

Universidad Autónoma de Nuevo León

Facultad de Ciencias Físico-Matemáticas



Tesis

Un algoritmo de búsqueda tabú bi-objetivo anidado para resolver un problema de logística verde

Presentada por

Lilian López Vera

Como requisito parcial para obtener el grado de:

Doctorado en Ciencias

con

Orientación en Matemáticas

Agosto 2021

Universidad Autónoma de Nuevo León

Facultad de Ciencias Físico-Matemáticas



Tesis

Un algoritmo de búsqueda tabú bi-objetivo anidado para resolver un problema de logística verde

Presentada por

Lilian López Vera

Como requisito parcial para obtener el grado de:

Doctorado en Ciencias

con

Orientación en Matemáticas

Agosto 2021

Universidad Autónoma de Nuevo León
Facultad de Ciencias Físico-Matemáticas
Centro de Investigación en Ciencias Físico-Matemáticas

Los miembros del comité de Tesis, por este medio, certifican que han leído la tesis presentada por la M.C. Lilian López Vera y que es totalmente adecuada en alcance y calidad como un requisito parcial para obtener el grado de Doctorado en Ciencias con Orientación en Matemáticas.

Dr. José Fernando Camacho Vallejo
Asesor Principal

Dr. José Luis Gonzalez Velarde
Co-asesor

Dra. Iris Abril Martínez Salazar
Miembro del Comité

Dr. Omar Jorge Ibarra Rojas
Miembro del Comité

Dr. Alvaro Eduardo Cordero Franco
Miembro del Comité

Vo. Bo.

Dr. Omar Jorge Ibarra Rojas
Coordinador del Posgrado en Ciencias con Orientación en Matemáticas

San Nicolás de los Garza, Nuevo León, México
Agosto 2021

Contenido

1	Introducción	1
1.1	Impacto ambiental y motivación	2
1.2	Problemática y contribución	3
1.3	Objetivos	5
1.4	Metodología	5
1.5	Estructura de la tesis	7
2	Revisión de Literatura	8
2.1	Programación Binivel	8
2.2	Problemas de logística o cadena de suministro bajo un enfoque binivel	9
2.3	Logística verde	10
3	Modelo Matemático	14
3.1	Descripción del problema	14
3.2	Modelo matemático	14
3.3	Enfoque optimista	20
3.4	Análisis de la reformulación a un modelo de un solo nivel	21
4	Algoritmo propuesto	23
4.1	Búsqueda tabú	24
4.2	Un algoritmo bi-objetivo anidado de búsqueda tabú	25
4.2.1	Codificación de la solución	25
4.2.2	Construcción de una solución inicial	26
4.2.3	Reparando una solución infactible	26
4.2.4	Evaluando la función objetivo del líder	27
4.2.5	Descripción del algoritmo NBOTS	27
5	Experimentación Computacional	33
5.1	Instancias y ambiente computacional	33
5.2	Descripción de un algoritmo voraz	34
5.3	Resultados numéricos	35
5.4	Evaluando la calidad de las aproximaciones a la frontera de Pareto	39
6	Conclusiones	41
6.1	Futuras direcciones de investigación	42

CONTENIDO

iii

Bibliografía

50

Capítulo 1

Introducción

En esta tesis doctoral nos enfocamos en estudiar un problema de logística verde. Los problemas clásicos de logística son problemas de producción y distribución de productos para atender a un conjunto de clientes, los cuales poseen una demanda específica que debe ser cubierta por una o varias plantas productoras. Usualmente, cada planta posee una capacidad de producción y otra de distribución. Un objetivo para estos problemas puede ser el de satisfacer a la mayor cantidad de clientes posible.

Esta tesis aborda un problema de logística verde, el cual además, considera la preocupación por el medio ambiente y el impacto que la producción y distribución de productos tiene al generar emisiones de CO_2 y liberarlas en el mismo. Dicho impacto se reduce al limitar la cantidad de clientes a los que hay que surtir. Este problema busca encontrar un balance entre la ganancia y la creciente preocupación por el medio ambiente.

Por lo tanto, este tipo de problemas de logística verde que buscan dicho balance poseen la característica de tener diversas decisiones, las cuales conducen a considerar objetivos en conflicto entre sí. Comúnmente, dichos objetivos suelen ser la ganancia de la compañía y la contaminación de emisiones de CO_2 . Es por esto, que resulta conveniente abordar este problema mediante programación bi-objetivo, en donde se buscará optimizar a ambas partes y encontrar un conjunto de soluciones que ayuden al tomador de decisiones a satisfacer sus necesidades de búsqueda.

Al mismo tiempo, en este problema de logística existe la problemática inherente a las diversas compañías o departamentos que intervienen para llevar un producto desde su fabricación hasta el cliente final. Dichas empresas o departamentos poseen un rango de jerarquía entre ellos. Es claro que, la decisión de un departamento afectará a la decisión del siguiente departamento a cargo. La jerarquía mencionada, proporciona la clave para modelar el problema utilizando la programación binivel.

También, es cierto que los problemas de logística pueden ser modelados como problemas de programación binivel, en donde en el nivel superior se toman las decisiones referentes a la selección de los clientes y la distribución de los productos hacia cada uno de ellos, por ejemplo: [Huang and Liu, 2004], [Safaei et al., 2018], [Marinakis and Marinaki, 2008a]. Aquí, se

buscará maximizar la ganancia generada por la venta y distribución de productos. Al mismo tiempo, este nivel buscará la minimización de las emisiones de CO_2 generadas en el proceso de producción y distribución. Posteriormente, el nivel inferior, asociado al proceso de producción de cada uno de los productos en las plantas manufactureras, busca minimizar costos de producción y traslado hacia un almacén central encargado de la distribución de los mismos.

Esta tesis busca medir y analizar el impacto que existe por la liberación de emisiones de CO_2 en el ambiente. Además, busca la manera en llegar a un balance estratégico con los objetivos principales de un negocio. Por consiguiente, intenta crear un modelo matemático capaz de identificar y asociar los factores que impactan (intervienen) en los problemas de logística verde, así como la correcta implementación de un algoritmo que pueda aproximar una frontera de posibles soluciones estratégicas para el problema aquí considerado.

1.1 Impacto ambiental y motivación

Las emisiones de CO_2 en la atmósfera han aumentado a un ritmo alarmante durante las últimas décadas. Esto es debido al auge de las fábricas y los vehículos impulsados por combustibles fósiles ([Forster et al., 2007]). El carbono en la atmósfera necesita cientos de años para ser absorbido por la biosfera y los océanos. Debido a las preocupaciones generadas por el cambio climático, las regulaciones gubernamentales y las metodologías para calcular los factores de emisión han aumentado. Se ha realizado un esfuerzo significativo para controlar las emisiones de gases de efecto invernadero en todo el mundo. Por ejemplo, en 42 países asiáticos (ver [Liu et al., 2018]), en 27 países europeos (ver [Zaman and Shamsuddin, 2017]), en Brasil, Rusia, India y China (ver [Aldakhil et al., 2018]), entre otros. En particular, en México se han decretado un gran número de regulaciones durante los últimos 40 años. En 1971 se promulgó la primera ley federal mexicana [Diario Oficial de la Federación, 1971], en donde algunos artículos enfatizan el control de emisiones de carbono. Es importante mencionar, que en [Diario Oficial de la Federación, 1988a] se publicó una de las leyes más importantes en términos de reducción de emisiones. Esta ley establece el control y prevención de la contaminación en la atmósfera. Además, en [Diario Oficial de la Federación, 1988b] se encuentra publicado el reglamento que define los procedimientos técnicos que deben obedecer las fuentes contaminantes. Además, se crearon licencias y certificados para gestionar y controlar las actividades industriales.

Actualmente, existen leyes que establecen las emisiones máximas permitidas para la industria y los vehículos. Por ejemplo, la ley [Diario Oficial de la Federación, 1993] en el Diario Oficial de la Federación establece el nivel máximo permitido de contaminación de acuerdo con su ubicación en el país. La ley [Diario Oficial de la Federación, 1999a] establece el nivel máximo de emisiones permitido para los vehículos de motor que utilizan gasolina como combustible. Las emisiones máximas se establecen en función del año y modelo del vehículo. Además, la ley [Diario Oficial de la Federación, 1999b] establece el nivel máximo de emisiones de hidrocarburos no quemados, monóxido de carbono, óxidos de nitrógeno producidos por los vehículos, donde las emisiones máximas dependen del tipo y peso del vehículo.

Además, se han implementado programas ambientales para reducir CO_2 . Uno de los más populares se llama *Hoy No Circula* en México en el cual se restringe la circulación de vehículos una vez por semana dependiendo del último número de placa. Estas leyes, regulaciones y programas han controlado las emisiones y la contaminación en México. Además, el número de leyes ha aumentado a lo largo de los años para mejorar el medio ambiente y la calidad de vida.

La creciente preocupación por la contaminación a nivel mundial ha llevado no solamente a los gobiernos a crear leyes que restrinjan la producción de dióxido de carbono. Además, ha logrado captar la atención de la industria privada por buscar alternativas de mejora e investigación en sus procesos para generar el menor impacto ambiental posible.

Estos hechos motivan este estudio, en donde se analiza un problema que considera las emisiones de CO_2 producidas por los vehículos usados para rutear y por las instalaciones de fabricación. El modelo estudiado, además de buscar la minimización de emisiones, se interesa en la maximización de la ganancia asociada con el proceso de distribución, lo que genera un problema bi-objetivo. Dado que el proceso de fabricación también se considera en el problema, las emisiones y los costos relacionados son tomados en cuenta.

1.2 Problemática y contribución

El problema aquí estudiado es el siguiente: considérese una situación en la que dos tipos de empresas interactúan entre sí de forma jerárquica dentro de una cadena de suministro. Una empresa adquiere y distribuye diferentes tipos de productos básicos sobre un subconjunto seleccionado de clientes; mientras que la otra empresa fabrica los productos básicos demandados por clientes seleccionados. En este problema, se asume que la empresa que distribuye los productos básicos diseña rutas para satisfacer al subconjunto seleccionado de clientes, con el objetivo de maximizar la ganancia. Además, debido a las regulaciones impuestas por la industria o el gobierno, se debe tener en cuenta la contaminación mínima. Hay disponible una flota heterogénea de vehículos para entregar productos a los clientes. De acuerdo con el supuesto anterior, cada tipo de vehículo tiene diferentes tasas de emisiones de CO_2 . Por tanto, la minimización de las emisiones de CO_2 se considera como otro objetivo.

Una vez que se ha seleccionado el subconjunto de clientes, la empresa distribuidora adquiere los productos necesarios para satisfacer la demanda acumulada de esos clientes específicos. El fabricante dispone de diferentes instalaciones capacitadas para producir la demanda solicitada. Esta empresa debe elegir la cantidad de productos básicos que se fabricarán en cada instalación. Luego, los productos se enviarán desde las instalaciones a un solo depósito. Para lograr esto último, se dispone de una flota de vehículos homogénea. El objetivo del fabricante es minimizar los costos de producción y envío. Además, en cada instalación, se impone una tasa de contaminación asociada con cada producto manufacturado y una tasa máxima de contaminación.

El problema aquí considerado se ilustra en la Fig. 1.1. En donde se puede observar el territorio de planeación para nuestro problema. Los clientes potenciales son identificados con un círculo verde, las fábricas productoras con un triángulo naranja, y finalmente el almacén central, el cual es identificado con un cuadro azul. Para la solución de la problemática bajo estudio, primero se procede a la selección de un subconjunto de clientes por parte de la empresa distribuidora, mostrado en la Fig. 1.1.a. (donde son marcados con una cruz blanca). Luego, se diseñan las mejores rutas posibles (ver Fig. 1.1.b), lo cual, genera una demanda acumulada en el depósito. Una vez que se han tomado estas decisiones, la empresa de fabricación establece el plan de producción en las instalaciones y las decisiones de envío (consulte la Fig. 1.1.c) para las cuales decidirá cuales plantas manufactureras desea utilizar y la transportación de cada una de ellas al almacén principal. Finalmente, el proceso de decisión completo se muestra en la Fig. 1.1.d.

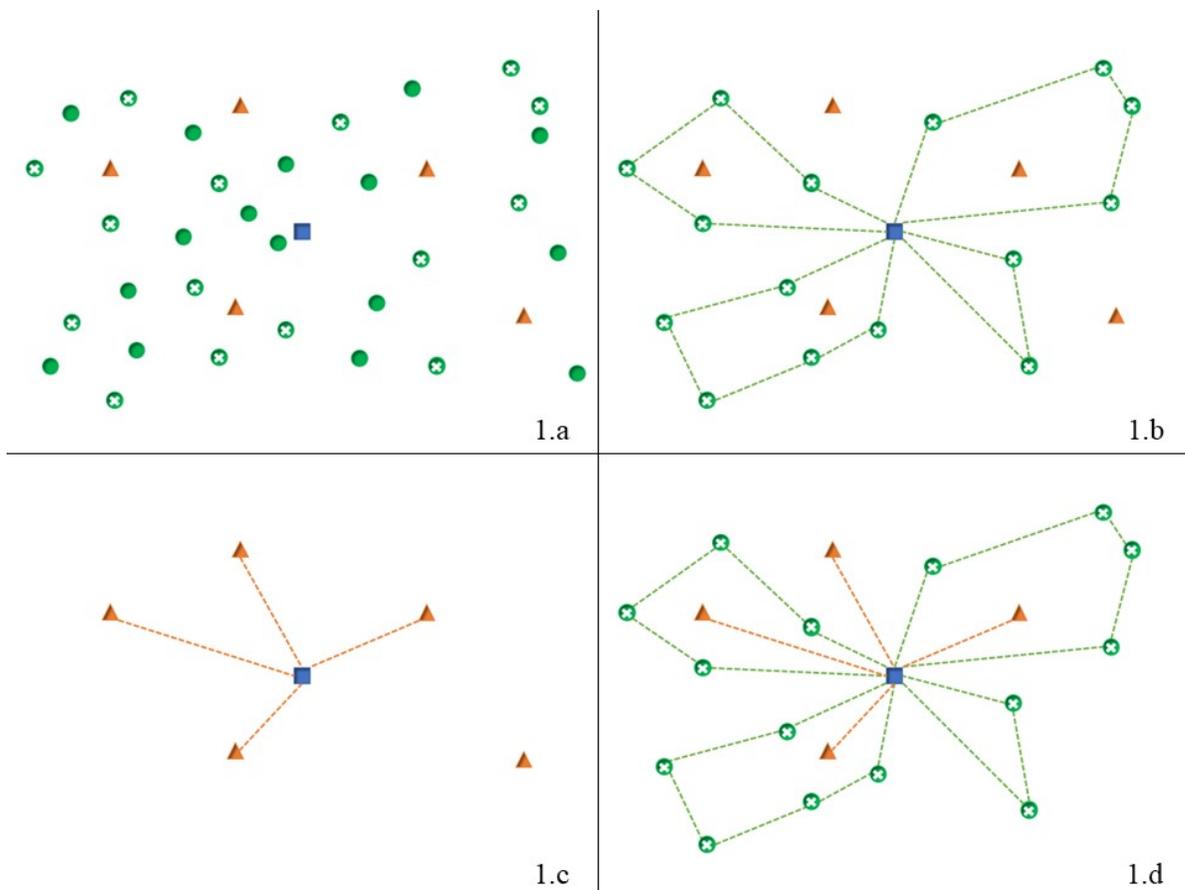


Figura 1.1: Ilustración del problema

Debido a la jerarquía existente entre las empresas, y la forma en que están interrelacionadas, el problema se modela como un modelo de programación bi-objetivo binivel, en el que el nivel superior tiene dos objetivos y aparece un único objetivo en el nivel inferior. La empresa distribuidora será el líder y la empresa de fabricación será el seguidor. Al líder le preocupa la maximización de la ganancia obtenida por el proceso de distribución y la minimización de

las emisiones de CO_2 , simultáneamente. El seguidor tiene como objetivo minimizar sus costos de fabricación y envío. Debido a las características del problema del seguidor, el modelo binivel no se puede reducir a uno de un solo nivel. Debido a esto, no es posible utilizar las reformulaciones clásicas utilizando condiciones de optimalidad. Por lo tanto, se propone un algoritmo de búsqueda tabú para resolver el problema bi-objetivo binivel, es decir, aproximar el frente de Pareto. El algoritmo propuesto considera las soluciones del líder con un enfoque anidado, es decir, resuelve de manera óptima el problema del seguidor para la decisión de cada líder. Los resultados obtenidos muestran fronteras de Pareto muy bien definidas que cubren una amplia gama del espacio de funciones objetivo.

Además, es importante mencionar nuestras principales contribuciones, donde en la literatura existen problemas de logística verde que han sido modelados como problemas bi-objetivo; sin embargo, muy pocos de ellos han sido modelados como problemas binivel. Los problemas existentes binivel enfocados a la logística verde no se enfrentan a la problemática de los esquemas de producción-distribución, donde se involucra la decisión de selección de clientes, transportación y la manufactura de los productos.

Por lo tanto, las principales contribuciones de nuestro problema son las siguientes:

- Un modelo de programación binivel bi-objetivo de logística verde realista y relevante.
- La consideración de un problema jerarquizado de producción-distribución amigable con el medio ambiente.
- Un algoritmo anidado de búsqueda tabú para aproximar las fronteras de Pareto.
- El análisis de los resultados obtenidos para identificar interesantes decisiones administrativas.

Por último, es importante resaltar que se propone un problema nuevo, el cual es la primera vez que se presenta con las suposiciones hechas en esta investigación, por lo cual los resultados obtenidos se consideran bastante buenos y el enfoque propuesto muy razonable en su relación de esfuerzo computacional contra la calidad de resultados obtenidos.

1.3 Objetivos

El principal objetivo de esta tesis es estudiar un problema complejo de logística que involucra el enfoque verde mediante un enfoque jerárquico. Además, proponer el modelo de programación binivel para formular esta situación.

Así mismo, se propone un algoritmo metaheurístico como metodología de solución a dicho problema. Se propone un algoritmo eficiente y funcional en términos del tiempo computacional requerido y de la calidad de las soluciones arrojadas. Para resolver este problema bi-objetivo binivel, lo que se busca es encontrar buenas soluciones en una frontera de Pareto

con un bajo costo computacional, utilizando una Búsqueda Tabú.

