria donde cualquier solución de prueba seleccionada es totalmente independiente de decisiones previas y de sus resultados. Aunque estos métodos han sido utilizados para encontrar soluciones óptimas, se ha encontrado que no siempre garantizan su hallazgo (Aguilar Arroyo, 2023). Sin embargo, para efectos de este estudio se propone el uso de la simulación Monte Carlo antes de ejecutar un algoritmo de optimización con el fin de:

- Obtener una idea previa del comportamiento del problema a partir de la simulación de soluciones.
- Librar de los posibles sesgos del decisor en la generación de soluciones, considerando únicamente la aleatoriedad del método de simulación.
- Aprovechar el componente aleatorio de la simulación para explorar ampliamente el espacio de soluciones factibles.
- Ofrecer al decisor el resultado de la simulación para que aprenda sobre las posibilidades del problema infiriendo las potencialidades del mismo y así obtenga una base para expresar sus preferencias.

En este estudio se propone someter un POM a una fase de simulación Monte Carlo donde se genere un conjunto muy amplio de z posibles soluciones (Ejemplo: $z = 1000$ o más) y posteriormente mostrarle de forma gráfica la simulación al decisor de manera que pueda formarse una idea visual de las posibilidades del problema antes de ejecutar un MOEA. En este punto se dotaría al decisor de información básica para conformar su estructura de preferencias.

La fase de simulación, además de brindar elementos de aprendizaje al decisor, será aprovechada para obtener un subconjunto de soluciones conformará la población inicial del proceso de optimización, la cual se denominará semilla. Con esta propuesta se busca

que el algoritmo evolutivo no tenga que descubrir buenas soluciones, sino que en lugar de comenzar en cero, mejore las soluciones que se le han proporcionado en la semilla.

2.2 Construcción de Estructura de Preferencias

Al retomar los hallazgos identificados en la literatura frente a las ventajas que trae expresar indirectamente las preferencias y con el ánimo de aprovechar las bondades que la LD tiene para aproximar modelos con imprecisión, se propone que las preferencias sean modeladas mediante un SID de forma que el decisor, con su experiencia, pueda deducirlas mediante reglas. A continuación, se enuncian una serie de pasos para realizar esta construcción:

1. Presentar los resultados de la simulación Monte Carlo de forma visual, de forma que el decisor pueda identificar visualmente el valor mínimo y máximo posible para cada objetivo. Vale la pena anticipar que, estos valores obtenidos a priori a la optimización, podrían ser reevaluados en las iteraciones de la ejecución del algoritmo evolutivo.
2. Construir una VL por cada objetivo involucrado en el problema. Cada variable lingüística puede definirse con distintos elementos, y a continuación se describe la manera cómo se deben establecer dichos elementos para cada VL:
 - a. Nombre de la VL: Satisfacción que el decisor experimenta por un valor determinado de dicho objetivo.

x : Satisfacción del decisor en el objetivo

- b. Universo de la VL: Definido como el rango comprendido entre el valor mínimo y máximo obtenido en cada objetivo durante la simulación Monte Carlo.

$$U_x \in [x_{min}, x_{max}]$$

- c. Términos lingüísticos: Representan al conjunto de r palabras que identifican la magnitud cualitativa que describe la VL. El criterio para establecer la cantidad de etiquetas r podría darse analizando el comportamiento del objetivo a partir de la simulación Monte Carlo, estableciendo zonas de mayor o menor concentración de valores para el objetivo en cuestión o incluso estableciendo zonas de soluciones de igual tamaño. Para efectos de estudio se toma como referencia el método de las restricciones donde las r subdivisiones del objetivo que hace parte de las restricciones se analizan a igual tamaño.
- d. Funciones de pertenencia: Para convertir las valoraciones cualitativas (términos lingüísticos) en representaciones cuantitativas, se hace necesario construir una función semántica (función de pertenencia) que indique cuál es la equivalencia que tienen los valores del universo del discurso de la variable difusa en un rango de 0 a 1.
Dicha función se asumirá que tiene una forma trapezoidal donde todas las etiquetas (términos) lingüísticos se distribuirán de forma equitativa en todo el intervalo del universo del discurso.
3. Construir una VL que represente la preferencia del decisor conformada por el comportamiento simultáneo de los objetivos que integran el problema. Se propone que dicha variable se defina así:
- a. Nombre de la VL: Disposición por mejorar la obtención del objetivo x_1 con respecto a sacrificar la obtención del objetivo x_2 .
 \mathcal{U} : Disposición por mejorar la obtención del objetivo x_1 con respecto a sacrificar la obtención del objetivo x_2 .
- b. Universo de la VL: Se establece que esta variable tendrá se expresará mediante un rango numérico comprendido entre 0 y 100.

$$U_{\mathcal{U}} \in [0,100]$$

- c. Términos lingüísticos: Se representan de forma análoga como se describió el diseño de la VL de los objetivos. Estas expresiones otro conjunto de r palabras que identifican la magnitud cualitativa que describe la VL de salida.
- d. Funciones de pertenencia: De la misma manera que se realizó para las VL de los objetivos, se construye la función semántica que permitirá convertir las valoraciones cualitativas de \mathcal{Y} en representaciones cuantitativas que indiquen cuál es la equivalencia que tienen los valores del universo del discurso de la variable difusa en un rango de 0 a 1.

Dicha función también se asumirá que tiene una forma trapezoidal donde todas las etiquetas (términos) lingüísticos se distribuirán de forma equitativa en todo el intervalo del universo del discurso.

4. Diseñar una base de reglas que representen una función de razonamiento aproximado, de tal forma que $F(x_1, x_2) = \mathcal{Y}$.

A partir de las variables difusas definidas: x_1, x_2, \mathcal{Y} , es posible relacionarlas de tal forma que, de acuerdo con la satisfacción que el decisor experimente en las variables x_i , se le pida establecer su disposición al sacrificio entre objetivos \mathcal{Y} . Por tanto, se construirán una base de reglas de la forma:

$$\begin{aligned} \text{Si } x_1 = x_1^{(1)} \wedge x_2 = x_2^{(1)} &\Rightarrow \mathcal{Y} = \mathcal{Y}^{(1)} \\ &\vdots \\ \text{Si } x_1 = x_1^{(t)} \wedge x_2 = x_2^{(t)} &\Rightarrow \mathcal{Y} = \mathcal{Y}^{(t)} \end{aligned}$$

t representa el número de reglas que tiene el sistema de inferencia difuso. $t = r_1 r_2$ (r_1 y r_2 corresponde a la cantidad de términos lingüísticos de las VL x_1 y x_2 respectivamente). Esta base de reglas representará un sistema de tipo MISO (Multiple Inputs – Single Output), donde las variables x_1 y x_2 corresponden a las entradas (Inputs) del sistema y la variable \mathcal{Y} corresponde a la salida (Output).

Para que el sistema de inferencia funcione de forma adecuada se debe cumplir que:

- El conjunto de reglas debe ser completo, es decir, debe cubrir todo el espacio de las variables de entrada.
- El conjunto de reglas debe ser coherente, es decir, no pueden existir dos reglas con igual valor en las entradas y diferente valor de salida.

El decisor deberá combinar todas las opciones de las variables de entrada, y para cada combinación deberá indicar cual es la variable de salida que mejor representa ese par de opciones.

Al momento de expresar estas reglas, el decisor debería utilizar la representación gráfica de las soluciones generadas por la simulación Monte Carlo. Allí deberá analizar el comportamiento de las soluciones, identificar las zonas dónde están más dispersas, dónde están más concentradas, dónde hay cambios más repentinos en la distribución de las soluciones, dónde se ubican posibles zonas infactibles, y además preguntarse si visualiza posibles soluciones que en la vida real podría satisfacerle más que otras. Tomando como base estas reflexiones, se espera que el decisor exprese sus preferencias a partir de las revelaciones que la simulación le aporta, y a partir del posible conocimiento previo que tenga del entorno del problema y que desee vincular a la reflexión. Esta propuesta asume racionalidad perfecta y que no existen incoherencias en el decisor.

Al finalizar estos pasos, el decisor tendrá una base de reglas que expresan sus preferencias mediante palabras y expresiones lingüísticas que dan cuenta del comportamiento del objetivo percibido como respuesta al aprendizaje que tuvo al estar expuesto a los resultados de la simulación. Esta base de reglas representa una función difusa de forma $F(\mathcal{X}_1, \mathcal{X}_2) = \mathcal{Y}$, y cumple con los siguientes atributos:

- No considera preferencias estáticas, sino que establece que son cambiantes y dependen del valor que tome cada objetivo de forma conjunta en una posible solución.

- Las preferencias no son especulativas sino que responden al análisis que el decisor hace del problema a partir de la información que recibe de la simulación.
- Son imprecisas y ello no impide que el decisor pueda hacerse una idea general del problema y así orientar el proceso de búsqueda en un MOEA.

Esta estructura de preferencias propuesta responde a una formulación mediante LD según lo planteado en el segundo objetivo específico de este trabajo. El presente estudio incorporará esta estructura de preferencias en un MOEA.

2.3 Elección del Algoritmo Evolutivo

De acuerdo con la revisión teórica presentada en este estudio, los algoritmos genéticos son un paradigma de los algoritmos evolutivos que han sido reconocidos por su éxito en el hallazgo de soluciones eficientes, y suelen ser robustos ante cambios (Cortez et al., 2019; Santana & Coello, 2006). Los algoritmos genéticos basados en dominancia de Pareto que han sido usados más frecuentemente en la literatura para problemas de múltiples y muchos objetivos son el NSGA-II y el NSGA-III respectivamente, y con base en ellos se ha verificado el rendimiento de otros algoritmos dado el buen desempeño que estos presentan. Además, una gran variedad de problemas de prueba se ha considerado para ser usados en estos algoritmos. Por esta razón, este estudio utilizará el NSGA-II y el NSGA-III para realizar el proceso de búsqueda de soluciones óptimas posterior a la incorporación de la estructura de preferencias.

Los algoritmos NSGA-II y NSGA-III inician el proceso de búsqueda con una población inicial que se genera aleatoriamente. En este estudio se propone inicializar el algoritmo con una población semilla la cual sería un subconjunto de las soluciones encontradas mediante simulación Monte Carlo. El propósito de esta solución semilla será brindar mayor dirección a la búsqueda y apoyar al algoritmo para converger a soluciones eficientes. A continuación, se describe una propuesta de pasos para obtener esta población semilla:

- Tomar cada una de las soluciones obtenida mediante simulación Monte Carlo y evaluarlas en el SID que describe las preferencias del decisor. Esta evaluación entregará el valor numérico y el valor lingüístico \mathcal{Y} correspondiente a cada solución.
- Clasificar las soluciones simuladas en cada uno de los términos lingüísticos de \mathcal{Y} .

- Seleccionar una muestra pequeña de soluciones simuladas perteneciente al término lingüístico de \mathcal{Y} que el decisor prefiera. La muestra total de soluciones seleccionadas es una cantidad w elegida por el decisor.
- Depositar en un archivo a los individuos correspondientes a las soluciones contenidas en la muestra de tamaño w , expresados en los valores de las variables de decisión del problema. El valor w corresponde a la cantidad de individuos que se espera que contenga la población del proceso de optimización.

Se recomienda que la población semilla contenga elementos de la expresión lingüística de \mathcal{Y} que el decisor prefiera. El decisor deberá definir una estrategia para construir esta población, por ejemplo: podría seleccionar una cantidad equitativa de individuos por cada expresión o seleccionar una cantidad diferente de individuos de cada expresión, eligiendo una mayor cantidad de individuos perteneciente de aquellas expresiones más preferidas y una menor cantidad de individuos perteneciente de aquellas expresiones menos preferidas. Los individuos de las etiquetas más preferidas aportarán a una mayor convergencia del proceso de búsqueda (explotación) y los individuos de las etiquetas menos preferidas aportarán a una menor convergencia del proceso de búsqueda (exploración).

Con el fin de dirigir la optimización, y para que el algoritmo no se concentre en soluciones ubicadas en una zona poco preferida del espacio de soluciones, se propone que el decisor elija el término lingüístico más preferido e incorpore el SID en una restricción de tal forma que si el algoritmo encuentra una solución que equivale a una etiqueta lingüística distinta a la preferida entonces el algoritmo, por influencia de la restricción, descartará dicha solución.

Posteriormente se definen los parámetros que conducirán los algoritmos:

- Cantidad de generaciones.
- Probabilidad de cruzamiento.
- Probabilidad de mutación.

Al final de las ejecuciones el algoritmo entregará el segmento de soluciones no dominadas que hacen parte de la FP del POM y que mejor se ajustan a las preferencias del decisor. El algoritmo concluye ofreciendo una solución óptima bajo algunos criterios adicionales.

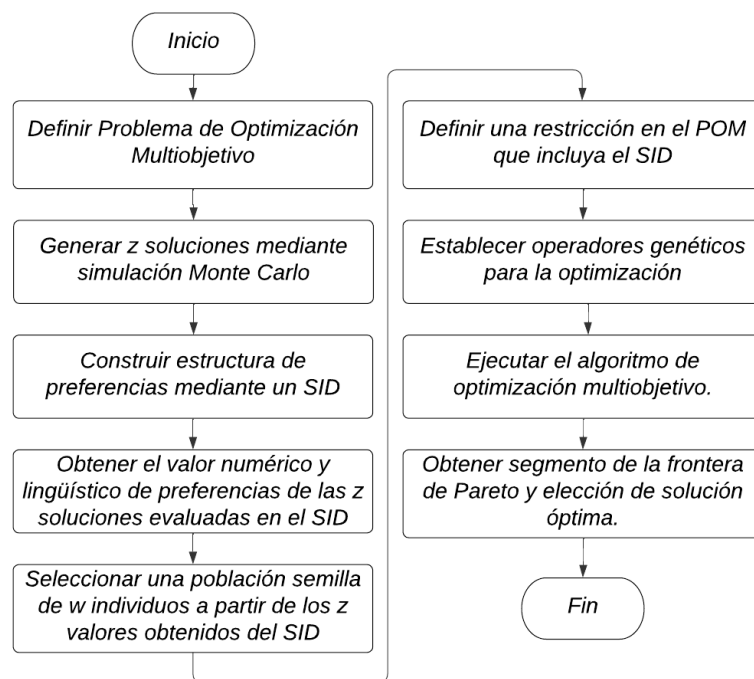
En conclusión, para dar cumplimiento con el tercer objetivo específico de este estudio, se propone utilizar los algoritmos NSGA-II y NSGA-III, con una población semilla que representa una muestra preferida proveniente de la simulación Monte Carlo e incorporando una restricción que vela por que las soluciones encontradas correspondan a la preferencia del decisor definida en el SID.

La implementación de la propuesta y los resultados que dan respuesta al cuarto y quinto objetivo específico de este trabajo se expondrán en los siguientes apartados.

3. Implementación de la propuesta

La propuesta desarrollada en este estudio, la cual se denomina **Algoritmo Genético Multiobjetivo con Sistema de Inferencia de Preferencias Difusas (AGMOSIPD)** sigue el diagrama de proceso que se presenta en la Figura 3-1:

Figura 3-1: Diagrama de flujo de AGMOSIPD.



Fuente: elaboración propia

3.1 Definición del Problema de Optimización Multiobjetivo

Para implementar y probar los resultados de la propuesta presentada en este trabajo se propone utilizar 6 problemas de prueba donde se minimizan 2 objetivos. En la Tabla 3-1 y la Tabla 3-2 se presentan las expresiones matemáticas de cada uno de ellos.

Tabla 3-1: Funciones de prueba.

Nro.	Nombre Problema	Variables de Decisión	Función Objetivo 1	Función Objetivo 2	Restricciones	Forma FP	Fuente
1	N/A	θ_i $i = 1, 2$	$x_1(\theta) = \theta_1$	$x_2(\theta) = (\theta_2 + 1)/\theta_1$	$0.1 \leq \theta_1 \leq 1$ $0 \leq \theta_2 \leq 5$	FP convexa	(Ramakrishnan & Hasan, 2013)
2	ZDT1	θ_i $i = 1, \dots, 30$	$x_1(\theta) = \theta_1$	$x_2(\theta) = \rho(\theta)\beta(x_1(\theta), \rho(\theta))$	$0 \leq \theta_i \leq 1$	FP convexa	(Zitzler et al., 2000)
3	ZDT2					FP no convexa	
4	ZDT3					FP discontinua	
5	ZDT4	θ_i $i = 1, \dots, 10$	$x_1(\theta) = \theta_1$	$x_2(\theta) = \rho(\theta)\beta(x_1(\theta), \rho(\theta))$	$0 \leq \theta_1 \leq 1$ $-10 \leq \theta_i \leq 10$ $i = 2, \dots, 10$	FP convexa	(Zitzler et al., 2000)
6	ZDT6					FP no convexa	
			$x_1(\theta) = 1 - e^{(-4\theta_1)} \sin^6(6\pi\theta_1)$		$0 \leq \theta_i \leq 1$		

Fuente: elaboración propia inspirada en Ramakrishnan & Hasan (2013) y Zitzler et al. (2000)

Tabla 3-2: Funciones del conjunto de problemas ZDT.

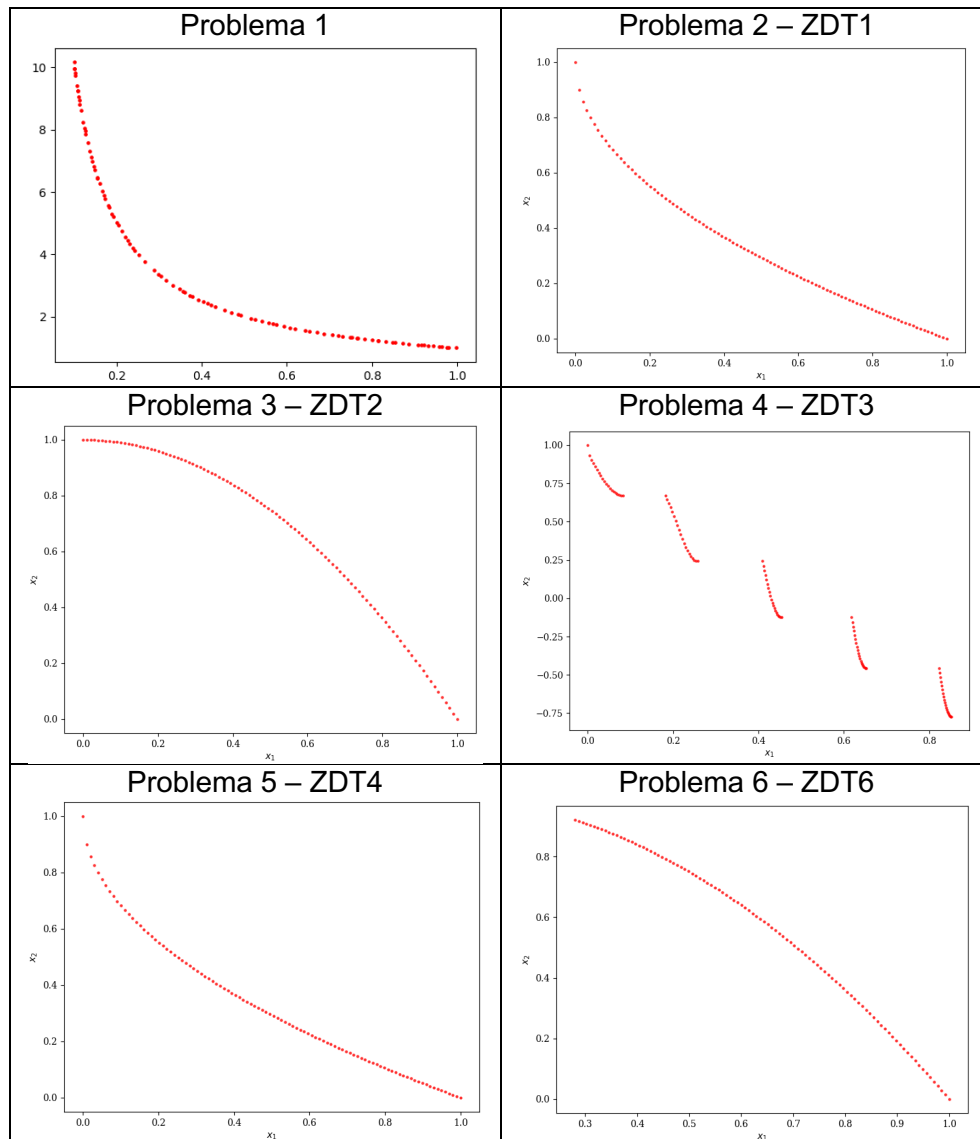
Nombre Problema	$\rho(\theta)$	$\beta(x_1(\theta), \rho(\theta))$
ZDT1	$\rho(\theta) = 1 + \frac{9}{n-1} \sum_{i=2}^n \theta_i$	$\beta(x_1(\theta), \rho(\theta)) = 1 - \sqrt{\frac{x_1(\theta)}{\rho(\theta)}}$
ZDT2	$\rho(\theta) = 1 + \frac{9}{n-1} \sum_{i=2}^n \theta_i$	$\beta(x_1(\theta), \rho(\theta)) = 1 - \left(\frac{x_1(\theta)}{\rho(\theta)}\right)^2$
ZDT3	$\rho(\theta) = 1 + \frac{9}{n-1} \sum_{i=2}^n \theta_i$	$\beta(x_1(\theta), \rho(\theta)) = 1 - \sqrt{\frac{x_1(\theta)}{\rho(\theta)} - \frac{x_1(\theta)}{\rho(\theta)} - \sin(10\pi x_1(\theta))}$
ZDT4	$\rho(\theta) = 1 + 10(n-1) + \sum_{i=2}^n (\theta_i^2 - 10 \cos(4\pi\theta_i))$	$\beta(x_1(\theta), \rho(\theta)) = 1 - \sqrt{\frac{x_1(\theta)}{\rho(\theta)}}$
ZDT6	$\rho(\theta) = 1 + 9 \left[\frac{\sum_{i=2}^n \theta_i}{9} \right]^{0.25}$	$\beta(x_1(\theta), \rho(\theta)) = 1 - \left(\frac{x_1(\theta)}{\rho(\theta)}\right)^2$

Fuente: elaboración propia inspirada en Zitzler et al. (2000)

El problema 1 fue abordado por Ramakrishnan & Hasan (2013) y fue inspirado en las funciones de prueba que ha investigado Kalyanmoy Deb. En dicho trabajo se incorporan preferencias en un MOEA mediante un SID y el resultado que ofrece es una solución discreta para elegir entre 4 computadores. Este problema se toma como referencia con el fin de implementar un SID que ofrezca una solución continua.

Los problemas 2 al 6 hacen parte del conjunto de funciones de prueba llamado ZDT (Zitzler – Deb – Thiele) en honor a sus creadores Eckart Zitzler, Kalyanmoy Deb y Lothar Thiele (Zitzler et al., 2000). Los problemas que se utilizarán son: ZDT1, ZDT2, ZDT3, ZDT4 y ZDT6. El problema ZDT5 no se abordará dado que está diseñado para variables discretas, y el presente trabajo delimita su alcance al ámbito continuo.

Figura 3-2: Representación gráfica de la FP de los problemas de prueba.



Fuente: elaboración propia

Cada uno de los problemas propuestos representa un POM de 2 objetivos que requieren ser minimizados y cuentan con una FP conocida. Estos problemas de prueba estudian

diferentes formas e inconvenientes de las FP estudiadas: algunas son convexas, no convexas, discontinuas, carecen de uniformidad en la densidad de soluciones a lo largo de la FP o incluso presentan muchas soluciones óptimas en ciertas regiones lo cual hace que un MOEA pueda converger rápidamente en un óptimo local. Estas funciones de prueba han sido ampliamente utilizadas en estudios para verificar la eficiencia de nuevos MOEA. En la Figura 3-2 se muestran las FP de los problemas de prueba que se abordarán en el presente estudio.

Para el uso de estas funciones se hace uso del lenguaje de programación de alto nivel Python. Específicamente para los problemas ZDT propuestos se empleó la librería “Pymoo” la cual fue desarrollada por (Blank & Deb, 2020). Dada la simpleza del Problema 1, éste fue codificado en Python por el autor del presente estudio.

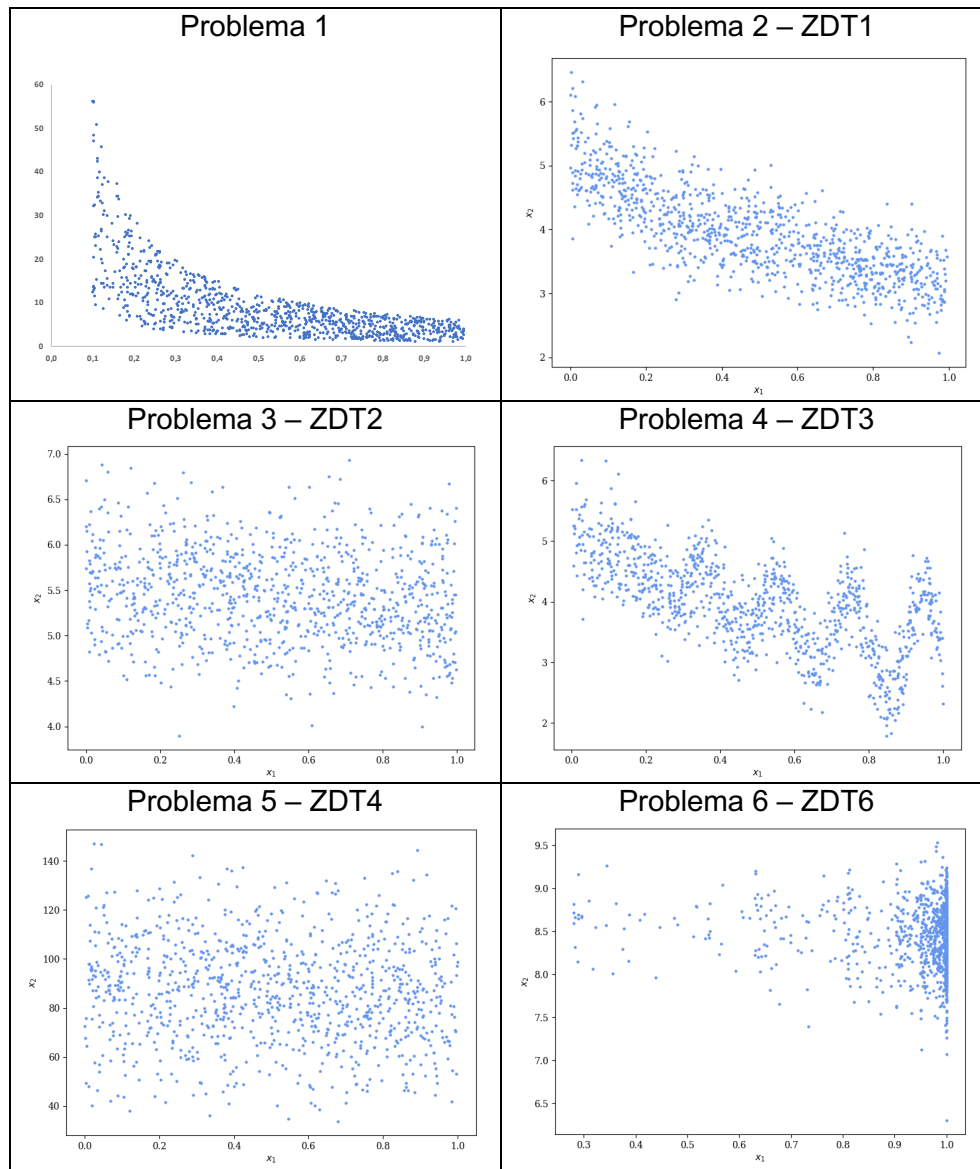
3.2 Generación de Soluciones mediante Simulación Monte Carlo

Para la ejecución de esta simulación se generaron z números aleatorios con distribución uniforme ($z = 1000$). La simulación se realizó mediante el uso de Python, y en la Figura 3-3 se presentan las gráficas de las simulaciones en el espacio de los objetivos x_1 y x_2 .

3.3 Construcción de la Estructura de Preferencias mediante un SID

Para todos los problemas se modelaron las preferencias mediante un SID de 2 entradas y 1 salida. La modelación que se presenta en este punto fue realizada en Python por medio de la librería “skfuzzy” para las variables difusas y “skfuzzy.control” para el SID.

En los próximos apartados se presentan la propuesta de diseño de las variables de entrada, la variable de salida y la base de reglas de la función de razonamiento aproximado.

Figura 3-3: Representación gráfica de las soluciones simuladas.

