



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

Programación multiobjetivo en el transporte del
Servicio Ordinario del Sistema de Movilidad 1 en
la Ciudad de México

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Actuario

PRESENTA:

Jorge Armando Serna Mendoza

TUTORA

Dra. Bibiana Obregón Quintana

Ciudad Universitaria, Cd. Mx., 2019





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

RESUMEN

El presente trabajo pretende modelar la red del transporte público del Sistema de Movilidad 1 (Sistema M1), en la zona de la Ciudad de México (CDMX), como una red de transporte para el servicio llamado “Ordinario”, es decir, proponer un modelo matemático multiobjetivo con el fin de optimizar el servicio. Así el modelo busca maximizar el flujo de usuarios dentro de la red, esto es, maximizando la utilidad generada del sistema y minimizando los tiempos de traslado para los usuarios.

ÍNDICE GENERAL

Introducción	7
1. Transporte	9
1.1. Historia del Transporte en la Ciudad de México	9
1.2. Actual y Futuro de Sistema M1	11
2. Redes	15
2.1. Definiciones Básicas	15
2.2. Conceptos Avanzados	19
2.3. Problema de Transporte	23
3. Programación Lineal y Programación Multiobjetivo	29
3.1. Antecedentes	29
3.2. Programación Lineal	30
3.2.1. Transporte	33
3.3. Programación Multiobjetivo	34
3.3.1. Modelo general de programación lineal multiobjetivo	35
3.3.2. Planteamiento de Transporte	37
3.3.3. Problemas multiobjetivo y métodos Heurísticos	44
4. Técnicas heurísticas	47
4.1. Definiciones básicas	47
4.2. Recocido Simulado	48

4.3. Reencadenamiento de Trayectorias	51
4.4. Búsqueda Armónica	53
4.5. Medidas de calidad de un algoritmo	55
5. Metodología y Resultados	57
Discusión y Conclusiones	71
Anexos	79

INTRODUCCIÓN

El trasladarse de un lugar a otro es una necesidad indispensable para la humanidad, que a través del tiempo se ha convertido en un problema en zonas con una alta concentración de personas debido a las actividades diarias, como son; trabajar, estudiar, entre otras. La saturación de las vialidades provocada por la concentración de personas ocasiona problemas de embotellamientos y tiempos de traslado tardados.

La Ciudad de México (CDMX) presenta severos problemas de transporte desde hace tiempo, debido al constante crecimiento de la urbe y a las nuevas actividades dentro de esta. Lo anterior ha provocado un gran interés científico, político y público para implementar políticas públicas de transporte que permitan a la red de transporte, transportar la máxima cantidad de personas posible.

A partir del siglo XXI, más del 77 % de la población en México reside en ciudades, la mayoría de esta en la CDMX, lo cual ha ocasionado que haya un crecimiento poblacional en algunas áreas de la ciudad y que haya una gran dispersión de localidades alejadas de los centros de trabajo. Por ello se creó el Sistema de Movilidad 1 (Sistema M1) de la CDMX, con el fin de trasladar a los residentes de zonas periféricas hacia los centros de trabajo, estudio, entre otros.

Por otra parte, el estudio de la teoría de redes ha permitido estudiar el transporte y conocer sus propiedades, este tipo de sistemas reales se pueden representar a través de redes complejas, debido a que están compuestos por una gran cantidad de elementos que interactúan entre sí y que presentan características propias de los sistemas complejos. En teoría de redes, una red se conforma por un conjunto de componentes (*nodos*) y una relación específica entre ellos (*arcos*).

Cuando un sistema de transporte se modela como una red, los nodos representan lugares geográficos (estaciones, paradas, terminales, etc.) y los arcos representan la ruta que conecta un lugar con otro (calles, vías, avenidas, caminos, etc.).

También, la investigación de operaciones se encarga de estudiar este tipo de problemas, aplicando metodologías que buscan optimizar recursos con ayuda de herramientas matemáticas, es decir, encontrar a través de un modelo matemático una solución que sea buena entre las soluciones posibles del problema.

Cabe destacar la importancia de la relación entre la investigación de operaciones con el estudio de la teoría de redes, esto es, conseguir la representación de sistemas reales mediante una red y poder llevar a cabo la aplicación de algoritmos de investigación de operaciones para resolver dicho problema.

Con base en lo anterior, se define como objetivo principal de esta tesis, proponer un modelo matemático del sistema de transporte del servicio Ordinario de Sistema M1 en la CDMX como una red, que encuentre soluciones que mejoren la problemática existente de asignación de autobuses por ruta. Dicho modelo busca maximizar la utilidad del servicio, minimizar los tiempos de traslado y el incumplimiento de satisfacer toda la demanda de usuarios. Algunos objetivos secundarios son, caracterizar la red de transporte del servicio Ordinario del Sistema M1 mediante un grafo, medir propiedades de la red y encontrar características de ésta.

Para lograr el objetivo, este trabajo está organizado como sigue:

1. **Capítulo 1.** Descripción de antecedentes y del estado actual del sistema de transporte de la CDMX.
2. **Capítulo 2.** Definición de conceptos básicos y avanzados de redes para la elaboración de la red de transporte y representación de la red.
3. **Capítulo 3.** Definición de conceptos de programación lineal, conceptos de programación multiobjetivo y de las variables de este trabajo, para la elaboración del modelo matemático.
4. **Capítulo 4.** Aplicación de técnicas heurísticas para encontrar la solución del modelo y comparación entre algoritmos como medida de calidad de algoritmos.
5. **Capítulo 5.** Metodología de este trabajo, seguido de la descripción de los resultados obtenidos.

CAPÍTULO 1

TRANSPORTE

En este capítulo se mencionan antecedentes históricos [1], junto con los cambios más relevantes del sistema de transporte de la actual CDMX, empezando desde la época prehispánica conformada por cuatro calzadas principales, luego la expansión hacia las periferias de la ciudad que propició la implementación de sistemas de transporte, como el Autotransporte Urbano de Pasajeros R-100, hasta llegar a conformar Sistema M1.

1.1. Historia del Transporte en la Ciudad de México

En la época prehispánica, Tenochtitlán, ahora Ciudad de México, era una ciudad bien conformada, con sistemas de tuberías, drenaje y de transporte, contaba con cuatro calzadas principales que apuntaban a los cuatro puntos cardinales. La primera, la calzada a Iztapalapa y Xochimilco, la segunda, la calzada a Tlacopan y Tacuba, que se dirigía al poniente, la tercera corría hacia el norte por la actual calle de Argentina y la cuarta, la calzada a Tepeyac, que iba al oriente por la actual calle de Guatemala.

Con la llegada de los españoles en el año 1519 y tras la conquista en 1521, se comenzó el inicio de la construcción de la nueva ciudad llamada Nueva España, antiguamente Tenochtitlán, dando así inicio al nuevo trazo urbanístico, conservando las cuatro calzadas establecidas. En el año 1522, se construyeron las primeras calles; Pino Suárez, Tacuba, Madero, Brasil, Argentina, Guatemala, Moneda, Corregidora, 5 de

Febrero y 16 de Septiembre. Durante el Virreinato de la Nueva España comprendido entre los años 1522 y 1810, el transporte se realizaba principalmente en caballos, mulas y carretas.

Con una población creciente, a principios del siglo XIX, la ciudad contaba con 160 mil habitantes y circulaban gran cantidad de vehículos. En el año de 1824, la geometría urbana había cambiado, aparecieron las primeras glorietas y cruceros, aunque la intensa actividad de autos seguía ocasionando problemas de vialidad. Para atender el problema, en 1830, se expidió el primer Reglamento de Tránsito, expedido por el gobierno de la capital, en el cual se establecían límites de velocidad y pretendía tener control sobre las tarifas y el uso de los vehículos.

En 1859, se introdujo en México el primer vehículo impulsado por energía eléctrica, conocido como tranvía, el cual se modernizó paulatinamente, hasta que en 1894 se le instaló un sistema de hilo aéreo. Hacia el año 1885, la Ciudad contaba con 300 mil habitantes y el desarrollo de vías públicas dentro del perímetro era de 13 mil 500 metros, los cuales sumaban 270 kilómetros.

En 1910, más del 70 % de la población en México vivía en medios rurales, más de un siglo después, más del 77 % de la población reside en las ciudades, lo que ha provocado una alta concentración poblacional en algunas áreas y una gran dispersión en localidades alejadas de los centros de trabajos. Aunado a lo anterior, el cambio en el uso del suelo provoca que las distancias entre viviendas y centros de trabajo sean más largas con el avance del tiempo.

En la década de 1980, la infraestructura del Sistema de Transporte Colectivo, mejor conocido como *Metro* y los ejes viales, provocó que se necesitara integrar un sistema de transporte urbano, por lo que se le dio al gobierno del Distrito Federal la responsabilidad de prestar el servicio de transporte de autobuses, así surgió Autotransporte Urbano de Pasajeros R-100, pero aún no era suficiente para cubrir la demanda creciente de usuarios. Por tales motivos, en el año 2000, se publicó en la Gaceta Oficial del Distrito Federal el decreto por el que se creó la Red de Transporte de Pasajeros del Distrito Federal, (RTP) como un organismo público. RTP inició operaciones a partir del primero de marzo del año 2000, con el fin de acercar a los residentes de

zonas periféricas hacia los centros de trabajo.

El cambio de nomenclatura, en el año 2016, de la Red de Transporte de Pasajeros (RTP) del Distrito Federal a Sistema de Movilidad 1 (Sistema M1) de la CDMX, se debe al reconocimiento del derecho humano de movilidad y a la modernización de la planta vehicular, que tiene como fin impulsar la renovación, expansión y reconfiguración del transporte.

1.2. Actual y Futuro de Sistema M1

El Sistema M1 cuenta actualmente con cinco diferentes tipos de servicio de transporte; servicio Ordinario, el cual brinda servicio de transporte en 91 de las 94 rutas de la red de transporte del Sistema M1, siendo el servicio que más rutas recorre de los cinco, contando con 599 unidades de 14 tipos distintos de autobuses. Los 14 tipos de autobús con los que cuenta el servicio Ordinario se muestran en la figura 1.1.



CDMX
CIUDAD DE MÉXICO



**SISTEMA
DE MOVILIDAD 1**
CDMX

CONFORMACIÓN DEL PARQUE VEHICULAR SERVICIO ORDINARIO

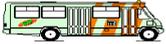
TIPO AUTOBÚS	TOTAL DE AUTOBUSES	MODELO	CAPACIDAD DE PASAJEROS SENTADOS/ DE PIE
AYCO 3000 RE 	70	2001 EPA 98	28 / 72 (100)
 AYCO DISCAP.	9	2002 EPA 98	21 / 79 (100)
AYCO 30030 RE 	29	2002 EPA 98	28 / 57 (85)
 AYCO 30030 SUSP. DEL. MEC.*	11	2003 EPA 98	28 / 57 (85)
RECO 4700 SFC 	2	2001 EPA 98	25 / 45 (70)
 TORINO 2002	72	2002 EURO III	28 / 62 (90)
TORINO 2004 	70	2004 EURO III	28 / 62 (90)
 TORINO 2006	115	2006 EURO III	28 / 62 (90)
TORINO 2009 	3	2009 EURO IV	28 / 62 (90)
TORINO 2006 EQUIP. 	17	2006 EURO III	28 / 62 (90)
AYCO M-B COSMOPOLITAN 	41	2016 EURO V	28 / 62 (90)
 AYCO M-B COSMO. C/RAMPA	9	2016 EURO V	28 / 62 (90)
VOLVO PROCITY DIESEL 	147	2016 EURO V	31 / 69 (100)
 VOLVO PROCITY C/RAMPA	4	2016 EURO V	31 / 69 (100)
TOTAL	599		

Figura 1.1: Conformación del parque vehicular del Servicio Ordinario

Fuente: Sistema M1 (2018)

Los otros cuatro servicios de transporte de Sistema M1 son; servicio Expreso, el cual consiste en hacer parada en algunas estaciones de las 23 rutas que recorre; servicio Atenea, que ofrece servicio exclusivo al sexo femenino en los principales corredores viales de la CDMX, recorriendo 50 rutas; servicio Ecobús, que consiste en usar un combustible más amigable con el medio ambiente, recorriendo 2 rutas; y el servicio Nochebús, que brinda servicio en las noches de 00:00 a 05:00 horas, en 7 rutas. Los servicios como Ordinario, Expreso, Atenea y Ecobús brindan un servicio de 06:00 a 23:00 horas en días hábiles.

En 2018, el costo de transporte para los usuarios de Sistema M1 era de; \$2.00 MXN para el servicio Ordinario y el servicio Atenea, \$4.00 MXN para el servicio Expreso, \$5.00 MXN para el servicio Ecobús y \$7.00 MXN para el servicio Nochebús [2] (Gobierno de la Ciudad de México 2018).

La Gerencia de Tesorería y Recaudación del Sistema M1 reportó una recaudación de 129,709,567 pesos durante el año 2017, transportando más de 62 millones de personas.

En resumen, actualmente (al 31 de diciembre de 2018) el Sistema Ordinario del Sistema M1:

- Recorre noventa y un rutas en 3,067.5 km.
- Cuenta con quinientos noventa y nueve autobuses.
- Recaudación de 129,706,567 pesos en el año 2017.
- 62,592,707 usuarios en el año 2017.

Actualmente (al 31 de diciembre de 2018), el sistema de transporte de la CDMX, presenta problemas como: el crecimiento exponencial en el uso de automóviles privados, el descuido político del parque vehicular, la falta de cultura en transporte público por parte de los usuarios y una infraestructura deficiente e inadecuada de éste [3]. Las causas del problema de movilidad en la CDMX son múltiples y complejas, tales como; la gran cantidad de personas que requieren moverse dentro y en las periféreas de la CDMX, ineficiencia en el flujo y movimiento de vehículos y personas,

un sistema de transporte descuidado que se volvió insuficiente, el comportamiento de los usuarios y conductores [4].

Además, los rápidos cambios en la urbanización y motorización en las megaciudades tienen un impacto directo en la calidad de vida de los habitantes, donde el sistema de transporte está integrado por usuarios, sistemas de transporte (como metro, automóviles, autobuses, red de transporte de pasajeros, etc.), zonas de trabajo y viviendas y medio ambiente, entre otros [5]. Por ello, la interacción entre los usuarios de transporte público y conductores de automóviles privados causa problemas en el flujo de la comunicación, estrés y descontento. Además, un sistema de transporte es eficiente si su construcción y operación considera su impacto en los residentes, el uso de la tierra, el medio ambiente y el desarrollo económico [6].

Por otro lado, el Sistema M1 tiene propuestas a futuro para hacer cambios dentro del organismo. La Gaceta Oficial de la Ciudad de México publicada el día 31 de diciembre del año 2017, publicó que para el Sistema M1, se destinarían 5 millones de pesos para la puesta en marcha de una nueva ruta que circularía de la estación del metro Miguel Ángel de Quevedo al Pueblo de la Magdalena Atlética.

También, durante el año 2017, fueron adquiridos nuevos autobuses para fortalecer el servicio de transporte del Sistema M1, sin embargo, el número de autobuses no es suficiente para cubrir la demanda en determinados horarios, por lo que propuso a futuro el arrendamiento de cincuenta autobuses, a fin de apoyar el servicio Ordinario de transporte del Sistema M1. El proyecto no ha sido aprobado (al 31 de diciembre de 2018) ya que falta determinar si es rentable el arrendamiento de los cincuenta autobuses, considerando que se deben de estimar todos los gastos correspondientes como: salario de los operadores, costo del combustible, costo del mantenimiento de las unidades, costo del arrendamiento, entre otros.

CAPÍTULO 2

REDES

En este capítulo se definen los conceptos de teoría de redes que se necesitan para desollar esta tesis, primero se definen conceptos básicos, después se definen conceptos avanzados, y por último, se define el problema de transporte.

2.1. Definiciones Básicas

El estudio de la teoría de redes ha permitido conocer propiedades de una gran cantidad de sistemas, contribuyendo al desarrollo y conocimiento sobre éstos, por ejemplo; el Internet, redes sociales, sistemas en el ámbito de la medicina, sistemas biológicos (redes genéticas, sistemas neuronales), entre otros [7]. Dichos sistemas pueden ser representados a través de *redes*, formadas por un conjunto de componentes (nodos) y una relación específica entre ellos (arcos).

La teoría de gráficas tiene sus inicios con el problema de los *puentes de Königsberg*, Prusia (actualmente Kaliningrado, Rusia). El problema consistía en pasar por los siete puentes una única vez y recorrerlos todos. El matemático Leonard Euler, dio respuesta al problema en 1736, sin embargo, lo más destacado fue que logró la representación gráfica de una red mediante puntos, los cuales representaban zonas de tierra y líneas, que representaban los puentes que unían las zonas de tierra, ver figura 2.1.

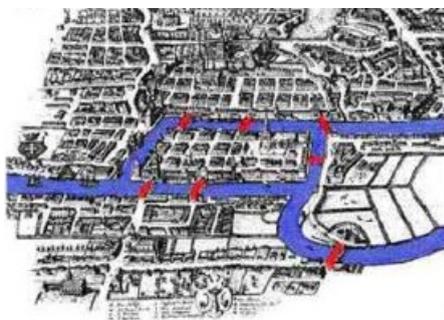


Figura 2.1: Puentes de Königsberg [8]

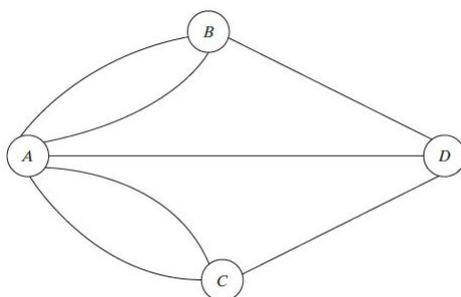


Figura 2.2: Representación en forma de red del problema de Königsberg [9]

Los estudios de la teoría de redes siguieron avanzando con trabajos como los de Ray Solomonoff y Anatol Rapoport, quienes en 1951, describieron inicialmente las redes complejas como redes aleatorias. La aportación más importante fue que conforme se incrementaban los arcos en una red, el grafo cambiaba de un conjunto de nodos desconectados a un grafo con la mayoría de nodos conectados, es decir, se formaba un componente gigante [10]. También, los grafos aleatorios fueron estudiados por Paul Erdős y Alfréd Rényi, quienes publicaron una serie de ocho artículos entre los años 1959 y 1968, en los cuales se dio origen formal al estudio de las redes complejas. Además, demostraron que en el momento de formación del componente gigante, surgen muchas propiedades de los grafos [11].

Por otro lado, el estudio de las redes sociales de Jeffrey Travers y Stanley Milgram en 1969, motivó el concepto de mundo pequeño [12]. El concepto de mundo pequeño trata de que la mayoría de los nodos en una red, están conectados por una ruta corta, que se incrementa de forma logarítmica con el número de nodos en la red [12], es decir,

$$l \sim \log N.$$

Una red R está compuesta por un conjunto de *nodos* (o vértices) N_1, N_2, \dots, N_k y por un conjunto de *arcos* (o aristas) a_1, a_2, \dots, a_n que representan la relación entre el conjunto de nodos. La notación para describir una red es $R = \{N, A\}$, donde N es el conjunto de nodos y A el conjunto de arcos.

Ejemplo: *Puentes de Königsberg*, véase figura 2.2

$$N = \{A, B, C, D\}$$

$$A = \{(A,B), (A,C), (A,D), (B,A), (B,D), (C,A), (C, D)\}$$

Una red R se dice que es una red dirigida cuando la red se conforma de arcos con dirección, ver figura 2.3.

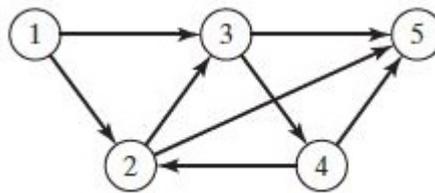


Figura 2.3: Ejemplo de una red dirigida [9]

Se denota la *dirección* D_{ij} de los arcos como, $i \rightarrow j$, cuando el sentido de la dirección va del nodo i al nodo j , en la figura 2.3 se pueden apreciar algunas direcciones como, $1 \rightarrow 2$, $2 \rightarrow 3$, $4 \rightarrow 2$, etc.