En la parte experimental, se busca analizar diversos casos y estudiar los resultados obtenidos de donde se encontraron soluciones para instancias de tamaño pequeños y medianos. Para los resultados, se busca aproximar una frontera de soluciones eficientes que contenga la mayor cantidad posible de soluciones no dominadas. Al mismo tiempo se busca aproximar la frontera más amplia posible de soluciones eficientes. Esto último, provee un mejor panorama para el tomador de decisiones o el consejo administrativo de una empresa, lo cual refleja su importancia y facilita su aplicación en la vida real.

1.4 Metodología

Para la realización de esta tesis se siguieron los siguientes pasos como parte de la metodología de investigación implementada.

1. Definición y acotación del problema bajo estudio.
2. Revisión de la literatura de programación binivel.
3. Revisión de la literatura sobre problemas de logística verde (enfocados en el medio ambiente).
4. Realización de una investigación de los antecedentes e importancia de los problemas de logística verde.
5. Revisión de la literatura sobre problemas de logística verde relacionados.
6. Análisis de las interrelaciones del problema.
7. Desarrollo de un modelo binivel bi-objetivo capaz de mostrar el impacto de cada variable de decisión.
8. Reformulación del modelo binivel en uno de un solo nivel.
9. Pruebas preliminares para medir la eficacia de la reformulación.
10. Presentación de los avances y resultados en EURO-ALIO, Bolonia, Italia, Junio 2018.
11. Seleccionar un algoritmo metaheurístico para aproximar el frente de Pareto del problema.
12. Revisión de literatura de algoritmos de búsqueda tabú.
13. Búsqueda y ajuste de instancias para trabajar el problema.
14. Desarrollo de un algoritmo metaheurístico implementando la búsqueda tabú como método de solución.

15. Implementación computacional del algoritmo.
16. Presentación de los avances y resultados en INFORMS International Conference, Cancun, México, Junio 2019.
17. Ajustes en el algoritmo de búsqueda para ampliar la frontera de soluciones eficientes.
18. Implementación computacional final del algoritmo.
19. Análisis de Resultados.
20. Presentación de los avances y resultados en VIII Congreso de la Sociedad Mexicana de Investigación de Operaciones (CSMIO), CDMX, México, Octubre 2019.
21. Escritura del artículo científico para publicación del problema.
22. Desarrollo de plan de trabajo a futuro y mejoras del problema.
23. Primera ronda de correcciones para publicación de artículo científico.
24. Revisión de la literatura de algoritmos de solución utilizando la heurística de Reencadenamiento de Trayectorias.
25. Propuesta de ajustes del modelo utilizando el algoritmo de Búsqueda Tabú, así como la incorporación de Reencadenamiento de Trayectorias.
26. Segunda ronda de correcciones para publicación de artículo científico.
27. Presentación de avances y resultados en Marzo 2021 en el Seminario del PCOM, UANL.
28. Revisión de la literatura de problemas de emisiones de CO_2 considerando carga y velocidad.
29. Planteamiento de modelo matemático considerando los factores de carga y velocidad.
30. Escritura de tesis.
31. Publicación del artículo en revista Annals of Operations Research (ANOR), titulado "A tabu search algorithm to solve a green logistics bi-objective bi-level problem".

1.5 Estructura de la tesis

La presente tesis está organizada de la siguiente manera:

El Capítulo 1 presenta una introducción del problema estudiado en esta tesis, así como la motivación para conducir la presente investigación. Se presenta una descripción detallada del problema. Y por último, se plantean los principales objetivos de la tesis.

El Capítulo 2 presenta una extensa revisión de la literatura para estudiar y analizar diversos aspectos relacionados a los problemas de logística verde que involucran rutas y múltiples objetivos.

El Capítulo 3 presenta a detalle el problema propuesto, así como una descripción formal del modelo matemático bi-objetivo binivel. Además, se presentan las características del modelo matemático planteado.

El Capítulo 4 describe el algoritmo anidado de búsqueda tabú bi-objetivo propuesto, el cual incluye una extensa explicación referente a la motivación y desarrollo del algoritmo propuesto. Además, se presenta el pseudocódigo seguido por ejemplos ilustrativos del funcionamiento del mismo.

El Capítulo 5 muestra la experimentación computacional. En este capítulo se define la creación de las instancias adaptadas, el ambiente computacional, así como los resultados obtenidos en dicha experimentación. Finalmente presenta el análisis de los resultados obtenidos, así como la discusión de la efectividad del algoritmo para aproximar la frontera eficiente.

El Capítulo 6 posee las conclusiones del presente trabajo y enlista algunas propuestas para futuras investigaciones. Principalmente, dichas propuestas de variantes para el presente modelo son extensiones interesantes que habrá que analizarse de manera independiente. Después de analizar su estructura se pueden definir nuevas líneas de investigación futura.

Capítulo 2

Revisión de Literatura

En este capítulo se presentan los antecedentes de programación binivel, conceptos relevantes y algunas aplicaciones interesantes. Se introduce un preámbulo sobre los problemas existentes en cadena de suministro y su importancia. Así como, se presenta una extensa revisión de literatura de los problemas encontrados en la literatura referentes a la logística verde y el impacto ambiental. Se hace énfasis en los modelos que han sido abordados mediante programación binivel. Además, se presenta una tabla comparativa de los métodos de solución encontrados con los cuales se han abordado dichos problemas.

2.1 Programación Binivel

Gran cantidad de problemas de la vida real involucran una relación jerárquica entre dos o más tomadores de decisiones. Estos problemas de tomadores de decisiones pueden verse traducidos como problemas de dos niveles (binivel), los cuales son conocidos como los problemas de Stackelberg. Su primer formulación de programación binivel apareció en 1934, propuesta por Heinrich Von Stackelberg [Stackelberg, 2011] enfocada en solucionar problemas en el área de economía. En dichos problemas económicos de Stackelberg, se demostró que era posible describir problemáticas complejas y reales de mercado, en un solo problema. Esto último, se lograba al definir un modelo jerárquico en donde dos tomadores de decisiones diferentes causan un impacto en el objetivo de un problema.

Este enfoque binivel consiste en definir dos tomadores de decisiones, donde, cada uno de ellos posee su propio objetivo e independencia directa con el otro tomador de decisiones. Sin embargo, las decisiones del líder dependen del seguidor, y debe tener en consideración que el seguidor es limitado por las decisiones del líder. Con esta idea, se desarrolló el concepto de tener un problema de optimización principal con sus restricciones asociadas. Adicional a esto, como una restricción del problema principal (líder) se establece un segundo problema de optimización (seguidor) con sus propias restricciones asociadas. Stackelberg definió este modelo jerárquico donde para cada movimiento particular del líder, el seguidor debe resolver su propio problema parametrizado en la decisión del líder. Esto último, crea una reacción particular del nivel inferior para cada posible decisión del líder [Réveillac, 2015].

Un problema de programación binivel representa la coordinación de dos cadenas de suministro jerárquicas en términos de un modelo de dos niveles, donde el problema principal es referido como el nivel superior o del líder y el problema anidado es referido como el nivel inferior o del seguidor.

Los problemas de programación binivel lineales, como se menciona en [Bard, 1991a] son clasificados como problemas no-convexos y además, clasificados como NP-duros, cuya característica principal significa que no pueden ser resueltos descomponiendo el problema de forma polinomial. Adicional a esto, son aquellos resueltos en un tiempo polinomial no determinista. Esta complejidad algorítmica y computacional de solución ha representado una gran área de interés en la literatura, abarcando problemas de decisión, búsqueda u optimización.

2.2 Problemas de logística o cadena de suministro bajo un enfoque binivel

Los problemas de logística y transporte son planteamientos enfocados en problemas de la vida real que buscan solucionar áreas de oportunidad en la toma de decisiones en diversos negocios (la industria). Buscan encontrar óptimas decisiones estratégicas para los procesos de producción y distribución. En el camino de solución se busca organizar los diversos procesos que una compañía tiene para lograr operar de la mejor manera buscando eficientizar sus recursos.

Estos problemas de logística o cadena de suministro pueden ser resueltos con diversos enfoques en el área de la investigación de operaciones. En específico, se van a abordar los problemas de programación binivel. La programación binivel ofrece la posibilidad de resolver problemas con una jerarquía predefinida gracias a su enfoque y modelación matemática.

El enfoque de modelación de problemas binivel ha demostrado tener gran impacto y ha sido capaz de encontrar soluciones a problemas complejos que eran imposibles de resolver en problemas de un solo nivel, debido a la jerarquía e independencia entre ambos niveles. Los primeros casos prácticos y teóricos que surgieron pueden ser encontrados en [Bard, 1983],[Bard, 1991b]. Estos dieron inicio a lo que hoy abarca una gran área de oportunidad en el planteamiento y solución de problemas complejos.

Por último, referenciamos diversos trabajos que abordan problemas de logística relacionados con el aquí estudiado. En particular, nos enfocamos en trabajos en los cuales, debido a la naturaleza del problema mismo, fueron analizados con un enfoque de programación binivel. Algunos ejemplos claros son los siguientes: en la investigación presentada en [Marinakis and Marinaki, 2008a], se aborda un problema real de localización y ruteo (LRP, por sus siglas en inglés) para una de las mayores compañías distribuidoras de madera en Grecia, donde para resolver su problemática, proponen una formulación del LRP basados en programación binivel; para este problema, el nivel superior es aquel con las decisiones del nivel estratégico que busca encontrar la ubicación óptima de las instalaciones, mientras el seguidor

está encargado por el nivel operativo, el cual busca la ruta óptima para los vehículos; para este problema proponen un algoritmo binivel que proporciona buenas soluciones y beneficios económicos para la empresa. En [Huang and Liu, 2004], se presenta un modelo de programación binivel el cual busca minimizar el costo de una red de distribución global y al mismo tiempo equilibrar la carga de trabajo de cada centro de distribución para la entrega de productos a sus clientes; debido a la complejidad de su problemática, es modelado como un modelo binivel, para el cual presentan un enfoque basado en algoritmos genéticos, y donde dan a los tomadores de decisiones una comparación entre el objetivo de minimización de costos y los requerimientos de balanceo. En [Zhang and Lu, 2007], abordan un problema binivel que busca desarrollar un modelo de toma de decisiones para la planificación logística; proponen un modelo binivel difuso y desarrollan un número difuso basado en el k -ésimo mejor enfoque para encontrar una solución óptima para el problema de decisión. El estudio presentado por [Sun et al., 2008] presenta un modelo de programación binivel para buscar la ubicación óptima de centros de distribución logística, donde el nivel superior busca determinar la ubicación óptima minimizando el costo de los planificadores (planeadores), y el nivel inferior proporciona una distribución de demanda equilibrada minimizando el costo de los clientes. Basado en la forma especial de sus restricciones, proponen un heurístico para solucionar el problema. Finalmente, otro claro ejemplo es el propuesto en [Safaei et al., 2018], donde abordan un problema de suministro y distribución en una red de ayuda humanitaria donde se presenta incertidumbre en los parámetros de demanda y suministro; para el cual, se busca optimizar el flujo de productos de ayuda en una cadena, así como seleccionar los mejores suministros y asignar órdenes; para dicho problema el enfoque binivel ayuda a seleccionar y asignar esa ayuda; en dicho problema el líder busca minimizar los costos operacionales totales, así como la cantidad esperada de demanda insatisfecha; el nivel inferior busca minimizar el riesgo total de suministrar los productos de ayuda.

2.3 Logística verde

Los estudios que buscan solucionar problemáticas ambientales han tenido un crecimiento y foco de interés muy grande en los últimos años, por lo cual han aparecido diferentes artículos para abordar problemas ambientales y sostenibles. En [Dekker et al., 2012], [Lin et al., 2014] y [Faulin et al., 2019], se toman diferentes decisiones en cadenas de suministro donde son considerados aspectos ambientales como transporte, rutas, producción, inventario, instalaciones (almacenes, puertos y terminales), diseño de la cadena de suministro, la recuperación de productos (cadenas de suministro de circuito cerrado) y el control operativo de las cadenas de suministro son mencionados.

En [Sbihi and Eglese, 2010], los autores involucran consideraciones ambientales y sociales más amplias en problemas dinámicos de lot-sizing, el modelo de costos de instalación conjunta y separada, la gestión de desechos, la recolección de desechos domésticos y el ruteo de vehículos. Además, en [Lai and Wong, 2012], se realizan importantes aportes en la gestión de cadenas logísticas, donde consideran el medio ambiente examinando los efectos de la presión regulatoria ambiental. En el caso de estudio de [Ubeda et al., 2011] problema relacionado

con la distribución de alimentos en España. Los autores muestran la forma en que se podría reducir el impacto ambiental en la práctica. En [Linton et al., 2007], los autores se centran en la interacción entre la sostenibilidad y las cadenas de suministro. Para comprender mejor las cadenas de suministro ambientales, [Beamon, 1999] analiza el impacto de los desechos sólidos y peligrosos, el uso de recursos naturales, la contaminación del agua y el aire, la presión pública, la legislación ambiental y las normas de gestión ambiental.

Una de las principales características de un problema de logística verde es que deben ser tomadas en cuenta las emisiones de CO_2 . Por ejemplo, el pollution-routing problem (PRP) introducido por [Bektaş and Laporte, 2011], que consiste en una extensión del problema de ruteo de vehículos, donde se consideran las emisiones de efecto invernadero. Extensiones del PRP se presentan en [Demir et al., 2012], [Franceschetti et al., 2013] y [Demir et al., 2014]. Además, en [Li et al., 2008], se analiza la relación entre las emisiones de CO_2 y la relación costo-ingreso de operación en la ubicación de los centros de distribución. Además, los autores presentaron un caso de estudio sobre el petróleo crudo, del cual se concluye que si el precio del crudo aumenta, las emisiones de carbono disminuirán. Otro hallazgo interesante es que si son abiertos más centros de distribución, las emisiones de carbono podrían disminuir. En [Diabat and Simchi-Levi, 2009] el impacto ambiental de las emisiones de CO_2 son consideradas como la novedad de un modelo de gestión de cadena de suministro verde, que integra la gestión y el impacto ambiental de la cadena de suministro. Un problema de ruteo de vehículos verde con efectos ambientales por utilizar vehículos de combustible alternativo es estudiado en [Erdoğan and Miller-Hooks, 2012]. Las investigaciones relacionadas que consideran decisiones de planificación aparecen en [Xiao and Konak, 2015] y [Qian and Eglese, 2016]. Además, en [Zhang et al., 2018] son incluidas las estaciones de combustible alternativo en el modelo de logística verde para reducir las emisiones de gases de efecto invernadero. Es posible encontrar soluciones prácticas y de buena calidad para problemas de logística verde en [Leggieri and Haouari, 2017], [Montoya et al., 2016], [Çağrı Koç and Karaoglan, 2016] y [Franceschetti et al., 2017]. Además, en [Çağrı Koç et al., 2014] es presentado un importante caso de estudio que muestra la ventaja de utilizar flotas heterogéneas dentro de este contexto verde, y en [Figliozzi et al., 2020] se presenta un interesante análisis de las emisiones causadas por diferentes vehículos.

Si se consideran al mismo tiempo otros objetivos, como los aspectos económicos o sociales, aparece un problema multiobjetivo. Diversos artículos abordan este tipo de problemas de logística verde. Por ejemplo, un problema de ubicación de una instalación sin capacidades se estudia en [Harris et al., 2009]. Los autores optimizan el costo, impacto ambiental y la demanda no cubierta, simultáneamente. Para resolver dicho problema, fue implementado un algoritmo evolutivo multiobjetivo. En [Harris et al., 2014], se presenta un problema de ubicación y asignación de instalaciones con capacidades, donde los objetivos están relacionados con el costo y el impacto ambiental. Es aplicado un algoritmo evolutivo simple para la optimización multiobjetivo para obtener un conjunto de soluciones no dominadas. Además, los costos de envío y las emisiones de gases se minimizan en [Pérez-Bernabeu et al., 2015], mostrando la importancia de la cooperación horizontal en una red de transporte por carretera. Este último concepto es explicado a profundidad en [Serrano-Hernández et al., 2017], en el que se señalan las amplias posibilidades para mejorar una cadena de suministro verde. Otros

estudios que consideran múltiples objetivos se pueden encontrar en [Sawik et al., 2017b] y [Sawik et al., 2017a], en los que se resuelve un caso de estudio real en España y algunas percepciones gerenciales sobre el tamaño de la flota, la importancia de tener una flota heterogénea, y el uso de diferentes tipos de camiones se concluye a partir de los resultados obtenidos.

En [Kannan et al., 2013] es estudiado un problema de selección de proveedores y asignación de pedidos, en el cual se considera el desempeño ambiental de los proveedores. Para resolver el problema, se integra un proceso de jerarquía analítica difusa (fuzzy), técnicas difusas para ordenar las preferencias por similitud con una solución ideal y un modelo de programación lineal multiobjetivo difuso. El estudio se centra en una empresa de fabricación de automóviles iraní que tiene como objetivo establecer un enfoque sistemático para cumplir con la selección de proveedores ecológicos. Un problema de selección de socios se introduce en [Yeh and Chuang, 2011], donde son considerados cuatro objetivos, (i) la minimización del costo total considerando los costos de producción y transporte, (ii) la minimización del tiempo de producción y transporte, (iii) la maximización de la calidad promedio del producto, y (iv) la maximización de un puntaje de evaluación ecológica. Se proponen dos algoritmos genéticos multiobjetivo. Además, para analizar las correlaciones entre los objetivos, se prueban cuatro esquemas: tres problemas biobjetivos (diferentes combinaciones de objetivos) y la consideración de todos los objetivos.

Por otro lado, existe una escasa investigación en la cual los aspectos ambientales sean considerados y modelados como problemas binivel. Bajo este esquema, se necesita una jerarquía entre los tomadores de decisiones. Después de una búsqueda intensiva, solo se encontraron algunos artículos. Por ejemplo, [Mathew and Sharma, 2006] aborda el problema de una cadena de suministro ecológica, en la que se estudia un problema de diseño de red. El líder determina la capacidad de expansión del enlace (the link capacity expansion) sujeta al comportamiento de viaje del usuario para minimizar los costos de viaje del sistema; y el seguidor determina los flujos de enlace sujetos a las condiciones de equilibrio del usuario. De manera similar, [Wen and Eglese, 2016] maneja un problema de precios de peaje para minimizar las emisiones de CO_2 . El líder impone peaje en las carreteras de la red, mientras los usuarios encuentran un equilibrio que minimiza los costos de viaje. Se pueden encontrar investigaciones previas sobre la reducción de emisiones en [Wang et al., 2011], en donde el gobierno (líder) elige el precio óptimo para las emisiones considerando la respuesta de las empresas (seguidoras) a ese precio; naturalmente, el seguidor tiene como objetivo maximizar su beneficio. En [Wang et al., 2014] se estudia una política de precios para la congestión del tráfico urbano, en donde el costo de las emisiones de carbono se considera parte del costo del viaje. El comportamiento de toma de decisiones de la gestión del tráfico está representado por el líder, maximizando el excedente del cliente; mientras que el seguidor describe el comportamiento de elección del usuario, minimizando los costos de viaje. Además, en [Hızır, 2006] se considera un modelo que mide las emisiones de gases en una red de tráfico para el transporte urbano. En la cual, el líder representa a los gerentes de transporte con el objetivo de hacer que los sistemas de transporte sean sostenibles; mientras que el seguidor representa las decisiones de los usuarios de la red minimizando sus costos de viaje.

