Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Subdirección de Estudios de Posgrado

modelos de optimización para el problema de inspección de rutas de camiones interurbanos

por

Evely Gutiérrez Noda

como requisito parcial para obtener el grado de

maestria en ciencias en ingeniería de Sistemas

Noviembre 2019

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Subdirección de Estudios de Posgrado

MODELOS DE OPTIMIZACIÓN PARA EL PROBLEMA DE INSPECCIÓN DE RUTAS DE CAMIONES INTERURBANOS

por

Evely Gutiérrez Noda

como requisito parcial para obtener el grado de

MAESTRIA EN CIENCIAS EN INGENIERÍA DE SISTEMAS

Noviembre 2019

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Subdirección de Estudios de Posgrado

Los miembros del Comité de Tesis recomendamos que la Tesis «MODELOS DE OPTIMIZACIÓN PARA EL PROBLEMA DE INSPECCIÓN DE RUTAS DE CAMIONES INTERURBANOS», realizada por el alumno Evely Gutiérrez Noda, con número de matrícula 1935050, sea aceptada para su defensa como requisito parcial para obtener el grado de Maestria en Ciencias en Ingeniería de Sistemas.

El Comité de Tesis

Dr. Fernando Lopez Irarragorri

Asesor

Dra. Maria De Los A. Báez Olvera

Revisor

Dr. Igor Semionovich Litvinchev

Revisor

Vo. Bo.

Dr. Simón Martínez Martínez

Subdirector de Estudios de Posgrado

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO

FIME

San Nicolás de los Garza, Nuevo León, agosto 2019

Dedico mi tesis a mi esposo Adalberto,
a mis hijos Abel y Erick y a mi mamá,
sin ellos no hubiese sido posible.

Índice general

Índice de figuras

Índice de tablas

Agradecimientos

Es un honor para mí agradecer a las personas e instituciones que jugaron un papel importante durante el tiempo que estuve estudiando la maestría, ya que fueron varios los obstáculos que tuve que superar, agradezco a:

En primer lugar, agradezco a mi esposo Adalberto, que sin él no hubiese sido posible pasar todo el proceso de selección para entrar en la maestría, además ha sido mi apoyo a lo largo de mis estudios, siempre dándome ánimos y repitiéndome que todo saldría bien en los momentos de miedo y desesperación.

A mis padres y familiares, por su apoyo, confianza y consejos. En especial a mi mamá, que gracias a ella pude continuar estudiando cuando nació mi segundo hijo Erick, ya que siempre los cuidó a él y a mi otro hijo Abel para que yo pudiera asistir a la universidad.

A mis hijos Abel y Erick que son la razón principal de superarme y ser mejor profesional cada día, para que puedan tener una educación y bienestar como se merecen.

A mi hermano Luis Gutiérrez porque gracias a él se pudo completar todo el trámite para poder ingresar en esta maestría y llegar a México.

Quisiera expresar mi profundo agradecimiento a Fernando López, mi asesor del trabajo, por la oportunidad que me ha brindado, por confiar en mí y por permitirme realizar y compartir este importante trabajo.

Y por último y no por ello menos importante, a mi comité de tesis, a mis profesores, a la facultad y a CONACyT por darme el beneficio de una beca que me permitió estudiar en este país.

Resumen

Evely Gutiérrez Noda.

Candidato para obtener el grado de Maestria en Ciencias en Ingeniería de Sistemas.

Universidad Autónoma de Nuevo León.

Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio: MODELOS DE OPTIMIZACIÓN PARA EL PROBLEMA DE INSPECCIÓN DE RUTAS DE CAMIONES INTERURBANOS.

Número de páginas: 76.

Objetivos y método de estudio: El objetivo del presente trabajo de tesis consiste en realizar una aproximación metodológica al problema de inspección de rutas interurbanas desde la perspectiva de investigación de operaciones. Otros objetivos son el diseño de una metodología de apoyo a la decisión para la planeación de las inspecciones a las rutas de trasporte interurbano en base a la criticalidad establecida por la empresa. El desarrollo de un modelo matemático para resolver instancias de tamaño real del problema de inspección de rutas interurbanas. Un ultimo objetivo es desarrollar una heurística eficiente que encuentre una planificación factible de recorrido de inspectores, en tiempos de cómputo aceptables en instancias reales y que garantice cubrir las rutas de mayor criticidad.

CONTRIBUCIONES Y CONCLUSIONES: La contribución científica de esta tesis es realizar un modelo matemático que sea eficiente en problemas de tamaño real. Se realiza una metodología de apoyo a la decisión desde una perspectiva a resultados, teniendo en cuenta las necesidades del problema. Se implementa una heurística que da solución a instancias de tamaño real. Se realizó una metodología que da solución a este problema, donde se tienen en cuenta las fases del proceso de toma de decisiones. En conclusión, esta metodología permite desarrollar un comportamiento racional en el proceso de toma de decisiones, por lo tanto no debe interpretarse como algo rígido sino más bien como un marco de trabajo donde se expone como llevar a cabo el apoyo a la decisión. A su ves esta metodología brinda un apoyo en la toma de decisiones para la planeación de las inspecciones a las rutas de trasporte interurbano en base a la criticidad establecida por la empresa, concluyendo el primer de los objetivos específicos trazados en este documento.

El desarrollo del modelo matemático para resolver instancias de tamaño real del problema de inspección de rutas interurbanas permite dar cumplimiento a uno de los objetivos específicos de esta tesis, concluyendo que este modelo permite desarrollar metodologías de apoyo a la decisión efectivas para la construcción de la planificación de las inspecciones de autobuses en una empresa de transporte interurbanos.

Se realizaron experimentos científicos que han mostrado que el modelo propuesto puede ser resuelto de manera eficiente para problemas reales en empresas medianas o pequeñas (con un máximo de rutas). Se desarrollo una heurística eficiente que logra encontrar resultados factibles en tiempos de computo aceptables y garantizando cubrir las rutas de mayor criticidad en diferentes tipos de instancias generadas aleatoriamente.

Firma del asesor: ______________________

Dr. Fernando López Irarragorri

Capítulo 1

Introducción

Este capítulo describe el diseño de la investigación que sigue este trabajo de tesis. El problema científico se basa en optimizar la cobertura de tramos críticos por un conjunto dado de inspectores, partiendo de una zona dada con un conjunto de rutas y salidas programas durante un periodo de tiempo. Se analiza la criticidad por tramos y esta es establecida por las empresas.

La estructura de este capítulo es la siguiente: en la Sección 1.1 se analiza brevemente el proceso de planeación de inspección de rutas interurbanas, así como la descripción breve de algunos trabajos anteriores sobre este proceso. La Sección 1.2 muestra algunos antecedentes del problema. A continuación, en la Sección 1.3 se describe de manera general el problema que se aborda en esta investigación. En la Sección 1.4 se presenta la justificación de este problema. En las Secciones 1.5 y 1.6 se describen: el objetivo general y los objetivos específicos que rigen esta investigación.

La novedad científica se presenta en la Sección 1.7. Por último, en la Sección 1.8 se describe el contenido de cada uno de los capítulos que conforman este documento.

1.1 Contexto

En las últimas décadas se ha notado un incremento de trabajos de optimización basados en técnicas de investigación de operaciones o programación matemática, con objetivos basados en garantizar sistemas de distribución para el manejo efectivo de la provisión de bienes o servicios. Estos trabajos buscan ahorros en los costos de transportación, teniendo en cuenta todas las características de los problemas de distribución en el mundo real.

El transporte público interurbano representa uno de los factores más importantes para el éxito de una región, afectando directamente la calidad de vida de sus habitantes y visitantes. Por distintas razones, muchas personas viajan todos los días de una ciudad a otra teniendo que enfrentar problemas como la congestión del tránsito, la escasez de lugares para estacionar, el estado de los autobuses o el incumplimiento de los horarios del servicio público y normas establecidas para este servicio [Gutiérrez, 2012, Le Grand, 2018].

Diariamente se registran distintas reclamaciones formuladas por los usuarios de líneas regulares de transporte de personal por carretera, por lo cual se hace conveniente intensificar las inspecciones sobre las mismas a fin de que cumplan las condiciones establecidas por las distintas empresas, especialmente las referidas a precio, prohibiciones de tráfico, trasbordos injustificados, etc.

A pesar de que es similar al transporte público urbano, el transporte público interurbano posee algunas diferencias importantes que requieren un tratamiento especial: a) por lo general tiene menos paradas y están más alejadas entre ellas, b) existe menos interacción entre vehículos, provocando un menor número de detenciones no planificadas, c) existen tramos de ruta con características especiales (los pasajeros compran el boleto en efectivo dentro del autobús)[Rodriguez et al., 2012, Black, 2018].

A nivel mundial, el interés de muchos trabajos va dirigido a la ejecución de un plan de mantenimiento preventivo de una flota de transporte, partiendo de que esta es una actividad programada de inspecciones al transporte, tanto de funcionamiento como de seguridad, que debe llevarse a cabo en forma periódica en base a un plan establecido. Todo esto logra disminuir problemas de servicio imprevistos, que constituyen una de las quejas más expuestas por los usuarios que utilizan este medio de transporte [Bauset et al., 2002, Montoro et al., 2018].

En Navarra, España, se establece que se perseguirá la eficacia de la función inspectora mediante la elaboración de planes de inspección con carácter sistemático. Plantean velar por el cumplimiento de la normativa que ordena la actividad del transporte por carretera, con el fin de comprobar el cumplimiento de las condiciones del servicio. Por parte de la Inspección de Transportes, tramitan y analizan todas las reclamaciones y quejas que puedan presentar los usuarios de los servicios regulares de transporte de viajeros por carretera [White, 2016].

Por otra parte, las empresas chilenas pueden optimizar su gestión para mejorar la calidad de su producción mediante la realización de planes de mantención e inspección preventiva al transporte. El objetivo que persiguen es minimizar la cantidad de trabajos hechos ante una emergencia, maximizando la cantidad de trabajos planificados y programados en un periodo de tiempo de acuerdo a los recursos disponibles. Para cumplir este objetivo realizan una planeación de inspección diaria de los camiones y realizan algunos servicios de rutina, incluyendo la asignación de tiempo calendario y personal para la ejecución de las distintas tareas de inspección o mantención de rutina a realizar, estableciendo las fechas de iniciación y termino de estas. [Narváez Fuentes, 2005, White, 2016]

En México, varios proyectos para investigar las principales características de la movilidad de las personas han demostrado que con el solo hecho de conocer la movilidad se puede diseñar una política de transporte que atienda adecuadamente la demanda de traslado de las personas y sus bienes [Izquierdo and María, 2008, Charlton and Vowles, 2008].

Para conocer las principales características de la movilidad interurbana de las personas en México, el IMT realizó en el año de 2007 un conjunto de encuestas en las diez ciudades más grandes del país. Un primer elemento de interés radica en el hecho de que la cantidad de viajes que salen anualmente de cada ciudad se asemeja a la cantidad de viajes que diariamente se realizan dentro de cada una de las ciudades. Estas encuestas permitieron conocer que un gran porcentaje de las personas se desplazan entre las principales ciudades de México [De Hoyos et al., 2010].

1.2 Antecedentes

Los problemas de rutas son problemas de Optimización Combinatoria, que consisten en encontrar la solución óptima entre un número finito o infinito numerable de soluciones. Estos problemas se suelen plantear como la búsqueda de la ruta óptima que atraviesa total o parcialmente las aristas y/o arcos dirigidos de un grafo dado. La existencia de casos reales en los que se pueden aplicar estos problemas, por ejemplo, los repartos de mercancías, recogida de basuras, transporte de pasajeros o la limpieza de las calles, los colocan en un nivel alto de importancia en la actualidad, ya que son muchos estos casos reales que necesitan una solución inmediata y eficaz. [Lüer et al., 2009]

Se debe tener cuidado con no confundir los problemas de ruta con los problemas de caminos, en estos últimos solo interesa escoger el camino óptimo que une dos nodos, sin embargo, los problemas de rutas se interesan por construir un ciclo (que será la ruta) tal que recorra ciertos nodos y/o arcos. [Medina et al., 2011, Patriksson, 2015] De igual modo estos problemas están relacionados y algunas veces podría pasar que se resuelva un problema de caminos para poder resolver un problema de rutas. [Das and Bhattacharyya, 2015]

Una parte de los esfuerzos en el área de transporte interurbano se ha dirigido a la solución del problema de ruteo de vehículos (VRP), el cual busca la determinación de la ruta óptima que permita la minimización de los costos, distancias, tiempos de operación y la optimización de los recursos existentes. Por otra parte, también se ah trabajado mucho en la planeación de inspecciones a rutas o de camiones durante las horas de trabajo, ya sea inspecciones de cumplimiento de servicio, de limpieza, de mecánica entre otras.

Algunos trabajos realizados sobre el tema del transporte público interurbano dan respuestas más eficientes al problema de la congestión. Mediante el uso intensivo de sistemas informáticos y de las telecomunicaciones aplicadas a la gestión del tráfico con los denominados Sistemas Inteligentes de Transporte (SIT). [Das and Bhattacharyya, 2015] Estos sistemas brindan mejoras en problemas de incumplimiento de horarios, en el incremento del tiempo de los viajes en transporte público y privado, en la contaminación del aire y niveles sonoros intolerables que llegan a afectar seriamente la salud. Todo lo anterior beneficia el bienestar de la población y tiene su correlación con importantes pérdidas económicas.

El problema que se tiene en este caso involucra un conjunto de inspectores que tienen que inspeccionar las rutas o tramos, teniendo en cuenta los factores críticos establecidos por la empresa que generalmente se expresa como un índice cualitativo. Este problema se caracteriza según la criticidad de los tramos que se obtiene mas adelante, los horizontes de tiempos de trabajo o tiempos de almuerzo de los inspectores, igual nodo de comienzo y fin de las inspecciones, asignación de tiempos de salidas. Anteriormente, en trabajos previos, no se aborda el problema de inspección en los autobuses interurbanos y en casos similares que presentan este problema no se encuentra automatizado.

1.3 Descripción del Problema Científico

Las empresas de transporte interurbano actualmente necesitan una planificación adecuada y eficaz para entregar a sus inspectores de autobuses, que garantice que se inspeccione la mayor cantidad de rutas o tramos de rutas, teniendo en cuenta el nivel de criticalidad de los tramos o las rutas establecido por la empresa. Para lograr esto se tienen en cuenta los siguientes aspectos:

- Tienen un horario de trabajo de X horas(varia entre 8 y 12).
- Las rutas inspeccionadas por cada inspector pueden ser múltiples, es decir, cada inspector inspecciona varias rutas o tramos durante su jornada de trabajo, buscando repetir lo menos posible las rutas o tramos ya inspeccionadas.
- Cada inspector puede inspeccionar solo una tramo a la vez.
- El inspector puede comenzar su trabajo en la parada que desee, pero siempre debe terminar en la misma parada donde comenzó.
- No es necesario inspeccionar todas las rutas y tramos en una jornada laboral.
- Se necesita cubrir la mayor cantidad de tramos o rutas caracterizados como críticos.

Para solucionar este problema, se propone una metodología de apoyo a la decisión para la planeación de las inspecciones a las rutas o tramos de trasporte interurbano, en base a la criticalidad que establece la empresa. Con esto se logra obtener una planificación eficaz de las inspecciones a las rutas, garantizando el cubrimiento de los tramos caracterizados como críticos.

1.4 Justificación

Los avances tecnológicos que actualmente se han presentado en el país y en el mundo, han simplificado el desarrollo de actividades que en el pasado requerían de mayor dinero y tiempo. En vista de ello, la tecnología ha influido en el progreso social y económico de las personas y las empresas que la utilicen, actuando como una herramienta de vital importancia para su crecimiento.

La planeación de las inspecciones a los autobuses permite lograr un proceso eficiente para las empresas, donde se tenga una flota de vehículo idónea y rentable para el trabajo diario. El problema de inspección que se presenta es sobre la mitigación de los faltantes en las entregas de efectivo por las ventas de boletos, para el caso en que los choferes son los que venden y entregan estos boletos en los tramos donde no existe terminal. La investigación de operaciones con la ayuda de herramientas tecnológicas se torna en una opción viable para la reducción de costos, tiempos de planeación y asignación de inspectores. Básicamente los encargados de realizar estas planificaciones lo realizan basándose en su experiencia, realizando estas planeaciones a mano. Las empresas utilizan la criticidad para establecer la prioridad de los tramos que se deben inspeccionar, y esta criticidad repercute en las fases de asignación de determinadas rutas a cada inspector y los horarios de salida de cada uno, ocasionando que se pueda asignar mayor numero de inspectores a rutas no necesarias, o de igual modo redundancia en los recorridos planificados para los inspectores, implicando el no abastecimiento de las inspecciones de rutas si necesarias y disminución del nivel de servicio que debe brindar el transporte interurbano.