Otro concepto es el de *arcos paralelos*, si en un grafo existe más de un arco que conecta al nodo i con el nodo j , se dice que el grafo tiene arcos paralelos o multiarcos. Ambos tipos de relaciones se pueden presentar en redes dirigidas y no dirigidas.

El *grado* de un nodo k_j en una red no dirigida, se refiere a la cantidad de nodos a los que es adyacente el nodo j a través de un arco (i, j) . Por otro lado, si la red es dirigida, el nodo j tiene dos tipos de grado, el grado interior, que es el número de arcos que tienen dirección de cualquier nodo i hacia el nodo j y el grado exterior que es el número de arcos que tienen dirección desde el nodo j hacia cualquier nodo i [13]. Además, cuando la red es no dirigida, no se puede realizar la separación de grados interior y exterior.

En la figura 2.2, para el arco (A, B) , por no tener una única dirección, la notación (D_{AB}, D_{BA}) denota las direcciones $A \rightarrow B$ y $B \rightarrow A$, para quitar la ambigüedad, se coloca D_{AB} junto a el arco (A, B) y en la figura 2.3, para el arco $(1, 3)$, por ser dirigido, se coloca junto a él la notación (D_{13}) , para denotar la dirección 1 al nodo 3, en la dirección $1 \rightarrow 3$.

En una red no dirigida, la *relación nodo-nodo* generalmente es representada por una matriz llamada *matriz de adyacencia* A_{ij} , donde la entrada a_{ij} vale 1 cuando existe el arco (i, j) , es decir, existe la relación que une al nodo i y al nodo j y 0 cuando no existe el arco. Además, el grado de un nodo i se puede calcular como:

$$k_i = \sum_{j=1}^N A_{ij}, \quad \text{donde } N \text{ es la cantidad de nodos en la red.} \quad (2.1)$$

Por otro lado, si se grafican los grados de los nodos k_j contra las *frecuencias* de los grados, es decir, cuantos nodos hay en la red con el mismo grado, se obtiene la *distribución de grado* de la red P_k . En general, la estructura de una red, puede entenderse mediante la distribución de grado, es decir, a partir de identificar cómo se están relacionando los nodos dentro de la red y qué tan conectada está [14], dicha medida se explicará en la siguiente sección. También, cuando un nodo tiene muchas incidencias de arcos, respecto a los demás nodos en la red, se dice que el nodo es un *nodo hub* de la red [9].

Un *camino* es una sucesión ordenada de arcos que une a dos nodos distintos, es decir, es una sucesión alternada y ordenada entre nodos y arcos, y una *ruta* es un camino que visita una única vez un nodo [9]. Por ejemplo, en la figura 2.3 se muestra que los arcos $(1, 3)$, $(3, 4)$ y $(4, 5)$ forman una ruta entre los nodos 1 y 5. Una ruta forma un *ciclo* o (bucle) si conecta un nodo de vuelta a sí mismo a través de otros nodos. Por ejemplo, en la figura 2.3 se muestra que los arcos $(2, 3)$, $(3, 4)$ y $(4, 2)$ forman un ciclo.

A cada nodo $i \in R$ se asocia un número b_i , que representará los recursos (si $b_i > 0$), o la demanda requerida (si $b_i < 0$). Por lo general, los nodos que cumplen $b_i > 0$, se les llama *orígenes* y a los nodos que cumplen $b_i < 0$, se les llama *destinos*. Si $b_i = 0$,

entonces no hay recursos en el nodo i y ninguna demanda se requiere, en este caso se dice que el nodo i es un nodo *intermedio* (o de *transbordo*).

2.2. Conceptos Avanzados

Hacia el siglo XX, se presenta el inicio en el estudio de las *redes complejas*, la motivación para el estudio de este tipo de redes, fue que las redes no permanecen estáticas sino que cambian a través del tiempo [10].

Las redes complejas se describieron en sus inicios como *gráficas aleatorias* por Ray Solomonoff y Anatol Rapoport en 1951, quienes vieron que si se agregaban más arcos a una red, la red estaría mejor conectada, es decir, la red cambiaba de un conjunto con componentes de nodos desconectados a una red que conectaba a la mayoría de los nodos, a esto se le nombró *componente gigante* [10]. Por otro lado, Paul Erdős y Alfréd Rényi introdujeron el *modelo de gráficas aleatorias* en 1960, donde lo más destacado que probaron fue que, se puede trabajar sólo con el componente gigante y dejar a un lado los componentes aislados [11].

Existen dos tipos de perturbaciones que pueden afectar a una red, la primera es interna y se refiere a fallas dentro de la red, es posible representarlás como la extracción de nodos de forma aleatoria, a esta perturbación se le conoce como *error*. A la segunda se le conoce como *ataque* y consiste en la eliminación intencional de nodos, siendo éste un evento externo a la red [15].

El daño provocado en la red se mide por el tamaño del componente gigante después de algún tipo de perturbación, esto es, si el comportamiento no disminuye de forma significativa, se dice que el daño no es grave. Además, si después de la perturbación la red conserva al componente gigante, entonces se puede decir que la red es *robusta* [9].

Cuando se tiene el componente gigante de una red, dicho componente se puede ver afectado ante circunstancias internas o externas de la red. La robustez es un indicador que mide la habilidad de que el sistema complejo conserve un comportamiento relativamente normal, es decir, la capacidad de la red de mantener la información

que proporciona el componente gigante [7, 15, 16]. Por ejemplo, si la red representa una red eléctrica, se desea saber si la red es capaz de soportar la pérdida de arcos sin que el flujo se interrumpa, si el flujo no se interrumpe con la pérdida de varios arcos, se dice que la red es bastante robusta.

Por otro lado, el estudio de las redes sociales de Travers y Milgram, motivó a realizar un experimento que consistía en seleccionar un grupo de personas, a los que se les envió una carta de instrucciones, pidiéndoles contactar a una persona específica. Los participantes debían conocer de forma directa a la persona para enviarle la carta, de lo contrario, debían buscar otro destinatario que sí conocieran y pensarán que podía conocer a la persona destino. Luego, debían de enviar su respuesta a Travers y Milgram, a fin de que estos pudieran reconstruir la ruta. Participaron 296 personas, 196 de Omaha, Nebraska y 100 de Boston, Massachusetts, y la persona destino era de Sharon, Massachusetts. El resultado fue: 64 de las 296 personas lograron una ruta con éxito, esto es, 21% de las personas que iniciaron. El número promedio de intermediarios fue 5.2, este resultado motivó la popular frase *seis grados de separación* [12].

Una métrica para entender características de una red es la *ruta más corta* (d_{ij}), que consiste en encontrar la distancia mínima de un nodo origen i a un nodo destino j . Además, el concepto de mundo pequeño consiste en que la mayoría de los nodos que no son vecinos entre sí, puedan ser alcanzados desde cualquier nodo origen a través de una ruta corta [9].

Otro concepto importante para definir la estructura de una red es el de distribución de grado, que para una red no dirigida, se define P_k como la probabilidad de elegir un nodo de grado k aleatoriamente, y la distribución de grado, es el histograma de los grados de los nodos contra la cantidad de nodos con el mismo grado. Entonces, si se grafican frecuencias contra el grado, la distribución muestra cómo se distribuyen los enlaces, por ejemplo: si el histograma tiene *sesgo* a la derecha, quiere decir que hay pocos nodos con un alto grado de conectividad, los llamados hubs, y si el histograma tiene sesgo a la izquierda se supone que la red tiene un mayor número de nodos con un alto grado de conectividad [17].

El histograma puede realizarse con escalas logarítmicas en ambos ejes, de esta forma puede observarse a la distribución de grado (P_k) como una línea recta, es decir,

$$\ln(P_k) = -\theta \ln(k) + c,$$

entonces el logaritmo de la distribución de grado (P_k), es una función lineal de grado, donde θ y c son constantes (el signo menos es conveniente debido a la negatividad de la pendiente).

Si bien la distribución de grado de una red permite conocer cuánto se conecta toda la red, no proporciona más información relevante, por lo que existen otras medidas importantes que se dedican a caracterizar una red compleja, como la *distancia media* (l), esta característica permite conocer qué tan rápido es llegar entre cualesquiera dos nodos en la red, es decir, la cantidad promedio de arcos que se necesitan recorrer para llegar entre cualesquiera dos nodos, dada por:

$$l = \frac{1}{\frac{1}{2}N(N+1)} \sum_{i \geq j} d_{ij}, \quad (2.2)$$

donde N es el número de nodos en toda la red [14].

Por otro lado, la *eficiencia local* (ε_{ij}) entre un par de nodos i y j , mide el intercambio de información en una red y es inversamente proporcional a la ruta más corta. Cuando no existe una ruta entre los nodos i y j , entonces:

$$d_{ij} = +\infty,$$

significa que $\varepsilon_{ij} = 0$ [18].

También se puede definir la *eficiencia global* como [18]:

$$E(G) = \frac{1}{N(N-1)} \sum_{i \neq j \in G} \frac{1}{d_{ij}}.$$

En conjunto se puede analizar que la robustez está relacionada con el componente gigante de forma directa, es decir, si la red es robusta, el componente gigante se mantiene, más aún, la ruta más corta entre pares de nodos, también se relaciona directamente, por lo que si la red no es tolerante, se puede afectar la distancia entre pares de nodos, por lo tanto si la red no es robusta, tampoco es eficiente [19].

La eficiencia puede ser *local* o *global*, cuando se considera la eficiencia local, se considera la primera vecindad del nodo específico, pero sin tomar en cuenta el nodo en cuestión y cuando es global considera a toda la red. La eficiencia global se utiliza en *sistemas paralelos*, cuando el envío de información es simultáneo entre los nodos, en cambio la eficiencia local, se utiliza en *sistemas secuenciales*, es decir, cuando sólo hay envío de información de un nodo en específico.

Otra medida utilizada es el *coeficiente de aglomeración* (C) o *clustering*, es una medida que se refiere al cociente entre el número de arcos de los vecinos del nodo i y el número máximo de posibles arcos entre los vecinos del nodo i , lo que indica la tendencia a formar triángulos dentro de la red y se calcula como:

$$C = \frac{1}{N} \sum_{i=1}^N C_i = \frac{1}{N} \sum_{i=1}^N \frac{2E_i}{p_i(p_i - 1)}, \quad (2.3)$$

donde E_i es el número de arcos entre los *vecinos* del nodo i y $p_i(p_i - 1)_{\frac{1}{2}}$ es el número máximo de la arcos con adyacencia probable entre los *vecinos* del nodo i [14]. También se le conoce como *propiedad de transitividad*.

En 1998, Duncan Watts y Steven Strogatz proponen un modelo de grafos, el cual se emplea para la construcción de algunas redes de mundo pequeño. El modelo consiste en la generación de grafos aleatorios con distancias medias pequeñas y valores altos del coeficiente de agrupamiento. El modelo presenta una desventaja, la cual consiste en que durante el proceso de reconexión de arcos, existe la posibilidad de separar el grafo original, esto se debe a que únicamente se reconecta el extremo de los arcos [20].

En 1999, Albert-László Barabási y Réka Albert en su interés de representar sistemas reales, establecieron la llamada *preferencia de enlace*, que consiste en añadir a la red

un nodo nuevo con n -arcos y conectarlo a los nodos existentes dependiendo del grado de dichos nodos. Así este modelo genera *redes de libre escala*, las cuales, como su nombre lo dice, no siguen una distribución probabilística [14].

Respecto a la conectividad, esta permite hacer clasificaciones más específicas en una red, es decir, las redes de libre escala pueden ser clasificadas por dos tipos; cuando la red tiene enlaces de nodos con grado similar, la relación es positiva (*asortativa*) y cuando en la red los nodos de alto grado se conectan con nodos de bajo grado, la relación es negativa (*disortativa*). Cuando no es posible identificar una tendencia, la relación es nula y la red se considera neutral [17].

2.3. Problema de Transporte

El transporte modelado a través de una red, es un sistema formado por un conjunto de nodos (estaciones, paradas, terminales, etc.) y arcos (rutas, avenidas, calles, etc.). El problema consiste en enviar una cantidad de objetos desde un nodo llamado origen (fuente), a otro nodo llamado destino (sumidero). Este problema puede ser planteado como un problema de *flujo máximo*, a fin de encontrar las rutas de origen a destino por las cuales sea posible trasladar la mayor cantidad de flujo, siempre y cuando se satisfagan las restricciones de capacidad en los arcos y las restricciones de conservación de flujo para nodos intermedios [21, 22].

El flujo $f(i, j)$, es una función no negativa definida en los arcos. La capacidad de arco $c(i, j)$, es un valor no negativo asociado al arco (i, j) . El valor de la capacidad de un arco se define en un intervalo cerrado $c(i, j) = [c^-(i, j), c^+(i, j)]$, donde $c^-(i, j)$ es la capacidad mínima y $c^+(i, j)$ es la capacidad máxima, es decir, el flujo a través de las trayectorias debe ser factible respecto a las capacidades, así $f(i, j) \in c(i, j)$, para todo arco (i, j) en la red.

Para determinar el flujo máximo en general, debe considerarse:

- Una red dirigida. Cuando la red es no dirigida, entonces el arco (i, j) se duplica formando los arcos (i, j) y (j, i) , y en ambos arcos la capacidad es la misma al

arco original.

- No debe existir ninguna ruta del origen al destino con capacidad infinita. Si existe una ruta con capacidad infinita, el problema no tiene solución.
- La red es simple. Esto es, no existen arcos rizos ni arcos paralelos entre dos nodos.
- La red no debe tener múltiples nodos fuente ni múltiples nodos sumidero. Cuando en la red existen varios nodos fuente y varios nodos sumidero, se sugiere añadir dos nuevos nodos ficticios llamados súper-fuente y súper-sumidero. Estos nodos se unen a la red mediante arcos extras con capacidades infinitas. Los nodos sumidero se unen al nodo súper-sumidero y los nodos fuente se unen al nodo súper-fuente.

Entonces, si se tiene una red $R = (N, A)$, donde N es el conjunto de nodos y A es el conjunto de arcos, con capacidades $c(i, j)$ no negativas asociadas a cada arco $(i, j) \in A$. También, sea s el nodo fuente y t el nodo sumidero de R . Se desea encontrar el flujo máximo desde el nodo fuente al nodo sumidero, que satisfaga las capacidades de los arcos y las restricciones de conservación de flujo en todos los nodos intermedios de la red. Esto es,

$$\text{Maximizar } z = \sum_j f(r, j) \quad (2.4)$$

$$\sum_i f(i, j) - \sum_k f(j, k) = \begin{cases} -z, & \text{si } j = r \\ 0, & \text{si } j \neq r, t \\ z, & \text{si } j = t \end{cases} \quad (2.5)$$

$$0 \leq f(i, j) \leq c(i, j), \forall i, j. \quad (2.6)$$

Las restricciones (2.5) se refieren a la conservación de flujo, esto es, el flujo que entra a un nodo intermedio debe de salir, excepto para los nodos origen y destino. Las

restricciones (2.6), se refieren a la capacidad, éstas indican que no se puede exceder la capacidad de los nodos.

Además, el transporte de flujo está relacionado de manera directa con la topología de la red, ya que el transporte es mejor en redes de libre escala en comparación con redes aleatorias, debido a la presencia de hubs [23].

Para la modelación de la red de transporte del servicio Ordinario de Sistema M1, se considera lo siguiente, además de tomar en cuenta la naturaleza de la ubicación geográfica de las rutas y la organización existente del organismo Sistema M1:

- Primero, se considera que la Gerencia de Tesorería y Recaudación, cuenta con siete módulos, en los cuales se lleva a cabo la recaudación de dinero obtenido de las alcancías de los autobuses del servicio Ordinario, esto es, que todos los autobuses que circulan en un subconjunto de rutas, depositan la recaudación en el módulo correspondiente al subconjunto de rutas. Cabe mencionar que las rutas pertenecen al módulo más cercano a ellas, ver figura 2.6.
- Segundo, sólo se consideran las estaciones (terminales y paradas en la intersección de dos o más rutas) de las rutas existentes, a continuación se explica por qué en la red propuesta no se consideran todas las paradas: la forma de recolección de dinero en los módulos se lleva a cabo después de haberse cumplido uno o más recorridos desde la estación origen a la estación destino, por lo que es difícil saber el flujo de personas en los nodos intermedios de la ruta, lo único que se puede conocer es el flujo total de personas de origen a destino en la ruta, por esto, sólo se consideran las estaciones que son terminales o que están en la intersección de dos o más rutas.

En la figura 2.4 se muestra el mapa de la red de metro de Bruselas (sólo líneas de metro), actualizado en 2008 por la Sociedad Bruselense de Transporte Intercomunales (STIB) y en la figura 2.5 se muestra la representación de la red de metro de Bruselas, considerando únicamente las estaciones terminales y las estaciones que se encuentran en la intersección de dos o más rutas [24].

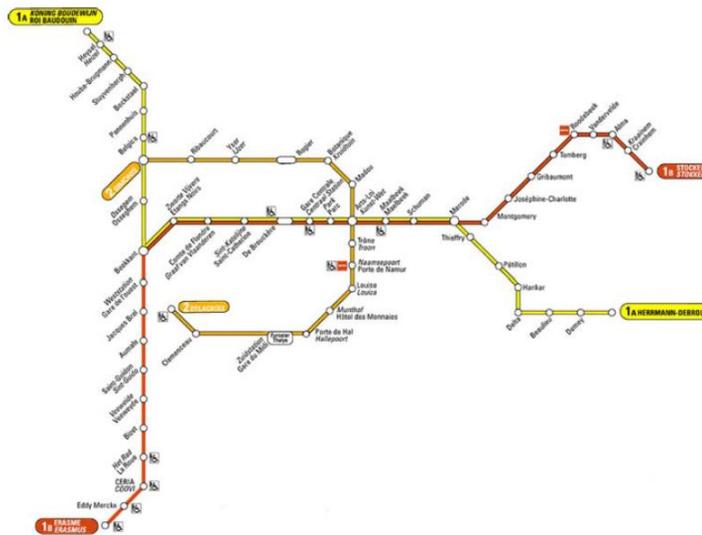


Figura 2.4: Mapa de la red de metro de Bruselas [24]

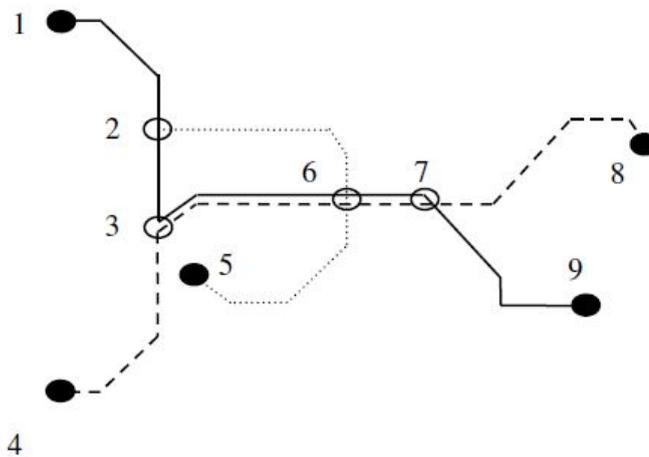


Figura 2.5: Representación de la red de metro de Bruselas [24]

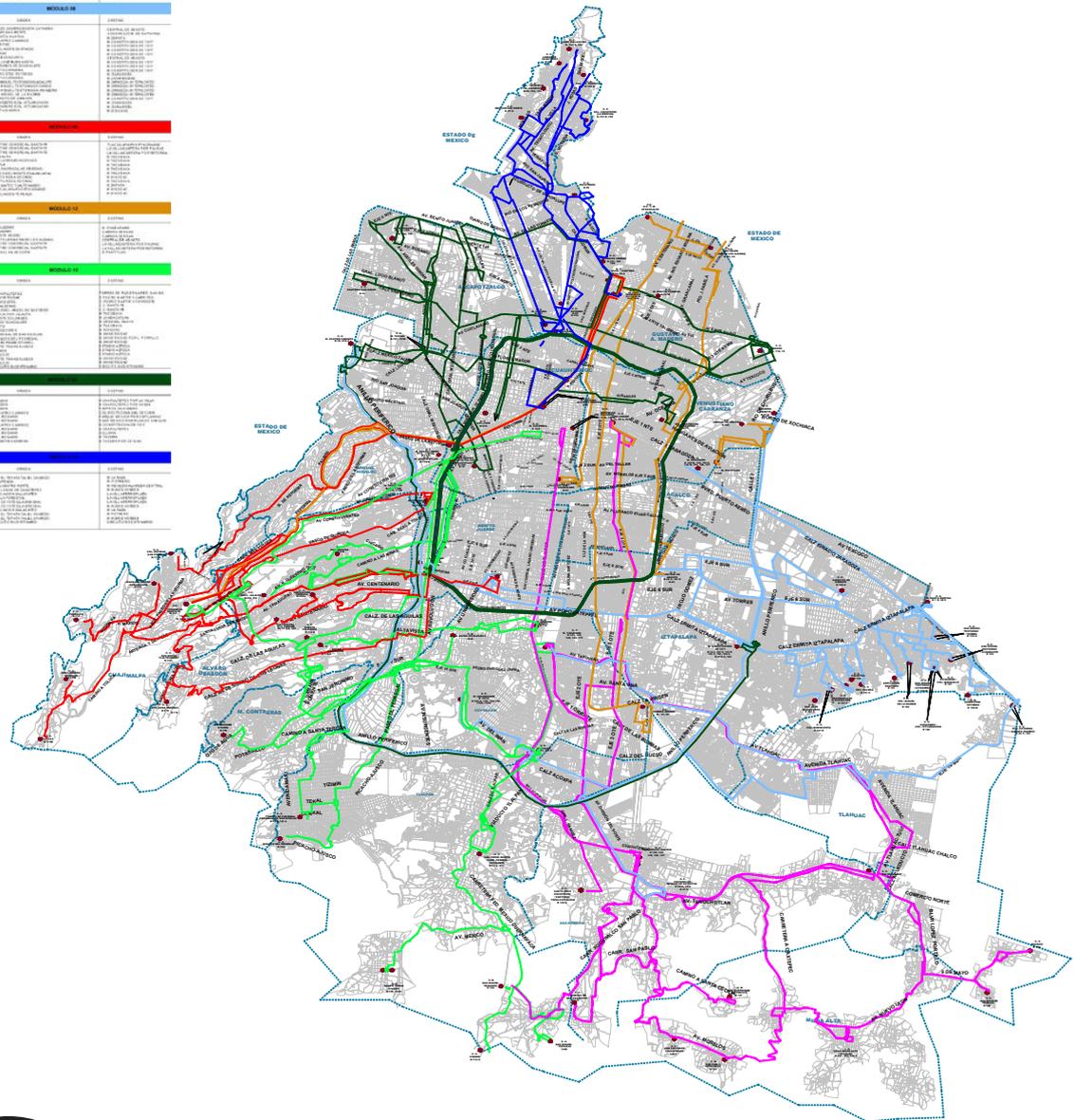
Con base en la modelación de la red propuesta en las figuras 2.4 y 2.5, se hará en esta tesis la modelación de la red de transporte del servicio Ordinario del Sistema M1 de la CDMX, ver figuras 2.6 y 2.7.

