



UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA
FACULTAD DE INGENIERÍA
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO
MAESTRÍA EN INGENIERÍA EN COMPUTACIÓN

UNA COMPARACIÓN ESTADÍSTICA DE FUNCIONES OBJETIVO PARA
EL PROBLEMA DE RUTEO DE VEHÍCULOS CON BALANCEO DE RUTAS

TESIS

PARA OBTENER EL GRADO DE MAESTRO EN INGENIERÍA EN COMPUTACIÓN

PRESENTA

JAIRO GIBRÁN LOZANO CARBALLO

DIRECTOR DE TESIS

DR. LUIS CARLOS GONZALES GURROLA

CHIHUAHUA, CHIHUAHUA.

NOVIEMBRE 2016



UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA
FACULTAD DE INGENIERÍA
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO
MAESTRÍA EN INGENIERÍA EN COMPUTACIÓN

UNA COMPARACIÓN ESTADÍSTICA DE FUNCIONES OBJETIVO PARA
EL PROBLEMA DE RUTEO DE VEHÍCULOS CON BALANCEO DE RUTAS

TESIS

PARA OBTENER EL GRADO DE MAESTRO EN INGENIERÍA EN COMPUTACIÓN

APROBADO:

DR. LUIS CARLOS GONZALES GURROLA, director

DR. FERNANDO MARTINEZ REYES, sinodal

Graciela Ramirez A.

DRA. GRACIELA RAMÍREZ ALONSO, sinodal

NOVIEMBRE 2016
CHIHUAHUA, CHIHUAHUA.

Derechos reservados

© Jairo Gibrán Lozano Carballo

Circuito No. 1, Nuevo Campus Universitario II

Chihuahua, Chih. C.P. 31100

2016

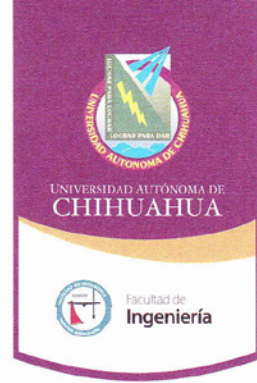
Copyright ©

por

JAIRO GIBRÁN LOZANO CARBALLO

Año 2016

24 de noviembre de 2016



ING. JAIRO GIBRÁN LOZANO CARBALLO

Presente

En atención a su solicitud relativa al trabajo de tesis para obtener el grado de Maestro en Ingeniería, nos es grato transcribirle el tema aprobado por esta Dirección, propuesto y dirigido por el director **Dr. Luis Carlos González Gurrola** para que lo desarrolle como tesis, con el título: **“UNA COMPARACIÓN ESTADÍSTICA DE FUNCIONES OBJETIVO PARA EL PROBLEMA DE RUTEO DE VEHÍCULOS CON BALANCEO DE RUTAS”**.

ÍNDICE

1. INTRODUCCIÓN

- 1.1. Antecedentes
- 1.2. Estado del Arte
- 1.3. Problema de Investigación
- 1.4. Preguntas de Investigación
- 1.5. Justificación
- 1.6. Delimitación
- 1.7. Objetivo General
- 1.8. Formulación de la hipótesis

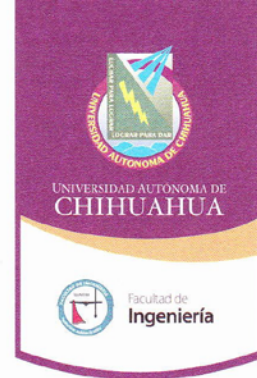
2. MARCO TEÓRICO

- 2.1. Problemas multi-objetivo, dominancia y optimalidad de pareto
- 2.3. Algoritmos genéticos

3. METODOLOGÍA

4. IMPLEMENTACIÓN DE ALGORITMO EVOLUTIVO 1+1

- 4.1. Heurística para construcción de la solución inicial
- 4.2. Transformación de espacios de búsqueda VRP a TSP
- 4.3. Operador de mutación SWAP
- 4.4. Operador de mutación REVERSE
- 4.5. Heurística para selección de punto de referencia z para cálculo de hipervolumen



5. EXPERIMENTOS

- 5.1. Ejecución de algoritmo AE (1+1) sobre las instancias VRP
- 5.2. Transformación de soluciones a un espacio objetivo común
- 5.3. Prueba t-test para comparar estadísticamente promedios de vectores de hipervolumenes
- 5.4. Esquema de puntajes en prueba t-test entre FOs

6. RESULTADOS

- 6.1. Condensado de resultados de pruebas t-test por instancia
- 6.2. Puntajes acumulados por FOs
- 6.3. Análisis de resultados
- 6.4. Condensado de resultados

7. CONCLUSIONES

- 7.1. Sobre las preguntas de investigación
- 7.2. Sobre la hipótesis
- 7.3. Sobre las nuevas interrogantes
- 7.4. Conclusión general

Referencias

Currículum vitae

Solicitamos a Usted tomar nota de que el título del trabajo se imprima en lugar visible de los ejemplares de las tesis.

ATENTAMENTE
"naturam subiecit aliis"

EL DIRECTOR

M.I. JAVIER GONZÁLEZ CANTÚ



EL SECRETARIO DE INVESTIGACIÓN
Y POSGRADO

DR. FERNANDO RAFAEL ASTORGA
BUSTILLOS

Abstract

Between August 2014 and September 2016 a research on statistically comparing the quality of the results obtained by using a simple evolutionary meta-heuristic for the vehicle routing problem with route balancing (VRPRB) when using different objective functions for measuring the balance was carried out. A basic genetic algorithm, the so called 1+1 evolutionary algorithm, was implemented and executed on 60 recently created VRP instances, 30 times per instance. A t-test between hypervolume vectors was carried out in order to find statistically significant differences. Results show that, there is indeed a significant difference on the quality of the solutions obtained when using different objective functions for measuring the balance between routes in a VRPRB. Moreover, the most widely used objective functions are not the ones leading to highest quality results. On the other hand, the best solutions were found while using objective functions which usage is not that common. This results provide researchers in the area with a reference point allowing them to take a smarter decision when choosing an objective function for measuring the balance between routes in a VRPRB.

Keywords: VRPRB, VRP, objective functions for balance, MAX-MIN, 1+1 evolutionary algorithm.

Resumen

Entre Agosto de 2014 y Septiembre de 2016 se llevó a cabo una investigación en torno a la comparación estadística de la calidad de los resultados obtenidos por una meta-heurística evolutiva simple aplicada al problema de ruteo de vehículos con balanceo de rutas (VRPRB) al utilizar diferentes funciones objetivo para medir el balanceo. Para esto se implementó un algoritmo genético básico, el llamado algoritmo evolutivo 1+1, y se ejecutó sobre 60 instancias VRP de reciente creación, 30 ocasiones por instancia. Se realizó una prueba t-test entre los vectores de hipervolumenes de los conjuntos de soluciones obtenidos por el algoritmo para determinar si existían diferencias estadísticamente significativas. Los resultados obtenidos muestran que, en efecto, existe diferencia significativa en la calidad de las soluciones obtenidas al utilizar diferentes funciones objetivo para el balanceo entre rutas en un VRPRB. Más aún, las funciones objetivo más ampliamente utilizadas en la literatura no son las que conducen a resultados de mayor calidad, mientras que las mejores soluciones fueron alcanzadas al utilizar funciones objetivo de uso menos común. Los resultados de esta investigación proporcionan un punto de referencia para investigadores del área permitiéndoles tomar una decisión más informada a la hora de seleccionar una función objetivo para medir el balanceo entre rutas en un VRPRB.

Palabras clave: VRPRB, VRP, funciones objetivo para balanceo, MAX-MIN, algoritmo evolutivo 1+1

Agradecimientos

Antes que a todos, a Dios. Por las bendiciones que, sin merecer, gozo al momento de escribir este mensaje. Gracias por mover mi vida con tanta perfección.

A mi familia... no existen palabras adecuadas para corresponder a tal sacrificio y apoyo incondicional. Nuestro humilde origen no ha sido obstáculo para que me lleven de la mano hasta alcanzar mis más increíbles sueños. Los amo a todos.

A mis amigos, a los que ya tenía y a los que adquirí. Por corresponder a mis locuras y tolerar mis fallos. A mis maestros, a todos quienes participaron en mi formación académica y en especial al Dr. Luis Carlos González, sin su asertividad no hubiera explotado tanto mi potencial. Al Dr. Fernando Martínez y la Dra. Graciela Ramírez, mis sinodales. A los Doctores Darío Landa Silva y Philippe Lacomme por recibirme en sus universidades en una de las experiencias más importantes de mi vida. Al Dr. Eduardo Rodríguez por su mentoría y su colaboración en mis proyectos de investigación.

A la Facultad de Ingeniería y a todos los ángeles que en ella trabajan. Gracias por hacer de cada trámite una grata experiencia.

Al CONACyT, por su noble labor para con nosotros los jóvenes investigadores con ambición de cambiar México para bien.

Dedicatoria

A Dios, y a todas las personas que de una u otra forma han tocado mi vida. Porque soy el resultado de la influencia de todos ellos. No les dedico únicamente este trabajo, les dedico el ser humano que soy en este momento. Ojalá tenga oportunidad de compensar todo el bien que me han hecho.



Índice de Contenido

1. INTRODUCCIÓN	1
1.1. Antecedentes.....	1
1.2. Estado del Arte	2
1.3. Problema de Investigación.....	3
1.4. Preguntas de Investigación	3
1.5. Justificación.....	4
1.6. Delimitación	4
1.6.1. Instancias de Matl <i>et al.</i>	4
1.7. Objetivo General	5
1.8. Formulación de la hipótesis.....	5
2. MARCO TEÓRICO	6
2.1. Problemas Multi-Objetivo, Dominancia y Optimalidad de Pareto.....	6
2.1.1. Hipervolumen como métrica de calidad de la aproximación a Frente de Pareto	9
2.2. Problema de Ruteo de Vehículos (VRP).....	9
2.2.1. Problema de Ruteo de Vehículos Capacitados (CVRP)	11
2.2.2. Problema de Ruteo de Vehículos con Balanceo de Rutas (VRPRB).....	12
2.2.3. Funciones Objetivo para balance entre rutas.....	13
2.3. Algoritmos Genéticos	15
2.3.1. Algoritmo Evolutivo (1+1).....	20
3. METODOLOGÍA	22
4. IMPLEMENTACIÓN DE ALGORITMO EVOLUTIVO 1+1	25
4.1. Heurística para construcción de la solución inicial	26
4.2. Transformación de espacios de búsqueda VRP a TSP.....	26
4.3. Operador de mutación SWAP.....	27
4.4. Operador de mutación REVERSE.....	27
4.5. Heurística para selección de punto de referencia z para cálculo de hipervolumen.....	27

Índice de Contenido

5. EXPERIMENTOS	29
5.1. Ejecución del algoritmo AE (1+1) sobre las instancias VRP	29
5.2. Transformación de soluciones a un espacio objetivo común	30
5.3. Prueba <i>t-test</i> para comparar estadísticamente promedios de vectores de hipervolumenes . . .	30
5.4. Esquema de puntajes en prueba <i>t-test</i> entre FOs	30
6. RESULTADOS	31
6.1. Condensado de resultados de pruebas <i>t-test</i> por instancia	31
6.2. Puntajes acumulados por FOs.....	32
6.3. Análisis de resultados	33
6.3.1. Comportamientos distintos al utilizar diferentes FOs por instancia	33
6.3.2. Comportamientos similares entre FOs al utilizar los dos operadores de mutación . .	33
6.3.3. Bajo desempeño del algoritmo en ambas variantes al usar FO MinMax.....	36
6.3.4. Empates entre todas las FOs.....	37
6.3.5. VAR, MAD, Gini y Std_Dev con mayor puntaje en más del 86 % de las comparaciones	37
6.3.6. Diferentes FOs obtienen el puntaje más alto dependiendo del operador de mutación utilizado.....	39
6.3.7. Casos particulares: instancias #31 y #43	39
6.4. Condensado de resultados.....	40
7. CONCLUSIONES	42
7.1. Sobre las preguntas de investigación	43
7.2. Sobre la hipótesis.....	44
7.3. Sobre las nuevas interrogantes.....	45
7.4. Conclusión general	46
Referencias	48
Curriculum vitae	51



Lista de Figuras

2.1.	Ejemplo de un Frente de Pareto para un problema de minimización de dos objetivos.	7
2.2.	Esquema de áreas de preferencia en dominancia entre soluciones. Para un problema de minimización la solución representada por un triángulo en el centro domina a las soluciones en B, es dominada por las soluciones en C y es indiferente con las soluciones en A y D.....	8
2.3.	Ejemplo de dominancia entre soluciones.....	9
2.4.	Ejemplo de aproximación a Frente de Pareto.	10
2.5.	a) Los aristas entre puntos representan una posible configuración de ruta TSP b) Generalización de la misma instancia TSP a VRP, seleccionando como depósito el cliente denotado mediante una estrella e involucrando tres vehículos.....	11
2.6.	Configuración de rutas factible para CVRP con capacidad Q por vehículo igual a 15 unidades. Se especifica la demanda de cada cliente, ninguna ruta sobrepasa Q	12
2.7.	a) El balance para esta solución, considerando la FO Max-Min, sería de 56 (136 - 80) b) El balance de esta solución sería de 5 (112 - 107)	13
2.8.	Codificación de una solución consistente en una cadena de 10 alelos usando un alfabeto binario	16
2.9.	Selección por ruleta.- Suponiendo que se generara un número aleatorio igual a .52, la solución elegida sería la número 2	17
2.10.	a) Crossover de 1 punto.- el punto de cruce es en la posición 6. El Hijo 1 se compone de la primera parte del Padre 1 (línea discontinua) y la segunda parte del Padre 2 (línea continua), el Hijo 2 se construye de forma inversa. b) Crossover de 3 puntos con puntos de cruce en las posiciones 2, 6 y 8. El Hijo 1 se compone de la primera y tercera sección del Padre 1 (línea discontinua) y la segunda y cuarta sección del Padre 2 (línea continua). El Hijo 2 se construye de forma inversa	18

2.11. Crossover por Máscara.- Un cero en la máscara indica que la información para ese alelo en la construcción del Hijo 1 será tomada del primer padre y la información para ese alelo en la construcción del Hijo 2 será tomada del segundo padre. Un uno en la máscara indica la operación inversa..... 18

2.12. Mutación.- El valor del séptimo alelo cambia de 1 a 0 19

6.1. Número de ocasiones en las que una FO dada obtuvo el puntaje más alto sobre las 60 instancias usando cada uno de los operadores de mutación..... 41



Lista de Tablas

2.1. Crecimiento exponencial de número de posibles rutas al incrementar número de puntos . . .	10
2.2. Funciones Objetivo para balanceo en VRPRB identificadas en [1].....	14
2.3. Funciones Objetivo para balanceo en VRPRB identificadas en [2].....	15
6.1. Condensado de resultados de pruebas <i>t-test</i> para instancia 29 con operador de mutación REVERSE.	32
6.2. Condensado de resultados de pruebas <i>t-test</i> para instancia 29 con operador de mutación SWAP.	32
6.3. Puntajes acumulados por cada FO por operador de mutación sobre las instancias 1 a 30.....	34
6.4. Puntajes acumulados por cada FO por operador de mutación sobre las instancias 31 a 60. . .	35



1

INTRODUCCIÓN

1.1. Antecedentes

En la actualidad, cualquier actividad económica involucra en pequeña o gran escala el transporte de bienes (*e.g.* comercio, sector agrícola, turismo, industria manufacturera, educación, por mencionar algunas). El transporte de bienes requiere de una inversión de recursos que, como cualquier otra inversión, busca hacerse de manera óptima. La optimización de recursos en el transporte de bienes depende mayormente de la planeación de las rutas para los vehículos involucrados, una tarea bastante compleja que requiere de mucho tiempo y talento administrativo y que la mayoría de las veces termina haciéndose de manera ineficaz, dando como resultado rutas de transporte ineficientes. Una configuración ineficiente de rutas se ve reflejada en la necesidad de un mayor número de vehículos para proveer el transporte, así como rutas excesivamente largas para cada uno de los vehículos involucrados. Esto a su vez se ve traducido en una mayor inversión de recursos por parte de la institución que paga por el transporte, esto por concepto de un mayor número de vehículos alquilados, el pago de salarios a un mayor número de conductores, mayor gasto en el suministro de combustible para los vehículos, mantenimiento, etc., además del impacto ambiental que dicha situación representa: mayores emisiones de dióxido de carbono y mayor congestión vehicular.

Como se mencionó en el párrafo anterior, la planeación de rutas de transporte eficientes es una tarea bastante compleja. Tanto es así, que constituye uno de los problemas más estudiados en el área de la optimización combinatoria: el Problema de Ruteo de Vehículos (VRP por sus siglas en inglés: *Vehicle Routing Problem*). Propuesto por Dantzing y Ramser [3], en su forma básica, el VRP busca generar rutas eficientes para una flota de vehículos con el objetivo de entregar (o recolectar) productos de almacenes a un conjunto de clientes minimizando la distancia total recorrida por las rutas [4].