En resumen, los problemas de logística verde se han modelado como programas bi-objetivo; pero pocos de ellos se han estudiado como problemas binivel. Sin embargo, los problemas de logística verde no se han enfrentado con un esquema de producción y distribución, donde la selección del cliente, el transporte y las decisiones de fabricación están involucradas. Por lo tanto, los principales aportes de esta investigación sean los siguientes: (i) un modelo de programación bi-objetivo binivel de logística verde realista y relevante, (ii) la consideración de un problema de producción-distribución jerarquizado y ecológico, (iii) un algoritmo de búsqueda tabú bi-objetivo anidado para aproximar los frentes de Pareto, y (iv) el análisis de los resultados obtenidos dando lugar a interesantes conocimientos de gestión. Es importante enfatizar que el problema aquí estudiado se presenta por primera vez y las suposiciones hechas a lo largo de la investigación son buenas y bastante razonables.

Capítulo 3

Modelo Matemático

En esta sección, se plantea claramente el problema bajo estudio y se describe a detalle el problema de programación binivel. Después, se define la terminología utilizada y se presenta formalmente el modelo matemático propuesto .

3.1 Descripción del problema

El problema aquí estudiado considera una situación en la que dos tipos de empresas interactúan entre sí de forma jerárquica dentro de una cadena de suministro. Una empresa adquiere y distribuye diferentes tipos de productos básicos sobre un subconjunto seleccionado de clientes; mientras que la otra empresa fabrica los productos básicos demandados por un grupo de clientes seleccionados. En este problema, se asume que la empresa que distribuye los productos básicos diseña rutas para satisfacer al subconjunto seleccionado de clientes con el objetivo de maximizar la ganancia. Además, debido a las regulaciones ambientales, se debe tener en cuenta la contaminación máxima. Hay disponible una flota heterogénea de vehículos para entregar los productos a los clientes. De acuerdo con el supuesto anterior, cada tipo de vehículo tiene diferentes tasas de emisiones de CO_2 . Por lo tanto, la minimización de las emisiones de CO_2 se considera como otro objetivo. Finalmente, la empresa manufacturera encargada de la fabricación de productos busca minimizar sus costos asociados a la producción y envío de productos. Ver Fig. 3.1.

3.2 Modelo matemático

En esta sección se describe la formulación matemática. Los conjuntos, parámetros, variables de decisión y los supuestos considerados se presentan a continuación. Sea I , L , M y N el conjunto de clientes, tipos de vehículos, instalaciones y productos respectivamente. Además, sea $V(l)$ el conjunto de vehículos disponibles de cada tipo $l \in L$, y $k(l)$ denota el k -ésimo vehículo del tipo $l \in L$ donde $k(l) \in V(l)$.

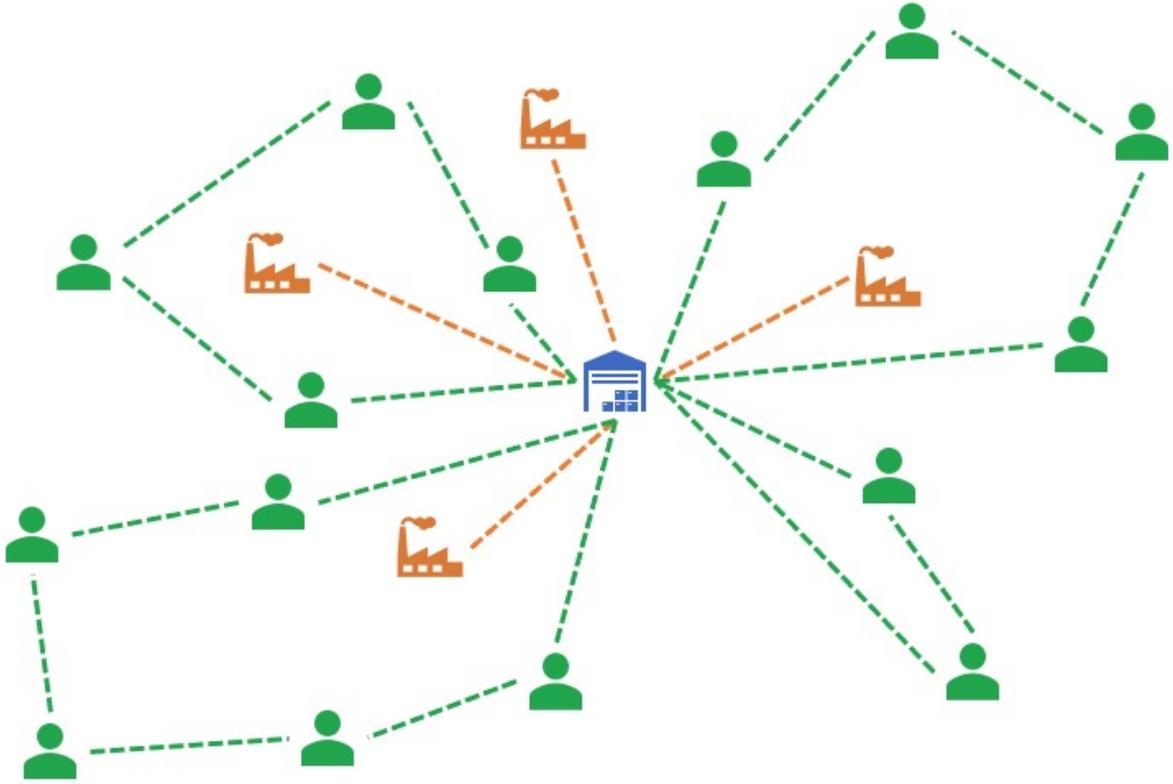


Figura 3.1: Ilustración del problema

De una forma similar a [Anderluh et al., 2019], [Eskandarpour et al., 2019] y [Li et al., 2019], las emisiones de CO_2 por unidad de distancia del tipo de vehículo $l \in L$ son consideradas como e_l . Sea ϵ_{mn} las emisiones de CO_2 generadas por la fabricación de cada producto $n \in N$ en la instalación $m \in M$. Además, considere ϵ como las emisiones de CO_2 por unidad de distancia causadas por el envío de productos desde una instalación al depósito. Sea g_n la ganancia asociada con cada producto $n \in N$. La distancia entre un par de clientes $i, j \in I$ se denota por d_{ij} , y δ_{in} representa la demanda del producto $n \in N$ ordenado por el cliente $i \in I$. Además, supongamos que w^{min} representa la ganancia mínima impuesta por la empresa distribuidora.

Respecto a los costos involucrados en la función objetivo del líder, sea r_l el costo de alquiler del tipo de vehículo $l \in L$; c_l el factor de corrección que convierte la distancia recorrida en un costo para cada tipo de vehículo $l \in L$; α_{mn} sea el costo de adquisición de cada producto $n \in N$ de la instalación $m \in M$ (incluye los costos de envío de cada instalación al depósito).

Además, u_l^{max} es la capacidad disponible asociada con el tipo de vehículo $l \in L$, y v^{max} representa la duración máxima de la ruta. Sea t_{ij} el tiempo requerido para llegar del cliente $i \in I$ al cliente $j \in I$, y se define como la suma de la distancia corregida y el tiempo de servicio

en el cliente $j \in I$; se calcula como $t_{ij} = d_{ij}\varphi + h_j$, donde φ es un factor que convierte la distancia en tiempo. Además, considere que el tiempo de servicio en el depósito puede ser omitido, es decir $h_0 = 0$, donde el depósito se indica mediante el subíndice 0. Sea γ^1 una constante positiva suficientemente grande que limitara el número de arcos en una ruta.

En el problema del seguidor, f_n representa la materia prima requerida para producir el producto $n \in N$, β_{mn} representa los costos de fabricación del producto $n \in N$ en la instalación $m \in M$, y ρ_m indica los costos de envío desde la instalación $m \in M$ al depósito. Además, s_m^{max} es la tasa de emisión máxima de CO_2 permitida en la instalación $m \in M$, y π_m^{max} representa la disponibilidad máxima que puede ser fabricada en la instalación $m \in M$. Finalmente, sea γ^2 una constante positiva suficientemente grande que limite el total de productos manufacturados en las instalaciones. Todos los conjuntos y parámetros se resumen en la Tabla 3.1.

Las variables de decisión binarias del líder son detalladas a continuación:

$$x_i = \begin{cases} 1, & \text{si el cliente } i \in I \text{ es seleccionado} \\ 0, & \text{si no} \end{cases}$$

$$y_{ij}^{k(l)} = \begin{cases} 1, & \text{si el arco } (i, j) \text{ esta en la ruta del vehículo } k \text{ del tipo } l \in L \\ 0, & \text{si no} \end{cases}$$

$$z^{k(l)} = \begin{cases} 1, & \text{si el } k\text{-ésimo vehículo del tipo } l \text{ es usado} \\ 0, & \text{si no} \end{cases}$$

Las variables de decisión del seguidor son denotadas por p_{mn} , que indica la cantidad de producto $n \in N$ fabricado en la instalación $m \in M$, y q_m , que denota si la instalación $m \in M$ se utiliza para fabricar productos o no. Todas las variables de decisión aquí consideradas se incluyen en la tabla 3.2.

Para simplificar la notación en el modelo matemático, definamos el almacén como el nodo 0 y el conjunto que contiene el almacen y los clientes como $I^* = I \cup \{0\}$. Por lo tanto, el modelo matemático de programación bi-objetivo binivel propuesto es el siguiente:

$$\min_{y,x,z} \sum_{i \in I^*} \sum_{j \in I^*} \sum_{l \in L} \sum_{k(l) \in V(l)} (e_l d_{ij}) y_{ij}^{k(l)} + \sum_{m \in M} \sum_{n \in N} \epsilon_{mn} p_{mn} + \sum_{m \in M} \varepsilon d_{m0} q_m \quad (3.1)$$

$$\max_{y,x,z} \sum_{i \in I} \left(\sum_{n \in N} g_n \delta_{in} \right) x_i - \sum_{l \in L} r_l \left(\sum_{k(l) \in V(l)} z^{k(l)} \right) - \sum_{m \in M} \sum_{n \in N} \alpha_{mn} p_{mn} \\ - \sum_{i \in I^*} \sum_{j \in I^*} \sum_{l \in L} \sum_{k(l) \in V(l)} (c_l d_{ij}) y_{ij}^{k(l)} \quad (3.2)$$

Tabla 3.1: Lista de conjuntos y parámetros

Notación	Descripción
Sets	
I	Clientes
L	Tipos de vehiculo
M	Instalaciones
N	Productos
$V(l)$	Vehiculos disponibles del tipo $l \in L$
Parametros	
e_l	CO_2 emisiones del tipo de vehículo $l \in L$ (por unidad de distancia, $kg\ CO_2/km$)
ϵ_{mn}	CO_2 emisiones por fabricar cada producto $n \in N$ en la instalación $m \in M$
ε	CO_2 Emisiones causadas por el envío de productos desde una instalación al depósito (por unidad de distancia)
g_n	Ganancia asociada con cada producto $n \in N$
d_{ij}	Distancia entre el par de clientes $i, j \in I$ (km)
δ_{in}	Demanda del producto $n \in N$ por el cliente $i \in I$
w^{min}	Ganancia mínima impuesta por la compañía distribuidora.
u_l^{max}	Capacidad máxima asociada con el tipo de vehículo $l \in L$
v^{max}	Duración máxima de la ruta
r_l	Costo de renta del vehículo tipo $l \in L$
c_l	Factor de corrección que convierte la distancia viajada en costo para cada tipo de vehículo $l \in L$
α_{mn}	Costo de adquisición para cada producto $n \in N$ de la instalación $m \in M$ (incluye los costos de envío de cada instalación al almacén)
t_{ij}	Tiempo requerido para llegar del cliente $i \in I$ al cliente $j \in I$
φ	Factor que convierte distancia en tiempo
γ^1	Constante positiva suficientemente grande
γ^2	Constante positiva suficientemente grande
f_n	Materia prima requerida para producir el producto $n \in N$
β_{mn}	Costo de manufactura para el producto $n \in N$ en la instalación $m \in M$
ρ_m	Costo de envío de la instalación $m \in M$ al almacén
s_m^{max}	Máxima tasa de emisiones de CO_2 permitida en la instalación $m \in M$ ($kg\ CO_2$)
π_m^{max}	Máxima disponibilidad de producción en la instalación $m \in M$

sujeto a:

Tabla 3.2: Lista de variables de decisión

Notación	Descripción
VARIABLES DEL LÍDER	
x_i	Selección binaria del i -ésimo cliente
$y_{ij}^{k(l)}$	Selección binaria del arco (i, j) en la ruta del vehículo k del tipo $l \in L$
$z^{k(l)}$	Selección binaria para el k -ésimo vehículo del tipo l
VARIABLES DEL SEGUIDOR	
p_{mn}	Cantidad de producto $n \in N$ manufacturado en la instalación $m \in M$
q_m	Selección binaria para la instalación $m \in M$

$$\sum_{j \in I} y_{0j}^{k(l)} = z^{k(l)} \quad \forall l \in L, k(l) \in V(l) \quad (3.3)$$

$$\sum_{i \in I} y_{i0}^{k(l)} = z^{k(l)} \quad \forall l \in L, k(l) \in V(l) \quad (3.4)$$

$$\sum_{j \in I} \sum_{l \in L} \sum_{k(l) \in V(l)} y_{ij}^{k(l)} = x_i \quad \forall i \in I \quad (3.5)$$

$$\sum_{j \in I} y_{ij}^{k(l)} = \sum_{j \in I} y_{ji}^{k(l)} \quad \forall i \in I, l \in L, k(l) \in V(l) \quad (3.6)$$

$$\sum_{i \in W} \sum_{j \in W} y_{ij}^{k(l)} \leq |W| - 1 \quad \begin{matrix} W \subseteq I^*, 2 \leq |W| \leq |I| + 1 \\ \forall l \in L, k(l) \in V(l) \end{matrix} \quad (3.7)$$

$$\sum_{i \in I^*} \sum_{\substack{j \in I \\ i \neq j}} t_{ij} y_{ij}^{k(l)} \leq v^{max} \quad \forall l \in L, k(l) \in V(l) \quad (3.8)$$

$$\sum_{i \in I} \sum_{\substack{j \in I^* \\ i \neq j}} \left(\sum_{n \in N} \delta_{in} \right) y_{ij}^{k(l)} \leq u_l^{max} \quad \forall l \in L, k(l) \in V(l) \quad (3.9)$$

$$\sum_{k(l) \in V(l)} z^{k(l)} \leq |V(l)| \quad \forall l \in L \quad (3.10)$$

$$\sum_{i \in I} \sum_{j \in I} y_{ij}^{k(l)} \leq z^{k(l)} \gamma^1 \quad \forall l \in L, k(l) \in V(l) \quad (3.11)$$

$$\sum_{i \in I} \left(\sum_{n \in N} g_n \delta_{in} \right) x_i - \sum_{l \in L} r_l \left(\sum_{k(l) \in V(l)} z^{k(l)} \right) - \sum_{i \in I^*} \sum_{j \in I^*} \sum_{l \in L} \sum_{k(l) \in V(l)} (c_l d_{ij}) y_{ij}^{k(l)} - \sum_{m \in M} \sum_{n \in N} \alpha_{mn} p_{mn} \geq w^{min} \quad (3.12)$$

$$y_{ij}^{k(l)}, x_i, z_l \in \{0, 1\} \quad \forall i, j \in I^*, l \in L, k(l) \in V(l) \quad (3.13)$$

En el cual para cada decisión del líder y, x, z , las variables del seguidor p y q resuelven

$$\min_{p,q} \sum_{m \in M} \sum_{n \in N} \beta_{mn} p_{mn} + \sum_{m \in M} \rho_m q_m \quad (3.14)$$

sujeto a:

$$\sum_{m \in M} p_{mn} = \sum_{i \in I} \delta_{in} x_i \quad \forall n \in N \quad (3.15)$$

$$\sum_{n \in N} \epsilon_{mn} p_{mn} \leq s_m^{max} \quad \forall m \in M \quad (3.16)$$

$$\sum_{n \in N} f_n p_{mn} \leq \pi_m^{max} \quad \forall m \in M \quad (3.17)$$

$$\sum_{n \in N} p_{mn} \leq q_m \gamma^2 \quad \forall m \in M \quad (3.18)$$

$$p_{mn} \in Z^+ \cup \{0\} \quad \forall m \in M, n \in N \quad (3.19)$$

$$q_m \in \{0, 1\} \quad \forall m \in M \quad (3.20)$$

El modelo definido por las Ec. (3.1)-(3.20) es un problema de programación lineal binivel bi-bjetivo. En la Ec. (3.1) la función objetivo del líder mide las emisiones de CO_2 causadas por la ruta de cada tipo de vehículo y las emisiones de CO_2 generados por el proceso de fabricación y envío desde las instalaciones. En la Ec. (3.2), la segunda función objetivo del líder corresponde a la maximización de beneficios. El primer término representa el ingreso total por cada producto demandado por los clientes, los términos restantes son el costo total de alquiler asociado con los vehículos, los costos totales de transporte de todos los vehículos y los costos de adquisición de los productos desde las instalaciones. Con respecto a las restricciones del líder, la Ec. (3.3) requiere que cada vehículo tenga una única salida del depósito, la Ec. (3.4) asegura que cada vehículo solo llegue una vez al depósito. La restricción (3.5) indica que los clientes seleccionados deben ser visitados solo una vez, la Ec. (3.6) es la restricción de conservación del flujo, y la ecuación (3.7) corresponde a una restricción clásica que evita los subtours. La restricción (3.8) asegura que el tiempo asociado con cada ruta no debe exceder el tiempo máximo establecido, la Ec. (3.9) establece que las mercancías incluidas en cada ruta no exceden la capacidad de cada tipo de vehículo, la Ec. (3.10) indica que hay una disponibilidad máxima de vehículos de cada tipo, la Ec. (3.11) asegura que solo los vehículos que se utilizan tengan una ruta asignada. La restricción (3.12) establece un beneficio mínimo requerido para la empresa. Es necesario observar que esta restricción involucra variables de líder y seguidor, lo que se conoce como restricción de acoplamiento en programación binivel. Por lo tanto, la factibilidad de una solución se logra teniendo un valor factible para las variables del líder, la solución óptima para el seguidor, y verificando que también se satisface la restricción (3.12). Además, la Ec. (3.13) establece las restricciones binarias para las variables de decisión de cada líder y , x y z .

