

SEP

TNM

INSTITUTO TECNOLÓGICO DE TIJUANA

**DIVISIÓN DE ESTUDIOS DE POSGRADO E
INVESTIGACIÓN**



**OPTIMIZACIÓN DE PROBLEMAS DINAMICOS EN
PRESENCIA DE UN GRAN NUMERO DE OBJETIVOS Y
PREFERENCIAS DEL DECISOR**

TRABAJO DE TESIS

Presentado por
M.C.C TEODORO EDUARDO MACIAS ESCOBAR

Para Obtener el Grado de
DOCTOR EN CIENCIAS EN COMPUTACIÓN

Director de Tesis
DRA. LAURA CRUZ REYES

Co-Director de Tesis
DR. BERNABÉ DORRONSORO DÍAZ

Tijuana, BC, Mayo, 2021



Instituto Tecnológico de Tijuana

Tijuana, Baja California,

26/marzo/2021

OFICIO No. 035/DEPI/2021

Asunto: Autorización de Impresión de Tesis

MARIBEL GUERRERO LUIS
JEFA DEL DEPARTAMENTO DE SERVICIOS ESCOLARES
PRESENTE

En lo referente al trabajo de tesis, "Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor". Presentado por C. **Teodoro Eduardo Macías Escobar**, alumno del Doctorado en Ciencias en Computación con número de control **D08070685**; informo a usted que, a solicitud del comité de tutorial, tengo a bien **Autorizar la impresión de Tesis**, atendiendo las disposiciones de los Lineamientos para la Operación de Estudios de Posgrado del Tecnológico Nacional de México.

Sin más por el momento le envío un cordial saludo.

ATENTAMENTE

Excelencia en Educación Tecnológica-
Por una juventud integrada al desarrollo de México-



INSTITUTO TECNOLÓGICO DE TIJUANA

**DIVISIÓN DE ESTUDIOS DE POSGRADO
E INVESTIGACIÓN**

YAZMIN MALDONADO ROBLES
JEFA DE DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

ccp. Archivo
YMR/lap



Calzada del Tecnológico S/N Esq. Av. Castillo de Chapultepec
y calle Cuauhtemotzin, Fracc. Tomás Aquino C.P. 22414,
Tijuana, Baja California.

(664) 6078400 Ext. 101 / e-mail: dir_tijuana@tecnm.mx

tecnm.mx | tijuana.tecnm.mx





Asunto: Se autoriza impresión de Trabajo de Tesis

Tijuana, B.C., 22 de Marzo del 2021

C. Dra. Yazmin Maldonado Robles

Jefe de la Div. de Estudios de Posgrado e Investigación
Presente.

En lo referente al trabajo de tesis escrito, con título "OPTIMIZACIÓN DE PROBLEMAS DINÁMICOS EN PRESENCIA DE UN GRAN NÚMERO DE OBJETIVOS Y PREFERENCIAS DEL DECISOR", presentado por el **C. TEODORO EDUARDO MACIAS ESCOBAR**, alumno del Doctorado en Ciencias en Computación con número de control **D08070685**, informamos a usted que se autoriza el escrito de tesis y se aprueba en todas sus partes, en virtud de reunir los requisitos de un trabajo de grado de Doctorado y a la vez se autoriza al interesado para que proceda de inmediato a la impresión del mismo y a presentar su examen de grado, ya que cumple con todos los requisitos.

ATENTAMENTE

DRA. ELBA PATRICIA MELIN OLMEDA
PRESIDENTE

DR. OSCAR CASTILLO LOPEZ
SECRETARIO

DR. JOSE MARIO GARCIA VALDEZ
VOCAL

DR. FEVRIER ADOLFO VALDEZ ACOSTA
VOCAL

DRA. DANIELA ADRIANA SANCHEZ VIZCARRA
VOCAL

c.c.p. Oficina de Titulación
c.c.p. División de Estudios de Posgrado e Investigación
c.c.p. Expediente
c.c.p. Interesado

EPMO/*inf



CARTA DECLARACIÓN DE PROPIEDAD INTELECTUAL

Tijuana, BC a 8 de marzo del 2021

Yo **TEODORO EDUARDO MACIAS ESCOBAR** reconozco que el Trabajo de Tesis de Doctorado que realice durante mis estudios en el Doctorado en Ciencias en Computación del Instituto Tecnológico de Tijuana fue parte del Proyecto de Investigación Titulado: **Apoyo a la decisión y optimización inteligente de sistemas complejos y de gran escala** NUMERO CLAVE CONACYT 269889 que desarrolla mi director de tesis la **Dra. Laura Cruz Reyes** y del cual es responsable del proyecto de investigación. Por esta razón, los métodos, modelos, algoritmos, y software realizados, así como datos y resultados obtenidos durante el desarrollo de mi tesis de doctorado son propiedad intelectual de mi Director y Codirector de Tesis, del Tecnológico Nacional de México, Instituto Tecnológico de Tijuana y del Conacyt, y No podré utilizarlos por mi cuenta durante, Ni después de terminar mi beca o estudios, excepto a solicitud escrita para poder utilizarlos bajo una colaboración directa con mi director el cual es responsable del proyecto de investigación. Por tanto, estoy de acuerdo en que no podre utilizar ni tomar modelos, ni datos utilizados en este proyecto de investigación y en el desarrollo de tesis para: presentaciones, publicaciones ni desarrollo de mi propia investigación que pudiera desarrollar una vez concluidos mis estudios.

Atentamente

TEODORO EDUARDO MACIAS ESCOBAR



Estudiante de Doctorado en Ciencias en Computación

DECLARACIÓN DE ORIGINALIDAD

Tijuana, BC., 8 de marzo de 2021,

Yo, **Teodoro Eduardo Macias Escobar**, estudiante del Doctorado en Ciencias en Computación, en mi calidad de autor manifiesto que este documento de tesis es producto de mi trabajo original y que no infringe los derechos de terceros, tales como derechos de publicación, derechos de autor, patente y similitud. Por lo tanto, la obra realizada es de mi exclusiva autoría y no infringí en copiar el texto o imágenes, de fuentes de información por lo cual soy responsable del escrito que aquí se presenta.

Así mismo, declaro que en las citas textuales que he incluido (las cuales aparecen entre comillas) y en los resúmenes que he realizado de publicaciones ajenas, indico explícitamente los datos de los autores y las publicaciones.

En caso de presentarse cualquier reclamación o acción por parte de terceros en cuanto a los derechos de autor sobre la obra en cuestión, acepto toda la responsabilidad de tal infracción y relevo de esta a mi director de tesis, así como al Tecnológico Nacional de México, al Instituto Tecnológico de Tijuana y a sus respectivas autoridades.



Teodoro Eduardo Macias Escobar
Estudiante de Doctorado en Ciencias en Computación

Resumen

La idea de obtener resultados óptimos ante cualquier situación que se presenta es común en todas las personas. Es por este motivo que la resolución de problemas de optimización tiene una importancia vital en cualquier estrato o nivel económico y social. Sin embargo, estos problemas tienden a tener varias características que dificultan obtener un resultado satisfactorio. Entre dichas características, las más relevantes son la existencia de múltiples objetivos, preferencias específicas por parte de algún individuo o grupo y los constantes cambios que ocurren en el entorno que rodea a dicho problema.

Se han propuesto una gran variedad de formas para solucionar estos problemas y evaluar los resultados obtenidos. No obstante, esto lleva a un problema en el cual se vuelve muy complicado determinar cuál alternativa es más viable para cada tipo de problema. Es por este motivo que esta tesis propone el desarrollo de un conjunto de relaciones entre diversos problemas, alternativas y formas de evaluación mediante el desarrollo de diversas metodologías que permitan unificarlos para poder solucionar un mayor abanico de problemas de forma satisfactoria.

Este trabajo presenta múltiples aportaciones dentro del área de optimización multiobjetivo. En primer lugar, el desarrollo una metodología mediante hiperheurísticas para solucionar problemas de optimización dinámicos, con preferencias determinadas e incluso con un número creciente de objetivos. También, se proponen estrategias para solucionar este tipo de problemas, así como una técnica que permite incorporar preferencias dentro de ellas. Estos dos elementos forman parte vital de dichas hiperheurísticas.

Además, dentro de este trabajo se propone el uso de diversos métodos de selección de heurísticas y criterios de aceptación basados en diferentes métricas de desempeño y el uso de evolucionabilidad poblacional, un método de análisis de aptitud de entornos. Cabe aclarar que varias de estas estrategias son contribuciones originales dentro del área para problemas de este tipo. Lo anterior permite establecer un punto de entrada para futuras investigaciones.

Los resultados obtenidos por las propuestas planteadas en esta tesis muestran que las metodologías utilizadas son capaces de producir soluciones de buena calidad para problemas de optimización de diversas características, tanto dinámicos, multiobjetivo y con preferencias definidas por agentes, lo cual enfoca la búsqueda de soluciones a un área específica dentro del espacio de soluciones. Esto muestra dos aspectos importantes de las propuestas. En primer lugar, la capacidad de obtener mejores soluciones con respecto a diversos algoritmos dentro del estado del arte. Y segundo, que la propuesta es capaz de solucionar de forma efectiva problemas de optimización con un número de características diferentes mayores a lo que son capaces dichos algoritmos del estado de arte usados de forma individual.

Abstract

The idea of obtaining optimal solutions for any situation that arises is a common thought to most people. It is for this reason that solving optimization problems is of vital importance in any economic and social stratum or level. However, these problems might present certain characteristics that makes obtaining a satisfactory result difficult. Among these characteristics, the most relevant are the existence of multiple objectives, specific preferences on the part of some individual or group, and the constant changes that occur in the environment surrounding the problem.

A great variety of alternatives have been proposed to solve these problems and to evaluate the results obtained. However, this leads to a problem in which it becomes very complicated to determine which alternative is more viable for each type of problem. It is for this motive that this thesis proposes the development of a set of relationships between diverse problems, alternatives and evaluation indicators by means of the development of diverse methodologies that allow to unify them to be able to solve a greater range of problems in a satisfactory way.

This work provides multiple contributions to research areas related to multi-objective optimization. First, the development of a methodology using hyper-heuristics to solve dynamic optimization problems, with preferences defined by a decision maker and even with an increasing number of objectives. Also, this work proposes strategies and methods to solve this type of problems, as well as a preference incorporation technique. These two elements form a vital part of the proposed hyper-heuristics.

In addition, this work proposes the use of different heuristic selection methods and solution acceptance criteria based on different performance metrics and the use of population evolvability, a fitness landscape analysis (FLA) method. It should be clarified that several of these strategies are original contributions within the research area for problems of this type. This study allows establishing an entry point for future research.

The results obtained by the proposals presented in this thesis show that the methodologies used are capable of producing good-quality solutions for optimization problems of diverse characteristics, such as being dynamic, multi-objective, and with preferences defined by a decision maker, which focuses the search for solutions to a specific area within the solution space. This displays two important aspects of the proposals. First, the ability to obtain better solutions with respect to several algorithms within the state-of-the-art. And second, that the proposal is capable of effectively solving optimization problems with a greater number of different characteristics than what the state-of-the-art algorithms used individually are capable of.

Agradecimientos

Agradezco al Consejo Nacional de Ciencia y Tecnología (CONACYT) y a la Dirección General de Educación Tecnológica (DGEST) por su apoyo durante todo el proceso que llevó a la conclusión de este documento.

También agradezco al Instituto Tecnológico de Ciudad Madero (ITCM), al Instituto Tecnológico de Tijuana (ITT) y la Universidad de Cádiz (UCA) por todas las disposiciones, facilidades y recursos proporcionados durante el periodo en el cual ejercí mis estudios. De igual manera agradezco a todos los profesores que me apoyaron durante este proceso, los cuales contribuyeron a mi formación académica.

De igual forma, agradezco a los profesores que conforman el comité tutorial: Dra. Laura Cruz Reyes, Dr. Héctor Joaquín Fraire Huacuja, Dr. Nelson Rangel Valdez, Dra. Claudia Guadalupe Gómez Santillán y Dr. Juan Frausto Solís.

Igualmente, estoy completamente agradecido con el tiempo, esfuerzo y apoyo otorgado por mi directora de tesis, Dra. Laura Cruz Reyes y mi codirector de tesis, Dr. Bernabé Dorronsoro durante el desarrollo de esta tesis. Ambos estuvieron siempre al pendiente de mi desarrollo no únicamente como estudiante, sino también como persona. Por este motivo no solamente tienen mi admiración y respeto a nivel académico, sino también a nivel personal, siendo ejemplos a seguir en mis futuros proyectos.

Agradezco a todos mis compañeros y miembros de generaciones anteriores y siguientes pertenecientes al Tecnológico Nacional de México y a la Universidad de Cádiz con los cuales tuve la oportunidad de relacionarme durante el transcurso del desarrollo de esta tesis. Su apoyo y conocimiento impulsaron el progreso continuo de este trabajo.

Finalmente quiero agradecer a mi familia, que siempre me ha apoyado cuando más la necesito, que ha soportado mis errores y defectos y que siempre ha estado a mi lado dándome ánimos y consejos, incluso estando a horas de distancia de ellos y en el otro lado del mundo.

Tabla de contenidos

Introducción.....	1
1.1 Justificación y beneficios.....	1
1.2 Objetivos	3
1.2.1 Objetivo general.....	3
1.2.2 Objetivos específicos	3
1.3 Hipótesis	3
1.4 Aportaciones	4
1.5 Producción científica	4
1.6 Organización del documento.....	5
Marco teórico.....	7
2.1 Toma de decisiones	7
2.1.1 El problema de toma de decisiones.....	7
2.1.2 Tomador de decisiones y sus preferencias.....	8
2.2 Optimización multiobjetivo	8
2.2.1 Variables de decisión.....	8
2.2.2 Restricciones.....	9
2.2.3 Modelo matemático de un problema multiobjetivo.....	9
2.2.4 Optimalidad de Pareto.....	9
2.2.5 Dominancia de Pareto.....	10
2.2.6 Frente de Pareto.....	11
2.2.7 Propósitos de la optimización multiobjetivo.....	11
2.3 Algoritmos evolutivos multiobjetivo.....	13
2.3.1 Metaheurísticas evolutivas	13
2.3.2 Algoritmos evolutivos multiobjetivo	13
2.3.3 Estrategias de incorporación de preferencias en MOEA	14
2.3.4 Métricas de desempeño.....	15
2.4 Optimización multiobjetivo dinámica.....	16
2.4.1 Modelo matemático de un problema multiobjetivo dinámico.....	16
2.4.2 Clasificación de los problemas de optimización multiobjetivo dinámicos.....	16
2.4.3 Cambios en el frente de Pareto	17
2.4.4 Algoritmos evolutivos multiobjetivo dinámicos	18
2.5 Superficie de aptitud dinámica	19
2.5.1 Unificación de valores objetivo	19
2.5.2 Evolucionabilidad.....	20
2.6 Hiperheurísticas.....	20
2.6.1 Concepto de hiperheurística	21
2.6.2 Clasificación de hiperheurísticas	21
2.6.3 Fases de una hiperheurística.....	22
Antecedentes	24
3.1 Algoritmos propuestos para solucionar DMOPs.....	24
3.2 Algoritmos propuestos para resolver MOPs preferenciales	25
3.3 Algoritmos propuestos para resolver MaOPs.....	27
3.4 Instancias para problemas de optimización multiobjetivo dinámicos	28
3.5 Métricas de evaluación de desempeño de algoritmos	30
3.5.1 Métricas para evaluar MOEAs	30
3.5.2 Métricas para evaluar algoritmos que solucionan MaOPs.....	32
3.5.3 Adaptación de las métricas de desempeño a problemas dinámicos	33
3.6 Análisis de superficie de aptitud.....	35
3.6.1 Análisis de superficie de aptitud dinámica	35
3.6.2 Evolucionabilidad poblacional	37
3.7 Hiperheurísticas.....	37
3.7.1 Métodos de selección de heurísticas de bajo nivel	38
3.7.2 Criterios de aceptación de soluciones	39
3.7.3 Hiperheurísticas para solucionar MOPs	39

3.7.4	Hiperheurísticas para solucionar problemas dinámicos	40
3.7.5	Hiperheurísticas con incorporación de preferencias	44
3.7.6	Hiperheurísticas para solucionar MaOPs.....	45
3.8	Problemas abiertos a la investigación y conclusiones	46
Planteamiento del problema		48
4.1	Problema multiobjetivo dinámico	48
4.1.1	Incorporación de preferencias en problemas dinámicos	49
4.1.2	Métodos de evaluación de problemas multiobjetivo dinámicos	50
4.2	Problema de muchos objetivos dinámico.....	51
4.2.1	Incorporación de preferencias en problemas dinámicos	52
4.2.2	Retos para la evaluación de problemas de muchos objetivos dinámicos	53
4.2.3	Métodos de evaluación de problemas de muchos objetivos dinámicos	53
Propuesta 1: Hiperheurísticas para la resolución de DMOPs y DMaOPs.....		54
5.1	Hiperheurísticas DHH y DPEM-HH	54
5.2	Hiperheurística DHHMa	63
5.3	Conclusiones	66
Propuesta 2: Método de incorporación de preferencias en DMOPs y DMaOPs		67
6.1	Método de Separación por Planos (PS).....	67
6.2	Incorporación de PS a hiperheurísticas	73
6.3	Incorporación de PS en hiperheurísticas para muchos objetivos.....	76
6.4	Conclusiones	78
Experimentación y análisis.....		80
7.1	Comparativa de DHH respecto a DMOEAs.....	80
7.1.1	Análisis de los resultados obtenidos.....	82
7.1.2	Conclusiones	83
7.2	Análisis de método de selección y criterios de aceptación en DPEM-HH	84
7.2.1	Análisis de los resultados para el conjunto FDA.....	87
7.2.2	Análisis de los resultados para el conjunto DMZDT.....	89
7.2.3	Análisis de los resultados para el conjunto dMOP	90
7.2.4	Conclusiones	92
7.3	Comparativa de DPEM-HH respecto a DMOEAs	93
7.3.1	Análisis de los resultados obtenidos.....	94
7.3.2	Análisis de DHH y DPEM-HH.....	96
7.3.3	Conclusiones	98
7.4	Solución de DMOPs preferenciales	99
7.4.1	Análisis de factibilidad del método de Separación por Planos	99
7.4.2	Análisis de DMOEAs con PS para resolver DMOPs preferenciales	101
7.4.3	Comparativa de métodos de agregación usados en hiperheurísticas.....	104
7.4.4	Análisis de métodos de selección usados en DHH-PS y DPEM-HH-PS	106
7.4.5	Análisis de criterios de aceptación usados en DHH-PS y DPEM-HH-PS	109
7.4.6	Análisis de DHH-PS y DPEM-HH-PS bajo diferentes condiciones dinámicas.....	111
7.4.7	Comparativa de hiperheurística respecto a DMOEAs	114
7.4.8	Conclusiones	117
7.5	Pruebas preliminares de DMOEAs para solucionar DMaOPs	118
7.5.1	Análisis de los resultados obtenidos.....	119
7.5.2	Conclusiones	122
7.6	Solución de DMaOPs mediante DHHMa	123
7.6.1	Comparación entre métodos de selección.....	123
7.6.2	Análisis de DHHMa respecto a sus LLHs	124
7.6.3	Conclusiones	126
7.7	Solución de DMaOPs preferenciales	126
7.7.1	Definición de valores de las variables del método de Separación por Planos	126
7.7.2	Comparación entre métodos de selección.....	129
7.7.3	Análisis de DHHMa-PS respecto a sus LLHs.....	130
7.7.4	Conclusiones	133
Conclusiones y trabajo futuro.....		135
8.3	Trabajos futuros.....	135

Referencias	137
Anexo A. Experimentaciones para definir valor de variables en Separación por Planos.....	145
Anexo B. Resultados de VD y HVR en la comparación de DHH-PS y DPEM-HH-PS.....	147
Anexo C. Clasificación de las configuraciones de parámetros para Separación por Planos	148
Anexo D. Resultados de la comparación entre métodos de selección de LLHs para DHHMa	149

Capítulo 1

Introducción

Muchos problemas en el mundo real son complejos (NP-duro), los cuales abordan múltiples objetivos a la vez. En dichos casos es común que algunos de los elementos involucrados en estos problemas sufran alguna clase de alteración durante el transcurso del tiempo. Para problemas estáticos con un número limitado de objetivos se ha demostrado que la aplicación de algoritmos metaheurísticos basados en dominancia de Pareto, tales como los algoritmos evolutivos multiobjetivo (MOEA por sus siglas en inglés) constituye una alternativa de solución que ha mostrado buenos resultados.

Sin embargo, la efectividad de estos algoritmos puede ver una reducción en su calidad cuando enfrenta problemas con una naturaleza dinámica. Esto quiere decir que algunas de sus características y propiedades sufren modificaciones durante el proceso de resolución. Esta misma situación puede presentarse cuando el número de objetivos a solucionar se incrementa, llegando incluso a transformarse en un problema de muchos objetivos (cuatro o más objetivos a resolver).

Determinar una estrategia correcta para mejorar la efectividad de estos algoritmos es complicado. Lo anterior debido tanto a la dificultad del problema que se busca resolver como al gran esfuerzo computacional requerido para obtener una solución utilizando dichos algoritmos, ya que existe una gran cantidad de requisitos a satisfacer.

Trabajos previos dentro del grupo al que se pertenece han mostrado que la incorporación de preferencias del tomador de decisiones durante el proceso de solución permite reducir el espacio de búsqueda, dirigiéndola hacia la región de interés del usuario final. En este sentido, consideramos que la incorporación de un conjunto de preferencias posibilitará estudios a profundidad en este área emergente.

La revisión de la literatura señala que se requiere identificar características retadoras de este tipo de problemas, buscando reflejarlas en casos de prueba con el objetivo de que sea posible llevar a cabo una evaluación de la efectividad de nuevos metaheurísticos.

En esta tesis se propone contribuir al crecimiento de este área abordando diversos retos en la línea de caracterización de los problemas dinámicos multiobjetivo (DMOP por sus siglas en inglés), así como en el diseño de algoritmos metaheurísticos que sean capaces de competir con aquellos que se mencionan dentro del estado del arte.

1.1 Justificación y beneficios

Como se mencionó previamente, muchos problemas que enfrentamos pueden ser complejos, dinámicos y con múltiples objetivos a resolver, dificultando su solución en tiempos computacionalmente aceptables. Debido a esto, el diseño y aplicación de algoritmos metaheurísticos capaces de adaptarse a los cambios ocurridos en el entorno para resolver

dichos problemas se ha vuelto una necesidad debido a la facilidad de éstos de obtener buenos resultados de manera rápida y eficiente. Generando soluciones que se acomoden al nuevo estado que rodea al problema en cada cambio que se presenta.

Un ejemplo de esto son los MOEAs dinámicos (DMOEA), los cuales incorporan técnicas para identificar y adaptarse de la mejor manera posible a los cambios. Cabe aclarar que las soluciones encontradas por enfoques estocásticos, aunque no son óptimos, son lo suficientemente cercanos a éstos como para ser considerados aceptables.

Hoy en día se han propuesto varias metaheurísticas capaces de solucionar problemas dentro de un entorno dinámico, así como situaciones y simulaciones en los cuales aplicar dichas propuestas. También se han desarrollado medidas para analizar las propiedades de dichos problemas, así como métricas para evaluar el desempeño y la calidad de las soluciones encontradas por estas metaheurísticas. Sin embargo, aún hace faltan definir ciertos aspectos dentro de este campo.

Entre dichos aspectos resaltan dos, primero la falta de estandarización entre algoritmos propuestos, problemas diseñados, métricas de evaluación de desempeño y medidas para el análisis de superficie de aptitud de problemas. Esto se debe a que cada autor utiliza un conjunto diferente de éstos para evaluar la calidad de su trabajo.

Por otro lado, también es necesario definir una metodología capaz de seleccionar a partir de un conjunto de algoritmos, aquellos que generen los mejores resultados para resolver un problema. Lo anterior se puede llevar a cabo si se conoce de forma previa el desempeño de los algoritmos considerados para el problema que se busca solucionar. Esto es posible si se tiene una relación definida entre estos elementos, lo cual nos regresa al punto establecido en el párrafo anterior.

El propósito de este trabajo es el desarrollo de una metodología que sea capaz de resolver problemas dinámicos de optimización multiobjetivo. Para esto, se identifican diversas propuestas y trabajos dentro de la literatura relevantes en el área de optimización y pertenecientes a los elementos mencionados en el párrafo anterior.

En base a dicha identificación, se planea llevar a cabo mediante un proceso de caracterización de los algoritmos y problemas seleccionados en base a un conjunto de identificadores que permitan evaluar diversas propiedades de éstos. Permitiendo de esta manera definir un conjunto de relaciones que contengan cada uno de los elementos.

La metodología propuesta se apoyará de estas relaciones, así como de un método de selección de algoritmos y de aceptación de soluciones adecuado, para seleccionar del conjunto de algoritmos disponibles, aquel o aquellos que generen los mejores resultados para el problema que se busca resolver.

Este trabajo también servirá de apoyo para futuras investigaciones que se busquen realizar dentro del campo, por lo que la contribución de este proyecto no se limita únicamente a la metodología desarrollada y la calidad de los resultados obtenidos, sino que también el contenido teórico generado puede servir de base para dichas investigaciones.

1.2 Objetivos

A continuación, se presenta el objetivo general y los objetivos específicos planteados para esta investigación.

1.2.1 Objetivo general

Desarrollar una metodología de solución de problemas dinámicos de optimización multiobjetivo comparable con los del estado del arte. Apoyándose de un conjunto de relaciones, las cuales se basan en una caracterización con indicadores de desempeño, de estos problemas y algunos algoritmos del estado del arte propuestos en la literatura para resolverlos.

1.2.2 Objetivos específicos

Los objetivos específicos que buscan apoyar el cumplimiento del objetivo general son:

- 1) Seleccionar un conjunto de DMOPs e identificar sus propiedades.
- 2) Seleccionar un conjunto de algoritmos para solucionar DMOPs, encontrando características que las identifiquen y distingan de las demás.
- 3) Seleccionar identificadores de la literatura que permitan caracterizar las propiedades de los problemas y algoritmos seleccionados y establecer relaciones entre ellos.
- 4) Desarrollar una metodología capaz de automatizar el proceso de selección de algoritmos para solucionar un DMOP en base a relaciones de desempeño.
- 5) Implementar una hiperheurística que sea capaz de resolver DMOPs, incluyendo la metodología de selección basado en caracterización del desempeño.
- 6) Probar los límites de la hiperheurística desarrollada en problemas de gran escala y con preferencias de un tomador de decisiones.

1.3 Hipótesis

A continuación, se describen las hipótesis planteadas para este proyecto de investigación:

- 1) Es factible llevar a cabo una caracterización de un conjunto de DMOPs, así como de algoritmos heurísticos para resolver dichos problemas bajo un conjunto de identificadores que analizan las propiedades de estos problemas y algoritmos.
- 2) Es posible establecer un conjunto de relaciones entre diversos DMOPs, algoritmos utilizados para solucionarlos e identificadores seleccionados para utilizar en este proyecto, con el objetivo de definir una estandarización entre estos elementos.
- 3) Se puede desarrollar una metodología de alto nivel capaz de aplicarse en los DMOPs seleccionados para su evaluación en este proyecto apoyándose del conjunto de relaciones desarrolladas. Dicha metodología aplicará de manera adecuada un método de selección de heurísticas de bajo nivel y un método de aceptación de soluciones.
- 4) Es viable la implementación de una hiperheurística en base de la metodología propuesta, dicha hiperheurística debe ser capaz de aplicarse en los DMOPs seleccionados previamente para su evaluación, generando resultados comparables con los de estado del arte.

1.4 Aportaciones

Las principales aportaciones presentadas en este trabajo son las siguientes:

- Hiperheurísticas para solucionar problemas de optimización dinámicos multiobjetivo, con muchos objetivos y preferencias definidas por un tomador de decisiones. Las principales características de las hiperheurísticas propuestas son:
 - Definición de relaciones entre múltiples algoritmos, instancias e identificadores de desempeño con diferentes características.
 - Capacidad de utilizar diferentes DMOEAs sin importar sus características. Debe tomarse en cuenta que lo más cercano dentro de la literatura es la resolución de MOPs usando MOEAs.
 - Capacidad de utilizar métodos de incorporación de preferencias para resolver DMOPs y DMaOPs preferenciales.
 - Uso de estrategias específicas para resolver DMaOPs preferenciales.
- Un nuevo método de incorporación de preferencias basado en punto de referencias y un vector de preferencias definidos por un DM. Este método busca definir un balance entre intensificación y diversificación para encontrar soluciones con buena convergencia y diversidad, enfocándose en la región de interés definida.
- Adaptación y propuesta de métodos de selección de LLHs y criterios de aceptación para utilizarse en problemas de optimización dinámica. Esto incluye estrategias basadas en función de elección, voracidad y aleatoriedad.
- Adaptación de MOEAs a dinamismo e incorporación de preferencias mediante:
 - Inserción de métodos de detección y adaptación a cambios.
 - Inserción del método de Separación por Planos dentro de su proceso.

1.5 Producción científica

A continuación, se enlistan las publicaciones generadas por la investigación realizada en este trabajo de tesis:

- Macias-Escobar, T., Cruz-Reyes, L., Dorronsoro, B., Fraire-Huacuja, H., Rangel-Valdez, N., & Gómez-Santillán, C. (2019). Application of population evolvability in a hyper-heuristic for dynamic multi-objective optimization. *Technological and Economic Development of Economy*, 25(5), 951-978. JCR Q2, FI 4.344
- Macias-Escobar, T., Cruz-Reyes, L., Fraire, H., & Dorronsoro, B. (2020). Plane Separation: A method to solve dynamic multi-objective optimization problems with incorporated preferences. *Future Generation Computer Systems*, 110, 864-875. JCR Q1, FI 5.768
- Macias-Escobar, T., Dorronsoro, B., Cruz-Reyes, L., Rangel-Valdez, N., & Gómez-Santillán, C. (2020). A Survey of Hyper-heuristics for Dynamic Optimization Problems. In *Intuitionistic and Type-2 Fuzzy Logic Enhancements in Neural and Optimization Algorithms: Theory and Applications* (pp. 463-477). Springer, Cham. SCOPUS Q4, SJR 0.22

De igual forma se enlistan los diferentes congresos en los cuales se presentaron diversos avances y descubrimientos resultantes de los resultados y experimentos realizados para este trabajo de tesis:

- “*Application of population evolvability in a hyper-heuristic for dynamic multi-objective optimization*” en **International Conference on Modelling and Simulation**. Abril, 2018.
- “*DPEM-HH: A Population Evolvability-based Multi-objective Hyperheuristic to Solve Dynamic Multi-objective Optimization Problems*” en **Numerical and Evolutionary Optimization 2018**. Septiembre, 2018.
- “*A Survey of Hyper-heuristics for Dynamic Optimization Problems*” en **International Seminar on Computing Intelligence (ISCI 2019)**, Enero, 2019.
- “*Application of Multiple Preference Incorporation Approaches to solve Dynamic Multi-objective Optimization Problems*” en **Numerical and Evolutionary Optimization 2019**. Septiembre, 2019.
- “*Introducción a la optimización dinámica y solución de problemas dinámicos multiobjetivo*” en **8° Seminario y 7° Escuela Nacional de Aprendizaje e Inteligencia Computacional**. Octubre, 2020.

1.6 Organización del documento

La estructura de este documento de tesis es la siguiente:

Capítulo 2 – Marco teórico. Este capítulo presenta los conceptos teóricos fundamentales necesarios para la comprensión adecuada de las definiciones y planteamientos presentados en las siguientes secciones de esta tesis. Los conceptos mostrados están relacionados con la optimización multiobjetivo, optimización dinámica, toma de decisiones, algoritmos evolutivos, hiperheurísticas, evaluación de desempeño y análisis de superficie de aptitud.

Capítulo 3 – Antecedentes. Esta sección presenta un estudio de las propuestas más relevantes, así como recientes, respecto a los conceptos mencionados en el capítulo anterior. Estos trabajos son organizados de forma que se presentan diversos DMOEAs, hiperheurísticas, DMOPs, métricas de evaluación de desempeño y de análisis de superficie de aptitud que tiene el mayor impacto con respecto a esta tesis. Se presentan las características esenciales y únicas de cada propuesta. Esto permite establecer cuáles son las áreas de oportunidad existentes en el campo que fueron tomadas en cuenta para este proyecto.

Capítulo 4 – Planteamiento del problema. Aquí, se hace una explicación de manera detallada de cada una de las cuestiones en las cuales se enfoca la propuesta de este documento de tesis. Esto se hace en consideración de las áreas de oportunidad detectadas en el capítulo anterior. Como se mencionó previamente, esta tesis busca establecer una estandarización para la solución de diversos DMOPs con diferentes propiedades mediante el uso de una metodología que permita unificar diferentes métodos propuestos.

Capítulo 5 – Propuesta de solución. En este capítulo se presenta la estrategia propuesta para resolver los diversos problemas planteados en el capítulo 4. Se presenta el enfoque

utilizado, así como cada uno de los procedimientos, algoritmos, métricas, métodos y técnicas usados para esta propuesta de manera detallada.

Capítulo 6 – Experimentación y análisis. La propuesta presentada es evaluada en cada una de las diferentes condiciones establecidas por cada uno de los problemas a resolver planteados. Se menciona a detalle las características y propiedades tanto de cada uno de la propuesta planteada, así como de los problemas a solucionar. Los resultados obtenidos son comparados con algoritmos pertenecientes al estado del arte relacionados al tipo de problema a resolver (dinámico, con preferencias incorporadas, de cuatro o más objetivos, etc.).

Capítulo 7 – Conclusiones y trabajo futuro. En base a los resultados obtenidos, este capítulo presenta las conclusiones alcanzadas. Si se llegó a satisfacer tanto el objetivo general como los específicos, así como si las hipótesis planteadas fueron aceptadas o no. De igual forma, se mencionan las áreas de investigación en las cuales puede continuarse el desarrollo de este proyecto en trabajos futuros.

Capítulo 2

Marco teórico

Este capítulo incluye los conceptos y fundamentos necesarios para comprender tanto el planteamiento del problema como la propuesta de solución presentados en capítulos posteriores a este.

2.1 Toma de decisiones

La toma de decisiones consiste en elegir la mejor opción disponible dentro de un conjunto de alternativas posibles, buscando tomar de manera racional decisiones difíciles y complejas [Keeney y Raiffa, 1993]. Este proceso se apoya de varias disciplinas tales como estadística, matemática o inteligencia artificial buscando desarrollar modelos para facilitar la toma de decisión por la persona o grupo encargado.

2.1.1 El problema de toma de decisiones

Una situación común en varios eventos de la vida diaria es enfrentar situaciones en las cuales debe tomarse una decisión entre múltiples alternativas disponibles. En varias ocasiones, estas situaciones también pueden involucrar múltiples criterios a considerar.

El contexto del párrafo anterior presenta un problema de toma de decisión, donde es necesario revisar cada alternativa individualmente, identificando aquella que satisfaga las necesidades actuales de la mejor manera posible, buscando maximizar la ganancia obtenida utilizando los recursos disponibles.

Según Chernoff [Chernoff, 1987], los problemas de decisión clásicos presentan los siguientes elementos:

- Objetivos por resolver.
- Las alternativas disponibles para satisfacer dichos objetivos.
- Factores o estados que definen el entorno que rodea al problema de decisión.
- Valores de utilidad o efectos relacionados a la comparativa de parejas de alternativas disponibles bajo el entorno actual del problema.

De acuerdo con Sánchez [Sánchez, 2007], la teoría de decisión establece una serie de criterios que permiten clasificar a los problemas de acuerdo con varios puntos de vista:

- Número de criterios o atributos que valoran la toma de decisión.
- Entorno del problema al llevar a cabo la toma de decisión.
- Número de expertos involucrados en la toma de decisión.

2.1.2 Tomador de decisiones y sus preferencias

El tomador de decisión (DM, por sus siglas en inglés) es la persona o grupo encargado de elegir las alternativas que se adecuen de mejor manera a sus preferencias. Las preferencias del DM determinan la dirección que toma la solución de problemas. Es común que cuando se solucionan problemas con múltiples objetivos a satisfacer, estos se encuentren en conflicto entre sí, donde el beneficio de un criterio significa el decremento o estancamiento de otro [Fernández et al., 2011a]. Son las preferencias del DM las que determinan a cuáles objetivos se les deben dar prioridad. Las preferencias del DM se presentan de tres formas distintas [Cohon y Marks, 1975]:

- A priori. El DM incorpora sus preferencias antes del análisis de las alternativas disponibles con el objetivo de que la búsqueda de soluciones se enfoque a dichas preferencias desde el comienzo.
- Progresivas o interactivas. Se determinan durante la resolución del problema.
- A posteriori. El DM incorpora sus preferencias al terminar el proceso de resolución y se aplica al conjunto de soluciones obtenidas para seleccionar las más adecuadas en bases a dichas preferencias.

El modelado de las preferencias dentro de la toma de decisión es un área de trabajo donde los expertos, apoyándose de sus conocimientos previos, habilidades, experiencias y creencias, hacen una valoración del conjunto de alternativas disponibles, estableciendo un orden de preferencia de cada una de ellas como solución preliminar del problema [Doyle, 2004; Oztürk et al., 2005; Roubens y Vincke, 1985]. Existen dos características fundamentales a considerar en el modelado de preferencias [Sánchez, 2007]:

- La estructura utilizada para representar las preferencias (relaciones binarias, vectores de peso, orden lexicográfico, etc.).
- El estilo o dominio utilizado para representar las preferencias (numérico, por intervalos, lingüístico, etc.).

2.2 Optimización multiobjetivo

Muchos de los problemas de toma de decisión involucran múltiples objetivos al mismo tiempo [Chankong y Haimes, 2008], los cuales debido tanto a las restricciones establecidas como a las preferencias determinadas por el DM se encuentran usualmente en conflicto.

Como se mencionó previamente, es común que, al existir múltiples objetivos, estos se encuentren en conflicto entre sí. Por lo tanto, usualmente determinar una sola solución como óptima no es posible. En este caso, se define un subconjunto de alternativas en las cuales no se puede determinar una absoluta dominancia de una de dichas soluciones sobre las demás, por lo que cada una de estas es una potencial solución óptima.

2.2.1 Variables de decisión

Las variables de decisión son valores numéricos dependientes o independientes que representan cada uno de los elementos a determinar de una solución. El vector de n variables

de decisión \vec{x} , compuesto por $\vec{x} = [x_1, x_2, \dots, x_n]$ representa una solución candidata del problema de optimización [Osyczka, 1985].

2.2.2 Restricciones

Los problemas de optimización están limitados por un conjunto de restricciones, ya sea para buscar simular las características del ambiente del problema o la limitación de recursos disponibles. Dichas restricciones están representadas de forma matemática en m desigualdades y p igualdades. Solamente aquellos vectores de variables de decisión \vec{x} que satisfagan todas las restricciones son considerados como soluciones factibles.

- Desigualdades $g_i(\vec{x}) \leq 0; \quad i = 1, 2, \dots, m .$
- Igualdades $h_j(\vec{x}) = 0; \quad j = 1, 2, \dots, p .$

2.2.3 Modelo matemático de un problema multiobjetivo

En un problema de optimización multiobjetivo (MOP) se busca el vector de variables de decisión $\vec{x} = [x_1, x_2, \dots, x_n]$ que optimice un conjunto de funciones objetivo F :

$$F(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})], \quad f_i: R^n \rightarrow R, \quad (2.1)$$

sujeto a:

$$g_i(\vec{x}) \leq 0; \quad i = 1, 2, \dots, m, \quad (2.2)$$

$$h_j(\vec{x}) = 0; \quad j = 1, 2, \dots, p, \quad (2.3)$$

donde:

- k es el número de funciones objetivo.
- n es el número de variables de decisión.
- m es el número de restricciones de desigualdad.
- p es el número de restricciones de igualdad.

En otras palabras, se busca definir un vector de variables de decisión que optimicen todas las funciones objetivo mientras que se satisfagan las restricciones, las cuales definen la región factible del problema.

Cuando este modelo matemático es utilizado para representar un problema de cuatro o más objetivos, se le denomina problema de optimización de muchos objetivos (MaOP por sus siglas en inglés). La estructura de este tipo de problemas es similar a la del MOP, siendo la única diferencia dentro del modelo, el número de objetivos a resolver.

Cada uno de los objetivos, tanto en MOPs como en MaOPs, puede buscar minimizar o maximizar sus valores. Para esta tesis, y sin pérdida de generalización, únicamente se trabajan problemas de minimización.

2.2.4 Optimalidad de Pareto

Debido al posible conflicto entre objetivos en MOPs es complicado determinar una única solución óptima. Esto, obliga a que la noción de “solución óptima” sea modificada para

adaptarse a este escenario. Tanto Edgeworth [Edgeworth, 1881] como Pareto [Pareto, 1896] establecen una noción de optimalidad que considera múltiples soluciones, definida como optimalidad de Pareto.

Un vector de decisión \vec{x}^* es considerado como un óptimo de Pareto si no existe otro vector dentro del área factible del espacio de búsqueda de funciones objetivo la cual mejore algún criterio sin causar deterioro a otro al mismo tiempo. En un MOP de minimización, la definición formal de este concepto es la siguiente:

$$f_i(\vec{x}^*) \leq f_i(\vec{x}), \quad \forall i \in [1, 2, \dots, k], \quad (2.4)$$

$$f_i(\vec{x}^*) < f_i(\vec{x}), \quad \exists i \in [1, 2, \dots, k]. \quad (2.5)$$

La Figura 2.1 muestra gráficamente la definición previa. Considérese un problema con dos objetivos f_1 y f_2 y un conjunto de soluciones para dicho problema. Para este caso el punto p_1 es único óptimo de Pareto, ya que, aunque p_1 y p_2 tienen el mismo valor en f_1 , mientras que p_1 y p_3 tienen el valor más bajo para f_2 , si se aplican las ecuaciones (2.4) y (2.5) el resultado favorece únicamente a p_1 .

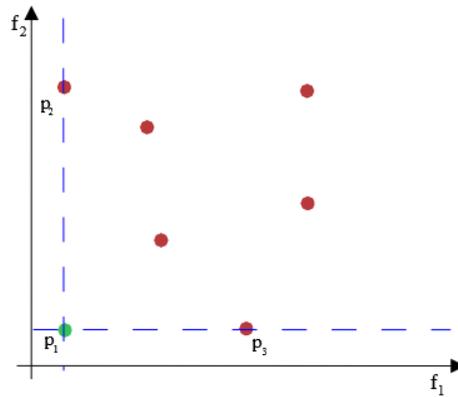


Figura 2.1. Representación de un óptimo de Pareto.

Bajo esta noción se pueden determinar múltiples soluciones óptimas que conforman el conjunto de soluciones óptimas de Pareto (POS^*). La representación del POS^* dentro del espacio de búsqueda de funciones objetivo se define como frente óptimo de Pareto (POF^*).

2.2.5 Dominancia de Pareto

Sean dos vectores de variables de decisión \vec{x}_a y \vec{x}_b . Se dice que \vec{x}_a tiene dominancia de Pareto sobre \vec{x}_b en un MOP de minimización sí y solo sí se cumplen las siguientes condiciones [Pareto, 1896]:

1. \vec{x}_a es al menos igual que \vec{x}_b en todos los objetivos:

$$f_i(\vec{x}_a) \leq f_j(\vec{x}_b), \quad \forall i \in [1, 2, \dots, k]. \quad (2.6)$$

2. \vec{x}_a es mejor que \vec{x}_b en al menos un objetivo:

$$f_i(\vec{x}_a) < f_j(\vec{x}_b), \quad \exists i \in [1, 2, \dots, k]. \quad (2.7)$$

Por lo anterior, la comparación de dominancia de Pareto entre \vec{x}_a y \vec{x}_b sólo pueden producir tres posibles resultados:

- \vec{x}_a domina a \vec{x}_b , $\vec{x}_a < \vec{x}_b$.
- \vec{x}_a es dominada por \vec{x}_b , $\vec{x}_b < \vec{x}_a$.
- \vec{x}_a y \vec{x}_b no se dominan entre sí (\vec{x}_a no domina a \vec{x}_b y \vec{x}_b no domina a \vec{x}_a).

La Figura 2.2 muestra el valor de varias soluciones para cada uno de los objetivos. Tomando en cuenta las ecuaciones (2.6) y (2.7), es posible concluir que para p_1, p_2, p_3, p_4 y p_5 no existe solución que las domine. Por ejemplo, en el caso de p_2 y p_6 , aunque tienen el mismo valor en f_1 no ocurre lo mismo en f_2 , donde p_2 tiene un mejor valor que p_6 , por lo que se dice que p_2 domina a p_6 .

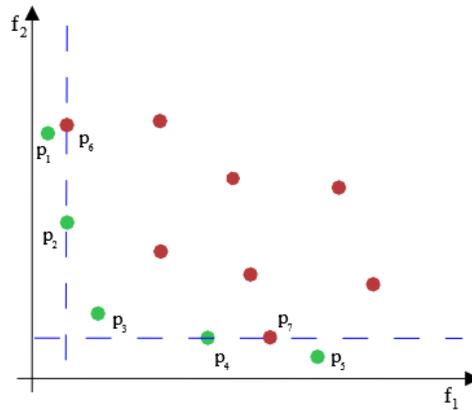


Figura 2.2. Elementos dominados y no dominados en un conjunto.

2.2.6 Frente de Pareto

La Figura 2.2 presentada previamente muestra las soluciones no dominadas dentro del espacio de búsqueda de las funciones objetivo. Cuando las soluciones se representan de esta manera se le denomina al conjunto frente de Pareto. La definición formal de este concepto se muestra en la ecuación (2.8):

$$POF^* = \{F(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})) \mid \vec{x} \in POS^*\} \quad (2.8)$$

2.2.7 Propósitos de la optimización multiobjetivo

Como se mencionó previamente, es común que en los MOPs se presenten múltiples elementos dentro del conjunto de soluciones óptimas de Pareto correspondiente. Debido a que cada una de estas soluciones es igualmente relevante, es deseable encontrar tantas como sea posible. A partir de este concepto, [Deb et al., 2002a] menciona dos propósitos que existen dentro de la optimización multiobjetivo.

- 1) Convergencia. El conjunto de soluciones encontrado debe ser lo más cercano posible al conjunto de óptimos de Pareto.

- 2) Diversidad. El conjunto de soluciones encontrado debe ser bien dispersado dentro del espacio de soluciones factibles.

La convergencia es esencial, ya que se busca que la solución encontrada sea de buena calidad, para ello debe encontrarse dentro del frente de Pareto o por lo menos lo suficientemente cercano de éste como para poder ser considerada como solución candidata. Si una solución está demasiado alejada de dicho frente es de poca calidad y por lo tanto irrelevante. En la Figura 2.3 se muestra un ejemplo de la convergencia mediante la clasificación de las soluciones en deseadas y no deseadas en base a la cercanía de dichas soluciones al frente de Pareto correspondiente al problema de optimización.

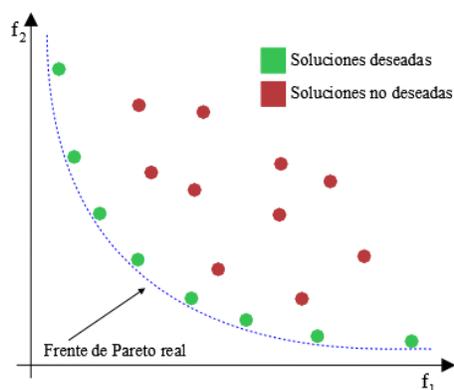


Figura 2.3. Clasificación de soluciones de acuerdo con su cercanía al frente de Pareto.

Por otro lado, es deseable que el conjunto de soluciones encontradas ofrezca variedad, en otras palabras, se busca que las soluciones candidatas se encuentren distribuidas dentro del espacio de soluciones factibles lo más uniformemente posible. Lo anterior con el objetivo de poder observar las diversas propiedades de dichas soluciones de una manera más clara. A su vez, esto le facilita al DM el llevar a cabo el proceso de toma de decisiones, al poder diferenciar más fácilmente aquellas soluciones que se acerquen más a sus preferencias.

En la Figura 2.4 se muestra la aplicación de la diversidad, ya que, aunque todas las soluciones se encuentran sobre el frente de Pareto, se descartan aquellas soluciones que están demasiado cercanas entre sí.

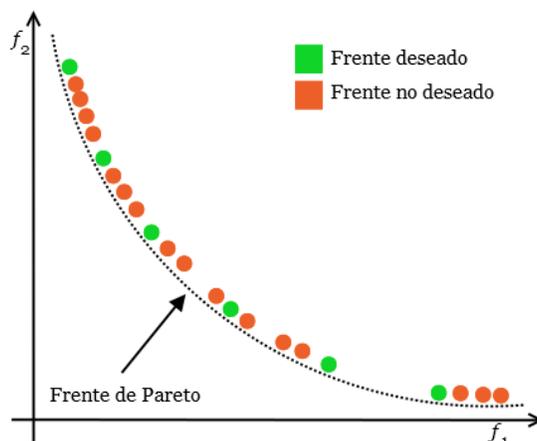


Figura 2.4. Clasificación de soluciones de acuerdo con su distribución en el frente de Pareto.

2.3 Algoritmos evolutivos multiobjetivo

Una de las técnicas utilizadas para solucionar problemas de optimización de alta dificultad es la aplicación de algoritmos que aplican un cierto grado de aleatoriedad para encontrar soluciones de buena calidad. Esto se conoce como optimización estocástica. Las metaheurísticas son la clase más general de este tipo de algoritmos [Talbi, 2009]. Dentro de las metaheurísticas se encuentran los algoritmos evolutivos, los cuales han sido ampliamente utilizados en el campo para solucionar MOPs.

Por la misma naturaleza aleatoria de estos algoritmos, no se puede garantizar que las soluciones generadas sean siempre óptimas, ya que pueden no ser necesariamente las mejores posibles, [Talbi, 2009]. Sin embargo, se ha demostrado anteriormente que una metaheurística bien diseñada puede sobreponerse a esta deficiencia, generando soluciones cercanas a las óptimas en tiempos computacionales razonables [Gendreau y Potvin, 2005].

2.3.1 Metaheurísticas evolutivas

Los algoritmos evolutivos (EA por sus siglas en inglés) son métodos que manejan un conjunto de soluciones candidatas dentro de una población, adaptando y aplicando diversas técnicas pertenecientes a la biología en dichas soluciones, tales como la selección de padres, cruza entre individuos para generar descendientes y la mutación. El EA básico se puede dividir en tres elementos [Talbi, 2009]:

- Evaluación de los individuos de una población y selección de padres.
- Cruza de estos individuos para generar una población hija.
- Técnica de unión de población padre e hija para definir una nueva población.

El último punto cobra vital importancia, ya que mediante esta técnica se busca que la población “evolucione” durante cada generación, buscando que la calidad de las soluciones encontradas vaya mejorando. [Deb et al., 2002a] hace mención que una ventaja de los EA sobre otras metaheurísticas que no manejan población es la posibilidad de encontrar múltiples óptimos de Pareto en una sola ejecución. Lo anterior no solo acelera el proceso de resolución, sino que también permite explorar múltiples áreas del espacio de soluciones factibles al mismo tiempo, reduciendo el riesgo de estancamiento en óptimos locales [Goldberg, 1989].

Además de lo anterior, [Goldberg, 1989] define otras ventajas de la aplicación de EAs para solucionar problemas de optimización, entre las que destacan las siguientes:

- No requieren conocimiento previo del problema a resolverse.
- Pueden combinarse con otras técnicas de búsqueda.
- Pueden trabajarse de forma paralela sin mayores complicaciones.
- Fáciles de implementar y usarse.

2.3.2 Algoritmos evolutivos multiobjetivo

Como se mencionó anteriormente, una ventaja de la aplicación de los EAs para solucionar problemas de optimización es su capacidad para encontrar múltiples soluciones óptimas de

Pareto en una sola ejecución. Esto, posiciona a los EAs como candidatos adecuados para solucionar MOPs. De hecho, se han propuesto múltiples trabajos dentro de la literatura para solucionar este tipo de problemas bajo diferentes condiciones y restricciones [Deb et al., 2002a; Tan et al., 2008; Li et al., 2012]. Los EAs adaptados para solucionar MOPs se denominan algoritmos evolutivos multiobjetivo (MOEA por sus siglas en inglés).

Una situación común que se presenta en los algoritmos para solucionar MOPs, incluidos los MOEAs, es generar demasiadas soluciones no dominadas, dificultando el proceso de toma de decisión del DM al tener demasiadas opciones a seleccionar. Dentro de la literatura se han propuesto diversas estrategias buscando reducir el número de soluciones disponibles para reducir la carga en el DM. Una de las formas más comúnmente utilizadas es el uso de una región de interés (ROI por sus siglas en inglés), la cual se define en base a las preferencias del DM y las soluciones no dominadas encontradas por el algoritmo [Adra et al., 2007].

La Figura 2.5 se muestra un ejemplo del conjunto de soluciones generadas por un MOEA que no maneja preferencias contra un MOEA que si las maneja. La imagen de la izquierda devolverá al DM todas las soluciones no dominadas pertenecientes al frente de Pareto. Por otro lado, la imagen de la derecha muestra un MOEA que incorpora las preferencias del DM, reduciendo el área del cual se toman las soluciones que se le presentaran al DM. Esto facilita su proceso de toma de decisión.

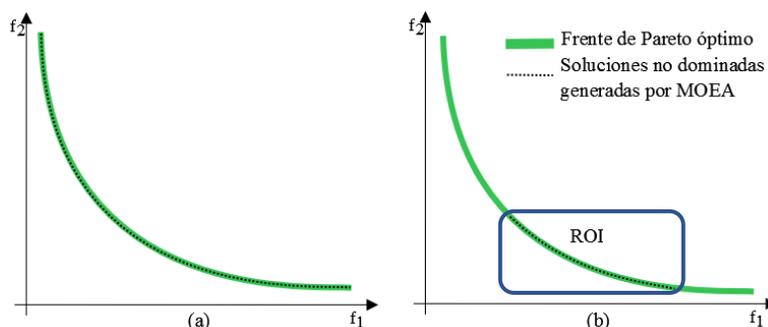


Figura 2.5. Conjunto de soluciones no-dominadas generadas por: (a) MOEA general (b) MOEA basado en preferencias.

2.3.3 Estrategias de incorporación de preferencias en MOEA

Dentro de la literatura se han propuesto diversas adaptaciones de MOEAs con el fin de incorporar las preferencias del DM dentro de su proceso y encontrar soluciones cercanas al ROI. En el trabajo presentado por Bechikh, et al., [Bechikh et al., 2015] se propuso una clasificación de los diversos enfoques utilizados por los MOEAs para manejar preferencias:

Asignación de pesos. A cada función objetivo se le define un cierto valor normalizado, el cual corresponde a la de preferencia del DM sobre dicho objetivo respecto a los demás. Una de las técnicas más comunes incluye la transformación de un MOP en un problema de un solo objetivo mediante métodos de agregación tales como la suma ponderada o el método de Tchebycheff [Borges y Hansen, 1998].

Ranking de soluciones. El DM clasifica el conjunto de soluciones (o un subconjunto de estas) para establecer un ranking que permita definir sus preferencias.

Ranking de objetivos. El DM establece sus preferencias de forma progresiva o *a priori*. A partir de esto, se clasifican un conjunto (o subconjunto) de las soluciones disponibles en base a los valores de sus funciones objetivo y las preferencias definidas. Usualmente, mediante la incorporación de un peso a cada objetivo.

Punto de referencia. Se establece uno o múltiples puntos de referencia dentro del espacio de búsqueda de las funciones objetivo. Dichos puntos representan las preferencias del usuario. El MOEA busca acercar sus soluciones a dichos puntos.

Compensación entre objetivos. Se establece un valor de intercambio aceptable para cada par de objetivos. Se define el nivel aceptable de degradación de un objetivo al mejorar el otro por una unidad. Esto permite comparar soluciones, revisando la diferencia entre sus funciones objetivo y si su nivel de mejora o degradación son aceptables. A partir de esto, es posible definir una relación de dominancia entre cada pareja de soluciones candidatas.

Superación entre soluciones. Las soluciones son ordenadas en base a valores de superación. Se define una relación de preferencia entre cada par de soluciones en base a un índice de credibilidad que utiliza los valores de las funciones objetivo de las soluciones y un vector de pesos que corresponde a las preferencias del DM.

Función de deseabilidad. Cada función objetivo es reemplazada con una función de deseabilidad, en el cual se incorporan las preferencias del DM. Es posible transformar un MOP en un problema de un solo objetivo mediante dicha función.

2.3.4 Métricas de desempeño

Es deseable conocer la calidad de las soluciones generadas por MOEAs al aplicarse en un problema de optimización. Por ello, es necesario tener algún tipo de criterio que evalúe la calidad de los resultados obtenidos mediante algún valor, lo cual permite llevar a cabo una comparación entre la calidad de las soluciones obtenidas entre diversos MOEAs. [Van Veldhuizen, 1999] utiliza el término de métricas para aquellos criterios que permitan evaluar alguno de los distintos aspectos sobre el desempeño de un algoritmo aplicado en un problema.

Como se mencionó en la Sección 2.2.7, la optimización multiobjetivo tiene dos propósitos. Primero, que el conjunto de soluciones encontradas sean lo más cercanas posibles a conjunto de óptimos de Pareto, y segundo, que dichas soluciones se encuentren lo más uniformemente distribuidas en el espacio de solución. Debido a esto se han propuesto múltiples métricas de desempeño que buscan evaluar uno o ambos aspectos de la mejor manera posible. Dos de las métricas de desempeño más comúnmente utilizadas son:

- Distancia generacional invertida [Sierra y Coello, 2005]. Para cada uno de los puntos del frente óptimo de Pareto, se encuentra la solución más cercana perteneciente al conjunto de soluciones obtenidas y se mide su distancia.
- Hipervolumen [Zitzler y Thiele, 1999]. Se mide el volumen del espacio dominado por cada solución del conjunto de soluciones obtenidas dentro del espacio de soluciones.

La primera métrica se enfoca únicamente en evaluar la convergencia del conjunto de soluciones obtenidas, mientras que la segunda busca evaluar tanto la convergencia como la diversidad bajo un mismo valor.

Ninguna métrica por si sola puede medir de manera satisfactoria el desempeño de un MOEA. Cada métrica puede proporcionar una cuantificación del desempeño específico, pero incompleto y solo puede ser efectiva bajo ciertas condiciones [Bechikh, 2013]. Es por ello por lo que usualmente se aplican múltiples métricas para evaluar un MOEA, buscando cubrir todos los aspectos que se deben evaluar.

2.4 Optimización multiobjetivo dinámica

En la vida real es bastante común que diversos elementos dentro de los problemas de optimización multiobjetivo sufran modificaciones durante el proceso de resolución. Una modificación en las restricciones o las funciones objetivo puede llevar a un cambio drástico en el entorno del problema, modificando el espacio de soluciones factibles e incluso el conjunto de soluciones óptimas para dicho problema [Helbig, 2012].

Debido a la naturaleza dinámica de este problema los algoritmos evolutivos multiobjetivo dinámicos (DMOEA por sus siglas en inglés) enfrentan un conjunto de retos a superar. Entre estos retos se encuentran la falta de funciones de referencia estándar, así como la falta de una métrica estándar para su desempeño, lo cual provoca que no sea posible comparar de manera regular la calidad de los resultados entre diferentes algoritmos [Helbig y Engelbrecht, 2013].

2.4.1 Modelo matemático de un problema multiobjetivo dinámico

El modelo matemático de un problema dinámico de optimización multiobjetivo (DMOP por sus siglas en inglés) es bastante similar con respecto a un MOP. El dinamismo en un DMOP es incorporado mediante la introducción de una variable de tiempo t , la cual determina los diferentes periodos de un DMOP. Cada periodo ofrece un entorno diferente del problema a resolver [Azzouz et al., 2017].

$$F(\vec{x}, t) = [f_1(\vec{x}, t), f_2(\vec{x}, t), \dots, f_k(\vec{x}, t)], \quad f_i: R^n \rightarrow R, \quad (2.9)$$

sujeto a:

$$g_i(\vec{x}, t) \leq 0; \quad i = 1, 2, \dots, m, \quad (2.10)$$

$$h_j(\vec{x}, t) = 0; \quad j = 1, 2, \dots, p. \quad (2.11)$$

En otras palabras, se busca definir el vector de variables de decisión que optimice todas las funciones objetivo mientras que se satisfagan las restricciones durante un periodo t , ya que estas definen la región factible del problema en ese periodo.

2.4.2 Clasificación de los problemas de optimización multiobjetivo dinámicos

Los cambios en el entorno de un DMOP pueden ser diferentes durante el transcurso del proceso de solución de este. Para dichos cambios se consideran dos factores [Helbig, 2012]:

- Severidad temporal. La frecuencia en la cual se presenta un cambio en el entorno.
- Severidad espacial. La extensión en el cambio de posición de las soluciones óptimas.

Debido a lo anterior se han sugerido múltiples formas de clasificación para los DMOPs [Azzouz et al., 2017]:

- Clasificación basada en la severidad temporal. Mientras mayor sea la frecuencia de los cambios en el entorno menor será el tiempo disponible para adaptarse a los cambios, lo cual aumenta la dificultad del problema.
- Clasificación basada en la severidad espacial. Dependiendo de qué tan drástico haya sido el cambio en el entorno mayor será la dificultad. Un problema que haya sufrido un cambio menor puede llegar a utilizar la información obtenida en estados previos de su entorno para acelerar el proceso de adaptación e identificación de las nuevas soluciones óptimas. Sin embargo, si el problema sufrió un cambio mayor existe la probabilidad de que cada instancia del entorno sea completamente diferente, por lo que debe reiniciarse el algoritmo, alentando drásticamente el proceso de adaptación.
- Clasificación basada en predictibilidad. La predictibilidad del cambio indica la regularidad con la ocurren las modificaciones en el entorno del problema. Bajo esta forma, los problemas pueden clasificarse como cíclicos si los cambios ocurren en periodos definidos y son deterministas o acíclicos si los cambios ocurren de forma completamente aleatoria.

Dentro de la literatura se propone una clasificación en base a los cambios ocurridos en el *POF** y el *POS** [Farina et al., 2004]. A partir de esto, se definen cuatro tipos diferentes de DMOPs, cuya relación se muestra en la Tabla 2.1:

- Tipo I. Existe un cambio en el conjunto óptimo de variables de decisión, pero el frente de Pareto se mantiene idéntico.
- Tipo II. Tanto el conjunto óptimo de variables de decisión, así como el frente de Pareto sufren un cambio.
- Tipo III. Existe un cambio en el frente de Pareto, pero el conjunto óptimo de variables de decisión se mantiene sin cambio.
- Tipo IV. Los cambios en el problema no afectan el conjunto óptimo de variables de decisión ni el frente de Pareto.

Tabla 2.1. Tipos de entorno de optimización multiobjetivo dinámico [Farina et al., 2004].