Para dar solución al VRP se han utilizado diversas estrategias. Los métodos exactos garantizan encontrar

siempre la solución óptima, sin embargo, requieren de tiempos de ejecución particularmente grandes para instancias del problema relativamente pequeñas [5] por lo cual dichas estrategias resultan incosteables en la práctica. Otra alternativa son los métodos constructivos, en los cuales, tal y como su nombre lo indica, se construye una solución desde cero, cuidando siempre que cumpla con las restricciones establecidas. Se obtiene así una solución viable pero, dado que estas aproximaciones generalmente carecen de una etapa de mejora, la mayoría de las veces se obtiene una solución ineficiente. Por otro lado, las *meta-heurísticas* (métodos heurísticos que pueden ser aplicados a una amplia variedad de problemas) conducen a soluciones sub-óptimas de calidad relativamente alta en una cantidad pequeña de tiempo [6].

Dentro de las aproximaciones meta-heurísticas se encuentra una de las estrategias quizá más ampliamente utilizadas para el VRP debido a su eficiencia y relativa simplicidad de implementación: los algoritmos genéticos (AG) [7] (Véase sección 2.3). Los AG buscan encontrar soluciones cercanas a las óptimas simulando el proceso evolutivo por selección natural mediante operadores de cruce, mutación y selección.

Al utilizar una meta-heurística como un AG se pueden encontrar soluciones de alta calidad para una instancia particular de VRP, es decir, una configuración eficiente de rutas. Sin embargo, estas soluciones generalmente representan una distribución desigual de las rutas entre los vehículos involucrados en términos, por ejemplo, de distancia recorrida. En otras palabras, algunos vehículos tienen que recorrer mayores distancias que otros. Esto puede traer descontento entre los empleados debido a las cargas desiguales de trabajo. Dado que el capital humano es clave para cualquier organización, es evidente que el balanceo de las rutas es un factor de gran trascendencia en este tipo de problemas [8].

Considerar el balanceo entre las rutas como un objetivo a optimizar adicional a la minimización de la distancia total recorrida por las mismas constituye precisamente un problema derivado del VRP, el Problema de Ruteo de Vehículos con Balanceo de Rutas (VRPRB por sus siglas en inglés: *Vehicle Routing Problem with Route Balancing*). Al igual que en el caso de VRP, los AG constituyen una aproximación bastante conveniente para buscar dar solución al VRPRB, ya que, al trabajar con un conjunto de soluciones, permiten una mejor exploración del espacio de búsqueda considerando los dos objetivos [9].

1.2. Estado del Arte

En un problema de optimización, como el VRP y sus derivados, la *función objetivo* FO es la función que representa el costo, la ganancia o alguna otra cantidad que se busca maximizar o minimizar sobre un conjunto de parámetros y sujeta a ciertas restricciones. Para maximizar el balance entre rutas (o minimizar el desbalance) en una solución de VRPRB diversas funciones objetivo FOs han sido propuestas. En [1], los

autores presentan un panorama general de las diferentes FOs utilizadas en la literatura del VRPRB para medir balance entre rutas, dando una breve descripción y análisis de cada una para justificar la selección de la FO Max (minimizar la ruta con mayor distancia recorrida) para su estudio, argumentando su naturaleza lineal y la ausencia de indicadores negativos (Véase 2.2.3). Sin embargo, no se provee evidencia objetiva de si, en efecto, la FO Max permite obtener mejores resultados que el resto de las FOs. En [2] los autores identifican un conjunto de propiedades axiomáticas que deben ser satisfechas por una FO ideal (para medir equidad) y las utilizan para evaluar seis FOs utilizadas en la literatura del VRPRB, señalando sus características y las propiedades de los *Frentes de Pareto* obtenidos (véase sección 2.1). Para el estudio, los autores crearon un conjunto de instancias VRP pequeñas, que pueden ser resueltas a optimalidad, y generaron todas las posibles soluciones. Los autores concluyen que ninguna de las FO satisface todos y cada uno de los axiomas y, por tanto, ninguna FO es mejor que las otras en todos los aspectos relevantes. A la fecha no se ha encontrado en la literatura una comparación cuantitativa de la calidad de las soluciones generadas por una meta-heurística como un AG aplicado al VRPRB al utilizar diferentes FOs para medir el balanceo entre las rutas.

1.3. Problema de Investigación

Hoy en día, hasta donde conocemos, no existe una comparación cuantitativa de la calidad de las soluciones generadas por un AG al seleccionar diferentes FOs para medir el balanceo entre rutas en el VRPRB. En la literatura del problema pueden identificarse al menos diez tipos diferentes de FOs para el balanceo [1, 2]. Si la selección de entre alguna de estas FO impacta directamente en la calidad de las soluciones generadas por un AG, resulta de gran importancia conocer la correlación entre la FO seleccionada y la calidad esperada de las soluciones. En otras palabras, si se puede comparar objetivamente la calidad de las soluciones para VRPRB generadas por un AG al utilizar las diferentes FOs existentes en la literatura, sería de gran interés conocer si alguna de ellas conduce a mejores resultados que las demás, o si de lo contrario, la selección de la FO para medir el balanceo no impacta directamente en la calidad de las soluciones.

1.4. Preguntas de Investigación

La situación descrita en párrafos anteriores da lugar a los siguientes cuestionamientos, a los cuales se busca dar respuesta mediante la investigación propuesta:

- ¿Existe alguna diferencia considerable en la calidad de las soluciones generadas por un AG para VRPRB al utilizar diferentes FOs para medir el balanceo entre rutas?

- ¿Utilizar diferentes FOs para balanceo conduce a obtener soluciones diferentes? o ¿se obtienen las mismas soluciones sin importar que FO se utilice?
- ¿Son las FOs más ampliamente utilizadas en la literatura del VRPRB las que permiten obtener mejores resultados?

1.5. Justificación

En [8] los autores afirman que la competitividad de una compañía depende ampliamente de su habilidad para tratar a sus empleados de manera justa. Debido a esto, resulta inaceptable optimizar rutas de transporte si esto implica generar cargas dispares de trabajo para los conductores (en los casos, por ejemplo, en que algunos deban conducir mayores distancias que otros), o tiempos de traslado desiguales para los empleados que se sirven del transporte de personal. Por tanto, es evidente que el balance es un factor importante al generar configuraciones de rutas para vehículos en un problema real.

Tal y como fue señalado en párrafos anteriores, a la fecha no se ha encontrado en la literatura del VRPRB una comparación cuantitativa de la calidad de las soluciones generadas por un AG al seleccionar diferentes FOs para medir el balanceo entre rutas. El desarrollo de esta investigación proveerá de una referencia objetiva para tomar la decisión de cuál FO utilizar para medir el balanceo al aplicar un AG a un VRPRB, beneficiando de manera directa futuras investigaciones en el área y de manera indirecta a todas aquellas personas que puedan servirse de las soluciones generadas por una meta-heurística evolutiva aplicada al VRPRB (empleados, conductores, administradores, etc.).

1.6. Delimitación

Con el objetivo de contar con un punto de referencia objetivo para situar los resultados obtenidos en la presente investigación, se optó por utilizar un conjunto de instancias de VRP de reciente creación, utilizadas en la literatura del VRPRB, mismas que se describen a continuación.

1.6.1. Instancias de Matl *et al.*

En una publicación reciente Matl *et al.* realizan una comparación entre seis FOs para balance de rutas evaluándolas bajo un conjunto de propiedades que dichas FOs deben satisfacer [2]. Para su estudio, los autores proponen un conjunto compuesto de 60 instancias VRP de tamaño reducido que pueden ser resueltas a optimalidad. Dichas instancias son creadas de la siguiente manera: tomando como referencia las instancias

1 a 10 de Christofides et al. [10], los autores toman conjuntos de 14 clientes para crear un subconjunto de instancias variando el número de vehículos y la capacidad máxima de cada vehículo y asignando demanda unitaria a todos los clientes.

Para estudiar las propiedades de las aproximaciones a frente de Pareto (véase sección 2.1), los autores resuelven cada instancia a optimalidad mediante la construcción de todas las posibles combinaciones de rutas. La presente investigación pretende complementar la aportación de [2] al utilizar el mismo conjunto de instancias para conocer la calidad de las aproximaciones a frente de Pareto obtenidas al utilizar cada una de las FOs bajo estudio utilizando una meta-heurística evolutiva.

1.7. Objetivo General

La presente investigación tiene como objetivo general el realizar una comparación cuantitativa de la calidad de las soluciones generadas por un AG (meta-heurística evolutiva) al utilizar nueve diferentes FOs para balanceo de rutas en el VRPRB.

Objetivos específicos:

- Diseñar e implementar un AG (meta-heurística evolutiva) aplicado al VRPRB con las diferentes FOs para balanceo bajo estudio.
- Analizar y comparar estadísticamente los resultados obtenidos al utilizar diferentes FOs para balanceo de rutas en la ejecución del AG implementado, sobre un conjunto de 60 instancias de VRP conocidas en la literatura.

1.8. Formulación de la hipótesis

La calidad de las soluciones generadas por un AG para VRPRB se ve influida por la selección de la FO utilizada para medir el balanceo entre las rutas.

2

MARCO TEÓRICO

2.1. Problemas Multi-Objetivo, Dominancia y Optimalidad de Pareto

Un problema *multi-objetivo* (PMO) se define de la siguiente manera¹:

$$(PMO) = \begin{cases} \min F(x) = (f_1(x), f_2(x), \dots, f_n(x)), \\ \text{sujeto a } x \in D \end{cases} \quad (2.1)$$

Donde $n \geq 2$ es el número de funciones objetivo, $x = (x_1, x_2, \dots, x_n)$ el vector de variables de decisión, D el espacio de soluciones factibles, y $F(x)$ el vector objetivo. El conjunto $O = F(D)$ corresponde a los puntos factibles en el espacio objetivo y $y = (y_1, y_2, \dots, y_n)$, donde $y_i = f_i(x)$, es un punto en el espacio objetivo [11].

Los objetivos que se buscan optimizar en un PMO generalmente se encuentran en conflicto, es decir, la mejora en alguno de los objetivos implica el deterioro en algún otro. Un ejemplo claro de esta situación sería buscar minimizar presupuesto total y tiempo de ejecución en la realización de un proyecto: generalmente si se reduce el presupuesto, el tiempo de ejecución tiende a aumentar y viceversa. Debido a lo anterior, no puede encontrarse una solución única que optimice todos los objetivos en un PMO (como sí es el caso en los problemas *mono-objetivo* en los que se busca optimizar un solo objetivo), en lugar de esto, se busca un conjunto de soluciones que representan la mejor compensación posible entre los objetivos a optimizar. Dicho conjunto de soluciones se conoce como *Frente de Pareto*. Un ejemplo del Frente de Pareto para un problema **bi-objetivo** (donde se busca minimizar ambos objetivos) puede apreciarse en la Figura 2.1.

Debido a la naturaleza altamente combinatoria de este tipo de problemas (VRP y sus variantes), el Frente

¹Para efectos de uniformidad en este documento se estará haciendo referencia en todo momento a problemas de minimización, pero las definiciones podrían ser generalizadas a maximización mediante el cambio de operadores relacionales.

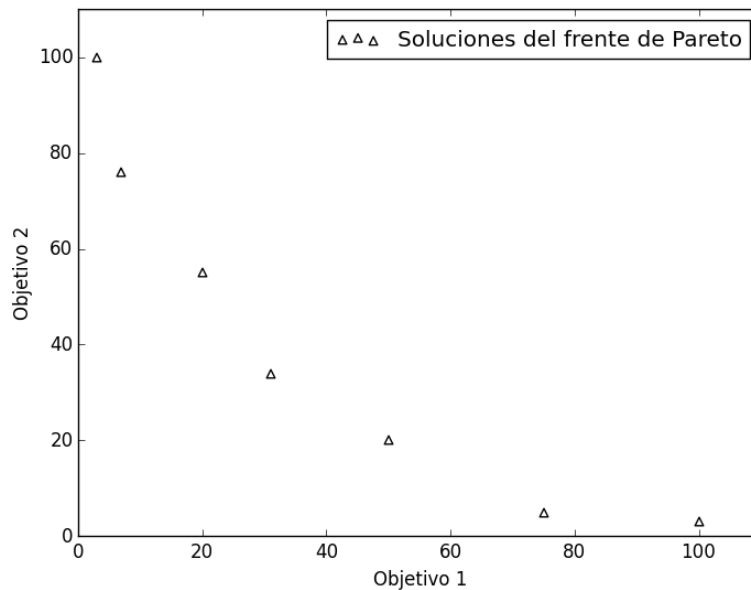


Figura 2.1: Ejemplo de un Frente de Pareto para un problema de minimización de dos objetivos.

de Pareto o conjunto de soluciones óptimas es desconocido y conceptualmente imposible de calcular ². Por tanto, la solución a un PMO consiste en un conjunto *aproximado* al Frente de Pareto. Del conjunto de soluciones generadas por una meta-heurística (como un AG) para un PMO, forman parte de la aproximación al Frente de Pareto únicamente aquellas soluciones *no dominadas*. La *dominancia* entre soluciones se define de la siguiente manera:

- Una solución $y = (y_1, y_2, \dots, y_n)$ domina (denotado por $<$) a una solución $z = (z_1, z_2, \dots, z_n)$ si y solo si $\forall i \in \{1, 2, \dots, n\}, f_i(y) \leq f_i(z)$ y $\exists i \in \{1, 2, \dots, n\}, f_i(y) < f_i(z)$

En términos coloquiales, una solución y domina a una solución z si y es menor o igual que z en todos los objetivos y estrictamente menor que z en al menos un objetivo (y es al menos tan buena como z en todos los objetivos y estrictamente mejor que z en al menos un objetivo). Para una mejor comprensión del concepto de dominancia en [12] los autores proponen un esquema de áreas asociadas con niveles de preferencia entre soluciones. Hablando específicamente de dos objetivos a minimizar, dicho esquema de áreas de preferencia se define como se muestra en la Figura 2.2.

²El VRP y sus variantes constituyen una familia de problemas de la categoría NP-Difícil, por tanto no existe, hasta el momento, algoritmo capaz de encontrar el conjunto de soluciones óptimas para todas las instancias en tiempo polinomial. Sin embargo, dependiendo del tamaño de la instancia, es posible encontrar el conjunto de soluciones óptimas mediante aproximaciones exactas [5] o construyendo todas las combinaciones posibles para luego encontrar el frente de Pareto [2].

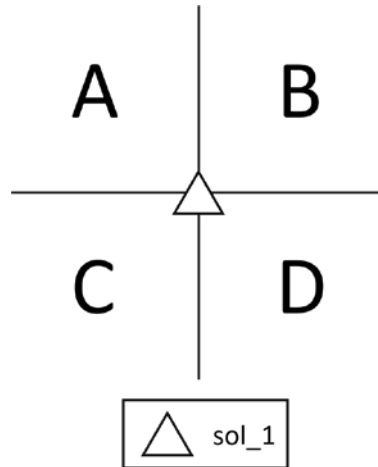


Figura 2.2: Esquema de áreas de preferencia en dominancia entre soluciones. Para un problema de minimización la solución representada por un triángulo en el centro domina a las soluciones en B, es dominada por las soluciones en C y es indiferente con las soluciones en A y D.

Si se ubica la solución sol_1 de referencia en el centro de la cruz, las soluciones en las áreas A y D (áreas de indiferencia) constituyen soluciones *no-comparables*, es decir, soluciones que no dominan a la solución sol_1 pero tampoco son dominadas por la misma, dado que es mejor en uno de los objetivos pero peor en cuanto al otro. No podría decirse entonces, si sol_1 es mejor o peor que alguna solución en las áreas A y D. Las soluciones en la sección B (área de dominación) son soluciones dominadas por sol_1 (sol_1 es mejor que las soluciones dentro de esta sección en los dos objetivos por tanto, sol_1 es preferible a cualquiera de las soluciones en esta sección). Las soluciones en la sección C (área de preferencia) son soluciones que dominan a sol_1 (son todas mejores que sol_1 en ambos objetivos, por tanto, son todas preferibles a sol_1). La Figura 2.3 muestra un ejemplo del uso de este esquema. En la imagen podemos observar que la solución representada por un triángulo domina a las soluciones representadas por cuadrados y a la vez es dominada por la solución representada por un círculo, mientras que es incomparable a las soluciones representadas por cruces.

Bajo este esquema de dominancia entre soluciones, la aproximación al Frente de Pareto obtenida por una meta-heurística se compone de aquellas soluciones que no son dominadas por ninguna otra solución en el conjunto completo. En la Figura 2.4 puede observarse un conjunto de 30 soluciones de las cuales solo las ocho representadas mediante triángulos forman parte de la aproximación al Frente de Pareto (*i. e.* son soluciones no dominadas), mientras que las soluciones representadas mediante círculos son soluciones dominadas.