El problema del seguidor está definido por las Ec. (3.14)-(3.20). La Ec. (3.14), presenta la función objetivo del seguidor, que tiene como objetivo minimizar los costos de fabricación y envío. En la Ec. (3.15), el suministro de toda la demanda acumulada está asegurada. La restricción (3.16) garantiza que las emisiones de CO_2 para la fabricación de los productos

demandados no pueden exceder la tasa de emisiones máxima de CO_2 permitida en cada instalación, y la Ec. (3.17) establece que los productos manufacturados no pueden exceder la producción máxima en cada instalación. La Ec. (3.18) asegura que si la instalación $m \in M$ fabrica algunos productos, entonces se tomarán en cuenta los costos de envío asociados. Finalmente, las Ec. (3.19) y (3.20) restringen que las variables del seguidor sean enteras no negativas y binarias, respectivamente.

3.3 Enfoque optimista

Al abordar nuestro problema, en el caso de que existan múltiples respuestas óptimas para el seguidor, para que el problema binivel esté bien definido, se necesita asumir un enfoque optimista o pesimista.

Es importante observar que las variables del seguidor p_{mn} y q_m están presentes en la función objetivo del líder, y la variable del líder x_i está presente en el problema del seguidor. En consecuencia, las decisiones del líder afectarán directamente los costos asociados con las asignaciones del plan de producción trazado por el seguidor. Sin embargo, es posible que puedan ser encontradas múltiples soluciones óptimas en el problema del seguidor para una decisión particular del líder.

Entonces, para tener un método de solución del problema binivel que sea adecuado, es importante considerar el caso en que el problema del seguidor tenga múltiples soluciones óptimas para una decisión determinada del líder. Para conseguir esto, se necesita asumir uno de los enfoques clásicos: el optimista o el pesimista.

En este problema, se decide asumir un enfoque optimista. Es por esto que optamos por la mejor conveniencia para el líder garantizando siempre el óptimo en el problema del seguidor. Esto es, una vez que el líder toma una decisión, el seguidor reacciona al tomar su mejor solución. Entonces, el problema del líder es resuelto nuevamente agregando una nueva restricción, fijando el valor de función objetivo del seguidor, así como su selección de variables.

Al asumir el enfoque optimista, este puede ser visto de dos formas: el seguidor posee una actitud cooperativa con el líder; o el líder selecciona la opción del seguidor más favorecedora con respecto a sus propios objetivos, sabiendo que esto no afectará la función objetivo del seguidor.

En nuestro problema, tomar la decisión que resulte más conveniente para el líder no es algo claro, ya que el líder considera simultáneamente dos objetivos. Y como se mencionó anteriormente, dichos objetivos se encuentran en conflicto. Esto es, de forma más clara, que si el líder decidiera enfocarse en un objetivo, por ejemplo, en la maximización de la ganancia, el tomador de decisiones tomaría la opción que para maximizar ganancia a su vez maximiza el número de viajes y clientes satisfechos. Esto último genera un panorama perjudicial para el otro objetivo, ya que a mayor cantidad de clientes satisfechos, mayor es la cantidad de

vehículos en circulación contaminando con CO_2 , y mayor es la cantidad de contaminación generada por las plantas productoras. Por el contrario, si el tomador de decisiones decidiera enfocarse en el objetivo de reducir al máximo las emisiones de CO_2 optaría por la opción de solución que tuviera el mínimo de clientes satisfechos, pero que garantizará cubrir la cuota mínima de ganancia. Esto último con la finalidad de tener la menor cantidad de camiones en circulación emitiendo CO_2 , tener el mínimo de carga en ellos y producir la menor contaminación posible por parte de las plantas manufactureras. Esto podría parecer una buena opción, sin embargo va en contra de los deseos y aspiraciones de una compañía en la industria privada. Así que, resulta importante generar un amplio panorama de dichas soluciones para proveer el mejor conjunto de posibles soluciones para el líder.

Nosotros, asumimos que se elige la solución del seguidor que implica mayor ganancia para el líder. El fundamento de este supuesto se basa en el hecho de que la restricción de acoplamiento está relacionada con la ganancia. Al asumir este enfoque optimista, se busca la factibilidad binivel. Esto puede verse como un esquema cooperativo. Además, dado que el seguidor no se beneficia al afectar la función objetivo del líder, comúnmente se asume el enfoque optimista, ver [Kalashnikov et al., 2015] y [Sinha et al., 2016].

3.4 Análisis de la reformulación a un modelo de un solo nivel

Los problemas que involucran el ruteo de vehículos se conocen por ser problemas difíciles de resolver, en particular, por las restricciones que evitan la creación de ciclos, las cuales, usualmente se reflejan en un número exponencial de restricciones que se agregan al modelo. En particular, en este problema la integración del proceso de fabricación y las decisiones de la distribución de productos complica aún más el problema. Además de lo anterior, como se ha mencionado en el Capítulo 2, los problemas binivel son complejos de resolver, aún en el caso más simple (caso lineal). Por lo tanto, considerando todas estas características dentro del mismo marco incrementarán el grado de intratabilidad del problema. En base a esto último, decidimos iniciar el análisis del mismo, considerado mediante el estudio de una hipotética simplificación al reformular el problema como uno de un solo nivel.

Para realizar dicha reformulación, se toma el problema del nivel inferior y se encuentra su respectivo problema dual. Sin embargo, en este caso, dicho problema involucra variables continuas y variables binarias. Por lo que una relajación lineal del problema del nivel inferior nos daría una solución que brindará una cota inferior para dicho problema. Lo cual, al evaluar esa solución en la restricción de acoplamiento (Ec. (3.5)), pudiera resultar infactible. De manera similar, al usar la solución óptima de la relajación lineal del nivel inferior obtendríamos diferentes cotas en ambas funciones objetivo del líder, sin poder clasificarlas como cotas inferiores o superiores (Ver [Casas-Ramírez et al., 2018] y [Maldonado-Pinto et al., 2016]).

Además, para lograr la reformulación indicada tendríamos un problema con todas las restricciones del líder, las restricciones del seguidor, las restricciones nivel inferior, y las restricciones que garanticen la optimalidad del nivel inferior. Esto último se puede lograr mediante las restricciones de holgura complementaria o una restricción que garantice la igualdad de las funciones objetivo del problema primal y dual del nivel inferior. Este último conjunto de restricciones nos llevaría a perder la linealidad del modelo. Entonces, tendríamos restricciones no lineales, las restricciones de prohibición de ciclos y un conjunto mucho más grande de restricciones y variables.

Más aún, el hecho de tener todos esos elementos en una supuesta reformulación del problema bi-objetivo binivel en un problema bi-objetivo de un solo nivel, nos conduciría a cotas del problema original, debido a la relajación del problema del nivel inferior. Es por esto, que después de este análisis y con las justificaciones ya mencionadas, se decide diseñar e implementar un algoritmo metaheurístico.

Capítulo 4

Algoritmo propuesto

En este capítulo se presenta el algoritmo propuesto para resolver el problema bajo estudio. En particular, se propone un algoritmo anidado de búsqueda tabú. El propósito de dicho algoritmo es encontrar diversas soluciones no-dominadas para aproximar la frontera de Pareto.

En los problemas binivel, el líder resuelve su problema sujeto a un conjunto de restricciones, en las que se encuentra otro problema de optimización (el del seguidor). Por cada esquema de variables fijadas por el líder que se envíen al grupo de seguidores, estos reaccionarán obteniendo la mejor decisión posible para cada uno de ellos. De esta manera, en cada esquema de variables fijadas por el líder, el seguidor también reaccionará tomando una decisión. Sin embargo, para resolver problemas binivel donde las variables del seguidor afectan la función objetivo del líder, primero, debemos enfocarnos en resolver el nivel inferior resultante a la optimalidad, y luego, completar la evaluación del nivel superior. Este es el enfoque comúnmente utilizado para la optimización binivel.

Como se mencionó anteriormente, la naturaleza entera y binaria de las variables del problema del nivel inferior prohíben aplicar las condiciones de optimalidad y reformular el modelo de programación binivel en uno de un solo nivel. No obstante, si se realiza una relajación lineal del nivel inferior, entonces si se podría proceder con dicha reformulación. Sin embargo, las ecuaciones para evitar los subtours aparecen en las restricciones del nivel superior. Esto último complica claramente la optimización exacta del problema binivel relajado. Por lo tanto, el uso de métodos exactos para resolver este problema binivel bi-objetivo es demasiado complicado y computacionalmente ineficiente. Tal y como se mostró en el capítulo anterior.

Por otro lado, el uso de metaheurísticas para resolver problemas binivel ha sido una opción exitosa en los últimos años (ver [Talbi, 2013]). Las metaheurísticas están diseñadas para resolver problemas complejos de optimización en donde otros métodos no serían capaces de llegar a una solución. Son conocidas por ser un mejor enfoque de solución para muchos problemas complejos, como lo son aquellos basados en el mundo real, los cuales suelen ser problemas con mucha combinatoriedad. Sus ventajas están basadas en su efectividad y su forma general de funcionar. Mientras que su complejidad se basa en adaptar la metaheurística a cada problema en particular. Aunque, esto suele requerir un menor esfuerzo que desarrollar un nuevo método específico de solución. Es importante señalar que una buena implementación de una

metaheurística es capaz de aproximar sus soluciones al óptimo en un tiempo computacional razonable.

Además, las metaheurísticas se han aplicado de manera eficiente para obtener soluciones de buena calidad de problemas binivel de producción-distribución en diferentes contextos, como en [Calvete et al., 2011], [Camacho-Vallejo et al., 2015] y [Nourifar et al., 2020]. En particular, la metaheurística de búsqueda tabú se ha utilizado para aproximar soluciones de problemas de ruteo binivel como en [Marinakis et al., 2007], [Mauttone et al., 2008] y [Marinakis and Marinaki, 2008b].

4.1 Búsqueda tabú

El algoritmo de búsqueda tabú es una metaheurística propuesta por primera vez por Fred Glover en [Glover, 1986]. Es un tipo de metaheurística, para la cual, su forma de operar es moverse de una solución inicial a otra nueva. Su idea principal se enfoca en convertir alguna solución o movimiento en uno tabú, con el fin de evitar ciclos en las búsquedas. Esto es, que se prohibirá realizar dicho cambio mientras este se encuentre en una lista denominada 'lista tabú'. La idea principal de hacer esto último, es ser capaz de realizar peores movimientos para buscar otro camino de soluciones. El estado tabú de un movimiento se pierde después de un tiempo predefinido, convirtiéndolo en un movimiento permitido. Es conveniente señalar que la búsqueda tabú se ha aplicado con éxito para resolver problemas de rutas de vehículos, como en [Li et al., 2012], [Potvin and Naud, 2011], [Renaud et al., 1996] y [Shen et al., 2009].

La idea general de la búsqueda tabú gira en torno a realizar una búsqueda local, sin embargo, el diseño de su algoritmo le permite dar saltos en el espacio de búsqueda con el fin de mejorar las soluciones. Para lograr esto, la búsqueda tabú utiliza las 'listas tabú' como método de exploración para los cuales, posee una lista dinámica y después de cada movimiento, dicho movimiento ingresará a la lista y el más viejo de ellos será removido de la misma. Además, al realizar la exploración siempre se seleccionará la mejor solución no tabú del vecindario, aún y que esta sea peor que la solución actual. Esto último dará la posibilidad de escapar de óptimos locales y redirigir la búsqueda para continuarla estratégicamente en otra área del espacio de soluciones. Dicha lista tabú posee además un tamaño de lista, el cual es el tiempo o número de iteraciones que un movimiento permanece en la lista tabú.

Una elección de tamaño de lista grande o pequeño nos conduce a la intensificación, la cual nos llevará a regresar a regiones ya exploradas para estudiarlas más a fondo. Y a su vez, para evitar regresar a óptimos locales cada cierto número de iteraciones. La búsqueda tabú utiliza además otra estrategia, como es la diversificación; la cual consistirá en visitar nuevas áreas no exploradas del espacio de soluciones. Para ello se modifican las reglas de elección para incorporar a las soluciones atributos que no han sido usados frecuentemente. Una forma clásica de diversificación consiste en reiniciar periódicamente la búsqueda desde puntos elegidos aleatoriamente, como lo es una nueva solución aleatoria del espacio de soluciones.

Debido a los beneficios de exploración presentados para problemas complejos, se decidió adaptar e implementar la metaheurística como principal método de exploración de soluciones. El objetivo consistirá en definir los parámetros adecuados de tamaño de lista tabú, tiempo, así como el momento correcto para recurrir a la diversificación de soluciones.

4.2 Un algoritmo bi-objetivo anidado de búsqueda tabú

En esta sección, se propone un algoritmo bi-objetivo anidado de búsqueda tabú (NBOTS, por sus siglas en inglés) para aproximar el frente de Pareto del problema bi-objetivo binivel definido por las Ec. (3.1) - (3.20). Nuestro algoritmo utiliza las ideas anteriormente descritas en [Kulturel-Konak et al., 2006] para seleccionar aleatoriamente uno de los objetivos para guiar la búsqueda. Además, se aplica el enfoque clásico anidado para resolver problemas de programación binivel; es decir, el problema del seguidor se resuelve de manera óptima para cada solución del líder. A continuación se presenta una descripción detallada del NBOTS propuesto, así como una serie de imágenes para ilustrar mejor el algoritmo propuesto.

4.2.1 Codificación de la solución

La solución es representada como una colección de $\sum_{l \in L} |V(l)| + 1$ cadenas, donde la primera cadena representa a los clientes no seleccionados y las restantes corresponden a cada vehículo.

Es importante especificar que solo las variables de decisión del líder están representadas explícita o implícitamente en la codificación de la solución. En el caso de que el i -ésimo cliente aparezca en una cadena diferente a la primera, significa que se selecciona al i -ésimo cliente, lo que implica que $x_i = 1$. Por lo tanto, la variable x se considera explícitamente en la solución. En contraste, la variable y está implícitamente representada. Dado que y está asociada con la selección de la ruta, la codificación de la solución brinda la información sobre cuales clientes están asociados con cada vehículo, y luego las rutas se crean de manera arbitraria. Con respecto a la variable z , notemos que si hay un cliente asignado en la j -ésima cadena, entonces se está utilizando el vehículo correspondiente y por lo tanto, $z^j = 1$. La variable de decisión del seguidor p se obtendrá al resolver de manera óptima el problema de nivel inferior para cada solución del líder.

Se puede apreciar en la Fig. 4.1 como es la representación de una solución completa para la cual, se puede observar que existen $|I| = 81$ clientes a evaluar y $|V(l)| = 7$ vehículos disponibles para utilizar. Además, como se mencionó anteriormente, para la solución existe una colección de $\sum_{l \in L} |V(l)| + 1 = 8$ cadenas al tomar en cuenta la cadena 0 como la cadena solución de los clientes que no se decide satisfacer su demanda, es decir, aquellos con $x_i = 0$. Cada cliente perteneciente a las cadenas de vehículos son aquellos con $x_i = 1$. También, se puede observar cada cadena con vehículo asignado. Por ejemplo, la cadena 1 nos muestra la ruta a seguir en donde y está implícitamente representada. En esta cadena podemos deducir que $y_{0,33}^{1(1)}, y_{33,12}^{1(1)}, y_{12,9}^{1(1)}, y_{9,45}^{1(1)}, y_{45,18}^{1(1)}, y_{18,20}^{1(1)}, y_{20,0}^{1(1)} = 1$, y así para cada cadena. Finalmente, en este ejemplo se observa que al utilizarse todos los vehiculos, se asocia $z^j = 1$ para cada uno de

ellos.

Parámetros:

$$\begin{aligned} |I| &= 81 & V(I) &= (2, 2, 3) \\ |L| &= 3 & |V(I)| &= 7 \end{aligned}$$

Solución:

0	4	13	16	29	30	44	47	50	53	62
 1	33	12	9	45	18	20				
 2	40	57	51	36	54	15	2			
 3	17	23	66	5	41	22	81	63	3	38
 4	35	46	11	49						
 5	52	21	39	64	1	59	42			
 6	14	56	58	65	79					
 7	43	6	55	19	24	61	48	60		

Figura 4.1: Representación de la solución

4.2.2 Construcción de una solución inicial

Se construye una solución factible inicial para entrar en el esquema de la búsqueda tabú. Primero, dado un rango predefinido por el número mínimo y máximo de clientes deseados, se genera un número aleatorio nc en ese rango. Después, los nc clientes son seleccionados al azar y asignados a un vehículo de manera aleatoria (entre todos los tipos de vehículos), si y solo si, no se excede la capacidad del vehículo.

Una vez que los nc clientes han sido seleccionados y asignados a un vehículo, se utiliza el optimizador CPLEX para obtener la ruta. Al solucionar CPLEX y proponer la ruta, esta entra en un proceso de evaluación de subtours, donde, en caso de existir alguno, se ingresa la restricción correspondiente para evitar dicho ciclo y se procede nuevamente a utilizar el optimizador. Además, el modelo resuelto por CPLEX descuida el tiempo máximo de duración de una ruta. Los resultados preliminares indicaron que se puede obtener un alto grado de infactibilidad al incluir esta restricción. Debido a esto, el esfuerzo computacional aumenta

significativamente. Por lo tanto, eliminamos esta restricción y la ruta construida puede ser infactible en términos del tiempo máximo de duración de la ruta. Para abordar este problema, se incluye un procedimiento que repara una solución infactible en una factible.