Un panorama similar al descrito anteriormente se presenta en México, donde el transporte público esta a cargo de empresas privadas. Muchas de ellas no cuentan con ningún sistema automático computarizado que determine la frecuencia de salida, las rutas a abordar asignados a los inspectores, etc. En México se ha implementado un mecanismo de inspección para intentar resolver o mitigar este problema: se identifican las salidas más críticos (baja recaudación respecto a los montos históricos) y se asignan inspectores que se suben a los autobuses en estos tramos para verificar la cantidad de pasajeros viajando en el autobús y checando sus boletos. El problema de inspección no ha sido formulado anteriormente en la literatura científica publicada sobre este tema y afines.

Resolviendo el problema de la inspección de rutas en el transporte interurbano se logra brindar a las empresas una planeación eficiente que garantice que se cubran todas o la mayor cantidad de las rutas críticas en tiempos razonables. Esto permite a las empresas contar con un servicio seguro y con la calidad requerida, cumpliendo con calidad los servicios brindados a los pasajeros.

1.5 Objetivo General

El objetivo del presente trabajo de tesis consiste en realizar una aproximación metodológica al problema de inspección de rutas interurbanas desde la perspectiva de investigación de operaciones.

1.5.1 Objetivos Específicos

1. Definir formalmente una metodología de apoyo a la decisión para la planeación de las inspecciones a las rutas de trasporte interurbano en base a la criticalidad establecida por la empresa.

2. Desarrollo de un modelo matemático para resolver instancias de tamaño real del problema de inspección de rutas interurbanas.

3. Desarrollar una heurística eficiente que encuentre una planificación factible de recorrido de inspectores, en tiempos de cómputo aceptables en instancias reales y que maximice la criticidad en tramos inspeccionados al menor costo

de inspección.

1.6 Novedad Científica

La contribución científica que se espera con esta tesis es realizar un modelo matemático que sea eficiente en problemas de tamaño real, con las siguientes características:

- Planteamiento formal del problema.
- Desarrollo de una metodología de apoyo a la decisión.
- Propuesta de un modelo de programación matemática para resolver el problema.
- Diferente distribución de criticidad de las rutas por tramos.
- Nodos de transfer con porcientos elevados de distribución de densidad.
- Número elevado de aristas.
- Horas de salidas de las rutas.

Se realiza una metodología de apoyo a la decisión desde una perspectiva a resultados, teniendo en cuenta las necesidades del problema. Se implementa una heurística que da solución a instancias de tamaño real que se detallan en el Capítulo 3 del documento.

1.7 Estructura de Documento

El presente documento está organizado de la siguiente manera: en el Capítulo 2, se presenta el marco teórico abarcando los temas que están relacionados con el problema. En el Capítulo 3 se plantea detalladamente el problema y se presenta el modelo matemático desarrollado. En el Capítulo 4 se describe la metodología general que se sigue para resolver el problema planteado y en el Capítulo 5 se presentan resultados experimentales. Para concluir el trabajo presentado, en el Capítulo 6 se presentan las conclusiones y recomendaciones para trabajos futuros.

Capítulo 2

Marco Teórico

2.1 Introducción

El capítulo siguiente presenta los fundamentos teóricos bajo los cuales se realizó la investigación de este trabajo, contiene además el estado del arte relacionado con el mismo. El capitulo se estructura comenzando por el Apoyo multicriterio a la decisión en la Sección 2.2, abordando el proceso de toma de decisiones y la clasificación de los problemas de decisión. En la Sección 2.3 se habla de la Programación lineal y se especifica la programación lineal entera y mixta, describiendo algunos algoritmos como el Branch & bound (B & B) y el Simplex, la programación lineal por si sola es otro de los temas tocados en esta sección. El tema de la Sección 2.4 refiere a Problemas de Ruteo por Arcos y contiene los problemas específicos del cartero chino y su variante rural. La Sección 2.5 aborda el tema de la Optimización con Heurísticas, abordando temas de las metaheuristicas y algunos algoritmos generales descritos, por último la sección 2.6 recrea las conclusiones del capítulo.

2.2 Apoyo multicriterio a la decisión

El análisis de la decisión multicriterio (en inglés: Multiple Criteria Decision Analysis (MCDA) es una subdisciplina de la investigación de operaciones que se relaciona estrechamente con las ciencias de la administración. El objeto de estudio del análisis de la decisión multicriterio es el desarrollo de metodologías para el apoyo a la decisión en problemas complejos de decisión bajo múltiples objetivos, criterios o metas y que están en conflicto. El uso de estos métodos se evidencia en situaciones de decisión como: servicios públicos, transporte, logística entre otras [Báez Olvera, 2015].

A lo largo de la historia, hombre ah estado obligado a tener que elegir entre diversidad de alternativas. A causa de esto, cada vez es más indispensable la necesidad de tomar decisiones, pero también cada vez es mayor el riesgo que se asume al decidir qué es lo mejor. Es constante el interés por buscar alternativas que ayuden a decidir, implementando modelos que provoquen la competitividad.

La búsqueda de la eficiencia y la productividad de las empresas, de los sectores industriales y de las regiones contribuyen a la exploración de metodologías de apoyo para la toma de decisiones en escenarios donde intervienen múltiples variables o criterios de selección [Bell, 1985, Arquero et al., 2009]. Lo anterior lleva a reconocer que cada vez es más necesaria la utilización de metodologías que permitan reducir o aliviar el riesgo que suponen las hipótesis y supuestos en la búsqueda de alcanzar mejores niveles de competitividad entre las empresas, los sectores industriales y las regiones.

Los métodos de decisión multicriterio no permiten encontrar una solución óptima, sino compromisos aceptables ante diferentes criterios en conflicto, son una base sustentada en elementos científicos, aportando mejoras distintivas para asumir una decisión. Se trata de decisiones basadas en componentes cuantificables que permiten ponderar el riesgo y, en virtud de ello, son capaces de elegir la decisión que resulta ser la más satisfactoria en el mejor de los casos, y en el peor, la menos insatisfactoria [Elwyn et al., 2017, Bell, 1985].

2.2.1 Proceso de toma de decisiones

El proceso de toma de decisión consiste en elegir alternativas de un conjunto dado para lograr ciertas metas expresadas por el tomador de decisiones y que sean consistentes con su sistema de preferencias. Este proceso de toma de decisiones consiste en tres fases: fase de inteligencia, fase de diseño y la fase de elección [Elwyn et al., 2017, Durán Gisbert and Climent, 2017].

La fase de inteligencia consiste en la identificación y estructuración del problema, definición de los objetivos, recopilación de datos. En la fase de diseño se construye y valida el modelo del problema es decir, se establecen los criterios de evaluación, se buscan soluciones y se prueba la factibilidad de las soluciones. Y finalmente la fase de elección consiste en la búsqueda, evaluación y recomendación de una solución apropiada al modelo [Torres Fernández et al., 2017].

la toma de decisiones se clasifica en dos tipos: toma de decisiones por máquinas y toma de decisiones por humanos. Esta última se realiza desde tres paradigmas: normativo, descriptivo y apoyo a la decisión [Castro, 2007]. La fase normativa incluye enfoques teóricos tales como teoría de la decisión, teoría de la utilidad multiatributo, entre otras. El caso del paradigma descriptivo esta relacionado con la psicología cognitiva, ciencias sociales y conductuales. Por último, el apoyo a la decisión consiste en el conjunto de disciplinas relacionadas para ayudar a las personas a tomar decisiones [Torres Fernández et al., 2017].

La toma de decisiones es el estudio para identificar y elegir alternativas basadas en valores y preferencias del decisor [Gutiérrez, 2009]. La toma de decisiones es el proceso de reducción de la incertidumbre sobre ciertas alternativas para permitir una elección razonable entre ellas. Los procesos involucrados en esto se describen a continuación [Chiavenato and Sapiro, 2017, Torres Fernández et al., 2017]

1. Identificación del decisor y de los interesados.

2. Definición del problema: Esta fase debe como mínimo, identificar las causas del problema y las interfaces con los interesados.

3. Determinar los requerimientos o restricciones: Las restricciones son condiciones que cualquier solución aceptable del problema debe cumplir. Estas restricciones van a determinar el conjunto de posibles soluciones al problema. Se formulan de forma cuantitativa.

4. Establecer las metas: Las metas son los objetivos que se desean alcanzar en la toma de decisiones.

5. Identificar alternativas: Una alternativa es una posibilidad que se puede escoger como solución o no. Tienen que ser identificadas o incluso desarrolladas. El número puede ser finito o infinito.

6. Valoración de alternativas: Definición de criterios o características que cada alternativa debe tener para una mayor o menor valoración de cómo la alternativa alcanza los objetivos definidos.

7. Elegir la herramienta de toma de decisiones: La elección depende del problema concreto de decisión, así como de los objetivos.

8. Evaluar las alternativas frente a los criterios: Los métodos de toma de decisiones necesitan la evaluación de las alternativas frente a los criterios. Dependiendo del criterio la evaluación puede ser: Objetiva o Subjetiva. Una vez evaluadas las alternativas y ordenadas el método escogerá la mejor o la que tenga mayor probabilidad de ser la mejor.

9. Validar las soluciones: Se valida la solución con objetivos, criterios y restricciones.

Aquellos problemas en los que las alternativas de decisión son finitas se denominan problemas de decisión multicriterio discretos. Por otro lado, cuando el problema toma un número infinito de valores y conduce a un número infinito de alternativas posibles, se llama decisión multiobjetivo. Los principales métodos de decisión multicriterio discretos son [Chiavenato and Sapiro, 2017]

- Ponderación lineal (scoring).
- Utilidad multiatributo (MAUT).
- Relaciones de sobreclasificación.
- Análisis jerárquico (AHP).

2.2.2 Clasificación de los problemas de decisión

Los problemas de decisión se clasifican según su problemática, según la naturaleza de las consecuencias asociadas a las alternativas, según la cardinalidad del conjunto de alternativas y según la cantidad de tomadores de decisiones [Roy, 2013, Stiggelbout et al., 2015]. A continuación se explican las características de cada una de las clasificaciones [Báez Olvera, 2015, Chiavenato and Sapiro, 2017].

Según su problemática [Bell, 1985]:

- Problemas de selección (P. α): esta clase de problemas se presenta cuando el tomador de decisiones escoge las mejores alternativas de un conjunto en base a sus preferencias.
- Problemas de ordenamiento (P.β): esta clase de problemas se presenta cuando el tomador de decisiones asigna cada alternativa a una única categoría previamente definida. El tomador de decisiones realiza la asignación tomando en cuenta sus preferencias.

- Problemas de jerarquización (P. γ): esta clase de problemas se presenta cuando el tomador de decisiones ordena, según una relación de preferencia, las alternativas a través de una jerarquización.
- Problemas de descripción (P. δ): esta clase de problemas se presenta cuando el tomador de decisiones desarrollar una descripción de cada alternativa y sus consecuencias en términos apropiados.

Según la naturaleza de las consecuencias asociadas a las alternativas:

- Decisiones bajo certeza: el tomador de decisiones conoce por adelantado las consecuencias de su decisión.
- Decisiones bajo incertidumbre: el tomador de decisiones no conoce de antemano las consecuencias de elegir una alternativa, pero si conoce una distribución de probabilidad asociada a la ocurrencia de las consecuencias asociadas a la alternativa.
- Decisiones bajo estricta incertidumbre: el tomador de decisiones no tiene conocimiento sobre la ocurrencia de las consecuencias asociadas a la alternativa.

Según la cardinalidad del conjunto de alternativas:

- Finito: número pequeño.
- Infinito: número muy grande o infinito contable o infinito incontable.

Según la cantidad de tomadores de decisiones:

- Una sola persona: las decisiones recaen sobre una sola persona.
- Un grupo de personas: las decisiones son tomadas por dos o más personas.

2.3 Programación lineal entera mixta

La programación matemática es una potente técnica de optimización utilizada en el proceso de toma de decisiones de numerosas organizaciones. Como otras ramas de la ciencia y la tecnología, la programación matemática se sirve de modelos para representar aquellos aspectos de la realidad que tienen influencia en su ámbito de interés, en este caso las decisiones que optimizan el funcionamiento de un sistema [Luo et al., 2017].

Una de las características de los modelos de optimización es la existencia de un único decisor frente a otras disciplinas donde puede existir más de un decir (por ejemplo, la teoría de juego) [Crouzeix et al., 2011]. La programación matemática permite identificar, especificar y resolver problemas de optimización de tipo lineal con variables de decisión continuas y discretas.

Según Beneke y Winterboer en el 1984, los métodos matemáticos de optimización (aquellos que permiten identificar los valores máximos o mínimos de determinadas expresiones matemáticas) alcanzaron un desarrollo notable en la década de los años 40. A partir de 1949 aparece un extraordinario número de publicaciones sobre la base teórica de la programación lineal así como de sus aplicaciones a las diversas ramas de la economía [Yin et al., 2017, Nemati et al., 2018].

La programación lineal es un método matemático que permite analizar y elegir la mejor entre muchas alternativas. En términos generales podemos pensar en la programación lienal como un medio para determinar la mejor manera de distribuir una cantidad de recursos limitados en función de lograr un objetivo que radica en maximizar o minimizar una determinada cantidad. El modelo general de un problema de programación lineal consta de dos partes muy importantes: la función objetivo y las restricciones [Hillier et al., 1997].

2.3.1 Programación Lineal

Un problema de optimización matemática, o simplemente un problema de optimización, tiene la forma [Mizuno et al., 1993, Nemati et al., 2018].

$$minimizar f_o(x) \qquad (1.1)$$

$$\text{sujeto a } f_i(x) \leq b_i, i = i, ..., m.$$

El vector $x = (x_1, ..., x_n)$ es la variable de optimización del problema, la función $f_o : R^n \longrightarrow R$ es la función objetivo, las funciones f_i :$R^n \longrightarrow R$, $i = 1, ..., m$, son las funciones de restricción (desigualdad) y las constantes $b1, ...bm$ son los límites, o límites, para las restricciones. Un vector x^* se llama óptimo, o un solución del problema (1.1), si tiene el valor objetivo más pequeño entre todos los vectores que satisfacen las restricciones [Mashayekh et al., 2017]: para cualquier z con $f_1(z) \leq b1, ..., f_m(z) \leq bm$, tenemos $f_o(z) \geq f_o(x^*)$.

El problema anterior se caracteriza por ser un problema de programación lineal si el objetivo y sus funciones de restricción $f_o, ..., f_m$ son lineales, es decir, si satisfacen que [Nemati et al., 2018]:

$$f_i(\alpha x + \beta y) = \alpha f_i(x) + \beta f_i(y)$$

para todo x, $y \in R^n$ y todos $\alpha, \beta \in R$.

La programación lineal puede también aplicarse a los problemas de minimización de costos y estos programas parten de un diferente conjunto de criterios para su optimización [Evans, 2017].

Forma estándar de programación lineal

La forma estándar o canónica del modelo de programación lineal está compuesta por una función objetivo y un conjunto de restricciones [Mashayekh et al., 2017, Bryson, 2018]. En general, la forma estándar del modelo de programación lineal puede expresarse como:

$$Z_{max} = C_1x_1 + C_2X_2 + ... + C_nx_n$$

Sujeto a:

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n \leqslant b_1$$

$$a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n \leqslant b_2$$

$$.$$

$$a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n \leqslant b_m$$

$$x_1, x_2 ... x_n \geqslant 0$$

Y su forma matricial está dada por la expresión:

$$Z_{max} = CX$$

Sujeto a:

$$AX \leqslant B$$

$$X \geqslant 0$$

Donde:

- **C**: Es la matriz de costos o utilidades, formada por los coeficientes de la función objetivo.
- **A**: Es la matriz de coeficientes del sistema formado por las restricciones.

- **B**: Es la matriz columna de términos independientes del sistema de restricciones.
- **X**: Es la matriz columna de las variables $x_1, x_2, x_3...x_n$ del sistema de restricciones.

El Algoritmo Simplex es un método analítico de solución de problemas de este tipo, capaz de resolver modelos más complejos que los resueltos mediante el método gráfico sin restricción en el número de variables.

Algoritmo Simplex

En 1947 el matemático norteamericano Jorge Dantzig desarrolló un algoritmo para resolver problemas de programación lineal de dos o más variables conocido como método Símplex [Bryson, 2018].

Alguno de los algoritmos de solución a problemas de programación lineal es el Simplex. El algoritmo del Simplex busca el óptimo de un problema de programación lineal recorriendo sólo algunos de los vértices del poliedro que representa el conjunto de soluciones factibles [Abdel-Basset et al., 2019]. En cada iteración, el algoritmo se desplaza de un vértice a otro de forma que el valor de la función objetivo mejore con el desplazamiento, de modo que aumente si el problema es de maximización, o disminuya si el problema es de minimización [Hillier et al., 1997].