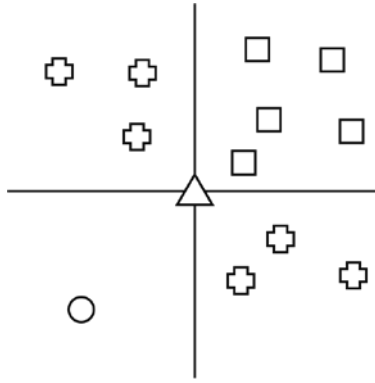


Figura 2.3: Ejemplo de dominancia entre soluciones.

2.1.1. Hipervolumen como métrica de calidad de la aproximación a Frente de Pareto

Varias métricas pueden ser utilizadas para medir la calidad de una aproximación a Frente de Pareto. Una de las más comunes es el *hipervolumen* [13], el cual consiste en el área dominada por un conjunto de soluciones A tomando como referencia un punto z en el espacio objetivo. Entre más grande el hipervolumen, mejor es la convergencia de un conjunto de soluciones dado hacia el conjunto óptimo o Frente de Pareto.

Una definición más formal del hipervolumen sería la siguiente. Considerando un espacio objetivo de dos dimensiones $f(x) = (f_1(x), f_2(x))$, cada solución $x_i \in A$ forma un rectángulo definido por sus coordenadas $(f_1(x), f_2(x))$ y las coordenadas del punto de referencia (z_1, z_2) . El hipervolumen entonces, es calculado como la unión de las áreas de todos los rectángulos definidos entre todos los puntos en A y z .

2.2. Problema de Ruteo de Vehículos (VRP)

El Problema de Ruteo de Vehículos (VRP por sus siglas en inglés: Vehicle Routing Problem) busca encontrar el conjunto de rutas con recorrido mínimo para una flota de vehículos con el objetivo de entregar (o recolectar) bienes a un conjunto de clientes. Consiste en una generalización del Problema del Agente Viajero (TSP por sus siglas en inglés: Traveling-Salesman Problem). En su forma más simple, el TSP consiste en la determinación de la ruta más corta posible que permita visitar una y solo una vez un conjunto de n puntos [3] (Figura 2.5(a)). Asumiendo que cada par de puntos está conectado por un enlace, el número de posibles configuraciones de ruta distintas es $\frac{n!}{2}$. La Tabla 2.1 muestra el crecimiento exponencial del número de rutas posibles al aumentar el número n de puntos a visitar en un TSP.

Para generalizar el TSP a VRP se introducen algunas restricciones adicionales. Primero, se requiere que el agente regrese al punto de partida una vez que ha visitado a m de los $n - 1$ puntos restantes ($n - 1$ debido a

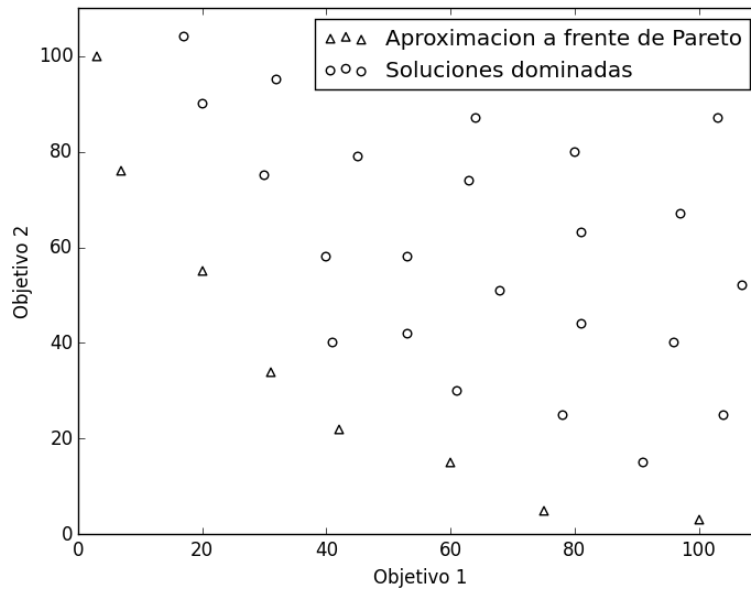


Figura 2.4: Ejemplo de aproximación a Frente de Pareto.

Tabla 2.1: Crecimiento exponencial de número de posibles rutas al incrementar número de puntos n

	Número de rutas posibles
9	181,440
10	1,814,400
11	19,958,400
12	239,500,800
15	6.53837E+11
20	1.21645E+18
50	1.5207E+64
100	4.6663E+157

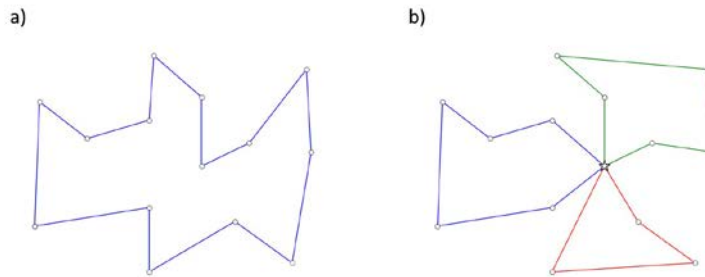


Figura 2.5: a) Los aristas entre puntos representan una posible configuración de ruta TSP b) Generalización de la misma instancia TSP a VRP, seleccionando como depósito el cliente denotado mediante una estrella e involucrando tres vehículos

que el punto de partida será el mismo para todos los recorridos). Además, se agrega una condición en la que demandas específicas q_i deben ser satisfechas en cada punto P_i (exceptuando el punto de inicio). Pudiendo especificar también el número de vehículos (agentes) a utilizar.

Bajo esta generalización, una solución al VRP consiste en un conjunto de rutas que parten de un punto en común o depósito, visitan un determinado número de clientes y luego regresan al depósito. Lo anterior bajo la restricción de que todos y cada uno de los clientes deben ser visitados una y solo una vez y buscando que la longitud acumulada de todas las rutas sea mínima. En la Figura 2.5(b) se puede observar la generalización de una instancia particular de TSP a un VRP involucrando tres vehículos.

El VRP y sus variantes constituyen una familia de problemas de optimización combinatoria NP-Difíciles [14] ampliamente estudiada por su gran número de aplicaciones a problemas reales [4, 15, 16].

2.2.1. Problema de Ruteo de Vehículos Capacitados (CVRP)

En su aplicación a situaciones reales, el VRP se ve sujeto a múltiples restricciones que son inherentes a las diversas variantes del problema. La adición de restricción de capacidad a los vehículos constituye la primera y más básica de las variantes del VRP, el *Problema de Ruteo de Vehículos Capacitados* (CVRP por sus siglas en inglés: *Capacitated Vehicle Routing Problem*). Éste puede ser modelado como un grafo en el que los vértices consisten en n clientes más un depósito único. Cada cliente debe servirse con una cantidad q_i de bienes ($i = 1, 2, \dots, n$) del depósito [17]. Para entregar (o recolectar) estos bienes se dispone de una flota de vehículos, cada uno de los cuales tiene asociada una máxima cantidad de bienes Q que puede transportar. Una solución de CVRP consiste entonces, en una colección de rutas donde cada cliente es visitado una y sólo una vez y la demanda total para cada vehículo es menor o igual a Q . La Figura 2.6 muestra una instancia

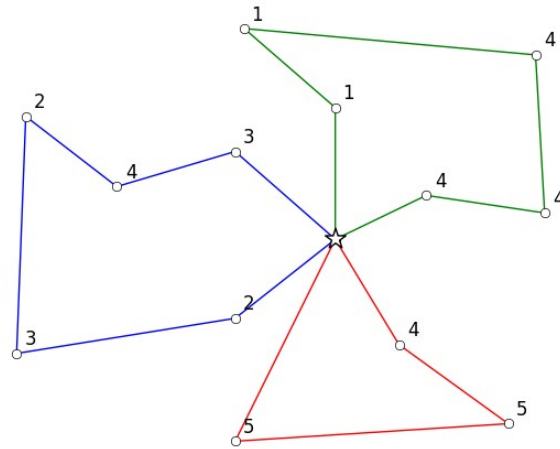


Figura 2.6: Configuración de rutas factible para CVRP con capacidad Q por vehículo igual a 15 unidades. Se especifica la demanda de cada cliente, ninguna ruta sobrepasa Q .

CVRP con una configuración de rutas factible, es decir, se satisfacen las demandas de todos los clientes y los vehículos no sobrepasan su capacidad máxima Q . De igual manera que en el VRP, en el CVRP se busca minimizar la distancia total recorrida por el conjunto de rutas.

2.2.2. Problema de Ruteo de Vehículos con Balanceo de Rutas (VRPRB)

El *Problema de Ruteo de Vehículos con Balanceo de Rutas* (VRPRB por sus siglas en inglés: *Vehicle Routing Problem with Route Balancing*) constituye una variante bi-objetivo del CVRP en la cual, además de buscar minimizar la distancia total recorrida por los vehículos respetando la restricción de capacidad máxima de los mismos, se busca lograr un balance entre las rutas, esto en términos de algún indicador como puede ser la carga total del vehículo o la longitud recorrida por cada ruta, siendo este último quizá el criterio más comúnmente utilizado. Con esto se busca introducir cierto grado de *justicia* a la solución sin descuidar el aspecto económico del problema [17].

Diversas *funciones objetivo* (FOs) han sido definidas en la literatura del VRPRB para tratar de captar el balance entre un conjunto de soluciones. Por ejemplo, la FO Max-Min busca minimizar la diferencia entre el indicador³ de la ruta más larga y el de la más corta (véase sección 2.2.3). La Figura 2.7(a) muestra la

³Para este estudio se utilizó la longitud de la ruta como valor indicador.

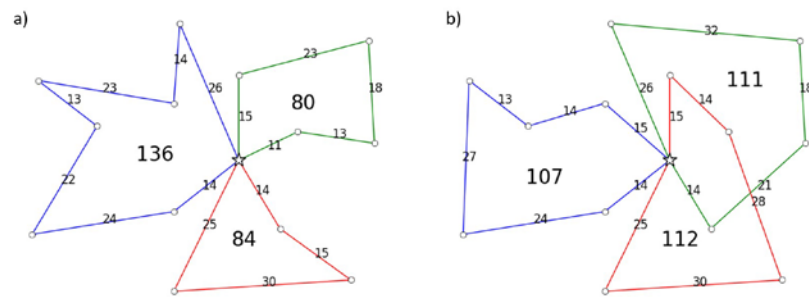


Figura 2.7: a) El balance para esta solución, considerando la FO Max-Min, sería de 56 ($136 - 80$) b) El balance de esta solución sería de 5 ($112 - 107$)

misma solución presentada en Figura 2.6 pero considerando ahora también su balance en términos de la FO Max-Min mientras que la figura Figura 2.7(b) muestra un conjunto de rutas mejor balanceadas para el mismo problema.

El VRPRB constituye un problema de gran relevancia ya que permite a una organización tratar a sus empleados con equidad, asignando cargas de trabajo similares a los conductores en términos, por ejemplo, de la longitud total que tienen que conducir. Esto representa una gran ventaja para las instituciones ya que el capital humano es un factor clave para el éxito de cualquier empresa.

2.2.3. Funciones Objetivo para balance entre rutas

Con el propósito de medir el balance entre un conjunto de rutas y contar así con una métrica que permita guiar la convergencia para el segundo objetivo del VRPRB (el balance), se han utilizado diferentes funciones objetivo (FOs). En [1] los autores realizan una revisión de la literatura del VRPRB e identifican seis tipos distintos de FOs para medir el balance entre las rutas, mismas que se enlistan en la Tabla 2.2, donde:

- T es el conjunto de rutas
- l_u es la longitud de la ruta u
- \bar{l} es un promedio anticipado, únicamente se utiliza en la FO REL' (véase tabla 2.2)

En [2], los autores realizan un análisis de FOs para balanceo de rutas en el VRPRBs mediante la identificación de un conjunto de propiedades axiomáticas que una FO ideal para balance debería satisfacer. Para su estudio los autores identifican también seis diferentes FOs: MinMax, Range, Standard Deviation (Std_Dev),

Tabla 2.2: Funciones Objetivo para balanceo en VRPRB identificadas en [1]

Nombre	Descripción	Expresión Matemática
Max/MinMax	Minimiza la longitud de la ruta más larga	$\min \max_{u \in T} l_u$
Max-Min	Minimiza la diferencia entre la ruta más larga y la más corta Minimiza la diferencia acumulada (entre todos los vehículos)	$\min(\max_{u \in T} l_u - \min_{u \in T} l_u)$
All-Min	entre la longitud de cada ruta y la longitud de la ruta más corta	$\min \sum_{t \in T} (l_t - \min_{u \in T} l_u)$
VAR	Minimiza la varianza entre las longitudes de las rutas	$\min \left(\frac{\sum_{t \in T} l_t^2}{ T } - \left(\frac{\sum_{t \in T} l_t}{ T } \right)^2 \right)$
REL	Minimiza la desviación relativa de la longitud de cada ruta respecto a la longitud de la ruta más larga	$\min \frac{1}{ T } \sum_{t \in T} \left(\frac{\max_{u \in T} l_u - l_t}{\max_{u \in T} l_u} \right)$
REL'	Minimiza la suma de la desviación absoluta de las longitudes de cada ruta respecto a un promedio anticipado,(parámetro predefinido)	$\min \sum_{t \in T} l_t - l' $

Lexicographic Min-max (Lex_Max), Mean Absolute Deviation (MAD) y Gini Coefficient (Gini). De dichas seis FOs dos coinciden con las identificadas en [1]: MinMax y Range, que corresponden Max y Max-Min respectivamente y que consisten quizá de las dos FOs más ampliamente usadas en la literatura del VRPRB. Así pues, se enlistan en la Tabla 2.3 cuatro FOs adicionales a las de la Tabla 2.2: Std_Dev, Lex_Max, MAD y Gini.

En la revisión de literatura sobre FOs para VRPRB, en [1] los autores reportan un solo trabajo en el que se hiciera uso de la FO REL' para medir balance [16]. En dicho trabajo se utilizó la carga de los vehículos como valor indicador de las rutas en lugar de la longitud de las mismas como es el caso en este estudio. Además, el uso de un parámetro preestablecido l' (promedio anticipado), que debe ser introducido por un usuario con cierto conocimiento de la instancia en cuestión, implica cierto grado de dependencia en el factor humano que a su vez puede introducir sesgo en el desempeño de una meta-heurística al utilizar dicha FO. Esta dependencia en el factor humano es, en cierto modo, erradicada por la FO MAD [2], en la que en lugar de medir la desviación absoluta respecto a un promedio anticipado l' se mide la desviación relativa respecto al promedio \bar{l} del conjunto de las longitudes de rutas en cuestión. Si se busca ajustar el parámetro l' para cada instancia VRP al usar la función objetivo REL', se pierde la objetividad de la comparación que se busca llevar a cabo en esta investigación, ya que se estaría beneficiando a la calidad de las aproximaciones a Frente de Pareto obtenidas mediante esta FO. Por otro lado, utilizar un valor arbitrario para l' conduce a resultados

Tabla 2.3: Funciones Objetivo para balanceo en VRPRB identificadas en [2]

Nombre	Descripción	Expresión Matemática
Std_Dev	Minimiza la desviación estándar de las longitudes de las rutas Además de minimizar la ruta más larga, se busca minimizar la segunda ruta más larga (sujeta a la minimización de la primera),	$\min \sqrt{\frac{1}{ T } \sum_{t \in T} (l_t - \bar{l})^2}$
Lex_Max	así como la tercera más larga (sujeta a la minimización de la segunda), hasta la $t - \text{sim}a$ más larga (sujeta a la minimización de la $(t - 1) - \text{sim}a$), donde t es igual al número de rutas ($ T $)	$\text{lex min } \{(l_1, l_2, \dots, l_t) : t \in T\}$
MAD	Minimiza el promedio de las diferencias entre la longitud de cada ruta respecto al promedio de las longitudes de las rutas	$\min \frac{1}{ T } \sum_{t \in T} l_t - \bar{l} $
Gini	Minimiza el coeficiente de Gini, uno de los índices más utilizados en estudios de equidad en general [2]	$\min \frac{1}{2 T ^2} \sum_{t \in T} \sum_{t' \in T} l_t - l_{t'} $

de tan pobre calidad que resultan de nula utilidad⁴ para los fines de este estudio. Debido a estas razones, se optó por descartar la FO REL' para el desarrollo de esta investigación.

2.3. Algoritmos Genéticos

En esta sección se pretende dar al lector un panorama general sobre el funcionamiento de los Algoritmos Genéticos (AGs). Para una visión más a fondo véase el trabajo de Colin R. Reeves [18] y D. Whitley [19].

