



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
INGENIERÍA DE SISTEMAS – TRANSPORTE

**HIPER-RUTAS MÍNIMAS MULTIMODALES EN TRANSPORTE PÚBLICO Y PRIVADO
PARA UN SISTEMA MULTIUSUARIO: CASO DE UNA SUBRED DEL DF**

TESIS
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA

PRESENTA:
JOSÉ FRANCISCO SALGADO RODRÍGUEZ

TUTOR PRINCIPAL
DRA. ANGÉLICA DEL ROCÍO LOZANO CUEVAS, INSTITUTO DE INGENIERÍA

MÉXICO, D. F. SEPTIEMBRE 2014



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



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.

JURADO ASIGNADO:

Presidente: Dr. Ricardo Aceves García
Secretario: M.I. José Antonio Rivera Colmenero
Vocal: Dra. Angélica del Rocío Lozano Cuevas
1 er. Suplente: M.I. Francisco Granados Villafuerte
2 do. Suplente: M.I. Luis Alejandro Guzman Castro

Lugar o lugares donde se realizó la tesis:

FACULTAD DE INGENIERÍA, UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

TUTOR DE TESIS:

DRA. ANGÉLICA DEL ROCÍO LOZANO CUEVAS

FIRMA

Índice

Introducción	1
Capítulo 1 . Problemas de transporte	4
1.1. Introducción.....	4
1.2. Transporte público y sostenibilidad.....	8
1.3. Transporte público en la Ciudad de México.....	9
1.3.1. Modos de transporte público en la Ciudad de México.....	10
1.3.2. Problemas de transporte en la Ciudad de México.....	21
1.3.3. Sostenibilidad del transporte en la Ciudad de México.....	24
1.3.4. Retos del transporte público sostenible en la Ciudad de México.....	28
1.4. Selección del área de estudio.....	29
1.4.1. Características socioeconómicas y demográficas.....	31
1.4.2. Modos de transporte público existentes en la zona de estudio.....	33
1.4.3. Conexión entre el transporte público con el privado en la zona de estudio.....	36
1.4.4. Implementación de un SIT en la zona de estudio.....	39
Capítulo 2 . Sistemas Inteligentes de Transporte	41
2.1. Definición de SIT.....	42
2.2. Clasificación de los SIT.....	44
2.2.1. Sistema Avanzado de Gestión del Transporte (SAGT).....	45
2.2.2. Sistema Avanzado de Información al Viajero (SAIV).....	47
2.2.3. Sistema Avanzado de Control de Vehículos (SACV).....	49
2.2.4. Operaciones de Vehículos Comerciales (OVC).....	51
2.2.5. Sistema Avanzado de Transporte Público (SATP).....	53
2.2.6. Algunas aplicaciones de los SIT.....	56
2.3. Organismos SIT alrededor del mundo.....	62
2.4. Congresos Internacionales.....	65
2.5. Sistema Avanzado de Información al Viajero (SAIV).....	68
2.5.1. Historia de los SAIV.....	68
2.5.2. Clasificación de los Sistemas Avanzados de Información al Viajero.....	71

Capítulo 3 . Hiperrutas viables multimodales	78
3.1. Teoría de grafos	78
3.2. Representación de redes	84
3.3. Viabilidad en rutas multimodales	89
3.4. Hipergrafos.....	91
3.5. Viabilidad en hiperrutas multimodales	96
3.6. Algoritmo Shortest Viable Hyperpath Problem.....	97
Capítulo 4 . Hiperred del área de estudio	105
4.1. Hiperred de transporte público	106
4.1.1. Hiperred del Metro	106
4.1.2. Hiperred del Metrobús.....	107
4.1.3. Hiperred de autobuses RTP.....	107
4.1.4. Hiperred del Trolebús y Tren Ligero.....	107
4.2. Red de transporte privado	112
4.3. Diagrama de estados para la zona de estudio.....	128
Capítulo 5 . Implementación del SAIV y obtención de resultados	131
5.1. Programas utilizados en la implementación del SAIV	131
5.2. Transformación de la tabla de datos asociada a la hiperred a un archivo separado por comas (CSV).....	132
5.3. Creación de la base de datos en PostgreSQL y llenado de las tablas a través de JDBC.....	133
5.3.1. Creación de las tablas en PostgreSQL	133
5.3.2. Adecuación de las tablas con JDBC	136
5.4. Configuración del Servidor Web	139
5.5. Programación del sistema en Java.....	141
5.5.1. Configuración de Eclipse JEE	142
5.6. Clases programadas	147
5.7. Incorporación de un servidor de mapas.....	149
5.8. Caso de estudio del sistema SAIV.....	149
5.9. Manejo del SAIV por el usuario final.....	150
5.10. Análisis de resultados.....	150
5.10.1. Ejemplo 1. Metro Iztapalapa-Metrobús Poliforum	151

5.10.2. Ejemplo 2. Metrobús Río frío-Tren ligero Periférico	153
5.10.3. Ejemplo 3. Automóvil Periférico/Picacho-Metro Tepalcates	154
5.10.4. Ejemplo 4. Automóvil Eje 10Sur-RTP Centenario	157
5.10.5. Ejemplo 5. Metro Lázaro Cárdenas-Automóvil Periférico Sur	158
5.10.6. Ejemplo 6. Metrobús Amores-Automóvil Circuito Interior	160
Conclusiones y recomendaciones	163
Referencias	167
Anexos.....	173
A. Estacionamientos en la zona de estudio	173
B. JDBC para estructuras de datos	174
C. Creación de tablas en SQL.....	181
D. Clase SVHP_Servlet.....	182
E. Clase SVHP	184
F. Clase Nodo	197
G. Clase escribeNodos	198
H. Clase escribeltinerario	200
I. Clase escribeLonLat.....	202

Índice de tablas

Tabla 1.1. Características de las líneas del Metro	11
Tabla 1.2. Trenes por línea.....	11
Tabla 1.3. Red de rutas del servicio expreso, RTP	13
Tabla 1.4. Líneas del Sistema de Transporte Público Trolebús	16
Tabla 1.5. Modalidad y tarifa de los taxis para el año 2014	19
Tabla 2.1. Clasificación de los subsistemas SIT	44
Tabla 2.2. Algunos lugares donde se ha implementado el sistema SIPARK SSD	59
Tabla 2.3. Aplicaciones de los SIT.....	59
Tabla 2.4. Servicios al usuario de la arquitectura SIT México, listados por orden de importancia	64
Tabla 2.5. Ediciones del Congreso Mundial sobre SIT	67
Tabla 5.1. Programas utilizados para la implementación el sistema	131

Índice de figuras

Figura 1.1. Las diez ciudades más pobladas del mundo en 2013	5
Figura 1.2. Red del Sistema de Transporte Colectivo Metro	12
Figura 1.3. Circuito Bicentenario (fragmento), autobuses RTP	14
Figura 1.4. Línea del Tren Ligero del Distrito Federal.....	15
Figura 1.5. Red del Metrobús.....	18
Figura 1.6. Reparto modal para viajes en dos tramos	22
Figura 1.7. Porcentaje de viajes mixtos.....	23
Figura 1.8. Indicadores del cumplimiento de la NOM-020-SSA1-1993.....	27
Figura 1.9. Frecuencia anual de días con concentraciones superiores a los indicados en la NOM-020-SSA1-1993	27
Figura 1.10. Delegaciones del área de estudio	30
Figura 1.11. Grado de marginación en el área de estudio.....	33
Figura 2.1. Elementos que conforman un SAIV.....	49
Figura 2.2. Componentes de un sistema AVCS	50
Figura 2.3. Elementos utilizados en un OVC	52
Figura 2.4. Sistema RTPI en el Reino Unido.....	55
Figura 2.5. Sistema RPTI. Ruta B63, Nueva York.....	56
Figura 2.6. Sistema SIPark SSD	57
Figura 2.7. Diagrama de funcionamiento del sistema SIPark SSD	58
Figura 2.8. Aeropuerto de Múnich.....	58
Figura 2.9. Logotipos de algunos organismos ITS	63
Figura 2.10. ITS México	65
Figura 3.1. Grafo dirigido	80
Figura 3.2. Red dirigida.....	80
Figura 3.3. Camino dirigido	80
Figura 3.4. Ruta dirigida.....	80
Figura 3.5. Ciclo dirigido	81
Figura 3.6. Ejemplo de dos árboles.....	82
Figura 3.7. Bosque.....	83
Figura 3.8. Árbol con raíz.....	83
Figura 3.9. Ejemplo de una red.....	84
Figura 3.10. Matriz de incidencia nodo-arco.....	85
Figura 3.11. Matriz de adyacencia nodo-nodo.....	86
Figura 3.12. Lista de adyacencia.....	87
Figura 3.13. Estrella saliente.....	88
Figura 3.14. Estrella entrante	89
Figura 3.15. Representación de hipergrafos	91
Figura 3.16. Procedimiento Shortest Viable Hyperpath Problem.....	102
Figura 3.17. Procedimiento Arc Concatenation	103
Figura 3.18. Procedimiento H-Arc Concatenation	104
Figura 3.19. Procedimiento Determine-S	104
Figura 4.1. Hiperred del Metro	108
Figura 4.2. Hiperred del Metrobús.....	109

Figura 4.3. Hiperred de autobuses RTP	110
Figura 4.4. Hiperredes del Tren Ligero y Trolebús	111
Figura 4.5. Hiperred de transporte público para el área de estudio	112
Figura 4.6. Vialidades principales y estacionamientos del área de estudio	113
Figura 4.7. Vialidades y estacionamientos considerados	114
Figura 4.8. Añadir Nueva Marca de posición.....	115
Figura 4.9. Ventana Nueva Marca de posición.....	115
Figura 4.10. Ventana Ícono	116
Figura 4.11. Ventana Agregar ruta	117
Figura 4.12. Ventana Guardar archivo	117
Figura 4.13. Opción Añadir capa vectorial del menú Capa.....	118
Figura 4.14. Ventana Añadir capa vectorial.....	119
Figura 4.15. Opción Guardar como del menú Capa	119
Figura 4.16. Ventana Guardar capa vectorial como	120
Figura 4.17. Ventana ESRI Shapefile.....	122
Figura 4.18. Ventana Import coordinates	122
Figura 4.19. Ventana Export Geography	123
Figura 4.20. Map Layers	123
Figura 4.21. Ventana Layers	123
Figura 4.22. Menú para seleccionar la herramienta Map Editing.....	125
Figura 4.23. Cuadro de herramientas Map Editing	125
Figura 4.24. Menú para seleccionar la opción Modify table	126
Figura 4.25. Ventana Modify Table	126
Figura 4.26. Opción Edit line attributes	127
Figura 4.27. Ventana para importar datos CSV en PostgreSQL.....	127
Figura 4.28. Estructura de la tabla asociada a la hiperred.....	127
Figura 4.29. Diagrama de estados	129
Figura 5.1. Archivo separado por comas (CSV)	132
Figura 5.2. Creación de una nueva tabla en PostgreSQL	134
Figura 5.3. Asignación del nombre a la nueva tabla.....	135
Figura 5.4. Creación de columnas para una tabla en PostgreSQL.....	135
Figura 5.5. Datos importados en PostgreSQL.....	136
Figura 5.6. Transferencia de los datos de PostgreSQL a Java.....	137
Figura 5.7. Creación de la tabla de arcos desde Java.....	138
Figura 5.8. Homologación del campo dirección en la tabla asociada a los arcos	139
Figura 5.9. Estructura general del sistema web.....	140
Figura 5.10. Apache Tomcat.....	141
Figura 5.11. Registro del nuevo servidor en Eclipse	142
Figura 5.12. Selección del tipo de servidor y su ubicación	143
Figura 5.13. Visualización del servidor en el explorador.....	144
Figura 5.14. Proyecto web dinámico en Eclipse	145
Figura 5.15. Registro del proyecto dinámico	146
Figura 5.16. Estructura del proyecto java	147
Figura 5.17. Página principal del sistema web	151
Figura 5.18. Hiperruta Metro Iztapalapa - Metrobús Polifórum	152

Figura 5.19. Hiperruta Metrobús Río Frío - Tren Ligero Periférico.....	154
Figura 5.20. Hiperruta Automóvil Periférico Sur N/S - Metro Tepalcates	156
Figura 5.21. Hiperruta Automóvil Eje 10 Sur E/O - RTP Centenario O/P.....	158
Figura 5.22. Hiperruta Metro Lázaro Cárdenas – Automóvil Periférico Sur S/N.....	159
Figura 5.23. Hiperruta Metrobús Amores – Automóvil Circuito Interior S/N	161

Agradecimientos

A la máxima casa de estudios del país y a su Facultad de Ingeniería, ejemplos de grandeza y generadoras de cambio.

A la Dra. Angélica del Rocío Lozano Cuevas, por su invaluable ayuda, vocación, guía y enseñanzas.

A mis maestros y sinodales, Dr. Ricardo Aceves García, M.I. José Antonio Rivera Colmenero, M.I. Francisco Granados Villafuerte y M.I. Luis Alejandro Guzman Castro, por sus consejos y recomendaciones.

A mis compañeros Alejandro Benítez, Minerva Soto y David López, por su colaboración en el desarrollo del presente trabajo.

A mis abuelitas Chuy y Mary, quienes desde el cielo siguen guiando mi camino, como lo hicieron en vida.

A Julieta Rodríguez Quezada, José Francisco Salgado Rico y Luis Ángel Salgado Rodríguez, mi familia, mi fortaleza, mi fuente de inspiración y amor inagotable.

A Nancy, por su apoyo incondicional, confianza y amor.

Cuauhtlehuani

Introducción

Los problemas relacionados con el transporte, como la disminución en la velocidad de recorrido y el aumento en los tiempos de viaje, aumento en las emisiones contaminantes, etc. y el deterioro generalizado en la salud de las personas que habitan en las grandes urbes como lo es la Zona Metropolitana del Valle de México (ZMVM), hace evidente que las supuestas soluciones planteadas hasta el momento no sean las adecuadas o en el mejor de los casos, solo llegan a ser parcialmente convenientes.

Con mucha frecuencia las acciones implementadas, se llevan a cabo desde el lado de la oferta y no desde la demanda. Los Sistemas Inteligentes de Transporte (SIT), buscan hacer un mejor uso de los recursos existentes, incorporando tecnologías de software y hardware. Al tratar de ordenar dichos recursos, en lugar de seguir incrementando la complejidad de los sistemas de transporte, se logran soluciones económicamente factibles.

La construcción de grandes obras de infraestructura destinadas al uso de modos privados de transporte, incentiva a los usuarios a utilizar más los modos privados que los públicos, esto provoca (entre otros factores, como los financiamientos automotrices) que cada vez sea mayor el número de viajes que se realizan en automóvil. Además, las deficiencias que presenta el transporte público y la baja integración entre los modos público y privado, provocan que los usuarios del automóvil, no consideren como una opción viable la realización de viajes que incluyan otros modos de transporte.

Es importante que las personas, puedan planear su viaje antes de realizarlo y que las decisiones que tomen, los lleven a reducir los tiempos de recorrido y costos. La falta de itinerarios en el transporte público de la ZMVM, el desconocimiento del tiempo de llegada y la frecuencia con la que los vehículos arriban a una parada, la carencia de paradas establecidas para ciertos modos de transporte, son algunas de las características del transporte público, que impiden al usuario la planeación de su viaje y el ahorro de tiempo.

El objetivo de la presente tesis es desarrollar un sistema multiusuario que permita a la gente, planear su itinerario, calculando las distintas posibilidades de viaje, utilizando tanto el automóvil como el transporte público (hiperrutas mínimas multimodales y viables), para

cualquier par origen - destino que introduzca el usuario, habiéndose adaptado y programado el algoritmo Shortest Viable Hyperpath Problem (SVHP), de Lozano y Storchi (2002).

Para lograr dicho objetivo la tesis se organizó de la siguiente manera. En el primer capítulo se plantea un contexto general sobre los problemas de transporte y movilidad, sostenibilidad en el transporte público, se describen las características de los modos de transporte en la Ciudad de México y se describen los problemas que existen para cada modo explicado, se determina la zona de estudio de forma geográfica y a través de distintos indicadores socioeconómicos, finalmente se concluye con el análisis sobre la interacción entre el transporte público y el transporte privado y la implementación de un SIT en la zona de estudio.

En el segundo capítulo se explica lo relacionado con los Sistemas Inteligentes de Transporte, su definición, clasificación, historia, los subsistemas que conforman a los SIT, y se explica de manera detallada cada uno de éstos, haciendo mayor énfasis en los Sistemas Avanzados de Información al Viajero (SAIV), ya que el sistema multiusuario programado para esta tesis es de este tipo.

En el tercer capítulo se explican conceptos de teoría de grafos, representación de redes, estructuras de datos, redes e hiperredes multimodales, y viabilidad, ya que al incluirse un modo que presenta restricciones como el automóvil debe estudiarse este concepto. Finalmente se explica a detalle el algoritmo SVHP y las modificaciones realizadas al mismo para poder utilizarse en el desarrollo del SAIV.

En el cuarto capítulo se analiza el proceso de realización de las hiperredes de transporte público y la red de transporte privado, y se establece el diagrama de estados utilizado para verificar la viabilidad en las rutas generadas por el SAIV.

En el quinto capítulo se explica el proceso de construcción del SAIV, comenzando por el diseño y desarrollo de la base de datos, posteriormente la configuración del servidor y la codificación de las clases en Java. También, se presentan algunos ejemplos de los resultados generados por el SAIV, cada ejemplo consta de una o más hiperrutas mínimas viables multimodales, las cuales son descritas en texto y representadas gráficamente.

Posteriormente se presentan las conclusiones y recomendaciones, y las referencias.

Finalmente se incluyen nueve anexos donde se muestra la información de los estacionamientos en el área de estudio, y el código de las distintas clases programadas en Java utilizadas para la creación y adecuación de las tablas en PostgreSQL, y para el funcionamiento del algoritmo SVHP, como parte del SAIV.

Capítulo 1 . Problemas de transporte

Este capítulo se subdivide en cuatro secciones, en la primera se presenta una introducción a los distintos problemas asociados al transporte y a la movilidad, posteriormente en la segunda sección se muestran los elementos necesarios para garantizar que el transporte sea sostenible, en la tercera sección se analizan las características de los modos de transporte público más utilizados en la Ciudad de México, así como, los problemas que presenta el sistema de transporte y sus retos a futuro. Finalmente en la cuarta sección, se analiza el área de estudio a través de sus características socioeconómicas y demográficas, los modos de transporte existentes, y la conexión entre el transporte público con el privado.

1.1. Introducción

Los problemas de movilidad asociados con el transporte, se han hecho presentes en la mayor parte del mundo, afectando el entorno social de las personas, su salud física y emocional, la economía de los países y de las personas, y el equilibrio ecológico del planeta, con los respectivos daños que se ocasionan a todos los seres vivos.

El rápido crecimiento poblacional en las grandes urbes, es producto de la combinación de muchos factores entre los que destacan; la concentración de actividades económicas, la creencia de que los trabajos mejor remunerados se encuentran en las grandes ciudades y que cualquiera puede encontrar trabajo en la ciudad, ya que las zonas urbanas y su entorno son el motor económico del país, pues generan el 95% del PIB (Lobo, 2012), mayor oferta educativa, oferta de una amplia gama de servicios, construcción de edificios habitacionales que permiten una gran concentración de personas en un espacio reducido, etc., lo que ocasiona una mayor demanda de servicios, ya sean de vivienda, agua, luz, drenaje, salud y por supuesto de transporte.

Según estimaciones de la Organización de las Naciones Unidas (Lupano y Sánchez, 2009), la tasa de crecimiento de la población urbana total es de 1.9% anual, lo que en escala planetaria equivale a la "aparición" cada día de una nueva ciudad de 160,000 habitantes. La población total de las grandes metrópolis (más de 2 millones de habitantes)

alcanza 940 millones, y algo más de un tercio (alrededor de 350 millones) viven en las 24 megalópolis con más de 10 millones de habitantes.

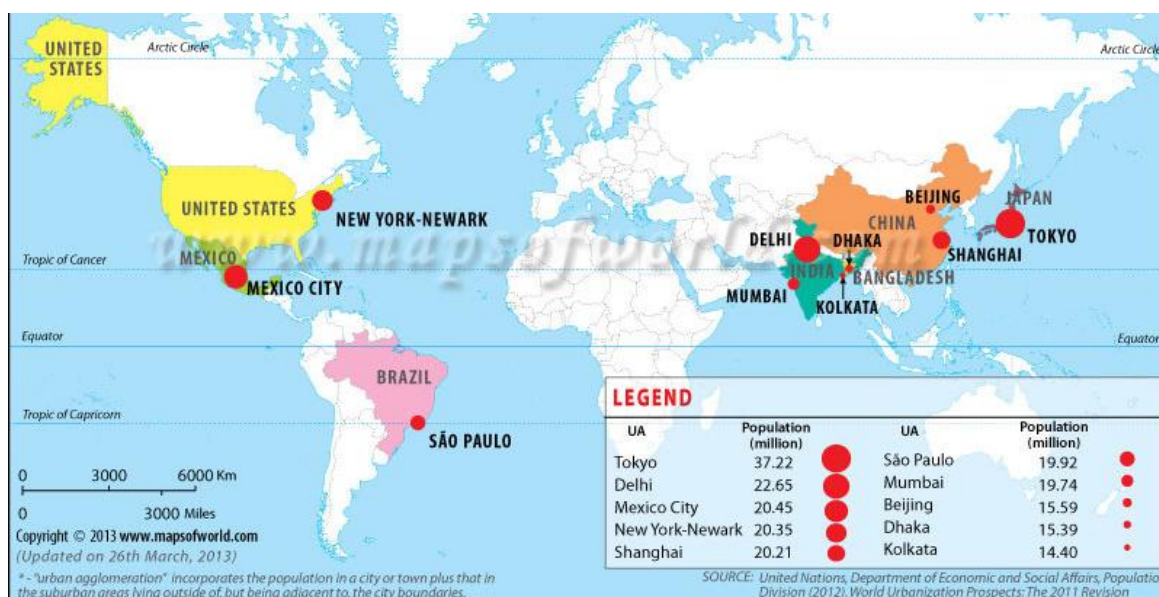


Figura 1.1. Las diez ciudades más pobladas del mundo en 2013

Fuente: (Maps of world, 2013)

Como se observa en la Figura 1.1, la Ciudad de México se encuentra entre las ciudades con mayor población a nivel mundial, este factor entre muchos otros, ha provocado que los problemas de transporte sean tan severos.

Además, del crecimiento poblacional, existen otros factores como el aumento en el parque vehicular, las malas condiciones de las vialidades, la asignación equívoca del número de autobuses y microbuses a cada línea a lo largo del día, el bloqueo de vías por manifestantes, la falta de espacios adecuados para descargar mercancías y para el transbordo de pasajeros, la mala educación vial, la impunidad y la falta de aplicación de los reglamentos, etc. que producen problemas mayores a nivel social, pérdidas económicas incalculables, aumento de la contaminación y daños ambientales, y lo más grave, la mala calidad de vida de los ciudadanos quienes pierden gran parte de su tiempo en transportarse.

Lupano y Sánchez (2009) mencionan que la congestión es causada principalmente por el uso intensivo del automóvil, cuya propiedad se ha masificado en América Latina en las últimas décadas. El automóvil posee ventajas en términos de facilitar la movilidad

personal, otorgar sensación de seguridad y un status o una posición, especialmente en países en vías de desarrollo. Sin embargo, en términos de espacio es poco eficiente para el traslado de personas, al punto que cada ocupante produce en las horas pico unas 11 veces la congestión atribuible a cada pasajero de autobús. Los mismos autores mencionan que la situación se ve agravada en América Latina debido a problemas de diseño y conservación de las calles, un estilo de conducción que no respeta a los demás, una defectuosa información sobre las condiciones del tránsito y la gestión inapropiada de las autoridades involucradas, muchas veces fragmentadas en una multiplicidad de entes.

En la Zona Metropolitana del Valle de México (ZMVM), diariamente se realizan un gran número de viajes con tiempos de recorrido cada vez mayores, según la Encuesta Origen-Destino (EOD, 2007), diariamente se realizan 21.9 millones de viajes y debido a que la expansión urbana se ha dado de forma desordenada, la gente cada vez tiene que recorrer mayores distancias, para trasladarse a su lugar de trabajo o de estudio. Al respecto, el Consejo de Evaluación del Desarrollo Social del Distrito Federal y el Programa Universitario de Estudios sobre la Ciudad de la Universidad Nacional Autónoma de México, menciona que en la Ciudad de México la concentración de actividades económicas, de infraestructura, equipamiento y servicios urbanos en las zonas y corredores centrales de la ciudad, así como la existencia de grandes zonas de vivienda de tipo social y popular ubicadas en las delegaciones periféricas y en municipios del Estado de México desprovistas de los servicios urbanos educativos, de comercio, de salud y lejanas a las fuentes de empleo, han ocasionado el incremento de los desplazamientos diarios de los habitantes de la ciudad (UNAM PUEC, 2011). Según el Fideicomiso para el Mejoramiento de las Vías de Comunicación del Distrito Federal (FIMEVIC), en 1994 se realizaban poco más de 4.2 millones de viajes metropolitanos por día. Se estima que para el 2020 esta cifra será cercana a los 5.6 millones de viajes (28.3 millones de viajes en total) y representará cerca del 20% del total de viajes en la ZMVM (FIMEVIC, 2011), es decir, tienden a predominar más los viajes largos que los viajes cortos.

Según la Encuesta Origen-Destino (EOD, 2007), aproximadamente uno de cada seis viajes que se generan en el Distrito Federal, tiene como destino alguno de los municipios del Estado de México (municipios que pertenecen a la ZMVM), mientras que uno de cada cuatro viajes que se generan en tales municipios, tienen como destino el Distrito Federal.

De los casi 22 millones de viajes, 14.8 millones se realizan en transporte público, 6.8 millones en transporte privado y el resto en transporte público-privado.

Las entidades con mayor densidad de población son el Distrito Federal y el Estado de México con 5,877 habitantes/km² y 627 habitantes/km² respectivamente (INEGI, 2005), lo que conlleva a tener una gran concentración de población en un espacio reducido, además existe un incremento en los desplazamientos diarios de los habitantes de la ciudad (UNAM PUEC, 2011), lo que produce una gran demanda de servicios de transporte que supera lo ofertado.

Como se ha mencionado con anterioridad, los tiempos de traslado y las distancias que recorren las personas han aumentado, y por lo tanto, el congestionamiento es mayor a lo largo del día. Según el FIMEVIC, en un análisis realizado en 1997 y 1998 para las vialidades primarias del DF por la Comisión Metropolitana de Transporte y Vialidad (COMETRAVI) se concluyó que: el 85% de las vialidades primarias tienen mala fluidez, que obliga a una velocidad entre 20 y 21 km/hr, mientras que los vehículos de transporte público se desplazan a 17 km/hr (FIMEVIC, 2011).

En nuestro país los problemas de transporte se han abordado principalmente desde el enfoque de construcción de infraestructura. Al respecto, la CEPAL menciona, que el objetivo principal de una política de transporte con enfoque social debe ser asegurar el acceso de la población a un transporte seguro, rápido, eficiente, cómodo, confiable y asequible para todos los usuarios. Este enfoque se orienta a elevar la calidad de vida de los habitantes y también la competitividad de las ciudades. Para lograr esto, se requiere de una nueva visión y de modelos alternativos de movilidad que den prioridad a las personas y no a los automóviles, a través de darle mayor peso presupuestal a programas de transporte público masivo en vez de darle prioridad a las grandes infraestructuras viales que tienden a saturarse (Lupano y Sánchez, 2009).

Como se menciona, al construir más infraestructura destinada al uso de vehículos particulares, se propicia un uso mayor del automóvil, además existen otros factores que contribuyen a ésta alza, como el aumento poblacional en la ZMVM y la facilidad para adquirir un auto, muestra de ello es que en el Distrito Federal y Estado de México en el año 2001 se tenían 3'759,595 vehículos de motor registrados en circulación, para el año

2010 la cifra aumentó a 7'390,069 vehículos, lo que representa un aumento del 96.5% (INEGI, 2010).

Aunado al aumento de vehículos particulares, el mayor porcentaje de viajes realizados en transporte público, lo ocupan el transporte de baja y mediana capacidad. De acuerdo con los datos aportados por la SETRAVI (SETRAVI, 2012), son los autobuses y microbuses los que acaparan el mayor número de viajes que se realizan en el DF. El transporte concesionado colectivo de pasajeros atiende a casi el 60% de la demanda, transportando por día hábil a más de 12 millones de pasajeros.

1.2. Transporte público y sostenibilidad

El uso de los vehículos privados se incrementa cuando el sistema de transporte público no es lo suficientemente bueno, y la gente prefiere (si se lo puede permitir) viajar en su propio vehículo, porque es percibido como más rápido, confortable y flexible para sus recorridos (Lozano et al., 2005).

Para impulsar el transporte público y que éste vaya ganando terreno respecto al transporte privado, se tienen que garantizar los beneficios, sociales, económicos y ambientales del uso de dicho transporte, es por ello que se ha contemplado la necesidad de implementar, a mediano y largo plazo, un transporte de carácter sostenible, esto se menciona en: "la Declaración de Río sobre el Medio Ambiente y Desarrollo (principios 1, 3, 4, 5), el Programa 21 de la ONU (capítulos 3, 6,7, 9) y el Protocolo de Kyoto (Artículo 2, VII), VIII)" (UNAM, 2011).

Para que el transporte público sea sostenible, es necesario que impacte de manera positiva (o que reduzca los impactos negativos), en el ámbito social, económico y ambiental, según el Consejo de Evaluación del Desarrollo Social del Distrito Federal (UNAM, 2011), la Asociación Internacional de Transporte Público (UITP), plasmó esta realidad en el Protocolo de Toronto 2, firmado y ratificado por 4000 profesionales del transporte durante el Congreso Mundial de la UITP de Toronto en 1999 y en la Declaración de las Ciudades en Movimiento (Comunicado de Melbourne), firmado en

Melbourne, por la División Asia Pacífico en el año 2000. Bajo esas directrices un sistema de transporte público sostenible es aquel sistema que (UNAM, 2011):

- a) Permite satisfacer las necesidades básicas de acceso a los bienes, el trabajo, la educación, el ocio y la información de forma segura para la salud pública y la integridad del medio ambiente, y a través de la equidad entre generaciones y dentro de una misma generación.
- b) Es asequible, opera de manera eficiente, ofrece diferentes modos de transporte y contribuye a una economía dinámica.
- c) Limita las emisiones y desechos dentro de la capacidad del planeta para absorberlos, minimiza el consumo de recursos no renovables, el uso del territorio y la producción de ruido, reutiliza y recicla sus componentes siempre que puede.

El transporte público debe tratarse como un bien social y cultural, y no fundamentalmente como un bien económico (UNAM, 2011), entendiéndolo de esta manera, se le debe dar un mayor apoyo al transporte público para que genere los beneficios sociales que se requieren, es decir, "que garantice el derecho a un medio ambiente sano y a la salud humana de todas las generaciones presentes y futuras, así como de todas las especies del planeta" (Declaración de Río, principios 1, 3, 4; Prog. 21 ONU, Cap. 9B. Numeral 2).

Desafortunadamente, en países como el nuestro, y en general en países en vías de desarrollo, no son comunes las prácticas sustentables, de lo plasmado por la UITP en el Protocolo de Toronto 2, se puede afirmar que no se cumplen en su totalidad ninguno de los tres incisos descritos con anterioridad.

1.3. Transporte público en la Ciudad de México

En esta sección se enlistan las principales características y problemas que definen a los distintos modos de transporte establecidos en la Ciudad de México.

En primera instancia se explican cada uno de los principales modos de transporte público, así como las características de cada uno de ellos, posteriormente se mencionan los principales problemas del sistema de transporte capitalino, después se analiza la sostenibilidad y los retos que presenta ésta para los distintos modos de transporte

1.3.1. Modos de transporte público en la Ciudad de México

En la Ciudad de México existen los siguientes modos de transporte público: Metro, Tren Ligero, Trolebús, autobús suburbano, autobús RTP, colectivos (microbuses y combis), Metrobús, taxis.

1.3.1.1 Metro

Es uno de los modos de transporte público más utilizados. En el año 2011 (STC, 2012), se transportaron a través del Metro un total de mil 487 millones 525 mil 176 usuarios. La red del Metro cuenta con una longitud de 201,388 kilómetros, y los viajes realizados en el Metro en el año 2011, sumaron una distancia total de 41 millones 49 mil 935.68 kilómetros.

El Metro de la ZMVM cuenta con 175 estaciones, 52 estaciones de correspondencia (transbordo), y 22 terminales (STC, 2012) las cuales permiten a los usuarios viajar a través de otros modos de transporte público, como colectivos, tren ligero, trolebús, etc. Las 52 estaciones de transbordo, posibilitan al viajero, trasladarse a través de toda la red del Metro con un sólo boleto y sin tener que salir de las estaciones. En la Tabla 1.1 se muestra para cada una de las 11 líneas que operan hasta el año 2012, la información del total de estaciones, estaciones de transbordo y longitud de la línea.

La red del Sistema de Transporte Colectivo (STC) tiene un total de 355 trenes asignados (322 neumáticos y 33 férreos para la línea A); para proporcionar el servicio a los usuarios en horas pico se tienen en operación 253 trenes, de los 102 trenes restantes se tienen en promedio: 14 trenes en mantenimiento sistemático, 8 trenes en mantenimiento mayor, 4 en rehabilitación y 5 en proyectos especiales, el resto de los trenes se encuentran disponibles para su explotación (STC, 2012). En la Tabla 1.2 se desglosa el número de

trenes por línea. Según el STC, la cantidad de trenes puede variar por una redistribución del parque vehicular en función de las necesidades del servicio (STC, 2012).

Tabla 1.1. Características de las líneas del Metro

Línea			Total de estaciones	Estaciones de transbordo	Longitud [km]
1	Observatorio	Pantitlán	20	7	18.828
2	Cuatro Caminos	Tasqueña	24	5	23.431
3	Indios Verdes	Universidad	21	6	23.609
4	Martín Carrera	Santa Anita	10	6	10.747
5	Pantitlán	Politécnico	13	5	15.675
6	El Rosario	Martín Carrera	11	4	13.947
7	El Rosario	Barranca del Muerto	14	3	18.784
8	Garibaldi	Constitución de 1917	19	5	20.078
9	Tacubaya	Pantitlán	12	5	15.375
A	Pantitlán	La Paz	10	1	17.192
B	Ciudad Azteca	Buenavista	21	5	23.722

Fuente: (STC, 2012)

Tabla 1.2. Trenes por línea

Línea	1	2	3	4	5	6	7	8	9	A	B	Total
Trenes	51	45	58	13	17	18	26	29	33	33	32	355

Fuente: (STC, 2012)

El Metro tiene una cobertura en las siguientes delegaciones del Distrito Federal: Álvaro Obregón, Azcapotzalco, Benito Juárez, Coyoacán, Cuauhtémoc, Gustavo A. Madero, Iztacalco, Iztapalapa, Miguel Hidalgo y Venustiano Carranza; y en los municipios del Estado de México: Ecatepec, Los Reyes, La Paz y Nezahualcóyotl. En la Figura 1.2 se muestra la red del sistema de transporte colectivo Metro.



Figura 1.2. Red del Sistema de Transporte Colectivo Metro

Fuente: (STC, 2012)

1.3.1.2. Autobús RTP

Una de las principales tareas de los autobuses RTP (Red de Transporte de Pasajeros del Distrito Federal), es trasladar a sus usuarios desde las terminales del Sistema Colectivo Metro a las zonas habitacionales que circundan el Distrito Federal. Cabe mencionar que también se conecta con el Metrobús, tren ligero y trolebús. Las rutas que ofrecen los autobuses RTP son: *Rutas de Servicio Atenea, Ruta del Circuito Bicentenario, Rutas de Servicio Expreso y Rutas de Servicio Ordinario.*

Los autobuses del servicio expreso, están equipados con Sistemas de Posicionamiento Global (GPS), con botones de alerta para cualquier contingencia y con radiocomunicación directa entre el operador y el Centro de Control de Servicio, para ofrecer mayor seguridad a los usuarios. La Tabla 1.3, muestra las rutas del sistema expreso de autobuses RTP.

La ruta del Circuito Bicentenario, brinda a los usuarios del sistema de transporte RTP la posibilidad de transbordar hacia el Metro, Metrobús y Trolebús, como lo muestra la Figura 1.3, donde se indica debajo del logotipo de cada estación, la conexión que existe entre el autobús RTP y el modo de transporte en cuestión.

Tabla 1.3. Red de rutas del servicio expreso, RTP

Ruta	Origen	Destino
12	Aragón	Panteón San Isidro
23	El Tepetatal (El charco)	Metro La Raza
39	Puente Negro	Carmen Serdán
39-A	Metro San Lázaro	Xochimilco/Bosque de Nativitas
46-C	Lienzo Charro/Santa Catarina	Central de Abastos
47-A	Alameda Oriente	Xochimilco/Bosque de Nativitas
57-A	Metro Cuatro Caminos	Metro Constitución de 1917
76	Centro Comercial Santa Fe	La Villa/Cantera por Palmas
76-A	Centro Comercial Santa Fe	La Villa/Cantera por Reforma
107-B	Metro Martin Carrera	Metro Tacuba por Ceylán
Cto. Bicentenario	Metro Chapultepec	Metro Chapultepec

Fuente: (RTP, 2012)



Figura 1.3. Circuito Bicentenario (fragmento), autobuses RTP

Fuente: (RTP, 2012)

1.3.1.3. Tren ligero

El Tren Ligero forma parte de la red del Servicio de Transportes Eléctricos del Distrito Federal, el cual opera en el Sur de la Ciudad de México prestando un servicio de transporte no contaminante a la población de las delegaciones Coyoacán, Tlalpan y Xochimilco, brindando su servicio a través de 16 estaciones y 2 terminales (ver Figura 1.4), mediante 20 trenes dobles acoplados, con doble cabina de mando y con capacidad máxima de 374 pasajeros por unidad (STE, 2012). La capacidad ofertada por este modo de transporte ha sido rebasada (El Universal, 2012), es necesaria la adquisición de nuevas unidades de mayor capacidad y la ampliación de ambas terminales, para que se puedan realizar las maniobras de vehículos de mayor tamaño. El resto de las estaciones tienen el tamaño suficiente para recibir vehículos de mayores dimensiones.