El método Símplex es otra de las herramientas importantes con que cuenta la investigación de operaciones para apoyar la toma de decisiones cuantitativas, es decir, este método se utiliza para resolver modelos de programación lineal, del mismo modo que el método gráfico, con la ventaja de no tener límite en la cantidad de variables de decisión que se incorporen al modelo. Por lo tanto se pueden manejar n variables y m restricciones, siempre y cuando cumplan con las características de la programación lineal [Bryson, 2018].

El método Simplex es un procedimiento iterativo que permite mejorar la solución de la función objetivo en cada paso. El proceso concluye cuando no es posible continuar mejorando dicho valor, es decir, se ha alcanzado la solución óptima (el mayor o menor valor posible, según el caso, para el que se satisfacen todas las restricciones) [Abdel-Basset et al., 2019].

El método Simplex se basa en la siguiente propiedad: si la función objetivo Z no toma su valor máximo en el vértice A, entonces existe una arista que parte de A y a lo largo de la cual el valor de Z aumenta [Abdel-Basset et al., 2019, Zhang et al., 2016]. La optimización de un problema de programación lineal puede dar lugar a cuatro posibles resultados [Zhang et al., 2016]

- Alcanzar un óptimo único.
- Alcanzar un óptimo que no es único (soluciones alternativas o múltiples).
- Concluir que el problema es no factible, esto es, que no existe ninguna solución que satisfaga simultáneamente todas las restricciones del problema.
- Concluir que el problema es no acotado, es decir, que el valor de la función objetivo en el óptimo es tan grande como se desee si el problema es de maximización, o tan pequeño como se quiera si el problema es de minimización.

El método Simplex alcanza siempre uno de estos resultados en un número finito de iteraciones. En cada iteración se pasa de una solución básica factible a otra, de manera que en el proceso, el valor de la función objetivo mejora en cada iteración. Cuando se determina que no existe ninguna solución buena factible con un mejor valor de la función objetivo que el actual, se detiene el proceso puesto que se ha llegado al óptimo [Hillier et al., 1997, Vanderbei et al., 2015].

Será necesario tener en cuenta que este método únicamente trabaja con restricciones del problema cuyas inecuaciones sean del tipo menor o igual y sus coeficientes independientes sean mayores o iguales a cero. Por tanto se debe estandarizar las restricciones para que cumplan estos requisitos antes de iniciar el algoritmo del Simplex. En caso de que después de éste proceso aparezcan restricciones del tipo mayor o igual, o igualdad, o no se puedan cambiar, será necesario emplear otros métodos de resolución, siendo el más común el método de las Dos Fases [Vanderbei et al., 2015, Zhang et al., 2016].

2.3.2 Programación lineal entera mixta

Un modelo de Programación Entera es aquel cuya solución óptima tiene sentido solamente si una parte o todas las variables de decisión toman valores restringidos a números enteros, permitiendo incorporar en el modelamiento matemático algunos aspectos que quedan fuera del alcance de los modelos de Programación Lineal [Garrido-Jurado et al., 2016, Bermúdez Colina, 2011]. En este sentido los algoritmos de resolución de los modelos de Programación Entera difieren a los utilizados en los modelos de Programación Lineal, destacándose entre ellos el Algoritmo de Ramificación y Acotamiento (o Branch and Bound), Branch and Cut, Planos Cortantes, Relajación Lagrangeana, entre otros.

Los modelos de Programación Entera se pueden clasificar en 2 grandes áreas: Programación Entera Mixta (PEM) y Programación Entera Pura (PEP).

En el caso de la PEP se agrupan aquellos modelos de Programación Entera que consideran exclusivamente variables de decisión que adoptan valores enteros o binarios, de forma exclusiva [Bermúdez Colina, 2011]. Generalmente el conjunto de las soluciones factibles (o dominio de soluciones factibles) es finito [Garrido-Jurado et al., 2016]. Alguna de las aplicaciones de este tipo de problema se encuentran en Problema de Asignación, Problema de Corte de Rollos, Problema de la Mochila, etc.

En el caso de la PEM se agrupan aquellos problemas de optimización que consideran variables de decisión enteras o binarias, pero no de forma exclusiva. Este tipo de problema puede considerarse como un híbrido entre distintas categorías de modelamiento, permitiendo la mezcla de variables enteras y variables continuas

[Demirel et al., 2016]. Algunos ejemplos de estos problemas son los conocidos problemas de Incorporación de Costos Fijos, Problemas de Localización y Transporte, Problema de Generación Eléctrica, etc.

Los modelos de optimización matemática pueden representar de manera exacta los problemas reales, permitiendo de esta manera implementar procedimientos exactos para la programación de la producción, programación de distribución, ruteo de vehículos, localización y distribución de planta, gestión de proyectos, gestión de proveedores, suplir nutrientes a una población con mínimo costo, entre otros. Dentro de los algoritmos de solución a problemas de programación entera mixta se encuentra el Branch & bound (B & B) que se describen más adelante.

Para definir el significado de un modelo matemático de programación lineal entera, se supone que se tiene un problema representado por un modelo lineal

$$max\{cx : Ax \leq b; x \geq 0\}$$

donde A es una matriz de m por n, c es un vector renglón *n-dimensional* y b es un vector columna *m-dimensional* de variables. Ahora se agregan restricciones lineales que obligan que ciertas variables solo tomen valores enteros. Este tipo de problemas son llamados problemas de programación entera[Ávila Torres et al., 2017].

La programación entera mixta (PEM, por sus siglas en ingles) se aplica cuando algunas variables, no todas, son enteras [Demirel et al., 2016].

$$maxcx + hy$$

$$Ax + Gy \leq b$$

$$x \geq 0; y \geq 0 \text{ entera}$$

Ciertos problemas enteros son muy difíciles de resolver computacionalmente, especialmente aquellos problemas combinatorios por ejemplo, el problema del agente viajero, el problema de la mochila, entre otros, la manera más común de resolver este tipo de problemas es mediante Ramificación y Acotamiento (Branch and Bound) donde las relajaciones de programación lineal brindan las cotas para estos problemas [Salas, 2009].

Algoritmo Branch & bound (B & B)

El método de ramificación y acotamiento o Branch & bound (B & B) es uno de los métodos exactos de resolución cuando el numero de máquinas es pequeño, ya que la exposición combinatoria puede llevar a tiempos de resolución inaceptables cuando aumenta la dimensión del problema. Este método B & B es una generalización de la técnica de de la técnica de backtracking backtracking, este consiste en una búsqueda estructurada del espacio de soluciones mediante el cual se divide el mismo de forma iterativa (ramificacion), resultando en problemas de cada ves menor dimensión [Demirel et al., 2016, Vigerske and Gleixner, 2018].

En problemas de maximización se calcula una cota inferior (acotación) para el coste de las soluciones de cada subconjunto. Además, se calcula una cota superior como el valor de la mejor solución admisible encontrada. Luego de cada subdivisión, aquellos subconjuntos con una cota inferior que excede el valor de una solución admisible son excluidos de posteriores particiones (poda). De este modo, es posible excluir de la búsqueda conjuntos de soluciones sin llegar a explorarlas. Se continua subdividiendo hasta que se encuentra una solución admisible cuyo coste no es mayor que la cota superior de cualquier subconjunto. El éxito de la técnica depende del numero de soluciones examinadas antes de alcanzar una solución optima [Demirel et al., 2016, Vigerske and Gleixner, 2018].

Generalmente, se representa el problema del árbol, siendo cada nodo del árbol un subconjunto con mejor dimension que si antecesor, etiquetando los nodos con la cota inferior para todas las ramas o subproblemas que parten de ese nodo [Luna et al., 2016]. Las variantes de B & B consisten en diferentes formas de hacer la subdivisión del espacio o ramificación, diferentes procedimientos para acotar o encontrar una cota inferior del problema y diferentes criterios para descartar o podar conjuntos de soluciones [Vigerske and Gleixner, 2018].

2.4 Metodo de Sumas Ponderadas

El método de las sumas ponderadas hace parte de los métodos clásicos utilizados para el manejo de problemas de optimización multiobjetivo, ya sea de decisiones tomadas de generación o basado en preferencia.

Como su nombre lo indica este método da cierto valor de peso a cada uno de los objetivos de acuerdo a lo investigado o al conocimiento que se tenga del problema. Este método se caracteriza por ser intuitivo, sencillo, todos los criterios son evaluados de acuerdo a su importancia o beneficio, garantizando encontrar solución con un punto optimo.

El método de la suma ponderada a menudo se presenta estrictamente como una herramienta, especialmente en los últimos años, y la literatura que existe con respecto a ejemplos de aplicaciones es extenso. Sin embargo, el foco está en la aplicación, y los problemas tienden a estar limitados a aquellos con solo dos funciones objetivas [Marler and Arora, 2010].

En este método, el objetivo menos importante recibe un peso de uno,y se asignan pesos enteros con incrementos consistentes a objetivos que son más importantes. El mismo enfoque se utiliza con métodos de categorización, en los que diferentes objetivos se agrupan en categorías amplias, como altamente importante y moderadamente importante.

El punto óptimo específico de Paretto que se proporciona como la solución depende de qué pesos se usen, por lo que es importante para determinar cómo se relacionan los pesos con las preferencias, al conjunto óptimo de Paretto y a la función objetivo individual. Una función de preferencia es una función abstracta (de puntos en el espacio de criterios) en la mente del tomador de decisiones, que incorpora perfectamente sus preferencias [Marler and Arora, 2010].

En cuanto a los pesos, el valor de un peso es significativo atento a los valores de otros pesos y relativo al valor de su correspondiente función objetivo. Suponiendo que los pesos son positivos y observando que calcular una suma ponderada siempre proporciona una solución óptima de Paretto, indica que en un punto óptimo de Paretto, el gradientes de las dos funciones objetivo son co-lineales y apuntan en direcciones opuestas. Esencialmente, la combinación lineal de los gradientes es igual a cero.

En el cumplimiento de la definición de la optimización de Paretto, esto sugiere que el movimiento de una solución que satisface, para mejorar una función, es perjudicial para al menos otra función. El método de suma ponderada proporciona un método básico y fácil con enfoque de uso para la optimización de objetivos múltiples y es útil como tal. La solución óptima de Paretto que resulta de un específico conjunto de pesos, depende de las restricciones que forman parte de el conjunto óptimo de Paretto, las relaciones entre los gradientes de las diferentes funciones objetivas, las magnitudes relativas de las funciones objetivas, y la forma de la hiper superficie óptima de Paretto [Marler and Arora, 2010].

2.5 Problemas de Ruteo por Arcos

Existe un área de investigación dentro de la Optimización Combinatoria conocida como Problemas de Rutas, los cuales son una de las partes más importantes de la logística del transporte. [Mauttone et al., 2010] Estos buscan la optimización ya sea en reducción de costos, vehículos o ambas cosas, de un conjunto de rutas a realizar por una flota de vehículos, teniendo en cuenta satisfacer las demandas de recogida, entrega o ambas cosas, de mercancías de algunos clientes geográficamente dispersos.

Estos problemas de rutas actualmente son difíciles de resolver óptimamente aun para tamaños relativamente pequeños, por este motivo la comunidad científica orientada a la resolución de estos problemas ha crecido notablemente utilizando diseños de métodos sofisticados tanto exactos como aproximados. Los problemas de Rutas se pueden clasificar atendiendo al lugar donde se produce la demanda, pueden ser Problemas de Rutas por Vértices o Problemas de Rutas por Arcos. En este epígrafe se abordan los problemas de rutas por arcos específicamente, dadas las características del problema abordado en esta tesis.

El problema de rutas por arcos tiene su origen en el siglo XVIII, cuando los habitantes de un pequeño pueblo de la actual Rusia, comenzaron a debatir si existía alguna ruta que pasase una única vez por los 7 puentes de atravesaba el rio Pregel y volviera al punto de origen [Calvo González, 2018]. Este problema fue propuesto por el matemático suizo llamado Leonhard Euler en su artículo del año 1736 [Calvino Martínez, 2011]. Euler demostró que no era posible puesto que hay puntos en la figura para los cuales el número de líneas que inciden no es par, condición necesaria para entrar y salir del mismo punto de partida por caminos diferentes [Bertero, 2015].

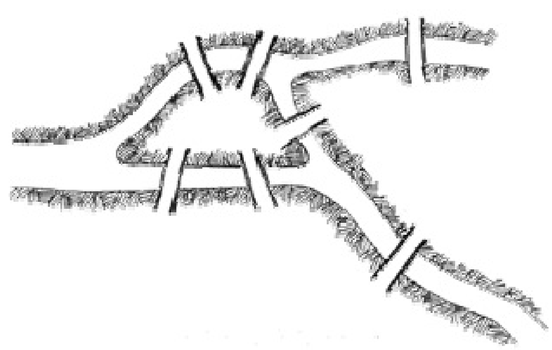

Figura 2.1: Puentes de Konigsberg

Los problemas de rutas por arcos se caracterizan por la necesidad de recorrer todos o parte de los arcos o aristas de un grafo. Uno de los problemas que presentan esta necesidad es el Problema de Cartero Chino (CPP), en el que un cartero debe completar un circuito recorriendo todas las calles de una parte de la cuidad, repartiendo el correo y minimizando la distancia total. El problema del Cartero Chino en cualquiera de sus variantes (Cartero Chino Rural, Cartero Chino Dirigido, Cartero Chino con Ventanas de Tiempo) tiene las características de un problema de rutas por arcos. [Alvarez Nuñez, 2013]

Otro de los grandes problemas con características similares que lo convierten en un problema de rutas por arcos es el Problema de Rutas con Capacidad de Arcos (CARP). En este problema a cada arco del grafo se le asocia una cantidad no negativa que representa la demanda de cada uno de los clientes. [Calvino Martínez, 2011]

Dadas las características de los problemas de rutas por arcos, se pueden mencionar algunos ejemplos típicos que dan gran importancia a estos problemas debido al gran número de situaciones reales en las que se pueden aplicar como, por ejemplo, la recogida de basuras, el transporte escolar, el reparto del correo, recogida de la nieve en las calles, la inspección al transporte urbano e interurbano, entre otros. Mas allá de que algunos casos prácticos requieran servir todos los arcos de la red, la mayoría de las aplicaciones de la vida real son modeladas como un problema del cartero rural.

2.5.1 Problema del Cartero Chino (CPP)

Uno de los problemas de ruteo de arcos es el problema del cartero chino (CPP), también conocido como problema del circuito del cartero, el problema de los correos o problema de la inspección y selección de rutas, es el primer problema de rutas por arcos en el que se plantea la posibilidad de construir un ciclo euleriano con coste óptimo. [Perez, 2014] Este problema fue planteado originalmente por el matemático chino Kwan Mei-Ko (en algunas bibliografías aparece como Guan Mei-Ko) en un artículo de un diario chino en 1960 y traducido al inglés en 1962 (Graphic

Programming using odd and even points) [López et al., 1983].

Este problema recibe el nombre de “Problema del cartero chino” debido al origen del autor. Mei-Ko planteaba el problema al que se enfrenta el cartero para repartir la correspondencia recorriendo la menor distancia posible, que esto matemáticamente consiste en encontrar un tour en el grafo de longitud mínima [Perez, 2014].

El CPP consiste en que un cartero debe completar un circuito recorriendo todas las calles de una parte de la ciudad (que se supone induce un grafo conexo) repartiendo el correo y minimizando la distancia total. [Benavent et al., 1992]

Al enfrentar el problema de optimización de cómo recorrer todas las rutas de manera que las rutas adicionales necesarias para realizar el circuito sumen lo menos posible, se refiere al Problema del Cartero Chino y tendrá diferentes características dependiendo de que se trate de un grafo dirigido, no dirigido o mixto. En todas las variantes se cuenta con una función costo (que puede estar medido en distancia, tiempo o cualquier otro atributo) asociado a cada arco o arista.

El objetivo de este problema consiste en encontrar un circuito de coste mínimo que atraviesa cada arista al menos una vez. Sin embargo, el problema original dió lugar a multitud de variantes, una de ellas es el Problema del Cartero Rural (RPP), en el que solamente un subconjunto de aristas debe ser necesariamente recorrido por el circuito.

2.5.2 Problema del Cartero Chino Rural (RPP)

Una generalización del CPP es el Problema del Cartero Rural (RPP), problema que cuando se define sobre un grafo dirigido es conocido como el Problema del Cartero Rural Dirigido (DRPP) [Bertero, 2015]. Ambos problemas están directamente relacionados con algunos problemas que surgen en la práctica tales como la recogida de basura, reparto de correo, inspección y mantenimiento de sistemas de distribución (conducciones de gas, tendidos eléctricos, redes de ferrocarril, etc.).