Introducidos por John Holland [20] en 1975, los AGs consisten en una familia de meta-heurísticas que pretenden simular el proceso de evolución *neo-Darwineana* mediante el uso de operadores de selección, recombinación y mutación sobre un conjunto (*población*) de potenciales soluciones a un problema específico codificadas en una estructura de datos particular utilizando un *alfabeto* determinado (simulando la estructura de un *cromosoma*). Por alfabeto se entiende el conjunto de caracteres utilizados para codificar las soluciones en un AG. Generalmente se utilizan alfabetos binarios ya que representan ciertas ventajas por su relativa simpleza, sin embargo, alfabetos no-binarios también pueden ser utilizados [19]. En la Figura 2.8 se muestra un ejemplo de la codificación de una solución para un algoritmo genético mediante un alfabeto binario: se trata de una cadena de 10 *alelos* (espacios) ocupados por símbolos (ceros o unos), que constituye el *genotipo*

⁴Experimentos preliminares mostraron que al usar un valor arbitrario para el parámetro l al usar la FO REL' conduce a resultados de calidad muy pobre: aproximaciones a Frente de Pareto excesivamente pobladas, constituidas por soluciones con muy altos valores objetivo generando por tanto hipervolumenes muy bajos. (véase sección 2.1.1)

0	1	1	1	1	0	1	0	0	1
---	---	---	---	---	---	---	---	---	---

Figura 2.8: Codificación de una solución consistente en una cadena de 10 alelos usando un alfabeto binario

(haciendo alusión a los términos utilizados en Genética) o información genética de la solución. El *fenotipo* o expresión del genotipo para esta solución depende del esquema de codificación. Si dicho esquema es por ejemplo, la conversión de decimal a binario, entonces el fenotipo de la solución en la Figura 2.8 sería el número 489, su equivalente en decimal.

La ejecución de un AG genético comienza con la creación de una población de soluciones iniciales. Generalmente, este conjunto inicial de individuos se crea de manera aleatoria, aunque existe también la posibilidad de *inyectar* la población inicial con soluciones de calidad. Algunos autores han demostrado que introducir buenas soluciones a la población inicial puede mejorar el desempeño de un AG [21], mientras que otras investigaciones indican que esta práctica puede conducir a una convergencia prematura a óptimos locales [22].

Luego de ser creadas, cada una de las soluciones iniciales es evaluada mediante una *función objetivo* (FO) (véase sección 2.2.3), que provee de una medida de la aptitud de la solución en base a un conjunto particular de parámetros o variables de la misma. Como resultado de la evaluación mediante la FO, a cada solución se le asigna una probabilidad de reproducción o *fitness*, de manera que las mejores soluciones tengan mayor oportunidad de ser elegidas para generar nuevos individuos.

El AG entra entonces en un proceso de dos etapas, primero la *población actual* pasa por un proceso de selección para determinar el conjunto de individuos que participarán en la siguiente etapa, en los procesos de *recombinación* y *mutación*, para crear la *población siguiente*. La selección de individuos puede llevarse a cabo de diversas maneras. En la *selección por ruleta* por ejemplo, se crea una distribución de probabilidad entre las soluciones en la que cada solución cuenta con una probabilidad de ser seleccionada proporcional a su *fitness*, luego se elige a la solución en base a un número generado aleatoriamente. El proceso debe su nombre a la analogía de girar una ruleta en la que a cada solución le corresponde un área proporcional a su *fitness*. Un ejemplo de selección por ruleta puede apreciarse en la Figura 2.9. Otro método de selección es la *selección por torneo*, en la que se eligen al azar n individuos de la población y se elige de ellas la que tenga mayor *fitness*. Ambos métodos de selección están sujetos a efectos estocásticos por lo que ninguno garantiza la participación de todos los individuos en el proceso.

Una vez concluido el proceso de selección, el AG continúa con el proceso de *recombinación* o *crossover*. El proceso de *crossover* consiste en generar nuevos individuos a partir de las soluciones seleccionadas en la

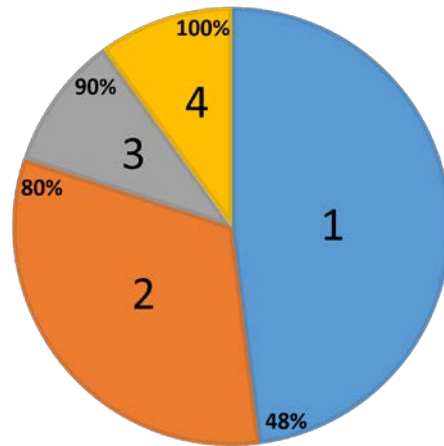


Figura 2.9: Selección por ruleta.- Suponiendo que se generara un número aleatorio igual a .52, la solución elegida sería la número 2

etapa anterior (*padres*) mediante la combinación de partes de las mismas en una forma específica, buscando retener información valiosa de ambos padres para generar *hijos* de mayor calidad. Una de las formas más sencillas de *crossover* es el *crossover de un punto*, en la que se elige un punto al azar entre 1 y la longitud del individuo seleccionado, luego, un primer hijo se conforma con la primer parte del primer padre (desde el inicio hasta el punto aleatorio seleccionado), y la segunda parte del segundo padre (desde el punto aleatorio hasta el final del segundo padre). Un segundo hijo se crea de manera inversa. La Figura 2.10(a) muestra un ejemplo del proceso de *crossover* de un punto. La estrategia anterior puede extenderse a n puntos de cruce de manera similar, intercalando las partes seleccionadas de los padres para generar a los hijos, tal y como puede apreciarse en la Figura 2.10(b).

Otra alternativa utilizada es el *crossover por máscara*, en la que se genera de manera aleatoria una cadena binaria de la misma longitud que una solución. Cada posición en esta cadena indica de cual padre se tomará la información en cada posición para generar al primer hijo, el segundo hijo se genera de manera inversa. La figura Figura 2.11 muestra un ejemplo de este tipo de *crossover*. El tipo de *crossover* depende del esquema de codificación de soluciones. Por ejemplo, los tres tipos de *crossover* antes descritos trabajan con un esquema de codificación con un alfabeto binario. Operadores de *crossover* específicos pueden ser diseñados para trabajar con esquemas de codificación más complejos.

Después del proceso de recombinación, cada nuevo individuo está sujeto a una pequeña probabilidad de *mutar*. En AGs, de manera análoga a como sucede en el proceso evolutivo natural, las mutaciones consisten en cambios *aleatorios* en la configuración genética del individuo que se dan con muy poca probabilidad con el paso de las generaciones. Para esquemas de codificación binarios generalmente la mutación consiste

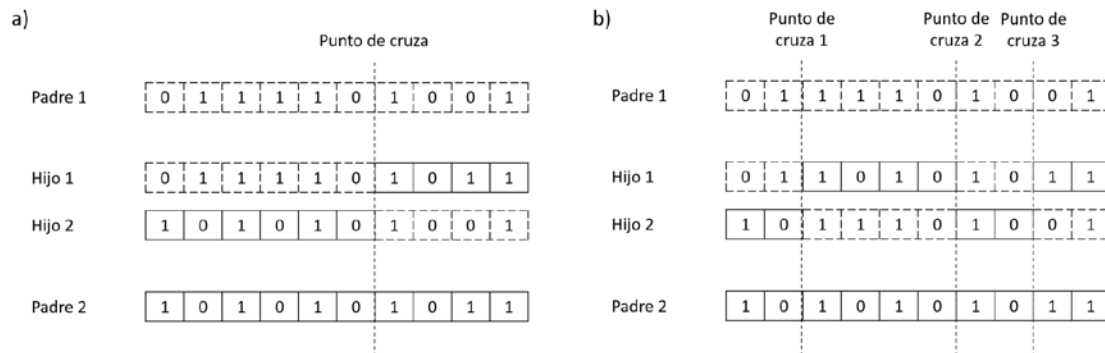


Figura 2.10: a) Crossover de 1 punto.- el punto de cruza es en la posición 6. El Hijo 1 se compone de la primera parte del Padre 1 (línea discontinua) y la segunda parte del Padre 2 (línea continua), el Hijo 2 se construye de forma inversa. b) Crossover de 3 puntos con puntos de cruza en las posiciones 2, 6 y 8. El Hijo 1 se compone de la primera y tercera sección del Padre 1 (línea discontinua) y la segunda y cuarta sección del Padre 2 (línea continua). El Hijo 2 se construye de forma inversa

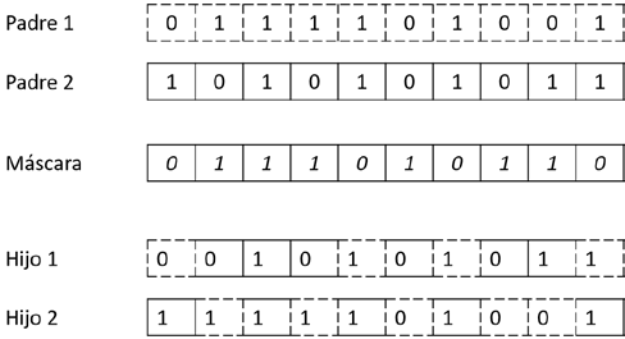


Figura 2.11: Crossover por Máscara.- Un cero en la máscara indica que la información para ese alelo en la construcción del Hijo 1 será tomada del primer padre y la información para ese alelo en la construcción del Hijo 2 será tomada del segundo padre. Un uno en la máscara indica la operación inversa

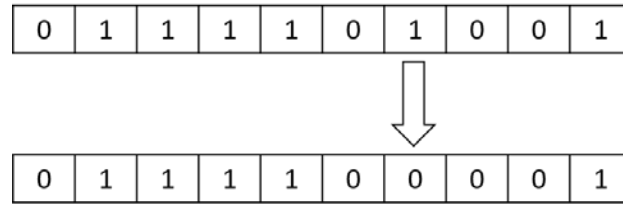


Figura 2.12: Mutación.- El valor del séptimo alelo cambia de 1 a 0

en cambiar el valor de uno de los alelos de la solución, tal y como puede apreciarse en la Figura 2.12. Al igual que en el caso del *crossover*, pueden definirse operadores de mutación específicos para esquemas de codificación más complejos.

Este proceso completo de evaluación, selección, recombinación y mutación forma una generación en la ejecución de un AG. Conceptualmente, el paso de las generaciones podría continuar de manera indefinida, pero en la práctica se requiere la definición de un criterio de parada en términos de número de generaciones, tiempo de ejecución o convergencia del objetivo a optimizar. En Algoritmo 1 se presenta la estructura de un AG básico.

Algoritmo 1 Algoritmo Genético Básico

```

1: procedure ALGORITMOGENETICO
2:   poblacion_inicial ← poblacion generada aleatoriamente
3:   evaluar_poblacion(poblacion_inicial)
4:   while ! CRITERIO_DE_PARADA do
5:     repeat
6:       hijos ← {}
7:       padres ← seleccion_de_padres(poblacion_inicial)
8:       hijos ← {hijos} ∪ {crossover(padres)}
9:       if CRITERIO_PARA_MUTACION then
10:        mutacion(hijos)
11:      until SUFICIENTE_DESCENDENCIA_CREADA
12:      evaluar_poblacion(hijos, padres)
13:      poblacion_inicial ← seleccionar_nueva_poblacion(hijos, padres)

```

Los AGs permiten realizar una exploración *no exhaustiva* (no se prueban todas las posibles soluciones) del espacio de búsqueda de manera inteligente, basándose en el principio de la supervivencia de los individuos más aptos durante el paso de las generaciones. Si bien, no garantizan encontrar siempre la solución

óptima, permiten encontrar soluciones de alta calidad en tiempos de ejecución sustancialmente menores a los requeridos por otras aproximaciones como los métodos exactos [5]. En problemas multi-objetivo como el VRPRB, en los que se pretende encontrar un conjunto de soluciones óptimas, en lugar de una única solución óptima, los AGs constituyen una familia de meta-heurísticas especialmente bien adaptadas ya que, al operar sobre un conjunto de soluciones potenciales, permiten una mayor exploración del espacio de búsqueda. Diversas investigaciones han demostrado la efectividad de los AGs en su aplicación a variantes multi-objetivo del VRPRB [7, 23, 24, 25, 26, 27, 11].

2.3.1. Algoritmo Evolutivo (1+1)

El objetivo de la investigación propuesta es el de realizar una comparación cuantitativa de la calidad de las soluciones generadas para un VRPRB por una meta-heurística evolutiva al utilizar diferentes FOs para medir el balanceo de las soluciones. A fin de introducir el menor sesgo posible y obtener resultados objetivos, se optó por implementar la versión más sencilla de un AG, el llamado *Algoritmo Evolutivo (1+1)* (AE (1+1)), el cual consiste en la mutación iterativa de un solo individuo *sol* creado de manera aleatoria. El individuo *sol* es reemplazado por el individuo mutado *sol'* únicamente cuando éste último es al menos tan bueno como el primero. Considerando la naturaleza bi-objetivo del VRPRB, se seleccionó una variante del AE (1+1) tal y como es propuesta por los autores en [26] en la que se incluye un archivo externo para guardar las soluciones no-dominadas que se encuentran durante el proceso de evolución de tal manera que un individuo mutado *sol'* se acepta únicamente cuando no es dominado por ninguno de los individuos que ya se encuentran en el archivo. Cuando un nuevo individuo *sol'* es aceptado, todos los individuos dominados por *sol'*, así como todos aquellos con los mismos valores objetivo, son eliminados del archivo. El AE (1+1) con archivo externo es formalmente descrito en Algoritmo 2.

Algoritmo 2 AE (1+1) con archivo externo

```

1: procedure RESOLVERINSTANCIA
2:    $sol \leftarrow$  solucion generada aleatoriamente
3:    $A \leftarrow \{sol\}$ 
4:    $iter \leftarrow 0$ 
5:   while  $iter < MAX\_ITER$  do
6:      $sol' \leftarrow$  operadorDeMutacion( $sol$ )
7:     if  $\nexists \hat{sol} \in A : \hat{sol} \prec sol'$  then
8:        $A \leftarrow \left\{ \hat{sol} \in A : sol' \not\prec \hat{sol} \wedge f(\hat{sol}) \neq f(sol') \right\} \cup \{sol'\}$ 
9:        $sol \leftarrow sol'$ 
10:     $iter \leftarrow iter + 1$ 

```



3

METODOLOGÍA

En el presente capítulo se describen todas las actividades que se llevarán a cabo para la culminación de cada uno de los objetivos específicos y, por tanto, del objetivo general de la investigación.

Durante el proceso de revisión de literatura del VRPRB fue de notarse la falta de consenso en cuanto a la FO utilizada para medir el balanceo entre rutas. Dentro de los trabajos revisados en una primera etapa, pudieron identificarse dos FO diferentes: Max-Min [28, 11, 9, 29] y MinMax [1]. Ninguna de las investigaciones revisadas proporciona argumentos sólidos para la selección de la FO utilizada. Más aún, en [1], se identifican otras cuatro FOs para medir balanceo entre rutas en un VRPRB.

De la situación descrita anteriormente surgieron varias interrogantes:

- ¿Por qué diferentes investigaciones utilizan diferentes FOs para medir el balanceo entre rutas en un VRPRB?
- Además de la complejidad de implementación, ¿existen otras diferencias al variar entre FOs?
- ¿Cuál FO conduce a mejores resultados?

La revisión de la literatura se enfocó entonces a la búsqueda de investigaciones en torno a las diferentes FOs utilizadas para medir el balanceo entre rutas en el VRPRB. Se identificaron dos trabajos con esta temática: [1] y [2]. En [1] los autores hacen una amplia revisión de literatura del VRPRB e identifican seis diferentes FOs (véase 2.2.3), proveen la definición matemática de cada una de ellas así como algunas de sus características, todo esto para argumentar la selección de la FO MinMax, o Max como se hace referencia en el texto. Aunque los autores llevan a cabo una decisión informada, el análisis sobre las diferentes FOs es básico, basado en simples observaciones y sin pruebas objetivas de las ventajas o desventajas de usar una u otra. En [2], los autores realizan un análisis a mayor profundidad, definiendo una serie de axiomas que

una FO para balanceo debería cumplir idealmente (véase 2.2.3). Los autores identifican seis FOs para balanceo entre rutas, cuatro adicionales a las identificadas por [1], y las prueban sobre 60 instancias pequeñas de VRPRB, construyendo todas las posibles soluciones con cada FO, para concluir en que ninguna cumple con todos los axiomas, por tanto, ninguna FO es mejor que las demás en todos los aspectos. A pesar de tratarse de un estudio a mayor profundidad, considerando propiedades matemáticas y analizando a detalle cada FO, la metodología de solución no es la comúnmente usada para las instancias de VRPRB conocidas en la literatura, o instancias de problemas reales, ya que, debido a su naturaleza altamente combinatoria, resulta prácticamente imposible construir todas las posibles soluciones. Por tanto, el análisis de FOs realizado en [2], si bien constituye una valiosa aportación a la literatura del problema, no responde las interrogantes que surgieron al inicio de la investigación, sigue siendo una incógnita si alguna de las FO, al ser utilizada en una meta-heurística, conduce a mejores resultados que las demás.