4.2.3 Reparando una solución infactible

Una vez establecida la ruta y en el caso de que esta exceda el tiempo máximo de duración, el proceso de reparación para dicho vehículo será el siguiente: se calcula la distancia de cada cliente contra los demás clientes incluidos en el vehículo, donde aquel cliente que posee la mayor distancia total en la ruta (con respecto a los demás clientes), se elimina y este será agregado a la cadena asociada con los clientes no seleccionados. Una vez eliminado este cliente, se volverá a calcular la duración de la ruta y se comprueba su factibilidad. Este procedimiento se repetirá y se continuarán eliminando clientes hasta que no se supere el tiempo máximo de duración de la ruta. Es importante recordar que en el momento de la creación de la solución inicial, los clientes fueron agregados garantizando siempre el cumplimiento de la restricción de capacidad de cada vehículo.

Para ilustrar la reparación de una solución infactible, ver Fig. 4.2, donde se puede observar el caso del vehículo 3 el cual excede v^{max} . Se realiza el cálculo de distancias correspondiente y se encuentra que el 'peor' cliente en base a este criterio es el 23, por lo tanto se procede a eliminarlo de la ruta y agregarlo a la cadena de clientes no seleccionados. Esto nos lleva a la factibilidad del vehículo 3.

4.2.4 Evaluando la función objetivo del líder

Una vez que se obtiene la solución factible del líder, se puede calcular la demanda requerida de los clientes seleccionados para cada producto. Estos valores son dados como parámetros al nivel inferior, que está definido por las Ec. (3.14)-(3.20). Este problema se resuelve de manera óptima utilizando CPLEX. Una vez que se conocen las variables óptimas del seguidor con respecto a las variables fijas del líder, se puede realizar la evaluación de ambas funciones objetivo del líder. Denotaremos a F_1 y F_2 como las emisiones de CO_2 y la función objetivo de ganancia, respectivamente. Luego, se realiza la búsqueda tabú. El algoritmo 1 describe el pseudocódigo para la construcción de una solución factible.

Parámetros:

$$v^{max} = 500$$

Solución:

	3	17	23	66	5	41	22	81	63	3	38	$v = 542$
---	---	----	----	----	---	----	----	----	----	---	----	-----------

$v > v^{max} \therefore$ INFECTIBLE

Procedimiento de calculo de distancias:

	3	17	23	66	5	41	22	81	63	3	38	$v = 542$
		↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	
		74	119	61	70	97	114	103	101	82	65	

Actualización de la ruta (eliminando cliente 23)

	3	17	66	5	41	22	81	63	3	38	$v = 496$
0		4	13	16	29	30	44	47	50	53	+23

$v < v^{max} \therefore$ FACTIBLE

Figura 4.2: Reparando una solución infactible

Algorithm 1: Construcción de una solución factible

Input : nc

- 1 $x' \leftarrow$ Seleccionar aleatoriamente nc clientes ;
- 2 $(y, z) \leftarrow$ Rutear_CPLEX(x');
- 3 **if** *infactible* **then**
- 4 $x \leftarrow$ Reparar(x');
- 5 $(y, z) \leftarrow$ Rutear_CPLEX(x);
- 6 **else**
- 7 $x \leftarrow x'$;
- 8 **end**
- 9 $p \leftarrow$ ResolverNivelInferior_CPLEX(x);
- 10 $F_1 \leftarrow$ Evaluar $F_1(y, p)$;
- 11 $F_2 \leftarrow$ Evaluar $F_2(x, y, z, p)$;

Output: x, y, z, p, F_1, F_2

4.2.5 Descripción del algoritmo NBOTS

A continuación se presenta una descripción detallada del algoritmo NBOTS:

1. **Inicialización** Construir una solución factible para el líder con el procedimiento descrito anteriormente y vaya al Paso 1. Inicialice la lista de soluciones no dominadas (ND) como vacía.
2. **Seleccionar el objetivo** Seleccione uno de los dos objetivos para activar utilizando una función de masa de probabilidad de Bernoulli. Esta probabilidad varía en cada iteración.
3. **Búsqueda en vecindario** Se consideran dos vecindarios (N1 y N2) para cada función objetivo. Sin embargo, N1 varía según el objetivo seleccionado. N2 sigue siendo el mismo para ambas funciones objetivo.

Para el objetivo relacionado con las emisiones (F_1), N1 consiste en eliminar una cantidad de clientes seleccionados de cada vehículo. El número de clientes a eliminar se elige aleatoriamente como un porcentaje de los clientes seleccionados (entre 0 y 40 %). Para decidir qué clientes serán eliminados del vehículo, se elige el cliente más lejano (en términos de distancia). El procedimiento continúa hasta que se alcanza el número de clientes a retirar de cada vehículo.

Para el objetivo relacionado con la ganancia (F_2), N1 consiste en ordenar a los clientes no seleccionados en términos de su demanda, en orden decreciente. Además, los vehículos se ordenan en función de su capacidad restante, en orden decreciente. Después, cada cliente no seleccionado con la mayor demanda se inserta en cada vehículo con suficiente capacidad restante. La inserción se realiza en la posición más conveniente posible para la ruta. Por lo tanto, se realiza el mejor movimiento. El procedimiento se repite hasta que ningún cliente no seleccionado pueda insertarse en una ruta, lo que ocurre cuando se viola la restricción de capacidad del vehículo o la duración máxima de la ruta.

Es importante mencionar que después de cada eliminación o inserción de un cliente en N1, varía el número de clientes seleccionados y la demanda total acumulada. Para abordar estos cambios, la ruta debe volver a optimizarse; y, para este último, el problema de nivel inferior debe resolverse nuevamente de manera óptima. Si se está considerando F_1 , entonces un movimiento en N1 conduce a una disminución en las emisiones de CO_2 ; por otro lado, la ganancia aumenta para F_2 .

Posteriormente, la solución mejorada entra en N2, que consiste en cambiar de clientes entre vehículos. Cada cliente se insertará en la mejor posición de un vehículo diferente. Se realizará el mejor movimiento. En este vecindario, los clientes se trasladan a un vehículo diferente, manteniendo la demanda acumulada al mismo nivel. En este caso, las rutas también deben volver a optimizarse, pero no es necesario resolver el problema del nivel inferior.

Una vez que se han explorado ambos vecindarios, se obtiene una solución candidata y se actualizan sus valores de función objetivo correspondientes.

4. **Actualizar la lista de soluciones ND** Compare la solución candidata con la lista de soluciones ND actual de la siguiente manera: si la solución candidata domina al menos

una solución en ND, elimine estas soluciones dominadas del conjunto ND y agregue la solución candidata a ND. Además, se debe agregar a ND una solución candidata que no esté dominada por ninguna solución actual que pertenezca a ND.

5. **Actualizar las estructuras tabú** Agregue el movimiento aceptado en el Paso 2 a la lista tabú y actualice las iteraciones restantes en tabú para otros movimientos prohibidos. El número de iteraciones que un movimiento permanecerá como tabú se elige aleatoriamente entre 8 y 15. En caso de que la solución candidata esté en la lista tabú, pero domine cualquier solución en la lista de soluciones ND, se incluirá en el conjunto ND (criterio de aspiración).
6. **Diversificación** Se utiliza un esquema de diversificación basado en el reinicio. Si la lista ND no se ha actualizado en las últimas iteraciones ($\text{MaxIter}/4$), una de las soluciones ND encontradas durante la búsqueda se selecciona uniformemente como la nueva solución actual. Ambas estructuras tabú se restablecen a vacías y la búsqueda se reinicia desde la solución seleccionada, es decir, regresa al Paso 1.
7. **Criterio de paro** Mientras no se satisfaga el criterio de paro, regrese al Paso 1. En este algoritmo, se consideran dos criterios de paro: un número máximo de iteraciones (MaxIter) realizadas sin actualizar ND o un límite de tiempo máximo de CPU.

En el Algoritmo 2 se representa el funcionamiento de los NBOTS. La búsqueda de los vecindarios para F_1 y F_2 se muestra en los algoritmos 3 y 4, respectivamente.

Algorithm 2: Pseudocódigo general para el algoritmo NBOTS

Input : x, y, z, p, F_1, F_2

```

1 Inicializar las estructuras tabú;
2  $\text{ND} \leftarrow \emptyset$ ;
3 while Criterio de paro no sea alcanzado do
4    $\text{rand} \leftarrow$  Generar un número aleatorio entre 0 y 1;
5   if  $\text{rand} \leq \text{Be}(0.5)$  then
6     //  $F_1$  es seleccionado ;
7      $(x, y, z, p)$  entra a N1 (remover);
8      $(x, y, z, p)$  entra a N2 (cambiar) ;
9   else
10    //  $F_2$  es seleccionado ;
11     $(x, y, z, p)$  entra a N1 (insertar);
12     $(x, y, z, p)$  entra a N2 (cambiar);
13  end
14  Actualización de las estructuras ND y tabú;
15  if Criterio de diversificación es alcanzado then
16    Seleccionar aleatoriamente una solución en ND;
17    Reiniciar las estructuras tabú;
18  end
19 end

```

Output: ND

Algorithm 3: Exploración de los vecindarios N1 y N2 para F_1

Input : x, y, z, p, F_1

```

1 // N1 es explorado;
2 for cada vehículo do
3    $rem \leftarrow$  Generar un número aleatorio entre 0 y 0.4;
4    $aux \leftarrow 1$ ;
5    $ncv \leftarrow$  Número de clientes en el vehículo ;
6   while  $aux \leq \lfloor rem \times ncv \rfloor$  do
7      $x \leftarrow x \setminus \{\text{cliente mas lejano}\}$  ;
8      $(y, z) \leftarrow$  Rutear_CPLEX( $x$ );
9      $aux \leftarrow aux+1$  ;
10  end
11   $p \leftarrow$  ResolverNivelInferior_CPLEX( $x$ );
12   $F_1 \leftarrow$  Evaluar  $F_1(y, p)$  ;
13 end
14 // N2 es explorado;
15 while  $F_1$  no pueda ser mejorado do
16   for cada cliente seleccionado do
17     for cada vehículo do
18        $x \leftarrow$  Insertar el cliente en la posición mas conveniente (si es posible);
19        $(y, z) \leftarrow$  Recalcular_Ruta( $x$ );
20        $F_1 \leftarrow$  Evaluar  $F_1(y, p)$  ;
21     end
22   end
23    $x \leftarrow$  Realizar el mejor cambio ;
24    $(y, z) \leftarrow$  Rutear_CPLEX( $x$ );
25    $p \leftarrow$  ResolverNivelInferior_CPLEX( $x$ );
26    $F_1 \leftarrow$  Evaluar  $F_1(y, p)$  ;
27 end
28  $F_2 \leftarrow$  Evaluar  $F_2(x, y, z, p)$  ;

```

Output: x, y, z, p, F_1, F_2

Algorithm 4: Exploración de los vecindarios N1 y N2 para F_2

Input : x, y, z, p, F_2

```

1 // N1 es explorado;
2 Ordenar los clientes no-seleccionados en orden decreciente (en términos de la demanda);
3 Ordenar los vehiculos con capacidad restante en orden decreciente (en terminos de su
  capacidad);
4 while un cliente no-seleccionado pueda ser insertado do
5   for cada cliente no-seleccionado do
6     for cada vehiculo con capacidad restante do
7        $x \leftarrow x \cup \{\text{cliente}\}$  ;
8        $(y, z) \leftarrow \text{Recalcular\_Ruta}(x)$ ;
9        $p \leftarrow \text{ResolverNivelInferior\_CPLEX}(x)$ ;
10       $F_2 \leftarrow \text{Evaluar } F_2(x, y, z, p)$  ;
11     end
12      $x \leftarrow \text{Insertar al cliente en el vehículo mas conveniente}$  ;
13   end
14    $(y, z) \leftarrow \text{Rutear\_CPLEX}(x)$ ;
15    $F_2 \leftarrow \text{Evaluar } F_2(x, y, z, p)$  ;
16 end
17 // N2 es explorado;
18 while  $F_2$  no pueda ser mejorado do
19   for cada cliente seleccionado do
20     for cada vehículo do
21        $x \leftarrow \text{Insertar al cliente en la posición mas conveniente (si es posible)}$ ;
22        $(y, z) \leftarrow \text{Recalcular\_Ruta}(x)$ ;
23        $F_2 \leftarrow \text{Evaluar } F_2(x, y, z, p)$  ;
24     end
25   end
26    $x \leftarrow \text{Realizar el mejor cambio}$  ;
27    $(y, z) \leftarrow \text{Rutear\_CPLEX}(x)$ ;
28    $F_2 \leftarrow \text{Evaluar } F_2(x, y, z, p)$  ;
29 end
30  $F_1 \leftarrow \text{Evaluar } F_1(y, p)$  ;

```

Output: x, y, z, p, F_1, F_2

Finalmente, se presentan las figuras Fig. 4.3 y 4.4, las cuales serán utilizadas para resumir el algoritmo del presente capítulo. El diagrama de flujo presentado en la Fig. 4.3 muestra los pasos descritos en las subsecciones 4.2.1 - 4.2.4 en las cuales se realiza la construcción de una solución inicial factible completa, la cual incluye la reparación de la solución en caso de ser necesario para garantizar factibilidad, así como la resolución del nivel inferior. Por último, el diagrama de flujo presentado en la Fig. 4.4 resume la construcción del algoritmo NBOTS presentado en la subsección 4.2.5. Este último muestra los 2 vecindarios de exploración $N1$ y $N2$, donde en el primer caso se elige un objetivo a explorar: beneficiando a la ganancia para el cual realiza la inserción de clientes en vehiculos o beneficiando el objetivo de emisiones para el cual removerá cierta cantidad de clientes de cada vehículo; posteriormente muestra el

vecindario N2 de intercambio de clientes. Además muestra el momento en el que se actualizan las estructuras y listas de soluciones no dominadas y el momento en el que puede ser necesario recurrir a la intensificación en la búsqueda de soluciones para aproximar la frontera eficiente.

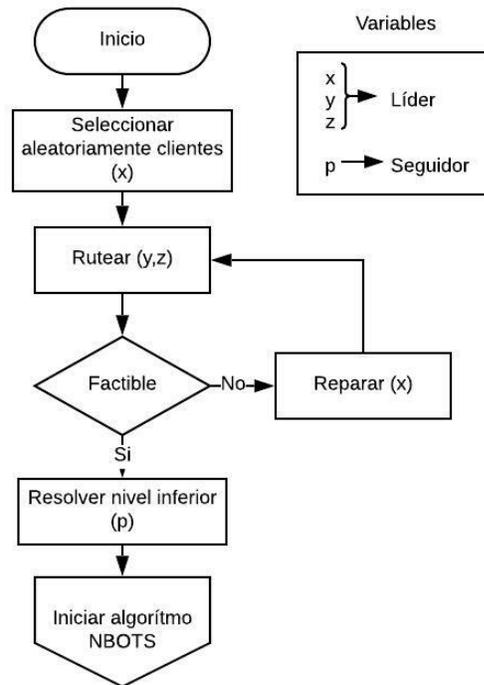


Figura 4.3: Construcción de una solución inicial factible

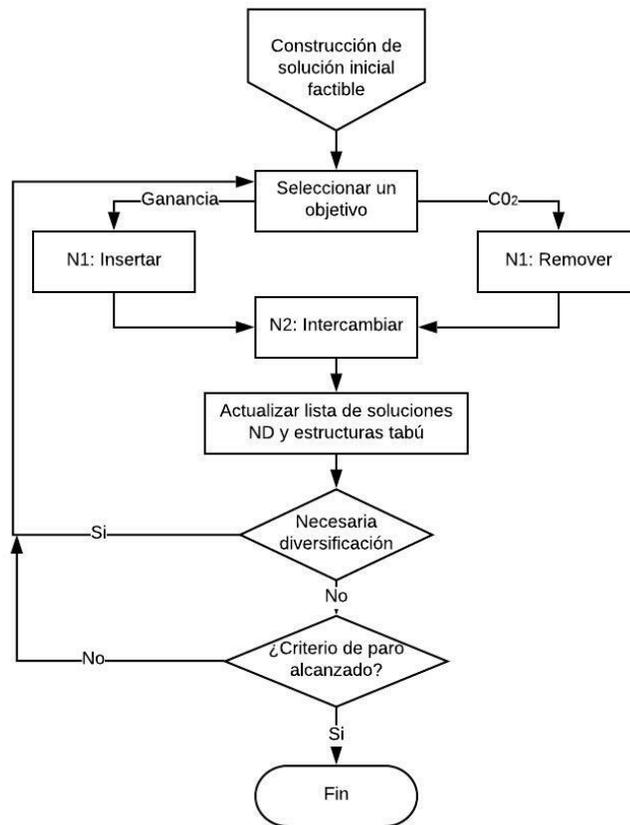


Figura 4.4: Algoritmo NBOTS completo

Capítulo 5

Experimentación Computacional

En este capítulo se presenta la experimentación computacional realizada para probar la eficiencia del algoritmo propuesto y su capacidad para aproximar el frente de Pareto. Particularmente, se detalla la generación de las instancias adaptadas y el ambiente computacional, así como la experimentación numérica. Además, se presenta el análisis de los resultados mostrados y la discusión de la efectividad del algoritmo para aproximar la frontera eficiente y así proponer cambios futuros.

5.1 Instancias y ambiente computacional

Para obtener un conjunto apropiado de instancias de referencia, se consideró el conjunto de instancias para el problema de generación de rutas de vehículos con entregas divididas (VRPwSD, por sus siglas en inglés) considerado en [Cordeau et al., 2002]. Dichas instancias fueron complementadas con los datos de emisiones y aquellos involucrados en el proceso de fabricación. Las instancias del VRPwSD contienen los datos sobre la demanda de los clientes y su ubicación, capacidad y número de vehículos, y tiempo máximo de duración de cada ruta.

Los parámetros relacionados con las plantas se generaron siguiendo el procedimiento descrito en [Calvete et al., 2011]. El número de plantas ubicadas se eligió entre 3 y 7. Luego, las ubicaciones de las plantas se fijaron aleatoriamente a partir del cuadrado de $[-200, 200] \times [-200, 200]$. Los costos de adquisición de los productos se seleccionaron del intervalo $[0.5, 1.5]$. Para los costos de envío, se generó aleatoriamente un número de $[2, 5]$ y se agregó al siguiente término $0.5 \times$ distancia (planta, centro de distribución). Finalmente, la capacidad de producción se generó aleatoriamente a partir del siguiente intervalo [demanda total / número de plantas, demanda total].