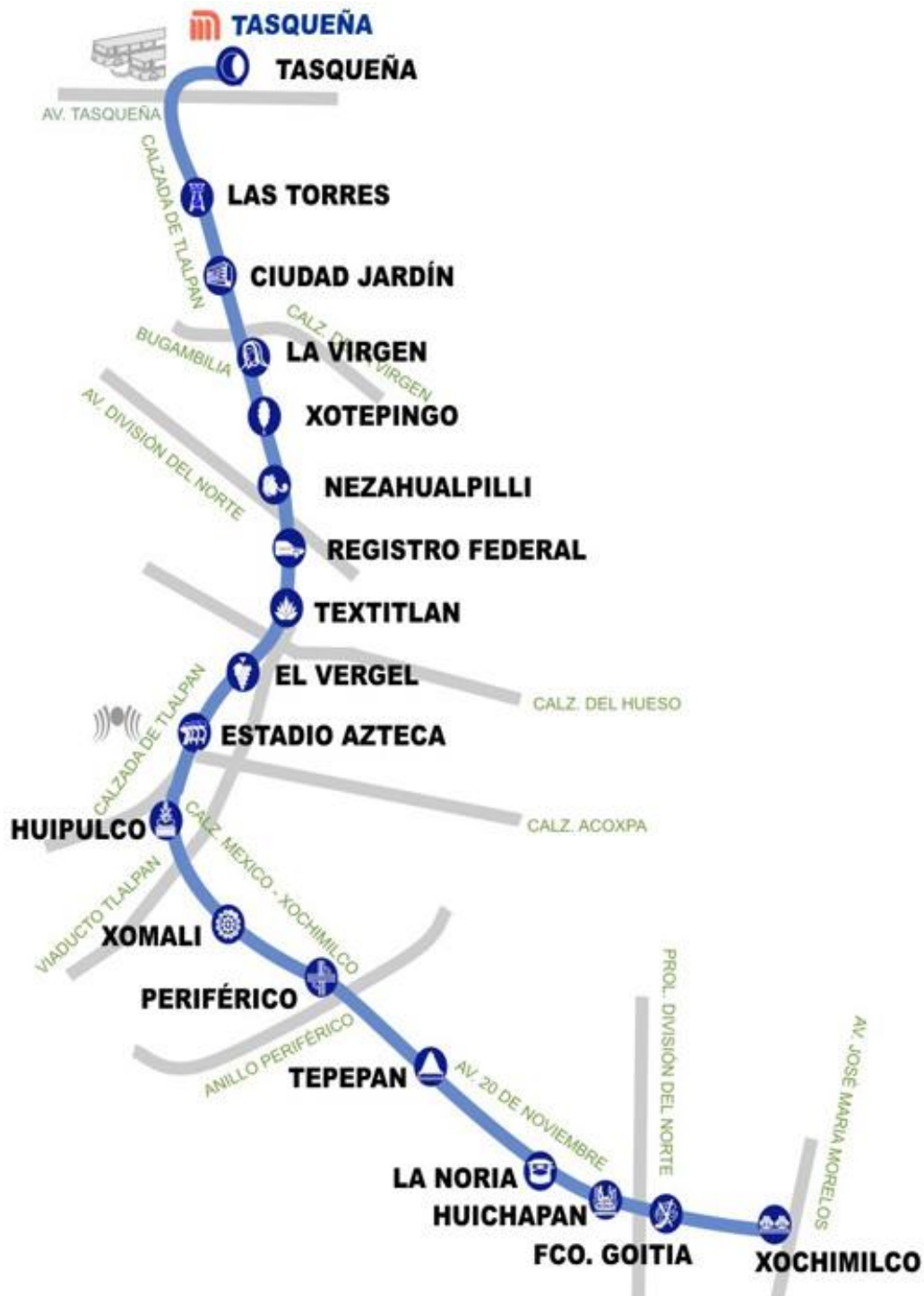










Figura 1.4. Línea del Tren Ligero del Distrito Federal

Fuente: (STE, 2012)

1.3.1.4. Trolebús

La Red del Servicio cuenta en 2014 con 8 Líneas de Trolebuses en servicio (Ver Tabla 1.4) con una longitud de operación 203.64 kilómetros. La flota vehicular programada en 2014 es de 290 trolebuses, los cuales operaban a un intervalo de paso promedio de 4.0 minutos, todas dentro del Distrito Federal (STE, 2014).

Tabla 1.4. Líneas del Sistema de Transporte Público Trolebús

Línea	Ruta
	Corredor cero emisiones eje central
	Circuito politécnico
	Corredor cero emisiones Eje 7 - 7A sur
	Metro boulevard puerto aéreo - Metro El Rosario
	Metro El Rosario - Metro Chapultepec
	UACM - CU
	San Felipe de Jesús - Metro Hidalgo
	Corredor cero emisiones eje 2-2A sur

Fuente: (STE, 2014)

El Sistema de Transporte Eléctrico Trolebús, permite el intercambio modal con el Metro, colectivos y autobuses RTP, además existen conexiones entre las distintas rutas del trolebús, lo que permite al usuario viajar a lo largo de la red de dicho modo de transporte.

La flota vehicular del trolebús está compuesta por vehículos fabricados entre 1975 y 1998, siendo el mayor número de unidades las que se fabricaron entre 1997 y 1998, con 189 unidades (STE, 2012). La mayor parte de los vehículos en operación, tienen más de 14 años de antigüedad, es por ello que de manera frecuente se observan trolebuses descompuestos, y como consecuencia se vuelve necesaria la renovación de la flota vehicular.

1.3.1.5. Metrobús

El Metrobús es un sistema BRT (Bus Rapid Transit o Autobús de Transito Rápido) inspirado en el sistema de autobuses de la ciudad de Curitiba en Brasil, que empezó a operar en la Ciudad de México en el año 2005, en la actualidad cuenta con 4 líneas que cubren aproximadamente 95 kilómetros de recorrido, como se muestra en la Figura 1.5.

La Línea 1 cuenta con 30 kilómetros de recorrido, inició sus operaciones el 19 de junio de 2005 y fue ampliada el 13 de marzo de 2008. Las vialidades que cubre son: Insurgentes en sus tramos Norte, Centro y Sur. Las delegaciones que atiende el Metrobús son: Gustavo A. Madero, Cuauhtémoc, Benito Juárez, Álvaro Obregón, Coyoacán y Tlalpan. La demanda aproximada de usuarios es 390 mil por día. Brinda conectividad con las líneas del Metro: 1, 2, 3, 9 y B y con el tren suburbano.

La Línea 2 tiene 20 kilómetros de longitud e inició sus operaciones el 16 de diciembre de 2009. Las vialidades que cubre son: Eje 4 Sur y Prolongación Plutarco Elías Calles. Las delegaciones que atiende son: Miguel Hidalgo, Cuauhtémoc, Benito Juárez, Iztacalco e Iztapalapa. La demanda aproximada de usuarios es de 146 mil por día. Permite al usuario conectarse con las líneas del Metro: 1, 2, 3, 7, 8, 9 y A y con el corredor cero emisiones del trolebús.

La Línea 3 cuenta con 17 kilómetros de recorrido e inició operaciones el 8 de febrero de 2011. Cubre las siguientes vialidades: Eje 1 Poniente, Puente de Alvarado, Balderas y Av. Chapultepec - Dr. Río de la Loza. Las delegaciones que atiende el Metrobús son: Azcapotzalco, Gustavo A. Madero, Cuauhtémoc, Benito Juárez y al municipio de Tlalnepantla (Estado de México). La demanda aproximada de usuarios es de 104 mil por día. Brinda conectividad con las líneas del Metro: 1, 2, 3, 6, 9 y B y con el tren suburbano.

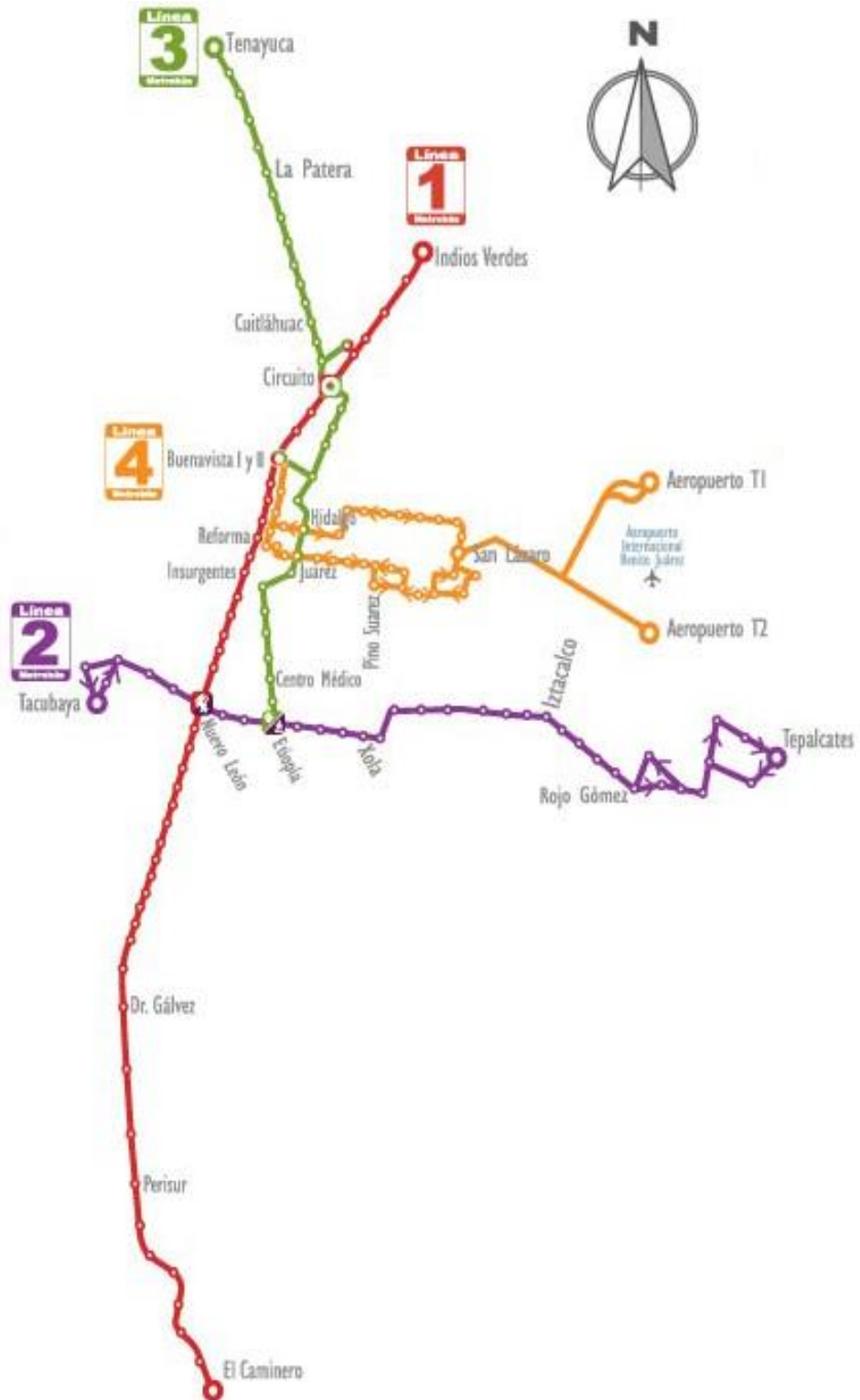


Figura 1.5. Red del Metrobús

Fuente: (Metrobús, 2012)

1.3.1.6. Taxi

El taxi es un transporte público de baja capacidad (o individual) que opera en toda la ZMVM, existen muchas irregularidades en su operación y en la expedición de elementos relacionados con dicho modo de transporte (placas, engomados, tarjetones, licencias, etc.). La ventaja es que el usuario puede abordar un taxi en cualquier punto de la ciudad y éste puede transferir modalmente al viajero hacia otro modo de transporte público o inclusive privado.

Según la clasificación de la Secretaría de Transportes y Vialidad del Distrito Federal (SETRAVI), los taxis se dividen en 5 distintas modalidades: libre, sitio, sitio radio taxi, sitio turístico y terminales. La tarifa vigente para el año 2014, se presenta en la Tabla 1.5.

Tabla 1.5. Modalidad y tarifa de los taxis para el año 2014

MODALIDAD	TARIFA AUTORIZADA DE BANDERAZO
Libre	\$ 8.74
Sitio	\$ 13.10
Sitio radio taxi	\$ 27.30
Sitio turístico	Definida por la organización
Terminales	De acuerdo a la zona

Fuente: (SETRAVI, 2014)

La tarifa por cada 250 metros o 45 segundos para el taxi libre es de \$1.07, para los taxis de sitio es de \$1.30 y para los radio taxis es de \$1.84. El servicio nocturno para los taxis, se considera a partir de las 23:00 hrs. y hasta las 6:00 hrs. del día siguiente, la tarifa autorizada será del 20% adicional a las señaladas en la Tabla 1.5, en función de la modalidad que se trate.

La Secretaría de Transportes y Vialidad (SETRAVI) a través de la Dirección General del Servicio de Transporte Público Individual de Pasajeros, no cuenta en su portal con la información de bases y número de unidades.

1.3.1.7. Colectivos (*microbuses y combis*)

Este tipo de transporte es el que cuenta con el mayor número de vehículos y rutas dentro del sistema de transporte en la Ciudad de México, son vehículos de baja capacidad, cuentan con poca organización en la cobertura de las rutas y en la asignación de vehículos. Por lo regular son vehículos inseguros, con una alta tasa de accidentes y de asaltos en unidades, al respecto, la ciudadanía percibe estos modos de transportes como los más inseguros. En general las unidades se encuentran en pésimas condiciones y los choferes no están correctamente capacitados. La Secretaría de Transportes y Vialidad (SETRAVI), no cuenta en su portal con la información de rutas y número de unidades que forman parte de este transporte público concesionado.

1.3.1. 8. Centros de Transferencia modal

Los distintos modos de transporte público anteriormente citados, tienen cierto grado de conexión entre ellos, sin embargo, en muchos de los casos, la transferencia modal se da sobre las vialidades, lo que ocasiona problemas de tránsito.

Existen algunos espacios físicos que forman parte de la infraestructura vial donde confluyen diversos modos de transporte terrestre de pasajeros (individual, colectivo y masivo) destinados a facilitar el transbordo de personas de un modo a otro.

También considerados como servicios auxiliares del transporte, los Centros de Transferencia Modal (CETRAM) se construyeron con el objetivo de dar solución a los problemas de congestión en vialidades aledañas a las estaciones del STC Metro o lugares donde se generan concentraciones considerables de vehículos de transporte periférico por ser bases o terminales de ruta (SETRAVI, 2012).

Existen en el Distrito Federal 45 CETRAM, los cuales ocupan una superficie de 791 mil 172 metros cuadrados. En éstos tienen su base 217 rutas y empresas de transporte concesionado, las cuales cubren alrededor de mil 217 destinos por la ciudad. Se calcula que el parque vehicular que entra diariamente a los CETRAM, es aproximadamente de 23

mil unidades, de las cuales el 45 por ciento proviene del Estado de México. El número de usuarios que utilizan los CETRAM en el área metropolitana es de 4.5 millones por día, aproximadamente. De éstos los de mayor afluencia son: Indios Verdes, Pantitlán, Tasqueña y Chapultepec, los cuales captan el 33 por ciento de la demanda. En 39 de los 45 CETRAM se tiene conexión con el STC-METRO y/o el tren ligero (SETRAVI, 2012).

Actualmente los CETRAM, se encuentran bajo resguardo y supervisión de la SETRAVI, entre 1999 y 2000, el Comité del Patrimonio Inmobiliario del Distrito Federal, acuerda por unanimidad que se autoriza a la SETRAVI, la posesión, vigilancia, conservación y administración de dichos espacios (SETRAVI, 2012). Sin embargo, por el gran número de personas que transitan en esos espacios, por el elevado número de vehículos que los ocupan, por el establecimiento del comercio ambulante y por una seguridad que se lleva a cabo de forma intermitente y deficiente, los CETRAM se han convertido en lugares inseguros, sucios y poco eficientes.

1.3.2. Problemas de transporte en la Ciudad de México

Si bien es cierto que el sistema de transporte de la Ciudad de México brinda ciertos beneficios sociales, y económicos, también es cierto que estos beneficios distan mucho de ser aceptables, esto se debe principalmente a la manera en la cual está organizado el sistema de transporte. Los viajes realizados se basan en una deficiente estructura modal, donde el mayor número de traslados se realiza en transporte de baja capacidad (microbuses, taxis y autos particulares), sin que las rutas se conecten con otros modos de transporte de forma adecuada y sin que se cuente con infraestructura auxiliar (estacionamientos) para que se favorezcan los viajes en automóvil combinado con cualquier otro modo de transporte público, es decir, el transporte de baja capacidad "tiene escasa integración con la gran infraestructura de transporte de alta capacidad –como Metro, RTP, STE, Metrobús- y utiliza de manera desventajosa las vialidades primarias, que también son saturadas por los autos privados" (UNAM PUEC, 2011) .

En la Figura 1.6, se muestran los porcentajes de los viajes cuya primera parte fue realizada en colectivo o en Metro, y la segunda parte del recorrido fue hecha en colectivo, Metro, autobús, taxi, Metrobús, Tren Ligero, Trolebús, etc. Como se puede observar la

mayor parte de los viajes se realizan en colectivos y en el metro, el resto de los porcentajes de viaje son muy bajos, lo que demuestra la poca integración modal que tiene nuestro sistema de transporte. El automóvil particular, se ubica en el rubro "otros", el cual en ambos casos es menor al 1%, muestra de la prácticamente nula integración de los viajes en automóvil con el transporte público. Lo anteriormente descrito se corrobora con los datos mostrados en la Figura 1.7, cuando los viajes se realizan tanto en transporte privado (automóvil, motocicleta y bicicleta) como público, solo se realizan 61,988 viajes de 21'954,157 viajes totales, lo que equivale al 0.3% del total de viajes (EOD, 2007).

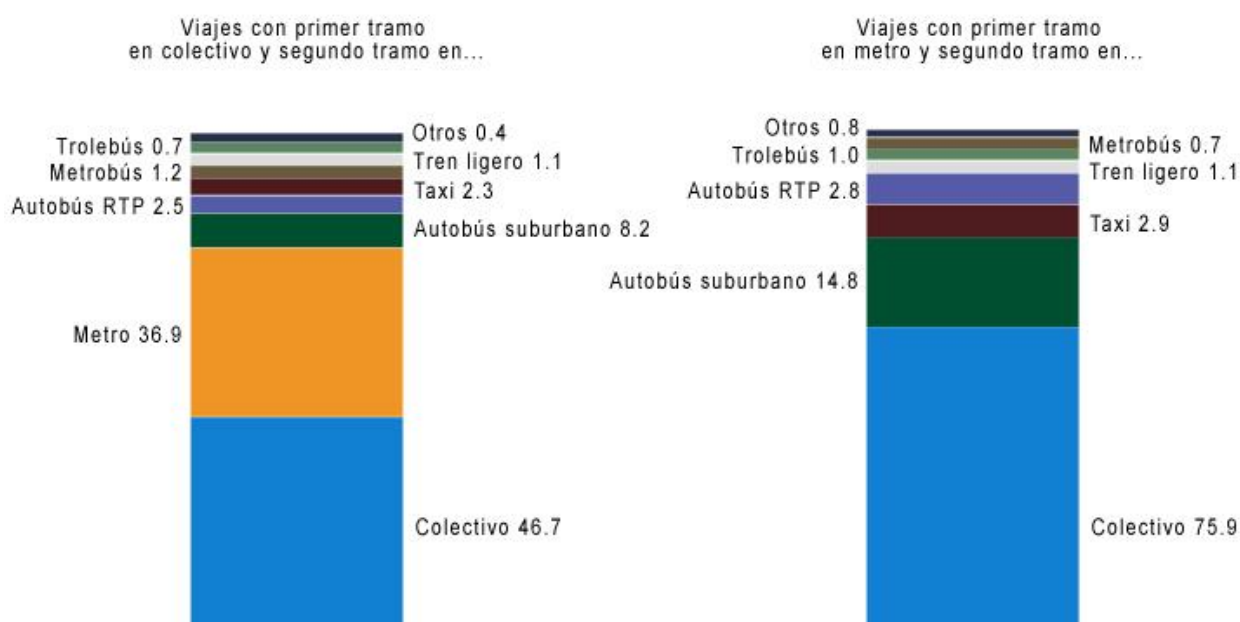


Figura 1.6. Reparto modal para viajes en dos tramos

Fuente: (Encuesta Origen Destino 2007)

Los principales problemas del transporte de baja capacidad en la Ciudad de México, se derivan de la inexistencia de paraderos establecidos, la falta de planeación para asignar los vehículos necesarios a lo largo del día en horas pico y valle, el congestionamiento que causan al subir/bajar a los pasajeros en carriles que no les corresponden y estacionándose en doble fila, provocando además que el cambio entre modos cuando se utiliza el transporte de baja capacidad no sea el adecuado.

VIAJES DE LOS RESIDENTES POR TIPO Y MODO DE TRANSPORTE CUADRO 12

Tipo y modo de transporte	Viajes	Porcentaje
Total	21 954 157	100.0
Transporte público	14 811 970	67.5
En un solo modo ¹	8 131 745	54.9
Metro ²	664 855	8.2
Tren ligero ²	3 787	0.0
Metrobús ²	42 633	0.5
Trolebús ²	71 224	0.9
Colectivo ²	5 243 743	64.5
Autobús RTP ²	185 525	2.3
Autobús suburbano ²	589 694	7.3
Taxi ²	1 330 284	16.4
En dos o más modos ¹	6 680 225	45.1
Transporte privado	6 806 735	31.0
En un solo modo ³	6 804 767	100.0
Automóvil ⁴	6 278 824	92.3
Bicicleta ⁴	433 981	6.4
Motocicleta ⁴	91 962	1.4
En dos o más modos ³	1 968	0.0
Transporte mixto⁶	61 988	0.3
Otros ⁷	273 464	1.2
En un solo modo ⁵	200 641	73.4
En dos o más modos ⁵	72 823	26.6

NOTA: La suma de los porcentajes puede no sumar 100% debido al redondeo.

¹ El porcentaje se obtiene respecto a los viajes en transporte público.

² El porcentaje se obtiene respecto a los viajes en transporte público en un solo modo.

³ El porcentaje se obtiene respecto a los viajes en transporte privado.

⁴ El porcentaje se obtiene respecto a los viajes en transporte privado en un solo modo.

⁵ El porcentaje se obtiene respecto a los viajes en otro tipo de transporte.

⁶ Transporte mixto incluye el tipo de transporte público y privado.

⁷ En Otro modo de transporte se incluyen autobuses foráneos, transporte escolar, transporte de personal, etc.

Figura 1.7. Porcentaje de viajes mixtos

Fuente: (Encuesta Origen Destino 2007)

1.3.3. Sostenibilidad del transporte en la Ciudad de México

Atender las necesidades en materia de transporte, desde una perspectiva de sostenibilidad exige considerar criterios urbanísticos, de infraestructura, sociales, económicos y ambientales. El objetivo es mejorar los sistemas en sí mismos, a través de una serie de instrumentos tecnológicos y de planeación que permitan optimizar su desempeño, pero también avanzar en la mejora de la calidad del aire, al tomar en cuenta que la mayoría de las emisiones contaminantes que llegan a la atmósfera provienen del sector transporte (SMA, 2012).

Actualmente, las zonas urbanas y su entorno son el motor económico del país, pues generan el 95% del PIB y albergan el 80% de la población. Sin embargo, su funcionamiento se ve amenazado por un modelo de transporte urbano que no es sustentable desde una perspectiva económica, financiera, social y ambiental (Lobo, 2012).

Por lo tanto, la solución que produzca mayores beneficios para atender la creciente demanda de movilidad en la ciudad de México (y en general en las grandes urbes) estará dada por el uso del transporte público masivo de calidad, que sustituya al transporte público de baja capacidad (microbuses, combis y taxis) y de baja calidad, y de igual forma reduzca el uso del automóvil.

Existen distintos factores que no permiten la sustentabilidad del sistema de transporte en la ciudad de México, entre los que destacan:

- Es la entidad federativa con la mayor densidad de población a nivel nacional.
- Se sigue autorizando la construcción de desarrollos urbanos, sin considerar cuales son las condiciones de transporte existentes, o el impacto que generará dicha construcción en la movilidad de la zona.
- Las malas condiciones de los vehículos: de las 29 mil 128 unidades de transporte que circulan en la Ciudad de México, 20 mil son obsoletas, lo que representa un riesgo para la seguridad de los pasajeros (Sol de México, 2012).
- Falta de regulación en el número de unidades que circulan: hay un exceso de flota cercano a las 10 mil unidades, lo cual provoca problemas serios como pelear el

pasaje, invasión de rutas, alta siniestralidad, unidades deterioradas e inseguras, entre otros (Sol de México, 2012). Esto se debe principalmente a que la forma en la que se produce el servicio de transporte, ya que no existen salarios ni prestaciones para los conductores de las unidades, son contratados por personas físicas y no por compañías dedicadas al transporte de pasajeros, motivo por el cual ellos deben obtener la mayor ganancia posible para garantizar el "salario" del día en curso.

- Altos costos por el uso excesivo del automóvil: En 2010 se gastaron 147 mil millones de pesos en gasolina importada al país y el subsidio que se asigna es de tal magnitud (76.6 mil millones de pesos en 2011 y se estima en 169.5 mil millones en 2012) que supera los montos erogados en los programas de combate a la pobreza. Política que es de carácter regresivo, pues el 70% de este subsidio se concentra en el tercio más rico de la población (ITPD, 2012).
- Falta de leyes en materia de sustentabilidad: El tamaño e importancia del desafío hace necesaria la participación directa y decidida de todos los niveles de gobierno pues el país no cuenta con una política pública robusta que dote a los gobiernos estatales y municipales con los recursos para asumir este reto. En este sentido, cambiar el modelo de transporte urbano es más un problema socio-político que técnico (Lobo, 2012).
- No se aplican las sanciones establecidas en el reglamento de tránsito: La corrupción conjunta de conductores y policías de tránsito, produce que las leyes y reglamentos vigentes, no se apliquen.
- Dificultad para cambiar de modo: En muchos casos los cambios de modo se realizan en la vía pública y fuera de espacios adecuados para el intercambio modal, como los CETRAM, bahías de ascenso/descenso, etc. Cuando se trata de viajes mixtos, no existen suficientes estacionamientos que permitan al conductor de un vehículo particular, dejar su coche cerca de una estación de transporte público, y continuar sin hacer uso del automóvil.
- Falta de información al usuario sobre rutas, alternativas, tiempos, ubicación de estacionamientos: Cuando los usuarios carecen de información o la información que poseen es insuficiente, las decisiones tomadas por estos son limitadas, esto impide que los usuarios puedan planear su viaje minimizando tiempos, costos, etc. Los Sistemas Avanzados de Información al Viajero (SAIV), brindan información a

los usuarios para que estos puedan realizar su viaje de la mejor manera (para mayor información consultar el tema 2.5, donde se detallan los SAIV).

Todo esto ocasiona que la movilidad urbana sea tan ineficiente que no sólo no contribuya al desarrollo económico del país, sino que afecta a la calidad de vida de sus habitantes, debido a los serios costos que genera y que no son cubiertos por quienes los generan (ITDP, 2012).

Como se explicó con anterioridad, existen distintos factores que impiden una adecuada movilidad en la Ciudad de México, como el uso indiscriminado del automóvil, el desorden del transporte público, la saturación de vialidades, y la falta de leyes en materia de sustentabilidad y transporte (y la no aplicación de las existentes); lo que ocasiona serios problemas ambientales y de salud.

Los problemas ambientales se acentúan debido a que la ciudad se ubica en una cuenca situada a 2 mil 240 metros sobre el nivel del mar, la ZMVM dispone de un 23% menos de oxígeno comparada con localidades costeras; esto afecta la eficiencia de los procesos de combustión y otros que requieren de este elemento para su óptima operación (SMA, 2012). Además, los vientos dominantes se direccionan de noreste a suroeste, encontrándose con la cadena montañosa que cierra la cuenca por el sur; de tal forma que las corrientes aéreas se detienen en esa zona, junto con los contaminantes que arrastran o se forman en el trayecto, los vientos son muy débiles la mayor parte del tiempo y, por lo tanto, se presenta una ventilación pobre. Ambas características impiden una circulación óptima de los vientos, generando en la atmósfera condiciones para una alta concentración de gases contaminantes (SMA, 2012).

Los habitantes de la ZMVM quedan expuestos a niveles de contaminación superiores a los recomendados por la Organización Mundial de la Salud (OMS) y por la Norma Oficial Mexicana de Salud (NOM-020-SSA1-1993), lo cual sin duda alguna afecta la salud de los pobladores de dicha región, en la Figura 1.8, se muestran los niveles de concentración de Ozono (ppb, partículas por billón) por año desde 1988 a la fecha, y aunque se muestra una tendencia decreciente en dichos niveles, estos aún exceden el límite establecido en la NOM-020-SSA1-1993. En la Figura 1.9, se muestra el número de días en los que se exceden los niveles máximos indicados en la misma norma.

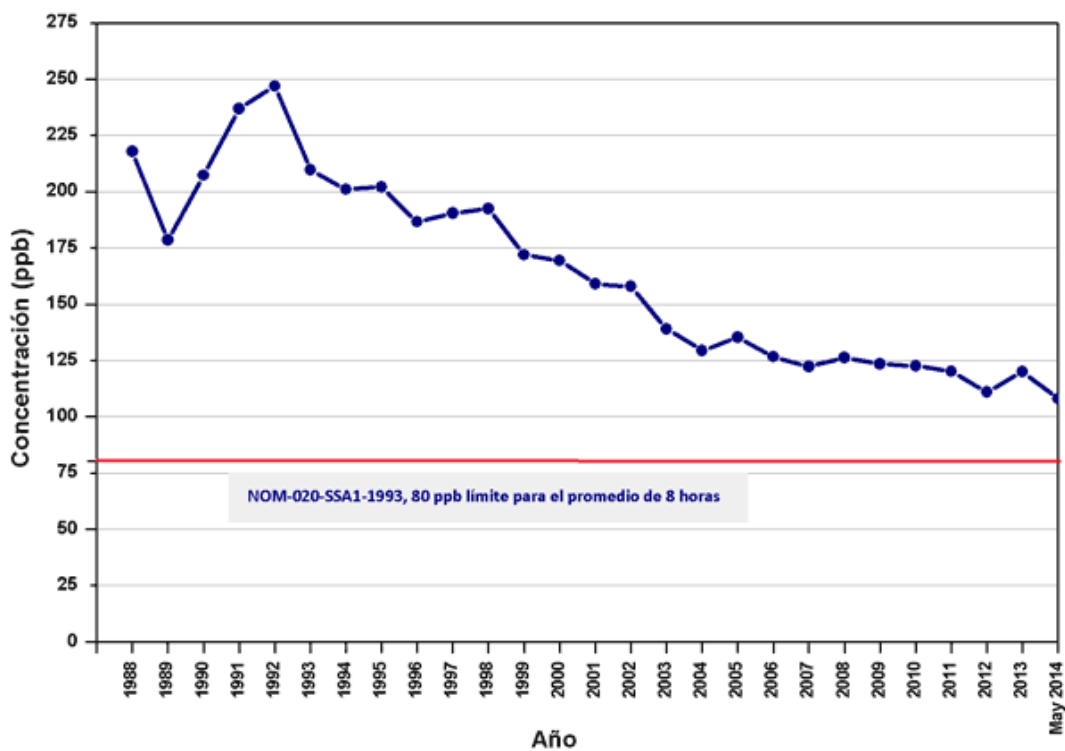


Figura 1.8. Indicadores del cumplimiento de la NOM-020-SSA1-1993

Fuente: (Sistema de Monitoreo Atmosférico, 2014)

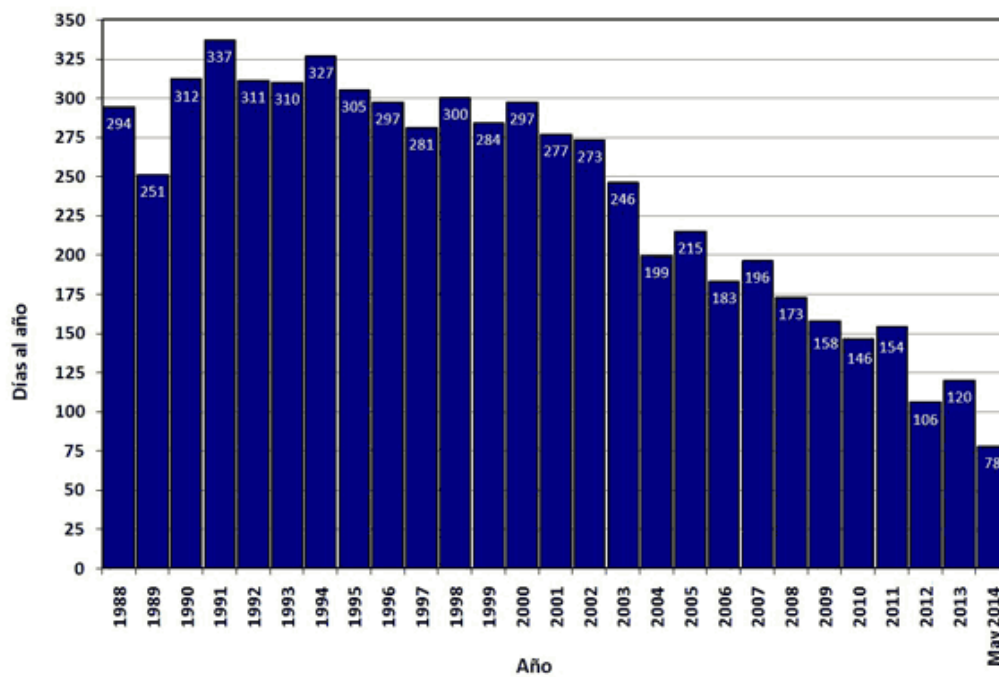


Figura 1.9. Frecuencia anual de días con concentraciones superiores a los indicados en la NOM-020-SSA1-1993

Fuente: (Sistema de Monitoreo Atmosférico, 2014)

La contaminación local que genera la combustión de gasolina, se estima que está ligada a casi 14 mil muertos en 2008 por mala calidad del aire, de acuerdo a la Organización Mundial de la Salud (OMS, 2012). En 2001, el doctor Mario Molina, premio nobel de química, recomendó al Gobierno de la Ciudad de México mejorar el transporte público como un requisito para atender el problema de la calidad del aire (SMA, 2012).

La participación activa del gobierno federal y de los gobiernos estatales, en la creación de estrategias, políticas e incentivos para impulsar un sistema de transporte sostenible, será decisiva. Las políticas implementadas deberán apoyar las tecnologías que resulten más costo-efectivas en la provisión de sistemas de transporte masivo, de acuerdo con el contexto específico de cada proyecto. Para ello, se requerirá realizar, en cada caso, un análisis de alternativas antes de establecer un esquema de financiamiento (Lobo, 2012).

En la actualidad el país carece de los elementos necesarios para afrontar los problemas y retos que plantea el transporte sustentable: la falta de una política nacional de movilidad urbana, de áreas específicas e instituciones para abordar el problema, los cortos periodos de gobierno de los municipios, las falta de coordinación de los tres niveles de gobierno, la falta de transparencia en el gasto público, el abandono de la planeación urbana y la falta de información sobre el uso del automóvil, entre otros (ITDP, 2012).

1.3.4. Retos del transporte público sostenible en la Ciudad de México

Distintos aspectos deben ser mejorados en el transporte público capitalino, entre estos se encuentran:

- La reducción de las emisiones contaminantes
- Disminuir el número de unidades de baja capacidad
- Reemplazar los vehículos en mal estado por unidades nuevas de mayor capacidad
- Mejorar la integración entre modos de transporte público
- Homologar el cobro entre los distintos modos de transporte público (tarjeta electrónica intermodal)
- Capacitación de conductores, aplicación de pruebas de manejo, y exámenes para detección de adicciones.

- Pagar salario y prestaciones a los conductores, para ofrecer mayor seguridad social y evitar la competencia para "ganar" el pasaje y "sacar la cuenta".
- Instalar sistemas de seguridad en las unidades (videocámaras y sistemas de comunicación con la SSP DF) para disminuir los asaltos.
- Uso de nuevas tecnologías, como los Sistemas Inteligentes de Transporte.

Lo anterior permitirá que la población use cada día más el transporte público, y esto a su vez, generará mejores ingresos a la gente que trabaja en los distintos modos de transporte y al mismo tiempo posibilite a la gente un traslado seguro, económico y con tiempos mejores (o en su defecto similares) a los hechos al viajar en automóvil.

Sin embargo, aunque se logre incentivar a la población para que use el transporte público, la gente seguirá usando su automóvil, por lo tanto será importante:

- Mejorar la integración del automóvil con el transporte público (estacionamientos cerca de terminales de Metro o autobuses)
- Regular el uso de espacios públicos a través de parquímetros
- Prohibir que los autos se estacionen en vías primarias, secundarias y alimentadoras.

Muchos son los retos que se tienen para mejorar el sistema de transporte en la Ciudad de México y como consecuencia lógica, mejorar la calidad de vida de sus habitantes. La magnitud de los problemas de movilidad, ocasiona que algunas soluciones deban ser implementadas a corto plazo, sin embargo debe existir una correcta planeación a mediano y largo plazo para producir soluciones efectivas.

1.4. Selección del área de estudio

El área de estudio se determinó con base en un par de factores, el primero, debido a que en la zona se combinan los distintos modos de transporte público mencionados en el punto 1.3.1 además de que la zona cuenta con vialidades de gran importancia, lo que permite incluir al automóvil dentro del análisis y de esta manera poder combinar el transporte público con el privado. En segunda instancia, la zona se eligió debido a que es una zona cercana a la UNAM y debido a esto, la información de campo es más fácil de obtener.