SISTEMA DE MOVILIDAD 1



CDMX
CIUDAD DE MÉXICO

LINEA	COLORES	ESTACIONES	TIPO DE SERVICIO
MOBILIDAD 01	MARROQUÍN	15	ORDINARIO
MOBILIDAD 02	ROJO	15	ORDINARIO
MOBILIDAD 03	VERDE	15	ORDINARIO
MOBILIDAD 04	AMARILLO	15	ORDINARIO
MOBILIDAD 05	ROSA	15	ORDINARIO
MOBILIDAD 06	AZUL	15	ORDINARIO
MOBILIDAD 07	NARANJA	15	ORDINARIO
MOBILIDAD 08	GRAN VERDE	15	ORDINARIO
MOBILIDAD 09	GRAN AZUL	15	ORDINARIO
MOBILIDAD 10	GRAN ROJO	15	ORDINARIO
MOBILIDAD 11	GRAN AMARILLO	15	ORDINARIO
MOBILIDAD 12	GRAN ROSA	15	ORDINARIO
MOBILIDAD 13	GRAN AZUL	15	ORDINARIO
MOBILIDAD 14	GRAN ROJO	15	ORDINARIO
MOBILIDAD 15	GRAN AMARILLO	15	ORDINARIO
MOBILIDAD 16	GRAN ROSA	15	ORDINARIO
MOBILIDAD 17	GRAN AZUL	15	ORDINARIO
MOBILIDAD 18	GRAN ROJO	15	ORDINARIO
MOBILIDAD 19	GRAN AMARILLO	15	ORDINARIO
MOBILIDAD 20	GRAN ROSA	15	ORDINARIO



SISTEMA DE MOVILIDAD 1
CDMX

Figura 2.6: Red de transporte del servicio Ordinario
Fuente: Sistema M1 (2018)

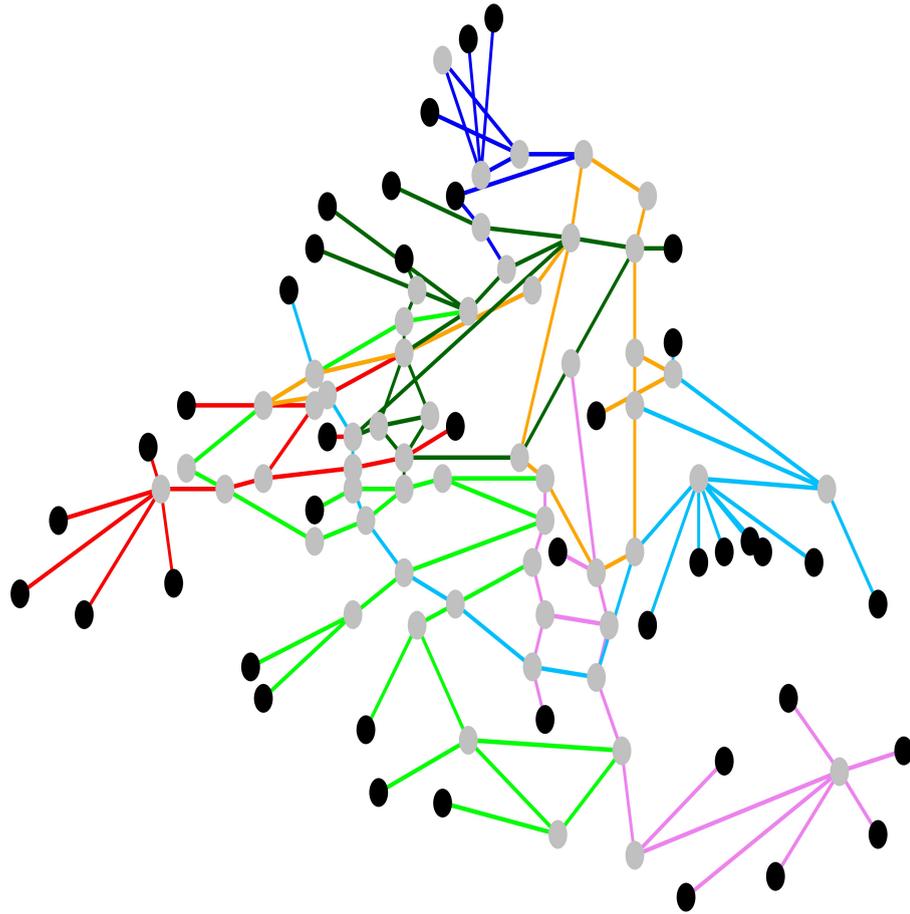


Figura 2.7: Representación de la red de transporte del servicio Ordinario

La figura 2.7 se construye como el ejemplo de la figura 2.5, para el caso de la red de transporte del servicio Ordinario, ver figura 2.6. Los nodos de color negro representan las estaciones terminales y los nodos de color gris representan las estaciones donde se intersectan dos o más rutas.

CAPÍTULO 3

PROGRAMACIÓN LINEAL Y PROGRAMACIÓN MULTI OBJETIVO

En este capítulo se introduce el uso de la *Programación Lineal (PL)* y de la *Programación Multiobjetivo (PM)*. Se define el problema de transporte visto desde la PL y se introducen las definiciones del modelo de decisión correspondiente al problema de transporte del servicio Ordinario del Sistema M1.

3.1. Antecedentes

Apartir de la aparición de la *Investigación de Operaciones (IO)*, hacia 1950, la IO pretende determinar la mejor opción (óptimo) de problemas de decisión con la restricción de recursos limitados, a través de técnicas matemáticas [9].

George B. Dantzig publicó el *método simplex* para resolver problemas de programación lineal. El método simplex en PL se le otorga mucha aceptación debido a:

- La habilidad que posee para modelar problemas importantes de decisión en áreas administrativas.
- La capacidad que posee para producir soluciones con costos y tiempos razonables computacionales.

Cabe destacar que, en los problemas de decisión suelen incluir factores que no pueden ser representados en el modelo matemático. El principal de estos factores es la presencia del ente humano, ya que se han visto casos donde la conducta humana dicta la decisión, por ejemplo, el problema del elevador, el problema empezó porque los inquilinos de un edificio se quejaban por los tiempos de espera para usar el elevador, al estudiar el problema a fondo se encontró que las quejas eran injustas, ya que el tiempo de espera era reducido. Se propuso como solución instalar espejos a lo largo de la entrada del elevador y las quejas desaparecieron porque los usuarios se mantenían ocupados viéndose o viendo a otras personas a través del espejo, mientras esperaban el servicio del elevador [9].

Este ejemplo muestra la importancia de ciertos factores que no se pueden representar en su totalidad en un modelo matemático, por lo que la IO debe visualizarse como una ciencia y como un arte. Una ciencia ya que debe proporcionar herramientas matemáticas como algoritmos y técnicas; y un arte, ya que depende de la habilidad del analista al momento de cómo llevar a cabo la decisión tomada [9].

3.2. Programación Lineal

La PL es una herramienta de la IO que nos permite llegar a la toma de decisiones después de haber obtenido la solución al problema en cuestión, a través de la aplicación de algoritmos específicos. La PL es una herramienta determinística, es decir, se conocen todos los parámetros del modelo con certeza.

Por otro lado, el modelo de decisión, es un medio para representar de manera resumida un problema, a través del lenguaje matemático, en el cual se permite identificar tres componentes principales: criterio objetivo, opciones de decisión y restricciones de recursos limitados [9].

Un problema en PL es un problema que consiste en minimizar o maximizar una función lineal bajo restricciones lineales del tipo de desigualdad, igualdad o ambas. Para poder representar un problema de optimización como un problema de PL, se requieren cuatro supuestos que están implícitos en la formulación de la PL [9] y son:

- *Proporcionalidad.* Dada una variable X_i , su contribución al costo total es $c_i X_i$ y su contribución a la i -ésima restricción es $a_{ij} X_i$.
- *Aditividad.* El costo total es la suma de los costos individuales y la contribución total de la i -ésima restricción es la suma de las contribuciones individuales.
- *Divisibilidad.* Las variables de decisión se pueden dividir en nivel fraccional.
- *Determinístico.* Supone la certeza de los valores de las variables constantes.

Para fines de esta tesis, se especifica la construcción de algunos parámetros y de algunas variables necesarias para el desarrollo del modelo de decisión:

- Longitud de ruta de la estación i a la estación j (DR_{ij}): Sea (D_{ij}) el valor que mide la distancia de la ruta existente del nodo i al nodo j , entonces, para calcular la distancia de origen a destino de una ruta, se suman todas las distancias entre todo par de nodos (estaciones) sucesivos que estén en la ruta, desde el nodo origen hasta el nodo destino, por lo que la distancia de la ruta se calcula como,

$$DR_{ij} = \sum_n d_{ij}, \quad \text{donde } n \text{ es el número de rutas en la red.} \quad (3.1)$$

- Capacidad del vehículo i (c_i): Es el número máximo de usuarios dentro de las unidades (autobuses), ya sean sentados o parados. Si el valor máximo es α , la unidad tendrá una capacidad en el intervalo $[0, \alpha]$, es decir que el número mínimo de personas que pueden llevar los camiones es 0 y el máximo α .
- Estimación de la cantidad de usuarios en la ruta i y el día j (P_{ij}): Es un valor que se calcula con ayuda de estadística, el cual describe la cantidad aproximada de usuarios que usan el transporte del Servicio Ordinario en la ruta i el día j , donde $j = 1, 2$.

$j= 1$. Día normal.

$j= 2$. Día de descanso o festivos.

- Salario del empleado i (S_i) : Los salarios de los empleados (operadores de autobuses) son distintos, dependiendo las características de cada uno de ellos, para solucionarlo se hace un promedio con los salarios, es decir, si s_i es el salario mensual del operador i , entonces:

$$S_i = \frac{1}{n} \sum_{i=1}^n s_i, \quad \text{donde } n \text{ es el número de operadores.} \quad (3.2)$$

- Costo del pasaje: El costo de pasaje del Servicio Ordinario es de \$2 pesos mexicanos (MXN) por persona (al día 31 de diciembre de 2018).
- Velocidad promedio del vehículo: Es un valor que se calcula con ayuda de estadística, el cual describe la velocidad promedio de las unidades.
- Datos sobre perturbaciones: Son datos anormales que nos ayudan a entender porqué los valores en ciertos días son distintos, por ejemplo:
 - 31 de diciembre (día festivo).
 - 1 de enero (día festivo).
 - 2 de octubre (día inhábil para la mitad de la planta vehicular).

Los días 31 de diciembre y 1 de enero son días que presentan poca actividad para transportar pasajeros, ya que son días festivos que la mayoría de las instituciones y empresas consideran como días de asueto. Por otro lado el día 2 de octubre, el sistema de transporte Sistema M1 sólo pone a su disposición la mitad de la planta vehicular, esto debido a que se llevan a cabo movimientos sociales que complican el tránsito vehicular.

Aunque el problema de un sistema puede implicar un gran número de restricciones y variables, generalmente un subconjunto de estas determina el comportamiento del sistema, de aquí la importancia de simplificar el problema para la construcción del modelo, el cual debe identificar las variables y restricciones dominantes.

3.2.1. Transporte

Considere m puntos de origen localizados en un mapa, donde el origen i tiene una provisión de a_i unidades de un cierto producto. Además, están localizados n puntos destino, en donde el destino j requiere de b_j unidades del producto. Suponga que $a_i, b_j > 0$. Asociado con cada arco (i, j) , del origen i al destino j , se tiene un costo unitario c_{ij} de transporte. El problema consiste en determinar la embarcación factible de los orígenes a los destinos que minimice el costo total del transporte [13].

Sea X_{ij} el número de unidades transportadas a lo largo del arco (i, j) , del origen i al destino j . Suponga, que la oferta total es igual a la demanda total, esto es,

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j.$$

Si la oferta total excede a la demanda total, entonces se puede introducir un destino ficticio con demanda $b_{n+1} = \sum_i a_i - \sum_j b_j$, y $c_{i,n+1} = 0$ para $i = 1, \dots, m$ y, de forma análoga cuando la demanda excede la oferta. Si la oferta total es igual a la demanda total, el modelo de programación lineal para el problema de transporte resulta ser el siguiente:

$$\begin{array}{ll} \text{Minimizar} & \\ c_{11}X_{11} + \dots + c_{1n}X_{1n} + c_{21}X_{21} + \dots + c_{2n}X_{2n} + \dots + c_{m1}X_{m1} + \dots + c_{mn}X_{mn} & \\ \\ \text{Sujeto a} & \\ X_{11} + \dots + X_{1n} & = a_1 \\ & X_{21} + \dots + X_{2n} & = a_2 \\ & \vdots \\ & \vdots \\ & \vdots \\ X_{11} + & & & X_{m1} + \dots + X_{mn} & = a_m \\ & X_{21} + \dots + & & X_{m1} & = b_1 \\ & \vdots \\ & \vdots \\ & \vdots \\ X_{11}, & \dots, X_{1n} + & & X_{2n} + \dots + & & X_{mn} = b_n \\ & & & , X_{2n}, & \dots & , X_{m1}, & \dots & , X_{mn} \geq 0. \end{array}$$

3.3. Programación Multiobjetivo

Una parte de la IO se ha encargado de estudiar problemas de optimización con un sólo objetivo, ofreciendo buenos resultados para determinados problemas. Este tipo de problemas tiene una estructura en común, la búsqueda de la mejor solución. Para resolverlos, se han propuesto técnicas como la programación lineal, simulación, métodos heurísticos, entre otros. Estos métodos consisten en determinar o elegir el valor máximo o mínimo de la función objetivo, eligiendo los valores de las variables en un espacio de búsqueda determinado por el problema.

Esta forma puede resultar válida en algunos problemas pero en otros no, como cuando es necesario contemplar más de un sólo objetivo. Este otro tipo de problemas tiene aplicaciones en la vida real y su aplicación por naturaleza es más compleja, ya que se requiere de poder visualizar integralmente todo el sistema. Por ejemplo, en el caso del sistema del transporte se desea cubrir todas las estaciones, que los tiempos de traslado sean cortos, las ganancias por servicio se maximicen o que los gastos por traslado sean lo más bajos posibles, esto hace que se requiera de la consideración de múltiples objetivos para llevar a cabo la toma de decisiones. Por lo tanto, resulta difícil encontrar una solución factible que simultáneamente optimice todas las funciones objetivo debido a factores como:

- Que las funciones objetivo conflictúen entre ellas, determinar cuántas y la importancia de ellas se vuelve complejo, ya que la mayoría de la veces no es posible reducirlas a una sola función objetivo [25].
- Por lo general, la optimización de una función objetivo puede afectar en los valores de las otras funciones, de tal forma que optimizar una función objetivo implica sacrificar el desempeño de otras [26].
- La PL tradicional es ineficiente al dar respuesta a este tipo de problemas, esto es, los métodos clásicos de programación lineal están enfocados a resolver

problemas con un sólo objetivo, y por tanto no son capaces de resolver este tipo de problemas [27].

- La literatura de problemas con múltiples objetivos es extensa, sin embargo, la importancia de las matemáticas no parece ser relevante para resolver problemas en la práctica [26].

Por estas y otras más razones, es que surge la necesidad de considerar más de un objetivo en los problemas de sistemas reales, ya que pocas veces es posible evaluar una solución y decidir con base a un criterio único.

3.3.1. Modelo general de programación lineal multiobjetivo

El problema de programación lineal con un sólo objetivo, se plantea mediante un modelo matemático donde la función objetivo toma valores en \mathbb{R} y las restricciones son de carácter lineal [13].

$$\begin{aligned} \text{Max (Min)} \quad & z = cx \\ \text{s.a :} \\ & Ax = b \\ & x \geq 0. \end{aligned}$$

Donde la función objetivo z se optimiza, c es un vector renglón de n componentes, A es la matriz de $m \times n$, b es un vector columna de m componentes y x es el vector renglón de las variables de decisión con n componentes.

Por otro lado, si la función objetivo toma valores en \mathbb{R}^n , y se consideran las restricciones y variables del problema de programación lineal, se obtiene un problema de programación lineal con k objetivos.

$$\text{Max (Min) } z = [f_1(x), f_2(x), \dots, f_k(x)]$$

s.a :

$$x \in X.$$

Donde z consta de k componentes, cada una de ellas a su vez, consta de una sólo función objetivo definida como:

$$f_\ell(x) = c_{\ell 1}x_{\ell 1} + c_{\ell 2}x_{\ell 2} + \dots + c_{\ell n}x_{\ell n}, \quad \ell = 1, \dots, k.$$

El conjunto de restricciones está conformado en un espacio vectorial de dimensión n .

$$\begin{array}{rcl} a_{11}X_{11} + \dots + a_{1n}X_{1n} & & = b_1 \\ a_{21}X_{21} + \dots + a_{2n}X_{2n} & & = b_2 \\ & \cdot & \\ & \cdot & \\ & \cdot & \\ & & a_{m1}X_{m1} + \dots + a_{mn}X_{mn} = b_m \\ X_i \geq 0, \quad i = 1, \dots, n. \end{array}$$

Este problema es el modelo general de un problema de *Programación Lineal Multi-objetivo (PLMO)* y se formula de la siguiente forma:

$$\text{Max (Min) } z = [f_1(x), f_2(x), \dots, f_k(x)]$$

s.a :

$$x \geq 0.$$

Donde la función objetivo z es un vector con k funciones objetivo, las restricciones se componen por la matriz A , que es una matriz de m renglones y n columnas, b es un vector de m renglones de términos independientes y las variables de decisión X_i ,

que se componen de un vector de n componentes no negativas.