Frente de Pareto	Conjunto óptimo de variables de decisión	
	Sin cambio	Con cambio
Sin cambio	Tipo IV	Tipo I
Con cambio	Tipo III	Tipo II

2.4.3 Cambios en el frente de Pareto

Los cambios en el entorno del problema de optimización multiobjetivo durante su proceso de resolución pueden llevar a diversos cambios en el frente de Pareto, entre los cuales se encuentran los siguientes [Helbig, 2012]:

- Varias soluciones dentro del frente de Pareto se vuelven dominadas por otras soluciones factibles y, por lo tanto, dejan de pertenecer al frente.
- La forma del frente de Pareto se mantiene igual. Sin embargo, su ubicación en el espacio de objetivos cambia en el transcurso del tiempo. Este tipo de cambio se relaciona con los DMOPs tipo I, explicados en la sección anterior.
- La forma del frente de Pareto cambia en el transcurso del tiempo. Un ejemplo de esto es que el frente pasa de ser continuo a segmentarse dentro del espacio de búsqueda. También puede ocurrir que, dentro de la representación gráfica, pase de tener una forma cóncava a una convexa o viceversa. Este tipo de cambio está relacionado con los entornos de un DMOP tipo II o III, en los cuales el frente de Pareto sufre un cambio. Lo anterior puede llevar a que los algoritmos utilizados tengan una mayor dificultad para encontrar un conjunto diverso de soluciones después de un cambio.
- La densidad de las soluciones en el frente de Pareto cambia durante el transcurso del tiempo. Esto se puede deber a dos motivos, primero, el aumento o la reducción del número de soluciones pertenecientes a este frente. El otro motivo es la afectación del espacio de soluciones factibles, lo cual cambia la densidad de cada solución factible, incluyendo las que pertenecen al frente.

2.4.4 Algoritmos evolutivos multiobjetivo dinámicos

Una complicación que puede surgir en los MOEAs al enfrentar problemas dinámicos, tales como DMOP, es la falta de adaptabilidad a los cambios que presentan dichos problemas, lo cual puede llevar a una pérdida drástica en la calidad de las soluciones. Es por ello por lo que se han propuesto dentro de la literatura múltiples MOEAs capaces de adaptarse a las modificaciones de este tipo de problemas, llamando diversas técnicas de adaptación en busca de mantener buenos resultados. Estos llevan el nombre de algoritmos evolutivos multiobjetivo dinámicos (DMOEA por sus siglas en inglés).

La técnica de adaptación de un DMOEA está obligada de manera implícita a cumplir con los dos propósitos de la optimización multiobjetivo, convergencia y diversidad. Debido a esto, varias técnicas han sido propuestas, las cuales afrontan un DMOP desde diferentes enfoques. Estos enfoques se pueden clasificar de la siguiente manera [Azzouz et al., 2017]:

- Enfoque basado en la diversidad (ED). Buscan mantener la diversidad de las soluciones alterando el contenido de la población en un cierto grado.
- Enfoque basado en la predicción de cambios (EPC). Predicen la localización de nuevas soluciones óptimas en base a información previamente recabada, detectando posibles patrones de movimiento.
- Enfoque basado en memoria (EM). Guían la dirección de búsqueda de futuras generaciones en base a información recabada por generaciones previas.
- Enfoques paralelos (EP). Aplican técnicas de paralelización para resolver un problema de manera rápida y eficiente.
- Enfoques que convierten los DMOPs en múltiples MOPs estáticos (EMOP). Trabajan el problema en periodos, donde cada periodo tiene un entorno estático.
- Otros enfoques.

2.5 Superficie de aptitud dinámica

Una superficie de aptitud estática se define formalmente [Schiavinotto y Stützle, 2007]:

$$\Lambda_S = (S, n, f). \quad (2.12)$$

Donde S es el espacio de búsqueda tal que una solución $x \in S$, $n(x)$ es la función de vecindad para cada elemento dentro de S , mientras que $f(x)$ es la función de aptitud correspondiente para los elementos dentro del espacio de búsqueda.

La noción anterior puede ser aplicada para el análisis de una superficie de aptitud dinámica. Sin embargo, se debe tomar en cuenta los constantes cambios a los que se enfrenta dicha superficie, lo cual a su vez puede llevar a modificaciones tanto en los valores obtenidos por la función de aptitud utilizada en cada una de las soluciones dentro del espacio de búsqueda como en la función misma. [Richter, 2013] define una nueva formulación tomando en cuenta estas consideraciones:

$$\Lambda_D = (S, n, K, F, \Phi_f). \quad (2.13)$$

Las variables S y n son equivalentes a su versión estática; K es una escala para medir y ordenar los cambios dinámicos; F son todas las funciones de aptitud en un tiempo $k \in K$ donde cada función $f \in F$ también es dependiente del tiempo al momento de dar valores de aptitud a cada uno de los elementos dentro del espacio de búsqueda S . Finalmente, Φ_f es un mapeo que describe la trayectoria de las funciones encontradas, con el objetivo de saber cómo se modifica la función de aptitud con el paso del tiempo.

2.5.1 Unificación de valores objetivo

La formulación anterior está enfocada en problemas de optimización dinámicos de un solo objetivo, por lo que no puede ser utilizada de forma directa en problemas multiobjetivo. Para poder utilizar dichas ecuaciones en un DMOP primero se requiere de transformar sus múltiples objetivos en una sola función objetivo.

Para llevar a cabo lo anterior [Borges y Hansen, 1998] y más recientemente [Li y Deb, 2017] propusieron la conversión de un MOP a un problema de un objetivo mediante dos métodos, la suma ponderada, y el método ponderado de Tchebycheff.

Suma ponderada. Los valores de los m objetivos del DMOP son agregados en una función escalar, en donde a cada función se le asigna un peso λ .

$$\min g_t^{ws}(x|\lambda) = \sum_{i=1}^m \lambda_i f_{i,t}(x), \quad (2.14)$$

sujeto a

$$\sum_{i=1}^m \lambda_i = 1. \quad (2.15)$$

Método ponderado de Tchebycheff. Se aplica una regla de máximos y mínimos apoyándose de un punto de referencia z para convertir un problema con m objetivos en un problema de un objetivo. De igual manera que la suma ponderada, se le asigna a cada función objetivo un cierto peso λ , donde el peso total se encuentra sujeto a la ecuación (2.15).

$$\min g_t^{tch}(x|\lambda) = \max \lambda_i |f_{i,t}(x) - z_i|, \quad i \in \{1, \dots, m\}. \quad (2.16)$$

Donde, para ambos métodos se consideran las siguientes nomenclaturas:

- x es el vector de variables de decisión de una solución candidata.
- m es el número de funciones objetivo del DMOP.
- $f_{i,t}(x)$ es el valor obtenido por x en el i -ésimo objetivo durante el periodo t .

2.5.2 Evolucionabilidad

Un inconveniente bastante común de las medidas de superficie de aptitud tanto estáticas como dinámicas es que solo se enfocan en obtener propiedades del problema que se busca resolver, sin embargo, esto puede presentar un problema. Lo anterior se debe a que estas medidas no toman en consideración el algoritmo de optimización que se está empleando en ese momento en busca de obtener soluciones de buena calidad.

La evolucionabilidad se define como la capacidad de una población de generar variantes con mayor aptitud que las existentes [Altenberg, 1994]. La ecuación (2.17) presenta una definición formal para buscar definir un valor a esta propiedad [Smith et al., 2002]. Sea una solución candidata x , se genera una vecindad $N(x)$ y un subconjunto $N^*(x) = \{c \mid c \in N(x), f(c) \geq f(x)\}$, la evolucionabilidad E_a de dicha solución se calcula de la siguiente manera,

$$E_a = \frac{|N^*(x)|}{|N(x)|}. \quad (2.17)$$

Mientras E_a tenga un mayor valor, más probable es que la solución evolucione de manera positiva, por lo que se dice que el problema es más fácil. Lo anterior se puede combinar con el algoritmo utilizado para solucionar el problema en cuestión calculando la evolucionabilidad de las soluciones generadas. Estableciendo de esta manera una relación entre la superficie de aptitud y un algoritmo de optimización.

2.6 Hiperheurísticas

Las heurísticas y metaheurísticas son capaces de explorar el espacio de búsqueda de soluciones, encontrando y seleccionando como soluciones candidatas aquellas que se apeguen más a sus criterios. Sin embargo, debido a la dificultad y características de algunos problemas, así como la existencia de una amplia variedad de alternativas disponibles para solucionar dichos problemas, es deseable que exista una metodología que permita seleccionar dentro de ese conjunto de heurísticas y metaheurísticas, aquellas que generen las mejores soluciones para el estado actual del problema.

Por otro lado, el teorema “*No Free Lunch*” [Wolpert y Macready, 1997] menciona que, dentro de un conjunto de problemas, tales como problemas de optimización, si existe un

grupo de algoritmos que funciona mejor en un cierto subconjunto de ellos, entonces deben existir otro grupo de algoritmos diferentes los cuales generan mejores resultados en otro subconjunto de estos problemas. Esto se puede aplicar incluso dentro de un mismo problema, donde dentro de un subconjunto específico de periodos estáticos de este, un conjunto de algoritmos puede generar buenos resultados, mientras que en otro subconjunto de periodos son otros algoritmos los que proveen mejores resultados.

2.6.1 Concepto de hiperheurística

Las hiperheurísticas se definen como una metodología automatizada de alto nivel para la selección o generación de heurísticas con el objetivo resolver problemas de búsqueda computacionalmente difíciles [Burke et al., 2010].

Dicha metodología o heurística de alto nivel (HLH por sus siglas en inglés) actúa sobre un espacio de búsqueda compuesto de un conjunto de heurísticas de bajo nivel (LLH por sus siglas en inglés) seleccionando aquella o aquellas que concluya como las más adecuadas a utilizar en cada punto de evaluación. Después, los LLHs actúan sobre el espacio de búsqueda de soluciones, seleccionando el conjunto de soluciones de mejor calidad en base a sus técnicas [Soubeiga, 2003].

Esto quiere decir que las hiperheurísticas no actúan directamente con el conjunto de soluciones, sino que definen cuales son los algoritmos, métodos y técnicas que llevaran a cabo el proceso de búsqueda dentro de dicho conjunto.

La complejidad de los HLHs como los LLHs puede variar dependiendo tanto de la metodología diseñada, así como de la complejidad del problema a resolver y las técnicas conocidas para solucionar dicho problema. Es por esta razón que tanto los HLHs como LLHs pueden ser heurísticas o metaheurísticas, sin embargo, es su aplicación dentro del proceso de resolución del problema lo que define su posición en el proceso.

2.6.2 Clasificación de hiperheurísticas

En un trabajo enfocado hacia en este rubro se propone una clasificación de las hiperheurísticas en base de dos criterios: la retroalimentación y la naturaleza del espacio de búsqueda de heurísticas [Burke et al., 2010]. En base a la retroalimentación de las hiperheurísticas se establecen tres diferentes categorías:

- *Sin aprendizaje*. No se utiliza retroalimentación durante todo el proceso. Esta clase de hiperheurísticas basan la selección de LLHs en la calidad de la solución actual o en la prueba de cada una de estas heurísticas siguiendo un orden predeterminado para determinar aquella que ofrezca un mejor resultado [Soubeiga, 2003].
- *Aprendizaje en línea*. Estas hiperheurísticas llevan el proceso de aprendizaje mientras se está llevando a cabo la solución de una de las instancias del problema definido. Lo anterior permite definir los LLHs a utilizar en cada etapa de dicha instancia.
- *Aprendizaje fuera de línea*. En esta categoría se establece un conjunto de instancias de entrenamiento, en donde se recopila toda la información obtenida previamente con el

objetivo de definir un método que pueda ser generalizado para las instancias aún no vistas por el sistema.

Por otro lado, en base de la naturaleza del espacio de búsqueda de las heurísticas, las hiperheurísticas se clasifican de la siguiente manera:

- *Selección de heurísticas.* A partir de un conjunto de LLHs existentes, se definen metodologías con el objetivo de seleccionar la combinación de éstas que produzca soluciones de mayor calidad. Para esto se puede construir una solución, seleccionando el LLH más adecuado en cada fase de la construcción; así como también se puede trabajar con soluciones completas eligiendo el subconjunto de LLHs cuya perturbación en la solución genera un resultado de alta calidad.
- *Generación de heurísticas.* Se enfocan en la generación de nuevas heurísticas en base de elementos de los LLHs existentes. Para esto, varios de los enfoques dentro de la literatura se apoyan del uso de la programación genética para el proceso de generación de heurísticas. Debido a su flexibilidad en el manejo de datos, la facilidad de identificar características positivas en el dominio del problema y la capacidad de expresar heurísticas diseñadas por el hombre en un lenguaje [Burke et al., 2009].

2.6.3 Fases de una hiperheurística

Como se mencionó previamente en la sección 2.6.1, las hiperheurísticas no actúan de manera directa con el conjunto de soluciones, sino que es el LLH seleccionado el que lleva a cabo este proceso. Sin embargo, es deseable que el resultado obtenido por dicha heurística sea satisfactorio, por lo que es necesario establecer un control sobre los resultados ofrecidos por los LLHs. Debido a esto [Bilgin et al., 2006] establece que una hiperheurística está compuesta por dos fases, cuya representación gráfica se presenta en la Figura 2.6:

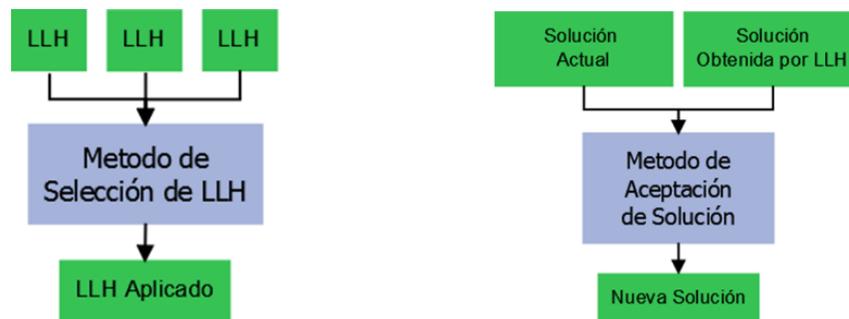


Figura 2.6. Fases de una hiperheurística.

Método de selección de heurísticas. Define cuál va a ser el LLH o subconjunto de LLHs dependiendo del funcionamiento del HLH, a utilizar en el problema que se busca resolver durante un cierto periodo hasta el próximo proceso de selección de LLH. Los métodos de selección de LLH se dividen en dos tipos [Burke et al., 2010]:

- *Adaptativo.* Estos métodos utilizan un proceso de aprendizaje que les permite llevar a cabo una selección utilizando información previamente obtenida.

- *No adaptativo*. Métodos de selección aleatorios o que utilizan información obtenida solamente durante el actual proceso de selección.

Método de aceptación de solución. Evalúa el resultado obtenido por el o los LLHs que trabajaron durante un periodo determinado, comparándolo con la solución que se tiene actualmente. El resultado obtenido se acepta como nueva solución solamente si satisfacen los criterios determinados por el método de aceptación de solución. Los criterios de aceptación se pueden dividir en dos tipos [Burke et al., 2010]:

- *Deterministas*. El resultado siempre va a ser el mismo para un conjunto de soluciones. Este tipo de criterios se enfocan en aspectos como los valores objetivo y valores de las variables de decisión.
- *No determinista*. El resultado puede variar para un conjunto de soluciones. Generalmente, estos criterios se apoyan en la aleatoriedad.

Capítulo 3

Antecedentes

En este capítulo se presenta una breve descripción de varios de los trabajos de investigación que tienen mayor relevancia dentro del campo de la optimización multiobjetivo dinámica. Dichos trabajos utilizan algoritmos evolutivos con el objetivo de resolver diversos DMOPs. Posterior a esto se da una relación de varios problemas dinámicos en los cuales se aplican dichos algoritmos, así como diversas métricas aplicadas para evaluar su desempeño. También se mencionan varias técnicas para llevar a cabo un análisis de la superficie de aptitud. Después, se describen diversas hiperheurísticas aplicadas en MOPs estáticos y dinámicos. Finalmente se mencionan varios problemas abiertos dentro de esta área.

3.1 Algoritmos propuestos para solucionar DMOPs

Los algoritmos evolutivos han sido utilizados previamente en los problemas de optimización dinámica de un solo objetivo (DSOP por sus siglas en inglés). Sin embargo, debido a que dentro de la optimización multiobjetivo estos algoritmos enfrentan un constante cambio en el entorno del problema, se generan problemas de adaptación a dichos cambios al sufrir una posible falta de diversidad bajo la convergencia obtenida [Azzouz et al., 2017]. Por esta razón se han propuesto varios mecanismos adicionales para mantener diversa la población. A continuación, se describen algunos trabajos relacionados a la aplicación de algoritmos evolutivos para la optimización multiobjetivo dinámica.

Dynamic Non-Dominated Sorting Genetic Algorithm II (DNSGA-II) [Deb et al., 2007]. Este algoritmo evolutivo es una modificación del NSGA-II [Deb et al., 2002a], el cual busca un nuevo frente óptimo de Pareto tan pronto como se detecte un cambio en el problema. Con el objetivo de introducir diversidad en cada cambio, se sustituye un porcentaje de la población actual por soluciones generadas aleatoriamente o mutaciones de soluciones existentes. Reevaluando posteriormente el nuevo conjunto obtenido.

Dynamic Queuing Multi-Objective Optimizer (D-QMOO) [Hatzakis y Wallace, 2006]. Este método combina una técnica de predicción con un algoritmo evolutivo para pronosticar la locación del conjunto óptimo de soluciones. Se definen un grupo de individuos en una posición cercana a donde estará el próximo óptimo. Este conjunto se utiliza para sembrar una nueva población cuando ocurra el cambio en el entorno.

The Dynamic Multi-objective EA with Core Estimation of Distribution (CDDMEA) [Liu, 2010]. En su trabajo, Liu incorpora una estimación básica del modelo de distribución con el objetivo de predecir la ubicación de las soluciones óptimas de Pareto dentro del espacio de búsqueda en el nuevo entorno que se genere. Cuando un cambio es detectado, el algoritmo utiliza la información recolectada de los entornos previos para predecir la ubicación del frente de Pareto en el siguiente periodo. La nueva población se obtiene agregando las diferencias

entre las soluciones centrales de entornos pasados con las soluciones actuales para generar una predicción de la nueva población.

The Dynamic Competitive Cooperative CO-EA (dCOEA) [Goh y Tan, 2009]. Para este trabajo se presenta un algoritmo multiobjetivo coevolutivo que hibrida mecanismos competitivos y cooperativos, simulando a la naturaleza, buscando resolver DMOPs. La población se divide en múltiples subconjuntos donde los individuos compiten entre sí, cada subpoblación representa a un subcomponente particular del problema. Los eventuales ganadores de cada subconjunto cooperaran para evolucionar en mejores soluciones.

The Multi-Strategy Ensemble MOEA (MS-MOEA) [Wang y Li, 2010]. La velocidad de convergencia es acelerada por un mecanismo de creación de descendientes alimentado por operadores genéticamente adaptativos y diferenciales. Se reinicia la nueva población mediante mutación cuando ocurre un cambio en el entorno.

The Dynamic Multi-objective EA (DMEA) [Liu y Wang, 2006]. En su trabajo, Liu y Wang proponen separar el periodo actual de un DMOP en subperiodos de una misma duración. En cada subperiodo el DMOP es aproximado por un MOP estático. Después, para cada MOP estático, se definen tanto la varianza de rango como la varianza de densidad de la población. Posteriormente, utilizando ambas varianzas, se transforma cada MOP estático en un problema estático de optimización de dos objetivos.

3.2 Algoritmos propuestos para resolver MOPs preferenciales

La existencia de un conjunto de preferencias establecidas por un DM genera otro reto que debe ser enfrentado por los algoritmos de optimización. El DM usualmente no se encuentra interesado en encontrar soluciones óptimas para todo el espacio de búsqueda disponible, sino que desea encontrar soluciones que se encuentren dentro de su región de interés. Varios trabajos han sido propuestos en los que se utilizan algoritmos evolutivos para solucionar problemas de optimización que incorporan las preferencias del DM. Dos de los trabajos más relevantes para esta tesis son NOSGA-II y WASF-GA.

Non-Outranked Sorting Genetic Algorithm II (NOSGAI) [Fernández et al., 2011b]. Es una adaptación de NSGA-II, el cual incorpora relaciones de preferencia entre cada uno de los individuos de una población y utilizando dichas referencias dentro del proceso de generación de frente no dominados y *Crowding Distance*. NOSGA-II hace comparaciones entre cada pareja de soluciones candidatas y determina una relación de preferencia entre ellas bajo el concepto de “la solución x es al menos tan buena como la solución y ”, esto se conceptualiza como superación. Esta superación está determinada no solamente por la dominancia de Pareto, sino también por un vector de pesos definido por las preferencias del DM. Dichas preferencias permiten determinar si la relación de superación es estricta, débil o existe indiferencia entre ambas soluciones.

Weighting achievement scalarizing function genetic algorithm (WASF-GA) [Ruiz et al., 2015]. Este MOEA considera las preferencias del DM para definir un punto de referencia dentro del espacio de búsqueda de funciones objetivo. La intención de WASF-GA es buscar

definir un ROI lo más cercano posible a dicho punto de referencia. Para esto, se utiliza una función de escalarización de logros (ASF por sus siglas en inglés) apoyada de un conjunto de vectores de pesos normalizados distribuidos de forma uniforme y el uso de múltiples frentes, siguiendo una estrategia similar a la utilizada por NSGA-II. Las soluciones candidatas se clasifican durante cada generación utilizando ASF y los múltiples vectores de peso. Toma consideración especial en penalizar el valor de aptitud de cada solución principalmente respecto al valor objetivo en el que se encuentre más al valor correspondiente del punto de referencia.

Además de los dos trabajos anteriores presentados, existe una amplia variedad de propuestas dentro de la literatura para solucionar MOPs estáticos preferenciales. El estudio realizado en trabajos previos [Bechikh et al., 2015] presenta un listado de los trabajos más notables de este rubro. En esta tesis solo se mencionó WASF-GA y NOSGA-II al ser relevantes para el contenido de este documento en las siguientes secciones.

Hasta nuestro conocimiento, existen pocos trabajos que toman este concepto y lo aplican dentro de la optimización dinámica multiobjetivo. Esto se debe a que resolver un DMOP con preferencias establece varios niveles de complejidad al problema. No solamente se debe hacer enfoque en considerar las preferencias del DM, sino que también debe tomarse en cuenta que el entorno que rodea al problema puede cambiar en cualquier momento. Entre los trabajos más relevantes dentro de esta área se encuentran:

DNSGA-II. Incorpora un vector de pesos que representan las preferencias del DM dentro del proceso de solución. Se obtiene un valor normalizado de la primera función objetivo, seleccionando la solución que más se acerque al peso preferencial definido por el DM para dicho objetivo. En este caso se enfoca únicamente en el primer objetivo.

DNSGA-II modificado [Roy y Mehmen, 2008]. Versión de DNSGAI la cual reemplaza las funciones objetivo por un conjunto de funciones de deseabilidad. Esto le permite al DMOEA propuesto trabajar bajo un conjunto de funciones definidas para el problema que se busca resolver y un conjunto específico de preferencias por parte del DM. Quizás el único inconveniente de esta modificación es que el DMOEA es aplicable únicamente para problemas similares al que fue presentado por los autores.

Sphere-dominance preference immune-inspired algorithm (SPIA) [Liu et al., 2010]. DMOEA que incorpora preferencias dentro de su proceso mediante la utilización de un punto de referencia. En base de ese punto se genera una hiperesfera dentro del espacio de búsqueda de funciones objetivo la cual define el ROI. Esto le permite al DM agregar sus preferencias sin afectar el proceso del DMOEA. El criterio de dominancia entre las soluciones se basa en si las soluciones se encuentran dentro de la hiperesfera.

Interactive Dynamic Multi-Objective Decision Making Evolutionary Algorithm (InDM2) [Nebro et al., 2018]. Incorpora preferencias mediante la inserción de puntos de referencia *a priori* y de manera interactiva durante el proceso de solución. InDM2 permite la incorporación de múltiples puntos de referencia, así como modificación de puntos ya existentes. Cada vez que el DMOEA detecta un cambio en el entorno del problema o en el

conjunto de puntos preferencias establecidos, se lleva a cabo un proceso de adaptación al cambio en el cual se sigue el proceso definido por DNSGA-II-A [Deb et al., 2007], al reemplazar un porcentaje de la población existentes por nuevas soluciones generadas de forma aleatoria. InDM2 permite trabajar con puntos de referencia factibles y no factibles.

Teniendo en cuenta la información anterior y los comentarios realizados dentro de la literatura [Helbig y Engelbrecht, 2016], se puede visualizar que el campo de la optimización de DMOPs preferenciales tiene una amplia área sin estudiar o con poca investigación. Existe la necesidad de profundizar la revisión de DMOEAs con incorporación de preferencias enfocados en puntos de referencia.

Una situación común al trabajar con este enfoque es la falta de flexibilidad debido a que los puntos son específicos a un problema. Por lo tanto, es necesario definir un método que permita generar puntos de referencia para cualquier tipo de problema en base a las preferencias del DM. De igual forma, hasta nuestro conocimiento, no existe un DMOEA preferencial basada en relaciones de superación. En este sentido, esta tesis propone...

3.3 Algoritmos propuestos para resolver MaOPs

Dentro de la literatura existen múltiples propuestas para solucionar MaOPs mediante el uso de algoritmos evolutivos. Entre los más relevantes o relacionados con el enfoque definido en este proyecto se encuentran los siguientes:

MOEA dinámico para resolver problemas de muchos objetivos (MDMOEA) [Zou et al., 2008]. MOEA que se apoya del principio de la termodinámica, simulando procesos de recocido simulado para resolver MaOPs. MDMOEA reemplaza el criterio de dominancia entre soluciones basado en dominancia de Pareto por una dominancia más flexible, denominada L-dominancia. Se dice que una solución L-domina a otra si es mejor en un mayor número de funciones objetivo y es superior bajo una función de agregación determinada. MDMOEA es capaz de solucionar de manera efectiva MaOPs de hasta nueve objetivos.

Modificaciones de NSGA-II para MaOP [Ishibuchi et al., 2008]. En este trabajo se llevaron a cabo pruebas del desempeño de NSGA-II para resolver MaOPs alterando dos de sus propiedades. La primera consistía en asignar una distancia cero a las soluciones extremas al evaluar *Crowding Distance*. El segundo cambio modifica la función objetivo buscando incrementar la presión de selección hacia el frente de Pareto. Los resultados muestran que modificar las funciones objetivo mediante la aplicación de funciones de agregación es una alternativa viable.

MOEA basado en índice de deseabilidad (DI-MOEA) [Trautmann et al., 2013]. Este enfoque busca solucionar MaOPs con preferencias incorporadas por un DM. Dichas preferencias son representadas mediante el uso de funciones de deseabilidad, las cuales son unificadas en una función de agregación. El valor resultante es definido por los autores como índice de deseabilidad (DI). DI se utiliza como parte de los criterios de selección cuando se define el conjunto de soluciones no dominadas. DI-EMOA fue probado en un conjunto de MaOPs de hasta seis objetivos, probando ser capaz de converger con el frente de Pareto

óptimo de forma más rápida y eficiente que diversos MOEAs que incorporan preferencias mediante el uso de funciones de deseabilidad y puntos de referencia.

Algoritmo de estimación de hipervolumen para optimización multiobjetivo (HypE) [Bader y Zitzler, 2011]. Es un algoritmo que se apoya de indicadores de hipervolumen para determinar la dominancia de Pareto entre las soluciones. Se utiliza una simulación de Monte Carlo para aproximar el hipervolumen a valores exactos. Esto permite un intercambio de exactitud por rapidez computacional, lo cual se denota de forma más crítica conforme aumenta el número de objetivos de un MaOP. HypE fue probado en MaOPs de hasta 50 objetivos, obteniendo resultados satisfactorios y efectivos en comparación de otros MOEAs.

Metaheurística de muchos objetivos basada en el indicador R2 (MOMBI-II) [Hernández y Coello, 2015]. Es un MOEA que considera el uso de funciones de escalarización por logros y la cercanía de las soluciones obtenidas al frente de Pareto óptimo. Los autores establecen el uso de tres puntos esenciales en el espacio de búsqueda: los puntos óptimo y nadir, así como un punto de referencia para el uso de la ASF. MOMBI-II utiliza el método de Tchebycheff para asignar un nivel de jerarquía a cada solución en base a los valores obtenidos por la ASF en cada uno de los diferentes vectores de peso. Dicho nivel de jerarquía se utiliza para determinar las soluciones que poblarán la siguiente generación. En el trabajo original, se utiliza el punto de origen como punto de referencia. Sin embargo, esto abre la posibilidad de utilizar este MOEA para MaOPs con preferencias definidas por un DM.

Como se puede ver, los trabajos previamente propuestos para la resolución de MaOPs se enfocan en entornos estáticos. Hasta nuestro conocimiento, solamente existe un trabajo propuesto para solucionar DMAOPs [Li et al., 2017]. En dicho trabajo se propone un algoritmo de optimización evolutivo para muchos objetivos con manejo dinámico de restricciones. Ese algoritmo soluciona MaOPs con restricciones dinámicas generadas a partir de problemas de un solo objetivo con restricciones.

Sin embargo, esta propuesta no puede ser aplicada en este proyecto al enfocarse en la resolución de problemas con restricciones, los cuales se definieron como parte de las alcances y limitaciones de este trabajo de tesis. Estudios relacionados al campo remarcan la falta de trabajos enfocados en este entorno [Helbig et al., 2016].

3.4 Instancias para problemas de optimización multiobjetivo dinámicos

Dentro del campo de la optimización multiobjetivo dinámica se han desarrollado un conjunto de instancias de prueba con el objetivo de evaluar la calidad de los DMOEAs desarrollados en diversos aspectos, tales como el manejo de pocos o muchos objetivos, cambios en el frente de Pareto o el conjunto de soluciones óptimas, la severidad temporal, entre otros.

Las instancias de prueba son funciones que permiten retar la capacidad de un algoritmo para solucionar problemas que trabajan bajo diferentes condiciones. Hoy en día existe un gran número de instancias de prueba enfocados en los MOPs. Sin embargo, dichas instancias se desarrollan dentro de un entorno estático, el cual no presenta algún cambio durante su proceso de resolución. Es por ello por lo que dentro de la literatura se han propuesto múltiples

instancias de prueba de DMOPs. A continuación, se hará una breve descripción de algunas de estas instancias, las cuales tiene la mayor relevancia dentro del campo.

El conjunto de problemas FDA [Farina et al., 2004]. Se desarrollaron un conjunto de instancias de prueba enfocadas a la optimización multiobjetivo dinámica basándose en un conjunto de escenarios y una clasificación, los cuales fueron establecidos dentro de ese mismo trabajo. Para generar dichas instancias se apoyaron del conjunto de instancias de prueba estáticas ZDT [Zitzler et al., 2000] y DTLZ [Deb et al., 2002b].

Problema de prueba (G2) [Jin y Sendhoff, 2004]. Basándose de un problema de optimización multiobjetivo estático se hace una reformulación de tal manera que el nuevo problema incluya una variable susceptible a cambio en base al tiempo.

El conjunto de problemas dMOP [Goh y Tan, 2009]. Apoyándose del conjunto de reglas que definieron la generación del conjunto FDA [Farina et al., 2004], se generó un conjunto de instancias de prueba resultado de hacer modificaciones en el conjunto de instancias FDA. Este juego de instancias prueba la capacidad de un DMOEA para solucionar problemas con frente de Pareto cóncavos, convexos y con una distribución cambiante.

El conjunto de problemas ZDT dinámicos (DMZDT) [Wang y Li, 2010]. Se adaptaron las instancias estáticas del conjunto ZDT para que el resultado obtenido en sus funciones objetivo fuera afectado por una variable susceptible al tiempo. Aunque todos estos problemas son de tipo I, cada una de estas instancias ofrece una situación diferente. Ya sea un frente de Pareto convexo, cóncavo o no uniforme.

El problema WYL [Wang y Li, 2010]. En este problema, la fórmula de una de las funciones objetivo cambia *constantemente* durante el transcurso del tiempo. Las funciones objetivo que se ven modificadas están basadas en el conjunto de instancias DMZDT propuesto en ese mismo trabajo.

Versiones modificadas de los problemas FDA. Dentro de la literatura se han propuesto múltiples modificaciones a instancias pertenecientes al conjunto FDA. Por ejemplo, en un caso se propusieron versiones de los problemas FDA2 y FDA3 en las cuales una de las funciones objetivo estaba alterada con respecto a la función original [Zheng, 2007].

A continuación, en la Tabla 3.1, se presentan las propiedades y características de las instancias de prueba previamente mencionadas, incluyendo el B número de problemas que componen a cada juego, los tipos de problemas que contiene uno de ellos en base a la clasificación de DMOPs basada en POF^* y POS^* [Farina et al., 2004], como se realiza su manejo de duración de cada periodo, ya sea por un número de generaciones o evaluaciones transcurridas. También se presenta el tamaño del vector de variables de decisión recomendado por los autores y el número de objetivos que manejan esto DMOPs.

Como se puede ver en dicha tabla, las instancias revisadas trabajan únicamente con un número reducido de objetivos, a excepción de los problemas FDA4 y FDA5 los cuales manejan un número no definido M de funciones objetivo al basarse en el conjunto de problemas DTLZ. Como puede verse en base a la clasificación de DMOPs utilizada existe

una amplia diversidad existente de instancias de prueba DMOPs, los cuales involucran frentes de Pareto convexos, no convexos, mixtos, continuos y discontinuos.

Tabla 3.1. Listado de instancias de prueba DMOP.

Instancia	<i>B</i>	Clasificación de DMOP	Manejo de periodos	VARIABLES de decisión	# de objetivos
FDA	5	I, II y III	Por generaciones	FDA1: 20 FDA2: 31 FDA3: 30 FDA4-5: <i>M</i> +9	FDA1-3: 2 FDA4.5: <i>M</i>
G2	1	II	Por generaciones	2	3
dMOP	3	I, II y III	Por generaciones	10	2
DMZDT	4	I	Por evaluaciones	DMZDT1-3: 30 DMZDT4: 10	2
WYL	1	II	Por evaluaciones	No especifica	2
FDA-modificado	2	II y III	Por generaciones	FDA2: 31 FDA3: 30	2

Lo anterior permite definir un buen conjunto de instancias para probar la capacidad tanto en convergencia como en diversidad de prácticamente cualquier algoritmo propuesto en la literatura, así como los algoritmos propuestos en esta tesis.

3.5 Métricas de evaluación de desempeño de algoritmos

Es deseable contar con algún método o técnica que permita evaluar la calidad de las soluciones obtenidos por los algoritmos propuestos para solucionar un cierto problema. Estas técnicas se definen como métricas de evaluación de desempeño.

Dichas métricas generalmente miden uno o varios aspectos del conjunto de las soluciones no dominadas: el número de soluciones generadas, la convergencia respecto a un frente de Pareto óptimo o la dispersión de las soluciones dentro del espacio de búsqueda de las funciones objetivo [Zitzler et al., 2000].

3.5.1 Métricas para evaluar MOEAs

A continuación, se presenta una lista de algunas de las métricas más comúnmente utilizadas para evaluar el desempeño de los resultados obtenidos por algoritmos evolutivos multiobjetivo (MOEAs) aplicados en los problemas de optimización de hasta tres objetivos.

Proporción de soluciones no dominadas (RNI) [Tan et al., 2002]. Se mide la proporción de soluciones no dominadas de una población *P*. Para esto, se obtiene el frente de Pareto (*POF*) de dicha población y se divide la cardinalidad de *POF* entre la cardinalidad de *P* tal como se muestra en la siguiente formula:

$$RNI = \frac{|POF|}{|P|}. \quad (3.1)$$

Hipervolumen (HV) [Zitzler y Thiele, 1999]. Se define como la sección dentro del espacio objetivo dominado por las soluciones pertenecientes a un frente de Pareto obtenido por el MOEA, denominado *POF*.

$$HV \triangleq \left\{ \bigcup_i a_i \mid v_i \in POF \right\}. \quad (3.2)$$

Donde:

- v_i es el vector de variables de decisión no dominado perteneciente a POF .
- a_i es el área determinada entre v_i y un punto de referencia.

Proporción de hipervolumen (HVR) [Van Veldhuizen, 1999]. En caso de conocer el frente de Pareto óptimo (POF^*) se puede medir la proporción entre éste y un frente obtenido (POF), esto se hace obteniendo el hipervolumen de ambos en base a un mismo punto de referencia y aplicando la siguiente formula:

$$HVR \triangleq \frac{HV(POF)}{HV(POF^*)}. \quad (3.3)$$

Distancia generacional del espacio de variables (VD) [Goh y Tan, 2009]. Una variación de la distancia generacional que puede ser aplicada en problemas estáticos y dinámicos. Se calcula la distancia entre cada uno los vectores de variables de decisión v pertenecientes al frente de Pareto obtenido POF y el vector de variables de decisión perteneciente al frente de Pareto óptimo POF^* .

$$VD = \frac{\sqrt{|POF| \sum_{v \in POF} d(v, POF^*)^2}}{|POF|}. \quad (3.4)$$

Distancia generacional invertida (IGD) [Sierra y Coello, 2005]. Se propuso una variación de la distancia generacional (GD por sus siglas en inglés), donde se mide la mínima distancia euclidiana entre cada uno de los vectores pertenecientes al frente de Pareto óptimo con respecto a los vectores del frente obtenido.

$$IGD(POF, POF^*) = \frac{\sum_{v \in POF^*} d(v, POF)}{|POF^*|}. \quad (3.5)$$

Propagación máxima (MS) [Goh y Tan, 2009]. Mide la proximidad de un frente de Pareto obtenido al frente óptimo en los M objetivos, en otras palabras, define que tan bien cubierto esta POF^* por POF .

$$MS = \sqrt{\frac{1}{M} \sum_{i=1}^M \left[\frac{\min[\overline{POF}_i, \overline{POF^*}_i] - \max[\underline{POF}_i, \underline{POF^*}_i]}{\overline{POF^*}_i - \underline{POF^*}_i} \right]^2}, \quad (3.6)$$

donde:

- \overline{POF}_i y \underline{POF}_i son los valores máximo y mínimo del i -ésimo objetivo en POF .
- $\overline{POF^*}_i$ y $\underline{POF^*}_i$ son los valores máximo y mínimo del i -ésimo objetivo en POF^* .

3.5.2 Métricas para evaluar algoritmos que solucionan MaOPs

Cuando se trabaja con MaOPs, tanto estáticos como dinámicos, conocer el frente de Pareto óptimo puede hacerse más complicado conforme aumenta el número de objetivos. Esto a su vez puede hacer que la confiabilidad de las métricas de desempeño que se basan en la comparación con el POF^* se vea afectada.

Por este motivo, varios autores han propuesto métricas de desempeño que no utilizan un POF^* para evaluarse sino la misma población o un punto de referencia. Estas medidas han mostrado ser formas de medición efectivas y comúnmente utilizadas en diversos trabajos dentro del estado del arte. A continuación, se menciona tres de las métricas que no utilizan POF^* más relevantes en el estado del arte y usadas en futuras secciones. La selección se basó considerando si su aplicación era posible en las condiciones establecidas por el problema, además de su relevancia dentro de diversos trabajos dentro del estado del arte.

L-dominancia [Zou et al., 2008]: Es una alternativa propuesta para manejar de forma más flexible el concepto de dominancia en un MaOP. Determina que una solución es superior a otra si el número de objetivos en la que es mejor supera el número de objetivos donde esta solución es peor. Estas condiciones se definen formalmente en las ecuaciones (3.7) y (3.8).

$$B_t(x, y) > W_s(x, y), \quad (3.7)$$

$$\|F(x)\|_p < \|F(y)\|_p, \quad \|F(x)\|_p = \left(\sum_{i=1}^m (f_i(x) - z_i)^p \right)^{1/p}, \quad (3.8)$$

donde:

- $B_t(x,y)$ es el número de objetivos en la que x es mejor que y .
- $W_s(x,y)$ es el número de objetivos en la que x es peor que y .
- z_i es el valor del i -ésimo objetivo de un punto de referencia z .
- p es un coeficiente de normalización definido por el usuario.
- m es el número de objetivos.

Indicador R2 [Hansen y Jaszkiwicz, 1994]: Este indicador permite evaluar la calidad relativa de dos conjuntos de aproximación. Como métrica de desempeño, estos conjuntos pueden considerarse como el frente de Pareto obtenido y un conjunto de puntos de referencia. R2 permite evaluar los aspectos deseados de un frente de Pareto (convergencia y diversidad). La ecuación (3.9) define formalmente como se evalúa R2. Mientras más pequeño sea el valor, mayor es la calidad.

$$R2 = \frac{1}{|U|} \sum_{u \in U} \min_{a \in A_t} \{u(a)\}, \quad (3.9)$$

donde:

- U es el vector de funciones de utilidad.
- A es un conjunto de soluciones obtenidas.

Función de escalarización de logros (ASF) [Wierzbicki, 1979]: Función de agregación que permite identificar que tan bien se ajustan los valores de las funciones objetivo de una solución en base a un punto de referencia. De igual manera que R2, se apoya de un conjunto de vectores de peso normalizados Λ para evaluar dicho ajuste bajo tantas condiciones dentro del problema como sea posible. Los valores de los vectores en Λ son determinados de forma que se evalúen todas las condiciones de preferencia posibles, desde preferencias totalmente enfocadas hacia un solo objetivo, a un vector de pesos con valores distribuidos de manera uniforme para cada objetivo. La ecuación (3.10) muestra la ASF de manera formal.

Esta función toma principal consideración en dos factores. Primero, la distancia de la solución a punto de referencia, penalizando el valor objetivo que este más alejado a la referencia. Segundo, los diferentes vectores de pesos, cuyos valores están distribuidos uniformemente, que se manejan para evaluar la solución bajo un conjunto de múltiples condiciones. El valor más pequeño de ASF equivale a una mejor adaptación del frente de Pareto obtenido respecto al punto de referencia definido.

$$ASF(q, f(x), \lambda) = \max_{i=1, \dots, m} \{ \lambda_i (f_i(x) - z_i) \} + q \sum_{i=1}^m \lambda_i (f_i(x) - z_i), \quad (3.10)$$

donde:

- λ es un vector de pesos de Λ de tamaño m .
- z es un punto de referencia en el espacio búsqueda de funciones objetivo.
- q es el coeficiente de aumento para ASF.

3.5.3 Adaptación de las métricas de desempeño a problemas dinámicos

Al evaluar algoritmos dinámicos también es deseable que se evalúe la capacidad de dicho algoritmo para adaptarse a los cambios del entorno. A diferencia de las métricas de desempeño estáticas, no solamente es importante el resultado obtenido después de haber finalizado la ejecución del algoritmo en el problema. También es sumamente importante conocer la información obtenida durante el proceso de solución en cada uno de los periodos en los que el problema se mantuvo estático.

Lo anterior se debe a que no solo se busca conocer la calidad del DMOEA aplicado en ese DMOP al final de su ejecución, sino que también es deseable conocer la capacidad de dicho algoritmo para adaptarse a los cambios dentro del entorno del problema resuelto. Por dicha razón, la aplicación de las métricas de desempeño en DMOPs puede dividirse en dos técnicas, en línea y fuera de línea:

En línea. Se obtiene un conjunto de valores, donde cada uno de ellos corresponde a la aplicación de una métrica de desempeño en un periodo específico durante el proceso de solución. Usualmente, dichos periodos corresponden a los momentos en los cuales el

problema se mantiene estático, en otras palabras, cada cambio en el entorno significa un nuevo periodo estático.

Fuera de línea. El valor resultante se obtiene hasta el final del proceso de solución en base de los frentes de Pareto obtenidos en cada periodo estático. En el trabajo de Goh y Tan [Goh y Tan, 2009] se utilizaron para su trabajo IGD y MS fuera de línea, cuyas formulas se presentaron anteriormente dentro de la sección 3.3.1. Los autores propusieron una forma de obtener un valor único para cada métrica evaluando los resultados al final de cada periodo estático y calculando el promedio de dichos valores en base a la siguiente ecuación:

$$\theta_{offline} = \frac{1}{\tau} \sum_{t=1}^{\tau} \theta \cdot I(t), \quad (3.11)$$

$$I(t) = \begin{cases} 1, & \text{si } (t \% \tau = 0) \\ 0, & \text{si } (t \% \tau \neq 0) \end{cases}, \quad (3.12)$$

donde:

- θ es la métrica de desempeño utilizada.
- $\theta_{offline}$ es el resultado final de la métrica de desempeño al terminar el proceso.
- τ es el número total de generaciones utilizadas por el DMOEA.

Por otra parte, hay múltiples trabajos se propuso el uso de la ecuación (3.13) para calcular el desempeño de un DMOEA aplicado para resolver un problema de optimización dinámico [Wang y Li, 2010; Azzouz et al., 2014]. Sea la variable θ como una métrica de desempeño:

$$\bar{\theta} = \frac{1}{nc} \sum_{i=1}^{nc} \theta_i, \quad (3.13)$$

donde:

- θ_i es el valor obtenido por θ durante el periodo i , antes del cambio al periodo $(i+1)$.
- nc es el número de cambios dentro del problema.

Para ambos casos se considera la unificación de los valores de la métrica para cada periodo mediante la obtención de un valor promedio. Para esta tesis, se considera el cálculo de los valores fuera de línea en base a la ecuación (3.13).

A partir de lo revisado anteriormente se puede generar una relación entre los algoritmos, instancias de prueba y métricas de desempeño revisadas en este trabajo. Dicha relación se presenta a continuación en la Tabla 3.2.

En base a dicha tabla, existe una falta de estandarización en los problemas de referencia, ya que, aunque las instancias FDA son comúnmente utilizados, no son las únicas aplicadas y en algunos casos incluso se utilizan únicamente instancias definidas por el mismo creador del DMOEA. Además de que el conjunto de métricas utilizadas para evaluar el desempeño de cada uno de los algoritmos diseñados es diferente en cada uno.

Tabla 3.2. Relación de DMOEAs, instancias de prueba y métricas revisadas.

Enfoque	Algoritmo	Instancias de prueba	Métricas de desempeño usadas
ED	DNSGA-II	FDA	HVR
EPC	D-QMOO	FDA	EO-EVD
EPC	CDDMEA	FDA y G2	U-measure [Leung y Wang, 2003]
EM	d-COEA	FDA y dMOP	VD y MS
EM	MS-MOEA	FDA, DMZDT y WYL	IGD y HV
EMOP	DMEA	FDA, G2	Ninguna

3.6 Análisis de superficie de aptitud

Es importante conocer de qué manera afectan las funciones objetivo el comportamiento de los algoritmos de búsqueda mediante un análisis de la superficie de aptitud. Esto debido a que obtener una superficie relativamente lisa significa que una solución de buena calidad en una región específica dentro de ella se encuentra rodeada de otras soluciones de buena calidad en regiones vecinas, por lo que la información obtenida al encontrar dicha solución puede ser aplicada para las vecindades.

Por otro lado, una superficie rugosa o inestable significa que la información obtenida para encontrar una solución de buena calidad dentro de una vecindad no puede necesariamente ser útil para encontrar soluciones de buena calidad en otras regiones dentro de la superficie [Garrett y Dasgupta, 2007].

Como se vio en la sección 2.5 existen técnicas para transformar un MOP en un SOP en base de un conjunto de pesos. Garrett y Dasgupta afirman que a partir de esto se puede caracterizar un MOP como “liso” si un ligero cambio en los valores de los pesos para cada función objetivo lleva a un ligero cambio en la superficie de aptitud para el SOP generado. Por otro lado, si dicho cambio lleva a una modificación drástica dentro de la superficie se considera a ese MOP como “rugoso”.

3.6.1 Análisis de superficie de aptitud dinámica

Debido a que los DMOPs presentan cambios en su entorno, es necesario que el análisis de superficie de aptitud realizado se adapte a dichos cambios. Diversos trabajos proponen diferentes métodos de análisis de superficie de aptitud (FLA por sus siglas en inglés) para evaluar tanto la topología como las propiedades dinámicas de una superficie de aptitud dinámica [Richter, 2013; Garrett y Dasgupta, 2007]:

Modalidad [Richter, 2013]. En esta medida, se hace un conteo del número de soluciones óptimas que existen en la superficie. Esta medida puede ser calculada fácilmente en superficies de aptitud para las cuales esté disponible una descripción matemática en forma de ecuación. Además, la modalidad puede ser considerada un factor influyente en la dificultad del problema. Se define un punto en el espacio de búsqueda x como óptimo local para un tiempo t si los demás puntos dentro de su vecindad no lo superan en el valor de aptitud obtenido.

$$f(x, t) \geq f(n(x), t). \quad (3.14)$$

Correlación de la distancia de aptitud [Garrett y Dasguspta, 2007]. Esta medida se utiliza para determinar el nivel de relación que existe entre los valores de aptitud de una solución y su distancia a la solución óptima más cercana. Primero es necesario llevar a cabo una caminata aleatoria de T pasos con una longitud l_s y una variable aleatoria *rand*, tal como muestran las ecuaciones 3.15 y 3.16. A partir de los puntos obtenidos se obtiene la distancia mínima $d_j(k) = \|x(j) - x_s(t)\|$ de dicho punto a un óptimo global $x_s(t)$.

Los autores de esta propuesta sugieren en su trabajo el cálculo de la distancia euclidiana entre los valores de aptitud del punto obtenido y los óptimos globales (véase ecuaciones 3.17 y 3.18). Esto con el objetivo de generar un solo valor correspondiente a la distancia para cada comparación y de esta manera poder seleccionar un valor mínimo para aplicar en el cálculo de la correlación.

$$x(j+1) \geq x(j) + l_s * rand, \quad (3.15)$$

$$f(j, t) \geq f(x(j), t), \quad j = 1, \dots, T, \quad (3.16)$$

$$\rho(t) = \frac{1}{\sigma_f \sigma_d T} \sum_{j=1}^T (f(j, t) - \bar{f}(t)) (d_j(t) - \bar{d}(t)), \quad (3.17)$$

$$\bar{d}(t) = \frac{1}{T} \sum_{j=1}^T d_j(t). \quad (3.18)$$

Donde σ_f y σ_d son las desviaciones estándar de $f(j, t)$ y $d(t)$ respectivamente. Un problema se considera fácil si una mayor cercanía de una solución al óptimo está fuertemente relacionada con el aumento en la aptitud de dicha solución. Lo anterior se representa en la medida mediante la obtención de un valor $\rho(t) = -1$. Por otro lado, un valor $\rho(t) = 1$ significa que existe poca relación entre ambos factores y por lo tanto el problema se considera difícil. De igual manera que las medidas anteriores, se considera un promedio en base de los periodos estáticos para la obtención del valor ρ .

Frecuencia de cambio. Esta medida se utiliza para considerar la frecuencia en la cual ocurren cambios que afectan al problema y que pueden modificar la superficie de aptitud. Para este caso se considera una frecuencia de cambio γ para cada periodo t , a partir de esto se puede determinar una generación τ en la cual se presentará un cambio.

$$\tau = \gamma t. \quad (3.19)$$

Severidad espacial. Otro factor que considerar en un problema dinámico es el grado de afectación que dichos cambios llevan a la superficie de aptitud. Un cambio puede llevar a la relocalización de las soluciones óptimas dentro del entorno. La medición de la severidad espacial en un periodo $n(t)$ puede obtenerse mediante el cálculo de la diferencia entre la posición de las soluciones óptimas $x_s(t)$ en dos periodos estáticos t anexos. Un valor único para esta medida puede definirse mediante el cálculo de un promedio de los valores obtenidos durante los T periodos estáticos que ocurren para ese problema.

$$\eta(t + 1) = ||x_s(t + 1) - x_s(t)||, \quad (3.20)$$

$$\eta = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \eta(t). \quad (3.21)$$

3.6.2 Evolucionabilidad poblacional

Es posible llevar a cabo la aplicación de medidas estáticas para el análisis de la superficie de aptitud en un DMOP si son utilizadas de manera correcta, siendo capaz de extender su uso si hace una adecuación correcta y se trabaja bajo un enfoque que pueda adaptar estas medidas a un entorno dinámico. En el caso particular de la evolucionabilidad, es posible aplicarla para evaluar problemas dinámicos si se trabaja bajo un enfoque que divide a este en múltiples problemas estáticos.

Como se mencionó en la sección 2.5.2, la mayoría de los métodos FLA no consideran el algoritmo utilizado para solucionar el problema en cuestión. Por ejemplo, si se hace consideración de los métodos FLA presentados en la sección 3.6.1, todos se enfatizan en el problema a resolver. Sin embargo, ninguno de ellos toma en consideración el método o algoritmo utilizado para resolver dicho problema. Métodos como la evolucionabilidad permiten identificar propiedades del problema a resolver y la capacidad del método utilizado para solucionarlo bajo el entorno actual.

Una variante de la evolucionabilidad es la evolucionabilidad poblacional [Wang et al., 2017], la cual se puede definir como la capacidad de una población de generar variantes con mayor aptitud que la actual en base de dos factores: i) la probabilidad de generar una población hija con mayor aptitud y ii) el grado de mejora de las poblaciones hijas. Este concepto se define formalmente:

$$evp(P_i) = \begin{cases} \frac{\sum_{P_{ij} \in N^+(P_i)} \frac{|f^b(P_i) - f^b(P_{ij})|}{NP}}{\frac{\sigma(f(P_i))}{|N(P_i)|}}, & |N^+(P_i)| > 0, \\ 0, & |N^+(P_i)| = 0 \end{cases}, \quad (3.22)$$

donde P_i es la población actual, NP es el tamaño de P_i , $N(P_i)$ es un conjunto de poblaciones hijas generadas por P_i , $N^+(P_i)$ es un subconjunto de $N(P_i)$ tal que para problemas de minimización $N^+(P_i) = \{P_{ij} | P_{ij} \in N(P_i), f^b(P_{ij}) < f^b(P_i)\}$ en donde $j = 1, \dots, |N^+(P_i)|$, $f^b(P)$ es el mejor valor de aptitud de la población P .

3.7 Hiperheurísticas

Dentro de la literatura se han propuesto múltiples hiperheurísticas para solucionar diversos problemas de optimización tanto estáticos como dinámicos. También, se han propuesto diversos métodos de selección de LLHs y criterios de aceptación, los cuales se han aplicado en diversos tipos de problemas con características similares a los revisados en este capítulo.

A continuación, se hará descripción de algunos de los métodos de selección, criterios de aceptación e hiperheurísticas más relevantes dentro del campo y con un mayor grado de importancia para esta tesis.

3.7.1 Métodos de selección de heurísticas de bajo nivel

En la sección 2.6.3 se hizo mención sobre los métodos de selección de heurísticas de bajo nivel, los cuales son métodos utilizados para elegir el LLH más adecuado para emplear en la solución de un problema considerando su estado actual. Estos métodos pueden ser adaptativos o no adaptativos. Algunos de los métodos de selección de LLH adaptativos son:

Función de elección [Cowling et al., 2000]. La selección se hace utilizando una función que considera varios factores. Primero la calidad de los resultados obtenidos por cada LLH de forma individual. Segundo, la comparación del desempeño de cada LLH respecto a cada una de las demás heurísticas candidatas, en otras palabras, una comparativa de parejas. Finalmente, se toma en cuenta el tiempo transcurrido desde su última selección.

Descenso aleatorio [Cowling et al., 2000]. Se elige un LLH (o permutación de LLHs) y se aplica hasta que no produzca mejora. Cuando se generan resultados de menor calidad que los actuales se lleva a cabo un proceso para seleccionar un nuevo LLH el cual depende de la hiperheurística que utiliza este método.

Búsqueda Tabú [Dowland et al., 2007]. Cuando un LLH genera soluciones de menor calidad se inserta en una lista tabú y no puede ser seleccionado mientras este allí.

Ruleta con memoria. Selección aleatoria en la que cada LLH es asignado un cierto porcentaje en base a la calidad de sus resultados actuales y los obtenidos en periodos previos.

Por otra parte, los métodos no adaptativos únicamente consideran la información obtenida durante el proceso de selección y no de periodos o generaciones previas. Entre los métodos no adaptativos más relevantes para esta tesis se encuentran los siguientes:

Aleatorio [Cowling et al., 2000]. Selección aleatoria donde cada LLH es otorgado el mismo porcentaje de selección. Este método no considera ningún valor o métrica, la selección es totalmente determinada por una elección aleatoria.

Voraz [Cowling et al., 2000]. Se elige el LLH que en ese momento obtenga el mejor resultado bajo una cierta métrica, función de aptitud, valores objetivo o uno o varios indicadores de desempeño.

Hambriento [Cowling y Chakhlevitch, 2003]. Variación del voraz en el cual se revisa únicamente un subconjunto de los LLH. Puede combinarse con una búsqueda tabú para separar ciertos LLHs del conjunto de heurística seleccionables.

Ruleta. Selección aleatoria en la que cada LLH es asignado un cierto porcentaje basado únicamente en resultados actuales. Muchos de sus aspectos son similares a la selección aleatoria. Sin embargo, la asignación de porcentajes se apoya en algún criterio, tal como los valores objetivo de las soluciones o los resultados obtenidos bajo indicadores de desempeño.

3.7.2 Criterios de aceptación de soluciones

Como se mencionó en la Sección 2.6.3, los criterios de aceptación son técnicas que permite definir si la solución obtenida por el LLH seleccionado se acepta (o rechaza) para reemplazar la solución actual. Los criterios pueden ser deterministas o no deterministas. Algunos ejemplos de criterios deterministas más relevantes son los siguientes:

Todos los movimientos (All Moves) [Cowling et al., 2000]. Siempre se aceptan las nuevas soluciones obtenidas, sin importar si la calidad de dichas soluciones es mejor o no.

Solo mejor [Cowling et al., 2000]. Solo se aceptan soluciones con mejor resultado en los valores objetivo o valores que determinan la aptitud de la solución. Esta consideración también se puede basar en otros indicadores tales como métricas de desempeño, métodos FLA o las preferencias del DM.

Mejor o igual [Cowling et al., 2000]. Sigue la misma estrategia que el criterio de aceptación “solo mejor”, pero en este caso también se aceptan las nuevas soluciones si su calidad es idéntica a las actuales. En otras palabras, la única condición de rechazo en este criterio es si las nuevas soluciones son de peor calidad bajo el evaluador usado por la hiperheurística que contiene este criterio.

En los criterios no deterministas, la aceptación o rechazo de las soluciones no siempre es idéntica aun cuando se presenten las mismas condiciones. Dichos criterios usualmente se apoyan de una técnica estocástica para llevar a cabo su decisión. Algunos de los criterios no deterministas dentro de la literatura más relevantes para esta tesis se encuentran:

Recocido simulado [Bai, 2005]. Acepta soluciones de mejor calidad y aplica el criterio de Metrópolis para aceptar o no soluciones de menor calidad. El recocido simulado utiliza una variable aleatoria dentro de su proceso, por tal motivo los resultados pueden variar aún con las mismas soluciones bajo el mismo entorno.

Búsqueda tabú [Chakhlevitch y Cowling, 2005]. Los LLH con soluciones de menor calidad se rechazan e insertan a lista tabú. En este caso, la separación de algunos LLHs de las opciones candidatas significa que el resultado puede variar aun teniendo las mismas soluciones disponibles.

3.7.3 Hiperheurísticas para solucionar MOPs

Dentro de la literatura se han propuesto múltiples hiperheurísticas para solucionar diversos problemas de optimización, a continuación, se hará descripción de algunas de las propuestas presentadas en la literatura para resolver MOPs, los cuales no presentan cambios en su entorno durante todo el transcurso del tiempo.

Hiperheurística basada en un algoritmo genético para resolver el problema de carteras públicas (HHGA_SPP) [García, 2010]: Hiperheurística enfocada en resolver un problema de selección de carteras de proyectos públicos apoyándose de un algoritmo genético. Cada individuo está compuesto por un subconjunto de LLHs en una cierta combinación, la cual define el orden con el que dichas heurísticas se aplican a una solución candidata.

Hiperheurística basada en una función de elección (HH_CF) [Maashi et al., 2014]. En cada intervalo de selección, a cada LLH r se aplicó una función de elección basada en dos elementos: un ranking de dos etapas (c_1) y una variable de tiempo (c_2), la cual se define como

$$CF(r) = \alpha c_1(r) + c_2(r). \quad (3.23)$$

La variable α es definida manualmente en base a la relevancia que se le desee dar al valor obtenido por el ranking sobre el tiempo transcurrido desde la última selección de un LLH. Para obtener c_1 , la primera etapa hace múltiples rankings donde se posiciona la calidad de las soluciones obtenidas por cada LLH en base a diversas métricas de desempeño. La segunda etapa hace un ranking en base al número de veces que cada LLH tuvo la primera posición en cada uno de los rankings generados en la primera etapa. Después de esto, c_1 se calcula utilizando la siguiente fórmula

$$c_1(r) = 2 * (N + 1) - \{Freq_{pos}(r) + \theta_{pos}(r)\}, \quad (3.24)$$

donde N es el número de LLHs, $Freq_{pos}$ es la posición de cada LLH en el ranking de la segunda etapa y θ_{pos} es el ranking correspondiente a la métrica de desempeño que tenga mayor relevancia θ . El valor de c_2 para cada LLH se obtiene en base al tiempo transcurrido desde su última aplicación en el problema, este valor puede definirse por unidades de tiempo (milisegundos, segundos, etc.) o por el número de evaluaciones o generaciones transcurridas.

Aquel LLH cuyo valor de función de elección sea mayor es seleccionado para aplicarse hasta el próximo proceso de selección. Esta hiperheurística puede ser aplicada para llevar a cabo la selección de LLHs tan complejos como metaheurísticas, ya que los autores propusieron originalmente su uso en un grupo de LLHs que consistía en tres MOEAs.

Hiperheurística con mecanismo de selección basado en un sistema de inferencia difuso [Zamli et al., 2017] (FIS): Esta hiperheurística utiliza cuatro diferentes metaheurísticas como LLHs (algoritmo genético básico, enseñanza/aprendizaje, polinización y Jaya [Rao, 2016]) para resolver un problema, proponiendo un nuevo mecanismo de aceptación de las nuevas soluciones el cual se basa en un sistema de inferencia difuso.

3.7.4 Hiperheurísticas para solucionar problemas dinámicos

Dentro de la literatura existen varios trabajos que se enfocan en solucionar problemas de optimización utilizando metaheurísticas como LLH. Sin embargo, pocos de ellos se enfocan en solucionar problemas de optimización dinámicos. Esto se debe a que la introducción de dinamismo en un problema significa que se deben considerar un número de condiciones adicionales, como se ha mencionado en capítulos anteriores.