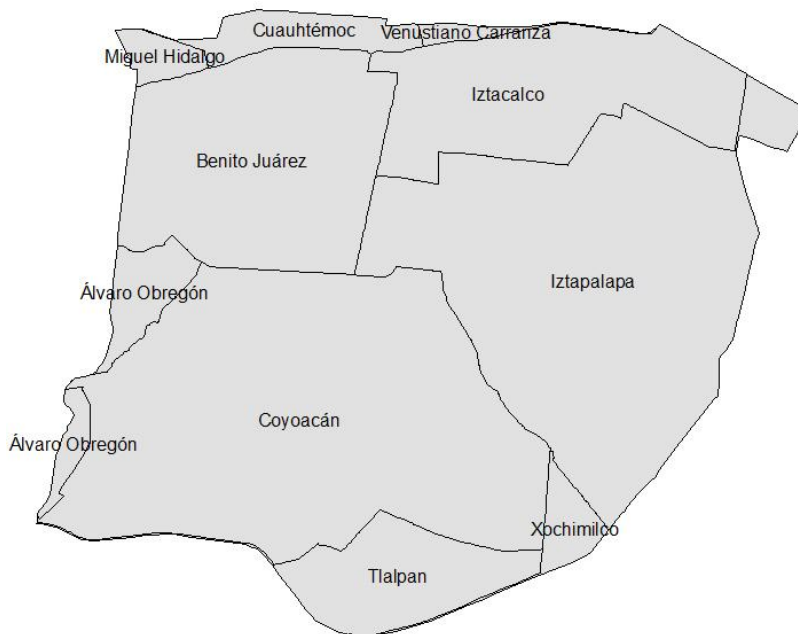


Figura 1.10. Delegaciones del área de estudio

Fuente: Propia

El área de estudio abarca la totalidad de la delegación Coyoacán, la mayor parte de las delegaciones Benito Juárez, Iztapalapa e Iztacalco y sólo una pequeña parte de las delegaciones Álvaro Obregón, Tlalpan, Xochimilco, Miguel Hidalgo, Cuauhtémoc y Venustiano Carranza, como se observa en la Figura 1.10. El área está delimitada por las siguientes vialidades:

- En el norte, por los Ejes 3 Sur y 4 Sur, por el Viaducto y la Calzada Ignacio Zaragoza.
- En el oeste, por la avenida Revolución y Paseos del Pedregal.
- En el sur, por el anillo Periférico en sus tramos, Boulevard Adolfo Ruiz Cortines y Canal de Garay.
- En el este, por el Anillo Periférico en sus tramos, Canal de Garay y Canal de San Juan.

1.4.1. Características socioeconómicas y demográficas

El análisis socioeconómico y demográfico se realizó únicamente en las delegaciones Benito Juárez, Iztacalco, Iztapalapa y Coyoacán, debido a que son las que ocupan la mayor parte del área de estudio. El Programa Delegacional de Desarrollo Urbano para las distintas delegaciones, cuenta con información a nivel de manzana, dimensión que contiene un mayor número de variables a las disponibles por el Área Geoestadística Básica (AGEB), permitiendo definir socio demográfica y económicamente a las Unidades Territoriales (UT) que conforman las delegaciones.

Según datos del Censo de Población y Vivienda 2010, la delegación Benito Juárez registró una población 385,439 habitantes en ese mismo año. En tanto que en el Programa Delegacional de Desarrollo Urbano (2005) define 56 UT's para la delegación Benito Juárez, de estas, 43 (80% de la población delegacional) presentan muy baja marginación, y sólo 13 UT's (20% de la población) presentan baja marginación.

Al efectuarse un análisis a nivel de manzana, se tiene que de las 2,082 manzanas que conforman el territorio delegacional, 1,464 manzanas habitadas por el 75% de la población delegacional presentan muy baja marginación, 468 manzanas que contienen el 21% de la población delegacional presentan baja marginación, y 95 con el 4% de la población presentan media, alta y muy alta marginación; mientras que el resto de las manzanas (55) no reportan información.

Para la delegación Coyoacán, el Censo de Población y Vivienda 2010, registró una población de 620,416 habitantes en ese mismo año. El Programa Delegacional de Desarrollo Urbano para la delegación Coyoacán (2010), define 116 Unidades Territoriales (UT) que conforman la delegación. Del total de UT's, 72 presentan muy baja marginación, representando el 38.9% de la población de la delegación, 21 UT's con baja marginación y el 20.3% de la población; en contraste, se registran 13 UT's con alta marginación habitadas por el 20.8% de la población delegacional, 7 UT's con grado de marginación media y 7.2% de la población y 3 UT's con muy alta marginación con el 12.8% de la misma.

Al bajar a nivel de las 4 mil 305 manzanas que conforman el territorio delegacional, se tienen 2 mil 528 manzanas de muy baja marginación habitadas por el 42.6% de la población delegacional, 649 manzanas de baja marginación habitadas por el 13% de la población delegacional, 625 manzanas con 20.6% de la población con marginación media, además de las 428 manzanas que representan alta marginación con 19.9% de la población de la delegación y las 75 manzanas con muy alta marginación habitadas por el 3.7% de la población de la delegación.

En el Censo de Población y Vivienda 2010, la delegación Iztapalapa registró una población 1'815,786 habitantes. El Programa Delegacional de Desarrollo Urbano en Iztapalapa (2003), dividió socio demográfica y económicamente a la delegación en 186 Unidades. Del total de UT's, 65 presentan una marginación muy alta, 46 una marginación media y 45 se presentan con alta marginación; en contraste, se encuentran 15 UT's con 107,638 habitantes, las cuales presentan grados de marginación baja y las 15 restantes con 71,471 habitantes, presentan una marginación muy baja.

Al bajar el nivel de las 13,641 manzanas que conforman el territorio delegacional se tiene que 9,632 manzanas (71%) habitadas por el 81.4% de la población delegacional, presentan una marginación que oscila entre muy alta y media marginación; mientras que el 26% de las manzanas (3,538), habitadas por el 17.3% de la población presentan baja y muy baja marginación; de 471 manzanas no se tiene datos.

Según datos del Censo de Población y Vivienda 2010, la delegación Iztacalco registró una población 384,326 habitantes. En relación a los niveles de ingreso de la población ocupada en 1990 se tuvieron para esta delegación las siguientes cifras. Entre 0 y 2 salarios mínimos, 62.4%; más de 2 y hasta 5 salarios mínimos, 27.7%; más de 5 salarios mínimos, 7.2% y, no especificados 2.7%.

El grado de marginación de las UT's de las delegaciones involucradas en el análisis, se muestran en la Figura 1.11.

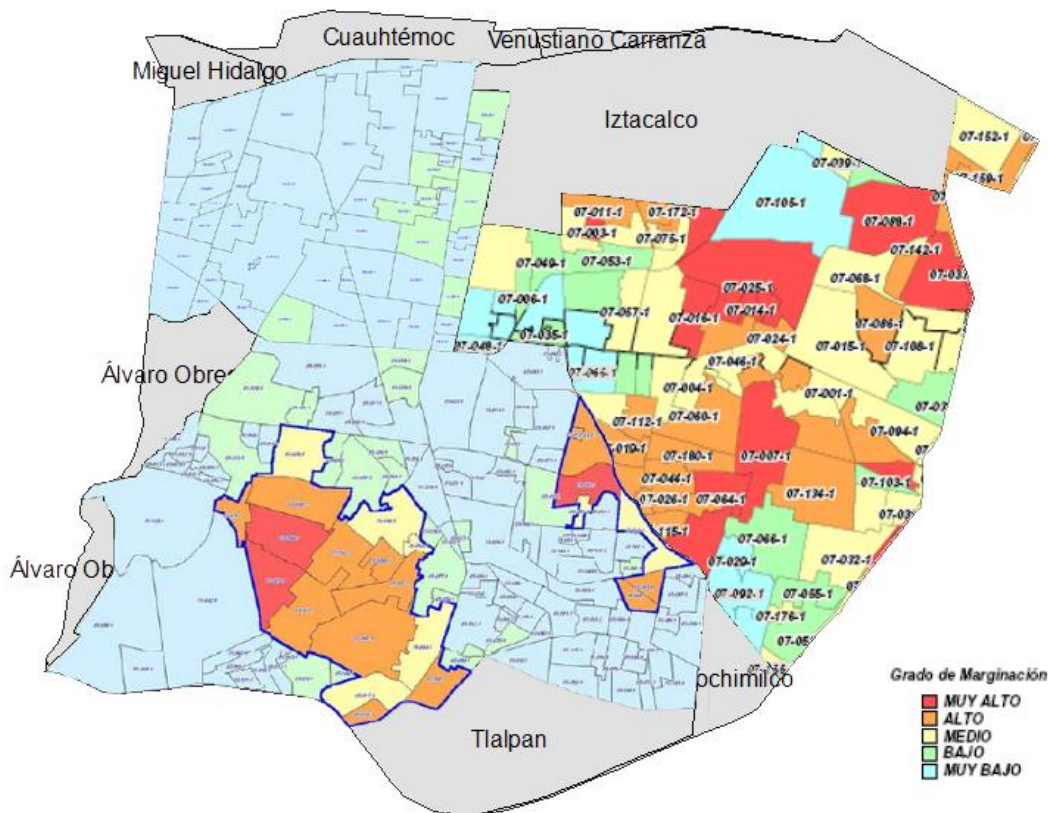


Figura 1.11. Grado de marginación en el área de estudio

Fuente: Propia, con información de los Programas Delegacionales de Desarrollo Urbano

1.4.2. Modos de transporte público existentes en la zona de estudio

A continuación se explican los distintos modos de transporte existentes en las delegaciones Benito Juárez, Coyoacán, Iztapalapa e Iztacalco.

El transporte público que existe en la *delegación Benito Juárez*, se encuentra integrado por: el Sistema de Transporte Colectivo Metro, la Red de Transporte de Pasajeros (RTP) y el Sistema de Transporte Eléctrico (Trolebús), los cuales se complementan con las rutas de servicio privado concesionado de taxis y colectivos (Micros). La delegación cuenta con 3 líneas del Metro y un total de 14 estaciones, como se desglosa a continuación:

- La Línea 2 (Cuatro Caminos - Tasqueña) con las estaciones Ermita, Portales, Nativitas, Villa de Cortés, Xola, Viaducto.

- La Línea 3 (Indios Verdes-Universidad) con las estaciones Coyoacán, Zapata, División del Norte, Eugenia y Etiopía.
- La Línea 7 (El Rosario-Barranca del Muerto) con las estaciones Mixcoac, San Antonio y San Pedro de los Pinos.

En lo referente a las rutas de la Red de Transporte de Pasajeros (RTP), existen alrededor de 20 rutas que sirven a la delegación Benito Juárez, dentro de éstas existen puntos de intercambio modal con 11 estaciones de 3 Líneas del Metro (Líneas 2, 3 y 7). De igual forma, existen 6 Líneas del Sistema de Transporte Eléctrico en su modalidad de Trolebús y puntos de intercambio modal con 10 estaciones de 3 líneas del Metro (Líneas 2, 3 y 7). Existen dos Centros de Transferencia Modal (CETRAM) en la delegación (metro Zapata y Mixcoac); además, debido a la dinámica vehicular que se presenta en las cercanías de las estaciones del Metro Ermita y Villa de Cortés, éstas se consideran como centros de transferencia, ya que son el destino de algunas rutas de transporte público como: microbuses, Red de Transporte de Pasajeros y Líneas de Transporte Eléctrico (Programa Delegacional de Desarrollo Urbano Benito Juárez, 2005).

De acuerdo con el Programa Delegacional de Desarrollo Urbano de la *Delegación Coyoacán* (2010), la delegación cuenta con los siguientes modos de transporte:

- El Sistema de Transporte Colectivo Metro, atraviesa la Delegación con dos líneas, la Línea 2 (Cuatro Caminos - Tasqueña), con 2 estaciones (General Anaya y Tasqueña) y la Línea 3 (Indios Verdes-Ciudad Universitaria) con 4 estaciones (Universidad, Copilco, Miguel A. de Quevedo y Viveros).
- El Sistema de Transportes Eléctricos en Coyoacán cuenta con cuatro líneas de trolebús, que recorren la delegación por Av. Miguel Ángel de Quevedo – Calzada Tasqueña de oriente a poniente, por Calzada Tasqueña - División del Norte, y en el sentido norte-sur sobre el Eje 3 Oriente Cafetales, con ramales hacia la Unidad Habitacional CTM hasta el Anillo Periférico.
- También, se cuenta con el Tren Ligero desde el CETRAM Tasqueña hasta el Estadio Azteca, con un total de diez estaciones.
- La red de autobuses concesionados cubre la delegación por sus vías principales y la red de transporte particular concesionado (microbuses y combis), sobre las vías

primarias y secundarias de la delegación adentrándose en algunos casos al tejido local.

De acuerdo con el Programa Delegacional de Desarrollo Urbano de la *Delegación Iztapalapa* (2008), la delegación cuenta con los siguientes modos de transporte:

- El Sistema de Transporte Colectivo Metro cubre las zonas centro y norte de la delegación mediante las Líneas 8 Garibaldi-Constitución de 1917 y “A” Metro Férreo Pantitlán-La Paz, respectivamente. La primera, que comunica el Centro de la Ciudad con el Centro de Iztapalapa corre por el Eje 3 Oriente hasta tomar la Calzada Ermita Iztapalapa. Las estaciones que se localizan dentro de la demarcación son: Apatlaco, Aculco, Escuadrón 201, Atlalilco, Cerro de la Estrella, Purísima y la terminal actual Constitución de 1917. La segunda sobre la Calzada Zaragoza incluye dentro de la demarcación las estaciones Canal de San Juan, Tepalcates, Guelatao, Peñón Viejo, Acatitla y Santa Martha.

El transporte público lo constituye principalmente el servicio concesionado a empresas transportistas que fundamentalmente articulan sus derroteros en función del intercambio modal con el STC Metro. Dicho servicio, es regulado a través de la Secretaría de Transportes y Vialidad, el parque vehicular de estas empresas y sus rutas, se encuentran en proceso de sustitución de unidades de baja capacidad, que saturan las vialidades con paradas continuas por autobuses de pasajeros mejor equipados, por alta capacidad, con paradas fijas.

- Cuenta con 11 CETRAM, 8 de ellos corresponden a las estaciones Apatlaco, Escuadrón 201, Iztapalapa, Constitución de 1917, Canal de San Juan, Tepalcates, Acatitla y Santa Martha, los tres restantes se localizan al oriente y poniente de la Central de Abasto y en Canal de Chalco. En conjunto la afluencia diaria a estos nodos de transferencia se estima en 500,000 usuarios (Programa Delegacional de Desarrollo Urbano de la delegación Iztapalapa, 2008).

De acuerdo con el Programa Delegacional de Desarrollo Urbano de la *Delegación Iztacalco* (1997), ésta cuenta con los siguientes modos de transporte:

- Diez estaciones del Metro pertenecientes a 5 líneas diferentes: Línea A (Pantitlán - Observatorio) con las estaciones San Juan y Agrícola Oriental, Línea 2 (Cuatro Caminos - Tasqueña) con la estación Viaducto, Línea 4 (Martín Carrera - Santa Anita) con la estación Santa Anita, Línea 8 (Garibaldi - Constitución de 1917) con las estaciones Apatlaco, Iztacalco Coyuya y Santa Anita y Línea 9 (Tacubaya - Pantitlán) con las estaciones Ciudad Deportiva y Puebla.
- Con relación al Sistema de Transporte Eléctrico existen 4 rutas con una longitud sobre la Delegación estimada en 15.31 kilómetros. Ruta M (Metro Iztacalco - Metro Villa de Cortés) con una longitud de 4.25 km, Ruta Ñ (Eje 3 y Eje 4 Sur Metro Tacubaya - ISSSTE Zaragoza) con una longitud de 5.20 km, Ruta O (Eje 5 y 6 Sur Central de Abastos - Metro San Antonio) con una longitud de 2.46 km y Ruta Q (Eje 5 Oriente Metro Pantitlán - Metro Iztapalapa) con una longitud de 3.4 km.
- Operan 32 rutas del servicio de transporte público de pasajeros en su modalidad de colectivo con paradas fijas.
- El RTP tiene en operación 15 rutas que cruzan la Delegación Iztacalco.
- Existen 3 Centros de Transferencia Modal: el CETRAM Coyuya que se conecta con la Línea 8 del STC-Metro, el CETRAM Pantitlán con capacidad para 260 autobuses y 440 microbuses y que conecta con las Líneas 1, 5, 9 y A del STC Metro (estas instalaciones son compartidas con la Delegación Venustiano Carranza), y el CETRAM Santa Anita con una capacidad de 42 autobuses y 64 microbuses y que conecta con las Líneas 4 y 8 del STC Metro (Programa Delegacional de Desarrollo Urbano de la Delegación Iztacalco, 1997).

1.4.3. Conexión entre el transporte público con el privado en la zona de estudio

Desafortunadamente, la integración modal de los viajes realizados por los capitalinos, es deficiente entre modos de transporte público, y prácticamente inexistente entre el transporte público y el privado. Existen distintos problemas que se presentan para ligar el transporte público con el privado en la zona de estudio, esto se debe principalmente a que

el número de estacionamientos públicos es muy bajo, y la mayor parte de éstos se encuentran lejos de CETRAMS, estaciones del Metro y en general, de los puntos de acceso a transporte público de alta capacidad.

El Gobierno del Distrito Federal, a través de las delegaciones debe planear la construcción de estacionamientos cercanos a estaciones de transporte público de alta demanda y CETRAMS, apegándose a lo establecido en los Programas Delegacionales de Desarrollo Urbano donde se indica que se pueden ampliar o construir nuevos estacionamientos públicos y privados en cualquier zonificación, excepto en Áreas Verdes (AV), Espacios Abiertos (EA) y en las correspondientes a Suelos de Conservación.

Estos estacionamientos tendrán el número de niveles que rija en la zona, considerando para ello la aplicación de otras Normas de Ordenación Generales, pudiendo ocupar el 85% de la superficie del terreno. La Delegación podrá regular el cobro al usuario considerando el tipo de uso al que el estacionamiento esté dando servicio (Programa Delegacional de Desarrollo Urbano Iztapalapa, 2008).

Para ejemplificar la necesidad de estacionamientos, se menciona lo que ocurre en las Delegaciones Benito Juárez y Coyoacán.

En la Delegación Benito Juárez, hay pocas áreas de estacionamientos públicos, únicamente se encuentran algunos estacionamientos privados integrados a diferentes inmuebles como: almacenes, centros comerciales y encierro de vehículos particulares; el equipamiento urbano existente y las tiendas departamentales no cuentan con los espacios suficientes para estacionamientos y sólo en determinadas instalaciones como hospitales, clínicas, deportivos y áreas administrativas delegacionales se proporciona el servicio al personal directivo. Además, se presentan diversos problemas viales alrededor de los espacios públicos que carecen de estacionamientos para vehículos y camiones que transportan mercancías, así como en las avenidas de mayor circulación debido a que su sección es reducida y no posibilita el estacionamiento en ambos lados (Programa Delegacional de Desarrollo Urbano Benito Juárez, 2005).

De acuerdo con el Programa Delegacional de Desarrollo Urbano de la *Delegación Coyoacán* (2010), las principales zonas que presentan problemas de estacionamiento son:

- a) Zonas aledañas al Estadio Azteca en los días de servicio, llega a afectar varias zonas habitacionales en un gran número de calles a su alrededor, principalmente Bosques de Tetlameya.
- b) La Av. División del Norte en la zona cercana a Av. Río Churubusco, debido al intenso uso comercial en la zona.
- c) La zona de poblados (Los Reyes, La Candelaria, San Francisco Culhuacán y San Pablo Tepetlapa) por la irregularidad y la reducida sección de sus calles no previstas para estacionamiento, además su traza irregular dificulta la posibilidad de ampliar estas calles.
- d) El Centro de Histórico de Coyoacán y la Colonia del Carmen son las zonas con mayores problemas de estacionamiento, especialmente en fines de semana y días festivos, cuando la demanda estimada de cajones asciende a más de 2,100 lugares. En el polígono existen cinco estacionamientos públicos y conforme la normatividad de los Programas Parciales Centro Histórico y Del Carmen, no está permitida la construcción de nuevos estacionamientos.
- e) Equipamientos con atención a la ciudadanía, como escuelas de nivel medio y superior, mercados, centros de salud, deportivos, etc.

Como se muestra en la Figura 1.7, el porcentaje de viajes mixtos es de 0.3% del total de viajes realizados en la ZMVM, para incrementar este porcentaje y generar beneficios como la disminución del tránsito vehicular y de la contaminación ambiental, se debe mejorar el transporte público, tanto en infraestructura, vehículos y capacitación del personal, además se deben construir, ampliar o habilitar estacionamientos cercanos a las principales estaciones de transporte público, lo que permitirá un incremento de este tipo de viajes.

1.4.4. Implementación de un SIT en la zona de estudio

El uso de nuevas tecnologías debe ser uno de los factores más importantes para plantear nuevas alternativas de solución a los problemas del transporte en las grandes urbes. Una de estas nuevas tecnologías (en nuestro país) son los Sistemas Inteligentes de Transporte, SIT, (Intelligent Transport Systems, ITS), los cuales utilizan la tecnología computacional y de comunicaciones para mejorar las condiciones del transporte.

Como se ha mencionado, la conexión entre el transporte público y privado es prácticamente inexistente, sin embargo existen algunos estacionamientos en el área de estudio que permiten al usuario transferirse entre los transportes privado y público.

En el caso de esta tesis, se propone la implementación de un sistema SIT como un Sistema Avanzado de Información al Viajero, SAIV (Advanced Traveler Information Systems, ATIS), que le permita al usuario seleccionar su ruta en el área de estudio, basándose en ciertos criterios como la distancia, tiempo, costo, número de transbordos, etc.

Al hacer uso del sistema resultado de esta tesis, el usuario podrá seleccionar desde una página web, su punto de origen y su destino final (siempre y cuando ambos estén dentro de la zona de estudio), de igual forma, podrá seleccionar que sea mínimo el número de transbordos, el costo, el tiempo, o la distancia, además seleccionará entre un viaje hecho en transporte público o uno hecho de forma mixta.

Si bien es cierto que a través de los Sistemas Inteligentes de Transporte no se mejorará la estructura modal del transporte de la Ciudad de México, sí es posible planear de manera más eficiente la forma en la que las personas se transportan, y además, hará evidente la necesidad de estacionamientos cercanos a las estaciones de transporte público, que permitirán la conexión entre el transporte privado con la red de transporte público capitalino.

Por ejemplo en el CETRAM Universidad, existe un gran estacionamiento el cual es usado como depósito (corralón) de taxis en la actualidad. Ese espacio podría ser utilizado para

brindar el servicio de estacionamiento a las personas que se desplazan desde la estación Universidad de la Línea 3 del STC Metro.

En el siguiente capítulo se analizan los Sistemas Inteligentes de Transporte (SIT), su clasificación, ejemplos, y la definición y características del Sistema Avanzado de Información al Viajero (SAIV).

Capítulo 2 . Sistemas Inteligentes de Transporte

Este capítulo se subdivide en cinco secciones, en la primera se presentan distintas definiciones de los Sistemas Inteligentes de Transporte (SIT), en la segunda sección se muestra una clasificación de los SIT, en la tercera sección se enlistan algunos de los organismos SIT más importantes alrededor del mundo, en la cuarta sección se presenta un resumen de los congresos internacionales sobre SIT, y finalmente en la quinta sección, se profundiza sobre los Sistemas Avanzados de Información al Viajero (SAIV) y sus distintos subsistemas, ya que el sistema programado para este trabajo de tesis se basa en un sistema de este tipo.

Debido a la complejidad de los sistemas de transporte, se deben implementar mecanismos, para controlar, regular y ordenar a los elementos de transporte ya existentes. Es indispensable hacer uso de tecnologías como, la computacional, electrónica, de la información, etc. para implementar soluciones que ayuden a aminorar los problemas relacionados con el transporte, como las que brindan los Sistemas Inteligentes de Transporte (SIT).

Como menciona Taylor (2001), el objetivo de usar los SIT es el mejorar las uniones entre la infraestructura, los vehículos, y los usuarios, para hacer que los componentes individuales, modos y el sistema de transporte trabajen como un todo de manera más efectiva y eficiente, en el contexto más amplio de transporte, desarrollo urbano y regional, políticas económicas y ambientales. Se sugieren los siguientes beneficios potenciales de las aplicaciones de los SIT:

- mejor utilización de la infraestructura;
- mejor flujo del tráfico;
- mejor servicio del transporte público;
- mejor seguridad;
- menores costos para el transporte de carga; y
- reducción del impacto ambiental.

2.1. Definición de SIT

Acha y Espinoza (2004) del Instituto Mexicano del Transporte mencionan que los Sistemas Inteligentes de Transporte, son la aplicación de tecnología avanzada de captura y proceso de información, comunicaciones y control para mejorar la eficiencia y seguridad en un sistema de transporte.

Por su parte el Departamento de Transporte de los E.U.A. estableció en 1999 la siguiente definición de SIT (Nelson, 2000): "Los sistemas inteligentes de transporte recolectan, almacenan, procesan y distribuyen información relacionada al movimiento de personas y bienes. Ejemplos incluyen los sistemas para la gestión de tránsito, la gestión del transporte público, el manejo de emergencias, la información a los usuarios, la seguridad y el control avanzado de los vehículos, las operaciones de vehículos comerciales, el pago electrónico y el cruce seguro a nivel de las líneas de ferrocarril".

El organismo ITS Canadá define a los Sistemas Inteligentes de Transporte de la siguiente manera (ITS Canadá, 2012): SIT es la aplicación de avanzadas y nuevas tecnologías (ordenadores, sensores, control, comunicaciones y dispositivos electrónicos) en el transporte, para salvar vidas, tiempo, dinero, energía y ayudar al medio ambiente.

La Secretaría de Estado de Transportes del Gobierno de España (2010), menciona que los SIT se pueden definir como un conjunto de aplicaciones avanzadas dentro de la tecnología informática, electrónica y de comunicaciones que, desde un punto de vista social, económico y medioambiental, están destinadas a mejorar la movilidad, seguridad y productividad del transporte, optimizando la utilización de las infraestructuras existentes, aumentando la eficiencia del consumo de energía y mejorando la capacidad del sistema de transportes.

ITS Europa (ERTICO, 2012) define a los SIT como la unión de la información y las tecnologías de la comunicación entre los vehículos y las redes carreteras para transportar gente y mercancías. Los SIT son "inteligentes" porque brindan conocimiento extra a los viajeros y a los operadores. En los vehículos, los sistemas SIT ayudan a los conductores a evitar congestionamientos y colisiones. En los trenes y autobuses, permiten a los directivos optimizar las operaciones de las flotas y ofrecen a los pasajeros cobros

automáticos e información en tiempo real sobre las corridas. En la red de carreteras, los SIT coordinan las señales de tránsito, detectan y manejan incidentes y despliegan información, orientación e instrucciones para los conductores.

De las anteriores definiciones se observa que el factor tecnológico es quizá el de mayor importancia, y lamentablemente en México creamos poca tecnología. La mayor parte de la tecnología es comprada a los países industrializados, lo que produce un incremento en los costos y una dependencia tecnológica hacia dichos países. Además, la gente responsable de la planeación y de las adquisiciones tecnológicas relacionadas con el área del transporte, es más burócrata que investigadora, donde muchas veces es influenciada por decisiones políticas y otras tantas desconoce el ámbito tecnológico.

La definición más completa es la que plantea la Secretaría de Estado de Transportes del Gobierno de España, ya que indica en primera instancia en qué consisten los SIT (conjunto de aplicaciones avanzadas dentro de la tecnología informática, electrónica y de comunicaciones); posteriormente plantea que los objetivos de un SIT deben estar enfocados a mejorar la interacción del transporte con las esferas social, económica y medioambiental, planteando objetivos concisos (mejorar la movilidad, seguridad y productividad del transporte), es decir, considera la necesidad de que el transporte sea sustentable y se minimicen los impactos negativos de este, en las tres esferas anteriormente mencionadas. Además, indica cómo cumplir con los objetivos planteados (optimizando la utilización de las infraestructuras existentes, aumentando la eficiencia del consumo de energía y mejorando la capacidad del sistema de transportes).

Lo que le falta a la definición, es una referencia a la interacción que debe existir entre vehículos y de los vehículos con la infraestructura.

Para conocer más sobre los SIT es necesario saber cómo se clasifican, de esta forma es posible entender cuáles son sus beneficios y aplicaciones posibles.

2.2. Clasificación de los SIT

Los Sistemas Inteligentes de Transporte (SIT) se clasifican en Sistema Avanzado de Gestión del Transporte (SAGT ó ATMS Advanced Transportation Management Systems), Sistema Avanzado de Información al Viajero (SAIV ó ATIS, Advanced Traveler Information Systems), Sistema Avanzado de Control de Vehículos (SACV ó AVCS, Advanced Vehicle Control Systems), Operación de Vehículos Comerciales (OVC ó CVO, Commercial Vehicle Operations), Sistema Avanzado para el Transporte Público (SATP ó APTS, Advanced Public Transportation Systems) y Sistema Avanzado para el Transporte Rural (SATR ó ARTS, Advanced Rural Transportation Systems) (Sussman, 2000).

En la Tabla 2.1, se enuncian las principales características de cada uno de ellos.

Tabla 2.1. Clasificación de los subsistemas SIT

	Subsistema SIT	Características
SAGT	Sistema Avanzado de Gestión del Transporte	Gestión de la red, incluyendo incidentes, control de tráfico, pago electrónico de peaje, predicción de congestamientos y estrategias para aminorarlos.
SAIV	Sistema Avanzado de Información al Viajero	Información a disposición del viajero antes de iniciar el viaje o en el vehículo durante el mismo. Los SAIV ayudan a proveer información de la red en tiempo real.
SACV	Sistema Avanzado de Control de Vehículos	Permiten mejorar el control del conductor sobre el vehículo y la seguridad del mismo mediante un grupo de tecnologías.
OVC	Operación de Vehículos Comerciales	Tecnologías para mejorar la productividad de las flotas comerciales, incluyendo el pesaje en movimiento (WIM), procedimientos para anticipar el despacho aduanero, bitácoras electrónicas y coordinación interestatal.
SATP	Sistema Avanzado para el Transporte Público	Información al pasajero y tecnologías para mejorar las operaciones del sistema, incluyendo la recaudación de las tarifas, transferencias intramodales e intermodales y programación de los vehículos de transporte público.
SATR	Sistema Avanzado de Transporte Rural	Principalmente seguridad y tecnologías de ayuda para viajeros en áreas rurales.

Fuente: (Sussman, 2000)

Para profundizar más sobre la clasificación descrita en la Tabla 2.1, en la siguiente subsección, se retoma de Mercado (2006) la descripción detallada de cada una de las características y funciones de los subsistemas que forman parte de los Sistemas Inteligentes de Transporte (SIT).

2.2.1. Sistema Avanzado de Gestión del Transporte (SAGT)

Mercado (2006), define las siguientes características de los Sistemas Avanzados de Gestión del Transporte (SAGT). La función de los SAGT, es la de recabar toda la información relacionada con las vías de comunicación, para esto utiliza una gran variedad de sensores y videocámaras. Después utiliza dicha información en diferentes cálculos y procesos, y finalmente la envía a los conductores o a los distintos módulos del SIT. Para transmitir la información a los conductores se puede recurrir a los sistemas de Señales de Mensaje Variable (SMV o VMS, Variable Message Signs), o por medio de señales directas a los sistemas de abordaje dentro de los vehículos como los sistemas de Asesoría por Radio (HAR, Highway Advisory Radio). La finalidad de los SAGT es la de mejorar el flujo dentro de una vía terrestre y asegurarse de que dicha vía se utiliza en las mejores condiciones de tiempo y espacio.

Como ejemplo de un sistema SAGT, se puede mencionar lo que se está realizando en Chile. El Ministerio de Transportes y Telecomunicaciones del Gobierno de Chile (UOCT, 2010), menciona que ese año estaban operando como parte de un SAGT, Sistemas de Control de Tránsito en las ciudades de Santiago, Antofagasta, Gran Valparaíso, Gran Concepción y Rancagua. También menciona que el sistema de control de tránsito de Santiago que administra la Unidad Operativa de Control de Tránsito (UOCT) controla actualmente 2,276 semáforos, de los cuales 1,941 operan con planes de tiempo prefijados según la hora del día; 320 lo hacen en control dinámico SCOOT; y, finalmente, 15 dispositivos localizados en la periferia de la ciudad, cuentan con sensores en todos sus accesos y otorgan los derechos de paso según la demanda puntual.

Según el Ministerio de Transporte y Telecomunicaciones de Santiago de Chile (UOCT, 2010), la UOCT para la gestión del tránsito cuenta con distintos sistemas como:

- **Circuito cerrado de televisión.** Permite monitorear las condiciones de operación de los cruces más conflictivos de la ciudad. Su objetivo es poder detectar oportunamente incidentes o situaciones de congestión, en cuyo caso se implementan desde el centro de control modificaciones en las programaciones o planes especiales para mitigar tales problemas.
- **Sistema de letreros y mensajes variables.** Orientado a informar a los usuarios sobre problemas de congestión en ciertas áreas, tiempos promedio de viaje entre determinados puntos, sobre la ocurrencia de accidentes u otro tipo de incidentes, así como recomendar rutas alternativas de desplazamiento.
- **Sistema de información geográfico.** Contiene un plano básico digitalizado de Santiago, escala 1:10.000, y una completa base de datos asociada a cada intersección semaforizada, con información de flujos vehiculares clasificados según tipo y horario, monografías, diseño de fases y planes de tiempos del semáforo según hora del día, estadísticas de falla del semáforo, señalización existente, etc.
- **Sistema de prioridad para los vehículos de emergencia.** En caso de un siniestro o emergencia, este sistema permite generar una onda de luces verdes por algunas de las rutas predefinidas. Este sistema ya se encuentra en aplicación en dos Compañías de Bomberos y en el corto plazo se extenderá a otros cuarteles.
- **Red de estaciones para el conteo automático de tráfico.** Se localizan en diferentes áreas de la ciudad y están conectadas en línea al sistema de control. Para cada una de ellas es posible conocer las mediciones de tránsito registradas cada 5 minutos durante todo el día, posibilitando de esta manera los análisis y estimaciones de crecimiento de los flujos vehiculares.
- **Sistema de detección por video.** Consiste en un software especial que puede ser asociado a cualquier monitor de televisión y, mediante la definición de detectores virtuales, el sistema puede realizar, entre otras cosas, conteos vehiculares, medición de velocidades e incluso, el manejo de intersecciones críticas, a través de una interfaz especial.

El SAGT está integrado por los sistemas que se mencionan a continuación (Mercado, 2006):

i) Control de Tráfico Urbano (CTU ó UTC, Urban Traffic Control)

El sistema CTU, es la parte principal dentro del SAGT y comparte sus principales objetivos con el SIT, como son, reducir los problemas de tráfico y la contaminación, reducir accidentes y ahorrar energía en el transporte. Su función es la de recolectar información desde la carretera, a través de varios tipos de sensores, controlar las señales de tráfico basándose en dicha información, darle información a los conductores y eventualmente enviar instrucciones a las oficinas de tránsito.

ii) Sistema de Gestión de Tráfico en Carretera (SGTC ó FTMS, Freeway Traffic Management Systems)

El objetivo del sistema SGTC, es el de optimizar la capacidad de una carretera sin tener que aumentar físicamente más carriles, mejorar la seguridad dentro de la misma y proveer de un mejor servicio a los usuarios. Para lograr esta meta se debe obtener la información de tráfico y ser capaz de proponer rutas alternas así como controlar la velocidad de los automovilistas. El sistema SGTC dará a conocer rápidamente a los viajeros los accidentes ocurridos y ayudará a balancear el tráfico entre la carretera y las rutas alternas a ésta.

iii) Gestión de la Demanda (GD ó DM, Demand Management)

El sistema GD, debe administrar la demanda de las vías, con la finalidad de moderar la congestión vial que pueda darse. Un método es el de utilizar sistemas electrónicos de cobro del peaje.

2.2.2. Sistema Avanzado de Información al Viajero (SAIV)

La función primordial de los Sistemas Avanzados de Información al Viajero (SAIV), es la de dar información a los viajeros, con relación a destinos específicos y cómo llegar a ellos. Todo esto se hace con el objetivo de dar mayor y mejor información a los viajeros para que puedan tomar mejores decisiones con respecto a su ruta de viaje (Mercado, 2006).

El principio básico del SAIV es que mientras más información se dé al conductor, mejores decisiones podrá tomar. Algunos ejemplos de SAIV son sistemas de abordaje en los vehículos, estaciones de reporte de tráfico, mapas de congestión vial, información de tránsito a través del Internet, información por medio de sistemas personales de comunicación tales como celulares, handhelds, relojes, etc. Toda la información que se proporciona a los usuarios es colectada primero desde el SAGT y después es procesada y difundida por medio del SAIV (Mercado, 2006).

Un ejemplo de SAIV que está funcionando de manera exitosa, es el sistema de medición implementado en Southampton, Inglaterra que se describe a continuación (Federal Signal Corporation, 2009).

A través de la lectura automática de las placas vehiculares en dos o más puntos de una red carretera, se puede obtener información de los tiempos de viaje, tránsito vehicular, velocidad promedio de los vehículos en las distintas vialidades, etc. Los datos de la placa captada por el sistema de medición ubicado en las orillas del camino, son transmitidos a un sistema central, donde los datos recolectados desde distintos puntos sirven para estimar los tiempos de viaje entre la posición en ese instante de tiempo y el destino, como se muestra en la Figura 2.1.

Esta información se muestra en las señales de gestión del tráfico en los accesos a la ciudad, se envía también a las estaciones de radio locales, y se publica en tiempo real en un mapa interactivo en Internet.

Los automovilistas tienen a su disposición información en tiempo real, antes de viajar, para poder planear su ruta, e información en el camino, para tomar rutas alternas si se presenta mayor congestión en puntos específicos de la ciudad. Los automovilistas también tienen información sobre la disponibilidad de estacionamientos y sus respectivos precios.

La ciudad de Southampton en el Reino Unido está haciendo uso de esta tecnología en sus carreteras y autopistas, lo cual ha ayudado a aumentar la velocidad promedio de 3 mph a 18 mph. A través de esta tecnología es posible reducir la congestión vehicular.



Figura 2.1. Elementos que conforman un SAIV

Fuente: (Federal Signal, 2009)

2.2.3. Sistema Avanzado de Control de Vehículos (SACV)

Los Sistemas Avanzados de Control de Vehículos (SACV), engloban todo tipo de tecnologías o sistemas aplicados a los vehículos que pueden hacer la conducción más segura ya sea proporcionando información al conductor, o actuando de manera directa sobre la conducción. Estos sistemas son desarrollados principalmente por las compañías automotrices. Algunos ejemplos de SACV son los sistemas ABS (Anti Blocking System), controles de tracción, alarmas de colisión, sistemas antivolcamiento, sistemas anticolidión etc. (Mercado, 2006).

Como ejemplo de SACV, están los sistemas anticolidión que funcionan en la actualidad, como el sistema CITY SAFETY de la compañía VOLVO. Estos sistemas operan únicamente de forma local, es decir, el vehículo donde se implementa es el único que tiene opción de evitar el impacto.

La velocidad máxima en la que puede funcionar el sistema CITY SAFETY no debe rebasar los 30 km/hr, es decir, dicho sistema está diseñado para prevenir colisiones traseras a bajas velocidades en la ciudad, ya que la mayoría de las colisiones en las grandes urbes ocurren de esta manera.

Cisneros (2009), menciona que después de ser perfeccionados este tipo de sistemas, deberán evolucionar hacia sistemas cooperativos, es decir, que permitirán la comunicación entre vehículos y con la infraestructura.