Habiendo identificado nueve FO diferentes¹, se procedió a seleccionar una herramienta adecuada para llevar a cabo la comparación que permitiera cumplir con los objetivos de esta investigación. Entre las meta-heurísticas más ampliamente utilizadas para VRPRB se encuentran las de naturaleza evolutiva, como los *algoritmos genéticos* (AGs). Entre la gran variedad de meta-heurísticas evolutivas pueden destacarse implementaciones como el NSGA-II que utiliza un operador de selección en la que se eligen los mejores individuos entre padres e hijos en base a su *fitness* y a que tan aisladas se encuentran (debido a que una solución aislada representa información importante referente a un espacio de búsqueda no explorado), además de proponer un método de ordenamiento de no-dominancia con complejidad computacional $O(M N^2)$ [30]. En [25], los autores proponen una versión de NSGA-II con paralelización mediante un modelo de islas aplicado al VRPRB obteniendo resultados de buena calidad. Otra implementación exitosa es el MOEA/D, la cual descompone un PMO en un número n de sub-problemas escalares y los optimiza de manera simultánea [31]. Estos y otros algoritmos resultaban una opción atractiva para ser utilizados como medio de comparación en el presente estudio. Sin embargo, debido a la complejidad de las implementaciones, la calidad de las soluciones obtenidas no podría ser directamente atribuida a la selección de la FO para balanceo, lo que es crucial para lograr la comparación objetiva que es objeto de este estudio. Más aún, la naturaleza de la implementación podría beneficiar o perjudicar los resultados obtenidos al usar una u otra FO. Por tal motivo se optó por buscar una implementación más sencilla, a fin de poder atribuir la calidad de las soluciones a la selección de FO con mayor objetividad y poder realizar una comparación más imparcial. Se optó entonces por la versión más simple de un algoritmo evolutivo, el Algoritmo Evolutivo (1+1) (AE (1+1)) (véase 2.3.1), en su versión con archivo externo para poder alojar el conjunto de soluciones que formen parte de la aproximación a Frente de

¹La FO REL' [1] se excluyó de este estudio (véase sección 2.2.3)

Pareto encontrada.

Para medir la calidad de las aproximaciones a Frente de Pareto encontradas, se utilizará la métrica de hipervolumen (véase sección 2.1.1). La ejecución del algoritmo implementado sobre una instancia en particular de VRPRB dará como resultado un conjunto de soluciones que constituyen la aproximación a Frente de Pareto. Dicho conjunto de soluciones será almacenado en un archivo de texto para ser utilizado en las posteriores comparaciones. Diferentes FOs implican diferentes espacios objetivo [2]. Por tanto, no se podrían comparar de manera directa, por ejemplo, dos soluciones que fueron generadas una con la FO Max-Min y la otra con la FO VAR, ya que cada solución mediría su balance en proporciones distintas. Más aún, no se podrían comparar directamente dos aproximaciones a Frente de Pareto generadas con FOs diferentes. Para llevar a cabo una comparación justa, se optará por la transformación de cada una de las soluciones de las aproximaciones a Frente de Pareto a un espacio objetivo común: el de la FO Max-Min por tratarse de la más ampliamente usada en la literatura.

Debido al componente estocástico del algoritmo que se implementará (el operador de mutación trabajará de manera aleatoria), será necesario contar con un número apropiado de ejecuciones del algoritmo, de manera que los promedios de hipervolumenes por FO puedan ser comparados mediante una prueba estadística. Para determinar si los promedios de hipervolumenes por FO son significativamente diferentes y así poder determinar, por ejemplo, que una FO condujo a resultados de mayor calidad que otra, se utilizará una prueba *t-test*.

Para concluir con la investigación, se aplicará un esquema de puntajes en el que, al comparar el desempeño del algoritmo al utilizar dos FO sobre una instancia en particular, se sumará 1 al puntaje de la FO que presente un promedio de hipervolumenes significativamente mayor según la prueba *t-test* y se restará 1 a la FO con menor promedio según la prueba. Si los promedios de hipervolumenes obtenidos por las dos FOs no presentan una diferencia significativa, sus puntajes se mantienen igual. Este esquema de puntajes nos permitirá visualizar de manera más clara si existe una FO en particular que conduzca a mejores resultados al ser utilizada en una meta-heurística evolutiva simple como la implementada en este estudio y responder así a los cuestionamientos que dieron origen a la presente investigación.



4

IMPLEMENTACIÓN DE ALGORITMO EVOLUTIVO 1+1

Tal y como se mencionó en el apartado de Marco Teórico del presente documento, se seleccionó como meta-heurística para el estudio el llamado *Algoritmo Evolutivo (1+1)*, en adelante AE (1+1) (véase sección 2.3.1). Se trata de la implementación más básica de una meta-heurística evolutiva en la que se trabaja con una solución única *sol*, la cual es iterativamente alterada mediante un operador de mutación. La solución *sol* es sustituida por la solución alterada *sol'* únicamente cuando ésta última es estrictamente mejor que la primera. Se optó por una meta-heurística con estas características para evitar cualquier sesgo en favor de alguna de las FO bajo estudio que pudiera ser introducido por la complejidad de meta-heurísticas evolutivas más eficientes como el MOEA/D [31] o el NSGA-II [30].

Debido a la naturaleza multi-objetivo del VRPRB, se implementó la versión AE (1+1) con archivo externo, tal y como es propuesta por los autores de [32]. En esta variante se trabaja también con una solución única, pero además se mantiene un archivo externo en el que se van almacenando las soluciones no dominadas encontradas durante las iteraciones del algoritmo de manera que la solución inicial *sol* es sustituida por la solución alterada *sol'* si y solo si ésta última no es dominada por ninguna de las soluciones en el archivo. Así pues, se implementó el algoritmo especificado en (Algoritmo 2). Se estableció a 30 mil el número de iteraciones *MAX_ITER*, una cantidad suficientemente grande de iteraciones para permitir la convergencia del algoritmo.

Se implementaron dos diferentes operadores de mutación: el operador de mutación SWAP y el operador de mutación REVERSE. Contar con dos operadores de mutación distintos permite comparar los resultados obtenidos a través de las FOs, pudiendo así determinar si los comportamientos con ambas variantes son similares o no. De ser similares podría atribuirse, con mayor objetividad, la calidad de los

resultados obtenidos a la selección de la FO en cuestión, ya que se estarían observando comportamientos similares independientemente del operador de mutación utilizado. En caso contrario, si los comportamientos entre las dos versiones del algoritmo presentan diferencias considerables, podría pensarse que la calidad de los resultados está siendo influenciada por el operador de mutación.

Con el objeto de alcanzar un balance adecuado entre *exploración* (buscar en una larga porción del espacio de búsqueda tratando de encontrar áreas prometedoras) y *explotación* (buscar en un área limitada y prometedora del espacio de búsqueda con el objeto de mejorar soluciones de alta calidad), se implementó un proceso de transformación de espacio objetivo entre VRP y TSP. Dicho proceso, los operadores de mutación SWAP y REVERSE, la heurística utilizada para generar la solución inicial así como la heurística utilizada para elegir el punto z de referencia para calcular el hipervolumen se describen en las siguientes secciones.

4.1. Heurística para construcción de la solución inicial

Para crear la solución inicial se genera una permutación aleatoria de los clientes de la instancia VRP en cuestión. Luego, uno a uno, los clientes son incluidos en una primer ruta hasta que no puedan agregarse más clientes debido a la restricción de capacidad de los vehículos. Cuando ya no pueden incluirse más clientes en la primer ruta, se genera una segunda ruta para insertar los clientes subsecuentes. Este proceso continúa hasta que todos los clientes han sido incluidos en una ruta.

Se generó una solución por cada una de las 60 instancias a utilizar en la etapa de experimentos, dado que todas las ejecuciones del algoritmo, utilizando cada una de las nueve FO bajo estudio, deberán partir de una solución inicial aleatoria idéntica, a fin de que el punto de partida del algoritmo sea el mismo para todos los casos.

4.2. Transformación de espacios de búsqueda VRP a TSP

Dado que el AE (1+1) sólo utiliza un operador de mutación (no utiliza operadores de selección o *crossover*), resulta complicado alcanzar un balance adecuado entre exploración y explotación. Con el objeto de alcanzar este balance exploración-explotación se utiliza en este estudio una transformación de espacio de búsqueda entre VRP y TSP tal y como se lleva a cabo en trabajos como [11] o [29]. Para lograr esto, justo antes de que la solución sea alterada por el operador de mutación, todas las rutas individuales de la solución de VRPRB son concatenadas para formar una permutación de TSP, es decir, una lista que contiene a todos los clientes.

Al operar sobre una permutación de clientes se trabaja indistintamente entre varias rutas como en cada

ruta individualmente. Operar entre varias rutas conduce a la creación de nuevas rutas, con la posibilidad de mover al algoritmo a un espacio de búsqueda bastante distinto (exploración). Operar en el interior de una ruta permite hacer *afinaciones* a la misma, pequeñas mejoras que tiendan a *perfeccionar* una solución de alta calidad (explotación).

Una vez que se lleva a cabo la perturbación, la solución es nuevamente transformada al espacio de búsqueda VRP mediante un proceso similar al utilizado para generar la solución inicial, con la diferencia de que en lugar de tener una permutación aleatoria de clientes se tiene una permutación que fue resultado del proceso de mutación.

4.3. Operador de mutación SWAP

Este operador consiste en intercambiar de lugar un número determinado de clientes de una solución TSP. Primero, la solución VRP es transformada al espacio de búsqueda TSP mediante el proceso descrito en la sección 4.2. Luego, un número aleatorio n de clientes para intercambiar es elegido (para este estudio n fue delimitado entre 2 y 4). Después, n posiciones de la permutación de clientes son seleccionadas e intercambiadas de manera aleatoria. Finalmente, la permutación de clientes resultante es transformada de nuevo a una solución en el espacio de búsqueda VRP.

4.4. Operador de mutación REVERSE

Este operador consiste en invertir (voltar) un segmento de la lista de clientes de una solución TSP. Primero, la solución VRP es transformada al espacio de búsqueda TSP mediante el proceso descrito en la sección 4.2. Luego, se invierte un segmento de la lista seleccionado de manera aleatoria. Finalmente, la permutación de clientes resultante es transformada de nuevo a una solución en el espacio de búsqueda VRP.

4.5. Heurística para selección de punto de referencia z para cálculo de hipervolumen

Para medir la calidad de las aproximaciones a Frente de Pareto obtenidas por el algoritmo implementado se utilizará la métrica de hipervolumen [13] (véase sección 2.1.1). Para el cálculo de dicha métrica se requiere la selección de un punto de referencia z en el espacio objetivo. En problemas de minimización, una práctica común es elegir los peores valores posibles en cada uno de los objetivos como coordenadas del punto de referencia. Para este estudio, el punto de referencia se calcula como se describe a continuación.

Para el balance, se construye una solución con solo dos rutas, en la primera ruta se incluyen todos los clientes exceptuando el último, el cual es incluido en la segunda ruta. De esta manera se obtiene una solución cuyo desbalance será muy alto. Para la distancia, se construye una solución con un solo cliente por ruta, lo cual constituye la configuración de rutas con la mayor distancia total recorrida posible. Mediante estas dos estrategias se obtiene un punto de referencia con valores objetivo extremadamente malos.

Es importante señalar que, a fin de obtener una medida de hipervolumen igualmente influenciada por los dos objetivos, los valores objetivo para cada solución en la aproximación a Frente de Pareto son normalizados en un rango entre 0 y 1.

Nótese que ni el AE (1+1), ni los operadores de mutación utilizados (SWAP y REVERSE) cuentan con alguna estrategia para hacer las rutas individuales óptimas en distancia (TSP óptimas), por tanto, las soluciones generadas por la meta-heurística implementada para este estudio podrían estar balanceadas de manera artificial mediante el incremento de la longitud de rutas pequeñas dentro de una solución, tal y como es señalado por los autores en [11]. Nuevamente, con la intención de mantener la objetividad de la comparación que es objeto de esta investigación, se optó por no introducir alguna estrategia de optimización de distancia en rutas individuales, a fin de evitar cualquier sesgo que dicha estrategia pudiera introducir para el beneficio o perjuicio de los resultados obtenidos al utilizar alguna de las FOs bajo estudio.



5

EXPERIMENTOS

En este capítulo se describen a detalle los experimentos que fueron llevados a cabo con el fin de comparar estadísticamente la calidad de los resultados obtenidos por la meta-heurística implementada al utilizar las diferentes FO bajo estudio.

5.1. Ejecución del algoritmo AE (1+1) sobre las instancias VRP

Para esta etapa se seleccionó un conjunto de instancias VRP recientemente propuestas en [2]. El conjunto consta de 60 instancias de VRP de un número reducido de clientes (14 clientes por instancia), creadas a partir de las bien conocidas instancias de Christofides *et al.* [10] (véase sección 1.6.1).

El algoritmo descrito en la sección 2.3.1 (Algoritmo 2) fue codificado en el lenguaje de programación Java. Para su ejecución se utilizó una computadora portátil con un procesador a 2.50 GHz y 6 GB de RAM. Se consideraron dos variantes del algoritmo por separado: la primera utilizando el operador de mutación SWAP y la segunda utilizando el operador de mutación REVERSE. Además, cada versión del algoritmo fue ejecutada usando cada una de las nueve FO bajo estudio, de tal manera que se probaron 18 configuraciones distintas (18 combinaciones FO-operador de mutación distintas). Cada una de las 18 configuraciones del algoritmo se ejecutó en 30 ocasiones sobre cada una de las 60 instancias VRP. Al final de cada ejecución, se obtuvo un archivo de texto con el conjunto de soluciones que formaron parte de la aproximación a Frente de Pareto.

5.2. Transformación de soluciones a un espacio objetivo común

De las ejecuciones del algoritmo AE (1+1) en sus dos variantes y con cada una de las nueve FOs bajo estudio, para cada una de las 60 instancias VRP de prueba se obtuvieron 30 aproximaciones a Frente de Pareto. Dado que cada FO trabaja en un espacio objetivo distinto en cuestiones del balance, para poder comparar estadísticamente la calidad de las aproximaciones obtenidas, fue necesario seleccionar un espacio objetivo común, en el que las aproximaciones obtenidas con diferentes FOs puedan ser comparadas objetivamente. Se seleccionó entonces, el espacio objetivo de la función Max-Min por tratarse quizá de la FO más ampliamente usada en la literatura del VRPRB. Cada solución en las aproximaciones a Frente de Pareto fue re-evaluada utilizando la FO Max-Min, luego, el hipervolumen de la aproximación fue calculado en el espacio objetivo común, permitiendo una comparación justa.

5.3. Prueba *t-test* para comparar estadísticamente promedios de vectores de hipervolumenes

Una vez re-evaluadas las aproximaciones, se procedió a comparar los vectores compuestos de 30 hipervolumenes obtenidos por cada una de las combinaciones FO-operador de mutación del algoritmo sobre las 60 instancias. Para este fin, se utilizó una prueba *t-test*, a un nivel de confianza de 95 % entre pares de vectores. Esta prueba permite determinar si los promedios de dos grupos de datos presentan una diferencia significativa, sin ser influida por valores atípicos (como sí lo es en ocasiones la media convencional) ya que la significancia estadística que arroja una prueba *t-test* no es determinada únicamente por el tamaño de la diferencia entre los promedios de los grupos de datos, sino también por el tamaño de la muestra y las desviaciones estándar de los mismos.

5.4. Esquema de puntajes en prueba *t-test* entre FOs

Para comparar los resultados obtenidos de las pruebas *t-test* se utilizó el siguiente esquema de puntajes entre FOs. Al comparar dos vectores de hipervolumenes obtenidos al utilizar dos FOs diferentes, con un mismo operador de mutación y sobre una instancia en particular, si la prueba *t-test* muestra una diferencia significativa a un nivel de confianza de 95 %, se incrementa 1 al puntaje de la FO con el mayor promedio y se decrementa -1 a la otra FO. Si no existe diferencia estadísticamente significativa entre los promedios de los vectores de hipervolumenes, los puntajes se dejan tal como estaban.



6

RESULTADOS

En este capítulo se presentan los resultados obtenidos en la etapa de experimentación de la investigación así como un análisis de los aspectos más sobresalientes que fueron observados.

6.1. Condensado de resultados de pruebas *t-test* por instancia

Para cada instancia y por cada uno de los dos operadores de mutación utilizados, se generó una matriz para condensar los resultados de las pruebas *t-test* de la siguiente manera. La intersección entre una columna y un renglón de la matriz representa el resultado de la prueba *t-test* entre la FO correspondiente al renglón y la FO correspondiente a la columna. Si la prueba *t-test* mostró que el promedio de los hipervolúmenes obtenidos mediante la FO correspondiente al renglón resultaba significativamente mayor al promedio de los hipervolúmenes obtenidos mediante la FO correspondiente a la columna, se asignó el valor 1 a la intersección, en caso contrario se asignó un -1. Si los promedios de los hipervolúmenes no presentaron diferencia significativa, se asignó 0 a la intersección. De esta manera, la suma de los valores por renglón representó el puntaje total obtenido por la FO correspondiente a ese renglón. Nótese que la diagonal principal de la matriz se queda sin valores puesto que representa una comparación de una FO contra sí misma, las intersecciones correspondientes a estos casos fueron denotados con un guión medio (-).