Para el caso de las hiperheurísticas esto significa que la selección de LLHs y aceptación de soluciones deben tomar en cuenta que las soluciones y entorno actual del problema no son necesariamente idénticos al estado en que se encontraban la última vez que se realizaron estos procesos. Entre los trabajos de la literatura con mayor relevancia se encuentran los siguientes:

Enfoque por hiperheurística [Özcan et al., 2009]. Es una de las investigaciones iniciales dentro del área. Se utiliza una hiperheurística con un método de selección voraz para resolver

problemas de optimización dinámicos de un solo objetivo bajo diferentes niveles de severidad de cambio y frecuencia de cambio. Se usaron seis LLH diferentes, la mutación de cambio de bits con cinco niveles de mutación diferentes y el método de escalada de bits de Davis [Davis, 1991]. Los resultados experimentales confirmaron que los diferentes cambios del entorno del problema con respecto a la frecuencia o la gravedad requieren diferentes enfoques de adaptación al cambio y que las hiperheurísticas son capaces de adaptarse a estos cambios, ya que pueden elegir qué heurística es apropiada para usar de acuerdo con las condiciones actuales del problema.

Estudio de hiperheurísticas [Kiraz et al., 2013]. Los autores de esta investigación experimentaron con diversas configuraciones de hiperheurísticas para resolver los problemas de referencia de picos móviles (MPB). Los métodos de selección de LLH usados fueron aleatorio, voraz, función de elección, aprendizaje reforzado [Kaelbling et al, 1996], descenso aleatorio y un nuevo método que usa una búsqueda de un solo punto basada en hipermutación. Los criterios de aceptación utilizados fueron “All Moves”, “mejor o igual”, “solo mejor” y estrategias basadas en recocido simulado y el método de Monte Carlo [Ayob y Kendall, 2003]. Todo lo anterior generó un total de 35 hiperheurísticas probadas en entornos dinámicos con diferentes severidad y frecuencia de cambio de picos.

Hibridación de hiperheurísticas con EAs [Topcuoglu et al., 2014]. El enfoque de este estudio fue respecto la hibridación de algoritmos evolutivos e hiperheurísticas. Se utilizó un manejo de memoria externa, insertando el mejor individuo de cada generación a un archivo. Se probaron 15 diferentes hiperheurísticas, las cuales consistían en diferentes combinaciones de métodos de selección de LLH y criterios de aceptación. Dichas hiperheurísticas se emplearon para solucionar problemas MPB y problemas dinámicos de asignación generalizado [Martello y Toth, 1990]. El método de adaptación al cambio consistía en unificar la población con el archivo, reevaluar el conjunto bajo el nuevo entorno y elegir las mejores soluciones para la nueva población usada en las siguientes generaciones.

Hibridación de hiperheurísticas con modelos probabilísticos [Uludag et al., 2013]. Se combina las capacidades de una hiperheurística con un algoritmo de aprendizaje incremental basado en la población [Yang y Yao, 2008]. La población es dividida en dos conjuntos, uno asignado a la hiperheurística y el otro al modelo probabilístico. La probabilidad de selección de los LLHs se determina mediante un proceso de aprendizaje fuera de línea. Los resultados de ambas poblaciones son unificados al final de cada generación y las mejores soluciones se utilizan para actualizar las probabilidades de cada LLH y mutar la subpoblación correspondiente al método probabilístico. Se probaron múltiples varios métodos de selección de LLH basados en estrategias aleatorias, aprendizaje reforzado y colonia de hormigas. Las líneas de estudio enfocadas en hibridación muestran la flexibilidad y adaptabilidad de una hiperheurística y cómo se puede utilizar como enfoque individual o hibridar con otras estrategias para resolver problemas de manera cooperativa.

Hiperheurísticas para solucionar problemas de enrutamiento de vehículos dinámicos. El problema de enrutamiento de vehículos dinámico (DVRP) [Kilby et al., 1998] es una variación del conocido VRP que incorpora elementos dinámicos al agregar nueva

información relacionada con el problema en cualquier momento. EH-DVRP [Garrido y Riff, 2010] es una hiperheurística propuesta que buscaba solucionar DVRPs mediante la generación y desarrollo de combinaciones de LLH para construir y mejorar rutas parciales, las cuales se unificaban al final. EH-DVRP utilizó aprendizaje en línea y un conjunto de heurísticas de construcción de rutas como LLHs. En un trabajo relacionado se propuso solucionar DVRP mediante una hiperheurística que utiliza un marco de referencia de expresión de genes (GEP-HH) [Sabar et al., 2015a]. GEP-HH genera y evoluciona un conjunto de individuos y los decodifica para generar un conjunto de LLHs de perturbación y criterios de aceptación. El método de selección de LLH de GEP-HH era voraz, utilizando el LLH con mejor aptitud para cada periodo.

Hiperheurísticas para planificación de programas dinámicos [Gökçe, et al., 2017]. En ese trabajo se propuso una hiperheurística de selección basada en un algoritmo genético para resolver problemas de planificación de programas. Se le asigna a cada máquina un cierto LLH para aplicar en un periodo determinado. Los LLHs que generen los mejores resultados se consideran como padres para generar nuevas permutaciones de LLHs. Esta hiperheurística usa un criterio de aceptación “All Moves”. Sin embargo, el criterio de parada está determinado por un cierto número de generaciones transcurridas sin mejora.

Hiperheurística basada en el retroceso exponencial binario [Chen et al., 2017]. Esta hiperheurística propuso la aplicación de búsqueda tabú como método de selección de LLHs dentro de un proceso de retroceso exponencial binario (BEBO) [Remde et al., 2009]. Esta propuesta se implementó para resolver problemas de reparación de caminos, los cuales obtenían constantemente llamadas de emergencia y reportes de daños. El objetivo era minimizar la cantidad de tiempo requerido para reparar todos los inconvenientes existentes. Los LLHs utilizados eran específicos al problema y solo se aceptaban las soluciones que fueran mejores que las actuales.

Los casos anteriores muestran hiperheurísticas aplicadas en problema de optimización dinámicos las cuales utilizan LLHs basados en heurísticas simples, tales como un intercambio de bits o un de una pareja de valores en un vector de variables de decisión. Sin embargo, la complejidad de un LLH no está limitada por ningún factor. De hecho, puede ser tan o más compleja como la misma hiperheurística en la que se utiliza. Pocos trabajos han propuesto el uso de metaheurísticas como LLHs para solucionar problemas dinámicos:

Escalada adaptativa (AHC) [Wang et al., 2009]. Es una estrategia de búsqueda local utilizada como un componente de algoritmo memético basado en algoritmo genético dinámico denominado AHMA. AHC utiliza dos métodos de escalada: una escalada voraz basada en cruza (GCHC) y una escalada empinada basada en mutaciones (SMHC). GCHC toma la solución con la mejor aptitud y un individuo seleccionado de la población seleccionado por ruleta como padres para generar una descendencia usando cruza uniforme, reemplazando al individuo élite si tiene una mejor aptitud. SMHC muta varios bits aleatorios del individuo élite. El individuo mutado reemplaza al individuo élite si su aptitud es mejor. Ambos métodos inician con una probabilidad idéntica para aplicarse a la población. Cuando todos los pasos de la búsqueda local terminaron para la generación actual, AHMA define

nuevas probabilidades para cada método en base a la mejora que hayan producido en la población en esa generación. Cuando se detecta un cambio AHC se usa justo antes de activar el método de adaptación al cambio de AHMA.

Búsqueda basada en hiperheurística (HH) [Baykasoğlu y Ozsoydan, 2017]. Se utilizan tres metaheurísticas basadas en población dentro de una hiperheurística para solucionar problemas de mochila dinámicos de múltiples dimensiones. El método de selección de heurística se basa en una ruleta donde en cada iteración se le asigna a cada uno de los LLH candidatos un porcentaje de selección en base a la calidad de sus resultados obtenidos en periodos previos. Se utiliza AHC y un método basado en inmigrantes aleatorios como métodos de adaptación al cambio. Esta hiperheurística usa un criterio de aceptación *All Moves*, ya que acepta cualquier nueva solución obtenida dentro de la población sin importar la calidad que tenga. El método de detección de cambios utilizado en esta hiperheurística es un umbral de diversidad, el cual determina que si las soluciones son demasiado homogéneas debe hacerse un cambio. La adaptación al cambio se hace mediante un método de inmigrantes aleatorios desencadenados [Wang et al., 2009] y cambios en el tamaño de la representación de la solución.

Versión binaria del algoritmo de atracción de superposición ponderada (bWSA) [Baykasoğlu y Ozsoydan, 2018]. De igual manera que el AHMA, bWSA es un algoritmo genético. Sin embargo, dentro de su proceso utiliza la hiperheurística AHC, llevando a cabo el mismo proceso de selección para determinar cuál método de escalada es el más adecuado para utilizar en base al estado actual del entorno del problema. La diferencia principal de bWSA respecto a AHMA es el momento en el que se utiliza AHC cuando se ha detectado un cambio. En este caso bWSA usa AHC hasta después de que hayan terminado de aplicarse los métodos de detección y adaptación al cambio.

Meta-hiper-heurística heterogénea (HMHH) [Stockt y Engelbrecht, 2018]. Basada en la propuesta original de HMHH [Grobler et al., 2010] y adaptado a problemas de optimización dinámicos. Esta línea de trabajo, iniciada desde el 2014, es hasta nuestro conocimiento la primera instancia en la cual se utiliza una hiperheurística como estrategia principal para resolver problemas de optimización dinámicos. La versión más reciente de este trabajo, propuesta en el 2018, utiliza HMHH con cinco diferentes metaheurísticas basadas en algoritmos genéticos, de evolución diferencial, así como de enjambre de partículas como LLHs para resolver DOPs. HMHH utiliza diez métodos de selección de LLH, tanto adaptativos como no adaptativos, basados en métodos como permutaciones, aleatoriedad o colonia de hormigas basadas en concentración de hormonas y mejora de aptitud para llevar a cabo la selección. También se propuso un método de selección de LLH el cual asigna probabilidades a cada metaheurística en base a la frecuencia que cada una mejoraba la aptitud de una solución respecto a un cierto número de iteraciones previas. Los resultados de esta línea de trabajo muestran la factibilidad de utilizar hiperheurísticas con metaheurísticas como LLHs para resolver problemas de optimización dinámicos de un solo objetivo.

Todos los trabajos presentados en esta subsección se enfocan en resolver únicamente problemas de optimización dinámicos de un solo objetivo. Esta situación provee una clara

área de investigación potencial. Debe considerarse que la mayoría de los problemas reales no solo cambian constantemente, sino que también tiene múltiples objetivos a satisfacer. Teniendo en cuenta los teoremas de "*No Free Lunch*", es comprensible creer que una hiperheurística podría obtener soluciones de alta calidad para DMOPs. Las hiperheurísticas propuestas anteriormente podrían adaptarse para resolver DMOP utilizando LLH centrados en la optimización multiobjetivo o métodos de agregación.

El estudio realizado generó un capítulo de libro [Macias-Escobar et al., 2020a], en el cual se presenta una investigación más profunda respecto a las diversas hiperheurísticas propuestas en la literatura para resolver problemas de optimización dinámicos. Dicho trabajo presenta una explicación a mayor detalle de los algoritmos revisados en esta sección, así como otros que también tiene un alto grado de relevancia en el campo. También, se hace una clasificación de los trabajos propuestos en base a su estrategia de retroalimentación, la naturaleza de sus LLHs [Burke et al., 2010] y el grado de complejidad de los LLHs utilizados por cada hiperheurística.

3.7.5 Hiperheurísticas con incorporación de preferencias

En la literatura existen varias propuestas de hiperheurísticas que utilizan métodos de incorporación de preferencias, entre los más relevantes para este trabajo y recientes se encuentran los siguientes:

Hyper-heuristic genetic algorithm for social portfolio problem (HHGA-SPP) [Rivera et al., 2012]. Esta hiperheurística resuelve problemas de cartera de proyectos generando una población donde cada individuo está conformado por una permutación de LLHs, los cuales se aplican en el orden establecido en la solución.

Genetic algorithm selection perturbative hyper-heuristic (GASPHH) [Raghavjee y Pillay, 2015]. Se utiliza una hiperheurística para solucionar el problema de asignación de horario en una escuela. Se utilizan permutaciones de LLHs, las cuales consisten en varias heurísticas de intercambio. El objetivo es obtener la menor cantidad de violaciones al horario posible, también se consideran las preferencias establecidas por los profesores respecto a horarios preferidos.

Hiperheurísticas de NSGA-II basados en puntos de referencia (r-NSGA-II-HH y R-NSGA-II-HH) [Jakubovski-Filho et al. 2018a; Jakubovski-Filho et al., 2018b]. Estos trabajos utilizan NSGA-II como hiperheurística, un conjunto de operadores de cruce y mutación actúan como LLHs. La selección de LLH se realiza utilizando un sistema en base de créditos y aleatoriedad. Para ambos casos, las preferencias del DM son incorporadas mediante la asignación de un punto de referencia.

Como se puede ver, y hasta nuestro conocimiento, las hiperheurísticas con incorporación de preferencias solo utilizan heurísticas específicas al problema como LLHs. Por otra parte, la mayoría de las hiperheurísticas propuestas para solucionar problemas de optimización dinámica se enfocan en problemas de un solo objetivo. El uso de hiperheurísticas para

resolver problemas de optimización dinámicos con preferencias es un área, hasta nuestro conocimiento en este momento, no explorada aún.

3.7.6 Hiperheurísticas para solucionar MaOPs

Varias de las hiperheurísticas que han sido propuestas previamente son capaces de solucionar problemas de optimización con más de tres objetivos. En teoría, el aumento de tres a cuatro objetivos a resolver puede parecer que no presenta un gran efecto en el desempeño de los algoritmos. Sin embargo, estrategias ampliamente conocidas como la generación de frentes por dominancia y *Crowding Distance* del NSGA-II sufren una pérdida de calidad bajo este cambio. Esto situación se nota aún más conforme el número de objetivos aumenta.

Por este motivo, y de forma similar a lo presentado en la sección 3.3, las hiperheurísticas propuestas para solucionar MaOPs se salen de las estrategias tradicionales de las hiperheurísticas que solucionar MOPs de hasta tres objetivos. Entre las hiperheurísticas de este rubro con mayor importancia y relevancia para esta tesis se encuentran:

Hiperheurística de optimización de enjambre de partículas multiobjetivo (H-MOPSO) [Castro y Pozo, 2014]: Esta propuesta busca resolver MaOPs mediante el uso de un algoritmo de optimización de enjambre de partículas (PSO). H-MOPSO maneja un conjunto de métodos de selección de partículas líder y de archivado de soluciones óptimas. Dichos métodos son considerados como el conjunto de LLHs, donde mediante la aplicación de una ruleta se selecciona uno de cada tipo de método para aplicarse durante una generación. Las probabilidades de cada LLH en la ruleta son actualizadas mediante el uso del indicador R2 [Hansen y Jaszkiwicz, 1994], el cual es una función de agregación que se basa en la cercanía de cada solución al punto utópico (en este caso, el origen) y un conjunto de vectores de peso. Se trataron MaOPs de hasta 20 objetivos, donde H-MOPSO probó ser mejor alternativa que las diferentes combinaciones posibles de LLHs utilizadas de manera independiente.

Hiperheurística multiobjetivo basada en secuencia (MOSSHH) [Walker y Keedwell, 2016]: MOSSHH es una hiperheurística que utiliza un conjunto de LLHs basados en ruina y recrear, mutación y selección desde archivo. Esta hiperheurística tiene un comportamiento similar a un algoritmo de estado estacionario, ya que, en cada iteración, se lleva a cabo la selección de LLH en base a una ruleta y se genera un único hijo a partir de la solución actual. Mediante la aplicación un indicador, el cual puede estar basado en dominancia de Pareto, hipervolumen o el número de funciones objetivo en las que es superior, se determina si la solución hija es agregada a un archivo y sustituye a la solución actual. MOSSHH fue probado en MaOPs de hasta seis objetivos y se encontró que la aplicación de indicadores basados en hipervolumen obtenía los mejores resultados para dichos problemas.

Hiperheurística para el problema de integración y orden de prueba (HITO) [Guizzo et al., 2017]: Esta hiperheurística utiliza un conjunto de LLHs basados en combinaciones de operadores de cruce y mutación para solucionar problemas de integración y orden de prueba de hasta cuatro objetivos. En ese trabajo, se probaron dos diferentes métodos de selección de LLH: función de elección, y selección basada en la estrategia de bandido multiarmado. Los

resultados para cuatro objetivos resultaron favorables cuando se aplica una función de elección sobre la otra estrategia de selección de LLH.

Metaheurística de muchos objetivos basada en el indicador R2 III (MOMBI-III) [Hernández-Gómez y Coello, 2017]: MOMBI-III utiliza como LLHs siete diferentes funciones de agregación. Mediante el uso de operadores genéticos se crea una población hija, la cual se unifica a la población padre. Las soluciones de la población unificada son evaluadas y clasificadas por cada LLH en cada vector de peso. Después, apoyado de un indicador R2 se define la posición más baja que haya obtenido cada solución en las comparaciones realizadas. Ese valor es utilizado para determinar cuáles soluciones van a formar parte de la población en la siguiente generación. MOMBI-III fue probado en MaOPs de hasta diez objetivos, teniendo un mejor desempeño en general que MOEA/D, NSGA-III y MOMBI-II.

Hiperheurística basada en cooperación (HH-CO) [Fritsche y Pozo, 2019]. En esta hiperheurística cada LLH era un MOEA con una población asignada. Cada LLH llevaba a cabo sus respectivos procedimientos e intercambiaba información entre ellos. En cada iteración se seleccionaba el mejor LLH en base a un criterio de recompensas basado en R2. Dicho LLH se aplicaba a la población actual. Después, cada LLH utilizaba su método de migración para generar una nueva población y comparar su mejora respecto a su población actual, aquel LLH con la mayor mejora era seleccionado para la siguiente iteración. HH-CO utilizó MOEAs con diferentes enfoques, tales como descomposición, uso de puntos de referencia, dominancia de Pareto e indicadores R2 como LLHs, buscando aprovechar las ventajas de cada uno de estos MOEAs para cubrir sus respectivas debilidades.

Se puede observar que la mayoría de estos trabajos usan LLHs de baja complejidad (heurísticas), con la excepción de HH-CO, el cual utiliza algoritmos como NSGA-II, MOMBI-II y MOEA/D. Sin embargo, HH-CO solo trabaja con MaOPs estáticos. Hasta nuestro conocimiento, en el momento que esta tesis fue escrita todavía no existen hiperheurísticas que trabajen con DMAOPs, y por consecuencia, tampoco existen hiperheurísticas que busquen solucionar DMAOPs preferenciales. Por este motivo este trabajo aborda un área sin explorar aún dentro del campo de optimización.

3.8 Problemas abiertos a la investigación y conclusiones

Aunque hoy en día existen varios algoritmos evolutivos enfocados hacia la optimización multiobjetivo dinámica, como se mencionó al final de la Sección 3.5 existe una falta de estandarización entre diversos DMOEAs propuestos (ver Sección 3.1 a 3.3), DMOPs (ver Sección 3.4) y métricas para evaluar el desempeño de dichos DMOEAs aplicados a estos problemas (ver Sección 3.5). Esta falta de estandarización se puede extender de igual manera a medidas para el análisis de superficie de aptitud en un problema dinámico propuestas en la literatura (ver Sección 3.6), ya que su aplicación ha sido escasa para este tipo de problemas, en especial para los DMOPs.

Por otro lado, dentro del trabajo que realizó la propuesta de HH_CF [Maashi et al., 2014] se menciona que la cantidad de estudios en los cuales se aplican hiperheurísticas en MOP es escaso. Dicha escasez de trabajos es aún mayor cuando los MOPs a resolver son dinámicos.

Esto sigue vigente hoy en día, ya que hasta donde se tiene conocimiento, las hiperheurísticas propuestas dentro de la literatura no se enfocan en DMOPs (ver Sección 3.7). En un estudio enfocado hacia la optimización dinámica [Helbig y Engelbrecht, 2013] se indica la existencia de dos grandes retos en la aplicación de hiperheurísticas para resolver problemas de optimización, cuya dificultad aumenta aún más cuando se tratan DMOPs:

- Incluso en un ambiente estático, el proceso de cambiar la hiperheurística utilizada para solucionar un cierto problema no es tarea fácil. Por esta razón, la dificultad aumenta todavía más dentro de un entorno en constante cambio.
- En un entorno dinámico, el número limitado de iteraciones disponibles en cada estado del problema antes de que suscite un cambio dificulta el proceso de afinación automática de los parámetros de las heurísticas utilizadas.

Además de lo anterior, también existe una falta de trabajos enfocados hacia ciertas direcciones, [Azzouz et al., 2017] hace un especial énfasis en diversos tópicos:

Enfoques dinámico paralelo. Cuando se están manejando DMOPs, se impone una restricción de tiempo debido a que el algoritmo evolutivo debe ser capaz de converger con el frente de Pareto óptimo lo más rápido que se pueda, antes de que ocurra un nuevo cambio en el ambiente. Los EA paralelos son relevantes en este contexto debido a eficiencia y su rapidez en comparación de los EA secuenciales. A pesar de lo anterior, existen pocos trabajos enfocados hacia esta dirección [Cámara et al., 2007; Zheng, 2007; Cámara et al., 2008].

Toma de decisiones automática. Cuando el DM tiene un conjunto de preferencias específicas, el EA debe ser capaz de dirigir la búsqueda hacia el ROI del frente de Pareto. Este objetivo ya ha sido estudiado en profundidad utilizando MOPs estáticos. Sin embargo, un ambiente dinámico sugiere que las preferencias son susceptibles a cambios, por lo que las técnicas de manejo de preferencias deben permitir que las preferencias sean adaptadas o automáticamente aprendidas durante el proceso de optimización. Hoy en día existen pocos trabajos que exploran este tema [Deb et al., 2007; Shen y Yao, 2015].

También es necesario tener en consideración que existen casos en donde los cambios dentro del entorno que rodea al problema son tan rápidos y constantes, que el proceso de toma de decisiones resulta imposible de realizar para un DM humano en el tiempo disponible entre cada cambio. Por dicha razón se requiere del desarrollo de un sistema que simule al DM. El cuál sea capaz de seleccionar dentro del conjunto de soluciones de alta calidad obtenidas en un periodo estático, aquella que satisfaga de mejor manera las preferencias definidas por el DM humano antes del próximo cambio en el entorno.

Por estos motivos es deseable que se realice un estudio a mayor profundidad sobre optimización dinámica. Especialmente en situaciones que involucran toma de decisión y preferencias de un DM, esta línea de investigación se aborda en esta tesis. También, se debe considerar que, con el aumento constante de propuestas, problemas e identificadores de desempeño, la dificultad de realizar un estudio satisfactorio aumenta constantemente. Por esta razón, es deseable el desarrollo de metodologías que permitan relacionar estos elementos, facilitando de esta manera la solución de problemas de optimización dinámicos.

Capítulo 4

Planteamiento del problema

Este capítulo muestra a mayor detalle cada uno de los problemas que las propuestas presentadas en esta tesis buscan resolver. En primer lugar, se presenta una descripción de los problemas multiobjetivo dinámico (DMOP) que se buscan solucionar, sus características y retos principales y se mencionan algunas estrategias propuestas dentro del estado del arte para resolverlos. La segunda parte de este capítulo contiene una descripción a detalle sobre en qué consiste un problema de muchos objetivos dinámico (DMaOP), su diferencia principal respecto a un DMOP, las estrategias dentro del estado del arte utilizadas para resolver algunas de las instancias que se buscan resolver y cuáles son los retos que este tipo de problemas presentan para los algoritmos propuestos. En ambas partes de este capítulo se menciona las condiciones adicionales presentadas al incorporarse a cada problema las preferencias de un DM, lo cual reduce y restringe las soluciones deseadas a una región de interés.

4.1 Problema multiobjetivo dinámico

La definición formal del DMOP se muestra en la ecuación (2.9) que se encuentra en la Sección 2.4.1. La estructura de este problema es bastante similar a la que presenta un problema de optimización multiobjetivo estático. La diferencia de mayor relevancia entre las versiones estáticas y dinámicas de un MOP es la existencia de una variable de tiempo t , la cual determina la existencia de múltiples periodos, teniendo diferentes características y restricciones cada uno. Esta condición agrega un nuevo reto para los algoritmos evolutivos que buscan solucionar DMOPs. Ya que no solamente deben buscar soluciones lo más cercanas al óptimo posible, sino que también deben adaptarse a los cambios constantes dentro del entorno y las nuevas restricciones y condiciones presenta el problema.

Dentro de la literatura se ha buscado representar la variable de tiempo t de formas distintas. Algunos autores enfocan el cambio de entorno a un cierto número de evaluaciones o segundos transcurridos. Para esta tesis. El valor de t se determina para cada uno de los problemas planteados siguiendo la estructura para el conjunto FDA [Farina et al., 2004]. La ecuación (4.1) presenta como se obtiene dicho valor. Se consideran tres variables: el número de generaciones transcurridas desde el comienzo τ ; el número de generaciones entre cada cambio, también llamado frecuencia de cambio τ_c ; y el número de diferentes entornos que presenta el problema, llamado severidad de cambio n_t .

$$t = \frac{t}{n_t} \left\lfloor \frac{\tau}{\tau_c} \right\rfloor. \quad (4.1)$$

Para esta tesis se consideran dentro del estado del arte tres conjuntos de instancias DMOP. El conjunto de instancias FDA, el conjunto dMOP y el conjunto DMZDT. En la sección 3.4 se presenta una breve descripción de cada uno de ellos. La selección de estas instancias se basa en las recomendaciones establecidas en una guía que propone estrategias

para analizar adecuadamente problemas de optimización dinámica [Helbig y Engelbrecht, 2014]. Dicha guía no solamente clasifica varias de las instancias DMOPs más comúnmente conocidas, sino que también explica a detalle varias de las características de sus frentes de Pareto óptimos, tales como su forma, continuidad, diversidad de las soluciones dentro del espacio de búsqueda y el grado de complejidad que imponen las restricciones a la búsqueda de mejores resultados. En base a esta guía se seleccionaron estos conjuntos de DMOPs debido a que contienen instancias de tipo I, II y III con frentes de Pareto óptimos convexos, no convexos, discontinuos, multimodales, con distribución de soluciones en el espacio de búsqueda de funciones objetivo uniforme y no uniforme.

Todas estas condiciones tienen el objetivo de enfrentar las propuestas de esta tesis a un conjunto de instancias con diferentes condiciones, buscando identificar sus límites y comparándolos con los límites alcanzados por otras propuestas presentadas dentro del estado del arte. La intención es mostrar las capacidades de un algoritmo hiperheurístico para solucionar DMOPs. Lo cual, como se mencionó en capítulos anteriores no ha sido propuesto anteriormente dentro de la literatura.

4.1.1 Incorporación de preferencias en problemas dinámicos

Una situación común dentro del área de optimización es la existencia de preferencias específicas en los objetivos definidas por el DM. Esto quiere decir que no solamente se deben encontrar las mejores soluciones, sino también aquellas que satisfagan los requerimientos que el DM tiene en ese momento. Esto se debe a que el DM usualmente no se encuentra interesado en soluciones óptimas dentro de todo el espacio de búsqueda disponible, sino solo en aquellas soluciones que se encuentren dentro de su región de interés.

Como se mencionó en la Sección 3.2, existen pocos trabajos dentro de la literatura que manejan técnicas de incorporación de preferencias en DMOPs. Esto se debe a que esta clase de problemas fuerzan a las técnicas propuestas no solamente a identificar un ROI, sino también adaptarse a los constantes cambios que se presentan en el entorno. Esta situación genera un nuevo grado de complejidad, ya que no solamente se debe ubicar el ROI de la forma más exacta posible, sino que también debe hacerse de la forma más rápida posible, buscando encontrar soluciones aceptables para las condiciones actuales antes de que se produzcan cambios, los cuales pueden hacer dichas soluciones de baja calidad y, en algunos casos, incluso situarlas fuera del área de soluciones factibles.

Las propuestas para solucionar DMOP actualmente existentes en la literatura se enfocan en tres técnicas de incorporación de preferencias:

Generación de punto de referencia: El DM define de forma a priori o interactiva uno o varios puntos de referencia dentro del espacio de búsqueda de funciones objetivo a los cuales se enfocan los DMOEAs.

Vector de pesos: El DM define a priori un vector de pesos que define la importancia de cada uno de los objetivos a solucionar. DNSGA-II enfoca su búsqueda basándose en el peso definido para el primer objetivo.

Función de deseabilidad: Las funciones objetivo son reemplazadas con funciones de deseabilidad que se enfocan en representar de forma específica los objetivos y restricciones de un problema a resolver.

Estas tres estrategias tienen una situación común, solo pueden estar enfocadas específicamente a un tipo de DMOP. En el caso de los puntos de referencia, estos valores no pueden ser aplicados en general al no estar normalizados. Las propuestas que trabajan actualmente utilizando vectores de pesos en DMOPs se enfocan únicamente en los valores de un objetivo, descartando la importancia de los demás objetivos. Por otra parte, la naturaleza de las funciones de deseabilidad no permite un uso general, ya que están diseñadas para utilizarse específicamente para un DMOP en específico.

En sí, existe la necesidad de profundizar la revisión de DMOEAs con incorporación de preferencias. Una situación común al trabajar con este enfoque es la falta de flexibilidad debido a las situaciones explicadas anteriormente. Por lo tanto, es necesario definir un método que permita tomar las preferencias definidas por el DM y ser capaz de incorporarlas en cualquier tipo de problema, o por lo menos en un mayor subconjunto de instancias en comparación a lo que se presenta en la situación actual.

Por lo tanto, el problema planteado para esta sección es el desarrollar una técnica de incorporación de preferencia que permita:

- Solucionar DMOPs con preferencias definidas por un DM.
- Adaptarse a los cambios en el entorno.
- Producir soluciones satisfactorias para cada periodo.
- Utilizarse en varios tipos de DMOPs con propiedades y retos específicos.

4.1.2 Métodos de evaluación de problemas multiobjetivo dinámicos

Como se ha mencionado durante las secciones anteriores una de las mayores situaciones a las cuales se enfrenta el área de optimización dinámica es la falta de estandarización en las métricas e instancias utilizadas para evaluar el desempeño de los algoritmos propuestos dentro de la literatura.

Un ejemplo que presenta la cuestión mencionada se muestra en la Tabla 3.2 dentro de la Sección 3.5.3. En la investigación de los antecedentes y trabajos relevantes realizada para esta tesis se identificaron múltiples propuestas para resolver DMOPs. Sin embargo, una constante entre ellas es que cada una utiliza un conjunto distinto de métricas para evaluar su desempeño. Aunque cada una de dichas métricas ha sido probada previamente y su efectividad y veracidad es válida, sigue haciendo falta una caracterización y determinación de relaciones entre los distintos algoritmos del estado del arte.

El problema de caracterización de algoritmos para solucionar DMOPs puede describirse de la siguiente manera.

Considérense los siguientes conjuntos:

- Un conjunto $P = \{p_1, p_2, \dots, p_n\}$ de DMOPs.

- Un conjunto $A = \{a_1, a_2, \dots, a_m\}$ de algoritmos heurísticos para solucionar un problema del conjunto P .
- Un conjunto $I = \{i_1, i_2, \dots, i_o\}$ de identificadores que permite caracterizar los problemas del conjunto P , así como los algoritmos del conjunto A .
- Un conjunto $E = \{f_1, f_2, \dots, f_q\}$ de elementos que evalúan los identificadores de I .

A partir de ello, se plantean tres situaciones a resolver:

- 1) Establecer un conjunto de relaciones $R = \{r_1, r_2, \dots, r_p\}$, tal que para cada $r_i \in R$ se establece una relación que contiene los siguientes elementos:
 - Un problema del conjunto P .
 - Un subconjunto de algoritmos de A , utilizados previamente para resolver el problema seleccionado.
 - Un subconjunto de identificadores I utilizados para caracterizar las propiedades del problema seleccionado, así como de dicho subconjunto de algoritmos.
 - Un subconjunto de aspectos de F correspondientes al subconjunto de I obtenido.
- 2) Aplicar el conocimiento obtenido en base al conjunto de relaciones R para el desarrollo de una metodología de alto nivel.
- 3) Utilizar dicha metodología en el proceso de selección de los algoritmos dentro del conjunto A para aplicar en la resolución de DMOPs pertenecientes al conjunto P .

4.2 Problema de muchos objetivos dinámico

Un problema de optimización es considerado como un problema de muchos objetivos (MaOP) cuando tiene cuatro o más objetivos a resolver. Por lo tanto, un problema de optimización dinámico de muchos objetivos (DMaOP) sigue la misma definición formal dada para DMOP presentada en la ecuación (2.9). Sin embargo, este tipo de problemas pueden brindar condiciones y retos diferentes a los exhibidos en DMOPs.

Debido a las nuevas condiciones que presenta el solucionar DMAOPs, es necesario también plantear modificaciones en los LLHs utilizados para adaptarlos en la solución de este tipo de problemas. En primer lugar, el uso de la dominancia de Pareto puede presentar inconvenientes el momento de trabajar con MaOPs y DMAOPs.

Para representar este ejemplo se muestra la Tabla 4.1, la cual presenta los valores de las funciones objetivo de dos soluciones, x y y . Dado un problema de minimización x no domina a y debido a que en f_6 y es menor que x . Si se considera a cada objetivo con la misma importancia entonces puede entenderse que x proveerá un mejor resultado que y . Esta diferencia en los resultados se puede identificar fácilmente mediante el uso de un método de agregación, tal como la suma ponderada. Sin embargo, cualquier algoritmo que utilice optimalidad de Pareto para determinar la importancia de las soluciones definirá ambas soluciones con la misma relevancia.

Tabla 4.1. Valores de los objetivos de dos soluciones.

Solución	f_1	f_2	f_3	f_4	f_5	f_6
x	0.01	0.01	0.01	0.01	0.01	0.3
y	0.9	0.9	0.9	0.9	0.9	0.29

Para evitar esta situación, uno de los primeros objetivos a plantear al resolver MaOPs y DMAOPs es buscar reemplazar el concepto de dominancia de Pareto por otras estrategias. El estado del arte presenta el uso de funciones de escalarización, como ASF y R2, así como la L-dominancia como posibles estrategias para reemplazar la no-dominancia.

4.2.1 Incorporación de preferencias en problemas dinámicos

Los DMAOPs son problemas de optimización, y como se ha mencionado previamente, este tipo de problemas pueden incluir un conjunto de preferencias establecidas por un DM. Lo anterior quiere decir que el enfoque de las estrategias utilizadas para resolver DMAOPs bajo esta condición deben enfocarse en una región de interés, ya que son las soluciones que le interesan a la persona encargada de tomar la decisión.

Existen múltiples trabajos dentro de la literatura que buscan solucionar instancias de MaOP mediante el uso de puntos de referencia, funciones de deseabilidad, L-dominancia y vectores de peso para determinar la dominancia entre soluciones, los trabajos más relevantes (al momento que esta tesis fue escrita) que llevan a cabo esta estrategia se mencionaron en la Sección 3.3. Varias de estas propuestas se apoyan de funciones de escalarización de aptitud para solucionar MaOP.

Al ser los DMAOPs una extensión del concepto DMOP con base al número de objetivos a resolver, es razonable considerar algunos de los inconvenientes presentes en las técnicas de incorporación de preferencias existentes usadas para resolver DMOPs se presenten también en las instancias de muchos objetivos. Particularmente hablando del uso de uno o múltiples puntos de referencia para dirigir la búsqueda de soluciones, la falta de una normalización de estos valores provoca que su uso requiera condiciones especiales para adaptarse a cada DMAOP. Para resolver esta situación, varias de las propuestas utilizan un punto de referencia ubicado en el punto de origen (vector cero), lo cual dirige la búsqueda hacia dicha área. Esto simula una estrategia de minimización. Sin embargo, el uso de esta estrategia para dirigir la búsqueda hacia cualquier otra área del espacio de búsqueda de funciones objetivo, hasta nuestro conocimiento, no ha sido probada aún.

Por tal motivo, es necesario determinar un punto de entrada dentro del área de investigación respecto a DMAOPs con preferencias definidas por un DM, ya que los trabajos dentro de la literatura únicamente se enfocan en MaOPs, y como se mencionó anteriormente, varios de ellos están enfocados en ciertos problemas en particular. La intención de esta tesis es el desarrollo de metodologías capaces de establecer una relación entre las instancias, algoritmos y métricas de desempeño, buscando presentar una alternativa que pueda ser utilizada en un mayor abanico de instancias, con diferentes retos y características.

Es por ello, que el problema principal planteado para esta sección consiste en el desarrollo de una técnica de incorporación de preferencias y su anexión dentro de un algoritmo o metodología para resolver DMAOPs. Dicha técnica debe permitir:

- Solucionar DMAOPs con preferencias definidas por un DM.
- Adaptabilidad a diferentes condiciones, instancias y entornos dinámicos

- Producir soluciones satisfactorias.

4.2.2 Retos para la evaluación de problemas de muchos objetivos dinámicos

Al momento de desarrollar una propuesta que permita incorporar de preferencias de un DM en un DMAOP de forma satisfactoria, se identificó la falta de un estado del arte enfocado totalmente a este rubro. Esta situación lleva a múltiples complicaciones que pueden concentrarse en dos conceptos:

Primero, resulta muy complicado y exigente en cuestión de esfuerzo y tiempo humano y computacional el obtener los frentes de Pareto óptimos de cada problema dinámico. Este cálculo se dificulta ya que puede haber múltiples frentes (problemas de tipo II y III). Además, se debe considerar la severidad de cambio y el número de objetivos que tratar. Por este motivo es preferible utilizar métricas que no consideraran un frente de Pareto óptimo.

Segundo, hasta nuestro conocimiento ningún trabajo del estado del arte se enfoca por completo al objetivo de este proyecto. Esto agrega complejidad desde la fase de configuración de los elementos de la hiperheurística. Esto significa que se requiere realizar pruebas preliminares para determinar la configuración más efectiva para cada situación evaluada, tanto para DMAOPs no preferenciales como para DMAOPs con preferencias incorporadas por un DM.

4.2.3 Métodos de evaluación de problemas de muchos objetivos dinámicos

La evaluación de DMAOPs presentan una situación en particular. Como se mencionó en el primer punto la Sección 4.2.2, la dificultad para identificar frentes de Pareto óptimo en este tipo de problemas limita el conjunto de métricas de evaluación posibles a utilizar. De hecho, dentro de la literatura los métodos utilizados más comúnmente no están basados en un frente de Pareto óptimo, sino en la comparación en base a hipervolumen o un punto de referencia.

Sin embargo, es necesario determinar una estandarización entre las métricas utilizadas respecto a las instancias resueltas. También, se debe considerar, como se mencionó anteriormente, que la falta de trabajos dentro del estado del arte aumenta la dificultad de este proceso, ya que no existe, hasta nuestro conocimiento, algún trabajo, el cual siga el mismo enfoque de la propuesta presentada en esta tesis. Por tal motivo, es razonable pensar que el objetivo a satisfacer en esta cuestión es generar un punto de entrada para el proceso de estandarización de métricas e instancias.

Para buscar obtener dicho punto de entrada se buscará llevar a cabo la definición de relaciones entre instancias DMAOP, algoritmos, identificadores de desempeño y los elementos que definen dichos identificadores. Esto, siguiendo la misma definición determinada en la Sección 4.1.2, pero aplicada en DMAOPs.

Capítulo 5

Propuesta 1: Hiperheurísticas para la resolución de DMOPs y DMaOPs

En este trabajo de investigación se propone definir una estandarización entre diversos DMOEAs propuestos en trabajos previos, así como definir una hiperheurística capaz de seleccionar dentro de un conjunto de heurísticas dinámicas aquellas que provean la solución de mayor calidad. En este capítulo se presentan las propuestas realizadas para solucionar cada uno de los problemas propuestos. Estos problemas son: la solución de DMOPs, la solución de DMaOPs, la incorporación de preferencias de un DM para estos problemas y la resolución de instancias preferenciales buscando un ROI cercano al óptimo.

5.1 Hiperheurísticas DHH y DPEM-HH

A partir de los elementos mencionados en las secciones previas se propone la idea de establecer una metodología que permite utilizar un conjunto de heurísticas para solucionar un problema estableciendo un conjunto de relaciones basados en métricas de desempeño. La metodología presentada es una hiperheurística denominada Hiperheurística Multiobjetivo Dinámica (DHH por sus siglas en inglés). DHH permite la unificación de varias metaheurísticas dentro de un mismo proceso. Esta hiperheurística está basada en la hiperheurística HH_CF [Maashi et al., 2014], en la cual se diseñó una hiperheurística capaz de resolver MOPs estáticos usando MOEAs como LLHs.

La Figura 5.1 presenta el diagrama de flujo de DHH. Primero, la hiperheurística crea una población inicial aleatoria. Después, se lleva a cabo la selección de un LLH otorgándole a cada LLH una copia de la población inicial y ejecutándose cada una por un número de generaciones de prueba. Los resultados de cada LLH son evaluados mediante un conjunto de métricas de desempeño y usados dentro del método de selección de LLH utilizado para llevar a cabo elegir el LLH más adecuado en base a dichas métricas. Luego, se lleva a cabo la ejecución del LLH seleccionado hasta alcanzar una de dos condiciones. La primera condición determina que si el LLH detecta un cambio en el entorno debe detener su ejecución y comparar las soluciones obtenidas con las soluciones actuales bajo un criterio de aceptación. Si el criterio determina que las nuevas soluciones son aceptables, estas se convierten en las nuevas soluciones actuales. Después de esto, se le otorga nuevamente a cada LLH una copia de la población de soluciones actuales para llevar nuevamente el proceso de selección de LLH de la misma forma que se mencionó anteriormente.

La segunda condición en la cual el LLH seleccionado se detiene es si la hiperheurística llega a alcanzar el criterio de terminación. En ese caso la última población obtenida es comparada contra la población actual bajo el criterio de aceptación. En base a dicho criterio se determina si se reemplaza la población actual o no por la última población obtenida por el LLH utilizado durante ese momento.

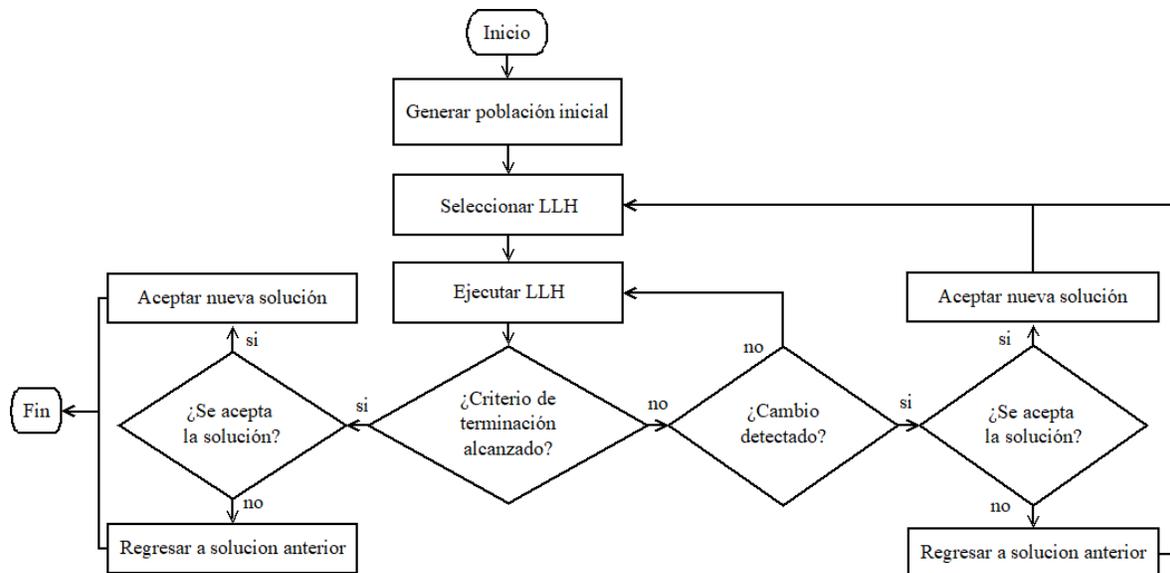


Figura 5.1. Diagrama de flujo de DHH.

Acorde a su estructura, DHH se clasifica como una hiperheurística de selección de heurísticas que utiliza aprendizaje en línea. Esto quiere decir que la información disponible de cada uno de los LLHs candidatos se actualiza en cada etapa de selección en base los resultados obtenidos durante el proceso de solución. También, significa que los LLHs disponibles para utilizar provienen de un conjunto de heurísticas previamente desarrolladas e incorporadas a la hiperheurística. Por lo que el objetivo de la hiperheurística es llevar a cabo la selección de aquel LLH que pueda proveer los mejores resultados bajo el entorno actual del problema en base a la información actual obtenida por los resultados presentados por cada LLH.

DHH está desarrollado para trabajar con un conjunto de DMOEAs, los cuales tiene cada uno un proceso complejo en cual una población sigue un proceso de selección, cruzamiento y mutación de soluciones candidatas pertenecientes a dicha población. Esto muestra una de las aportaciones presentadas por DHH dentro del área de optimización dinámica. Hasta el momento cuando esta tesis fue presentada, no existía, hasta el conocimiento del autor, dentro del estado del arte un hiperheurística enfocada en solucionar DMOPs mediante el uso de DMOEAs como LLHs.

El método de selección de LLH utilizado por DHH es la función de elección (CF), basada en las ecuaciones (3.23) y (3.24), el valor de c_2 para DHH está determinado en base a la cantidad de milisegundos transcurridos desde la última vez que cada LLH fue seleccionado. Un conjunto de métricas de desempeño sea utiliza para evaluar la calidad de los resultados obtenidos por cada LLH después de un conjunto de generaciones de prueba, en el cual cada heurística se ejecuta sobre una misma población inicial. El uso de CF tiene como objetivo de analizar la capacidad de exploración e intensificación de cada uno de los LLHs, apoyando la selección en estos criterios.

La función de elección es utilizada tomando en consideración la búsqueda un balance entre la exploración y la intensificación. Por una parte, los valores obtenidos por la variable c_1 de la ecuación (3.23) promueven la intensificación. Esto, se debe a que los resultados por la doble clasificación utilizada enfocan su búsqueda hacia los LLHs que proveen mejores soluciones para el periodo actual, de una manera similar a un método voraz. Por otro lado, los valores definidos en c_2 promueven la exploración, ya que no consideran la calidad de los resultados obtenidos por el LLH, sino únicamente el tiempo transcurrido desde su última selección. Esto quiere decir, que, si un LLH no ha sido utilizado por un largo periodo de tiempo, el valor de esta variable se hará mayor hasta el punto de dirigir la selección hacia dicho LLH, incluso si no provee los mejores resultados para el periodo actual. Esto permite a DHH explorar áreas que hubiesen sido muy complicado llegar utilizando únicamente una estrategia voraz, tal como se presenta en c_1 , las cuales pueden potencialmente llevar a mejoras en las soluciones a mediano y largo plazo, así como evitar estancamientos en óptimos locales.

El criterio de aceptación de soluciones utilizado por DHH es “*All Moves*”, esto quiere decir que cualquier nueva población presentada por el LLH seleccionado será designado como la nueva población actual para la siguiente etapa de selección de LLH. La elección de este criterio se basa en su simplicidad y buscando aprovechar una de las propiedades principales de una hiperheurística, la cual le permite para generar mejores soluciones combinando las fortalezas múltiples LLHs. La intención principal de DHH es llevar a cabo una prueba para comprobar la factibilidad de la aplicación de una hiperheurística para resolver DMOPs mediante el uso de DMOEAs como LLHs. Después de comprobar su funcionalidad se busca agregar los componentes adicionales propuestos en esta tesis (evolucionabilidad poblacional, incorporación de preferencias, métodos para el manejo de problemas con muchos objetivos).

El Algoritmo 5.1 describe a detalle el procedimiento llevado a cabo por DHH, las variables que se introducen a este algoritmo son la frecuencia de cambio, el número de generaciones de prueba por periodo (st) y el conjunto de LLHs. Primero se genera una población inicial pop y se asigna el valor de st a una variable que controlara el total de generaciones de prueba por cada LLH en cada ciclo τ_{test} (línea 4). Después, mientras no se haya alcanzado el criterio de terminación se lleva a cabo el resto del algoritmo. En caso de haber detectado un cambio en el entorno o ser la primera generación se le asigna a cada LLH una copia de la población actual (np_i). Posterior a eso, cada LLH es ejecutado por τ_{test} generaciones (línea 8).

Después de que cada LLH haya terminado de ejecutarse, se hace una evaluación de las poblaciones obtenidas por cada LLH bajo las métricas utilizadas por CF generando una matriz de dos dimensiones MET donde las filas representan cada LLH y las columnas el valor de cada una de las métricas utilizadas (línea 11). Un LLH es seleccionado siguiendo la estrategia establecida las ecuaciones (3.23) y (3.24), la población actual del LLH seleccionado es asignada a pop (líneas 12 y 13). Finalmente, el LLH seleccionado es ejecutado usando pop (línea 15) hasta que se detecte un cambio, en cuyo caso, se iniciara el proceso de selección de heurísticas de bajo nivel nuevamente (líneas 3 a 13).

Algoritmo 5.1 DHH. Hiperheurística multiobjetivo dinámica.

INICIO

```

1:  $pop \leftarrow \text{GenerarPoblacionInicial}()$ 
2: mientras no se alcance criterio de paro
3:   si se detecta cambio o primera generación
4:      $\tau_{test} \leftarrow st$ 
5:     para  $i = 1$  hasta  $|LLH|$ 
6:        $np_i \leftarrow \text{CopiarPoblacion}(pop)$ 
7:       para  $j = 1$  hasta  $st$ 
8:          $np_i \leftarrow \text{EjecutarLLH}(llh_i, np_i, \tau_{test})$ 
9:       fin para
10:    fin para
11:     $MET \leftarrow \text{EvaluacionMetricas}(NP)$ 
12:     $LLH_{seleccionado} \leftarrow \text{SeleccionarLLH}(LLH, MET)$ 
13:     $pop \leftarrow \text{AsignarPoblacion}(LLH_{seleccionado}, NP)$ 
14:  sino
15:     $pop \leftarrow \text{EjecutarLLH}(s, pop, \tau_t - \tau_{test})$ 
16:     $\tau_{test} \leftarrow 0$ 
17:  fin si
18: fin mientras
FIN
```

También, se propone el uso de una versión de DHH la cual incorpora el cálculo y uso de evolucionabilidad poblacional dentro de su proceso. Esta metodología se denomina como Hiperheurística Multiobjetivo Dinámica basada en Evolucionabilidad Poblacional (DPEM-HH por sus siglas en inglés). Su procedimiento es bastante similar al que realiza DHH. La Figura 5.2 muestra el diagrama de flujo de las operaciones realizadas por DPEM-HH. De igual forma que DHH, el proceso inicia con la generación de una población inicial aleatoria.

Sin embargo, a diferencia de DHH, no se procede directamente a la selección de LLH, sino que primero cada LLH es asignado una copia de la población inicial y ejecutado por un cierto número de generaciones de prueba. La evolucionabilidad poblacional de cada generación es evaluada mediante una adaptación de la ecuación (3.22) para poder aplicarse a problemas dinámicos. Dicha adaptación se representa en la ecuación (5.1). Después, se obtiene un valor promedio siguiendo la ecuación (5.2), donde $evp_t(P_i)$ representa el valor de la evolucionabilidad poblacional en el periodo t para la población actual en la generación de prueba i . Este valor apoya la toma de decisión de la hiperheurística para definir cuál es el LLH a utilizar durante el primer periodo.

$$evp_t(P_i) = \begin{cases} \frac{\sum_{P_{ij} \in N^+(P_i)} \frac{|f_t^b(P_i) - f_t^b(P_{ij})|}{NP}}{\sigma(f_t(P_i))} \cdot \frac{1}{|N(P_i)|}, & |N^+(P_i)| > 0, \\ 0, & |N^+(P_i)| = 0 \end{cases} \quad (5.1)$$

$$\overline{evp}_i = \frac{1}{st} \sum_{i=1}^{st} evp_t(P_i). \quad (5.2)$$

El resto del procedimiento de DPEM-HH sigue los mismos pasos que DHH hasta que se detecte un cambio. Cuando un cambio es detectado, se usa el criterio de aceptación de

soluciones determinado y se asigna una copia de la población actual a cada LLH para ejecutarse por un número de generaciones de prueba, evaluando en cada generación su evolucionabilidad poblacional. Al final, se obtiene un valor promedio de este método FLA. Este valor se utiliza como un apoyo para llevar a cabo el proceso de selección de LLH para el periodo correspondiente.

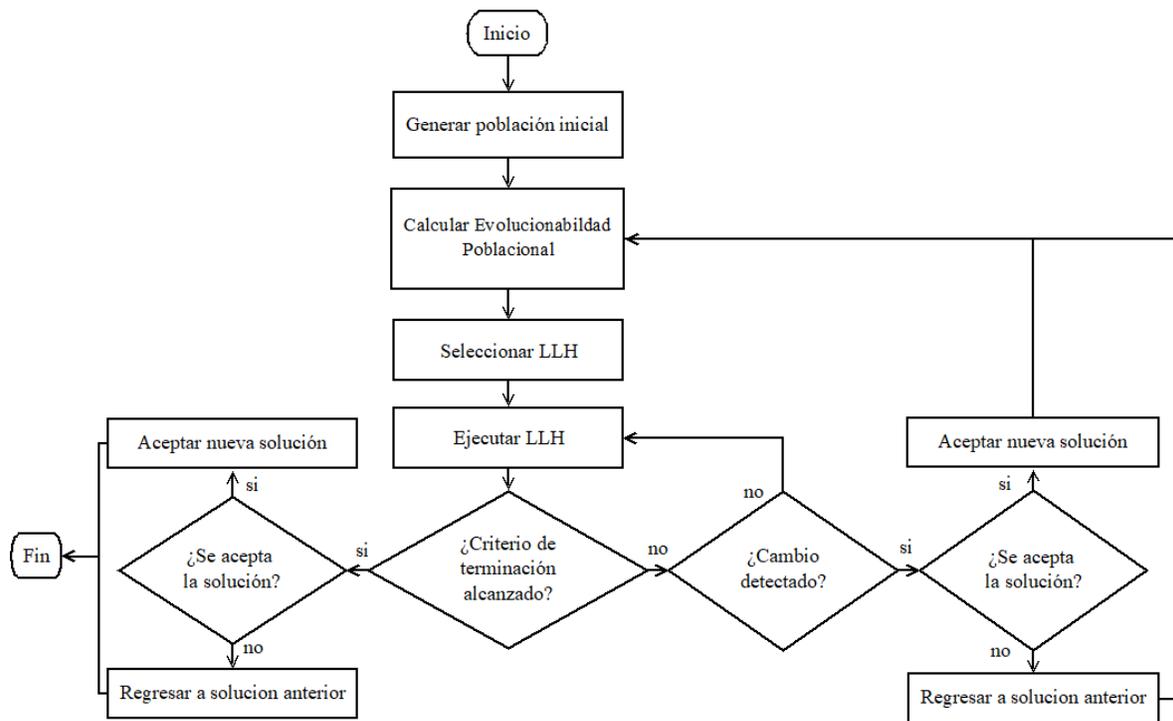


Figura 5.2. Diagrama de flujo de DPDM-HH.

En esta tesis se propone el uso de múltiples combinaciones de métodos de selección de LLH y criterios de aceptación de solución para DPDM-HH los cuales se van a comparar para determinar cual tiene mayor efectividad en un conjunto de DMOPs con diferentes características. El objetivo de esto es observar los efectos producidos por el uso de la evolucionabilidad poblacional en la hiperheurística y revisar su efectividad al ser utilizado junto con el conjunto de métricas de desempeño usado por DHH. Los métodos de selección de LLH usados por DPDM-HH se explican a continuación:

Voraz (G). En cada generación de prueba se crea un conjunto de poblaciones hijas y se calcula la evolucionabilidad poblacional para cada LLH siguiendo la ecuación (5.1) y (5.2). Como se puede ver en dichas ecuaciones, se utiliza un valor único para definir la aptitud. Por ello, se aplicó un método de agregación de valores para un DMOP utilizando una sumatoria ponderada [Li y Deb, 2017] para los k objetivos como se muestra en la ecuación (5.3). A cada objetivo se le asignó un peso λ_i idéntico ($\lambda_i = 1/k$). Cabe aclarar que aun cuando se conoce que esta aproximación no es adecuada para problemas con frentes no convexos, solo es utilizado para proceso comparativo en la selección de LLH. Por otra parte, la dominancia de Pareto se utiliza para determinar las soluciones que conforman las siguientes generaciones.

$$\min f(x|\lambda) = \sum_{i=1}^m \lambda_i f_i(x), \quad s. t. \lambda_i \geq 0, \sum_{i=1}^m \lambda_i = 1. \quad (5.3)$$

Función de Elección (CF). Se combinan los resultados obtenidos por la evolucionabilidad poblacional promedio y un conjunto de métricas de desempeño dentro de una función de elección. Con esto se busca analizar la posibilidad de aplicar este método FLA en un método de selección de heurística más complejo para obtener soluciones de buena calidad. La función de elección utilizada se basa en las ecuaciones (3.23) y (3.24). La variable θ en la función c_1 , mostrada en la ecuación (3.24) es asignada al ranking correspondiente a la evolucionabilidad poblacional promedio, ya que se considera como el valor más relevante para esta hiperheurística. El valor de c_2 se basa en los milisegundos transcurridos desde última vez que cada LLH fue seleccionado.

Ruleta con memoria (ROU). El proceso de este método de selección se representa en el Algoritmo 5.2, la cual presenta una variación del método de selección por ruleta [Santiago et al., 2019]. Inicialmente se le asigna una probabilidad idéntica ($1/|LLH|$) a cada LLH. Para los periodos posteriores a la inicial se calcula la evolucionabilidad poblacional promedio de cada LLH y se definen nuevas probabilidades. Estas probabilidades se suman en una sola variable. Posteriormente se selecciona un LLH aleatoriamente y se resta su probabilidad a la sumatoria. Cuando la sumatoria sea menor o igual que 0 se termina el proceso y el último LLH seleccionado es utilizado en el siguiente periodo.

Para definir los nuevos valores de probabilidad de cada LLH en un periodo de selección, tal como se ve en la línea 6 del Algoritmo 5.2, se define un valor d , que es la combinación de la probabilidad actual y la diferencia de evolucionabilidad poblacional promedio obtenida durante el periodo actual y el periodo anterior para cada LLH r . Se utiliza la variable β , llamada coeficiente de evolucionabilidad, para definir la potencia de esta diferencia.

La probabilidad de selección de cada LLH para el siguiente periodo se define en base al valor d obtenido por cada LLH dividido entre la suma de todos los valores d . Las ecuaciones (5.4) y (5.5) muestran el cálculo de d y la nueva probabilidad de un LLH, respectivamente.

$$d_{(r,t+1)} = p_{(r,t)} + \beta * (evp_{t+1}(r) - evp_t(r)), \quad (5.4)$$

$$p_{(r,t+1)} = \frac{d_{(r,t+1)}}{\sum_{j=1}^{|LLH|} d_{(j,t+1)}}. \quad (5.5)$$

Aleatorio (SR). A cada LLH se le asigna una misma probabilidad ($1/|LLH|$) y se selecciona de forma aleatoria un LLH. Este método de selección no utiliza evolucionabilidad poblacional dentro de su proceso.

Por otra parte, se asignaron tres diferentes criterios de aceptación para trabajar con cada uno de los métodos de selección de LLH incorporados a DPEM-HH.

Todos los movimientos “All Moves” (AM): Se aceptan todas las soluciones obtenidas por el LLH, sin considerar si las soluciones son o no de mejor calidad que las actuales.

Algoritmo 5.2. Método de selección - Ruleta con memoria.

INICIO

```
1: si  $t = 0$ 
2:   definir probabilidades idénticas para cada LLH
3: fin si
4: sino
5:   calcular evolucionabilidad poblacional promedio de cada LLH
6:   calcular nueva probabilidad de cada LLH usando evolucionabilidad poblacional
7: fin sino
8: se suman las probabilidades en la variable suma
9: mientras  $suma > 0$ 
10:  seleccionar un LLH usando las nuevas probabilidades
11:  restar probabilidad del LLH seleccionado a suma
12: fin mientras
```

FIN

Solo Mejor Dinámico (IOD): Al final de un periodo estático, el último conjunto de soluciones obtenido por un LLH es comparado con el conjunto de soluciones actuales en base al conjunto de métricas de desempeño presentados previamente. Solamente se aceptan las nuevas soluciones si presentan un mejor valor que el conjunto actual.

El proceso de análisis y aceptación o rechazo de las nuevas soluciones está representado en el Algoritmo 5.3. En el primer periodo ($t = 0$), se evalúa el LLH seleccionado obteniendo sus valores de métricas de desempeño y guardando su *POF*.

Después, se evalúa el nuevo conjunto de soluciones y se obtiene su valor de métricas de desempeño y *POF*. Se comparan los valores de las métricas del conjunto de nuevas soluciones y las soluciones actuales siguiendo un orden lexicográfico. Esto significa que, si un *POF* supera a otro en una métrica con rango superior, se declara ganador.

El orden de las métricas de desempeño es IGD, RNI, MS y HVR. Se definió este orden por la relevancia de información que provee cada métrica. IGD se colocó como el más importante debido a que provee información tanto sobre convergencia como diversidad de un *POF* respecto al *POF**.

En caso de que el nuevo *POF* supere al actual, se actualizan los valores y se acepta la solución. En caso contrario se reevalúa el *POF* actual bajo el frente de Pareto óptimo (*POF**) correspondiente al periodo actual, obteniendo nuevos valores de métricas. Si el nuevo *POF* supera al actual bajo este *POF** se actualizan los valores y se acepta la solución. En caso contrario se rechaza la solución y se mantiene la solución previa.

Mejor o Igual Dinámico (IED): Este criterio de aceptación de soluciones lleva a cabo un procedimiento similar al criterio Solo Mejor Dinámico. IED sigue el proceso determinado en el Algoritmo 5.3. Sin embargo, el nuevo conjunto de soluciones generado por el LLH utilizado durante el periodo actual es aceptado como la nueva población si sus valores bajo las métricas de desempeño de mayor rango lexicográfico son mejores o iguales que los valores obtenidos por la población actual bajo las condiciones establecidas por el entorno en el periodo donde se está evaluando.

Algoritmo 5.3. Método de aceptación - Solo mejor Dinámico.