El mismo autor (Cisneros, 2009) menciona también que diferentes sensores instalados en el vehículo monitorizan constantemente su entorno, como se muestra en la Figura 2.2, de forma que se puedan detectar inmediatamente aquellos vehículos u obstáculos que entren en el rango de alcance de dichos sensores, entre los que se encuentran tecnologías basadas en detección por infrarrojos, radares laser y detección por tratamiento de imágenes de video (cámara CMOS). Cuando un objeto (vehículo, peatón, obstáculo, etc.) es detectado, el sistema determina si el vehículo está en una situación en la que pueda producirse una colisión inminente y, en caso afirmativo, se envía al conductor un primer aviso para que realice una maniobra de frenado que pueda evitar el impacto. Las formas en que el sistema avisa al conductor de lo que podría ser una colisión inminente pueden ser variadas, según el fabricante de la tecnología, incluyendo:

- Proyección de aviso visual en el parabrisas del vehículo.
- Señales acústicas que, interfieren menos en la conducción.
- Otros avisos, como vibraciones en el volante o, incluso, frenadas intensas pero de corta duración que atraigan la atención del conductor.

Normalmente la señal de aviso suele ser una combinación de los diferentes tipos mencionados. Después de enviar una señal de aviso, si el conductor no modifica su velocidad, el sistema frena su vehículo automáticamente.

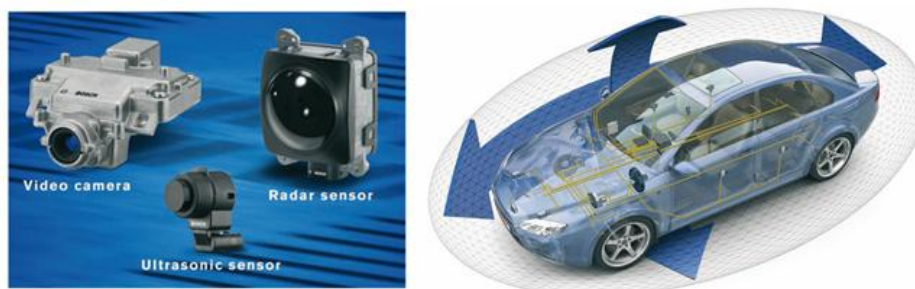


Figura 2.2. Componentes de un sistema AVCS

Fuente: (Cisneros, 2009)

Sin embargo, antes de evolucionar hacia un sistema cooperativo, el sistema CITY SAFETY, deberá de mejorar su eficiencia, ya que en algunas de las pruebas que la compañía Volvo realizó ante los medios de comunicación el sistema no funcionó, inclusive, algunos de los usuarios que han adquirido un coche con esta tecnología han subido sus videos a Youtube, donde se demuestra también que el sistema es poco confiable. En poco tiempo el sistema deberá mejorar y quizá hasta ampliar el rango de funcionamiento, el cual es muy limitado (hasta 30 km/h).

También hace falta generar un estándar de comunicación entre vehículos, para que a futuro, los vehículos de distintas compañías puedan compartir información entre sí y también puedan compartir información con la infraestructura.

2.2.4. Operaciones de Vehículos Comerciales (OVC)

Los sistemas para Operaciones de Vehículos Comerciales (OVC), tienen la finalidad de ayudar tanto a los transportistas grandes como a los pequeños. Se implementan sistemas OVC para facilitar y economizar la logística de transporte de vehículos de carga y de transporte humano. Con estos sistemas se evitan paradas e inspecciones innecesarias y facilitan la identificación del vehículo, conductor y carga. Simplifican la comunicación entre el transportista, la base, y los centros de ayuda y monitoreo. A la vez ayudan a evitar congestiones provocadas por vehículos de carga. Ejemplos de estos sistemas son credenciales electrónicas del vehículo de carga, sistemas de cobro automático de peaje, entre otros (Mercado, 2006).

A continuación se cita un ejemplo de lo ocurrido con los vehículos comerciales en carreteras estadounidenses.

La tecnología de Reconocimiento Automático de Matrículas (RAM ó ALPR, Automatic License Plate Recognition) (Federal Signal Corporation, 2009), ha sido implementada con éxito en aplicaciones para vehículos comerciales, integrándose con los Sistemas Inteligentes de Imagen (IIS), formando lo que se conoce como Solución Inteligente de Inspección en Carretera (SIIC). La solución es la integración de varias tecnologías, incluyendo: RAM, sensores de pesaje en movimiento (WIM, weight-in-motion), detectores

de radiación, sensores térmicos, video digital y otras tecnologías de punta, como se muestra en la Figura 2.3.

A través de esta solución, se puede centrar la atención y resolver los problemas en los vehículos de alto riesgo, como los que transportan sustancias o materiales peligrosos, también se puede identificar vehículos que transporten sustancias ilegales, o vehículos robados o vinculados con algún hecho ilícito. Esta nueva solución, está en uso actualmente a lo largo de las redes de transporte de los Estados Unidos.



Figura 2.3. Elementos utilizados en un OVC

Fuente: (Federal Signal, 2009)

El Departamento del Transporte de los Estados Unidos opera dos tipos de sistemas OVC en las carreteras de ese país, el PrePass y el Advantage 1-75. El PrePass, es un sistema de acreditación electrónica que permite a los camiones ser pesados en movimiento y verificar los datos del vehículo, sin necesidad de detenerlo, cuando éstos se aproximan a las estaciones de pesaje. Después de que los camiones se pesan y se comprueban las credenciales, un dispositivo en la cabina avisa al conductor si la estación de pesaje se puede omitir.

El Sistema Advantage 1-75 opera en los 3,500 km de la autopista de Ontario a Miami, a través de seis estados (Florida, Georgia, Tennessee, Kentucky, Ohio y Michigan), los conductores de los camiones deben detenerse una sola vez para registrarse en el sistema, posteriormente podrán moverse a lo largo del área que cubre el sistema Advantage 1-75, usando el Sistema Automático de Acreditación.

Daza y de la Torre (2006) mencionan que en México el conocimiento de las tecnologías de SIT aún no está totalmente divulgado. Sólo algunos sectores hacen uso de manera aislada de algunos de los servicios ofrecidos por estas nuevas tecnologías. En particular, algunas empresas del sector transportista de carga utilizan en forma limitada los servicios de cobro electrónico de peajes, y la localización por sistemas de posicionamiento global de sus vehículos.

La transferencia electrónica de fondos y los sistemas de información para los conductores muestran una alta utilización por parte de las empresas transportistas de carga y con flota propia. Es importante recordar que los sistemas de información a los conductores se encuentran en una etapa de desarrollo, en la que el uso de redes de telefonía celular para transmitir información en forma oral es el medio prevaleciente de comunicación.

En la actualidad, se ha incrementado el número de empresas que ofrecen y utilizan servicios para el rastreo satelital de vehículos, incluyendo también funciones como paro de motor vía remota, encendido de luces y sirena, apertura remota de seguros, detección de eventos a través de sensores como exceso de temperatura, apertura de puertas, encendido y apagado de motor, etc. Esta información ayuda a generar un historial de viajes, con estadísticas como consumo de combustible, exceso de velocidad, horas de trabajo de los vehículos, kilometraje, etc.

2.2.5. Sistema Avanzado de Transporte Público (SATP)

Los Sistemas Avanzados de Transporte Público (SATP), son sistemas y tecnologías enfocadas a mejorar el sistema público de transporte. Utilizan los SAGT, SAIV y SACV para mejorar sus viajes y sus servicios al usuario. También mejoran la seguridad de los vehículos, pueden contener sistemas de localización automática de vehículos (LAV ó AVL, Automatic Vehicle Location) y ofrecen al usuario información actualizada de los autobuses que espera, y de las rutas y los tiempos asociados (Federal Signal Corporation, 2009). Los sistemas LAV y de Información al Pasajero en Tiempo Real (IPTR ó RTPI, Real Time Passenger Information), son ejemplos de aplicaciones para los sistemas SATP.

El Departamento de Transporte del Reino Unido (2010), menciona que en la actualidad los sistemas IPTR, proporcionan información acerca de los servicios de autobús a través de:

- Monitores en las paradas, como el mostrado en la Figura 2.4.
- Páginas web.
- Dispositivos móviles.

El mismo Departamento de Transporte del Reino Unido (2010), indica que, la información que se muestra al usuario (en tiempo real), informa sobre los tiempos de llegada del autobús, y el estatus del mismo, es decir, si está a tiempo o se ha retrasado.

El Departamento de Transporte del Reino Unido (2010) menciona que los sistemas IPTR cuentan con un sistema de localización automática de vehículos, a través de equipos GPS instalados en cada autobús, cuentan también con un transmisor de radio, un servidor central y señales en las paradas de autobús.

El autobús comunica su posición de forma regular (cada 30 segundos), cuando se alcanza cierto punto o no llega a un punto determinado en un período de tiempo establecido. El servidor central interpreta la información del autobús y envía los datos a través del transmisor de radio, a los monitores ubicados en las paradas de autobús. La información también se envía a páginas web y a dispositivos móviles que así lo soliciten.



Figura 2.4. Sistema RTPI en el Reino Unido

Fuente: (Departamento de Transporte, UK, 2010)

Este sistema ha demostrado una gran efectividad y ha tenido gran aceptación por parte de los usuarios del transporte público del Reino Unido, por ejemplo, en la ciudad de Liverpool, según cifras del Departamento de Transporte del Reino Unido, el sistema (Time Checker-Liverpool) ha mostrado una precisión del 90%. El 68% de los usuarios utilizan el sistema Time checker de manera constante. El 92% percibe a los sistemas de Información en Tiempo Real (ITR) como "muy precisos" o "precisos". El 87% tuvo una sensación de mayor tranquilidad y un 73% dijo que se sentía más seguro al esperar el autobús en la noche. El 57% cree que el sistema ITR, ha provocado una disminución en los tiempos de espera, el 71.5% cree que, los servicios de autobús mejoraron en general cuando las pantallas electrónicas se instalaron. Finalmente el 89% quiere ver que los sistemas ITR se implementen en todas las paradas de Liverpool.

Existen casos como el de la ruta B63 en Nueva York (Cobble Hill-Fort Hamilton) donde inclusive algunos de los negocios, como los ubicados en Atlantic Avenue en Brooklyn, muestran en sus aparadores la información de los autobuses. La autoridad encargada de poner este proyecto en marcha es la MTA (Metropolitan Transportation Authority). También los usuarios de esta ruta pueden acceder a la página web de la MTA para ver la

distancia y tiempo que tardarán los autobuses en llegar a la parada donde se ubique el usuario, como se muestra en la Figura 2.5. Se puede verificar directamente esta información, a través de teléfonos Smartphone con lectores de código QR (Quick Response); los códigos bidimensionales se ubican en cada una de las paradas. También se puede mandar un mensaje de texto vía celular, con el código de la parada de autobús, para obtener los tiempos y distancias de los autobuses de la ruta. Existen aplicaciones para otro tipo de dispositivos móviles (Iphone, Ipod, Ipad) que muestran la información completa sobre la ruta B63.

Con lo anteriormente descrito, se observa, que a través del uso de distintas tecnologías se pueden conformar sistemas SATP, que resuelvan las necesidades de los usuarios.

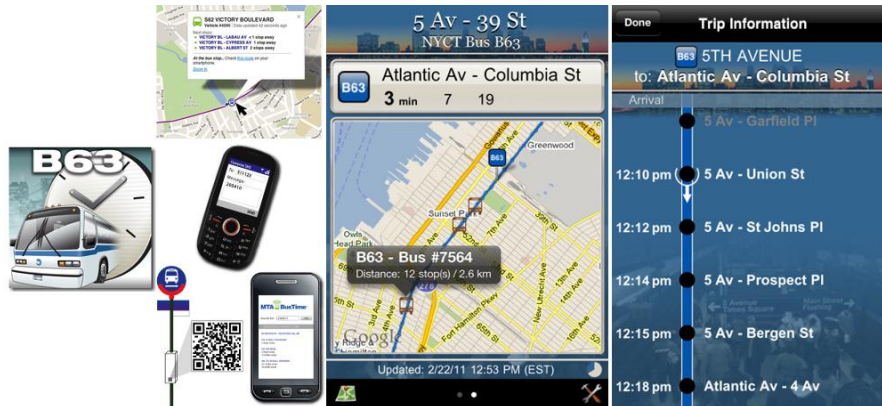


Figura 2.5. Sistema RPTI. Ruta B63, Nueva York

Fuente: Propia

2.2.6. Algunas aplicaciones de los SIT

La compañía Siemens creó un sistema en el que cada espacio en el estacionamiento está equipado con un sensor ultrasónico, que detecta si el lugar está ocupado o disponible (SIEMENS AG, 2007). Los operadores del estacionamiento tienen una visión completa del porcentaje de ocupación del estacionamiento y de la distribución de dicha ocupación, como lo muestra la Figura 2.6. Esto facilita la administración del estacionamiento, elimina los errores de recuento de vehículos, evita el congestionamiento interno, reduce los tiempos de acceso y espera para los usuarios y reduce las emisiones contaminantes.

Incluso es posible evitar la entrada de los conductores en un sector con pocos espacios libres disponibles, para distribuir la demanda y evitar aglomeramientos.



Figura 2.6. Sistema SIPark SSD

Fuente: (Siemens AG, 2007)

Los datos y la energía eléctrica viajan por el mismo cable de red (AS interface bus), esto produce una reducción en los costos de cableado. Los datos viajan vía Ethernet hasta unos concentradores de datos y posteriormente hasta el centro de control y operación. La información regresa a los usuarios de forma visual a través de distintas pantallas, por ejemplo, pantallas que muestran las zonas disponibles para estacionarse, o pantallas que muestran los niveles de ocupación, etc., el diagrama de funcionamiento del sistema SIPark, se muestra en la Figura 2.7.

Siguiendo con la idea de los Sistemas Inteligentes de Transporte, dicho sistema está diseñado para acoplarse con otros sistemas de estacionamiento, o con sistemas de reservaciones.

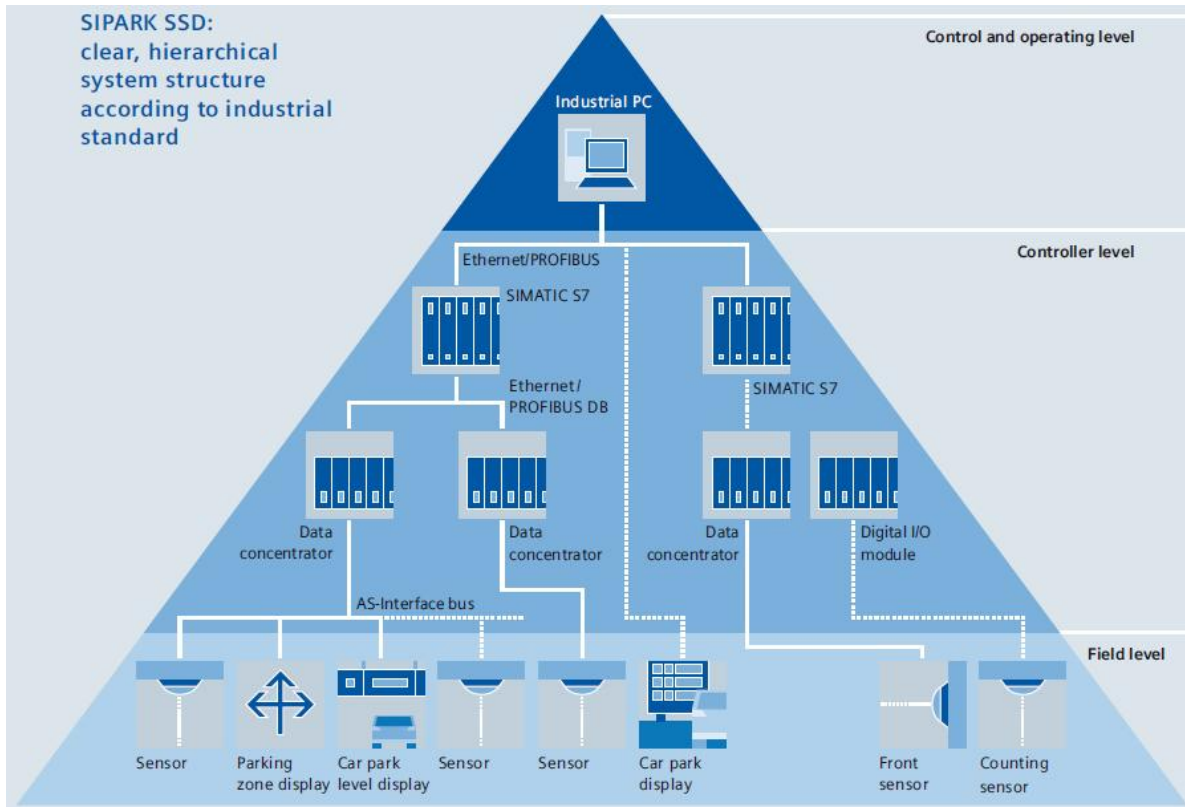


Figura 2.7. Diagrama de funcionamiento del sistema SIPark SSD

Fuente: (Siemens AG, 2007)

El estacionamiento más grande del mundo donde se está aplicando este sistema, es el de la terminal 1 del aeropuerto "Franz-Josef Strauss" de la ciudad de Múnich (Figura 2.8). A partir del año 2009 los sistemas guía con los que cuentan los estacionamientos de las terminales 1 y 2 controlan en más de 15,000 espacios de estacionamiento. Este sistema puede aplicarse en cualquier estacionamiento que se quiera controlar, sin importar su tamaño, algunos lugares donde se ha implementado se muestran en la Tabla 2.2.



Figura 2.8. Aeropuerto de Múnich

Fuente: (Munich Airport, 2014)

Tabla 2.2. Algunos lugares donde se ha implementado el sistema SIPARK SSD

Cliente	Ubicación	Espacios de estacionamiento
Modern Universe Estate Shanghai	China	358
Leppävaara Espoo	Finlandia	1,620
Tienda KPMG, Berlín	Alemania	10
Cruz Roja de Múnich	Alemania	40
Gobierno de la Provincia de Linz	Austria	1,152
Centro de Servicios Langovtrg Zagreb	Croacia	303
Hipermercado Carrefour, Estambul	Turquía	2,400
Estacionamiento City Hall, St. Gallen	Suiza	85
Torre mayor de Singapur	Singapur	600
Convento Carmelita de Linz	Austria	405
Estacionamiento 983 de Abu Dhabi	Emiratos Árabes Unidos	786

Fuente: (SIEMENS AG, 2007)

En la Tabla 2.3, se muestran algunas de las aplicaciones y la clasificación de los SIT mencionada por Taylor (2001).

Tabla 2.3. Aplicaciones de los SIT

Aplicación SIT	Descripción	Ejemplos
Control de tráfico automático (ATC, Automatic traffic control)	Sistema que controla y facilita el flujo del tráfico y peatones, coordinando de forma adaptativa en tiempo real las señales de tráfico. Los elementos en los caminos son monitoreados para proveer información sobre volúmenes vehiculares y velocidades. El sistema puede dar prioridad a los vehículos de emergencia.	FETSIM en California y los sistemas de control de tráfico SCATS Y SCOOT alrededor del mundo.

Fuente: (Taylor, 2001)

Tabla 2.3. Aplicaciones de los SIT (continuación)

Aplicación SIT	Descripción	Ejemplos
Control automático del vehículo (AVC, Automatic Vehicle Control)	Sistemas que hacen uso de computadoras, sensores en el camino, radares y otros dispositivos, para controlar la dirección y velocidad de los vehículos, sin involucrar necesariamente a un operador humano en algunas o en todas las tareas de manejo.	Carriles de autopista automatizados probados cerca de San Diego CA., como parte del proyecto CALTRANS PATH. Pruebas sobre adaptación inteligente de velocidad en Suecia.
Sistemas de información al conductor (DIS, Driver Information System)	Sistemas que proporcionan al conductor información acerca de las condiciones del tráfico y congestión en la red. La información puede proporcionarse antes o durante el viaje.	Sistema de guía en San Antonio (Departamento de Transporte de Texas).
Monitoreo del ambiente/Contaminación (EPM, Environment / pollution monitoring)	Sistema que monitorea y procesa las mediciones de un número de sensores en un área o a lo largo del camino y transmite advertencias e implementa estrategias para el control del tráfico en respuesta al cambio en los niveles de contaminación o de las condiciones ambientales.	Sistemas de alerta de niebla y nieve usados en distintos países.
Cobro Electrónico de peaje (ETC, Electronic Toll Collection)	Sistemas de pago automático de cuotas o cargos a los usuarios de carreteras y puentes sin la necesidad de frenar los vehículos. Comúnmente los vehículos están equipados con una "tarjeta inteligente" o son reconocidos electrónicamente de tal forma que al pasar el vehículo el cargo se realiza a la tarjeta o a una cuenta.	Muchos sistemas ETC han sido instalados en autopistas de distintos países. El ejemplo más notable es el sistema electrónico de precios de caminos en Singapur.

Fuente: (Taylor, 2001)

Tabla 2.3. Aplicaciones de los SIT (continuación)

Aplicación SIT	Descripción	Ejemplos
Sistema de Gestión de Carga (FMS, Freight management system)	Sistema diseñado específicamente para mejorar la confiabilidad, calidad de servicio, y eficiencia del transporte de carga, con un gran control sobre las operaciones de la flota a través del seguimiento e identificación de vehículos, calendarización computarizada y comunicaciones de dos vías.	Booz Allen y Hamilton (1998) describen una serie de sistemas propietarios usados por las compañías de carga en los Estados Unidos.
Sistema de Gestión de Incidentes (INM, Incident Management System)	Sistemas que buscan detectar rápidamente incidentes (como choques, descompostura de vehículos, o bloqueo de carriles) en un camino y recomendar planes para una rápida respuesta de tal manera que el incidente se pueda resolver.	Muchas de las carreteras de los Estados Unidos y en cualquier sitio equipado con sistemas INM, por ejemplo el sistema CHART de Maryland.
Información de Transporte Público (PTI, Public Transport Information)	Sistemas que proporcionan información a los viajeros acerca de la disponibilidad de los servicios del transporte público, pueden recomendar servicios específicos para un viaje dado. La información puede ser proporcionada antes o durante el viaje y puede estar basada en horarios o en información en tiempo real sobre las progresiones de los vehículos.	Existen una serie de sistemas PTI operando actualmente en el Reino Unido.
Gestión de Transporte Público (PTM, Public Transport Management)	Sistema que mejora la eficiencia y seguridad de los vehículos de transporte público, como los horarios computarizados, turnos y envíos de vehículos, monitoreo de flotas y sistemas de control inteligente.	Un ejemplo familiar es el sistema computarizado de envío de vehículos usado comúnmente por las compañías de taxis.

Fuente: (Taylor, 2001)

Tabla 2.3. Aplicaciones de los SIT (continuación)

Aplicación SIT	Descripción	Ejemplos
Sistemas de Guía de Rutas (ROG, Route Guidance System)	Sistemas que proporcionan navegación, información de guía de rutas, planificación de la información de viaje, a través de medios como los mapas electrónicos o instrucciones sintetizadas de voz.	Muchas compañías de alquiler de coches proporcionan sistemas de navegación en el vehículo rentado.
Mejoramiento de la Seguridad en el Camino (RSE, Road Safety Enhancement)	Sistemas que ayudan a los conductores a evitar choques y/o proporcionan una mejor protección y respuesta a emergencias si ocurre el choque.	Sistemas de frenado automático para evitar colisiones y sistemas de asistencia automática en emergencias.

Fuente: (Taylor, 2001)

2.3. Organismos SIT alrededor del mundo

Debido al impacto positivo que genera la aplicación de los SIT en cuanto a movilidad, seguridad, costos sociales, financieros, y medio ambiente, existen organismos SIT en una gran cantidad de países. Todos ellos en esencia tienen los mismos objetivos: investigación, creación de nuevas tecnologías, promoción y estandarización de sistemas SIT, destinados a mejorar la calidad en el transporte.

Algunos de los organismos SIT de mayor relevancia son: ERTICO (ITS Europa), PAITS (ITS Panamericano), ITS Iberoamérica, ITS Asia-Pacífico, ITS América (E.U.), ITS Canadá, ITS Reino Unido, ITS Argentina, ITS Brasil, ITS Chile, ITS Alemania, ITS Francia, ITS España, ITS Italia, ITS Australia, ITS Japón e ITS Corea. Los logotipos de las organizaciones más importantes se muestran en la Figura 2.9.

Existen algunos países que impulsan más la investigación, generación e implementación de tecnología SIT; en otros prevalecen los enfoques normativo y político, como ocurre en el nuestro.



Figura 2.9. Logotipos de algunos organismos ITS

Fuente: Propia con imágenes de ITS Iberoamérica

Países como Japón, Estados Unidos, Canadá, Inglaterra, Francia (y en general la Comunidad Europea), Alemania, Australia, Brasil, Chile, Argentina entre otros, han implementado en sus respectivos países y bajo ciertos estándares los Sistemas Inteligentes de Transporte.

Para lograr una estandarización, es necesaria la implementación de una arquitectura de SIT, para ello ITS Japón recomienda seguir el siguiente procedimiento (ITS Japón, 2002 citado por Acha y Espinoza, 2004):

- 1) Definir con detalle los servicios que se ofrecerán a los usuarios de los SIT
- 2) Diseñar la Arquitectura Lógica, también llamada Arquitectura Funcional
- 3) Diseñar la Arquitectura Física, e
- 4) Identificar aquellos elementos susceptibles de estandarización.

Algunas preguntas que podrían plantearse son, ¿Qué se ha implementado en nuestro país? ¿Cuáles son los elementos más importantes a considerar para la implementación de la arquitectura SIT en México?

Para ello la Secretaría de Comunicaciones y Transportes a través del Instituto Mexicano del Transporte, consideró que los servicios listados (en orden de importancia) en la Tabla 2.4, deberían desarrollarse para conformar la Arquitectura SIT mexicana.

Tabla 2.4. Servicios al usuario de la arquitectura SIT México

Servicio al usuario
Administración del transporte público
Información de servicios al viajero
Información previa al viaje (peatones, conductores)
Administración de vehículos de emergencia
Monitoreo de seguridad a bordo del vehículo
Procesos de administración de vehículos comerciales
Inspección al lado del camino
Control de tránsito
Proveer información de ocurrencia de restricciones al tránsito
Seguridad para el usuario del transporte público
Servicios de pago electrónico
Administración de flotillas comerciales
Administración de condiciones ambientales (clima y medio ambiente)
Administración de la demanda de viajes
Respuesta a incidentes con materiales peligrosos
Verificación y reducción de emisiones
Activación de restricciones al movimiento antes de impactos
Liberación electrónica de vehículos comerciales
Administración de carga intermodal
Eliminación de colisiones basada en los vehículos
Selección de rutas
Administración de incidentes y desastres
Apoyo al viajero en tránsito (navegación)
Operaciones de mantenimiento y construcción
Operaciones y mantenimiento
Reservaciones y vehículos compartidos
Seguridad basada en infraestructura
Mejoras a la seguridad basada en sensores
Notificación automática de emergencias y personal de seguridad
Transporte público personalizado
Sistemas de carreteras automáticas
Operación automática de vehículos
Funciones de información almacenada

Fuente: (Acha y Espinoza, 2004)

Para implementar la arquitectura SIT, la SCT tomó como referencia el modelo implementado en EU, lo que puede ocasionar que no se resuelvan los problemas de la mejor manera, o que ni siquiera puedan resolverse, ya que existen muchas diferencias de funcionamiento y organización de un país a otro. Además, en nuestro país apenas se comienza con el uso de los SIT, y los que se han implementado son muy pocos, entre los que se encuentran las tarjetas para el cobro de peaje automatizado, y los sistemas adaptativos de control del tráfico (en muy pocas ciudades y que incluyen un limitado número de intersecciones semaforizadas).

El organismo ITS México presenta deficiencias, falta de cohesión con otros organismos encargados del transporte en México, y la información publicada en su página web es inexistente, a diferencia de otros organismos SIT cuyas páginas web muestran lo que hacen, proyectos a futuro, socios comerciales, proyectos de investigación, etc.



Figura 2.10. ITS México

Fuente: (ITS México)

2.4. Congresos Internacionales

Existen diversos congresos que se realizan en todo el mundo impulsados por los organismos SIT de cada uno de los países, entre los más importantes se encuentran:

- Congreso Mundial de SIT
- Congreso Europeo de SIT
- Congreso SIT Asia
- Reunión anual y exposición, SIT América
- Reunión general y conferencia anual, SIT Canadá
- Congreso Español de SIT.

En cada uno de ellos se abordan temas sobre nuevas tecnologías, tendencias, normatividad, sustentabilidad, transporte público, transporte de carga, etc. El congreso más importante es el *Congreso Mundial de SIT*, y debido a la gran cantidad de temas que allí se abordan, han establecido la siguiente clasificación:

- Movilidad cooperativa
- Recopilación de datos y medición del desempeño
- Vehículos de carga y comerciales
- Infraestructura, tránsito y manejo de congestión
- Financiamiento
- Siguiete generación de información al viajero y aplicaciones del consumidor
- Cuestiones políticas e institucionales
- Movilidad y operaciones
- Políticas y estrategias
- Transporte público
- Sustentabilidad
- Seguridad
- Sistemas en vehículos y electrónicos

De la anterior clasificación, el rubro que se relaciona más con el trabajo de tesis es: “Siguiete generación de información al viajero y aplicaciones del consumidor”; en el XVIII Congreso Mundial, llevado a cabo en Orlando, E.U. en el año 2011 se abordaron temas como:

- Integración de servicios de información al viajero para viajes intermodales.
- Servicios basados en localización para el transporte público.
- El futuro de internet y el futuro de los SIT.
- Planeación de rutas personalizadas: siguiete generación, información al viajero.

Estos temas se relacionan de manera directa con la problemática que se aborda en el presente trabajo de tesis, donde se implementará un Sistema Avanzado de Información al Viajero (SAIV). El sistema web proporcionará al usuario una hiperruta mínima (para mayor referencia ir al tema 3.3) personalizada entre dos puntos ubicados dentro del área de

estudio (definida en el punto 1.4.), la cual podrá combinar el transporte público con el privado y por lo tanto tendrá que analizar la viabilidad de las combinaciones que se pueden generar, además las rutas generadas estarán sujetas a un número máximo de transferencias modales establecidas por el usuario.

El *Congreso Mundial de SIT* tiene 19 años de llevarse a cabo, como se muestra en la Tabla 2.5.

Tabla 2.5. Ediciones del Congreso Mundial sobre SIT

I	1994	Paris, Francia
II	1995	Yokohama, Japón
III	1996	Orlando, E.U.
IV	1997	Berlín, Alemania
V	1998	Seúl, Corea del Sur
VI	1999	Toronto, Canadá
VII	2000	Torino, Italia
VIII	2001	Sídney, Australia
IX	2002	Chicago, E.U.
X	2003	Madrid, España
XI	2004	Nagoya, Aichi
XII	2005	San Francisco, E.U.
XIII	2006	Londres, Inglaterra
XIV	2007	Beijing, China
XV	2008	Nueva York, E.U.
XVI	2009	Estocolmo, Dinamarca
XVII	2010	Busan, Corea del Sur
XVIII	2011	Orlando, E.U.
XIX	2012	Viena, Austria

Fuente: Toyota, 2012

Todos los países que han sido sede del Congreso Mundial de SIT son países desarrollados económica y tecnológicamente, quedando marginadas regiones como Latinoamérica, África y la mayor parte de Asia, esto es lógico hasta cierto punto, ya que

es en estos países donde se generan los mayores avances tecnológicos, sin embargo, deberían involucrarse los países en desarrollo (incluyendo al nuestro) para que a través del uso de tecnología de vanguardia se generen los cambios que requieren estos países y de esta forma poder solucionar los grandes problemas que les aquejan.

2.5. Sistema Avanzado de Información al Viajero (SAIV)

Rilling y Brestold (Gandy y Meitner, 2007) definen a los Sistemas Avanzados de Información al Viajero (SAIV), como aquella información que ayuda al conductor con la planeación, la percepción, el análisis y la toma de decisiones para mejorar la comodidad y la eficiencia del viaje. La definición que plantean los autores considera únicamente a los SAIV como parte de un viaje realizado solo en transporte privado y deja de lado la posibilidad de que el viaje se realice en transporte público o combinando ambos modos.

Como introducción a los SAIV, se abordará en primera instancia la historia de dichos sistemas, a través de lo mencionado por Adler y Blue (1998).

2.5.1. Historia de los SAIV

En la década de los 60's y a principios de los 70's las grandes urbes como Los Ángeles, Detroit y Chicago, comenzaron a investigar, a desarrollar y a implementar tecnologías avanzadas para el monitoreo del tránsito y la propagación de dicha información en tiempo real. Trabajaron en sistemas de información al viajero, enfocados en el uso de letreros electrónicos, que de manera visual daban información a los usuarios sobre las condiciones del tránsito y alternativas o desviaciones que podían tomar.

En la década de los 70's el proyecto más importante que fue patrocinado por la Administración de Carreteras Federales (Federal Highway Administration), fue el denominado ERGS (Electronic Route Guidance System) o Sistema Electrónico de Orientación y Rutas (SEOR). Su objetivo era proveer a los conductores dentro del vehículo con información de orientación, que les permitiera ir de un origen a un destino previamente establecido. En el año de 1971 el gobierno de los Estados Unidos abandonó

tempranamente el proyecto. A mediados de la misma década, Japón comenzó con una serie de proyectos de investigación basados en el modelo SEOR.

Hacia finales de la década de los 80's los sistemas de señales viales controlados de manera centralizada y los sistemas de advertencia en carreteras, fueron cada vez más comunes en las áreas metropolitanas. Además en la misma década, fueron probados en muchas partes del mundo los Sistemas de Orientación y Rutas en los Vehículos (SORV ó IVRGS, In-vehicle Route Guidance Systems), particularmente en Japón, Europa y en los Estados Unidos. Con la rápida evolución de los SIT y el énfasis puesto en el control del tránsito en tiempo real, en los sistemas dinámicos de orientación y rutas, y en los sistemas de asignación dinámica del tránsito, provocó un mayor desarrollo de los SAIV durante la década de los 90's. (Adler y Blue, 1998)

Los sistemas de información al viajero, pueden ser clasificados por su temporalidad y avances, de la siguiente manera (Adler y Blue, 1998):

Primera generación. Surge gracias a las tecnologías emergentes de cómputo, el monitoreo de tránsito y los sistemas de control a finales de la década de los 60's y principios de los 70's. En muchos casos esos sistemas fueron diseñados para mejorar el flujo del tránsito en ciertos puntos de la red, como cruces altamente congestionados, o para evitar que los conductores por falta de información se dirijan hacia un sitio con congestión no recurrente, debido a alguna eventualidad, por ejemplo, un accidente. Los sistemas de Señales de Mensaje Variable (SMV ó VMS Variable Message Signs) y de Aviso por Radio para Carreteras (HAR, Highway Advisory Radio), son representativos de esta primera generación.

Segunda generación. Los sistemas SAIV de la actualidad pertenecen a la segunda generación, éstos cuentan con un amplio rango de tecnologías que han sido diseñadas para proveer a los conductores de guías y rutas dinámicas, información en tiempo real sobre las condiciones del tránsito y servicios información para los conductores. Los sistemas de información multimodal permiten a los conductores estar más informados y tener mayores posibilidades para elegir sus rutas. Los sistemas de la primera generación eran sistemas de una sola vía de comunicación, la responsabilidad de elección caía sobre el usuario, a diferencia de los sistemas de la segunda generación donde la cantidad de

información que se le ofrece al conductor es reducida, enfocándose en las necesidades específicas del usuario. Las propiedades de los sistemas de segunda generación son las siguientes:

- Interfaz de usuario interactiva
- Localización del vehículo y mapeo inteligente
- Búsqueda de rutas individualizadas
- Directorio de páginas amarillas
- Información multimodal
- Orientación de rutas dinámicas

Los sistemas computacionales tienden a ser cada vez mas intuitivos y sencillos de utilizar, pero de manera interna son cada vez más complejos y en ciertos casos tienden a imitar el comportamiento humano, como ocurre con los programas que hacen uso de la inteligencia artificial (IA), definida por el diccionario Webster's New World Dictionary of Computer Terms (Adler y Blue, 1998) como el conjunto de tecnologías que tratan de emular ciertos aspectos del comportamiento humano, como el razonamiento y la comunicación, y la imitación de sentidos biológicos como la vista y el oído. Lo mismo ocurrirá con los sistemas SAIV quienes incorporarán la inteligencia artificial para personalizar la información que se le da al viajero, y adaptarse a las necesidades del conductor a través de un aprendizaje.

Tercera Generación. Como mencionan Adler y Blue (1998), el siguiente gran salto de los sistemas de información al viajero, ocurrirá a través de la integración de la IA con los sistemas SAIV, llevando a la creación de los sistemas de tercera generación, llamados Sistemas Inteligentes de Información al Viajero o (SIIV ó ITIS, Intelligent Traveler Information Systems).

Las actividades básicas que pueden conseguirse a través de la unión de la IA con los sistemas ATIS, según Adler y Blue (1998) son las siguientes:

i) Reconocimiento del conductor. Cada conductor tiene sus propios objetivos de viaje, hábitos de manejo, comprensión y percepción de la red. El sistema almacena información específica sobre el usuario, la cual es invocada cuando el usuario ingresa al sistema.

ii) Comunicación con el conductor. La interface de usuario define la forma en la que el conductor interactúa con el sistema de guía. En la actualidad la mayoría de los sistemas de abordaje requieren una interacción manual para que el usuario opere el sistema, los sistemas con mayor inteligencia integran interfaces auditivas, donde los conductores dan instrucciones de voz y reciben instrucciones de navegación audibles.

iii) Aprender las preferencias del conductor. Ya sea implícitamente a través del histórico de viajes o explícitamente pidiendo a los conductores califiquen su percepción de las experiencias de viaje, los sistemas de guía deberán aprender de las preferencias de cada conductor.

iv) Interfaces con distintas fuentes de información en tiempo real. En un ambiente donde la información en tiempo real, incluyendo condiciones de tránsito y clima, esté disponible, los sistemas guías de ruta buscarán servidores de información, probablemente a través de transmisiones de radio, para desarrollar un itinerario de viaje que cumpla con los objetivos personales del conductor.

v) Minimizar los distractores que producen los sistemas a bordo. Los sistemas dentro del vehículo tienden a generar distracciones en el conductor, cuando éste desvía su atención del camino para centrarla en los dispositivos a bordo. El sistema deberá estar diseñado para alertar al conductor en caso de existir riesgo, y para no distraer la atención del conductor, dándole únicamente la información necesaria.