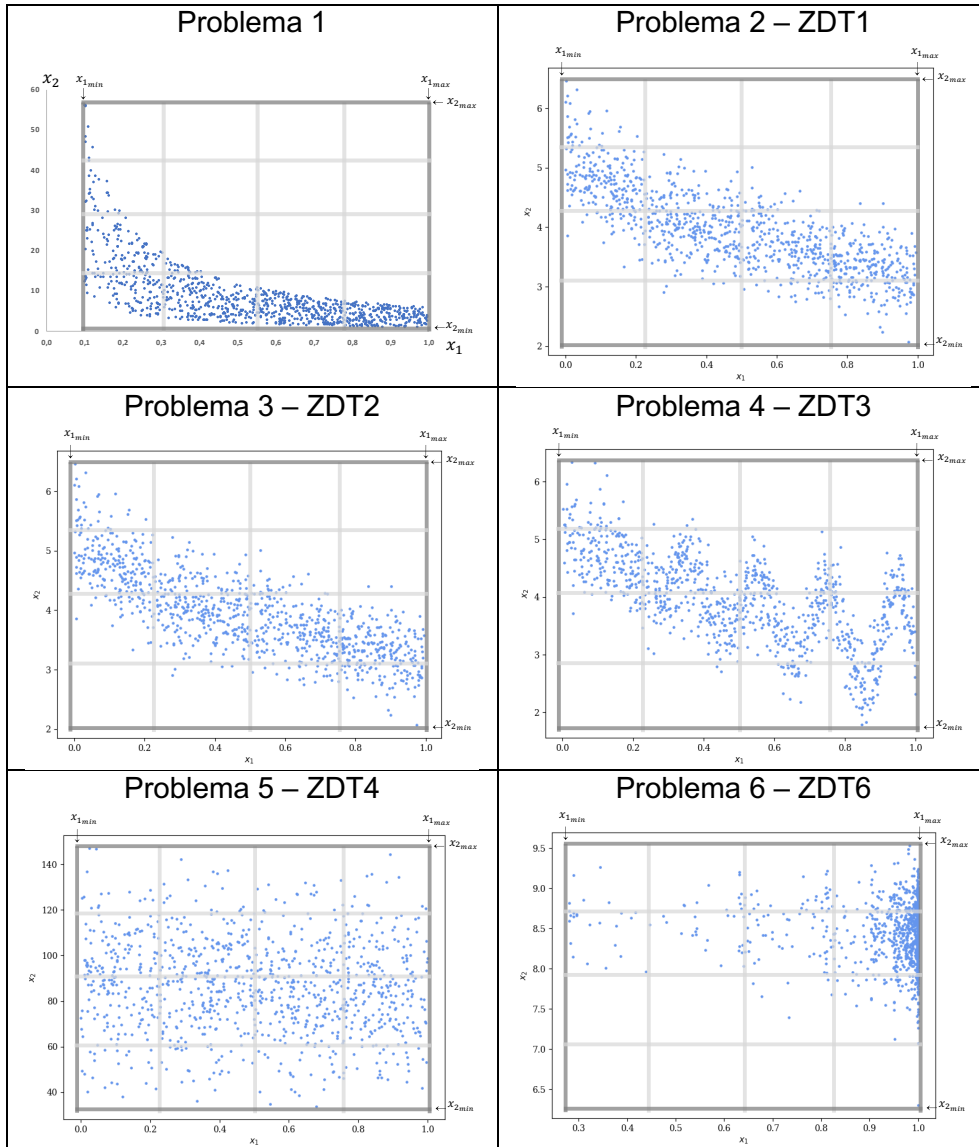
Fuente: elaboración propia

3.3.1 Diseño de las Variables de Entrada

Las VL de entrada se diseñan tomando como base el comportamiento que toma cada objetivo en las soluciones simuladas. La visualización gráfica le debe brindar al decisor un apoyo para construir sus percepciones de satisfacción sobre cada objetivo. Por ejemplo, las soluciones simuladas del problema 1 brindan una idea del comportamiento de los objetivos, el cual se muestra en la Figura 3-4, donde se proponen 4 posibles subdivisiones

para los objetivos, que dividen el espacio de soluciones en una matriz 4 x 4. Estas 4 subdivisiones se utilizaron como punto de partida para definir etiquetas lingüísticas para las variables de entrada. Otro decisor, con otro análisis, podría proponer una subdivisión distinta.

Figura 3-4: Representación gráfica para identificar zonas de interés.



Fuente: elaboración propia

En la Tabla 3-3 se presentan los elementos definidos para las VL de entrada, y en la Figura 3-5 se presenta su representación gráfica. Los valores Δ , a , b , c , d , e , f que se indican en

dicha Figura se definen mediante la siguiente expresión (con $r = 4$ que corresponde a la cantidad de términos lingüísticos de cada VL de entrada):

$$\Delta = \frac{|x_{i_{max}} - x_{i_{min}}|}{2r - 1}$$

$$a = x_{i_{min}} + 1\Delta = x_{i_{max}} - 1\Delta$$

$$b = x_{i_{min}} + 2\Delta = x_{i_{max}} - 2\Delta$$

$$c = x_{i_{min}} + 3\Delta = x_{i_{max}} - 3\Delta$$

$$d = x_{i_{min}} + 4\Delta = x_{i_{max}} - 4\Delta$$

$$e = x_{i_{min}} + 5\Delta = x_{i_{max}} - 5\Delta$$

$$f = x_{i_{min}} + 6\Delta = x_{i_{max}} - 6\Delta$$

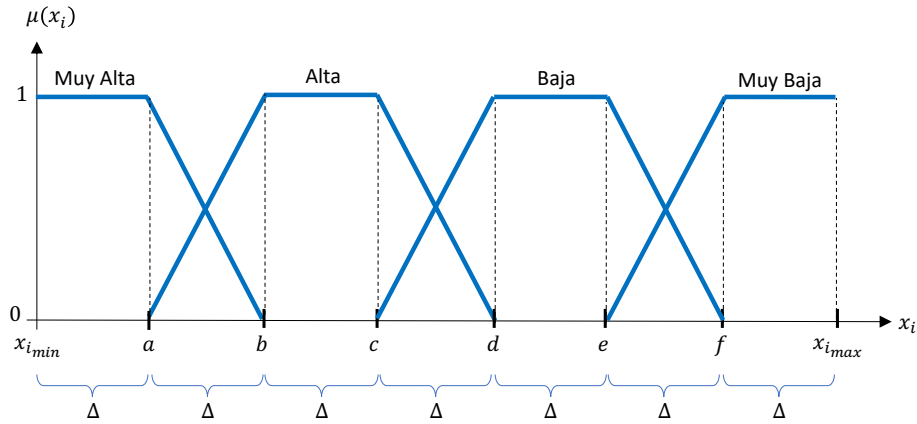
Tabla 3-3: Definición de VL de entrada del SID.

Elemento	VL del Objetivo 1: X_1	VL del Objetivo 2: X_2
Nombre	Satisfacción del decisor en el objetivo 1	Satisfacción del decisor en el objetivo 2
Términos lingüísticos	MA: Muy Alta A: Alta B: Baja MB: Muy Baja	MA: Muy Alta A: Alta B: Baja MB: Muy Baja
Universo de discurso	$U_{x_1} \in [x_{1_{min}}, x_{1_{max}}]$	$U_{x_2} \in [x_{2_{min}}, x_{2_{max}}]$
Función de pertenencia	Funciones trapezoidales	Funciones trapezoidales

Fuente: elaboración propia

La expresión matemática de las funciones de pertenencia de cada término lingüístico de las VL de entrada se presenta en la Tabla 3-4.

Figura 3-5: Representación gráfica de las VL de entrada del SID.



Fuente: elaboración propia

Tabla 3-4: Funciones de pertenencia correspondientes a la Figura 3-5.

Función de pertenencia Valor lingüístico "Muy Alta"	Función de pertenencia Valor lingüístico "Alta"
$\mu_{MA}(x_i) = \begin{cases} 1, & x_{i_{min}} \leq x_i < a \\ \frac{b-x_i}{b-a}, & a \leq x_i < b \\ 0, & b \leq x_i \leq x_{i_{max}} \end{cases}$ $\mu_{MA}(x_i) = T(x_{i_{min}}, x_{i_{min}}, a, b)$	$\mu_A(x_i) = \begin{cases} 0, & x_{i_{min}} \leq x_i < a \\ \frac{x_i-a}{b-a}, & a \leq x_i < b \\ 1, & b \leq x_i < c \\ \frac{d-x_i}{d-c}, & c \leq x_i < d \\ 0, & d \leq x_i \leq x_{i_{max}} \end{cases}$ $\mu_A(x_i) = T(a, b, c, d)$
Función de pertenencia Valor lingüístico "Baja"	Función de pertenencia Valor lingüístico "Muy Baja"
$\mu_B(x_i) = \begin{cases} 0, & x_{i_{min}} \leq x_i < c \\ \frac{x_i-c}{d-c}, & c \leq x_i < d \\ 1, & d \leq x_i < e \\ \frac{f-x_i}{f-e}, & e \leq x_i < f \\ 0, & f \leq x_i \leq x_{i_{max}} \end{cases}$ $\mu_B(x_i) = T(c, d, e, f)$	$\mu_{MB}(x_i) = \begin{cases} 0, & x_{i_{min}} \leq x_i < e \\ \frac{x_i-e}{f-e}, & e \leq x_i < f \\ 1, & f \leq x_i \leq x_{i_{max}} \end{cases}$ $\mu_{MB}(x_i) = T(e, f, x_{i_{max}}, x_{i_{max}})$

Fuente: elaboración propia

3.3.2 Diseño de la Variable de Salida

La VL de salida se define tomando como base el comportamiento simultáneo de los objetivos en las soluciones simuladas. La visualización gráfica le debe brindar al decisor un apoyo para construir sus percepciones sobre esas zonas donde podría tener una mayor

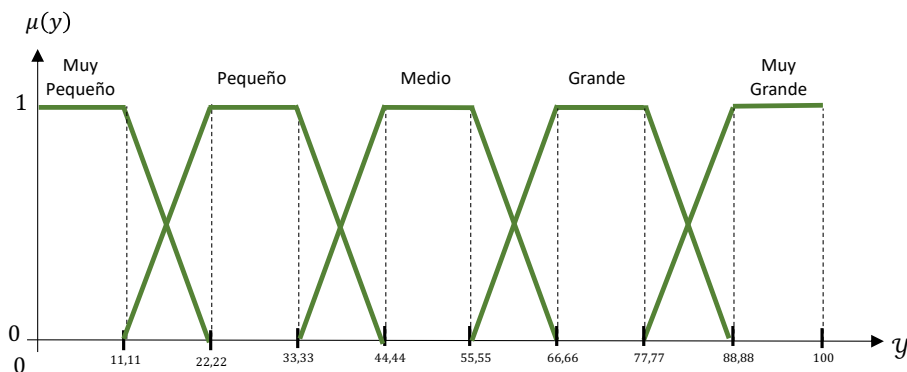
preferencia sobre ciertas soluciones. En la Tabla 3-5 se presentan los elementos definidos para la VL de salida y en la Figura 3-6 se presenta su representación gráfica.

Tabla 3-5: Definición de VL de salida del SID.

Elemento	VL de Salida: y
Nombre	Disposición por mejorar la obtención del objetivo x_1 con respecto a sacrificar la obtención del objetivo x_2 .
Términos lingüísticos	MP: Muy Pequeña P: Pequeña M: Media G: Grande MG: Muy Grande
Universo de discurso	$U_y \in [0,100]$
Función de pertenencia	Funciones trapezoidales

Fuente: elaboración propia

Figura 3-6: Representación gráfica de la VL de salida del SID.



Fuente: elaboración propia

La expresión matemática de las funciones de pertenencia de cada término lingüístico de la VL de salida se presenta en la Tabla 3-6.

3.3.3 Diseño de la Base de Reglas de la Función de Razonamiento Aproximado

A partir de las variables difusas definidas: x_1, x_2, y , se diseña un conjunto de reglas de tal forma que, de acuerdo con la satisfacción que el decisor experimente en las variables x_i

se le pida establecer su disposición al sacrificio de perder en un objetivo para ganar en otro. Esta relación entre variables permitirá el funcionamiento de la función de razonamiento aproximado del SID.

Tabla 3-6: Funciones de pertenencia correspondientes a la Figura 3-6.

Función de pertenencia Valor lingüístico “Muy Pequeño”	Función de pertenencia Valor lingüístico “Pequeño”
$\mu_{MP}(y) = \begin{cases} 1, & 0 \leq y < 11,11 \\ \frac{22,22 - y}{11,11}, & 11,11 \leq y < 22,22 \\ 0, & 22,22 \leq y \leq 100 \end{cases}$ $\mu_{MP}(y) = T(0,0,11,11,22,22)$	$\mu_P(y) = \begin{cases} 0, & 0 \leq y < 11,11 \\ \frac{y - 11,11}{11,11}, & 11,11 \leq y < 22,22 \\ 1, & 22,22 \leq y < 33,33 \\ \frac{44,44 - y}{11,11}, & 33,33 \leq y < 44,44 \\ 0, & 44,44 \leq y \leq 100 \end{cases}$ $\mu_P(y) = T(11,11,22,22,33,33,44,44)$
Función de pertenencia Valor lingüístico “Medio”	Función de pertenencia Valor lingüístico “Grande”
$\mu_M(y) = \begin{cases} 0, & 0 \leq y < 33,33 \\ \frac{y - 33,33}{11,11}, & 33,33 \leq y < 44,44 \\ 1, & 44,44 \leq y < 55,55 \\ \frac{66,66 - y}{11,11}, & 55,55 \leq y < 66,66 \\ 0, & 66,66 \leq y \leq 100 \end{cases}$ $\mu_M(y) = T(33,33,44,44,55,55,66,66)$	$\mu_G(y) = \begin{cases} 0, & 0 \leq y < 55,55 \\ \frac{y - 55,55}{11,11}, & 55,55 \leq y < 66,66 \\ 1, & 66,66 \leq y < 77,77 \\ \frac{88,88 - y}{11,11}, & 77,77 \leq y < 88,88 \\ 0, & 88,88 \leq y \leq 100 \end{cases}$ $\mu_G(y) = T(55,55,66,66,77,77,88,88)$
Función de pertenencia Valor lingüístico “Muy Grande”	
$\mu_{MG}(y) = \begin{cases} 0, & 0 \leq y < 77,77 \\ \frac{y - 77,77}{11,11}, & 77,77 \leq y < 88,88 \\ 1, & 88,88 \leq y \leq 100 \end{cases}$ $\mu_{MG}(y) = T(77,77,88,88,100,100)$	

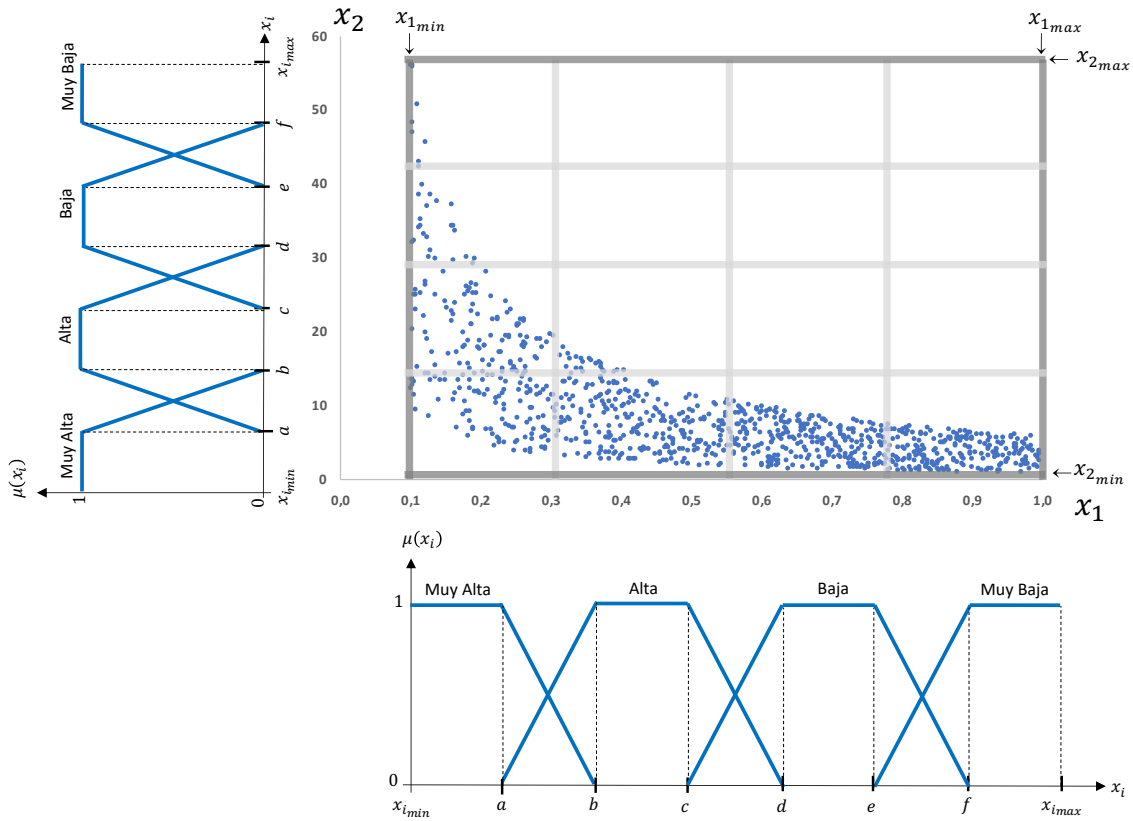
Fuente: elaboración propia

Al momento de definir el SID debe ocurrir que:

- Las reglas deben cubrir todo el espacio de las variables de entrada.
- El conjunto de reglas debe ser coherente.

Una forma de relacionar las variables de entrada consiste en aprovechar el comportamiento de las soluciones simuladas y, para el caso del Problema 1 utilizar la matriz 4x4 de la Figura 3-4, para compararla con los CD de las variables de entrada. Esta comparación se presenta en la Figura 3-7.

Figura 3-7: Relación entre los CD de las VL de entrada y la simulación.



Fuente: elaboración propia

Para relacionar las variables que integran el SID en forma de reglas, se debe buscar que ellas describan las preferencias del decisor. Para poblar las reglas se propone utilizar como referencia el análisis de las tasas de intercambio entre ambos objetivos.

Una tasa de intercambio, para una curva de soluciones como la que se simula para el problema 1, se interpreta como la pendiente de la recta tangente en cada punto de la curva que contiene soluciones óptimas, es decir de la FP. Dicha pendiente podemos denotarla como \emptyset , donde $\emptyset \in (-\infty, 0)$.

$$\emptyset = \frac{\Delta x_2}{\Delta x_1}$$

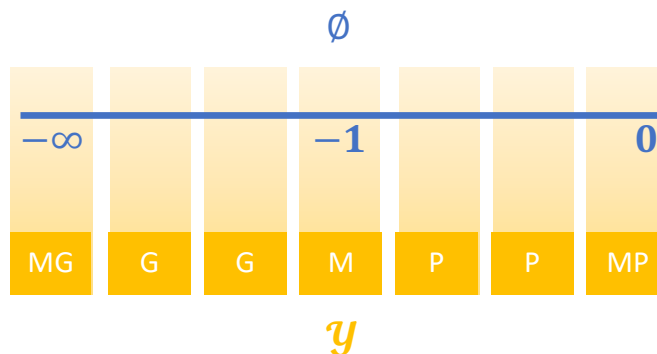
- Si $\emptyset \rightarrow 0$, quiere decir que pequeñas variaciones en el objetivo 2 producirán grandes variaciones en el objetivo 1.

- Si $\emptyset \rightarrow -1$, quiere decir que ante una variación del objetivo 1 se produce una variación muy similar (o igual) en el objetivo 2.
- Si $\emptyset \rightarrow -\infty$, quiere decir que pequeñas variaciones en el objetivo 1 producirán grandes variaciones en el objetivo 2.

Cada valor de \emptyset representa una tasa de intercambio distinta y el decisor podría sentir preferencia por alguna tasa en particular. Para el decisor es muy difícil expresar de forma precisa la tasa de intercambio preferida, por tanto, convendría aprovechar el SID para que este refleje zonas difusas de tasas de intercambio donde el decisor exprese las zonas donde se encuentren sus tasas preferidas.

En la Figura 3-8 se ilustra una propuesta para relacionar el comportamiento de la tasa de intercambio \emptyset (expresada en color azul) con la variable γ (cuyas etiquetas se expresan en color amarillo). Allí se puede apreciar que, cuando la tasa de intercambio tiene un comportamiento cercano a cero, la etiqueta lingüística de salida correspondiente sería MP (que equivale a Muy Pequeño) y a medida que el valor de \emptyset va siendo más negativo el valor de γ va aumentando hasta llegar a la etiqueta MG (que equivale a "Muy Grande").

Figura 3-8: Comparación de \emptyset con etiquetas de γ .



MG: "Muy Grande", G: "Grande", M: "Medio", P: "Pequeño" y MP: "Muy Pequeño".

Fuente: elaboración propia

En la Tabla 3-7 se resaltan en amarillo 7 sectores que representan zonas de tasas de intercambio para una eventual FP como se ilustra en la Figura 3-7. Los sectores blancos podrían recibir el valor de γ de forma análoga representando el comportamiento de

posibles tasas de intercambio para una FP en dicha región. En la Tabla 3-8 y la Tabla 3-9 se ilustran propuestas para esta asignación del valor de y .

Tabla 3-7: Reglas del SID que representa preferencias – Parte I.

x_2: Satisfacción Objetivo 2	Muy Baja	MG			
	Baja	G			
	Alta	G			
	Muy Alta	M	P	P	MP
y: Disposición de mejorar x_1 con respecto a x_2	Muy Alta	Alta	Baja	Muy Baja	
		x_1: Satisfacción Objetivo 1			

Fuente: elaboración propia

Tabla 3-8: Reglas del SID que representa preferencias – Parte II.

x_2: Satisfacción Objetivo 2	Muy Baja	MG	MG		
	Baja	G	G		
	Alta	G	M	P	MP
	Muy Alta	M	P	P	MP
y: Disposición de mejorar x_1 con respecto a x_2	Muy Alta	Alta	Baja	Muy Baja	
		x_1: Satisfacción Objetivo 1			

Fuente: elaboración propia

Tabla 3-9: Reglas del SID que representa preferencias – Parte III.

x_2: Satisfacción Objetivo 2	Muy Baja	MG	MG	G	M
	Baja	G	G	M	P
	Alta	G	M	P	MP
	Muy Alta	M	P	P	MP
y: Disposición de mejorar x_1 con respecto a x_2	Muy Alta	Alta	Baja	Muy Baja	
		x_1: Satisfacción Objetivo 1			

Fuente: elaboración propia

A manera de ejemplo, para indicar la forma como se deben leer las tablas anteriores se presentan algunas reglas:

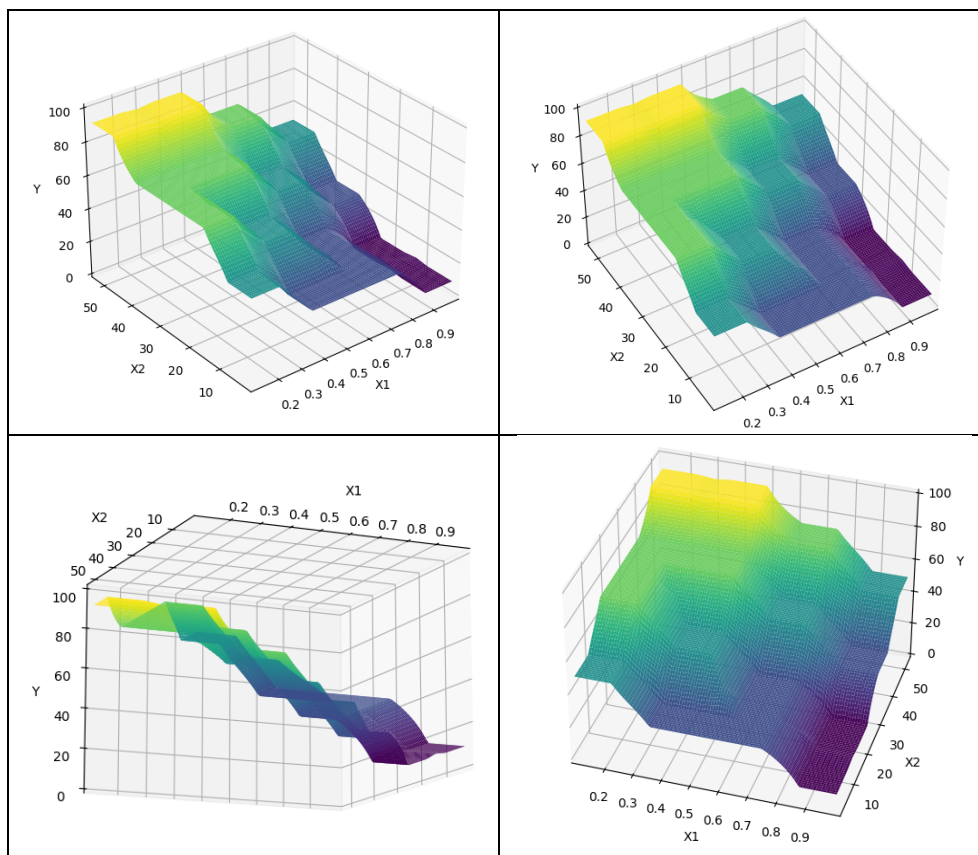
$$\text{Si } x_1 = \text{“Muy Alta”} \wedge x_2 = \text{“Muy Alta”} \Rightarrow y = \text{“Medio”}$$

$$\text{Si } x_1 = \text{“Baja”} \wedge x_2 = \text{“Muy Baja”} \Rightarrow y = \text{“Grande”}$$

La representación matricial de la Tabla 3-9 coincide con la cuadrícula formada en la Figura 3-7, por tanto, puede aprovecharse el uso de esta última para que, a partir de la información visual, el decisor pueda definir sus preferencias al asignar a cada zona del espacio de soluciones una etiqueta lingüística de la variable de salida y . En conclusión, esta matriz expresa las preferencias del decisor ante el comportamiento simultáneo de ambos objetivos.

Es posible obtener una gráfica (superficie) aproximada de la función $F(x_1, x_2) = y$, la cual se presenta en la Figura 3-9. Si bien cada uno de los problemas de prueba tiene un SID distinto, debido a que los universos del discurso de la variable de entrada son diferentes, la variable de salida y , por ende, la expresión gráfica de la superficie del SID se propone que sean iguales para todos ellos.

Figura 3-9: Representación gráfica de la superficie de reglas del SID.



Fuente: elaboración propia

3.4 Evaluación de las Soluciones simuladas en el SID

Posterior a la construcción de la estructura de preferencias, se deben evaluar las soluciones de los 6 problemas en su correspondiente SID. Cuando se evalúa una solución de la forma (x_1, x_2) , el SID proyecta estos valores en los términos lingüísticos de las VL de entrada $\mathcal{X}_1, \mathcal{X}_2$ mediante el uso de aritmética difusa. Finalmente, estos valores se computan en la base de reglas y se obtiene un valor numérico y , que puede ser asociado a un término lingüístico de la VL de salida \mathcal{Y} .

3.5 Selección de la Población Semilla

Luego de la evaluación de las soluciones simuladas en el SID, el decisor debe elegir el valor de la variable \mathcal{Y} que prefiere y que representa la zona de intercambio más representativa para él. La población semilla se definirá como $w=100$, y para poblarla se seleccionarán las w soluciones de la simulación que cuyo valor de \mathcal{Y} sea el elegido por el decisor.

La estrategia propuesta para poblar la semilla será la siguiente: Si se supone que el valor elegido por el decisor es $\mathcal{Y} = \text{Muy Pequeño}$ y si se establece que φ corresponde a la cantidad de soluciones simuladas donde $\mathcal{Y} = \text{Muy Pequeño}$, entonces la semilla se poblará así:

- Si $w = \varphi$, entonces la semilla estará compuesta por todas las φ soluciones.
- Si $w < \varphi$, entonces la semilla contendrá w soluciones donde $\mathcal{Y} = \text{Muy Pequeño}$ elegidas al azar.
- Si $w > \varphi$, entonces la semilla contendrá todas las soluciones donde $\mathcal{Y} = \text{Muy Pequeño}$ y la diferencia $w - \varphi$ que falta será poblada con otro subconjunto de soluciones para otro valor de \mathcal{Y} .