Los parámetros restantes se generaron de la siguiente manera: en base a la proporcionalidad existente en las instancias, φ se establece en 1. Además, el factor que relaciona el costo y la distancia recorrida por cada vehículo (c_l) y la ganancia por producto (g_n) se establecen en 100. Con respecto a los parámetros de emisión de CO_2 , se considera 210 como base para las emisiones de los vehículos, que se multiplica por los porcentajes correspondientes obtenidos de [Hill et al., 2013] para reflejar los factores de emisión del transporte terrestre de carga. Se

considera como base un promedio de la capacidad máxima de producción de las plantas, las emisiones máximas de CO_2 por planta serían la base multiplicada por la capacidad de producción. Se realiza un proceso similar para las emisiones de CO_2 por planta. Se toma como base la relación entre el promedio de la capacidad máxima de producción y el menor costo de producción y envío de las plantas, luego este número se multiplica por el costo de producción y envío de cada planta. El beneficio mínimo aceptable para la empresa distribuidora se establece en 2000.

El proceso explicado anteriormente se implementó para generar 12 instancias diferentes para llevar a cabo nuestra experimentación computacional. Las principales diferencias entre estas instancias consisten en lo siguiente: el número de clientes varía entre 48 y 1008; el tipo de vehículos entre 4 y 6; el número de vehículos de cada tipo entre 1 y 21; y las plantas entre 3 y 7.

Las pruebas se realizaron en una computadora con una CPU Intel Core i5-4440U de 3.10 GHz con 8 GB de RAM. Se utilizaron las librerías con la versión 12.6 de CPLEX para el ruteo y se resolvió el problema de nivel inferior a través de Visual Studio 2012. La tabla 5.1 muestra los tamaños de las instancias.

Instancia	Clientes ($ I $)	Plantas ($ M $)	Tipos de vehículo ($ L $)	Cantidad por tipo ($ V(l) $)
1	48	4	4	1
2	96	3	4	2
3	144	4	4	3
4	192	3	4	4
5	240	3	4	5
6	288	4	4	6
7	73	3	6	1
8	145	4	6	2
9	217	7	6	3
10	289	5	6	4
11	1008	4	4	21
12	721	3	6	10

Tabla 5.1: Tamaño de las instancias

5.2 Descripción de un algoritmo voraz

Con el objetivo de evaluar el desempeño de nuestro algoritmo propuesto, es necesario establecer un punto de comparación contra resultados encontrados mediante otros métodos. Debido a que nuestra problemática presenta un conjunto de instancias nuevas y adaptadas de la literatura, resulta imposible comparar los resultados con otros trabajos propuestos con anterioridad. Debido a esto, es necesaria la creación e implementación de otro método de solución.

Por lo cual, se propone comparar el desempeño del algoritmo propuesto en el capítulo anterior con un algoritmo voraz (greedy).

Los algoritmos heurísticos voraces emplean estrategias de búsqueda que se basan en encontrar la solución más prometedora en cada paso de la búsqueda durante la creación de una solución. Esto permite, encontrar de forma rápida a una buena solución que satisfaga cada una de las restricciones del problema. Esto último, generalmente se realiza en tiempos computacionales pequeños, lo cual resulta atractivo como un método de comparación.

La idea general de la implementación del algoritmo será la siguiente: el algoritmo voraz será utilizado para calcular una cota en la ganancia y, en consecuencia, en las emisiones de CO_2 . La característica clave del algoritmo voraz consiste en incluir a los clientes en los vehículos, de modo que se logre el máximo beneficio posible. Las limitaciones de capacidad y duración máxima de una ruta se consideran durante la asignación de los clientes. En el caso en que no se pueda incluir a un cliente en ningún vehículo, se mantienen como no seleccionados. Este procedimiento finaliza cuando se han considerado todos los clientes. Finalmente, se evalúan las ganancias y las emisiones de CO_2 asociadas con la solución voraz factible.

Al implementar este algoritmo voraz, lo que se encuentra es una única solución en cada corrida del algoritmo para cada instancia. Es decir, debido a los criterios voraces durante la construcción, el algoritmo siempre encontrará la misma solución. Esta es una de las principales desventajas de los algoritmos voraces en problemas en los que buscamos aproximar a una frontera eficiente, en comparación con el algoritmo propuesto en el capítulo anterior, el cual busca encontrar la mayor cantidad de soluciones factibles no dominadas.

5.3 Resultados numéricos

El algoritmo NBOTS propuesto contiene muy pocos parámetros que necesitan calibración. Recuerde que la probabilidad de seleccionar un objetivo en la búsqueda local se actualiza a través de las iteraciones. El tiempo que un movimiento permanecerá como tabú se elige al azar entre 8 y 15 en cada iteración. Estos valores se fijan basados en [Kulturel-Konak et al., 2006] y la experimentación preliminar. Hay que tomar en cuenta que la longitud de la lista tabú es dinámica y depende de la cantidad de clientes y vehículos considerados en la instancia.

En la búsqueda local, particularmente en $N1$ para F_1 , se elige la cantidad de clientes que se eliminarán. Para mantener la diversidad, eliminamos aleatoriamente entre el 0 y el 40 % de los clientes seleccionados. Los demás parámetros están relacionados con la diversidad y con los criterios de paro: un número máximo de iteraciones sin actualizar la lista ND y un límite de tiempo de ejecución de la CPU, respectivamente. El primero está relacionado con el tamaño de la instancia y el segundo se establece en 7200 segundos. La experimentación mostró que el criterio siempre utilizado para detener la ejecución del programa fue el límite de tiempo máximo. Debido a esto, el tiempo de ejecución no se muestra en los resultados.

Considerando la aleatoriedad involucrada en el algoritmo, se realizaron 10 ejecuciones por

Instancia	Min	Max	Promedio
1	31	50	42
2	49	78	65
3	42	65	52
4	52	62	58
5	45	62	55
6	34	49	46
7	46	60	53
8	34	42	37
9	37	50	46
10	39	67	58
11	32	41	36
12	35	54	50

Tabla 5.2: Número de soluciones no dominadas obtenidas por el NBOTS

instancia. Los resultados obtenidos de la experimentación se muestran en la Tabla 5.2. La primera columna representa el número de la instancia a la que hacemos referencia. Las columnas restantes están asociadas con las soluciones no dominadas. La segunda y tercera columnas muestran el número mínimo y máximo de soluciones no dominadas obtenidas por el algoritmo, respectivamente. Finalmente, la última columna muestra el promedio de las soluciones no dominadas entre las diez ejecuciones del algoritmo.

En la Tabla 5.2 se puede ver que el NBOTS obtuvo al menos 32 soluciones no dominadas. El mayor número de soluciones no dominadas corresponde a la instancia 4, en la que se obtuvieron 69 soluciones. En promedio, hay al menos 38 soluciones no dominadas para cada instancia. Estos últimos nos permiten aproximarnos adecuadamente a los frentes de Pareto.

Para validar lo anterior, los frentes de Pareto aproximados se trazan en las Figuras 5.1 a 5.12. La corrida con el mayor número de soluciones no dominadas se graficó para cada instancia. Las aproximaciones de los frentes de Pareto son representativos y cubren un amplio espacio. Es importante recordar que estamos asumiendo que la empresa distribuidora tiene una ganancia mínima aceptable de 2000.

Los frentes de Pareto aproximados obtenidos por el algoritmo NBOTS propuesto se trazan en azul, mientras que el límite obtenido por el algoritmo greedy se traza en rojo.