INICIO

```
1: en  $t = 0$  evaluar LLH seleccionado y obtener  $MET_t$  y  $POF_t$ 
2: mientras no se alcance criterio de paro
3:   evaluar conjunto de soluciones actual y obtener  $MET_{t+1}$  y  $POF_{t+1}$ 
4:   comparar valores de  $MET_t$  y  $MET_{t+1}$  por orden lexicográfico
5:   si  $MET_{t+1}$  es mejor que  $MET_t$ 
6:     actualizar  $MET_t$  y  $POF_t$  con  $MET_{t+1}$  y  $POF_{t+1}$ 
7:     aceptar solución
8:   si no
9:     actualizar valores de  $MET_t$  evaluando  $POF_t$  con  $POF_{t+1}^*$ 
10:    comparar valores de  $MET_t$  y  $MET_{t+1}$  por orden lexicográfico
11:    si  $MET_{t+1}$  es mejor que  $MET_t$ 
12:      actualizar  $MET_t$  y  $POF_t$  con  $MET_{t+1}$  y  $POF_{t+1}$ 
13:      aceptar solución
14:    si no
15:      rechazar solución
16:    fin si
17:  fin si
18:   $t = t+1$ 
19: fin mientras
```

FIN

De igual forma que para DHH, un conjunto de DMOEA se utiliza como heurísticas de bajo nivel para DPEM-HH. La clasificación de DPEM-HH es idéntica a DHH (selección de LLHs con aprendizaje en línea). Además, como se mencionó anteriormente, la evolucionabilidad poblacional se considera en el proceso de selección de LLH. La introducción de un método FLA dentro de una hiperheurística para resolver problemas de optimización dinámica es una de las aportaciones presentadas por esta tesis. Al momento que esta tesis fue presentada, el uso de métodos FLA dentro de estrategias de selección de algoritmos solo había sido incorporado en problemas de optimización estáticos y de un solo objetivo. La propuesta de DPEM-HH presenta por lo tanto un punto de entrada dentro del campo. La intención de esta hiperheurística es aprovechar la capacidad de la evolucionabilidad poblacional para obtener información que permite establecer una relación entre la complejidad del problema y la capacidad de cada algoritmo utilizado para generar mejores resultados. Esto en teoría, le debe permitir a DPEM-HH seleccionar aquella heurística que produzca soluciones de mejor calidad para el entorno actual. Esto también significa una potencial mejor adaptación a los cambios en el problema, ya que no solamente se detecta el cambio en el entorno, sino también se detecta cuáles son los algoritmos candidatos capaces de sobreponerse a las alteraciones en el nuevo entorno y generar soluciones de mejor calidad.

El proceso de DPEM-HH se representa en el Algoritmo 5.4. De igual forma que DHH, los valores de entrada son la frecuencia de cambio, el número de generaciones de prueba por periodo y el conjunto de LLHs. El valor de salida es la última población aceptada por el criterio de aceptación usado por la hiperheurística. Primero, se genera una población inicial aleatoria. Después, mientras no se alcance el criterio de terminación se lleva a cabo el ciclo principal de DPEM-HH. Dentro de este ciclo se asigna el valor de la frecuencia de cambio a st y se asigna una copia de la población actual (pop_evp_i) a cada LLH (líneas 4 a 6).

Cada LLH es ejecutado por st generaciones de prueba. Al final de cada generación se genera una vecindad de poblaciones NP , las cuales consisten en poblaciones obtenidas después de un ciclo evolutivo (selección, cruzamiento y mutación) a partir de la población actual. Usando esta vecindad se calcula la evolucionabilidad poblacional del LLH para la generación actual. Los valores obtenidos en cada generación de prueba se suman en una variable evp_i (líneas 8 a 10). Después de concluir con este ciclo, se obtiene un valor promedio de la evolucionabilidad poblacional (línea 12).

Siguiendo la estrategia definida por DHH, al final de las generaciones de prueba DPEM-HH evalúa las poblaciones obtenidas por cada LLH en base a un conjunto de métricas de desempeño (línea 14). Después, se lleva a cabo la selección del LLH y asignación de la población actual apoyado de los valores obtenidos por las métricas y las evolucionabilidad poblacional, acorde a como los utilice el método de selección de LLH incorporado en la hiperheurística (línea 15 y 16). Finalmente, se usa el LLH seleccionado durante el resto de las generaciones determinadas en el periodo, las cuales equivalen al valor de la frecuencia de cambio menos el número de generaciones de prueba usadas por el LLH seleccionado.

Algoritmo 5.4 DPEM-HH. Hiperheurística multiobjetivo dinámica basada en evolucionabilidad poblacional.

INICIO

```
1:  $pop \leftarrow$  GenerarPoblacionInicial()
2: mientras no se alcance criterio de paro
3:   si se detecta cambio o primera generación
4:      $\tau_{rest} \leftarrow st$ 
5:     para  $i = 1$  hasta  $|LLH|$ 
6:        $pop\_evp_i \leftarrow$  CopiarPoblacion( $pop$ )
7:       para  $j = 1$  hasta  $st$ 
8:          $pop\_evp_i \leftarrow$  EjecutarLLH( $llh_i, pop\_evp_i, \tau_{rest}$ )
9:          $NP \leftarrow$  CrearVecindad( $llh_i, pop\_evp_i$ )
10:         $evp_i \leftarrow evp_i +$  CalcularEvolucionabilidad( $pop\_evp_i, NP$ )
11:      fin para
12:       $evp_i \leftarrow evp_i/st$ 
13:    fin para
14:     $MET \leftarrow$  EvaluacionMetricas( $POP\_EVP$ )
15:     $LLH_{seleccionado} \leftarrow$  SeleccionarLLH( $LLH, EVP, MET$ )
16:     $pop \leftarrow$  AsignarPoblacion( $LLH_{seleccionado}, POP\_EVP$ )
17:    sino
18:       $pop \leftarrow$  EjecutarLLH( $s, pop, \tau_t - \tau_{rest}$ )
19:       $\tau_{rest} \leftarrow 0$ 
20:    fin si
21: fin mientras
FIN
```

Respecto a los LLHs a utilizar, se deben considerar dos criterios principales. En primer lugar, la heurística debe ser capaz de solucionar problemas dinámicos. También, debe estar diseñada para la solución de problemas multiobjetivo. En base a esta información y la accesibilidad para trabajar con este tipo de algoritmos, se busca utilizar ambas versiones de la metaheurística DNSGA-II [Deb et al., 2007]. Además, dentro de esta tesis se propone el uso de una versión, denominada DNSGA-II-AB. Este algoritmo utiliza ambos métodos de adaptación al cambio, propuestos en el trabajo original, en una misma subpoblación al momento de detectar un cambio en el entorno.

Dentro de la literatura existen múltiples trabajos enfocados a la solución de problemas de optimización multiobjetivo. Estas propuestas, sin embargo, son en su mayoría enfocadas a problemas que no presentan cambios con el paso del tiempo. En base a la accesibilidad a los algoritmos y su relevancia dentro del campo, se busca utilizar y adaptar a dinamismo los MOEAs MOCell [Nebro et al., 2009], SPEA2 [Zitzler et al., 2001] y GDE3 [Kukkonen y Lampinen, 2005]. Para ello, a cada uno se le incorpora un método de detección al cambio basado en el número de generaciones transcurridas y un método de adaptación al cambio que se basa en el reemplazo de un subconjunto de soluciones por nuevas soluciones generadas de forma aleatoria, siguiendo el método definido en DNSGA-II-A.

5.2 Hiperheurística DHHMa

En base a los resultados obtenidos en las experimentaciones usando las propuestas sobre problemas de optimización de hasta tres objetivos se llevó a cabo un análisis de la estructura de DHH. En esta sección se propone un rediseño de tal forma que la hiperheurística sea capaz de solucionar DMAOPs de una forma adecuada. Para ello se realiza una reestructuración de varios elementos dentro de la hiperheurística desarrollada, tales como el proceso de selección de LLHs, así como diversas estrategias aplicadas por los LLHs utilizados.

Como se analizó en la Sección 3.7.6, existen dentro de la literatura múltiples hiperheurísticas propuestas para solucionar MaOP. Los métodos de selección de heurísticas de bajo nivel utilizados por dichas propuestas se basan principalmente en el uso de una ruleta o una función de elección basada en desempeño y tiempo transcurrido desde la última selección de cada heurística.

Una parte principal del objetivo establecido para esta tesis es el desarrollo de una metodología capaz de solucionar problemas con diferentes condiciones, sin importar el número de objetivos o condiciones del problema. Por este motivo, la estructura utilizada en DHH en la Figura 5.1 es utilizada para la hiperheurística propuesta en esta sección. Sin embargo, los métodos de selección, métricas de desempeño utilizadas y la estructura de sus heurísticas de bajo nivel son diferentes. Por este motivo, y para evitar confusión con las propuestas anteriores, se nombra a la versión de la hiperheurística usada en esta sección como Hiperheurística para Muchos Objetivos Dinámica (DHHMa).

Los resultados obtenidos por las experimentaciones realizadas respecto a las propuestas de secciones previas en este capítulo son presentados en las Secciones 6.1 a 6.4. En dichos experimentos se llegó a la conclusión que dentro de un entorno dinámico preferencial de hasta tres objetivos el uso de métodos de selección de LLH relacionados a la función de elección de HH_CF [Maashi et al., 2014] tienen un mayor impacto positivo con respecto a métodos basados en ruleta. Sin embargo, ambos métodos deben considerarse, ya que hay casos en los cuales ambos métodos presentan ventaja.

Por ese motivo, se propone el diseño de una hiperheurística para solucionar DMAOPs utilizando diversos métodos de selección. Los cuales consisten en el método de función de elección basado en las ecuaciones (3.23) y (3.24). De igual forma, se busca usar un método voraz usando dichas ecuaciones. Su naturaleza voraz se debe a que únicamente se considera

el valor de la variable c_1 . También, se utiliza un método de selección basado en la ruleta propuesta en el algoritmo FAME [Santiago et al., 2019]. El criterio de aceptación de soluciones utilizado es *All Moves* [Cowling et al., 2000] debido a su simplicidad y que mostró ser efectivo para DMOPs con y sin preferencias de dos y tres objetivos.

Cuando se trabaja con MaOPs, tanto estáticos como dinámicos, conocer el frente de Pareto óptimo puede hacerse más complicado conforme aumenta el número de objetivos, tal como se menciona en la Sección 4.2.2. Esto, puede provocar que la confiabilidad de las métricas de desempeño que se basan en la comparación con el POF^* se vea afectada. Por tal motivo se plantea el uso de un conjunto de métricas de desempeño diferentes. Dichas métricas han demostrado ser formas de medición efectivas en diversos trabajos dentro del estado del arte. Sin embargo, como se concluyó en la revisión de la literatura, las métricas propuestas para este tipo de problemas estas enfocados en problemas que no presentan cambios en su entorno. Por lo tanto, es necesario adaptarlas a problemas de optimización dinámicos. Las métricas utilizadas y su adaptación a dinamismo se presentan a continuación.

Proporción de soluciones no L-dominadas (LRNI): Siguiendo la estrategia definida en el RNI, se obtiene una proporción del número de soluciones no L-dominadas en base a la ecuación (3.7) y el número de soluciones de una población P en el periodo t . El coeficiente de normalización p este definido en 2 para permitir definir el valor de $\|F(x)_p\|$ en base a una distancia euclidiana.

$$LRNI_t = \frac{|LND_t|}{|P_t|}, \quad (5.6)$$

$$LND_t = \{x \in P_t \mid \nexists y \in P_t, B_s(x, y) < W_s(x, y) \wedge \|F(x)_p\| > \|F(y)_p\|\}. \quad (5.7)$$

Indicador R2: Para este caso se adapta la ecuación (3.9) para un entorno dinámico mediante la inserción de una variable de tiempo t . En esta tesis se propone el uso de la función de agregación de Tchebycheff como la función de utilidad usada para evaluar a los algoritmos al usar esta métrica. Además, con el objetivo de obtener una mayor información por parte de esta métrica, se busca utilizar múltiples vectores de peso (Λ) en los cuales se evalúan las soluciones obtenidas. Por lo tanto, la definición formal del indicador R2 utilizado es

$$R2_t = \frac{1}{|\Lambda|} \sum_{\lambda \in \Lambda} \min_{a \in A_t} \{ \max_{i \in \{1, \dots, m\}} \{\lambda_i |z_{i,t} - f_{i,t}(a)|\} \}, \quad (5.8)$$

donde:

- Λ es un conjunto de vectores de peso.
- A_t es un conjunto de soluciones obtenidas en el periodo t .
- $f_{i,t}(a)$ es un valor del objetivo i en el tiempo t para la solución a .
- $z_{i,t}$ es el valor del objetivo i en el tiempo t del punto de referencia z .

Función de escalarización de logros (ASF): Esta métrica se basa en la definición presentada en la ecuación (3.10). De igual forma que R2, se requiere la inserción de una variable de tiempo t la cual permita adaptar la métrica a problemas de optimización dinámica.

La propuesta en WASF-GA [Ruiz et al., 2015] define que el uso de un conjunto de vectores de peso permite una diversidad en el conjunto de soluciones no dominadas. Por tal motivo, esta métrica utiliza un conjunto de vectores de pesos Λ para evaluar las soluciones, definiendo un valor único mediante un promedio. Para evaluar a una población P , se sigue la definición establecida por el método de agregación de Tchebycheff. La definición formal se presenta en la ecuación (5.9). Cabe mencionar que el valor de q para esta tesis está definido en 0.001, siguiendo las recomendaciones establecidas en el diseño de WASF-GA [Ruiz et al., 2015].

$$ASF(q, P_t, \Lambda) = \frac{1}{|\Lambda|} \sum_{\lambda \in \Lambda} \min_{x \in P_t} \left\{ \max_{i=1, \dots, m} \{ |\lambda_i (f_{i,t}(x) - z_{i,t})| \} + q \sum_{i=1}^m |\lambda_i (f_{i,t}(x) - z_{i,t})| \right\}. \quad (5.9)$$

Hipervolumen (HV) [Zitzler y Thiele, 1998]: Se define como la porción en el espacio objetivo dominada por una población P que va desde el frente de Pareto generado por P hasta un punto de referencia z . En este caso, mientras mayor sea el hipervolumen, mejor es la calidad de la población. Su forma de evaluar permite obtener información respecto a convergencia y diversidad del frente de Pareto respecto a un POF* real (si es que puede ser calculado) o tentativo.

Debido a las nuevas condiciones que presenta el solucionar DMAOPs, es necesario también plantear modificaciones en los LLHs utilizados para adaptarlos en la solución de este tipo de problemas. En las propuestas previas presentadas dentro de este capítulo se han utilizado DMOEAs basados en el criterio de dominancia de Pareto. Como se mencionó en la Sección 4.2, dicha estrategia puede verse afectada conforme aumenta el número de objetivos a resolver. Por dicho motivo, se propone para los trabajos relacionados en esta tesis utilizar un conjunto de LLHs que hayan sido probados con anterioridad en MaOPs. En específico, se utiliza MOMBI-II, un MOEA que se basa en el indicador R2 para definir un criterio de dominancia y solucionar problemas de muchos objetivos. También, se busca utilizar MOEAs que apliquen estrategias de dominancia diferentes a la dominancia de Pareto. Para este trabajo se utiliza WASF-GA, el cual es un MOEA que hace uso de la función de escalarización por logros como criterio de dominancia. Por tal motivo, este MOEA también es considerado como LLH para resolver DMAOPs.

Tanto MOMBI-II como WASF-GA son algoritmos propuestos para resolver problemas estáticos. Por lo tanto, requieren ser adaptados para poder resolver problemas de optimización dinámicos. Para esto, se requiere la inserción de dos nuevos métodos: un método de detección de cambio y un método de adaptación al cambio. Ambos MOEAs toman en cuenta el número de generaciones transcurridas como su criterio de detección de cambio. Esto quiere decir que cuando un cierto número de generaciones ha transcurrido, se considera que el entorno del problema ha cambiado.

Respecto al método de adaptación al cambio, en el caso de WASF-GA, su versión dinámica, denominada DWASF-GA realiza el mismo procedimiento que DNSGA-II-A, en donde un porcentaje de su población es reemplazado por soluciones generadas aleatoriamente. Para MOMBI-II se utilizan dos versiones dinámicas. La primera, denominada DMOMBI-II-A utiliza el método de detección de cambio usado por DNSGA-

II-A. La segunda versión, denominada DMOMBI-II-B, lleva a cabo su método en base al proceso realizado por DNSGA-II-B, en el cual el subconjunto de soluciones seleccionadas es mutada y reinsertada a la población.

5.3 Conclusiones

Las propuestas presentadas en este capítulo buscan solucionar los problemas planteados para este trabajo de tesis. Para ello, dichas propuestas se enfocan en dos criterios principales dentro de la optimización de problemas: dinamismo y manejo de múltiples objetivos.

Dentro de la literatura existen varias metaheurísticas propuestas que buscan solucionar problemas que contienen una o ambos de los criterios previamente mencionados. Para las propuestas presentadas en este capítulo se busca utilizar tanto DMOEAs previamente desarrollados, específicamente, múltiples versiones de DNSGA-II, diferenciadas en base al método de adaptación al cambio utilizado. Por otra parte, también se propone la adaptación de múltiples MOEAs poder solucionar DMOPs y DMaOPs. Dichos MOEAs son MOCeLL, SPEA2, GDE3, WASF-GA y MOMBI-II. Para transformar estos MOEAs a DMOEAs se les incorpora métodos de detección al cambio y adaptación al cambio.

La metodología propuesta se encuentra representada en la estructura de las hiperheurísticas DHH, DPEM-HH y DHHMa, los cuales se utilizan para resolver problemas de optimización dinámicos y con múltiples objetivos. Dicha metodología busca satisfacer el objetivo general y los objetivos específicos planteados. Primero, se busca desarrollar una técnica que permita solucionar DMOPs y DMaOPs de forma efectiva. También, se busca que dicha metodología sea capaz de trabajar con LLHs complejos, en este caso, con diferentes DMOEAs. Este conjunto de LLHs está conformado por varios de los DMOEAs mencionados previamente en este capítulo.

Parte de los objetivos establecidos mencionan la necesidad de definir los identificadores más adecuados que permitan evaluar el desempeño la metodología propuesta y cada uno de los LLHs utilizados. Por este motivo en este capítulo también se propone el uso de un conjunto de métricas de desempeño, así como de la evolucionabilidad poblacional, para evaluar la capacidad y habilidad de cada uno de los algoritmos utilizados. En base a estudios previos dentro de la literatura, se determinó que dichos indicadores son capaces de proveer suficiente información para llevar a cabo una evaluación adecuada. Esto, debido a que el conjunto de indicadores seleccionados considera tanto la convergencia de las soluciones encontradas por cada algoritmo respecto a un frente de Pareto óptimo, así como su diversidad dentro del espacio de búsqueda de funciones objetivo.

Por otra parte, la variedad de método de selección de LLHs y criterios de aceptación de solución tiene dos objetivos en particular. En primer lugar, se busca identificar patrones en el desempeño de la metodología y la configuración más adecuada para solucionar los DMOPs y DMaOPs considerados. En segundo lugar, se buscan probar los límites y alcances actuales que presenta la metodología propuesta, lo cual permita establecer conclusiones en base a los resultados obtenidos y definir el rumbo del trabajo futuro.

Capítulo 6

Propuesta 2: Método de incorporación de preferencias en DMOPs y DMaOPs

Además de las propuestas presentadas para solucionar problemas multiobjetivo dinámicos, dentro de este trabajo de tesis se hace consideración de las preferencias definidas por un DM. La incorporación de preferencias para problemas de optimización dinámica es un área poco explorada, tal como se muestra en la revisión de los antecedentes y el estado del arte. Por dicho motivo, este trabajo propone llevar a cabo un análisis profundo dentro de esta área. Este análisis genera dos beneficios. Primero, la propuesta de un nuevo método de incorporación de preferencias para utilizar en este tipo de problemas. Segundo, la obtención de nueva información relevante al tema de manejo de preferencias dentro de esta área, que puede ser utilizada como punto de partida para estudios y trabajos futuros.

6.1 Método de Separación por Planos (PS)

Como se mencionó en la Sección 4.1.1, aun cuando existen DMOEAs que incorporan preferencia dentro de su proceso mediante la aplicación de puntos de referencia, la variedad de instancias DMOPs, así como sus diferentes características presentan una limitante importante para los DMOEAs que aplican este enfoque.

Específicamente hablando en la técnica basada en la incorporación de puntos de referencia definidos para un DMOP, se presenta la situación de no ser necesariamente aplicables para otro. A diferencia de una asignación de pesos o un conjunto de relaciones de superación, los valores preferenciales del DM deben cambiarse cada vez que se ataque un problema diferente. Esto obliga a cada una de las partes involucradas en el proceso de solución, tanto DM como el conjunto de expertos que llevan a cabo el modelado de preferencias deben entender en profundidad cada problema a resolver. Lo cual, aunque en teoría suena necesario, puede resultar en una gran inversión de tiempo y esfuerzo cuando se trabaja con múltiples problemas con características similares, pero diferentes funciones objetivo.

En esta sección se presenta un método de incorporación de preferencias que fue propuesto para este trabajo, el cual permite generar puntos de referencia de manera dinámica. Este método es definido como Separación por Planos (PS). PS genera de manera automática un conjunto de puntos de referencia L basándose en un vector de configuración de preferencias W establecido por el DM. Para cada objetivo, el DM asigna un peso máximo y mínimo de preferencia. Utilizando esta configuración y los valores máximos y mínimos encontrados para cada uno de los objetivos por la población, se definen los vértices de un plano ROI. Toda solución que se encuentre dentro de dicho plano se dice que pertenece al ROI. A partir del ROI se generan nuevos planos, los cuales incrementan su tamaño respecto al ROI en base a un porcentaje de expansión R .

En la Figura 6.1 se representa un sistema de tres planos, ROI, así como soluciones cercanas y de media distancia al ROI. Cada plano está conformado de las soluciones que se encuentren dentro de su rango y a su vez fuera del rango de planos de menor tamaño. Por ejemplo, una solución que pertenece al plano ROI, solamente pertenece a ese plano. Por otra parte, una solución que está dentro del rango del plano de soluciones cercanas, pero fuera del ROI pertenece al plano de Cercanas. Las soluciones que no pueden ser asignadas a un plano se definen como Lejanas.

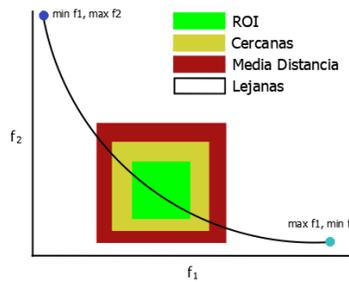


Figura 6.1 Separación por Planos

El proceso de Separación por Planos se describe en el Algoritmo 6.1. Primero, se toma una población unificada Q , que es el resultado del uso de una heurística H en la población actual. Q contiene tanto a la población padre como a la hija. Sea min_i y max_i los valores mínimo y máximo por objetivo i obtenidos por Q , respectivamente. L_{i,min_i} y L_{i,max_i} representan los valores extremos para el objetivo i del ROI obtenido, dichos valores son calculados teniendo en cuenta la distancia entre min_i y max_i , denotada como d , y W .

Después de obtener el ROI, se calculan los valores extremos por objetivo de cada plano siguiente, los cuales son asignados a L . Dichos valores se obtienen tomando en cuenta los valores extremos del ROI y el porcentaje de expansión R correspondiente al objetivo i del plano j . Con los planos generados, Q es separada en subpoblaciones S , dependiendo de su posición en el espacio de búsqueda. Finalmente se inserta a la nueva población NP soluciones de cada plano siguiendo el porcentaje de contribución C establecido para cada plano y el proceso definido por la heurística H utilizada en ese momento.

Como se puede ver, PS es un enfoque que permite la incorporación de preferencias de manera flexible, ya que puede ser insertado en cualquier parte de un algoritmo de optimización multiobjetivo. El único requisito es que debe ser aplicado cuando se presenta la unificación de las poblaciones padre e hija. Después de que PS haya dividido la población unificada en planos, el algoritmo puede generar una nueva población con los individuos más cercanos al ROI siguiendo su correspondiente estrategia, aplicándola de manera aislada para cada uno de los planos. Así, se obtienen los mejores resultados de cada subconjunto.

Dos factores críticos y que deben ser considerados al solucionar DMOPs son la convergencia del frente de Pareto obtenido (POF) respecto al frente de Pareto óptimo (POF^*) y la diversidad de POF en el espacio de búsqueda. Aunque a primera vista pueda parecer que el mantener de soluciones diversas no es un aspecto relevante al solucionar problemas con ROI definido. La falta de diversidad puede llevar a situaciones críticas tales como convergencia prematura o estancamiento en óptimos locales. Debido a lo anterior, es

necesario considerar mantener un cierto grado de diversidad en el frente de Pareto obtenido, incluso cuando hay un ROI definido por un DM.

Algoritmo 6.1. Separación por Planos.

Entrada: R, W, C, P, H

Salida: NP

INICIO

1: $Q \leftarrow \text{GenerarPoblacionUnificada}(H, P)$

2: $\{max, min\} \leftarrow \text{CalcularMaxMin}(Q)$

3: $i \leftarrow 1$

4: **mientras** $i \leq m$

5: $d \leftarrow max_i - min_i$

6: $L_{i,min_i} \leftarrow min_i + (W_{i,min} + d)$

7: $L_{i,max_i} \leftarrow min_i + (W_{i,max} + d)$

8: $i \leftarrow i+1$

9: **fin mientras**

10: $j \leftarrow 2$

11: **mientras** $j \leq |S|$

12: $i \leftarrow 1$

13: **mientras** $i \leq m$

14: $L_{j,min_i} \leftarrow L_{i,min_i} - (L_{i,min_i} * R_{j,i})$

15: $L_{j,max_i} \leftarrow L_{i,max_i} + (L_{i,max_i} * R_{j,i})$

16: $i \leftarrow i+1$

17: **fin mientras**

18: $j \leftarrow j+1$

19: **fin mientras**

20: $S \leftarrow \text{InsertarEnPlano}(L, Q)$

21: $NP \leftarrow \text{GenerarNuevaPoblacion}(S, C, H)$

FIN

Con el objetivo de mantener un equilibrio entre las capacidades de intensificación y exploración de los DMOEAs, PS se activa durante la primera generación y, después, cada vez que un cambio es detectado. En este caso, al utilizarse el número de generaciones transcurridas para determinar dicho cambio, es la variable t_{ps} la cual determina cuando se aplica PS y tiene el mismo valor que τ_r . También, PS utiliza un vector de contribución C , el cual define el máximo porcentaje de soluciones que un plano puede agregar a la nueva población respecto al tamaño máximo de dicha población.

Dentro de este trabajo se plantea el uso de tres versiones de DNSGA-II. Todas tienen el mismo procedimiento y método de detección de cambios. Sin embargo, cada uno tiene un método de adaptación al cambio diferente. DNSGA-II-A-PS, el cual reemplaza un subconjunto de la población padre por nuevas soluciones. DNSGA-II-B-PS toma dicho subconjunto y los modifica mediante mutación. Estas dos versiones fueron propuestas en su trabajo original [Deb et al., 2007]. La tercera versión DNSGA-II-AB-PS, fue propuesta por primera vez en un artículo derivado de esta tesis [Macias-Escobar et al., 2019]. Esta nueva versión reemplaza la mitad del subconjunto por nuevas soluciones y muta la otra mitad. La Figura 6.2 muestra de forma gráfica los métodos de adaptación al cambio utilizados por cada versión de DNSGA-II.

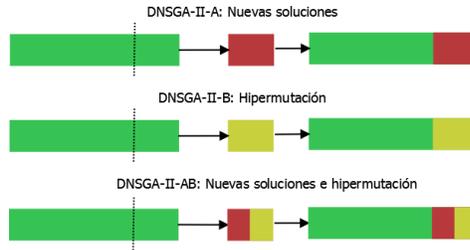


Figura 6.2 Método de adaptación al cambio de las versiones de DNSGA-II.

Para las experimentaciones relacionadas con esta sección se propone la incorporación de PS en DNSGA-II-A-PS y DNSGA-II-AB-PS. La inserción de PS en DNSGA-II significa que se requiere definir por parte del DM los valores para W , R y C . Después de llevar a cabo los procesos evolutivos correspondientes y haber unificado la población padre e hija, PS separa la población unificada en múltiples planos utilizando estos nuevos parámetros. Después, ambas versiones de DNSGA-II-PS llevan a cabo el proceso de generación de nueva población en cada plano, el cual consiste en una clasificación de soluciones no dominadas y *Crowding Distance*. Un subconjunto de las mejores soluciones de cada plano es entonces enviado para formar una nueva población en base al valor del vector C correspondiente a cada plano. El Algoritmo 6.2 muestra el procedimiento de DNSGA-II con PS incorporado. Sean τ_{max} el número de generaciones iteradas, N el tamaño máximo de la población, τ el número de generación actual; cd y ca los porcentajes de población revisada para detección de cambio y modificada por técnica de diversificación, respectivamente, y H el LLH actual.

Algoritmo 6.2. DNSGA-II-PS.

Entrada: $W, R, C, \tau_{max}, \tau, N, cd, ca$

Salida: P_t

INICIO

```

1:  $\tau \leftarrow 1$ 
2:  $P_t \leftarrow \text{GenerarPoblaciónInicial}()$ 
3:  $P_t \leftarrow \text{EvaluarPoblación}(P_t)$ 
4: repetir
5:   si ( $\text{RevisarCambio}(P_t, cd)$ )
6:      $P_t \leftarrow \text{ModificarPoblación}(P_t, ca)$ 
7:      $P_t \leftarrow \text{EvaluarPoblación}(P_t)$ 
8:   fin si
9:    $P_a \leftarrow \text{SeleccionarPadres}(P_t)$ 
10:   $Q \leftarrow \text{Cruza}(P_a)$ 
11:   $Q \leftarrow \text{Mutación}(Q)$ 
12:   $Q \leftarrow \text{EvaluarPoblación}(Q)$ 
13:  si  $\tau = 1$  o  $\tau = \tau_{ps}$ 
14:     $P_{t+1} \leftarrow \text{PS}(R, W, C, Q, H)$ 
15:  sino
16:     $R \leftarrow P_t \cup Q$ 
17:     $F \leftarrow \text{ClasificaciónPorNoDominancia}(R)$ 
18:     $P_{t+1} \leftarrow \emptyset, i \leftarrow 0$ 
19:    mientras ( $|P_{t+1}| < N$ )
20:       $F_i \leftarrow \text{DistanciaCrowding}(F_i)$ 
21:       $P_{t+1} \leftarrow P_{t+1} \cup F_i [1:(N-|P_{t+1}|)]$ 
22:       $i \leftarrow i+1$ 
23:    fin mientras
24:  fin si
25:   $\tau \leftarrow \tau + 1$ 
26: hasta ( $\tau > \tau_{max}$ )

```

FIN

También, se incorpora PS en una versión dinámica del algoritmo de Evolución Diferencial Generalizada 3 (GDE3) [Kukkonen y Lampinen, 2005], denominada DGDE3-PS. Este algoritmo se enfoca en la aplicación del método DE/rand/1/bin en MOPs. Cada solución genera un hijo y dependiendo de la factibilidad y relación de dominancia entre estas dos soluciones se determina si uno o ambos individuos son agregados a la nueva población. Después de llevar a cabo este proceso, si el número de soluciones en la nueva población sobrepasa el límite máximo, GDE3 elimina las soluciones excedentes en base a una clasificación de soluciones no dominadas y *Crowding Distance*.

Para este trabajo, GDE3 se adapta para usarse en DMOPs incorporando un método de detección de cambios y un método de adaptación a cambios. Ambos métodos están basados en la misma estrategia seguida por DNSGA-II-A. Esto quiere decir que la detección de cambio se basa en el número de generaciones transcurridas y la adaptación al cambio consiste en un reemplazo de soluciones por nuevas soluciones generadas aleatoriamente.

Algoritmo 6.3. DGDE3-PS

Entrada: $W, R, C, \tau_{max}, \tau, N, cd, ca$

Salida: P_t

INICIO

```
1:  $\tau \leftarrow 1$ 
2:  $P_t \leftarrow \text{GenerarPoblaciónInicial}()$ 
3:  $P_t \leftarrow \text{EvaluarPoblación}(P_t)$ 
4: repetir
5:   si ( $\text{RevisarCambio}(P_t, cd)$ )
6:      $P_t \leftarrow \text{ModificarPoblación}(P_t, ca)$ 
7:      $P_t \leftarrow \text{EvaluarPoblación}(P_t)$ 
8:   fin si
9:   para  $i = 1$  hasta  $N$ 
10:     $P_a \leftarrow \text{SeleccionarPadres}(P_t)$ 
11:     $u \leftarrow \text{Cruza}(P_a)$ 
12:     $u \leftarrow \text{Mutación}(u)$ 
13:     $u \leftarrow \text{EvaluarSolución}(u)$ 
14:     $P_{t,i} \leftarrow \text{CompararDominancia}(P_{t,i}, u)$ 
15:    si ( $\text{NoDominancia}(P_{t,i}, u)$ )
16:       $P_t \leftarrow P_t \cup u$ 
17:    fin si
18:   fin para
19:   si ( $P_t > N$ )
20:     si  $\tau = 1$  o  $\tau = \tau_{ps}$ 
21:        $P_{t+1} \leftarrow \text{PS}(R, W, C, P_t, H)$ 
22:     sino
23:        $F \leftarrow \text{ClasificaciónPorNoDominancia}(P_t)$ 
24:        $P_{t+1} \leftarrow \emptyset, i \leftarrow 0$ 
25:       mientras ( $|P_{t+1}| < N$ )
26:          $F_i \leftarrow \text{DistanciaCrowding}(F_i)$ 
27:          $P_{t+1} \leftarrow P_{t+1} \cup F_i$  [ $1:(N-|P_{t+1}|)$ ]
28:          $i \leftarrow i+1$ 
29:       fin mientras
30:     fin si
31:      $\tau \leftarrow \tau + 1$ 
32:   fin si
33: hasta ( $\tau > \tau_{max}$ )
```

FIN

PS es aplicado durante el proceso de la comparación entre las soluciones de la población padre e hija para determinar cuáles conformaran la próxima generación. Esto ocurre después de haber llevado a cabo los procesos de selección y cruzamiento de padres, así como la mutación de las soluciones hijas, siguiendo una estructura similar a la de DNSGA-II-PS. De igual manera, se requiere definir W , R y C de forma a priori por parte del DM, la generación de frentes no dominados y *Crowding Distance* se hace de forma aislada en cada plano y el número de soluciones enviadas por cada plano hacia la nueva población depende de C . El Algoritmo 6.3 explica el procedimiento llevado a cabo por DGDE3-PS.

Algoritmo 6.4. DSPEA2-PS.

Entrada: $W, R, C, \tau_{max}, \tau_{ps}, N, N_a, cd, ca$

Salida: P_t

N_a = tamaño máximo del archivo

INICIO

```
1:  $\tau \leftarrow 1$ 
2:  $P_t \leftarrow \text{GenerarPoblaciónInicial}()$ 
3:  $A_t \leftarrow \text{InicializarArchivo}()$ 
4:  $P_t \leftarrow \text{EvaluarPoblación}(P_t)$ 
5: repetir
6:    $A_t \leftarrow \text{EvaluarArchivo}(A_t)$ 
7:   si ( $\text{RevisarCambio}(P_t, cd)$ )
8:      $P_t \leftarrow \text{ModificarPoblación}(P_t, ca)$ 
9:      $P_t \leftarrow \text{EvaluarPoblación}(P_t)$ 
10:     $A_t \leftarrow \emptyset$ 
11:   fin si
12:    $A_{t+1} \leftarrow \text{InsertarSolucionesNoDominadas}(P_t, A_t)$ 
13:   si  $|A_{t+1}| < N_a$ 
14:      $A_{t+1} \leftarrow \text{InsertarSolucionesDominadas}(P_t, A_t)$ 
15:   fin si
16:   si  $|A_{t+1}| > N_a$ 
17:     si  $\tau = 1$  o  $\tau = \tau_{ps}$ 
18:        $A_{t+1} \leftarrow \text{PS}(R, W, C, A_{t+1}, H)$ 
19:     sino
20:        $A_{t+1} \leftarrow \text{TruncarPorAptitud}(A_{t+1})$ 
21:     fin si
22:   fin si
23:    $P_a \leftarrow \text{SeleccionarPadres}(A_{t+1})$ 
24:    $Q \leftarrow \text{Cruza}(P_a)$ 
25:    $Q \leftarrow \text{Mutación}(Q)$ 
26:    $P_t \leftarrow Q$ 
27:    $P_t \leftarrow \text{EvaluarPoblación}(P_t)$ 
28:    $\tau = \tau + 1$ 
29: hasta ( $\tau > \tau_{max}$ )
FIN
```

El algoritmo evolutivo de fuerza de Pareto 2 (SPEA2) es una versión mejorada del algoritmo evolutivo de fuerza de Pareto, el cual incluye un proceso de asignación de aptitud basada en la fuerza de las soluciones que lo dominan (aptitud pura), un método de selección de entorno que permite mantener el *POF* disperso y un proceso de truncamiento de archivo que descarta las soluciones con mayor vecindad.

Para adaptar SPEA2 al dinamismo (DSPEA2) se incorporaron métodos de detección y adaptación al cambio similares a DNSGA-II. El archivo de soluciones no dominadas es descartado cada vez que se genera un cambio. PS es incorporado dentro de SPEA2 justo

después de que el archivo y la población actual se unifican. En ese momento, PS separa las subpoblaciones, las cuales usan de manera aislada el proceso de selección de SPEA2, devolviendo las mejores soluciones de cada frente para generar la nueva población. El Algoritmo 6.4 muestra a detalle la incorporación de PS dentro de este DMOEA. De igual forma que los otros DMOEAs previos, DSPEA2-PS requiere una definición de W , R y C .

6.2 Incorporación de PS a hiperheurísticas

Así como se propone en la Sección 6.1 la incorporación de PS dentro del proceso de múltiples DMOEAs, este trabajo también propone que esta nueva técnica de incorporación de preferencias puede ser utilizadas dentro de hiperheurísticas. Por este motivo se proponen múltiples hiperheurísticas, las cuales incluyen PS dentro de su proceso con el objetivo de desarrollar metodologías capaces de solucionar DMOPs con preferencias definidas.

La primera hiperheurística propuesta es una variante de la Hiperheurística Multiobjetivo Dinámica (DHH), la cual utiliza Separación por Planos (DHH-PS). DHH-PS busca utilizar múltiples DMOEAs con PS incorporado como heurísticas de bajo nivel.

El Algoritmo 6.5 presenta el proceso realizado por DHH-PS. Primero, cada heurística LLH_i es ejecutada un número de generaciones de prueba τ_{test} sobre una copia de la población actual. Para cada LLH_i , PS es aplicado en la primera generación de prueba y cada τ_{ps} generaciones. Si la condición anterior no se cumple, entonces el método de reemplazo de población definido de cada heurística es aplicado en Q . La última población obtenida por cada LLH es evaluada y el conjunto de métricas MET es utilizado conforme el método de selección de heurísticas en uso para definir cuál LLH es seleccionado. Después, la última población del LLH seleccionado se define como la población actual. Finalmente, este LLH es usado en la población (utilizando PS o su método de reemplazo de población en base a τ_{ps}) hasta el final del periodo.

DPEM-HH es una hiperheurística que se apoya de evolucionabilidad poblacional para llevar a cabo el proceso de selección de LLHs. La incorporación de PS en DPEM-HH se presenta en el Algoritmo 6.6. Sea $P_{evp,i}$ la población generada por el LLH_i , evp_i la evolucionabilidad poblacional calculada para cada LLH_i después de ejecutarse por τ_{test} generaciones de prueba y MET el conjunto de valores correspondientes a las métricas de desempeño utilizadas para el proceso de selección de heurísticas. DPEM-HH lleva a cabo el mismo proceso que se definió en el periodo anterior, con la diferencia de que cada LLH ejecuta PS dentro de su proceso. De igual manera de DHH-PS, al final de proceso de selección se toma la última población del LLH elegido y se utiliza como población actual. El LLH seleccionado se sigue ejecutando (utilizando PS o su método de reemplazo de población en base a τ_{ps}) hasta el próximo periodo.

Para DHH-PS y DPEM-HH-PS se propone el uso de varios métodos de selección de LLHs y criterios de aceptación de solución, con el objetivo de determinar cuál configuración de método y criterio presentan los mejores resultados para un cierto conjunto de DMOPs. Varios de los métodos de selección utilizados para los experimentos de esta sección se basan

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

en aquellos presentados en la Sección 5.1. Además, se plantea el uso de dos métodos de selección adicionales:

Algoritmo 6.5. DHH-PS.

Entrada: τ_{ps} , τ_{test} , LLH , R , W , C

Salida: P

INICIO

```
1:  $P \leftarrow \text{GenerarPoblacionInicial}()$ 
2: mientras no se alcance criterio de paro
3:   si se detecta cambio o primera generación
4:     para  $i = 1$  hasta  $|LLH|$ 
5:        $P_{copy,i} \leftarrow \text{CopiarPoblacion}(P)$ 
6:       para  $j = 1$  hasta  $\tau_{test}$ 
7:          $Q \leftarrow \text{CrearPoblacionUnida}(LLH_i, P_{copy,i})$ 
8:         si  $j = 1$  o  $j = \tau_{ps}$ 
9:            $P_{copy,i} \leftarrow \text{PS}(R, W, C, Q, LLH_i)$ 
10:        sino
11:           $P_{copy,i} \leftarrow \text{Reemplazo}(Q, LLH_i)$ 
12:        fin si
13:      fin para
14:    fin para
15:     $MET \leftarrow \text{EvaluacionMetricas}(P_{copy})$ 
16:     $\{P_{copy,seleccionado}, LLH_{seleccionado}\} \leftarrow \text{SeleccionarLLH}(LLH, MET, P_{copy})$ 
17:     $P \leftarrow P_{copy,seleccionado}$ 
18:  sino
19:    si  $i \bmod \tau_{ps} = 0$ 
20:       $Q \leftarrow \text{CrearPoblacionUnida}(P, LLH_{seleccionado})$ 
21:       $P \leftarrow \text{PS}(R, W, C, Q, LLH_{seleccionado})$ 
22:    sino
23:       $P \leftarrow \text{EjecutarLLH}(P, LLH_{seleccionado})$ 
24:    fin si
25:  fin si
26: fin mientras
FIN
```

Voraz con Ranking (GR). Basado en la función de elección mostrada en las ecuaciones (3.23) y (3.24). Sin embargo, solamente se utiliza los resultados obtenidos en c_1 por cada LLH para llevar a cabo la selección, eligiendo la heurística con el mayor valor. Este método, en comparación del voraz presentado en la Sección 5.1, utiliza múltiples métricas de desempeño para llevar a cabo el proceso de selección.

Ruleta (R). Sigue la misma estrategia establecida por Algoritmo 5.2. Pero, a diferencia de la ruleta con memoria, este método elige el primer LLH que haya sido seleccionado. Este método de selección sigue la estructura tradicional del método de ruleta.

De igual forma, también se propone el análisis del criterio de aceptación “*All Moves*”, presentado en la Sección 5.1, junto con un criterio presentado para las experimentaciones de correspondientes a esta sección denominada Distancia σ ($D(\sigma)$).

Se considera el punto utópico μ , nadir φ y centro γ de una población obtenida, tal como se muestra en la Figura 6.3. A partir de ellos se define una solución σ , la cual corresponde a la solución más cercana a μ . La ecuación (6.1) define como calcular distancia σ ($D(\sigma)$). Δ representa la distancia euclidiana entre dos puntos.

$$D(\sigma) = \frac{\Delta(\mu, \sigma) + \Delta(\sigma, \gamma)}{\Delta(\mu, \sigma) + \Delta(\sigma, \varphi)} \quad (6.1)$$

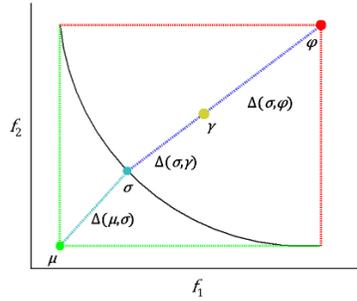


Figura 6.3 Distancia σ .

Algoritmo 6.6. DPEM-HH-PS.

Entrada: τ_{ps} , τ_{test} , LLH , R , W , C

Salida: P

INICIO

```

1:  $P \leftarrow$  GenerarPoblacionInicial()
2: mientras no se alcance criterio de paro
3:   si se detecta cambio o primera generación
4:     para  $i = 1$  hasta  $|LLH|$ 
5:        $P_{evp,i} \leftarrow$  CopiarPoblacion( $P$ )
6:       para  $j = 1$  hasta  $\tau_{test}$ 
7:          $Q \leftarrow$  CrearPoblacionUnida( $LLH_i$ ,  $P_{evp,i}$ )
8:         si  $j = 1$  o  $j = \tau_{ps}$ 
9:            $P_{evp,i} \leftarrow$  PS( $R$ ,  $W$ ,  $C$ ,  $Q$ ,  $llh_i$ )
10:        sino
11:           $P_{evp,i} \leftarrow$  Reemplazo( $Q$ ,  $llh_i$ )
12:        fin si
13:         $NP \leftarrow$  CrearVecindad( $llh_i$ ,  $P_{evp,i}$ )
14:         $evp_i \leftarrow evp_i +$  CalcularEvolucionabilidad( $P_{evp,i}$ ,  $NP$ )
15:      fin para
16:       $evp_i \leftarrow evp_i / \tau_{test}$ 
17:    fin para
18:     $MET \leftarrow$  EvaluacionMetricas( $P_{evp}$ )
19:     $LLH_{seleccionado} \leftarrow$  SeleccionarLLH( $LLH$ ,  $evp$ ,  $MET$ )
20:     $P \leftarrow$  AsignarPoblacion( $LLH_{seleccionado}$ ,  $P_{evp,seleccionado}$ )
21:  sino
22:    si  $i \bmod \tau_{ps} = 0$ 
23:       $Q \leftarrow$  CrearPoblacionUnida( $P$ ,  $LLH_{seleccionado}$ )
24:       $P \leftarrow$  PS( $R$ ,  $W$ ,  $C$ ,  $Q$ ,  $llh_{seleccionado}$ )
25:    sino
26:       $P \leftarrow$  EjecutarLLH( $P$ ,  $LLH_{seleccionado}$ )
27:    fin si
28:  fin si
29: fin mientras
FIN

```

Para esta sección se analizan dos diferentes criterios para aceptar la nueva población basándose en los resultados obtenidos por $D(\sigma)$:

- 1) Si $D(\sigma_{t+1}) > 0.95D(\sigma_t)$, se acepta la nueva población. ($D(\sigma)$).
- 2) Igual que 1). Si no se cumple, pero $D(\sigma_{t+1}) \geq 0.9D(\sigma_t)$, entonces se define una probabilidad en base a la ecuación (6.2). Sea $rand$ un valor aleatorio entre 0 y 1, se acepta la nueva población si acc es mayor o igual que $rand$. ($D(\sigma_{rand})$).

$$acc = 2 * \frac{D(\sigma_{t+1}) - 0.9D(\sigma_t)}{0.1D(\sigma_t)}. \quad (6.2)$$

6.3 Incorporación de PS en hiperheurísticas para muchos objetivos

Después de definir los cambios en la hiperheurísticas y sus LLHs para DMAOPs, también es necesario determinar la técnica de incorporación de preferencias a utilizar para manejar este tipo de problemas bajo un conjunto de preferencias definidas por un DM.

Considerando los resultados obtenidos en los experimentos que se presentan en la Sección 7.4, basadas en las propuestas en secciones anteriores dentro de este capítulo, es posible considerar el uso de PS para resolver este tipo de instancias. El método de Separación por Planos presenta una ventaja en el proceso de incorporación de preferencias al resolver problemas de muchos objetivos ya que no está limitado por el número de objetivos que presenta una instancia.

El único requisito que necesita cumplirse para poder utilizar PS en cualquier problema de optimización dinámico es definir los valores de los vectores de configuración preferencial W , porcentaje de expansión de planos R y el porcentaje de contribución por plano C . Específicamente, en W solamente es necesario determinar de forma obligatoria un valor específico para uno de los objetivos.

Por ejemplo, el valor de W_{f_1} puede determinarse en [0.1-0.4], mientras que el resto de los objetivos pueden determinarse con un rango completo [0.0-1.0] si se considera una configuración con valores normalizados. Con haber definido el valor preferencial de f_1 es suficiente para generar un ROI dentro del espacio de búsqueda de funciones objetivo. Obviamente, mientras más específicas sean las definiciones preferenciales para cada objetivo, más específica será la ubicación del ROI dentro del espacio de búsqueda. Por lo tanto, mientras más específicas sean las preferencias del DM, más estricto será el algoritmo que utiliza PS al momento de buscar soluciones que satisfagan dichas preferencias.

Las métricas y funciones de escalarización utilizadas en los LLHs y DHHMa, vistas en la sección anterior, pueden adaptarse a PS de la siguiente manera:

LRNI: La población P consiste en soluciones dentro del plano ROI, el cual es el primer plano generado por PS que representa las preferencias del DM. Por lo tanto, las soluciones del conjunto LND serán aquellas soluciones no L-dominadas del ROI.

R2 y ASF: En base a las propiedades de la distancia σ , se pueden presentar variaciones de $R2$ y ASF que son compatibles con PS. El punto de referencia z es el punto central del plano ROI. Además, los valores de las funciones objetivo son normalizadas cuando se evalué tanto de $R2$ y ASF , apoyándose en estrategias de normalización [Ruiz et al., 2015]. μ y φ son los puntos utópicos y nadir del plano evaluado en ese momento, respectivamente.

Por ejemplo, en el caso de la Figura 6.4, si se está trabajando con las soluciones del plano azul, se toman sus soluciones utópica y nadir para normalizar las soluciones. Cada solución de este plano es evaluada en base a su distancia respecto al punto utópico del plano ROI.

HV: El cálculo de hipervolumen se hace únicamente utilizando soluciones que pertenezcan al plano ROI, de igual forma que las evaluaciones de hipervolumen en problemas sin preferencias definidas, se utiliza como punto de referencia el punto origen ($\vec{0}$).

Para esta propuesta se hace un énfasis en la aplicación de funciones de escalarización, en especial en *ASF*, la cual se propone incorporar dentro de PS. Para eso, se utiliza la estrategia presentada previamente y apoyada por la Figura 6.4 para cada uno de los planos. La intención es combinar la capacidad de explotación de *ASF* con la capacidad de exploración de PS, al permitir revisar incluso soluciones lejanas al ROI.

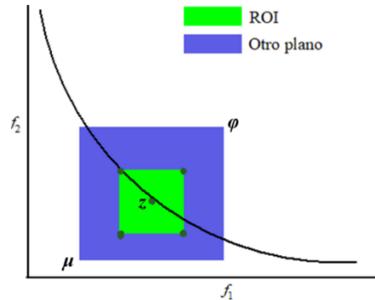


Figura 6.4 Definición de puntos para R2 y ASF usando Separación por Planos.

El Algoritmo 6.7 muestra cómo se usa la función de escalarización *ASF* dentro de PS. Como puede verse, la obtención de planos sigue la misma estrategia utilizada para resolver problemas con menos de cuatro objetivos. Esta estrategia se basa en los rangos que representan las preferencias del DM y la población padre e hija generadas por la heurística y unificadas en un solo conjunto.

PS inicia realizando el mismo proceso que el definido en el Algoritmo 6.1. Se determinan los valores máximos y mínimos por objetivo. Después, define los valores máximos y mínimo de cada objetivo por plano en L usando los valores objetivos extremos, las preferencias del DM y un criterio de expansión respecto al ROI. En base a L , se inserta cada solución a su respectivo plano.

El primer cambio buscando adaptar PS a DMaOP consiste en generar un punto de referencia z , el cual será el punto central del plano ROI (línea 21). El segundo cambio consiste en utilizar las propiedades de la heurística H y z para generar la nueva población usando cada uno de los S planos en base a un porcentaje de contribución C (línea 22). Cada plano selecciona las $N \cdot C_i$ soluciones en base a la estrategia de reemplazo de soluciones para generar la población de la siguiente generación de cada LLH para formar la nueva población NP . Sea N el tamaño máximo de la población y C_i el porcentaje de contribución del plano i .

De igual forma que para las hiperheurísticas enfocadas en problemas de hasta tres objetivos. PS es incorporado dentro de los procesos de DHHMa siguiendo la misma estructura definida para DHH en el Algoritmo 6.4. Esta versión de DHHMa con PS es denominada como DHHMa-PS. También, PS es insertado en cada uno de los LLHs utilizados por DHHMa mencionados en el capítulo 5. Tanto DWASF-GA-PS y DMOMBI-II-PS aplican PS al momento de llevar a cabo el proceso de unificación de poblaciones padre e hija

y comparación de soluciones para generar una nueva población, siguiendo la misma estructura de los LLHs utilizados en DHH-PS.

Algoritmo 6.7. Separación por Planos aplicada en DMaOPs (PS)

Entrada: R, W, C, P, H

Salida: NP

INICIO

```
1:  $Q \leftarrow \text{GenerarPoblacionUnificada}(H, P)$ 
2:  $\{max, min\} \leftarrow \text{CalcularMaxMin}(Q)$ 
3:  $i \leftarrow 1$ 
4: mientras  $i \leq m$ 
5:    $d \leftarrow max_i - min_i$ 
6:    $L_{1,min_i} \leftarrow min_i + (W_{i,min} + d)$ 
7:    $L_{1,max_i} \leftarrow min_i + (W_{i,max} + d)$ 
8:    $i \leftarrow i+1$ 
9: fin mientras
10:  $j \leftarrow 2$ 
11: mientras  $j < |S|$ 
12:    $i \leftarrow 1$ 
13:   mientras  $i \leq m$ 
14:      $L_{j,min_i} \leftarrow L_{1,min_i} - (L_{1,min_i} * R_{j,i})$ 
15:      $L_{j,max_i} \leftarrow L_{1,max_i} + (L_{1,max_i} * R_{j,i})$ 
16:      $i \leftarrow i+1$ 
17:   fin mientras
18:    $j \leftarrow j+1$ 
19: fin mientras
20:  $S \leftarrow \text{InsertarEnPlano}(L, Q)$ 
21:  $z \leftarrow \text{GenerarPuntoReferencia}(S_1)$ 
22:  $NP \leftarrow \text{GenerarNuevaPoblacion}(S, C, H, z)$ 
```

FIN

6.4 Conclusiones

Un área relevante a investigación y considerar relacionada a los problemas de optimización es el manejo e incorporación de preferencias establecidas por un DM. Como se mostró en capítulos anteriores, existe una amplia cantidad de propuestas y una clasificación establecida respecto a la incorporación de preferencias es problemas estáticos. Sin embargo, su uso en problemas dinámicos sigue siendo un área con poca exploración.

La propuesta presentada en este capítulo se enfoca en el área de incorporación de preferencias de un DM para DMOPs y DMaOPs. Esto, permitirá no solamente establecer, hasta el conocimiento del autor, el primer uso de estas técnicas en hiperheurísticas para resolver problemas dinámicos. También, permitirá generar nueva información que establezca un punto de partida para los investigadores en cualquier trabajo futuro relacionado a esta área. Por dicho motivo, puede decirse el nuevo conocimiento proporcionado por esta investigación tiene el mismo valor que la propuesta misma.

Una de las principales intenciones de la propuesta establecida en este capítulo es expandir el estudio realizado por las propuestas desarrolladas en el capítulo anterior. Esto quiere decir, que, además de hacer enfoque en el dinamismo y manejo de múltiples objetivos, también se busca que los algoritmos propuestos sean capaces de solucionar instancias que contengan preferencias determinadas por un DM.

Para esto, se plantea una nueva técnica de incorporación de preferencias, denominada Separación por Planos (PS). Esta técnica se basa en un conjunto de preferencias determinadas *a priori* para enfocar la búsqueda de soluciones hacia la región de interés dentro del espacio de búsqueda de la función objetivo. En esta propuesta PS es implementado dentro de los procesos de DHH, DPEM-HH y DHHMa, así como diversos DMOEAs propuestos en el estado del arte y presentados en el capítulo anterior. como es el caso de DNSGA-II-PS, DSPEA2-PS, DGDE3-PS, DWASF-GA-PS y DMOMBI-II-PS.

Capítulo 7

Experimentación y análisis

En este capítulo se muestran los diseños, resultados, análisis y conclusiones de todas las experimentaciones realizadas durante el transcurso de esta tesis. Cada una de las propuestas realizadas en los capítulos anteriores son evaluadas y comparadas respecto tanto a metaheurísticas pertenecientes al estado del arte como aquellas propuestas dentro de la literatura que tienen mayor relevancia dentro del campo de optimización dinámica, incorporación de preferencias y solución de problemas dinámicos con muchos objetivos. Las secciones de este capítulo están divididas de forma que se pueden revisar cada uno de los planteamientos de problema realizados de forma separada. Esto quiere decir que para cada problema planteado existe una hiperheurística propuesta que fue aplicada en instancias que reflejan dicho tipo de problema.

7.1 Comparativa de DHH respecto a DMOEAs

Como se mencionó anteriormente, esta tesis tiene como objetivo el desarrollo de una metodología de solución de problemas dinámicos. En esta sección se busca analizar la factibilidad de dicho objetivo mediante una serie de pruebas utilizando un conjunto limitado de DMOPs y DMOEAs. Con el objetivo de analizar la factibilidad del proceso de caracterización de algoritmos y DMOPs bajo métricas de desempeño, se llevó a cabo un estudio en el cual se buscó implementar DMOEAs que solucionaran DMOPs, evaluando la calidad de sus soluciones mediante métricas de desempeño.

Después, estos DMOEAs se incorporan el de LLHs disponibles para DHH, la cual se utilizará para resolver los mismos problemas y comparar los resultados contra el desempeño de cada LLH al ejecutarse de manera independiente. Para analizar la factibilidad de la aplicación de una metodología de alto nivel para solucionar un problema dinámico, se llevó a cabo el desarrollo e implementación de DHH para resolver DMOPs de tipo I y III.

Como se mencionó en la sección 5.1, DHH está basado en la estructura propuesta en HH_CF y adaptada para resolver problemas dinámicos. Con el objetivo de adaptar la hiperheurística a un entorno dinámico se utilizaron métricas que han sido aplicadas previamente para evaluar DMOEAs aplicados en DMOPs.

En la propuesta de HH_CF, no se hacía consideración del frente de Pareto óptimo (POF^*), sin embargo, al conocerse el POF^* del problema tratado, se decidió utilizar métricas que involucraban dicho frente. Las métricas seleccionadas fueron distancia generacional invertida (IGD), proporción de hipervolumen (HVR), propagación máxima (MS) y una proporción de soluciones no dominadas con respecto al tamaño de la población (RNI).

Para el conjunto de LLHs, se implementaron diversas variaciones del algoritmo DNSGA-II. Estas variaciones constan de la técnica utilizada en el momento de reemplazo de población, ya sea reemplazo de un porcentaje de individuos de la población por nuevas

soluciones aleatorias (DNSGA-II-A) o mutación de un subconjunto de individuos de la población (DNSGA-II-B) [Deb et al., 2007]. Además, se definió una nueva variación que aplica ambas técnicas dentro de un porcentaje de la población, reemplazando por soluciones aleatorias la mitad del subconjunto y mutando la otra mitad (DNSGA-II-AB).

Para esta experimentación tanto DHH como las tres versiones de DNSGA-II se utilizaron para resolver las instancias FDA1, FDA3, DMZDT1, DMZDT2, DMZDT3 y DMZDT4. Las características de los algoritmos utilizados, así como de ambos DMOPs son los siguientes:

- Datos de DHH:
 - LLHs: DNSGA-II-A, DNSGA-II-B, DNSGA-II-AB.
 - Población = 100.
 - Clasificación: Selección de heurísticas de perturbación/Aprendizaje en línea.
 - Método de selección de LLH: Función de elección, $\alpha = 1000$.
 - Métricas de desempeño: RNI, IGD, HVR y MS.
 - Criterio de aceptación de soluciones: *All Moves*.
 - Método de detección de cambio: Número de generaciones transcurridas.
 - Método de adaptación al cambio:
 - Generación de soluciones aleatorias (DNSGA-II-A, DNSGA-II-AB).
 - Hipermutación (DNSGA-II-B, DNSGA-II-AB).
- Datos de los DMOEAs:
 - Cruza: Cruza binaria simulada al 90%, índice de distribución = 20.
 - Mutación: Mutación polinomial a 0.05%, índice de distribución = 20.
 - Porcentaje de población modificada por periodo = 20%.
- Datos de los DMOPs:
 - Tamaño del vector de variables de decisión.
 - FDA1: 20; FDA3: 30.
 - DMZDT1, DMZDT2, DMZDT3: 30; DMZDT4: 10.
 - Severidad de cambio (n_t) = 10.
 - Frecuencia de cambio (τ_t) = 25.
 - Generaciones totales = $n_t * \tau_t = 250$.
 - Clasificación de DMOPs: Tipo I.

A partir de lo anterior se puede definir relaciones entre un problema, un conjunto de DMOEAs que resuelven dicho problema, un conjunto de métricas de desempeño de dichos algoritmos, así como los aspectos evaluados por esas métricas. Estas relaciones se muestran en la Tabla 7.1.

Después, se reúnen los algoritmos seleccionados dentro de una hiperheurística, en este caso DHH, buscando automatizar el proceso de selección de algoritmos para solucionar el DMOP en cuestión en base a la relación definida. De esta manera se sigue lo mencionado dentro de los objetivos y el planteamiento del problema.

Tabla 7.1. Relación problema-algoritmo-métrica-aspecto.

Problema	Algoritmo	Métrica	Aspecto
FDA1	DNSGA-II-A DNSGA-II-B DNSGA-II-AB	RNI	Diversidad
FDA3		IGD	Exactitud
DMZDT1		MS	Diversidad
DMZDT2		HV-Ratio	Exactitud y diversidad
DMZDT3			
DMZDT4			

7.1.1 Análisis de los resultados obtenidos

Se llevaron a cabo 30 ejecuciones de DHH y de cada uno de los LLH de forma independiente. Las Tablas 7.2 a 7.5 se presentan los resultados fuera de línea obtenidos en base a la mediana y desviación estándar para cada una de las métricas usadas para evaluar los resultados obtenidos (RNI, IGD, MS y HVR), de forma respectiva. La Figura 7.1 muestra los *POFs* obtenidos por cada uno de estos algoritmos en DMZDT2.

Tabla 7.2. Mediana y desviación estándar fuera de línea para RNI.

Problema	DNSGA-II-A	DNSGA-II-B	DNSGA-II-AB	DHH
FDA1	7.615E+1(2.058E+0)	7.795E+1(2.466E+0)	7.675E+1(2.418E+0)	8.015E+1(2.446E+0)
FDA3	9.120E+1(4.727E+0)	9.265E+1(5.921E+0)	9.080E+1(3.985E+0)	9.240E+1(3.675E+0)
DMZDT1	9.874E+1(1.490E-1)	9.875E+1(1.436E-1)	9.876E+1(1.453E-1)	9.881E+1(1.294E-1)
DMZDT2	8.754E+1(1.093E+0)	8.790E+1(1.036E+0)	8.796E+1(1.022E+0)	8.936E+1(1.085E+0)
DMZDT3	9.904E+1(1.404E-1)	9.911E+1(9.395E-2)	9.908E+1(1.008E-1)	9.901E+1(9.995E-2)
DMZDT4	5.636E+1(2.993E+0)	5.677E+1(3.957E+0)	5.760E+1(3.976E+0)	5.917E+1(3.931E+0)

Tabla 7.3. Mediana y desviación estándar fuera de línea para IGD.