Por ejemplo, la tabla 6.1 muestra el condensado de resultados de las pruebas *t-test* para la instancia #29 usando el operador de mutación REVERSE. Podemos observar que la FO REL obtuvo un promedio de hipervolúmenes sin diferencias estadísticamente significativas a los obtenidos por las FOs All-Min y Max-Min, superior al de la FO MinMax e inferior a los de las FOs VAR, MAD, Gini, Lex_Max y Std_Dev, obteniendo así un puntaje total de -4.

La tabla 6.2 muestra el condensado de pruebas *t-test* para la misma instancia (#29), pero para el ope-

Tabla 6.1: Condensado de resultados de pruebas *t-test* para instancia 29 con operador de mutación REVERSE.

	All-Min	Max-Min	MinMax	REL	VAR	MAD	Gini	Lex_Max	Std_Dev	Puntaje Total
All-Min	-	0	1	0	-1	-1	-1	-1	-1	-4
Max-Min	0	-	1	0	-1	-1	-1	-1	-1	-4
MinMax	-1	-1	-	-1	-1	-1	-1	-1	-1	-8
REL	0	0	1	-	-1	-1	-1	-1	-1	-4
VAR	1	1	1	1	-	-1	0	1	0	4
MAD	1	1	1	1	1	-	0	1	1	7
Gini	1	1	1	1	0	0	-	1	0	5
Lex_Max	1	1	1	1	-1	-1	-1	-	-1	0
Std_Dev	1	1	1	1	0	-1	0	1	-	4

Tabla 6.2: Condensado de resultados de pruebas *t-test* para instancia 29 con operador de mutación SWAP.

	All-Min	Max-Min	MinMax	REL	VAR	MAD	Gini	Lex_Max	Std_Dev	Puntaje Total
All-Min	-	0	1	0	-1	-1	-1	1	-1	-2
Max-Min	0	-	1	0	-1	-1	-1	1	-1	-2
MinMax	-1	-1	-	-1	-1	-1	-1	-1	-1	-8
REL	0	0	1	-	-1	-1	-1	1	-1	-2
VAR	1	1	1	1	-	0	0	1	0	5
MAD	1	1	1	1	0	-	0	1	0	5
Gini	1	1	1	1	0	0	-	1	0	5
Lex_Max	-1	-1	1	-1	-1	-1	-1	-	-1	-6
Std_Dev	1	1	1	1	0	0	0	1	-	5

rador de mutación SWAP. En esta tabla podemos observar por ejemplo, que las FOs VAR, MAD, Gini y Std_Dev obtuvieron, entre ellas mismas, promedios de hipervolumenes indiferentes (denotado por ceros en las intersecciones entre estas FOs) mientras que superaron los promedios de hipervolumenes del resto de las FOs: All-Min, Max-Min, MinMax, REL y Lex_Max, obteniendo de esta manera, un puntaje final de cinco unidades cada una.

6.2. Puntajes acumulados por FOs

Los tablas de resultados condensados por cada instancia con cada operador de mutación fueron integradas en dos tablas de puntajes acumulados obtenidos por cada FO por instancia. La tabla 6.3 muestra los puntajes acumulados por las FOs en las instancias 1 a 30 y la tabla 6.4 en las instancias 31 a 60. Para una mejor apreciación de la información, por cada instancia y por cada operador de mutación, se marcó el puntaje más alto en negritas.

Por ejemplo, para la instancia #29 y considerando el operador de mutación REVERSE, la FO MAD

obtiene un puntaje acumulado de 7, ya que supera los promedios de hipervolumenes de las FOs All-Min, Max-Min, MinMax, REL, VAR, Lex_Max y Std_Dev (como puede apreciarse en la tabla 6.1 de puntajes condensados por instancia), por tanto se agregan siete unidades a su puntaje acumulado. En relación a la FO Gini, la prueba *t-test* no mostró diferencia significativa en los promedios de hipervolumenes, así que el puntaje se queda tal como estaba. De esta manera, la FO MAD obtiene el mayor puntaje entre las nueve FOs al utilizar el operador de mutación REVERSE para la instancia #29.

Para esa misma instancia pero considerando el operador de mutación SWAP se observan resultados diferentes ya que, tal y como fue descrito en la sección 6.1, las FOs VAR, MAD, Gini y Std_Dev obtuvieron todas un puntaje máximo de cinco unidades.

6.3. Análisis de resultados

En esta sección se realiza una discusión exhaustiva de los resultados obtenidos en la etapa de experimentación, acentuando las situaciones particulares que dan respuesta a las interrogantes que dieron origen a la presente investigación, así como también señalando nuevas incógnitas que surgen de manera natural con los nuevos descubrimientos y que constituyen grandes áreas de oportunidad para futuras investigaciones. Los principales puntos a resaltar se describen en las siguientes sub-secciones.

6.3.1. Comportamientos distintos al utilizar diferentes FOs por instancia

Lo primero que es importante señalar es la considerable diferencia en el desempeño del algoritmo al utilizar las diferentes FOs bajo estudio, lo cual es evidente en los puntajes obtenidos por instancia, considerando por separado los dos operadores de mutación. Exceptuando algunos casos particulares en los que la selección de la FO parece no influir significativamente en los resultados obtenidos por el algoritmo (véase sección 6.3.4), en más del 95 % de las ocasiones, las diferentes FO obtuvieron puntajes muy distintos, lo que implica variaciones importantes en la calidad de las aproximaciones a Frente de Pareto que cada FO permitió alcanzar. De entrada, esta situación constituye un primer argumento sólido en favor de la comprobación de la hipótesis planteada al inicio de esta investigación: *"la calidad de las soluciones generadas por un AG para VRPRB se ve influida por la selección de la FO utilizada para medir el balanceo entre rutas"*.

6.3.2. Comportamientos similares entre FOs al utilizar los dos operadores de mutación

Otro aspecto que salta a la vista en las tablas de resultados es la gran similitud en el comportamiento del algoritmo al utilizar los dos operadores de mutación sobre las 60 instancias VRP. Comportamientos tan

Tabla 6.3: Puntajes acumulados por cada FO por operador de mutación sobre las instancias 1 a 30.

Instancia	Operador REVERSE										Operador SWAP									
	All-Min	Max-Min	MinMax	REL	VAR	MAD	Gini	Lex_Max	Std	Dev	All-Min	Max-Min	MinMax	REL	VAR	MAD	Gini	Lex_Max	Std	Dev
1	1	1	-8	-2	2	1	1	2	2	1	1	-8	1	1	1	1	1	1	1	1
2	-3	-3	-8	-3	5	5	5	-3	5	-2	-2	-8	-2	5	5	5	5	-6	5	5
3	-3	-3	-8	-3	5	6	3	-2	5	-3	-3	-8	-3	5	4	5	5	-3	6	6
4	-1	-1	-8	-1	-2	6	6	-5	6	-1	-2	-8	-3	5	5	5	5	-6	5	5
5	-5	-2	-8	-4	5	4	7	-1	4	-2	-2	-8	-2	5	5	5	5	-6	5	5
6	-6	-3	-8	-3	5	5	5	0	5	-4	-3	-8	-1	6	5	5	5	-4	4	4
7	1	-1	-8	2	1	1	1	2	1	1	0	-8	1	2	2	4	0	-2	2	-2
8	-3	-2	-8	-3	2	7	1	-1	7	-3	-3	-8	-3	5	5	5	5	-3	5	5
9	-3	-3	-8	-3	5	5	5	-3	5	-3	-3	-8	-3	5	5	5	5	-3	5	5
10	-5	-3	-7	-4	5	5	5	-1	5	-2	-2	-7	-2	4	5	6	-7	5	5	5
11	-4	-4	-8	-4	5	5	5	0	5	-2	-2	-8	-2	5	6	5	-6	4	4	4
12	-4	-2	-8	-5	5	5	5	-1	5	-3	-3	-8	-3	6	3	6	-3	5	5	5
13	0	0	-2	0	-2	0	2	2	0	-1	1	-6	0	-1	0	1	0	6	6	6
14	-3	-3	-8	-3	5	5	5	-3	5	-2	-2	-8	-3	5	4	6	-5	5	5	5
15	-4	-4	-4	-4	4	5	7	-4	4	-2	-3	-8	-3	5	5	5	-4	5	5	5
16	-3	-3	-7	-4	5	5	5	-3	5	-2	-2	-7	-2	5	5	5	-7	5	5	5
17	-4	-4	-8	-4	5	5	5	0	5	-2	-1	-8	-3	5	5	5	-6	5	5	5
18	-3	-3	-8	-3	5	5	5	-3	5	-3	-3	-8	-3	6	5	5	-3	4	4	4
19	-1	2	-8	-3	2	4	1	4	-1	1	3	-8	0	1	1	1	0	1	1	1
20	-3	-2	-8	-2	5	5	5	-5	5	-2	-2	-8	-2	5	5	5	-6	5	5	5
21	-3	-3	-8	-3	4	5	4	-3	7	-2	-2	-8	1	2	5	4	-4	4	4	4
22	-4	-3	-8	-4	5	4	6	-1	5	-2	-2	-8	-2	5	5	5	-6	5	5	5
23	-5	-4	-6	-2	5	5	5	-3	5	-2	-3	-7	-1	5	5	5	-7	5	5	5
24	-4	-3	-4	-6	5	5	5	-3	5	-4	0	-8	-4	6	5	4	-4	5	5	5
25	-1	-1	-7	-2	7	-1	7	-1	-1	0	1	-8	4	1	0	1	-1	2	2	2
26	-3	-3	-8	-3	4	6	5	-3	5	-2	-2	-8	-2	5	5	5	-6	5	5	5
27	-3	-3	-8	-3	5	5	5	-3	5	-3	-3	-8	-3	5	5	5	-3	5	5	5
28	-4	-4	-8	-4	5	5	5	0	5	-2	-2	-8	-2	5	5	5	-6	5	5	5
29	-4	-4	-8	-4	4	7	5	0	4	-2	-2	-8	-2	5	5	5	-6	5	5	5
30	-4	-4	-8	-4	6	3	5	0	6	-3	-1	-8	-4	6	5	5	-4	4	4	4

Tabla 6.4: Puntajes acumulados por cada FO por operador de mutación sobre las instancias 31 a 60.

Instancia	Operador REVERSE									Operador SWAP								
	All-Min	Max-Min	MinMax	REL	VAR	MAD	Gini	Lex_Max	Std Dev	All-Min	Max-Min	MinMax	REL	VAR	MAD	Gini	Lex_Max	Std Dev
31	-1	0	-8	0	0	1	0	8	0	1	1	-8	1	1	1	1	1	1
32	-3	-3	-8	-3	5	5	5	-3	5	-3	-3	-8	-3	5	4	4	-3	7
33	-4	-4	-8	-3	4	5	5	0	5	-3	-3	-8	-3	5	5	5	-3	5
34	-4	-4	-8	-4	5	5	5	0	5	-2	-2	-8	-2	5	5	5	-6	5
35	-4	-4	-8	-4	5	4	6	0	5	-2	-2	-8	-2	5	5	5	-6	5
36	-3	-4	-8	-5	5	3	6	0	6	-1	-2	-8	-4	5	5	5	-5	5
37	1	1	-8	1	1	1	1	1	1	2	1	-8	1	1	0	1	1	1
38	-4	-4	-4	-4	5	5	5	-4	5	-3	-3	-7	-3	5	5	5	-4	5
39	-2	-3	-8	-4	5	5	5	-3	5	-2	-3	-8	-2	3	5	5	-3	5
40	-2	-2	-8	-2	-2	6	6	-2	6	-2	-2	-8	-2	5	5	5	-6	5
41	-4	-4	-7	-5	5	5	5	0	5	-3	-3	-8	-3	5	5	5	-3	5
42	-2	-4	-8	-5	5	5	5	-1	5	-1	-3	-8	-4	5	5	5	-4	5
43	-1	0	-8	0	0	0	0	7	2	1	1	-8	1	2	1	-1	1	2
44	-6	-4	-3	-4	5	5	5	-3	5	-3	-3	-3	-4	5	5	5	-7	5
45	-3	-3	-8	-3	5	5	5	-3	5	-3	-3	-8	-2	5	5	5	-4	5
46	-5	-4	-5	-5	5	5	5	-1	5	-2	-2	-7	-2	5	5	5	-7	5
47	-4	-4	-5	-5	5	5	5	-2	5	-2	-2	-7	-2	3	4	6	-7	7
48	-4	-4	-8	-4	5	5	5	0	5	-3	-3	-8	-3	5	5	5	-3	5
49	1	1	-8	1	1	0	1	2	1	1	1	-8	1	1	1	1	1	1
50	-3	-3	-8	-3	5	5	5	-3	5	-2	-2	-8	-2	5	5	5	-6	5
51	-3	-3	-8	-3	5	5	5	-3	5	-2	-2	-8	-3	5	5	5	-5	5
52	-1	-1	-8	-1	-6	3	8	0	6	-2	-3	-8	-1	5	5	5	-6	5
53	-3	-3	-8	-3	5	5	5	-3	5	-2	-2	-8	-2	5	5	5	-6	5
54	-2	-4	-8	-5	5	1	6	2	5	-3	-3	-8	-3	5	5	6	-3	4
55	-1	-5	-8	2	2	3	2	3	2	1	1	-8	1	1	1	1	1	1
56	-3	-3	-8	-3	5	5	5	-3	5	-2	-2	-8	-2	5	5	5	-6	5
57	-3	-3	-8	-3	5	5	5	-3	5	-3	-3	-8	-3	4	7	4	-3	5
58	-4	-4	-8	-4	5	5	5	0	5	-2	-2	-7	-2	5	5	5	-7	5
59	-5	-2	-8	-4	5	5	5	-1	5	-2	-2	-8	-2	5	5	5	-6	5
60	-4	-2	-8	-5	4	7	4	-1	5	-4	0	-8	-4	5	5	5	-4	5

similares al utilizar dos operadores de mutación considerablemente diferentes nos otorga cierta certeza de que la calidad de los resultados obtenidos por el algoritmo está, en efecto, siendo influida en mayor medida por la selección de la FO que por las características del algoritmo en sí. Esta situación permite expresar los resultados y conclusiones con mayor objetividad ya que, exceptuando algunos casos particulares en los que para una misma instancia diferentes FOs obtienen el mayor puntaje dependiendo del operador de mutación que se use (véase sección 6.3.6), en más del 88 % de ocasiones un mismo subconjunto de FOs, o un subconjunto muy similar de estas, obtiene el mayor puntaje independientemente del operador de mutación utilizado.

6.3.3. Bajo desempeño del algoritmo en ambas variantes al usar FO MinMax

Otro aspecto que atrae atención es la clara desventaja de la FO MinMax en comparación con el resto de las FOs, ya que en ambos operadores de mutación pueden observarse únicamente resultados negativos y en la mayoría de los casos se trata de un puntaje de -8 unidades, lo que significa que fue superada por todas las FOs (el promedio de los hipervolumenes obtenidos mediante esta FO fue menor al obtenido mediante el resto de las FOs). Esta situación resulta particularmente interesante ya que se trata de una FO bastante utilizada en la literatura [1, 33, 34, 35].

Este bajo desempeño de la FO MinMax podría deberse a lo siguiente. Remontando a la sección 2.2.3, esta FO busca minimizar la longitud de la ruta más larga de la solución. Por tal motivo, para generar una nueva (y potencialmente mejor) solución, es necesario que el operador de mutación altere directamente la ruta de mayor longitud, lo cual no siempre ocurre debido a la naturaleza estocástica del algoritmo. De esta manera, un porcentaje indeterminado de alteraciones a la solución es despreciado por no modificar directamente a la ruta de mayor longitud, incluso cuando se trata de soluciones potencialmente mejor balanceadas en otros espacios de búsqueda.

Es evidente entonces que, si se utilizara un operador de mutación especializado, que buscara alterar directamente la ruta de mayor longitud, el desempeño del algoritmo al utilizar la FO MinMax podría mejorar considerablemente. Mas tal y como se ha acentuado a lo largo de este documento, el objetivo de esta investigación es determinar si alguna FO conduce a resultados de mayor calidad meramente por su uso y con la menor influencia de las características de la meta-heurística en que es utilizada.