2.5.2. Clasificación de los Sistemas Avanzados de Información al Viajero

La Administración Federal de Carreteras del Departamento de Transporte de los Estados Unidos de América, clasifica a los sistemas avanzados de información al viajero, de la siguiente manera (U.S. Department of Transportation. Federal Highway Administration, 1996):

2.5.2.1. Sistemas de Navegación y Enrutamiento en el Vehículo

El Sistema de Navegación y Enrutamiento en el Vehículo (SNEV ó IRANS, In-Vehicle Routing and Navigation Systems), proporciona a los conductores información acerca de cómo llegar de un lugar a otro, y puede incluir información de operaciones de tránsito y congestinamiento urbano recurrente y no recurrente. Han sido identificados siete componentes funcionales que tienen aplicación tanto en los sistemas SAIV como en OVC, éstos son:

i) Planificación del viaje. Comprende la planificación de rutas con múltiples destinos, sitios visitados frecuentemente (históricos), así como la información para alojarse en hoteles, servicios de restaurantes y servicios información en el vehículo.

ii) Coordinación y planificación de viajes multimodales. La información es proporcionada a la persona que viaje o conduzca para su uso en diferentes modos de transporte (camión, tren, metro, vehículo particular etc.). Esta información puede incluir actualizaciones en tiempo real sobre el tiempo de llegada y los tiempos de viajes previstos.

iii) Preselección de rutas y selección de destino. Permite seleccionar al conductor cualquier destino y/o ruta antes de la partida. Las características de dicha preselección incluyen la selección del destino, tiempo de salida y las rutas hacia el destino. Los sistemas de información podrán incluir información sobre la congestión en tiempo real o de manera histórica, tiempo estimado de viaje, y rutas que optimicen parámetros como el tiempo o la distancia.

iv) Selección dinámica de rutas. Se refiere a cualquier sistema de selección de rutas que operen mientras el vehículo esté en movimiento, incluye la presentación de tránsito actualizado e información sobre incidentes que puedan afectar la selección de rutas del conductor. Además, el sistema podría alertar al conductor si tomó equivocadamente un camino. La selección de ruta dinámica puede generar una nueva ruta a partir de la posición que tenga en ese instante del conductor.

v) Guía de rutas. Esta capacidad incluye información direccional. Esta información puede ser en forma de una ruta seleccionada en un mapa electrónico, iconos indicando las direcciones a seguir en pantalla a través de una interface HUD (Head-up Display), o a través señales de voz.

vi) Navegación de rutas. Proporciona información para seleccionar el destino; sin embargo, no incluye la guía de rutas. Puede incluir dentro del vehículo la presentación de mapas electrónicos con calles, direcciones, localización actual del vehículo, localización del destino, y localización de servicios o lugares de interés. La navegación de rutas provee información que típicamente puede ser encontrada en mapas de papel.

vii) Cobro de peaje automatizado. Este sistema permite a un vehículo viajar a través de un camino de cuota sin la necesidad de pararse para pagar el peaje; el cobro puede ser deducido automáticamente de las cuentas de los conductores cuando pasan a través de las áreas de cobro de peaje.

2.5.2.2. Sistema de Información y Servicios al Conductor dentro del Vehículo

El Sistema de Información y Servicios al Conductor (IMSIS, In-Vehicle Motorist Services Information Systems), proporciona las siguientes capacidades:

i) Difusión de la información de servicios y sitios de interés. Proporciona información a los conductores que de otra forma podría ser encontrada en los señalamientos ubicados a la orilla de la carretera. Funciona de manera similar a la información sobre servicios y sitios interés que típicamente dan los señalamientos; pero éstos serán desplegados dentro del vehículo en movimiento.

ii) Acceso al directorio de servicios y sitios de interés. Proporciona información de moteles, hoteles, servicios de emergencia médica, entretenimiento y servicios de recreación. Accediendo esta capacidad, el conductor puede acceder a información relevante sobre negocios o atracciones.

iii) Coordinación de destinos. Permite al conductor comunicarse para hacer los arreglos necesarios en el lugar de destino. Esto puede incluir reservaciones en restaurantes y hoteles.

iv) Capacidad para la transferencia de mensajes. Permite al conductor comunicarse con otros. En la actualidad, esta función está implementada en los teléfonos celulares y en los radios de banda ciudadana (Citizens Band Radio). En el futuro, los sistemas SAIV pueden mejorar esta tecnología mediante la generación de mensajes automática con el toque de un botón y recibir los mensajes para su uso futuro. La transferencia de mensajes puede involucrar a los mensajes de texto y voz.

2.5.2.3. Sistema de Advertencia y Asesor de Seguridad dentro del Vehículo

El Sistema de Advertencia y Asesor de Seguridad dentro del Vehículo (SAASV ó IVSAWS, In-Vehicle Safety Advisory and Warning Systems), proporciona advertencias sobre peligro inmediato y situaciones o condiciones del camino delante del conductor. Se ofrece la advertencia suficiente que permite al conductor tomar las medidas correctivas necesarias (por ejemplo, reducir la velocidad). Este sistema no ofrece dentro del vehículo las advertencias de seguridad que alerten de un peligro inminente que requiera de acción inmediata (por ejemplo, cambio de carriles, advertencias sobre los puntos ciegos en el automóvil, dispositivos de pertenencia ante una colisión inminente, etc.). Además proporciona asistencia manual y automática.

i) Advertencia de peligro inmediato. Se puede advertir de un peligro inmediato al conductor indicando la posición relativa del peligro, el tipo de peligro, y el estado de los vehículos de emergencia en el área. Específicamente puede incluir la notificación al conductor, de la cercanía de un vehículo de emergencia o el aviso al conductor de un accidente ubicado en la cercanía del vehículo.

ii) Información sobre las condiciones del camino. Proporciona información sobre las condiciones del camino, tal como la congestión, obras en construcción, etc. todo esto dentro de un rango de proximidad definido previamente en el vehículo o en la ruta del conductor.

iii) Solicitud automática de ayuda. Genera una señal en aquellas circunstancias donde la solicitud manual de ayuda no sea factible y donde la respuesta inmediata se es esencial (accidentes graves). La señal ofrecerá información sobre la ubicación del accidente y su envío al personal que atenderá la emergencia.

iv) Solicitud manual de ayuda. Comprende todos aquellos servicios requeridos en una emergencia (L, ambulancia, remolque, bomberos). Permiten al conductor tener acceso desde el vehículo sin la necesidad de localizar un teléfono, conocer el número telefónico apropiado, o incluso conocer la ubicación del vehículo. Esta función puede incluir también la retroalimentación al notificar al conductor del estatus de la respuesta, como el tiempo esperado de arribo del servicio solicitado.

v) Monitoreo de las condiciones del vehículo. Permite dar seguimiento a las condiciones generales del vehículo y notificar conductor de los problemas actuales, así como los problemas potenciales. Esta función puede ser interactiva, proporcionando conductor la capacidad de interrogar al sistema acerca de los problemas y la obtención de más detalles sobre la situación.

2.5.2.4. Sistema de Información con Símbolos dentro del Vehículo

El sistema de Información con Símbolos dentro del Vehículo (ISIS, In-Vehicle Signing Information Systems), ofrece el despliegue dentro del vehículo de rutas no comerciales, advertencias, y asesoramiento e información que se encuentra actualmente representados las señales carreteras.

i) Guía de información carretera. Incluye las señales de tránsito, marcadores de ruta, y puestos de milla, que tradicionalmente se encuentran las señales externas al vehículo, esta información será desplegada dentro del vehículo.

ii) Notificación de información carretera. Las señales de carretera que actualmente notifican a los conductores de peligros potenciales o cambios en el camino, señalamientos

combinados, aviso sobre límite de velocidad, señales de curvas en el camino, serán desplegados en el vehículo.

iii) Información reguladora en carretera. Información tal como los límites de velocidad, paradas, señales para ceder el paso, revisiones de vuelta y uso de carriles serán proporcionados dentro del vehículo.

2.5.2.5. Funciones específicas de Operaciones de Vehículos Comerciales (CVO)

Las Funciones específicas de Operaciones de Vehículos Comerciales, ofrecen las siguientes funciones al usuario:

i) Programación de rutas. Coordinan los distintos destinos introducidos por el conductor, para minimizar el tiempo de viaje o para acomodar los horarios de entrega.

ii) Operación asistida por computadora. Ayuda al operador en la generación de rutas que coordine múltiples paradas definidas por el operador. Esta función podría indicar qué flota vehicular es la más adecuada para un destino en particular, basándose en información como el tipo de carga, localización del vehículo, y tiempos de entrega y recolección.

El sistema que se describe en el punto 2.5.2.6 no pertenece a la clasificación hecha por la Administración Federal de Carreteras del Departamento de Transporte de los Estados Unidos de América, sin embargo, el presente trabajo de tesis se centra en el desarrollo de un sistema de información para viajes multimodales, como parte de un SAIV, motivo por el cual, se incluye este sistema en la clasificación de los SAIV.

2.5.2.6. Sistema de información para viajes multimodales

El Sistema de información para viajes multimodales, proporciona al usuario información sobre los distintos modos de transporte público y privado que el usuario podrá utilizar en cierta área geográfica determinada, las alternativas de viaje entre el origen y destino

seleccionados y los tiempos de viaje, transbordos, transferencias, costos, etc. de cada una de estas alternativas, con el propósito de que el usuario pueda planificar su viaje antes de que se realice, o pueda modificar su itinerario al enterarse de algún percance que se presente en su ruta cuando el sistema recibe información en tiempo real de cada uno de los modos de transporte involucrados.

El SAIV programado en este trabajo de tesis no recibe información en tiempo real, por lo tanto, sirve para que el usuario pueda tomar mejores decisiones de viaje antes de que este se realice. En un futuro este tipo de sistemas deberán actualizarse en tiempo real y recibir la información que cada modo de transporte genere, como horas de llegada de los vehículos a cierta estación, retrasos en los arribos, percances, nivel de ocupación en los vehículos de transporte público, congestionamiento, etc.

A manera de conclusión en este segundo capítulo, si bien es cierto que nuestro país no es un productor tecnológico y que está rezagado en el uso de nuevas tecnologías aplicadas al transporte (específicamente el uso de los sistemas SIT), existen grandes posibilidades de avance y de innovación, ya que al uso de los SIT en México no se les da la debida importancia.

Si el gobierno destina suficientes recursos y establece la normatividad adecuada que permita regular el comportamiento de los elementos participantes en el sistema de transporte, se podrían establecer las bases de un sistema SIT que opere de forma eficiente en nuestro país, y que privilegie el uso de nuevas tecnologías sobre la construcción de infraestructura.

En este segundo capítulo se abordó lo relacionado con el marco tecnológico en el que se sustenta la tesis, en el siguiente capítulo se revisará la teoría de grafos, redes e hiperredes para conformar el marco teórico.

Capítulo 3 . Hiperrutas viables multimodales

El tercer capítulo se subdivide en seis secciones, en la primera se presenta un resumen con las definiciones de teoría de grafos que se utilizan en la tesis, en la segunda sección se plantean algunas de las estructuras de datos que se utiliza con mayor frecuencia para la representación de las redes, en la tercera sección se analizan las características de viabilidad en rutas multimodales, es decir, la factibilidad de realizar una secuencia de cambios de modo cuando se realiza un viaje, en la cuarta sección se definen los conceptos y características de hipergrafos e hiperrutas, que a su vez sirven para definir el concepto de viabilidad en hiperrutas multimodales tratado en la quinta sección, por último en la sexta sección se analiza el algoritmo del Problema de la hiperruta viable más corta, el cual es utilizado en el sistema programado para esta tesis.

El estudio de los problemas de transporte puede sustentarse en la teoría de redes, para ello es de gran importancia plantear los conceptos básicos de dicha teoría. En primera instancia, una red de transporte puede representarse a través de nodos (estaciones, andenes, paradas) y arcos (vialidades, cambios de modo, transbordos, puentes, caminos peatonales, lugares de ascenso y descenso), que en su conjunto forman un grafo. A continuación se describen los conceptos más importantes de la teoría de grafos, los cuales permitirán modelar las redes e hiperredes del área de estudio (Capítulo 4), e implementar los algoritmos y estructuras de datos (Capítulo 3) que posibiliten la obtención de las hiperrutas, dado un par origen-destino (Capítulo 6).

3.1. Teoría de grafos

Los arcos de un grafo pueden tener o no dirección, es decir, existen grafos dirigidos y grafos no dirigidos. Para poder representar la dirección de las vialidades, será necesaria la inclusión de grafos dirigidos. A continuación se presentan las definiciones de grafo dirigido, camino dirigido, ruta dirigida, ciclo dirigido, grafo dirigido conexo, árbol y bosque (Ahuja et al., 1993).

Grafo Dirigido. Un digrafo o grafo dirigido es una pareja $D = (V, F)$, donde V es un conjunto de puntos llamados nodos, finito y no vacío, y F es un conjunto de parejas

ordenadas de elementos en V , llamados arcos. Al conjunto de nodos y arcos de D se le denota por $V(D)$ y $F(D)$ respectivamente. En la Figura 3.1 se muestra un ejemplo de grafo dirigido.

Debido a las características de las redes de transporte, es común asociar valores o "pesos" a cada uno de los arcos de la red, para representar los costos de cada uno de los arcos, éstos pueden ser, costos monetarios, tiempos, distancias, capacidades, oferta, demanda, etc. El grafo dirigido cuyos nodos y/o arcos tienen asociados dichos valores numéricos, se conoce con el nombre de *Red dirigida*. En la Figura 3.2 se muestra un ejemplo de red dirigida.

Cuando un usuario de transporte público o privado se transporta entre dos puntos de una red, interesa saber de qué forma lo hace. Para ello se definen a continuación los conceptos de camino dirigido, ruta dirigida y ciclo dirigido.

Camino dirigido. Sean u y v nodos de un digrafo D . Un camino dirigido entre u y v , es una sucesión de nodos $\vec{C}_a = (u = u_1, u_2, u_3, \dots, u_n = v)$ tal que $(u_i, u_{i+1}) \in F(D) \forall i \in \{1, 2, \dots, n-1\}$. En un camino dirigido se puede pasar dos veces por un mismo nodo, pero no por un mismo arco, en la Figura 3.3 se muestra un ejemplo de camino dirigido, donde el nodo 3 es visitado dos veces (El orden de los nodos visitados es: 3-1-2-3-6).

Ruta dirigida. Sean u y v nodos de un digrafo D . Una ruta dirigida entre u y v llamada \vec{T} , es un camino dirigido que no repite arcos. En la Figura 3.4 se observa una ruta dirigida, donde no se repiten arcos (ni nodos), como lo establece la definición.

Ciclo dirigido. Un ciclo dirigido \vec{C}_l , es una ruta dirigida donde los únicos nodos que se repiten son el primero y el último. En la Figura 3.5 se muestra un ejemplo de ciclo dirigido. Si el ciclo contiene a todos los nodos, entonces a éste se le conoce como Ciclo Hamiltoniano, en el caso del ejemplo de la Figura 3.5 no se trata de un ciclo Hamiltoniano, ya que hace falta incluir en el ciclo a los nodos 5, 6 y 7.

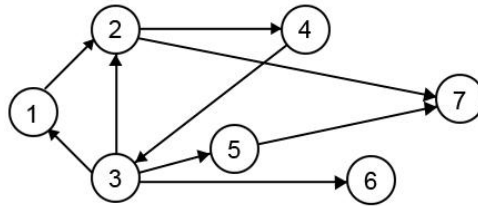


Figura 3.1. Grafo dirigido

Fuente: Propia

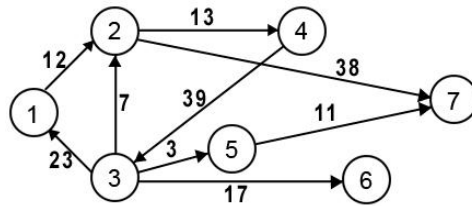


Figura 3.2. Red dirigida

Fuente: Propia

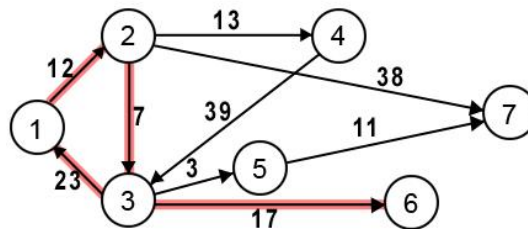


Figura 3.3. Camino dirigido

Fuente: Propia

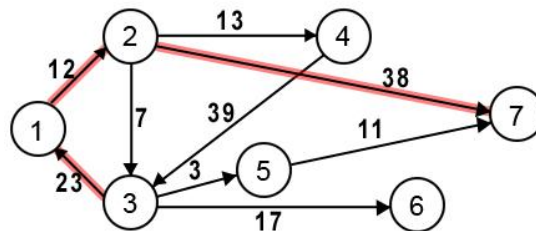


Figura 3.4. Ruta dirigida

Fuente: Propia

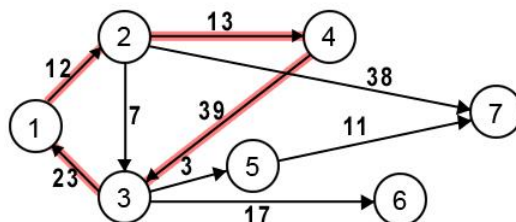


Figura 3.5. Ciclo dirigido

Fuente: Propia

Los nodos pueden estar conectados de distintas maneras, de ellos pueden salir o llegar un número distinto de arcos, por ello, es conveniente definir los conceptos de grafo dirigido conexo y árbol.

Grafo dirigido conexo. Sea D un digrafo.

- 1) D es conexo si $\forall u, v \in V(D)$ existe un camino entre u y v contenido en D .
- 2) D es fuertemente conexo si $\forall u, v \in V(D)$ existe al menos una ruta dirigida de cada uno de los nodos a los demás.

Un grafo es conexo si:

$$n - 1 \leq m \leq n(n - 1).$$

Donde:

- n es el número de nodos y
- m es el número de arcos.

Un par de nodos pueden estar conectados de distintas maneras, una posibilidad es que exista un solo camino entre ambos nodos, otra posibilidad es que exista más de un camino que conecte ambos nodos o inclusive que no exista un camino directo entre pares de nodos. Existe una clasificación para saber el grado de conectividad que presenta un grafo, la cual se muestra a continuación:

- Grafo conexo: al menos $m = n - 1$
- Grafo completo (denso): $m = n(n-1)$
- Grafo esparzo: $2n \leq m \leq 4n$.

Los grafos que representan redes viales se clasifican como grafos esparzos, donde el número de arcos oscila entre el doble y el cuádruple del número de nodos.

Si además de ser conexo, el grafo no tiene ciclos, entonces a este se le conoce como **árbol**, el cual es muy importante en la teoría de grafos, ya que una gran variedad de algoritmos de flujo en la red se aplican a esta estructura. Los árboles tienen las siguientes propiedades:

- Un árbol con n nodos contiene exactamente $n-1$ arcos
- Un árbol tiene al menos dos nodos colgantes (con grado 1)
- Cada par de nodos del árbol están conectados por una única ruta

Bosque. Un grafo sin ciclos es un bosque. De manera alternativa se puede decir que un bosque es una colección de árboles (dos o más árboles), como se muestra en las Figuras 3.6 y 3.7.

Subárbol. Una subgráfica conexa de un árbol es un subárbol.

Árbol con raíz. Un árbol con raíz, es un árbol que se dice es sostenido por un nodo especialmente designado llamado raíz. En un árbol con raíz es posible definir relaciones jerárquicas o de parentesco padre-hijo. El nodo padre del nodo i , es único y corresponde al nodo siguiente en la ruta del nodo i al nodo raíz. En la Figura 3.8 se muestra un árbol cuya raíz es el nodo 1.

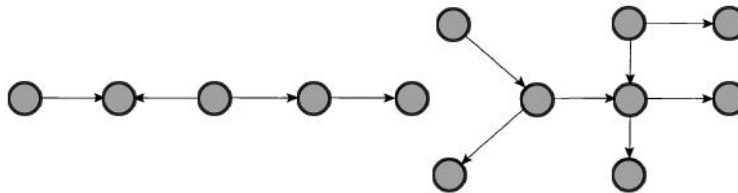


Figura 3.6. Ejemplo de dos árboles

Fuente: (Ahuja et al., 1993)

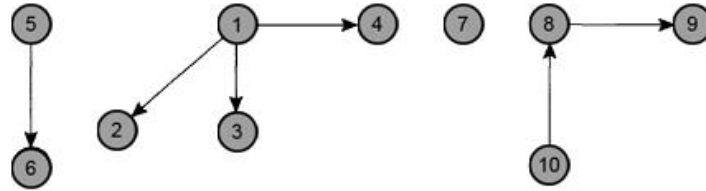


Figura 3.7. Bosque

Fuente: (Ahuja et al., 1993)

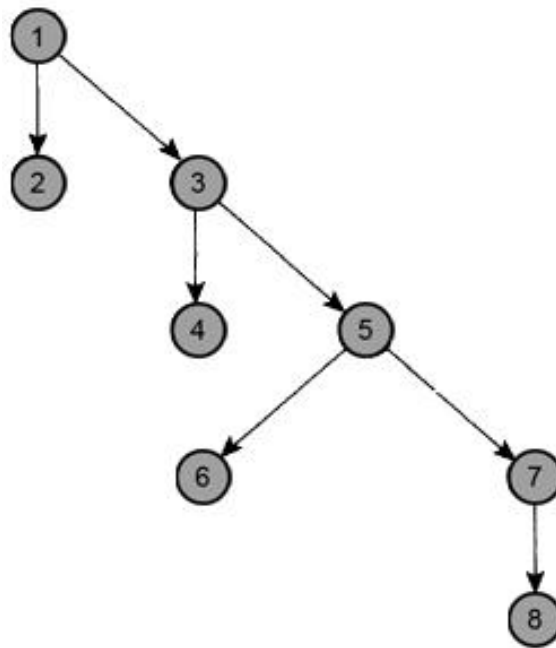


Figura 3.8. Árbol con raíz

Fuente: Elaboración propia

De las relaciones jerárquicas se desprenden los conceptos de predecesor, sucesor, descendientes y ancestros, los cuales se ejemplifican a continuación, haciendo uso de la Figura 3.8.

Predecesor: El nodo 1 es el predecesor de los nodos 2 y 3, y el nodo 5 es el predecesor de los nodos 6 y 7.

Sucesor: Los nodos sucesores del nodo 5 son los nodos 6 y 7. El nodo sucesor del nodo 7 es el nodo 8.

Descendientes: Los descendientes del nodo 5 son los nodos 5, 6, 7 y 8. De la misma figura también se desprende que los descendientes del nodo 3 son los nodos 3, 4, 5, 6, 7 y 8.

Ancestro: Para los nodos 6, 7 y 8, el ancestro es el nodo 5.

3.2. Representación de redes

La forma en la que se representa la topología de una red en la computadora y las estructuras de datos usadas para almacenar la información asociada a la misma, junto con los algoritmos usados, inciden en el desempeño computacional de los algoritmos de redes. Para ello se explican a continuación los conceptos de matriz de incidencia nodo-arco, matriz de adyacencia nodo-nodo, lista de adyacencia, estrella entrante y estrella saliente. Para ello se hará uso de la red de la Figura 3.9 y de los conceptos escritos por Ahuja et al. (1993).

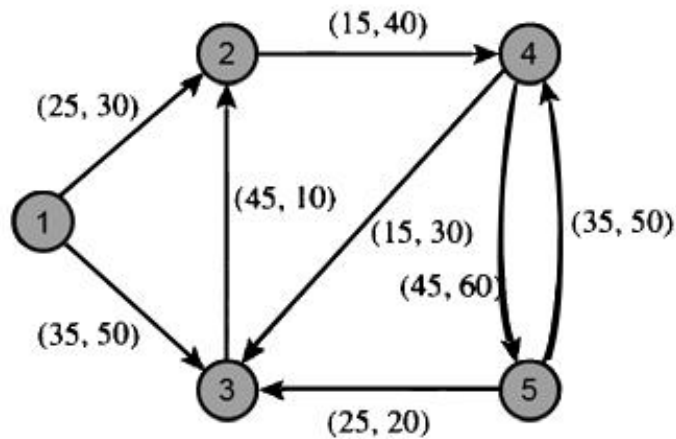


Figura 3.9. Ejemplo de una red

Fuente: (Ahuja et al., 1993)

Matriz de incidencia nodo-arco

Las relaciones entre nodos y arcos pueden representarse de forma matricial a través de la matriz de incidencia los renglones de esta matriz representan los nodos de la red, mientras que las columnas representan a cada uno de los arcos de la red, para el ejemplo

de la red de la Figura 3.9 se tienen 5 nodos y 8 arcos, por lo que se obtendrá una matriz de tamaño 5x8. El valor almacenado en cada posición (Mij) de la matriz será igual a 1 si el arco indicado por la columna "j" sale del nodo indicado por el renglón "i", vale -1 si el arco "j" entra al nodo "i", y 0 cuando no existe relación entre nodos y arcos. La Figura 3.10 representa la matriz de incidencia nodo-arco asociada a la red de la Figura 3.9.

Matriz de adyacencia nodo-nodo

Para establecer las relaciones entre nodos se puede utilizar la matriz de adyacencia nodo-nodo. A diferencia de las matrices de incidencia, en este caso se tendrán únicamente matrices cuadradas de orden "n", donde "n" es el número de nodos de la red. En el caso de la red de la Figura 3.9 se tendrá una matriz cuadrada de orden 5. El valor almacenado (Mij) será igual a 1 si los nodos i y j son conectados de forma directa por un arco, es decir, son adyacentes, si no existe un arco que una un par de nodos el valor almacenado (Mij) será igual a cero. La matriz de adyacencia tendrá n² valores, de los cuales únicamente m (número de arcos) serán diferentes de cero. Este tipo de representación conviene ser utilizada para redes densas, para redes esparzas como las que se utilizan en transporte no conviene este tipo de representación, ya que se desperdicia mucho espacio almacenando muchos ceros en la matriz. La Figura 3.11 representa la matriz de adyacencia nodo-nodo de la red de la Figura 3.9.

$$\begin{array}{c}
 (1,2) \ (1,3) \ (2,4) \ (3,2) \ (4,3) \ (4,5) \ (5,3) \ (5,4) \\
 \begin{array}{l}
 1 \\
 2 \\
 3 \\
 4 \\
 5
 \end{array}
 \left[\begin{array}{cccccccc}
 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\
 0 & -1 & 0 & 1 & -1 & 0 & -1 & 0 \\
 0 & 0 & -1 & 0 & 1 & 1 & 0 & -1 \\
 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1
 \end{array} \right]
 \end{array}$$

Figura 3.10. Matriz de incidencia nodo-arco

Fuente: (Ahuja et al., 1993)

Lista de adyacencia

La lista de arcos de adyacencia $A(i)$ de un nodo i , es el conjunto de arcos salientes del nodo, es decir, $A(i) = \{(i, j) \in A: j \in N\}$, de forma similar, se define a la lista de nodos de adyacencia $A(i)$ de un nodo i como el conjunto de nodos adyacentes al nodo, es decir, $A(i) = \{j \in N: (i, j) \in A\}$.

La representación mediante una lista de adyacencia guarda los nodos de la lista de adyacencia de cada nodo como una lista ligada sencilla. Una lista ligada es un conjunto de celdas conectadas, cada una con uno o más campos. La celda correspondiente al arco (i, j) tendrá todos los campos necesarios para guardar la atributos de ese arco, en este caso, para un nodo "i", se tiene una lista ligada con los campos del nodo adyacente "j", el costo del arco "Cij" y la capacidad del arco "Uij", respectivamente. Si de un nodo sale un solo arco el último campo deberá valer 0, pero si del nodo sale más de un arco, el último campo tendrá el valor de un apuntador hacia otra lista ligada; existirán tantas listas ligadas como arcos salientes del nodo, además, en la última lista ligada, también se coloca el valor 0 para la última celda de la lista. La lista de adyacencia de la red de la Figura 3.9 se muestra en la Figura 3.12.

	1	2	3	4	5
1	0	1	1	0	0
2	0	0	0	1	0
3	0	1	0	0	0
4	0	0	1	0	1
5	0	0	1	1	0

Figura 3.11. Matriz de adyacencia nodo-nodo

Fuente: (Ahuja et al., 1993)

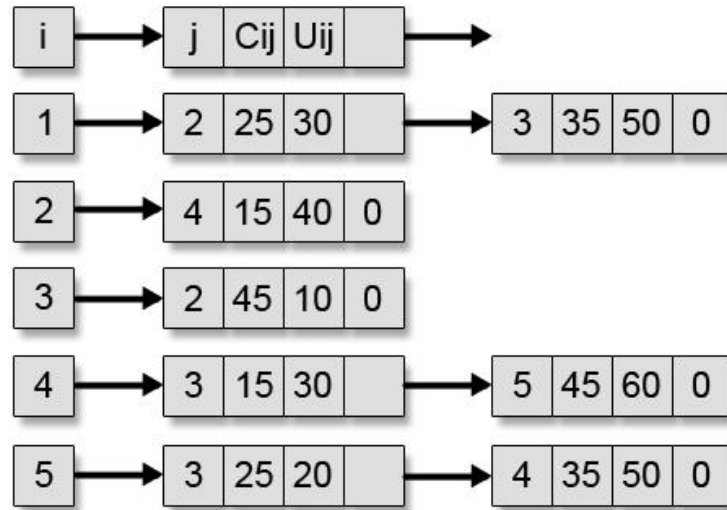


Figura 3.12. Lista de adyacencia

Fuente: (Ahuja et al., 1993)

La representación de una red a través de una estrella saliente es similar a la representación a través de una lista de adyacencia, en cuanto a que solo almacena la lista de adyacencia de nodos para cada nodo de la red, pero en lugar de almacenar los valores en listas ligadas, estos valores son almacenados dentro de un único arreglo.

Estrella saliente

La representación de una red como estrella saliente, guarda en un solo arreglo la lista de nodos de adyacencia de cada nodo de la red. Los arcos se enlistan, asociando a cada uno de ellos un número único. Los arcos se enumeran de la siguiente manera: primero los que salen del nodo 1, posteriormente los que salen del nodo 2 y así sucesivamente hasta llegar al último nodo de la red. En forma secuencial se guarda la información de cada arco en la lista de arcos. En los arreglos cola, cabeza, costo y capacidad, se guardan cada uno de estos atributos de cada arco. También mantiene un apuntador con cada nodo (apuntador (i)), que indica el arco numerado con el número menor en la lista de arcos que salen del nodo i, si este nodo no tiene arcos salientes entonces el apuntador (i) = apuntador (i+1). Por lo tanto la representación de estrella saliente guarda los arcos que salen del nodo i en las posiciones del apuntador (i), hasta el apuntador (i+1) - 1 de la lista

de arcos. Abarcando la totalidad de los nodos, se tiene que el rango para el apuntador es:

$$\text{apuntador (1) = 1} \rightarrow \text{apuntador (n + 1) = m + 1}$$

Es decir su rango va de 1 hasta el número de arcos en la red más uno. La estructura de estrella saliente asociada a la red de la Figura 3.9., se muestra en la Figura 3.13.

Estrella entrante

Para determinar simultáneamente el conjunto de arcos que entran a cada nodo, se utiliza la representación de estrella entrante. Se examinan en orden los nodos de 1 a n, y se guardan en forma secuencial las cabezas, colas, los costos y las capacidades de los arcos que entran al nodo i. También mantiene un apuntador de reversa con cada nodo i (r-apuntador (i)), para indicar la primera posición en esos arreglos que contiene la información sobre un arco entrante al nodo i. Si este nodo no tiene arcos entrantes entonces r-apuntador (i) = r-apuntador (i+1). Los arcos que entran al nodo i, se guardan en las posiciones de r-apuntador (i), hasta r-apuntador (i+1)-1 de la lista de arcos. Abarcando la totalidad de los nodos, se tiene que el rango para el r-apuntador es:

$$\text{r-apuntador (1) = 1} \rightarrow \text{r-apuntador (n + 1) = m + 1}$$

Es decir su rango va de 1 hasta el número de arcos en la red más uno. La estructura de estrella entrante asociada a la red de la Figura 3.9., se muestra en la Figura 3.14.

apuntador		cola	cabeza	costo	capacidad
1	1	1	2	25	30
2	3	1	3	35	50
3	4	2	4	15	40
4	5	3	2	45	10
5	7	4	3	15	30
6	9	4	5	45	60
		7	3	25	20
		8	4	35	50

Figura 3.13. Estrella saliente

Fuente: (Ahuja et al., 1993)

	costo	capacidad	cola	cabeza	r-apuntador
1	45	10	3	2	1
2	25	30	1	2	1
3	35	50	1	3	3
4	15	30	4	3	6
5	25	20	5	3	8
6	35	50	5	4	9
7	15	40	2	4	
8	45	60	4	5	

Figura 3.14. Estrella entrante

Fuente: (Ahuja et al., 1993)

Para los algoritmos de redes es más conveniente el uso de la representación de estrella entrante (o saliente), en lugar de la lista de adyacencia, ya que la primera ocupa menos espacio que la segunda, motivo por el cual será la estrella entrante la representación utilizada para los algoritmos de hiper rutas más cortas, como se verá más adelante.

3.3. Viabilidad en rutas multimodales

Para explicar la viabilidad o factibilidad, al realizar una secuencia de cambios de modo cuando se realiza un viaje, se citarán primero los conceptos de árbol de ruta más corta y red multimodal planteados por Lozano y Storchi (2001).

Árbol de ruta más corta

Sea $G=(N,A)$ un grafo dirigido, donde N y A son respectivamente el conjunto de nodos y arcos, con $|N| = n$ y $|A| = m$. Sea $c: A \rightarrow R$ la función de costo. El problema del árbol de ruta más corta consiste en encontrar un árbol dirigido T con raíz en o , tal que la ruta al nodo i es la ruta más corta desde o hasta $i \forall i \in N$ conectada a o (Lozano y Storchi, 2001).

Red multimodal

En una red multimodal, un nodo es un lugar donde uno puede seleccionar entre continuar en el modo actual o cambiar de modo. El cambio de modo o transferencia modal está

representado por un arco llamado *arco de transferencia*. Un arco que conecta dos nodos con un mismo modo de viaje, se llama *arco de viaje*.

Sea M el conjunto de modos, donde la transferencia modal es considerada un modo. Uno y solo un modo $r \in M$ está asociado con cada arco $(i, j) \in A$. Por lo tanto, existen $|M - 1|$ redes simples, una para cada modo de viaje, conectadas entre ellas mismas por arcos de transferencia. El conjunto de nodos y arcos que conforman cada una de esas $|M - 1|$ redes, están respectivamente denotadas por N_r y A_r , donde r ($r \in M$) representa un modo de viaje. Si r indica una transferencia modal, N_r está vacío y A_r incluye todos los arcos de transferencia modal (Lozano y Storchi, 2001).

Ruta viable multimodal

Para que exista viabilidad en una ruta que incluye dos o más modos de transporte, se deben cumplir con ciertas restricciones en la secuencia de modos utilizados, dada una red. Cada arco del viaje, tiene asociado un costo generalizado, éste no está formado exclusivamente por el costo de sus arcos de viaje, también incluye el costo por cambiar de modo. Cada usuario asigna un costo personalizado a cada cambio de modo, y no está dispuesto a llevar a cabo más de un cierto número de transferencias modales, por lo tanto el usuario prefiere la ruta más corta que no exceda ese número de transferencias modales (Lozano y Storchi, 2001).

Las personas cuando viajan consideran una serie de alternativas, por lo general viajan usando más de un modo de transporte, sin embargo no están dispuestos a realizar muchos cambios de modo.

Cuando se utiliza un modo de transporte que presenta ciertas restricciones (como el automóvil), se debe analizar la viabilidad del uso del modo restringido en el viaje realizado, ya que se pueden presentar rutas 'ilógicas' o infactibles, al combinar los modos de transporte sin restricción con el modo restringido, por ejemplo, si se viaja de inicio en automóvil, luego en autobús y una vez más en automóvil. También ciertas rutas pueden ser viables o no, dependiendo de la ciudad o región geográfica, por ejemplo, en París o Londres, no sería lógico tomar el metro, luego un autobús y nuevamente el metro, ya que la cobertura geográfica de dicho modo, abarca la mayor parte de la ciudad en cuestión, sin embargo en el Distrito Federal esto sí es factible ya que el metro no abarca grandes

zonas de la ciudad y en ocasiones los usuarios de transporte público utilizan el Metro, luego otro modo de transporte público (usualmente microbús o autobús) y nuevamente el Metro.

3.4. Hipergrafos

El transporte público de baja capacidad es el que opera en mayor medida en la Ciudad de México. Este tipo de transporte carece de itinerarios fijos, lo que se conoce es una frecuencia aproximada de las distintas rutas que pasan en una parada X. A través de esta frecuencia, se pueden establecer un conjunto de probabilidades (una por línea) de que un usuario que espera en la parada X pueda abordar una ruta específica.

Por lo tanto, no basta tener un grafo dirigido con costos en cada uno de sus arcos, para poder modelar la espera de un usuario de transporte público, es adecuado utilizar los hipergrafos para este tipo de problema. A continuación se definen los hipergrafos e hiperarcos.

Hipergrafo e hiperarco

Un hipergrafo dirigido o h-grafo es una pareja $H=(V, E)$, en donde $V=(v_1, v_2, \dots, v_n)$ es un conjunto de nodos y $E=(e_1, e_2, \dots, e_m)$ es un conjunto de hiperarcos. Un h-arco $e \in E$ es la pareja $e=(t(e), h(e))$ en donde $t(e) \subset V$ denota el conjunto de los nodos cola y $h(e) \subset V$, denota el conjunto de los nodos cabeza. Es decir el h-arco conecta subconjuntos de V .

Existen distintas maneras de representar a los hipergrafos, un par de ellas se muestran en la Figura 3.15, en la parte derecha, se muestra un grafo conexo, mientras que en la izquierda se muestra un grafo no conexo.

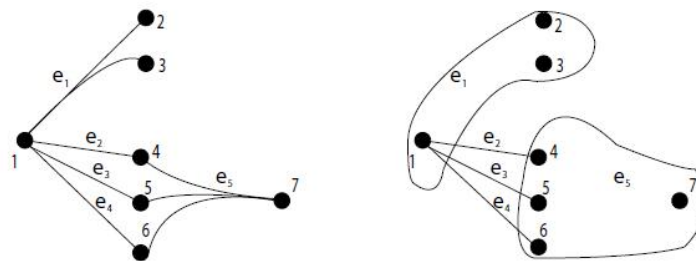


Figura 3.15. Representación de hipergrafos

Fuente: (Ahuja et al. 1993)

Hiperruta

Un grafo parcial definido sobre todas las alternativas (del pasajero), es un conjunto de rutas del origen al destino del viaje. Tal conjunto es llamado hiperruta, de manera formal se tiene la siguiente definición.

Sea H un h-grafo. Una ruta, r_{OD} , que conecta un origen O y un destino D , es una secuencia de nodos y h-arcos, $r_{OD} = (O, t(e_1), e_1, t(e_2), e_2, \dots, e_m, D)$, en donde $t(e_{i+1}) \in h(e_i)$ para $i=1, 2, \dots, m-1$ y $d \in h(e_m)$.