3.3.2. Planteamiento de Transporte

Un modelo matemático es la generalización de un sistema que aplica conceptos y un lenguaje matemático. En la literatura, hay varios modelos matemáticos del comportamiento humano, que se basan en encuestas u observaciones empíricas, donde los modelos matemáticos sirven para determinar y analizar los efectos de agentes y factores en el sistema [28].

En términos generales, un sistema de transporte consiste en los medios y equipos necesarios para el movimiento de los pasajeros. El sistema de transporte es un servicio esencial en todas las ciudades, ya que contribuye a la calidad de vida de las personas [5].

Carlos Gershenson dice que el transporte urbano es un fenómeno complejo [29]. En términos generales, un sistema complejo no se puede separar en sus componentes, ya que las interacciones entre sus elementos lo definen y son relevantes para explicar la evolución del propio sistema [4].

Por otro lado, una red compleja puede verse como una estructura que describe una amplia variedad de sistemas de gran importancia tecnológica e intelectual. En las últimas décadas, ha crecido el interés en los sistemas de transporte, por ejemplo: Michael Wegener presenta una revisión de los desarrollos en el campo de las operaciones que integran el uso de la tierra y las políticas del transporte [28].

Attila Kovacs en el año 2015 [30], da una generalización del problema de enrutamiento de vehículos, considerando varias funciones objetivo. La mejora de la consistencia del conductor y la consistencia del tiempo de llegada, junto con la minimización del costo de enrutamiento son objetivos independientes del problema. Este modelo se llama *MOGenConVRP*. También, Xiaolei Ma en el mismo año [31], presenta un modelo basado en la aplicación de la teoría del aprendizaje en una red de transporte a gran escala. Este modelo se utilizó para modelar y predecir la congestión y la

evolución del tráfico.

Un año después, Dragan Pamučar muestra un modelo de soporte de decisión espacial de transporte, que se basa en la integración del método de criterios múltiples de la combinación lineal ponderada y el algoritmo Dijkstra modificado dentro de un sistema de información geográfica [32]. Por otra parte, Gershenson en el año 2009 [33], analizó el fenómeno de inestabilidad en un caso específico de sistemas de transporte públicos. Fernando Lámbarry en el mismo año [34], menciona que el total de viajes generados en la Ciudad de México se produce en un 20 % en automóviles privados y en un 80 % en transporte público [35].

También, Miguel López en el año 2011 [36], mencionó que el transporte de pasajeros del sistema de la CDMX está integrado por:

- Transporte público gubernamental (compuesto por el Sistema de Transporte Colectivo del Metro (STCM)).
- Servicio de Transporte Eléctrico (ETS).
- Sistema M1.
- Metrobús.
- Transporte público concesionado (microbuses, autobuses, combis y taxis).
- Transporte particular.

Por otro lado, varios estudios en la literatura señalan que el sistema de transporte urbano presentado en la CDMX es deficiente, por lo que la mayoría de los viajes en transporte público todavía se caracterizan por la baja calidad del servicio, la regulación deficiente y la alta contaminación [37].

Para la construcción de un modelo de decisión, se define una función objetivo, la cual indica el curso para determinar la solución óptima. Después se establecen las opciones de decisión, las cuales delimitan el problema a través de un conjunto de soluciones posibles y por último, se establecen las restricciones de los recursos con los que se puede resolver el problema.

A continuación se presenta el modelo para la maximización de utilidades, minimización de tiempos muertos y del tiempo de transporte para los usuarios del sistema de transporte Ordinario del Sistema M1. El modelo propuesto está descrito por día de funcionamiento del Sistema M1 (al día 31 de enero de 2018).

Descripción del modelo:

- El servicio Ordinario de Sistema M1 está integrado por 599 autobuses y 1797 operadores en el sistema, de los cuales sólo hay 1198 operadores disponibles por día.
- Los costos de administración son de \$ 585,501 MXN en promedio por día. Además, el valor por salario de los operadores es de \$ 701,404 MXN en promedio por día.
- El gobierno de la Ciudad de México otorgó para el año 2017, un subsidio anual a Sistema M1 de \$ 3,050, 101 MXN, que equivale a \$ 8,356.441 MXN por día.
- El presupuesto máximo para combustible del año 2017 fue de \$ 374,879,583 MXN, que equivale a \$ 1,027,067 MXN por día.
- Un autobús recolecta \$ 104 MXN en promedio al terminar una ruta.
- Cada autobús en el sistema gasta un promedio de \$ 160.11 MXN por día en mantenimiento.
- El costo de un litro de combustible diesel es de \$ 17.66 MXN. Además, el rendimiento del diesel por km es de 0.30 l/km, en promedio. Por esta razón, una corrida de un km implica un costo de \$ 5.29 MXN.
- La regulación de tráfico actual en el CDMX, establece que cada autobús debe conducirse a menos de 60 km/h.
- Un autobús se puede quedar esperando en una estación terminal un mínimo de 5 minutos (1/12 hr).
- Un autobús permanece en cualquier estación un promedio de 2 min (1/30 h).

Definición de variables:

- j es la j -ésima ruta, $j = 1, \dots, 91$.
- L_j es la longitud de recorrido en km. de la ruta j .
- ns_j es el número de estaciones de la ruta j .
- P_j es la prioridad que se le da a la ruta j .
- TT_j es el tiempo de recorrido de la ruta j .
- TB_j es el tiempo en la estación terminal de la ruta j .
- TA_j es el tiempo total de recorrido de la ruta j .
- x_j es el número de autobuses en la ruta j .
- C_j es el número de viajes realizados en la ruta j .
- C_T es el total de viajes realizados del servicio Ordinario.
- I son los ingresos totales del servicio Ordinario.
- ct son los costos totales del servicio Ordinario.

Modelo:

$$\text{Max } z = I - ct \quad (3.3)$$

$$\text{Min } z = \sum_{j=1}^m TB_j \quad (3.4)$$

$$\text{Min } z = 1 - \sum_{j=1}^m (P_j \cdot \frac{C_j}{C_T})^{\frac{1}{2}} \quad (3.5)$$

sujeto a :

$$TT_j = \frac{L_j + 2 ns_j}{60}, \quad \forall j = 1, \dots, m \quad (3.6)$$

$$TB_j = \text{máx}(\frac{1}{12}, \frac{TT_j}{x_j}), \quad \forall j = 1, \dots, m \quad (3.7)$$

$$x_j \geq 1, \quad \forall j = 1, \dots, m \quad (3.8)$$

$$x_j \leq [12 TT_j], \quad \forall j = 1, \dots, m \quad (3.9)$$

$$\sum_{j=1}^m x_j \geq 91 \quad (3.10)$$

$$\sum_{j=1}^m x_j \leq 599 \quad (3.11)$$

$$TA_j = TT_j + TB_j, \quad \forall j = 1, \dots, m \quad (3.12)$$

$$C_j \leq \left\lfloor \frac{16}{TA_j} \right\rfloor x_j, \quad \forall j = 1, \dots, m \quad (3.13)$$

$$C_T = \sum_{j=1}^m C_j \quad (3.14)$$

$$\sum_{j=1}^m 5.298L_jC_j \leq 1027067.351 \quad (3.15)$$

$$I = \sum_{j=1}^m 104C_j \quad (3.16)$$

$$ct = 160.11 \sum_{j=1}^m x_j + \sum_{j=1}^m 5.298L_jC_j + 1278550.227 \quad (3.17)$$

La función objetivo 3.3 representa la maximización de la utilidad, donde I es el ingreso total y ct son los costos de operación totales.

La función objetivo 3.4 representa la minimización de los tiempos de llegada de los autobuses a las estaciones terminales por ruta.

La función objetivo 3.5 representa la minimización del incumplimiento de la demanda de usuarios, donde $\frac{C_j}{C_T}$ representa el promedio de corridas por ruta, ponderado con la preferencia por ruta P_j .

La restricción 3.6 representa el tiempo promedio de recorrido de la ruta por minuto, donde L_j representa la longitud de la ruta y ns_j es el número de estaciones por ruta.

La restricción 3.7 representa el tiempo máximo que puede estar un autobús en la estación terminal por ruta en minutos.

La restricción 3.8 representa que por ruta debe ser asignado mínimo un autobús.

La restricción 3.9 representa el número máximo de autobuses asignados por ruta.

La restricción 3.10 representa el número mínimo de autobuses a asignar.

La restricción 3.11 representa el número máximo de autobuses a asignar.

La restricción 3.12 representa al tiempo el tiempo total de recorrido por ruta.

La restricción 3.13 representa el número máximo de viajes realizados por ruta, donde $\frac{16}{TA_j}$ representa el tiempo de servicio entre el tiempo de recorrido total por ruta.

La restricción 3.14 representa el número total de corridas.

La restricción 3.15 representa el presupuesto máximo de consumo de gasolina, donde L_j es la longitud por ruta y C_j el número de viajes por ruta.

La restricción 3.16 representa la utilidad total generada del servicio Ordinario.

La restricción 3.17 representa los costos totales del servicio Ordinario.

Cuadro 3.1: Características del servicio Ordinario de SM1.

j	P_j	L_j	ns_j	N	P_j	L_j	ns_j	N	P_j	L_j	ns_j
1	0.006943564	13.2	42	31	0.010722154	48.8	137	61	0.003471782	45.2	141
2	0.010980519	54.5	148	32	0.019764934	54	133	62	0.002777425	43.5	113
3	0.008836926	38	79	33	0.003423338	24.1	73	63	0.013766018	46.2	123
4	0.001453304	43	77	34	0.015259692	56.6	155	64	0.014728209	20	106
5	0.008836926	44	83	35	0.002139586	32.5	126	65	0.027354411	56	148
6	0.007364105	20.2	82	36	0.010512232	24.5	74	66	0.03092924	37.6	82
7	0.011782567	35	101	37	0.010415345	19.4	66	67	0.027612776	36	100
8	0.010309747	43	128	38	0.021395864	44.5	126	68	0.027451298	50.9	121
9	0.011782567	37	97	39	0.002567504	22.8	82	69	0.012659893	8	50
10	0.008836926	39	128	40	0.002567504	30.2	93	70	0.017245874	24.2	72
11	0.007364105	18	79	41	0.008836926	58.9	152	71	0.00800932	15.4	46
12	0.013255388	16	56	42	0.001062527	72	163	72	0.032360236	32	71
13	0.011782567	30	102	43	0.049920992	43	101	73	0.004418463	24.8	96
14	0.011782567	16	81	44	0.030826193	54	110	74	0.006701346	25.5	98
15	0.013255388	19.2	72	45	0.045052424	42.9	99	75	0.006701346	23.7	89
16	0.005891284	38.7	157	46	0.011782567	81.5	114	76	0.006701346	29.8	103
17	0.005457964	22.7	55	47	0.008836926	12.9	34	77	0.004418463	15.5	103
18	0.005457964	26.4	68	48	0.011782567	22.2	74	78	0.006467203	34.1	108
19	0.012522636	28.5	82	49	0.003730147	14.2	45	79	0.003003495	17	114
20	0.005891284	17.9	87	50	0.005893955	26.9	88	80	0.004418463	22.6	87
21	0.011263106	25.3	98	51	0.004418463	11	57	81	0	47.2	130
22	0.007339185	34.1	132	52	0.004418463	12.6	50	82	0.038544852	40.8	127
23	0.00978558	31.3	128	53	0.012030128	47	109	83	0.007750955	43.2	111
24	0.003326451	21.7	48	54	0.002083069	37	115	84	0.00978558	36.2	109
25	0.004844347	36.2	90	55	0.043792894	67	185	85	0.003003495	30.1	94
26	0.002494839	50.6	79	56	0.008041615	28.3	119	86	0.003003495	32.3	97
27	0.002494839	40.7	79	57	0.018093635	29.8	83	87	0.004418463	81.5	114
28	0.006459129	46.8	103	58	0.012062423	44.8	83	88	0.017148987	29.5	70
29	0.007339185	50.5	129	59	0.002777425	37	97	89	0.004505242	32.4	94
30	0.012062423	48.8	137	60	0.010309747	20	61	90	0.004505242	18.9	67
								91	0.004418463	42.2	122

En el cuadro 3.1 se presentan las características de la red de transporte del servicio Ordinario de SM1 por ruta, la información de la 91 rutas se presenta por renglones en donde; j es la j -ésima ruta, P_j es la prioridad de la ruta j , esto es, entre mayor demanda tenga la ruta, mayor será la prioridad de ésta. Este parámetro se obtiene a través de estandarizar en el intervalo $[0, 1]$ los valores de la matriz de demanda de transporte de pasajeros de la CDMX, que coinciden con las rutas del servicio

Ordinario de Sistema M1, L_j es la longitud en km. de recorrido del circuito de la j -ésima ruta y ns_j es el número de estaciones en la j -ésima ruta.

3.3.3. Problemas multiobjetivo y métodos Heurísticos

Los métodos heurísticos sirven para resolver problemas en donde las variables son enteras o bien cuando el problema resulta ser complejo para resolver en costos y tiempos razonables computacionalmente. Los métodos heurísticos y metaheurísticos son métodos que han surgido para resolver problemas multiobjetivo.

En los métodos heurísticos clásicos, Paolo Serafini desarrolló un algoritmo voraz basado en órdenes topológicos y con ello, muestra todas las bases eficientes de un problema con un matroide multiobjetivo [38]; Gerald Warburton dio un algoritmo de ϵ -aproximación del conjunto pareto para el problema multiobjetivo de la ruta más corta [39]; Thomas Erlebach desarrolló un esquema de una aproximación en tiempo polinomial completo para el problema de la mochila [40]; y Matthias Ehrgott se enfocó en combinar heurísticos, a partir de cotas inferiores y hacer la extensión a cotas superiores sobre valores óptimos al caso multiobjetivo [41].

Una matroide M es un par (S, I) [42], tal que:

- S es un conjunto finito.
- $I \subseteq \mathbb{P}(S)$ es una familia de subconjuntos independientes de S , tales que si $B \in I$ y $A \subseteq B$, entonces $A \in I$ (propiedad hereditaria).
- Si $A \in I$, $B \in I$ y $|A| < |B|$, entonces existe algún elemento $x \in B \setminus A$ tal que $A \cup x \in I$ (propiedad de intercambio).

En teoría de redes, la matroide (M) de una red (R) , cumple que:

- S representa el conjunto de arcos de R .
- I representa el conjunto de arcos acíclicos de R .

Con base en lo mencionado al inicio de este capítulo, se propone para este trabajo la implementación de técnicas heurísticas conocidas, que han sido muy estudiadas en la literatura y que se explicarán en el siguiente capítulo. Se proponen estas técnicas ya que el problema tiene naturaleza de ser un problema multiobjetivo.

TÉCNICAS HEURÍSTICAS

En este capítulo se introduce la justificación del uso de métodos heurísticos para la elaboración de este trabajo, se menciona exactamente qué heurísticas fueron utilizadas, junto con una descripción de la heurística y un pseudocódigo que explica el método computacional.

4.1. Definiciones básicas

Existen métodos exactos para resolver problemas de programación lineal multiobjetivo, los cuales a veces no pueden garantizar que la solución obtenida sea óptima, ya que estos son poco eficientes y hasta inoperables debido a los altos costos y tiempos computacionales. Para este tipo de problemas más complejos de optimización, es cuando se justifica el uso de los métodos heurísticos.

La palabra heurística 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. De manera más precisa, estos problemas se pueden expresar como, encontrar el valor de las variables de decisión para los que una determinada función objetivo alcanza su valor máximo o mínimo [43].

Algunas clases de problemas de optimización son relativamente fáciles de resolver, los cuales pueden ser resueltos con el método *simplex*, sin embargo, muchos otros tipos

de problemas de optimización son difíciles de resolver. En general, la mayor parte de problemas en la práctica son difíciles. La idea de problema "difícil de resolver" queda reflejada en el término científico *NP-hard*, utilizado en el contexto de la complejidad algorítmica, es decir, un problema de optimización difícil es aquel para el que no podemos garantizar encontrar la mejor solución posible en un tiempo razonable computacionalmente [43].

Sin embargo, en contraposición a los métodos exactos que proporcionan una solución óptima del problema, los métodos heurísticos se limitan a proporcionar una buena solución factible del problema, que no necesariamente es la óptima. La justificación a esto es que, el tiempo invertido por un método exacto para encontrar la solución óptima de un problema difícil, si es que existe, es de un orden de magnitud muy superior al del heurístico (llegando a ser tan grande en algunos casos que incluso puede llegar a ser inaplicable) [43]. Para este trabajo se hace uso de tres heurísticas conocidas, *Recocido Simulado*, *Reencadenamiento de Trayectorias* y *Búsqueda Armónica*.

4.2. Recocido Simulado

Recocido Simulado (RS) es un método de optimización inspirado en el proceso de templado de metales. El proceso de templado de metales consiste en tres fases; la primera fase es de calentamiento, en la segunda fase mantiene una temperatura alta, lo cual permite a las moléculas acomodarse en estados de mínima energía y la tercera fase es de enfriamiento controlado para aumentar el tamaño de sus cristales y reducir los defectos.

El algoritmo de Nicholas Metropolis [44] propuesto en 1953, es el pilar de los métodos de recocido simulado, sin embargo, serían Scott Kirkpatrick y Daniel Gelatt los primeros en aplicarlo a problemas de optimización para encontrar soluciones al problema del *agente viajero* con un número relativamente grande de ciudades [45].

El algoritmo de RS es un método iterativo que inicia con un cierto estado s_0 . Mediante un proceso en particular, genera un estado vecino s_1 del estado actual. Si la

evaluación del estado s_1 es mejor que la del estado s_0 , se cambia el estado s_0 por el estado s_1 . Si la evaluación de s_1 es peor a la de s_0 se elige el estado s_0 por el estado s_1 con una cierta probabilidad, que depende de las diferencias de las evaluaciones $\Delta f = f(s_0) - f(s_1)$ y de la temperatura actual del sistema T [43].

La posibilidad de elegir un estado peor al actual es lo que le permite al algoritmo RS salir de óptimos locales para poder llegar a óptimos globales. La posibilidad de aceptar elegir un peor estado, se calcula con la fórmula,

$$P(\Delta f, T) = e^{\Delta f/T},$$

se puede ver que una cualidad del algoritmo RS es que la temperatura va disminuyendo gradualmente conforme avanza el método.

A continuación, se muestra un pseudocódigo que describe el método de RS que se adaptó para el problema de interés de este trabajo.

Algoritmo 1: Pseudocódigo del algoritmo de Recocido Simulado

Entrada: parametros

1 recocido(parametros) = función(soluciones, no_dominado, contador_evaluaciones)

Salida: resultado_recocido

2 contador = data(todos, 1) + 1

3 pesos = $\frac{i=0,1}{parametros}$

4 contador_evaluaciones = 0

5 **mientras** contador_evaluaciones < maximo_evaluaciones **hacer**

6 temperatura_minima = temperatura_minima · λ

7 maximo = max(soluciones)

8 minimo = min(soluciones)

9 **mientras** temperatura > temperatura_minima **hacer**

10 j = 0

11 **mientras** j < repeticiones **hacer**

12 auxiliar = pesos(1, i)

13 solucion_actual = soluciones(i, 1:91)

14 solucion_actual = vecino(solucion_actual, parametros_vecino, maximo, minimo)

15 solucion_1 = evaluar(solucion_actual)

16 **para** m = 1 **hasta** parametros_vecinos **hacer**

17 solucion_vecino = vecino(solucion_actual, parametros_vecino)

18 solucion_2 = evaluar(solucion_vecino)

19 contador_evaluaciones = contador_evaluaciones + parametros_vecinos

20 mejor_solucion = encontrar_solucion(solucion_1, solucion_2, auxiliar)

21 auxiliar_2 = soluciones(i, todos)

22 **si** indice = 2 **entonces**

23 soluciones = solucion_actual

24 sin_movimiento = 0

25 **en otro caso**

26 e = $\exp\left(\frac{\delta}{temperatura}\right)$

27 **si** random < e **entonces**

28 soluciones = solucion_1

29 sin_movimiento = 0

30 **en otro caso**

31 soluciones = soluciones

32 sin_movimiento = sin_movimiento + 1

33 **si** sin_movimiento ≤ maximo_movimientos **entonces**

34 j = repeticiones

35 **en otro caso**

36 j = j + 1

37 solucion_inicial = encontrar(solucion_inicial)

38 temperatura = temperatura · λ

39 resultado = (soluciones, solucion_inicial)

40 **para** i = 1 **hasta** parametros **hacer**

41 **si** sin_movimiento = movimientos **entonces**

42 soluciones = encontrar_solucion(resultado, soluciones)

43 sin_movimiento = 0

44 resultado_recocido = (soluciones, solucion_inicial)

45 **devolver** evaluar(resultado_recocido) = (soluciones, no_dominado)

4.3. Reencadenamiento de Trayectorias

El método de Reencadenamiento de Trayectorias (RT), fue originalmente propuesto como una estrategia que posee procesos de intensificación y diversificación. Fred Glover estableció que el principio de operación de RT dice que el algoritmo genera nuevas soluciones mediante la exploración de trayectorias en el espacio de soluciones. Para ello, necesita dos soluciones de alta calidad, una funciona como *origen* y la otra como *destino*, se genera un camino en el espacio de soluciones entre origen y destino [46].