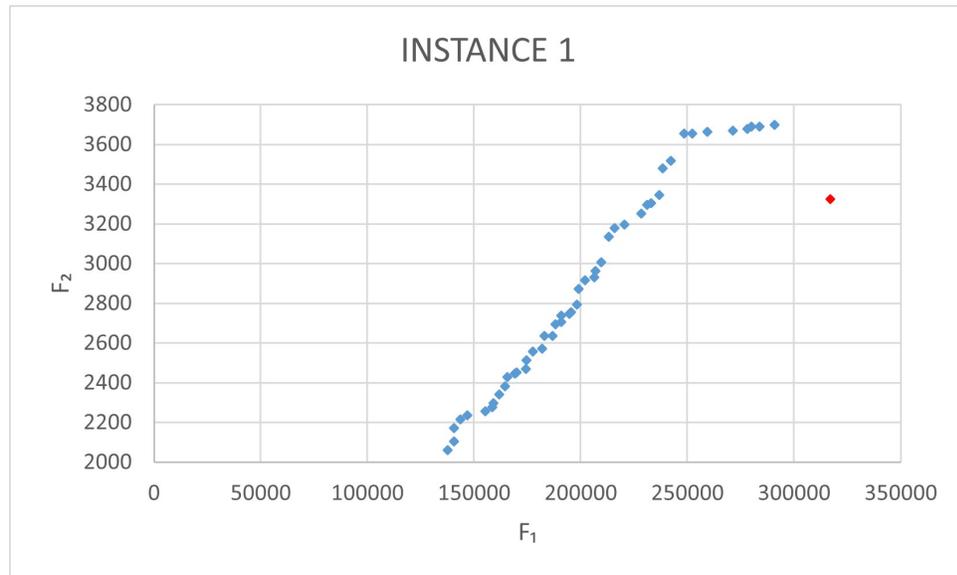


Figura 5.1: Frente de pareto aproximado para la instacia 1

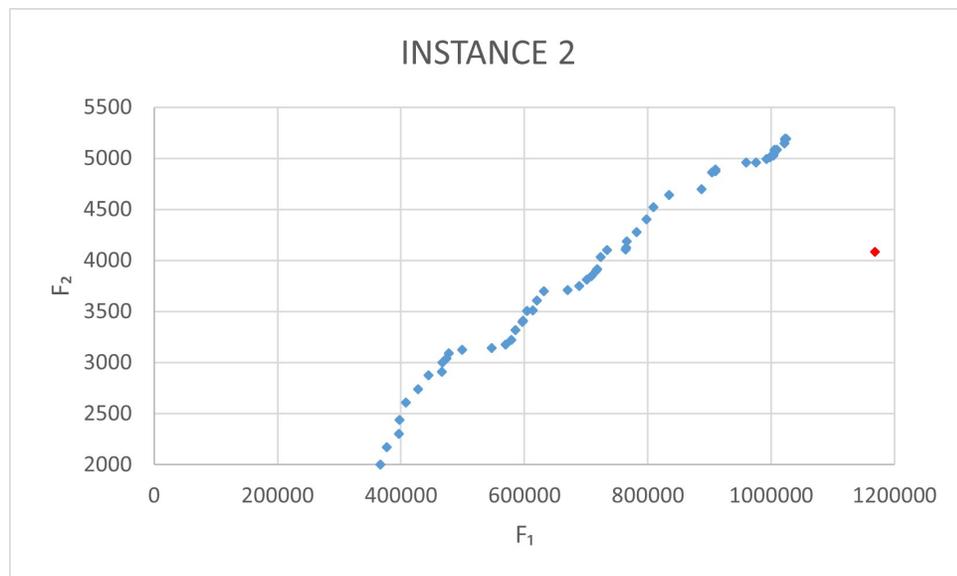


Figura 5.2: Frente de pareto aproximado para la instacia 2

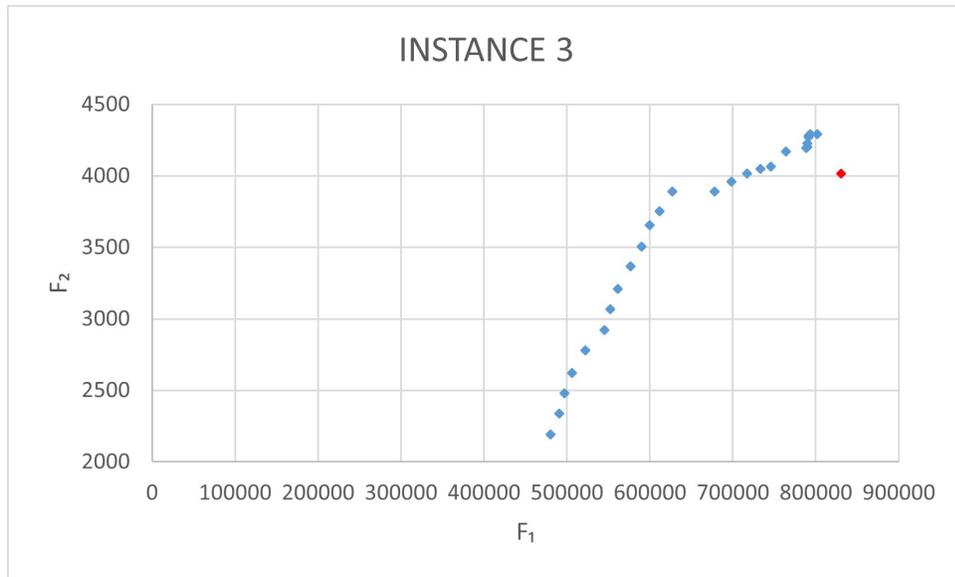


Figura 5.3: Frente de pareto aproximado para la instacia 3

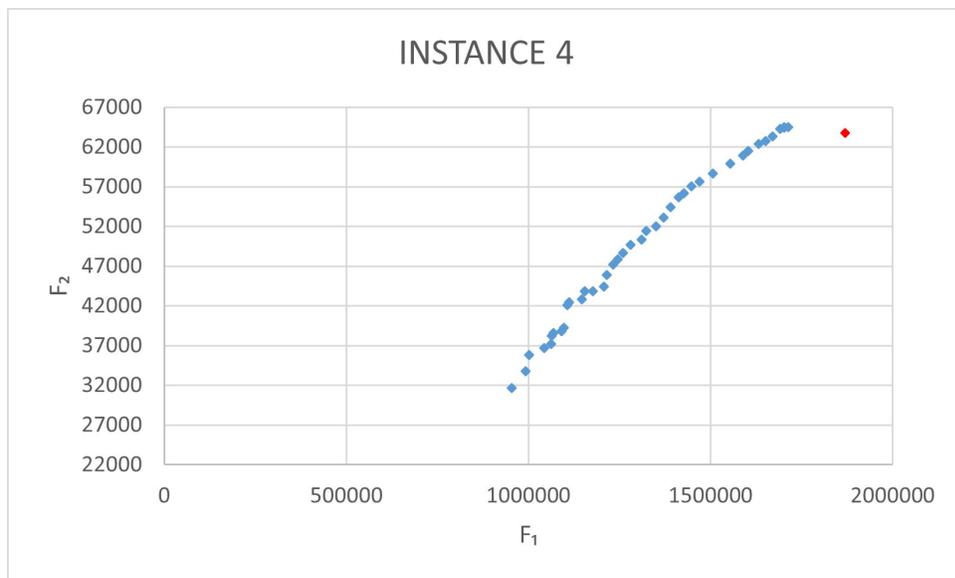


Figura 5.4: Frente de pareto aproximado para la instacia 4

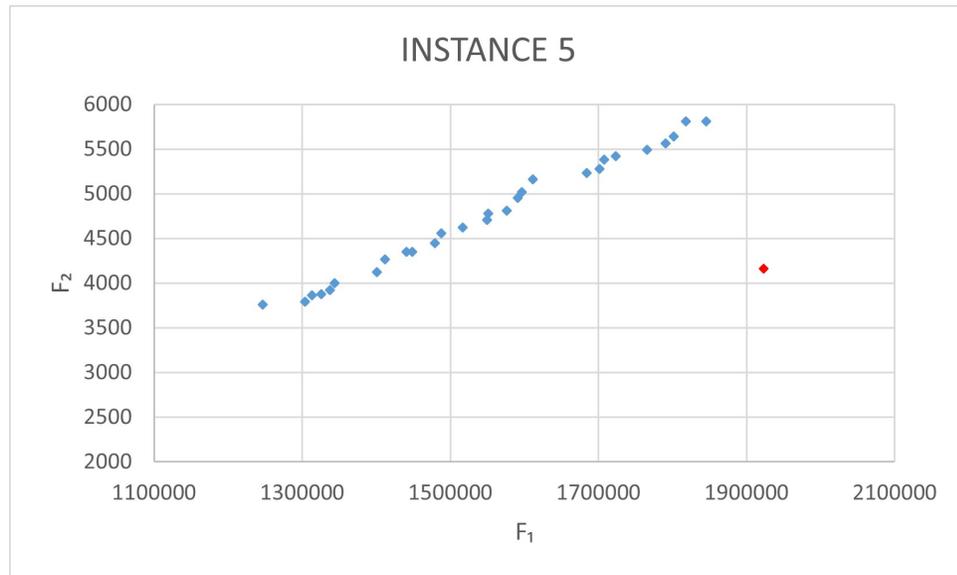


Figura 5.5: Frente de pareto aproximado para la instacia 5

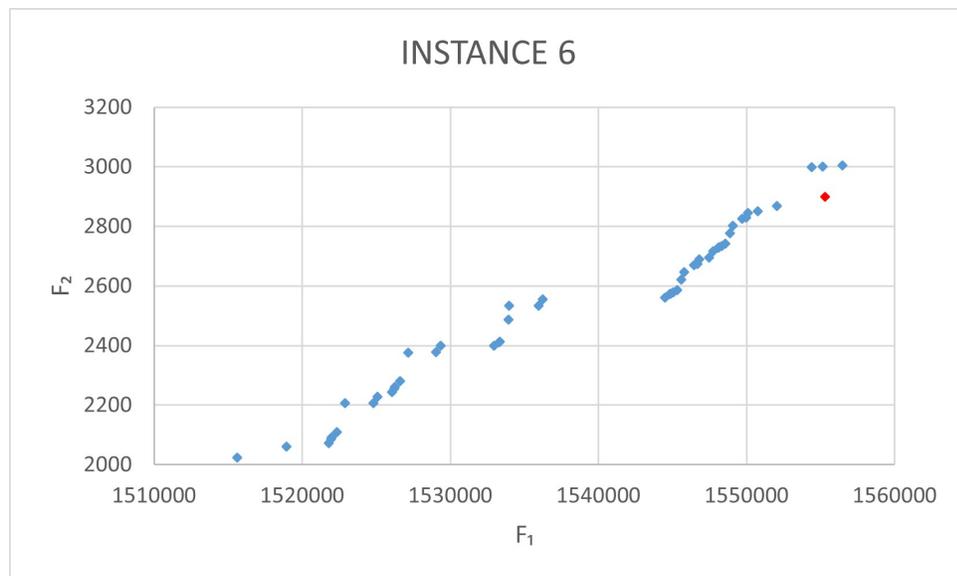


Figura 5.6: Frente de pareto aproximado para la instacia 6

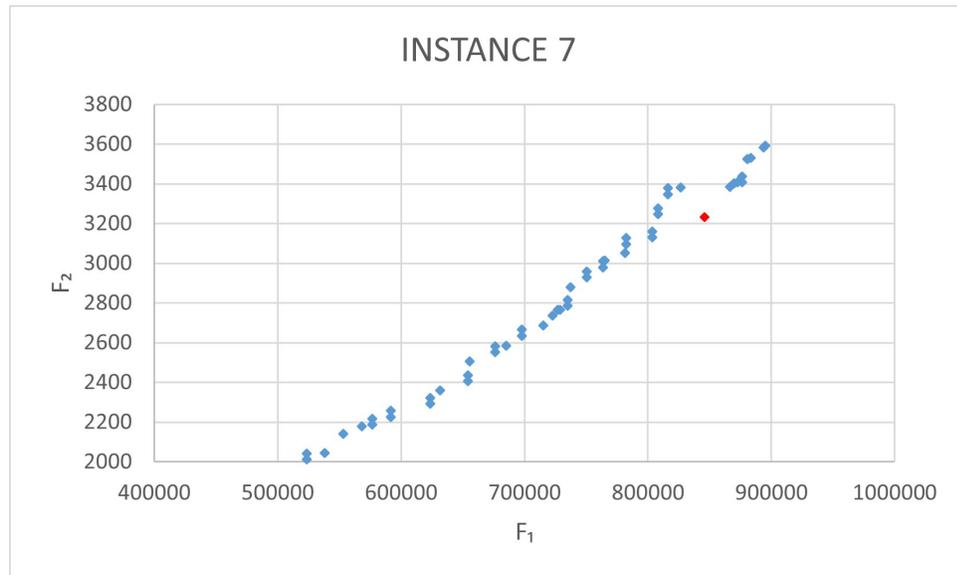


Figura 5.7: Frente de pareto aproximado para la instacia 7

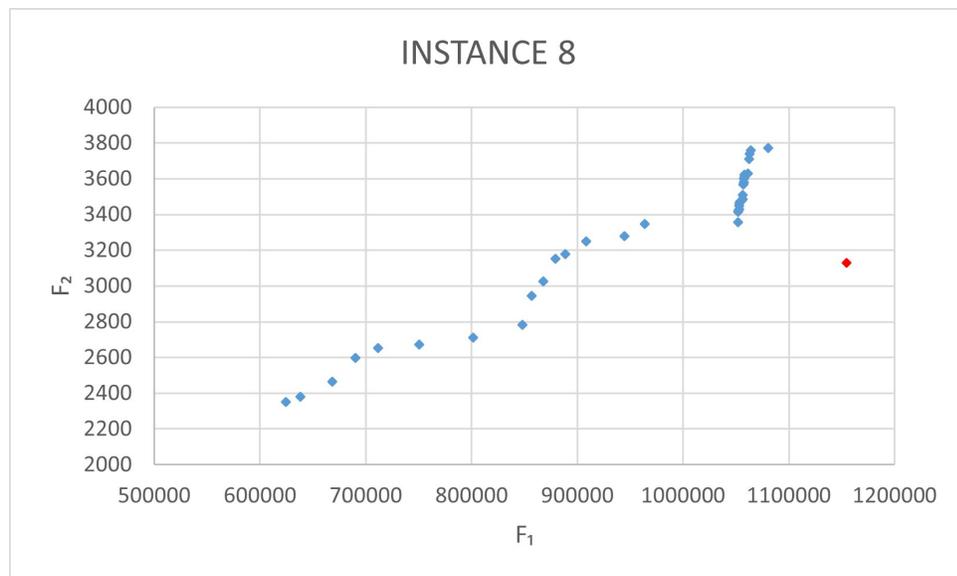


Figura 5.8: Frente de pareto aproximado para la instacia 8

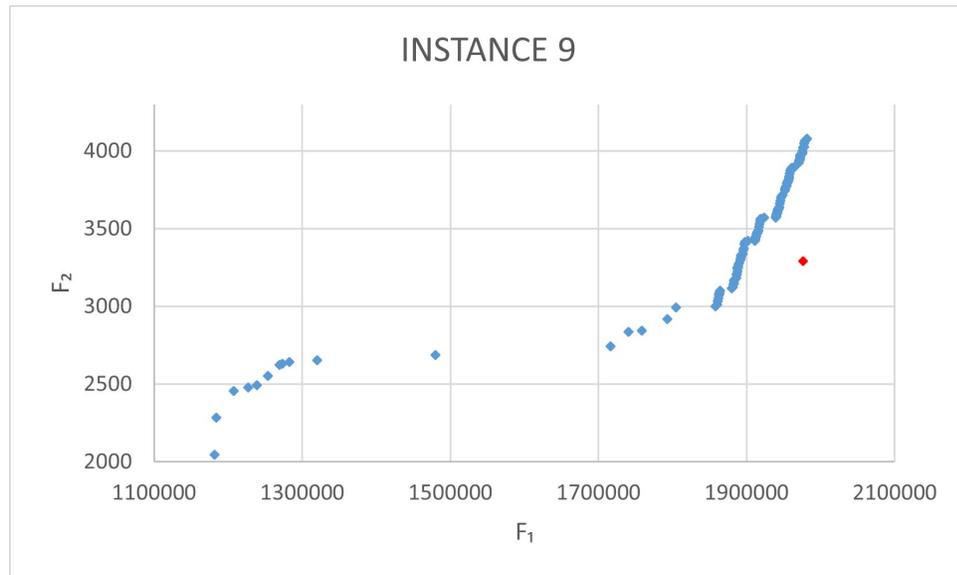


Figura 5.9: Frente de pareto aproximado para la instacia 9

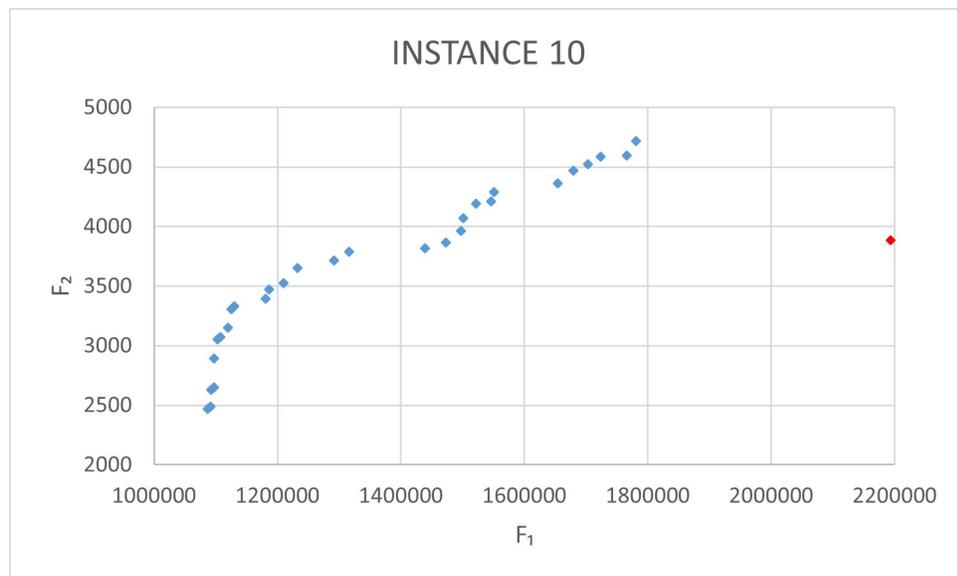


Figura 5.10: Frente de pareto aproximado para la instacia 10

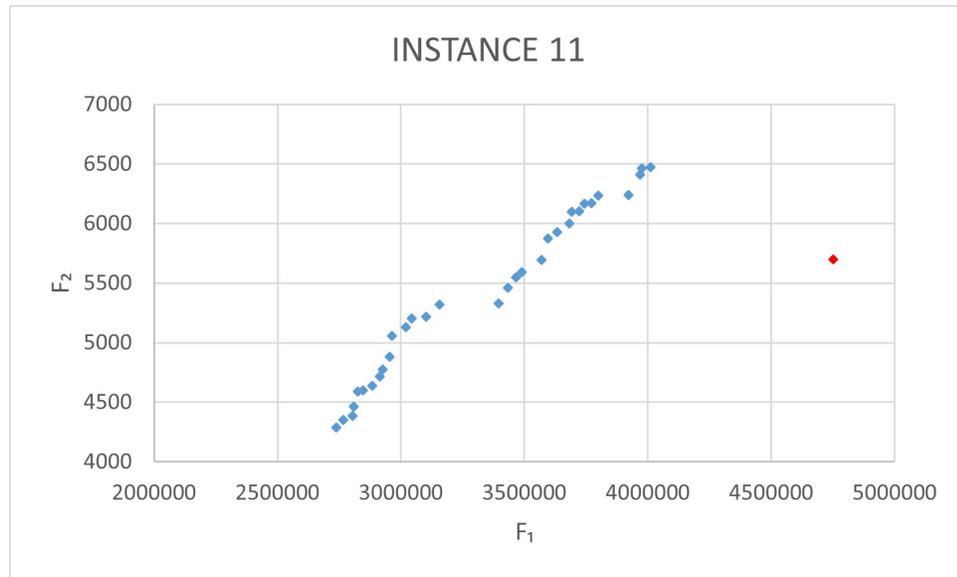


Figura 5.11: Frente de pareto aproximado para la instancia 11

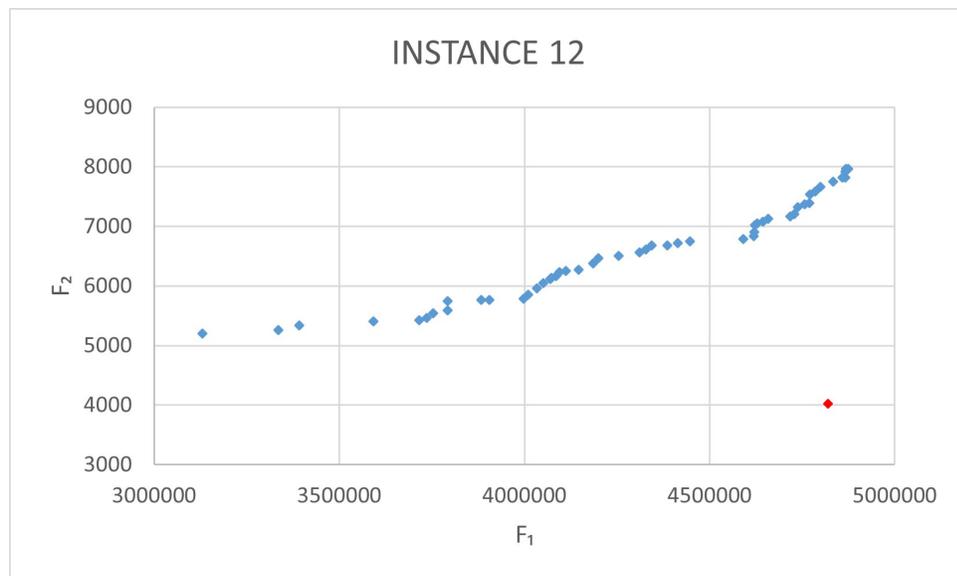


Figura 5.12: Frente de pareto aproximado para la instancia 12

Con respecto a la ganancia, las instancias 1, 5, 7, 10 y 12 (ver Figuras 5.1, 5.5, 5.7, 5.10, y 5.12) encontraron soluciones que proporcionen un beneficio superior al mínimo aceptable (2000). Esta es una buena característica del NBOTS e implica que no tiene problemas con la restricción de acoplamiento, la cual complica enormemente el problema. La instancia 4 tiene un valor de umbral más alto. Sus ganancias comenzaron alrededor de 35000 (ver Figura 5.4), por debajo de ese valor, se reportaron pérdidas. Un caso similar ocurrió en la instancia 11, como se muestra en la Figura 5.11, donde la ganancia comenzó en 5500.

Los resultados obtenidos muestran que el NBOTS propuesto es capaz de encontrar un número suficiente de soluciones no dominadas. En otras palabras, el algoritmo es capaz de encontrar los movimientos más convenientes en función de la función objetivo considerada, es decir, incluir clientes para aumentar la ganancia o deseleccionar clientes que generan las mayores emisiones de CO_2 . Como resultado, se aproxima un frente de Pareto de buena forma. Además, el límite obtenido por el algoritmo voraz siempre está dominado por la aproximación obtenida de los frentes de Pareto. Es de destacar que todos los frentes de Pareto aproximados parecen no ser convexos, lo que indica que el NBOTS propuesto es capaz de superar este problema. Entonces, parece que se está realizando una buena exploración del espacio con la búsqueda propuesta.

5.4 Evaluando la calidad de las aproximaciones a la frontera de Pareto

La importancia de tener un conjunto de soluciones no dominadas en lugar de una solución única está en el apoyo que brinda al tomador de decisiones. Por ejemplo, si la contaminación en el medio ambiente alcanza un nivel alto, el tomador de decisiones puede considerar las emisiones máximas de CO_2 que podrían producirse pero teniendo en cuenta la ganancia. En este caso, ya se conoce la solución factible que el tomador de decisiones puede implementar para superar este problema, la cual es dada por el algoritmo NBOTS. Por otro lado, si el aire está más limpio, se podría implementar una solución que brinde más ganancias aunque se contamine más. Se podría lograr una compensación entre las ganancias y las emisiones de CO_2 en función de las condiciones reales del medio ambiente.

Capítulo 6

Conclusiones

En esta tesis se estudió un problema de cadena de suministro en la que una empresa distribuye productos entre un subconjunto seleccionado de clientes y en donde se considera el proceso de fabricación de dichos productos. Además, se abordó un enfoque de un problema de logística verde, el cual considera la preocupación por el medio ambiente y el impacto que la producción y distribución de productos tiene al generar emisiones de CO_2 y liberarlas en el medio ambiente. Una preocupación que ha ido en aumento por el daño que estas producen en nuestra sociedad y por las nuevas restricciones gubernamentales que se están estableciendo. Es conocido que este impacto de contaminación se reduce al limitar la cantidad de clientes a los que es necesario surtir. Por lo cual, este problema buscó encontrar un balance entre la ganancia (principal objetivo en toda industria) y la creciente preocupación por el medio ambiente. Para esto la empresa distribuidora definió dos objetivos: la maximización del beneficio obtenido por el proceso de distribución y la minimización de las emisiones de CO_2 . La mejor forma de abordar esta situación fue modelarla como un problema de programación bi-objetivo binivel, la cual era capaz de plasmar la problemática completa y necesidades de cada tomador de decisiones. En donde, la principal motivación del problema era ayudar al tomador de decisiones (empresa distribuidora) a encontrar el punto de quiebre entre las ganancias y las emisiones de CO_2 . Esto se lograba encontrando la mejor aproximación a la frontera de Pareto.

Debido a la complejidad inherente para encontrar la frontera de Pareto exacta, se propuso un algoritmo basado en la búsqueda tabú. En donde, una de las principales características del algoritmo propuesto es tener dos vecindarios de búsqueda diferentes para la intensificación. El primero de ellos, varía según el objetivo seleccionado para mejorar la ganancia o las emisiones de CO_2 . El objetivo de la ganancia mejoraba en dicho vecindario mediante un proceso de inserción de clientes, mientras el objetivo de las emisiones de CO_2 mejoraba con un proceso de remover clientes. Después, ambos objetivos compartían un segundo vecindario de búsqueda propuesto basado en el intercambio de clientes. Ambos vecindarios diversificaban sus búsquedas con el principal atributo de este algoritmo: la lista tabú.

Los resultados numéricos de la experiencia computacional muestran que el rendimiento del algoritmo de búsqueda tabú propuesto es muy aceptable. El método reduce el esfuerzo computacional al realizar una búsqueda local controlada en diferentes vecindarios y al realizar una exploración diversa del espacio de decisión. Los resultados indican que el algoritmo es

capaz de encontrar una gran cantidad de soluciones no dominadas. Ésta es la aproximación de la frontera de Pareto. En consecuencia, las fronteras de Pareto aproximadas cubren una gran región en el espacio de decisión. Esto último representa un gran logro y ventaja para el tomador de decisiones, en donde se ve reflejada la importancia de tener un conjunto grande de soluciones no dominadas, comparado al hecho de encontrar una única solución. Por ejemplo, si la contaminación en el medio ambiente alcanza un nivel alto o si estas se ven reguladas repentinamente por el gobierno, el tomador de decisiones puede considerar las emisiones máximas de CO_2 que podrían producirse pero teniendo en cuenta la ganancia. En este caso, ya se conoce la solución factible que el tomador de decisiones puede implementar para superar este problema, la cual es dada por el algoritmo NBOTS. Por otro lado, si el aire está en un mejor punto de calidad, se podría implementar una solución que brinde más ganancias aunque se contamine más. En consecuencia, este algoritmo provee un amplio rango de opciones que brinda una compensación entre las ganancias y las emisiones de CO_2 en función de las condiciones reales del medio ambiente, facilitando el proceso al tomador de decisiones.

6.1 Futuras direcciones de investigación

Vale la pena recordar que en cada movimiento de la búsqueda local se resuelven dos problemas distintos mediante el uso de un optimizador, es decir, el problema de ruteo y el problema del nivel inferior. Por lo tanto, para hacer una reducción significativa en el esfuerzo computacional, se podría proponer una búsqueda tabú controlada hibridizada con un procedimiento de reencadenamiento de trayectorias (path-relinking). Esta nueva dirección de investigación tiene como objetivo obtener soluciones no dominadas en ambos extremos de la frontera de Pareto mediante el uso de la búsqueda tabú, y utilizar el procedimiento de reencadenamiento de trayectorias para obtener los puntos no dominados restantes a partir de los extremos de la frontera. Esta hibridización ha logrado una disminución significativa del tiempo de cómputo y, a veces, una mejor aproximación de la frontera de Pareto (ver [Barbalho et al., 2013]).

Otra idea para ampliar esta investigación es aplicar un enfoque microscópico para calcular las emisiones de CO_2 , el cual aportaría un cálculo más fiel de las mismas. Por ejemplo, se podrían considerar las ideas de [Bektaş and Laporte, 2011], en las que se tiene en cuenta la carga en el vehículo y la velocidad a la que viaja el vehículo en cada arco de la ruta. Ese enfoque conduce a un cálculo de emisiones más confiable. La complejidad de aplicar este enfoque microscópico se basa en la incorporación de nuevas variables que serán las encargadas de medir la carga de cada arco activo. De la misma forma nos enfrentamos a ser capaces de encontrar y establecer la relación existente entre la velocidad y la producción de emisiones de CO_2 e incorporarlo a nuestro modelo. Esto proveería al tomador de decisiones de datos más certeros y con el impacto más realista posible.