En algunos casos no nos interesa atravesar todas las aristas, o arcos, sino solo un subconjunto de ellos. El RPP puede plantearse como un cartero que tiene que repartir el correo en varios pueblos, tiene que recorrer todas las calles de los pueblos, pero no todas las carreteras que los unen. Más allá de que algunos casos prácticos requieren servir todos los arcos de la red, la mayoría de las aplicaciones de la vida real son modeladas como un problema del cartero rural [Benavent et al., 1985].

En general se trata de un problema NP-duro, sin embargo, si es un grafo estrictamente dirigido o no dirigido se puede resolver en tiempo polinómico siempre y cuando el subgrafo formado por las aristas o arcos requeridos sea fuertemente conexo [Perez, 2014, Benavent et al., 1985]. Este problema, al ser definido sobre un grafo dirigido, se conoce como el Problema del cartero rural dirigido (DRPP), así mismo se puede definir el problema sobre grafos no dirigidos o mixtos. A pesar de la similitud en el enunciado de estos problemas, cada uno de ellos presenta características muy específicas que se reflejan en la dificultad y estrategias de resolución.

El problema del cartero rural ha sido objeto de mucho interés por parte de la comunidad de optimización combinatoria, tanto para grafos dirigidos, no dirigidos o grafos mixtos. [Benavent et al., 1985] En general se trata de un problema NP-duro, sin embargo, si es un grafo estrictamente dirigido o no dirigido se puede resolver en tiempo polinómico siempre y cuando el subgrafo formado por las aristas o arcos requeridos sea fuertemente conexo [Calvino Martínez, 2011].

2.6 Optimización con Heurísticas

Los métodos de optimización heurística reciben el nombre de algoritmos heurísticos, metaheurísticos o heurísticos. Este término deriva de la palabra griega heuriskein que significa encontrar o descubrir y se usa en el ámbito de la optimización para describir una clase de algoritmos de resolución de problemas [Cunquero, 2003].

La palabra heurística es definida por el diccionario de la Real Academia Española como ‘Técnicas de la indagación y el descubrimiento’, “Busqueda o investigacion de documentos o fuentes historicas”, “Maneras de buscar la solución a un problema mediante métodos no rigurosos” [González et al., 2001].

La heurística es vista como el arte de inventar por parte de los seres humanos, con la intención de procurar estrategias, métodos, criterios, que permitan resolver problemas a través de la creatividad, pensamiento divergente o lateral [Silva Clavería and Silva Clavería, 2004]. En los tiempos de antes, optimizar significa poco más que mejorar; sin embargo, en el contexto científico la optimización es el proceso de tratar de encontrar la mejor solución posible para un determinado problema. Existe una infinidad de problemas teóricos y prácticos que involucran a la optimización. Los procedimientos heurísticos se dividen en [González et al., 2001]:

- **Principios heurísticos**, son los que establecen sugerencias para encontrar la solución idónea al problema.
- **Reglas heurísticas**, son las que señalan los medios para resolver el problema.
- **Estrategias heurísticas**, son aquellas que permiten organizar los materiales o recursos compilados que contribuyen a la búsqueda de la solución del problema.

Existen diferentes soluciones a un determinado problema de optimización y uno de los criterios para seleccionar una solución es encontrar la mejor. En términos científicos, estos problemas se pueden expresar como encontrar el valor de unas variables de decisión para los que una determinada función objetivo alcanza su valor máximo o mínimo, siendo el valor de las variables en ocasiones sujeto a unas restricciones [Payá Zaforteza, 2009].

Se denomina Métodos Heurísticos al conjunto de métodos y técnicas que se emplean con el fin de encontrar y solucionar un problema en aquellos casos que es difícil hallar una solución óptima o satisfactoria [González et al., 2001]. Se puede encontrar una gran cantidad de problemas de optimización, tanto en la industria como en la ciencia. Desde los clásicos problemas de diseño de redes de telecomunicación u organización de la producción hasta los más actuales en ingeniería de software. Algunas clases de problemas de optimización son relativamente fáciles de resolver [Cunquero, 2003]. Este es el caso, por ejemplo, de los problemas lineales, en los que tanto la función objetivo como las restricciones son expresiones lineales. Estos problemas pueden ser resueltos con el conocido método Simplex; sin embargo, muchos otros tipos de problemas de optimización son muy difíciles de resolver y no se encuentra solución con métodos de solución exacta.

Se puede decir que un problema de optimización difícil es aquel para el que no se puede garantizar el encontrar la mejor solución posible en un tiempo razonable. Una gran cantidad y variedad de problemas difíciles que aparecen en la práctica y que necesitan ser resueltos de forma eficiente son la causa del desarrollo de procedimientos eficientes para encontrar buenas soluciones, aunque no fueran óptimas. Dado esta necesidad surgen las heurísticas, métodos en los que la rapidez del proceso es tan importante como la calidad de la solución obtenida [Olivera, 2004, Cunquero, 2003].

Los problemas combinatorios pueden ser divididos en dos grandes grupos considerando la existencia de algoritmos polinomiales para resolver cada tipo de problema [Isaza et al., 2004]. El primero es el problema tipo P (polinomial) para el cual existen algoritmos con esfuerzos computacionales de tipo polinomial para encontrar la solución óptima. El segundo grupo de problemas combinatorios es el problema tipo NP (no polinomial) para el cual no se conocen algoritmos con esfuerzos computacionales de tipo polinomial para encontrar la solución óptima.

Los métodos exactos que proporcionan una solución óptima del problema, a diferencia de los métodos heurísticos que se limitan a proporcionar una buena solución del problema no necesariamente óptima [Silva Clavería and Silva Clavería, 2004]. Sin embargo, el tiempo invertido por un método exacto para encontrar la solución óptima de un problema difícil, es de un orden de magnitud muy superior al del heurístico, llegando a ser imposible en muchos casos. En otras palabras, los métodos heurísticos proporcionan soluciones rápidas y factibles a problemas complejos.

Uno de los métodos heuristicos mas usados son los llamados heuristicos greedy o voraz. Estos métodos funcionan escogiendo en cada paso al mejor elemento posible, conocido como el elemento más prometedor. Se elimina ese elemento del conjunto de candidatos y, acto seguido, comprueba si la inclusión de este elemento en el conjunto de elementos seleccionados produce una solución factible. En caso de que así sea, se incluye ese elemento en S. Si la inclusión no fuera factible, se descarta el elemento. Iteramos el bucle, comprobando si el conjunto de seleccionados es una solución y, si no es así, pasando al siguiente elemento del conjunto de candidatos [Serna and Uran, 2015].

Algorithm 1 Algoritmo Greedy(C) C: es el conjunto de todos los candidatos

S = vacio S es el conjunto en el que se construye la solución
Mientras
solucion(S) y C diferente de vacio
Hacer
x = el elemento de C que maximiza seleccionar(x)
C = C x
Si $completable(S \cup x)$
$S = S \cup x$
Si solución(S)
Retornar S
Sino no hay solución

Una vez finalizado el bucle, el algoritmo comprueba si el conjunto S es una solución o no, devolviendo el resultado apropiado en cada caso. El algoritmo se muestra a continuación:

En los últimos años ha habido un crecimiento espectacular en el desarrollo de procedimientos heurísticos para resolver problemas de optimización. En 1995 se edita el primer número de la revista Journal of Heuristics dedicada íntegramente a la difusión de los procedimientos heurísticos [Cunquero, 2003]. Existen otras razones para utilizar métodos heurísticos, entre las que se pueden destacar [Zanakis and Evans, 1981]

- La no existencia de ningún método exacto para su solución.
- Aún existiendo un método exacto para resolver el problema, este es computacionalmente muy costoso.
- El método heurístico es más flexible que un método exacto, permitiendo, por ejemplo, la incorporación de condiciones de difícil modelización.
- El método heurístico se utiliza como parte de un procedimiento global que garantiza el óptimo de un problema. Una variante de lo anterior es el cuando el método heurístico proporciona una buena solución inicial de partida. Otra variante es cuando el método heurístico participa en un paso intermedio del procedimiento.

Los algoritmos heurísticos dependen en gran medida del problema concreto para el que se van a diseñar. Las técnicas e ideas aplicadas para la resolución de un problema son específicas de este, pero en algunos casos, pueden ser trasladadas a otros problemas [Silva Clavería and Silva Clavería, 2004].

Los algoritmos heurísticos se clasifican en algoritmos constructivos, algoritmos de descomposición y división, algoritmos de reducción, algoritmos de manipulación del modelo y algoritmos de búsqueda usando vecindad [Gallego et al., 2003, RAMON and ROMERO,].

En general, y sin tener en cuenta un problema específico, una buena heurística debería tener por lo menos las siguientes propiedades o características [González et al., 2001, Zanakis and Evans, 1981]:

1. **Eficiente:** Un esfuerzo computacional realista para obtener la solución.

2. Simple: que facilita la comprensión y aceptación del usuario.

3. Robustez: la heurística debe obtener buenas soluciones, en condiciones razonables.

Para medir la calidad de un heurístico existen diversos procedimientos, entre los que se encuentran los siguientes [Cunquero, 2003]:

1. Comparación con la solución óptima: Esta medida de calidad se puede realizar cuando se disponga de un procedimiento que proporcione el óptimo para un conjunto limitado de ejemplos. Este conjunto de ejemplos puede servir para medir la calidad del método heurístico.

2. Comparación con una cota: En ocasiones el óptimo del problema no está disponible ni siquiera para un conjunto limitado de ejemplos. Este método de evaluación consiste en comparar el valor de la solución que proporciona el heurístico con una cota del problema (inferior si es un problema de minimización y superior si es de maximización).

3. Comparación con un método exacto truncado: Se puede establecer un límite de iteraciones (o de tiempo) máximo de ejecución para el algoritmo exacto. La mejor solución encontrada con procedimientos truncados proporciona una cota con la que comparar el heurístico.

4. Comparación con otros heurísticos: Este es uno de los métodos más empleados en problemas difíciles (NP-duros) sobre los que se ha trabajado durante tiempo y para los que se conocen algunos buenos heurísticos. Al igual que ocurre con la comparación con las cotas, la conclusión de dicha comparación está en función de la bondad del heurístico escogido.

5. Análisis del peor caso: Consiste en analizar el comportamiento en el peor caso del algoritmo heurístico; es decir, considerar los ejemplos que sean más desfavorables para el algoritmo y acotar la máxima desviación respecto del óptimo del problema.

2.6.1 Optimización con Metaheuristicas

Las metaheurísticas pueden surgir como estrategias generales de diseño de procedimientos heurísticos para la resolución de problemas con un alto rendimiento. En la resolución de problemas específicos han surgido procedimientos heurísticos exitosos, de los que se ha tratado de extraer lo que es esencial en su éxito para aplicarlos a otros problemas o a otro contexto mas extenso. De esta forma se han obtenido, tanto técnicas y recursos computacionales específicos, como estrategias de diseños generales para procedimientos heurísticos de resolución de problemas. Estas estrategias generales para construir algoritmos, que quedan por encima de las heurísticas y van algo mas allá, se les denomina metaheurísticas. Las metaheurísticas pueden integrarse como un sistema experto para facilitar su uso genérico a la vez que mejorar su rendimiento [Melián et al., 2003].

Las metaheuristicas se pueden clasificar en metaheurísticas de búsqueda, de relajación, constructivas, evolutivas y otras.

- La metaheurística de *búsqueda* guía los procedimientos que buscan transformaciones o movimientos para recorrer el espacio de soluciones alternativas y explotar las estructuras de entornos asociadas.
- La metaheurística de *relajación* se refiere al procedimiento de resolución de problemas que utilizan relajaciones del modelo general haciéndolo mas fácil de resolver, cuya solución facilita la solución del problema original.
- La metaheurística *constructiva* se orienta a los procedimientos que tratan de la obtención de una solución a partir del análisis y selección paulatina de las componentes que la forman.
- La metaheurística *evolutiva* está enfocada a procedimientos basados en conjuntos de soluciones que evolucionan sobre el espacio de soluciones.

Algunas metaheurísticas surgen combinando metaheurísticas de distintos tipos, ejemplo la GRASP (Greedy Ramdomized Adaptative Search Procedure, por sus siglas en inglés) que combina una fase constructiva con una fase de búsqueda de mejora [RESENDE and RIBEIRO, 2002, Resende and Velarde, 2003]. Otras metaheurísticas se centran en el uso de algun tipo de recurso computacional o forma especial como las redes neuronales, los sistemas de hormigas o la programación por restricciones y no se incluyen claramente en ninguno de los cuatro tipos de clasificaciones anteriormente mencionados [Resende and Velarde, 2003].

Búsqueda local Iterada (ILS)

La búsqueda local iterada (ILS) es una metaheurística general y poderosa que proporciona una forma flexible de mejorar el rendimiento de los algoritmos de búsqueda local. La búsqueda local iterada es una metaheurística simple y de aplicación general que se aplica iterativamente. El procedimiento de la ILS se describe en [Lourenzo, 2001], esta se basa en aplicar la búsqueda local a las modificaciones del punto de búsqueda actual. Estos son mecanismos para generar una solución inicial s_o, un procedimiento de modificación, que modifica la solución actual s llevando a alguna solución intermedia s'. Luego un procedimiento de búsqueda local que lleva a s' a un mínimo local s'' y un criterio de aceptación que decide qué solución será la próxima que se le aplique una modificación [Stützle, 1998].

Algorithm 2 Búsqueda Local Iterada

Se genera una solucion inicial s_o

s = BúsquedaLocal(s_o)

Repetir

s' = Modificacion(s, historico)

s'' = BúsquedaLocal(s')

s = CriterioAceptacion(s, s'', historico)

hasta que se cumpla el criterio de aceptación

End

2.7 Conclusiones del Capítulo 2

A través de la literatura, se pueden encontrar algunos trabajos que se asemejan en algunos aspectos del problema aquí planteado. Los problemas más parecidos son los de ruteo de arcos, que son problemas difíciles. Por esta razón se espera que la solución de este problema para instancias grandes, probablemente deba realizarse por medio de algoritmos heurísticos.

Capítulo 3

Planteamiento del problema y metodología de solución

Este capítulo describe detalladamente el problema que se aborda en este trabajo. Se describe el mismo con cada especificación y características tenidas en cuenta a la hora del análisis y confección del modelo matemático. Se explica el enfoque desde el cual se modelaron las restricciones mas importantes en el modelo matemático así como se explica el modelo matemático desarrollado. Luego se describe la metodología utilizada y cada etapa que se realizo.

El contenido de este capítulo se estructura como sigue: en la Sección 3.1 se describe el problema a abordar, en la Sección 3.2 se presentan los supuestos del modelo, en la sección 3.3 se presenta el modelo matemático y se definen cada una de las restricciones que lo conforman. La sección 3.4 abarca la metodología utilizada para dar solución a este problema así como las etapas que la componen y finalmente en la Sección 3.5 se presentan las conclusiones de este capítulo.

3.1 Descripción del problema

El problema a tratar consiste en planear las inspecciones de un conjunto de inspectores activos que deben hacer en una jornada laboral, especificando que ruta y en que horarios comenzaran, si deben cambiar o no de ruta a lo largo de su inspección y que tramos específicos van a inspeccionar. Cada zona esta compuesta por rutas y cada ruta esta particionada en tramos donde se realizan las inspecciones. Las rutas puede tener diferentes horarios de salida (varias corridas). Para cada uno de los tramos de las rutas se conoce una criticidad y esta va definida por las empresas. Cada inspector tiene una jornada laboral y debe realizar la inspección sin pasarse de esta jornada. Los inspectores pueden salir de diferentes nodos pero deben retornar al nodo de salida al final de su recorrido de inspección. Los inspectores deben almorzar en el mismo autobús donde se encuentren inspeccionando o en un nodo de transfer que se encuentren en el horario que deseen. Se busca que los inspectores verifiquen la mayor cantidad de tramos críticos en una jornada laboral, típicamente entre 8 y 12 horas. Este problema se asemeja a un problema de ruteo de arcos con múltiples depósitos y la planeación de múltiples vehículos simultáneamente con restricciones de capacidad (solo un inspector en una corrida a la vez).

Como se ha planteado en el capítulo de introducción el problema científico que se aborda en este trabajo se resume en: desarrollar una metodología de apoyo a la decisión para la planeación de las inspecciones a las rutas de trasporte interurbano en base a la criticalidad establecida por la empresa. Se realiza una metodología de apoyo a la decisión desde una perspectiva a resultados, teniendo en cuenta las necesidades del problema. Esto implica integrar la toma de decisiones estratégicas, tácticas y operativas, mismas que se traducen en:

- Que tramos y rutas formarán parte del diseño de la planeación de las inspecciones.
- Cual inspector inspeccionara cual ruta, en cual tramo y en que horario.