El método de combinación en RT se basa en la generación de trayectorias entre soluciones en el espacio de búsqueda, en lugar de llevar a cabo combinaciones lineales entre ellas, esto es, dadas dos buenas soluciones x_1 y x_2 , RT construye un camino entre ambas, comenzando por la solución x_1 , denominada solución origen y llevando a cabo una serie de movimientos, se intenta llegar a la solución x_2 , denominada solución destino [46].

RT construye la trayectoria de x_1 a x_2 , eliminando paulatinamente los atributos de la solución origen para introducir atributos que pertenecen a la solución destino. El objetivo es encontrar en estos caminos soluciones que mejoren aquellas que originaron la trayectoria.

RT se basa en cinco procedimientos que operan sobre un conjunto de soluciones de referencia que son:

- **Método de generación de soluciones diversas.** Dada una solución origen se le hacen movimientos a la trayectoria para generar una trayectoria vecina a ésta.
- **Método de mejora.** Transforma las soluciones de prueba en una o más soluciones de prueba mejoradas.
- **Método de actualización del conjunto de referencia.** Construye y mantiene un conjunto de referencia, compuesto por las n mejores soluciones encontradas hasta el momento. A partir del conjunto de soluciones iniciales se extrae

un subconjunto de soluciones siguiendo un criterio de calidad y diversidad.

- **Método de generación de subconjuntos.** Se basa en la combinación exhaustiva de todas las soluciones del subconjunto del método anterior. Este método especifica la forma en que se construyen los subconjuntos a los que posteriormente se les aplica el método de combinación.
- **Método de combinación de soluciones.** Combina las soluciones de los subconjuntos generados en una o más soluciones.

A continuación, se muestra un pseudocódigo que describe el método de RT que se adaptó para el problema de interés de este trabajo.

Algoritmo 2: Pseudocódigo del algoritmo de Reencadenamiento de Trayectorias

Entrada: parametros

1 reencadenamiento(parametros) = función(soluciones, no_dominado, contador_evaluaciones)

Salida: resultado_reencadenamiento

2 contador = data(todos, 1) + 1

3 pesos = $\frac{i=0,1}{parametros}$

4 contador_evaluaciones = 0

5 **mientras** contador_parametros < maximo_evaluaciones **hacer**

6 maximo = max(sin_movimiento)

7 **si** maximo = 0 **entonces**

8 maximo = 1

9 auxiliar = $\left\lceil \frac{sin_movimiento}{maximo} \right\rceil$

10 **si** suma(auxiliar) < sin_movimiento **entonces**

11 **para** i = 1 **hasta** parametros **hacer**

12 solucion_nueva (k, todos) = construccion(soluciones, parametros)

13 contador_evaluaciones = contador_evaluaciones + parametros

14 **en otro caso**

15 **para** i = 1 **hasta** parametros **hacer**

16 solucion_actual = soluciones(i, 1:91)

17 menor_solucion_actual = solucion_actual

18 **para** m = 1 **hasta** parametros_vecino **hacer**

19 aux_2 = vecino(solucion_actual, parametras_vecino)

20 solucion_vecino = evaluar(aux_2)

21 mejor_solucion(solucion_actual, solucion_vecino) =

 mejor_solucion_actual

22 solucion_nueva = mejor_solucion_actual

23 contador_evaluaciones = contador_evaluaciones + parametros_vecino

24 **para** i = 1:parametros **hacer**

25 **si** aux_2 = 0 **entonces**

26 sin_movimiento = sin_movimiento + 1

27 **en otro caso**

28 sin_movimiento = 0

29 **para** i = 1 **hasta** parametros **hacer**

30 **si** sin_movimiento(k) > sin_movimiento **entonces**

31 soluciones = encontrar_solucion(soluciones)

32 sin_movimiento = 0

33 **devolver** evaluar(resultado_reecadenamiento) = (soluciones, no_dominado)

4.4. Búsqueda Armónica

El método de Búsqueda Armónica (BA) es una estrategia basada en optimización local. Es similar a los algoritmos genéticos pero menos aleatorio. BA comienza con una solución inicial y luego la cadena empieza a mutar, si la mutación proporciona una solución con mayor aptitud que la solución anterior, se conserva la nueva solu-

ción, en caso contrario, se conserva la solución actual. Luego el algoritmo se repite hasta que no se pueda encontrar una mutación, que provoque un incremento en la aptitud de la solución actual y esta solución se devuelve como resultado [43].

El algoritmo BA se conoce como algoritmo voraz, es decir, siempre hace la mejor elección disponible en cada paso, con la esperanza de que de esta manera se puede obtener el mejor resultado global. Algunas características de BA son:

- **Documentado.** Utiliza información del estado por elegir un vecino u otro.
- **No exhaustivo.** No explora todo el espacio de estados.
- **Eficiente.** Encuentra buenas soluciones evitando explorar una parte de todas las soluciones

A continuación, se muestra un pseudocódigo que describe el método de BA que se adaptó para el problema de interés de este trabajo.

Algoritmo 3: Pseudocódigo del algoritmo Búsqueda Armónica

Entrada: *parametros*

1 ascenso(*parametros*) = **función**(*soluciones*, *no_dominado*,
 contador_evaluaciones)

Salida: *resultado_ascenso*

2 contador = data(*todos*, 1) + 1

3 pesos = $\frac{i=0,1}{parametros}$

4 contador_evaluaciones = 0

5 **mientras** *contador_evaluaciones* < *maximo_evaluaciones* **hacer**

6 **para** *i = 1 hasta parametros* **hacer**

7 *j = 0*

8 **mientras** *j* < *repeticiones* **hacer**

9 *auxiliar = pesos(1, i)*

10 *solucion_actual = soluciones(i, 1:91)*

11 *solucion_actual = vecino(solucion_actual, parametros_vecino)*

12 *solucion_1 = evaluar(solucion_actual)*

13 **para** *m = 1 hasta parametros_vecinos* **hacer**

14 *solucion_vecino = vecino(solucion_actual, parametros_vecino)*

15 *solucion_2 = evaluar(solucion_vecino)*

16 *contador_evaluaciones = contador_evaluaciones +*
 parametros_vecinos

17 *mejor_solucion = encontrar_solucion(solucion_1, solucion_2,*
 auxiliar)

18 *auxiliar_2 = soluciones(i, todos)*

19 **si** *indice = 1 entonces*

20 *auxiliar_3 = mejor_solucion*

21 *sin_movimiento = sin_movimiento + 1*

22 **en otro caso**

23 *auxiliar_3 = auxiliar_2*

24 *sin_movimiento = 0*

25 **si** *sin_movimiento* > *maximo_movimientos* **entonces**

26 *j = repeticiones*

27 **en otro caso**

28 *j = j + 1*

29 **para** *i = 1 hasta parametros* **hacer**

30 **si** *movimientos = 0 entonces*

31 *soluciones = encontrar_solucion(auxiliar_3, soluciones)*

32 *resultado = (soluciones, auxiliar_3)*

33 **devolver** *evaluar(resultado_ascenso) = (soluciones, no_dominado)*

4.5. Medidas de calidad de un algoritmo

Un buen algoritmo heurístico debe tener las siguientes propiedades [43]:

- **Eficiente.** Un esfuerzo computacional realista para obtener la solución
- **Bueno.** La solución debe de estar, en promedio, cerca del óptimo.

- **Robusto.** La probabilidad de obtener una mala solución (lejos del óptimo) debe ser baja.

Para medir la calidad de un heurístico existen diversos procedimientos, algunos son [43]:

- **Comparación con el óptimo.** Aunque normalmente se recurre a un algoritmo aproximado por no existir un método exacto para obtener el valor óptimo, o por ser computacionalmente muy costoso, en ocasiones puede disponer de un procedimiento que proporcione el óptimo para un conjunto limitado de ejemplos, regularmente de tamaño reducido. Este conjunto de ejemplos puede servir para medir la calidad del método heurístico.
- **Comparación con una cota.** En ocasiones el óptimo del problema no está disponible ni siquiera para un conjunto limitado de ejemplos. Un método alternativo de evaluación consiste en comparar el valor de la solución que proporciona el heurístico con una cota del problema, entonces la bondad de esta medida dependerá de la bondad de la cota, por lo que, de alguna manera, se tiene que tener información de lo buena que es la cota.
- **Comparación con un método exacto truncado.** Para problemas de grandes dimensiones puede establecerse un número fijo de iteraciones, en los que el método exacto es capaz de otorgar una solución al problema. De esta forma, se garantiza que el valor de la mejor solución proporcionada por el procedimiento no dista más de α del valor óptimo del problema.
- **Comparación con otros heurísticos.** Este es uno de los métodos más empleados en problemas difíciles, sobre los que se ha trabajado durante el tiempo y para los que se conocen algunos buenos heurísticos. Al igual que ocurre con la comparación de las cotas, la conclusión de dicha comparación está en función de la bondad del heurístico elegido.

CAPÍTULO 5

METODOLOGÍA Y RESULTADOS

En este capítulo se describe la secuencia de pasos que se hicieron para llevar a cabo este trabajo.

Obtención de los datos

Los datos utilizados en este trabajo fueron del año 2017 y del servicio Ordinario de Sistema M1. La obtención de los datos se llevó a cabo de dos maneras.

La primera, se realizó dentro de las instalaciones de la Gerencia de Tesorería y Recaudación, Dirección Ejecutiva de Administración y Finanzas, Sistema de Movilidad 1, Gobierno de la Ciudad de México. Se obtuvieron los siguientes datos:

- Recaudación monetaria de los autobuses en los siete módulos de Sistema M1. Los cuales se extrajeron del Sistema Web de Aplicaciones Programadas (SWAP).
- Nómina de operadores de autobuses. Los cuales fueron proporcionados por la Gerencia de Tesorería y Recaudación.

La segunda, se realizó a través de la solicitud de datos ingresada y registrada en el Sistema Electrónico de Transparencia con el número de folio 0320000013718, para Sistema de Movilidad 1. La respuesta a la solicitud fue otorgada con fundamento

en lo dispuesto en el artículo 212 de la Ley de Transparencia, Acceso a la Información Pública y Redención de Cuentas de la Ciudad de México, a través del oficio SM1/DEOM/0837/18. Se obtuvieron los siguientes datos:

- Número y nombre (cómo se identifica cada ruta) de rutas.
- Número de paradas de cada ruta.
- Longitud o distancia entre cada una de las paradas en km.
- Longitud total de cada ruta.
- Cuántos y cuáles tipos de autobuses se utilizan por ruta.
- Capacidad de pasajeros (parados y sentados) por tipo de autobús.
- Gastos de mantenimiento por autobús dependiendo el tipo.
- Velocidad promedio de autobús por ruta.
- Gasto promedio de combustible de autobús en km.
- Archivo digital del mapa de rutas de Sistema M1.

Construcción del modelo de transporte del servicio Ordinario del Sistema M1 e implementación de técnicas heurísticas

Con los datos obtenidos y revisión de la literatura mencionada en el capítulo 3, se llevó a cabo la construcción del modelo multiobjetivo de transporte del servicio Ordinario de Sistema M1 (ecuaciones 3.3-3.17). Para la construcción de dicho modelo, primero se consideraron las características de la estructura del modelo de programación lineal para el problema de transporte, después el modelo se modificó considerando las restricciones y variables de programación lineal y un modelo con tres objetivos. Luego, se cuantificó la tabla de Características del servicio Ordinario (cuadro 3.1), tomando en cuenta la matriz de priorización del transporte de la CDMX y con base en los datos obtenidos en dicha tabla, se realizaron programas a través del sistema de cómputo Matlab. Los siguientes programas fueron programados e implementados según su numeración:

1. Generador: este programa genera 20 soluciones iniciales para cada una de las tres heurísticas. A continuación, se muestra un pseudocódigo que describe dicho programa.

Algoritmo 4: Pseudocódigo Generador

Entrada: data
 1 data = leer (datos)
Salida: resultado
 2 generador(parametros) = función(soluciones, no_dominado)
 3 costo = CC · NV
 4 **para** $i = 1$ **hasta** $(2n)$ **hacer**
 5 **para** $j = 1$ **hasta** 91 **hacer**
 6 solucion_1(i, j) = redondear(0 + random · X(j, 1))
 7 auxiliar_1(1, j) = 140 · solucion_1(i, j) · NV(j, 1)
 8 auxiliar_2(1, j) = NV(j, 1) · solucion_1(i, j) · GV(j, 1)
 9 auxiliar_3(1, j) = TV(j, 1) · solucion_1(i, j)
 10 objetivo_1(i, 1) = suma(auxiliar_1) - suma(auxiliar_2) + 3,050,101
 11 objetivo_2(i, 1) = suma(auxiliar_3)
 12 z_1 = solucion_1(i, todos) · costo
 13 **si** $z_1 \leq 374,879,585$ **entonces**
 14 solucion_2(i, 1) = 0
 15 solucion_3(i, 1) = $|z_1 - 374,879,585|$
 16 **en otro caso**
 17 solucion_2(i, 1) = 1
 18 solucion_4(i, 1) = $z_1 - 374,879,585$
 19 z_2 = suma(solucion_1(i, todos))
 20 **si** $z_2 \leq 1,710$ **entonces**
 21 solucion_2(i, 2) = 0
 22 solucion_3(i, 2) = $|z_2 - 1,710|$
 23 **en otro caso**
 24 solucion_2(i, 2) = 1
 25 solucion_4(i, 2) = $z_2 - 1,710$
 26 **si** $z_2 \geq 91$ **entonces**
 27 solucion_2(i, 3) = 0
 28 solucion_3(i, 3) = $|91 - z_2|$
 29 **en otro caso**
 30 solucion_2(i, 3) = 1
 31 solucion_4(i, 3) = $91 - z_2$
 32 z_3 = solucion_1(i, todos) · NV
 33 **si** $z_3 \leq 16$ **entonces**
 34 solucion_2(i, 4) = 0
 35 solucion_3(i, 4) = $|z_3 - 16|$
 36 **en otro caso**
 37 solucion_2(i, 4) = 1
 38 solucion_4(i, 4) = $z_3 - 16$
 39 solucion_5(i, 1) = suma(solucion_2(i, todos))
 40 resultado = (solucion_1, solucion_2, solucion_4, solucion_3, solucion_5,
 objetivo_1, objetivo_2)
 41 generador(soluciones, no_dominado) = solucion_generador(n, resultado)
 42 **devolver** generador(soluciones, no_dominado)

2. RS: este programa realiza 20 iteraciones de las soluciones obtenidas por el programa Generador, haciendo los cambios en los puntos solución correspondientes al algoritmo RS (algoritmo 1).

3. RT: este programa realiza 20 iteraciones de las soluciones obtenidas por el programa Generador, haciendo los cambios en los puntos solución correspondientes al algoritmo RT (algoritmo 2).

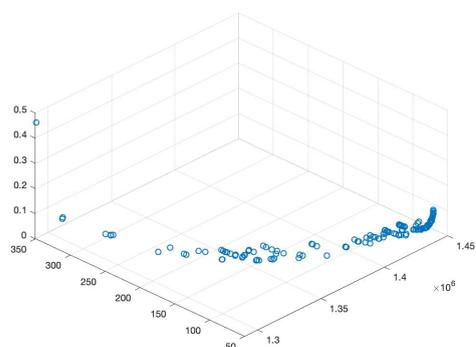
4. BA: este programa realiza 20 iteraciones de las soluciones obtenidas por el programa Generador, haciendo los cambios en los puntos solución correspondientes al algoritmo BA (algoritmo 3).

5. Maestro: este programa ejecuta de manera simultánea los programas RS, RT y BA, guardando las posibles soluciones del modelo, en soluciones dominadas y soluciones no dominadas.

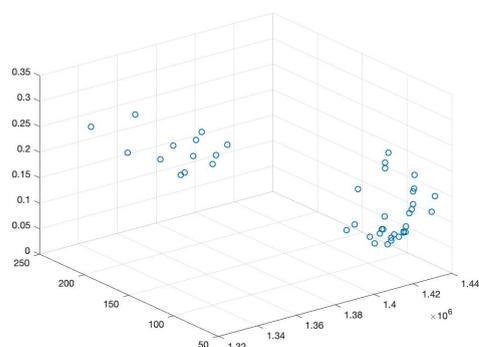
Algoritmo 5: Pseudocódigo Maestro

```

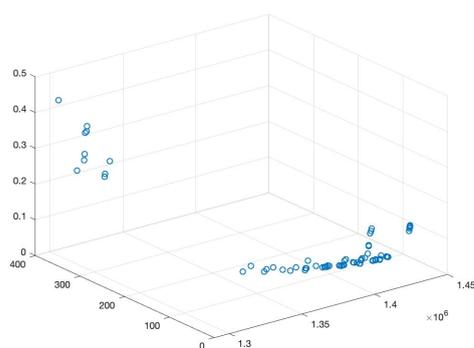
Entrada: data
1 data= leer (datos)
Salida: solucion_1
2 para heuristica = 1 hasta 3 hacer
3   parametros = parametros (heuristica)
4   solucion_1
5   para corrida = 1 hasta repeticiones hacer
6     soluciones
7     (soluciones, no_dominado) = data(parametros)
8     si heuristica = 1 entonces
9       (soluciones, no_dominado, evaluacion) = ascenso(soluciones,
10        parametros, data, heuristica, corrida)
11     si no, si heuristica = 2 entonces
12       (soluciones, no_dominado, evaluacion) =
13       recocido_simulado(soluciones, parametros, data, heuristica, corrida)
14     en otro caso
15       (soluciones, no_dominado, evaluacion) =
16       recocido_simulado_con_vecindades(soluciones, parametros, data,
17        heuristica, corrida)
18 devolver solucion_1 = (solucion_1, no_dominado)
  
```



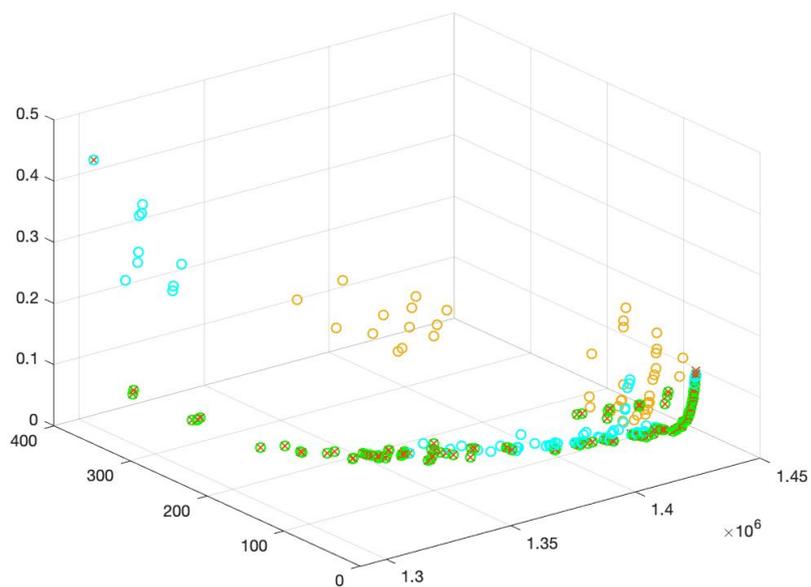
(a) RT



(b) RS



(c) BA



(d) RT contra RS contra BA

Figura 5.1: Soluciones no dominadas

En la figura 5.1 se grafican los puntos no dominados por heurística, esto es; la figura 5.1(a), muestra los puntos no dominados obtenidos en RT (algoritmo 1); la figura 5.1(b), muestra los puntos no dominados obtenidos en RS (algoritmo 2); la figura 5.1(c), muestra los puntos no dominados obtenidos en BA (algoritmo 3) y la figura 5.1(d), muestra los puntos no dominados obtenidos de las tres heurísticas, RT (color verde), RS (color amarillo), BA (color turquesa) y las equis (color rojo) representan las soluciones no dominadas del conjunto de las tres heurísticas.