Una hiperruta r_{OD} es el conjunto mínimo sin ciclos de rutas r_{OD} , tal que el destino D , está conectado a cualquier nodo que pertenezca a r_{OD} .

Hipergrafo multimodal

Una hiperred multimodal es la tripleta (N, E, M) , donde N es el conjunto de nodos, E es el conjunto de los h-arcos y M son los modos de transporte asociados a los h-arcos.

Lozano y Storchi (2002) consideran seis modos de transporte, éstos son:

M_1 : Transporte sobre rieles.

M_2 : Transporte público en superficie.

M_3 : Transporte privado con necesidades de estacionamiento (automóviles y motocicletas).

M_4 : Transporte privado sin necesidades de estacionamiento (bicicletas).

M_5 : Transporte a pie (peatonal).

M_6 : Transferencias modales. Por ejemplo, abandonar M_2 y continuar por algún M_i , tal que $i \neq 2$.

El modo M_6 , puede representar distintas acciones, dependiendo de los modos que intervienen en el cambio de modo. El cambio de modo es un proceso que puede incluir las siguientes acciones:

- (i) dejar una línea del modo actual;
- (ii) dejar el modo actual;
- (iii) acercarse a un nuevo modo 'r';
- (iv) esperar en el modo 'r' y
- (v) tomar el modo 'r'.

Un hipergrafo multimodal, está compuesto por tres distintos tipos de hiper arcos, éstos son: hiper arcos de abordaje, arcos de viaje y arcos de transferencia modal (Lozano y Storchi, 2002).

Los mismos autores (Lozano y Storchi, 2002) definen los conceptos de hiper arcos de abordaje, arcos de viaje y arcos de transferencia modal, a través de las acciones asociadas a ellos, estos conceptos se citan a continuación:

Un hiperarco de abordaje está representado por las acciones (iv) y (v) y existe solo para los modos de $M2$ (aquellos modos que pueden ser representados mediante hiper grafos).

Un arco de viaje para el modo 'r', representa la acción de viajar por el modo 'r' o la acción de viajar por una línea del modo 'r' y abandonar tal línea (acción (i)) para toda r que pertenece a la unión de $M1$ y $M2$. La acción de dejar el modo 'r', significa que no se toma una línea de dicho modo de manera inmediata. Esta acción no se incluye en el arco de viaje.

Un arco de transferencia modal puede estar representado por las acciones (ii) y (iii) o por las acciones (ii), (iii) y (v), dependiendo de los modos involucrados en el cambio modal.

Los arcos de viaje y los arcos de transferencia modal tienen asociados un tiempo como costo, a diferencia de los h-arcos de abordaje quienes tienen asociados las frecuencias de las líneas que pueden ser abordadas.

En el transporte público es común que existan diferentes líneas L pertenecientes a algún modo de transporte $M2$, por consiguiente, en cierta estación existe un subconjunto de líneas L que pasan por dicha estación, es decir, $L_i \subset L$, el subconjunto de líneas que pasa por el nodo $i \in N$.

Cuando un usuario arriba a la estación i , existe la posibilidad de que más de una línea de transporte lo lleve a su destino. El conjunto de líneas de transporte que pasan por i y que

llegan al destino seleccionado por el usuario recibe el nombre de *conjunto atractivo*. Es decir, el conjunto atractivo es el conjunto de líneas que van de i a d , tal que en el nodo i el usuario está dispuesto a abordar la primera corrida que arribe a la estación i . Además, por construcción L'_i tiene asociado un h-arco de abordaje $e'_i = (i, h(e'_i))$, donde $h(e'_i)$ es el conjunto de las paradas de cada línea en i (Lozano y Storchi, 2002 citado por López, 2010).

Para definir el tiempo esperado de viaje del conjunto atractivo L'_i , es necesario se precisen los siguientes términos:

- ϕ_j la frecuencia de la línea $l_j \in L_i$, es decir, la frecuencia de alguna línea l_j que pasa por la estación i .
- $\phi(L'_i)$ la frecuencia combinada del conjunto de líneas L'_i . Esto es, la frecuencia combinada del conjunto atractivo.
- $\pi_j(L'_i)$ la probabilidad de que la línea l_j arribe a la estación i antes que cualquier otra línea de L'_i .
- t_j tiempo de viaje esperado entre la parada i y el destino si la línea l_j es utilizada. Este tiempo no incluye el tiempo de espera en i .
- $\varpi(L'_i)$ es el tiempo promedio de espera del conjunto atractivo en la parada i .

Si se supone que el usuario toma el primer autobús de su conjunto atractivo, que los pasajeros arriban a la estación aleatoriamente y que los vehículos de las diferentes líneas llegan a las paradas con una distribución exponencial, esto es, el tiempo de espera promedio es igual al inverso de la frecuencias (Lozano y Storchi 2002), entonces, se cumple que:

$$\Phi(L'_i) = \sum_{l_j \in L'_i} \phi_j$$

$$\varpi(L'_i) = \frac{1}{\Phi(L'_i)}$$

$$\pi_j(L'_i) = \frac{\phi_j}{\Phi(L'_i)}$$

En general, los arcos de una hiperred multimodal tienen asociado un tiempo de viaje $c(i,j)$ y una probabilidad $\pi_j(L'_i) = 1$, ya que no existen otras opciones en la parada i .

Por otro lado, para un h-arco de abordaje, existen diferentes opciones en la parada i y el tiempo de espera depende de la probabilidad de abordar cierto h-arco del conjunto atractivo, esto es, el tiempo de espera es $C(e'_i) = w(e'_i)$ más tantos coeficientes como número de nodos existan en $h(e'_i)$, donde estos coeficientes son $\pi(e'_i, j) \forall j \in h(e'_i)$ y se cumple que:

$$\sum_{j \in h(e'_i)} \pi(e'_i, j) = 1$$

Es decir, la suma de las probabilidades de cada arco del conjunto atractivo dará como resultado 1, ya que el usuario siempre elige una opción para viajar.

El tiempo esperado de viaje $V_p(i)$ al ir del nodo i al destino d , asociado a cada h-ruta p_{id} , está dado por:

$$V_p(i) = \begin{cases} c(i, j) + V_p(j) & \text{si } e = (i, j) \\ w(e'_i) + \sum_{j \in h(e'_i)} \pi(e'_i, j) V_p(j) & \text{si } e' \text{ es un h - arco de abordaje} \end{cases}$$

Cada uno de los términos y expresiones anteriormente citadas, sirven para que una hiper red pueda modelar a un sistema de transporte multimodal.

A continuación se explica la viabilidad para las hiper redes multimodales; los conceptos son similares a los descritos en la sección 3.3, por lo tanto, se expondrán solamente las definiciones y conceptos distintos a los anteriormente expuestos en la sección citada.

3.5. Viabilidad en hiperrutas multimodales

Definición. Una hiperruta multimodal es viable si las rutas que componen a la hiperruta no incluyen más de una sub-ruta de modo- r maximal para cada $r \in M_v$. Es decir, las rutas de una hiper ruta multimodal no usan un modo restringido más de una vez (Lozano y Storchi, 2002).

Un modo- r , con $r \in M_2$, cuenta con las dos siguientes importantes características:

- a) Una red de modo- r está representada por un hiper grafo de modo- r .
- b) Un modo- r no está restringido, si cualquier ruta o hiper ruta puede incluir más de una sub ruta maximal de modo- r .

La concatenación de las rutas P_1 y P_2 , denotado por $(P_1 \circ P_2)$, es una ruta P formada por P_1 y P_2 , tal que el destino de P_1 es el origen de P_2 (Lozano y Storchi, 2002).

La generación de una nueva hiperruta viable a través de la concatenación, puede ser realizada de dos maneras distintas:

1. Sea $e = (i, j)$ un arco de modo- r y sea p_{jd} una hiper ruta viable de j a d . La concatenación de p_{jd} con el arco e es una hiper ruta viable p_{id} si, la concatenación del arco e con cada ruta viable q_{jd} que compone a la hiper ruta p_{jd} genera una ruta viable q_{id} .
2. Sea $e'_i = (i, h(e'_i))$ un hiperarco de abordaje y sea $p_{jd} \forall j \in h(e'_i)$ las hiperrutas viables de los nodos j a d . La concatenación de las hiperrutas viables (p_{jd}) con el hiperarco e'_i es siempre una hiperruta viable p_{id} .

Por lo tanto la viabilidad deberá ser verificada cuando la nueva hiperruta sea construida a través de arcos, cuando se construya a través de hiperarcos siempre se tendrá una concatenación viable.

Para ampliar el análisis de la viabilidad sobre las hiperredes multimodales, es necesario realizar un diagrama de estados sobre las hiperredes, este punto se desarrolla a profundidad en la sección 4.4.

3.6. Algoritmo Shortest Viable Hyperpath Problem

El algoritmo de Hiperrutas mínimas viables (SVHP) encuentra el conjunto de hiperrutas desde un cierto origen hasta el destino especificado, con el tiempo mínimo esperado de viaje y un límite superior de transferencias modales entre 0 y k . El algoritmo guarda el tiempo esperado de viaje, el límite superior de transferencias modales y el estado, para las hiperrutas desde cada nodo visitado hasta el destino. Las hiperrutas viables desde el nodo i hasta el destino d (p_{id}) pueden tener distintos estados asociados. Una pareja nodo-estado $[i,s]$, representa una hiperruta viable p_{id} , para el estado s (Lozano y Storchi, 2002).

El algoritmo consta de un procedimiento principal (Shortest Viable Hyperpath Problem) y tres procedimientos complementarios (*Arc-Concatenation*, *h-Arc-Concatenation* y *Determine-s*).

Las variables asociadas con cada par nodo-estado $[i, s]$ son las siguientes (Lozano y Storchi, 2002):

- $V_{sx}^*(j)$ = tiempo esperado de viaje en la actual hiperruta más corta $[i, s]$.
- $w_s(i)$ = cota superior del número de transferencias modales en la actual hiperruta más corta $[i, s]$.
- $SA_s(i)$ = hiperarco sucesivo del nodo i , en la actual hiperruta más corta $[i, s]$.
- $ST_s(i)$ = conjunto de estados de las hiperrutas más cortas que están actualmente en las cabezas del hiper arco $SA_s(i)$.
- $lastlabel_s(i)$ = tiempo esperado de viaje de la hiperruta más corta $[i, s]$ con cota superior de transferencias modales menor que $w_s(i)$.

Para cada hiperarco que cumpla que $|h(e)| > 1$, es decir, aquellos hiperarcos que representan más de dos rutas diferentes de transporte público, la siguiente información se conserva.

- $h(e)$ = el conjunto de nodos que compone la cabeza de e .
- $\phi(e)$ = la frecuencia de las líneas que pueden ser tomadas a través de e .
- $C^*(e)$ = el tiempo esperado de viaje mínimo que se obtiene utilizando e .

En el procedimiento SVHP se etiqueta una pareja $[i, s]$ cuando el tiempo esperado de viaje de la híper ruta p_{id} para el estado s se mejora utilizando el arco (i, j) . Si el arco (i, j) no es transferencia modal, entonces $[i, s]$ se inserta en Q_{now} , de lo contrario se inserta en Q_{next} .

En cada iteración sólo se examinan los elementos de Q_{now} y se van eliminando hasta que Q_{now} se vacíe; después Q_{now} toma los elementos de Q_{next} hasta que este conjunto esté vacío y la iteración $k+1$ comienza.

Para entender cómo funciona el algoritmo SVHP (Figura 3.16), a continuación se detallan cada una de las instrucciones de la función principal; además se realiza el mismo análisis para el resto de las funciones del algoritmo SVHP.

La línea 2, **for each j in N do $V_{sx}^*(j) = \infty \forall s$** , indica que todos los nodos deben ser etiquetados con valor infinito en su respectivo campo $V_{sx}^*(j)$ (tiempo mínimo de viaje para el nodo j con estado S_x).

Posteriormente el ciclo comprendido entre las líneas 3 y 5,

for each e in E do

$\{ C^*(e) = \infty; h^*(e) = \emptyset; \phi^*(e) = \mathbf{0} \}$ establece las condiciones iniciales de cada híper arco, donde el tiempo de viaje óptimo inicial es infinito, el conjunto atractivo de cabezas del híper arco es el conjunto vacío y la frecuencia combinada óptima es cero.

La instrucción de la línea 6, **$Q_{now} = \{[d, \mathbf{0}]\}; V_0^*(d) = \mathbf{0}$** , establece las condiciones iniciales para el destino, indica que este nodo debe ser agregado a la lista Q_{now} con un estado cuyo valor es cero y que el tiempo de viaje en el destino es igual a cero.

El ciclo comprendido entre las líneas 7 y 24:

repeat {

...

} until $h > k$ or $Q_{now} = \emptyset$ indica que las instrucciones dentro del ciclo deben repetirse hasta que el contador de transferencias del algoritmo (h) exceda al número máximo de transferencias establecidas por el usuario (k) o cuando la lista Q_{now} se vacíe, es decir, cuando ya no existan más nodos dentro de la lista para ser analizados.

Mientras que la lista Q_{now} no se haya vaciado (ciclo que va de la línea 8 a la línea 22, **while** $Q_{now} \neq \emptyset$), se deben realizar las siguientes acciones:

Seleccionar de Q_{now} al primer par nodo-estado que aparezca en la lista, **select** $[j, s_x]$ from Q_{now} (línea 9).

El nodo seleccionado se quita de la lista $Q_{now} = Q_{now} / \{[j, s_x]\}$, (línea 10).

El resto del algoritmo es ejecutado solo si el tiempo mínimo de viaje para el nodo j ($V_{sx}^*(j)$), es menor que el mejor de los tiempos almacenados para dicho nodo ($lastlabel_{sx}(j)$), si se cumple la condición (línea 11) se actualiza el valor del menor tiempo almacenado con el valor del tiempo mínimo de viaje para el nodo j , es decir:

$$lastlabel_{sx}(j) = V_{sx}^*(j) \text{ (Línea 12)}$$

Para cada arco de la estrella entrante, **for each** $e \in B(j)$ **do** (línea 13), guarda el valor del nodo cola del hiper arco en la variable "i"; si la cardinalidad del hiper arco es igual a 1, es decir si solo tiene una cabeza, se trata de un arco (**if** $(|h(e)| = 1)$, línea 15) por lo tanto debe llamarse al procedimiento Arc-Concatenation (Concatena Arco, línea 16); si la cardinalidad es mayor a uno, se tienen dos o más cabeza conectadas al nodo cola, por lo tanto se trata de un hiper arco y debe llamarse al procedimiento h-Arc-Concatenation (Concatena Hiper Arco, línea 18).

Al finalizar un ciclo del procedimiento principal se deben copiar en la lista Q_{now} , los nodos almacenados en la lista Q_{next} , vaciar posteriormente la lista Q_{next} e incrementar el contador de transferencias en una unidad, esto es:

$$Q_{now} = Q_{next}; Q_{next} = \emptyset; h = h + 1 \text{ (Línea 23)}$$

El procedimiento *Arc-Concatenation* (Figura 3.17) analiza los arcos $e = (i, j)$ de la hiper red y determina si existe una nueva hiper ruta viable p con *estado-s* y que no supere el límite de transferencias modales. Para comprobar si es posible crear p con *estado-s* se compara con la hiper ruta actual y con las demás hiper rutas con *estados-sy*, en donde sy es el

conjunto de estados preferidos a s . Si p mejora en costo a las otras, entonces se concatena (i, j) con p , en caso contrario, no se genera la híper ruta (López, 2010).

El procedimiento *h-Arc-Concatenation* (Figura 3.18) examina aquellos híper arcos e , tales que $|h(e)| > 1$, en donde la híper ruta p_{jd} que se concatena cumple que $j \in h(e)$. En el procedimiento primero se determina el estado- s de la híper ruta usando el sub procedimiento *Determine-s* (Figura 3.19), después se actualizan las etiquetas de los nodos cabeza de e , de la frecuencia combinada de e , el conjunto de hiperrutas viables por e y el tiempo de viaje esperado mínimo. Dicha actualización se efectúa siempre y cuando $V_{sx}^*(j) < V_s^*(i)$, es decir, siempre que el tiempo esperado de viaje de la híper ruta p_{jd} mejore el tiempo esperado de viaje de la híper ruta p_{id} , en caso contrario, la concatenación no se realiza y por lo tanto tampoco se actualizan las etiquetas (López, 2010).

Para conocer el estado asociado a un nodo en particular, se utilizan los procedimientos State-M₁ (S_x, S), State-M₂ (S_x, S), State-M₃ (S_x, S), y State-M₄ (S_x, S), los cuales determinan el estado final en un nodo, a partir del estado del nodo predecesor y del modo asociado al arco que conecta al predecesor con el nodo actual. Dichos procedimientos se basan en el diagrama de estados que se muestra en la sección 4.4 y que está ligado con la viabilidad de los híper arcos a concatenar. El pseudocódigo asociado a cada uno de los estados es el que se muestra a continuación:

```

Procedimiento State-M1 ( $S_x, S$ ) {
    If ( $S_x=0$  or  $S_x=1$ ) then  $S=1$ 
    If ( $S_x=4$ ) then  $S=4$ 
}

```

```

Procedimiento State-M2 ( $S_x, S$ ) {
    If ( $S_x=0$ ) then  $S=2$ 
    If ( $S_x=1$ ) then  $S=1$ 
    If ( $S_x=4$ ) then  $S=4$ 
}

```

```

Procedimiento State-M3 (Sx, S) {
    If (Sx=0 or Sx=3) then S=3
    If (Sx=1 or Sx=5) then S=5
}

```

```

Procedimiento State-M4 (Sx, S) {
    If (Sx=1 or Sx=2) then S=1
    If (Sx=3 or Sx=4) then S=4
}

```

El procedimiento *Determine-s* (Figura 3.19), obtiene el estado resultante para las transiciones entre la hiper ruta actual y la nueva hiper ruta. Cuando dos hiperrutas convergen en el mismo nodo, este procedimiento se encarga de determinar cuál será el estado final. La revisión de la viabilidad para los modos de transporte utilizados puede ser simplificada de la siguiente manera:

If [(S_x=4 and State-M ≠ M₃) or (S_x=5 and State-M=M₃)] then "Concatenación viable"

Esta simplificación está basada en el diagrama de estados que se muestra en la sección 4.4 y se puede realizar ya que el único modo de transporte restringido es el automóvil, y los modos de transporte público que existen en el área de estudio, no presentan restricción alguna.

El procedimiento principal del algoritmo SVHP, se presenta en la Figura 3.16.

```

Procedure SHORTEST-VIABLE-HYERPATH-PROBLEM
1 begin
2   foreach  $j \in N$  do  $V_{s_x}^*(j) = \infty \forall s$ ;
3   foreach  $e \in E$  do
4     |  $C^*(e) = \infty; h^*(e) = \phi; Phi^*(e) = 0$ ;
5   end
6    $Q_{now} = \{[d, 0]\}; V_0^*(d) = 0$ ;
7   repeat
8     while  $Q_{now} \neq \phi$  do
9       select  $[j, s_x]$  from  $Q_{now}$ ;
10       $Q_{now} = Q_{now} / \{[j, s_x]\}$ ;
11      if ( $V_{s_x}^* < lastlabel_{s_x}(j)$ ) then
12        |  $lastlabel_{s_x}(j) = V_{s_x}^*$ 
13        | foreach  $e \in B(j)$  do
14          |  $i = t(e)$ 
15          | if ( $|h(e)| = 1$ ) then
16            | call Procedure Arc-Concatenation( $e, s_x, V^*, c, w$ );
17          | else
18            | call Procedure
19            | h-Arc-Concatenation( $e, s_x, V^*, C^*, w, h^*, \phi$ );
20          | end
21        | end
22      end
23       $Q_{now} = Q_{next}; Q_{next} = \phi; h = h + 1$ ;
24    until  $h > k$  or  $Q = \phi$ ;
25 end

```

Figura 3.16. Procedimiento Shortest Viable Hyperpath Problem

Fuente: (Lozano y Storchi, 2002)

```

Procedure Arc-Concatenation( $e, s_x, V^*, e, w$ )
1  $wt = 0; s = 0; con = 0;$ 
2 switch ( $mode_e$ ) do
3   case ( $\in M1 :$ )
4     if  $s_x \neq 5, 12, 14, 20$  then call  $State-M1(s_x, s)$ 
5     break
6   case ( $\in M3 :$ )
7     if  $s_x \neq 8, 14, 18, 20$  then call  $State-M3(s_x, s)$ 
8     break
9   case ( $\in M4 :$ )
10    if  $s_x \neq 11, 16, 18, 20$  then call  $State-M4(s_x, s)$ 
11    break
12  case ( $\in M6 :$ )
13    if  $w_{s_x}(j) < k$  and  $s_x \neq 0$  then  $wt = 1;$  call  $State-M6(s_x, s)$ 
14    break
15  otherwise
16    if  $s_x = 0$  then
17       $s = 1$ 
18    else
19       $s = s_x$ 
20    end
21 endsw
22 if ( $s \neq 0$  y  $V_{s_x}^*(j) + c(e) < V_s^*(i)$ ) then call  $State-M6(s_x, s)$ 
23 else  $SM = \phi$ 
24 while ( $EOF(SM) = 0$  and  $con \neq 1$ ) do
25   select  $s_y$  from  $SM$ 
26   if ( $V_{s_x}^*(j) + c(e) < V_{s_y}^*(i)$ ) then
27      $V_s^*(i) = V_{s_x}^*(j) + c(e)$ 
28      $w_s(i) = w_{s_x}(j) + wt$ 
29      $SA_s(i) = e$ 
30      $ST_s(i) = s_x$ 
31     if ( $wt = 0$  and  $[i, s] \notin Q_{now}$ ) then  $Q_{now} = Q_{now} \cup \{[i, s]\}$ 
32     if ( $wt = 1$  and  $[i, s] \notin Q_{next}$ ) then  $Q_{next} = Q_{next} \cup \{[i, s]\}$ 
33      $con = 1$ 
34   end
35 end

```

Figura 3.17. Procedimiento Arc Concatenation

Fuente: (Lozano y Storchi, 2002)

```

Procedure h-Arc-Concatenation( $e, s_x, V^*, C^*, w, h^*, \varphi$ )
1  $con = 0$ ;
2 if ( $\Phi^*(e) \neq 0$ ) then call Determine-s( $s, s_x, s_z(e)$ )
3 else  $s = s_x$ 
4 if ( $V_{s_x}^*(j) < V_s^*(i)$ ) then
5    $\Phi^*(e) = \Phi^*(e) + \varphi_j$ 
6    $h^*(e) = h^*(e) \cup \{j\}$ 
7    $state(e) = state(e) \cup \{s_x\}$ 
8    $s_z(e) = s$ 
9   if ( $\Phi^*(e) = \varphi_j$ ) then  $C^*(e) = 1/\varphi_j + V_{s_x}^*(j)$ 
10  else  $C^*(e) = C^*(e) - (C^*(e) - V_{s_x}^*(j))\varphi_j/\Phi^*(e)$ 
11  if ( $C^*(e) < V_s^*(i)$ ) then call states( $s, PS$ )
12  else  $PS = \phi$ 
13  while ( $EOF(SM) = 0$  and  $con \neq 1$ ) do
14    select  $s_y$  from  $SM$ 
15    if ( $C^*(e) < V_{s_y}^*(i)$ ) then
16       $V_s^*(i) = C^*(e)$ 
17       $w_s(i) = w_{s_x}(j)$ 
18       $SA_s(i) = e$ 
19       $ST_s(i) = state(e)$ 
20      if ( $[i, s] \notin Q_{now}$ ) then  $Q_{now} = Q_{now} \cup \{[i, s]\}$ 
21       $con = 1$ 
22    end
23  end
24 end

```

Figura 3.18. Procedimiento H-Arc Concatenation

Fuente: (Lozano y Storchi, 2002)

```

Procedure Determine-s( $s, s_x, s_z(e)$ )
1  $sum = s_z(e) + s_x$ 
2 if ( $s_z(e) = 1$ ) then  $s = s_x$ 
3 else if ( $s_x = 1$  or ( $s_z(e) = s$ )) then  $s = s_z(e)$ 
4 else if ( $sum = 19$ ) then
5   if ( $s_z(e) = 14$  or  $s_x = 14$ ) then  $s = 14$ 
6   else  $s = 18$ 
7 end
8 else if ( $sum = 26$  or  $29$ ) then  $s = 18$ 
9 else if  $sum = 16, 21$  or  $27$  then  $s = 16$ 
10 else if  $sum = 16, 21$  or  $27$  then  $s = 16$ 
11 else if  $sum = 13$  or  $22$  then  $s = 14$ 
12 else  $s = 20$ 

```

Figura 3.19. Procedimiento Determine-S

Fuente: (Lozano y Storchi, 2002)

Capítulo 4 . Hiperred del área de estudio

El cuarto capítulo está conformado por tres secciones, en las dos primeras se analiza la estructura general de las hiperredes de transporte público y de la red de transporte privado, así como su digitalización y conversión en un archivo DBD de TransCad para su futura exportación a PostgreSQL. En la tercera sección se plantea el diagrama de estados que se utiliza para verificar la viabilidad en una hiperruta.

Para que el algoritmo SVHP pueda funcionar es necesario que se tengan modeladas las hiperredes que representan la estructura de los sistemas de transporte público que ofrecen su servicio en el área de estudio (Metro, Metrobús, Trolebús, Tren Ligero y RTP), y también que se tenga modelada la red de vialidades por las que circulan los automóviles.

La mayoría de los problemas de hiperredes mínimas multimodales se han abordado sin incluir al automóvil, trabajos de tesis como “Hiperrutas más cortas con respecto del tiempo entre las instalaciones dentro de Ciudad Universitaria: Un modelo de transporte multimodal” (López, 2010), incluye la bicicleta como único modo restringido de transporte y “Modelación de hiperrutas mínimas multimodales del transporte público de pasajeros: Zona centro sur del DF” (Soto, 2013), incluye únicamente transporte público. El incluir el automóvil en este trabajo presenta ciertas ventajas, las personas que utilicen el SAIV podrán obtener hiperrutas que combinen el transporte público con el privado, en lugar de que una persona utilice el automóvil desde el origen hasta el destino se propone utilizarlo hasta llegar a un estacionamiento que esté cerca de una estación de algún modo de transporte público, para continuar el viaje en alguno de los modos de transporte anteriormente mencionados, con ello, se reducirán los tiempos de traslado, el costo de viaje, las emisiones contaminantes y el estrés del conductor. Sin embargo, existen algunos factores que impiden que esto se lleve a cabo, por ejemplo, los estacionamientos en la zona de estudio son insuficientes, los existentes son de baja capacidad, muchos se ubican lejos de alguna estación de transporte público, en ocasiones el personal que labora en los estacionamientos es deshonesto, las cuotas son elevadas, etc.

De poder mejorarse esto, y con el uso de un sistema SAIV como el que se desarrolla en esta tesis, los viajeros podrán tomar mejores decisiones sobre su viaje, haciendo uso de manera combinada del automóvil con el transporte público y gozando de beneficios económicos, de reducción en los tiempos de traslado, etc.

A continuación, se detallarán cada una de las redes e hiperredes que componen al área de estudio.

4.1. Hiperred de transporte público

En el presente trabajo de tesis se utilizaron las hiperredes de transporte público modeladas por Soto (2013). El proceso de construcción de dichas hiperredes es similar al proceso de construcción descrito en el punto 4.2 para la red de vialidades asociadas al automóvil, por lo que no se ahondará en la explicación técnica para la elaboración de las hiperredes, pero si se detallarán las características de cada una de ellas para saber cómo están conformadas (líneas, paradas, modos, etc.).

El transporte público que circula en el área de estudio comprende al Sistema de Transporte Colectivo Metro, RTP, Tren Ligero, Trolebús y Metrobús (Figura 4.5). Aunque también circulan combis y microbuses, éstos fueron excluidos del análisis por la falta de información que presentan dichos modos de transporte.

4.1.1. Hiperred del Metro

La hiperred del Metro está conformada por la Línea 2 (estaciones: Tasqueña a Chabacano), Línea 3 (Estaciones: Universidad a Centro médico), Línea 4 (estaciones: Santa Anita y Jamaica), Línea 7 (estaciones: Barranca del Muerto a Tacubaya), Línea 8 (estaciones: Constitución de 1917 a Chabacano), Línea 9 (estaciones: Tacubaya a Puebla) y Línea A (estaciones: Agrícola Oriental a Tepalcates). En la Figura 4.1, se muestra la hiperred hecha en TransCad (formato shp).

4.1.2. Hiperred del Metrobús

La hiperred del Metrobús está conformada por la Línea 1 (estaciones: PeriSur a Campeche), Línea 2 (todas las estaciones) y Línea 3 (estaciones: Etiopía a Dr. Márquez), como se muestra en la Figura 4.2.

4.1.3. Hiperred de autobuses RTP

La hiperred asociada a la Red de Transporte de Pasajeros del Distrito Federal (RTP), se hizo con base únicamente en el servicio expreso, las líneas que se consideraron son: la Línea 57-A Metro Cuatro Caminos-Metro Constitución de 1917 (estaciones: Picacho a Tren Ligero), la Línea 39 Metro San Lázaro-Carmen Serdán, la Línea 39-A Metro San Lázaro-Xochimilco, la Línea 47-A Alameda Oriente-Xochimilco y la Línea del Circuito Bicentenario. Dicha información se muestra en la Figura 4.3.

4.1.4. Hiperred del Trolebús y Tren Ligero

Para el Tren Ligero se consideraron las estaciones Tasqueña a Periférico, el Trolebús está conformado por la Línea Corredor Cero Emisiones (estaciones: Terminal de autobuses del sur a Antonio Solís), Línea K (estaciones: Ciudad Universitaria a Av. Miguel Hidalgo), Línea M (estaciones: Villa de Cortés a Canal de Tezontle) y Línea Q (estaciones: Puente de Títla a Javier Rojo Gómez intersección con Av. 4 Sur), lo anteriormente descrito se observa en la Figura 4.4.

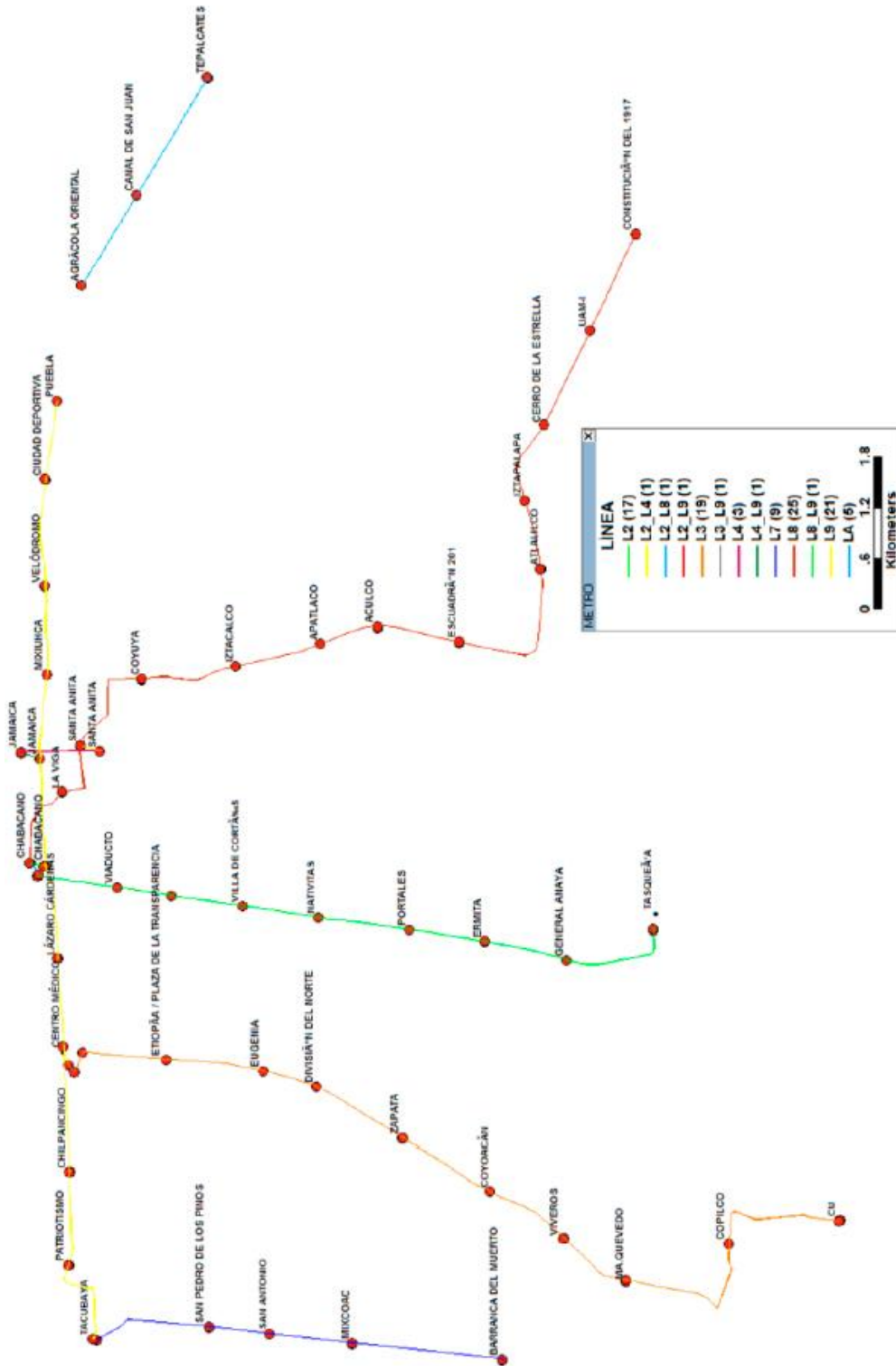


Figura 4.1. Hiperred del Metro

Fuente: (Soto, 2013)

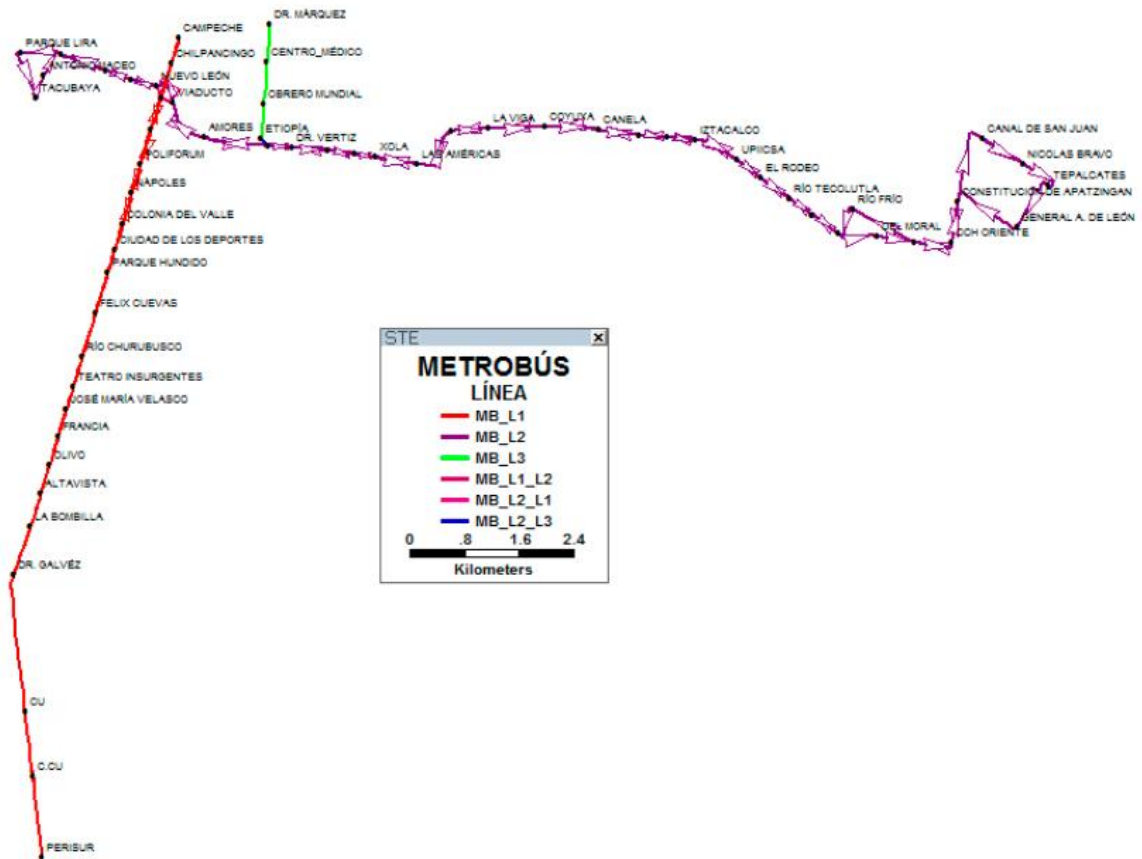


Figura 4.2. Hiperred del Metrobús

Fuente: (Soto, 2013)