Si el valor preferido de \mathcal{Y} es Muy Pequeño y las soluciones simuladas de este valor de \mathcal{Y} no alcanzan a poblar la semilla, entonces se utilizará el orden de preferencia que se muestra en la Tabla 3-10:

Tabla 3-10: Orden de preferencia de valores de \mathcal{Y} para poblar la semilla.

Si \mathcal{y} preferido es:	El orden de prioridad de preferencias en caso de que \mathcal{y} preferido no alcance a poblar la semilla es:
Muy Pequeño	Pequeño, Medio, Grande y Muy Grande
Pequeño	Medio, Grande, Muy Grande y Muy Pequeño
Medio	Grande, Muy Grande, Muy Pequeño y Pequeño
Grande	Muy Grande, Muy Pequeño, Pequeño y Medio
Muy Grande	Muy Pequeño, Pequeño, Medio y Grande

Fuente: elaboración propia

3.6 Definición de Restricción Incorporando el SID

Con el fin de que AGMOSIPD pueda dedicar su procesamiento a la obtención de soluciones preferidas y no desperdicie sus recursos en la obtención de puntos de la FP que no son preferidos, se ha incorporado una restricción que permitirá que el algoritmo se dirija hacia zonas de soluciones deseables, utilizando el término lingüístico más preferido en la variable de salida \mathcal{Y} e introduciendo el SID en dicha restricción.

Siendo el SID representado por la expresión $F(\mathcal{X}_1, \mathcal{X}_2) = \mathcal{Y}$, la formulación del problema para los 6 casos de prueba es la siguiente:

$$\text{Minimizar } x_m(\theta), \quad m = 1, 2;$$

Sujeto a:

$$\theta_i^{(L)} \leq \theta_i \leq \theta_i^{(U)}, \quad i = 1, 2, \dots, n$$

$$g(\theta) = a \sum_{i=1}^n \theta_i \leq 0$$

$$\text{Donde } a = \begin{cases} 0, & \mathcal{Y} = \psi \\ 1, & \mathcal{Y} \neq \psi \end{cases}$$

ψ = Etiqueta preferida por el decisor

$\mathcal{Y} = F(\mathcal{X}_1, \mathcal{X}_2)$ es un Sistema de Inferencia Difusa

$\psi \in \mathcal{Y}$

$n = 2, 10$ o 30 según el caso

Durante cada generación debe suceder lo siguiente:

- **En la generación 1:** Los universos del discurso de las variables de entrada del SID están acotados por los valores: x_{1min} , x_{1max} , x_{2min} y x_{2max} . Estos valores surgen de las z soluciones de la Simulación Montecarlo x_{1min}^S , x_{1max}^S , x_{2min}^S y x_{2max}^S . Por tanto se indica que:

$$x_{i_{min}} = \min (x_{i_{min}}^S, x_{i_{min}}^g)$$

$$x_{i_{max}} = \max (x_{i_{max}}^S, x_{i_{max}}^g)$$

$x_{i_{min}}^S$: Valor mínimo del objetivo x_i en la simulación.

$x_{i_{max}}^S$: Valor máximo del objetivo x_i en la simulación.

$x_{i_{min}}^g$: Valor mínimo del objetivo x_i en la generación actual.

$x_{i_{max}}^g$: Valor máximo del objetivo x_i en la generación actual.

- **En las siguientes generaciones:** Los valores x_{1min} , x_{1max} , x_{2min} y x_{2max} se actualizan con los valores mínimos y máximos que el proceso de optimización va encontrando x_{1min}^g , x_{1max}^g , x_{2min}^g y x_{2max}^g . La actualización sigue las siguientes instrucciones:

$$x_{i_{min}} = \min (x_{i_{min}}^{g-1}, x_{i_{min}}^g)$$

$$x_{i_{max}} = \max (x_{i_{max}}^{g-1}, x_{i_{max}}^g)$$

$x_{i_{min}}^g$: Valor mínimo del objetivo x_i en la generación actual.

$x_{i_{max}}^g$: Valor máximo del objetivo x_i en la generación actual.

$x_{i_{min}}^{g-1}$: Valor mínimo del objetivo x_i en la generación anterior.

$x_{i_{max}}^{g-1}$: Valor máximo del objetivo x_i en la generación anterior.

Esta actualización de valores en cada generación permite que el SID amplíe los universos del discurso de sus variables de entrada. Siendo así, se promueve que el mismo pueda evaluar cualquier solución que surja dentro del algoritmo evolutivo y que, además, la estructura de preferencias del decisor abarque el máximo espacio de soluciones detectado en la optimización.

Se propone que al finalizar la ejecución del proceso de búsqueda evolutivo AGMOSIPD se obtengan los siguientes puntos:

- **Cota ideal:** En un problema de minimización, corresponde al vector de valores x , que contiene el mínimo valor alcanzado en AGMOSIPD para cada objetivo. Este punto se considera relevante porque se asume que el decisor preferiría obtener una solución como esta, donde se obtiene lo mejor de ambos objetivos, además corresponde al límite inferior de las variables de entrada del SID.
- **Cota nadir:** En un problema de minimización, corresponde al vector de valores x , que contiene el máximo valor alcanzado en AGMOSIPD para cada objetivo. Este punto corresponde al límite superior de las variables de entrada del SID, y se espera que el decisor quiera alejarse de este punto, el cual es dominado por soluciones óptimas.

Los anteriores puntos de referencia serán aprovechados en la parte final del algoritmo para elegir una única solución final.

3.7 Definición de Operadores Genéticos y Ejecución del Algoritmo

Para realizar el proceso de optimización con los algoritmos NSGA-II y NSGA-III se utilizará la librería Pymoo de Python. Previo a la optimización se definieron los siguientes operadores genéticos:

- **Operador de selección:** Este operador es el definido por NSGA-II (dominancia de Pareto y distancia de aglomeración) y NSGA-III (dominancia de Pareto y distancia a puntos de referencia). El proceso evolutivo comenzará con la semilla definida para la población inicial.
- **Operador de cruzamiento:** Esta instrucción es invocada mediante la instrucción “SBX” (Simulated Binary Crossover) que cruza un par de individuos que representan números continuos.
- **Operador de mutación:** Este operador se define mediante el comando “PM” (Polynomial Mutation), el cual evalúa la probabilidad de que un gen de la cadena genética sea mutado.

3.8 Obtención del Segmento de la FP y Elección de Solución Óptima

Posterior a la ejecución del proceso de optimización, se incorpora en AGMOSIPD un procedimiento que permita elegir una o varias soluciones del segmento de la FP encontrado. En este trabajo se propone calcular las siguientes métricas:

- **Distancia a la cota ideal:** Tomando como referencia la cota ideal, encontrada en AGMOSIPD, se propone calcular la distancia euclidiana de dicha cota a cada solución encontrada en la optimización. Este valor se calcula así:

V : Cantidad de soluciones encontrada en AGMOSIPD

$x_{m,v}^*$: Solución en el objetivo m encontrada en AGMOSIPD donde $v = 1, \dots, V$

x_m^{CI} : Valor de cota ideal del objetivo m

DCI_v : Distancia de la solución v encontrada en AGMOSIPD con respecto a la cota ideal

$$DCI_v = \sqrt{(x_1^{CI} - x_{1,v}^*)^2 + (x_2^{CI} - x_{2,v}^*)^2}$$

- **Distancia a la cota nadir:** Tomando como referencia la cota nadir, encontrada en AGMOSIPD, se propone calcular la distancia euclidiana de dicha cota a cada solución encontrada en la optimización. Este valor se calcula así:

x_m^{CN} : Valor de cota nadir del objetivo m

DCN_v : Distancia de la solución v encontrada en AGMOSIPD con respecto a la cota nadir

$$DCN_v = \sqrt{(x_1^{CN} - x_{1,v}^*)^2 + (x_2^{CN} - x_{2,v}^*)^2}$$

- **Solución preferida ideal:** Corresponde a la solución obtenida en AGMOSIPD con menor DCI. Se representa x_m^{CI*} .
- **Solución preferida nadir:** Corresponde a la solución obtenida en AGMOSIPD con mayor DCN. Se representa x_m^{CN*} .
- **Punto medio:** Se define como x_m^{PM} .

$$x_m^{PM} = \frac{x_m^{CI*} + x_m^{CN*}}{2}$$

- **Distancia al punto medio:** Tomando como referencia el punto medio, se propone calcular la distancia euclidiana de dicho punto a cada solución encontrada en la optimización. Este valor se calcula así:

DPM_v: Distancia de la solución *v* encontrada en AGMOSIPD con respecto al punto medio.

$$DPM_v = \sqrt{(x_1^{PM} - x_{1,v}^*)^2 + (x_2^{PM} - x_{2,v}^*)^2}$$

- **Solución preferida punto medio:** Corresponde a la solución obtenida en AGMOSIPD con menor DPM. Se representa x_i^{PM*} .

Luego del cálculo de las métricas anteriores, AGMOSIPD entregará al decisor 3 soluciones: x_i^{CI*} , x_i^{CN*} y x_i^{PM*} . Con estas soluciones sugeridas el decisor podrá elegir aquella que más le resulte conveniente, ya sea la más cercana al ideal, la más lejana al nadir o un punto intermedio entre los dos anteriores.

Esta propuesta de identificación de la distancia a la cota ideal, nadir y punto medio se inspira en el método TOPSIS que aborda la elección de alternativas, en análisis multicriterio, mediante la identificación de soluciones que tienen mayor o menor cercanía o proximidad a puntos de referencia como los aquí descritos (Yoon & Kim, 2017).

Con la presentación de este procedimiento se sustenta la implementación de la metodología propuesta y planteada como propósito en el cuarto objetivo específico de este trabajo. La presentación de los resultados de la ejecución del algoritmo, la obtención del segmento de la frontera de Pareto que da cuenta de las preferencias del decisor y la validación de los hallazgos, se presentan en el siguiente apartado.

4. Resultados de la Implementación de la Propuesta

4.1 Premisas de Ejecución de la Propuesta

La propuesta diseñada en AGMOSIDP se ejecutó para los 6 problemas propuestos (Problema 1, ZDT1, ZDT2, ZDT3, ZDT4 y ZDT6), utilizando 5 perfiles de preferencias (Muy Pequeño, Pequeño, Medio, Grande y Muy Grande) en 2 algoritmos genéticos (NSGA-II y NSGA-III), lo cual representa un total de 60 ejecuciones.

Para cada problema y perfil de preferencia se realizó una simulación Monte Carlo, para un total de 30 simulaciones de 1000 soluciones, y en cada simulación se seleccionó una semilla de 100 soluciones que cumplen con el perfil de preferencia establecido. La misma semilla de cada problema y perfil de preferencia se utilizó al ejecutar el algoritmo NSGA-II y NSGA-III.

En la Tabla 4-1 se muestra un ejemplo de la población semilla seleccionada para el Problema 1 para un perfil de preferencias Muy Pequeño. Allí se muestra el valor de las variables de decisión, la evaluación en las funciones objetivos, el valor obtenido en el SID para la variable de salida y y su correspondiente etiqueta lingüística denominada Conjunto de y .

Los parámetros con los cuales se ejecutó el NSGA-II y NSGA-III en AGMOSIPD son:

- **Operador de Cruzamiento:** Se utilizó el operador de cruzamiento SBX (Simulated Binary Crossover) propuesto por la librería Pymoo de Python. Los parámetros de este operador son:
 - a. Probabilidad de cruzamiento: 0.7, representa la probabilidad de que un par de soluciones sean cruzadas.

- b. Parámetro de distribución eta: 15, controla la diversidad de la distribución binomial utilizada en el operador de cruzamiento. Mientras mayor sea este valor, mayor será la probabilidad de generar soluciones diversas. Para esta implementación se utiliza el valor estándar de 15 sugerido por Pymoo.

Tabla 4-1: Ejemplo de semilla para el problema 1 para $\gamma = \text{"Muy Pequeño"}$.

#	θ_1	θ_2	X1	X2	Y	Conjunto de Y	#	θ_1	θ_2	X1	X2	Y	Conjunto de Y
131	0,9121	0,7496	0,9121	1,9182	8,6477	Muy Pequeño	456	0,8810	0,9385	0,8810	2,2004	8,6477	Muy Pequeño
134	0,8775	1,1263	0,8775	2,4232	8,6477	Muy Pequeño	458	0,9733	2,1561	0,9733	3,2427	8,6477	Muy Pequeño
150	0,9562	3,8169	0,9562	5,0374	8,6477	Muy Pequeño	471	0,9108	0,7053	0,9108	1,8722	8,6477	Muy Pequeño
154	0,8876	1,8226	0,8876	3,1799	8,6477	Muy Pequeño	475	0,9683	2,1457	0,9683	3,2487	8,6477	Muy Pequeño
158	0,8774	1,9470	0,8774	3,3589	8,6477	Muy Pequeño	480	0,9262	1,2438	0,9262	2,4224	8,6477	Muy Pequeño
171	0,9237	2,2108	0,9237	3,4760	8,6477	Muy Pequeño	481	0,8727	1,6934	0,8727	3,0864	8,6477	Muy Pequeño
177	0,9167	4,6276	0,9167	6,1390	8,6477	Muy Pequeño	488	0,9176	0,9799	0,9176	2,1578	8,6477	Muy Pequeño
178	0,9225	1,4420	0,9225	2,6471	8,6477	Muy Pequeño	499	0,9092	4,1772	0,9092	5,6941	8,6477	Muy Pequeño
198	0,9583	3,5900	0,9583	4,7895	8,6477	Muy Pequeño	502	0,8887	3,4313	0,8887	4,9862	8,6477	Muy Pequeño
199	0,9725	4,8552	0,9725	6,0210	8,6477	Muy Pequeño	505	0,9086	1,2889	0,9086	2,5193	8,6477	Muy Pequeño
227	0,9458	3,9797	0,9458	5,2649	8,6477	Muy Pequeño	508	0,9517	3,9230	0,9517	5,1728	8,6477	Muy Pequeño
228	0,9461	0,5565	0,9461	1,6452	8,6477	Muy Pequeño	514	0,9930	2,0244	0,9930	3,0458	8,6477	Muy Pequeño
240	0,9631	3,5329	0,9631	4,7067	8,6477	Muy Pequeño	523	0,9039	0,2424	0,9039	1,3745	8,6477	Muy Pequeño
245	0,8757	2,7025	0,8757	4,2279	8,6477	Muy Pequeño	524	0,8824	0,7939	0,8824	2,0329	8,6477	Muy Pequeño
246	0,9265	4,2848	0,9265	5,7044	8,6477	Muy Pequeño	528	0,9661	4,9573	0,9661	6,1662	8,6477	Muy Pequeño
257	0,9931	2,7052	0,9931	3,7308	8,6477	Muy Pequeño	547	0,9275	4,2360	0,9275	5,6455	8,6477	Muy Pequeño
261	0,9587	3,2048	0,9587	4,3857	8,6477	Muy Pequeño	554	0,8831	0,7505	0,8831	1,9822	8,6477	Muy Pequeño
269	0,9143	1,6292	0,9143	2,8757	8,6477	Muy Pequeño	562	0,9703	2,3096	0,9703	3,4108	8,6477	Muy Pequeño
271	0,9193	0,7924	0,9193	1,9497	8,6477	Muy Pequeño	747	0,9036	3,9207	0,9036	5,4458	8,6477	Muy Pequeño
275	0,9377	1,6858	0,9377	2,8642	8,6477	Muy Pequeño	756	0,9072	3,6227	0,9072	5,0953	8,6477	Muy Pequeño
278	0,9958	2,9982	0,9958	4,0153	8,6477	Muy Pequeño	762	0,9077	4,9530	0,9077	6,5586	8,6477	Muy Pequeño
287	0,9049	0,7765	0,9049	1,9632	8,6477	Muy Pequeño	773	0,9035	2,6582	0,9035	4,0490	8,6477	Muy Pequeño
288	0,9955	0,7494	0,9955	1,7573	8,6477	Muy Pequeño	775	0,8826	3,0352	0,8826	4,5721	8,6477	Muy Pequeño
304	0,9496	2,3284	0,9496	3,5051	8,6477	Muy Pequeño	777	0,9502	4,0259	0,9502	5,2894	8,6477	Muy Pequeño
306	0,8982	3,6794	0,8982	5,2099	8,6477	Muy Pequeño	795	0,9670	2,5620	0,9670	3,6835	8,6477	Muy Pequeño
311	0,9269	4,8908	0,9269	6,3554	8,6477	Muy Pequeño	797	0,8809	4,0836	0,8809	5,7712	8,6477	Muy Pequeño
342	0,9185	3,6153	0,9185	5,0250	8,6477	Muy Pequeño	799	0,9041	2,6412	0,9041	4,0273	8,6477	Muy Pequeño
343	0,9032	0,4200	0,9032	1,5722	8,6477	Muy Pequeño	819	0,9002	1,0221	0,9002	2,2463	8,6477	Muy Pequeño
345	0,9418	3,1542	0,9418	4,4111	8,6477	Muy Pequeño	841	0,9470	3,3951	0,9470	4,6409	8,6477	Muy Pequeño
346	0,9142	0,3184	0,9142	1,4422	8,6477	Muy Pequeño	842	0,9688	1,7419	0,9688	2,8300	8,6477	Muy Pequeño
350	0,8772	3,2303	0,8772	4,8225	8,6477	Muy Pequeño	852	0,9854	0,6758	0,9854	1,7007	8,6477	Muy Pequeño
352	0,9376	3,7738	0,9376	5,0916	8,6477	Muy Pequeño	859	0,9936	3,0754	0,9936	4,1017	8,6477	Muy Pequeño
362	0,9610	4,6191	0,9610	5,8470	8,6477	Muy Pequeño	861	0,9554	4,0468	0,9554	5,2824	8,6477	Muy Pequeño
365	0,9717	0,2057	0,9717	1,2408	8,6477	Muy Pequeño	864	0,8944	2,0162	0,8944	3,3721	8,6477	Muy Pequeño
370	0,9340	3,7483	0,9340	5,0840	8,6477	Muy Pequeño	893	0,9800	3,0112	0,9800	4,0932	8,6477	Muy Pequeño
375	0,9017	3,5717	0,9017	5,0702	8,6477	Muy Pequeño	898	0,9034	0,2425	0,9034	1,3753	8,6477	Muy Pequeño
377	0,9301	0,3899	0,9301	1,4944	8,6477	Muy Pequeño	899	0,9012	0,3904	0,9012	1,5428	8,6477	Muy Pequeño
379	0,9814	1,3750	0,9814	2,4201	8,6477	Muy Pequeño	903	0,8926	0,0999	0,8926	1,2323	8,6477	Muy Pequeño
385	0,9637	2,1459	0,9637	3,2643	8,6477	Muy Pequeño	904	0,9011	0,2945	0,9011	1,4366	8,6477	Muy Pequeño
396	0,9301	3,0003	0,9301	4,3007	8,6477	Muy Pequeño	906	0,9225	4,1252	0,9225	5,5555	8,6477	Muy Pequeño
406	0,9699	3,3976	0,9699	4,5339	8,6477	Muy Pequeño	911	0,9408	3,8948	0,9408	5,2028	8,6477	Muy Pequeño
410	0,8979	4,4384	0,8979	6,0566	8,6477	Muy Pequeño	914	0,9400	2,9117	0,9400	4,1616	8,6477	Muy Pequeño
421	0,9628	0,9131	0,9628	1,9869	8,6477	Muy Pequeño	925	0,9954	4,7600	0,9954	5,7869	8,6477	Muy Pequeño
430	0,9526	2,5341	0,9526	3,7099	8,6477	Muy Pequeño	929	0,9014	3,7820	0,9014	5,3051	8,6477	Muy Pequeño
433	0,9699	1,5928	0,9699	2,6734	8,6477	Muy Pequeño	932	0,9001	2,4080	0,9001	3,7861	8,6477	Muy Pequeño
441	0,9506	2,2063	0,9506	3,3731	8,6477	Muy Pequeño	967	0,9619	1,5009	0,9619	2,5999	8,6477	Muy Pequeño
444	0,9430	3,5402	0,9430	4,8144	8,6477	Muy Pequeño	969	0,9674	1,6192	0,9674	2,7075	8,6477	Muy Pequeño
452	0,9444	3,8507	0,9444	5,1362	8,6477	Muy Pequeño	977	0,9711	2,7198	0,9711	3,8305	8,6477	Muy Pequeño
454	0,9669	1,7500	0,9669	2,8443	8,6477	Muy Pequeño	985	0,8907	0,8216	0,8907	2,0452	8,6477	Muy Pequeño
455	0,9204	2,3014	0,9204	3,5868	8,6477	Muy Pequeño	988	0,9400	3,5065	0,9400	4,7940	8,6477	Muy Pequeño

Fuente: elaboración propia

- **Operador de Mutación:** Se utilizó el operador de mutación PM (Polynomial Mutation) propuesto por la librería Pymoo de Python. Los parámetros de este operador son:

- a. Probabilidad de mutación: 0.9, representa la probabilidad de que un gen de una cadena genética sea mutado.
 - b. Parámetro de distribución eta: 20, controla la diversidad de la distribución utilizada en el operador de mutación. Para esta implementación se utiliza el valor estándar sugerido por Pymoo.
- **Puntos de referencia (solo para NSGA-III)**: Se utilizó el método “das-dennis” para la selección de 12 puntos de referencia en 2 objetivos, propuesto por Pymoo.
 - **Parámetros de búsqueda**:
 - a. Tamaño de población: 100
 - b. Número de generaciones: 200
 - c. Población inicial: Corresponde a las soluciones semilla seleccionadas de la simulación Monte Carlo.

También se ejecutaron los algoritmos NSGA-II y NSGA-III para cada uno de los problemas propuestos sin incorporar preferencias. Estas ejecuciones se realizaron con los mismos parámetros planteados para AGMOSIPD con el fin de poder comparar, en un escenario de iguales condiciones, los resultados de una optimización sin incorporar preferencias con los resultados de una optimización con preferencias difusas.

Los resultados de las ejecuciones de AGMOSIPD se representaron mediante gráficos en el espacio de los objetivos (x_1, x_2) , y en ellos se representaron los siguientes elementos:

- Soluciones obtenidas en la simulación Monte Carlo.
- Soluciones semilla seleccionadas de la simulación.
- Soluciones obtenidas mediante AGMOSIPD.
- Soluciones obtenidas mediante una optimización sin incorporar preferencias.

Cada gráfico se muestra en 3 escalas gráficas (zoom) de visualización para enfocar (ampliar) la apreciación de los resultados. A continuación, en la Figura 4-1 se muestran las convenciones presentes en los gráficos:

Figura 4-1: Convenciones para interpretar los gráficos de los resultados.

○	Simulación
●	Semilla
○	Frontera Pareto
●	Fragmento Difuso
+	Cota Ideal
⊗	Cota Nadir
+	Punto Preferido Ideal
⊗	Punto Preferido Nadir
●	Punto Medio
○	Punto Preferido Medio

Fuente: elaboración propia

Seguidamente, se describe cada una de las expresiones que hacen parte de la leyenda de las gráficas:

- **Simulación:** Corresponde a cada uno de los 1000 puntos obtenidos mediante simulación Monte Carlo.
- **Semilla:** Equivale a los 100 puntos seleccionados de la simulación cuya evaluación en el SID corresponde a la expresión preferida por el decisor y que se utilizarán como punto inicial del proceso de optimización.
- **Frontera de Pareto:** Representa el conjunto de soluciones óptimas obtenidas mediante NSGA-II o NSGA-III sin recurrir a la utilización del SID en la optimización.
- **Fragmento Difuso:** Hace referencia al conjunto de soluciones obtenidas mediante AGMOSIPD. Se espera que estas soluciones alcancen o superen soluciones óptimas obtenidas mediante una optimización sin preferencias.
- **Cota Ideal:** Es el punto de mínimo valor para cada objetivo obtenido en AGMOSIPD durante todas las generaciones. No necesariamente es el punto ideal.
- **Cota Nadir:** Es el punto de máximo valor para cada objetivo obtenido en AGMOSIPD durante todas las generaciones. No necesariamente es el punto nadir.
- **Punto Preferido Ideal:** Es la solución del fragmento difuso que tiene menor distancia a la cota ideal. Es decir, es una solución preferida por el decisor más cercana a la cota ideal.

- **Punto Preferido Nadir:** Es la solución del fragmento difuso que tiene mayor distancia a la cota nadir. Es decir, es una solución preferida por el decisor más lejana a la cota nadir.
- **Punto Medio:** Es la coordenada del espacio (x_1, x_2) que representa el punto medio entre el punto preferido ideal y el punto preferido nadir.
- **Punto Preferido Medio:** Es la solución del fragmento difuso que tiene menor distancia al punto medio. Es decir, es una solución preferida por el decisor más cercano al punto medio.

En las siguientes secciones se presentan representaciones gráficas de las 60 ejecuciones. Para los perfiles de preferencias Muy Pequeño, Pequeño y Medio se muestran 3 enfoques gráficos distintos para una mejor visualización de los resultados. Para los perfiles de preferencias Grande y Muy Grande solo se presenta un enfoque gráfico el cual muestra de forma suficiente su comportamiento.

4.2 Resultados Problema 1

Los resultados gráficos de este problema se pueden apreciar en la Figura A-1, Figura A-2, Figura A-3, Figura A-4, y Figura A-5.

En este problema se puede observar que los valores obtenidos mediante la simulación Monte Carlo cubren ampliamente el espacio de soluciones y dan una idea de la curvatura que podría tener la Frontera de Pareto. De hecho, es posible identificar que las soluciones de la optimización del problema sin preferencias (curva azul) bordean adecuadamente los puntos simulados. Esto permite inferir que la simulación le entrega al decisor información relevante, que le permite aprender del problema y elaborar sus preferencias.

Las soluciones que conforman la semilla y que corresponden al perfil de preferencia expresado como \mathcal{Y} en el SID se aprecian gráficamente en la zona correspondiente. Para valores de \mathcal{Y} equivalentes a Muy Pequeño, Pequeño y Medio se nota como los puntos semilla se encuentran muy cercanos a la FP, por tanto, es de esperarse que faciliten la optimización.

Se percibe que los puntos marcados en color rojo, que corresponden a las soluciones propuestas por AGMOSIPD, para γ equivalente a Muy Pequeño, Pequeño y Medio, delinean de forma similar la curva azul dando una idea de la efectividad del algoritmo para alcanzar soluciones Pareto óptimas. Para el caso donde γ equivale a Grande y Muy Grande el algoritmo no alcanza soluciones óptimas dado el SID no permite que la optimización llegue hasta ellas. En un escenario de racionalidad perfecta no es coherente que un decisor prefiera soluciones con un valor de γ igual a Grande o Muy Grande. Este mismo comportamiento es evidente en los problemas ZDT1, ZDT2, ZDT3, ZDT4 y ZDT6, donde también se observa que, en todos los problemas, la FP no registra puntos donde las tasas de intercambio entre objetivos correspondan a la zona donde el perfil de preferencias es Grande o Muy Grande, por tanto, se puede omitir el análisis de estos perfiles.

Con el fin de comprobar la eficiencia de las soluciones de AGMOSIPD, se realizó una comparación entre estas soluciones y las obtenidas mediante optimización sin preferencias y se cuantificó la cantidad de soluciones no dominadas presentes en AGMOSIPD. Es decir, se identificó la cantidad de soluciones eficientes que el algoritmo con preferencias aportó a la FP obtenida sin preferencias. En la Tabla 4-2 y la Tabla 4-3 se presentan los resultados:

Tabla 4-2: Nro. de soluciones de AGMOSIPD no dominadas en el problema 1.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
Problema 1	NSGA-II	98	100	99	297
Problema 1	NSGA-III	99	100	100	299
Total Problema 1		197	200	199	596

Fuente: elaboración propia

Tabla 4-3: % de soluciones de AGMOSIPD no dominadas en el problema 1.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
Problema 1	NSGA-II	98%	100%	99%	99%
Problema 1	NSGA-III	99%	100%	100%	100%
Total Problema 1		99%	99%	100%	100%

Fuente: elaboración propia

A partir de estos resultados se identifica que, para un perfil de preferencias Pequeño, el algoritmo propuesto es capaz de encontrar soluciones eficientes en un 100%. Para Muy Pequeño y Medio se obtienen soluciones dominadas entre 1% y 2%. Al comparar los algoritmos se percibe que NSGA-III tiene mejor desempeño que NSGA-II al encontrar soluciones no dominadas.

También se identificó que AGMOSIPD, para el Problema 1 en ambos algoritmos, no aportó soluciones que dominaran las soluciones obtenidas en la búsqueda sin preferencias. Es decir, AGMOSIPD no obtuvo resultados que superaran la FP obtenida sin preferencias.