- Que horarios de salida tendrá cada inspector.
- Que horario de salida tendrá cada ruta.
- Que cantidad de inspecciones diarias se podrá programar para cada inspector.

Estas decisiones acercan el problema en cuestión a problemas re ruteo por arcos (cartero o recolector de basura). En este trabajo se propone resolver estos problemas de forma integrada, para lo cual se ha propuesto una metodología de apoyo a la decisión que se presenta en el siguiente capitulo.

3.2 Supuestos del modelo

Para la elaboración del modelo matemático, se han definido las condiciones bajo las cuales funciona el modelo, teniendo en cuenta que se trata de un multigrafo, se describen las características de este modelo a continuación donde se supone no hay retrasos en llegadas o salidas:

1. Los tiempos de espera en un nodo hasta tomar el siguiente camión son despreciados en el modelo, es decir, toman valor cero.

   Rutas, Salidas y Nodos

2. Cada ruta esta particionada en tramos donde se realizan las inspecciones.
3. Un tramo es un recorrido entre dos paradas consecutivas.
4. Cada inspector inspecciona varias rutas o tramos durante su jornada de trabajo, buscando repetir lo menos posible las rutas o tramos ya inspeccionadas.
5. Cada ruta puede tener diferentes horarios de salida en un día.
6. No es necesario inspeccionar todas las rutas y tramos en una jornada laboral.
7. Se necesita cubrir la mayor cantidad de tramos caracterizados como críticos.

8. Cada tramo tiene una criticidad diferente especificada por la empresa.

9. Existen nodos de transfer donde coinciden dos o más rutas.

Inspectores

10. Si un inspector no está disponible, no se toma en cuenta para planificación.

11. Debe haber un solo inspector inspeccionando el mismo tramo en el mismo horario.

12. Los inspectores deben iniciar y terminar en el mismo nodo.

13. La jornada de trabajo de cada inspector dura de 8 a 12 horas.

14. Los inpectores al llegar a un nodo de transfer buscarán tomar el autobús que llegue primero con la mayor criticidad.

15. Los inspectores almuerzan en los autobuses o cuando esperan en un nodo transfer para cambiar de ruta.

Autobuses

16. Los autobuses cumplen el horario al 100 por ciento no hay retardos ni adelantos en las llegadas o salidas.

Los supuestos anteriores son las condiciones que se tomaron en cuenta a la hora de formular y dar solución a este problema. Sin embargo, se pueden considerar otros aspectos pero esto pudiera implicar una reformulación del problema.

3.3 Modelo lineal entero mixto

Esta sección presenta el modelo matemático que se ha desarrollado para atacar el problema que se aborda en este trabajo de investigación. Dicho modelo, consta de una función objetivo y una serie de restricciones que debe cumplir.

1. **Conjuntos**

   I : Inspectores

   S : Salidas

   S_i^o :Conjunto de salidas que tienen como origen el nodo desde donde inicia la inspección el inspector i.

   S_i^f: Conjunto de salidas que tienen como nodo destino el nodo desde donde inicia el inspector i.

   $S_{s'}^+$: Conjunto de todas las salidas que salen de la terminal a la que llega s' en un horario dado

   $S_{s'}^-$: Conjunto de todas las salidas que llegan de la terminal a la que llega s' en un horario dado.

   t_i^o: Tiempo en el que inicia su jornada laboral el inspector i.

2. **Parámetros**

   T : duración de la jornada laboral en horas

   C_s : Conjunto de criticidades.

   O_i : Costo por inspector considerando los salarios por inspector por jornada laboral.

3. **Variables**

   X : variable que indica si una salida específica es inspeccionada (1) por un inspector o no (0)

   Y : variable que representa si un inspector esta activo o no en la planeación

4. **Restricciones**

   Criticidad.

$$\sum_{s \in S} C_s * X_{is} <= y_i \qquad \forall i \in I \qquad (0)$$

Inspectores.

$$1 <= \sum_{i \in I} y_i \qquad \forall i \in I \qquad (1)$$

Establece que el tiempo total en minutos de cada inspector no debe sobrepasar la jornada laboral.

$$\sum_{s \in S} t_s * X_{is} <= y_i * \sum_{s \in S} t_s \qquad \forall i \in I \qquad (2)$$

Establece que si un inspector está activo entonces debe salir de su tramo asignado.

$$\sum_{s \in S_i^o} x_{is} = y_i \qquad \forall i \in I \qquad (3)$$

Establece que si se activa el inspector debe estar activo en el tramo fin.

$$y_i <= \sum_{s \in \ S_i^f} x_{is} <= y_i * \mid S_i^f \mid \qquad \forall i \in I \quad (4)$$

Establece que un inspector no puede tener inspecciones antes de su horario de salida.

$$\sum_{s \in \ S_i^-} x_{is} = 0 \qquad \forall i \in I \qquad (5)$$

Establece que un tramo solo puede ser inspeccionado por un inspector al mismo tiempo.

$$\sum_{i \in\ I} x_{is} <= 1 \qquad \forall s \in S \quad (6)$$

Establece que un inspector no puede inspeccionar 2 tramos al mismo tiempo.

$$y_i <= \sum_{s \in\ S^+_{s'}} X_{is} <= y_i * \mid S^+_{s'} \mid \qquad \forall i \in I \wedge \forall s' \in S \quad (7)$$

Establece la **Conservacion de Flujo**, indicando que si no es tiempo de inicio del inspector entonces los inspectores que ingresan a un nodo deben salir de el.

$$\sum_{s \in\ S^+_{s'}} X_{is} = X_{is'} \qquad \forall i \in I \wedge \forall s' \in S \quad (8)$$

5. **Objetivo**

   El objetivo es la suma ponderada de dos objetivos:

   - maximizar el total de salidas criticas inspeccionados
   - minimizar el costo de los inspectores empleados (considerando los salarios por inspector por jornada laboral)

$$Minimizar\ \ \lambda * \sum_{i \in\ I} c_s * x_{is} + (1 - \lambda) * \sum_{i \in\ I} y_i * O_i \qquad (1)$$

3.4 Metodología de Apoyo a la Decisión

Una metodología de apoyo a la decisión es un conjunto de axiomas, reglas, métodos, procedimientos, que son desarrollados para ayudar al tomador de decisiones durante todas las etapas del proceso de decisión. El propósito de la metodología de apoyo a la decisión es garantizar que el proceso de decisión sea realizado por el tomador del decisiones de forma eficiente, eficaz y racionalmente.

Simón en (1960) planteo que el proceso de toma de decisiones consiste en tres fases: fase de inteligencia, fase de diseño y la fase de elección. La fase de inteligencia consiste en la identificación y estructuración del problema, definición de los objetivos, recopilación de datos. En la fase de diseño se construye y valida el modelo del problema es decir, se establecen los criterios de evaluación, se buscan soluciones y se prueba la factibilidad de las soluciones. Y finalmente la fase de elección consiste en la búsqueda, evaluación y recomendación de una solución apropiada al modelo. En la figura 3.1 se esquematiza el proceso de la toma de decisiones representado en sus tres fases.

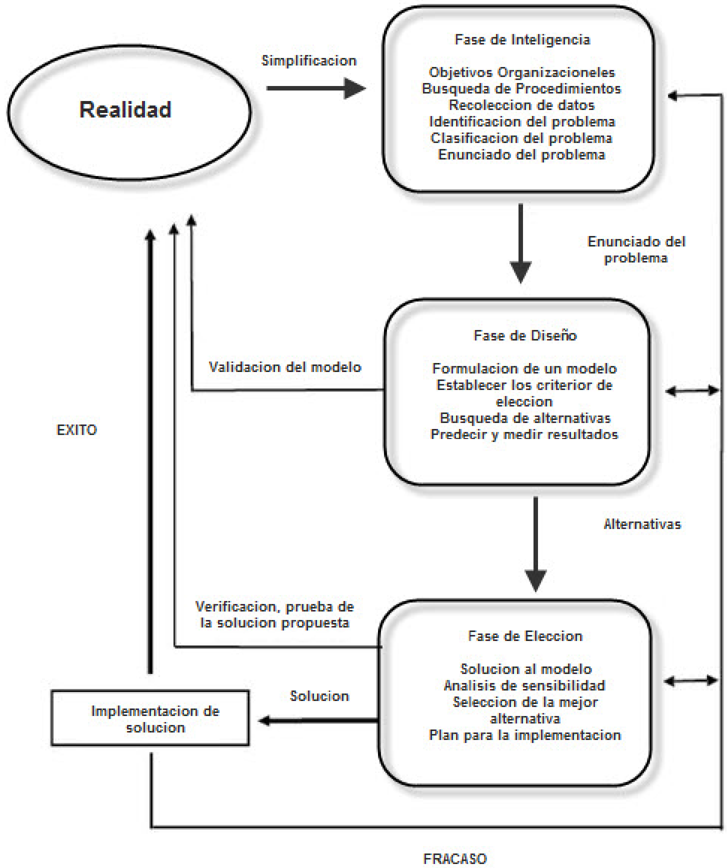

Figura 3.1: Proceso de toma de decisiones

En este trabajo se propone una metodología de apoyo a la decisión desarrollada en las tres fases antes descritas: fase de modelo, fase de optimización y fase de selección. En la figura 3.2 se esquematiza la metodología a la par de las fases con la toma de decisiones, las cuales se explicaron anteriormente.

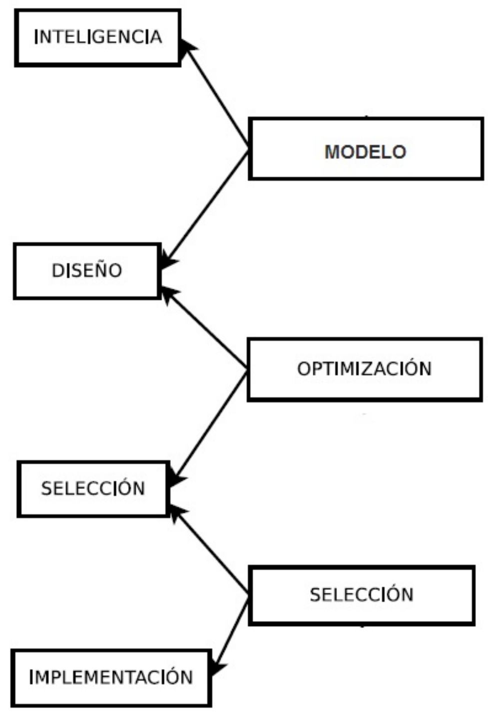

Figura 3.2: Metodología Propuesta

3.4.1 Fase 1 o de Pre-procesamiento

Se aplican una algunos aspectos que ayudan a realizar un pre-procesamiento en el modelo matemático con el objetivo de reducir el número de variables y el tiempo computacional. Los pre - procesamientos realizados fueron:

1. La reducción de dominio aplicado a las variables de decisión. Este es un procedimiento simple pero muy efectivo ya que se realiza un pre-calculo del rango de valores que puede tener la variable de decisión.

2. A partir de las propiedades descritas en el capítulo anterior, se realiza un análisis que conlleva a dejar solamente una variable de decisión, la cual seria $Xijrt$ que indica si el inspector i inspecciona la ruta j, en el tramo r, en el tiempo t, tomando valor (1) si está activo, (0) si no está activo.

3. Análisis realizado para definir el objetivo de optimización que llevara el modelo, centrando varios puntos en uno solo maximizando la cantidad de inspecciones diarias a las rutas o tramos con mayor criticidad establecida. Ademas este análisis ayudo a reducir el numero de variables del modelo matemático.

3.4.2 Fase 2 o de optimización

Esta es la fase encargada de obtener opciones de diseño y planeación de la planeacion de inspecciones que resultan de un proceso de optimización en el cual se consideran todos los aspectos discutidos en el capitulo anterior. Se plantea un problema de optimización y se diseña un modelo matemático lineal entero mixto que representa el problema.

Primeramente se crea este modelo como inicios de estudios del problema para poder obtener soluciones exactas utilizando instancias pequeñas y que son relativamente fáciles de generar las cotas que servirían para la implementación posterior de un algoritmo heurístico, ademas que estas instancias arrojan estas soluciones exactas con facilidad en cuanto a términos computacionales. Ahora los algoritmos heurísticos se utilizaran para poder resolver instancias que resulten ser difíciles en términos computacionales.

La base para poder modificar el problema y dar solución a cualquier situación que se desee implementar es el modelo matemático. Este modelo permite modificaciones tanto en la función objetivo, restricciones o variable de decisión. Se le da entrada al modelo de optimización con aquellos inspectores que están disponibles a trabajar en el día indicado. En esta fase se va a obtener como solución del problema de optimización, la planeación de las rutas a inspeccionar por cada inspector en una jornada laboral. Esta solucion garantiza el objetivo de optimización planteado en el modelo, incorporando en cada planeacion de cada inspector los tramos de mayor criticidad establecidos en la instancia.

3.4.3 Fase 3 o de selección

El propósito de esta fase consiste en apoyar al tomador de decisiones en la búsqueda de la ROI (región de interés). Para poder encontrarla se pueden aplicar métodos de búsqueda que incorporan preferencias a priori, durante o luego del proceso de optimización. En la programación matemática han resultado populares los métodos de incorporación de metas, estos métodos incorporan preferencias a priori (Romero, 1993) o durante el proceso. Para problemas multiobjetivo bajo certeza, existen métodos basados en la metodología PROMETHE [Fernández Barberis, 2002], entre otros, que incorporan preferencias a posteriori para encontrar la ROI [Jimenez et al., 2017].

Esta fase no es más que el proceso de seleccionar la mejor alternativa de todas las valoradas. Generalmente las metodologías desarrolladas para resolver problemas de optimización terminan cuando se obtiene un frente satisfactorio (para el tomador de decisiones) que contiene imágenes de soluciones eficientes. Sucede que cuando al tomador de decisiones se le presenta el conjunto de soluciones eficientes encontradas se origina otro problema de decisión, ya que entre estas soluciones el tomador de decisiones debe decidir cuáles son las que representan compromisos aceptables entre los múltiples criterios u objetivos que han sido considerados en su problema de decisión. Finalmente el tomador de decisiones escogerá una o un conjunto muy reducido de soluciones eficientes (ROI), convirtiéndose en un problema de selección multicriterio.

Esta búsqueda no es un proceso simple ya que implica la modelación de preferencias del tomador de decisiones. Este debe realizar comparaciones entre las alternativas cuando se tienen más de tres criterios ya que para el caso de menos criterios puede hacerse de forma visual, solo analizando el frente no denominado.

3.5 Conclusiones del Capítulo 3

El problema que se aborda aquí es de gran importancia para la planificación de la inspección al transporte interurbano, especialmente para México donde son escasas las empresas de este tipo de transporte que utilizan algún tipo de sistema automatizado para estas labores. Con el modelo matemático que se presentó se quiere lograr una planificación de inspección óptima de los tramos que contienen las rutas de transporte interurbanas caracterizados como críticos. No existe formulación matemática previa de este problema, aunque existen similares no contienen todas las características de este problema considerado.

En este capitulo se ha presentado una metodología de apoyo a la decisión que permite desarrollar un comportamiento racional en el proceso de toma de decisiones. Esta metodología no debe interpretarse como algo rígido sino más bien como un marco de trabajo donde se expone como llevar a cabo el apoyo a la decisión.

Capítulo 4

Experimentacion

Este capítulo describe la el experimento que se realizo para probar la heurística construida para dar solución al problema de inspección de rutas interurbanas. En la sección 4.1 se describe la naturaleza de los datos con una subseccion que describe las instancias utilizadas para realizar los experimentos. En la siguiente sección se describe la Heurística y por ultimo se muestran los resultados que se obtuvieron por medio del experimento y comenta el análisis de estos resultados.

4.1 Naturaleza de los datos

Los datos para el análisis pueden recogerse de muchas maneras, pero fundamentalmente se pueden establecer dos tipos de datos según su forma de obtención: datos observacionales y datos experimentales. Para el trabajo en cuestión los datos son del tipo experimentales.

Los datos experimentales son aquellos cuyo valor los fija el experimentador. El analista manipula deliberadamente los valores del factor con el fin de poder establecer una relación de causalidad entre dicho factor y la variable respuesta. El analista también consigue de esta forma aquellos valores en los que está interesado. La principal ventaja de los datos experimentales es que es más fácil establecer relaciones de causalidad entre las variables, pues el analista puede observar la evolución de los resultados a medida que va manipulando los factores.

La capacidad de poder elegir los valores de las variables que interesen hace que se necesiten menos datos para sacar conclusiones que si se usase datos observacionales. La necesidad de economizar a la hora de recoger datos es muy importante en ingeniería, por ejemplo los costes de manipular procesos industriales o realizar ensayos de laboratorio pueden ser muy elevados.