Bajo esta perspectiva, la FO MinMax representa una desventaja evidente en comparación al resto de las FOs bajo estudio, tanto que pierde sentido considerarla en las observaciones realizadas. Así por ejemplo, para la instancia #1, considerando el operador de mutación SWAP, se afirma que las FOs All-Min, Max-Min, REL, VAR, MAD, Gini, Lex_Max y Std_Dev, obtuvieron el mismo puntaje, sin prestar especial atención a

que todas superaron a la FO MinMax en términos de las pruebas *t-test*, considerando por tanto, que se trató de un caso en el que la selección de la FO no impactó significativamente en la calidad de los resultados obtenidos.

Haciendo referencia a la sección 1.4, podemos comenzar a dar respuesta a la tercer pregunta de investigación: *¿Son las FOs más ampliamente utilizadas en la literatura del VRPRB las que permiten obtener mejores resultados?*. En base a la contundente evidencia observada en torno a la FO MinMax podemos argumentar que este no es el caso, al menos en lo que respecta al mero uso de la FO, habiendo aclarado que una meta-heurística especializada para trabajar específicamente con esta FO podría mejorar la calidad de las soluciones obtenidas.

6.3.4. Empates entre todas las FOs

Resulta interesante que en algunas instancias, todas las FOs exceptuando MinMax (véase sección 6.3.3), obtienen el puntaje más alto, lo que significa que ninguna FO mostró una ventaja clara sobre las demás en términos de la calidad de las aproximaciones a Frente de Pareto obtenidas.

Por ejemplo, en las instancias #1, #31, #49 y #55, mediante el operador de mutación SWAP, todas las FOs excepto MinMax obtienen un máximo puntaje de 1, implicando que ninguna representó una ventaja considerable al ser utilizada para esa instancia en particular (fuera de superar a MinMax). Para estas mismas 4 instancias, al utilizar el operador de mutación REVERSE, se dan situaciones muy cercanas al empate entre todas las FOs, en las que los puntajes más altos no sobrepasan las 3 unidades, exceptuando los casos particulares de las instancias #31 y #43, en los que la FO Lex_Max superó a todas las demás FOs al utilizarse en conjunto con el operador de mutación REVERSE, obteniendo un puntaje final de 8 y 7 unidades respectivamente (véase sección 6.3.7). En cuanto al operador de mutación REVERSE, únicamente en la instancia #37 se da una situación de empate entre todas las FOs, también con resultados cercanos al empate al utilizar el operador de mutación SWAP.

Lo anterior indica que para algunas instancias en particular, la selección de la FO no impacta de gran manera en la calidad de los resultados obtenidos. Las características de las instancias VRP que conducen a esta situación particular constituyen todavía una incógnita que queda fuera del alcance de esta investigación pero constituye un tema de gran interés para trabajos futuros.

6.3.5. VAR, MAD, Gini y Std_Dev con mayor puntaje en más del 86 % de las comparaciones

En poco más del 86 % de las ocasiones, las FOs VAR, MAD, Gini y Std_Dev o un subconjunto de estas cuatro, constituyen el conjunto exclusivo de FOs que obtienen el puntaje más alto respecto a las pruebas *t-*

test. En gran cantidad de estos casos, las FOs VAR, MAD, Gini y Std_Dev empatan con un máximo puntaje de 5 unidades, significando que superaron a las otras 5 FOs pero fueron indiferentes entre ellas mismas.

Para algunas instancias esto sucede solo al utilizar un operador de mutación en particular, por ejemplo las instancias #6, #10, #14 y #47 con el operador de mutación REVERSE y las instancias #5, #8, #15 y #22 con el operador de mutación SWAP, por mencionar algunos casos. En muchas otras instancias, este empate entre las cuatro funciones objetivo se da para ambos operadores de mutación, como sucede con las instancias #9, #16, #17, #20 y #23 entre otras. Esto podría significar que, de cierto modo, las FOs VAR, MAD, Gini y Std_Dev sí representan cierta ventaja al ser utilizadas para medir el balanceo entre rutas en una meta-heurística evolutiva aplicada al VRPRB, en cuanto a la calidad de las aproximaciones a Frente de Pareto obtenidas considerando la métrica de hipervolumen, reforzando así la hipótesis planteada en la presente investigación.

Es interesante que sean estas cuatro FOs en específico las que presentan un mejor desempeño en cuanto a las pruebas llevadas a cabo, ya que se trata de cuatro de las cinco FOs que involucran para su cálculo medidas de tendencia central, en este caso los promedios, de las longitudes de las rutas en una solución, o de desviaciones de las mismas¹. Tal y como se describió en la sección 2.2.3:

- VAR involucra el **promedio** de los cuadrados de las longitudes de las rutas, así como el **promedio** cuadrado de las longitudes de las rutas
- Std_Dev involucra el **promedio** de las diferencias al cuadrado entre la longitud de cada ruta y la longitud **promedio** de las rutas en la solución
- MAD involucra el **promedio** de la desviación absoluta de las longitudes de las rutas respecto al **promedio** de las rutas en la solución
- Gini involucra el **promedio** de las desviaciones entre todos los pares de rutas en la solución

Este detalle podría marcar la pauta para una nueva hipótesis: *"las FOs para balanceo que involucran para su cálculo medidas de tendencia central, conducen a mejores resultados que aquellas que trabajan con valores absolutos"*.

La única excepción sería la FO REL, que también involucra para su cálculo el promedio de la desviación relativa de la longitud de cada ruta respecto a la longitud de la ruta más larga. Quizá esto último sea la causa de su bajo desempeño²: por el hecho de tomar como referencia la longitud de la ruta más larga, una solución

¹Las otras cuatro FOs (MinMax, All-Min, REL y Lex_Max) involucran únicamente los valores absolutos de las longitudes de las rutas

²REL resultó entre las FOs con mayor puntaje en menos de 10 de las 120 contiendas entre FOs

cuya ruta más corta es incrementada en distancia se consideraría *mejor balanceada* ya que dicha ruta se acercaría en distancia a la ruta más larga, disminuyendo su desviación relativa y por tanto obteniendo un mejor valor objetivo en balance. Esta situación podría conducir a aproximaciones a Frentes de Pareto con soluciones "*bien balanceadas*" a expensas de un deterioro innecesario del objetivo de minimizar la distancia total recorrida, y con mayor probabilidad de incurrir en balanceo artificial al balancear configuraciones de rutas mediante el incremento de la distancia de rutas pequeñas.

6.3.6. Diferentes FOs obtienen el puntaje más alto dependiendo del operador de mutación utilizado

En algunas instancias, dependiendo del operador de mutación utilizado, diferentes FOs obtuvieron el puntaje más alto. Por ejemplo, en la instancia #3, al utilizar el operador de mutación REVERSE, la FO que obtuvo el mayor puntaje fue MAD con 6 puntos, mientras que al utilizar el operador de mutación SWAP, la FO con mayor puntaje fue Std_Dev también con 6 puntos. En la instancia #7, con el operador de mutación REVERSE, REL y Lex_Max presentaron el puntaje más alto con 2 unidades mientras que al utilizar el operador de mutación SWAP Gini obtuvo el máximo puntaje con 4 unidades. Situaciones similares se presentan en las instancias #13, #19, #21, #25, y #43. Este comportamiento distinto al trabajar sobre una misma instancia utilizando dos operadores de mutación distintos podría implicar que, debido a ciertas características de la instancia en cuestión, el operador de mutación ejerció mayor influencia en el desempeño del algoritmo. Por tanto, para la obtención de aproximaciones de Frente de Pareto de calidad en dichas instancias influyen más las características del algoritmo utilizado que la FO para balanceo que se utilice. Se desconocen las características de las instancias VRP que conducen a esta situación particular, mas constituyen un tema de gran importancia para futuras investigaciones.

6.3.7. Casos particulares: instancias #31 y #43

Dos casos particulares llaman la atención: las instancias #31 y #43. En ambos casos, al utilizar el operador de mutación SWAP, parece no haber diferencia entre utilizar cualquiera de las FOs (exceptuando MinMax). En la instancia #31 todas las FOs empatan con 1 punto, mientras que en la instancia #43 las dos FOs con puntaje más alto, VAR y Std_Dev, obtienen solo 2 puntos. Sería de esperarse una situación similar al usar el operador de mutación REVERSE, tal y como sucedió en la gran mayoría de los casos en que se presentaban empates entre todas las FOs. Sin embargo, al utilizar el operador de mutación REVERSE en estas dos instancias, la FO Lex_Max obtiene un puntaje total de 8 y 7 puntos respectivamente. Esto constituye una situación bastante peculiar ya que, de entrada, solo en 11 ocasiones una FO obtuvo un puntaje de

7 unidades y solo en 2 ocasiones una FO obtuvo 8 puntos, lo que significa que superó a todas las FOs. No sería extraño encontrar en estos casos esporádicos a FOs como VAR, MAD, Gini o Std_Dev, que como ya se ha mencionado, resultan ser las FOs que demuestran conducir a resultados de mayor calidad. No obstante se trata de la FO Lex_Max que no mostró un desempeño particularmente bueno en la etapa de experimentación pero que en estas dos instancias, y en combinación con el operador de mutación REVERSE superó en términos de las pruebas *t-test* a todas las FOs contrincantes.

Así pues, las características de estas dos instancias en particular, hacen que la combinación entre el operador de mutación REVERSE y la FO Lex_Max resulte particularmente atractiva, lo que genera nuevas interrogantes:

- ¿Qué combinaciones de operador de mutación-FO conducen a mejores resultados?
- ¿Qué características de las instancias VRP permiten elegir de manera adecuada esta combinación?

7.1. Condensado de resultados

El histograma 6.1 muestra gráficamente el número de ocasiones en las que cada FO, utilizando cada uno de los operadores de mutación, obtuvo el mayor puntaje por cada una de las 60 instancias VRP.

Este gráfico permite apreciar de manera visual algunos de los puntos señalados en las secciones anteriores:

- La gran similitud entre los comportamientos presentados por el algoritmo con cada FO al variar entre las dos vertientes del mismo: utilizando cada uno de los operadores de mutación
- El bajo desempeño de la FO MinMax con ambas vertientes del algoritmo, sin obtener una sola vez el puntaje más alto entre las FOs.
- La contundente ventaja mostrada por las FOs VAR, MAD, Gini y Std_Dev

Pero además pone de manifiesto de forma todavía más evidente el bajo desempeño de las FOs que involucran para su cálculo únicamente valores absolutos de las longitudes de las rutas: All-Min, Max-Min, MinMax y Lex_Max. Ya se ha hablado en párrafos anteriores de la FO MinMax y su bajo desempeño, pero el hecho de que entre estas FOs se encuentre Max-Min refuerza los argumentos para dar respuesta a una de las preguntas de investigación del presente trabajo, ya que se trata quizá de la FO más ampliamente utilizada [36, 28, 11, 9, 29, 37] pero, tal y como puede apreciarse en el histograma, no se trata de una de las FOs que conduzca a resultados de mayor calidad.

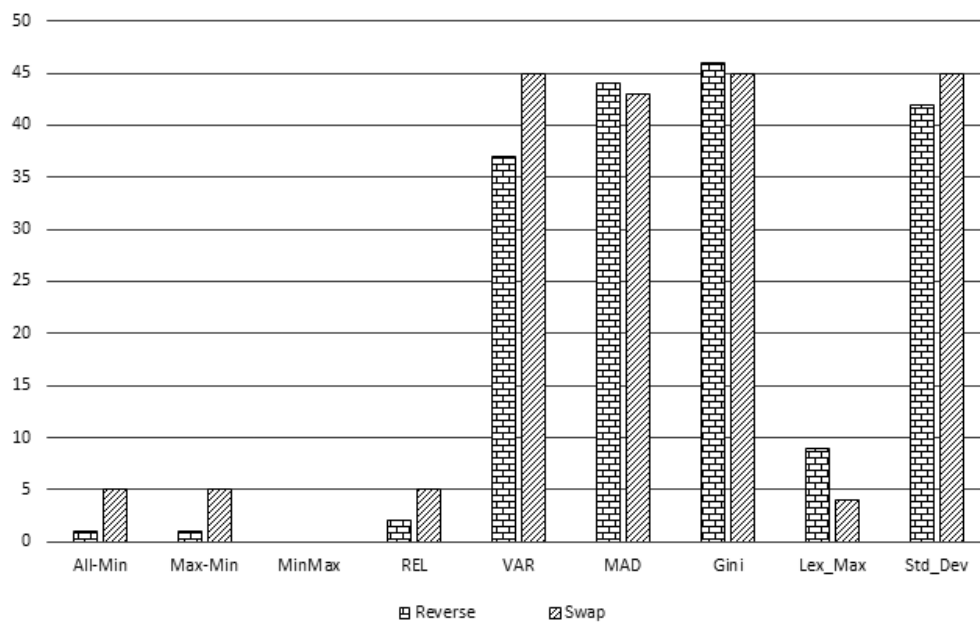


Figura 6.1: Número de ocasiones en las que una FO dada obtuvo el puntaje más alto sobre las 60 instancias usando cada uno de los operadores de mutación.

7

CONCLUSIONES

La presente investigación surgió como resultado de un minucioso análisis del estado del arte en torno al VRP y sus variantes, análisis en que terminó por identificarse un hueco en el conocimiento del área: ninguna investigación referente al VRPRB provee evidencia concreta de si el uso de alguna FO para balanceo en particular permite obtener mejores resultados o si, por el contrario, la selección de la FO para balanceo no influye en la calidad de los resultados obtenidos.

Se planteó entonces como objetivo general de la investigación el siguiente: *realizar una comparación cuantitativa de la calidad de las soluciones generadas por un AG (meta-heurística evolutiva) al utilizar las diferentes FO para balanceo de rutas en el VRPRB*". Para esto fue necesaria la conclusión de dos objetivos específicos:

- El diseño e implementación de un AG (meta-heurística evolutiva) aplicado al VRPRB con nueve de las diferentes FOs para balanceo identificadas en la literatura
- Realizar una serie de experimentos con el algoritmo implementado sobre un conjunto de instancias VRP, para analizar y comparar estadísticamente los resultados obtenidos al utilizar nueve de las diferentes FOs conocidas en la literatura

Para el primer objetivo específico se diseñó e implementó la versión más simple de un AG, el llamado *Algoritmo Evolutivo 1+1* (AE (1+1)). El motivo de selección de este algoritmo fue evitar cualquier tipo de sesgo en beneficio o prejuicio de alguna FO que la complejidad de una implementación más avanzada pudiera introducir a los resultados de la investigación. Los procesos de diseño e implementación del algoritmo se describen en el capítulo 4. Se desarrollaron dos operadores de mutación distintos: el operador REVERSE y el operador SWAP, esto con la finalidad de contar con dos variantes del algoritmo y tener una forma de corroborar que los resultados obtenidos se mantuvieran entre diferentes implementaciones.

Contando ya con una herramienta de comparación (la implementación del AE (1+1)), se procedió a una etapa de experimentación que permitiera la conclusión del segundo objetivo específico. Los detalles de dicha etapa se describen en la sección 5 mientras que las principales observaciones a partir de la experimentación se presentan en la sección 6.

En los siguientes párrafos de este apartado se presentan las conclusiones generales de la investigación, comenzando por dar respuesta puntual a cada una de las preguntas de investigación planteadas al inicio del proceso, continuando con el veredicto final respecto a la hipótesis establecida y finalizando con las nuevas interrogantes que surgieron como resultado del nuevo conocimiento adquirido.

7.1. Sobre las preguntas de investigación

A continuación se aborda de manera concreta cada una de las preguntas que dieron origen a la presente investigación, respondiendo cada una de manera objetiva de acuerdo al conocimiento generado del proceso completo, desde la revisión de literatura hasta el análisis de resultados de la etapa de experimentación.

¿Existe alguna diferencia considerable en la calidad de las soluciones generadas por un AG para VRPRB al utilizar diferentes FOs para medir el balance entre rutas?

Sí existe tal diferencia. Esto puede ser observado en el simple hecho de que las diferentes FOs evaluadas obtienen puntajes tan distintos al ser utilizadas sobre la misma instancia VRP y con un mismo operador de mutación y bajo exactamente los mismos parámetros (véanse tablas 6.3 y 6.4). Dicha situación se dio en un 95.83 % de los casos.

Estos resultados fueron generados a través de pruebas *t-test*, con un nivel de confianza de 95 %, entre pares de vectores de 30 hipervolumenes obtenidos mediante cada FO por cada instancia VRP. Por tanto se tiene la certeza de que los resultados obtenidos no están influenciados por la naturaleza estocástica del algoritmo ni por valores atípicos en los resultados obtenidos por el mismo, ya que la prueba *t-test* minimiza dicha influencia al considerar la distribución y desviación estándar del conjunto de datos.

¿Utilizar diferentes FOs para balanceo conduce a obtener soluciones diferentes? o ¿Se obtienen las mismas soluciones sin importar que FO se utilice?