4.3 Resultados Problema ZDT1

Los resultados gráficos de este problema se pueden apreciar en la Figura A-6, Figura A-7, Figura A-8, Figura A-9, y Figura A-10.

En los problemas ZDT1, ZDT2, ZDT3, ZDT4 y ZDT6 se aprecia que los valores simulados solo cubren un fragmento superior del espacio de soluciones y no otorgan una aproximación al lugar donde se encuentra la FP. Sin embargo, con la simulación, el decisor podría inferir una idea de sus preferencias e indicar una zona de interés de modo que el SID, dentro del optimizador, amplíe el universo del discurso de sus variables de entrada y le otorgue una trayectoria al proceso de búsqueda. En estos casos el decisor aprende con información visible y el SID se encarga de proyectar la preferencia del decisor a medida que amplía el espacio de soluciones. También se encuentra que las soluciones que conforman la semilla se distribuyen adecuadamente en el espacio de soluciones.

Al igual que en el problema 1, se encuentra que en el problema ZDT1, las soluciones obtenidas en AGMOSIPD para γ equivalente a Muy Pequeño, Pequeño y Medio, alcanzan la curva azul y en algunos casos la superan. Esto permite inferir la efectividad preliminar del algoritmo. En la Tabla 4-4 y la Tabla 4-5 se presentan los datos que muestran la comparación de AGMOSIPD con una optimización sin preferencias:

Tabla 4-4: Nro. de soluciones de AGMOSIPD no dominadas en ZDT1.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT1	NSGA-II	98	99	100	297
ZDT1	NSGA-III	100	99	100	299
Total ZDT1		198	198	200	596

Fuente: elaboración propia

Tabla 4-5: % de soluciones de AGMOSIPD no dominadas en ZDT1.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT1	NSGA-II	98%	99%	100%	99%
ZDT1	NSGA-III	100%	99%	100%	100%
Total ZDT1		99%	99%	99%	100%

Fuente: elaboración propia

En las tablas anteriores se encuentra que, en un perfil de preferencias Medio, el algoritmo encuentra soluciones eficientes en un 100%. Para Muy Pequeño y Pequeño se obtienen soluciones dominadas entre 1% y 2%. En este caso también se percibe que NSGA-III tiene mejor desempeño que NSGA-II al encontrar soluciones no dominadas.

Al analizar la dominancia entre las soluciones de AGMOSIPD y la optimización sin preferencias para el problema ZDT1, como se indica en la Tabla 4-6 y la Tabla 4-7, se encuentra que las soluciones de AGMOSIPD superan en eficiencia a algunas soluciones de la optimización sin preferencias. Se percibe que el algoritmo NSGA-II aplicado en AGMOSIPD es capaz de empujar la curva roja más allá del ámbito óptimo de la curva azul en todos los perfiles de preferencia. En definitiva, para todo el problema ZDT1 en los perfiles de preferencia analizados, AGMOSIPD podría ser capaz de dominar entre y 3% y 7% de soluciones de una optimización sin preferencias.

Tabla 4-6: Nro. de soluciones dominadas en AG sin preferencias en ZDT1.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT1	NSGA-II	6	8	7	21
ZDT1	NSGA-III	3	8	6	17
Total ZDT1		9	16	13	38

Fuente: elaboración propia

Tabla 4-7: % de soluciones dominadas en AG sin preferencias en ZDT1.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT1	NSGA-II	6%	8%	7%	7%
ZDT1	NSGA-III	3%	8%	6%	6%
Total ZDT1		5%	8%	7%	6%

Fuente: elaboración propia

4.4 Resultados Problema ZDT2

Los resultados gráficos de este problema se pueden apreciar en la Figura A 11, Figura A 12, Figura A 13, Figura A 14, y Figura A 15.

Para el problema ZDT2 también ocurre que las soluciones que conforman la semilla se distribuyen adecuadamente en el espacio de soluciones. De manera análoga a los problemas anteriores se encuentra que en el problema ZDT2, las soluciones para y equivalente a Muy Pequeño, Pequeño y Medio, alcanzan la curva azul de forma nítida, permitiendo suponer que AGMOSIPD ha encontrado soluciones Pareto óptimas.

Los resultados de la comparación de soluciones dominadas se indican en la Tabla 4-8 y la Tabla 4-9 como se ha abordado en los problemas anteriores:

Tabla 4-8: Nro. de soluciones de AGMOSIPD no dominadas en ZDT2.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT2	NSGA-II	99	99	100	298
ZDT2	NSGA-III	96	99	100	295
Total ZDT2		195	198	200	593

Fuente: elaboración propia

Tabla 4-9: % de soluciones de AGMOSIPD no dominadas en ZDT2.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT2	NSGA-II	99%	99%	100%	99%
ZDT2	NSGA-III	96%	99%	100%	98%
Total ZDT2		98%	99%	100%	99%

Fuente: elaboración propia

Con la información de las tablas anteriores se encuentra que el perfil Medio obtiene soluciones eficientes en un 100%. Los perfiles Muy Pequeño y Pequeño obtienen soluciones dominadas entre 1% y 2%. Al contrario que en los problemas anteriores se identifica que NSGA-II tiene mejor desempeño que NSGA-III al encontrar soluciones no dominadas.

En el problema ZT2 se encuentra que las soluciones de AGMOSIPD también superan en eficiencia algunas soluciones de la optimización sin preferencias. Tanto el NSGA-II como el NSGA-III encuentran soluciones eficientes más allá de la curva azul en todos los perfiles de preferencia, y la dominancia de soluciones de una optimización sin preferencias se produce entre 2% y 13%. La Tabla 4-10 y Tabla 4-11 soportan estos hallazgos.

Tabla 4-10: Nro. de soluciones dominadas en AG sin preferencias en ZDT2.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT2	NSGA-II	5	3	10	18
ZDT2	NSGA-III	2	4	16	22
Total ZDT2		7	7	26	40

Fuente: elaboración propia

Tabla 4-11: % de soluciones dominadas en AG sin preferencias en ZDT2.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT2	NSGA-II	5%	3%	10%	6%
ZDT2	NSGA-III	2%	4%	16%	7%
Total ZDT2		4%	4%	13%	7%

Fuente: elaboración propia

4.5 Resultados Problema ZDT3

Los resultados gráficos de este problema se pueden apreciar en la Figura A 16, Figura A 17, Figura A 18, Figura A 19, y Figura A 20.

Visualmente, es posible inferir que al aplicar AGMOSIPD en el problema ZDT3 las soluciones resaltadas en color rojo se aproximan exitosamente a la curva azul para y igual

a Muy Pequeño, Pequeño y Medio. Incluso parece suceder que, a pesar de la forma discontinua de la FP, AGMOSIPD no tiene inconvenientes para encontrar soluciones óptimas.

Para confirmar las sospechas de la interpretación gráfica se realiza la comparación de soluciones dominadas a partir de los valores que se expresan en la Tabla 4-12 y la Tabla 4-13:

Tabla 4-12: Nro. de soluciones de AGMOSIPD no dominadas en ZDT3.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT3	NSGA-II	71	95	99	265
ZDT3	NSGA-III	75	99	99	273
Total ZDT3		146	194	198	538

Fuente: elaboración propia

Tabla 4-13: % de soluciones de AGMOSIPD no dominadas en ZDT3.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT3	NSGA-II	71%	95%	99%	88%
ZDT3	NSGA-III	75%	99%	99%	91%
Total ZDT3		73%	97%	99%	90%

Fuente: elaboración propia

A partir de los resultados indicados en la tabla anterior se percibe que, para un perfil Muy Pequeño, el porcentaje de soluciones no dominadas cae por debajo del 90% con respecto a los problemas anteriores. Los perfiles Pequeño y Medio obtienen soluciones dominadas entre 1% y 5%. Para este caso el algoritmo NSGA-III obtiene mayor cantidad de soluciones no dominadas. Al analizar la forma de la FP se encuentran diferentes curvaturas en su extensión, lo que parece sugerir que la curvatura que tiene la FP en un perfil Muy Pequeño podría aportar dificultad al proceso de búsqueda mediante AGMOSIPD.

En el problema ZDT3 es posible superar en eficiencia algunas soluciones de la optimización sin preferencias. Sin embargo, su comportamiento de superación es inferior a la superación detectada en ZDT1 y ZDT2, aunque superior al Problema 1. A lo sumo el porcentaje de soluciones que domina AGMOSIPD de una optimización sin preferencias es del 7%, y este comportamiento se puede apreciar en la Tabla 4-14 y la Tabla 4-15.

Tabla 4-14: Nro. de soluciones dominadas en AG sin preferencias en ZDT3.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT3	NSGA-II	3	6	4	13
ZDT3	NSGA-III	0	7	3	10
Total ZDT3		3	13	7	23

Fuente: elaboración propia

Tabla 4-15: % de soluciones dominadas en AG sin preferencias en ZDT3.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT3	NSGA-II	3%	6%	4%	4%
ZDT3	NSGA-III	0%	7%	3%	3%
Total ZDT3		2%	7%	4%	4%

Fuente: elaboración propia

4.6 Resultados Problema ZDT4

Los resultados gráficos de este problema se pueden apreciar en la Figura A 21, Figura A 22, Figura A 23, Figura A 24 y Figura A 25.

Al realizar una inspección visual de los resultados del problema ZDT4 es posible detectar que la ejecución de AGMOSIPD no alcanza a identificar completamente soluciones eficientes y pareciera que no es capaz de limitar el espacio de soluciones a las restringidas por el SID. Se observa que para γ igual a Medio, el algoritmo encuentra soluciones eficientes.

Con los datos que se indican en la Tabla 4-16 y la Tabla 4-17 se confirma la sospecha detectada en la interpretación gráfica, ya que la cantidad de soluciones no dominadas que aporta AGMOSIPD, en este problema, cae drásticamente con respecto a los valores encontrados en los problemas analizados hasta este punto. Ni siquiera para un perfil de preferencia Muy Pequeño AGMOSIPD alcanza un 50% de soluciones no dominadas, un perfil de preferencia Pequeño no supera el 90% y, como ya se sospechaba, un perfil de preferencia Medio se comporta satisfactoriamente para NSGA-II (con 90% de soluciones no dominadas) y excelente para NSGA-III (con 100% de soluciones no dominadas).

Tabla 4-16: Nro. de soluciones de AGMOSIPD no dominadas en ZDT4.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT4	NSGA-II	15	74	90	179
ZDT4	NSGA-III	26	88	100	214
Total ZDT4		41	162	190	393

Fuente: elaboración propia

Tabla 4-17: % de soluciones de AGMOSIPD no dominadas en ZDT4.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT4	NSGA-II	15%	74%	90%	60%
ZDT4	NSGA-III	26%	88%	100%	71%
Total ZDT4		21%	81%	95%	66%

Fuente: elaboración propia

Al revisar la información de la Tabla 4-18 y la Tabla 4-19 se percibe que el problema ZDT4 supera débilmente, con respecto a ZDT1, ZDT2 y ZDT3, en eficiencia algunas soluciones de la optimización sin preferencias.

Tabla 4-18: Nro. de soluciones dominadas en AG sin preferencias en ZDT4.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT4	NSGA-II	0	3	0	3
ZDT4	NSGA-III	1	16	9	26
Total ZDT4		1	19	9	29

Fuente: elaboración propia

Tabla 4-19: % de soluciones dominadas en AG sin preferencias en ZDT4.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT4	NSGA-II	0%	3%	0%	1%
ZDT4	NSGA-III	1%	16%	9%	9%
Total ZDT4		1%	10%	5%	5%

Fuente: elaboración propia

El modelo propuesto en ZDT4 es un problema desafiante debido a la alta “no linealidad” de las funciones objetivo y, además, el comportamiento de los objetivos es no convexo, lo

cual produce que los algoritmos evolutivos puedan atascarse fácilmente en óptimos locales y converger prematuramente. Este problema contiene 21⁹ fronteras de Pareto locales y por tanto representa un desafío para los algoritmos evolutivos lidiar con la presencia de esa multimodalidad (Zitzler et al., 2000). Dada esta limitación matemática, es posible inferir que AGMOSIDP no contribuye a encontrar soluciones óptimas para preferencias distintas a las que ofrece el codo de la FP.

4.7 Resultados Problema ZDT6

Los resultados gráficos de este problema se pueden apreciar en la Figura A 26, Figura A 27, Figura A 28, Figura A 29, y Figura A 30.

Las gráficas para cada una de las ejecuciones de AGMOSIPD en este problema muestran un comportamiento bien definido en las soluciones que ofrece, alcanzando una forma similar a la curva azul. También es posible detectar que algunas soluciones de color rojo dominan algunas soluciones de color azul, lo cual es observable para los perfiles Muy Pequeño, Pequeño y Medio.

A través de la comparación de soluciones entre AGMOSIPD y la optimización sin preferencias, que se muestra en la Tabla 4-20 y la Tabla 4-21, se confirman los buenos resultados en el problema ZDT6. La efectividad en los perfiles Muy Pequeño y Medio es del 100% y, para el caso de Pequeño su efectividad alcanza el 99% en NSGA-II y el 88% para NSGA-III.

Tabla 4-20: Nro. de soluciones de AGMOSIPD no dominadas en ZDT6.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT6	NSGA-II	100	99	100	299
ZDT6	NSGA-III	100	88	100	288
Total ZDT6		200	187	200	587

Fuente: elaboración propia

Tabla 4-21: % de soluciones de AGMOSIPD no dominadas en ZDT6.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT6	NSGA-II	100%	99%	100%	100%
ZDT6	NSGA-III	100%	88%	100%	96%
Total ZDT6		100%	94%	100%	98%

Fuente: elaboración propia

Al revisar los resultados planteados en la Tabla 4-22 y la Tabla 4-23, se percibe que el problema ZDT6 encuentra soluciones óptimas más allá de las que se pueden encontrar en una optimización sin preferencias, superior al 10% en la mayoría de los casos. Esto permite deducir que la incorporación de preferencias promueve la obtención de mejores resultados en los perfiles Muy Pequeño, Pequeño y Medio.

Tabla 4-22: Nro. de soluciones dominadas en AG sin preferencias en ZDT6.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT6	NSGA-II	21	12	15	48
ZDT6	NSGA-III	24	6	21	51
Total ZDT6		45	18	36	99

Fuente: elaboración propia

Tabla 4-23: % de soluciones dominadas en AG sin preferencias en ZDT6.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
ZDT6	NSGA-II	21%	12%	15%	16%
ZDT6	NSGA-III	24%	6%	21%	17%
Total ZDT6		23%	9%	18%	17%

Fuente: elaboración propia

4.8 Comportamiento de los Perfiles de Preferencia Grande y Muy Grande

En el desarrollo del presente estudio se ha detectado que la ejecución del AGMOSIPD en los perfiles Grande y Muy Grande no aporta valores de soluciones óptimas que puedan ser de interés para el decisor por las siguientes razones:

- La forma de la FP de todos los problemas tiende a aproximarse o a tocar el eje de las ordenadas (eje del objetivo x_2), por tanto, no es viable obtener una tasa de intercambio grande o muy grande que ofrezca soluciones óptimas. Estos perfiles de preferencia tendrían sentido en fronteras de Pareto que tengan una inclinación más pronunciada, casi que con un comportamiento asintótico con respecto a las ordenadas. Como trabajo futuro podría probarse la ejecución de AGMOSIPD en este tipo de funciones de Pareto.
- En los casos donde las funciones objetivo se comportan de forma no convexa o incluso tienen un comportamiento multimodal (es decir, que contienen múltiples picos y valles), se encuentra que soluciones de perfil Grande y Muy Grande no alcanzan a llegar a soluciones óptimas, como en ZDT4 (ver Tabla 4-24 y la Tabla 4-25), o presentan un comportamiento errático donde al parecer el SID no logra ubicar de forma adecuada estas soluciones, como en ZDT6 donde el espacio de búsqueda es no uniforme (Zitzler et al., 2000), ya que muestra soluciones con perfil Muy Grande donde en realidad el perfil debería ser Muy Pequeño.

Tabla 4-24: Nro. de soluciones no dominadas en ZDT4 con perfil GG y MG.

Problema	Algoritmo	Grande	Muy Grande	Total
ZDT4	NSGA-II	0	1	1
ZDT4	NSGA-III	1	1	2
Total ZDT4		1	2	3

Fuente: elaboración propia

Tabla 4-25: % de soluciones no dominadas en ZDT4 con perfil GG y MG..

Problema	Algoritmo	Grande	Muy Grande	Total
ZDT4	NSGA-II	0,00%	1,00%	0,50%
ZDT4	NSGA-III	1,00%	1,00%	1,00%
Total ZDT4		0,50%	1,00%	0,75%

Fuente: elaboración propia

En vista de la carencia de aporte de estos perfiles de preferencia y bajo el supuesto de racionalidad perfecta en el decisor, se descarta la evaluación de la calidad de las soluciones obtenidas en las ejecuciones de estos perfiles.

4.9 Resumen de Resultados

Al realizar una evaluación conjunta de los resultados en la obtención de soluciones no dominadas, a través de la Tabla 4-26, se evidencian los siguientes elementos:

- La aplicación de AGMOSIPD en el Problema 1 y en ZDT1, ZDT2 y ZDT6 en NSGA-II y NSGA-III producen al menos un 95% de soluciones no dominadas en los perfiles de preferencia Muy Pequeño, Pequeño y Medio.
- Se percibe una marcada tendencia de AGMOSIPD a alcanzar soluciones eficientes en el perfil de preferencia Medio, el cual equivale al codo o zona de inflexión de la FP donde ocurren las tasas de intercambio más equilibradas.
- El algoritmo NSGA-III proporciona mayor o igual cantidad de soluciones no dominadas en la mayoría de las ejecuciones realizadas, cuando se compara con el algoritmo NSGA-II.

Tabla 4-26: % de soluciones de AGMOSIPD no dominadas.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
Problema 1	NSGA-II	98%	100%	99%	99%
Problema 1	NSGA-III	99%	100%	100%	100%
Total Problema 1		99%	100%	100%	99%
ZDT1	NSGA-II	98%	99%	100%	99%
ZDT1	NSGA-III	100%	99%	100%	100%
Total ZDT1		99%	99%	99%	100%
ZDT2	NSGA-II	99%	99%	100%	99%
ZDT2	NSGA-III	96%	99%	100%	98%
Total ZDT2		98%	99%	100%	99%
ZDT3	NSGA-II	71%	95%	99%	88%
ZDT3	NSGA-III	75%	99%	99%	91%
Total ZDT3		73%	97%	99%	90%
ZDT4	NSGA-II	15%	74%	90%	60%
ZDT4	NSGA-III	26%	88%	100%	71%
Total ZDT4		21%	81%	95%	66%
ZDT6	NSGA-II	100%	99%	100%	100%
ZDT6	NSGA-III	100%	88%	100%	96%
Total ZDT6		100%	94%	100%	98%
Total NSGA-II		80%	94%	98%	91%
Total NSGA-III		83%	96%	100%	93%

Fuente: elaboración propia

Al revisar la comparación de soluciones dominadas frente a la optimización sin preferencias con AGMOSIPD se concluye, a partir de la información de la Tabla 4-27, que:

- En la mayoría de las ejecuciones del problema ZDT6, AGMOSIPD proporciona mayor cantidad de soluciones óptimas con respecto a la optimización de preferencias bajo una misma cantidad de generaciones.
- En 200 generaciones, AGMOSIPD no aporta mayor cantidad de soluciones con respecto a la optimización tradicional en el Problema 1.
- En la mayoría de los casos, el algoritmo NSGA-III produce mayor o igual número de soluciones óptimas por encima de la optimización sin preferencias.
- En problemas cuya FP es no convexa, como en ZDT2 y ZDT6, AGMSIPD aporta más soluciones eficientes con respecto a la optimización sin preferencias cuando el perfil es Medio.

Tabla 4-27: % de soluciones de la optimización sin preferencias dominadas.

Problema	Algoritmo	M. Pequeño	Pequeño	Medio	Total
Problema 1	NSGA-II	0%	0%	0%	0%
Problema 1	NSGA-III	0%	0%	0%	0%
Total Problema 1		0%	0%	0%	0%
ZDT1	NSGA-II	6%	8%	7%	7%
ZDT1	NSGA-III	3%	8%	6%	6%
Total ZDT1		5%	8%	7%	6%
ZDT2	NSGA-II	5%	3%	10%	6%
ZDT2	NSGA-III	2%	4%	16%	7%
Total ZDT2		4%	4%	13%	7%
ZDT3	NSGA-II	3%	6%	4%	4%
ZDT3	NSGA-III	0%	7%	3%	3%
Total ZDT3		2%	7%	4%	4%
ZDT4	NSGA-II	0%	3%	0%	1%
ZDT4	NSGA-III	1%	16%	9%	9%
Total ZDT4		1%	10%	5%	5%
ZDT6	NSGA-II	21%	12%	15%	16%
ZDT6	NSGA-III	24%	6%	21%	17%
Total ZDT6		23%	9%	18%	17%
Total NSGA-II		6%	5%	6%	6%
Total NSGA-III		5%	7%	9%	7%

Fuente: elaboración propia

4.10 Elección de Soluciones Óptimas

La aplicación de un algoritmo de optimización multiobjetivo con incorporación de preferencias debe producir una única solución que satisfaga al decisor. Con el hallazgo de estas soluciones finaliza la ejecución del AGMOSIPD y corresponderá al decisor llevar a cabo una decisión con esta información.

En las tablas que se presentan A continuación, se expresan las soluciones óptimas que AGMOSIPD sugiere como solución final para cada problema, perfil de preferencia y algoritmo, y a partir de ellas el decisor deberá elegir un criterio que se ajuste mejor a sus necesidades:

- Criterio de solución más cercana a la cota ideal.
- Criterio de solución más lejana a la cota nadir.
- Criterio de solución más cercana al punto medio entre la solución más cercana a la cota ideal y la solución más lejana a la cota nadir.

Las convenciones de las tablas son:

1. PRB: Problema, PBR1 significa Problema 1
2. PR: Perfil de Preferencia que puede ser MP (Muy Pequeño), PP (Pequeño) y MM (Medio).
3. AG: Algoritmo evolutivo utilizado en AGMOSIPD.
4. X1: Valor obtenido en la función objetivo 1
5. X2: Valor obtenido en la función objetivo 2
6. DCI: Valor de distancia euclidiana de la solución con respecto a la cota ideal.
7. DCN: Valor de distancia euclidiana de la solución con respecto a la cota nadir.
8. DPM: Valor de distancia euclidiana de la solución con respecto al punto medio.

Los valores resaltados corresponden a:

1. En la columna DCI es la mínima distancia obtenida en AGMOSIPD.
2. En la columna DCN es la máxima distancia obtenida en AGMOSIPD
3. En la columna DPM es la mínima distancia obtenida en AGMOSIPD

Seguidamente se concluyen algunos aspectos analizando los resultados en cada uno de los problemas:

- En la Tabla A-1 se visualizan las soluciones del problema 1 donde se puede apreciar que, para el perfil Medio se encuentra la misma solución para los criterios DCI, DCN y DPM.
- Para los problemas ZDT1, ZDT2, ZDT3, ZDT4 y ZDT6 se percibe que el valor de distancia para DCI y DCN disminuye conforme el perfil de preferencia pasa de MP a MM.
- Las soluciones del problema ZDT1, ZDT2 y ZDT6 se presentan en la Tabla A-2, Tabla A-3 y Tabla A-6 respectivamente. Para estos problemas se sugiere ofrecerle al decisor todas las alternativas encontradas en los perfiles MP, PP y MM dado el éxito que obtuvieron en alcanzar mejores soluciones que una optimización sin preferencias. El problema que presenta mayor nivel de eficiencia en soluciones no dominadas y soluciones dominadas con respecto a un problema sin preferencia es el ZDT6.
- Las soluciones del problema ZDT3 están expuestas en la Tabla A-4. En este caso, las soluciones para el perfil MP no se recomiendan debido a que el porcentaje de soluciones eficientes que se obtuvieron fue inferior al 80%. Únicamente se recomienda ofrecer al decisor las soluciones del perfil PP y MM, los cuales obtuvieron más de un 90% de soluciones eficientes.
- Las soluciones del problema ZDT4 se indican en la Tabla A-5. En este punto las soluciones para el perfil MP y PP no se recomiendan debido a que el porcentaje de soluciones eficientes que se obtuvieron fue inferior al 30% para MP e inferior al 80% para PP. Únicamente se recomienda ofrecer al decisor las soluciones del MM, los cuales obtuvieron un 90% o más de soluciones eficientes.

Con los resultados expuestos se comprueba el funcionamiento de la propuesta y se verifica la obtención de soluciones eficientes tal como se plantea en el quinto objetivo específico de a presente investigación.

5. Conclusiones, limitaciones y trabajos futuros

5.1 Conclusiones

La presente investigación se realizó con el propósito de diseñar una metodología que permita incorporar un Sistema de Inferencia Difusa para asignar las preferencias en la optimización de un POM mediante AG. Durante el desarrollo de este propósito fue posible concluir lo siguiente:

- **Elección del método de asignación de preferencias:** El estudio de la asignación de preferencias en un POM, por parte de un decisor, contempla la incorporación a priori, a posteriori o interactiva de las mismas. Estos mecanismos de incorporación presentan ventajas y desventajas en las distintas propuestas que se han abordado en el estado del arte; sin embargo, se percibe un elemento común entre ellas, que corresponde al nivel de conocimiento que se le brinda al decisor sobre el comportamiento del problema, de tal forma que este pueda elaborar sus preferencias. Podría existir mayor riesgo de que el decisor no tenga información sobre el comportamiento del problema en los métodos a priori y por ende podría enfrentarse a elaborar sus preferencias mediante especulación. Con el fin de proporcionar mayor conocimiento al decisor sobre el problema, se incorporó un enfoque de aprendizaje previo a la optimización buscando reducir el riesgo de que el decisor especule su construcción de preferencias de forma a priori. Los resultados de las ejecuciones de AGMOSIPD presentados en esta propuesta permiten observar que la etapa de aprendizaje previo mediante simulación Monte Carlo facilita un conocimiento del problema para que el decisor pueda elaborar sus preferencias antes de la optimización, proporcionando soporte metodológico a la elección del método de asignación de preferencias de forma a priori.

- **Aplicación de lógica difusa en la estructuración de preferencias:** Se han desarrollado diferentes técnicas para que el decisor pueda construir una estructura para definir sus preferencias, y varias de ellas requieren que el decisor pueda expresarlas de forma numérica. Sin embargo, se ha encontrado que establecerlas de esta manera puede conllevar a sobrecarga cognitiva, cometer errores al momento de valorar cuantitativamente percepciones subjetivas y no garantizan que el decisor dimensione adecuadamente las limitaciones del problema. Como alternativa para combatir estas limitaciones es posible establecer preferencias de forma cualitativa mediante la formulación de reglas de comparación o la utilización de términos lingüísticos a través de lógica difusa. Esta última contempla la utilización de SID para representar el conocimiento mediante palabras. En este estudio se incorporaron preferencias mediante un SID representando diferentes niveles de tasas de intercambio entre objetivos a través de palabras. Esta propuesta se aleja de los métodos que trabajan con pesos de ponderación (que suponen una tasa de intercambio única definida por el decisor para todo el espacio de soluciones incluso si se consideran pesos difusos), AGMOSIPD considera que las tasas de intercambio dependen de la zona del espacio de soluciones en que se está evaluando el proceso. Durante la aplicación de AGMOSIPD puede observarse que es posible definir preferencias mediante el uso de términos lingüísticos para representar un problema de optimización multiobjetivo.
- **Desempeño de los algoritmos NSGA-II y NSGA-III:** Entre las técnicas de mayor utilización en el campo de los MOEAs se ubican los algoritmos NSGA-II y NSGA-III, que han mostrado mayor efectividad en la resolución de diferentes problemas de prueba y aplicaciones de la vida real. La finalidad de estas técnicas es entregar muchísimas soluciones repartidas en toda la FP sin inclusión de preferencias, por lo cual el elector, a posteriori, debe elegir una sola de ellas por cualquier otro método multicriterio. El algoritmo NSGA-II presenta mejor desempeño en problemas de múltiples objetivos y el algoritmo NSGA-III tiene mayor efectividad en problemas de muchos objetivos. En consonancia con esto, se detectó que la aplicación de AGMOSIPD en el Problema 1 y en ZDT1, ZDT2 y ZDT6 en NSGA-II y NSGA-III producen al menos un 95% de soluciones no dominadas en los perfiles

de preferencia Muy Pequeño, Pequeño y Medio. Adicionalmente, el algoritmo NSGA-III proporciona mayor o igual cantidad de soluciones no dominadas, en la mayoría de las ejecuciones realizadas, cuando se compara con el algoritmo NSGA-II. Esto permite concluir que en un escenario de incorporación de preferencias difusas el NSGA-II y NSGA-III obtienen un desempeño satisfactorio e, incluso, pueden superar su desempeño cuando se comparan con una optimización sin preferencias. En definitiva, AGMOSIPD usa las ventajas del NSGA pero lo mejora ya que el resultado final es un pequeño conjunto de soluciones no dominadas preferidas por el decisor, lo que implica que se requiere un proceso, a posteriori, mínimo o incluso ninguno si se elige el centroide de la zona como se propone en este trabajo.