Problema	DNSGA-II-A	DNSGA-II-B	DNSGA-II-AB	DHH
FDA1	4.769E-3(2.120E-3)	4.429E-3(1.919E-3)	4.658E-3(2.804E-3)	4.254E-3(1.690E-3)
FDA3	1.235E-2(9.520E-4)	1.240E-2(1.384E-3)	1.198E-2(1.084E-3)	1.316E-2(1.587E-3)
DMZDT1	4.882E-3(1.420E-4)	4.833E-3(1.830E-4)	4.861E-3(1.610E-4)	4.745E-3(8.800E-5)
DMZDT2	1.003E-2(1.261E-3)	9.932E-3(1.339E-3)	9.901E-3(1.104E-3)	9.349E-3(1.144E-3)
DMZDT3	2.282E-2(1.090E-4)	2.285E-2(5.500E-5)	2.282E-2(1.050E-4)	2.285E-2(2.600E-5)
DMZDT4	3.009E-2(1.269E-2)	3.555E-2(2.151E-2)	3.460E-2(1.854E-2)	2.539E-2(1.828E-2)

Tabla 7.4. Mediana y desviación estándar fuera de línea para MS.

Problema	DNSGA-II-A	DNSGA-II-B	DNSGA-II-AB	DHH
FDA1	9.408E-1(3.864E-2)	9.517E-1(4.387E-2)	9.454E-1 0.054377	9.508E-1(3.189E-2)
FDA3	9.947E-1(3.002E-3)	9.951E-1(1.511E-2)	9.956E-1(4.420E-3)	9.957E-1(2.786E-3)
DMZDT1	9.691E-1(3.041E-3)	9.692E-1(3.825E-3)	9.690E-1(3.693E-3)	9.699E-1(2.304E-3)
DMZDT2	8.972E-1(1.452E-2)	8.984E-1(1.466E-2)	8.997E-1(1.3181E-2)	9.089E-1(1.327E-2)
DMZDT3	9.106E-1(1.431E-2)	9.110E-1(8.060E-3)	9.108E-1(1.452E-2)	9.106E-1(1.168E-3)
DMZDT4	8.222E-1(4.425E-2)	8.291E-1(4.865E-2)	8.186E-1(4.649E-2)	8.486E-1(3.688E-2)

Tabla 7.5. Mediana y desviación estándar fuera de línea para HVR.

Problema	DNSGA-II-A	DNSGA-II-B	DNSGA-II-AB	DHH
FDA1	9.028E-1(1.126E-2)	9.065E-1(9.023E-2)	9.016E-1(1.321E-2)	9.123E-1(1.027E-2)
FDA3	7.289E-1(2.043E-2)	7.305E-1(3.159E-2)	7.365E-1(2.384E-2)	7.099E-1(3.276E-2)
DMZDT1	8.982E-1(1.406E-3)	8.988E-1(1.508E-3)	8.985E-1(1.087E-3)	9.016E-1(1.061E-3)
DMZDT2	7.091E-1(1.041E-2)	7.126E-1(1.062E-2)	7.115E-1(8.866E-3)	7.207E-1(9.305E-3)
DMZDT3	9.331E-1(3.382E-3)	9.343E-1(2.419E-3)	9.339E-1(3.537E-3)	9.339E-1(9.030E-4)
DMZDT4	5.624E-1(1.580E-1)	5.415E-1(1.885E-1)	5.046E-1(1.938E-1)	6.305E-1(2.131E-1)

Los resultados de la Tabla 7.2 muestran que DHH es capaz de encontrar un mayor número de soluciones óptimas en relación con el resto de los LLHs tanto para la mayoría de

las instancias, a excepción de FDA3 y DMZDT3. Esta mejora puede deberse a que se apoya de múltiples métodos de adaptación al cambio en vez de únicamente uno como manejan cada una de las versiones de DNSGA-II aplicadas de forma independiente.

Analizando la capacidad de convergencia de los algoritmos evaluados, es posible ver que DHH presenta mejora tanto en IGD como en HVR para la mayoría de las instancias, siendo nuevamente FDA3 y DMZDT3 las excepciones, donde solo logra mejorar el resultado en comparación de un subconjunto de LLHs. En base a dichas métricas se puede suponer que la hiperheurística es capaz de encontrar soluciones que se encuentran más cerca del POF^* en relación con las soluciones obtenidas por los LLH. Esto se apega al concepto de que una hiperheurística puede aprovechar las fortalezas de sus LLHs para cubrir debilidades.

Respecto a diversidad. DHH presenta mejores resultados para la mayoría de las instancias en comparación con los demás LLHs para IGD y MS. La hiperheurística únicamente presenta menores resultados para DMZDT3 para ambas métricas. Sin embargo, el resultado general tiende a favorecer a la hiperheurística para las demás instancias. Estos valores pueden suponer la idea de que DHH no solamente es capaz de encontrar soluciones cercanas al óptimo, sino que también, aprovechando cada uno de los LLH es capaz de encontrar soluciones con una mejor diversidad.

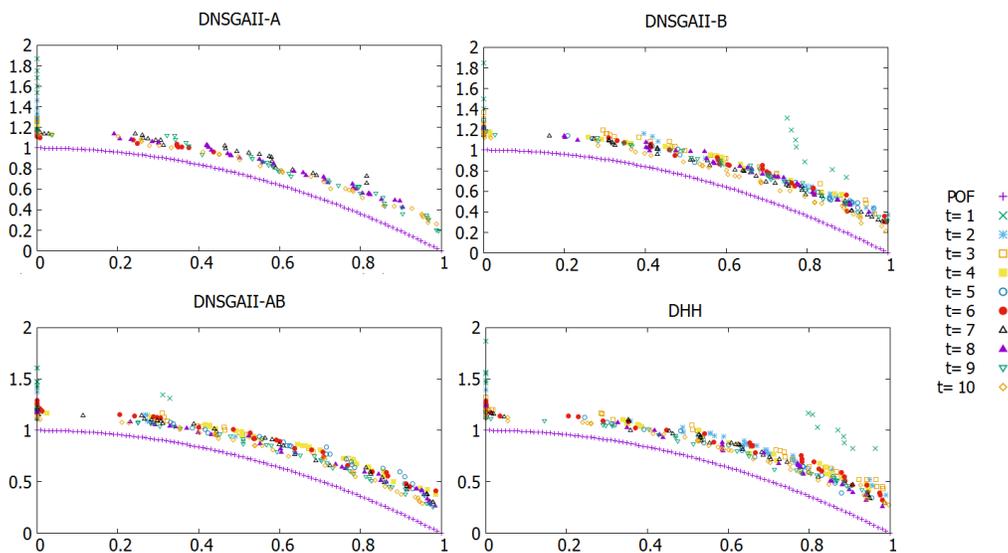


Figura 7.1. Frente óptimo de DMZDT2 y frentes obtenidos por DHH y variaciones de DNSGA-II.

7.1.2 Conclusiones

Esta primera prueba tiene el objetivo de mostrar que el uso de hiperheurísticas en el área de optimización de problemas dinámicos es factible. El diseño de los DMOEAs utilizados permite su fácil incorporación en esta clase de metodologías para su uso en la solución de DMOPs. Además, la estructura de cada LLH no se ve alterada respecto a su diseño original, ya que todos los procedimientos adicionales en este caso recaen en DHH.

A partir de los resultados obtenidos, se puede creer que es factible la aplicación de esta metodología de alto nivel para solucionar estos dos DMOP, obteniendo soluciones

satisfactorias comparables, y en algunos casos mejores, a las obtenidas por los LLHs ejecutados de manera individual.

7.2 Análisis de método de selección y criterios de aceptación en DPEM-HH

En esta experimentación se llevó a cabo un análisis de DPEM-HH con el objetivo de revisar y comparar los efectos de diferentes combinaciones de métodos de selección y criterios de aceptación de soluciones implementados dentro de la hiperheurística. El procedimiento consiste en utilizar un mismo conjunto de DMOEAs como LLHs para resolver un conjunto de DMOPs. Con esto, se busca encontrar la configuración de la metodología más adecuada para cada uno de los problemas, así como para todo el conjunto.

La intención de DPEM-HH es ser capaz de incorporar DMOEAs, métodos FLA y métricas de desempeño en su metodología para solucionar DMOPs. Para el conjunto de LLHs se utilizaron DNSGA-II-A y variantes dinámicas de MOCeCell [Nebro et al., 2009] y SPEA2 [Zitzler et al., 2001]. Para adaptar estos últimos dos DMOEAs se siguieron las recomendaciones establecidas por los autores del marco de referencia jMetal. Este marco propone como un posible método de respuesta al cambio el reemplazar un cierto porcentaje de la población por soluciones generadas aleatoriamente y completo vaciado del archivo utilizado por estos dos DMOEAs.

La evolucionabilidad poblacional fue insertada en varios métodos de selección de LLH (voraz, función de elección y ruleta). Las métricas de evaluación utilizadas tanto por la función de elección, los criterios de aceptación, así como para evaluar las soluciones obtenidas por las diferentes configuraciones de DPEM-HH en esta experimentación fueron RNI, IGD, MS y HVR. Estas métricas han sido antes aplicadas dentro de la optimización dinámica. Por lo que pueden ser utilizadas.

Once instancias de prueba fueron usadas para este experimento con el objetivo de evaluar el desempeño de las diferentes configuraciones de DPEM-HH propuestas. FDA1, FDA3, FDA4 y FDA5 del conjunto FDA, el conjunto DMZDT y el conjunto dMOP. Todos estos problemas son de dos objetivos, salvo FDA4 y FDA5, los cuales pueden manejar un número ilimitado de objetivos (dos o más objetivos). Para esta experimentación, FDA4 y FDA5 manejan tres objetivos, proporcionando así otro reto para analizar, al manejar problemas de más de dos objetivos.

Para cada uno de estos problemas se definió la instancia de tiempo $t = (1/n_t)\lfloor \tau/\tau_t \rfloor$ para los problemas FDA y dMOP y $t = \lfloor \tau/\tau_t \rfloor$ para los problemas DMZDT, donde n_t representa la severidad de cambio, τ_t la frecuencia de cambio y τ la generación actual. La selección de estas instancias se debe a la relevancia de FDA y dMOP dentro del campo, así como la poca exploración existente respecto a DMZDT. También, se siguieron las recomendaciones establecidas por los autores de dichos DMOPs y la guía previamente utilizada [Helbig y Engelbrecht, 2014] para la selección de DMOPs y la asignación de sus valores.

Para todas las instancias de prueba se definió un tamaño de población de 100 individuos en para cada uno de los LLHs en todas las configuraciones de DPEM-HH. El tamaño del archivo utilizado por DMOCeCell y DSPEA2 se definió en 100, Los tres DMOEAs utilizados como LLH utilizan cruce binaria simulada como operador de cruce y distribución polinomial como operador de mutación [Deb y Agrawal, 1995]. Las probabilidades de estos operadores se definieron en 0.9 y $1/n$, donde n es el número de variables del DMOP, respectivamente. Los índices de distribución fueron definidos en 10 para la cruce y 20 para mutación

La detección de cambio se basa en el número de generaciones evaluadas, cuando el problema alcanza una generación que provoca un cambio en t , todos los algoritmos inician su respectivo método de respuesta al cambio. Los tres DMOEAs manejan un reemplazo del 20% de la población cada vez que se detecta un cambio. Además, en el caso de DMOCeCell y DSPEA2 el archivo es completamente vaciado al terminar un periodo. Para el proceso de selección de LLH, el número de generaciones de prueba por periodo se definió como el 20% de τ_r . El valor α de la función de elección se definió en 2000.

Para los problemas pertenecientes a los conjuntos FDA y dMOP, se define $n_t = 10$ y $\tau_r = 25$. Mientras que para los problemas del conjunto DMZDT $n_t = 100$ y $\tau_r = 25$. Estos valores se definieron acorde a las recomendaciones de sus autores y un trabajo enfocado en guiar el diseño de experimentos para optimización dinámica [Helbig y Engelbrecht, 2014]. El número de generaciones totales se definió como $n_t * \tau_r$. El número de generaciones de prueba de DPEM-HH para cada proceso de selección de LLH se define en base a las recomendaciones de los autores de la evolucionabilidad poblacional [Wang et al., 2017], que equivale a 20% de las evaluaciones totales. Por ello, se define el 20% de τ_r para cada LLH para llevar a cabo la evaluación de su evolucionabilidad poblacional.

Todas las configuraciones de DPEM-HH usando el marco jMetal 5.1, el cual contiene NSGA-II, MOCeCell y SPEA2. Todos los algoritmos fueron ejecutados 30 veces en cada DMOP utilizando una laptop con procesador AMD Ryzen 5 2.0 GHz y 12GB de memoria RAM. El tiempo de ejecución para los problemas FDA y dMOP se encontró en un rango de 10 a 15 segundos. En el caso de los problemas DMZDT el rango de tiempo de ejecución iban de 70 a 120 segundos.

Se realizó una comparación entre las diferentes configuraciones de DPEM-HH en base a la media y desviación estándar fuera de línea (promedio de todos los periodos) obtenida por el conjunto de soluciones respecto a las métricas de desempeño utilizadas. La Tabla 7.6 muestra la configuración que obtuvo el mejor para cada DMOP en cada métrica.

Los resultados muestran que tanto la selección de heurísticas basadas en una estrategia voraz como en función de elección obtuvieron los mejores resultados en base a RNI. Esto infiere que el método voraz es capaz de generar una mayor cantidad de soluciones no dominadas en comparación con los otros tres métodos de selección propuestos para el conjunto FDA. Mientras tanto, la función de elección mostró mayor efectividad para los conjuntos DMZDT y dMOP. Respecto a esta misma métrica también es importante notar que los criterios de aceptación AM e IED tuvieron un efecto cercano entre ellos y no existe una diferencia notable. Aunque, todos los movimientos (AM) parece mostrar una ligera

dominancia sobre los otros criterios para problemas de tipo I. Por lo tanto, se puede suponer que estos dos criterios generan un número de soluciones no dominadas cercano entre ellos.

Tabla 7.6. Mejor configuración de DPEM-HH por DMOP en todas las métricas evaluadas.

Problema	RNI	IGD	MS	HVR
FDA1	G, AM	CF, IED	CF, IED	CF, IED
FDA3	ROU, IED	CF, IOD	CF, AM	SR, IED
FDA4	CF, IOD	CF, IED	ROU, AM	CF, IED
FDA5	G, AM	CF, IED	ROU, AM	ROU, IOD
DMZDT1	G, AM	CF, IED	CF, IOD	CF, IED
DMZDT2	CF, IED	CF, IOD	CF, IOD	CF, IOD
DMZDT3	CF, AM	G, IOD	CF, AM	CF, AM
DMZDT4	CF, IED	G, IED	G, IED	G, IED
dMOP1	CF, AM	CF, IED	CF, IED	CF, IED
dMOP2	CF, IED	CF, IED	CF, IED	CF, IED
dMOP3	G, IED	G, IED	G, IED	G, IED

G, CF, ROU y SR indican el uso de método de selección Voraz, Función de Elección, Ruleta o Aleatorio, respectiva. AM, IOD e IED indican el uso del criterio de aceptación Todos los Movimientos, Solo Mejor Dinámico y Mejor e Igual Dinámico, respectivamente.

Si se hace enfoque respecto al IGD, se puede notar una amplia superioridad por parte del método CF. Aunque, en algunos casos, como se ve en DMZDT3, DMZDT4 y dMOP3, el método voraz genera mejores valores. También se debe observar que en todas instancias los mejores criterios de aceptación fueron IOD o IED. Lo que permite suponer que aplicar una función de elección con un criterio de aceptación que solo acepte soluciones con igual o mejor desempeño puede producir soluciones más cercanas al *POF** en comparación de las demás configuraciones.

En MS los resultados parecen seguir favoreciendo a la Función de Elección. Sin embargo, para los problemas FDA4 y FDA5, los cuales son de tres objetivos, la ruleta con memoria generó un conjunto de soluciones más disperso. No parece haber una superación notable de algún criterio de aceptación sobre los demás. Aun así, IED parece tener una ligera ventaja sobre los demás criterios.

Finalmente, para HVR se sigue observado una ventaja de la función de elección respecto a los demás valores. Además, también puede que notar que la mayoría de los DMOPs presentan mejores valores cuando utilizan criterios de aceptación que solo aceptan soluciones con igual o mejor desempeño. Esto permite inferir que la aplicación del método de selección función de elección con el criterio de aceptación de IED genera conjunto de soluciones con una mejor relación entre convergencia y diversidad para este conjunto de DMOPs en comparación con las demás configuraciones disponibles.

Debido a que cada una de las métricas de desempeño utilizadas provee información relevante acerca del conjunto de soluciones obtenidas por cada versión de DPEM-HH, todas son igualmente importantes. Por lo tanto, buscando hacer una comparación justa se hizo una sumatoria considerando el número de veces que una configuración era mejor para una métrica en cada uno de los DMOPs probados. Estos valores se muestran en la Tabla 7.7, la cual divide los resultados en base a los métodos de selección de LLH y criterios de aceptación utilizados.

Respecto a método de selección, la función de elección muestra una ventaja sobre los demás métodos. Vale la pena notar que el método voraz también tuvo un efecto fuerte en los resultados, especialmente en DMZDT4 y dMOP3, los cuales son problemas tipo I. Por otra

parte, en el caso de la función de elección, este método obtuvo resultados positivos para todas las demás instancias de los tres conjuntos probados.

Tabla 7.7. Suma de mejor posición en cada métrica por cada configuración de DPEM-HH.

	AM	IOD	IED	Total
G	3	1	7	11
CF	5	6	17	28
ROU	2	1	1	4
SR	0	0	1	1
Total	10	8	26	44

G, CF, ROU y SR indican el uso de método de selección Voraz, Función de Elección, Ruleta o Aleatorio, respectiva. AM, IOD e IED indican el uso del criterio de aceptación Todos los Movimientos, Solo Mejor Dinámico y Mejor e Igual Dinámico, respectivamente.

Revisando los resultados en base a los criterios de aceptación se puede ver una amplia ventaja por parte de IED en comparación con los demás criterios. Esto puede suponer que un criterio de aceptación que realiza un análisis, comparación y acepta resultados con un cierto grado de flexibilidad puede presentar mejores resultados que criterios más estrictos o que no tengan ninguna restricción.

Revisando las diferentes combinaciones de métodos de selección y criterios de aceptación utilizados en este periodo, se puede observar que CF con el IED es la configuración más efectiva, ya que supera altamente las demás configuraciones al punto que duplica la cantidad de veces que gana en alguna métrica en comparación con la segunda mejor configuración (G, IED).

Estos resultados permiten tener una idea del efecto que produce cada uno de los diferentes métodos de selección de LLH y criterios de aceptación de soluciones probados para esta experimentación. De igual forma, permite establecer una pauta a seguir para las experimentaciones para los demás problemas planteados en este trabajo, los cuales incorporan propiedades adicionales a los DMOPs y presentan nuevos retos a resolver.

7.2.1 Análisis de los resultados para el conjunto FDA

Las Tablas 7.8 a 7.11 muestran los valores de la media y desviación estándar fuera de línea obtenidos por cada configuración de DPEM-HH para cada instancia para cada una de las métricas de desempeño utilizadas para esta experimentación. Posteriormente se presenta un breve análisis de los resultados para cada una de las instancias de este conjunto.

Tabla 7.8. Media y desviación estándar fuera de línea de RNI obtenidas por DPEM-HH. para FDA.

DPEM-HH	FDA1	FDA3	FDA4	FDA5
G, AM	8.350E+1(4.300E+0)	8.820E+1(6.700E+0)	9.940E+1(1.300E+0)	9.990E+1(3.100E-1)
G, IOD	8.070E+1(8.200E+0)	8.330E+1(1.600E+0)	9.960E+1(1.100E+0)	9.980E+1(5.600E-1)
G, IED	7.910E+1(5.800E+0)	8.700E+1(1.100E+0)	9.920E+1(1.700E+0)	9.990E+1(3.200E-1)
CF, AM	7.850E+1(4.400E+0)	8.500E+1(6.800E+0)	9.940E+1(1.300E+0)	9.940E+1(1.210E+0)
CF, IOD	7.780E+1(7.700E+0)	8.690E+1(1.200E+0)	9.970E+1(7.000E-1)	9.960E+1(1.040E+0)
CF, IED	7.840E+1(8.900E+0)	8.760E+1(9.400E+0)	9.940E+1(1.200E+0)	9.960E+1(9.600E-1)
ROU, AM	8.100E+1(1.050E+1)	8.040E+1(1.600E+0)	9.820E+1(2.300E+0)	9.840E+1(2.070E+0)
ROU, IOD	7.920E+1(1.020E+1)	9.130E+1(8.900E+0)	9.860E+1(1.900E+0)	9.920E+1(1.700E+0)
ROU, IED	7.850E+1(1.170E+1)	9.200E+1(9.900E+0)	9.890E+1(1.700E+0)	9.890E+1(1.640E+0)
SR, AM	7.370E+1(9.200E+0)	7.980E+1(1.300E+0)	9.880E+1(1.700E+0)	9.830E+1(2.250E+0)
SR, IOD	7.520E+1(1.260E+1)	8.070E+1(1.800E+0)	9.840E+1(2.000E+0)	9.900E+1(1.650E+0)
SR, IED	7.480E+1(1.140E+1)	8.310E+1(1.800E+0)	9.910E+1(1.600E+0)	9.840E+1(2.170E+0)

FDA1. La aplicación del método voraz con AM generó los mejores valores para RNI. Sin embargo, el método CF con IED obtuvo los mejores valores para IGD, MS y HVR. Para ambos métodos de selección la evolucionabilidad poblacional es utilizada, por lo que puede entenderse su relevancia en base a los resultados obtenidos.

Tabla 7.9 Media y desviación estándar fuera de línea de IGD obtenidas por DPEM-HH para FDA.

DPEM-HH	FDA1	FDA3	FDA4	FDA5
G, AM	7.081E-3(3.558E-3)	1.753E-2(3.484E-3)	1.222E-2(9.240E-4)	1.101E-2(1.034E-3)
G, IOD	7.044E-3(4.496E-3)	1.139E-2(1.735E-3)	8.638E-3(1.218E-3)	9.873E-3(8.690E-4)
G, IED	7.342E-3(3.352E-3)	1.109E-2(1.606E-3)	8.515E-3(1.491E-3)	9.794E-3(8.820E-4)
CF, AM	7.332E-3(2.115E-3)	1.637E-2(2.132E-3)	1.243E-2(1.016E-3)	1.154E-2(1.005E-3)
CF, IOD	6.471E-3(2.093E-3)	1.090E-2(1.372E-3)	8.488E-3(1.174E-3)	1.012E-2(1.160E-3)
CF, IED	6.104E-3(2.512E-3)	1.116E-2(2.006E-3)	8.428E-3(1.252E-3)	9.644E-3(1.008E-3)
ROU, AM	9.996E-3(4.967E-3)	2.229E-2(7.664E-3)	1.525E-2(2.060E-3)	1.450E-2(2.508E-3)
ROU, IOD	7.762E-3(3.379E-3)	1.167E-2(1.669E-3)	1.058E-2(1.972E-3)	1.079E-2(1.871E-3)
ROU, IED	8.209E-3(4.066E-3)	1.159E-2(2.008E-3)	9.903E-3(1.338E-3)	1.133E-2(2.287E-3)
SR, AM	1.105E-2(4.270E-3)	2.764E-2(7.550E-3)	1.497E-2(2.121E-3)	1.379E-2(1.690E-3)
SR, IOD	8.623E-3(3.687E-3)	1.285E-2(1.784E-3)	9.656E-3(1.835E-3)	1.085E-2(1.614E-3)
SR, IED	8.895E-3(4.813E-3)	1.310E-2(1.862E-3)	9.609E-3(2.036E-3)	1.116E-2(1.637E-3)

FDA3. Varios métodos de selección resultaron efectivos para diferentes situaciones. Ruleta generó más resultados, aleatorio produce un mejor HVR y función de elección obtiene una mejor convergencia y diversidad en base a IGD y MS. Los criterios que revisan resultados (IOD e IED) produjeron mejores resultados que AM salvo en MS.

Tabla 7.10 Media y desviación estándar fuera de línea de MS obtenidas por DPEM-HH para FDA.

DPEM-HH	FDA1	FDA3	FDA4	FDA5
G, AM	9.228E-1(5.051E-2)	9.826E-1(3.398E-2)	9.997E-1(2.870E-4)	9.998E-1(1.880E-4)
G, IOD	9.146E-1(7.203E-2)	9.672E-1(6.074E-2)	9.993E-1(1.129E-3)	9.887E-1(7.726E-3)
G, IED	9.144E-1(6.147E-2)	9.813E-1(1.425E-2)	9.992E-1(9.670E-4)	9.916E-1(4.371E-3)
CF, AM	9.174E-1(3.899E-2)	9.879E-1(1.481E-2)	9.998E-1(2.150E-4)	9.998E-1(2.720E-4)
CF, IOD	9.231E-1(3.744E-2)	9.794E-1(1.551E-2)	9.994E-1(8.080E-4)	9.898E-1(8.825E-3)
CF, IED	9.334E-1(4.019E-2)	9.813E-1(1.480E-2)	9.995E-1(6.780E-4)	9.891E-1(7.499E-3)
ROU, AM	8.862E-1(8.132E-2)	9.677E-1(5.065E-2)	9.999E-1(1.050E-4)	9.999E-1(8.600E-5)
ROU, IOD	9.071E-1(6.236E-2)	9.866E-1(1.054E-2)	9.998E-1(4.570E-4)	9.904E-1(8.455E-3)
ROU, IED	9.006E-1(6.596E-2)	9.787E-1(2.042E-2)	9.999E-1(2.700E-4)	9.915E-1(8.814E-3)
SR, AM	8.803E-1(7.055E-2)	9.638E-1(5.864E-2)	9.999E-1(1.500E-4)	9.999E-1(1.070E-4)
SR, IOD	8.931E-1(6.762E-2)	9.792E-1(1.816E-2)	9.997E-1(6.260E-4)	9.878E-1(1.352E-2)
SR, IED	8.909E-1(7.938E-2)	9.745E-1(2.707E-2)	9.995E-1(8.570E-4)	9.878E-1(9.198E-3)

Tabla 7.11 Media y desviación estándar fuera de línea de HVR obtenidas por DPEM-HH para FDA

DPEM-HH	FDA1	FDA3	FDA4	FDA5
G, AM	8.661E-1(2.885E-2)	6.365E-1(6.187E-2)	6.421E-1(2.741E-2)	6.717E-1(3.841E-2)
G, IOD	8.795E-1(3.729E-2)	8.851E-1(5.451E-2)	8.120E-1(3.715E-2)	8.889E-1(5.803E-2)
G, IED	8.743E-1(2.733E-2)	8.745E-1(5.200E-2)	8.114E-1(4.651E-2)	8.758E-1(6.451E-2)
CF, AM	8.646E-1(2.066E-2)	6.570E-1(3.969E-2)	6.262E-1(3.431E-2)	6.566E-1(3.221E-2)
CF, IOD	8.843E-1(2.105E-2)	8.855E-1(4.597E-2)	8.110E-1(4.545E-2)	8.672E-1(7.282E-2)
CF, IED	8.882E-1(2.638E-2)	8.606E-1(6.884E-2)	8.180E-1(4.055E-2)	8.795E-1(5.675E-2)
ROU, AM	8.201E-1(5.298E-2)	5.634E-1(1.208E-1)	5.376E-1(5.781E-2)	5.549E-1(7.858E-2)
ROU, IOD	8.606E-1(3.530E-2)	8.636E-1(5.670E-2)	7.324E-1(7.570E-2)	9.226E-1(8.064E-2)
ROU, IED	8.540E-1(4.712E-2)	8.831E-1(6.036E-2)	7.520E-1(4.713E-2)	8.794E-1(6.961E-2)
SR, AM	7.934E-1(5.024E-2)	4.801E-1(1.051E-1)	5.428E-1(6.183E-2)	5.716E-1(6.343E-2)
SR, IOD	8.489E-1(4.578E-2)	8.889E-1(7.680E-2)	7.721E-1(6.168E-2)	9.120E-1(9.446E-2)
SR, IED	8.414E-1(6.154E-2)	8.894E-1(1.022E-1)	7.732E-1(7.079E-2)	8.969E-1(6.819E-2)

FDA4. CF con IOD o IED obtuvieron los mejores resultados respecto a número de soluciones (RNI), convergencia (IGD) y HVR. Sin embargo, los procesos aleatorios parecen

generar una mejor diversidad para este problema ya que la ruleta utilizando AM obtuvo mejores valores para MS.

FDA5. Igual que en *FDA3*, diferentes configuraciones fueron efectivas para una métrica en específico. Ruleta generó soluciones con mejor MS y HVR, voraz obtuvo más soluciones no dominadas y CF mejor convergencia. El criterio AM parece obtener un mayor número de soluciones no dominadas con buena dispersión (RNI y MS). IOD e IED generaron mejores valores para HVR e IGD, respectivamente.

La revisión anterior permite ver que para este juego de problemas CF fue efectivo para encontrar soluciones con buena convergencia (IGD y HVR), ROU para diversidad (MS) y G para un mayor número de soluciones no dominadas. Respecto a criterios de aceptación IED fue efectivo para IGD y HVR mientras que AM produjo una mayor cantidad de soluciones no dominadas con buena diversidad.

7.2.2 Análisis de los resultados para el conjunto DMZDT

Las Tablas 7.12 a 7.15 muestran los valores de la media y desviación estándar fuera de línea obtenidos por cada configuración de DPEM-HH para cada instancia del conjunto DMZDT en cada una de las métricas de desempeño utilizadas. Posteriormente se presenta un breve análisis de los resultados para cada una de las instancias de este conjunto.

Tabla 7.12. Media y desviación estándar fuera de línea de RNI obtenidas por DPEM-HH para DMZDT.

DPEM-HH	DMZDT1	DMZDT2	DMZDT3	DMZDT4
G, AM	9.851E+1(2.028E-1)	8.612E+1(1.089E+0)	9.875E+1(2.281E-1)	5.723E+1(3.300E+0)
G, IOD	9.843E+1(2.865E-1)	8.553E+1(3.418E+0)	5.633E+1(1.405E+1)	5.987E+1(4.925E+0)
G, IED	9.840E+1(4.196E-1)	8.482E+1(3.301E+0)	6.111E+1(1.791E+1)	6.038E+1(5.431E+0)
CF, AM	9.847E+1(1.797E-1)	8.625E+1(1.421E+0)	9.884E+1(2.026E-1)	5.728E+1(4.678E+0)
CF, IOD	9.849E+1(3.184E-1)	8.509E+1(3.077E+0)	6.337E+1(1.944E+1)	6.026E+1(6.874E+0)
CF, IED	9.846E+1(4.916E-1)	8.644E+1(3.487E+0)	6.082E+1(1.650E+1)	6.183E+1(7.300E+0)
ROU, AM	9.820E+1(3.121E-1)	8.589E+1(3.992E+0)	9.874E+1(2.855E-1)	5.186E+1(6.237E+0)
ROU, IOD	9.831E+1(3.495E-1)	8.452E+1(4.608E+0)	5.875E+1(1.646E+1)	5.106E+1(8.782E+0)
ROU, IED	9.825E+1(4.822E-1)	8.610E+1(5.018E+0)	5.663E+1(1.674E+1)	5.558E+1(6.728E+0)
SR, AM	9.816E+1(2.776E-1)	8.511E+1(1.376E+0)	9.871E+1(2.447E-1)	4.655E+1(4.857E+0)
SR, IOD	9.821E+1(4.511E-1)	8.337E+1(3.087E+0)	5.417E+1(1.555E+1)	5.625E+1(5.910E+0)
SR, IED	9.841E+1(3.296E-1)	8.353E+1(3.557E+0)	5.621E+1(1.744E+1)	5.758E+1(6.214E+0)

Tabla 7.13 Media y desviación estándar fuera de línea de IGD obtenidas por DPEM-HH para DMZDT.

DPEM-HH	DMZDT1	DMZDT2	DMZDT3	DMZDT4
G, AM	5.520E-3(2.160E-4)	1.126E-2(1.069E-3)	2.268E-2(1.050E-4)	4.909E-2(2.257E-2)
G, IOD	4.956E-3(3.830E-4)	1.042E-2(1.004E-3)	1.971E-2(1.174E-3)	4.064E-2(1.480E-2)
G, IED	4.973E-3(2.790E-4)	1.051E-2(9.750E-4)	2.011E-2(1.129E-3)	3.908E-2(1.663E-2)
CF, AM	5.600E-3(2.000E-4)	1.137E-2(1.138E-3)	2.264E-2(5.800E-5)	5.765E-2(2.520E-2)
CF, IOD	4.948E-3(2.950E-4)	9.497E-3(1.044E-3)	2.051E-2(9.230E-4)	4.461E-2(1.805E-2)
CF, IED	4.814E-3(2.810E-4)	9.843E-3(1.217E-3)	2.046E-2(1.153E-3)	4.703E-2(1.989E-2)
ROU, AM	6.095E-3(4.880E-4)	1.198E-2(1.153E-3)	2.254E-2(3.040E-4)	6.271E-2(2.663E-2)
ROU, IOD	5.077E-3(4.510E-4)	1.052E-2(2.017E-3)	2.048E-2(1.136E-3)	4.462E-2(1.851E-2)
ROU, IED	5.107E-3(4.480E-4)	9.916E-3(1.968E-3)	2.003E-2(1.276E-3)	4.502E-2(2.073E-2)
SR, AM	6.210E-3(2.830E-4)	1.120E-2(2.198E-3)	2.253E-2(9.100E-5)	5.931E-2(2.752E-2)
SR, IOD	5.105E-3(3.520E-4)	1.033E-2(1.451E-3)	1.979E-2(1.025E-3)	4.169E-2(1.703E-2)
SR, IED	5.022E-3(3.530E-4)	1.053E-2(1.297E-3)	2.031E-2(1.242E-3)	4.749E-2(1.881E-2)

DMZDT1. El método voraz encontró mayor número de soluciones no dominadas (RNI). Sin embargo, CF obtuvo resultados con mejor convergencia y diversidad (IGD, MS y HVR).

Los tres criterios de aceptación fueron efectivos en diversos elementos, el AM respecto a RNI, IOD respecto a MS e IED fue mejor respecto a IGD y HVR.

DMZDT2. La configuración CF, IED obtuvo el mejor valor de RNI. Pero, en el resto de las métricas, la configuración CF, IOD supero a las demás versiones de DPEM-HH. Esto supone que esta configuración es ideal para este problema.

Tabla 7.14 Media y desviación estándar fuera de línea de MS obtenidas por DPEM-HH para DMZDT.

DPEM-HH	DMZDT1	DMZDT2	DMZDT3	DMZDT4
G, AM	9.616E-1(2.070E-3)	8.825E-1(1.398E-2)	9.041E-1(5.274E-3)	7.860E-1(5.620E-2)
G, IOD	9.653E-1(5.749E-3)	8.906E-1(1.184E-2)	8.314E-1(4.625E-2)	7.914E-1(4.706E-2)
G, IED	9.652E-1(3.786E-3)	8.906E-1(1.405E-2)	8.281E-1(4.739E-2)	8.054E-1(5.486E-2)
CF, AM	9.621E-1(2.285E-3)	8.820E-1(1.581E-2)	9.062E-1(1.497E-3)	7.544E-1(4.865E-2)
CF, IOD	9.670E-1(4.113E-3)	9.013E-1(1.259E-2)	8.384E-1(4.719E-2)	7.879E-1(5.070E-2)
CF, IED	9.659E-1(4.308E-3)	8.981E-1(1.763E-2)	8.385E-1(5.164E-2)	7.807E-1(5.496E-2)
ROU, AM	9.580E-1(4.228E-3)	8.892E-1(2.379E-2)	8.988E-1(1.904E-2)	7.337E-1(6.247E-2)
ROU, IOD	9.653E-1(4.370E-3)	8.945E-1(2.476E-2)	8.068E-1(8.413E-2)	7.590E-1(8.423E-2)
ROU, IED	9.652E-1(5.069E-3)	9.004E-1(2.319E-2)	8.187E-1(6.167E-2)	7.772E-1(6.756E-2)
SR, AM	9.578E-1(3.633E-3)	8.802E-1(1.579E-2)	9.031E-1(6.080E-3)	7.342E-1(4.949E-2)
SR, IOD	9.636E-1(5.457E-3)	8.934E-1(1.977E-2)	8.132E-1(4.813E-2)	7.788E-1(5.297E-2)
SR, IED	9.655E-1(4.253E-3)	8.914E-1(1.767E-2)	8.107E-1(7.013E-2)	7.573E-1(6.230E-2)

Tabla 7.15 Media y desviación estándar fuera de línea de HVR obtenidas por DPEM-HH para DMZDT.

DPEM-HH	DMZDT1	DMZDT2	DMZDT3	DMZDT4
G, AM	8.871E-1(2.389E-3)	6.742E-1(9.850E-3)	9.186E-1(2.886E-3)	2.946E-1(1.182E-1)
G, IOD	8.993E-1(5.364E-3)	6.987E-1(1.200E-2)	6.562E-1(1.042E-1)	2.806E-1(1.080E-1)
G, IED	8.993E-1(3.602E-3)	6.987E-1(1.323E-2)	6.741E-1(1.008E-1)	3.087E-1(1.269E-1)
CF, AM	8.851E-1(2.142E-3)	6.721E-1(1.096E-2)	9.201E-1(2.206E-3)	2.127E-1(1.145E-1)
CF, IOD	8.995E-1(3.241E-3)	7.160E-1(1.459E-2)	7.054E-1(1.309E-1)	2.690E-1(1.255E-1)
CF, IED	9.024E-1(3.489E-3)	7.126E-1(1.401E-2)	7.155E-1(1.065E-1)	2.691E-1(1.412E-1)
ROU, AM	8.764E-1(8.999E-3)	6.738E-1(5.297E-2)	9.103E-1(8.033E-3)	1.995E-1(1.342E-1)
ROU, IOD	8.967E-1(7.206E-3)	7.048E-1(4.103E-2)	6.705E-1(1.163E-1)	2.591E-1(1.269E-1)
ROU, IED	8.964E-1(9.632E-3)	7.115E-1(4.196E-2)	6.501E-1(1.094E-1)	2.531E-1(1.407E-1)
SR, AM	8.733E-1(3.578E-3)	6.502E-1(1.216E-2)	9.128E-1(3.012E-3)	2.583E-1(9.697E-2)
SR, IOD	8.970E-1(4.867E-3)	7.049E-1(1.732E-2)	6.112E-1(1.039E-1)	2.922E-1(1.230E-1)
SR, IED	8.977E-1(5.207E-3)	7.006E-1(1.556E-2)	6.503E-1(1.165E-1)	2.460E-1(1.392E-1)

DMZDT3. El método voraz generó mejores resultados respecto a IGD. Por otra parte, la función de elección mostró mejores resultados para las otras tres métricas (RNI, MS y HVR). El criterio de aceptación AM fue efectivo en todos los ámbitos con excepción de la convergencia (IGD) en la cual IOD presentó un mejor valor.

DMZDT4. Una misma configuración fue la mejor para todas las métricas salvo RNI. DPEM-HH usando (G, IED) produjo un mejor IGD, MS y HVR mientras que la función de elección usando un criterio de aceptación IED produjo más soluciones no dominadas.

Los métodos voraz y función de elección tuvieron un alto impacto en estos DMOPs, ya que estos dos métodos fueron prácticamente los únicos en producir los mejores resultados. Respecto a los criterios de aceptación se observan resultados mixtos, aunque se presenta una ligera superioridad de IOD e IED sobre AM para este conjunto de problemas.

7.2.3 Análisis de los resultados para el conjunto dMOP

Las Tablas 7.16 a 7.19 muestran los valores de la media y desviación estándar fuera de línea obtenidos por cada configuración de DPEM-HH para cada instancia del conjunto dMOP para

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

RNI, IGD, MS y HVR, respectivamente. También, se muestra breve análisis de los resultados para cada una de las instancias de este conjunto.

Tabla 7.16. Media y desviación estándar fuera de línea de RNI obtenidas por DPEM-HH para dMOP.

DPEM-HH	dMOP1	dMOP2	dMOP3
G, AM	8.966E+1(2.901E+0)	9.337E+1(2.896E+0)	9.553E+1(2.053E+0)
G, IOD	8.997E+1(2.932E+0)	9.596E+1(3.336E+0)	9.574E+1(4.463E+0)
G, IED	9.080E+1(1.876E+0)	9.517E+1(3.094E+0)	9.765E+1(3.477E+0)
CF, AM	9.138E+1(8.764E-1)	9.243E+1(3.317E+0)	9.497E+1(2.780E+0)
CF, IOD	9.025E+1(2.332E+0)	9.627E+1(2.408E+0)	9.581E+1(4.042E+0)
CF, IED	9.111E+1(1.721E+0)	9.642E+1(2.324E+0)	9.580E+1(4.102E+0)
ROU, AM	8.947E+1(3.220E+0)	8.798E+1(5.955E+0)	9.427E+1(3.274E+0)
ROU, IOD	8.881E+1(3.748E+0)	9.486E+1(2.328E+0)	9.469E+1(4.456E+0)
ROU, IED	8.950E+1(3.198E+0)	9.520E+1(3.266E+0)	9.550E+1(4.495E+0)
SR, AM	8.936E+1(2.843E+0)	8.134E+1(6.982E+0)	9.298E+1(3.272E+0)
SR, IOD	8.907E+1(3.427E+0)	9.445E+1(2.346E+0)	9.492E+1(5.251E+0)
SR, IED	8.788E+1(3.757E+0)	9.436E+1(4.188E+0)	9.624E+1(4.066E+0)

Tabla 7.17 Media y desviación estándar fuera de línea de IGD obtenidas por DPEM-HH para dMOP.

DPEM-HH	dMOP1	dMOP2	dMOP3
G, AM	9.696E-3(5.793E-3)	2.054E-3(6.740E-4)	5.109E-3(1.789E-3)
G, IOD	9.572E-3(5.311E-3)	1.903E-3(1.355E-3)	2.364E-3(1.348E-3)
G, IED	9.986E-3(3.317E-3)	1.987E-3(1.049E-3)	1.760E-3(1.008E-3)
CF, AM	9.312E-3(3.103E-3)	2.247E-3(9.770E-4)	3.953E-3(1.346E-3)
CF, IOD	1.064E-2(4.531E-3)	1.636E-3(5.850E-4)	1.994E-3(1.058E-3)
CF, IED	8.151E-3(4.409E-3)	1.584E-3(6.690E-4)	2.204E-3(1.081E-3)
ROU, AM	1.046E-2(5.157E-3)	2.621E-3(8.530E-4)	4.076E-3(1.558E-3)
ROU, IOD	1.157E-2(6.467E-3)	2.115E-3(8.030E-4)	1.939E-3(9.010E-4)
ROU, IED	1.176E-2(5.345E-3)	1.795E-3(5.560E-4)	2.284E-3(1.601E-3)
SR, AM	1.164E-2(6.638E-3)	3.493E-3(1.134E-3)	4.446E-3(1.332E-3)
SR, IOD	1.219E-2(6.920E-3)	2.064E-3(1.024E-3)	2.359E-3(1.705E-3)
SR, IED	1.234E-2(7.030E-3)	2.063E-3(9.760E-4)	1.987E-3(1.294E-3)

Tabla 7.18 Media y desviación estándar fuera de línea de MS obtenidas por DPEM-HH para dMOP.

DPEM-HH	dMOP1	dMOP2	dMOP3
G, AM	8.807E-1(7.048E-2)	9.742E-1(1.358E-2)	9.149E-1(2.990E-2)
G, IOD	8.769E-1(6.607E-2)	9.729E-1(2.525E-2)	9.615E-1(3.093E-2)
G, IED	8.609E-1(4.383E-2)	9.719E-1(2.121E-2)	9.746E-1(2.205E-2)
CF, AM	8.660E-1(4.223E-2)	9.732E-1(1.781E-2)	9.323E-1(2.636E-2)
CF, IOD	8.624E-1(5.164E-2)	9.785E-1(1.186E-2)	9.706E-1(2.024E-2)
CF, IED	8.945E-1(6.178E-2)	9.788E-1(1.386E-2)	9.646E-1(2.210E-2)
ROU, AM	8.617E-1(7.175E-2)	9.648E-1(1.709E-2)	9.337E-1(2.263E-2)
ROU, IOD	8.662E-1(7.208E-2)	9.743E-1(1.799E-2)	9.721E-1(1.784E-2)
ROU, IED	8.574E-1(4.762E-2)	9.773E-1(1.078E-2)	9.615E-1(2.495E-2)
SR, AM	8.611E-1(6.736E-2)	9.555E-1(1.879E-2)	9.367E-1(2.736E-2)
SR, IOD	8.553E-1(6.906E-2)	9.745E-1(1.585E-2)	9.632E-1(3.100E-2)
SR, IED	8.636E-1(7.891E-2)	9.703E-1(2.042E-2)	9.713E-1(2.498E-2)

Tabla 7.19 Media y desviación estándar fuera de línea de HVR obtenidas por DPEM-HH para dMOP.

DPEM-HH	dMOP1	dMOP2	dMOP3
G, AM	8.833E-1(5.054E-2)	9.409E-1(1.079E-2)	9.112E-1(2.990E-2)
G, IOD	8.862E-1(4.446E-2)	9.709E-1(1.897E-2)	9.735E-1(3.093E-2)
G, IED	8.831E-1(2.931E-2)	9.697E-1(1.597E-2)	9.828E-1(2.205E-2)
CF, AM	8.886E-1(3.475E-2)	9.368E-1(1.537E-2)	9.327E-1(2.636E-2)
CF, IOD	8.776E-1(4.328E-2)	9.744E-1(1.324E-2)	9.726E-1(2.024E-2)
CF, IED	8.996E-1(4.322E-2)	9.753E-1(1.046E-2)	9.717E-1(2.210E-2)
ROU, AM	8.730E-1(4.748E-2)	9.219E-1(1.941E-2)	9.398E-1(2.736E-2)
ROU, IOD	8.717E-1(6.234E-2)	9.736E-1(1.420E-2)	9.740E-1(1.784E-2)
ROU, IED	8.715E-1(4.446E-2)	9.715E-1(1.670E-2)	9.688E-1(2.495E-2)
SR, AM	8.714E-1(5.086E-2)	8.909E-1(2.739E-2)	9.288E-1(2.263E-2)
SR, IOD	8.643E-1(5.596E-2)	9.750E-1(1.714E-2)	9.741E-1(3.100E-2)
SR, IED	8.577E-1(6.212E-2)	9.751E-1(2.576E-2)	9.764E-1(2.498E-2)

dMOP1. DPEM-HH utilizando el método de selección basada en una función de elección y criterio de aceptación IED produce mejores resultados en base a IGD, MS y HVR. Este mismo método de selección también es capaz de producir el mayor número de soluciones no dominadas, siendo la única diferencia el uso del criterio de aceptación AM para este caso.

dMOP2. Para las cuatro métricas, los mejores resultados fueron presentados en la hiperheurística usando la configuración CF, IED. Esto permite suponer que esta configuración es altamente efectiva para esta instancia en particular.

dMOP3. DPEM-HH aplicando un método de selección voraz y el criterio de aceptación de solución Mejor o Igual Dinámico resulto ser altamente efectivo en comparación de las demás configuraciones ya que obtuvo los mejores resultados para todas las métricas.

Los resultados obtenidos por este juego de problemas presentan un patrón fácil de reconocer. La aplicación de un método de selección voraz es sumamente efectiva para *dMOP3* en comparación con las demás versiones, reforzando de esta manera la menciones previamente hechas sobre la efectividad de este método para problemas de tipo I. CF es el método de selección más efectivo para obtener buena convergencia y diversidad para *dMOP1* y *dMOP2*. Respecto a los criterios de aceptación de soluciones, para las todas las métricas utilizadas para evaluar en general existe una tendencia de obtener mejores resultados al utilizar el criterio de aceptación IED. La única excepción de esta condición es el resultado obtenido en RNI para *dMOP1*, donde AM presenta mejores resultados.

7.2.4 Conclusiones

El objetivo de las pruebas realizadas en esta subsección fue llevar a cabo un análisis y comparación de diversos métodos de selección de LLHs y criterios de aceptación de soluciones, tanto propuestos dentro de la literatura, así como otros propuestos en este trabajo basados en dichos documentos. Los métodos y criterios probados incluyen estrategias enfocadas en la intensificación basándose en la voracidad, así como estrategias enfocadas en la diversidad basándose en una función de elección y el uso de técnicas aleatorias.

Los resultados de la experimentación revelaron que la aplicación de un método de selección de heurística complejo puede tener buena efectividad para los problemas revisados en comparación de métodos más sencillos. Esto es debido a que la función de elección, un método de selección que combina métricas de desempeño y evolucionabilidad poblacional para llevar a cabo una selección adecuada, resultó ser más efectiva que las otras alternativas, las cuales son más sencillas. Además, se observó que la aplicación de aleatoriedad tiene poco efecto en estos problemas y que incluso una búsqueda voraz tiene mejor resultado.

Respecto a los criterios de aceptación de soluciones, se pudo observar que la aplicación de un criterio que revisa profundamente los resultados y que tenga un umbral de aceptación no tan estricto, permite realizar una mejor exploración de las soluciones, obteniendo así mejores resultados. Esto se refleja en el criterio de aceptación IED.

En conclusión, el uso de métodos y criterios basados en estrategias voraces y en la función de elección son probablemente los enfoques más adecuados para solucionar DMOPs

de tipo I, II y III bajo las condiciones dinámicas probadas. Por lo que se recomienda su uso en futuras experimentaciones.

7.3 Comparativa de DPEM-HH respecto a DMOEAs

En esta experimentación se lleva a cabo la comparativa de DPEM-HH con los diferentes DMOEAs que utiliza como LLHs. La intención de esta sección es no solamente revisar la efectividad de DPEM-HH bajo un cierto conjunto de DMOPs, sino también identificar si la hiperheurística es capaz de igualar o mejorar su desempeño en comparación del resultado de los DMOEAs utilizados de forma independiente. Los resultados de esta sección se presentan a mayor detalle en un artículo derivado de la investigación realizada para esta tesis [Macias-Escobar et al., 2019].

Once instancias pertenecientes a los conjuntos FDA, DMZDT y dMOP son utilizadas para examinar el rendimiento de DPEM-HH en comparación con otros los DMOEAs utilizados como LLHs. FDA1, FDA3, FDA4 y FDA5 del conjunto de pruebas de la FDA, el conjunto de pruebas DMZDT y el conjunto de pruebas dMOP. Esto permite probar DPEM-HH en DMOPs de tipo I (frente óptimo de Pareto no cambia), II (POF^* y POS^* cambiantes) y III (conjunto de soluciones óptima no cambia). Este conjunto de DMOPs contiene POF^* convexos, no convexos, discontinuos y multimodales con conjuntos de soluciones óptimas uniformes y no uniformes. El valor definido de t para todos los DMOPs es $t = (1/n_t)[t/\tau_t]$.

La configuración utilizada por DPEM-HH para este experimento consiste en utilizar tres versiones de DNSGA-II (DNSGA-II-A, DNSGA-II-B y DNSGA-II-AB) como LLHs. El método de selección de LLH es el voraz, utilizando los valores obtenidos por la evolucionabilidad poblacional para determinar la selección siguiendo la definición presentada en la ecuación (5.3). *All Moves* es utilizado como el criterio de aceptación de soluciones. El objetivo de utilizar esta combinación de método y criterio es trabajar con una configuración básica pero efectiva que permita evaluar los resultados de la hiperheurística respecto a los DMOEAs con un menor esfuerzo computacional.

Teniendo en cuenta lo anterior, se debe considerar que el cálculo de evolucionabilidad poblacional puede ser un proceso muy costoso debido a la exploración que realiza durante cada generación en el espacio de búsqueda de las soluciones. Por lo tanto, el número de generaciones muestreadas debe definirse cuidadosamente para evitar un proceso computacionalmente complejo. Los autores de este concepto [Wang et al., 2017] sugieren utilizar el 20% del número total de evaluaciones. Como este trabajo está basado en el número de generaciones, dicho valor equivale al 20% de las generaciones transcurridas entre cada cambio, esto quiere decir el 20% del valor de τ_t .

Al analizar el desempeño de un algoritmo con respecto a la calidad de las soluciones obtenidas para la optimización multiobjetivo, es deseable utilizar un conjunto de métricas que puedan cubrir tantas propiedades como sea posible. Tales como la cercanía de las soluciones encontradas respecto al frente óptimo de Pareto (POF^*) y la uniformidad de la distribución entre las soluciones en el espacio de búsqueda. Por este motivo las métricas que se utilizan para evaluar los resultados son IGD, MS y HVR. Se mencionó previamente dentro

de este capítulo que estas métricas ya han sido utilizadas previamente en problemas de optimización dinámica, por lo que pueden ser utilizadas.

El tamaño de la población para todas las instancias se establece en 100 para DPEM-HH y las tres versiones de DNSGA-II. Cada versión de DNSGA-II, tanto cuando se ejecuta individualmente como cuando se utiliza como LLH, utiliza una cruce binaria simulado como operador de cruce y una distribución polinomial como operador de mutación. Las probabilidades de cruce y mutación se establecen en 0,9 y $1/n$, respectivamente, donde n es el tamaño del vector de variables del DMOP evaluado. Los índices de distribución se establecen en 10 para la cruce y 20 para la mutación. Estos valores se definen siguiendo la configuración original de DNSGA-II.

El método de detección de cambio se basa en el número de generaciones transcurridas. Cuando se llega a una generación que produce un cambio en el valor de t se considera un cambio en el entorno y se lleva a cabo el método de adaptación al cambio. Dicho método consiste en un seleccionar un 20% de la población actual. DNSGA-II-A genera nuevas soluciones que reemplazan dicho porcentaje de soluciones. DNSGA-II-B muta el subconjunto de soluciones seleccionadas. DNSGA-II-AB reemplaza la mitad de dicho subconjunto por nuevas soluciones y muta la otra mitad.

Los conjuntos FDA y dMOP definen el valor de $n_t = 10$ y $\tau_t = 25$. Mientras tanto las instancias DMZDT utilizan $n_t = 100$ y $\tau_t = 25$. Esto, siguiendo las configuraciones establecidas por los autores y una guía que propone una metodología para llevar a cabo experimentos en problemas de optimización dinámicos [Helbig y Engelbrecht, 2014]. El total de generaciones máximas por instancia se define como $n_t * \tau_t$. Los experimentos fueron llevados a cabo bajo el marco jMetal 5.1, cada algoritmo fue ejecutado 30 veces por instancia utilizando una laptop con un procesador Intel Celeron de 2.13 GHz y un 4GB de RAM.

El tiempo de ejecución de las instancias de los conjuntos de pruebas FDA y dMOP está en un rango de 10 a 30 segundos. Mientras tanto, los problemas de DMZDT tardan entre 60 y 120 segundos por ejecución.

7.3.1 Análisis de los resultados obtenidos

La media y desviación estándar fuera de línea de IGD, MS y HVR obtenidas de todos los algoritmos evaluados para cada instancia se presentan en las Tablas 7.20, 7.21 y 7.22, respectivamente. El resultado obtenido por DPEM-HH se compara con las tres versiones de DNSGA-II ejecutadas individualmente. Los mejores resultados se resaltan en negrita. Para determinar si existe una diferencia significativa entre los resultados obtenidos de cada uno de los algoritmos probados se aplica la prueba de rangos alineados de Friedman [Hodges y Lehmann, 1962] con $p = 0.05$. Después, el procedimiento post-hoc de Holm [Holm, 1979] se utiliza para comparar DPEM-HH con cada una de las versiones probadas de DNSGA-II.

DPEM-HH supera en convergencia a todas las versiones de DNSGA-II probadas en esta sección en varias instancias al tener un menor valor en IGD. Para todos los casos, excepto FDA3, DPEM-HH presenta un mejor desempeño con respecto al menos una de las versiones

de DNSGA-II. Estos resultados pueden significar que DPEM-HH es capaz de encontrar soluciones con mayor cercanía al frente óptimo en comparación con los otros algoritmos para la mayoría de los casos. En varios casos, estas afirmaciones están respaldadas por una diferencia estadísticamente significativa. El desempeño de DPEM-HH en FDA1, DMZDT1 y dMOP2 muestra una mejora estadísticamente significativa sobre todas las versiones de DNSGA-II. La Tabla 7.20 también muestra que DPEM-HH es particularmente dominante cuando se compara con DNSGA-II-B.

Tabla 7.20 Media y desviación estándar fuera de línea de IGD obtenidas por DPEM-HH y DNSGA-II.

Problema	DPEM-HH	DNSGA-II-A	DNSGA-II-B	DNSGA-II-AB
FDA1	1.959E-3(4.710E-4)	5.525E-3(2.120E-3)*	5.151E-3(1.919E-3)*	5.996E-3(2.804E-3)*
FDA3	1.331E-2(1.342E-3)	1.248E-2(9.520E-4) ■	1.269E-2(1.384E-3) ■	1.208E-2(1.084E-3) ■
FDA4	1.181E-2(7.560E-4)	1.179E-2(4.690E-4)	1.310E-2(8.470E-4)*	1.240E-2(1.011E-3)*
FDA5	1.091E-2(6.310E-4)	1.094E-2(7.570E-4)	1.184E-2(6.990E-4)*	1.134E-2(8.310E-4)
DMZDT1	4.741E-3(8.800E-5)	4.898E-3(1.420E-4)*	4.879E-3(1.830E-4)*	4.911E-3(1.610E-4)*
DMZDT2	9.334E-3(1.144E-3)	9.800E-3(1.261E-3)	9.693E-3(1.339E-3)	9.687E-3(1.104E-3)
DMZDT3	2.285E-2(2.600E-5)	2.285E-2(1.090E-4)	2.285E-2(5.500E-5)	2.284E-2(1.050E-4) ■
DMZDT4	3.579E-2(1.964E-2)	3.294E-2(1.270E-2)	3.873E-2(2.151E-2)	3.783E-2(1.854E-2)
dMOP1	6.616E-3(4.683E-3)	8.353E-3(4.305E-3)	8.669E-3(4.501E-3)	7.603E-3(4.832E-3)
dMOP2	1.221E-3(1.020E-4)	1.378E-3(1.540E-4)*	1.341E-3(2.450E-4)*	1.335E-3(1.300E-4)*
dMOP3	6.703E-3(1.984E-3)	1.667E-3(9.640E-4) ■	1.024E-2(2.131E-3)*	2.788E-3(1.483E-3) ■

Nota: * y ■ indican que DPEM-HH es significativamente mejor o peor que el DMOEA en base a las pruebas de Friedman y Holm, respectivamente.

Aunque existen casos como FDA3, DMZDT3 y dMOP3, donde la hiperheurística únicamente es capaz de superar a subconjunto de las versiones de DNSGA-II utilizadas. El concepto de la hiperheurística aún se mantiene, ya que es capaz de producir un mejor resultado que algunos de sus LLHs al trabajar de manera independiente. Estos resultados demuestran que DPEM-HH aprovecha las fortalezas de su conjunto de LLH buscando cubrir sus respectivas debilidades, produciendo de esta forma soluciones con mejor convergencia al frente óptimo en comparación con lo que logran las LLH aplicadas solas. Por tanto, se puede decir que DPEM-HH muestra su eficacia con respecto a esta característica.

Tabla 7.21 Media y desviación estándar fuera de línea de MS obtenidas por DPEM-HH y DNSGA-II.

Problema	DPEM-HH	DNSGA-II-A	DNSGA-II-B	DNSGA-II-AB
FDA1	9.796E-1(1.134E-2)	9.302E-1(3.865E-2)*	9.346E-1(4.387E-2)*	9.205E-1(5.438E-2)*
FDA3	9.938E-1(2.519E-3)	9.940E-1(3.002E-3)	9.918E-1(1.511E-2)	9.941E-1(4.420E-3)
FDA4	1.000E-0(3.000E-6)	1.000E-0(1.000E-6)	1.000E-0(1.000E-6)	1.000E-0(1.000E-6)
FDA5	1.000E-0(3.000E-6)	1.000E-0(1.000E-6)	9.999E-1(2.900E-5)	1.000E-0(0.000E-0)
DMZDT1	9.693E-1(2.304E-3)	9.681E-1(3.040E-3)	9.679E-1(3.825E-3)	9.675E-1(3.693E-3)*
DMZDT2	9.049E-1(1.327E-2)	8.996E-1(1.452E-2)	9.001E-1(1.466E-2)	9.008E-1(1.318E-2)
DMZDT3	9.103E-1(1.168E-3)	9.051E-1(1.430E-2)	9.083E-1(8.060E-3)	9.078E-1(1.451E-2)
DMZDT4	8.109E-1(5.226E-2)	8.251E-1(4.425E-2)	8.278E-1(4.865E-2)	8.112E-1(4.649E-2)
dMOP1	9.170E-1(6.169E-2)	8.885E-1(6.101E-2)	8.818E-1(6.243E-2)	9.001E-1(6.643E-2)
dMOP2	9.855E-1(3.367E-3)	9.830E-1(5.359E-3)	9.844E-1(6.070E-3)	9.839E-1(5.919E-3)
dMOP3	8.374E-1(4.451E-2)	9.740E-1(1.698E-2) ■	7.694E-1(3.972E-2)*	9.520E-1(2.627E-2) ■

Nota: * y ■ indican que DPEM-HH es significativamente mejor o peor que el DMOEA en base a las pruebas de Friedman y Holm, respectivamente.

En base a los resultados mostrados en la Tabla 7.21, DPEM-HH proporciona resultados iguales o mejores que el resto de los algoritmos para varios casos. Además, en varios de los casos en donde los resultados de DPEM-HH no superan a todos los DMOEAs, la diferencia entre la hiperheurística y el DMOEA ganador no es estadísticamente significativa, con excepción de FDA4 y dMOP3. Estos resultados permiten suponer que DPEM-HH es capaz de obtener soluciones con igual o mejor diversidad con respecto a las versiones de DNSGA-II probadas en esta sección. También, DPEM-HH ofrece los mejores valores para las

instancias de DMZDT, excepto DMZDT4, y ofrece una ventaja significativa sobre todas las versiones de DNSGA-II respecto a FDA1. Todos estos DMOPs son de tipo I, esto significa que es posible que DPEM-HH funciona especialmente bien para DMOPs de este tipo. Sin embargo, si el rendimiento de un LLH es no es satisfactorio, los resultados de una hiperheurística podrían verse afectados. Esta situación se presenta en FDA3 y dMOP3, donde el desempeño de DNSGA-II-B es bajo, tal como se presenta en la Tabla 7.21. Esto afecta el desempeño de la hiperheurística y, aunque todavía es capaz de obtener mejores resultados que DNSGA-II-B, no puede superar a uno o al resto de DMOEA.

Tabla 7.22 Media y desviación estándar fuera de línea de HVR obtenidas por DPEM-HH y DNSGA-II.