De entrada, tal y como demuestran Matl *et al.* en [2], diferentes FOs permiten encontrar Frentes de Pareto (Frentes Óptimos) diferentes, por tanto, hay soluciones que no pueden encontrarse con determinadas FOs. Esto lo determinaron al construir todas las posibles soluciones para cada una de las 60 instancias que generaron para su estudio (mismas instancias que fueron utilizadas para esta investigación) determinando

así, los Frentes de Pareto reales entre este conjunto de soluciones, para luego percatarse de que algunas soluciones que aparecen en uno o varios Frentes generados mediante una FO, no aparecen en uno o varios Frentes generados mediante otra u otras FOs. Esto tiene sentido, ya que cada FO trabaja en un espacio de búsqueda distinto, cada uno con limitantes específicas, lo cual se ve reflejado en los resultados obtenidos en la presente investigación en las diferencias tan marcadas en el desempeño de las variantes del algoritmo al utilizar las diferentes FO. De llegar a las mismas soluciones, o al menos a conjuntos de soluciones similares, las pruebas *t-test* arrojarían puntajes muy cerrados al comparar los promedios de vectores de hipervolumenes alcanzados mediante las diferentes FOs.

En base a estos argumentos, la respuesta a esta interrogante es que, si bien habrá soluciones que puedan ser encontradas mediante varias FOs, los conjuntos de soluciones que forman el Frente de Pareto o Frente Óptimo de cada una no tienen que ser necesariamente iguales, más aún es bastante probable que sean diferentes.

¿Son las FOs más ampliamente utilizadas en la literatura del VRPRB las que permiten obtener mejores resultados?

Si hablamos estrictamente de la influencia de la FO por su mero uso y sin considerar algún tipo de adaptación de los operadores de la meta-heurística en que se usa entonces la respuesta es no. Las evaluaciones realizadas en esta investigación mostraron coincidencia en el comportamiento del algoritmo en sus dos variantes al utilizar las nueve FOs bajo estudio en un 88.33 % de las ocasiones, lo que significa que el comportamiento del algoritmo fue mayormente influido por el uso de la FO en particular y no por algún factor externo de la implementación. Esto nos da la certeza para afirmar que la calidad de las aproximaciones a Frentes de Pareto obtenidas se debieron en gran medida al hecho de operar sobre el espacio objetivo de la FO que se utilizó. Habiendo establecido esto basta con observar el histograma 6.1 para darse cuenta de que las dos FOs más ampliamente utilizadas en la literatura: Max-Min y MinMax no son las que conducen a resultados de mayor calidad.

7.2. Sobre la hipótesis

Tal y como puede apreciarse en las tablas de resultados 6.3 y 6.4, la selección de la FO influye de manera importante en la calidad de las soluciones encontradas por una meta-heurística evolutiva aplicada al VRPRB. Esto puede apreciarse en la variabilidad de los puntajes acumulados obtenidos por las nueve FOs bajo estudio al ser utilizadas en el algoritmo con cada uno de los operadores de mutación implementados sobre una misma instancia. Esto implica que, dependiendo de la instancia, alguna o algunas FOs permitieron alcanzar

mejores resultados que el resto de las mismas. Puede observarse incluso, y de manera más contundente en el histograma 6.1, que las FOs VAR, MAD, Gini y Std_Dev muestran un desempeño sobresaliente, obteniendo los puntajes más altos en 86.66 % de las comparaciones. Lo cual implica que fueron estas cuatro FOs las que permitieron encontrar aproximaciones a Frente de Pareto de mayor calidad en la mayoría de las instancias VRP utilizadas.

Por tanto, se cuenta con evidencia suficiente para concluir la presente investigación dando por comprobada la hipótesis planteada en sus inicios:

"La calidad de las soluciones generadas por un AG para VRPRB se ve influida por la selección de la FO utilizada para medir el balanceo entre rutas".

Sin embargo, también es evidente que esta influencia se ve afectada por ciertas características inherentes a las instancias VRP en que se trabaja. Prueba de ello es el hecho de que en diferentes instancias se observan diferentes comportamientos entre las FOs, lo que implica que determinadas características de la instancia VRP conducen a un mejor o peor desempeño del algoritmo al utilizar ciertas FOs. La identificación de dichas características constituye un tema de gran interés que dio lugar a nuevas incógnitas las cuales se presentan en la siguiente sección.

7.3. Sobre las nuevas interrogantes

Como sucede de manera natural, con la generación de nuevo conocimiento surgen nuevas interrogantes en torno al problema estudiado. La presente investigación no fue la excepción. A continuación se presentan algunas incógnitas que constituyen potenciales temas de investigación para futuros trabajos relacionados con el área de estudio.

¿Qué características inherentes a las FOs VAR, MAD, Gini y Std_Dev permiten alcanzar resultados de mayor calidad al ser estas utilizadas en una meta-heurística evolutiva?

Como ya se ha mencionado en el presente capítulo, las FOs VAR, MAD, Gini y Std_Dev mostraron un desempeño sobresaliente, conduciendo a aproximaciones a Frente de Pareto de mayor calidad que sus contrincantes en un 86.66 % de las ocasiones. La diferencia más evidente entre estas cuatro FOs y las restantes (exceptuando REL, véase sección 6.3.5) es que VAR, MAD, Gini y Std_Dev involucran para su cálculo promedios de las longitudes de las rutas, mientras que las otras FOs involucran únicamente los valores absolutos de las longitudes de las rutas.

Dicha observación da pie a la generación de una nueva hipótesis: *"las FOs para balanceo que involucran para su cálculo medidas de tendencia central, conducen a mejores resultados que aquellas que trabajan con valores absolutos"*.

¿Qué características inherentes a una instancia VRP podrían informar la selección de FO y/o operador de mutación?

La tendencia en la gran mayoría de las comparaciones realizadas entre promedios de hipervolúmenes obtenidos mediante las diferentes FOs estudiadas fue una ventaja considerable al utilizar las FOs VAR, MAD, Gini y Std_Dev. Sin embargo, existieron casos particulares en los que las FOs con mejor desempeño no fueron estas cuatro. Dichos casos dan indicios de que existen características particulares de las instancias VRP que permiten alcanzar resultados de mayor o menor calidad con determinadas FOs, con determinado operador de mutación o con determinadas combinaciones de ambos factores.

En la sección 6.3.4 se analizan casos particulares de instancias en las que aparentemente, la selección de la FO no influye de manera significativa en el comportamiento del algoritmo, ya que se obtuvieron promedios de hipervolúmenes sin diferencias estadísticamente significativas mediante todas las FOs. Conocer las características de estas instancias que conllevan a esta situación permitiría, entre otras cosas, tomar la decisión de utilizar la FO de menor complejidad computacional dado que se alcanzarían resultados con un nivel de calidad similar independientemente de cuál FO se utilice.

También en la mayoría de los casos se observaron comportamientos similares entre las FOs al variar el operador de mutación. Sin embargo, en algunas instancias las FOs mostraron comportamientos diferentes dependiendo del operador de mutación utilizado. En sección 6.3.6 se hace un análisis de los casos en los que, para una misma instancia, diferentes FOs obtuvieron el mayor puntaje al cambiar de operador de mutación. Mas aún, en la sección 6.3.7 se describen dos casos particulares en los que no solo varió la FO con mayor puntaje al cambiar de operador de mutación, sino que además dicha FO resultó ser una que en general mostró un desempeño bastante pobre. Esto podría significar que las características inherentes a las instancias VRP podrían incluso proveer información para la selección del operador de mutación o para la mejor combinación operador de mutación-FO.

7.4. Conclusión general

En la presente investigación se ha demostrado, con pruebas objetivas, que la selección de la FO para medir el balance entre rutas en una meta-heurística evolutiva aplicada al VRPRB influye significativamente

en la calidad de los resultados obtenidos. También se provee de evidencia que muestra que las FOs para balanceo más ampliamente utilizadas en la literatura del VRPRB no son las que permiten obtener resultados de mayor calidad. Además, se han planteado nuevas interrogantes en torno al problema, buscando determinar las características inherentes a las instancias VRP que permitan informar la selección tanto de FO como de operador de mutación.

Estas nuevas interrogantes constituyen un rumbo de investigación de gran importancia ya que, tal y como señalan los autores en [38], a partir de la difusión del NFLT¹ en el área de optimización: *"se ha vuelto cada vez más importante comprender la relación entre los componentes del algoritmo propuesto A y un problema de optimización dado f . Así pues, el problema f se convierte en el punto de partida para construir un algoritmo adecuado. El algoritmo de optimización necesita enfocarse específicamente en las características del problema f "*. En otras palabras, entre más se conozca el problema y como sus características pueden ayudarnos a definir un mejor algoritmo, mayor será la capacidad para resolverlo.

¹El NFLT (*No Free Lunch Theorem*) prueba que el desempeño promedio de cualquier par de algoritmos *A* y *B* entre todos los problemas posibles es idéntico. En otras palabras, no existe un *optimizador universal*.



Referencias

- [1] S. Schwarze and S. Voß, “Improved load balancing and resource utilization for the Skill Vehicle Routing Problem,” *Optimization Letters*, vol. 7, no. 8, pp. 1805–1823, 2013.
- [2] P. Matl, R. F. Hartl, and T. Vidal, “Equity Objectives in Vehicle Routing: A Survey and Analysis,” 2016.
- [3] G. B. Dantzing and J. H. Ramser, “The Truck Dispatching Problem,” *Management Science*, vol. 6, no. 1, pp. 80–91, 1959.
- [4] P. Toth and D. Vigo, “THE VEHICLE ROUTING PROBLEM,” 2002.
- [5] R. Baldacci and A. Mingozzi, “A unified exact method for solving different classes of vehicle routing problems,” *Mathematical Programming*, vol. 120, no. 2, pp. 347–380, 2009.
- [6] a. E. Rizzoli, R. Montemanni, E. Lucibello, and L. M. Gambardella, “Ant colony optimization for real-world vehicle routing problems,” *Swarm Intelligence*, vol. 1, no. 2, pp. 135–151, 2007.
- [7] W. Zhou, T. Song, F. He, and X. Liu, “Multiobjective Vehicle Routing Problem with Route Balance Based on Genetic Algorithm,” vol. 2013, 2013.
- [8] T.-R. Lee and J.-H. Ueng, “A study of vehicle routing problems with load-balancing,” *International Journal of Physical Distribution & Logistics Management*, vol. 29, no. 10, pp. 646–657, 1999.
- [9] N. Jozefowicz, F. Semet, and E.-G. Talbi, “Parallel and Hybrid Models for Multi-objective Optimization :,” pp. 271–280, 2002.
- [10] N. Christofides, A. Mingozzi, P. Toth, and C. Sandi, “The vehicle routing problem,” in *Christofides, N., Mingozzi, A., Toth, P., and Sandi, C., editors, Combinatorial Optimization*, ch. 11, pp. 315–338, 1979.
- [11] N. Jozefowicz, F. Semet, and E.-G. Talbi, “An evolutionary algorithm for the vehicle routing problem with route balancing,” *European Journal of Operational Research*, vol. 195, no. 3, pp. 761–769, 2009.

- [12] C. Yann and P. Siarry, "Introduction: multiobjective optimization and domination," in *Multiobjective Optimization*, pp. 15–43, Springer Berlin Heidelberg, 2004.
- [13] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, 2003.
- [14] J. K. Lenstra and A. H. G. Rinnooy-Kan, "Complexity of Vehicle Routing and Scheduling Problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.
- [15] N. Jozefowicz, F. Semet, and E.-G. Talbi, "Multi-objective vehicle routing problems," *European Journal of Operational Research*, vol. 189, no. 2, pp. 293–309, 2008.
- [16] M. N. Kritikos and G. Ioannou, "The balanced cargo vehicle routing problem with time windows," *International Journal of Production Economics*, vol. 123, no. 1, pp. 42–51, 2010.
- [17] A. Pessoa, M. P. D. A. Ao, and E. Uchoa, "The Vehicle Routing Problem: Latest Advances and New Challenges," *The Vehicle Routing Problem Latest Advances and New Challenges*, vol. 43, pp. 297–325, 2008.
- [18] C. R. Reeves, "Genetic Algorithms," in *Handbook of Metaheuristics* (M. Gendreau and J.-Y. Potvin, eds.), ch. 5, pp. 109–139, Springer, 2nd ed., 2010.
- [19] D. Whitley, *A Genetic Algorithm Tutorial*. 1993.
- [20] J. H. Holland, *Adaptation in Natural and Artificial Systems*, vol. Ann Arbor. 1975.
- [21] C. R. Reeves, "A genetic algorithm for flowshop sequencing," *Computers & Operations Research*, vol. 22, no. 1, pp. 5–13, 1995.
- [22] N. V. H. N. V. Huy and N. D. N. N. D. Nghia, "Solving graphical Steiner tree problem using parallel genetic algorithm," *2008 IEEE International Conference on Research, Innovation and Vision for the Future in Computing and Communication Technologies*, 2008.
- [23] J. Pasia, K. Doerner, R. Hartl, and M. Reimann *Evolutionary Computation in . . .*, vol. 4446, pp. pp 166–175, 2007.
- [24] A. Konstantinidis, S. Pericleous, and C. Charalambous, "MOEA/D for a tri-objective Vehicle Routing Problem," *IFIP Advances in Information and Communication Technology*, vol. 412, pp. 131–140, 2013.

- [25] N. Jozefowicz, “Enhancements of NSGA II and Its Application to the Vehicle Routing Problem with Route Balancing,” pp. 131–142, 2006.
- [26] A. García-Nájera, J. a. Bullinaria, and M. a. Gutiérrez-Andrade, “An evolutionary approach for multi-objective vehicle routing problems with backhauls,” *Computers & Industrial Engineering*, vol. 81, pp. 90–108, 2015.
- [27] L. Yexing, C. Xinye, F. Zhun, and Z. Qingfu, “An external archive guided multiobjective evolutionary approach based on decomposition for continuous optimization,” *Evolutionary Computation (CEC), 2014 IEEE Congress on*, vol. 19, no. 4, pp. 1124–1130, 2014.
- [28] N. Jozefowicz, F. Semet, and E. G. Talbi, “Target aiming Pareto search and its application to the vehicle routing problem with route balancing,” *Journal of Heuristics*, vol. 13, no. 5, pp. 455–469, 2007.
- [29] P. Lacomme, C. Prins, C. Prodhon, and L. Ren, “A Multi-Start Split based Path Relinking (MSSPR) approach for the vehicle routing problem with route balancing,” *Engineering Applications of Artificial Intelligence*, vol. 38, pp. 237–251, 2015.
- [30] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [31] Q. Z. Q. Zhang and H. L. H. Li, “MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition,” *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [32] M. Garza-Fabre, G. Toscano-Pulido, and E. Rodriguez-Tello, “Locality-based multiobjectivization for the HP model of protein structure prediction,” *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference. ACM, 2012*, pp. 473–480, 2012.
- [33] D. Applegate, W. Cook, S. Dash, and A. Rohe, “Solution of a Min-Max Vehicle Routing Problem,” *INFORMS Journal on Computing*, vol. 14, no. 2, pp. 132–143, 2002.
- [34] A. Corberán, E. Fernández, M. Laguna, and R. Martí, “Heuristic solutions to the problem of routing school buses with multiple objectives,” *Journal of the Operational Research Society*, vol. 53, no. 4, pp. 427–435, 2002.
- [35] C. R. Delgado Serna and J. Pacheco Bonrostro, “Minmax Vehicle Routing Problems : Application to School Transport in the Province of B urgós,” no. December 1997, 2001.

- [36] I. Borgulya, “An algorithm for the capacitated vehicle routing problem with route balancing,” *Central European Journal of Operations Research*, vol. 16, no. 4, pp. 331–343, 2008.
- [37] T. Keskinurk and M. B. Yildirim, “A genetic algorithm metaheuristic for bakery distribution vehicle routing problem with load balancing,” *INISTA 2011 - 2011 International Symposium on INnovations in Intelligent SysTems and Applications*, pp. 287–291, 2011.
- [38] F. Neri and C. Cotta, “Memetic algorithms and memetic computing optimization: A literature review,” *Swarm and Evolutionary Computation*, vol. 2, pp. 1–14, 2012.

Currículum vitae

Jairo Gibrán Lozano Carballo nace en la Ciudad de Cuauhtémoc, Chihuahua, México, en 1992. Obtuvo el título de Ingeniero de Software en 2014, egresado de la Facultad de Ingeniería de la Universidad Autónoma de Chihuahua. En 2014 inicia el posgrado de Maestría en Ingeniería en Computación en la misma institución, bajo la supervisión del Dr. Luis Carlos González Gurrola y comienza a trabajar en torno al Problema de Ruteo de Vehículos con Balanceo de Rutas. En Febrero de 2016 inicia una estancia de investigación por tres meses en *The University of Nottingham*, en Nottingham, Inglaterra, bajo la supervisión del Dr. Darío Landa Silva, periodo durante el cual realizó también una estadía de una semana en *Université Blaise Pascal* en Clermont-Ferrand, Francia, bajo la supervisión del Dr. Philippe Lacomme. La presente tesis resulta como conclusión de su trabajo durante el programa de posgrado.