6. Filtrador: este programa toma las soluciones obtenidas del programa maestro y hace un filtrado de las soluciones para cada uno de los tres algoritmos, esto es, comparar las soluciones no dominadas de cada una de las tres heurísticas y guarda un conjunto de soluciones para cada heurística que cumplan ser equidistantes entre ellas.

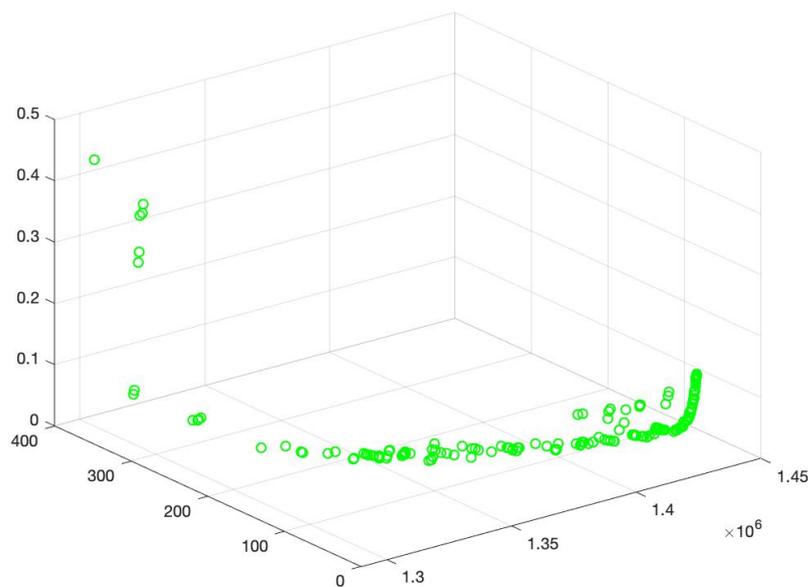


Figura 5.2: Filtrado de las soluciones no dominadas

En la figura 5.2 se grafican los puntos no dominados de las tres heurísticas y que además son equidistantes entre ellos, la distancia se considera a partir de

los valores mínimos y valores máximos obtenidos.

Algoritmo 6: Pseudocódigo Filtrador

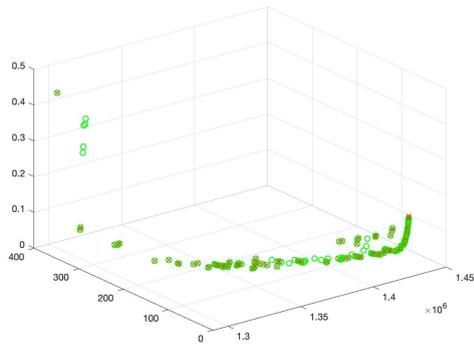
Entrada: resultados
Salida: filtrado

```

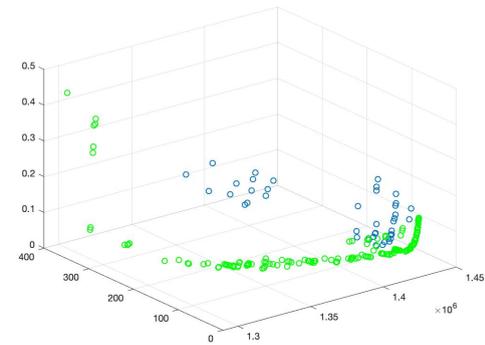
1 filtrar(n, resultado) = función(soluciones, no_dominado)
2 auxiliar_1 = solucion(objetivo_1)
3 auxiliar_2 = solucion(objetivo_2)
4 auxiliar_3 = solucion(objetivo_3)
5 mejor_1 = max(auxiliar_1)
6 peor_1 = min(auxiliar_1)
7 distancia_1 = max(auxiliar_1)-min(auxiliar_1)
8 si  $dist_1 = 0$  entonces
9   |  $dist_1 = 1$ 
10 mejor_2 = max(auxiliar_2)
11 peor_2 = min(auxiliar_2)
12 distancia_2 = max(auxiliar_2)-min(auxiliar_2)
13 si  $dist_2 = 0$  entonces
14   |  $dist_2 = 1$ 
15 mejor_3 = max(auxiliar_3)
16 peor_3 = min(auxiliar_3)
17 distancia_3 = max(auxiliar_3)-min(auxiliar_3)
18 si  $dist_3 = 0$  entonces
19   |  $dist_3 = 1$ 
20 o1=(mejor_1-auxiliar_1)/dist_1
21 o2=(mejor_2-auxiliar_2)/dist_2
22 o3=(mejor_3-auxiliar_3)/dist_3
23 q1=min(o1)
24 q2=min(o2)
25 q3=min(o3)
26 dom1=max([o1(q1,1),o1(q2,1),o1(q3,1)])
27 dom2=max([o2(q1,1),o2(q2,1),o2(q3,1)])
28 dom3=max([o3(q1,1),o3(q2,1),o3(q3,1)])
29 contador=0
30 para  $i=1$  hasta contador hacer
31   | si  $o1(i,1) \leq dom1 \cap o2(i,1) \leq dom2 \cap o3(i,1) \leq dom3$  entonces
32     | contador=contador+1
33     | sol1a(contador)=sol(i)
34     | o21(contador)=o1(i,1)
35     | o22(contador)=o2(i,1)
36     | o23(contador)=o3(i,1)
37 para  $i=1$  hasta contador hacer
38   | para  $j=1$  hasta contador hacer
39     | si  $i \simeq j$  entonces
40       | si  $o21(i,1) \leq o21(j,1) \cap o22(i,1) \leq o22(j,1) \cap o23(i,1) \leq o23(j,1)$ 
41         | entonces
42           |  $no\_dominado(i,j)=1$ 
43   | filtrado= $\sum_{\forall i} no\_dominado_i$ 
44 devolver filtrado

```

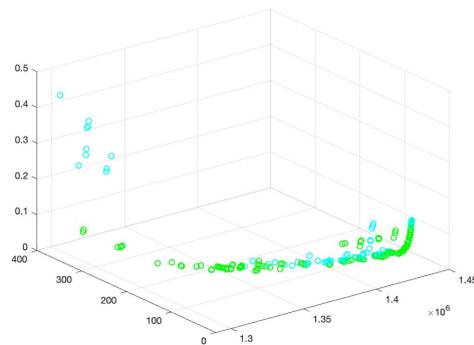
7. Evaluador: este programa toma las soluciones obtenidas del programa maestro y las compara, para así generar un nuevo conjunto de soluciones que relacionen las tres heurísticas, las tres funciones objetivo y regresa el conjunto de soluciones de la frente de Pareto del modelo.



(a) RT contra Filtrado



(b) RS contra Filtrado



(c) BA contra Filtrado

Figura 5.3: Soluciones contra heurísticas

En la figura 5.3 se grafican los puntos no dominados en el conjunto de puntos filtrados contra los puntos no dominados por heurística, esto es; la figura 5.3(a), muestra los puntos no dominados obtenidos en RS (algoritmo 1) contra las puntos obtenidos del programa 6; la figura 5.3(b), muestra los puntos no dominados obtenidos en RT (algoritmo 2) contra las puntos obtenidos del programa 6; la figura 5.3(c), muestra los puntos no dominados obtenidos en

AC (algoritmo 3) contra las puntos obtenidos del programa 6.

Algoritmo 7: Pseudocódigo Evaluador

Entrada: resultados
 1 evaluar(n, resultado) = función(soluciones, no_dominado)
Salida: evaluaciones
 2 auxiliar_1 = max(soluciones)
 3 auxiliar_2 = min(soluciones)
 4 **si** $auxiliar_2 = 0 \cap auxiliar_1 = 0$ **entonces**
 5 **para** $i = 1$ **hasta** 91 **hacer**
 6 auxiliar_3 = suma(solucion_3(i))
 7 resultado(objetivo_1)
 8 resultado(objetivo_2)
 9 **si no, si** $auxiliar_2 \simeq 0 \cap auxiliar_2 = auxiliar_1$ **entonces**
 10 **para** $i = 1$ **hasta** 91 **hacer**
 11 auxiliar_3 = suma(solucion_4(i))
 12 resultado(objetivo_1)
 13 resultado(objetivo_2)
 14 **en otro caso**
 15 **para** $i = 1$ **hasta** 91 **hacer**
 16 **si** $solucion_5 = auxiliar_2$ **entonces**
 17 contador_soluciones = contador_soluciones + 1
 18 objetivo_1(contador_soluciones)
 19 objetivo_2(contador_soluciones)
 20 resultado(contador_soluciones)
 21 contador_soluciones = auxiliar_4
 22 distancia_minima = min(auxiliar_4)
 23 **para** $i = 1$ **hasta** distancia_minima - 1 **hacer**
 24 a = resultado(i)
 25 **para** $j = i+1$ **hasta** distancia_minima **hacer**
 26 b = resultado(j)
 27 distancia = $\sqrt{\text{suma}(a - b)^2}$
 28 **si** distancia = 0 **entonces**
 29 distancia_minima(1, j) = min(distancia(distancia_minima(1, j))
 30 distancia_minima(1, i) = min(distancia(distancia_minima(1, i))
 31 mejor_objetivo_1 = max(objetivo_1)
 32 peor_objetivo_1 = min(objetivo_1)
 33 diferencia_1 = mejor_objetivo_1 - peor_objetivo_1
 34 **si** diferencia_1 = 0 **entonces**
 35 diferencia_1 = 1
 36 valor_1 = $\frac{\text{objetivo}_1 - \text{peor_objetivo}_1}{\text{diferencia}_1}$
 37 mejor_objetivo_2 = min(objetivo_2)
 38 peor_objetivo_2 = max(objetivo_2)
 39 diferencia_2 = mejor_objetivo_2 - peor_objetivo_2
 40 valor_2 = $\frac{\text{objetivo}_2 - \text{peor_objetivo}_2}{\text{diferencia}_2}$
 41 mejor_objetivo_3 = min(objetivo_3)
 42 peor_objetivo_3 = max(objetivo_3)
 43 diferencia_3 = mejor_objetivo_3 - peor_objetivo_3
 44 valor_3 = $\frac{\text{objetivo}_3 - \text{peor_objetivo}_3}{\text{diferencia}_3}$
 45 conjunto_solucion = evaluar(solucion, max(valor_1), max(valor_2),
 max(valor_3))
 46 **devolver** conjunto_solucion

Construcción de la red de transporte del servicio Ordinario del Sistema M1 y cuantificación de métricas y características

Con los datos obtenidos y revisión de la literatura mencionada en el capítulo 2, se llevó a cabo la caracterización de la red transporte del servicio Ordinario de Sistema M1 (figura 2.7), primero se consideraron todas las paradas por ruta, después la red se modificó considerando sólo las estaciones transbordo y las estaciones donde se intersectan dos o más rutas.

Luego, se realizó un programa a través del lenguaje de programación Python. Las siguientes figuras fueron obtenidas conforme el orden del programa, el cual sigue el mismo avance con base en las características mencionadas en el capítulo 2.

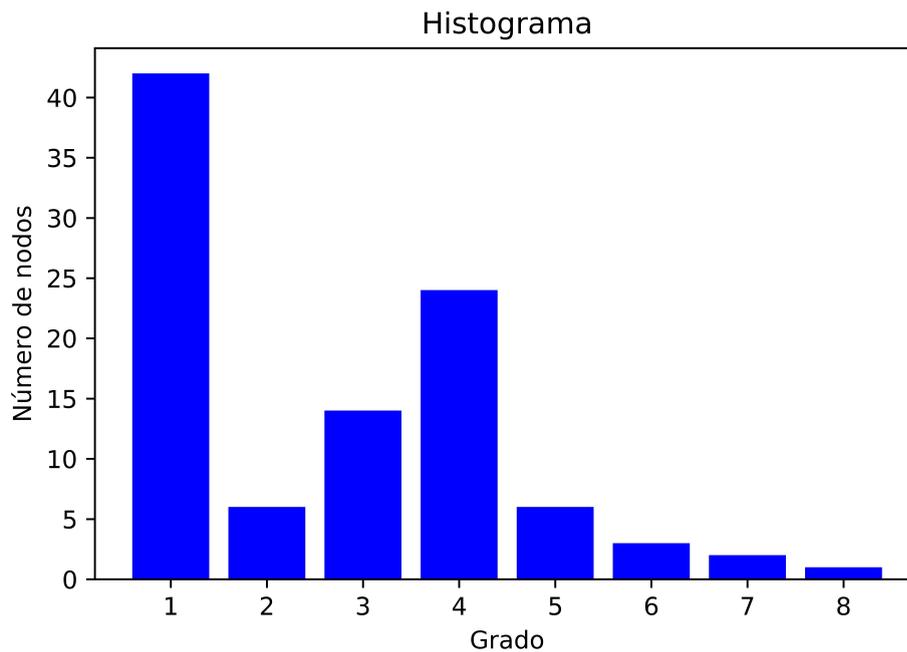


Figura 5.4: Distribución de grado.

La figura 5.4, representa el histograma de los grados de los nodos de la red, se puede apreciar que en su mayoría los nodos son de grado $k = 1$, esto se debe a que en la red hay muchos nodos terminales. Por la manera en cómo se comportan los grados de la red, la red tiene tendencia a ser de libre escala. Además, el histograma tiene sesgo a la izquierda, lo que quiere decir que hay pocos nodos con un alto grado.

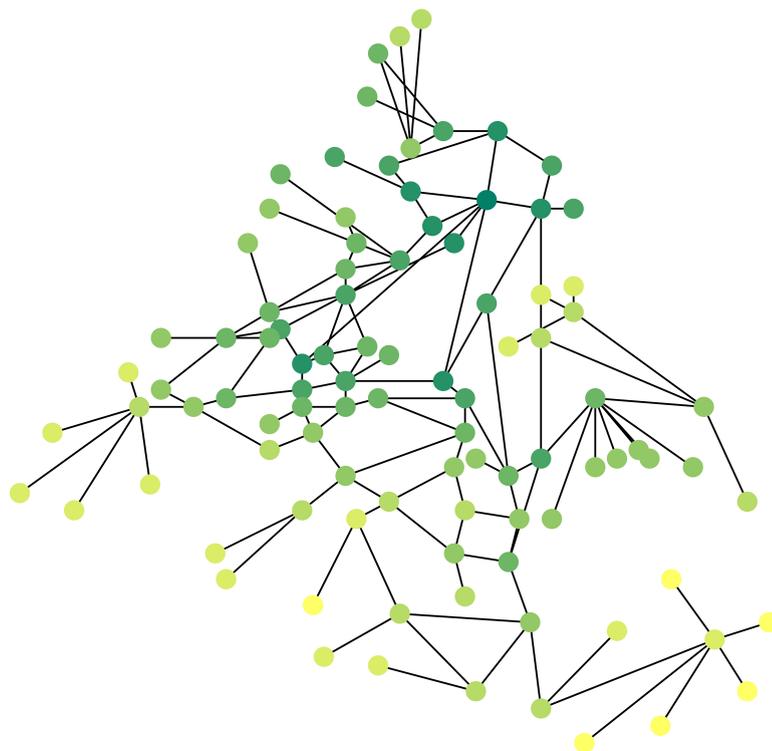


Figura 5.5: Grado de cercanía.

La figura 5.5, colorea los nodos dependiendo su valor de grado de cercanía, es decir, entre más oscuro es el color del nodo, quiere decir que tiene un grado de cercanía mayor, por lo que, por ese nodo suelen pasar más caminos cortos. En la figura podemos ver que el grado de cercanía está fuertemente relacionado con la topología de la red, ya que la mayoría de las rutas cortas pasan por el centro de la red, la cual representa la zona centro de la CDMX.

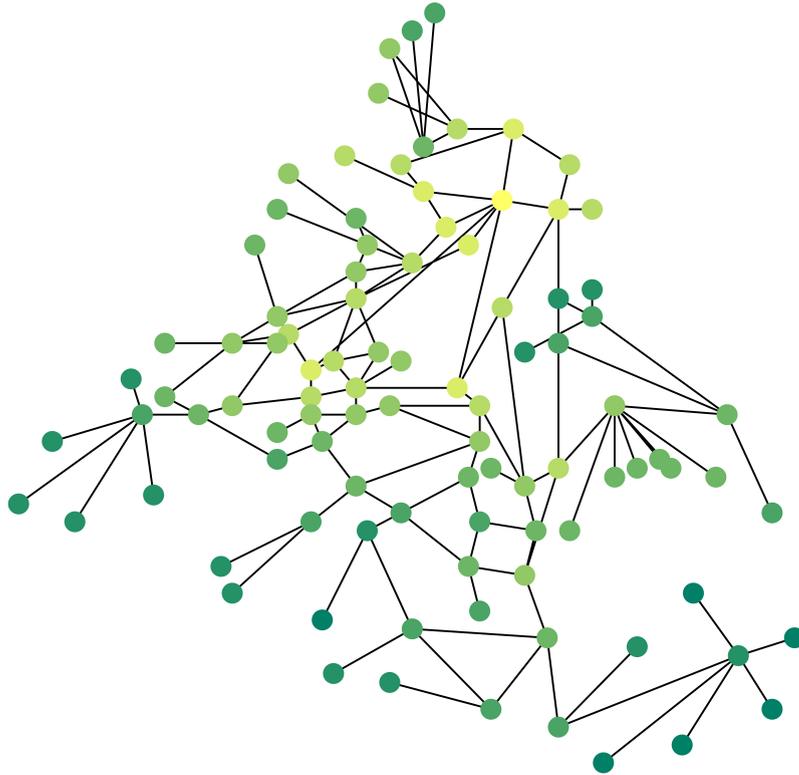


Figura 5.6: Excentricidad local.

La figura 5.6, colorea los nodos dependiendo su valor de excentricidad local, es decir, entre más obscuro es el color del nodo, quiere decir que tiene una excentricidad local mayor, por lo que por ese nodo pasan distancias máximas más grandes.

La figura 5.7, colorea los nodos dependiendo su valor de coeficiente de aglomeración, es decir, entre más obscuro es el color del nodo, quiere decir que tiene un mayor coeficiente de aglomeración local.

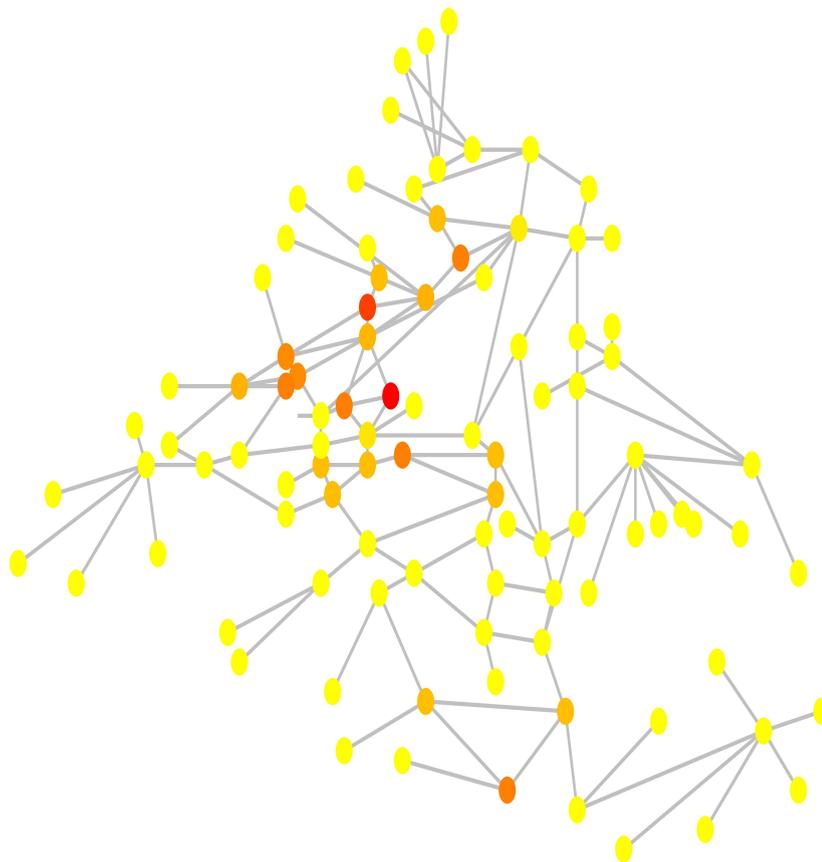


Figura 5.7: Coeficiente de aglomeración.