Otra ventaja de los datos experimenales es que permiten provocar situaciones de interés que difícilmente puedan observarse en la realidad o que se precise de demasiado tiempo de observación. En ocasiones, un experimento será la única forma de obtener información.

4.1.1 Descripción de las instancias

A continuación se realiza una descripción de los datos utilizados para realizar el experimento, los cuales se convierten en instancias. Se explica la naturaleza de los datos que se van a utilizar y la estructura de la instancia.

Para este experimento se utilizó un generador de instancias desarrollado en el lenguaje de programación *R-Studio*. Se toman en cuenta para esta generación de las instancias los factores antes descritos. Además el generador toma en cuenta los parámetros generales para las rutas, que serian: posibles tiempos que tarde un tramo, la posible criticidad que tiene un tramo, tiempos entre salidas ya que para cada nueva salida se calcula el tiempo, el posible número de salidas por ruta y las posibles horas de inicio la primera salida.

Para la creación de la ruta se tienen en cuenta el número de nodos que pertenecen a cada ruta, este debe ser mayor que dos, se añade como nodo final el mismo nodo inicial para cerrar la ruta. Con todas las especificaciones anteriores entre otras, se generan las instancias. La siguiente tabla muestra un resumen de algunas instancias que se utilizaron en el experimento.

Tabla 4.1: Resumen parcial de algunas instancias generadas

Inst	**Rutas**	**Insp.**	**C. T O.**	**C. T. D.**	**C. O.**	**C. D.**	**T. Tram**	**Jornada**
1	2	1	5	5	3	3	6	8
2	2	2	5	5	3	3	6	8
3	2	2	9	9	5	5	11	12
3	3	3	8	8	11	11	16	12

4.2 Descripción de la Heurística y Resultados Obtenidos

Para dar solución al problema planteado de rutas de inspección al transporte interurbano, se desarrolló una heurística greedy en el lenguaje de programación *R-Studio*, la cual se describe a continuación en el diagrama de flujo de la figura 4.1.

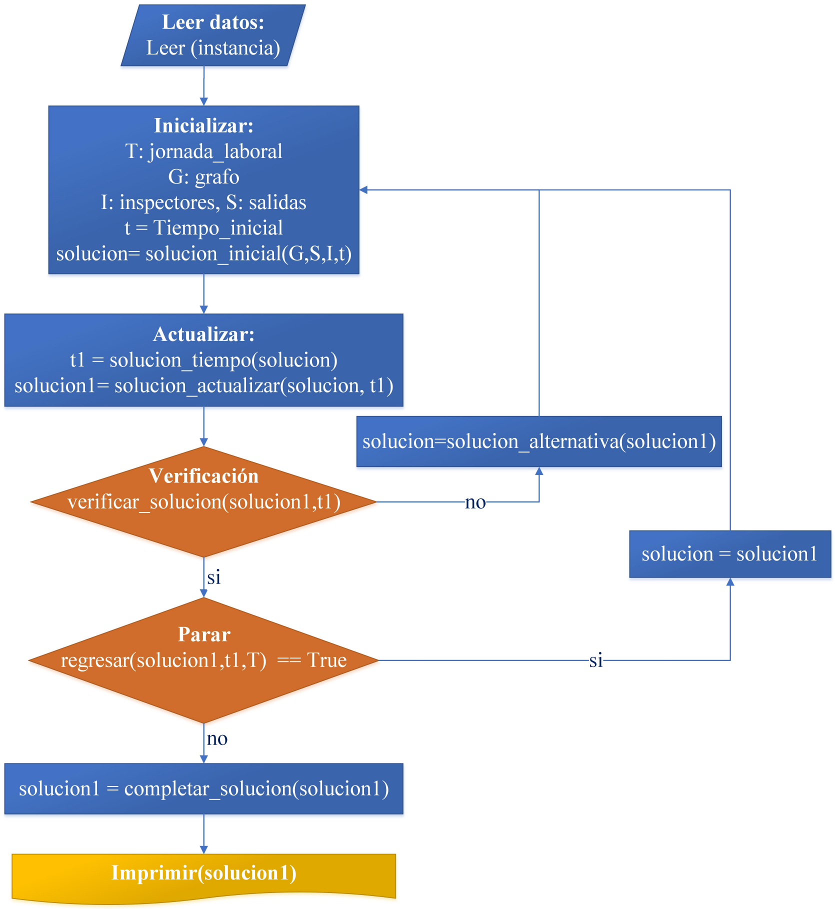

Figura 4.1: Diagrama de flujo que representa funcionamiento de la heurística greedy.

Para una mejor comprensión del diagrama anterior se describen las funciones desarrolladas:

- $solucion_inicial$: ubica los inspectores y las salidas en sus respectivas terminales de partida (nodos), actualiza el tiempo transcurrido hasta ubicar el último inspector o la salida inmediata a la ubicación del ultimo inspector.

- $solucion_tiemp$: devuelve el tiempo real hasta el que se avanzo al construir la solución.

- $solucion_actualizar$: actualiza la solución actual a partir del tiempo en que finalizo esta.

- $solucion_alternativa$: calcula una solución alternativa.
- $verificar_solucion$: checa que la solución sea factible.
- $regresar$: verifica si los inspectores pueden regresar al punto de partida desde su posición actual, antes de que finalice la jornada laboral.
- $completar_solucion$: finaliza la generación de la solución completando los recorridos de los inspectores, desde su punto actual al punto respectivo desde donde salieron.

4.3 Resultados Obtenidos

En esta sección se analizan los resultados obtenidos al correr las instancias generadas utilizando un poderoso solucionador de Programas lineales (LP)y Programación entera mixta (MIP) llamado PyMathProg. Este proporciona una sintaxis de modelado fácil y flexible usando Python para crear y optimizar modelos de programación matemática. Es una especie de reencarnación de AMPL y GNU MathProg en Python de código libre.

PyMathProg permite hacer programación matemática en Python con facilidad y flexibilidad, para crear, optimizar, cambiar y volver a optimizar su modelo sin esfuerzo de más. Todo lo anterior se logra con un buen lenguaje de modelado (Python) un solucionador potente y flexible, que es GLPK, un conjunto de herramientas de integrado, con base de datos, trazado, etc y un entorno de modelado interactivo para un fácil aprendizaje.

Al estar integrado al Python, puede aprovechar todas os beneficios disponibles en Python: como el fácil acceso a la base de datos, la presentación gráfica de su solución, el análisis estadístico o el uso de pymprog para la inteligencia artificial en los juegos, etc.

Se muestran los resultados obtenidos en las instancias descritas anteriormente en la siguiente tabla. Por cada valor de lambda que se tuvo en cuenta (0.0001 para minimizar inspectores - costos, 0.999 para maximizar criticidad y 0.5 para equilibrar las funciones objetivos) se capturan la solución de cada función objetivo.

Tabla 4.2: Valores de Criticidad y cantidad de inspectores en las soluciones encontradas

		Lambda = 0.001		**Lambda = 0.5**		**Lambda = 0.999**	
No.	**Instancias**	**Crit.**	**Insp.**	**Crit.**	**Insp.**	**Crit.**	**Insp.**
2	2_insp_2_rut_3_nod	11	1	11	1	20	2
3	2_insp_2_rut_5_nod	22	1	22	1	44	1
4	3_insp_2_rut_13_nod	28	1	53	2	67	3

Se muestra en los resultados que a mayor valor de Lambda (0.999) se busca maximizar la criticidad en la solución encontrada, en el caso de las instancias ultimas, se evidencia un aumento en la criticidad total. Esta máxima criticidad se logra con la mayor cantidad de inspectores, o en el caso de una de las instancias, con la menor cantidad. Esto se debe a que las conexiones de una instancia respecto a la otra es mucho mayor, ocasionando que se necesite utilizar mas inspectores.

Para cuando los valores de lambda fueron 0.5, el cambio no fue significativo en todas las instancias, a pesar de buscar un equilibrio entre las funciones objetivos. Los valores de criticidad aumentaron sin aumentar el numero de inspectores.

En los casos con lambda 0.001, donde se busca minimizar la cantidad de inspectores y con esto el costo, en todos los casos se logro tener la menor cantidad de inspectores posibles, logrando la minimizacion de los costos.

El aumento de la cantidad de nodos, beneficia el resultado en la maximizacion de la criticidad y no afecto el resultado de la cantidad de inspectores, por tanto el aumento de la cantidad de nodos, influye positivamente en los resultados de las funciones objetivos. Por otro lado el aumento de la cantidad de inspectores afecta positivamente la solución pero no a gran escala, por lo que se logra minimizar siempre el costo.

Para consultar las soluciones encontradas en las instancias descritas consulte la sección de Anexos.

Por otro lado, los resultados de la Heurística arrojan varios óptimoscon respecto a la criticidad máxima. El resultado se muestra en la siguiente tabla.

Tabla 4.3: Resultados de la Heuristica Greedy

		Optimo 1	**Optimo 2**	**Optimo 3**
No.	**Instancias**	**Crit.**	**Crit.**	**Crit.**
1	1_insp_2_rut_3_nod	14	8	7
2	2_insp_2_rut_3_nod	12	8	9
3	2_insp_2_rut_5_nod	23	12	
4	3_insp_2_rut_13_nod	30	26	15

Capítulo 5

Conclusiones

Este capítulo plasma las conclusiones y recomendaciones generales con las cuales se pretende que la propuesta presentada actual se complete más. En la Sección 6.1 se presentan las conclusiones, mientras que en la Sección 6.2 se habla acerca del trabajo futuro.

5.1 Conclusiones

El objetivo general de este trabajo de tesis consiste en realizar una aproximación metodológica al problema de inspección de rutas interurbanas desde la perspectiva de investigación de operaciones. Este objetivo fue cumplido en su totalidad, ya que se realizó una metodología que da solución a este problema, donde se tienen en cuenta las fases del proceso de toma de decisiones. Esta es una metodología de apoyo a la decisión que permite desarrollar un comportamiento racional en el proceso de toma de decisiones, por lo tanto no debe interpretarse como algo rígido sino más bien como un marco de trabajo donde se expone como llevar a cabo el apoyo a la decisión. A su ves esta metodología brinda un apoyo en la toma de decisiones para la planeación de las inspecciones a las rutas de trasporte interurbano en base a la criticidad establecida por la empresa, dando cumplimiento al primer de los objetivos específicos trazados en este documento.

Se da cumplimiento al segundo objetivo especifico trazado con el desarrollo del modelo matemático para resolver instancias de tamaño mediano o pequeño del problema de inspección de rutas interurbanas. Este modelo permite desarrollar metodologías de apoyo a la decisión efectivas para la construcción de la planificación de las inspecciones de autobuses en una empresa de transporte interurbanos.

Se realizaron experimentos científicos que han mostrado que el modelo propuesto puede ser resuelto de manera eficiente para problemas reales en empresas medianas o pequeñas (con un máximo de rutas).

Se desarrolló además una heurística eficiente que logra encontrar resultados factibles en tiempos de computo aceptables y garantizando cubrir las rutas de mayor criticidad en diferentes tipos de instancias generadas aleatoriamente, por lo cual se da cumplimiento al ultimo objetivo especifico trazado al inicio del documento.

5.2 Trabajo Futuro

El problema presentado en este trabajo de investigación tiene una amplia gama de oportunidades para continuar su estudio. A continuación, mencionamos algunas de ellas.

Trabajar con instancias reales para darle una mayor aproximación al modelo propuesto a la vida actual y la necesidad actual de alguna empresa de transporte interurbano.

Incorporar los horarios de comida de los inspectores y denominar algunos nodos como los lugares donde los inspectores pueden comer.

Al concluir el trabajo de investigación, se observó que las características del problema son ideales para realizar una simulación del mismo, por lo cual es un trabajo futuro que seria indicado llevar a cabo.

Capítulo 6

Anexos

```
MIP solver is set to intopt



-----------------------objetivos MIP--------------------------


lambda:  0.001  ** inspectores:  1.0  ** criticidad:
i1:  11.0
i2:  0.0

-----------------------primal valor: y MIP ----------------
i1  :  1.0
i2  :  0.0

-----------------------primal valor: x MIP ----------------
('i1', (1, '9:00', 'A', 'B', '10:00'))  :  0.0
('i1', (1, '10:00', 'B', 'C', '13:00'))  :  0.0
('i1', (1, '13:00', 'C', 'A', '14:00'))  :  0.0
('i1', (2, '8:00', 'A', 'C', '10:00'))  :  1.0
('i1', (2, '10:00', 'C', 'B', '12:00'))  :  1.0
('i1', (2, '12:00', 'B', 'A', '16:00'))  :  1.0
('i2', (1, '9:00', 'A', 'B', '10:00'))  :  0.0
('i2', (1, '10:00', 'B', 'C', '13:00'))  :  0.0
('i2', (1, '13:00', 'C', 'A', '14:00'))  :  0.0
('i2', (2, '8:00', 'A', 'C', '10:00'))  :  0.0
('i2', (2, '10:00', 'C', 'B', '12:00'))  :  0.0
('i2', (2, '12:00', 'B', 'A', '16:00'))  :  0.0

---------------MIP solución x---------------
('i1', (2, '8:00', 'A', 'C', '10:00'))
('i1', (2, '10:00', 'C', 'B', '12:00'))
('i1', (2, '12:00', 'B', 'A', '16:00'))

Karush-Kuhn-Tucker optimality conditions:
=========================================
Solver used for this solution: intopt

1) Error for Primal Equality Constraints:
----------------------------------------
Largest absolute error: 0.000000 (row id: 0)
Largest relative error: 0.000000 (row id: 0)

2) Error for Primal Inequality Constraints:
-------------------------------------------
Largest absolute error: 0.000000 (row id: 0)
```

```
MIP solver is set to intopt



----------------------objetivos MIP------------------------

lambda:  0.999  ** inspectores:  2.0  ** criticidad:
i1:  11.0
i2:  9.0

----------------------primal valor: y MIP ---------------
i1  :  1.0
i2  :  1.0

----------------------primal valor: x MIP ---------------
('i1', (1, '9:00', 'A', 'B', '10:00'))  :  0.0
('i1', (1, '10:00', 'B', 'C', '13:00'))  :  0.0
('i1', (1, '13:00', 'C', 'A', '14:00'))  :  0.0
('i1', (2, '8:00', 'A', 'C', '10:00'))  :  1.0
('i1', (2, '10:00', 'C', 'B', '12:00'))  :  1.0
('i1', (2, '12:00', 'B', 'A', '16:00'))  :  1.0
('i2', (1, '9:00', 'A', 'B', '10:00'))  :  1.0
('i2', (1, '10:00', 'B', 'C', '13:00'))  :  1.0
('i2', (1, '13:00', 'C', 'A', '14:00'))  :  1.0
('i2', (2, '8:00', 'A', 'C', '10:00'))  :  0.0
('i2', (2, '10:00', 'C', 'B', '12:00'))  :  0.0
('i2', (2, '12:00', 'B', 'A', '16:00'))  :  0.0

--------------MIP solución x--------------
('i1', (2, '8:00', 'A', 'C', '10:00'))
('i1', (2, '10:00', 'C', 'B', '12:00'))
('i1', (2, '12:00', 'B', 'A', '16:00'))
('i2', (1, '9:00', 'A', 'B', '10:00'))
('i2', (1, '10:00', 'B', 'C', '13:00'))
('i2', (1, '13:00', 'C', 'A', '14:00'))

--------------MIP solución x--------------
('i1', (2, '8:00', 'A', 'C', '10:00'))
('i1', (2, '10:00', 'C', 'B', '12:00'))
('i1', (2, '12:00', 'B', 'A', '16:00'))
('i2', (1, '9:00', 'A', 'B', '10:00'))
('i2', (1, '10:00', 'B', 'C', '13:00'))
('i2', (1, '13:00', 'C', 'A', '14:00'))

Karush-Kuhn-Tucker optimality conditions:
=========================================
Solver used for this solution: intopt

1) Error for Primal Equality Constraints:
----------------------------------------
Largest absolute error: 0.000000 (row id: 0)
Largest relative error: 0.000000 (row id: 0)

2) Error for Primal Inequality Constraints:
-------------------------------------------
Largest absolute error: 0.000000 (row id: 0)
Largest relative error: 0.000000 (row id: 0)


###>MILP Objective value: 0

estado:  solution is optimal
```