Problema	DPEM-HH	DNSGA-II-A	DNSGA-II-B	DNSGA-II-AB
FDA1	9.690E-1(4.508E-3)	9.015E-1(1.127E-2)*	9.063E-1(9.023E-3)*	9.001E-1(1.321E-2)*
FDA3	7.073E-1(2.866E-2)	7.262E-1(2.043E-2) ■	7.217E-1(3.159E-2) ■	7.339E-1(2.384E-2) ■
FDA4	6.291E-1(2.742E-2)	6.355E-1(1.862E-2)	5.871E-1(2.958E-2)*	6.176E-1(2.595E-2)
FDA5	6.638E-1(2.752E-2)	6.678E-1(2.965E-2)	6.298E-1(3.319E-2)*	6.503E-1(3.324E-2)
DMZDT1	9.016E-1(1.061E-3)	8.982E-1(1.406E-3)*	8.987E-1(1.508E-3)*	8.985E-1(1.087E-3)*
DMZDT2	7.191E-1(9.305E-3)	7.095E-1(1.041E-2)*	7.116E-1(1.063E-2)*	7.120E-1(8.866E-3)*
DMZDT3	9.337E-1(9.030E-4)	9.321E-1(3.382E-3)*	9.339E-1(2.420E-3)	9.331E-1(3.537E-3)
DMZDT4	5.222E-1(1.908E-1)	5.391E-1(1.580E-1)	5.131E-1(1.885E-1)	4.864E-1(1.938E-1)
dMOP1	9.133E-1(4.727E-2)	8.923E-1(4.509E-2)	8.898E-1(4.720E-2)	9.037E-1(4.886E-2)
dMOP2	9.642E-1(3.075E-3)	9.599E-1(3.382E-3)*	9.615E-1(2.961E-3)*	9.616E-1(3.384E-3)*
dMOP3	9.428E-1(1.725E-2)	9.816E-1(4.870E-3) ■	9.130E-1(1.964E-2)*	9.767E-1(6.899E-3) ■

Nota: * y ■ indican que DPEM-HH es significativamente mejor o peor que el DMOEA en base a las pruebas de Friedman y Holm, respectivamente.

La Tabla 7.22 presenta los resultados obtenidos por DPEM-HH y DNSGA-II para la métrica HVR. DPEM-HH supera los resultados obtenidos de al menos una de las versiones de DNSGA-II en la mayoría de los DMOPs, siendo excepción únicamente FDA3. La hiperheurística puede obtener conjuntos de soluciones que presentan una mejor relación entre convergencia y diversidad en comparación con DNSGA-II. Para FDA1, DMZDT1, DMZDT2 y dMOP2. DPEM-HH supera significativamente los resultados obtenidos por todos los demás algoritmos probados. Mientras tanto, para FDA4, FDA5, DMZDT3 y dMOP3, la hiperheurística muestra una ventaja estadísticamente significativa sobre algunas de las versiones de DNSGA-II. Cabe señalar que DPEM-HH muestra una superioridad significativa frente a DNSGA-II-B para la mayoría de las instancias probadas.

En esta misma tabla debe considerarse que en para algunos DMOP el desempeño de DPEM-HH se ve afectado por los resultados de baja calidad obtenidos por alguno de los LLH. Si bien DPEM-HH proporciona mejores resultados que los LLH de bajo rendimiento, esta situación presenta complicaciones a la hiperheurística para alcanzar mejores resultados que todas las versiones de DNSGA-II. Estos escenarios muestran una vez más que DPEM-HH cumple el concepto de hiperheurística utilizando las fortalezas de sus LLH para cubrir sus debilidades. Sin embargo, hay casos en los que el bajo rendimiento de LLH puede afectar el resultado proporcionado por la hiperheurística.

7.3.2 Análisis de DHH y DPEM-HH

A partir de los resultados obtenidos por DHH y DPEM-HH es posible pensar que estas metodologías son funcionales y efectivas para solucionar el tipo de DMOPs probados en esta sección. En las Tablas 7.23, 7.24 y 7.25 se hace una comparativa entre los resultados obtenidos de DHH y DPEM-HH para FDA1, FDA3 y el conjunto de problemas DMZDT

bajo IGD, MS y HVR, respectivamente. Para determinar si la diferencia de los valores obtenidos por cada hiperheurística es significativa se lleva a cabo una prueba estadística de Wilcoxon [Wilcoxon, 1946].

Tabla 7.23 Media y desviación estándar fuera de línea de IGD obtenidas por DHH y DPEM-HH

Problema	DHH	DPEM-HH
FDA1	4.254E-3(1.690E-3)	1.959E-3(4.710E-4)*
FDA3	1.316E-2(1.587E-3)	1.331E-2(1.342E-3)
DMZDT1	4.745E-3(8.800E-5)	4.741E-3(8.800E-5)
DMZDT2	9.349E-3(1.144E-3)	9.334E-3(1.144E-3)*
DMZDT3	2.285E-2(2.600E-5)	2.285E-2(2.600E-5)
DMZDT4	2.539E-2(1.828E-2)*	3.579E-2(1.964E-2)

Nota: * indican que la hiperheurística es significativamente mejor para dicho DMOP en base a la prueba de Wilcoxon.

Los resultados presentados en la Tabla 7.23 muestran que DPEM-HH es capaz de encontrar soluciones con una mejor convergencia respecto al frente de Pareto óptimo respecto a DHH. Esta situación se presenta especialmente para DMOPs de tipo I, ya que existe una mejora significativa para FDA1 y DMZDT2. Esto puede deberse a la capacidad de exploración adicional que provee la evolucionabilidad poblacional. Por otra parte, DHH presenta un mejor resultado para FDA3, un DMOP tipo III. Aunque la diferencia no es significativa, esto puede suponer que es ligeramente mejor que DPEM-HH para este tipo de instancias. DHH se comporta particularmente bien en comparación con DPEM-HH para DMZDT4, este también es un problema tipo I, lo cual abre la posibilidad de pensar que ambas metodologías son capaces de solucionar de forma satisfactoria este tipo de problemas.

Tabla 7.24 Media y desviación estándar fuera de línea de MS obtenidas por DHH y DPEM-HH

Problema	DHH	DPEM-HH
FDA1	9.508E-1(3.189E-2)	9.796E-1(1.134E-2)
FDA3	9.957E-1(2.786E-3)	9.938E-1(2.519E-3)
DMZDT1	9.699E-1(2.304E-3)	9.693E-1(2.304E-3)
DMZDT2	9.089E-1(1.327E-2)*	9.049E-1(1.327E-2)
DMZDT3	9.106E-1(1.168E-3)*	9.103E-1(1.168E-3)
DMZDT4	8.486E-1(3.688E-2)	8.109E-1(5.226E-2)

Nota: * indican que la hiperheurística es significativamente mejor para dicho DMOP en base a la prueba de Wilcoxon.

En la Tabla 7.24 se presentan los resultados de ambas hiperheurísticas en base a MS. DHH presenta mejores resultados que DPEM-HH para la mayoría de las instancias probadas, a excepción de FDA1. Sin embargo, la diferencia entre ambas hiperheurísticas para este DMOP no es significativa. Esto permite suponer que la capacidad de DHH para encontrar soluciones bien diversificadas en el espacio de búsqueda de función objetivo es mejor con respecto a DPEM-HH. Cabe aclarar que solamente se identificó diferencia significativa para DMZDT2 y DMZDT3, los cuales presentan *POS** convexo y de forma irregular, respectivamente. Esto puede significar que DHH es capaz de encontrar mejores soluciones cuando el *POS** tiene dichas formas. También puede suponerse que, aunque la evolucionabilidad poblacional permite una mejor convergencia, esto viene a costo de la diversidad de sus soluciones.

La Tabla 7.25 presenta los resultados de DHH y DPEM-HH para HVR en cada instancia probada. En este caso el desempeño de ambas hiperheurísticas es similar, siendo ligeramente mejor los resultados en general para DHH, únicamente siendo significativamente mejor para DMZDT2. Esto, junto con los mejores resultados para DMZDT3 y DMZDT4, complementan

el análisis de la tabla anterior, ya que DHH presenta una mejor relación entre diversidad y convergencia con respecto a DPEM-HH para estas instancias. Por otra parte, DPEM-HH es significativamente superior a DHH para FDA1, el cual es un DMOP tipo I. Esto, junto con los resultados de las dos tablas anteriores, permiten ver que DPEM-HH produce mejores resultados en esta instancia respecto a DHH tanto en convergencia, como en diversidad.

Tabla 7.25 Media y desviación estándar fuera de línea de HVR obtenidas por DHH y DPEM-HH

Problema	DHH	DPEM-HH
FDA1	9.123E-1(1.027E-2)	9.690E-1(4.508E-3)*
FDA3	7.099E-1(3.276E-2)	7.073E-1(2.866E-2)
DMZDT1	9.016E-1(1.061E-3)	9.016E-1(1.061E-3)
DMZDT2	7.207E-1(9.305E-3)*	7.191E-1(9.305E-3)
DMZDT3	9.339E-1(9.030E-4)	9.337E-1(9.030E-4)
DMZDT4	6.305E-1(2.131E-1)	5.222E-1(1.908E-1)

Nota: * indican que la hiperheurística es significativamente mejor para dicho DMOP en base a la prueba de Wilcoxon.

7.3.3 Conclusiones

Los resultados de los experimentos realizados en esta sección permiten indicar la factibilidad de la aplicación de la evolucionabilidad poblacional dentro de una hiperheurística. Además, el uso de este método FLA puede permitir obtener de manera efectiva soluciones de alta calidad si es utilizado correctamente, ya sea usado como el único factor para llevar a cabo la selección de LLHs, o combinado con otras técnicas y métricas para formar un método de selección más complejo.

Debe tomarse en cuenta que la combinación entre el enfoque voraz del método de selección y el uso de la evolucionabilidad poblacional permiten establecer un balance entre la búsqueda de convergencia de las soluciones respecto al *POF**, así como la diversidad de la población dentro del espacio de búsqueda.

La comparativa entre DHH y DPEM-HH muestra que el primero es capaz de encontrar soluciones con una mayor diversidad para varias de los DMOPs probados. Por otra parte, la evolucionabilidad poblacional permite que DPEM-HH produzca en general soluciones con una mayor convergencia hacia el *POF**. A partir de estos resultados es posible concluir que ambas metodologías son adecuadas para resolver este tipo de problemas. También en base a los resultados obtenidos esta sección y secciones anteriores de este capítulo, se puede establecer la utilidad de las hiperheurísticas, ya que tanto DHH como DPEM-HH son capaces de producir resultados de igual o mejor calidad para un mayor número de DMOPs en comparación con sus LLHs ejecutados de manera independiente.

En estos resultados se muestran una de las principales ventajas de las hiperheurísticas sobre las metaheurísticas convencionales. Dicha ventaja es su generalidad, ya que es posible incorporar dentro de una hiperheurística diferentes tipos de LLHs con propiedades y capacidades distintas. De esta forma la hiperheurística buscará aprovechar las fortalezas de cada uno sus LLHs para buscar cubrir sus debilidades. Por este motivo, debe tenerse una precaución mayor al seleccionar los LLHs a utilizar dentro de estas metodologías.

7.4 Solución de DMOPs preferenciales

En esta sección se hace revisión de todas las experimentaciones realizadas para evaluar y analizar el desempeño de múltiples DMOEAs e hiperheurísticas con PS incorporado propuestas para resolver diversos DMOPs preferenciales. Estos experimentos tienen la intención de identificar las capacidades y máximos alcances actuales a los que son capaces de llegar los algoritmos propuestos, además de analizar posibles futuras mejoras identificadas, así como trabajos futuros.

Se comparan los resultados en contra de algoritmos relevantes que usan la misma técnica de incorporación. Además, de igual forma que las secciones previas, se busca hacer una comparación de los resultados obtenidos por las hiperheurísticas respecto a sus LLHs utilizados de forma independiente. Buscando confirmar de esta forma la factibilidad y efectividad de estas metodologías para establecer una relación entre instancias, metaheurísticas y métricas de desempeño que permita definir un punto inicial rumbo a la normalización dentro del campo de optimización dinámica preferencial.

7.4.1 Análisis de factibilidad del método de Separación por Planos

Inicialmente, es deseable conocer el grado de efectividad del uso del método de Separación por Planos (PS) en un DMOEA al momento de ser incorporado en comparación a cuando no se utiliza. Por este motivo, la primera experimentación de esta Sección se enfoca en el análisis de los efectos ocurridos respecto al desempeño de un DMOEA al incorporar PS.

Para este experimento se utilizan dos versiones de DNSGA-II-A. La primera es usada para solucionar un DMOP sin considerar las preferencias del DM hasta después de haber generado sus soluciones. Al final de la ejecución, se toman las soluciones que están dentro del ROI, definido por dichas preferencias, como el conjunto de soluciones óptimas que se le devuelve al DM. La segunda versión de DNSGA-II-A agrega PS dentro de su proceso (DNSGA-II-A-PS), ejecutando dicha técnica de incorporación de preferencias durante su ejecución. De igual forma que la primera versión, DNSGA-II-A-PS devuelve al DM las soluciones encontradas dentro del ROI al final de su ejecución.

Se emplearon cinco DMOPs para llevar a cabo la comparación entre ambos DMOEAs utilizados, FDA1, FDA3 y las instancias del conjunto dMOP. La variable n_t se definió con un valor de 10, τ_t en 25 y el número de generaciones totales por cada instancia en $n_t * \tau_t$, que equivale a 250 generaciones. Para este caso se probaron configuraciones preferenciales (W), las cuales se enfocan en los extremos máximos y mínimos de f_1 . Los valores normalizados de ambas configuraciones son los siguientes:

- 1) $W_{f_1} = [0.1, 0.4]$.
 - $W_{f_2} = [0.3, 0.7]$ (FDA1, FDA3 y dMOP3).
 - $W_{f_2} = [0.65, 1.0]$ (dMOP1, dMOP2).
- 2) $W_{f_1} = [0.6, 0.9]$.
 - $W_{f_2} = [0.05, 0.3]$ (FDA1, FDA3 y dMOP3).
 - $W_{f_2} = [0.1, 0.65]$ (dMOP1, dMOP2).

Los rangos considerados para f_2 en cada uno de los DMOPs están definidos en base el área factible dentro del espacio de búsqueda de funciones objetivo de dicho objetivo tomando en cuenta el valor definido en el rango de f_1 .

La cantidad de planos que genera PS está definida en tres, uno que representa al ROI. El segundo, definido como Cercanas, representa a las soluciones cerca del ROI. El tercer plano se define como Media Distancia y son soluciones alejadas del ROI, pero cercanas al plano anterior. Cualquier solución que no pertenezca a alguno de los planos anteriores se inserta en un plano llamado Lejanas, dentro de ese grupo se incluyen soluciones tales como las soluciones extremas por cada función objetivo. Se realizó una experimentación preliminar para definir R , C , así como que tan constantemente se aplica PS en un DMOEA. Los resultados se muestran en Anexo A. La expansión por plano R se define en $\{0\%, 15\%, 30\%\}$, donde el primer valor corresponde al plano ROI, el segundo a Cercanas y el tercero a Media Distancia. El porcentaje de distribución C se define en $\{80\%, 15\%, 3\%, 2\%\}$ con cada valor representando un plano. El cuarto valor (2%) representa las soluciones aportadas por el subconjunto que no pudo ser asignado a un plano, consideradas como Lejanas. PS se aplicará cada 5 generaciones para buscar mantener un balance entre la intensificación y exploración.

Las Tablas 7.26 y 7.27 presentan los resultados de ambos DMOEAs bajo VD y HVR para ambos entornos preferenciales. Los resultados muestran de forma clara que DNSGA-II-A-PS es capaz de encontrar mejores resultados tanto en convergencia como diversidad respecto a un POF* en comparación de la versión DNSGA-II-A que no utiliza PS. La superioridad de DNSGA-II-A-PS es comprobada de forma estadística mediante el uso de una prueba de Wilcoxon con $p = 0.05$. Los resultados de esta prueba muestran que, en la mayoría de los casos, la superioridad presentada por DNSGA-II-A-PS es significativa. Bajo HVR, para todas las instancias probadas, el DMOEA con PS incorporado es superior de forma significativa respecto a su versión sin PS para ambos entornos preferenciales. También, es importante mencionar que DNSGA-II-A-PS presenta una superioridad en ambos entornos tanto para VD como HVR en FDA3, dMOP2 y dMOP3, los cuales son problemas de tipo I y II. Los resultados obtenidos pueden sugerir que PS es efectivo para DMOPs de este tipo.

Tabla 7.26. Comparativa entre DNSGA-II-A y DNSGA-II-A-PS, $W_{f1} = [0.1, 0.4]$.

Métrica	VD		HVR	
DMOEA	DNSGA-II-A	DNSGA-II-A-PS	DNSGA-II-A	DNSGA-II-A-PS
FDA1	1.594e-1(1.735e-2)	1.569e-1(1.808e-2)	6.248e-1(2.964e-2)*	6.555e-1(1.741e-2)
FDA3	1.405e+0(4.693e-1)*	5.380e-1(9.498e-2)	1.403e-1(8.757e-2)*	3.095e-1(8.590e-2)
dMOP1	1.206e-2(1.800e-2)	7.485e-3(1.401e-2)	8.288e-1(3.071e-2)*	8.517e-1(3.331e-2)
dMOP2	4.944e-2(4.961e-3)*	3.993e-2(5.434e-3)	8.147e-1(1.987e-2)*	8.506e-1(2.586e-2)
dMOP3	2.977e-2(4.172e-3)*	2.531e-2(4.627e-3)	9.077e-1(1.652e-2)*	9.293e-1(1.580e-2)

* La diferencia entre este DMOEA y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a la prueba de Wilcoxon.

Una posible razón por la cual el uso de PS es efectivo en comparación a una selección de soluciones dentro del ROI al final de la ejecución es que PS segmenta la población en subconjunto, los cuales trabajan de forma aislada y devuelven sus mejores soluciones, las cuales se unifican en una nueva población. Esta estrategia permite combinar un enfoque de intensificación al buscar las mejores soluciones de cada subconjunto, con un enfoque de diversificación al buscar soluciones ubicadas en varias áreas dentro del espacio de búsqueda, incluso aquellas que están alejadas del ROI. De esta forma, el DMOEA es capaz explotar

soluciones dentro del ROI y, al mismo tiempo, explorar áreas las cuales un método puramente enfocado a la detección de soluciones que satisfacen las preferencias del DM tendría muchas dificultades en alcanzar, como, por ejemplo, un método enfocado en un punto de referencia.

Tabla 7.27. Comparativa entre DNSGA-II-A y DNSGA-II-A-PS, $W_{f1} = [0.6, 0.9]$.

Métrica	VD		HVR	
	DNSGA-II-A	DNSGA-II-A-PS	DNSGA-II-A	DNSGA-II-A-PS
DMOEA	DNSGA-II-A	DNSGA-II-A-PS	DNSGA-II-A	DNSGA-II-A-PS
FDA1	2.829e-1(1.250e-1)*	2.432e-1(4.617e-2)	4.717e-1(4.126e-2)*	5.474e-1(3.861e-2)
FDA3	1.097e+0(4.658e-1)*	8.471e-1(1.690e-1)	1.634e-2(2.960e-2)*	1.708e-1(2.273e-2)
dMOP1	1.178e-1(7.760e-2)	1.078e-1(8.301e-2)	7.789e-1(6.266e-2)*	8.834e-1(2.770e-2)
dMOP2	3.426e-2(4.884e-3)*	2.926e-2(4.057e-3)	8.887e-1(1.280e-2)*	9.088e-1(8.507e-3)
dMOP3	4.602e-2(4.928e-2)*	3.339e-2(1.205e-2)	8.536e-1(4.224e-2)*	8.997e-1(1.951e-2)

* La diferencia entre este DMOEA y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a la prueba de Wilcoxon.

7.4.2 Análisis de DMOEAs con PS para resolver DMOPs preferenciales

Cinco instancias preferenciales fueron utilizadas para este primer experimento buscando evaluar el desempeño de varios de los algoritmos propuestos en la Sección 6.1. FDA1 y FDA3 del conjunto FDA y el conjunto dMOP. Todos estos problemas manejan dos objetivos y originalmente no tiene preferencias definidas. Por ello, se ha agregado un DM el cual establece un ROI definiendo un rango de preferencia por objetivo.

Para cada uno de estos problemas se definió la instancia de tiempo t en base a la ecuación (4.1). La selección de estas instancias se debe a la relevancia de FDA y dMOP dentro del campo. También, se siguieron las recomendaciones establecidas por los autores de dichos DMOPs y la guía para llevar a cabo experimentos de optimización dinámica seguida anteriormente [Helbig y Engelbrecht, 2014] para la selección de DMOPs y la asignación de sus valores, buscando que el conjunto tuviese una variedad de retos y características diferentes. El valor de n_t se asignó en 10, mientras que τ_t se definió en 25. El número de generaciones totales en cada DMOP equivale a $n_t * \tau_t$, que equivale a 250 generaciones.

Para todas los DMOPs se definió un tamaño de población de 100 individuos en para cada DMOEAs. DNSGA-II-A-PS, DGDE3-PS y DSPEA2-PS fueron aplicados en los cinco DMOPs preferenciales. El tamaño del archivo utilizado por DSPEA2 se definió en 100. DNSGA-II-A-PS y DSPEA2-PS utilizan cruce binaria simulada como operador de cruce y distribución polinomial como operador de mutación. De igual forma que en secciones anteriores, las probabilidades de estos dos operadores se definieron en 0.9 y $1/n$, donde n es el número de variables del DMOP, respectivamente. Los índices de distribución fueron definidos en 10 para la cruce y 20 para mutación. DGDE3-PS utiliza el método DE/rand/1/bin siguiendo los parámetros establecidos en el trabajo original [Kukkonen y Lampinen, 2005].

Los DMOEAs seleccionados para resolver los cinco DMOPs utilizados en este experimento trabajan bajo la siguiente configuración de preferencias W :

- $f_1 = [0.1, 0.4]$.
 - $f_2 = [0.3, 0.7]$ (FDA1, FDA3 y dMOP3).
 - $f_2 = [0.65, 1.0]$ (dMOP1, dMOP2).

Estos valores se asignaron buscando que ROI generado se enfoque en áreas factibles del espacio de búsqueda de soluciones. Los frentes de Pareto obtenidos se compararon el POF^*

del DMOP correspondiente al ROI. En este caso se buscó hacer énfasis en la búsqueda de soluciones bajo una sola configuración preferencial ya que este experimento se trabaja como punto de entrada para las pruebas realizadas en la Sección 7.4.3. Los valores para las demás variables requeridas al utilizar PS (R y C) son los mismos que se definieron para el experimento realizado en la Sección 7.4.1.

La detección de cambio para los DMOEAs se basa en el número de la generación evaluada, cuando el problema alcanza una generación que provoca un cambio en t , todos los algoritmos inician su respectivo método de respuesta al cambio. Los métodos de adaptación al cambio siguen el procedimiento que se mencionó para cada DMOEA en la Sección 6.1. Además, para DSPEA2-PS el archivo es completamente vaciado al terminar un periodo.

El desempeño de los tres DMOEAs propuestos se comparó con una versión dinámica de RNSGA-II (DRNSGA-II) [Deb y Sundar, 2006]. Este MOEA se adaptó para solucionar problemas dinámicos mediante la asignación de método de detección y adaptación de cambios similar a DNSGA-II. Para establecer un punto de referencia para DRNSGA-II se define un punto equivalente al centroide del ROI obtenido aplicando el vector de pesos preferenciales W .

Todos los DMOEAs se configuraron usando el marco jMetal 5.2, el cual contiene NSGA-II, GDE3, SPEA2 y RNSGA-II. Todos los algoritmos fueron ejecutados 30 veces en cada DMOP utilizando una laptop con procesador AMD Ryzen 5 2.0 GHz y 12GB de memoria RAM. El tiempo de ejecución para los problemas FDA y dMOP se encontró en un rango de 5 a 10 segundos por instancia.

Para este experimento, los frentes de Pareto obtenidos son evaluados respecto al POF^* correspondiente al ROI de cada DMOP utilizando dos métricas de desempeño para evaluar los DMOEA utilizados: distancia generacional (GD) y HVR. La definición de GD se muestra en la ecuación (7.1). Se utilizó GD en vez de IGD para esta experimentación, debido a que existe la posibilidad de que los puntos generados por los DMOEA se alejen mucho del POF^* definido por el ROI. Este aspecto podría ser ignorado por IGD si existen soluciones cercanas al ROI. Por lo tanto, se utiliza GD para la analizar la cercanía de cada una de las soluciones encontradas por los DMOEAs.

$$GD(POF, POF^*) = \frac{\sum_{v \in POF} d(v, POF^*)}{|POF|}. \quad (7.1)$$

Para determinar si la diferencia obtenida entre el mejor resultado y los demás valores obtenidos dentro de esta experimentación es significativa se utilizó la prueba de rangos alineados de Friedman con un valor de significancia de 0.05 y la prueba post-hoc de Holm.

Los resultados en base a GD pueden visualizarse en la Tabla 7.28. DNSGA-II-A-PS presento un gran desempeño bajo esta métrica para cada uno de los DMOPs a excepción de dMOP3, donde es superado significativamente por DGDE3-PS. En general los DMOEAs con PS incorporado mostraron una mejora significativa para todos los DMOEAs sobre DRNSGA-II con la excepción de FDA3, en el cual DRNSGA-II se comportó mejor que los

demás DMOEAs. Con esto podemos establecer que los DMOEAs con PS son capaces de encontrar soluciones cercanas a un ROI dentro del *POF** de DMOPs, enfocando las soluciones obtenidas a dicha región de interés.

Tabla 7.28. Mediana y desviación estándar fuera de línea de GD.

Instancia	DNSGA-II-A-PS	DGDE3-PS	DSPEA2-PS	DRNSGA-II
FDA1	4.260E-2(1.004E-2)	1.287E-1(2.522E-2)*	1.193E-1(8.919E-2)*	1.108E-1(5.955E-3)*
FDA3	1.847E-1(1.179E-1)	5.256E-1(1.013E-1)*	3.750E-1(1.803E-1)*	1.689E-1(1.517E-2)
dMOP1	1.538E-3(2.268E-2)	7.373E-3(2.522E-3)*	5.433E-3(2.920E-3)*	9.638E-2(7.856E-2)*
dMOP2	7.049E-3(1.101E-3)	7.217E-3(3.450E-4)	1.719E-2(6.612E-3)*	2.182E-1(5.370E-3)*
dMOP3	5.208E-3(1.493E-3)*	3.947E-3(2.740E-4)	1.392E-2(8.763E-3)*	9.132E-2(4.066E-3)*

* La diferencia entre este DMOEA y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

En la Tabla 7.29 se muestran los resultados obtenidos por HVR. DGDE3-PS y DNSGA-II-A-PS presentan una mejora significativa con respecto al desempeño de DRNSGA-II para las instancias dMOP. Esto significa que los DMOEAs con PS incorporado pueden encontrar soluciones cercanas y con buena dispersión respecto al ROI de DMOPs con diferentes características, ya que este juego de problemas incluye problemas de tipo I, II y III. Ambos DMOEAs también un mejor desempeño para el juego de problemas FDA, sin embargo, en este caso la diferencia no es significativa. Aun así, debe tomarse en cuenta la capacidad de mejora que ofrece PS a los DMOEAs para adaptarse mejor a los cambios de un entorno.

Tabla 7.29. Mediana y desviación estándar fuera de línea de HVR.

Instancia	DNSGA-II-A-PS	DGDE3-PS	DSPEA2-PS	DRNSGA-II
FDA1	3.511E-1(2.148E-2)	3.383E-1(8.658E-3)	1.755E-1(3.500E-2)*	3.475E-1(1.168E-2)
FDA3	3.037E-1(1.152E-1)	3.264E-1(1.039E-2)	1.240E-1(5.231E-2)*	3.159E-1(2.973E-2)
dMOP1	8.494E-1(3.301E-2)	9.088E-1(1.043E-2)	5.582E-1(1.413E-1)*	8.442E-1(5.021E-2)*
dMOP2	8.606E-1(2.804E-2)	8.588E-1(5.065E-3)	7.133E-1(4.349E-2)*	7.842E-1(1.234E-2)*
dMOP3	8.917E-1(3.263E-2)*	9.214E-1(9.006E-3)	7.373E-1(5.573E-2)*	8.847E-1(9.544E-2)*

* La diferencia entre este DMOEA y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

En general, los resultados obtenidos en esta experimentación fueron satisfactorios, ya que en la gran mayoría de los casos los DMOEAs con PS incorporado tuvieron un mejor desempeño que la versión dinámica de DRNSGA-II. En algunos casos esta mejora fue incluso significativa, por lo que se demuestra no solo la factibilidad de la aplicación de PS en DMOEAs, sino también que esta técnica de incorporación de preferencias puede mejorar el desempeño de diversos algoritmos evolutivos dinámicos.

En la Figura 7.2 podemos observar una gráfica que muestra el desempeño de DNSGA-II-A-PS en FDA1 bajo las preferencias establecidas. Como se puede ver, los *POF* encontrados por este DMOEA en cada periodo están enfocados en el ROI del problema. Esta imagen permite reconocer la factibilidad y efectividad del uso de PS en los DMOEAs.

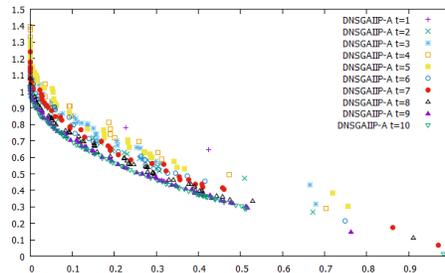


Figura 7.2 DNSGA-II-A-PS aplicado en FDA1 con $W = \{(0.1, 0.4), (0.3, 0.7)\}$.

Por otra parte, la Figura 7.3 muestra el resultado de DGDE3-PS en dMOP1, como se puede observar PS le permite adaptarse a los cambios del problema y encontrar la ubicación del nuevo ROI de forma efectiva.

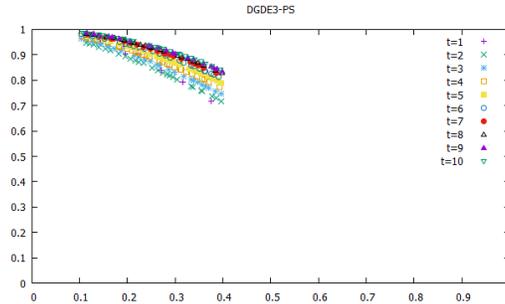


Figura 7.3 DGDE3-PS aplicado en dMOP1 con $W = \{(0.1, 0.4), (0.65, 1.0)\}$.

Algo que es necesario notar respecto a los resultados mostrados en ambas tablas, es que el desempeño de DSPEA2-PS para cada uno de los problemas fue de bajo rendimiento. Esto se puede deber a que este DMOEA no pueda adaptarse para aplicarse para solucionar DMOPs preferenciales. Por otra parte, considerando el teorema de “*Not Free Lunch*” [Wolpert y Macready, 1997], existe la posibilidad de que, aunque el desempeño de DSPEA2-PS sea bajo para estos DMOPs, exista otro subconjunto de problemas preferenciales en los cuales tenga un mejor desempeño que los demás DMOEAs utilizados en este experimento.

7.4.3 Comparativa de métodos de agregación usados en hiperheurísticas

Para iniciar los diversos experimentos llevados a cabo respecto a la inserción de PS dentro de las hiperheurísticas propuestas se realizó una comparativa entre dos métodos de agregación, los cuales son utilizados para calcular el valor de la evolucionabilidad poblacional al utilizar DPEM-HH-PS.

Como se mencionó al presentar el concepto de evolucionabilidad poblacional, este método FLA requiere del uso de un método de agregación para unificar los valores de problemas de optimización multiobjetivo en un valor único. Para la experimentación en esta sección se realizó una comparación entre el método de suma ponderada (WS) y el método de Tchebycheff (T), presentados en las ecuaciones (2.14) y (2.16), respectivamente.

Ambas versiones de DPEM-HH-PS utilizan una función de elección basada en las ecuaciones (3.23) y (3.24). Sin embargo, tienen una diferencia particular. El valor de c_2 corresponde al número de generación transcurridas y no a los milisegundos como experimentaciones anteriores. El objetivo es considerar unidades de tiempo que puedan ser utilizadas en cualquier dispositivo o equipo para realizar la prueba, sin importar sus características. Este cambio también lleva a una modificación del valor de α , el cual tiene el mismo valor asignado a la frecuencia de cambio ($\alpha = \tau_t$).

Las métricas utilizadas fueron RNI, IGD, VD y HVR. Junto a estos valores se adjuntó dentro de la ecuación el valor de la evolucionabilidad poblacional, cuyo valor define θ para la ecuación (3.24).

Siete instancias preferenciales de prueba fueron utilizadas en este experimento para evaluar el desempeño de las hiperheurísticas propuestas. FDA1, FDA3, FDA4, FDA5 y el conjunto dMOP. Estos problemas son de dos y tres objetivos y para cada uno de ellos se estableció un conjunto de preferencias por objetivo definidas por un DM *a priori*.

Cada DMOP es probado en dos configuraciones preferenciales W diferentes. Estos valores se asignaron buscando que ROI generado se enfoque en áreas factibles del espacio de búsqueda de soluciones. También, se busca probar la capacidad de las hiperheurísticas propuestas para ROIs cercanos a extremos máximos y mínimos de una función objetivo. Los *POFs* obtenidos se compararon el *POF** del DMOP correspondiente al ROI.

- 3) $W_{f1} = [0.1, 0.4]$.
 - $W_{f2} = [0.3, 0.7]$ (FDA1, FDA3 y dMOP3).
 - $W_{f2}, W_{f3} = [0.0, 1.0]$ (FDA4 y FDA5).
 - $W_{f2} = [0.65, 1.0]$ (dMOP1, dMOP2).
- 4) $W_{f1} = [0.6, 0.9]$.
 - $W_{f2} = [0.05, 0.3]$ (FDA1, FDA3 y dMOP3).
 - $W_{f2}, W_{f3} = [0.0, 0.8]$ (FDA4 y FDA5).
 - $W_{f2} = [0.1, 0.65]$ (dMOP1, dMOP2).

Para cada DMOP se definió la instancia de tiempo $t = (1/n_t) \lfloor \tau / \tau_t \rfloor$, donde n_t representa la severidad de cambio, τ_t la frecuencia de cambio y τ la generación actual. La severidad de cambio n_t se asignó en 10 y la frecuencia de cambio τ_t en 25. El número de generaciones totales se definió como $n_t * \tau_t$, lo cual equivale a 250 generaciones.

Los DMOEAs utilizados como LLHs para DPEM-HH-PS son DNSGA-II-A-PS, DNSGA-II-AB-PS y DGDE3-PS. Los operadores de selección, cruce y mutación de cada uno de los DMOEAs utilizados es similar a los definidos en la Sección 7.4.2. La detección de cambio se basa en el número de generaciones evaluadas. El método de adaptación al cambio de cada LLH sigue la estructura definida para cada uno en la Sección 6.1.

La cantidad de planos que genera PS está definida en tres (ROI, Cercanas y Media Distancia). Cualquier solución que no pertenezca a alguno de los planos anteriores se inserta en un plano llamado Lejanas. Los valores de las variables de PS se definen como $R = \{0\%, 15\%, 30\%\}$, $C = \{80\%, 15\%, 3\%, 2\%\}$ y PS se aplicará cada 5 generaciones. Esto siguiendo los resultados obtenidos en el Anexo A.

Todos los DMOEAs se configuraron usando el marco jMetal 5.2. Todos los algoritmos fueron ejecutados 30 veces en cada DMOP utilizando una laptop con procesador AMD Ryzen 5 2.0 GHz y 12GB de memoria RAM. El tiempo de ejecución para los problemas FDA y dMOP se encontró en un rango de 5 a 10 segundos por instancia.

En las Tablas 7.30 y 7.31 se presentan la mediana y desviación estándar fuera de línea obtenidas para ambas hiperheurísticas en base a VD y HVR bajo las dos configuraciones preferenciales previamente mencionadas. Los resultados marcados en negritas representan el mejor valor ente ambos métodos de agregación para la respectiva métrica. Se realizó una

comparación en a pruebas de Wilcoxon [Wilcoxon, 1946] para determinar si la diferencia entre ambas hiperheurística es estadísticamente significativa.

Tabla 7.30. DPEM-HH-PS bajo diferentes métodos de agregación, $W_{f1} = [0.1, 0.4]$.

Métrica	VD		HVR	
Método	Suma	Tchebycheff	Suma	Tchebycheff
FDA1	1.671e-1(6.683e-2)	2.284e-1(8.516e-2)	6.530e-1(1.835e-2)*	6.441e-1(2.520e-2)
FDA3	5.221e-1(2.622e-1)	5.754e-1(2.303e-1)	3.571e-1(2.645e-2)	3.615e-1(3.117e-2)
FDA4	1.221e-1(1.577e-2)	1.131e-1(1.863e-2)*	5.619e-1(2.663e-2)	5.692e-1(2.789e-2)
FDA5	1.159e-1(1.240e-2)	1.190e-1(1.993e-2)	5.656e-1(2.661e-2)	5.567e-1(2.976e-2)
dMOP1	2.616e-2(1.043e-2)*	2.841e-2(9.961e-2)	9.179e-1(1.744e-2)*	3.840e-1(2.122e-2)
dMOP2	3.825e-2(3.950e-3)	3.679e-2(4.036e-3)	8.614e-1(1.011e-2)	8.662e-1(1.209e-2)
dMOP3	1.052e-1(5.929e-2)	1.127e-1(5.804e-2)	7.967e-1(5.985e-2)	7.469e-1(7.233e-2)

* La diferencia entre este método de agregación y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a la prueba de Wilcoxon.

Como puede verse en la Tabla 7.30, la mayoría de los resultados obtenidos por DPEM-HH-PS utilizando suma ponderada como método de agregación superan a los obtenidos por Tchebycheff. En algunos casos, como en dMOP1, dicha superación es significativa. Sin embargo, se debe notar en esa misma tabla que el método de Tchebycheff es estadísticamente superior para FDA4, el cual es un problema de tres objetivos.

Tabla 7.31. DPEM-HH-PS bajo diferentes métodos de agregación, $W_{f1} = [0.6, 0.9]$.

Métrica	VD		HVR	
Método	Suma	Tchebycheff	Suma	Tchebycheff
FDA1	2.310e-1(3.157e-2)*	2.833e-1(8.257e-2)	5.420e-1(2.132e-2)	5.493e-1(2.254e-2)
FDA3	8.438e-1(1.937e-1)	8.564e-1(3.095e-1)	2.230e-1(2.165e-2)	2.222e-1(2.302e-2)
FDA4	1.802e-1(6.843e-2)	1.683e-1(3.984e-2)	5.405e-1(3.408e-2)	5.416e-1(3.509e-2)
FDA5	1.388e-1(4.136e-2)	1.438e-1(5.135e-2)	5.779e-1(3.722e-2)	5.623e-1(3.640e-2)
dMOP1	7.695e-2(1.146e-1)	2.918e-2(6.530e-2)*	8.812e-1(3.439e-2)	9.301e-1(3.118e-2)*
dMOP2	2.662e-2(2.929e-3)	2.720e-2(5.811e-3)	9.142e-1(8.015e-3)	9.125e-1(1.177e-2)
dMOP3	1.161e-1(4.584e-2)	9.284e-2(5.548e-2)	7.484e-1(6.140e-2)	7.608e-1(7.666e-2)

* La diferencia entre este método de agregación y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a la prueba de Wilcoxon.

En la Tabla 7.31, los resultados muestran que ambos métodos son idénticamente efectivos para el conjunto de DMOPs utilizados. Suma ponderada se comporta especialmente bien para la mayoría de los problemas FDA, mientras que Tchebycheff es efectivo para la mayoría de los problemas dMOP. Además, nuevamente Tchebycheff es mejor para FDA4. Los resultados del método de Tchebycheff sobre este problema de tres objetivos dan una idea de su posible utilidad para DMOPs con muchos objetivos (cuatro o más).

Los resultados obtenidos muestran que la aplicación de evolucionabilidad poblacional permite cubrir las posibles desventajas que tenga el uso del método de suma ponderada en problemas con frentes de Pareto no convexos. Por otra parte, en base al impacto que tuvo el método de Tchebycheff en DMOPs con más de dos objetivos, especialmente el caso de FDA4, así como entornos preferenciales cercanos a la cota superior para f_1 . Se debe investigar el impacto de este método en otras hiperheurísticas, utilizando métodos de selección de LLH. Por lo tanto, ambos métodos de agregación se seguirán utilizando en experimentos relacionados en la siguiente subsección.

7.4.4 Análisis de métodos de selección usados en DHH-PS y DPEM-HH-PS

En base los resultados obtenidos en la Sección 7.4.3, se llevaron una experimentación y análisis de los resultados obtenidos por diversas hiperheurísticas con PS incorporado,

utilizando diferentes métodos de selección de LLH. Las configuraciones probadas se presentan en la Tabla 7.32, todas las hiperheurísticas en esta subsección utilizan *All Moves* como criterio de aceptación.

Tabla 7.32. Hiperheurísticas utilizadas en experimentación.

Hiperheurística	Método de selección	Método de agregación	θ
DPEM-HH-PS	GR	WS	VD, HVR, EVP
	CF	WS y T	VD, HVR, EVP
	R	No aplica	HVR
	RE	No aplica	HVR
DHH-PS	GR	No aplica	VD, HVR
	CF	No aplica	VD, HVR

En las Tablas 7.33 y 7.34 se muestra la configuración que obtuvo los mejores resultados tanto para DHH-PS como DPEM-HH-PS en cada una de las instancias probadas bajo los dos entornos preferenciales definidos en la Sección 7.4.3. La configuración de ambas hiperheurísticas sigue la misma estructura definida en esa misma sección. La alternativa que obtuvo los mejores valores de los dos para cada métrica está marcada en negritas. Los valores correspondientes a dichas configuraciones para cada métrica se presentan en el Anexo B.

Tabla 7.33. Mejor configuración por problema $W_1 = [0.1,0.4]$.

Métrica	VD		HVR	
	DHH-PS	DPEM-HH-PS	DHH-PS	DPEM-HH-PS
Hiperheurística				
FDA1	GR-VD	CF-T-VD	GR-VD	CF-T-VD
FDA3	CF-WS-VD	CF-WS-EVP	CF-WS-VD	CF-T-EVP
FDA4	GR-VD	GR-VD	GR-VD	GR-VD
FDA5	GR-VD	CF-T-VD	GR-VD	CF-T-VD
dMOP1	GR-HVR	CF-T-HVR	GR-HVR	CF-WS-EVP
dMOP2	GR-HVR	CF-T-HVR	GR-HVR	GR-HVR
dMOP3	RE-HVR	GR-EVP	GR-HVR	CF-WS-EVP

Tabla 7.34. Mejor configuración por problema $W_1 = [0.6,0.9]$.

Métrica	VD		HVR	
	DHH-PS	DPEM-HH-PS	DHH-PS	DPEM-HH-PS
Hiperheurística				
FDA1	GR-VD	GR-VD	GR-VD	GR-VD
FDA3	GR-VD	CF-T-HVR	CF-WS-VD	CF-T-HVR
FDA4	GR-VD	GR-EVP	GR-VD	GR-EVP
FDA5	CF-WS-HVR	CF-WS-EVP	CF-WS-HVR	CF-WS-EVP
dMOP1	GR-HVR	CF-T-EVP	GR-HVR	CF-T-EVP
dMOP2	GR-VD	CF-T-HVR	GR-VD	CF-T-HVR
dMOP3	GR-HVR	GR-EVP	GR-HVR	GR-EVP

Como se puede observar en ambas tablas, los métodos de selección basados en una función de elección (CF y GR) son los que presentan una mayor efectividad para la gran mayoría de los problemas, siendo dMOP3 la única excepción, donde RE es efectivo en base a los resultados obtenidos por VD cuando hay una preferencia en f_1 a valores cercanos a punto de origen. Lo anterior puede deberse a que tanto CF como GR hacen un análisis exhaustivo del frente de Pareto obtenido, analizando tanto su convergencia como diversidad respecto a un frente de Pareto óptimo. Esto le permite elegir con suficiente información cual es el LLH más adecuado para el periodo actual buscando acercarse a los resultados óptimos y evitando problemas de falta de diversidad, tal como estancamiento en óptimos locales.

También, estos resultados permiten ver que los criterios basados en aleatoriedad no son efectivos dentro de un método de selección de LLH en problemas dinámicos preferenciales. El uso de este tipo de métodos de selección tiene resultados de similar bajo impacto en DMOPs no preferenciales, como se mostró en los experimentos realizados en la Sección 7.2. Por lo tanto, es factible descartar este tipo de método de selección para los siguientes experimentos, ya que no mostraron un tener un efecto relevante en la mayoría de los DMOPs.

VD y HVR como Θ fueron sumamente efectivos bajo este mismo entorno preferencial. Por otra parte, cuando existe una preferencia hacia valores cercanos al máximo para f_1 , hay una mayor efectividad al utilizar HVR y EVP como Θ .

Respecto a la comparativa entre DHH-PS y DPEM-HH-PS, es relevante la diferencia entre su efectividad al trabajar con ambos entornos definidos por el DM. Cuando se define el valor de $W_{f_1} = [0.1, 0.4]$, ambas hiperheurísticas son idénticamente efectivas, siendo DHH-PS más efectivo en el juego de instancias FDA y DPEM-HH-PS en los problemas dMOP. Sin embargo, cuando $W_{f_1} = [0.6, 0.9]$, DPEM-HH-PS tiene un mejor desempeño en todas las instancias, con excepción de dMOP3, donde DHH-PS es superior tanto en VD como en HVR.

Los resultados permiten ver la efectividad de la aplicación de la evolucionabilidad poblacional y HVR como métrica principal cuando se trabaja bajo preferencias cercanas a la cota superior de las funciones objetivo. Lo anterior puede deberse a que el uso de evolucionabilidad conlleva a una mayor exploración dentro del espacio de búsqueda, lo cual permite a los LLHs de la hiperheurística examinar soluciones que quizás hubiera tomado un mayor número de evaluaciones alcanzar usando metodologías que no utilizan EVP.

Por otra parte, el uso de hiperheurísticas que no utilicen evolucionabilidad poblacional resulta más adecuado cuando la búsqueda se enfoca en áreas cercanas a la cota inferior de las funciones objetivo. En este caso, al ser en énfasis en soluciones cercanas a los valores más bajos para cada función objetivo, podría ser preferible realizar una explotación de los resultados obtenidos, ya que dirigen su búsqueda a menores valores. Curiosamente, VD, la cual es una métrica que se enfoca totalmente en convergencia, es sumamente efectiva bajo estas condiciones preferenciales. Debido a esto, esta suposición podría ser válida.

La Tabla 7.35 presenta una comparativa entre las métricas y métodos utilizados como Θ , comparándolos en base al número de veces que fueron superiores con respecto a las otras métricas para ambos entornos preferenciales. En la Tabla 7.36 se hace un análisis similar, pero comparando el número de veces que DHH-PS es superior a DPEM-HH-PS y viceversa.

Tabla 7.35. Suma de superioridad por métrica.

Preferencia	$W_{f_1} = [0.1, 0.4]$	$W_{f_1} = [0.6, 0.9]$
VD	8	2
HVR	5	6
EVP	1	6

Tabla 7.36. Suma de superioridad por hiperheurística.

Preferencia	$W_{f_1} = [0.1, 0.4]$	$W_{f_1} = [0.6, 0.9]$
DHH-PS	7	3
DPEM-HH-PS	6	12

Estas tablas muestran la efectividad de VD en cuando $W_{f1} = [0.1,0.4]$, mientras que HVR y EVP son efectivos cuando $W_{f1} = [0.6,0.9]$. Como se mencionó anteriormente, esto puede deberse a la necesidad de explotar las soluciones existentes o explorar nuevas soluciones, en base a las preferencias establecidas por el DM. Por otra parte, también permite ver de forma más clara la efectividad de DHH-PS para problemas que requieran una mayor explotación, mientras que el uso de DPEM-HH-PS es más adecuado en situaciones donde se requiera una mayor capacidad de exploración del espacio de búsqueda de soluciones.

En base a los resultados obtenidos y revisando cada hiperheurística por separado es posible notar que DHH-PS es más efectivo cuando utiliza GR como método de selección de LLH manejando VD como su métrica principal (DHH-PS-GR-VD). Por otra parte, DPEM-HH-PS tuvo un mejor desempeño en general cuando se utilizó CF como método de selección, calculando la evolucionabilidad mediante el método de agregación de Tchebycheff y operando HVR como Θ (DPEM-HH-PS-CF-T-HVR).

7.4.5 Análisis de criterios de aceptación usados en DHH-PS y DPEM-HH-PS

En base a los resultados obtenidos anteriormente se tomaron las mejores configuraciones de hiperheurística tanto para DHH-PS como para DPEM-HH-PS y se probaron los diferentes criterios de aceptación propuestos. Los DMOPs utilizados y la configuración de cada hiperheurística y sus LLHs son acorde a lo planteado en las Secciones 7.4.3 y 7.4.4.

Las Tablas 7.37 y 7.38 muestran la comparativa de las medianas y desviaciones estándar de VD fuera de línea obtenidas por versiones de DHH-PS utilizando los criterios de aceptación *All Moves*, Distancia σ ($D(\sigma)$) y Distancia σ con aleatoriedad ($D(\sigma_{rand})$). Por otra parte, las Tablas 7.39 y 7.40 muestran esta comparativa, pero utilizando la hiperheurística DPEM-HH-PS. Para determinar si la diferencia obtenida entre el mejor resultado y los demás valores obtenidos dentro de esta experimentación es significativa se utilizó la prueba de rangos alineados de Friedman con un $p = 0.05$ y la prueba post-hoc de Holm.

Tabla 7.37. DHH-PS con diferentes criterios de aceptación, $W_{f1} = [0.1-0.4]$.

Problema	AM	$D(\sigma)$	$D(\sigma_{rand})$
FDA1	1.521e-1(3.042e-2)	4.883e-1(7.563e-1)*	3.626e-1(1.038e+0)*
FDA3	6.663e-1(2.807e-2)	8.412e-1(9.689e-1)	4.615e-1(8.006e-1)
FDA4	1.080e-1(2.175e-2)	1.085e-1(1.270e-2)	1.071e-1(1.698e-2)
FDA5	1.073e-1(1.835e-2)	1.137e-1(4.017e-2)	1.142e-1(1.904e-2)
dMOP1	3.107e-2(2.949e-2)	3.090e-2(3.129e-2)	2.742e-2(9.416e-3)
dMOP2	3.645e-2(2.867e-3)	3.887e-2(1.576e-2)*	3.650e-2(7.632e-3)
dMOP3	1.424e-1(7.105e-2)*	8.171e-2(5.893e-2)	7.972e-2(3.101e-2)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Tabla 7.38. DHH-PS con diferentes criterios de aceptación, $W_{f1} = [0.6-0.9]$.

Problema	AM	$D(\sigma)$	$D(\sigma_{rand})$
FDA1	2.256e-1(6.990e-2)	1.046e+0(1.155e+0)*	5.466e-1(1.337e+0)*
FDA3	3.083e+0(4.571e-2)	2.191e+0(1.806e+0)	2.611e+0(1.490e+0)
FDA4	1.350e-1(7.042e-2)	1.489e-1(3.668e-2)	1.465e-1(6.182e-2)
FDA5	1.529e-1(6.204e-2)	1.928e-1(7.925e-2)*	1.789e-1(7.643e-2)*
dMOP1	6.344e-2(9.586e-2)	1.240e-1(2.773e-1)*	1.072e-1(1.791e-1)*
dMOP2	2.717e-2(3.015e-3)	2.965e-2(6.224e-3)*	2.653e-2(4.244e-3)
dMOP3	1.232e-1(6.703e-2)	1.096e-1(6.165e-2)	1.352e-1(7.535e-2)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Analizando los resultados obtenidos en DHH-PS, se puede observar que *All Moves* y $D(\sigma_{rand})$, cuando $W_{f1} = [0.1, 0.4]$, son efectivos para el conjunto de DMOPs utilizados en esta experimentación, siendo superior en todos los casos sobre la distancia σ sin aleatoriedad, en algunos casos incluso de manera significativa (FDA1 y dMOP2). Esto es concordante con los resultados obtenidos en las subsecciones anteriores. Ya que, al buscar soluciones cercanas al valor mínimo de las funciones objetivo, los mejores resultados se consiguen cuando se obtiene el máximo beneficio de cada una de las soluciones existentes. Lo anterior se debe a que tanto AM como $D(\sigma_{rand})$ permiten utilizar el mismo conjunto de soluciones, aun cuando su calidad no demuestre mejoría a corto plazo.

Por otra parte, cuando se utiliza DHH-PS bajo $W_{f1} = [0.6, 0.9]$, AM y $D(\sigma)$ presentan mayor efectividad. En este caso $D(\sigma)$ tiene un mejor rendimiento que $D(\sigma_{rand})$ ya que el primer criterio permite explorar una mayor área en el espacio de soluciones. Este resultado se debe a que, si las soluciones no son mejores, se descartan y se exploran nuevas soluciones dentro de la misma área, buscando evitar estancamientos en óptimos locales.

Tabla 7.39. DPEM-HH-PS con diferentes criterios de aceptación, $W_{f1} = [0.1-0.4]$.

Problema	AM	$D(\sigma)$	$D(\sigma_{rand})$
FDA1	1.813e-1(6.839e-2)	7.952e-1(2.001e+0)*	4.194e-1(1.268e+0)*
FDA3	6.892e-1(4.741e-2)	9.810e-1(6.203e-1)*	8.095e-1(1.337e+0)*
FDA4	1.150e-1(1.868e-2)	1.143e-1(5.662e-2)	1.180e-1(5.122e-2)
FDA5	1.161e-1(1.021e-2)	1.184e-1(1.887e-2)	1.198e-1(2.671e-2)
dMOP1	2.287e-2(1.015e-2)	2.990e-2(1.155e-2)	2.608e-2(1.694e-2)
dMOP2	3.542e-2(3.750e-3)	4.156e-2(2.970e-2)*	3.917e-2(5.528e-3)
dMOP3	1.242e-1(6.010e-2)*	8.184e-2(5.667e-2)	1.010e-1(5.976e-2)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Tabla 7.40. DPEM-HH-PS con diferentes criterios de aceptación, $W_{f1} = [0.6-0.9]$.

Problema	AM	$D(\sigma)$	$D(\sigma_{rand})$
FDA1	2.494e-1(6.061e-2)	2.115e+0(1.962e+0)*	5.577e-1(1.384e+0)*
FDA3	3.078e+0(5.014e-2)*	2.422e+0(1.480e+0)	2.576e+0(1.432e+0)
FDA4	1.734e-1(6.760e-2)	3.941e-1(7.272e-1)*	2.624e-1(5.956e-1)*
FDA5	1.535e-1(6.163e-2)	2.134e-1(1.338e-1)*	1.679e-1(6.834e-2)
dMOP1	6.761e-2(8.332e-2)	9.305e-2(8.455e-2)	5.343e-2(6.992e-2)
dMOP2	2.590e-2(1.886e-3)	2.855e-2(6.312e-3)*	2.809e-2(5.231e-3)*
dMOP3	1.502e-1(6.647e-2)*	1.015e-1(5.218e-2)	1.227e-1(6.162e-2)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Los resultados para DPEM-HH-PS son en su mayoría efectivos para AM, especialmente para FDA1, FDA5 y dMOP2, en los cuales para ambos entornos preferenciales obtiene el mejor VD. $D(\sigma)$ es efectivo en problemas tipo I con dos y tres objetivos, ya que obtiene los mejores resultados para FDA4 bajo $W_{f1} = [0.1, 0.4]$ y dMOP3 para ambas configuraciones de preferencia. $D(\sigma_{rand})$ solo fue efectivo bajo dMOP1. Los resultados anteriores respaldan las conclusiones previas respecto a que la aplicación de DPEM-HH-PS se ve beneficiado al utilizar métodos enfocados en exploración del espacio de búsqueda de soluciones. Esta misma situación resulta cierta con respecto a los criterios de aceptación.

Otro aspecto interesante, es que, a diferencia de los métodos de selección, la aplicación de aleatoriedad dentro de los criterios de aceptación resultó efectiva. Bajo un entorno preferencial enfocado hacia $W_{f1} = [0.1, 0.4]$, $D(\sigma_{rand})$ obtuvo los mejores resultados para la mayoría de los DMOPs probados. Esto nos permite concluir que la aplicación de aleatoriedad

puede ser efectiva dentro un criterio de aceptación al solucionar DMOPs con preferencias definidas por un DM.

7.4.6 Análisis de DHH-PS y DPEM-HH-PS bajo diferentes condiciones dinámicas

Los descubrimientos y conclusiones alcanzadas las secciones anteriores permitieron obtener nueva información dentro de áreas pertenecientes al campo de la optimización dinámica que han sido poco exploradas, en este caso el uso de hiperheurísticas y la incorporación de preferencias de un DM. Estos descubrimientos también ofrecen puntos en los cuales dicha investigación puede profundizarse aún más para obtener nuevo conocimiento. Para esta sección, se realizaron nuevos experimentos para DHH-PS y dos versiones DPEM-HH-PS (DPEM-HH-PS-A y DPEM-HH-PS-B) bajo la siguiente configuración:

- 1) DHH-PS:
 - a) Método de selección: Función de elección, basada en ecuaciones (3.23) y (3.24), métricas: RNI, VD, IGD, HVR, $\theta = VD$, c_2 equivale al número de generaciones ocurridas desde la última selección del LLH, $\alpha = \tau_r$.
 - b) Criterio de aceptación: *All Moves*.
 - c) LLHs: DGDE3-PS, DNSGA-II-A-PS, DNSGA-II-AB-PS.
- 2) DPEM-HH-PS-A:
 - a) Método de selección: Función de elección, basada en ecuaciones (3.23) y (3.24), métricas: RNI, VD, IGD, HVR, EVP, $\theta = EVP$, c_2 equivale al número de generaciones ocurridas desde la última selección del LLH, $\alpha = \tau_r$.
 - b) Criterio de aceptación: *All Moves*.
 - c) LLHs: DGDE3-PS, DNSGA-II-A-PS, DNSGA-II-AB-PS.
- 3) DPEM-HH-PS-B:
 - a) Método de selección: Función de elección, basada en ecuaciones (3.23) y (3.24), métricas: RNI, VD, IGD, HVR, EVP $\theta = EVP$, c_2 equivale al número de generaciones ocurridas desde la última selección del LLH, $\alpha = \tau_r$.
 - b) Criterio de aceptación: *All Moves*.
 - c) LLHs: DNSGA-II-A-PS, DNSGA-II-B-PS, DNSGA-II-AB-PS.

La diferencia entre DPEM-HH-PS-A y DPEM-HH-PS-B radica en los LLHs utilizado. DPEM-HH-PS-A utiliza el mismo conjunto de LLHs usados en los experimentos previos relacionados (Sección 7.4.3, 7.4.4 y 7.4.5), mientras que DPEM-HH-PS-B utiliza tres versiones de DNSGA-II con PS incorporado. La intención de esto es buscar evaluar el desempeño de DPEM-HH-PS utilizando como LLHs únicamente algoritmos genéticos. También, se busca evaluar la capacidad de la hiperheurística para combinar algoritmos genéticos con algoritmos de evolución diferencial.

Seis instancias con preferencias fueron utilizadas para evaluar el desempeño de las hiperheurísticas propuestas. FDA1, FDA4 y FDA5, dMOP1, dMOP2 y dMOP3. Estos problemas son de dos y tres objetivos y para cada uno de ellos se estableció un conjunto de preferencias por objetivo definidas por un DM *a priori* usando las dos mismas

configuraciones preferenciales definidas en la Sección 7.4.3. Para cada DMOP se definió la instancia de tiempo $t = (1/n_t)\lceil \tau/\tau_t \rceil$.

Buscando probar la capacidad de las hiperheurísticas, se evaluó cada DMOP bajo tres diferentes entornos dinámicos, con diferentes frecuencias de cambio $\tau_t = \{10, 25, 50\}$. La intención es probar la capacidad de las hiperheurísticas para adaptarse a diferentes velocidades de cambio. La severidad de cambio n_t se asignó en 10.

La configuración de cada uno de los DMOPs utilizados como LLHs y de cada una de las variables requeridas para incorporar PS (W , R y C) es idéntica a la utilizada en los experimentos realizados desde la Sección 7.4.3. De igual forma, las herramientas y el hardware utilizados para llevar a cabo los experimentos de esta sección son los mismos.

Las Tablas 7.41 y 7.42 muestran los resultados obtenidos por las heurísticas utilizadas en base a la mediana y desviación estándar fuera de línea en VD para cada uno de los DMOPs. Los mejores resultados están marcados en negritas. Para determinar si la diferencia en los valores obtenidos identificada entre cada una de las hiperheurística presenta una relevancia significativa se utilizó de la prueba de rangos alineados de Friedman con un valor $p = 0.05$ y la prueba post-hoc de Holm para comparar los resultados entre cada pareja de algoritmos.

Comparando DHH-PS contra ambas versiones de DPEM-HH-PS, podemos ver que DHH-PS tiene un mejor comportamiento cuando el entorno del problema presenta cambios rápidos, esto equivale a un valor de τ_t bajo (10). Conforme va aumentando este valor, los resultados de ambas versiones de DPEM-HH-PS van mejorando, un ejemplo de esto se refleja en los resultados de FDA1. Esta condición se presenta principalmente cuando existe una preferencia hacia los valores superiores extremos de f_1 ($W_{f_1} = [0.6-0.9]$).

Tabla 7.41. Mediana y desviación estándar fuera de línea para VD, $W_{f_1} = [0.1-0.4]$.

τ_t	Problema	DHH-PS	DPEM-HH-PS-A	DPEM-HH-PS-B
10	FDA1	1.460e+0(2.282e-1)	1.467e+0(2.561e-1)	1.973e+0(3.045e-1)*
	FDA4	4.273e-1(8.853e-2)*	3.122e-1(6.394e-2)	1.536e+0(1.520e-1)*
	FDA5	3.853e-1(8.892e-2)*	2.966e-1(6.473e-2)	5.206e-1(9.022e-2)*
	dMOP1	2.319e+0(8.881e-1)	2.658e+0(1.151e+0)*	3.906e+0(1.483e+0)*
	dMOP2	4.358e-1(1.142e-1)	4.655e-1(1.350e-1)	6.790e-1(1.290e-1)*
	dMOP3	2.301e-1(4.211e-2)	2.238e-1(6.034e-2)	3.569e-1(5.845e-2)*
25	FDA1	1.532e-1(2.734e-2)	1.680e-1(4.569e-2)*	1.574e-1(2.048e-2)
	FDA4	1.072e-1(1.682e-2)	1.164e-1(2.394e-2)	1.417e-1(2.405e-2)*
	FDA5	1.158e-1(1.238e-2)	1.114e-1(1.090e-2)	1.371e-1(1.300e-2)*
	dMOP1	2.747e-2(1.244e-2)	3.077e-2(1.121e-1)*	1.347e-1(1.219e-1)*
	dMOP2	3.661e-2(2.907e-3)*	3.890e-2(3.224e-3)*	3.426e-2(3.889e-3)
	dMOP3	2.347e-2(1.820e-3)	2.305e-2(3.141e-3)	7.766e-2(3.296e-2)*
50	FDA1	3.101e-2(6.285e-3)*	3.274e-2(1.008e-2)*	2.787e-2(1.811e-3)
	FDA4	8.911e-2(2.880e-3)	9.040e-2(2.534e-3)	9.062e-2(3.481e-3)*
	FDA5	8.752e-2(2.661e-3)	8.926e-2(2.826e-3)	8.873e-2(3.588e-3)
	dMOP1	5.143e-3(1.056e-3)	5.390e-3(2.579e-3)	8.742e-3(4.928e-3)*
	dMOP2	9.783e-3(1.196e-3)*	9.722e-3(1.162e-3)*	8.972e-3(5.889e-4)
	dMOP3	6.996e-3(2.973e-4)	6.829e-3(2.574e-4)	7.038e-3(5.461e-3)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Una posible razón para explicar este comportamiento puede ser que a mayor número de generaciones que se presentan entre cada cambio del entorno, también le permite a DPEM-HH-PS utilizar una mayor cantidad de generaciones de prueba al llevar el proceso de

selección de LLH. Con más generaciones durante este proceso, DPEM-HH-PS puede explorar a mayor detalle la vecindad de la población actual y analizar de forma adecuada la evolucionabilidad poblacional de cada LLH. Al tener más información cuando realiza una selección de heurísticas, DPEM-HH-PS probablemente puede seleccionar el LLH que producirá mejores resultados.