Las figuras anteriores presentan las características de la red de forma particular para cada nodo. En el cuadro siguiente (cuadro 5.1), se muestran las características globales de la red.

Cuadro 5.1: Medidas globales de la red.

Medida	Valor
Distancia media (l)	5.9259415106248685
Eficiencia (ε)	0.22190635883040677
Clustering global (C)	0.057531584062196314

En los anexos correspondientes a esta tesis se encuentran tablas con las soluciones encontradas por los algoritmos y tablas correspondientes a las características por nodo de la red.

DISCUSIÓN Y CONCLUSIONES

En este trabajo se utilizaron tres heurísticas para examinar el problema de transporte del servicio Ordinario de SM1 en la CDMX, con el fin de encontrar soluciones que mejoren la problemática existente de asignación de autobuses por ruta, maximizando la utilidad del servicio, minimizando los tiempos de traslado y el incumplimiento de satisfacer toda la demanda de usuarios. El problema se solucionó planteando un modelo multiobjetivo $max-min-min$, tomando en cuenta tres objetivos; maximizar la utilidad total del servicio Ordinario, minimizar los tiempos de traslado de los usuarios por ruta y minimizar el incumplimiento de satisfacer toda la demanda de usuarios del servicio Ordinario. En la cuadro 5.2 se muestran algunos de los valores obtenidos en las funciones objetivo que muestran una mejoría en los resultados si se hace la reasignación de autobuses como en las soluciones encontradas.

Cuadro 5.2: Comparación entre la asignación real y las asignaciones encontradas.

Asignación	Objetivo 1	Objetivo 2	Objetivo 3
Real	360296	70.76333796	0.880799899
1	1420000	61.4	0.082399
7	1440000	52.469	0.10784
16	1440000	53.706	0.09342
17	1440000	70.249	0.023407
20	1430000	73.896	0.020452
36	1440000	54.531	0.083519
51	1440000	62.995	0.039328
84	1410000	66.726	0.087418
85	1430000	56.582	0.091659
94	1430000	59.712	0.07667

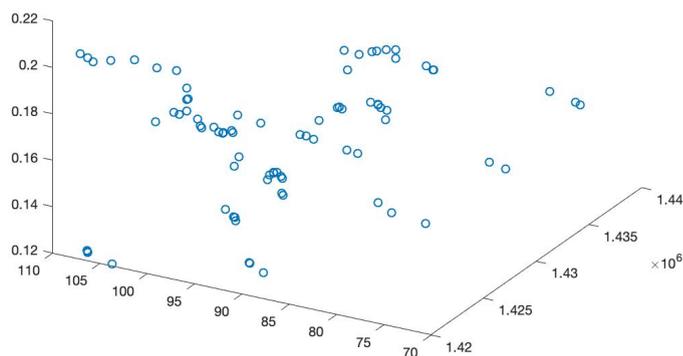
En el cuadro 5.2 se puede ver que los resultados encontrados en las soluciones pueden llegar a mejorar las tres funciones objetivo con respecto a la asignación real, ya que para el objetivo 1 el valor de las soluciones encontradas se maximiza con respecto a la asignación real y para los objetivos 2 y 3 los valores de las soluciones encontradas se minimizan con respecto a la asignación real.

Los resultados obtenidos por los algoritmos RS y BA, indican que las soluciones encontradas tienden a aproximarse a valores buenos para una función objetivo, esto es, los resultados suelen encontrarse cerca de valores buenos para un sólo objetivo, formando así nichos de soluciones, que si bien son resultados buenos para esa función objetivo, resultan ser malos para el resto de las otras funciones objetivo, esto por la naturaleza voraz de las heurísticas de ir mejorando conforme a cada iteración el valor global del modelo.

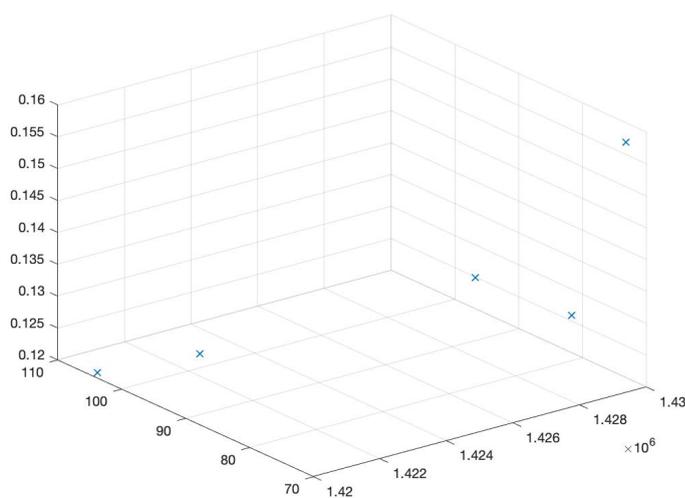
Por otro lado, se tiene que los resultados obtenidos por el algoritmo RT son buenos para representar soluciones del modelo, es decir, el algoritmo no sólo obtiene soluciones buenas, sino que además las soluciones son diversas, ya que, por la naturaleza de la heurística, ésta considera la generación de soluciones diversas empleando métodos de mejora.

Luego, se consideró las medidas de calidad de un algoritmo, y se observó que el algoritmo RT a diferencia de los otros dos, cumple ser eficiente, bueno y robusto. Así, se obtuvo que, al hacer la comparación entre los tres algoritmos, el algoritmo RT es mejor que los algoritmos de RS y BA. Lo anterior, sólo da información de qué tan bueno fue el algoritmo RT respecto a la bondad de los otros dos algoritmos utilizados, por lo que también se consideró comparar el algoritmo RT contra el método *NSGA-2* [47–49], el cual ha sido utilizado ampliamente en la práctica por obtener buenas soluciones para una gran cantidad de problemas.

Las soluciones del algoritmo *NSGA-2* y sus respectivas soluciones no dominadas se pueden observar en la figura 5.8.



(a) Soluciones del método NSGA-2



(b) Soluciones no dominadas del método NSGA-2

Figura 5.8: Soluciones encontradas por el método NSGA-2

De la figura 5.8, se puede apreciar que las soluciones encontradas son diversas, pero al obtener las soluciones no dominadas del conjunto de soluciones, éste resulta ser pequeño, lo cual le impide ser un algoritmo robusto, por lo que el algoritmo RT indica ser el mejor algoritmo para solucionar el problema.

Una vez que se analizaron las características de los algoritmos, también se consideraron métricas para observar los valores obtenidos por los cuatro métodos.

Cuadro 5.3: Hipervolumen respecto al Frente de Pareto ideal.

Corridas	RT	RS	BA	NSGA-2
1	0.0226	0.0087	0.0136	0.0016
2	0.0191	0.0109	0.017	0.0035
3	0.0245	0.0111	0.0164	0.0023
4	0.0189	0.0106	0.0076	0.0019
5	0.0202	0.013	0.0134	0.0016
6	0.0194	0.0116	0.0141	0.001
7	0.0218	0.0117	0.0168	0.0035
8	0.0213	0.0111	0.0112	0
9	0.0227	0.0113	0.012	0.0017
10	0.0191	0.0094	0.0128	0.0022
11	0.0177	0.0099	0.0188	0.0033
12	0.0249	0.0114	0.0155	0.0011
13	0.019	0.0127	0.0117	0.0037
14	0.0209	0.0124	0.0154	0.0031
15	0.0259	0.01	0.0135	0.0016
16	0.0229	0.0119	0.0159	0.0032
17	0.0213	0.0099	0.0136	0.0012
18	0.0242	0.0139	0.0086	0.0038
19	0.0186	0.0102	0.0105	0.001
20	0.0231	0.0114	0.0124	0.0023
Máximo	0.0259	0.0139	0.0188	0.0038
Mínimo	0.0177	0.0087	0.0076	0
Media	0.0213	0.0112	0.01355	0.00205
Mediana	0.021315789	0.011142105	0.0136	0.002173684
Varianza	5.69524E-06	1.62892E-06	8.08779E-06	0.000001188

En el cuadro 5.3, se presentan los resultados de las veinte corridas para cada uno de los cuatro algoritmos, se puede ver que los valores obtenidos para la métrica de hipervolumen se encuentran en el intervalo $[0.0177, 0]$, siendo el valor máximo encontrado por el algoritmo RT, lo que quiere decir que, las mejores soluciones han sido encontradas por este algoritmo.

Cuadro 5.4: Porcentaje del hipervolumen respecto al Frente de Pareto ideal.

Corridas	RT	RS	BA	NSGA-2
1	0.81294964	0.31294964	0.489208633	0.057553957
2	0.68705036	0.392086331	0.611510791	0.125899281
3	0.881294964	0.399280576	0.589928058	0.082733813
4	0.679856115	0.381294964	0.273381295	0.068345324
5	0.726618705	0.467625899	0.482014388	0.057553957
6	0.697841727	0.417266187	0.507194245	0.035971223
7	0.784172662	0.420863309	0.604316547	0.125899281
8	0.76618705	0.399280576	0.402877698	0
9	0.816546763	0.40647482	0.431654676	0.061151079
10	0.68705036	0.338129496	0.460431655	0.079136691
11	0.636690647	0.356115108	0.676258993	0.118705036
12	0.895683453	0.410071942	0.557553957	0.039568345
13	0.683453237	0.456834532	0.420863309	0.133093525
14	0.751798561	0.446043165	0.553956835	0.111510791
15	0.931654676	0.35971223	0.485611511	0.057553957
16	0.823741007	0.428057554	0.571942446	0.115107914
17	0.76618705	0.356115108	0.489208633	0.043165468
18	0.870503597	0.5	0.309352518	0.136690647
19	0.669064748	0.366906475	0.377697842	0.035971223
20	0.830935252	0.410071942	0.446043165	0.082733813
Máximo	0.931654676	0.5	0.676258993	0.136690647
Mínimo	0.636690647	0.31294964	0.273381295	0
Media	0.76618705	0.402877698	0.487410072	0.073741007
Mediana	0.766755017	0.400795153	0.489208633	0.07819008
Varianza	0.007369231	0.002107708	0.010465024	0.001537188

En el cuadro 5.4, se presentan los resultados de las veinte corridas para cada uno de los cuatro algoritmos, se puede ver que los valores obtenidos para el porcentaje del hipervolumen se encuentran en el intervalo $[0.931, 0]$, siendo el valor máximo encontrado por el algoritmo RT, lo que quiere decir que, la mayoría de las soluciones en la Frente de Pareto han sido encontradas por este algoritmo.

Cuadro 5.5: Distancia Generacional Inversa.

Corridas	RT	RS	BA	NSGA-2
1	28846.12036	-	-	0.514871141
2	45548.21948	37711.35215	4330.12836	1.292503775
3	12649.07853	-	0.022545302	-
4	34636.24958	41459.44644	-	-
5	42408.04614	-	21343.79679	1.837977047
6	30149.64492	4330.135436	-	0.762551923
7	4000.017699	-	50040.09532	1.205166902
8	54182.58381	-	16408.92133	1.279863745
9	45580.12291	51852.95823	-	0.49989997
10	9258.219751	-	7999.994529	0.388618678
11	32863.36042	-	9428.197467	0.011367061
12	30357.61568	-	14142.14234	0.777470543
13	4714.04307	-	56276.67817	4948.547418
14	9797.956236	-	11780.31631	0.794845531
15	39707.70341	-	7990.199891	0.408733024
16	39191.86189	28276.7039	41216.17338	2.766118801
17	23563.72356	-	39433.45244	-
18	43969.71128	-	4707.257992	4999.634668
19	0.348844921	4999.989107	-	1.419147317
20	40850.91799	-	4714.041547	-
Máximo	54182.58381	51852.95823	56276.67817	4999.634668
Mínimo	0.348844921	4330.135436	0.022545302	0.011367061
Media	28613.77728	28105.09754	19320.76123	622.6338263
Mediana	31610.48805	32994.02802	11780.31631	1.000006216
Varianza	268620759.2	386740791.4	333494253.1	2885448.181

En el cuadro 5.5, se presentan los resultados de las veinte corridas para cada uno de los cuatro algoritmos, se puede ver que el valor de la varianza obtenido para la distancia generacional inversa se encuentra en el intervalo [268620759, 386740791], siendo el valor mínimo encontrado por el algoritmo RT, lo que quiere decir que, las distancias entre las soluciones encontradas por este algoritmo son las menos dispersas, lo que las vuelve equidistantes y buenas soluciones para el problema.

Por otro lado, los valores obtenidos para los indicadores nos proporcionan información de la estructura de la red. El coeficiente de aglomeración local, ver cuadro 5.5, muestra que hay pocos nodos con un coeficiente de aglomeración local alto y la ma-

yoría poseen un coeficiente de aglomeración muy bajo. Esta propiedad se ve reflejada en las asignaciones encontradas, ya que los nodos con coeficiente de aglomeración alto se encuentran sobre las rutas con mayor asignación promedio de autobuses.

La distancia media es de $5.92 \simeq 6$, ver cuadro 5.1, lo que quiere decir que la distancia media es baja, entonces, si la red posee distancia media baja y pocos nodos con coeficiente de aglomeración alto, estos tendrían que estar al centro de la red, mientras que los nodos con bajo coeficiente de aglomeración a las periferias de la red. Estos dos indicadores propician a que las soluciones encontradas presenten asignaciones de camiones tales que para las rutas que no pasan por nodos con un alto grado de aglomeración, la asignación promedio es de pocos autobuses, mientras que para las rutas que pasan por el centro de la red tiene una asignación promedio de muchos autobuses.

También, el grado de los nodos, ver cuadro 5.8, muestra que hay pocos nodos con grado alto y muchos nodos con grado bajo, los nodos de bajo grado se encuentran a la periferia de la red y cuentan con un coeficiente de aglomeración local bajo, por lo que las asignaciones encontradas muestran preferencia por asignar autobuses a las rutas que se encuentran al centro de la red, en las cuales se encuentran nodos con un grado alto y un coeficiente de aglomeración alto.

En conclusión, se lograron varios objetivos en este trabajo, los cuales son:

- Primero, se logró el objetivo principal de este trabajo, esto es, se encontraron soluciones que resultan ser mejores asignaciones de autobuses por ruta para el servicio Ordinario de SM1, viéndose un aumento en la utilidad total diaria del servicio Ordinario y tiempos promedio de traslado para los usuarios más cortos que los tiempos existentes. Además, actualmente la asignación de autobuses del servicio Ordinario de SM1 es infactable, pues el problema de transporte debe asignar a lo menos un autobús por ruta, lo cual no cumple la ruta número ochenta y uno con la asignación actual, la cual corre desde la estación Aragón hasta la estación Metro Chapultepec por la Villa.
- Segundo, se lograron objetivos secundarios como; la representación de la red de transporte del servicio Ordinario de SM1 y la caracterización de la red a

través de las medidas propuestas en la metodología.

También, se puede decir que el mejor método para solucionar el problema de transporte del servicio Ordinario en la CDMX fue el algoritmo RT. Además, se puede ver que la estructura de la red influye en el funcionamiento del transporte, ya que los nodos que poseen mayor coeficiente de agrupamiento y menor grado de cercanía en la red, son nodos que representan estaciones que se encuentran en las rutas con mayor demanda en el modelo. Así, es posible concluir que los resultados obtenidos muestran que la eficiencia del transporte no sólo depende de los recursos del servicio de transporte, sino también de las características topológicas de la red.

Por otro lado, este trabajo puede ser mejorado en varias direcciones, como son; la representación de la red de transporte del servicio Ordinario de SM1 considerando todas las paradas de la red real, la eliminación de la variable en el modelo correspondiente a la asignación de autobuses en la ruta ochenta y uno, ya que actualmente esa ruta no se encuentra en operación, y la elaboración de un programa que calcule soluciones en las que se consideren la asignación de autobuses en trabajo paralelo.

ANEXOS

En este apartado se encuentran los datos relacionados a las características calculadas de la red, primero se presenta la numeración de los nodos en la red y después las tablas con los valores de las características calculadas para cada nodo. También, una tabla con la numeración de las rutas y la asignación de autobuses actual por ruta, después se encuentran tablas con las nuevas asignaciones encontradas por los algoritmos.

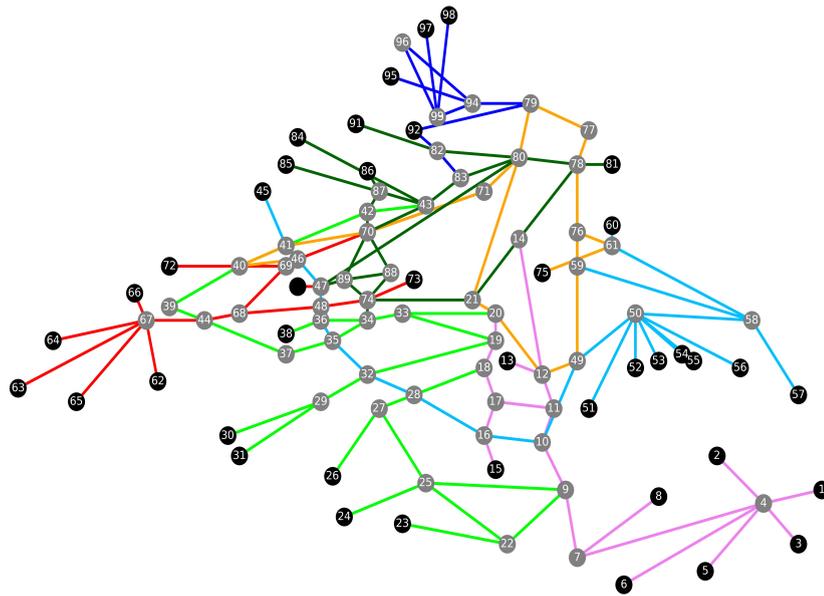


Figura 5.9: Numeración de los nodos

Cuadro 5.6: Coeficiente de aglomeración local.

Nodo	Valor	Nodo	Valor	Nodo	Valor
1	0	34	0.166666667	67	0
2	0	35	0.166666667	68	0
3	0	36	0.166666667	69	0.333333333
4	0	37	0	70	0.19047619
5	0	38	0	71	0
6	0	39	0	72	0
7	0	40	0.2	73	0
8	0	41	0.3	74	0.066666667
9	0.166666667	42	0.5	75	0
10	0	43	0.2	76	0
11	0	44	0	77	0
12	0	45	0	78	0
13	0	46	0.3	79	0
14	0	47	0	80	0.047619048
15	0	48	0	81	0
16	0	49	0	82	0.166666667
17	0	50	0	83	0.333333333
18	0	51	0	84	0
19	0.166666667	52	0	85	0
20	0.166666667	53	0	86	0
21	0	54	0	87	0.166666667
22	0.333333333	55	0	88	0.666666667
23	0	56	0	89	0.333333333
24	0	57	0	90	0
25	0.166666667	58	0	91	0
26	0	59	0	92	0
27	0	60	0	93	0
28	0	61	0	94	0
29	0	62	0	95	0
30	0	63	0	96	0
31	0	64	0	97	0
32	0	65	0	98	0
33	0.333333333	66	0	99	0

Cuadro 5.7: Grado de cercanía local.