- **Obtención de soluciones eficientes:** La aplicación de AGMOSIPD en los problemas de prueba 1, ZDT1, ZDT2 y ZDT6, muestra que en ellos se alcanzan soluciones eficientes en más del 90% de las soluciones obtenidas en los perfiles de preferencia Muy Pequeño, Pequeño y Medio, tanto en NSGA-II como en NSGA-III. Ello da cuenta de la efectividad de esta propuesta para alcanzar Fronteras de Pareto. Existen problemas como el ZDT3 y ZDT4 donde la geometría y formulación matemática de los mismos es compleja y donde, además, el porcentaje de soluciones eficientes cae con respecto a los demás casos de prueba. Esto permite especular que es posible que exista relación entre la discontinuidad, no linealidad, multimodalidad del problema, y las dificultades que experimenta AGMOSIPD para encontrar soluciones eficientes. También se identifica que, en la mayoría de los casos, las soluciones que obtiene AGMOSIPD corresponde al 99% o al 100% de soluciones eficientes, este hallazgo generalmente coincide con un nivel de preferencia Medio.
- **Comparación de soluciones eficientes con y sin preferencia:** Al comparar las soluciones obtenidas por AGMOSIPD y las obtenidas por una optimización sin preferencias, se encuentra que AGMOSIPD es capaz de encontrar soluciones que dominan algunas soluciones pertenecientes a la optimización sin preferencias. Es importante destacar que AGMOSIPD tiende a favorecer la explotación de soluciones más que la exploración, debido a la influencia que tiene el SID incluido como restricción, lo que permite presumir que la restricción contribuye a la mejora

de las soluciones encontradas en zonas muy focalizadas sin considerar diversidad. Este hallazgo es importante porque podría pensarse que un AG sin preferencias hace uso de costo computacional en encontrar soluciones en toda la FP, mientras que con un AG con preferencias, como AGMOSIPD, se enfoca solo en una parte, por lo que puede encontrar mejores soluciones FP con una misma cantidad de iteraciones.

- **Pertinencia en la definición de base de reglas:** Durante las ejecuciones de AGMOSIPD se encontró que el perfil de preferencias Grande y Muy Grande no aporta soluciones eficientes y coherentes para los problemas analizados, lo que da cuenta de la importancia de definir unas variables lingüísticas y una base de reglas coherente y acorde con las potencialidades del problema.

En definitiva, este trabajo logra que algoritmos tan prometedores para la solución de POM complejos, como NSGA II y III, no requieran concentrarse en el hallazgo de toda la FP, lo cual supone, para el decisor, una carga excesiva de elección, a posteriori, pues en definitiva solo se requiere implementar una única solución de ella. El método propuesto, a diferencia de las técnicas de optimización sin preferencias, entrega una única solución (o pocas) de la FP (o cercanas a ella) considerando preferencias cuya expresión por parte del decisor se hace a partir de expresiones lingüísticas, lo cual cognitivamente es más realista con la forma en que los seres humanos piensan y definen sus percepciones.

5.2 Limitaciones

La ejecución del trabajo desarrollado presenta las siguientes limitaciones:

- La implementación de esta propuesta requirió de un esfuerzo computacional importante, que se tradujo en una dedicación en tiempo relevante. Las pruebas se realizaron en varios equipos de cómputo donde se produjeron ejecuciones que oscilaron entre 1.680 segundos y 3.600 segundos. Esto impidió que se repitieran las 60 ejecuciones muchas más veces o que, incluso, se incrementara el número de generaciones, dado que podía generar bloqueos en el procesamiento de cómputo.

- Se percibió que las ejecuciones de AGMOSIPD experimentaban mayor tiempo de procesamiento durante el momento en el que los algoritmos genéticos evaluaban cada individuo en la restricción que contenía el SID. Una posible solución a este punto podría abordarse interviniendo el código fuente de la librería skfuzzy para mejorar el rendimiento de la función que pone el SID en operación. Esta intervención no fue posible realizarla debido a la alta complejidad que implica su modificación.
- No fue posible detectar posibles relaciones entre las dificultades de AGMOSIPD para encontrar soluciones óptimas en los problemas ZDT3, ZDT4 y ZDT6, y la complejidad de la estructura matemática que poseen estos problemas.
- Este trabajo asume que el decisor actúa bajo criterios de racionalidad perfecta. Por tanto, AGMOSIPD exige que el diseño de la VL de salida y la base de reglas sea coherentes con el comportamiento de la FP del problema. Este supuesto de racionalidad perfecta no siempre se cumple por lo que podría afectar la efectividad de AGMOSIPD para ofrecer soluciones satisfactorias a un decisor.

5.3 Trabajos Futuros

A continuación, se presentan algunas recomendaciones que permitirían encontrar nuevas rutas de investigación futura para ampliar esta propuesta:

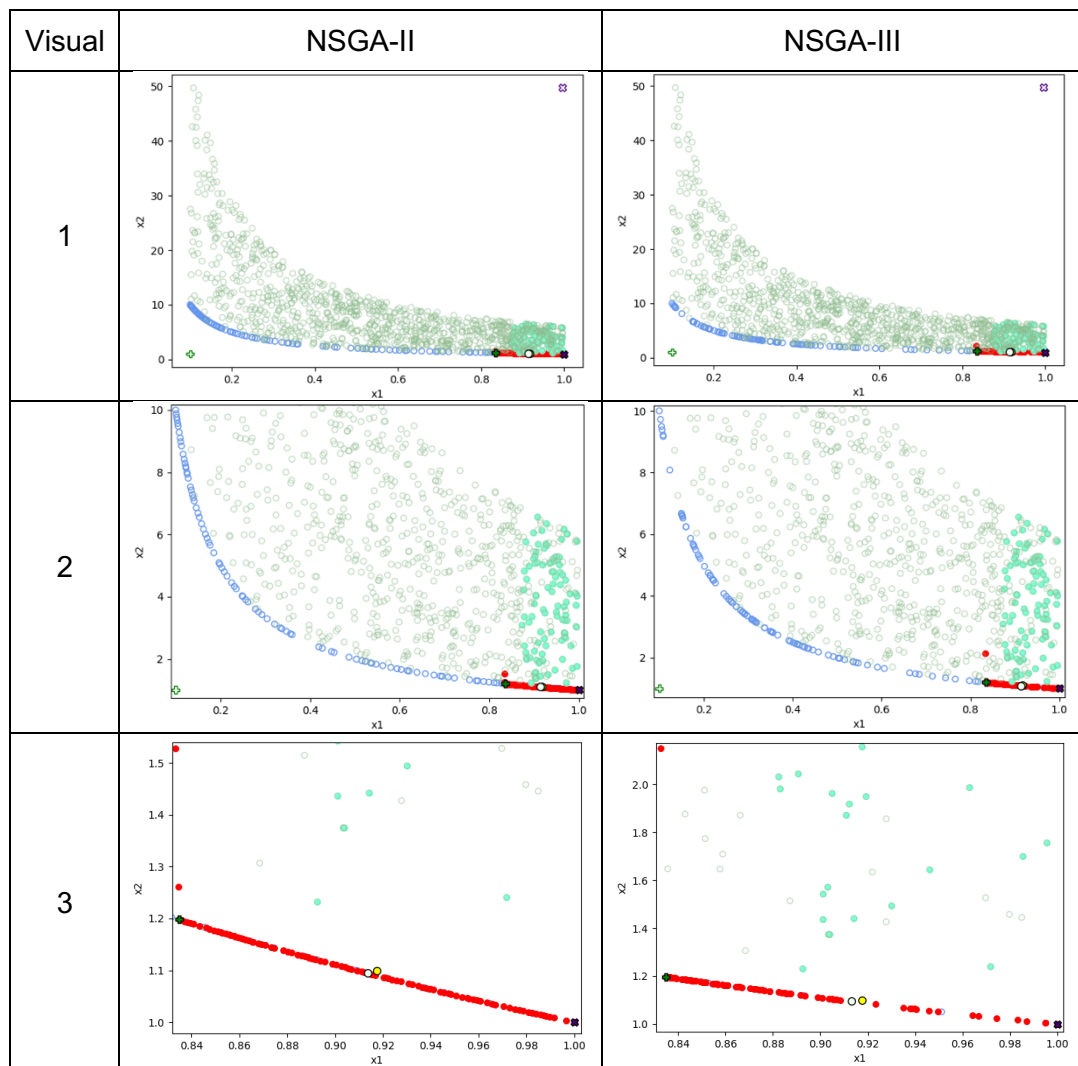
- **Utilización de otros algoritmos evolutivos:** Se sugiere considerar otros algoritmos evolutivos, como por ejemplo los basados en descomposición (MOEA/D), para verificar el comportamiento de AGMOSIPD en la obtención de soluciones eficientes.
- **Incorporación de muchos objetivos:** Se recomienda agregar más de dos objetivos y considerar la aplicación de múltiples Sistemas de Inferencia Difusa, considerando la complejidad que podría resultar para el decisor modelar sus preferencias por pares de objetivos.

- **Exploración de problemas:** Para continuar revisando la pertinencia de AGMOSIPD, podría verificarse su aplicación en problemas discretos, combinatoriales, con restricciones, e incluso considerando aplicaciones de la realidad donde quizás la estructura matemática del problema no es tan compleja y es más cercana a casos reales.
- **Inclusión de técnicas de aprendizaje inicial:** La incorporación de otras técnicas de simulación o la utilización de redes neuronales en una fase previa a la optimización para encontrar soluciones que aporten conocimiento para la construcción de preferencias mediante un SID podría proporcionar mayores avances en este campo de investigación.
- **Variación en el diseño del SID:** Se recomienda explorar el diseño de VL mediante funciones de pertenencia distintas a trapezoidales, otras configuraciones de etiquetas lingüísticas y cantidad de conjuntos difusos, diseñar otras variables de salida del SID con una estructura de preferencias distinta a tasas de intercambio.
- **Evaluación del impacto de la racionalidad limitada del decisor:** La incorporación de preferencias en los problemas de optimización asumen racionalidad perfecta, por lo que se sugiere estudiar de qué manera se pueden estructurar preferencias en un modelo como AGMOSIPD que se base en criterios de racionalidad limitada.
- **Revisión de la incidencia de la geometría del problema:** Se propone verificar si existe relación entre la forma del problema, tales como FP convexas, no convexas, discontinuas, multimodales; y una posible incidencia en la efectividad de AGMOSIPD para encontrar soluciones eficientes.

A. Anexo: Figuras y Tablas Adicionales

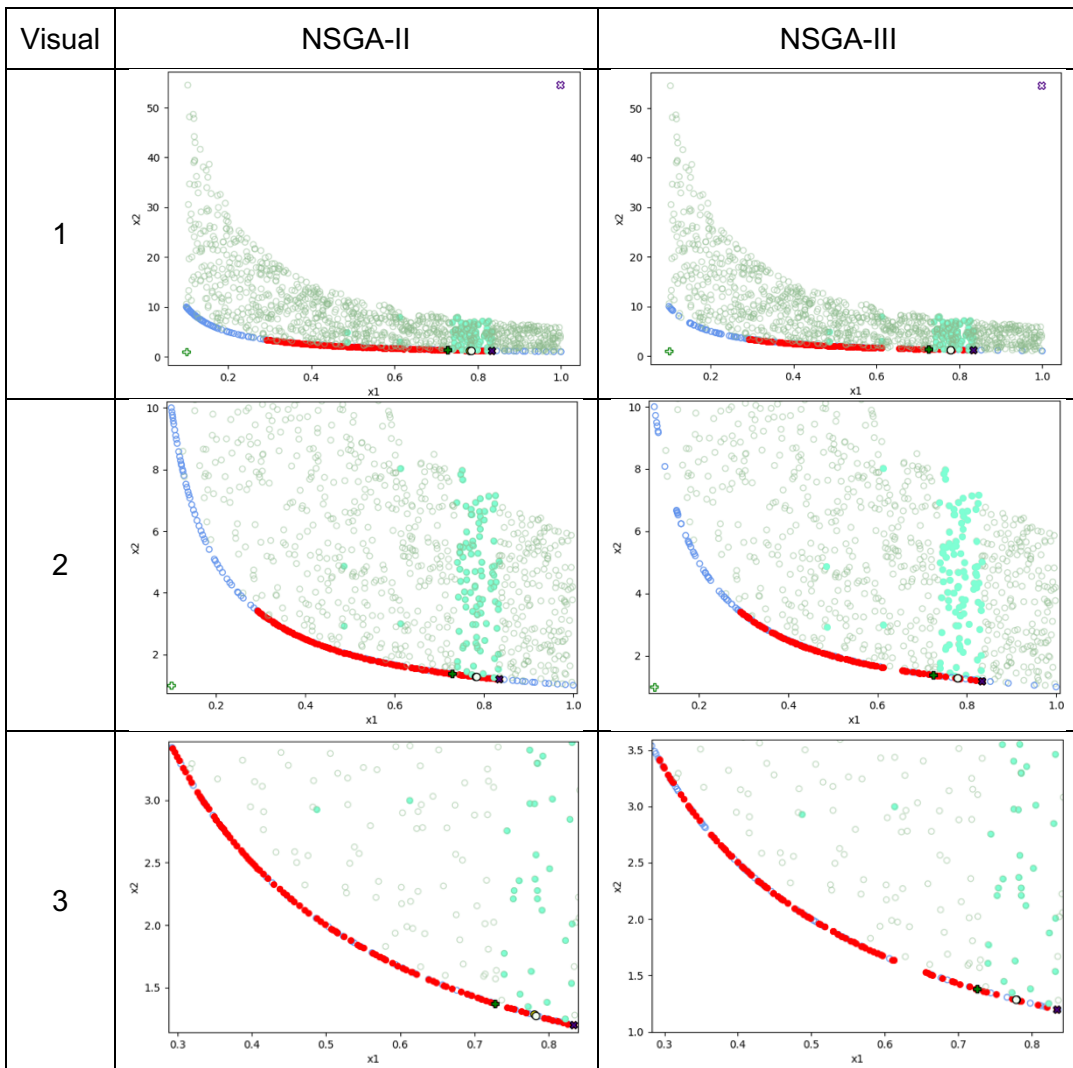
En este anexo se presentan algunas figuras y tablas que se mencionan en el desarrollo del presente trabajo:

Figura A-1: Resultado de AGMOSIPD en el problema 1 con $\gamma = \text{“Muy Pequeño”}$.



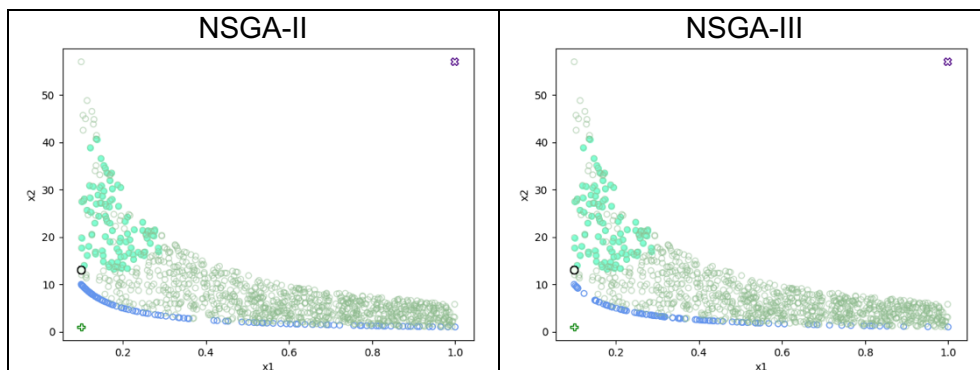
Fuente: elaboración propia

Figura A-2: Resultado de AGMOSIPD en el problema 1 con $\gamma = \text{“Pequeño”}$.



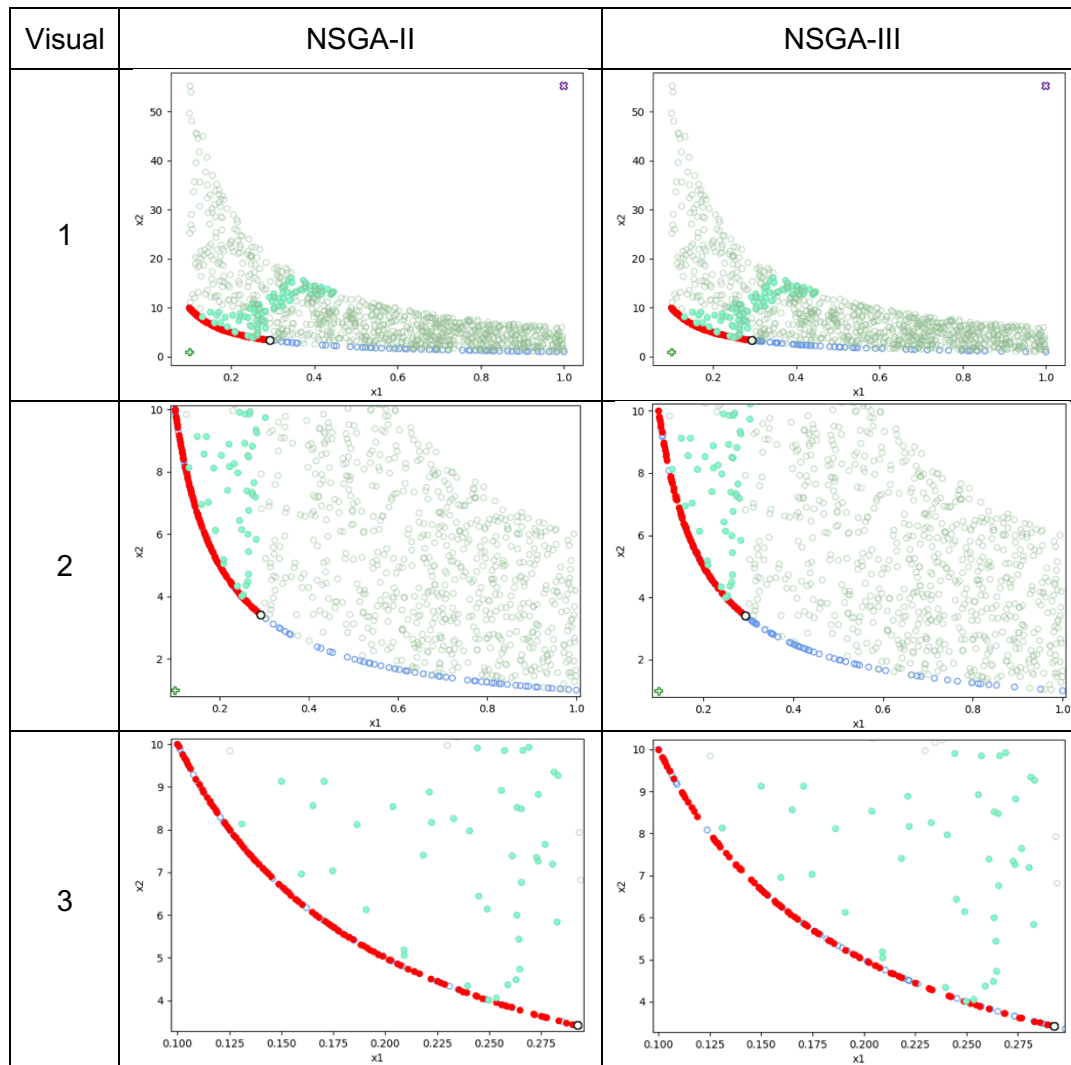
Fuente: elaboración propia

Figura A-3: Resultado de AGMOSIPD en el problema 1 con $\gamma = \text{“Grande”}$.



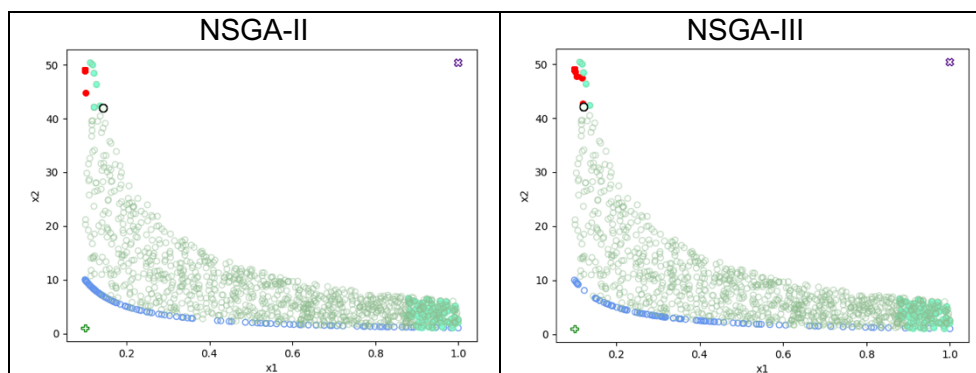
Fuente: elaboración propia

Figura A-4: Resultado de AGMOSIPD en el problema 1 con $\gamma = \text{“Medio”}$.



Fuente: elaboración propia

Figura A-5: Resultado de AGMOSIPD en el problema 1 con $\gamma = \text{“Muy Grande”}$.



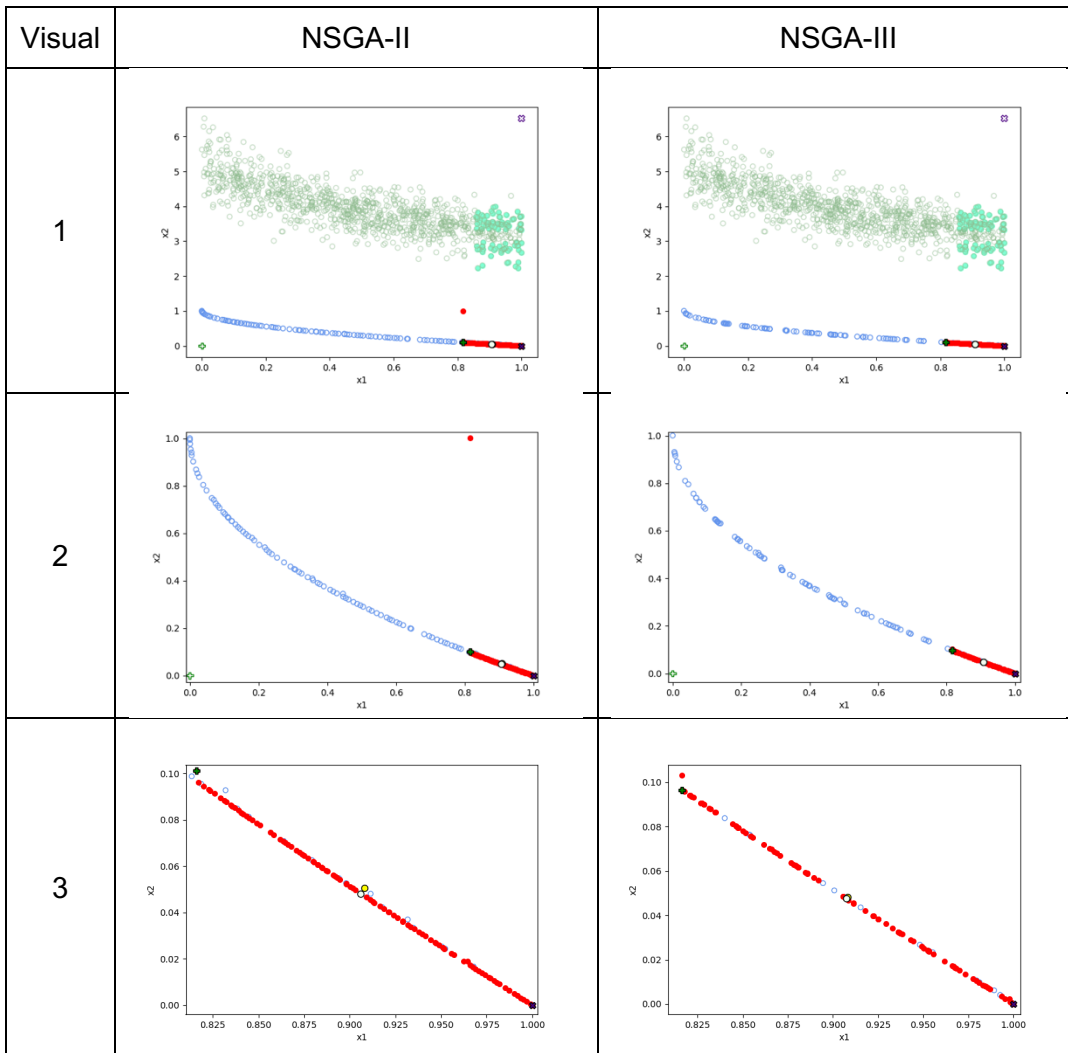
Fuente: elaboración propia

Tabla A-1: Soluciones del Problema 1 seleccionadas por AGMOSIPD.

PRB	PR	AG	X1	X2	DCI	DCN	DPM
PRB1	MP	NSGA2	0,834713	1,198062	0,759939	48,520202	0,128985
PRB1	MP	NSGA3	0,834951	1,197675	0,760068	48,520588	0,128760
PRB1	MP	NSGA3	1,000000	1,000000	0,898962	48,717982	0,128760
PRB1	MP	NSGA2	1,000000	1,000000	0,898962	48,717982	0,128985
PRB1	MP	NSGA3	0,913089	1,095183	0,817611	48,622877	0,005709
PRB1	MP	NSGA2	0,913778	1,094358	0,818199	48,623701	0,005886
PRB1	PP	NSGA3	0,724935	1,379434	0,706171	53,140093	0,105002
PRB1	PP	NSGA2	0,728257	1,373141	0,706211	53,146368	0,101473
PRB1	PP	NSGA2	0,833571	1,199658	0,748425	53,319419	0,101473
PRB1	PP	NSGA3	0,833545	1,199695	0,748407	53,319382	0,105002
PRB1	PP	NSGA3	0,778354	1,284763	0,717559	53,234515	0,004883
PRB1	PP	NSGA2	0,783367	1,276542	0,719670	53,242715	0,010159
PRB1	MM	NSGA2	0,292723	3,416204	2,385377	51,795151	0,000000
PRB1	MM	NSGA3	0,292714	3,416308	2,385481	51,795047	0,000000

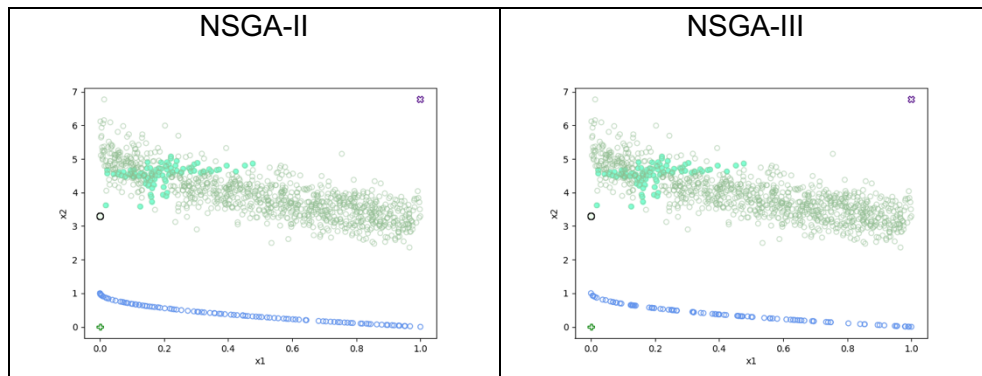
Fuente: elaboración propia

Figura A-6: Resultado de AGMOSIPD en ZDT1 con $\gamma = \text{“Muy Pequeño”}$.