La aleatoriedad de dMOP3 es reto considerable al momento de buscar resolverlo. DPEM-HH-PS es superior a DHH-PS en la mayoría de los entornos presentados para este DMOP. El uso de la evolucionabilidad poblacional permite a los LLH explorar el nuevo entorno y analizar la calidad de soluciones vecinas. Esta ventaja le permite que a DPEM-HH-PS adaptarse mejor a los cambios presentados por dMOP3, ya que puede identificar de mejor forma la probabilidad de que cada LLH obtenga mejores soluciones inmediatamente considerando la variable que actualmente este asignada como f_1 .

La efectividad de DPEM-HH-PS en dMOP2, una instancia de tipo II, puede sugerir que el uso de la evolucionabilidad poblacional es efectiva cuando se trata este tipo de problema. Lo anterior debido a que los resultados bajo ambos entornos preferenciales resultaron superiores en alguna de las dos versiones de DPEM-HH-PS sobre DHH-PS en todas menos una de las configuraciones dinámicas ($\tau_r = 10$). Esto, permite volver a suponer que DHH-PS tiene buena capacidad para resolver DMOPs bajo cambios rápidos.

Tabla 7.42. Mediana y desviación estándar fuera de línea para VD, $W_{f1} = [0.6-0.9]$.

τ_r	Problema	DHH-PS	DPEM-HH-PS-A	DPEM-HH-PS-B
10	FDA1	2.185e+0(2.889e-1)	1.994e+0(3.201e-1)	2.987e+0(4.744e-1)*
	FDA4	9.045e-2(2.264e-2)	9.416e-2(3.216e-2)	2.506e-1(4.097e-2)*
	FDA5	8.410e-1(2.283e-1)	8.639e-1(3.170e-1)	1.435e+0(2.333e-1)*
	dMOP1	1.050e+0(2.548e-1)	1.130e+0(3.089e-1)	1.279e+0(4.795e-1)*
	dMOP2	2.446e-1(3.911e-2)	2.420e-1(3.844e-2)	2.867e-1(5.926e-2)*
	dMOP3	2.578e-1(4.865e-2)	2.747e-1(4.590e-2)	4.595e-1(1.038e-1)*
	25	FDA1	2.305e-1(4.860e-2)	2.351e-1(4.276e-2)
FDA4		1.402e-1(2.503e-2)	1.734e-1(7.943e-2)*	3.768e-1(8.484e-2)*
FDA5		1.441e-1(1.020e-1)	1.642e-1(1.279e-1)	2.781e-1(5.972e-2)*
dMOP1		1.058e-1(1.449e-1)*	6.005e-2(6.643e-2)	1.360e-1(8.296e-2)*
dMOP2		2.831e-2(4.908e-3)*	2.699e-2(4.304e-3)	2.798e-2(3.704e-3)
dMOP3		3.066e-2(4.480e-3)	3.100e-2(6.230e-3)	1.053e-1(5.268e-2)*
50		FDA1	3.989e-2(6.420e-3)	4.145e-2(9.943e-3)*
	FDA4	9.346e-2(4.421e-3)	9.682e-2(9.856e-3)	1.087e-1(1.127e-2)*
	FDA5	9.267e-2(4.951e-3)	9.221e-2(6.135e-3)	1.043e-1(1.167e-2)*
	dMOP1	4.954e-3(2.742e-2)	5.007e-3(3.605e-2)	5.580e-3(4.324e-2)
	dMOP2	7.503e-3(1.139e-3)*	7.024e-3(3.349e-4)	7.383e-3(2.366e-4)*
	dMOP3	8.496e-3(5.550e-4)	8.083e-3(5.456e-4)	7.968e-3(6.799e-3)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Con base en los resultados obtenidos en dMOP1, un problema de tipo III, el uso de algoritmos de evolución diferencial (DE) como LLH es efectivo, ya que sus propiedades pueden obtener mejores resultados en comparación con DPEM-HH-PS-B, que solo usa algoritmos genéticos (GA). DHH-PS es particularmente eficaz, por lo que centrarse solo en las métricas de desempeño y no en la evolucionabilidad poblacional puede ser una mejor alternativa para solucionar DMOPs de tipo III.

DHH-PS es capaz de adaptarse mejor a DMOPs de más de dos objetivos, principalmente para FDA4, que es un DMOP tipo I. Tanto DHH-PS como DPEM-HH-PS-A se comportan

de manera similar al resolver FDA5, que se clasifica como tipo II. La calidad de DPEM-HH-PS está respaldada por el análisis realizado al revisar los resultados en dMOP2, reafirmando su capacidad para resolver DMOP de este tipo.

Para FDA4 y FDA5, aunque en general DHH-PS tiene un mejor comportamiento, hay múltiples casos en los que la aplicación de evolucionabilidad poblacional es altamente efectiva, generando mejores resultados que DHH-PS.

Comparando ambas versiones de DPEM-HH-PS, se observa que el rendimiento de DPEM-HH-PS-A es mejor que DPEM-HH-PS-B para FDA1 cuando $\tau_\tau = 10$. Sin embargo, a medida que aumenta τ_τ , la calidad de los resultados de DPEM-HH-PS-B mejora, hasta el punto de superar la otra versión cuando $\tau_\tau = 50$.

Esta misma situación se observa para dMOP2, un problema de tipo II, cuando $W_{f1} = [0.1-0.4]$. Es posible suponer que, para este tipo de instancias, el uso de un conjunto de algoritmos de evolución genética y diferencial como LLH es efectivo cuando se manejan entornos que cambian rápidamente. Por otro lado, a medida que aumenta el tiempo entre cada cambio, el uso de un conjunto de LLH formados únicamente por algoritmos genéticos parece ser una opción más apropiada. Esto en base a los resultados obtenidos por DPEM-HH-PS-B.

Los resultados obtenidos para dMOP3 en ambos entornos preferenciales muestran que el uso de DE dentro del conjunto de LLHs permite que DPEM-HH-PS-A se adapte mejor a la aleatoriedad de esta instancia. Esto no solo sugiere que la evolucionabilidad poblacional permite una buena adaptación, como se ha mencionado anteriormente, sino también que el uso de algoritmos DE aparentemente permite llevar a cabo una mejor exploración del vecindario de la población actual.

DPEM-HH-PS-A encuentra dificultades para resolver los DMOP de tres objetivos (FDA4 y FDA5). Esto denota los problemas encontrados por sus LLH para resolver problemas con más de dos objetivos. Por otra parte, DPEM-HH-PS-B enfatiza una de las principales propiedades de la hiperheurística. En este caso, las fortalezas de DGDE3-PS para resolver estos DMOP se utilizan en la búsqueda de cubrir las debilidades presentadas por los otros LLH en este entorno.

7.4.7 Comparativa de hiperheurística respecto a DMOEAs

El objetivo de los experimentos realizados en esta sección es mostrar la factibilidad de PS como un método de incorporación de preferencias para problemas de optimización dinámica. Para ello, se hace una comparación de los resultados obtenidos por DHH-PS respecto a sus LLHs. Los resultados de esta sección fueron presentados en un artículo [Macias-Escobar et al., 2020b], el cual se puede revisar por si se desea conocer a mayor detalle la estructura de cada algoritmo propuesto y el proceso de experimentación.

Para esta sección, se manejó DHH-PS utilizando DNSGA-II-A-PS, DNSGA-II-AB-PS y DGDE3-PS como LLHs. El método de selección de LLHs fue la función de elección basada en las ecuaciones (3.23) y (3.24), las métricas utilizadas son RNI, IGD, VD y HVR, donde $\theta = VD$ y el valor de c_2 está determinado por el número de generaciones transcurridas desde

la última selección de cada LLH. El criterio de aceptación es *All Moves*. La configuración de cada uno de los DMOEAs usados como LLH siguió la definición mostrada en la Sección 6.1 y utilizada en este capítulo desde la Sección 7.4.3.

Se realizaron pruebas en cinco instancias con preferencias incorporadas. FDA1, FDA3, dMOP1, dMOP2 y dMOP3. Todos los problemas son de dos objetivos y siguen la definición formal establecida en sus trabajos originales. Estas instancias cubren múltiples características dinámicas al ser de tipo I, II y III, además de tener frentes de Pareto convexos y no convexos. El valor de la variable de tiempo t para cada DMOP se basa en la ecuación (4.1). La frecuencia de cambio τ_r se definió en 25, mientras que la severidad n_t en 10. Para todos los DMOPs, el número máximo de generaciones es $n_t * \tau_r$, lo cual equivale a 250 generaciones.

Lo valores de las variables requeridas para incorporar PS en los DMOEAs y DHH-PS (W , R , C) son los mismos que están definidos en la Sección 7.4.3. Cabe recordar que dichos valores fueron obtenidos en base a un conjunto de pruebas experimentales, las cuales se muestran en el Anexo A.

Las Tablas 7.43 a 7.45 muestran la mediana y desviación estándar fuera de línea de VD, IGD y HVR, respectivamente, de los resultados obtenidos por DHH-PS, sus LLHs utilizados de forma independiente y DRNSGA-II, el cual utiliza la misma configuración definida para la Sección 7.4.2, bajo el entorno preferencial en el cual $W_{f1} = [0.1-0.4]$. Las Tablas 7.46 a 7.48 muestran los valores correspondientes a estas métricas en el entorno $W_{f1} = [0.6-0.9]$. Los mejores resultados para cada DMOP están marcados con negritas. Se utilizó una prueba de rangos alineados de Friedman y una prueba post-hoc de Holm para determinar si se presentó una diferencia significativa entre los resultados obtenidos. Las Tablas 7.49 y 7.50 muestran una comparación de DHH-PS con cada uno de los algoritmos probados para cada uno de los entornos preferenciales definidos.

Tabla 7.43. Mediana y desviación estándar fuera de línea de VD, $W_{f1} = [0.1-0.4]$.

Instancia	DHH-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS	DGDE3-PS	DRNSGA-II
FDA1	1.577e-1(6.200e-2)	1.569e-1(1.808e-2)	1.615e-1(2.620e-2)	3.324e-1(5.201e-2)*	9.453e-1(4.253e-2)*
FDA3	4.751e-1(1.434e-1)	5.380e-1(9.498e-2)	5.194e-1(1.242e-1)	1.236e+0(1.174e-1)*	1.139e+0(6.350e-2)*
dMOP1	2.490e-2(1.442e-2)*	7.485e-3(1.401e-2)	9.053e-3(2.463e-2)	2.278e-2(6.455e-3)*	5.445e-1(7.442e-1)*
dMOP2	3.690e-2(3.508e-3)	3.993e-2(5.434e-3)	3.912e-2(4.345e-3)	3.705e-2(1.714e-3)	2.166e+0(5.132e-2)*
dMOP3	1.427e-1(5.552e-2)*	2.531e-2(4.627e-3)*	3.417e-2(9.236e-3)*	2.277e-2(1.405e-3)	9.102e-1(4.187e-2)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significante en base a las pruebas de Friedman y Holm.

Tabla 7.44. Mediana y desviación estándar fuera de línea de IGD, $W_{f1} = [0.1-0.4]$.

Instancia	DHH-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS	DGDE3-PS	DRNSGA-II
FDA1	1.607e-2(5.012e-3)*	1.616e-2(3.519e-3)*	1.804e-2(5.739e-3)*	3.094e-2(5.278e-3)*	1.447e-2(1.115e-3)
FDA3	4.783e-2(1.283e-2)*	5.683e-2(1.237e-2)*	5.389e-2(1.850e-2)*	1.146e-1(1.022e-2)*	3.756e-2(3.079e-3)
dMOP1	3.629e-3(2.344e-3)	3.719e-3(3.770e-3)	5.318e-3(4.126e-3)	4.016e-3(8.809e-4)	1.440e-2(4.942e-3)*
dMOP2	4.391e-3(4.506e-4)	4.692e-3(1.956e-3)*	4.481e-3(1.606e-3)	4.270e-3(1.772e-4)	6.665e-3(5.481e-4)*
dMOP3	2.904e-2(9.467e-3)*	2.933e-3(1.756e-3)	7.465e-3(3.374e-3)*	4.453e-3(1.356e-3)*	4.661e-3(6.652e-4)*

* La diferencia entre este algoritmo y el de mejor valor (marcado en negrita) es estadísticamente significante en base a las pruebas de Friedman y Holm.

Tabla 7.45. Mediana y desviación estándar fuera de línea de HVR, $W_{f1} = [0.1-0.4]$.

Instancia	DHH-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS	DGDE3-PS	DRNSGA-II
FDA1	6.551e-1(2.034e-2)	6.555e-1(1.741e-2)	6.511e-1(3.902e-2)	6.176e-1(1.573e-2)*	6.321e-1(2.123e-2)*
FDA3	3.631e-1(3.177e-2)	3.095e-1(8.590e-2)*	3.037e-1(1.152e-1)*	3.264e-1(1.040e-2)*	3.159e-1(2.974e-2)*
dMOP1	9.199e-1(1.792e-2)	8.517e-1(3.331e-2)*	8.495e-1(3.302e-2)*	9.089e-1(1.043e-2)	8.442e-1(5.021e-2)*
dMOP2	8.658e-1(8.794e-3)	8.506e-1(2.586e-2)*	8.607e-1(2.805e-2)	8.588e-1(5.065e-3)	7.842e-1(1.234e-2)*
dMOP3	7.215e-1(7.721e-2)*	9.293e-1(1.580e-2)	8.918e-1(3.263e-2)*	9.214e-1(9.006e-3)	8.848e-1(9.544e-3)*

* La diferencia entre este algoritmo y el de mejor valor (marcado en negrita) es estadísticamente significante en base a las pruebas de Friedman y Holm.

DHH-PS presentan una mejora estadísticamente significativa en su desempeño en a VD y HVR respecto a DRNSGA-II para cada una de las instancias probadas en este experimento, siendo la única excepción HVR en dMOP3. DRNSGA-II solo presenta mejores resultados que la hiperheurística para IGD en FDA1 y FDA3. Considerando los resultados obtenidos por ambos algoritmos, es posible decir que DHH-PS presenta una mejora general tanto en convergencia como diversidad respecto a DRNSGA-II. Esta mejora permite concluir un mejor desempeño de PS respecto a métodos de incorporación de preferencia basados en un solo punto para estas instancias.

Los resultados de DHH-PS cuando se presenta una preferencia por parte del DM hacia los valores mínimos de f_1 muestran que la hiperheurística es particularmente efectiva para DMOPs de tipo II y III. Lo anterior se puede observar al analizar los resultados obtenidos en FDA3, dMOP1 y dMOP2. DHH-PS supera de forma significativa varios, y en algunos casos, incluso todos los DMOEAs respecto a HVR para estas tres instancias. Esta superación se presenta principalmente sobre DNSGA-II-A-PS. Esto demuestra no solamente que DHH-PS es una alternativa factible para solucionar estos problemas, sino que también es capaz de producir soluciones de igual o mejor calidad que la mayoría de los DMOEAs utilizados en este experimento de forma independiente.

Tabla 7.46. Mediana y desviación estándar fuera de línea de VD, $W_f = [0.6-0.9]$.

Instancia	DHH-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS	DGDE3-PS	DRNSGA-II
FDA1	2.325e-1(2.744e-2)	2.432e-1(4.617e-2)	2.297e-1(4.244e-2)	4.583e-1(5.085e-2)*	2.829e+0(6.960e-2)*
FDA3	7.924e-1(1.883e-1)	8.471e-1(1.690e-1)	7.538e-1(1.717e-1)	1.872e+0(1.687e-1)*	3.611e+0(1.449e-1)*
dMOP1	5.465e-2(1.437e-1)*	1.078e-1(8.301e-2)*	1.071e-1(7.384e-2)*	2.630e-2(9.255e-3)	1.314e+0(3.506e-1)*
dMOP2	2.766e-2(2.764e-3)*	2.926e-2(4.057e-3)*	2.915e-2(5.491e-3)*	2.577e-2(1.506e-3)	8.681e-1(2.346e-2)*
dMOP3	1.545e-1(7.718e-2)*	3.339e-2(1.205e-2)*	3.431e-2(2.533e-2)*	2.826e-2(1.536e-3)	2.404e+0(5.773e-2)*

* La diferencia entre este algoritmo y el de mejor valor (marcado en negrita) es estadísticamente significante en base a las pruebas de Friedman y Holm.

Tabla 7.47. Mediana y desviación estándar fuera de línea de IGD, $W_f = [0.6-0.9]$.

Instancia	DHH-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS	DGDE3-PS	DRNSGA-II
FDA1	2.302e-2(3.563e-3)	2.516e-2(6.158e-3)	2.321e-2(5.622e-3)	4.211e-2(4.958e-3)*	2.121e-2(1.272e-2)
FDA3	7.862e-2(1.725e-2)*	8.501e-2(1.840e-2)*	7.603e-2(1.949e-2)*	1.732e-1(1.762e-2)*	6.142e-2(2.381e-2)
dMOP1	7.483e-3(1.905e-2)	1.496e-2(1.014e-2)*	1.586e-2(1.038e-2)*	4.403e-3(1.196e-3)	5.898e-2(5.672e-2)*
dMOP2	3.276e-3(5.192e-4)	3.166e-3(4.358e-4)	3.294e-3(8.486e-4)	3.793e-3(6.816e-4)*	3.847e-3(1.451e-3)*
dMOP3	2.662e-2(1.108e-2)*	4.145e-3(1.872e-3)	4.426e-3(3.689e-3)	3.631e-3(4.322e-4)	7.281e-3(6.689e-3)*

* La diferencia entre este algoritmo y el de mejor valor (marcado en negrita) es estadísticamente significante en base a las pruebas de Friedman y Holm.

Tabla 7.48. Mediana y desviación estándar fuera de línea de HVR, $W_f = [0.6-0.9]$.

Instancia	DHH-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS	DGDE3-PS	DRNSGA-II
FDA1	5.359e-1(2.473e-2)	5.474e-1(3.861e-2)	5.428e-1(2.068e-2)	5.150e-1(1.293e-2)*	4.980e-1(4.432e-2)*
FDA3	2.250e-1(1.565e-2)	1.708e-1(2.273e-2)*	1.882e-1(2.014e-2)*	2.106e-1(9.920e-3)	1.790e-1(2.006e-2)*
dMOP1	8.851e-1(6.176e-2)*	8.834e-1(2.770e-2)*	8.809e-1(3.967e-2)*	9.264e-1(1.412e-2)	5.722e-1(3.661e-1)*
dMOP2	9.110e-1(8.472e-3)	9.088e-1(8.507e-3)	9.083e-1(1.094e-2)	9.144e-1(5.460e-3)	8.796e-1(1.639e-2)*
dMOP3	7.355e-1(8.143e-2)*	8.997e-1(1.951e-2)	8.981e-1(2.794e-2)	9.028e-1(5.036e-3)	8.245e-1(4.351e-2)*

* La diferencia entre este algoritmo y el de mejor valor (marcado en negrita) es estadísticamente significante en base a las pruebas de Friedman y Holm.

Tabla 7.49. Comparación de DHH-PS con DMOEAs, $W_f = [0.1-0.4]$.

DMOEa	FDA1			FDA3			dMOP1			dMOP2			dMOP3		
	VD	IGD	HVR	VD	IGD	HVR	VD	IGD	HVR	VD	IGD	HVR	VD	IGD	HVR
DNSGA-II-A-PS	=	=	=	=	=	+	-	=	+	=	=	+	-	-	-
DNSGA-II-AB-PS	=	=	=	=	=	+	-	=	+	=	=	=	-	-	-
DGDE3-PS	+	+	+	+	+	+	=	=	=	=	=	=	-	-	-
DRNSGA-II	+	-	+	+	-	+	+	+	+	+	+	+	+	-	-

+, - = representan si DHH-PS presenta mejor, peor e igual resultado de forma significativa bajo las pruebas de rangos alineados de Friedman y Holm.

Por otra parte, los resultados de DHH-PS cuando se tiene una preferencia hacia los valores superiores de f_1 , como se muestra en las Tablas 7.46 a 7.48 presentan información interesante. DHH-PS obtuvo resultados significativamente mejores que DGDE3-PS y

DRNSGA-II para FDA1 y FDA3 en la mayoría de las métricas utilizadas. Sin embargo, sus resultados no presentan una diferencia significativa para estas instancias respecto a ambas versiones de DNSGA-II-PS. En cambio, para el conjunto de instancias dMOP, DHH-PS es capaz de generar soluciones con mejores valores en múltiples métricas respecto a DNSGA-II-A-PS, DNSGA-II-AB-PS y DRNSGA-II. Sin embargo, se ve superado en la mayoría de los casos por DGDE3-PS, especialmente en dMOP1 y dMOP3.

Tabla 7.50. Comparación de DHH-PS con DMOEAs, $W_{f1} = [0.6-0.9]$.

DMOEA	FDA1			FDA3			dMOP1			dMOP2			dMOP3		
	VD	IGD	HVR	VD	IGD	HVR	VD	IGD	HVR	VD	IGD	HVR	VD	IGD	HVR
DNSGA-II-A-PS	=	=	=	=	=	+	=	=	=	=	=	=	-	-	-
DNSGA-II-AB-PS	=	=	=	=	=	+	=	=	+	=	=	=	-	-	-
DGDE3-PS	+	+	+	+	+	=	-	=	-	-	+	=	-	-	-
DRNSGA-II	+	=	+	+	-	+	+	+	+	+	+	+	+	-	=

+, - y = representan si DHH-PS presenta mejor, peor e igual resultado de forma significativa bajo las pruebas de rangos alineados de Friedman y Holm.

Es necesario recordar que el desempeño de una hiperheurística es dependiente del desempeño cada uno de sus LLHs. En el caso de DHH-PS, los resultados obtenidos son dependientes de cada uno de los DMOEAs utilizados. Esto refleja un detalle importante en estas metodologías, ya que una hiperheurística puede ver afectada su calidad si uno o varios de sus LLHs presentan un desempeño de menor calidad. Como se puede observar en las Tablas 7.43 a 7.48, existen casos en los cuales ciertos DMOEAs funcionan de manera excepcional sobre varios DMOPs. Debido a la naturaleza de las hiperheurísticas, DHH-PS puede buscar cubrir las debilidades de cada uno de sus LLHs aprovechando la fortaleza de dichos DMOEAs, y así, buscar obtener mejores resultados. Sin embargo, también es necesario considerar que, si las debilidades superan a las fortalezas, la hiperheurística será capaz de mejorar el desempeño únicamente hasta donde dichas fortalezas se lo permitan.

Un ejemplo claro de la situación anterior se presenta en los resultados obtenidos para dMOP3. Donde, la naturaleza aleatoria del proceso de selección de padres de DGDE3-PS resulto ser altamente efectiva para esta instancia, ya que le permite desempeñarse de manera significativamente mejor que los demás DMOEAs probados. Sin embargo, el desempeño de ambas versiones de DNSGA-II-PS en esta instancia afectan de manera directa el desempeño de DHH-PS. Aun así, el concepto de hiperheurística se mantiene, ya que DHH-PS aprovecha las fortalezas de DGDE3-PS para generar mejores resultados que DNSGA-II-A-PS y DNSGA-II-AB-PS usados de manera individual.

7.4.8 Conclusiones

La intención de los experimentos anteriores era probar la factibilidad del uso de hiperheurísticas en problemas de optimización dinámica utilizando preferencias determinadas por un DM. También, se buscó analizar e identificar el potencial del método de Separación por Planos para este tipo de problemas.

De igual forma, se hizo la revisión de múltiples métodos de selección de LLHs y criterios de aceptación de soluciones con diferentes propiedades y estrategias para identificar aquellos que fueron más adecuados para resolver DMOPs preferenciales. Esto, busca expandir la investigación realizada sobre estos mismos elementos de la hiperheurística realizada previamente en este capítulo.

En base a las experimentaciones realizadas en esta subsección es posible concluir que el uso de DHH-PS es factible y efectivo para este tipo de problemas, especialmente para DMOPs de tipo II y III. Pero, es necesario considerar al momento de seleccionar el conjunto de LLHs a utilizar aquellos que sean capaces de proveer de buenos resultados, o al menos, que dichos resultados sean aceptables por su propia cuenta. También es importante considerar muy cuidadosamente el método de selección de LLH y criterio de aceptación de soluciones a utilizar en la hiperheurística.

7.5 Pruebas preliminares de DMOEAs para solucionar DMaOPs

Antes de hacer modificaciones en las hiperheurísticas propuestas para adaptarlas a problemas de optimización de muchos objetivos, es deseable observar los efectos que provocan en la calidad de las soluciones el resolver problemas con un número de objetivos en aumento. Además, también es deseable observar los efectos que ocurren en el desempeño de diversos DMOEAs al trabajar con DMaOPs con preferencias establecidas.

En primer lugar, se debe tomar en cuenta que las configuraciones que resultaron efectivas en problemas menores de cuatro objetivos no son necesariamente efectivas en MaOPs. Tal como se mencionó la Sección 4.2, la dominancia de Pareto por sí sola no es un método adecuado para establecer una jerarquía de soluciones. Por ese motivo se busca apoyar esta estrategia con una función de escalarización como ASF.

La configuración utilizada para PS tiene definidos las mismas dos configuraciones preferenciales W definidas en la Sección 7.4.6. Los valores de R y C se definen en $\{ROI, ROI+15\%, ROI+30\%\}$ y $\{40\%, 35\%, 20\%, 5\%\}$, respectivamente. PS es aplicado cada 2 generaciones, acentuando la rotación entre intensificación y explotación. Con esto se busca mantener un balance entre intensificación y exploración de los DMOEAs. La definición de estos valores se basa en un conjunto de experimentaciones preliminares cuyos resultados se presentan en el Anexo C.

Las instancias de prueba utilizadas son FDA4 y FDA5 con cuatro, cinco y seis objetivos. Los algoritmos utilizados son DNSGA-II-PS-A, DNSGA-II-PS-AB y DGDE3-PS. La configuración de los experimentos realizados sigue la misma estructura definida en la Sección 7.4.6, salvo la frecuencia de cambio (τ_r), la cual se define en 25.

Cada una de las instancias es probada bajo dos condiciones preferencias W las cuales de igual forma que en los experimentos anteriores, se enfocan en f_1 , buscando que los demás objetivos sean definidos un rango normalizado que se encuentre dentro del área factible del espacio de búsqueda de funciones objetivo.

- 1) $W_{f_1} = [0.1, 0.4]$.
 - $W_{f_2}, W_{f_3}, W_{f_4}, W_{f_5}, W_{f_6} = [0.0, 1.0]$.
- 2) $W_{f_1} = [0.6, 0.9]$.
 - $W_{f_2} = [0.0, 0.8]$.
 - $W_{f_3}, W_{f_4}, W_{f_5}, W_{f_6} = [0.0, 1.0]$.

Para evaluar las soluciones obtenidas por cada algoritmo se unifican las poblaciones de los tres algoritmos, para obtener una población unificada. Se define un frente de Pareto verdadero en base a las soluciones no dominadas de dicha población. La anterior definición se apoya en trabajos previos [Jakubovski-Filho et al., 2018]. De dicho frente de Pareto verdadero, se obtiene un punto de referencia que consiste en el punto central de su ROI. Este punto de referencia se utiliza para evaluar ASF y R2 de cada una de las nueve heurísticas. Cada uno de los valores se organizó en un ranking del mejor al peor valor.

7.5.1 Análisis de los resultados obtenidos

Los valores obtenidos para *ASF*, *R2* y *HV* para cada algoritmo en cada uno de los problemas se muestra en las Tablas 7.51-7.53 para $W_{f1} = [0.1-0.4]$ y Tablas 7.54-7.56 para $W_{f1} = [0.6-0.9]$. Para determinar si existe diferencia significativa entre los resultados encontrados por cada algoritmo para cada métrica presentada en las tablas se utilizó la prueba de rangos alineados de Friedman con $p = 0.05$ y una prueba post-hoc de Holm.

Tabla 7.51. Mediana y desviación estándar fuera de línea para ASF con $W_{f1} = [0.1,0.4]$.

DMOP	DGDE3-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS
<i>m</i> = 4			
FDA4	7.086e+02(1.333e+04)*	4.440e+00(2.964e+02)	8.785e+00(4.903e+01)
FDA5	1.148e+13(1.024e+02)*	5.811e+00(1.190e+07)	3.321e+01(4.868e+06)
<i>m</i> = 5			
FDA4	1.938e+01(2.842e+01)*	3.200e+00(7.190e+00)	4.385e+00(4.310e+00)
FDA5	6.252e+06(2.316e+10)*	1.026e+01(6.090e+02)	1.213e+03(1.395e+09)
<i>m</i> = 6			
FDA4	9.045e+00(2.358e+01)*	2.997e+00(4.720e+00)	2.340e+00(5.846e+00)
FDA5	2.647e+03(3.191e+12)	3.502e+01(1.967e+07)	5.396e+05(2.507e+12)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Tabla 7.52. Mediana y desviación estándar fuera de línea para R2 con $W_{f1} = [0.1,0.4]$.

DMOP	DGDE3-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS
<i>m</i> = 4			
FDA4	6.927e+00(5.990e+03)	1.029e+06(4.969e+05)*	9.988e+05(4.744e+05)*
FDA5	2.326e-01(7.669e-02)	3.370e-01(9.591e-02)*	3.317e-01(1.052e-01)*
<i>m</i> = 5			
FDA4	3.029e-01(2.383e+00)	2.233e+01(1.470e+01)*	2.027e+01(1.308e+01)*
FDA5	2.094e-01(1.137e-01)	3.108e-01(9.197e-02)*	3.851e-01(9.831e-02)*
<i>m</i> = 6			
FDA4	1.466e-01(3.107e-02)	3.327e-01(1.033e-01)*	3.092e-01(1.104e-01)*
FDA5	1.684e-01(8.049e-02)	3.083e-01(9.849e-02)*	2.895e-01(7.720e-02)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Tabla 7.53. Mediana y desviación estándar fuera de línea para HV con $W_{f1} = [0.1,0.4]$.

DMOP	DGDE3-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS
<i>m</i> = 4			
FDA4	2.498e-01(9.760e-02)	0.000e+00(5.841e-03)*	0.000e+00(2.439e-03)*
FDA5	1.741e-01(4.424e-02)	2.911e-02(2.039e-02)*	2.549e-02(1.956e-02)*
<i>m</i> = 5			
FDA4	3.026e-01(8.762e-02)	3.088e-02(3.085e-02)*	4.223e-02(4.029e-02)*
FDA5	5.973e-02(3.057e-02)	1.105e-02(6.745e-03)*	5.656e-03(8.347e-03)*
<i>m</i> = 6			
FDA4	1.756e-01(6.820e-02)	2.636e-02(4.589e-02)*	3.669e-02(3.976e-02)*
FDA5	1.299e-01(4.485e-02)	3.573e-02(3.221e-02)*	4.942e-02(3.135e-02)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Los resultados bajo *ASF* en la Tabla 7.51 muestran que DNSGA-II-A-PS presenta los mejores resultados para casi todos los DMOPs probados bajo $W_{f1} = [0.1-0.4]$, con excepción

de FDA4 con seis objetivos, donde DNSGA-II-AB-PS es superior. Para esta condición preferencial, las versiones de DNSGA-II-PS son significativamente superiores a DGDE3-PS para ambos DMOPs bajo cuatro y cinco objetivos, además de FDA4 para seis objetivos.

ASF es una función que no solamente penaliza una solución en base de su mayor distancia, sino que también considera la distancia de cada uno de sus valores objetivo con respecto a los del punto de referencia mediante una sumatoria. Esto quiere decir que, si existe un valor objetivo de la solución demasiado alejado del punto de preferencia, esto causará que su valor de ASF aumente drásticamente. Esta situación se refleja en los resultados obtenidos en la Tabla 7.51. Ambas versiones de DNSGA-II-PS presentan un buen desempeño para encontrar soluciones buena calidad cuando hay preferencias del DM cercanas al punto origen.

Las Tablas 7.52 y 7.55 muestran que DGDE3-PS presenta los mejores resultados para R2 en la mayoría de los DMOPs probados bajo ambas condiciones preferenciales. Mostrando diferencia significativa en la mayoría de los casos respecto a ambas versiones de DNSGA-II-PS, siendo la única excepción DNSGA-II-AB-PS para FDA5 con cinco y seis objetivos.

Tabla 7.54. Mediana y desviación estándar fuera de línea para ASF con $W_{f1} = [0.6, 0.9]$.

DMOP	DGDE3-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS
<i>m</i> = 4			
FDA4	7.184e-01(2.966e-01)	7.976e+00(3.237e+01)*	7.962e+00(3.277e+01)*
FDA5	3.755e+03(5.668e+17)	6.514e+01(2.675e+15)	1.079e+05(5.173e+15)
<i>m</i> = 5			
FDA4	8.182e-01(2.510e+00)	6.742e+00(1.284e+01)*	6.103e+00(2.644e+01)*
FDA5	1.845e+00(1.323e+08)	3.045e+02(5.892e+10)	7.013e+07(1.465e+18)*
<i>m</i> = 6			
FDA4	8.715e-01(1.912e-01)	7.776e+00(5.574e+01)*	7.671e+00(6.485e+00)*
FDA5	1.727e+01(3.262e+21)	9.463e+03(1.837e+20)	1.433e+05(2.357e+26)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Tabla 7.55. Mediana y desviación estándar fuera de línea para R2 con $W_{f1} = [0.6, 0.9]$.

DMOP	DGDE3-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS
<i>m</i> = 4			
FDA4	1.414e-01(3.642e-02)	4.027e-01(7.463e-02)*	3.502e-01(1.004e-01)*
FDA5	3.082e-01(1.122e-01)	3.754e-01(1.271e-01)*	3.889e-01(8.645e-02)*
<i>m</i> = 5			
FDA4	1.810e-01(4.117e-02)	3.483e-01(1.120e-01)*	3.134e-01(1.023e-01)*
FDA5	3.231e+03(1.692e+03)*	1.067e+03(2.267e+03)*	1.301e+02(1.357e+03)
<i>m</i> = 6			
FDA4	2.752e-01(8.146e-02)	4.016e-01(9.689e-02)*	3.544e-01(8.174e-02)*
FDA5	9.736e-01(1.270e+00)*	4.743e-01(1.062e+00)	4.494e-01(2.054e-01)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Tabla 7.56. Mediana y desviación estándar fuera de línea para HV con $W_{f1} = [0.6, 0.9]$.

DMOP	DGDE3-PS	DNSGA-II-A-PS	DNSGA-II-AB-PS
<i>m</i> = 4			
FDA4	2.215e-01(4.683e-02)	5.160e-03(1.861e-02)*	1.873e-02(1.917e-02)*
FDA5	1.562e-01(8.078e-02)	7.786e-02(6.532e-02)*	1.556e-01(6.730e-02)
<i>m</i> = 5			
FDA4	9.698e-02(4.172e-02)	6.617e-03(1.121e-02)*	1.410e-02(3.230e-02)*
FDA5	1.399e-01(6.351e-02)	8.951e-02(5.457e-02)*	1.342e-01(9.934e-02)
<i>m</i> = 6			
FDA4	4.007e-02(2.449e-02)	1.732e-03(1.142e-02)*	4.712e-03(2.841e-02)*
FDA5	4.907e-02(3.695e-02)*	1.066e-01(6.783e-02)*	1.418e-01(8.675e-02)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

R2 es una función de escalarización que se enfoca en el valor de la mayor distancia existente entre cada solución y el punto de referencia respecto a sus valores objetivo. A diferencia de ASF, no considera la distancia de todos los valores objetivo, sino solamente del mayor. Esta estrategia presenta un mayor criterio de penalización que ASF, ya que, si alguno de los valores de la solución en espacio objetivo es muy alejado al punto de referencia, R2 solamente considerará esa distancia, descartando a los demás objetivos, aun cuando la calidad de estos haya sido de alto nivel. Con esto se busca analizar que las soluciones satisfagan las condiciones definidas por el tomador de decisiones en cada uno de los objetivos.

En base a esta consideración es posible indicar que DGDE3-PS presenta una mayor uniformidad en las soluciones obtenidas para cada uno de los objetivos de cada DMAOP con respecto a ambas versiones DNSGA-II-PS. Esta diferencia tiene mayor notoriedad cuando $W_{f1} = [0.1-0.4]$, donde DGDE3-PS tiene una diferencia estadísticamente significativa sobre DNSGA-II-A-PS para la mayoría de las instancias probadas. Esto muestra, que aun cuando DNSGA-II-PS es capaz de encontrar soluciones de buena calidad cuando las preferencias del DM se acercan al punto origen, conforme el número de objetivos aumenta, puede presentar problemas en alguno de los objetivos, alejándolo de manera significativa respecto al punto de referencia. Aun cuando para el resto de los objetivos sea más cercano que su comparativa en DGDE3-PS, ese valor objetivo rezagado causa una alta penalización bajo R2.

Esta misma situación también se presenta bajo preferencias cercanas al límite superior de f_1 . Como se ha mencionado anteriormente, DGDE3-PS presenta un mejor desempeño bajo esta situación. Por lo que es comprensible suponer que la misma situación que se presentó en la Tabla 7.52, se presente en los resultados obtenidos en la Tabla 7.54. Curiosamente, aunque en la mayoría de los casos esto resulta verdadero, para el caso de FDA5 bajo cinco y seis objetivos, DNSGA-II-AB-PS muestra resultados significativamente superiores que DGDE3-PS. Esto permite suponer que la combinación de mutación de soluciones y generación de nuevas soluciones aleatorias en el proceso de adaptación al cambio de DNSGA-II-AB-PS afecta positivamente la búsqueda de soluciones de buena calidad para esta instancia.

El hipervolumen permite identificar una relación entre la convergencia y diversidad que tienen las soluciones obtenidas por un algoritmo respecto a un frente de Pareto. En este caso se utiliza el punto de origen. Ya que la herramienta jMetal, en la cual fueron implementados y ejecutados los DMOEAs desarrollados, invierte el frente obtenido por un algoritmo en base a un frente de referencia, en este caso el frente verdadero generado en base de las soluciones no dominadas de cada uno de los algoritmos, para analizarlo sobre dicho punto, se puede entender que mientras mayor sea el hipervolumen, mejor será la relación entre la convergencia y diversidad de la población.

En base a este análisis es posible pensar en la posibilidad de que DGDE3-PS presenta una mejor relación de convergencia y diversidad para todas las instancias probadas respecto a las dos versiones de DNSGA-II-PS cuando $W_{f1} = [0.1-0.4]$, tal como se muestra en la Tabla 7.53. En este caso la diferencia entre DGDE3-PS y los demás algoritmos es tan amplia que existe una diferencia estadísticamente significativa para FDA4 y FDA5 usando cuatro, cinco y seis objetivos. Esto claramente demuestra que DGDE3-PS puede no solamente obtener

soluciones cercanas a un frente de Pareto verdadero o un punto de referencia, sino que también puede mantener suficiente diversidad dentro del ROI. Lo cual, le ofrece al DM una mayor variedad de soluciones a elegir.

La Tabla 7.56 presenta la misma situación que la Tabla 7.53, donde DGDE3-PS es significativamente superior a ambas versiones de DNSGA-II-PS. La única excepción siendo DNSGA-II-AB-PS para FDA5 con 6 objetivos. Esta situación apoya dos de las suposiciones presentadas previamente. Primero, DGDE3-PS siendo capaz de encontrar mejores soluciones cuando existe una preferencia del DM en valores cercanos al límite máximo para f_1 . Segundo, que la adaptación al cambio mediante la combinación de generación de soluciones aleatorias y soluciones mutadas pueden tener producir un buen desempeño en problemas de muchos objetivos dinámicos.

Estos resultados permiten entender que, en general, DGDE3-PS tiene una mejor adaptación a problemas de muchos objetivos con respecto a ambas versiones de DNSGA-II, incluso si estas fueron modificadas para adaptarse a dichos problemas. Una de las posibles razones de esta situación puede deberse al proceso de selección de padres que tiene DGDE3-PS. Su selección se basa en un criterio de aleatoriedad, lo cual permite al algoritmo explorar el espacio de soluciones de mejor manera, mientras que la estrategia de generación de nueva población, junto con PS, se encargan de llevar a cabo una explotación de la región donde se encuentran las soluciones candidatas.

Por otro lado, como ambas versiones DNSGA-II-PS utilizan un torneo binario en base al desempeño de una solución y utilizando un criterio de dominancia de Pareto. El uso de este criterio de dominancia causa que su capacidad de exploración se ve mermada en comparación con el DGDE3-PS. Aún más al enfrentarse a soluciones con muchos objetivos y en constante cambio, ya que puede llevarlo a sufrir un problema de estancamiento en óptimos locales.

7.5.2 Conclusiones

La intención de los experimentos previos es identificar de qué forma se presentan los efectos de utilizar estrategias de búsqueda de soluciones basadas en la dominancia de Pareto en problemas de optimización dinámica con más de tres objetivos. En capítulos anteriores se mostró de forma teórica un ejemplo de las situaciones que puede enfrentar esta estrategia. Sin embargo, para complementar dicha mención se llevó a cabo este experimento para identificar de forma directa los efectos.

Para esto, se comparó el desempeño de algoritmos basados enteramente en la dominancia de Pareto en contra de otro que, además de usar esta dominancia, se apoya de métodos basados en la aleatoriedad. Para la mayoría de las instancias y métricas de evaluación utilizadas, el algoritmo que usó la segunda estrategia presentó mejores resultados que algoritmos basado únicamente en dominancia de Pareto.

De esta forma, se comprueba la existencia de los potenciales problemas que puede presentar el uso de la dominancia y optimalidad de Pareto al resolver problemas de

optimización dinámicos con más de tres objetivos. A partir de estos resultados se puede concluir que el uso de algoritmos con estrategias que no estén basadas en su totalidad en estos criterios puede adaptarse de mejor manera a este tipo de problemas. Por lo tanto, para las siguientes experimentaciones se utilizan DMOEAs con criterios de dominancia diferentes, tal como se hizo mención en las Secciones 5.2 y 6.3.

7.6 Solución de DMaOPs mediante DHHMa

Después de haberse realizado un análisis de los resultados de las experimentaciones realizadas anteriormente, se busca llevar a cabo la solución de DMaOPs mediante el uso de una hiperheurística. En este caso, se busca utilizar una versión de DHH adaptada para resolver este tipo de problemas, denominada DHHMa, los datos correspondientes a su desarrollo se explicaron a detalle en la Sección 5.2. Primero, se realiza una comparativa entre ambas versiones DHHMa propuestas en base los diferentes métodos de selección de LLH utilizados. Después, se analizan los resultados obtenidos por la versión de DHHMa seleccionada con respecto a su LLHs.

7.6.1 Comparación entre métodos de selección

Para esta experimentación se usan dos versiones de DHHMa, cuya diferencia radica en el método de selección de LLH. La primera versión, denominada DHHMa-A para este experimento, utiliza un método voraz en base a la función de elección utilizada por HH_CF [Maashi et al., 2014] y adaptada para problemas de optimización dinámica. Este método únicamente considera la primera parte de la función (c_1), la cual usa únicamente la calidad de los resultados obtenidos bajo diversas métricas en el periodo actual. Para este caso se utilizan como métricas de desempeño R2, ASF y HV, donde $\Theta = R2$ y el valor de α es descartado al no ser necesario definir un balance múltiples criterios, como se presenta en la versión original de la función de elección.

La segunda versión, denominada DHHMa-B, se basa en un método de selección voraz básico, el cual usa únicamente los valores obtenidos por R2 para cada LLH en el periodo actual. La razón de elegir R2 como una métrica principal para ambas versiones es su capacidad de evaluar tanto convergencia como diversidad de las soluciones obtenidas en base a un punto de referencia.

Ambas versiones de DHHMa usan como LLHs una versión dinámica del WASF-GA (DWASF-GA) y dos versiones dinámicas de MOMBI-II (DMOMBI-II-A y DMOMBI-II-B) donde su diferencia radica en el método de adaptación al cambio. La detección de cambios se basa en el número de generaciones transcurridas. El tamaño del vector de pesos y población máxima de cada LLH es de 200 individuos.

Las instancias utilizadas son FDA4 y FDA5, cada una de ellas es probada bajo cuatro, cinco y seis objetivos y ejecutadas 30 veces cada uno. La frecuencia de cambio $\tau_r = 25$ y la severidad de cambio $n_r = 10$. El total de generaciones este definido en $\tau_r * n_r$, dando un total de 250 generaciones para cada DMaOP. La variable de tiempo t utilizada para definir un cambio dentro de estas instancias se basa en la ecuación (4.1).

Los resultados obtenidos son evaluados mediante tres métricas, R2, ASF y HV. Tanto ASF como R2 utilizan un punto de referencia. Teniendo en cuenta que estos DMOEs no están manejando preferencias definidas por un DM y los problemas son de minimización se define el punto de referencia como el vector en el punto de origen, siguiendo la definición propuesta por los autores de los LLHs utilizados.

La intención de esta experimentación es comparar cual alternativa de DHHMa ofrece los mejores resultados. Por dicho motivo la Tabla 7.57 presentan una tabla comparativa en la cual se marca la versión que obtuvo el mejor valor para cada métrica en cada instancia. Los resultados obtenidos se muestran a mayor detalle en el Anexo D.

Tabla 7.57: Revisión de mejor versión por métrica e instancia

Métrica	FDA4-4	FDA4-5	FDA4-6	FDA5-4	FDA5-5	FDA5-6
ASF	DHHMa-B	DHHMa-B	DHHMa-A	DHHMa-A	DHHMa-A	DHHMa-A
R2	DHHMa-B	DHHMa-B	DHHMa-B	DHHMa-B	DHHMa-B	DHHMa-A
HV	DHHMa-A	DHHMa-A	DHHMa-A	DHHMa-A	DHHMa-B	DHHMa-B
Total	DHHMa-A	ASF: 4	R2: 1	HV: 4	SUMA: 9	
	DHHMa-B	ASF: 2	R2: 5	HV: 2	SUMA: 9	

DHHMa-A presenta mejores resultados ligeramente para ASF y HV, lo cual puede significar que esta versión de DHHMa es capaz de obtener soluciones más cercanas al punto de referencia. Sin embargo, los resultados respecto a R2 son ampliamente mejores para DHHMa-B, esto permite suponer que esta versión de la hiperheurística es capaz de no solamente acercarse de manera efectiva al punto de referencia definido, sino también que las soluciones obtenidas presentan una buena diversidad dentro del espacio de búsqueda de funciones objetivo.

La sumatoria de las veces en que es mejor cada versión de DHHMa en cada métrica es idéntica para ambas versiones. Además, en base a los resultados presentados en el Anexo D, no hay una diferencia significativa en ninguno de los resultados. En base a lo anterior es posible determinar cualquiera de las alternativas para utilizar en la comparación con respecto a los LLHs, la cual se realiza en la siguiente sección. En base de esto, y por su simplicidad, se utiliza DHHMa-B para dicha experimentación. Esta versión se considera como DHHMa para la experimentación realizada en la Sección 7.6.2.

7.6.2 Análisis de DHHMa respecto a sus LLHs

Para esta sección se hace una comparación entre los resultados de la versión de DHHMa seleccionada en base a los experimentos realizados en la Sección 7.6.1 y los tres DMOEs utilizados como LLHs ejecutados de manera independiente. En la Tabla 7.58 y 7.59 se presentan los resultados obtenidos por cada algoritmo para ASF y R2, respectivamente. Para determinar existe una diferencia significativa entre los valores de cada algoritmo se lleva a cabo una prueba de rangos alineados de Friedman con $p = 0.05$ y una prueba post-hoc de Holm, de manera similar a la que se ha estado realizando en pruebas anteriores.

Los resultados obtenidos en la Tabla 7.58 permiten generar varias observaciones. En primer lugar, DWASF-GA presenta un buen comportamiento para FDA4 y FDA5 con cuatro a seis objetivos bajo esta ASF. Esta situación se enfatiza aún más cuando ambas instancias

tienen cuatro objetivos, donde la diferencia es significativa sobre todos los demás algoritmos probados. Esto puede significar que DWASF-GA tiene una mejor capacidad de convergencia para estos problemas.

Tabla 7.58. Mediana y desviación estándar fuera de línea para ASF.

DMOP $m = 4$	DHHMa	DWASF-GA	DMOMBI-II-A	DMOMBI-II-B
FDA4	1.375e-2(7.039e-4)*	1.042e-2(4.612e-4)	1.716e-2(1.546e-4)*	1.730e-2(1.195e-4)*
FDA5	1.449e-2(5.735e-4)*	1.089e-2(7.077e-4)	1.662e-2(1.551e-4)*	1.689e-2(1.823e-4)*
$m = 5$				
FDA4	1.825e-3(1.253e-4)	1.659e-3(8.501e-5)	4.195e-3(2.787e-4)*	4.265e-3(2.596e-4)*
FDA5	1.896e-3(1.471e-4)	1.818e-3(9.629e-5)	3.675e-3(2.664e-4)*	3.694e-3(2.486e-4)*
$m = 6$				
FDA4	4.927e-4(5.542e-5)	4.022e-4(7.680e-5)	9.414e-4(4.698e-5)*	9.385e-4(6.883e-5)*
FDA5	4.351e-4(4.472e-5)	4.104e-4(3.576e-5)	6.293e-4(2.579e-5)*	6.517e-4(3.262e-5)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Es necesario notar que el concepto de hiperheurística se mantiene, ya que DHHMa es capaz de utilizar las ventajas de cada uno de sus LLHs para cubrir sus respectivas debilidades. Esto puede verse en esta tabla, ya que para todas las condiciones DHHMaOP se posiciona en segundo lugar, solamente detrás de DWASF-GA y únicamente cuando los DMAOPs manejan cuatro objetivos se presenta una diferencia significativa entre ambos algoritmos.

Por otra parte, DHHMa es mejor de forma estadísticamente significativa en todas las instancias sobre DMOMBI-II-A y DMOMBI-II-B para esta métrica, esto infiere que es capaz de acercarse de mejor forma los resultados al punto de origen que estos dos LLHs.

Tabla 7.59. Mediana y desviación estándar fuera de línea para R2.

DMOP $m = 4$	DHHMa	DWASF-GA	DMOMBI-II-A	DMOMBI-II-B
FDA4	1.805e-2(3.758e-4)	1.878e-2(3.194e-4)*	1.819e-2(1.539e-4)	1.834e-2(1.352e-4)*
FDA5	2.739e-2(3.548e-4)	2.863e-2(6.535e-4)*	2.728e-2(2.213e-4)	2.788e-2(3.195e-4)*
$m = 5$				
FDA4	3.656e-3(1.670e-4)	3.755e-3(2.079e-4)	4.519e-3(2.972e-4)*	4.631e-3(2.724e-4)*
FDA5	5.356e-3(2.418e-4)	5.669e-3(2.501e-4)*	6.138e-3(4.438e-4)*	6.170e-3(4.360e-4)*
$m = 6$				
FDA4	8.668e-4(8.822e-5)	9.414e-4(1.210e-4)*	1.051e-3(5.325e-5)*	1.064e-3(7.787e-5)*
FDA5	1.425e-3(1.655e-4)*	1.379e-3(8.820e-5)*	1.033e-3(3.493e-5)	1.076e-3(4.773e-5)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Por otra parte, en base a los resultados de la Tabla 7.59 es posible ver que DHHMa es mejor que todos los LLHs para FDA4 y en FDA5 con cinco objetivos. Incluso, la hiperheurística llega a ser significativamente superior en varios casos, especialmente respecto a DMOMBI-II-B.

Es necesario notar que para FDA5 con cuatro objetivos DHHMa se encuentra en segundo lugar, Sin embargo, no presenta una desventaja significativa respecto al mejor, que es DMOMBI-II-A. Por otra parte, DHHMa presenta resultados significativamente superiores para FDA4 con cuatro objetivos sobre DWASF-GA y DMOMBI-II-B.

La única instancia donde DHHMa presenta algunos inconvenientes es al momento de resolver FDA5 con seis objetivos, donde DMOMBI-II-A tiene una ventaja significativa respecto a los demás DMOEAs y a la hiperheurística.

Estos resultados permiten suponer que en general DHHMa es capaz de aprovechar las ventajas de cada uno de los LLHs para generar soluciones que son capaces de acercarse al punto de referencia mientras mantiene una diversidad adecuada dentro del espacio de búsqueda de mejor forma que los demás DMOEAs implementados de forma individual para estos dos DMOEs para cuatro a seis objetivos.

Considerando los resultados presentados en estas dos tablas puede concluirse que DHHMa posee una buena habilidad de convergencia respecto al punto de referencia utilizado, que en este caso es el punto de origen, esto puede ser de ayuda para su aplicación en DMOEs preferenciales.

7.6.3 Conclusiones

Se llevaron a cabo los experimentos anteriores con el objetivo de introducir de forma práctica la metodología propuesta dentro de un entorno no solamente dinámico, sino de muchos objetivos. Esto, es un seguimiento de las experimentaciones realizadas en la subsección anterior, en la cual se analizó la capacidad de múltiple DMOEAs para resolver DMOEs.

Para esto, se trabajó con una versión de DHH adaptada para solucionar DMOEs, también, se utilizaron DMOEAs tanto previamente utilizados en problemas de optimización muchos objetivos como aquellos que utilizan una estrategia de búsqueda de soluciones diferente a la dominancia de Pareto.

Los resultados obtenidos muestran que DHHMa es capaz de encontrar soluciones de buena calidad y con un nivel de convergencia a un punto de referencia definido de forma satisfactoria. Esto demuestra la factibilidad y potencial del uso de esta metodología para este tipo de DMOEs.

7.7 Solución de DMOEs preferenciales

La experimentación realizada en esta sección tiene como objetivo encontrar los límites actuales de la hiperheurística propuesta. Para esto, DHHMa es utilizado para resolver DMOEs con preferencias incorporadas por el usuario mediante el uso de método de Separación por Plano. La nueva hiperheurística presentada se denomina DHHMa-PS. El análisis de DHHMa-PS es llevado a cabo en esta sección en tres etapas. Primero, se lleva a cabo una reevaluación de las diversas variables requeridas por PS para identificar la combinación más adecuada para la hiperheurística. Después, se hace una revisión de diversos métodos de selección de LLHs utilizados por DHHMa-PS. Finalmente, se comparan los resultados obtenidos por DHHMa-PS con respecto a sus LLHs ejecutados de forma independiente. Dicha comparación permite ver cuáles son los límites actuales de la hiperheurística y sus posibles áreas de desarrollo para trabajos futuros.

7.7.1 Definición de valores de las variables del método de Separación por Planos

Teniendo en cuenta que este experimento representa la primera introducción de PS a una hiperheurística para solucionar DMOEs y la escasa información dentro de los antecedentes y el estado del arte respecto al uso de estas metodologías para resolver este tipo de problemas,

es necesario llevar a cabo una serie de pruebas preliminares para definir los valores más adecuados en cada una de las variables requeridas por PS. Esta subsección se enfoca en determinar los valores más adecuado para los vectores de expansión por plano R y contribución por plano C .

Para estos experimentos se utiliza una versión de DHHMa la cual incorpora PS dentro de su proceso de una forma similar a la mostrada en el Algoritmo 6.5. Esta hiperheurística, denominada DHHMa-PS utiliza versiones de los LLHs utilizados en el experimento anterior, los cuales, incorporan PS dentro de su proceso (DWASF-GA-PS, DMOMBI-II-A-PS y DMOMBI-II-B-PS). Teniendo las mismas propiedades y características definidas por sus autores en sus respectivos trabajos originales y apoyado en el diseño de dichas metaheurísticas bajo el marco de trabajo jMetal 5.2.

La configuración inicial de DHHMa-PS está basada en las conclusiones obtenidas en las experimentaciones sobre DHH-PS. Por lo tanto, su método de selección de LLHs es la función de elección y su criterio de aceptación es *All Moves*. Sin embargo, es necesario mencionar que debido a que a diferencia de los problemas resueltos por DHH-PS, no se considera el *POF** de los DMAOPs resueltos. Por este motivo, las métricas de desempeño utilizadas deben cambiarse, para DHHMa-PS se utilizan ASF, R2 y HV para la función de elección, siendo R2 la métrica que toma el rol de Θ . El valor de α equivale a τ_r y el valor de c_2 se define como el número de generaciones transcurridas desde la última vez que cada LLH fue seleccionado.

Primero, se lleva a cabo una experimentación para definir el valor del vector del porcentaje de expansión por plano (R). Para esto, se utiliza la misma configuración de PS definida en la Sección 7.5 en tres versiones de DHHMa-PS, donde la diferencia radica en el cambio de los valores de R . Este experimento prueba tres diferentes configuraciones:

- $R_1 = \{ROI, ROI+5\%, ROI+10\% \}$
- $R_2 = \{ROI, ROI+15\%, ROI+30\% \}$
- $R_3 = \{ROI, ROI+25\%, ROI+50\% \}$

Para buscar mantener un balance entre diversificación e intensificación se aumentó el valor de τ_{ps} a 5 generaciones. Se usan instancias de FDA4 y FDA5 con cuatro, cinco y seis objetivos en dos diferentes entornos preferenciales que siguen los valores de W definidos en la Sección 7.5 para evaluar el desempeño de tanto DHHMa-PS como cada uno de sus LLHs ejecutado de forma individual.

La Tabla 7.60 muestra cual versión de DHHMa-PS presenta los mejores resultados para ambas configuraciones preferenciales W en base a ASF, R2 y HV. Cabe mencionar que v_1 y v_2 hacen referencia a $W_{f1} = [0.1-0.4]$ y $W_{f2} = [0.6-0.9]$, respectivamente. Al final, en esa misma tabla, se muestra una suma de las veces en que resulta mejor cada versión por métrica. Esto en base a la consideración de que cada métrica tiene el mismo grado de relevancia.

Los resultados presentados en la Tabla 7.60 muestran un patrón a favor de la configuración definida por R_1 , especialmente para FDA5 para cuatro, cinco y seis objetivos

en ambos entornos preferenciales. La única instancia donde dicha configuración es superada es las pruebas realizadas para FDA4 con cinco objetivos bajo el entorno de v_1 , donde R_3 presenta mejores resultados para cada una de las métricas usadas para evaluar el desempeño.

Tabla 7.60: Revisión de mejor versión por métrica e instancia para ambos entornos

Métrica	FDA4-4		FDA4-5		FDA4-6		FDA5-4		FDA5-5		FDA5-6	
	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2
ASF	R_3	R_1	R_3	R_1	R_1	R_1	R_1	R_1	R_1	R_3	R_1	R_3
R2	R_1	R_3	R_3	R_1	R_1	R_1	R_1	R_1	R_1	R_3	R_1	R_1
HV	R_3	R_1	R_3	R_1	R_3	R_1	R_1	R_1	R_1	R_1	R_3	R_1
Total	R_1		ASF: 8		R2: 9		HV: 8		Suma: 25			
	R_2		ASF: 0		R2: 0		HV: 0		Suma: 0			
	R_3		ASF: 4		R2: 3		HV: 4		Suma: 11			

El motivo por el cual R_1 muestra un mejor desempeño en dichas métricas respecto a las otras configuraciones puede deberse a que la hiperheurística se ve beneficiada si la búsqueda de soluciones cercanas al ROI es más estricta. La misma naturaleza de PS obliga a tanto DMOEAs como hiperheurísticas a explorar una mayor área del espacio de búsqueda de soluciones al dividir la población actual en diferentes planos, estudiando cada uno de forma aislada. Por lo tanto, la reducción del tamaño de los planos puede permitir proporcionar un balance entre la capacidad de exploración de PS y un proceso de intensificación.

Después de evaluar los valores de R , se realizan pruebas con diferentes valores para el vector de contribución por plano (C). Para ello se usa la misma versión de DHHMa-PS usada en la prueba anterior, donde $R = \{ROI, ROI+15\%, ROI+30\%\}$, buscando tener la configuración más parecida posible a la establecida por DHH-PS, lo cual permite evaluar los valores de cada variable por separado.

En este caso, se prueban dos versiones de DHHMa-PS con diferentes valores en C . Ambas versiones consideran tres planos (ROI, cercanas y media distancia), donde el cuarto valor de C corresponde a las soluciones que se encuentran fuera de los límites establecidos por cada uno de los planos (soluciones lejanas).

- $C_1 = \{80\%, 15\%, 3\%, 2\%\}$
- $C_2 = \{40\%, 35\%, 20\%, 5\%\}$

La Tabla 7.61 presenta la configuración que obtiene los mejores resultados para cada DMAOP probado bajo ambos entornos preferenciales. Esta tabla también muestra una sumatoria de las veces en que cada versión de la hiperheurística obtuvo el mejor desempeño por cada métrica de desempeño usada para evaluar los resultados obtenidos.

Tabla 7.61: Revisión de mejor versión por métrica e instancia para ambos entornos

Métrica	FDA4-4		FDA4-5		FDA4-6		FDA5-4		FDA5-5		FDA5-6	
	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2
ASF	C_2	C_1	C_2	C_1	C_2	C_2	C_1	C_2	C_1	C_2	C_1	C_2
R2	C_2	C_2	C_1	C_2	C_2	C_1	C_1	C_2	C_2	C_2	C_1	C_1
HV	C_1	C_2	C_2	C_1	C_2	C_1	C_2	C_2	C_1	C_2	C_2	C_2
Total	C_1		ASF: 5		R2: 5		HV: 4		Suma: 14			
	C_2		ASF: 7		R2: 7		HV: 8		Suma: 22			

Como puede verse, la configuración C_2 presenta mejores resultados para la mayoría de las instancias probadas. Los valores de C_2 enfocan la búsqueda de soluciones en cada uno de los planos utilizados de una forma más balanceada en comparación con los valores de C_1 , los cuales se enfocan mayoritariamente en las soluciones catalogadas dentro del primer plano. Estos resultados concuerdan con las conclusiones llegadas en la definición del valor de R . Permitir una mayor exploración del espacio de búsqueda y reducir el tamaño de los planos generados puede llevar a soluciones con mayor diversificación, pero con una convergencia satisfactoria al punto de referencia. Esto a su vez, permite evitar problemas comunes como el estancamiento en óptimos locales o realizar una búsqueda de soluciones sin rumbo.