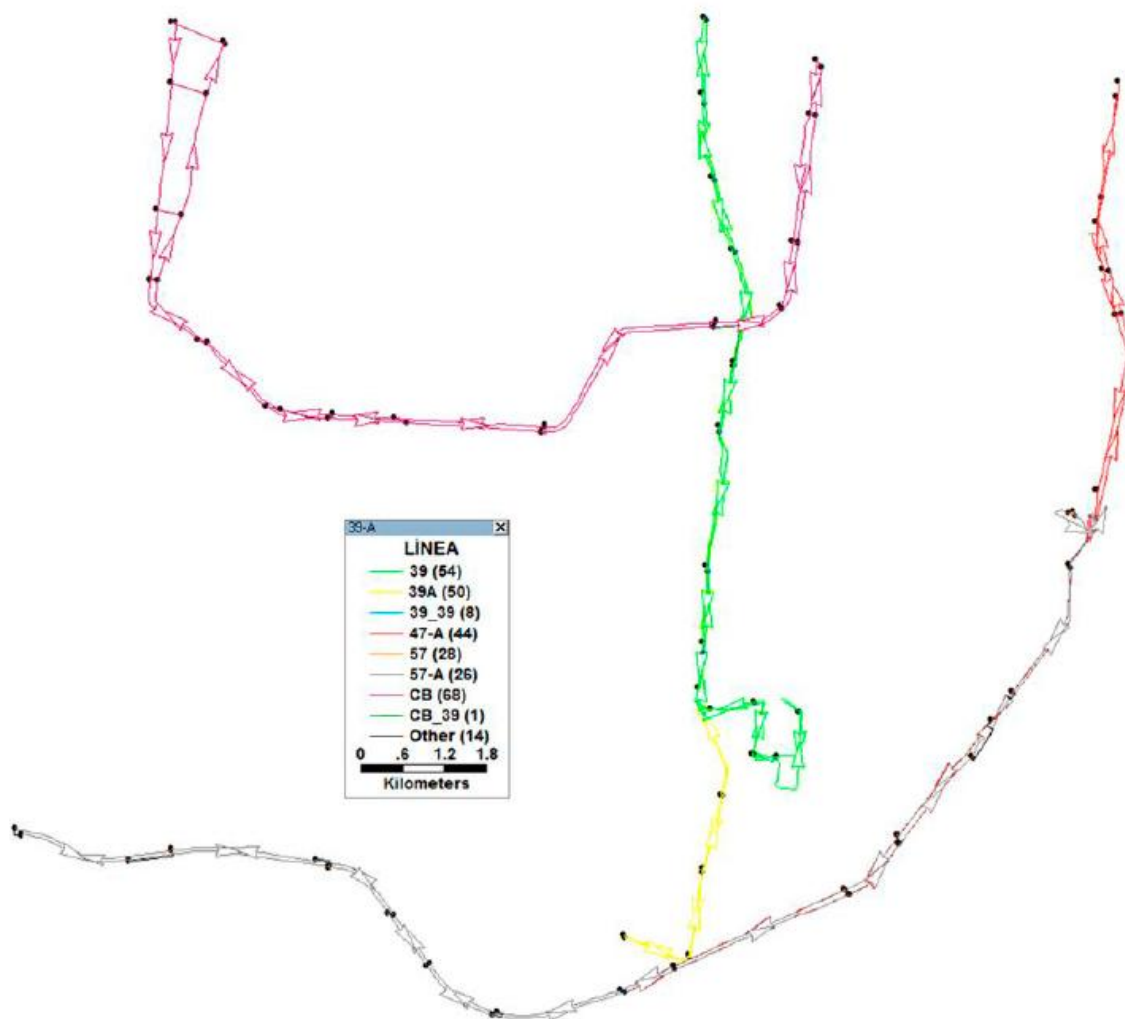


Figura 4.3. Hiperred de autobuses RTP

Fuente: (Soto, 2013)

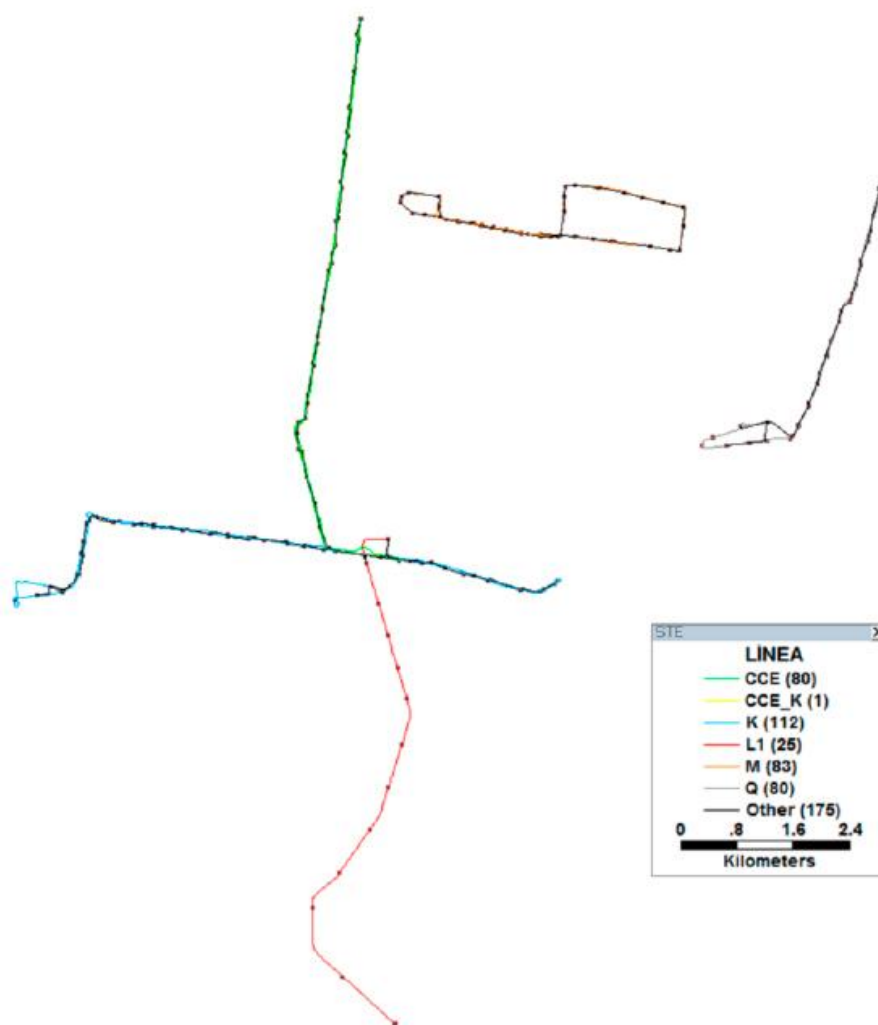


Figura 4.4. Hiperredes del Tren Ligero y Trolebús

Fuente: (Soto, 2013)

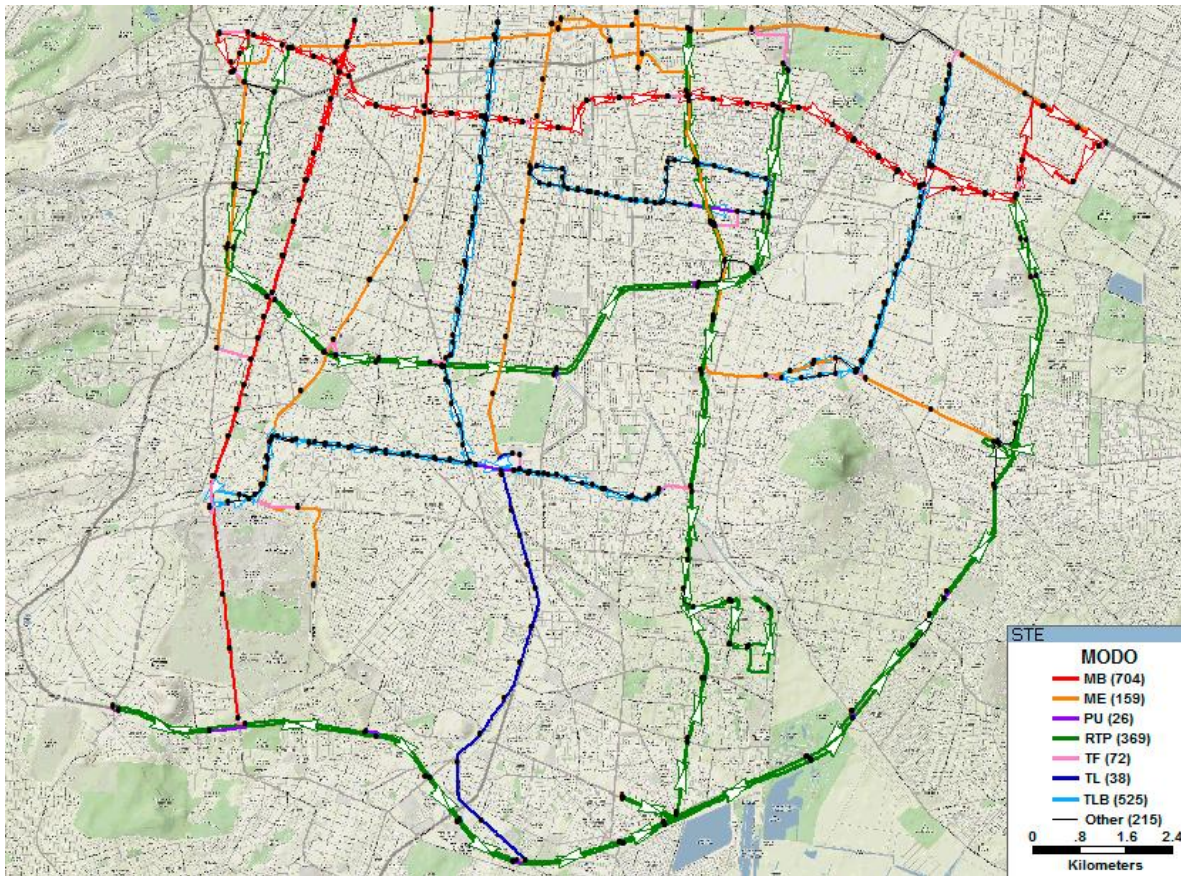


Figura 4.5. Hiperred de transporte público para el área de estudio

Fuente: (Soto, 2013)

4.2. Red de transporte privado

La red de transporte privado, incluye algunas vialidades principales y estacionamientos públicos de las cuatro delegaciones que conforman el área de estudio.

Con el fin de mostrar el potencial del SAIV desarrollado, y por el alcance de la investigación, sólo fueron consideradas algunas vialidades para el automóvil, ya que el objetivo de la tesis es demostrar que se puede combinar un modo restringido con los modos de transporte público de la zona de estudio y aplicar el algoritmo SVHP analizado en la sección 3.6. En la Figura 4.7, se muestran las vialidades y estacionamientos utilizados en el SAIV, los cuales fueron digitalizadas con los programas Google Earth y TransCad.

Queda pendiente la integración de toda la red vial y de estacionamientos presentada en la Figura 4.6.

Las vialidades consideradas fueron: Periférico Sur en el tramo comprendido entre Viaducto Tlalpan y PeriSur, Insurgentes Sur entre Periférico y la Zona Cultural de Ciudad Universitaria, el Circuito Mario de la Cueva de la Zona Cultural hasta el Metro CU, Eje 10 Sur en el tramo comprendido entre Av. Aztecas y Antonio Delfín Madrigal, la avenida Delfín Madrigal del Eje 10 Sur hasta el Centro Nacional de Prevención de Desastres, la Calzada México-Xochimilco, desde su intersección con Periférico Sur hasta el entronque con Calzada de Tlalpan, Calzada de Tlalpan desde Huipulco hasta Tasqueña, Calzada Tasqueña y su continuación Miguel Ángel de Quevedo desde el Eje 2 Oriente hasta Avenida Universidad y Circuito interior desde Canal de Tezontle hasta el Eje 7 Sur Municipio Libre.



Figura 4.6. Vialidades principales y estacionamientos del área de estudio

Fuente: Propia

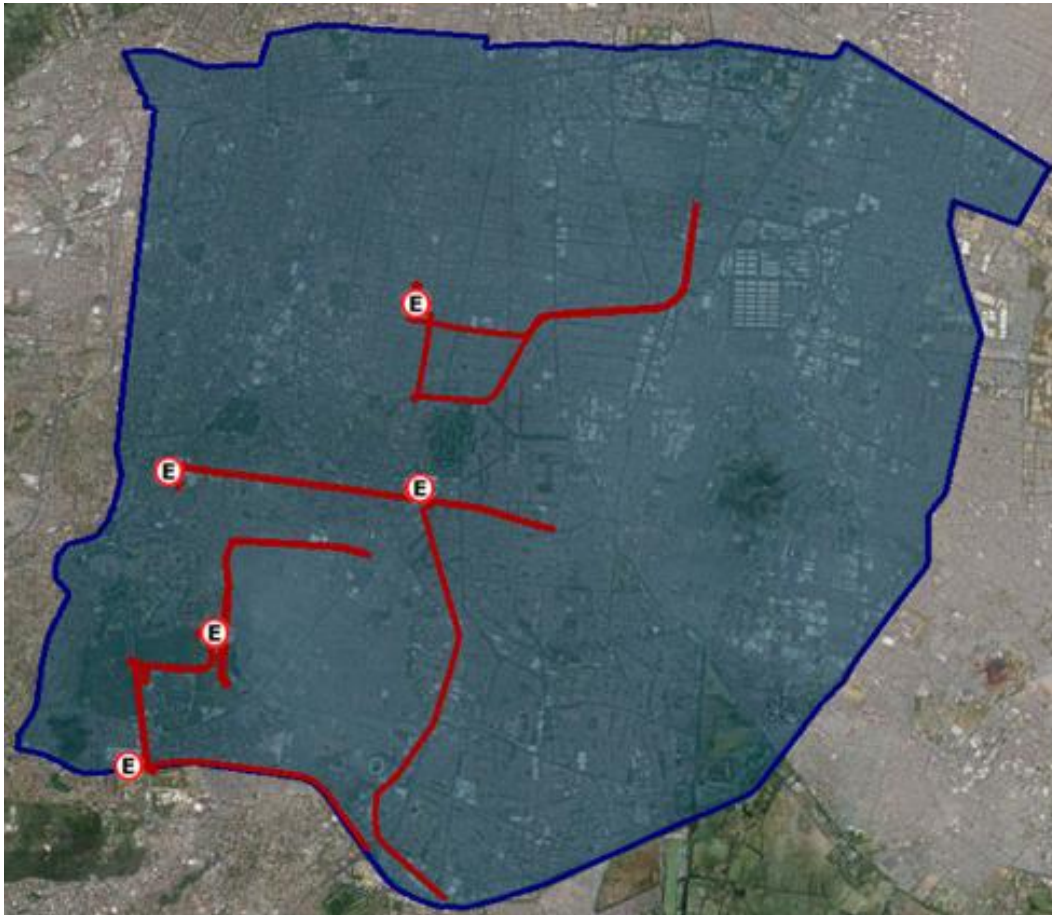


Figura 4.7. Vialidades y estacionamientos considerados

Fuente: Propia

Digitalización de la red de transporte privado

El software utilizado para la captura y digitalización de la red de transporte privado fue Google Earth, debido a que es un software gratuito y de fácil utilización. Para el transporte privado se capturaron la ubicación de los estacionamientos del área de estudio y las vialidades descritas al inicio del tema 4.2. El orden de digitalización entre estacionamientos y vialidades es irrelevante, en primera instancia se explicará la digitalización de los estacionamientos y posteriormente la digitalización de las vialidades.

Para indicar la ubicación de los estacionamientos se presiona el botón *Agregar marca de posición*, como lo muestra la Figura 4.8, en la ventana *Nueva Marca de Posición* (Figura

4.9) se introduce el nombre del estacionamiento que se va a agregar, si se quiere cambiar la imagen de la marca se debe seleccionar el ícono ubicado en la esquina superior derecha de la misma ventana (Figura 4.9) y seleccionar la ubicación de la imagen que se quiere asignar (Figura 4.10). Para agregar el resto de los estacionamientos, se repite el proceso anteriormente descrito.



Figura 4.8. Añadir Nueva Marca de posición

Fuente: Propia

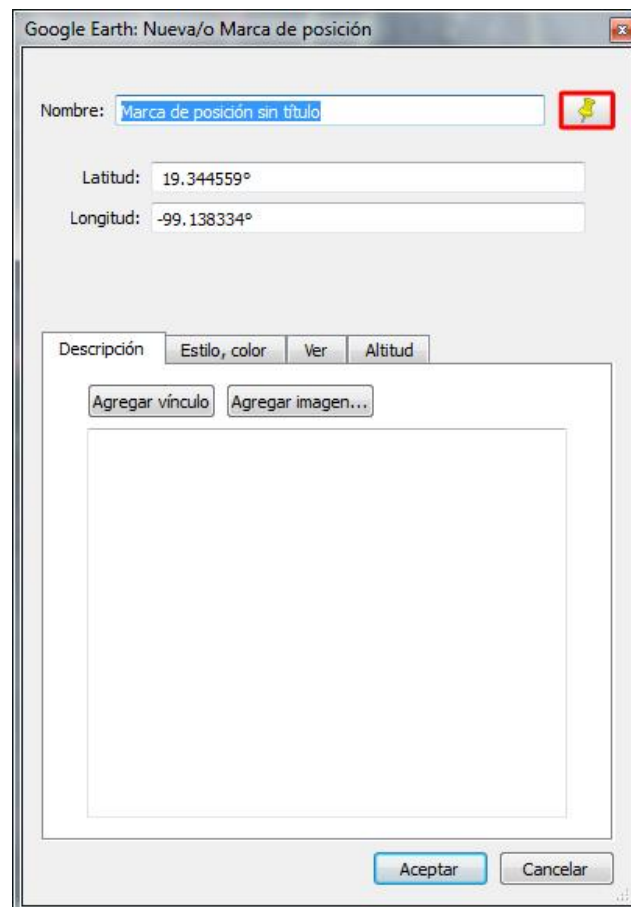


Figura 4.9. Ventana Nueva Marca de posición

Fuente: Propia

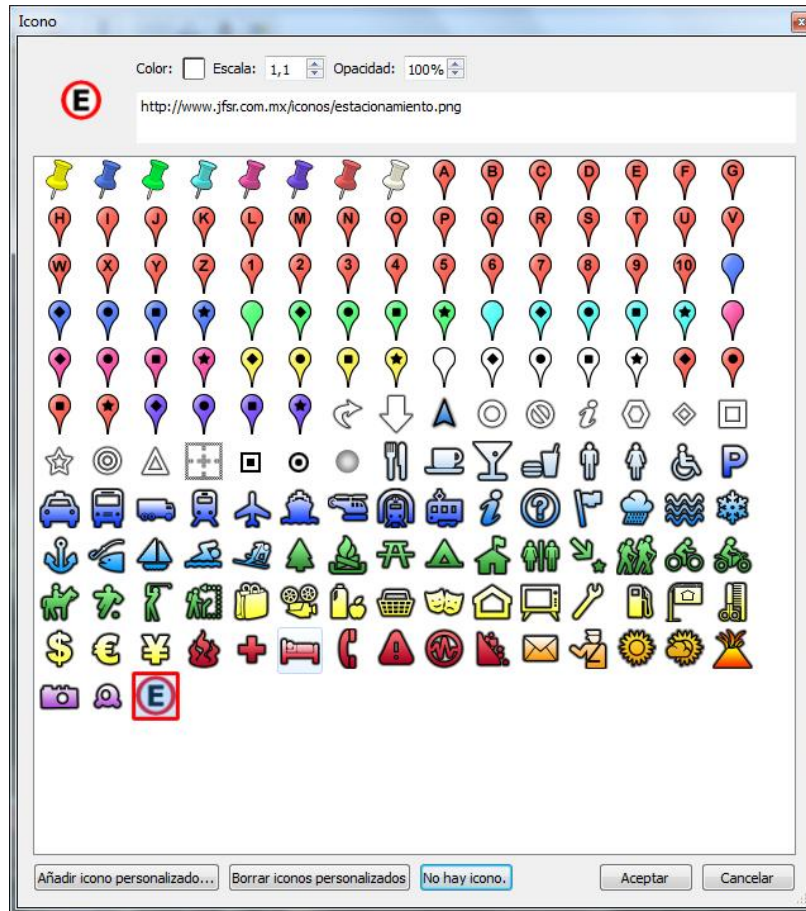


Figura 4.10. Ventana Ícono

Fuente: Propia

Para agregar las vialidades del transporte privado, se selecciona el botón *Agregar ruta*, (Figura 4.11) y posteriormente se dibuja la ruta a través de segmentos de recta, dando clic de manera reiterada a lo largo de la vialidad que se va a trazar y tomando como referencia el mapa de Google Earth.

Cuando se terminan de agregar todas las rutas, se guardan por separado las capas de estacionamientos y vialidades. Para guardar la información capturada, se selecciona la carpeta que contiene los datos (por ejemplo la carpeta “Estacionamientos”) y posteriormente se presionan las teclas Ctrl+S, esto abre la ventana *Guardar archivo...* (Figura 4.12) donde se asigna el nombre del archivo y en el campo tipo, se selecciona KML (Keyhole Markup Language).

El siguiente paso es transformar los archivos KML a SHP, ya que la información digitalizada en Google Earth, será procesada en TransCad para unir la red de transporte privado con las hiperredes de transporte público, y TransCad no reconoce los archivos con formato KML.



Figura 4.11. Ventana Agregar ruta

Fuente: Propia

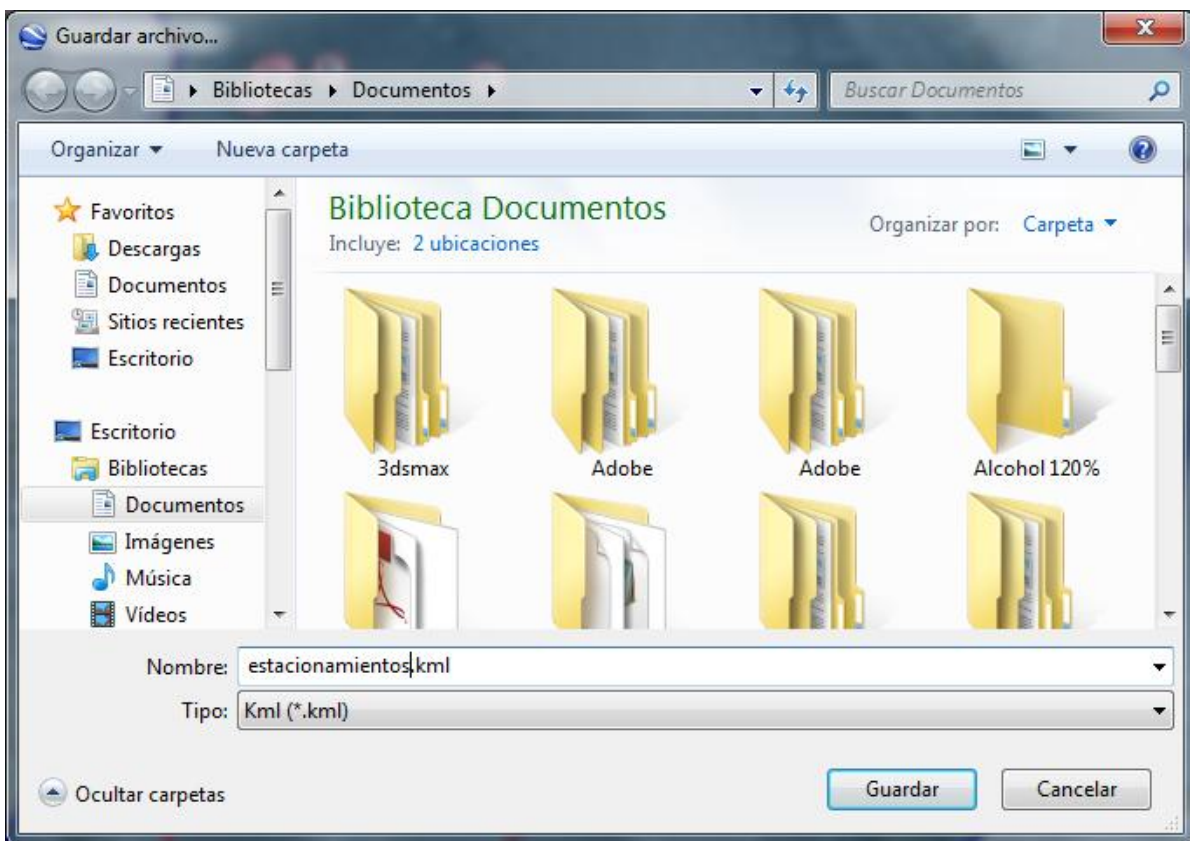


Figura 4.12. Ventana Guardar archivo

Fuente: Propia

Transformación de archivos KML a SHP

Para transformar los archivos a formato SHP, se utilizó el software libre Quantum GIS, que permite transformar fácilmente los archivos al formato requerido. Para realizar la transformación se selecciona del menú *Capa*, la opción *Añadir capa vectorial...* como se muestra en la Figura 4.13, o bien se presionan las teclas Ctrl+V. En la ventana *Añadir capa vectorial...*, se selecciona el archivo KML que se quiere transformar, presionando el botón *Explorar*, finalmente se presiona el botón *Open* para que Quantum GIS abra el archivo (Figura 4.14). Para guardar el archivo en formato SHP, se selecciona del menú *Capa*, la opción *Guardar como...* (Figura 4.15), en la ventana *Guardar capa vectorial como...*, se asigna el nombre de archivo y en el campo *Formato* se selecciona el valor *Archivo Shape de ESRI* (Figura 4.16).

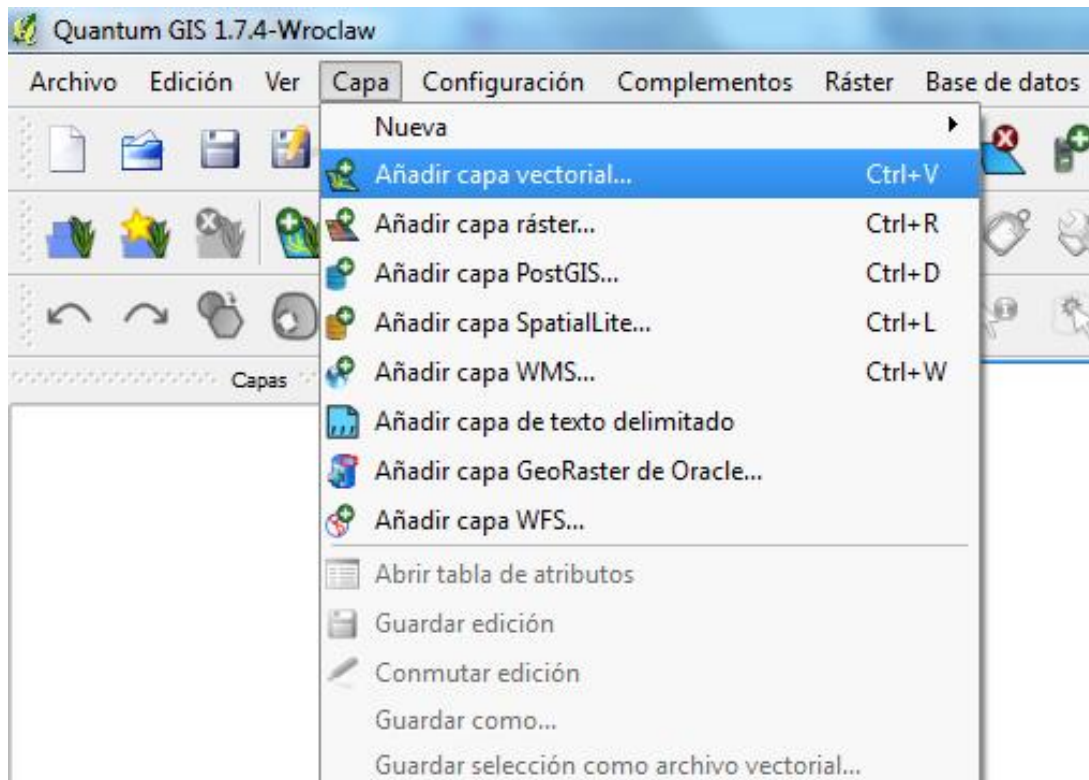


Figura 4.13. Opción Añadir capa vectorial del menú Capa

Fuente: Propia

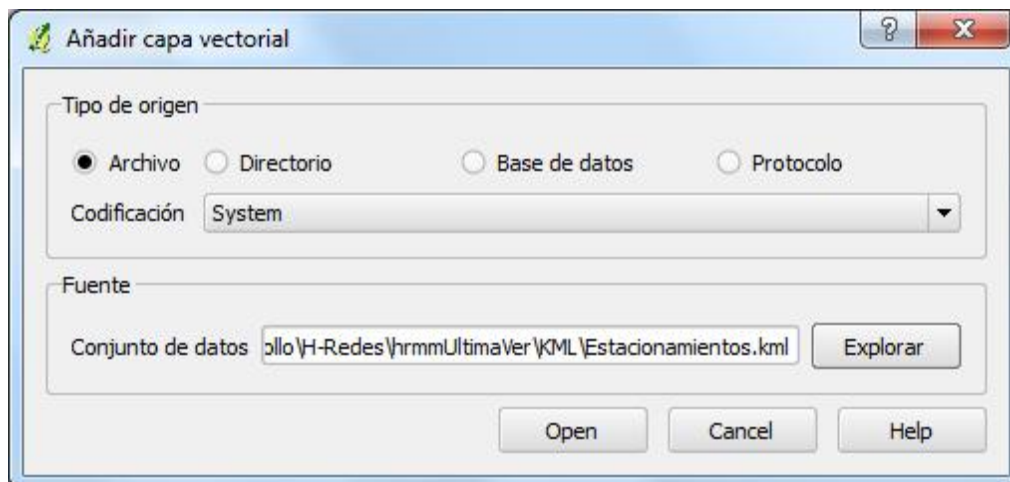


Figura 4.14. Ventana Añadir capa vectorial
Fuente: Propia

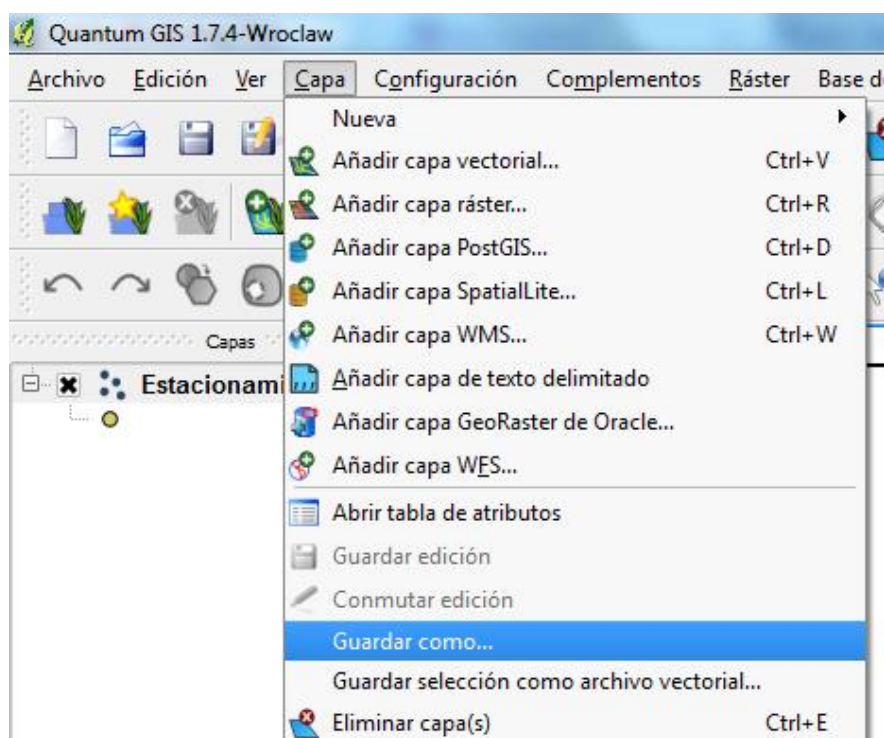


Figura 4.15. Opción Guardar como del menú Capa
Fuente: Propia

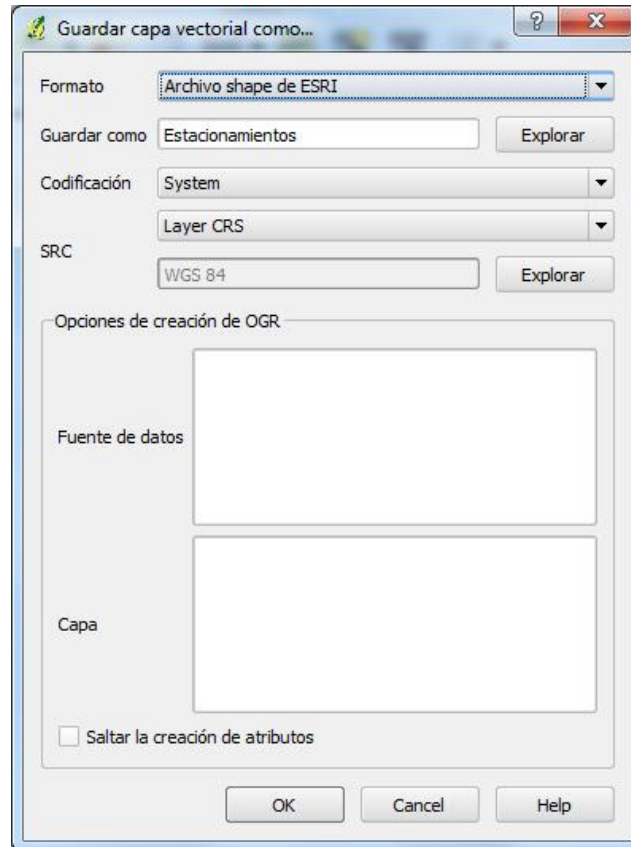


Figura 4.16. Ventana Guardar capa vectorial como
Fuente: Propia

Conversión de formato SHP a formato DBD

Teniendo los archivos en formato shape, es posible agregar información como los modos, rutas, líneas, frecuencias, tiempos de recorrido etc., esta información será agregada en TransCad y estará almacenada en *tablas de atributos*. La información contenida en dicha tabla y asociada tanto a las redes de transporte privado, como a las hiperredes de transporte público, tiene la siguiente estructura:

- **ID.** Identificador único para cada arco.
- **Length.** Longitud del arco.
- **Dir.** Dirección del arco. Valores almacenados: 0 (doble sentido), 1 (sentido en el que se dibujo el arco) y -1 (sentido contrario respecto al dibujo del arco).

- **Línea.** Línea asociada a un modo de transporte.
- **Modo.** Modo de transporte asociado al arco.
- **Ruta.** Ruta a la que pertenece el arco.
- **Frec_H_Pic.** Tiempo de espera en minutos asociado con la frecuencia del hiperarco en hora pico.
- **Frec_H_Val.** Tiempo de espera en minutos asociado con la frecuencia del hiperarco en hora valle.
- **T_D_Rec.** Tiempo de recorrido del arco.
- **Tipo.** Clasificación del arco/hiperarco, los valores son: A-Arco, H-Hiperarco y P-Peatonal.
- **TiempoApro.** Tiempo aproximado del arco/hiperarco (cifra redondeada a dos dígitos).
- **FROM_ID.** Nodo del que parte el arco/hiperarco.
- **TO_ID.** Nodo al que llega el arco/hiperarco.

Para asociar los datos al archivo SHP, debemos abrirlo en TransCad, a través de la opción *File* → *Open*. Al abrir el archivo aparece la ventana *ESRI Shapefile* (Figura 4.17), en la cual presionaremos el botón *Coordinates*, esto abre una nueva ventana de nombre *Import Coordinates*, donde se debe verificar que los datos sean los que se muestran en la Figura 4.18.

El archivo SHP abierto, se transforma al formato nativo de TransCad (DBD), para ello se selecciona la opción *Tools* → *Export*, y en la ventana *Export Geography*, seleccionar para el campo *Export* el valor *All features*, para el campo *To* el valor *Standard Geographic File* y para *ID Field* el valor *ID*, como se muestra en la Figura 4.19. Al exportar el archivo se generan 15 nuevos archivos con extensión: bin, bx, cdd, cdk, dbd, dcb, des, dsk, grp, lok, pnk, pts, r0, r1, sty.

El siguiente paso es abrir el archivo DBD para agregar la información al shape, para ello se selecciona la opción *File* → *Open* y en tipo de archivo se elige DBD. Para agregar más capas y poder trabajar de manera conjunta las hiperredes de transporte público y la red de vialidades para el automóvil, se presiona el botón *Map Layers* (Figura 4.20) y en la ventana *Layers* (Figura 4.21), se presiona el botón *Add Layer*.

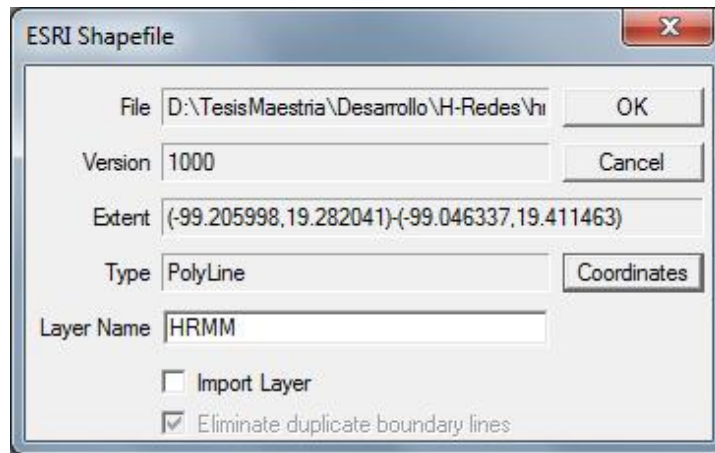


Figura 4.17. Ventana ESRI Shapefile

Fuente: Propia

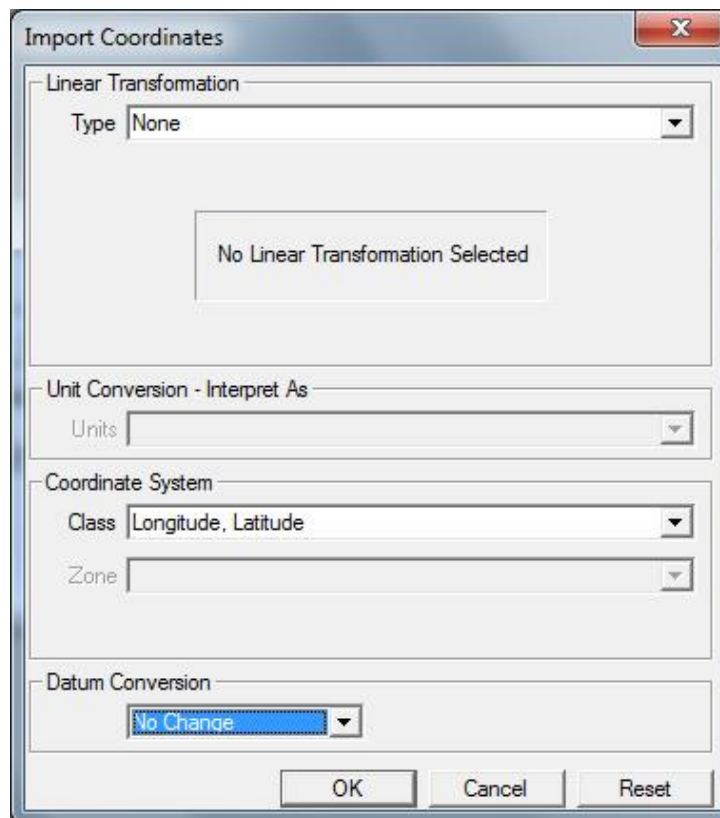


Figura 4.18. Ventana Import coordinates

Fuente: Propia

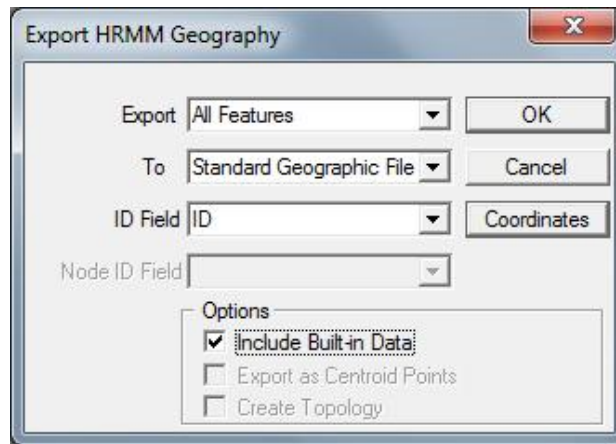


Figura 4.19. Ventana Export Geography

Fuente: Propia



Figura 4.20. Map Layers

Fuente: Propia

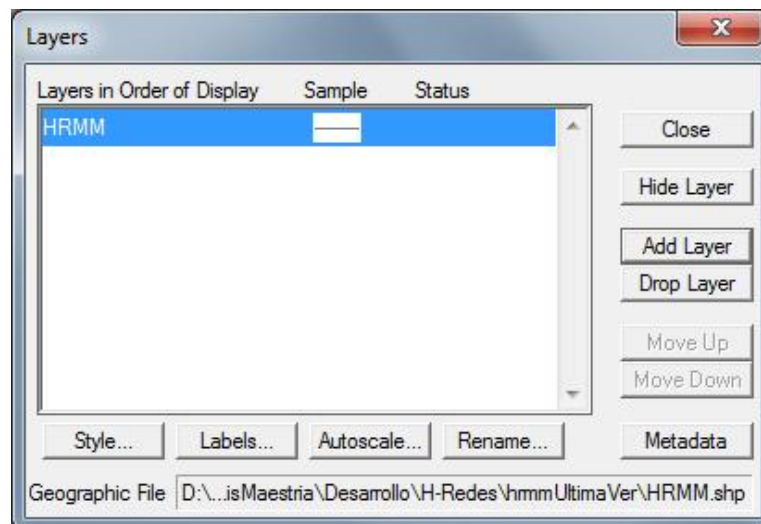


Figura 4.21. Ventana Layers

Fuente: Propia

Transformación del shape en arcos y nodos

Cada una de las curvas trazadas en Google Earth y que forman parte del archivo DBD, representan los recorridos de las distintas líneas de transporte público y vialidades para el automóvil. El trayecto entre dos estaciones de una misma línea se representa a través de un arco y las estaciones se representan como nodos, se deben agregar tanto arcos como nodos a cada shape del proyecto en TransCad, por lo tanto, se seccionarán cada una de las curvas para que estas sean una sucesión de segmentos (arcos) conectados por puntos (nodos). Para realizarlo se selecciona la opción *Tools* → *Map Editing* → *Toolbox* (Figura 4.22).