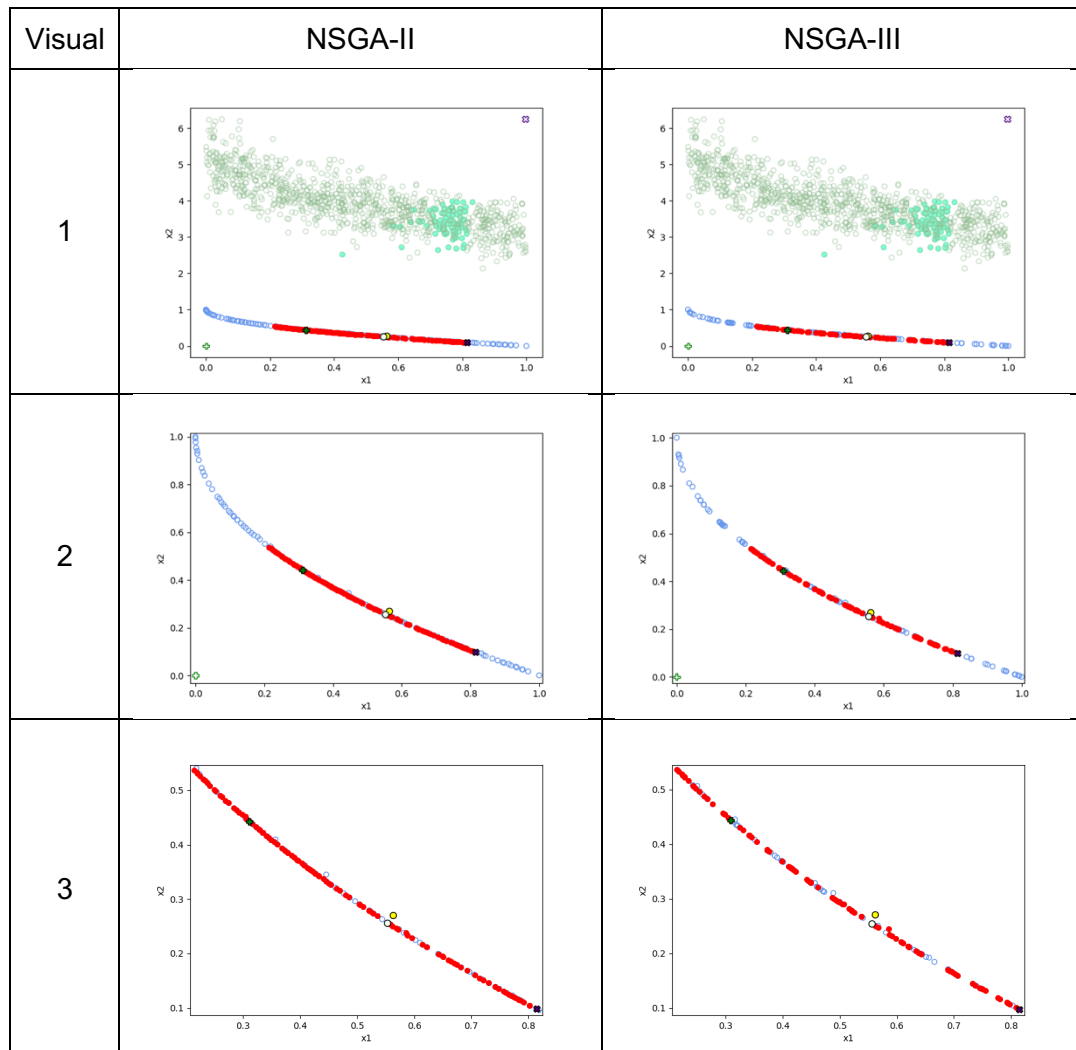
Fuente: elaboración propia

Figura A-7: Resultado de AGMOSIPD en ZDT1 con $\gamma = \text{“Grande”}$.



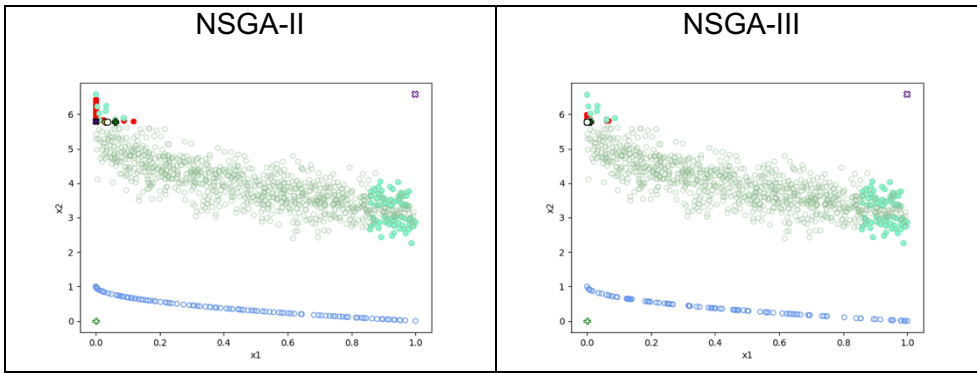
Fuente: elaboración propia

Figura A-8: Resultado de AGMOSIPD en ZDT1 con $\gamma = \text{“Pequeño”}$.



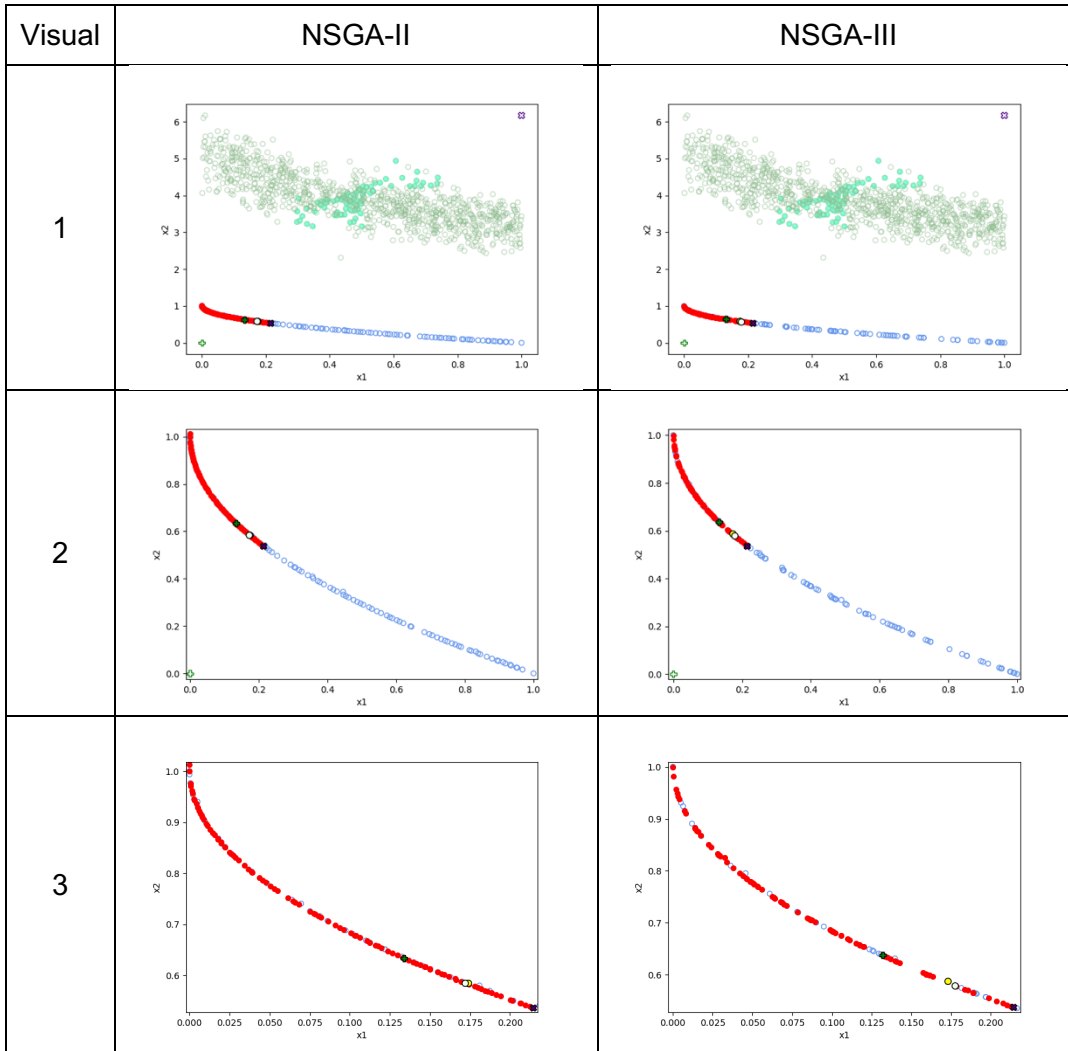
Fuente: elaboración propia

Figura A-9: Resultado de AGMOSIPD en ZDT1 con $\gamma = \text{“Muy Grande”}$.



Fuente: elaboración propia

Figura A-10: Resultado de AGMOSIPD en ZDT1 con $\gamma = \text{“Medio”}$.



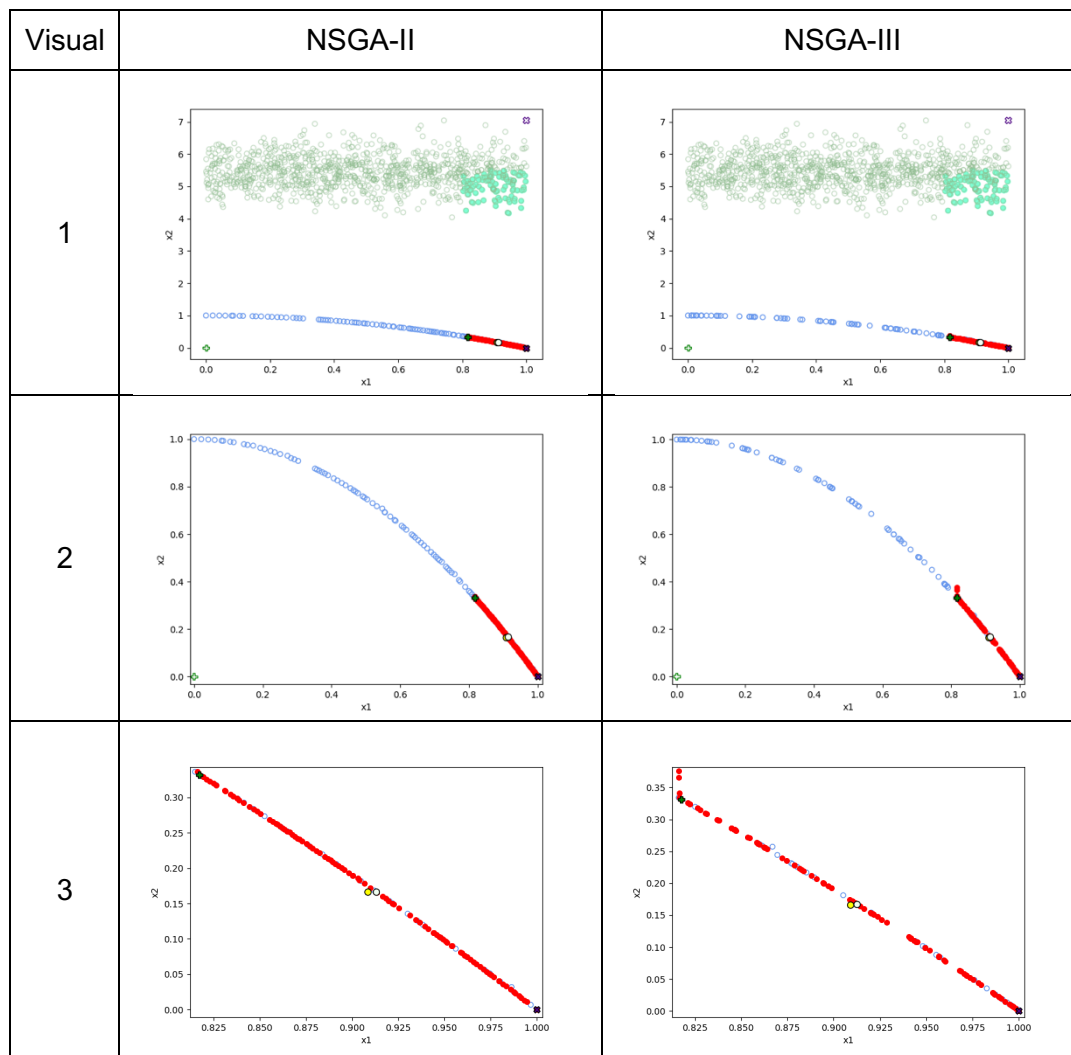
Fuente: elaboración propia

Tabla A-2: Soluciones del problema ZDT1 seleccionadas por AGMOSIPD.

PRB	PR	AG	X1	X2	DCI	DCN	DPM
ZDT1	MP	NSGA3	0,816415	0,096568	0,821763	6,426154	0,103689
ZDT1	MP	NSGA2	0,816169	0,101279	0,822091	6,421451	0,104925
ZDT1	MP	NSGA2	0,999991	0,000055	0,999658	6,520045	0,104925
ZDT1	MP	NSGA3	0,999994	0,000108	0,999661	6,519992	0,103689
ZDT1	MP	NSGA3	0,907296	0,047572	0,908204	6,473191	0,001188
ZDT1	MP	NSGA2	0,906072	0,048205	0,907018	6,472576	0,003177
ZDT1	PP	NSGA2	0,311337	0,442203	0,463871	5,842885	0,304720
ZDT1	PP	NSGA3	0,308969	0,444429	0,463910	5,840953	0,306358
ZDT1	PP	NSGA2	0,814146	0,097819	0,813576	6,149562	0,304720
ZDT1	PP	NSGA3	0,814259	0,097872	0,813689	6,149506	0,306358
ZDT1	PP	NSGA2	0,552956	0,256561	0,574743	6,004570	0,016633
ZDT1	PP	NSGA3	0,556100	0,254495	0,577187	6,006399	0,017545
ZDT1	MM	NSGA3	0,131895	0,637065	0,165289	5,602608	0,064619
ZDT1	MM	NSGA2	0,133912	0,634246	0,165365	5,605082	0,062987
ZDT1	MM	NSGA2	0,214263	0,537225	0,214263	5,689307	0,062987
ZDT1	MM	NSGA3	0,214226	0,537445	0,214226	5,689094	0,064619
ZDT1	MM	NSGA2	0,172100	0,585377	0,178709	5,647645	0,002020
ZDT1	MM	NSGA3	0,177696	0,578702	0,182423	5,653432	0,009729

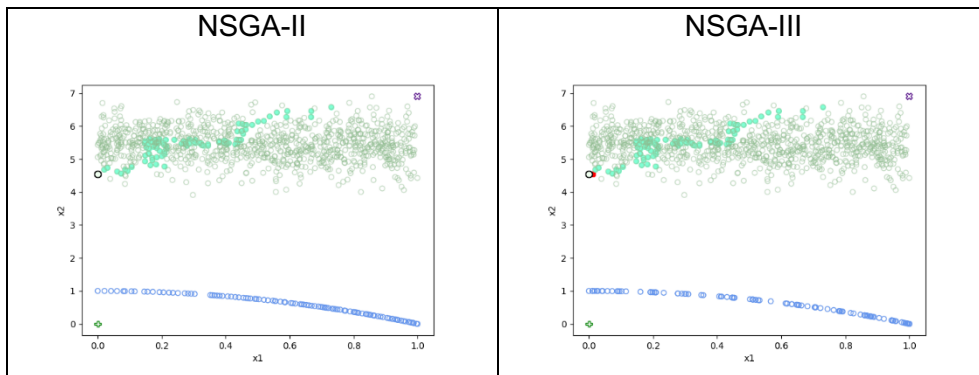
Fuente: elaboración propia

Figura A-11: Resultado de AGMOSIPD en ZDT2 para $\gamma = \text{“Muy Pequeño”}$.



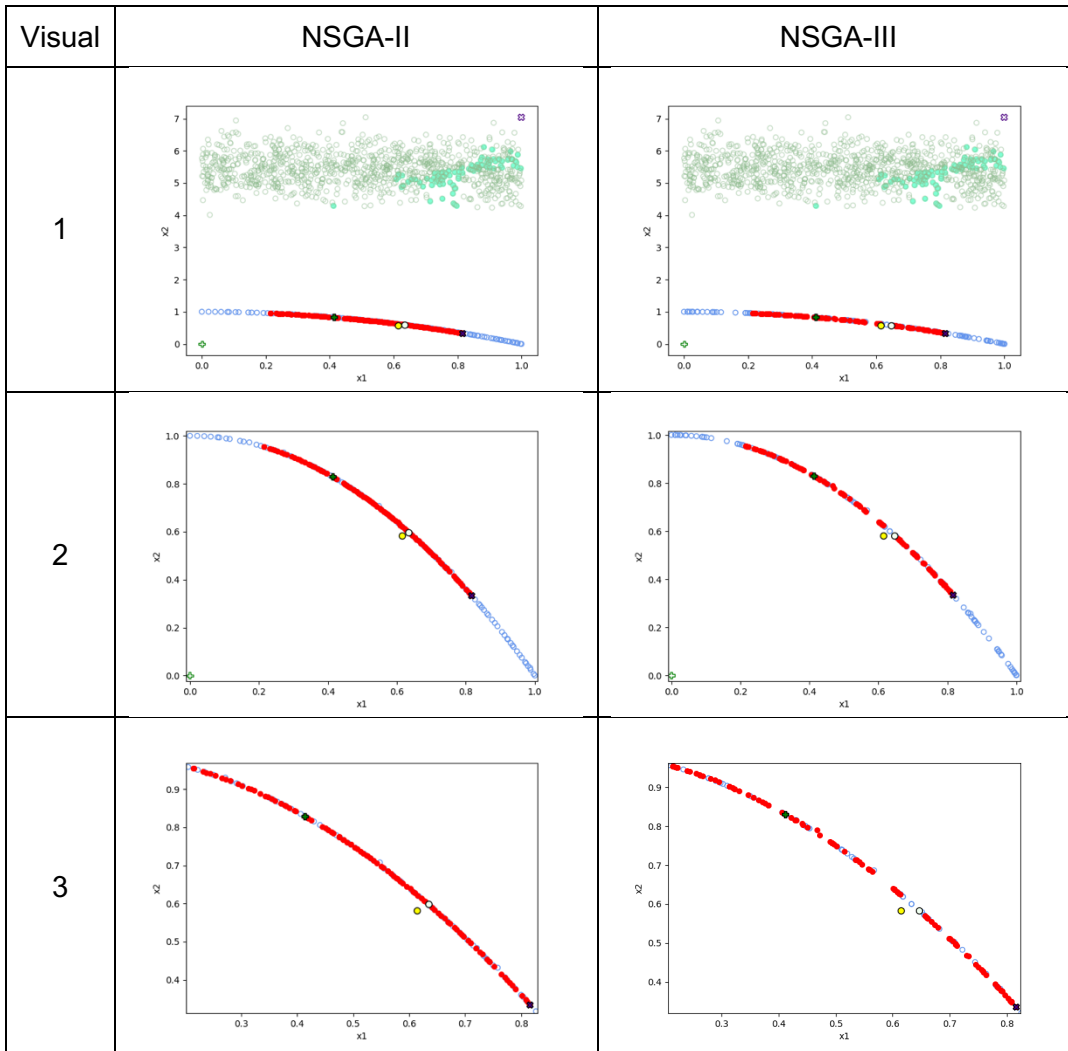
Fuente: elaboración propia

Figura A-12: Resultado de AGMOSIPD en ZDT2 para $\gamma = \text{“Grande”}$.



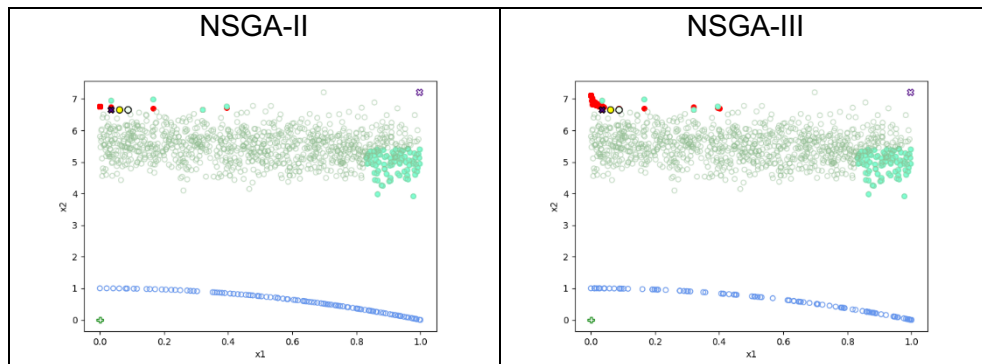
Fuente: elaboración propia

Figura A-13: Resultado de AGMOSIPD en ZDT2 para $\gamma = \text{“Pequeño”}$.



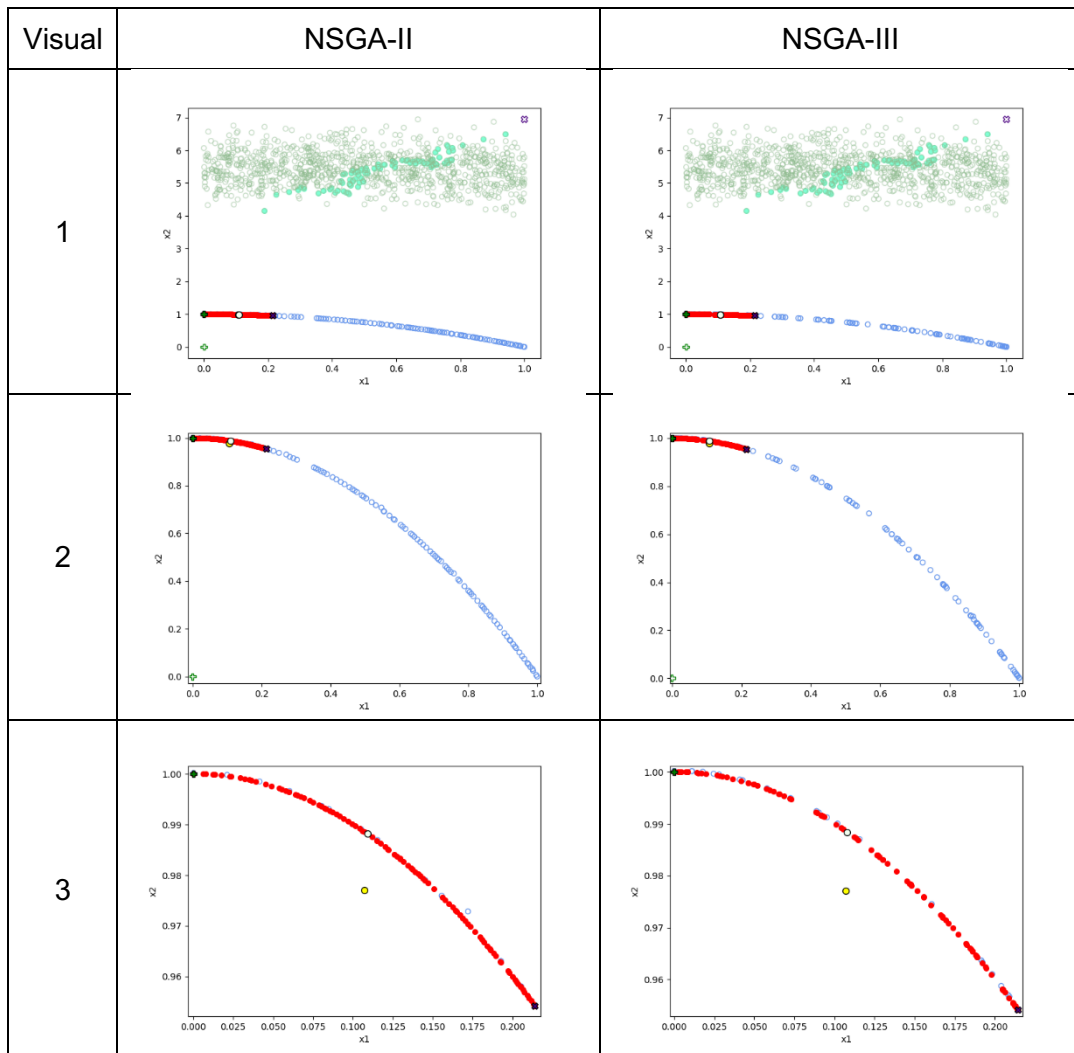
Fuente: elaboración propia

Figura A-14: Resultado de AGMOSIPD en ZDT2 para $\gamma = \text{“Muy Grande”}$.



Fuente: elaboración propia

Figura A-15: Resultado de AGMOSIPD en ZDT2 para $\gamma = \text{“Medio”}$.



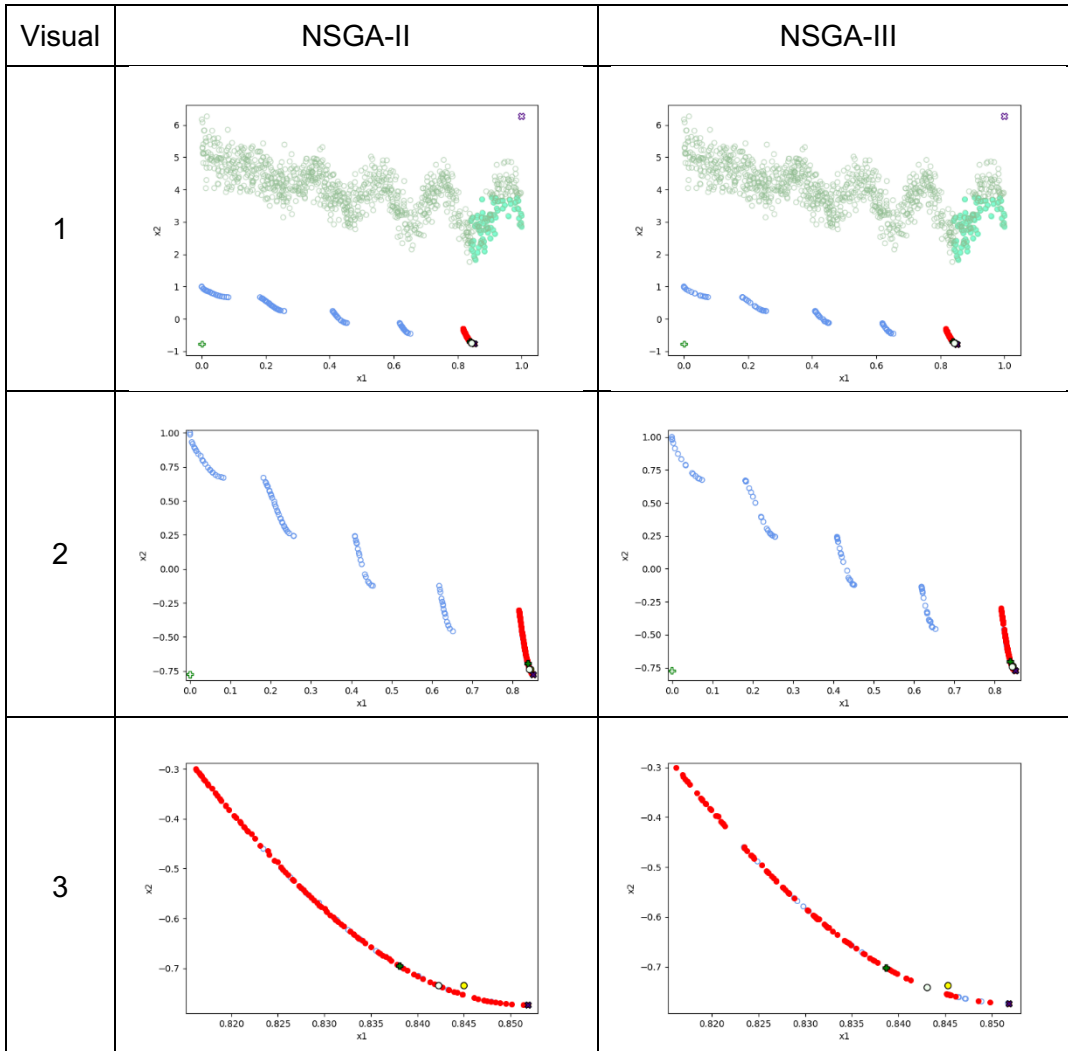
Fuente: elaboración propia

Tabla A-3: Soluciones del problema ZDT2 seleccionadas por AGMOSIPD.