Nodo	Valor	Nodo	Valor	Nodo	Valor
1	0.126963351	34	0.207264957	67	0.153968254
2	0.126963351	35	0.199588477	68	0.187258687
3	0.126963351	36	0.195564516	69	0.179297597
4	0.145209581	37	0.176684882	70	0.200413223
5	0.126963351	38	0.163851351	71	0.212253829
6	0.126963351	39	0.167530225	72	0.152276295
7	0.166666667	40	0.179297597	73	0.177330896
8	0.143067847	41	0.177981651	74	0.215077605
9	0.194	42	0.174774775	75	0.131258457
10	0.224537037	43	0.189823875	76	0.131258457
11	0.195171026	44	0.178308824	77	0.20464135
12	0.218961625	45	0.151326053	78	0.244949495
13	0.179962894	46	0.201663202	79	0.210869565
14	0.219457014	47	0.22985782	80	0.253926702
15	0.170175439	48	0.212719298	81	0.197154472
16	0.20464135	49	0.243107769	82	0.207708779
17	0.177330896	50	0.206823028	83	0.211328976
18	0.186180422	51	0.171681416	84	0.159802306
19	0.207708779	52	0.171681416	85	0.141605839
20	0.229314421	53	0.171681416	86	0.141605839
21	0.22985782	54	0.171681416	87	0.164685908
22	0.170175439	55	0.171681416	88	0.187984496
23	0.145645646	56	0.171681416	89	0.205073996
24	0.152037618	57	0.149460709	90	0.172291297
25	0.17896679	58	0.175406872	91	0.177981651
26	0.147865854	59	0.149460709	92	0.1515625
27	0.173214286	60	0.131258457	93	0.178308824
28	0.197154472	61	0.150855365	94	0.1515625
29	0.171378092	62	0.133608815	95	0.152996845
30	0.14652568	63	0.133608815	96	0.118004866
31	0.14652568	64	0.133608815	97	0.118004866
32	0.20464135	65	0.133608815	98	0.133608815
33	0.207264957	66	0.133608815	99	0.133608815

Cuadro 5.8: Excentricidad local.

Nodo	Valor	Nodo	Valor	Nodo	Valor
1	12	34	9	67	11
2	12	35	9	68	10
3	12	36	9	69	10
4	11	37	10	70	9
5	12	38	10	71	8
6	12	39	11	72	11
7	10	40	10	73	10
8	11	41	10	74	9
9	9	42	10	75	12
10	8	43	9	76	12
11	9	44	10	77	8
12	8	45	11	78	7
13	9	46	9	79	8
14	7	47	8	80	7
15	10	48	9	81	8
16	9	49	8	82	8
17	10	50	9	83	8
18	9	51	10	84	10
19	8	52	10	85	11
20	7	53	10	86	11
21	8	54	10	87	10
22	10	55	10	88	10
23	11	56	10	89	9
24	11	57	11	90	9
25	10	58	10	91	9
26	12	59	11	92	10
27	11	60	12	93	9
28	10	61	11	94	10
29	10	62	12	95	10
30	11	63	12	96	12
31	11	64	12	97	12
32	9	65	12	98	11
33	8	66	12	99	11

Cuadro 5.9: Grado de nodo.

Nodo	Valor	Nodo	Valor	Nodo	Valor
1	1	34	4	67	1
2	6	35	4	68	3
3	1	36	4	69	6
4	1	37	2	70	1
5	1	38	1	71	3
6	1	39	4	72	1
7	3	40	2	73	7
8	1	41	5	74	2
9	4	42	5	75	1
10	4	43	4	76	1
11	3	44	5	77	5
12	5	45	1	78	2
13	1	46	5	79	4
14	3	47	4	80	7
15	3	48	4	81	1
16	4	49	4	82	4
17	1	50	8	83	3
18	3	51	1	84	1
19	4	52	1	85	1
20	4	53	1	86	4
21	4	54	1	87	1
22	3	55	1	88	1
23	4	56	1	89	3
24	1	57	4	90	4
25	1	58	1	91	2
26	3	59	1	92	4
27	1	60	4	93	1
28	4	61	1	94	1
29	4	62	6	95	2
30	3	63	1	96	3
31	1	64	1	97	1
32	1	65	1	98	1
33	3	66	1	99	1



ORDINARIO



DISTRIBUCIÓN DE PARQUE VEHICULAR ASIGNADO POR RUTA

N°	Ruta	Origen	Destino	Ordinario
1	9-C	CENTRO COMERCIAL SANTA FE	TLACUITLAPA/PUERTA GRANDE	5
2	76*	CENTRO COMERCIAL SANTA FE (PERIFERICO)	LA VILLA/CANTERA POR PALMAS	4
3	110	CHIMALPA	METRO TACUBAYA	6
4	110-B	SAN LORENZO ACOPILCO	METRO TACUBAYA	6
5	110-C	LA PILA	METRO TACUBAYA	6
6	112	AMPLIACIÓN JALALPA	METRO TACUBAYA	5
7	113-B	COL. NAVIDAD (LAS PIEDRAS)	METRO TACUBAYA	8
8	115	JESUS DEL MONTE (CUAJIMALPA)	METRO TACUBAYA	7
9	116	SANTA ROSA XOCHIAC	METRO MIXCOAC	8
10	118	SANTA ROSA XOCHIAC	METRO TACUBAYA (CHAPULTEPEC)	6
11	119	PILOTO	METRO TACUBAYA	5
12	119-B	PRESIDENTES	METRO MIXCOAC	9
13	120	SAN MATEO TLALTENANGO	METRO ZAPATA	8
14	124	TLACUITLAPA/PUERTA GRANDE	METRO MIXCOAC	8
15	124-A	AMPLIACIÓN TEPEACA	METRO MIXCOAC	9
16	13-A	METRO CHAPULTEPEC	TORRES DE PADIERNA/PEDREGAL DE SAN NICOLAS	4
17	17-E	METRO UNIVERSIDAD	SAN PEDRO MÁRTIR POR CARRETERA FEDERAL	4
18	17-F	METRO TASQUEÑA	SAN PEDRO MÁRTIR POR FOVISSSTE	4
19	115-A	PUENTE COLORADO	METRO CHAPULTEPEC/METRO JUANACATLÁN	3
20	116-A	RÍO DE GUADALUPE	METRO GENERAL ANAYA	4
21	123-A	PEDREGAL DE SAN NICOLAS	METRO UNIVERSIDAD	3
22	125	BOSQUES DEL PEDREGAL	METRO UNIVERSIDAD POR LÓPEZ PORTILLO	3
23	128	SAN BERNABE/OYAMEL	METRO UNIVERSIDAD	4
24	131	CASETA DE COBRO	ESTADIO AZTECA	4
25	134	SANTO TOMÁS AJUSCO	ESTADIO AZTECA	3
26	134-A	PARRES	ESTADIO AZTECA	3
27	134-B	TOPILEJO	ESTADIO AZTECA	3
28	134-C	SANTO TOMÁS AJUSCO	METRO UNIVERSIDAD	2
29	134-D	TOPILEJO	METRO UNIVERSIDAD	3
30	39-A	METRO SAN LÁZARO	XOCHIMILCO/BOSQUE DE NATIVITAS POR CAFETALES	9
31	39-B	METRO SAN LÁZARO	XOCHIMILCO/BOSQUE DENATIVITAS POR MIRAMONTES	8
32	141	VILLA MILPA ALTA	METRO TASQUEÑA/TLÁHUAC	6
33	142	TULYEHUACO	XOCHIMILCO/PALMAS	8
34	143	VILLA MILPA ALTA	METRO TASQUEÑA/XOCHIMILCO	7
35	144	SAN PABLO OZTOTEPEC	XOCHIMILCO/PALMAS	5
36	144-C	SAN SALVADOR CUAUHTENCO	VILLA MILPA ALTA	6
37	145	PEDREGAL DE SAN FRANCISCO	XOCHIMILCO/PALMAS	6
38	145-A	SANTIAGO TEPALCATLAPAN	REPUBLICA DEL SALVADOR	5
39	146	SAN MIGUEL TEHUIZCO	XOCHIMILCO/PALMAS	6
40	147	SAN BARTOLOME XICOMULCO	XOCHIMILCO/PALMAS	6
41	148	SAN NICOLAS TETELCO	METRO TASQUEÑA	6
42	149	MIXQUIC	METRO TASQUEÑA	7
43	46-C	LIENZO CHARRO/SANTA CATARINA	CENTRAL DE ABASTO	27
44	47-A	ALAMEDA ORIENTE	XOCHIMILCO/BOSQUE DE NATIVITAS	23
45	52-C	METRO SANTA MARTHA	METRO ZAPATA	36
46	57-C	METRO CUATRO CAMINOS	METRO CONSTITUCIÓN DE 1917	8
47	159	PALMITAS	METRO CONSTITUCIÓN DE 1917	6
48	161	AMPLIACIÓN SANTIAGO	METRO CONSTITUCIÓN DE 1917	8
49	161-C	PALMAS	METRO CONSTITUCIÓN DE 1917	3
50	161-D	COL. BUENAVISTA	CENTRAL DE ABASTO	5
51	161-E	SAN JOSE BUENAVISTA	METRO CONSTITUCIÓN DE 1917	3
52	161-F	BARRANCA DE GUADALUPE	METRO CONSTITUCIÓN DE 1917	3
53	162	SANTA CATARINA	METRO CONSTITUCIÓN DE 1917	10
54	162-B	CAMPESTRE POTRERO	METRO ZARAGOZA/METRO TEPALCATES	3
55	162-D	SANTA CATARINA	METRO UNIVERSIDAD	12
56	163	SAN MIGUEL TEOTONGO/GUADALUPE	METRO ZARAGOZA/METRO TEPALCATES	4
57	163-A	SAN MIGUEL TEOTONGO/TORRES	METRO ZARAGOZA/METRO TEPALCATES	9
58	163-B	SAN MIGUEL TEOTONGO/AVISADERO	METRO ZARAGOZA/METRO TEPALCATES	6
59	164	COL. MIGUEL DE LA MADRID	METRO ZARAGOZA/METRO TEPALCATES	4
60	165-A	EJÉRCITO DE ORIENTE	METRO CONSTITUCIÓN DE 1917	7
61	166	AVISADERO/COL. IXTLAHUACAN	METRO ZARAGOZA/METRO TEPALCATES	5
62	167	AVISADERO/COL. MIRAVALLE	METRO ZARAGOZA/METRO TEPALCATES	4
63	1-D	METRO SANTA MARTHA	METRO MIXCOAC	11

DISTRIBUCIÓN DE PARQUE VEHICULAR ASIGNADO POR RUTA

N°	Ruta	Origen	Destino	Ordinario
64	33	LEÓN DE LOS ALDAMA	METRO CHABACANO	10
65	37	U.C.T.M. ATZACOALCO	CARMEN SERDÁN	22
66	39	SAN LÁZARO	CARMEN SERDÁN	21
67	43	SAN FELIPE/LEÓN DE LOS ALDAMA	CENTRAL DE ABASTO	9
68	76-A	CENTRO COMERCIAL SANTA FE	LA VILLA/CANTERA POR REFORMA	10
69	168	ARENAL 4A. SECCIÓN	METRO PANTITLÁN	8
70	23	COL. EL TEPETATAL (EL CHARCO)	METRO LA RAZA	8
71	25	ZACATENCO	METRO POTRERO	8
72	27-A	RECLUSORIO NORTE	METRO HIDALGO/ALAMEDA CENTRAL	6
73	101	COL. LOMAS DE CUAUTEPEC	METRO INDIOS VERDES	3
74	101-A	AMPLIACIÓN MALACATES	LA VILLA FERROPLAZA	5
75	101-B	COL. FORESTAL	LA VILLA FERROPLAZA	5
76	101-D	COL. COCOYOTES (LA BRECHA)	LA VILLA FERROPLAZA	5
77	102	COL. COCOYOTES (LA BRECHA)	METRO INDIOS VERDES	3
78	103	AMPLIACION MALACATES	METRO LA RAZA	3
79	104	COL. EL TEPETATAL (EL CHARCO)	METRO POTRERO	3
80	108	COL. EL TEPETATAL (EL CHARCO)	METRO INDIOS VERDES	3
81	11	ARAGÓN	METRO CHAPULTEPEC POR LA VILLA	0
82	11-A	ARAGÓN	METRO CHAPULTEPEC POR AV. 604	7
83	12	ARAGÓN	PANTEÓN SAN ISIDRO	6
84	18	METRO CUATRO CAMINOS	COL. MOCTEZUMA 2A. SECCIÓN	6
85	19	METRO EL ROSARIO	PARQUE MÉXICO POR CUITLÁHUAC	2
86	19-A	METRO EL ROSARIO	PARQUE MÉXICO POR PLAN DE SAN LUIS	2
87	57-A	METRO CUATRO CAMINOS	METRO CONSTITUCIÓN DE 1917	3
88	59	METRO EL ROSARIO	METRO CHAPULTEPEC	3
89	59-A	METRO EL ROSARIO	SULLIVAN	3
90	107	METRO EL ROSARIO	METRO TACUBA	3
91	107-B	METRO MARTÍN CARRERA	METRO TACUBA POR CEYLÁN	3
TOTAL				599

BIBLIOGRAFÍA

- [1] Secretaría de Movilidad. Cinco siglos de Transporte en la Ciudad de México. http://data.semovi.cdmx.gob.mx/wb/stv/acerca_de_setravi.html. Accedido el 2018-08-16.
- [2] Gabriela Sánchez. Transformarán a la RTP en el Sistema M1 con unidades de nueva generación, 2015-01-25. <https://www.jornada.com.mx/2015/01/25/capital/030n2cap#>. Accedido el 2018-12-05.
- [3] Cláudia Sheinbaum. Plan estratégico de movilidad de la Ciudad de México 2019, 2019-04-15. <https://semovi.cdmx.gob.mx/storage/app/media/uploaded-files/plan-estrategico-de-movilidad-2019.pdf>. Accedido el 2019-05-01.
- [4] Israel Colchado. La movilidad urbana en la Ciudad de México: un problema complejo, 2017-04-28. <https://www.c3.unam.mx/boletines/boletin5.html>. Accedido el 2018-11-22.
- [5] Johanna Camargo, Martha Helena Carrillo & Jairo R. Montoya-Torres. Multi-criteria approaches for urban passenger transport systems: a literature review. *Annals of operations research*, 226(1):69–87, 2015.
- [6] T. Moore & J. Pulidindi. Understanding urban transportation systems: an action guide for city leaders. national league of cities, 2013.
- [7] Réka Albert & Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

- [8] Leonhard Euler. Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, pages 128–140, 1741.
- [9] Hamdy A. Taha. *Operations Research: An Introduction*. Pearson Educación de México, 9th edition edition, 2011.
- [10] R. Solomonoff & A. Rapoport. Connectivity a random nets. *The Structure and Dynamics of Networks*, pages 27–37, 1951.
- [11] Paul Erdős & Alfred Rényi. On the evolution of random graphs. *The Structure and Dynamics of Networks*, pages 38–82, 1960.
- [12] J. Travers & S. Milgram. An experimental study of the small world problem. *The Structure and Dynamics of Networks*, pages 130–148, 1969.
- [13] M. Bazaraa, J. Jarvis & H. Sherali. *Linear Programming and Netwok Flows*. Wiley, 4th edition edition, 2010.
- [14] Albert-László Barabási & Réka Albert. Emergence of scaling in random networks. *Science*, pages 509–512, 1999.
- [15] Albert-László Barabási & Eric Bonabeau. Scale-free networks. *Scientific american*, 288(5):60–69, 2003.
- [16] Sergei N. Dorogovtsev & José F.F. Mendes. *Evolution of networks: From biological nets to the Internet and WWW*. OUP Oxford, 2013.
- [17] Mark E.J. Newman. Assortative mixing in networks. *Physical review letters*, 89(20):208701, 2002.
- [18] Vito Latora & Massimo Marchiori. Efficient behavior of small-world networks. *Physical review letters*, 87(19):198701, 2001.
- [19] Paolo Crucitti, Vito Latora, Massimo Marchiori & Andrea Rapisarda. Efficiency of scale-free networks: error and attack tolerance. *Physica A: Statistical Mechanics and its Applications*, 320:622–642, 2003.
- [20] Duncan J. Watts & Steven H. Strogatz. Collective dynamics of ‘small-world’ networks. *nature*, 393(6684):440, 1998.

- [21] R.T. Rockafellar. Networks flows and monotropic optimization. [*Athena Scientific, Belmont, Massachusetts*], 1998.
- [22] R. Ahuja, T. Magnanti & J. Orlin. Network Flows: Theory, Algorithms and Applicattions. [*Prentice Hall*], 1993.
- [23] S. Carmi, Z. Wu, E. López, H.E. Stanley & S. Havlin. Transport between multiple sources in complex networks. *The European Physical Journal B*, pages 165–174, 2007.
- [24] S. Derrible & C. Kennedy. Characterizing metro networks: state, form, and structure. *Springer Science*, pages 275–297, 2009.
- [25] Carlos Romero. *Análisis de las decisiones multicriterio*. Isdefe, 1996.
- [26] Katta G Murty. *Linear programming*. Springer, 1983.
- [27] Carlos Romero. El enfoque multiobjetivo en los modelos matemáticos de planificación de cultivos. 1981.
- [28] M Wegener. Overview of land use transport models. handbook of transport geography and spatial systems. In *Handbook 5 of the Handbook in Transport*. Pergamon/Elsevier Science Kidlington, 2004.
- [29] Carlos Gershenson. Self-organizing urban transportation systems. In *Complexity Theories of Cities Have Come of Age*, pages 269–279. Springer, 2012.
- [30] Attila A. Kovacs, Sophie N. Parragh & Richard F. Hartl. The multi-objective generalized consistent vehicle routing problem. *European Journal of Operational Research*, 247(2):441–458, 2015.
- [31] Xiaolei Ma, Haiyang Yu, Yunpeng Wang & Yinhai Wang. Large-scale transportation network congestion evolution prediction using deep learning theory. *PloS one*, 10(3):e0119044, 2015.
- [32] Dragan Pamučar, Ljubomir Gigović, Goran Ćirović & Miodrag Regodić. Transport spatial model for the definition of green routes for city logistics centers. *Environmental Impact Assessment Review*, 56:72–87, 2016.

- [33] Carlos Gershenson & Luis A. Pineda. Why does public transport not arrive on time? the pervasiveness of equal headway instability. *PloS one*, 4(10):e7292, 2009.
- [34] Fernando Lámbarry, Mara Maricela Trujillo & Cintia Guadalupe Cumbres. Stress from an administrative perspective in public transport drivers in mexico city: Minibus and metrobus. *Estudios Gerenciales*, 32(139):112–119, 2016.
- [35] Instituto Nacional de Estadística y Geografía. Encuesta origen destino en hogares de la zona metropolitana del valle de México 2017, 2018-02-19. https://www.inegi.org.mx/contenidos/saladeprensa/boletines/2018/EstSociodemo/OrigenDest2018_02.pdf. Accedido el 2019-03-02.
- [36] Miguel López. El transporte de pasajeros y el sistema vial en la ciudad de México. *México: Instituto de Investigaciones Jurídicas de la UNAM*, 2011.
- [37] Sebastián Varela. Urban and suburban transport in Mexico City. in the international transport forum discussion paper, 2015. https://www.itf-oecd.org/sites/default/files/docs/varela_0.pdf. Accedido el 2019-01-15.
- [38] Paolo Serafini. Simulated annealing for multi objective optimization problems. In *Multiple criteria decision making*, pages 283–292. Springer, 1994.
- [39] Gerald W. Evans. An overview of techniques for solving multiobjective mathematical programs. *Management Science*, 30(11):1268–1282, 1984.
- [40] Thomas Erlebach, Klaus Jansen & Eike Seidel. Polynomial-time approximation schemes for geometric graphs. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 671–679. Society for Industrial and Applied Mathematics, 2001.
- [41] Matthias Ehrgott. *Multicriteria optimization*, volume 491. Springer Science & Business Media, 2005.
- [42] Dante Zanarini. Algoritmos greedy, 2009-06-25. <https://www.yumpu.com/es/document/view/14171352/algoritmos-greedy/19>. Accedido 2018-11-28.
- [43] Rafael Martí Cunquero. Algoritmos heurísticos en optimización combinatoria. *Valencia: Universidad de Valencia. Retrieved*, 11(01):2012, 2003.

- [44] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller & Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.
- [45] Scott Kirkpatrick, C. Daniel Gelattand & Mario P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.
- [46] Fred Glover. Tabu search and adaptive memory programming—advances, applications and challenges. In *Interfaces in computer science and operations research*, pages 1–75. Springer, 1997.
- [47] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal & Tanaka Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.
- [48] Kalyanmoy Deb, Samir Agrawal, Amrit Pratap & Tanaka Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In *International conference on parallel problem solving from nature*, pages 849–858. Springer, 2000.
- [49] Hui Li & Qingfu Zhang. Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. *IEEE transactions on evolutionary computation*, 13(2):284–302, 2009.
- [50] Sistema de Movilidad 1. Correo institucional. <http://www.sm1.cdmx.gob.mx/comunicacion>. Accedido 2018-07-28.