Figura 6.2: Resultado MIP Instancia 2 con Lambda 0,999

```
MIP solver is set to intopt



-----------------------objetivos MIP--------------------------


lambda:  0.001  ** inspectores:  1.0  ** criticidad:
i1:  22.0
i2:  0.0

-----------------------primal valor: y MIP ----------------
i1  :  1.0
i2  :  0.0

-----------------------primal valor: x MIP ----------------
('i1', (1, '8:00', 'A', 'B', '9:00'))  :  0.0
('i1', (1, '9:00', 'B', 'C', '10:30'))  :  0.0
('i1', (1, '10:30', 'C', 'A', '12:30'))  :  0.0
('i1', (1, '10:30', 'C', 'D', '11:00'))  :  0.0
('i1', (1, '11:00', 'D', 'A', '13:00'))  :  0.0
('i1', (2, '10:00', 'A', 'D', '13:00'))  :  1.0
('i1', (2, '13:00', 'D', 'E', '17:00'))  :  1.0
('i1', (2, '17:00', 'E', 'A', '20:00'))  :  1.0
('i1', (2, '14:00', 'A', 'D', '17:00'))  :  0.0
('i1', (2, '17:00', 'D', 'E', '21:00'))  :  0.0
('i1', (2, '21:00', 'E', 'A', '24:00'))  :  0.0
('i2', (1, '8:00', 'A', 'B', '9:00'))  :  0.0
('i2', (1, '9:00', 'B', 'C', '10:30'))  :  0.0
('i2', (1, '10:30', 'C', 'A', '12:30'))  :  0.0
('i2', (1, '10:30', 'C', 'D', '11:00'))  :  0.0
('i2', (1, '11:00', 'D', 'A', '13:00'))  :  0.0
('i2', (2, '10:00', 'A', 'D', '13:00'))  :  0.0
('i2', (2, '13:00', 'D', 'E', '17:00'))  :  0.0
('i2', (2, '17:00', 'E', 'A', '20:00'))  :  0.0
('i2', (2, '14:00', 'A', 'D', '17:00'))  :  0.0
('i2', (2, '17:00', 'D', 'E', '21:00'))  :  0.0
('i2', (2, '21:00', 'E', 'A', '24:00'))  :  0.0

---------------MIP solución x---------------
('i1', (2, '10:00', 'A', 'D', '13:00'))
('i1', (2, '13:00', 'D', 'E', '17:00'))
('i1', (2, '17:00', 'E', 'A', '20:00'))

Karush-Kuhn-Tucker optimality conditions:
=========================================
Solver used for this solution: intopt

1) Error for Primal Equality Constraints:
----------------------------------------
Largest absolute error: 0.000000 (row id: 0)
Largest relative error: 0.000000 (row id: 0)

2) Error for Primal Inequality Constraints:
-------------------------------------------
Largest absolute error: 0.000000 (row id: 0)
Largest relative error: 0.000000 (row id: 0)


###>MILP Objective value: -4

estado:  solution is optimal
```

Figura 6.3: Resultado MIP Instancia 3 con Lambda 0,001

```
MIP solver is set to intopt



----------------------objetivos MIP-------------------------


lambda:  0.999  ** inspectores:  2.0  ** criticidad:
i1:  22.0
i2:  22.0

----------------------primal valor: y MIP ---------------
i1  :  1.0
i2  :  1.0

----------------------primal valor: x MIP ---------------
('i1', (1, '8:00', 'A', 'B', '9:00'))  :  0.0
('i1', (1, '9:00', 'B', 'C', '10:30'))  :  0.0
('i1', (1, '10:30', 'C', 'A', '12:30'))  :  0.0
('i1', (1, '10:30', 'C', 'D', '11:00'))  :  0.0
('i1', (1, '11:00', 'D', 'A', '13:00'))  :  0.0
('i1', (2, '10:00', 'A', 'D', '13:00'))  :  1.0
('i1', (2, '13:00', 'D', 'E', '17:00'))  :  1.0
('i1', (2, '17:00', 'E', 'A', '20:00'))  :  1.0
('i1', (2, '14:00', 'A', 'D', '17:00'))  :  0.0
('i1', (2, '17:00', 'D', 'E', '21:00'))  :  0.0
('i1', (2, '21:00', 'E', 'A', '24:00'))  :  0.0
('i2', (1, '8:00', 'A', 'B', '9:00'))  :  0.0
('i2', (1, '9:00', 'B', 'C', '10:30'))  :  0.0
('i2', (1, '10:30', 'C', 'A', '12:30'))  :  0.0
('i2', (1, '10:30', 'C', 'D', '11:00'))  :  0.0
('i2', (1, '11:00', 'D', 'A', '13:00'))  :  0.0
('i2', (2, '10:00', 'A', 'D', '13:00'))  :  0.0
('i2', (2, '13:00', 'D', 'E', '17:00'))  :  0.0
('i2', (2, '17:00', 'E', 'A', '20:00'))  :  0.0
('i2', (2, '14:00', 'A', 'D', '17:00'))  :  1.0
('i2', (2, '17:00', 'D', 'E', '21:00'))  :  1.0
('i2', (2, '21:00', 'E', 'A', '24:00'))  :  1.0

--------------MIP solución x--------------
('i1', (2, '10:00', 'A', 'D', '13:00'))
('i1', (2, '13:00', 'D', 'E', '17:00'))
('i1', (2, '17:00', 'E', 'A', '20:00'))
('i2', (2, '14:00', 'A', 'D', '17:00'))
('i2', (2, '17:00', 'D', 'E', '21:00'))
('i2', (2, '21:00', 'E', 'A', '24:00'))

Karush-Kuhn-Tucker optimality conditions:
=========================================
Solver used for this solution: intopt

1) Error for Primal Equality Constraints:
----------------------------------------
Largest absolute error: 0.000000 (row id: 0)
Largest relative error: 0.000000 (row id: 0)

2) Error for Primal Inequality Constraints:
-------------------------------------------
Largest absolute error: 0.000000 (row id: 0)
Largest relative error: 0.000000 (row id: 0)


###>MILP Objective value: 0

estado:  solution is optimal
```

Figura 6.4: Resultado MIP Instancia 3 con Lambda 0,999

```
-----------------------objetivos MIP--------------------------


lambda:  0.001  ** inspectores:  1.0  ** criticidad:
i1:  1.0
i2:  1.0
i3:  26.0

-----------------------primal valor: y MIP ----------------
i1  :  0.0
i2  :  0.0
i3  :  1.0

----------------------primal valor: x MIP ----------------
('i1', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
('i1', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
('i1', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
('i1', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
('i1', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
('i1', (2, '10:00', 'E', 'F', '11:00'))  :  0.0
('i1', (2, '11:00', 'F', 'A', '12:00'))  :  0.0
('i1', (2, '12:00', 'A', 'B', '13:00'))  :  0.0
('i1', (2, '13:00', 'B', 'C', '15:00'))  :  0.0
('i1', (2, '15:00', 'C', 'E', '16:00'))  :  0.0
('i1', (3, '9:00', 'E', 'L', '11:00'))  :  0.0
('i1', (3, '11:00', 'L', 'M', '12:00'))  :  0.0
('i1', (3, '12:00', 'M', 'A', '14:00'))  :  0.0
('i1', (3, '14:00', 'A', 'B', '15:00'))  :  0.0
('i1', (3, '15:00', 'B', 'D', '16:00'))  :  0.0
('i1', (3, '16:00', 'D', 'E', '17:00'))  :  0.0
('i2', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
('i2', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
('i2', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
('i2', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
('i2', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
('i2', (2, '10:00', 'E', 'F', '11:00'))  :  0.0
('i2', (2, '11:00', 'F', 'A', '12:00'))  :  0.0
('i2', (2, '12:00', 'A', 'B', '13:00'))  :  0.0
('i2', (2, '13:00', 'B', 'C', '15:00'))  :  0.0
('i2', (2, '15:00', 'C', 'E', '16:00'))  :  0.0
('i2', (3, '9:00', 'E', 'L', '11:00'))  :  0.0
('i2', (3, '11:00', 'L', 'M', '12:00'))  :  0.0
('i2', (3, '12:00', 'M', 'A', '14:00'))  :  0.0
('i2', (3, '14:00', 'A', 'B', '15:00'))  :  0.0
('i2', (3, '15:00', 'B', 'D', '16:00'))  :  0.0
('i2', (3, '16:00', 'D', 'E', '17:00'))  :  0.0
('i3', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
('i3', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
('i3', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
('i3', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
('i3', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
('i3', (2, '10:00', 'E', 'F', '11:00'))  :  0.0
('i3', (2, '11:00', 'F', 'A', '12:00'))  :  0.0
('i3', (2, '12:00', 'A', 'B', '13:00'))  :  0.0
('i3', (2, '13:00', 'B', 'C', '15:00'))  :  0.0
('i3', (2, '15:00', 'C', 'E', '16:00'))  :  0.0
('i3', (3, '9:00', 'E', 'L', '11:00'))  :  1.0
('i3', (3, '11:00', 'L', 'M', '12:00'))  :  1.0
('i3', (3, '12:00', 'M', 'A', '14:00'))  :  1.0
('i3', (3, '14:00', 'A', 'B', '15:00'))  :  1.0
('i3', (3, '15:00', 'B', 'D', '16:00'))  :  1.0
('i3', (3, '16:00', 'D', 'E', '17:00'))  :  1.0

--------------MIP solución x--------------
('i3', (3, '9:00', 'E', 'L', '11:00'))
('i3', (3, '11:00', 'L', 'M', '12:00'))
('i3', (3, '12:00', 'M', 'A', '14:00'))
('i3', (3, '14:00', 'A', 'B', '15:00'))
('i3', (3, '15:00', 'B', 'D', '16:00'))
('i3', (3, '16:00', 'D', 'E', '17:00'))
```

Figura 6.5: Resultado MIP Instancia 4 con Lambda 0,001

```
-----------------------objetivos MIP--------------------------

lambda:  0.5  ** inspectores:  2.0  ** criticidad:
i1:  1.0
i2:  26.0
i3:  26.0

-----------------------primal valor: y MIP ----------------
i1  :  0.0
i2  :  1.0
i3  :  1.0

-----------------------primal valor: x MIP ----------------
('i1', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
('i1', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
('i1', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
('i1', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
('i1', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
('i1', (2, '10:00', 'E', 'F', '11:00'))  :  0.0
('i1', (2, '11:00', 'F', 'A', '12:00'))  :  0.0
('i1', (2, '12:00', 'A', 'B', '13:00'))  :  0.0
('i1', (2, '13:00', 'B', 'C', '15:00'))  :  0.0
('i1', (2, '15:00', 'C', 'E', '16:00'))  :  0.0
('i1', (3, '9:00', 'E', 'L', '11:00'))  :  0.0
('i1', (3, '11:00', 'L', 'M', '12:00'))  :  0.0
('i1', (3, '12:00', 'M', 'A', '14:00'))  :  0.0
('i1', (3, '14:00', 'A', 'B', '15:00'))  :  0.0
('i1', (3, '15:00', 'B', 'D', '16:00'))  :  0.0
('i1', (3, '16:00', 'D', 'E', '17:00'))  :  0.0
('i2', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
('i2', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
('i2', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
('i2', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
('i2', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
('i2', (2, '10:00', 'E', 'F', '11:00'))  :  1.0
('i2', (2, '11:00', 'F', 'A', '12:00'))  :  1.0
('i2', (2, '12:00', 'A', 'B', '13:00'))  :  1.0
('i2', (2, '13:00', 'B', 'C', '15:00'))  :  1.0
('i2', (2, '15:00', 'C', 'E', '16:00'))  :  1.0
('i2', (3, '9:00', 'E', 'L', '11:00'))  :  0.0
('i2', (3, '11:00', 'L', 'M', '12:00'))  :  0.0
('i2', (3, '12:00', 'M', 'A', '14:00'))  :  0.0
('i2', (3, '14:00', 'A', 'B', '15:00'))  :  0.0
('i2', (3, '15:00', 'B', 'D', '16:00'))  :  0.0
('i2', (3, '16:00', 'D', 'E', '17:00'))  :  0.0
('i3', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
('i3', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
('i3', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
('i3', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
('i3', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
('i3', (2, '10:00', 'E', 'F', '11:00'))  :  0.0
('i3', (2, '11:00', 'F', 'A', '12:00'))  :  0.0
('i3', (2, '12:00', 'A', 'B', '13:00'))  :  0.0
('i3', (2, '13:00', 'B', 'C', '15:00'))  :  0.0
('i3', (2, '15:00', 'C', 'E', '16:00'))  :  0.0
('i3', (3, '9:00', 'E', 'L', '11:00'))  :  1.0
('i3', (3, '11:00', 'L', 'M', '12:00'))  :  1.0
('i3', (3, '12:00', 'M', 'A', '14:00'))  :  1.0
('i3', (3, '14:00', 'A', 'B', '15:00'))  :  1.0
('i3', (3, '15:00', 'B', 'D', '16:00'))  :  1.0
('i3', (3, '16:00', 'D', 'E', '17:00'))  :  1.0

---------------MIP solución x---------------
('i2', (2, '10:00', 'E', 'F', '11:00'))
('i2', (2, '11:00', 'F', 'A', '12:00'))
('i2', (2, '12:00', 'A', 'B', '13:00'))
('i2', (2, '13:00', 'B', 'C', '15:00'))
('i2', (2, '15:00', 'C', 'E', '16:00'))
('i3', (3, '9:00', 'E', 'L', '11:00'))
('i3', (3, '11:00', 'L', 'M', '12:00'))
('i3', (3, '12:00', 'M', 'A', '14:00'))
('i3', (3, '14:00', 'A', 'B', '15:00'))
('i3', (3, '15:00', 'B', 'D', '16:00'))
('i3', (3, '16:00', 'D', 'E', '17:00'))
```

Figura 6.6: Resultado MIP Instancia 4 con Lambda 0,5

```
MIP solver is set to intopt



----------------------objetivos MIP-------------------------


lambda:  0.999  ** inspectores:  3.0  ** criticidad:
i1:  15.0
i2:  26.0
i3:  26.0

----------------------primal valor: y MIP ---------------
i1  :  1.0
i2  :  1.0
i3  :  1.0
 ----------------------primal valor: x MIP ---------------
 ('i1', (1, '9:00', 'E', 'H', '10:00'))  :  1.0
 ('i1', (1, '10:00', 'H', 'I', '11:00'))  :  1.0
 ('i1', (1, '11:00', 'I', 'J', '12:00'))  :  1.0
 ('i1', (1, '12:00', 'J', 'C', '13:00'))  :  1.0
 ('i1', (1, '13:00', 'C', 'E', '16:00'))  :  1.0
 ('i1', (2, '10:00', 'E', 'F', '11:00'))  :  0.0
 ('i1', (2, '11:00', 'F', 'A', '12:00'))  :  0.0
 ('i1', (2, '12:00', 'A', 'B', '13:00'))  :  0.0
 ('i1', (2, '13:00', 'B', 'C', '15:00'))  :  0.0
 ('i1', (2, '15:00', 'C', 'E', '16:00'))  :  0.0
 ('i1', (3, '9:00', 'E', 'L', '11:00'))  :  0.0
 ('i1', (3, '11:00', 'L', 'M', '12:00'))  :  0.0
 ('i1', (3, '12:00', 'M', 'A', '14:00'))  :  0.0
 ('i1', (3, '14:00', 'A', 'B', '15:00'))  :  0.0
 ('i1', (3, '15:00', 'B', 'D', '16:00'))  :  0.0
 ('i1', (3, '16:00', 'D', 'E', '17:00'))  :  0.0
 ('i2', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
 ('i2', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
 ('i2', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
 ('i2', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
 ('i2', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
 ('i2', (2, '10:00', 'E', 'F', '11:00'))  :  1.0
 ('i2', (2, '11:00', 'F', 'A', '12:00'))  :  1.0
 ('i2', (2, '12:00', 'A', 'B', '13:00'))  :  1.0
 ('i2', (2, '13:00', 'B', 'C', '15:00'))  :  1.0
 ('i2', (2, '15:00', 'C', 'E', '16:00'))  :  1.0
 ('i2', (3, '9:00', 'E', 'L', '11:00'))  :  0.0
 ('i2', (3, '11:00', 'L', 'M', '12:00'))  :  0.0
 ('i2', (3, '12:00', 'M', 'A', '14:00'))  :  0.0
 ('i2', (3, '14:00', 'A', 'B', '15:00'))  :  0.0
 ('i2', (3, '15:00', 'B', 'D', '16:00'))  :  0.0
 ('i2', (3, '16:00', 'D', 'E', '17:00'))  :  0.0
 ('i3', (1, '9:00', 'E', 'H', '10:00'))  :  0.0
 ('i3', (1, '10:00', 'H', 'I', '11:00'))  :  0.0
 ('i3', (1, '11:00', 'I', 'J', '12:00'))  :  0.0
 ('i3', (1, '12:00', 'J', 'C', '13:00'))  :  0.0
 ('i3', (1, '13:00', 'C', 'E', '16:00'))  :  0.0
 ('i3', (2, '10:00', 'E', 'F', '11:00'))  :  0.0
 ('i3', (2, '11:00', 'F', 'A', '12:00'))  :  0.0
 ('i3', (2, '12:00', 'A', 'B', '13:00'))  :  0.0
 ('i3', (2, '13:00', 'B', 'C', '15:00'))  :  0.0
 ('i3', (2, '15:00', 'C', 'E', '16:00'))  :  0.0
 ('i3', (3, '9:00', 'E', 'L', '11:00'))  :  1.0
 ('i3', (3, '11:00', 'L', 'M', '12:00'))  :  1.0
 ('i3', (3, '12:00', 'M', 'A', '14:00'))  :  1.0
 ('i3', (3, '14:00', 'A', 'B', '15:00'))  :  1.0
 ('i3', (3, '15:00', 'B', 'D', '16:00'))  :  1.0
 ('i3', (3, '16:00', 'D', 'E', '17:00'))  :  1.0

 --------------MIP solución x--------------
 ('i1', (1, '9:00', 'E', 'H', '10:00'))
 ('i1', (1, '10:00', 'H', 'I', '11:00'))
 ('i1', (1, '11:00', 'I', 'J', '12:00'))
 ('i1', (1, '12:00', 'J', 'C', '13:00'))
 ('i1', (1, '13:00', 'C', 'E', '16:00'))
 ('i2', (2, '10:00', 'E', 'F', '11:00'))
 ('i2', (2, '11:00', 'F', 'A', '12:00'))
 ('i2', (2, '12:00', 'A', 'B', '13:00'))
 ('i2', (2, '13:00', 'B', 'C', '15:00'))
 ('i2', (2, '15:00', 'C', 'E', '16:00'))
 ('i3', (3, '9:00', 'E', 'L', '11:00'))
 ('i3', (3, '11:00', 'L', 'M', '12:00'))
 ('i3', (3, '12:00', 'M', 'A', '14:00'))
 ('i3', (3, '14:00', 'A', 'B', '15:00'))
 ('i3', (3, '15:00', 'B', 'D', '16:00'))
 ('i3', (3, '16:00', 'D', 'E', '17:00'))
```

Figura 6.7: Resultado MIP Instancia 4 con Lambda 0,999

Bibliografía

[Abdel-Basset et al., 2019] Abdel-Basset, M., Gunasekaran, M., Mohamed, M., and Smarandache, F. (2019). A novel method for solving the fully neutrosophic linear programming problems. *Neural Computing and Applications*, 31(5):1595–1605.