En base a los resultados obtenidos se lleva a la conclusión de que la configuración de PS para las ejecuciones de esta sección será la siguiente:

- 1) $W_{f1} = [0.1, 0.4]$.
 - $W_{f2}, W_{f3}, W_{f4}, W_{f5}, W_{f6} = [0.0, 1.0]$.
- 2) $W_{f1} = [0.6, 0.9]$.
 - $W_{f2} = [0.0, 0.8]$.
 - $W_{f3}, W_{f4}, W_{f5}, W_{f6} = [0.0, 1.0]$.
 - $R = \{ROI, ROI+5\%, ROI+10\% \}$
 - $C = \{40\%, 35\%, 20\%, 5\% \}$

7.7.2 Comparación entre métodos de selección

Después de definir los valores de las variables de PS el siguiente paso de esta experimentación es buscar determinar el método de selección de LLHs más adecuado para DHHMa-PS bajo la estructura actual. Para ello se prueban tres diferentes métodos de selección usando estrategias utilizadas en las pruebas llevadas a cabo para resolver DMOPs:

- 1) DHHMa-PS-A: Función de elección, métricas: ASF, R2 y HV, $\Theta = R2$, $\alpha = \tau_r$.
- 2) DHHMa-PS-B: Voraz en base a función de elección, métricas: LRNI, ASF, R2 y HV, $\Theta = R2$.
- 3) DHHMa-PS-C: Ruleta con memoria usando R2 para definir las probabilidades de cada LLH.

Estas tres versiones de DHHMa-PS siguen la configuración de PS previamente definida y el criterio de aceptación de soluciones All Moves. La Tabla 7.62 muestra los resultados obtenidos por cada versión de forma similar a la estructura presentada en las Tablas 7.60 y 7.61, exhibiendo cuál es la versión de DHHMa-PS que obtuvo los mejores resultados para cada DMaOP bajo ASF, R2 y HV. En este caso A, B y C representan a DHHMa-PS-A, DHHMa-PS-B y DHHMa-PS-C, respectivamente.

Los resultados obtenidos muestran una tendencia hacia dos métodos de selección. En específico, DHHMa-PS-C, el cual usa ruleta, presenta mejores resultados mayoritariamente para las instancias FDA4, en especial cuando manejan cuatro y cinco objetivos. Por otra parte, DHHMa-PS-B muestra mejores resultados en general para la mayoría de las métricas de evaluación usadas para la instancia FDA5 para cuatro, cinco y seis objetivos.

Tabla 7.62: Revisión de mejor versión por métrica e instancia

Métrica	FDA4-4		FDA4-5		FDA4-6		FDA5-4		FDA5-5		FDA5-6	
	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2
ASF	C	B	C	C	B	B	B	B	B	B	B	C
R2	C	C	C	C	B	B	C	B	B	B	B	C
HV	C	C	C	B	C	C	C	B	C	B	C	B
Total	DHHMa-PS-A		ASF: 0		R2: 0		HV: 0		SUMA: 0			
	DHHMa-PS-B		ASF: 8		R2: 6		HV: 4		SUMA: 18			
	DHHMa-PS-C		ASF: 4		R2: 6		HV: 8		SUMA: 18			

El método de ruleta presenta mejores resultados cuando se maneja una preferencia cercana a los valores mínimos de f_1 , los cuales son más cercanos al punto de origen. Mientras que, cuando existe una preferencia más cercana a los valores máximos que puede alcanzar f_1 , o sea, más alejados al punto de origen, es el método voraz el cual muestra una mejor capacidad para encontrar soluciones en base a las métricas usadas.

Un aspecto que debe notarse es que DHHMa-PS-A no logra presentar superioridad en ninguna instancia. Esto puede suponer que el uso de la variable c_2 de la ecuación, la cual busca diversificar la selección de LLHs puede afectar el desempeño de la hiperheurística. Esto puede deberse a que PS ya está enfocado a establecer un balance entre intensificación y diversificación, donde la introducción de más elementos que promueven la diversidad provoca una situación en la cual se pierde el rumbo del punto de referencia en la búsqueda de soluciones, llevando a soluciones de menor calidad.

En base los datos presentados en la tabla anterior se abre la posibilidad de utilizar tanto DHHMa-PS-B como DHHMa-PS-C. Con el objetivo de mantener una consistencia con las pruebas en secciones anteriores, se usa la versión DHHMa-PS-B para las comparaciones realizadas en la siguiente subsección. Esta versión de la hiperheurística es denominada DHHMa-PS para el resto de esta sección.

7.7.3 Análisis de DHHMa-PS respecto a sus LLHs

Para esta parte de la sección se lleva a cabo una comparación y análisis de los resultados obtenidos por DHHMa-PS con respecto a los tres DMOEAs utilizados como LLHs ejecutados de forma independiente. Las Tablas 7.63 y 7.64 presentan los valores de ASF y R2 en base a las soluciones obtenidas por cada algoritmo bajo la primera configuración preferencial, respectivamente. Dicha configuración, enfoca el ROI a valores pertenecientes a los valores mínimos de f_1 . En otras palabras, son más cercanos al punto de origen.

Tabla 7.63. Mediana y desviación estándar fuera de línea para ASF con $W_{f1} = [0.1, 0.4]$.

DMOP $m = 4$	DHHMa-PS	DWASF-GA-PS	DMOMBI-II-A-PS	DMOMBI-II-B-PS
FDA4	3.708e-1(1.623e-1)*	1.015e+0(1.692e-1)*	5.853e-2(7.789e-3)	5.836e-2(8.900e-3)
FDA5	1.634e-1(6.211e-2)*	2.282e-1(7.484e-2)*	6.517e-2(4.243e-3)	6.228e-2(6.043e-3)
$m = 5$				
FDA4	2.317e-1(7.756e-2)*	5.598e-1(1.053e-1)*	9.614e-2(9.580e-3)	1.052e-1(1.390e-2)
FDA5	8.938e-2(2.248e-2)	1.597e-1(4.402e-2)*	7.363e-2(5.624e-3)	6.866e-2(3.061e+41)
$m = 6$				
FDA4	1.558e-1(4.988e-2)	2.821e-1(4.309e-2)*	1.259e-1(6.420e-2)	1.435e-1(1.996e-1)*
FDA5	4.877e-2(8.945e-3)	7.149e-2(1.959e-2)*	8.032e-2(3.087e-2)	1.043e-1(1.522e+64)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Los resultados obtenidos por DHHMa-PS y cada LLH para esta configuración permite generar varias observaciones. En primer lugar, en base a los valores vistos en la Tabla 7.63, es necesario mencionar que ambas versiones de DMOMBI-II-PS presentan un mejor desempeño ASF respecto a DHHMa-PS, esto se puede deber a que su búsqueda se basa totalmente en un indicador R2, que utiliza puntos de referencia lo cual dirige la búsqueda de mejor manera al ROI.

También, se debe notar que la calidad de DHHMa-PS para ASF tiende a mejorar conforme aumenta el número de objetivos de las instancias. Incluso en el caso de FDA5 con seis objetivos, los resultados de la hiperheurística son mejores que todos los LLHs e incluso supera significativamente a DWASF-GA-PS y DMOMBI-II-B-PS.

Como anteriormente se mencionó, la efectividad de una hiperheurística es dependiente de la efectividad de sus LLHs para solucionar problemas. La calidad de los resultados de DWASF-GA presenta inconvenientes en base a esta métrica. Estos resultados pueden afectar a la hiperheurística, ya que pueden presentarse casos en los cuales la selección de heurísticas lleve a elegir este LLH y se produzca un estancamiento en un óptimo local, como ocurre con este DMOEA.

Tabla 7.64. Mediana y desviación estándar fuera de línea para R2 con $W_{f1} = [0.1, 0.4]$.

DMOP $m = 4$	DHHMa-PS	DWASF-GA-PS	DMOMBI-II-A-PS	DMOMBI-II-B-PS
FDA4	1.070e-1(3.744e-2)*	3.258e-1(3.693e-2)*	3.909e-2(3.327e-3)	4.001e-2(3.588e-3)
FDA5	4.782e-2(6.384e-3)*	7.070e-2(1.536e-2)*	4.210e-2(1.764e-3)	4.031e-2(3.949e-3)
$m = 5$				
FDA4	9.163e-2(2.429e-2)*	2.003e-1(3.133e-2)*	6.024e-2(3.856e-3)	6.140e-2(5.196e-3)
FDA5	3.751e-2(7.094e-3)	7.514e-2(2.052e-2)*	5.353e-2(2.174e-3)*	4.938e-2(1.123e-2)*
$m = 6$				
FDA4	6.264e-2(8.625e-3)	1.333e-1(1.963e-2)*	7.305e-2(8.699e-3)*	8.354e-2(9.841e-3)*
FDA5	3.150e-2(6.307e-3)	5.257e-2(1.280e-2)*	5.432e-2(3.690e-3)*	7.098e-2(1.638e-2)*

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significante en base a las pruebas de Friedman y Holm.

Los resultados de la Tabla 7.64 muestran que DHHMa-PS, en base a R2, es capaz de encontrar mejores resultados conforme aumenta el número de objetivos en las instancias. Este caso, se presenta especialmente para FDA5, donde la hiperheurística es mejor de manera significativa respecto a todas los LLHs bajo cinco y seis objetivos. También, esta situación se repite al revisar FDA4 con seis objetivos.

La posibilidad de que DWASF-GA-PS sufra de estancamiento en óptimos locales también se muestra bajo esta métrica, ya que en todos los casos sus valores son significativamente inferiores que el resto de los LLHs y DHHMa-PS.

En general, Los resultados presentados en estas dos tablas deja claro que DHHMa-PS puede obtener soluciones de igual o mejor calidad en múltiples casos en comparación a varios MOEAs existentes dentro del estado del arte para estas instancias bajo la condición de que las preferencias sean cercanas al punto de origen. Esto, a su vez, permite entender que la metodología propuesta en esta tesis puede expandirse de manera satisfactoria a DMaOPs.

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

Las Tablas 7.65 y 7.66 presentan los resultados de ASF y R2 bajo la segunda configuración preferencial W utilizada, la cual determina que $W_{f1} = [0.6, 0.9]$. Esto significa que el ROI y el punto de referencia se encuentran cercanos al límite máximo de este objetivo.

Tabla 7.65. Mediana y desviación estándar fuera de línea para ASF con $W_{f1} = [0.6, 0.9]$.

DMOP	DHHMa-PS	DWASF-GA-PS	DMOMBI-II-A-PS	DMOMBI-II-B-PS
$m = 4$				
FDA4	3.625e-1(4.781e+2)*	1.059e+0(1.563e-1)*	3.535e-2(1.402e-3)	3.478e-2(1.439e-3)
FDA5	4.107e-1(3.620e+0)*	1.235e+0(2.532e+0)*	3.249e-2(1.723e-3)	3.115e-2(1.898e-3)
$m = 5$				
FDA4	4.791e-1(9.363e+0)*	7.142e-1(2.231e-1)*	6.348e-2(2.591e-3)	6.264e-2(1.781e-3)
FDA5	4.205e-1(5.132e+30)*	9.183e-1(4.417e+0)*	5.950e-2(1.677e-3)	5.763e-2(1.437e-3)
$m = 6$				
FDA4	9.529e-1(3.723e+2)*	4.494e-1(8.328e-2)*	6.625e-2(1.693e-2)	5.640e-2(5.722e-3)
FDA5	2.927e-1(4.611e+2)*	5.814e-1(2.672e+1)*	6.853e-2(1.763e-2)	6.861e-2(7.198e+92)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

Tabla 7.66. Mediana y desviación estándar fuera de línea para R2 con $W_{f1} = [0.6, 0.9]$.

DMOP	DHHMa-PS	DWASF-GA-PS	DMOMBI-II-A-PS	DMOMBI-II-B-PS
$m = 4$				
FDA4	1.450e-1(4.410e-2)*	5.228e-1(6.611e-2)*	2.992e-2(1.276e-3)	2.933e-2(1.361e-3)
FDA5	1.839e-1(6.268e-2)*	5.384e-1(2.661e-2)*	3.053e-2(1.620e-3)	2.936e-2(1.832e-3)
$m = 5$				
FDA4	1.521e-1(4.571e-2)*	3.634e-1(5.194e-2)*	5.072e-2(2.289e-3)	5.041e-2(1.588e-3)
FDA5	1.587e-1(3.731e-2)*	3.932e-1(3.341e-2)*	4.923e-2(1.642e-3)	4.841e-2(1.545e-3)
$m = 6$				
FDA4	2.148e-1(6.312e-2)*	2.818e-1(3.777e-2)*	5.659e-2(1.706e-2)	4.832e-2(5.199e-3)
FDA5	1.422e-1(1.956e-2)*	2.255e-1(2.188e-2)*	5.095e-2(1.773e-2)	5.215e-2(4.228e-2)

* La diferencia entre este criterio y el de mejor valor (marcado en negrita) es estadísticamente significativa en base a las pruebas de Friedman y Holm.

En base a la Tabla 7.65 se puede observar que ambas versiones de DMOMBI-II-PS se comportan mejor de forma significativa cuando existe una preferencia por parte del DM que se enfoca en los límites superiores de los valores de los objetivos. Esto se debe a que DMOMBI-II-PS tiene una buena capacidad para explorar el espacio de soluciones, lo cual le permite encontrar evitar posibles estancamientos en óptimos locales, como ocurre en el caso de DWASF-GA-PS. En la Tabla 7.66, la cual muestra los valores de R2 de cada algoritmo, se repite la misma situación que se exhibió al analizar los resultados mediante ASF.

DHHMa-PS, en su estado actual, presenta limitaciones al solucionar este tipo de problemas, en este punto, es posible identificar cuáles son las limitantes actuales de la hiperheurística propuesta en este proyecto. Debemos recordar que no existe un algoritmo inmune al teorema de “No Free Lunch”. Sin embargo, la capacidad de la metodología propuesta le permite cubrir un área de problemas mayor que los LLHs aplicados de forma individual (problemas dinámicos, multiobjetivo, muchos objetivos, con preferencias de un DM incorporadas).

Los experimentos en otras fases y en periodos previos muestran que la hiperheurística es capaz de adaptarse a diferentes entornos y configuraciones. Por lo que es factible pensar que es cuestión de una mayor experimentación para descubrir una configuración que funcione de forma completamente efectiva para este entorno

7.7.4 Conclusiones

El objetivo de los experimentos realizados en esta subsección es probar la capacidad y límites actuales de la metodología propuesta en este trabajo mediante el uso DHHMa para solucionar DMaOP con preferencias definidas por DM. Para esto, se llevó a cabo un conjunto de experimentos preliminares para definir los valores de las variables de PS y el método de selección de LLHs más adecuado. Estos experimentos siguen el patrón definido en experimentos pasados.

Después, se comparan y analizan los resultados de DHHMa-PS con respecto a sus LLHs ejecutados de forma individual. El análisis nos muestra de manera exitosa los límites y alcances actuales que posee la metodología propuesta, donde DHHMa-PS parece ser capaz de encontrar buenos resultados en ROIs cercanos al punto de origen y su calidad mejora conforme aumenta el número de objetivos de un DMaOP. Esto permite visualizar el potencial de esta metodología para trabajos futuros relacionados en este campo.

También, es necesario aclarar que las aportaciones y experimentos dentro de toda esta sección son, hasta el conocimiento del autor, la primera vez que hace uso de hiperheurísticas para solucionar DMaOPs preferenciales mediante el uso de DMOEAs. Por lo que estos resultados e información puede utilizarse como un punto de partida para llevar a cabo más investigaciones dentro de este campo.

Por otra parte, los límites actuales de la hiperheurística identificados no son necesariamente el punto de terminación de esta propuesta para futuros trabajos. Debe considerarse que el uso de esta metodología se ha podido extender de forma satisfactoria en diversas áreas de la optimización dinámica en comparativa con otras propuestas, tal como se presenta de forma visual en la Tabla 7.67. Por tal motivo es razonable pensar que los límites actuales pueden ser superados si llevan a cabo estudios a mayor detalle en el futuro.

Tabla 7.67. Capacidad de hiperheurísticas propuestas en comparación con otros algoritmos.

Algoritmo	DMOP	DMOP preferencial	DMaOP	DMaOP preferencial
DNSGA-II	✓	✓		
GDE3*	✓	✓		
R-NSGA-II*		✓		
NOSGA*		✓		
WASF-GA*		✓	✓	
MOMBI-II*			✓	✓
DHH	✓	✓	✓	✓
DPEM-HH	✓	✓	✓	✓

*Adaptados para problemas dinámicos, preferenciales y de muchos objetivos en este proyecto.

Los resultados obtenidos por las hiperheurísticas propuestas en esta tesis y evaluadas en este capítulo muestra su capacidad y utilidad para resolver una amplia variedad de problemas de optimización. El enfoque de esta tesis se dirige a tres propiedades relevantes dentro del campo: dinamismo, manejo de múltiples y muchos objetivos, y el manejo e incorporación de preferencias definidas por un DM. Como muestra la Tabla 7.67, una situación detectada con las metaheurísticas pertenecientes al estado del arte es su enfoque a únicamente un cierto tipo de problemas. Por otra parte, las hiperheurísticas son capaces de aplicarse en diferentes tipos

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

de instancias al ser metodologías enfocadas en revisar dentro del espacio de búsqueda de heurísticas de bajo nivel disponibles y no en el espacio de soluciones. Esto les permite aplicarse con mínimas modificaciones a diferentes tipos de problemas.

A partir de esto es razonable pensar en la utilidad de las hiperheurísticas para la resolución de problemas relacionados a la vida real. Esto es, pensando en situaciones en las cuales los algoritmos pueden suponer diversas opiniones de cómo solucionar un problema, siendo la hiperheurísticas la herramienta encargada de seleccionar la alternativa más viable y con mejores resultados.

Capítulo 8

Conclusiones y trabajo futuro

Esta tesis propuso el desarrollo de una metodología que permita solucionar problemas de optimización dinámicos, con preferencias incorporadas y multiobjetivo. Además de eso se buscó el desarrollo de relaciones entre algoritmos, instancias y métricas de desempeño que permitieran presentar una posible estandarización para este tipo de problemas.

En base a los resultados obtenidos se puede afirmar que dichos objetivos fueron alcanzados de forma satisfactoria, ya que la estructura propuesta presenta una hiperheurística aplicable en problemas de optimización con las características mencionadas previamente. Siendo capaz de adaptarse y usarse para obtener soluciones con una calidad satisfactoria. A continuación, se enlista de forma breve las aportaciones principales de este trabajo, la producción científica y las posibles áreas en las que se detectaron oportunidades para trabajos en futuras investigaciones.

8.3 Trabajos futuros

A continuación, se presentan las áreas de oportunidad con poca o nula exploración identificadas durante esta investigación en los cuales puede llevarse un estudio futuro:

- Incorporación de diferentes DMOEAs como LLHs. Previamente se mencionó que dentro de la literatura existe una variedad de MOEAs y DMOEAs propuestos para solucionar problemas de optimización tanto estáticos como dinámicos. En vista de esta situación es razonable pensar que existen varias metaheurísticas con diferentes enfoques que, por cuestiones de tiempo o el rumbo tomado por esta tesis, no fueron considerados. Esto abre la posibilidad de utilizar algoritmos como LLHs e incorporarlos dentro de la metodología propuesta. Existen múltiples razones por las cuales dirigir un trabajo futuro a este rubro puede ser interesante. Primero, esto permitiría expandir la aplicación de las hiperheurísticas propuestas en otros tipos de problemas. También, hay que considerar que cada nueva investigación puede significar nuevos descubrimientos dentro del campo de optimización dinámica, lo que significa a su vez la apertura de nuevas áreas de investigación dentro del campo.
- Uso de diferentes métodos de selección de LLH y criterios de aceptación de solución. Dentro varios trabajos dentro del estado del arte existen diversos métodos de selección de LLH y criterios de aceptación que no fueron probados dentro de las experimentaciones de esta tesis [Cowling et al., 2000]. La cantidad de propuestas existentes en ambos elementos abre la posibilidad de llevar a cabo un mayor estudio en el cual se revisen diferentes métodos y criterios a los utilizados en los experimentos previamente realizados. Dichas experimentaciones se centraron en el uso de métodos basados en función de elección y aleatoriedad, así como criterios de aceptación deterministas. Debido a esto, una posible área de investigación para trabajos futuros

es el uso de métodos de selección fundamentados en estrategias diferentes, tales como permutaciones o incluso algoritmos como recocido simulado y colonia de hormigas. También, puede estudiarse el efecto del uso de criterios de aceptación no deterministas, tales como ruleta o selección aleatoria, en los resultados obtenidos.

- Pruebas usando diferentes DMOPs y preferencias. En este trabajo se hizo enfoque en tres juegos de instancias DMOP: FDA, dMOP y DMZDT. Estos tres conjuntos proveen diversos retos y propiedades únicas que retan a los algoritmos utilizados a solucionar DMOPs bajo diferentes condiciones. Debe considerarse que dentro de la literatura existen más DMOPs propuestos, tales como los que se revisaron en el Capítulo 3. Además, para este trabajo, los DMOPs considerados no poseen restricciones que vayan más allá de la definición un rango de valores para cada uno de los valores dentro del vector de variables de decisión de cada solución. Un ejemplo de ello son las instancias DCTP [Azzouz et al., 2015], las cuales además de estos límites incluyen restricciones adicionales, las cuales son susceptibles a cambios durante el transcurso del tiempo. Otra área pendiente por explorar por la propuesta presentada en esta tesis es su aplicación en problemas reales. Para este caso pueden considerarse simulaciones de problemas reales de control [Farina et al., 2004], programación de actividades y manejo de recursos [Deb et al., 2007], entre otros. Estos problemas sufren cambios en diversos momentos del proceso de solución. El uso de una hiperheurística para este tipo de soluciones puede ser atractivo, ya que permite la posibilidad de definir múltiples algoritmos como opciones para solucionar un problema. Mientras, la hiperheurística se enfoca en seleccionar aquella que sea más adecuada en el entorno actual. Estos problemas también pueden considerar ciertas preferencias definidas por un DM que no sean necesariamente las mismas definidas en este trabajo. Todo lo anterior presenta la posibilidad de llevar a cabo una extensión del estudio realizado en esta tesis en instancias de prueba no exploradas aún o dirigir el uso de la metodología propuesta hacia instancias que simulan situaciones reales.
- Extensión del uso de las hiperheurísticas propuestas en DMAOPs. La propuesta experimentada en este capítulo para solucionar DMAOP se enfoca en el uso de metaheurísticas basadas en funciones de escalarización de logros y puntos de referencia como LLHs. Es posible considerar que el uso de LLHs que manejan diferentes estrategias, tales como la superación (NOSGA-II), puede proveer nuevos resultados e información relevante respecto al impacto de la hiperheurística en la resolución de DMAOPs. De igual forma que en los puntos anteriores, un posible trabajo futuro puede enfocarse en la extensión del estudio actual sobre DMAOPs mediante el uso de diferentes metaheurísticas usadas como LLHs a las que fueron usadas para esta experimentación.

Referencias

- [Adra et al., 2007] Adra, S. F., Griffin, I., and Fleming, P. J. (2007). A comparative study of progressive preference articulation techniques for multiobjective optimisation. In Proceedings of the 4th international conference on Evolutionary Multi-criterion Optimization (EMO'07), Springer, Matsushima, Japan, 908–921.
- [Altenberg, 2014] Altenberg, L. (1994). The evolution of evolvability in genetic programming. *Advances in genetic programming*, 3, 47-74.
- [Azzouz et al., 2014] Azzouz, R., Bechikh, S., & Said, L. B. (2014, July). A multiple reference point-based evolutionary algorithm for dynamic multi-objective optimization with undetectable changes. In 2014 IEEE Congress on Evolutionary Computation (CEC) (pp. 3168-3175). IEEE.
- [Azzouz et al., 2015] Azzouz, R., Bechikh, S., & Ben Said, L. (2015, July). Multi-objective optimization with dynamic constraints and objectives: new challenges for evolutionary algorithms. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation* (pp. 615-622).
- [Azzouz et al., 2017] Azzouz, R., Bechikh, S., & Said, L. B. (2017). Dynamic Multi-objective Optimization Using Evolutionary Algorithms: A Survey. In *Recent Advances in Evolutionary Multi-objective Optimization* (pp. 31-70). Springer International Publishing.
- [Ayob y Kendall, 2003] Ayob, M., & Kendall, G. (2003, December). A monte carlo hyper-heuristic to optimise component placement sequencing for multi head placement machine. In *Proceedings of the international conference on intelligent technologies, InTech* (Vol. 3, pp. 132-141).
- [Bader y Zitzler, 2011] Bader, J., & Zitzler, E. (2011). HypE: An algorithm for fast hypervolumE-based many-objective optimization. *Evolutionary computation*, 19(1), 45-76.
- [Bai, 2005] Bai, R. (2005). An investigation of novel approaches for optimising retail shelf space allocation (Doctoral dissertation, University of Nottingham).
- [Baykasoğlu y Ozsoydan, 2017] Baykasoğlu, A., & Ozsoydan, F. B. (2017). Evolutionary and population-based methods versus constructive search strategies in dynamic combinatorial optimization. *Information Sciences*, 420, 159-183.
- [Baykasoğlu y Ozsoydan, 2018] Baykasoğlu, A., & Ozsoydan, F. B. (2018). Dynamic optimization in binary search spaces via weighted superposition attraction algorithm. *Expert Systems with Applications*, 96, 157-174.
- [Bechikh, 2013] Bechikh, S. (2013). Incorporating Decision Maker's Preference Information in Evolutionary Multi-objective Optimization. Diss. PhD thesis, High Institute of Management of Tunis, University of Tunis, Tunisia. <http://delta.cs.cinvestav.mx/~ccoello/EMOO/thesisbechikh.pdf.gz>.
- [Bechikh et al., 2015] Bechikh, S., Kessentini, M., Said, L. B., & Ghédira, K. (2015). Preference incorporation in evolutionary multiobjective optimization: A survey of the state-of-the-art. In *Advances in Computers* (Vol. 98, pp. 141-207). Elsevier.
- [Bilgin et al., 2006] Bilgin, B., Özcan, E., & Korkmaz, E. E. (2006, August). An experimental study on hyper-heuristics and exam timetabling. In *International Conference on the Practice and Theory of Automated Timetabling* (pp. 394-412). Springer, Berlin, Heidelberg.
- [Borges y Hansen, 1998] Borges, P. C., & Hansen, M. P. (1998). A basis for future successes in multiobjective combinatorial optimization. Technical Report, Department of Mathematical Modelling, Technical University of Denmark.

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

[Burke et al., 2009] Burke, E. K., Hyde, M. R., Kendall, G., Ochoa, G., Ozcan, E., & Woodward, J. R. (2009). Exploring hyper-heuristic methodologies with genetic programming. In *Computational intelligence* (pp. 177-201). Springer Berlin Heidelberg.

[Burke et al., 2010] Burke, E. K., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., & Woodward, J. R. (2010). A classification of hyper-heuristic approaches. In *Handbook of metaheuristics* (pp. 449-468). Springer US.

[Cámara et al., 2007] Cámara, M., Ortega, J., & Toro, F. J. (2007, March). Parallel processing for multi-objective optimization in dynamic environments. In *2007 IEEE International Parallel and Distributed Processing Symposium* (pp. 1-8). IEEE.

[Cámara et al., 2008] Cámara, M., Ortega, J., & de Toro, F. (2008). Parallel multi-objective optimization evolutionary algorithms in dynamic environments. In *Proceedings of the first international workshop on parallel architectures and bioinspired algorithms* (Vol. 1, pp. 13-20).

[Castro y Pozo, 2014] Castro, O. R., & Pozo, A. (2014, December). A MOPSO based on hyper-heuristic to optimize many-objective problems. In *2014 IEEE Symposium on Swarm Intelligence* (pp. 1-8). IEEE.

[Chankong y Haimes, 2008] Chankong, V., & Haimes, Y. Y. (2008). *Multiobjective decision making: theory and methodology*. Courier Dover Publications.

[Chen et al., 2017] Chen, Y., Cowling, P., Polack, F., Remde, S., & Mourdjis, P. (2017). Dynamic optimisation of preventative and corrective maintenance schedules for a large scale urban drainage system. *European journal of operational research*, 257(2), 494-510.

[Chernoff, 1987] Chernoff, H. (1987). *Elementary Decision Theory*. Dover Publications.

[Cohon y Marks, 1975] Cohon, J. L., & Marks, D. H. (1975). A review and evaluation of multiobjective programming techniques. *Water Resources Research*, 11(2), 208-220.

[Cowling et al., 2000] Cowling, P., Kendall, G., & Soubeiga, E. (2000, August). A hyperheuristic approach to scheduling a sales summit. In *International Conference on the Practice and Theory of Automated Timetabling* (pp. 176-190). Springer, Berlin, Heidelberg.

[Cowling y Chakhlevitch, 2003] Cowling, P., & Chakhlevitch, K. (2003, December). Hyperheuristics for managing a large collection of low level heuristics to schedule personnel. In *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on* (Vol. 2, pp. 1214-1221). IEEE.

[Deb y Agrawal, 1995] Deb, K., & Agrawal, R. B. (1995). Simulated binary crossover for continuous search space. *Complex Systems*, 9(3), 1-15.

[Deb et al., 2001] Deb, K., Pratap, A., & Meyarivan, T. (2001, March). Constrained test problems for multi-objective evolutionary optimization. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 284-298). Springer Berlin Heidelberg.

[Deb et al., 2002a] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2), 182-197.

[Deb et al., 2002b] Deb, K., Thiele, L., Laumanns, M., & Zitzler, E. (2002, May). Scalable multi-objective optimization test problems. In *Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on* (Vol. 1, pp. 825-830). IEEE.

[Deb y Sundar, 2006] Deb, K., & Sundar, J. (2006, July). Reference point based multi-objective optimization using evolutionary algorithms. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation* (pp. 635-642). ACM.

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

[Deb et al., 2007] Deb, K., Rao, U., & Karthik, S. (2007, March). Dynamic multi-objective optimization and decision-making using modified NSGA-II: a case study on hydro-thermal power scheduling. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 803-817). Springer Berlin Heidelberg.

[Dowsland et al., 2007] Dowsland, K. A., Soubeiga, E., & Burke, E. (2007). A simulated annealing based hyperheuristic for determining shipper sizes for storage and transportation. *European Journal of Operational Research*, 179(3), 759-774.

[Doyle, 2004] Doyle, J. (2004). Prospects for preferences. *Computational Intelligence*, 20(2):111–136.

[Edgeworth, 1881] Edgeworth, F. (1881). *Mathematical Psychics*. P. Keagan, London, England.

[Farina et al., 2004] Farina, M., Deb, K., & Amato, P. (2004). Dynamic multiobjective optimization problems: test cases, approximations, and applications. *IEEE Transactions on evolutionary computation*, 8(5), 425-442.

[Fernández et al. 2011a] Fernández, E., López, E., Navarro, J., Vega, I. (2011): “Aplicación de metaheurísticas multiobjetivo a la solución de problemas de cartera de proyectos públicos con una valoración multidimensional de su impacto”, *Gestión y Política Pública*, vol. XX, no. 2, 2011, pp. 381- 432.

[Fernández et al., 2011b] Fernandez, E., Lopez, E., Lopez, F., & Coello, C. A. C. (2011). Increasing selective pressure towards the best compromise in evolutionary multiobjective optimization: The extended NOSGA method. *Information Sciences*, 181(1), 44-56.

[Fritsche y Pozo, 2019] Fritsche, G., & Pozo, A. (2019, July). Cooperative based hyper-heuristic for many-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference* (pp. 550-558).

[García, 2010] García, R.: *Hiper-heurístico para Resolver el Problema de Cartera de Proyectos Sociales*. Tesis de Maestría, Instituto Tecnológico de Cd. Madero (2010).

[Garrett y Dasgupta, 2007] Garrett, D., & Dasgupta, D. (2007, December). Multiobjective landscape analysis and the generalized assignment problem. In *International Conference on Learning and Intelligent Optimization* (pp. 110-124). Springer Berlin Heidelberg.

[Garrido & Riff, 2010] Garrido, P., & Riff, M. C. (2010). DVRP: a hard dynamic combinatorial optimisation problem tackled by an evolutionary hyper-heuristic. *Journal of Heuristics*, 16(6), 795-834.

[Gendreau y Potvin, 2005] Gendreau, M., y Potvin, J-Y. (2005). Meta-heuristics in combinatorial optimization, *Annals of Operations Research* 140 (1) 189-213.

[Goh y Tan, 2009] Goh, C. K., & Tan, K. C. (2009). A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 13(1), 103-127.

[Gökçe et al., 2017] Gökçe, M. A., Beygo, B., & Ekmekçi, T. (2017). A Hyperheuristic Approach for Dynamic Multilevel Capacitated Lot Sizing with Linked Lot Sizes for APS implementations. *Journal of Yaşar University*, 12(45), 1-13.

[Goldberg, 1989] Goldberg, E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Reading, Massachusetts.

[Grobler et al., 2010] Grobler, J., Engelbrecht, A. P., Kendall, G., & Yadavalli, V. S. S. (2010, July). Alternative hyper-heuristic strategies for multi-method global optimization. In *IEEE Congress on Evolutionary Computation* (pp. 1-8). IEEE.

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

[Guizzo et al., 2017] Guizzo, G., Vergilio, S. R., Pozo, A. T., & Fritsche, G. M. (2017). A multi-objective and evolutionary hyper-heuristic applied to the integration and test order problem. *Applied Soft Computing*, 56, 331-344.

[Hansen y Jaskiewicz, 1994] Hansen, M. P., & Jaskiewicz, A. (1994). *Evaluating the quality of approximations to the non-dominated set*. IMM, Department of Mathematical Modelling, Technical University of Denmark.

[Hatzakis y Wallace, 2006] Hatzakis, I., & Wallace, D. (2006, July). Dynamic multi-objective optimization with evolutionary algorithms: a forward-looking approach. In Proceedings of the 8th annual conference on Genetic and evolutionary computation (pp. 1201-1208). ACM.

[Helbig, 2012] Helbig, M. (2012). Solving dynamic multi-objective optimisation problems using vector evaluated particle swarm optimisation (Doctoral dissertation, University of Pretoria).

[Helbig y Engelbrecht, 2013] Helbig, M., & Engelbrecht, A. P. (2013, September). Challenges of Dynamic Multi-objective Optimisation. In 2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence (pp. 254-261). IEEE.

[Helbig y Engelbrecht, 2014] Helbig, M., & Engelbrecht, A. P. (2014). Benchmarks for dynamic multi-objective optimisation algorithms. *ACM Computing Surveys (CSUR)*, 46(3), 37.

[Helbig y Engelbrecht, 2016] Helbig, M., Deb, K., & Engelbrecht, A. (2016, July). Key challenges and future directions of dynamic multi-objective optimisation. In *2016 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1256-1261). IEEE.

[Hernández-Gómez y Coello, 2015] Hernández Gómez, R., & Coello Coello, C. A. (2015, July). Improved metaheuristic based on the R2 indicator for many-objective optimization. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation* (pp. 679-686).

[Hernández-Gómez y Coello, 2017] Hernández-Gómez, R., & Coello, C. A. C. (2017, July). A hyper-heuristic of scalarizing functions. In *Proceedings of the Genetic and Evolutionary Computation Conference* (pp. 577-584).

[Hodges y Lehmann] Hodges, J. L., & Lehmann, E. L. (1962). Rank methods for combination of independent experiments in analysis of variance. *The Annals of Mathematical Statistics*, 33(2), 482-497.

[Holm, 1979] Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, 65-70.

[Ishibuchi et al., 2008] Ishibuchi, H., Tsukamoto, N., & Nojima, Y. (2008, June). Evolutionary many-objective optimization: A short review. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)* (pp. 2419-2426). IEEE.

[Jakubowski-Filho et al., 2018a] Jakubowski-Filho, H. L., Ferreira, T. N., & Vergilio, S. R. (2018, July). Incorporating User Preferences in a Software Product Line Testing Hyper-Heuristic Approach. In *2018 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1-8). IEEE.

[Jakubowski-Filho et al., 2018b] Jakubowski-Filho, H. L., Ferreira, T. N., & Vergilio, S. R. (2018, September). Multiple objective test set selection for software product line testing: evaluating different preference-based algorithms. In *Proceedings of the XXXII Brazilian Symposium on Software Engineering* (pp. 162-171). ACM.

[Jin y Sendhoff, 2004] Jin, Y., & Sendhoff, B. (2004, April). Constructing dynamic optimization test problems using the multi-objective optimization concept. In Workshops on Applications of Evolutionary Computation (pp. 525-536). Springer Berlin Heidelberg.

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

[Kaelbling et al., 1996] Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4, 237-285.

[Keeney y Raiffa, 1993], Keeney, R.L., and Raiffa, H. (1993) Decisions with Multiple Objectives: Preferences and Value Tradeoffs. Cambridge University Press.

[Kilby et al., 1998] Kilby, P., Prosser, P., & Shaw, P. (1998). Dynamic VRPs: A study of scenarios. *University of Strathclyde Technical Report*, 1-11.

[Kiraz et al., 2013] Kiraz, B., Etaner-Uyar, A. Ş., & Özcan, E. (2013, April). An ant-based selection hyper-heuristic for dynamic environments. In *European Conference on the Applications of Evolutionary Computation* (pp. 626-635). Springer, Berlin, Heidelberg.

[Kukkonen y Lampinen, 2005] Kukkonen, S., & Lampinen, J. (2005, September). GDE3: The third evolution step of generalized differential evolution. In *2005 IEEE congress on evolutionary computation* (Vol. 1, pp. 443-450). IEEE.

[Leung y Wang, 2003] Leung, Y. W., & Wang, Y. (2003). U-measure: a quality measure for multiobjective programming. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 33(3), 337-343.

[Li y Deb, 2017] Li, H., & Deb, K. Challenges for Evolutionary Multiobjective Optimization Algorithms for Solving Variable-length Problems.

[Li et al., 2012] Li, K., Kwong, S., Cao, J., Li, M., Zheng, J., Shen, R. (2012). Achieving balance between proximity and diversity in multi-objective evolutionary algorithm, *Information Sciences* 182 (1) 220–242.

[Li et al., 2017] Li, X., Zeng, S., Li, C., & Ma, J. (2017). Many-objective optimization with dynamic constraint handling for constrained optimization problems. *Soft Computing*, 21(24), 7435-7445.

[Liu y Wang, 2006] Liu, C. A., & Wang, Y. (2006, September). New evolutionary algorithm for dynamic multiobjective optimization problems. In *International Conference on Natural Computation* (pp. 889-892). Springer Berlin Heidelberg.

[Liu, 2010] Liu, C. A. (2010, June). New dynamic multiobjective evolutionary algorithm with core estimation of distribution. In *Electrical and Control Engineering (ICECE), 2010 International Conference on* (pp. 1345-1348). IEEE.

[Liu et al., 2010] Liu, R., Zhang, W., Jiao, L., Liu, F., & Ma, J. (2010, July). A spherE-dominance based preference immunE-inspired algorithm for dynamic multi-objective optimization. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation* (pp. 423-430). ACM.

[Maashi et al., 2014] Maashi, M., Özcan, E., & Kendall, G. (2014). A multi-objective hyper-heuristic based on choice function. *Expert Systems with Applications*, 41(9), 4475-4493.

[Macias-Escobar et al., 2019] Macias-Escobar, T., Cruz-Reyes, L., Dorronsoro, B., Fraire-Huacuja, H., Rangel-Valdez, N., & Gómez-Santillán, C. (2019). Application of population evolvability in a hyper-heuristic for dynamic multi-objective optimization. *Technological and Economic Development of Economy*, 25(5), 951-978.

[Macias-Escobar et al., 2020a] Macias-Escobar, T., Dorronsoro, B., Cruz-Reyes, L., Rangel-Valdez, N., & Gómez-Santillán, C. (2020). A Survey of Hyper-heuristics for Dynamic Optimization Problems. In *Intuitionistic and Type-2 Fuzzy Logic Enhancements in Neural and Optimization Algorithms: Theory and Applications* (pp. 463-477). Springer, Cham.

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

[Macias-Escobar et al., 2020b] Macias-Escobar, T., Cruz-Reyes, L., Fraire, H., & Dorronsoro, B. (2020). Plane Separation: A method to solve dynamic multi-objective optimization problems with incorporated preferences. *Future Generation Computer Systems*, 110, 864-875.

[Martello y Toth, 1990] Martello, S., & Toth, P. (1990). Knapsack problems: algorithms and computer implementations. *Wiley-Interscience series in discrete mathematics and optimization*.

[Nebro et al., 2009] Nebro, A. J., Durillo, J. J., Luna, F., Dorronsoro, B., & Alba, E. (2009). MOCcell: A cellular genetic algorithm for multiobjective optimization. *International Journal of Intelligent Systems*, 24(7), 726-746.

[Nebro et al., 2018] Nebro, A. J., Ruiz, A. B., Barba-González, C., García-Nieto, J., Luque, M., & Aldana-Montes, J. F. (2018). InDM2: Interactive dynamic multi-objective decision making using evolutionary algorithms. *Swarm and Evolutionary Computation*, 40, 184-195.

[Osyczka, 1985] Osyczka, A. (1985). Multicriteria optimization for engineering design. *Design optimization*, 1, 193-227.

[Özcan et al., 2009] Ozcan, E., Uyar, S. E., & Burke, E. (2009, July). A greedy hyper-heuristic in dynamic environments. In *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers* (pp. 2201-2204). ACM.

[Oztürk et al., 2005] Oztürk, M., Tsoukiàs, A., and Ph. Vincke. (2005). Preference Modeling, pages 27–72. In: *State of the Art in Multiple Criteria Decision Analysis*, M. Ehrgott, S. Greco and J. Figueira (Ed.). Wiley Series on Intelligent Systems. Springer-Verlag.

[Pareto, 1896] Pareto, V. (1896). *Cours d'économie politique*, volume 1 & 2. Lausanne.

[Pérez et al., 2007] Pérez, F. Molina, J., Caballero, R., Coello, C., Hernández, A., (2007). Hibridación de métodos exactos y heurísticos para el problema multiobjetivo. *Journal Economic Literature*: C61; C63. XV Jornadas de ASEPUMA y III Encuentro Internacional.

[Raghavjee y Pillay, 2015] Raghavjee, R., & Pillay, N. (2015). A genetic algorithm selection perturbative hyper-heuristic for solving the school timetabling problem. *ORiON*, 31(1), 39-60.

[Rao, 2016] Rao, R. (2016). Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. *International Journal of Industrial Engineering Computations*, 7(1), 19-34.

[Remde et al., 2009] Remde, S., Dahal, K., Cowling, P., & Colledge, N. (2009, April). Binary exponential back off for tabu tenure in hyperheuristics. In *European Conference on Evolutionary Computation in Combinatorial Optimization* (pp. 109-120). Springer, Berlin, Heidelberg.

[Richter, 2013] Richter, H. (2013). Dynamic fitness landscape analysis. In *Evolutionary computation for dynamic optimization problems* (pp. 269-297). Springer Berlin Heidelberg.

[Rivera et al., 2012] Rivera, G., Gómez, C., Cruz, L., García, R., Balderas, F. A., Fernández, E. R., & López, F. (2012). Solution to the social portfolio problem by evolutionary algorithms. *International Journal*, 21-30.

[Roubens y Vincke, 1985] Roubens, M., and Ph. Vincke. (1985). *Preference modeling*. Springer-Verlag.

[Roy y Mehnen, 2008] Roy, R., & Mehnen, J. (2008). Dynamic multi-objective optimisation for machining gradient materials. *CIRP Annals-Manufacturing Technology*, 57(1), 429-432.

[Ruiz et al., 2015] Ruiz, A. B., Saborido, R., & Luque, M. (2015). A preference-based evolutionary algorithm for multiobjective optimization: the weighting achievement scalarizing function genetic algorithm. *Journal of Global Optimization*, 62(1), 101-129.

[Sabar et al., 2015] Sabar, N. R., Ayob, M., Kendall, G., & Qu, R. (2015). Automatic design of a hyper-heuristic framework with gene expression programming for combinatorial optimization problems. *IEEE Trans. Evolutionary Computation*, 19(3), 309-325.

[Sánchez, 2007] Sánchez, P. (2007). Modelos para la combinación de preferencias en toma de decisiones: herramientas y aplicaciones. Tesis Doctoral, Universidad de Granada.

[Santiago et al., 2019] Santiago, A., Dorronsoro, B., Nebro, A. J., Durillo, J. J., Castillo, O., & Fraire, H. J. (2019). A novel multi-objective evolutionary algorithm with fuzzy logic based adaptive selection of operators: FAME. *Information Sciences*, 471, 233-251.

[Schiavinotto y Stützle, 2007] Schiavinotto, T., & Stützle, T. (2007). A review of metrics on permutations for search landscape analysis. *Computers & operations research*, 34(10), 3143-3153.

[Shen y Yao, 2015] Shen, X. N., & Yao, X. (2015). Mathematical modeling and multi-objective evolutionary algorithms applied to dynamic flexible job shop scheduling problems. *Information Sciences*, 298, 198-224.

[Sierra y Coello, 2005] Sierra, M. R., & Coello, C. A. C. (2005, March). Improving PSO-based multi-objective optimization using crowding, mutation and ϵ -dominance. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 505-519). Springer Berlin Heidelberg.

[Smith et al., 2002] Smith, T., Husband, P., & O'Shea, M. (2002). Fitness landscapes and evolvability. *Evolutionary computation*, 10(1), 1-34.

[Soubeiga, 2003] Soubeiga, E. (2003). Development and application of hyperheuristics to personnel scheduling (Doctoral dissertation, University of Nottingham).

[Talbi, 2009] Talbi, E. G. (2009). *Metaheuristics: from design to implementation* (Vol. 74). John Wiley & Sons.

[Tan et al., 2002] Tan, K. C., Lee, T. H., & Khor, E. F. (2002). Evolutionary algorithms for multi-objective optimization: Performance assessments and comparisons. *Artificial intelligence review*, 17(4), 251-290.

[Tan et al., 2008] Tan, K. C., Goh, C. K., Mamun, A. A., & Ei, E. Z. (2008). An evolutionary artificial immune system for multi-objective optimization. *European Journal of Operational Research*, 187(2), 371-392.

[Topcuoglu et al., 2014] Topcuoglu, H. R., Ucar, A., & Altin, L. (2014). A hyper-heuristic based framework for dynamic optimization problems. *Applied Soft Computing*, 19, 236-251.

[Trautmann et al., 2013] Trautmann, H., Wagner, T., Biermann, D., & Weihs, C. (2013). Indicator-based selection in evolutionary multiobjective optimization algorithms based on the desirability index. *Journal of Multi-Criteria Decision Analysis*, 20(5-6), 319-337.

[Stockt y Engelbrecht, 2018] van der Stockt, S. A., & Engelbrecht, A. P. (2018). Analysis of selection hyper-heuristics for population-based meta-heuristics in real-valued dynamic optimization. *Swarm and evolutionary computation*, 43, 127-146.

[Uludağ et al., 2013] Uludağ, G., Kiraz, B., Etaner-Uyar, A. Ş., & Özcan, E. (2013). A hybrid multi-population framework for dynamic environments combining online and offline learning. *Soft Computing*, 17(12), 2327-2348.

[Van Veldhuizen, 1999] Van Veldhuizen, D. A. (1999). Multiobjective evolutionary algorithms: classifications, analyses, and new innovations (No. AFIT/DS/ENG/99-01). AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING.

Optimización de problemas dinámicos en presencia de un gran número de objetivos y preferencias del decisor

[Walker y Keedwell, 2016] Walker, D. J., & Keedwell, E. (2016, September). Towards many-objective optimisation with hyper-heuristics: identifying good heuristics with indicators. In *International Conference on Parallel Problem Solving from Nature* (pp. 493-502). Springer, Cham.

[Wang y Li, 2010] Wang, Y., & Li, B. (2010). Multi-strategy ensemble evolutionary algorithm for dynamic multi-objective optimization. *Memetic Computing*, 2(1), 3-24.

[Wang et al., 2009] Wang, H., Wang, D., & Yang, S. (2009). A memetic algorithm with adaptive hill climbing strategy for dynamic optimization problems. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 13(8), 763-780.

[Wang et al., 2017] Wang, M., Li, B., Zhang, G., & Yao, X. (2017). Population Evolvability: Dynamic Fitness Landscape Analysis for Population-based Metaheuristic Algorithms. *IEEE Transactions on Evolutionary Computation*.

[Wierzbicki, 1979] Wierzbicki, A. P. (1979). The use of reference objectives in multiobjective optimization-theoretical implications and practical experience

[Wilcoxon, 1992] Wilcoxon, F. (1992). Individual comparisons by ranking methods. In *Breakthroughs in statistics* (pp. 196-202). Springer, New York, NY.

[Wolpert y Macready, 1997] Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1), 67-82.

[Yang y Yao, 2008] Yang, S., & Yao, X. (2008). Population-based incremental learning with associative memory for dynamic environments. *IEEE Transactions on Evolutionary Computation*, 12(5), 542-561.

[Zamli et al., 2017] Zamli, K. Z., Din, F., Kendall, G., & Ahmed, B. S. (2017). An experimental study of hyper-heuristic selection and acceptance mechanism for combinatorial t-way test suite generation. *Information Sciences*, 399, 121-153.

[Zheng, 2007] Zheng, B. (2007). A new dynamic multi-objective optimization evolutionary algorithm. In: *Proceedings of the Third International Conference on Natural Computation*, pp. 565–570

[Zitzler y Thiele, 1998] Zitzler, E., & Thiele, L. (1998, September). Multiobjective optimization using evolutionary algorithms—a comparative case study. In *International conference on parallel problem solving from nature* (pp. 292-301). Springer, Berlin, Heidelberg.

[Zitzler y Thiele, 1999] Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE transactions on Evolutionary Computation*, 3(4), 257-271.

[Zitzler et al., 2000] Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, 8(2), 173-195.

[Zitzler et al., 2001] Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm. *TIK-report*, 103.

[Zou et al., 2008] Zou, X., Chen, Y., Liu, M., & Kang, L. (2008). A new evolutionary algorithm for solving many-objective optimization problems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(5), 1402-1412.

Anexo A. Experimentaciones para definir valor de variables en Separación por Planos

Primero, se llevó a cabo una experimentación para definir los parámetros del porcentaje de expansión R y el porcentaje de contribución C para cada plano. Para ello se definieron múltiples configuraciones probando valores extremos para R . Dichas configuraciones se presentan en la Tabla A.1.

Tabla A.1. Configuraciones de porcentajes de expansión y contribución.

Configuración	C por plano	R por plano
cr1	{100, 0, 0, 0}	{0%, 15%, 30%}
cr2	{100, 0, 0, 0}	{0%, 10%, 20%}
cr3	{100, 0, 0, 0}	{0%, 5%, 10%}
cr4	{60, 25, 10, 5}	{0%, 15%, 30%}
cr5	{60, 25, 10, 5}	{0%, 10%, 20%}
cr6	{60, 25, 10, 5}	{0%, 5%, 10%}

Cada una estas configuraciones fueron probadas en DNSGA-II-PS para todos los DMOPs seleccionados. IGD, MS y HVR fueron utilizados para comparar los resultados obtenidos por cada configuración. Para cada DMOP se probaron tres diferentes combinaciones de valores asignados en W . Dichos valores se asignaron teniendo en cuenta que se enfocaran en zonas factibles del espacio de búsqueda:

- 1) $f1 = [0.1, 0.4]$.
 $f2 = [0.3, 0.7]$ (FDA1, FDA3 y dMOP3).
 $f2 = [0.65, 1.0]$ (dMOP1, dMOP2).
- 2) $f1 = [0.4, 0.6]$.
 $f2 = [0.2, 0.4]$ (FDA1, FDA3 y dMOP3).
 $f2 = [0.5, 0.85]$ (dMOP1, dMOP2).
- 3) $f1 = [0.6, 0.9]$.
 $f2 = [0.05, 0.3]$ (FDA1, FDA3 y dMOP3).
 $f2 = [0.1, 0.65]$ (dMOP1, dMOP2).

En la Tabla A.2 se presentan los resultados en base al número de veces que cada configuración obtuvo los mejores valores para cada una de las métricas. La configuración cr4 generó los mejores resultados para cada una las métricas, por lo que se concluyó que R se asigna en {0%, 15%, 30%}. En otras palabras, cada plano tiene un 15% extra de expansión respecto al ROI.

Tabla A.2. Sumatoria por configuración de mejor valor por métrica en DNSGA-II-PS.

	cr1	cr2	cr3	cr4	cr5	cr6
IGD	0	0	1	9	1	4
MS	0	0	0	7	4	4
HVR	0	1	2	6	1	5
Total	0	1	3	22	6	13

Los resultados también muestran que una contribución del 60% del plano ROI y 40% de soluciones los demás planos (25% Cercanos, 10% Media Distancia, 5% Lejanos) presenta

los mejores resultados. Se expandió la revisión probando valores C aledaños, las configuraciones probadas fueron $cr1 - \{40, 35, 20, 5\}$, $cr2 - \{50, 30, 15, 5\}$, $cr3 - \{60, 25, 10, 5\}$, $cr4 - \{70, 20, 6, 4\}$ y $cr5 - \{80, 15, 3, 2\}$. Los resultados, siguiendo el patrón de la tabla anterior se presentan en la Tabla A.3. Se observa que $C = \{80, 15, 3, 2\}$ provee los mejores resultados. Por lo tanto, se le asignó a C dicho valor para el resto de la experimentación.

Tabla A.3. Segunda etapa de evaluación de porcentaje de contribución en DNSGA-II-PS.

	cr1	cr2	cr3	cr4	cr5
IGD	1	0	2	3	9
MS	2	0	2	6	5
HVR	0	1	1	1	12
TOTAL	3	1	5	10	26

De igual manera también se evaluó que tan constante debe ser la aplicación del PS dentro del DMOEA para generar mejores resultados. Utilizando la configuración ganadora de las experimentaciones anteriores se desarrollaron múltiples versiones de DNSGA-II-PS, las cuales aplicaban PS cada 1, 5, 10 o 20 generaciones. Los resultados se presentan en la Tabla A.4. Aunque PS aplicado cada 20 generaciones presenta un mayor número de victorias, la mayoría son basadas en diversidad. Por otra parte, el mejor desempeño respecto a convergencia es presentado por PS aplicado cada 5 generaciones. Por lo tanto, se define que PS será aplicado cada 5 generaciones para cada DMOEA utilizado en esta experimentación.

Tabla A.4. Comparativa de efectividad de PS bajo diferentes generaciones.

	g=1	g=5	g=10	g=20
IGD	3	4	4	4
MS	1	2	4	8
HVR	3	5	3	4
TOTAL	7	11	11	16

Anexo B. Resultados de VD y HVR en la comparación de DHH-PS y DPEM-HH-PS

Los valores correspondientes a la mediana y desviación estándar fuera de línea de la mejor configuración de hiperheurística para cada DMOP que se mostraron en las Tablas 7.28 y 7.29 se muestran a continuación.

Tabla B.1. VD y HVR de la mejor configuración de hiperheurística, $W_{f1} = [0.1, 0.4]$.

Métrica	VD		HVR	
	DHH-PS	DPEM-HH-PS	DHH-PS	DPEM-HH-PS
FDA1	1.521e-1(3.042e-2)	1.636e-1(2.141e-2)	6.590e-1(2.100e-2)	6.592e-1(2.550e-2)
FDA3	4.751e-1(1.434e-1)	5.221e-1(2.622e-1)	3.631e-1(3.177e-2)	3.615e-1(3.117e-2)
FDA4	1.080e-1(2.175e-2)	1.080e-1(1.765e-2)	5.924e-1(3.003e-2)	5.949e-1(3.254e-2)
FDA5	1.073e-1(1.835e-2)	1.109e-1(1.425e-2)	5.718e-1(4.154e-2)	5.713e-1(2.742e-2)
dMOP1	2.418e-2(7.429e-3)	2.287e-2(1.015e-2)	9.218e-1(1.529e-2)	9.179e-1(1.745e-2)
dMOP2	3.592e-2(3.080e-3)	3.542e-2(3.750e-3)	8.681e-1(9.117e-3)	8.751e-1(1.016e-2)
dMOP3	1.049e-1(6.097e-2)	9.620e-2(5.948e-2)	8.030e-1(1.015e-1)	7.967e-1(5.985e-2)

Tabla B.2. VD y HVR de la mejor configuración de hiperheurística, $W_{f1} = [0.6, 0.9]$.

Métrica	VD		HVR	
	DHH-PS	DPEM-HH-PS	DHH-PS	DPEM-HH-PS
FDA1	2.256e-1(6.990e-2)	2.226e-1(3.020e-2)	5.542e-1(2.464e-2)	5.548e-1(2.461e-2)
FDA3	7.558e-1(9.886e-2)	7.386e-1(1.102e-1)	2.250e-1(1.565e-2)	2.369e-1(1.846e-2)
FDA4	1.350e-1(7.042e-2)	1.349e-1(3.459e-2)	5.683e-1(3.006e-2)	5.773e-1(2.957e-2)
FDA5	1.420e-1(2.887e-2)	1.388e-1(4.136e-2)	5.609e-1(3.958e-2)	5.779e-1(3.722e-2)
dMOP1	5.007e-2(9.862e-2)	2.918e-2(6.530e-2)	8.882e-1(3.488e-2)	9.301e-01(3.118e-2)
dMOP2	2.717e-2(3.015e-3)	2.590e-2(1.886e-3)	9.141e-1(9.522e-3)	9.159e-01(6.741e-3)
dMOP3	7.421e-2(6.864e-2)	9.068e-2(6.216e-2)	7.914e-1(7.943e-2)	7.815e-01(5.379e-2)

Anexo C. Clasificación de las configuraciones de parámetros para Separación por Planos

En las siguientes tablas se muestran los rankings obtenidos por cada una de las configuraciones de C y generaciones entre uso de Separación por Planos mencionadas en la Sección 7.5. En la Tabla C.1, v_1 hace referencia a $W_{f1} = [0.1-0.4]$ y v_2 a $W_{f1} = [0.6-0.9]$. Dentro de cada comparativa se presenta el número de la versión que tuvo el mejor valor.

La Tabla C.2 muestra los resultados de la sumatoria de las veces que cada versión tuvo el mejor ranking para cada métrica en cada uno de los diferentes DMAOPs. El valor después de la instancia representa el número de objetivos. Por ejemplo, FDA4-4 representa FDA4 con 4 objetivos.

Las configuraciones probadas con las siguientes:

- ps_1 : $C = \{80\%, 15\%, 3\%, 2\%\}$, $R = \{ROI, ROI+15\%, ROI+30\%\}$, $\tau_{ps} = 5$
- ps_2 : $C = \{40\%, 35\%, 20\%, 5\%\}$, $R = \{ROI, ROI+15\%, ROI+30\%\}$, $\tau_{ps} = 5$
- ps_3 : $C = \{40\%, 35\%, 20\%, 5\%\}$, $R = \{ROI, ROI+15\%, ROI+30\%\}$, $\tau_{ps} = 2$

La primera versión es la que se utiliza actualmente para DMOPs. La segunda se enfatiza en la exploración del espacio de soluciones, permitiendo a más soluciones fuera del ROI introducirse en la nueva población. La tercera versión realiza lo mismo, pero con PS ejecutándose un mayor número de veces, buscando acentuar la rotación entre exploración y explotación.

Tabla C.1: Revisión de mejor versión por métrica e instancia

Métrica	FDA4-4		FDA4-5		FDA4-6		FDA5-4		FDA5-5		FDA5-6	
	v_1	v_2										
LRNI	1	3	3	3	1	3	3	3	3	3	3	1
ASF	3	3	3	3	3	3	1	1	2	2	2	2
R2	1	3	3	3	1	3	1	3	2	1	2	3
HV	2	1	3	1	3	1	3	1	3	1	3	3

Tabla C.2: Sumatoria de mejores rankings

SUMA	FDA4-4		FDA4-5		FDA4-6		FDA5-4		FDA5-5		FDA5-6		
	v_1	v_2											
ps_1	2	1	0	1	2	1	2	2	0	2	0	1	
ps_2	1	0	0	0	0	0	0	0	2	1	2	1	
ps_3	1	3	4	3	2	3	2	2	2	1	2	2	
TOTAL												TOTAL	
ps_1	3		1		3		4		2		1		14
ps_2	1		0		0		0		3		3		7
ps_3	4		7		5		4		3		4		27

Anexo D. Resultados de la comparación entre métodos de selección de LLHs para DHHMa

En esta sección se presentan a mayor detalle los resultados obtenidos y presentando en la Sección 7.6.1, las Tablas D.1, D.2 y D.3 muestran los valores de DHHMa-A y DHHMa-B para ASF, R2 y HV, respectivamente. Para determinar si existe una diferencia significativa entre ambas versiones de DHHMa se utiliza una prueba de Wilcoxon con $p = 0.05$ y una prueba post-hoc de Holm. Sin embargo, es necesario aclarar que ninguno de los resultados presenta una diferencia significativa entre ellos.

Tabla D.1. Mediana y desviación estándar fuera de línea para ASF.

DMOP $m = 4$	DHHMa-A	DHHMa-B
FDA4	1.387e-02 (6.690e-04)	1.375e-02 (7.039e-04)
FDA5	1.441e-02 (5.146e-04)	1.449e-02 (5.735e-04)
$m = 5$		
FDA4	1.843e-03 (1.559e-04)	1.825e-03 (1.253e-04)
FDA5	1.853e-03 (2.136e-04)	1.896e-03 (1.471e-04)
$m = 6$		
FDA4	4.811e-04 (5.243e-05)	4.927e-04 (5.542e-05)
FDA5	4.241e-04 (6.083e-05)	4.351e-04 (4.472e-05)

Tabla D.2. Mediana y desviación estándar fuera de línea para R2.

DMOP $m = 4$	DHHMa-A	DHHMa-B
FDA4	1.823e-02 (2.922e-04)	1.805e-02 (3.758e-04)
FDA5	2.762e-02 (3.152e-04)	2.739e-02 (3.548e-04)
$m = 5$		
FDA4	3.675e-03 (1.750e-04)	3.656e-03 (1.670e-04)
FDA5	5.368e-03 (3.031e-04)	5.356e-03 (2.418e-04)
$m = 6$		
FDA4	8.829e-04 (1.033e-04)	8.668e-04 (8.822e-05)
FDA5	1.409e-03 (1.482e-04)	1.425e-03 (1.655e-04)

Tabla D.3. Mediana y desviación estándar fuera de línea para HV.

DMOP $m = 4$	DHHMa-A	DHHMa-B
FDA4	2.993e-01 (2.454e-02)	2.944e-01 (3.255e-02)
FDA5	4.249e-02 (1.097e-02)	4.099e-02 (1.397e-02)
$m = 5$		
FDA4	5.678e-02 (2.546e-02)	5.094e-02 (2.008e-02)
FDA5	8.865e-03 (1.913e-02)	1.038e-02 (1.803e-02)
$m = 6$		
FDA4	7.176e-02 (2.790e-02)	7.002e-02 (3.158e-02)
FDA5	2.267e-02 (1.609e-02)	2.274e-02 (1.590e-02)