PRB	PR	AG	X1	X2	DCI	DCN	DPM
ZDT2	MP	NSGA2	0,817142	0,332477	0,881593	6,711944	0,189635
ZDT2	MP	NSGA3	0,817749	0,331495	0,881725	6,712909	0,188991
ZDT2	MP	NSGA2	0,999986	0,000190	0,999418	7,041739	0,189635
ZDT2	MP	NSGA3	1,000000	0,000353	0,999432	7,041576	0,188991
ZDT2	MP	NSGA3	0,912602	0,167406	0,927207	6,875080	0,004012
ZDT2	MP	NSGA2	0,912899	0,166741	0,927408	6,875740	0,004354
ZDT2	PP	NSGA3	0,412151	0,830401	0,643998	6,238252	0,319428
ZDT2	PP	NSGA2	0,413322	0,829479	0,644152	6,239059	0,318760
ZDT2	PP	NSGA2	0,815683	0,334970	0,815148	6,708505	0,318760
ZDT2	PP	NSGA3	0,815673	0,335116	0,815138	6,708359	0,319428
ZDT2	PP	NSGA2	0,634397	0,597957	0,686252	6,453329	0,025363
ZDT2	PP	NSGA3	0,646399	0,582782	0,691721	6,467813	0,032487
ZDT2	MM	NSGA2	0,000000	1,000053	0,045881	6,026800	0,109538
ZDT2	MM	NSGA3	0,000000	1,000075	0,045920	6,026778	0,109547
ZDT2	MM	NSGA3	0,214227	0,954155	0,214227	6,040488	0,109547
ZDT2	MM	NSGA2	0,214217	0,954171	0,214217	6,040474	0,109538
ZDT2	MM	NSGA2	0,109070	0,988202	0,114255	6,021389	0,011262
ZDT2	MM	NSGA3	0,107921	0,988440	0,113236	6,021323	0,011353

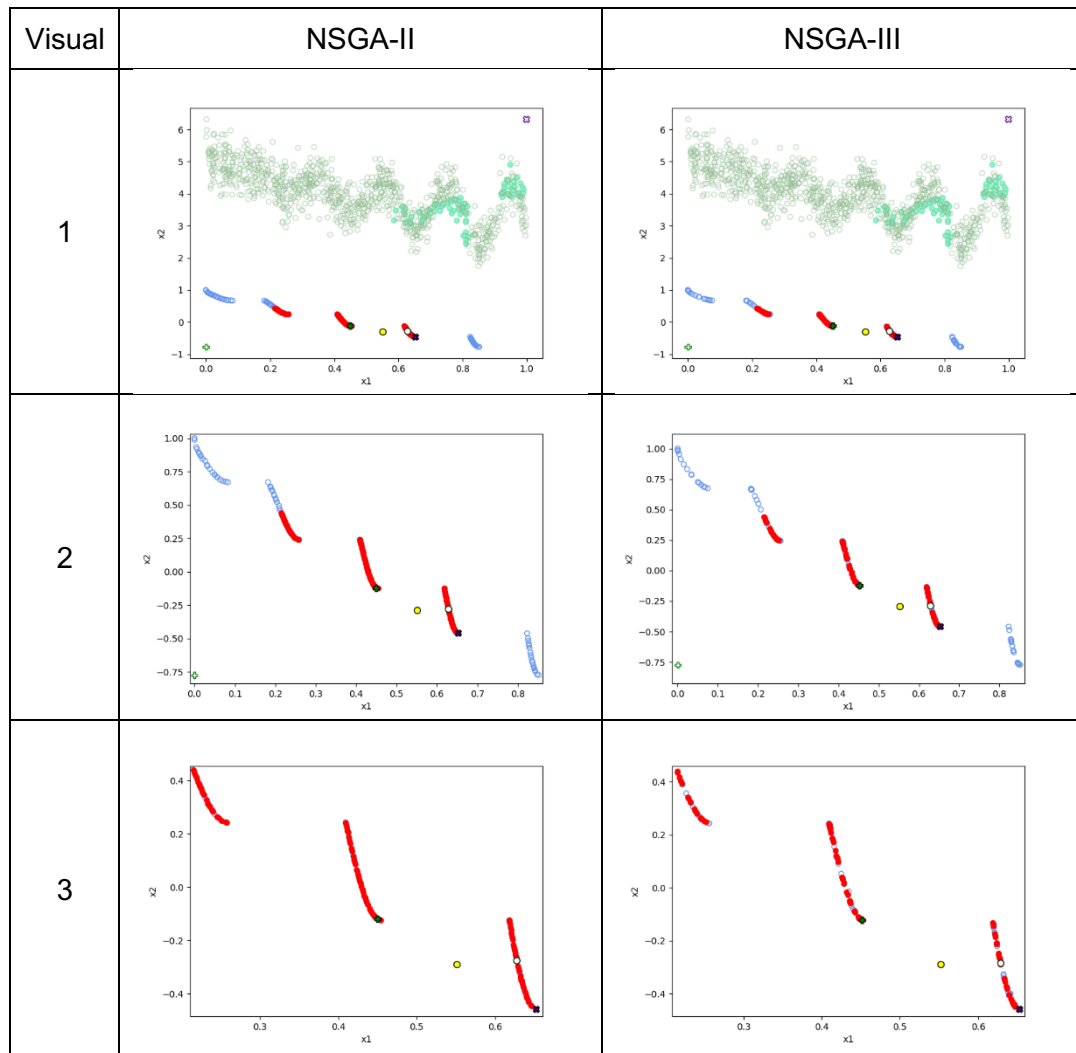
Fuente: elaboración propia

Figura A-16: Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{“Muy Pequeño”}$.



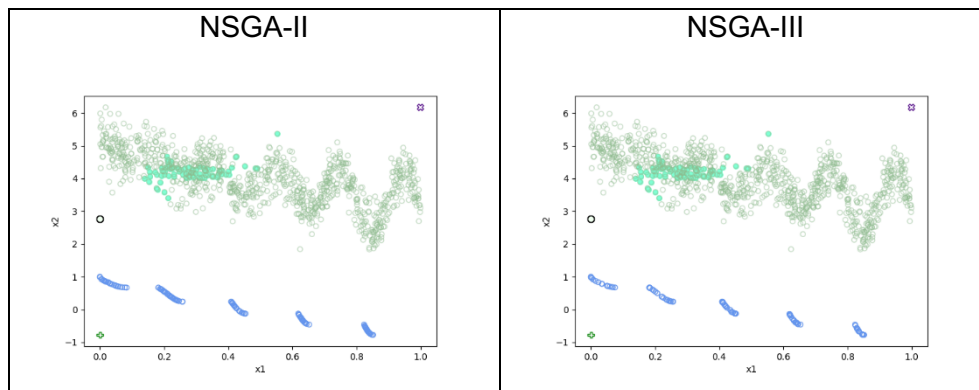
Fuente: elaboración propia

Figura A-17: Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{“Pequeño”}$.



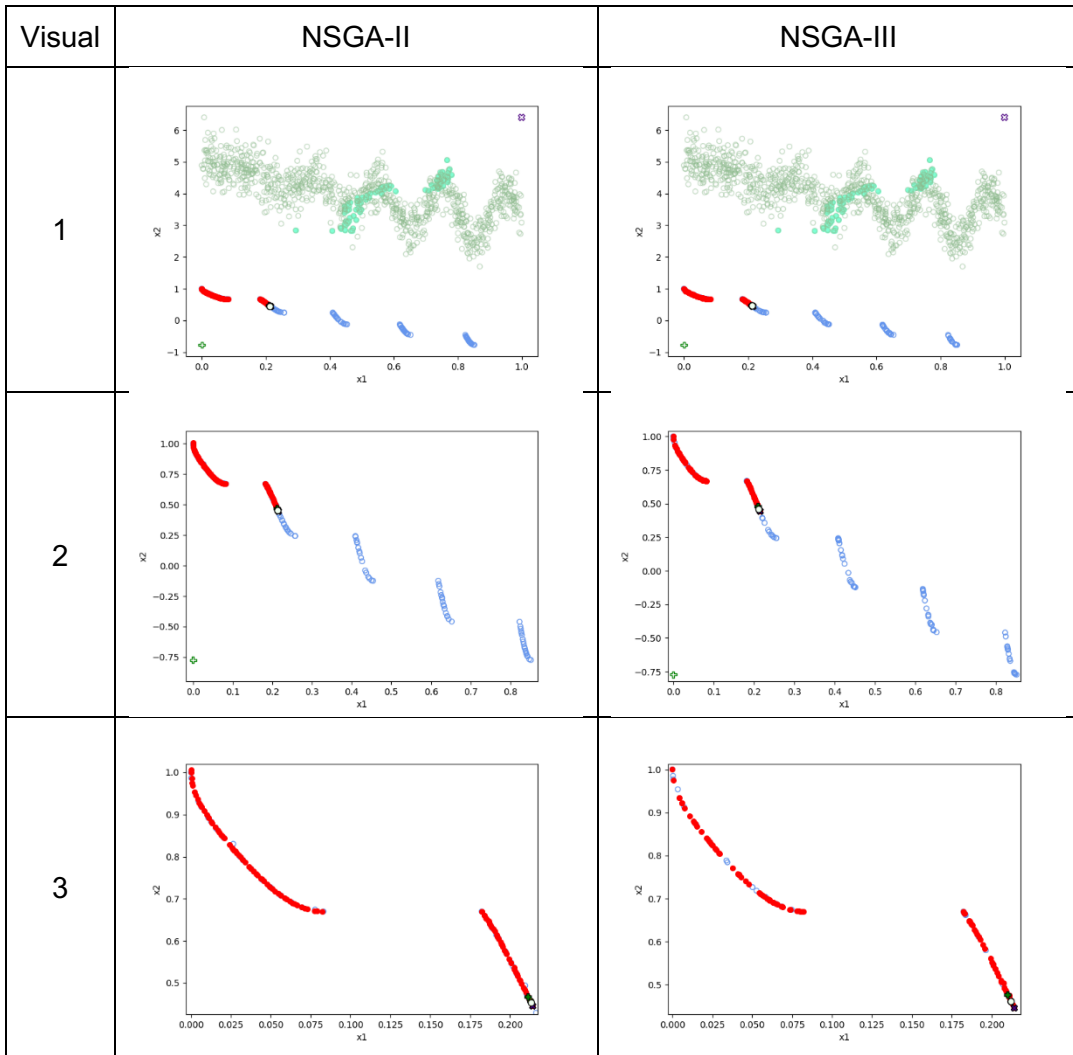
Fuente: elaboración propia

Figura A-18: Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{“Grande”}$.



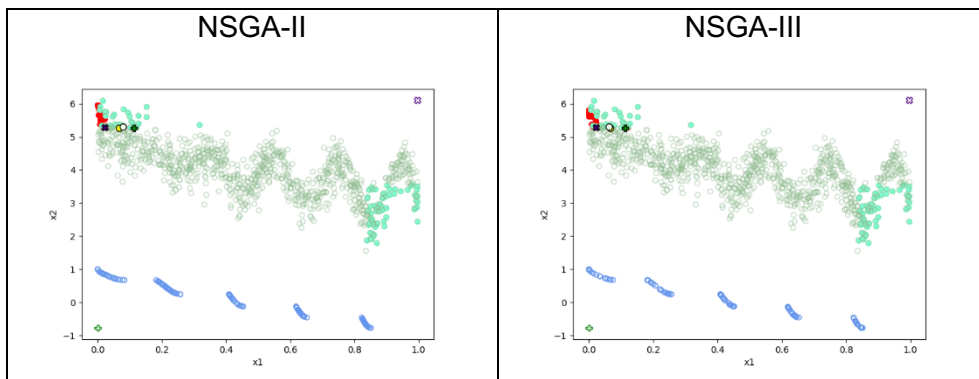
Fuente: elaboración propia

Figura A-19: Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{“Medio”}$.



Fuente: elaboración propia

Figura A-20: Resultado de AGMOSIPD en ZDT3 para $\gamma = \text{“Muy Grande”}$.



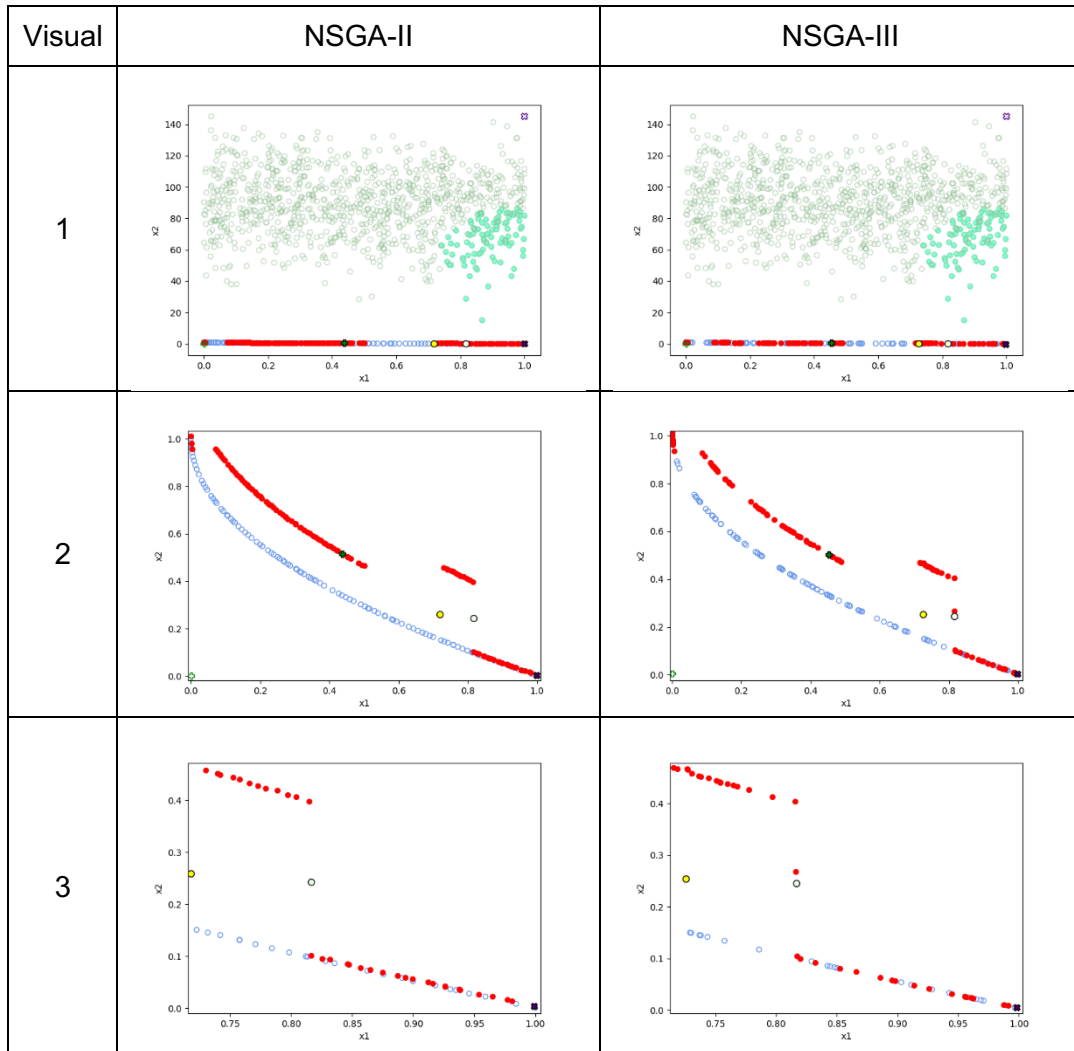
Fuente: elaboración propia

Tabla A-4: Soluciones del problema ZDT3 seleccionadas por AGMOSIPD.

PRB	PR	AG	X1	X2	DCI	DCN	DPM
ZDT3	MP	NSGA2	0,838091	-0,695362	0,841325	6,961729	0,039545
ZDT3	MP	NSGA3	0,838655	-0,701465	0,841341	6,967817	0,036476
ZDT3	MP	NSGA2	0,851861	-0,773243	0,851483	7,039286	0,039545
ZDT3	MP	NSGA3	0,851844	-0,773214	0,851465	7,039257	0,036476
ZDT3	MP	NSGA2	0,842237	-0,734384	0,842755	7,000645	0,002740
ZDT3	MP	NSGA3	0,843027	-0,740809	0,843271	7,007052	0,004121
ZDT3	PP	NSGA3	0,451397	-0,122357	0,561430	6,460143	0,195553
ZDT3	PP	NSGA2	0,450116	-0,120254	0,561843	6,458157	0,196897
ZDT3	PP	NSGA2	0,652437	-0,458100	0,651240	6,781491	0,196897
ZDT3	PP	NSGA3	0,652493	-0,457804	0,651295	6,781193	0,195553
ZDT3	PP	NSGA3	0,628698	-0,285592	0,650703	6,610503	0,076885
ZDT3	PP	NSGA2	0,627971	-0,275655	0,652787	6,600622	0,077877
ZDT3	MM	NSGA3	0,210028	0,477153	0,212405	5,977055	0,015976
ZDT3	MM	NSGA2	0,211344	0,466972	0,212438	5,986974	0,010859
ZDT3	MM	NSGA2	0,214153	0,445436	0,214153	6,007956	0,010859
ZDT3	MM	NSGA3	0,214161	0,445470	0,214161	6,007921	0,015976
ZDT3	MM	NSGA3	0,212116	0,461239	0,212702	5,992556	0,000076
ZDT3	MM	NSGA2	0,213104	0,453571	0,213259	6,000028	0,002657

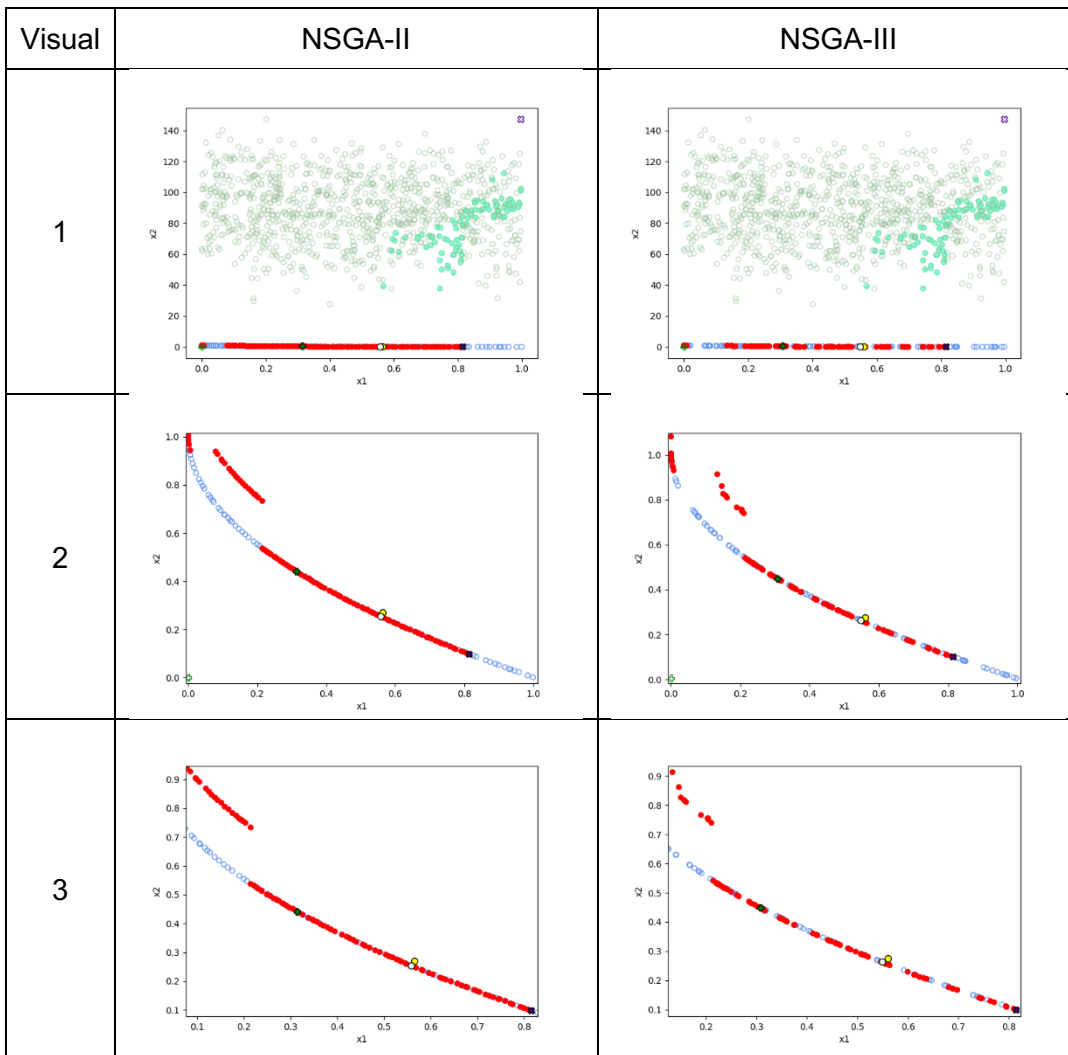
Fuente: elaboración propia

Figura A-21: Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{“Muy Pequeño”}$.



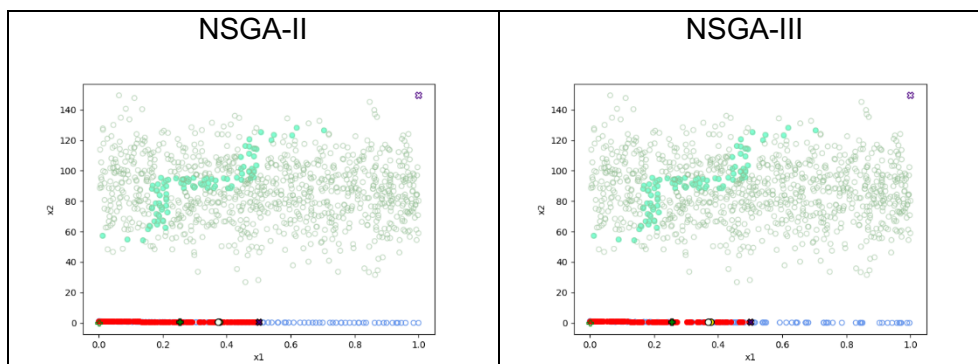
Fuente: elaboración propia

Figura A-22: Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{“Pequeño”}$.



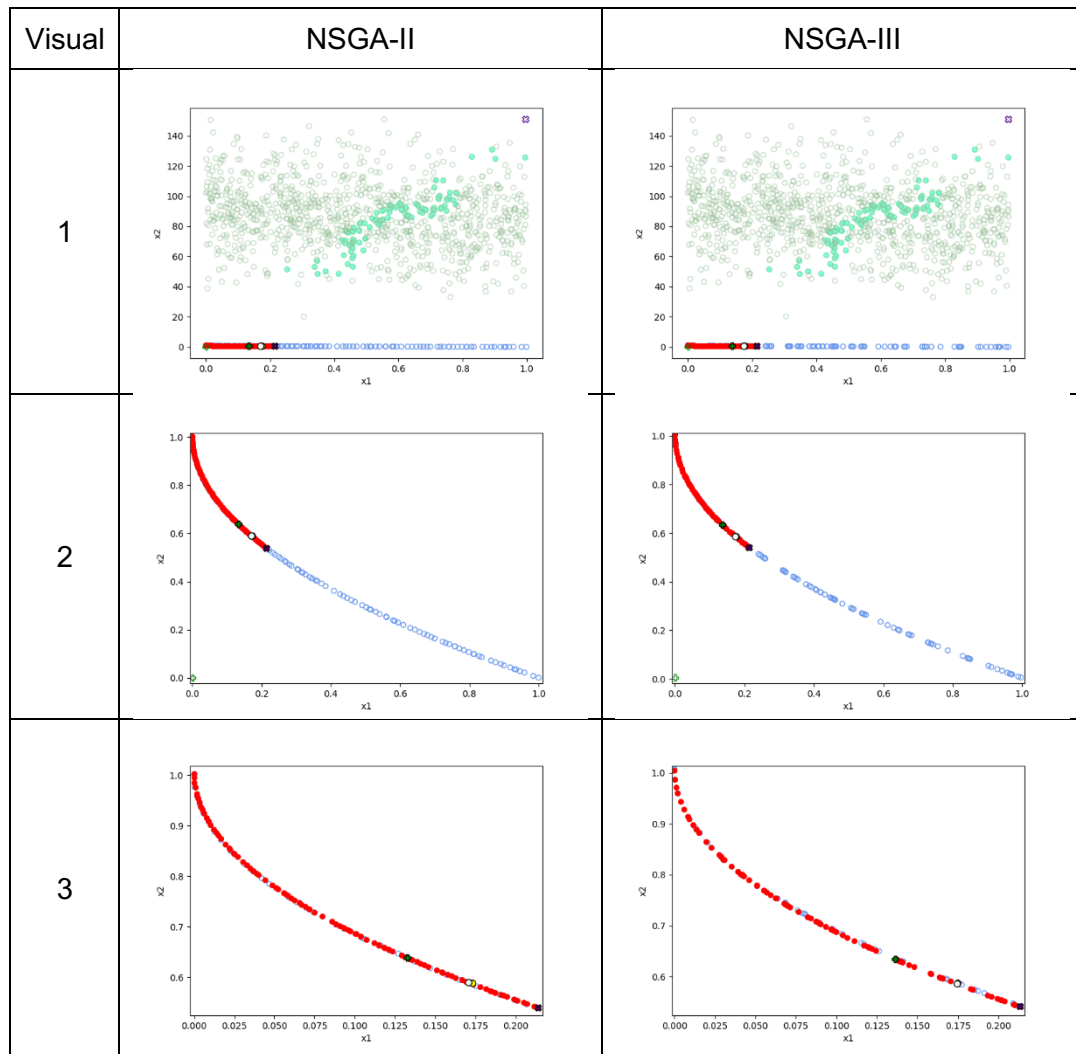
Fuente: elaboración propia

Figura A-23: Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{“Grande”}$.



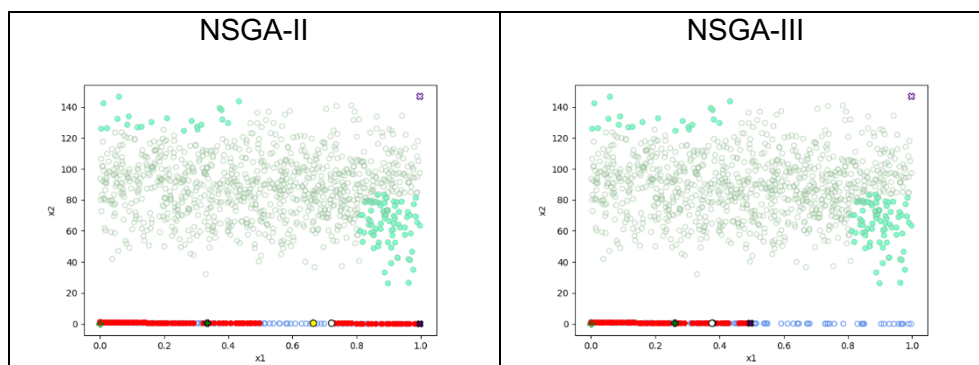
Fuente: elaboración propia

Figura A-24: Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{“Medio”}$.



Fuente: elaboración propia

Figura A-25: Resultado de AGMOSIPD en ZDT4 para $\gamma = \text{“Muy Grande”}$.