[Alvarez Nuñez, 2013] Alvarez Nuñez, M. F. (2013). Teoría de grafos.

[Arquero et al., 2009] Arquero, A., Alvarez, M., and Martinez, E. (2009). Decision management making by ahp (analytical hierarchy process) trought gis data. *IEEE Latin America Transactions*, 7(1):101–106.

[Ávila Torres et al., 2017] Ávila Torres, P. A. et al. (2017). Planificación multiperiodo de las frecuencias de paso y las tablas de tiempo con incertidumbre en la demanda y tiempo de viaje.

[Báez Olvera, 2015] Báez Olvera, M. d. l. Á. (2015). *Apoyo a la decisión para el diseño y la planeación integrados de una cadena de suministro*. PhD thesis, Universidad Autónoma de Nuevo León.

[Bauset et al., 2002] Bauset, S., González, P., Martínez, V. M., and Martínez, B. T. (2002). El mantenimiento de las flotas de transporte. *Técnicas Industriales*, pages 42–47.

[Bell, 1985] Bell, D. E. (1985). Disappointment in decision making under uncertainty. *Operations research*, 33(1):1–27.

[Benavent et al., 1985] Benavent, E., Campos, V., Corberán, A., and Mota, E. (1985). Analisis de heuristicos para el problema del cartero rural. *Trabajos de estadística y de investigación operativa*, 36(2):27–38.

[Benavent et al., 1992] Benavent, E., Campos, V., Corberán, A., and Mota, E. (1992). The capacitated arc routing problem: lower bounds. *Networks*, 22(7):669–690.

[Bermúdez Colina, 2011] Bermúdez Colina, Y. (2011). Aplicaciones de programación lineal, entera y mixta. *Ingeniería Industrial. Actualidad y Nuevas Tendencias*, (7).

[Bertero, 2015] Bertero, F. (2015). Optimización de recorridos en ciudades. una aplicación al sistema de recolección de residuos sólidos urbanos en el municipio de concordia. B.S. thesis, Facultad de Ciencias Exactas, Ingeniería y Agrimensura. Universidad Nacional ....

[Black, 2018] Black, J. (2018). *Urban transport planning: Theory and practice.* Routledge.

[Bryson, 2018] Bryson, A. E. (2018). *Applied optimal control: optimization, estimation and control.* Routledge.

[Calvino Martínez, 2011] Calvino Martínez, A. (2011). Cooperación en los problemas del viajante (tsp) y de rutas de vehículos (vrp): Una panorámica. *Santiago de Compostela: USC Compostela.*

[Calvo González, 2018] Calvo González, M. (2018). Problemas de rutas de vehículos por arcos.

[Castro, 2007] Castro, M. (2007). *Desarrollo e Implementación de un Framework para la Formación de Carteras de Proyectos de I&D en Organizaciones Publicas.* PhD thesis, Tesis de Grado de maestría.

[Charlton and Vowles, 2008] Charlton, C. and Vowles, T. M. (2008). Inter-urban and regional transport.

[Chiavenato and Sapiro, 2017] Chiavenato, I. and Sapiro, A. (2017). *Planeación estratégica*. McGraw-Hill Interamericana.

[Crouzeix et al., 2011] Crouzeix, J. P., Keraghel, A., and Sosa, W. (2011). Programación matemática diferenciable. *Universidad Nacional de Ingenieria, Lima, Peru*.

[Cunquero, 2003] Cunquero, R. M. (2003). Algoritmos heurísticos en optimización combinatoria. *Valencia: Universodad de Valencia. Retrieved*, 11(01):2012.

[Das and Bhattacharyya, 2015] Das, S. and Bhattacharyya, B. K. (2015). Optimization of municipal solid waste collection and transportation routes. *Waste Management*, 43:9–18.

[De Hoyos et al., 2010] De Hoyos, R., Martínez, J. M., and Székely, M. (2010). Educación y movilidad social en méxico. *Movilidad social en México. Población, desarrollo y crecimiento. México DF: Centro de Estudios Espinosa Yglesias*.

[Demirel et al., 2016] Demirel, E., Demirel, N., and Gökçen, H. (2016). A mixed integer linear programming model to optimize reverse logistics activities of end-of-life vehicles in turkey. *Journal of Cleaner Production*, 112:2101–2113.

[Durán Gisbert and Climent, 2017] Durán Gisbert, D. and Climent, G. G. (2017). La formación del profesorado para la educación inclusiva: Un proceso de desarrollo profesional y de mejora de los centros para atender la diversidad.

[Elwyn et al., 2017] Elwyn, G., Durand, M. A., Song, J., Aarts, J., Barr, P. J., Berger, Z., Cochran, N., Frosch, D., Galasiński, D., Gulbrandsen, P., et al. (2017). A three-talk model for shared decision making: multistage consultation process. *bmj*, 359:j4891.

[Evans, 2017] Evans, J. (2017). *Optimization algorithms for networks and graphs*. Routledge.

[Fernández Barberis, 2002] Fernández Barberis, G. (2002). Los métodos promethee: Una metodología de ayuda a la toma de decisiones multicriterio discreta. *Revista Electrónica de Comunicaciones y Trabajos de ASEPUMA*, 1:5–28.

[Gallego et al., 2003] Gallego, R., Romero, R., and Escobar, A. (2003). Algoritmos genéticos. *Texto Guía de la Maestría en Ingeniería Eléctrica. Ed. Universidad Tecnológica de Pereira, Pereira.*

[Garrido-Jurado et al., 2016] Garrido-Jurado, S., Munoz-Salinas, R., Madrid-Cuevas, F. J., and Medina-Carnicer, R. (2016). Generation of fiducial marker dictionaries using mixed integer linear programming. *Pattern Recognition*, 51:481–491.

[González et al., 2001] González, M. P., Pascual, A., and Lorés, J. (2001). Evaluación heurística. *2001). Introducción a la Interacción Persona-Ordenador. AIPO: Asociación Interacción Persona-Ordenador.*

[Gutiérrez, 2012] Gutiérrez, A. (2012). Qué es la movilidad?. elementos para (re) construir las definiciones básicas del campo del transporte. *Bitácora Urbano-Territorial*, 21(2):3.

[Gutiérrez, 2009] Gutiérrez, Ó. P. (2009). Un enfoque multicriterio para la toma de decisiones en la gestión de inventarios. *Cuadernos de Administración*, 22(38).

[Hillier et al., 1997] Hillier, F. S., Lieberman, G. J., and Osuna, M. A. G. (1997). *Introducción a la Investigación de Operaciones*, volume 1. McGraw-Hill.

[Isaza et al., 2004] Isaza, R. A. H., Rendón, R. A. G., and Porras, C. A. R. (2004). Técnicas heurísticas aplicadas al problema del cartero viajante (tsp). *Scientia et technica*, 1(24):1–6.

[Izquierdo and María, 2008] Izquierdo, C. and María, J. (2008). Estudios sobre movilidad cotidiana en méxico. *Scripta Nova*, 12(273).

[Jimenez et al., 2017] Jimenez, C., Cid, H. A., and Figueroa, I. (2017). Prometheus: Procedural methodology for developing heuristics of usability. *IEEE Latin America Transactions*, 15(3):541–549.

[Le Grand, 2018] Le Grand, J. (2018). *The strategy of equality: redistribution and the social services*. Routledge.

[López et al., 1983] López, E. B., Aucejo, V. C., Salvador, Á. C., and Vidal, E. M. (1983). Problemas de rutas por arcos. *Qüestiió: quaderns d'estadística i investigació operativa*, 7(3):479–490.

[Lourenzo, 2001] Lourenzo, H. (2001). A beginner's introduction to iterated local search.

[Lüer et al., 2009] Lüer, A., Benavente, M., Bustos, J., and Venegas, B. (2009). El problema de rutas de vehículos: Extensiones y métodos de resolución, estado del arte. In *EIG*.

[Luna et al., 2016] Luna, A. C., Diaz, N. L., Graells, M., Vasquez, J. C., and Guerrero, J. M. (2016). Mixed-integer-linear-programming-based energy management system for hybrid pv-wind-battery microgrids: Modeling, design, and experimental verification. *IEEE Transactions on Power Electronics*, 32(4):2769–2783.

[Luo et al., 2017] Luo, Z., Qi, L., and Xiu, N. (2017). The sparsest solutions to z-tensor complementarity problems. *Optimization letters*, 11(3):471–482.

[Marler and Arora, 2010] Marler, R. and Arora, J. (2010). The weighted sum method for multi-objective optimization: New insights. *Structural and Multidisciplinary Optimization*, 41:853–862.

[Mashayekh et al., 2017] Mashayekh, S., Stadler, M., Cardoso, G., and Heleno, M. (2017). A mixed integer linear programming approach for optimal der portfolio, sizing, and placement in multi-energy microgrids. *Applied Energy*, 187:154–168.

[Mauttone et al., 2010] Mauttone, A., Cancela, H., and Urquhart, M. (2010). Diseño y optimización de rutas y frecuencias en el transporte colectivo urbano, modelos y algoritmos. *Universidad de la República Facultad de Ingeniería, Uruguay.*

[Medina et al., 2011] Medina, L. B. R., La Rotta, E. C. G., and Castro, J. A. O. (2011). Una revisión al estado del arte del problema de ruteo de vehículos: Evolución histórica y métodos de solución. *Ingeniería*, 16(2):35–55.

[Melián et al., 2003] Melián, B., Pérez, J. A. M., and Vega, J. M. M. (2003). Metaheurísticas: Una visión global. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 7(19):0.

[Mizuno et al., 1993] Mizuno, S., Todd, M. J., and Ye, Y. (1993). On adaptive-step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations research*, 18(4):964–981.

[Montoro et al., 2018] Montoro, L., Useche, S., Alonso, F., and Cendales, B. (2018). Work environment, stress, and driving anger: a structural equation model for predicting traffic sanctions of public transport drivers. *International journal of environmental research and public health*, 15(3):497.

[Narváez Fuentes, 2005] Narváez Fuentes, R. A. (2005). Diseño de un plan de mantención preventiva para camiones de multiaseo.

[Nemati et al., 2018] Nemati, M., Braun, M., and Tenbohlen, S. (2018). Optimization of unit commitment and economic dispatch in microgrids based on genetic algorithm and mixed integer linear programming. *Applied energy*, 210:944–963.

[Olivera, 2004] Olivera, A. (2004). Heurísticas para problemas de ruteo de vehículos. *Reportes Técnicos 04-08*.

[Patriksson, 2015] Patriksson, M. (2015). *The traffic assignment problem: models and methods*. Courier Dover Publications.

[Payá Zaforteza, 2009] Payá Zaforteza, I. J. (2009). *Optimización heurística de pórticos de edificación de hormigón armado*. PhD thesis.

[Perez, 2014] Perez, J. (2014). El problema del cartero chino.

[RAMON and ROMERO,] RAMON, A. G. R. and ROMERO, L. Statical planning of colombia's transmission system using genetics algorithm.

[RESENDE and RIBEIRO, 2002] RESENDE, M. and RIBEIRO, C. (2002). Greedy ramdomized adaptative search procedure. *Handbook of metaheuristics*.

[Resende and Velarde, 2003] Resende, M. G. and Velarde, J. L. G. (2003). Grasp: Procedimientos de búsqueda miopes aleatorizados y adaptativos. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 7(19):0.

[Rodriguez et al., 2012] Rodriguez, D. A., Olivera, A. C., and Brignole, N. B. (2012). Una estrategia paralela con simulated annealing para el problema del transporte público interurbano.

[Roy, 2013] Roy, B. (2013). *Multicriteria methodology for decision aiding*, volume 12. Springer Science & Business Media.

[Salas, 2009] Salas, H. G. (2009). *Programación lineal aplicada*. Ecoe Ediciones.

[Serna and Uran, 2015] Serna, M. D. A. and Uran, C. A. S. (2015). A memetic algorithm for the traveling salesman problem. *IEEE Latin America Transactions*, 13(8):2674–2679.

[Silva Clavería and Silva Clavería, 2004] Silva Clavería, A. and Silva Clavería, R. (2004). Heurística: Origen y consecuencias.

[Stiggelbout et al., 2015] Stiggelbout, A. M., Pieterse, A. H., and De Haes, J. C. (2015). Shared decision making: concepts, evidence, and practice. *Patient education and counseling*, 98(10):1172–1179.

[Stützle, 1998] Stützle, T. (1998). Applying iterated local search to the permutation flow shop problem. Technical report, Technical Report AIDA-98-04, FG Intellektik, TU Darmstadt.

[Torres Fernández et al., 2017] Torres Fernández, J. P., Gallo Mendoza, J. G., Hallo Alvear, R. F., Abcarius, J. J., Páez, M., Humberto, M., and Fernández Lorenzo, A. (2017). Gestión de la información como herramienta para la toma de decisiones en salud: escenarios más probables. *Revista Cubana de Investigaciones Biomédicas*, 36(3):0–0.

[Vanderbei et al., 2015] Vanderbei, R. J. et al. (2015). *Linear programming*. Springer.

[Vigerske and Gleixner, 2018] Vigerske, S. and Gleixner, A. (2018). Scip: Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework. *Optimization Methods and Software*, 33(3):563–593.

[White, 2016] White, P. R. (2016). *Public transport: its planning, management and operation*. Routledge.

[Yin et al., 2017] Yin, J., Yang, L., Tang, T., Gao, Z., and Ran, B. (2017). Dynamic passenger demand oriented metro train scheduling with energy-efficiency and waiting time minimization: Mixed-integer linear programming approaches. *Transportation Research Part B: Methodological*, 97:182–213.

[Zanakis and Evans, 1981] Zanakis, S. H. and Evans, J. R. (1981). Heuristic "optimization": Why, when, and how to use it. *Interfaces*, 11(5):84–91.

[Zhang et al., 2016] Zhang, Y., Sun, X., and Wang, B. (2016). Efficient algorithm for k-barrier coverage based on integer linear programming. *China Communications*, 13(7):16–23.

Resumen autobiográfico

Evely Gutiérrez Noda

Candidato para obtener el grado de
Maestria en Ciencias en Ingeniería
de Sistemas

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

MODELOS DE OPTIMIZACIÓN PARA EL PROBLEMA DE INSPECCIÓN DE RUTAS DE CAMIONES INTERURBANOS

Aquí va tu historia. Recuerda que debe incluir: lugar y fecha de nacimiento, nombre de los padres, escuelas y universidades en las que se graduó después de la preparatoria, títulos o grados obtenidos (no incluir los estudios que se están concluyendo), experiencia profesional y organizaciones profesionales a las que pertenece (no incluir lista de publicaciones).