Sin duda alguna existen propuestas de mejora muy buenas para nuestro problema, el cual fue diseñado con la idea de crear un modelo general capaz de adaptarse a cambios y nuevos parámetros. De la misma forma, el diseño del algoritmo propuesto nos presenta la posibilidad de incorporar estrategias de intensificación como lo es la incorporación del reencadenamiento de trayectorias a la búsqueda de soluciones. Ciertamente, es un problema relevante que día

a día genera mayor interés en la industria, el gobierno y la academia, el cual brinda la satisfacción de ser conscientes del daño que generamos en nuestro entorno y busca ser capaces de brindar una alternativa de solución que sea más conveniente.

Bibliografía

- [Aldakhil et al., 2018] Aldakhil, A. M., Nassani, A. A., Awan, U., Abro, M. M. Q., and Zaman, K. (2018). Determinants of green logistics in brics countries: An integrated supply chain model for green business. *Journal of Cleaner Production*, 195:861–868.
- [Anderluh et al., 2019] Anderluh, A., Nolz, P. C., Hemmelmayr, V. C., and Crainic, T. G. (2019). Multi objective optimization of a two echelon vehicle routing problem with vehicle synchronization and ‘grey zone’ customers arising in urban logistics. *European Journal of Operational Research*.
- [Çağrı Koç et al., 2014] Çağrı Koç, Bektaş, T., Jabali, O., and Laporte, G. (2014). The fleet size and mix pollution-routing problem. *Transportation Research Part B: Methodological*, 70:239 – 254.
- [Çağrı Koç and Karaoglan, 2016] Çağrı Koç and Karaoglan, I. (2016). The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing*, 39:154 – 164.
- [Barbalho et al., 2013] Barbalho, H., Rosseti, I., Martins, S. L., and Plastino, A. (2013). A hybrid data mining grasp with path-relinking. *Computers & Operations Research*, 40(12):3159–3173.
- [Bard, 1983] Bard, J. F. (1983). An algorithm for solving the general bilevel programming problem. *Mathematics of Operations Research*, 8(2):260–272.
- [Bard, 1991a] Bard, J. F. (1991a). Some properties of the bilevel programming problem.
- [Bard, 1991b] Bard, J. F. (1991b). Some properties of the bilevel programming problem. *Journal of optimization theory and applications*, 68(2):371–378.
- [Beamon, 1999] Beamon, B. M. (1999). Designing the green supply chain. *Logistics Information Management*, 12(4):332–342.
- [Bektaş and Laporte, 2011] Bektaş, T. and Laporte, G. (2011). The pollution-routing problem. *Transportation Research Part B: Methodological*, 45(8):1232 – 1250. Supply chain disruption and risk management.
- [Calvete et al., 2011] Calvete, H. I., Galé, C., and Oliveros, M.-J. (2011). Bilevel model for production–distribution planning solved by using ant colony optimization. *Computers and Operations Research*, 38(1):320 – 327. Project Management and Scheduling.

- [Camacho-Vallejo et al., 2015] Camacho-Vallejo, J.-F., Muñoz-Sánchez, R., and González-Velarde, J. L. (2015). A heuristic algorithm for a supply chains production-distribution planning. *Computers and Operations Research*, 61:110 – 121.
- [Casas-Ramírez et al., 2018] Casas-Ramírez, M.-S., Camacho-Vallejo, J.-F., and Martínez-Salazar, I.-A. (2018). Approximating solutions to a bilevel capacitated facility location problem with customer’s patronization toward a list of preferences. *Applied Mathematics and Computation*, 319:369–386. Recent Advances in Computing.
- [Cordeau et al., 2002] Cordeau, J.-F., Gendreau, M., Laporte, G., Potvin, J.-Y., and Semet, F. (2002). A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, 53(5):512–522.
- [Dekker et al., 2012] Dekker, R., Bloemhof, J., and Mallidis, I. (2012). Operations research for green logistics—an overview of aspects, issues, contributions and challenges. *European Journal of Operational Research*, 219(3):671–679.
- [Demir et al., 2012] Demir, E., Bektaş, T., and Laporte, G. (2012). An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, 223(2):346 – 359.
- [Demir et al., 2014] Demir, E., Bektaş, T., and Laporte, G. (2014). The bi-objective pollution-routing problem. *European Journal of Operational Research*, 232(3):464 – 478.
- [Diabat and Simchi-Levi, 2009] Diabat, A. and Simchi-Levi, D. (2009). A carbon-capped supply chain network problem. In *2009 IEEE International Conference on Industrial Engineering and Engineering Management*, pages 523–527. IEEE.
- [Diario Oficial de la Federación, 1971] Diario Oficial de la Federación (1971). Ley federal para prevenir y controlar el medio ambiente.
- [Diario Oficial de la Federación, 1988a] Diario Oficial de la Federación (1988a). Ley general del equilibrio ecológico y la protección del medio ambiente. DOF 13-05-2016.
- [Diario Oficial de la Federación, 1988b] Diario Oficial de la Federación (1988b). Reglamento de la ley general del equilibrio ecológico y la protección al ambiente en materia de prevención y control de la contaminación de la atmósfera. DOF 31-10-2014.
- [Diario Oficial de la Federación, 1993] Diario Oficial de la Federación (1993). Norma oficial mexicana nom-043-semarnat-1993.
- [Diario Oficial de la Federación, 1999a] Diario Oficial de la Federación (1999a). Norma oficial mexicana nom-041-ecol-1999.
- [Diario Oficial de la Federación, 1999b] Diario Oficial de la Federación (1999b). Norma oficial mexicana nom-042-ecol-1999.
- [Erdoğan and Miller-Hooks, 2012] Erdoğan, S. and Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):100–114.

- [Eskandarpour et al., 2019] Eskandarpour, M., Ouelhadj, D., Hatami, S., Juan, A. A., and Khosravi, B. (2019). Enhanced multi directional local search for the biobjective heterogeneous vehicle routing problem with multiple driving ranges. *European Journal of Operational Research*, 277(2):479 – 491.
- [Faulin et al., 2019] Faulin, J., Grasman, S., Juan, A., and Hirsch, P. (2019). *Sustainable Transportation and Smart Logistics: Decision Making Models and Solutions*.
- [Figliozzi et al., 2020] Figliozzi, M., Saenz, J., and Faulin, J. (2020). Minimization of urban freight distribution lifecycle co2e emissions: Results from an optimization model and a real-world case study. *Transport Policy*, 86:60 – 68.
- [Forster et al., 2007] Forster, P., Ramaswamy, V., Artaxo, P., Bernsten, T., Betts, R., Fahey, D. W., Haywood, J., Lean, J., Lowe, D. C., Myhre, G., et al. (2007). Changes in atmospheric constituents and in radiative forcing. chapter 2. In *Climate Change 2007. The Physical Science Basis*.
- [Franceschetti et al., 2017] Franceschetti, A., Demir, E., Honhon, D., Van Woensel, T., Laporte, G., and Stobbe, M. (2017). A metaheuristic for the time-dependent pollution-routing problem. *European Journal of Operational Research*, 259(3):972 – 991.
- [Franceschetti et al., 2013] Franceschetti, A., Honhon, D., Woensel, T. V., Bektaş, T., and Laporte, G. (2013). The time-dependent pollution-routing problem. *Transportation Research Part B: Methodological*, 56:265 – 293.
- [Glover, 1986] Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5):533–549.
- [Harris et al., 2009] Harris, I., Mumford, C., and Naim, M. (2009). The multi-objective uncapacitated facility location problem for green logistics. In *2009 IEEE Congress on Evolutionary Computation*, pages 2732–2739. IEEE.
- [Harris et al., 2014] Harris, I., Mumford, C. L., and Naim, M. M. (2014). A hybrid multi-objective approach to capacitated facility location with flexible store allocation for green logistics modeling. *Transportation Research Part E: Logistics and Transportation Review*, 66:1–22.
- [Hill et al., 2013] Hill, N., Venfield, H., Dun, C., and James, K. (2013). Government ghg conversion factors for company reporting: methodology paper for emission factors. *DEFRA and DECC*.
- [Hızır, 2006] Hızır, A. E. (2006). *Using emission functions in mathematical programming models for sustainable urban transportation: An application in bilevel optimization*. PhD thesis.
- [Huang and Liu, 2004] Huang, B. and Liu, N. (2004). Bilevel programming approach to optimizing a logistic distribution network with balancing requirements. *Transportation Research Record*, 1894(1):188–197.

- [Kalashnikov et al., 2015] Kalashnikov, V. V., Dempe, S., Pérez-Valdés, G. A., Kalashnykova, N. I., and Camacho-Vallejo, J.-F. (2015). Bilevel programming and applications. *Mathematical Problems in Engineering*, 2015.
- [Kannan et al., 2013] Kannan, D., Khodaverdi, R., Olfat, L., Jafarian, A., and Diabat, A. (2013). Integrated fuzzy multi criteria decision making method and multi-objective programming approach for supplier selection and order allocation in a green supply chain. *Journal of Cleaner Production*, 47:355–367.
- [Kulturel-Konak et al., 2006] Kulturel-Konak, S., Smith, A. E., and Norman, B. A. (2006). Multi-objective tabu search using a multinomial probability mass function. *European Journal of Operational Research*, 169(3):918–931.
- [Lai and Wong, 2012] Lai, K.-h. and Wong, C. W. (2012). Green logistics management and performance: Some empirical evidence from chinese manufacturing exporters. *Omega*, 40(3):267–282.
- [Leggieri and Haouari, 2017] Leggieri, V. and Haouari, M. (2017). A practical solution approach for the green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 104:97 – 112.
- [Li et al., 2008] Li, F., Liu, T., Zhang, H., Cao, R., Ding, W., and Fasano, J. P. (2008). Distribution center location for green supply chain. In *2008 IEEE International Conference on Service Operations and Logistics, and Informatics*, volume 2, pages 2951–2956. IEEE.
- [Li et al., 2012] Li, X., Leung, S. C., and Tian, P. (2012). A multistart adaptive memory-based tabu search algorithm for the heterogeneous fixed fleet open vehicle routing problem. *Expert Systems with Applications*, 39(1):365 – 374.
- [Li et al., 2019] Li, Y., Soleimani, H., and Zohal, M. (2019). An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives. *Journal of Cleaner Production*, 227:1161 – 1172.
- [Lin et al., 2014] Lin, C., Choy, K., Ho, G., Chung, S., and Lam, H. (2014). Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications*, 41(4, Part 1):1118 – 1138.
- [Linton et al., 2007] Linton, J. D., Klassen, R., and Jayaraman, V. (2007). Sustainable supply chains: An introduction. *Journal of Operations Management*, 25(6):1075–1082.
- [Liu et al., 2018] Liu, J., Yuan, C., Hafeez, M., and Yuan, Q. (2018). The relationship between environment and logistics performance: evidence from asian countries. *Journal of Cleaner Production*, 204:282–291.
- [Maldonado-Pinto et al., 2016] Maldonado-Pinto, S., Casas-Ramírez, M.-S., and Camacho-Vallejo, J.-F. (2016). Analyzing the performance of a hybrid heuristic for solving a bilevel location problem under different approaches to tackle the lower level. *Mathematical Problems in Engineering*, 2016:1–10.

- [Marinakis and Marinaki, 2008a] Marinakis, Y. and Marinaki, M. (2008a). A bilevel genetic algorithm for a real life location routing problem. *International Journal of Logistics: Research and Applications*, 11(1):49–65.
- [Marinakis and Marinaki, 2008b] Marinakis, Y. and Marinaki, M. (2008b). A bilevel genetic algorithm for a real life location routing problem. *International Journal of Logistics Research and Applications*, 11(1):49–65.
- [Marinakis et al., 2007] Marinakis, Y., Migdalas, A., and Pardalos, P. (2007). A new bilevel formulation for the vehicle routing problem and a solution method using a genetic algorithm. *Journal of Global Optimization*, 38:555–580.
- [Mathew and Sharma, 2006] Mathew, T. V. and Sharma, S. (2006). Continuous network design with emission pricing as a bi-level optimization problem. In *Applications of Advanced Technology in Transportation*, pages 804–809.
- [Mauttone et al., 2008] Mauttone, A., Labbé, M., and Figueiredo, R. (2008). A Tabu Search approach to solve a network design problem with user-optimal flows. In *ALIO/EURO Workshop on Applied Combinatorial Optimization*, Buenos Aires, Argentina.
- [Montoya et al., 2016] Montoya, A., Guéret, C., Mendoza, J. E., and Villegas, J. G. (2016). A multi-space sampling heuristic for the green vehicle routing problem. *Transportation Research Part C: Emerging Technologies*, 70:113 – 128.
- [Nourifar et al., 2020] Nourifar, R., Mahdavi, I., Mahdavi-Amiri, N., and Paydar, M. M. (2020). Mathematical modelling of a decentralized multi-echelon supply chain network considering service level under uncertainty. *Scientia Iranica*, 27(3):1634–1654.
- [Potvin and Naud, 2011] Potvin, J.-Y. and Naud, M.-A. (2011). Tabu search with ejection chains for the vehicle routing problem with private fleet and common carrier. *Journal of the Operational Research Society*, 62(2):326–336.
- [Pérez-Bernabeu et al., 2015] Pérez-Bernabeu, E., Juan, A. A., Faulin, J., and Barrios, B. B. (2015). Horizontal cooperation in road transportation: a case illustrating savings in distances and greenhouse gas emissions. *International Transactions in Operational Research*, 22(3):585–606.
- [Qian and Eglese, 2016] Qian, J. and Eglese, R. (2016). Fuel emissions optimization in vehicle routing problems with time-varying speeds. *European Journal of Operational Research*, 248(3):840 – 848.
- [Renaud et al., 1996] Renaud, J., Laporte, G., and Boctor, F. F. (1996). A tabu search heuristic for the multi-depot vehicle routing problem. *Computers & Operations Research*, 23(3):229 – 235.
- [Réveillac, 2015] Réveillac, J.-M. (2015). 1 - operational research. pages 1 – 9.
- [Safaei et al., 2018] Safaei, A. S., Farsad, S., and Paydar, M. M. (2018). Robust bi-level optimization of relief logistics operations. *Applied Mathematical Modelling*, 56:359–380.

- [Sawik et al., 2017a] Sawik, B., Faulin, J., and Pérez-Bernabeu, E. (2017a). Multi-criteria optimization for fleet size with environmental aspects. *Transportation Research Procedia*, 27:61 – 68. 20th EURO Working Group on Transportation Meeting, EWGT 2017, 4-6 September 2017, Budapest, Hungary.
- [Sawik et al., 2017b] Sawik, B., Faulin, J., and Pérez-Bernabeu, E. (2017b). A multicriteria analysis for the green vrp: A case discussion for the distribution problem of a spanish retailer. *Transportation Research Procedia*, 22:305 – 313.
- [Sbihi and Eglese, 2010] Sbihi, A. and Eglese, R. W. (2010). Combinatorial optimization and green logistics. *Annals of Operations Research*, 175(1):159–175.
- [Serrano-Hernández et al., 2017] Serrano-Hernández, A., Angel A., J., Javier, F., and Elena, P.-B. (2017). Horizontal collaboration in freight transport: concepts, benefits and environmental challenges. *SORT-Statistics and Operations Research Transactions*, 1:393–414.
- [Shen et al., 2009] Shen, Z., Ordóñez, F., and Dessouky, M. M. (2009). *The Stochastic Vehicle Routing Problem for Minimum Unmet Demand*, pages 349–371. Springer US, Boston, MA.
- [Sinha et al., 2016] Sinha, A., Malo, P., and Deb, K. (2016). Solving optimistic bilevel programs by iteratively approximating lower level optimal value function. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 1877–1884. IEEE.
- [Stackelberg, 2011] Stackelberg, H. V. (2011). *Market Structure and Equilibrium*. Springer-Verlag Berlin Heidelberg.
- [Sun et al., 2008] Sun, H., Gao, Z., and Wu, J. (2008). A bi-level programming model and solution algorithm for the location of logistics distribution centers. *Applied mathematical modelling*, 32(4):610–616.
- [Talbi, 2013] Talbi, E.-G. (2013). *Metaheuristics for Bi-level Optimization*, volume 482.
- [Ubeda et al., 2011] Ubeda, S., Arcelus, F. J., and Faulin, J. (2011). Green logistics at eroski: A case study. *International Journal of Production Economics*, 131(1):44–51.
- [Wang et al., 2011] Wang, G.-M., Ma, L.-M., and Li, L.-L. (2011). An application of bilevel programming problem in optimal pollution emission price. *Journal of Service Science and Management*, 4(03):334.
- [Wang et al., 2014] Wang, J., Chi, L., Hu, X., and Zhou, H. (2014). Urban traffic congestion pricing model with the consideration of carbon emissions cost. *Sustainability*, 6(2):676–691.
- [Wen and Eglese, 2016] Wen, L. and Eglese, R. (2016). Minimizing co2e emissions by setting a road toll. *Transportation Research Part D: Transport and Environment*, 44:1 – 13.
- [Xiao and Konak, 2015] Xiao, Y. and Konak, A. (2015). A simulating annealing algorithm to solve the green vehicle routing and scheduling problem with hierarchical objectives and weighted tardiness. *Applied Soft Computing*, 34:372 – 388.

- [Yeh and Chuang, 2011] Yeh, W.-C. and Chuang, M.-C. (2011). Using multi-objective genetic algorithm for partner selection in green supply chain problems. *Expert Systems with Applications*, 38(4):4244–4253.
- [Zaman and Shamsuddin, 2017] Zaman, K. and Shamsuddin, S. (2017). Green logistics and national scale economic indicators: Evidence from a panel of selected european countries. *Journal of Cleaner Production*, 143:51–63.
- [Zhang and Lu, 2007] Zhang, G. and Lu, J. (2007). Model and approach of fuzzy bilevel decision making for logistics planning problem. *Journal of Enterprise Information Management*.
- [Zhang et al., 2018] Zhang, S., Gajpal, Y., and Appadoo, S. (2018). A meta-heuristic for capacitated green vehicle routing problem. *Annals of Operations Research*, 269(1-2):753–771.