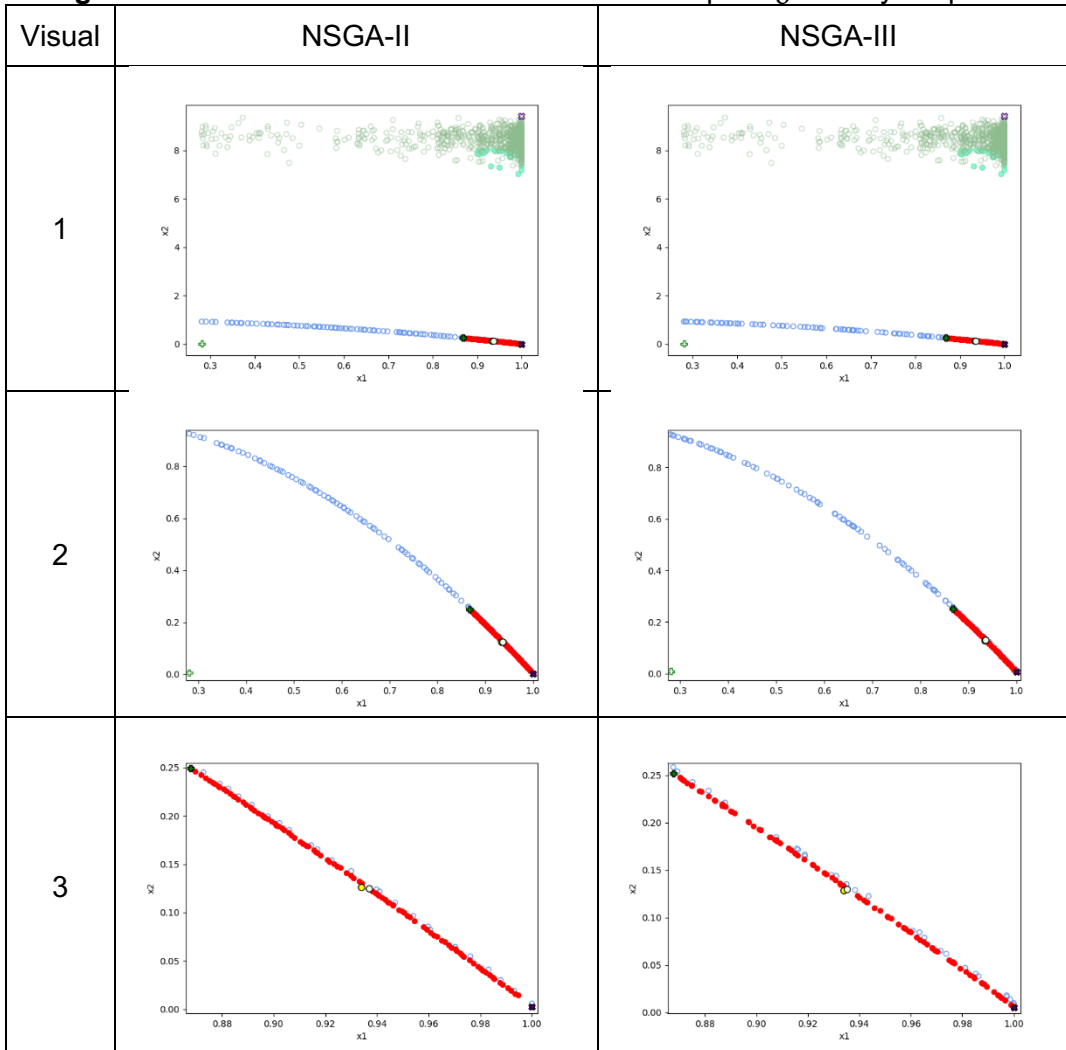
Fuente: elaboración propia

Tabla A-5: Soluciones del problema ZDT4 seleccionadas por AGMOSIPD.

PRB	PR	AG	X1	X2	DCI	DCN	DPM
ZDT4	MP	NSGA2	0,436892	0,514513	0,671880	144,472577	0,379641
ZDT4	MP	NSGA3	0,452911	0,502517	0,672816	144,484511	0,369042
ZDT4	MP	NSGA2	0,998995	0,004074	0,998995	144,981920	0,379641
ZDT4	MP	NSGA3	0,998084	0,004970	0,998084	144,981023	0,369042
ZDT4	MP	NSGA3	0,816391	0,245744	0,851156	144,740365	0,091244
ZDT4	MP	NSGA2	0,816236	0,243391	0,850596	144,742719	0,099570
ZDT4	PP	NSGA2	0,314195	0,440861	0,464494	146,865572	0,302925
ZDT4	PP	NSGA3	0,307510	0,448729	0,464687	146,857735	0,307413
ZDT4	PP	NSGA2	0,814213	0,098756	0,814213	147,206202	0,302925
ZDT4	PP	NSGA3	0,814107	0,100345	0,814107	147,204613	0,307413
ZDT4	PP	NSGA2	0,557896	0,254649	0,579267	147,050851	0,016419
ZDT4	PP	NSGA3	0,548152	0,263011	0,571779	147,042519	0,017118
ZDT4	MM	NSGA3	0,136328	0,634432	0,164724	150,296441	0,060099
ZDT4	MM	NSGA2	0,132717	0,638500	0,165050	150,292394	0,063616
ZDT4	MM	NSGA2	0,213712	0,540380	0,213712	150,390069	0,063616
ZDT4	MM	NSGA3	0,213131	0,541973	0,213131	150,388479	0,060099
ZDT4	MM	NSGA3	0,174391	0,585871	0,179832	150,344789	0,002356
ZDT4	MM	NSGA2	0,170739	0,589573	0,177684	150,341107	0,002479

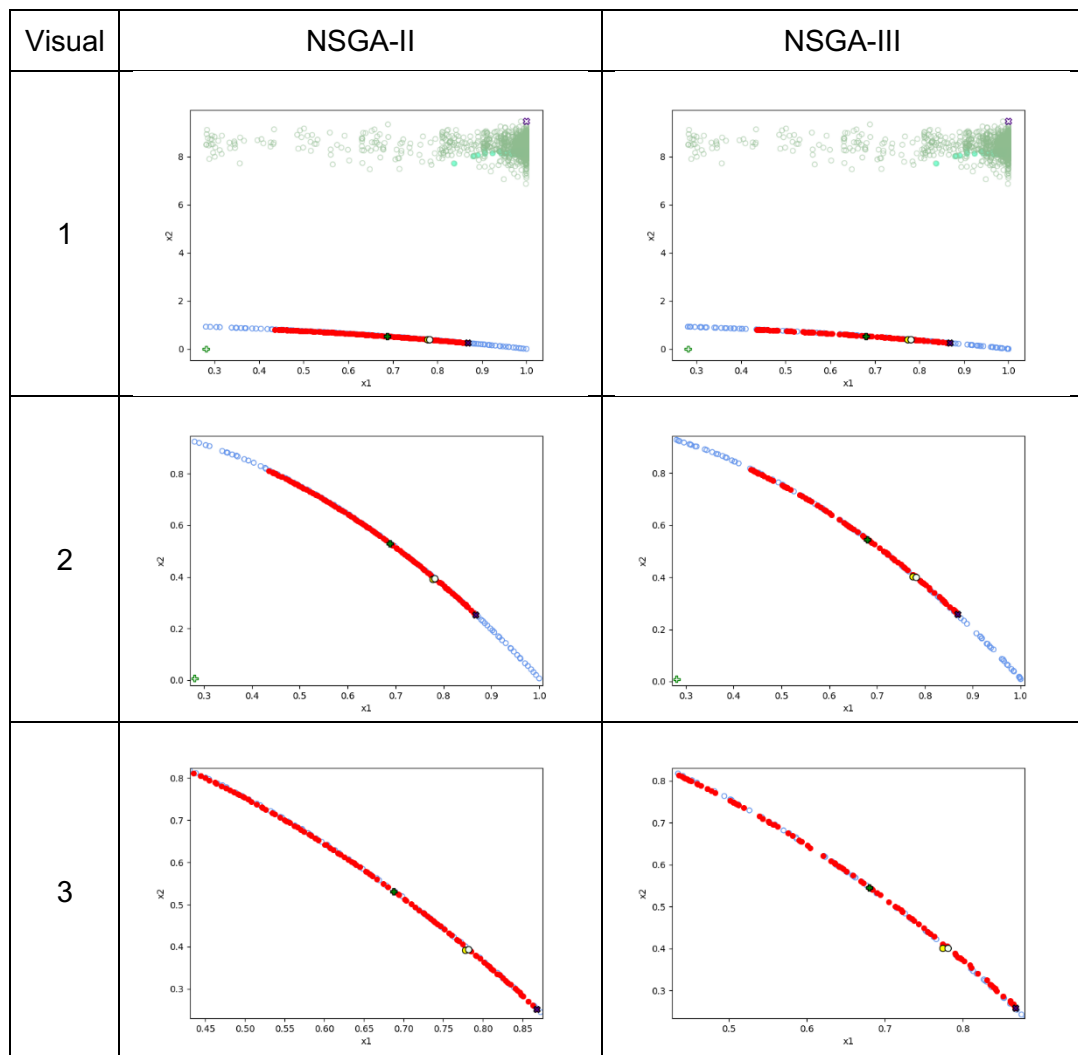
Fuente: elaboración propia

Figura A-26: Resultado de AGMOSIPD en ZDT6 para $\gamma =$ "Muy Pequeño".



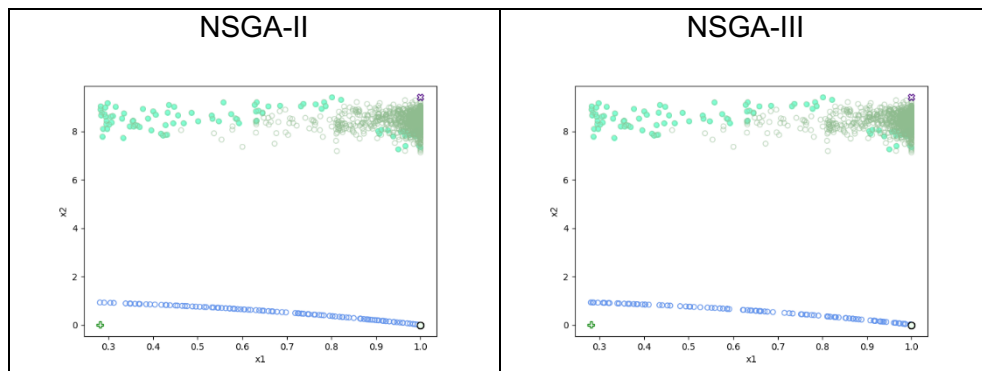
Fuente: elaboración propia

Figura A-27: Resultado de AGMOSIPD en ZDT6 para $y = \text{“Pequeño”}$.



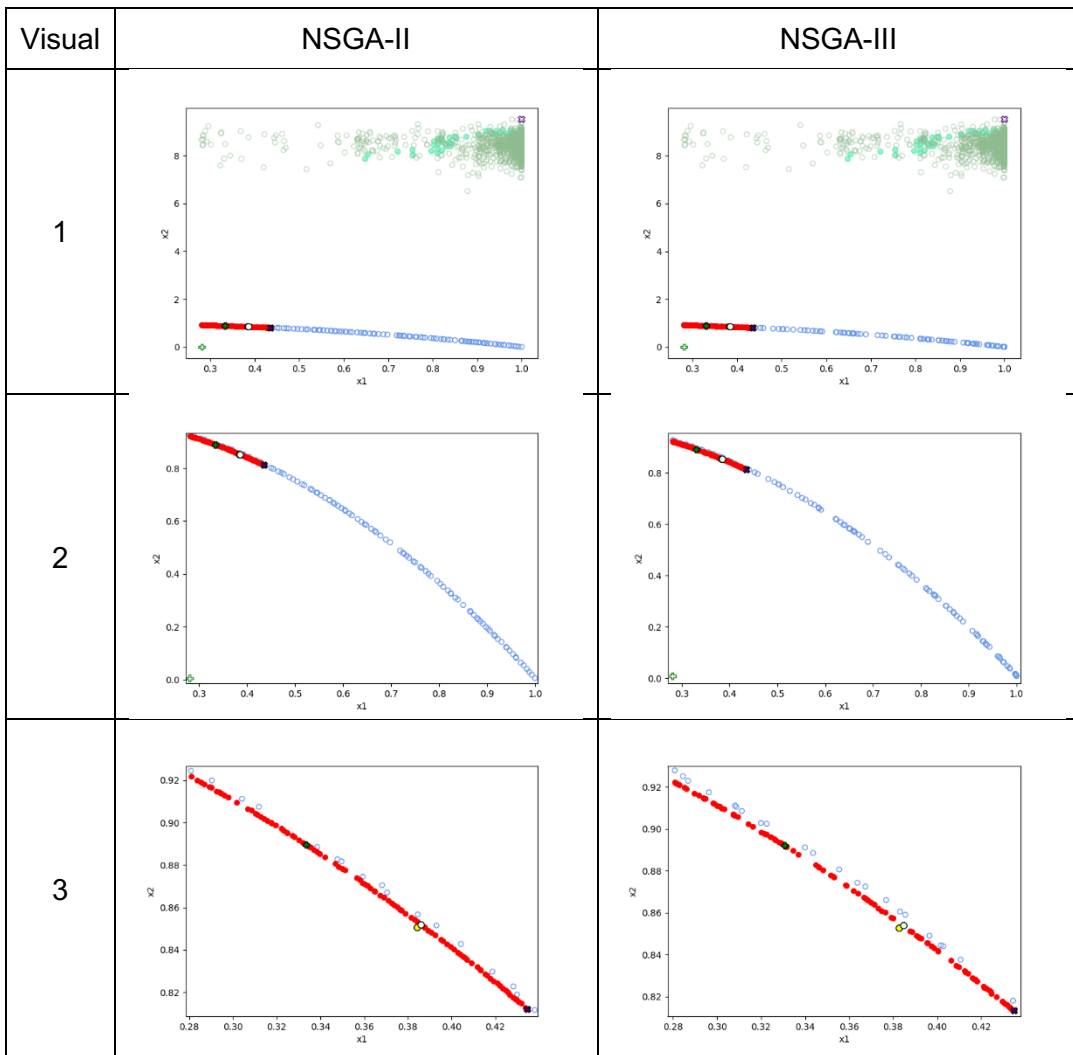
Fuente: elaboración propia

Figura A-28: Resultado de AGMOSIPD en ZDT6 para $y = \text{“Grande”}$.



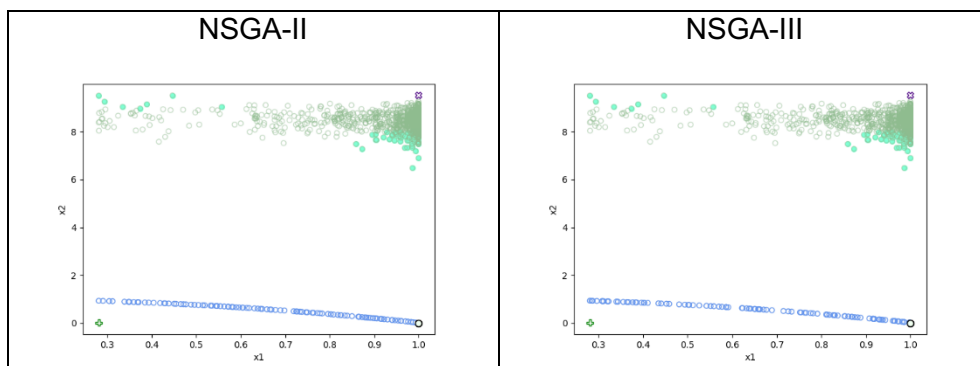
Fuente: elaboración propia

Figura A-29: Resultado de AGMOSIPD en ZDT6 para $\gamma = \text{“Medio”}$.



Fuente: elaboración propia

Figura A-30: Resultado de AGMOSIPD en ZDT6 para $\gamma = \text{“Muy Grande”}$.



Fuente: elaboración propia

Tabla A-6: Soluciones del problema ZDT6 seleccionadas por AGMOSIPD.

PRB	PR	AG	X1	X2	DCI	DCN	DPM
ZDT6	MP	NSGA3	0,867812	0,252039	0,636693	9,158695	0,139868
ZDT6	MP	NSGA2	0,867815	0,249416	0,636903	9,161318	0,140108
ZDT6	MP	NSGA2	0,999999	0,002335	0,719207	9,407445	0,140108
ZDT6	MP	NSGA3	0,999973	0,005493	0,719182	9,404287	0,139868
ZDT6	MP	NSGA3	0,935293	0,129624	0,666169	9,280381	0,001643
ZDT6	MP	NSGA2	0,936820	0,124641	0,667332	9,285354	0,003164
ZDT6	PP	NSGA3	0,679983	0,544727	0,490381	8,921528	0,171085
ZDT6	PP	NSGA2	0,687394	0,530719	0,492606	8,935265	0,165894
ZDT6	PP	NSGA2	0,867652	0,252170	0,586548	9,209295	0,165894
ZDT6	PP	NSGA3	0,867888	0,258769	0,586278	9,202693	0,171085
ZDT6	PP	NSGA2	0,781776	0,394092	0,520398	9,069048	0,005010
ZDT6	PP	NSGA3	0,780832	0,401134	0,519125	9,062030	0,006923
ZDT6	MM	NSGA3	0,330560	0,892084	0,093062	8,647063	0,065326
ZDT6	MM	NSGA2	0,333418	0,889615	0,093774	8,649305	0,063791
ZDT6	MM	NSGA2	0,434686	0,812011	0,153911	8,719529	0,063791
ZDT6	MM	NSGA3	0,434905	0,813458	0,154130	8,718070	0,065326
ZDT6	MM	NSGA2	0,385860	0,851916	0,112406	8,683025	0,002118
ZDT6	MM	NSGA3	0,384655	0,854097	0,111546	8,680935	0,002335

Fuente: elaboración propia

B. Anexo: Código de AGMOSIPD

Los diferentes códigos utilizados para cada una de las ejecuciones desarrolladas en esta investigación pueden ser consultados en el siguiente enlace:

<https://github.com/diazguerra1103/AGMOSIPD>

Bibliografía

- Aguilar Arroyo, E. A. (2023). *Un nuevo sistema inmune artificial para problemas de optimización multi-objetivo* [Tesis de maestría, Centro de Investigación y Estudios Avanzados del Instituto Politécnico Nacional].
<https://repositorio.cinvestav.mx/bitstream/handle/cinvestav/4776/SSIT0018189.pdf?sequence=1>
- Bechikh, S., Kessentini, M., Said, L. B., & Ghédira, K. (2015). Preference Incorporation in Evolutionary Multiobjective Optimization. En *Advances in Computers* (Vol. 98, pp. 141-207). Elsevier. <https://doi.org/10.1016/bs.adcom.2015.03.001>
- Blank, J., & Deb, K. (2020). Pymoo: Multi-Objective Optimization in Python. *IEEE Access*, 8, 89497-89509. <https://doi.org/10.1109/ACCESS.2020.2990567>
- Bonissone, S. R. (2001). *Evolutionary algorithms for multi-objective optimization: Fuzzy preference aggregation and multisexual EAs* (B. Bosacchi, D. B. Fogel, & J. C. Bezdek, Eds.; pp. 157-164). <https://doi.org/10.1117/12.448334>
- Branke, J., Kaußler, T., & Schmeck, H. (2001). Guidance in evolutionary multi-objective optimization. *Advances in Engineering Software*, 32(6), 499-507. [https://doi.org/10.1016/S0965-9978\(00\)00110-1](https://doi.org/10.1016/S0965-9978(00)00110-1)
- Choon, O. H., & Tilahun, S. L. (2011). Integration fuzzy preference in genetic algorithm to solve multiobjective optimization problems. *Far East Math. Sci*, 55, 165-179.
- Coello, C. A. C. (2019). *Introducción a la Computación Evolutiva (Notas de Curso)* [Notas de Curso].

- https://gc.scalahed.com/recursos/files/r161r/w25199w/s1_introduccionalacomputacionevolutiva.pdf
- Coello, C. A. C., Lamont, G. B., & Veldhuizen, D. A. V. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems Second Edition* (2.^a ed.). Springer.
<http://tinyurl.com/4b2cp7ef>
- Cortez, V. F., Cruz, D. V., & Margolis, P. E. L. (2019). Optimización de Portafolios de Inversión con Algoritmos Genéticos. *Revista de Investigación en Ciencias Contables y Administrativas*, 4(2), Article 2.
- Cuartas Torres, B. A. C. (2009). *Metodología para la optimización de múltiples objetivos basada en ag y uso de preferencias* [Tesis de maestría].
<https://repositorio.unal.edu.co/handle/unal/70080>
- Cvetković, D., & Coello, C. A. C. (2005). Human Preferences and their Applications in Evolutionary Multi—Objective Optimization. En Y. Jin (Ed.), *Knowledge Incorporation in Evolutionary Computation* (Vol. 167, pp. 479-502). Springer Berlin Heidelberg.
https://doi.org/10.1007/978-3-540-44511-1_22
- Deb, K., & Chaudhuri, S. (2005). I-EMO: An Interactive Evolutionary Multi-objective Optimization Tool. En S. K. Pal, S. Bandyopadhyay, & S. Biswas (Eds.), *Pattern Recognition and Machine Intelligence* (Vol. 3776, pp. 690-695). Springer Berlin Heidelberg. https://doi.org/10.1007/11590316_111
- Deb, K., & Jain, H. (2014). An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Transactions on Evolutionary Computation*, 18(4), 577-601.
<https://doi.org/10.1109/TEVC.2013.2281535>
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182-197.
<https://doi.org/10.1109/4235.996017>

- Duarte, O. G. (1999). Sistemas de lógica difusa: Fundamentos. *Ingeniería e Investigación*, 42, 22-30. <https://doi.org/10.15446/ing.investig.n42.21065>
- Duarte, O., Sarmiento, C., Barrera, M., Márquez, M., Culma, J. E., & Ramirez, J. J. (2022). *Modelos matemáticos para la gestión curricular* (1.ª ed.). Universidad Nacional de Colombia, Facultad de Ingeniería. <https://repositorio.unal.edu.co/handle/unal/83381>
- Fonseca, C., & Fleming, P. (1999). Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization. *the fifth Intl conference on Genetic Algorithms*, 93.
- Ishibuchi, H., Imada, R., Setoguchi, Y., & Nojima, Y. (2016). Performance comparison of NSGA-II and NSGA-III on various many-objective test problems. *2016 IEEE Congress on Evolutionary Computation (CEC)*, 3045-3052. <https://doi.org/10.1109/CEC.2016.7744174>
- Jamwal, P. K., Abdikenov, B., & Hussain, S. (2019). Evolutionary Optimization Using Equitable Fuzzy Sorting Genetic Algorithm (EFSGA). *IEEE Access*, 7, 8111-8126. <https://doi.org/10.1109/ACCESS.2018.2890274>
- Jin, Y., & Sendhoff, B. (2002). Incorporation of fuzzy preferences into evolutionary multiobjective optimization. En *4th Asia-Pacific Conference on Simulated Evolution and Learning* (Vol. 1, pp. 26-30).
- Kaci, S. (2011). *Working with Preferences: Less Is More*. Springer Science & Business Media.
- Kahneman, D. (2012). *Pensar rápido, pensar despacio*. Debate. <http://tinyurl.com/4rf8zpj>
- Kim, J.-H., Han, J.-H., Kim, Y.-H., Choi, S.-H., & Kim, E.-S. (2012). Preference-Based Solution Selection Algorithm for Evolutionary Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 16(1), 20-34. <https://doi.org/10.1109/TEVC.2010.2098412>
- Kingsley, D. C. (2006). Preference Uncertainty, Preference Refinement and Paired Comparison Choice Experiments. *University of Colorado, Boulder*.

- Lai, G., Liao, M., & Li, K. (2021). Empirical Studies on the Role of the Decision Maker in Interactive Evolutionary Multi-Objective Optimization. *2021 IEEE Congress on Evolutionary Computation (CEC)*, 185-192. <https://doi.org/10.1109/CEC45853.2021.9504980>
- Leyva-Lopez, J. C., & Aguilera-Contreras, M. A. (2005). A Multiobjective Evolutionary Algorithm for Deriving Final Ranking from a Fuzzy Outranking Relation. En C. A. Coello Coello, A. Hernández Aguirre, & E. Zitzler (Eds.), *Evolutionary Multi-Criterion Optimization* (Vol. 3410, pp. 235-249). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-31880-4_17
- Li, B., Li, J., Tang, K., & Yao, X. (2015). Many-Objective Evolutionary Algorithms: A Survey. *ACM Computing Surveys*, 48(1), 1-35. <https://doi.org/10.1145/2792984>
- Li, J., Li, Y., & Wang, Y. (2021). Fuzzy Inference NSGA-III Algorithm-Based Multi-Objective Optimization for Switched Reluctance Generator. *IEEE Transactions on Energy Conversion*, 36(4), 3578-3581. <https://doi.org/10.1109/TEC.2021.3099961>
- Li, K., Chen, R., Min, G., & Yao, X. (2018). Integration of Preferences in Decomposition Multiobjective Optimization. *IEEE Transactions on Cybernetics*, 48(12), 3359-3370. <https://doi.org/10.1109/TCYB.2018.2859363>
- Li, K., Chen, R., Savic, D., & Yao, X. (2019). Interactive Decomposition Multiobjective Optimization Via Progressively Learned Value Functions. *IEEE Transactions on Fuzzy Systems*, 27(5), 849-860. <https://doi.org/10.1109/TFUZZ.2018.2880700>
- Li, K., Liao, M., Deb, K., Min, G., & Yao, X. (2020). Does Preference Always Help? A Holistic Study on Preference-Based Evolutionary Multiobjective Optimization Using Reference Points. *IEEE Transactions on Evolutionary Computation*, 24(6), 1078-1096. <https://doi.org/10.1109/TEVC.2020.2987559>
- Lichtenstein, S., & Slovic, P. (Eds.). (2006). *The Construction of Preference* (1.^a ed.). Cambridge University Press. <https://doi.org/10.1017/CBO9780511618031>

- Luo, B., Lin, L., & Zhong, S. (2018). PGA/MOEA/D: A preference-guided evolutionary algorithm for multi-objective decision-making problems with interval-valued fuzzy preferences. *International Journal of Systems Science*, 49(3), 595-616. <https://doi.org/10.1080/00207721.2017.1412537>
- Pedrycz, W., Ekel, P., & Parreiras, R. (2011). *Fuzzy Multicriteria Decision-Making: Models, Methods and Applications*. John Wiley & Sons.
- Rachmawati, L., & Srinivasan, D. (2006). Preference Incorporation in Multi-objective Evolutionary Algorithms: A Survey. *2006 IEEE International Conference on Evolutionary Computation*, 962-968. <https://doi.org/10.1109/CEC.2006.1688414>
- Ramakrishnan, S., & Hasan, Y. A. (2013). Fuzzy preference-based multi-objective optimization method. *Artificial Intelligence Review*, 39(2), 165-181. <https://doi.org/10.1007/s10462-011-9264-4>
- REAL ACADEMIA ESPAÑOLA. (2023). *Diccionario de la lengua española* (23.^a ed.). <https://dle.rae.es>
- Rosenthal, R. E. (1984). Principles of multiobjective optimization. *Naval Postgraduate School*.
- Santana, L. V. S., & Coello, C. A. C. (2006). Una introducción a la Computación Evolutiva y alguna de sus aplicaciones en Economía y Finanzas. *Revista de Métodos Cuantitativos para la Economía y la Empresa*, 2, páginas 3 a 26-páginas 3 a 26. <https://doi.org/10.46661/revmetodoscuanteconempresa.2057>
- Shen, X., Guo, Y., Chen, Q., & Hu, W. (2010). A multi-objective optimization evolutionary algorithm incorporating preference information based on fuzzy logic. *Computational Optimization and Applications*, 46(1), 159-188. <https://doi.org/10.1007/s10589-008-9189-2>
- Shen, X., Li, T., & Zhang, M. (2009). A Fuzzy Multi-objective Optimization Evolutionary Algorithm Incorporating Preference Information. *2009 Second International Symposium*

- on Knowledge Acquisition and Modeling*, 143-146.
<https://doi.org/10.1109/KAM.2009.12>
- Smith, R., Mesa, O., Dyner, I., Jaramillo, P., Poveda, G., & Valencia, D. (2000). *Decisiones con Múltiples Objetivos e Incertidumbre* (2.^a ed.). Universidad Nacional de Colombia.
- Taylor, K. P. (2022). *Preference Learning for Multi-objective Optimisation Problems* [Tesis de doctorado].
- Thiele, L., Miettinen, K., Korhonen, P. J., & Molina, J. (2009). A Preference-Based Evolutionary Algorithm for Multi-Objective Optimization. *Evolutionary Computation*, 17(3), 411-436.
<https://doi.org/10.1162/evco.2009.17.3.411>
- Tomczyk, M. K., & Kadziński, M. (2020). On the elicitation of indirect preferences in interactive evolutionary multiple objective optimization. *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*, 569-577.
<https://doi.org/10.1145/3377930.3389826>
- Tversky, A., & Kahneman, D. (1974). Judgment under Uncertainty: Heuristics and Biases: Biases in judgments reveal some heuristics of thinking under uncertainty. *Science*, 185(4157), 1124-1131. <https://doi.org/10.1126/science.185.4157.1124>
- Xin, B., Chen, L., Chen, J., Ishibuchi, H., Hirota, K., & Liu, B. (2018). Interactive Multiobjective Optimization: A Review of the State-of-the-Art. *IEEE Access*, 6, 41256-41279.
<https://doi.org/10.1109/ACCESS.2018.2856832>
- Xiong, J., Tan, X., Yang, K., & Chen, Y. (2013). Fuzzy Group Decision Making for Multiobjective Problems: Tradeoff between Consensus and Robustness. *Journal of Applied Mathematics*, 2013, 1-9. <https://doi.org/10.1155/2013/657978>
- Yoon, K. P., & Kim, W. K. (2017). The behavioral TOPSIS. *Expert Systems with Applications*, 89, 266-272. <https://doi.org/10.1016/j.eswa.2017.07.045>

Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, 8(2), 173-195.
<https://doi.org/10.1162/106365600568202>