En la ventana *Toolbox Map Editing* (Figura 4.23), se puede realizar la edición tanto de líneas como de nodos, entre las herramientas del *Toolbox* están: modificar, agregar y borrar una línea, unir y separar líneas, editar los atributos de una línea y de un nodo, salvar y cancelar los cambios hechos en la edición.

Cuando se han creado todos los nodos, arcos e hiperarcos, se genera la estructura de la tabla, para ello se selecciona la opción *Dataview* → *Modify Table* (Figura 4.24) y en la ventana *Modify Table*, se agregan los campos necesarios y a cada campo se le asigna el tipo de datos que le corresponda, como se muestra en la Figura 4.25.

Habiéndose agregado todos los campos a la *tabla de atributos*, se debe llenar cada registro, para que todos los arcos y nodos tengan sus datos, y cuando se ejecute el algoritmo *SVHP*, se haga el cálculo utilizando dicha información. Para llenar los registros, se selecciona la opción *Edit line attributes* del cuadro de herramientas *Map editing* (Figura 4.26).

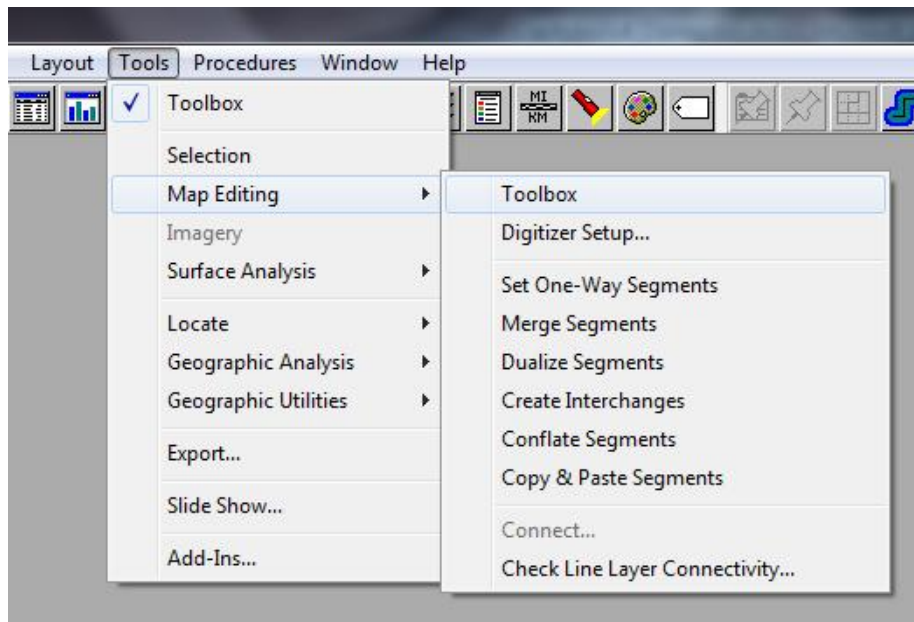


Figura 4.22. Menú para seleccionar la herramienta Map Editing
Fuente: Propia



Figura 4.23. Cuadro de herramientas Map Editing
Fuente: Propia

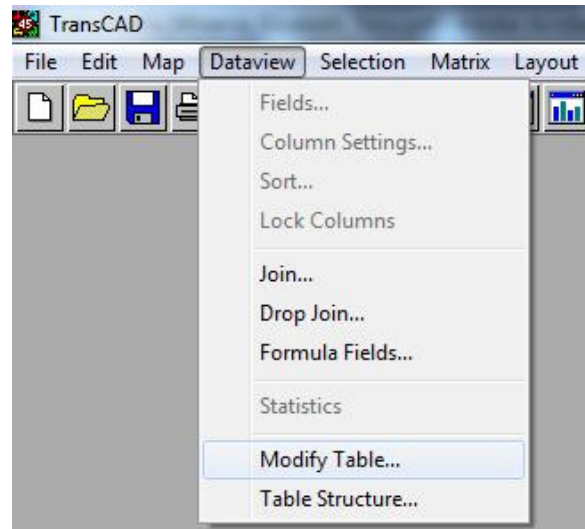


Figura 4.24. Menú para seleccionar la opción Modify table

Fuente: Propia

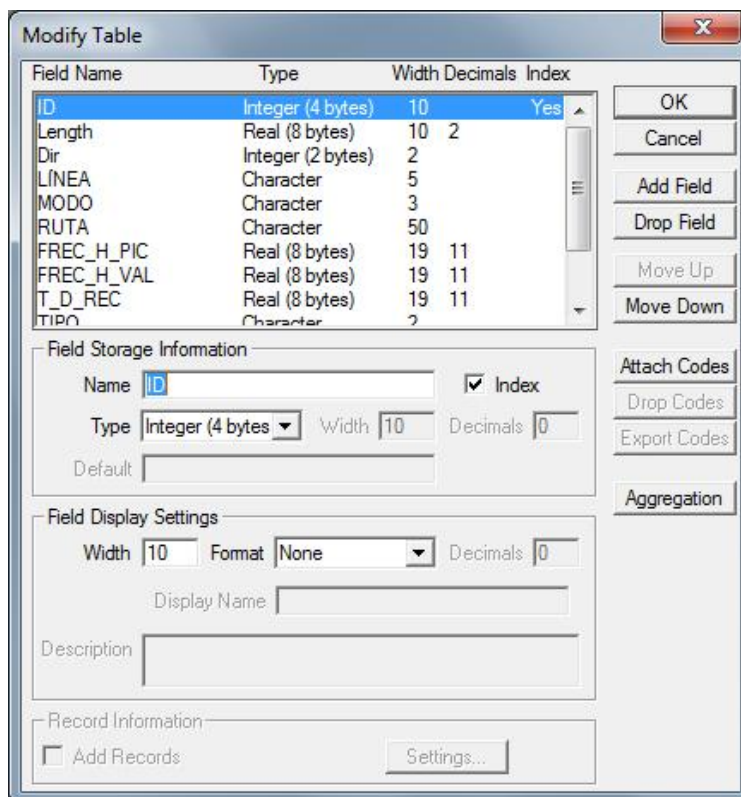


Figura 4.25. Ventana Modify Table

Fuente: Propia



Figura 4.26. Opción Edit line attributes

Fuente: Propia

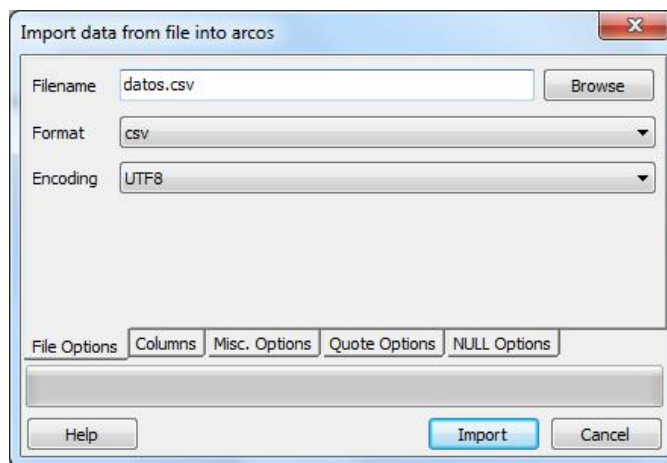


Figura 4.27. Ventana para importar datos CSV en PostgreSQL

Fuente: Propia

	ID integer	LENGTH real	DIR integer	LINEA text	MODO character(3)	RUTA text	FREC_H_PIC real	FREC_H_VAL real	T_D_REC real	TIPO character(1)	TIEMPOAPRO real	FROM_ID integer	TO_ID integer
1	2	0.27	1	Q	TLB	SUR_NORTE	0	0	2.7071	A	2.71	3	66
2	3	0.04	1	M	TLB	VILLA-DE-CO	0	0	0.161929	A	0.16	5	68
3	5	0.15	1	CCE	TLB	SUR_NORTE	0	0	1.06415	A	1.06	9	156
4	7	0.22	1	K	TLB	PONIENTE_OR	0	0	1.73779	A	1.74	13	201
5	9	0.62	1	L1	TL	TREN_LIGERO	0	0	2	A	2	17	18
6	9	0.62	1	L1	TL	TREN_LIGERO	0	0	2	A	2	18	17
7	10	0.71	1	L1	TL	TREN_LIGERO	0	0	2	A	2	18	19
8	10	0.71	1	L1	TL	TREN_LIGERO	0	0	2	A	2	19	18
9	11	0.4	1	L1	TL	TREN_LIGERO	0	0	2	A	2	20	19
10	11	0.4	1	L1	TL	TREN_LIGERO	0	0	2	A	2	19	20
11	12	0.45	1	L1	TL	TREN_LIGERO	0	0	2	A	2	20	21
12	12	0.45	1	L1	TL	TREN_LIGERO	0	0	2	A	2	21	20
13	13	0.41	1	L1	TL	TREN_LIGERO	0	0	2	A	2	21	22
14	13	0.41	1	L1	TL	TREN_LIGERO	0	0	2	A	2	22	21
15	14	0.39	1	L1	TL	TREN_LIGERO	0	0	2	A	2	22	23
16	14	0.39	1	L1	TL	TREN_LIGERO	0	0	2	A	2	23	22
17	15	0.41	1	L1	TL	TREN_LIGERO	0	0	2	A	2	23	24
18	15	0.41	1	L1	TL	TREN_LIGERO	0	0	2	A	2	24	23
19	16	0.28	1	L1	TL	TREN_LIGERO	0	0	2	A	2	24	25
20	16	0.28	1	L1	TL	TREN_LIGERO	0	0	2	A	2	25	24
21	17	0.3	1	L1	TL	TREN_LIGERO	0	0	2	A	2	25	26
22	17	0.3	1	L1	TL	TREN_LIGERO	0	0	2	A	2	26	25

Figura 4.28. Estructura de la tabla asociada a la hiperred

Fuente: Propia

Después de haber llenado todos los registros de la tabla de atributos, se exporta la información a formato CSV, seleccionando del menú *Save As* la opción *Comma-delimited text (*.txt, *.csv)*, posteriormente se importa la información al manejador de base de datos utilizado, en este caso PostgreSQL, como se muestra en la Figura 4.27. Los datos debidamente importados se deben observar en las tablas creadas en PostgreSQL como lo muestra la Figura 4.28.

4.3. Diagrama de estados para la zona de estudio

Se puede modelar la situación de un usuario en un viaje, haciendo uso de un diagrama de estados. Un estado, puede representar que el viajero esté caminando, en transporte público, haciendo una transferencia modal, usando el automóvil, etc.

Los elementos utilizados del diagrama de estados son los siguientes:

- **Estado.** Representa situaciones que ocurren durante el tiempo de análisis. Se simboliza gráficamente con un rectángulo.
- **Transición.** Representa el paso entre distintos estados, se etiqueta con el evento que lo provoca y con la acción resultante, en este caso se etiquetará con el modo de transporte utilizado en el trayecto que une dos estados. Se simboliza gráficamente con una flecha.

Considerando los conceptos de viabilidad explicados en las secciones 3.3 y 3.5, y considerando las restricciones sobre el uso del automóvil y la factibilidad de tomar cualquier modo de transporte público más de una vez sin importar el modo, se construye el diagrama de estados. Con ello es posible saber cuáles son las restricciones que se imponen para los modos restringidos (en este caso el automóvil). A partir del diagrama de estados, se determinarán las secuencias 'lógicas' o 'viables' para los modos de transporte en el área de estudio.

Tomando en cuenta que el automóvil solo puede ser utilizado al inicio o al final del viaje, se obtiene el siguiente diagrama de estados (Figura 4.29).

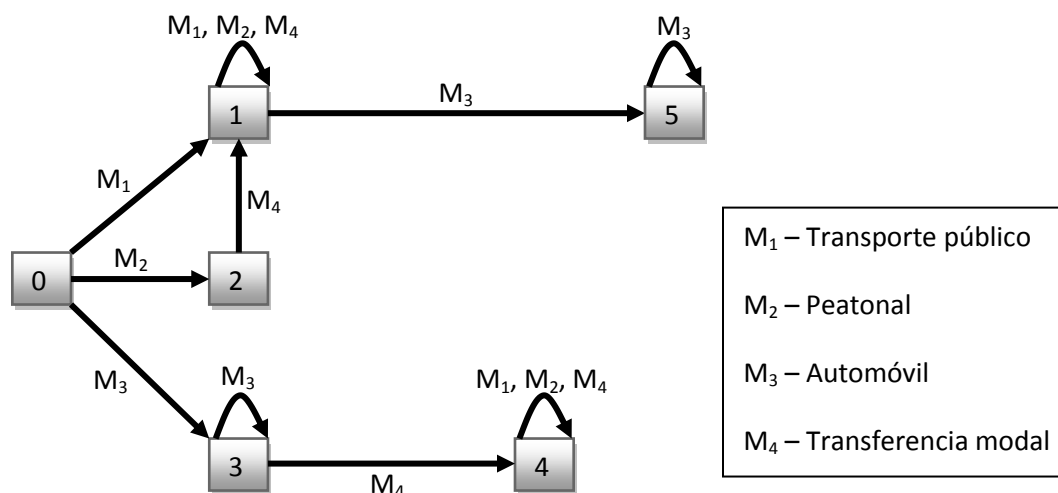


Figura 4.29. Diagrama de estados

Fuente: Propia

El usuario puede partir del origen en transporte público, caminando o en automóvil. Si aborda de inicio el transporte público (estado 1) puede continuar en transporte público, caminar o realizar una transferencia (sigue en el estado 1), o bien, puede abordar el coche para finalizar su recorrido (estado 5). Si el viaje comienza en automóvil (estado 3), el usuario puede continuar haciendo uso del mismo (estado 3), o bien, dejar su coche en un estacionamiento y terminar su viaje en transporte público (estado 4).

A través del diagrama de estados, se observa que el automóvil es un modo restringido, ya que no puede usarse en más de una ocasión y solamente puede ser utilizado al inicio o al final del viaje.

El diagrama de estados se incluye como un procedimiento ligado al procedimiento SVHP para verificar la viabilidad de los modos concatenados, y con ello, generar un viaje haciendo uso de uno o más modos, con una secuencia lógica.

En el siguiente capítulo se describe la implementación del SAIV desde el enfoque de base de datos y de programación del sistema multiusuario, también se presentan los resultados obtenidos al ejecutar el algoritmo SVHP sobre las hiperredes y redes que se analizaron en este capítulo.

Capítulo 5 . Implementación del SAIV y obtención de resultados

En el presente capítulo se explica el proceso de implementación del SAIV, comenzando por el diseño y desarrollo de la base de datos, posteriormente se explica la configuración del servidor y la codificación de las clases en java. También se presenta un análisis sobre algunos ejemplos generados por el SAIV, cada ejemplo consta de una o más hiperrutas mínimas viables multimodales, las cuales son descritas en texto y representadas gráficamente.

5.1. Programas utilizados en la implementación del SAIV

Para implementar el algoritmo SVHP descrito en el capítulo 3, fue necesario construir la hiperred de transporte público, así como la red de transporte privado, para ello se utilizaron los Sistemas de Información Geográfica, ArcGIS y TransCAD. Para la implementación del sistema SAIV, se programó en lenguaje JAVA a través del entorno de desarrollo Eclipse Java EE IDE for Web Developers, a través del Java Development Kit (JDK) en su versión 7.0; para almacenar los datos y estructuras se utilizó el manejador de base de datos PostgreSQL.

El usuario podrá hacer uso del SAIV a través de una página web, para ello se utiliza Apache TomCat como servidor web y OpenLayers para mostrar las estaciones de los diversos modos de transporte en un mapa. En la Tabla 5.1, se muestran cada una de las secciones de la implementación del sistema y los programas utilizados.

Tabla 5.1. Programas utilizados para la implementación el sistema

SIG	Realización de la hiperred	ArcGIS TransCAD
Desarrollo del Algoritmo	Codificación	Eclipse Java EE IDE For web developers, JDK 7.0
	Bases de datos	PostgreSQL
Sistema Web	Servidor Web	Apache TomCat
	Servidor de mapas	OpenLayers

Fuente: Propia

El primer paso de la etapa de desarrollo, consistió en digitalizar las hiperredes y redes que forman el área de estudio tanto para el transporte público como para el privado, esta explicación se encuentra en las secciones 4.2 y 4.3. Una vez digitalizadas las hiperredes, fue necesario transformar las tablas de datos (en formato dbf y asociadas al archivo shp) en archivos separados por comas (CSV), de tal forma que se pudiera exportar la información al manejador de base de datos (PostgreSQL).

5.2. Transformación de la tabla de datos asociada a la hiperred a un archivo separado por comas (CSV)

Cada uno de los grafos modelados (hiperredes de transporte público y la red de vialidades principales) tienen asociada una tabla de datos, la cual en primera instancia es exportada desde TransCAD a formato DBF para poder ser editada en Excel 2007, en este paquete se quitan los caracteres no contemplados en codificaciones de caracteres como la ANSI o la UTF-8, finalmente el archivo se guarda como un archivo separado por comas o CSV (Comma Separated Values, como se muestra en la Figura 5.1).

El archivo CSV deberá ser guardado desde el programa NotePad++ con la codificación UTF-8, para evitar errores en la importación a PostgreSQL.

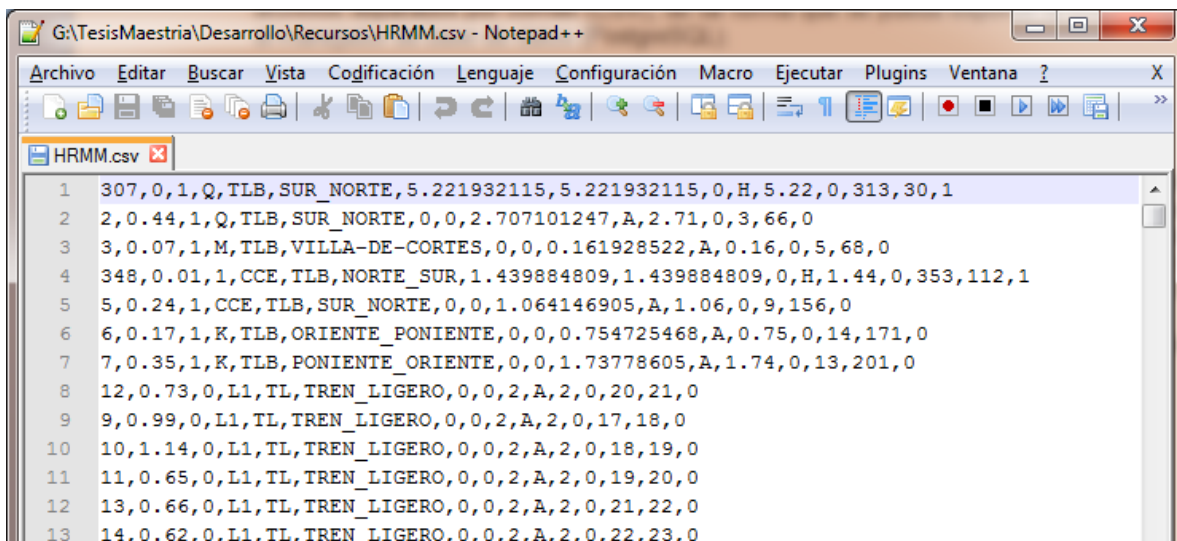


Figura 5.1. Archivo separado por comas (CSV)

Fuente: Propia

5.3. Creación de la base de datos en PostgreSQL y llenado de las tablas a través de JDBC

La primera parte del proceso (creación de las tablas de la base de datos) fue realizada en PostgreSQL, la segunda parte fue programada en Java con la API JDBC (Java DataBase Connectivity) para conectarse a PostgreSQL y llenar las tablas de manera automática.

5.3.1. Creación de las tablas en PostgreSQL

Para generar una nueva tabla se presiona con el botón derecho en "Tables" y en el menú emergente se selecciona la opción "New Table...", como se muestra en la Figura 5.2.

En la pestaña "Properties" de la ventana "New Table..." se asigna el nombre de la tabla en el campo "Name", en este caso se asignó el nombre "hrmm", como se muestra en la Figura 5.3. En la pestaña "Columns" de la misma ventana, se agregan las columnas que tendrá la tabla; en el campo "Column name" se asigna el nombre de la columna, mientras que en el campo "Definition" se asigna el tipo de dato de la columna. Los tipos de datos utilizados fueron entero, flotante y texto (integer, real y text), como se muestra en la Figura 5.4.

Al terminar se presiona el botón "OK" y con esto la tabla habrá sido creada, con las características previamente referidas.

Para importar los datos del archivo CSV a la tabla hrmm, se ejecuta el código SQL:

```
copy "hrmm" from 'D:/BD/HRMM.csv' WITH DELIMITER ',';
```

Los datos copiados en la tabla hrmm, se muestran en la Figura 5.5.

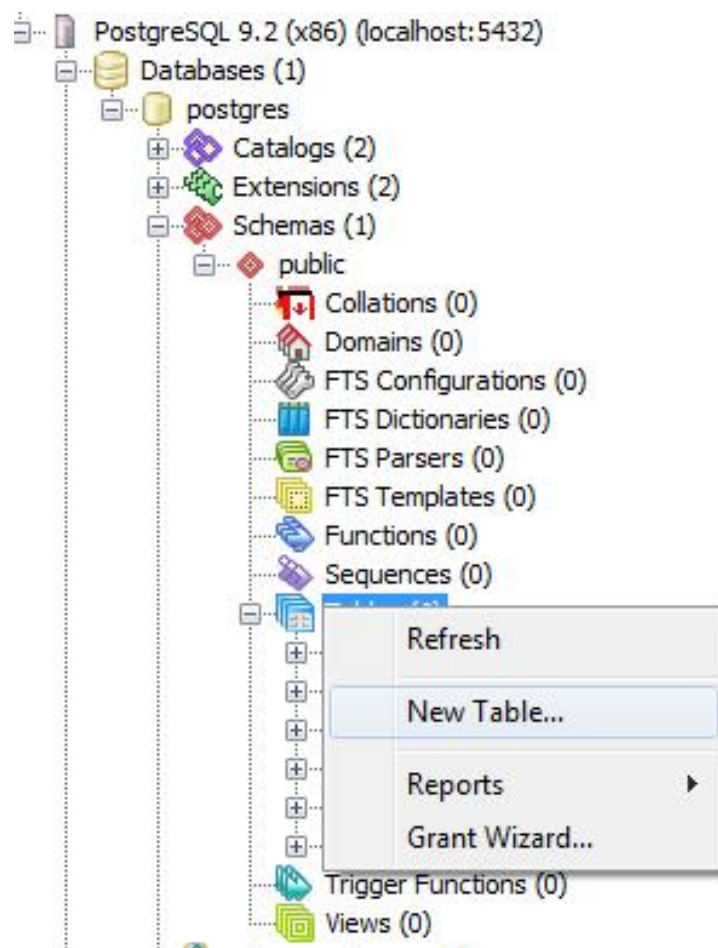


Figura 5.2. Creación de una nueva tabla en PostgreSQL

Fuente: Propia

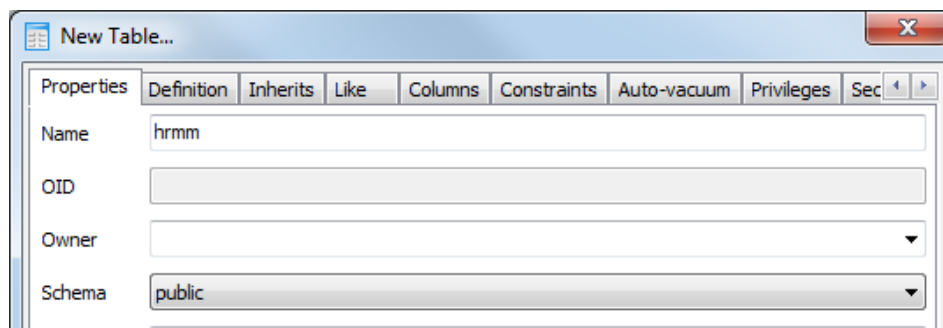


Figura 5.3. Asignación del nombre a la nueva tabla

Fuente: Propia

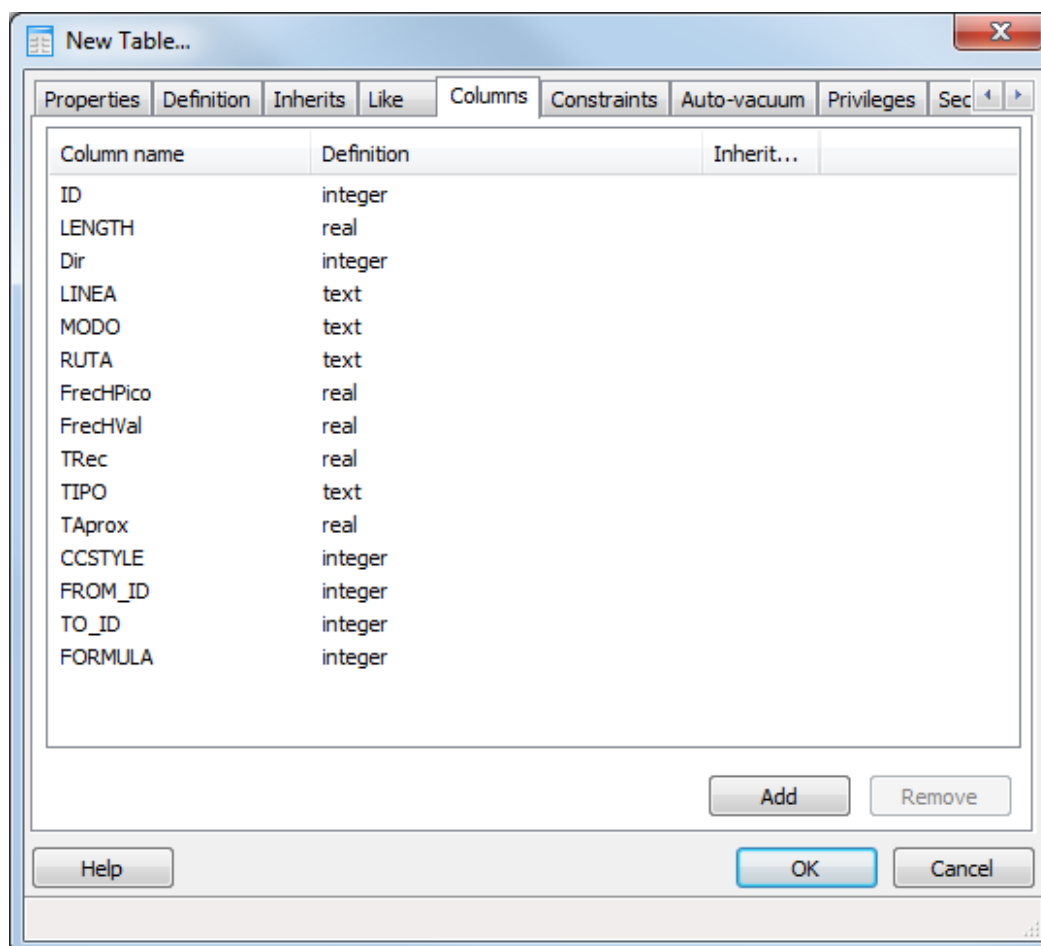


Figura 5.4. Creación de columnas para una tabla en PostgreSQL

Fuente: Propia

ID integer	LENGTH real	Dir integer	LINEA text	MODO text	RUTA text	FrechPico real	FrechVal real	TRec real	TIPO text	TAprox real	CCSTYLE integer	FROM_ID integer	TO_ID integer	FORMULA integer	
1	307	0	1	Q	TLB	SUR_NORTE	5.22193	5.22193	0	H	5.22	0	313	30	1
2	2	0.44	1	Q	TLB	SUR_NORTE	0	0	2.7071	A	2.71	0	3	66	0
3	3	0.07	1	M	TLB	VILLA-DE-CC	0	0	0.161929	A	0.16	0	5	68	0
4	348	0.01	1	CCE	TLB	NORTE_SUR	1.43988	1.43988	0	H	1.44	0	353	112	1
5	5	0.24	1	CCE	TLB	SUR_NORTE	0	0	1.06415	A	1.06	0	9	156	0
6	6	0.17	1	K	TLB	ORIENTE_PON	0	0	0.754725	A	0.75	0	14	171	0
7	7	0.35	1	K	TLB	PONIENTE_OR	0	0	1.73779	A	1.74	0	13	201	0
8	12	0.73	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	20	21	0
9	9	0.99	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	17	18	0
10	10	1.14	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	18	19	0
11	11	0.65	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	19	20	0
12	13	0.66	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	21	22	0
13	14	0.62	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	22	23	0
14	15	0.66	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	23	24	0
15	16	0.44	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	24	25	0
16	17	0.48	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	25	26	0
17	18	0.46	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	26	27	0
18	19	0.59	0	L1	TL	TREN_LIGERC	0	0	2	A	2	0	27	28	0
19	145	0.27	1	CCE	TLB	NORTE_SUR	0	0	1.17652	A	1.18	0	153	8	0
20	21	0.25	1	Q	TLB	NORTE_SUR	0	0	1.81476	A	1.81	0	29	31	0
21	306	0	1	Q	TLB	NORTE_SUR	5.22193	5.22193	0	H	5.22	0	312	29	1
22	23	0.32	1	Q	TLB	NORTE_SUR	0	0	2.32606	A	2.33	0	31	33	0
23	24	0.28	1	Q	TLB	SUR_NORTE	0	0	1.69604	A	1.7	0	32	30	0
24	25	0.34	1	Q	TLB	NORTE_SUR	0	0	2.47975	A	2.48	0	33	35	0
25	26	0.25	1	Q	TLB	SUR_NORTE	0	0	1.52137	A	1.52	0	34	32	0
26	27	0.24	1	Q	TLB	NORTE_SUR	5.22193	5.22193	1.72818	A	1.73	0	35	37	0
27	28	0.31	1	Q	TLB	SUR_NORTE	0	0	1.88876	A	1.89	0	36	34	0
28	29	0.51	1	Q	TLB	NORTE_SUR	0	0	3.70377	A	3.7	0	37	39	0
29	30	0.3	1	Q	TLB	SUR_NORTE	0	0	1.86606	A	1.87	0	38	36	0
30	31	0.39	1	Q	TLB	NORTE_SUR	0	0	2.77896	A	2.78	0	39	41	0

Figura 5.5. Datos importados en PostgreSQL

Fuente: Propia

Habiéndose creado la tabla a través del procedimiento descrito, se procedió a adecuar la misma a través de la API JDBC de JAVA.

5.3.2. Adecuación de las tablas con JDBC

Para que el algoritmo de búsqueda funcione correctamente, es necesario que el campo DIR (dirección de flujo del arco) cuente únicamente con valores 1; de inicio este campo creado automáticamente en TransCAD almacena los valores 0 (doble sentido), 1 (sentido en el que se dibujo el arco) y -1 (sentido contrario respecto al dibujo del arco). Un arco cualquiera está formado por dos nodos, una cola y una cabeza (capítulo 3), en este caso los nodos están almacenados en los campos FROM_ID y TO_ID.

El algoritmo programado en JDBC de java, lee en primera instancia el campo DIR, si el valor es 1, el algoritmo no altera los valores de los campos FROM_ID y TO_ID, si el valor es -1 invierte los valores escritos en dichos campos, y si es cero, duplica los registros

dejando en el primero de ellos los campos FROM_ID y TO_ID sin cambio alguno y en el segundo registro invierte los valores asignados a estos campos.

Los datos almacenados en PostgreSQL deben ser transferidos a arreglos en java, para ello se utiliza la función "get", como se muestra a continuación para los tipos de datos entero, flotante y texto.

Las variables ID, LONGITUD y LINEA son arreglos del tipo integer, float y string respectivamente. Para el tipo de datos entero se utiliza la función "getInt", como parámetro de la función se usa el campo de la tabla de PostgreSQL, en este caso el campo es "ID", lo mismo ocurre para los otros dos tipos de datos utilizados. Después de haber transferido todos los datos de la tabla "hrmm" a arreglos en java, se deberá hacer la copia de los datos a la tabla "arcos", eliminando los valores de 0 y -1 del campo DIR, como se mencionó anteriormente, para ello se utiliza un objeto del tipo "PreparedStatement" en el cual se almacena un query SQL. El código que inserta los datos en la nueva tabla, se muestra en la Figura 5.7.

Después de escribir en las 13 columnas de la tabla, se debe ejecutar la instrucción executeUpdate () para grabar los datos de manera permanente en la base de datos. Para entender mejor la codificación de esta sección se recomienda leer el Anexo B (JDBC para estructuras de datos).

Al ejecutar el código java, se puede observar dentro de PostgreSQL como la tabla arcos se rescribió dejando únicamente el valor de 1 para el campo DIR, siguiendo el procedimiento descrito con anterioridad y cuyo resultado se muestra en la Figura 5.8.

```
ID = rs.getInt ("ID");  
LONGITUD = rs.getFloat ("LENGTH");  
LINEA = rs.getString ("LINEA");
```

Figura 5.6. Transferencia de los datos de PostgreSQL a Java

Fuente: Propia

```
PreparedStatement escribe = con.prepareStatement("INSERT INTO arcos VALUES
(?,?,?,?,?,?,?,?,?,?,?,?);");
for (k=1; k<j; k++) {
    escribe.setInt (1, identificador[k]);
    escribe.setFloat (2, longitud[k]);
    escribe.setInt (3, direccion[k]);
    escribe.setString (4, linea[k]);
    escribe.setString (5, modo[k]);
    escribe.setString (6, ruta[k]);
    escribe.setFloat (7, frecHpico[k]);
    escribe.setFloat (8, frecHvalle[k]);
    escribe.setFloat (9, tiempo[k]);
    escribe.setString (10, tipo[k]);
    escribe.setFloat (11, tiempoa[k]);
    escribe.setInt (12, fromID[k]);
    escribe.setInt (13, toID[k]);
    escribe.executeUpdate ();
}
}
```

Figura 5.7. Creación de la tabla de arcos desde Java

Fuente: Propia

ID	LENGTH	Dir	LINEA	MODO	RUTA	FrechHPico	FrechHVal	TRec	TIPO	TAprox	FROM_ID	TO_ID	
integer	real	integer	text	text	text	real	real	real	text	real	integer	integer	
1	307	0	1	Q	TLB	SUR_NORTE	5.22193	5.22193	0	H	5.22	313	30
2	2	0.44	1	Q	TLB	SUR_NORTE	0	0	2.7071	A	2.71	3	66
3	3	0.07	1	M	TLB	VILLA-DE-CC	0	0	0.161929	A	0.16	5	68
4	348	0.01	1	CCE	TLB	NORTE_SUR	1.43988	1.43988	0	H	1.44	353	112
5	5	0.24	1	CCE	TLB	SUR_NORTE	0	0	1.06415	A	1.06	9	156
6	6	0.17	1	K	TLB	ORIENTE_PON	0	0	0.754725	A	0.75	14	171
7	7	0.35	1	K	TLB	PONIENTE_OR	0	0	1.73779	A	1.74	13	201
8	12	0.73	1	L1	TL	TREN_LIGERC	0	0	2	A	2	20	21
9	12	0.73	1	L1	TL	TREN_LIGERC	0	0	2	A	2	21	20
10	9	0.99	1	L1	TL	TREN_LIGERC	0	0	2	A	2	17	18
11	9	0.99	1	L1	TL	TREN_LIGERC	0	0	2	A	2	18	17
12	10	1.14	1	L1	TL	TREN_LIGERC	0	0	2	A	2	18	19
13	10	1.14	1	L1	TL	TREN_LIGERC	0	0	2	A	2	19	18
14	11	0.65	1	L1	TL	TREN_LIGERC	0	0	2	A	2	19	20
15	11	0.65	1	L1	TL	TREN_LIGERC	0	0	2	A	2	20	19
16	13	0.66	1	L1	TL	TREN_LIGERC	0	0	2	A	2	21	22
17	13	0.66	1	L1	TL	TREN_LIGERC	0	0	2	A	2	22	21
18	14	0.62	1	L1	TL	TREN_LIGERC	0	0	2	A	2	22	23
19	14	0.62	1	L1	TL	TREN_LIGERC	0	0	2	A	2	23	22
20	15	0.66	1	L1	TL	TREN_LIGERC	0	0	2	A	2	23	24
21	15	0.66	1	L1	TL	TREN_LIGERC	0	0	2	A	2	24	23
22	16	0.44	1	L1	TL	TREN_LIGERC	0	0	2	A	2	24	25
23	16	0.44	1	L1	TL	TREN_LIGERC	0	0	2	A	2	25	24
24	17	0.48	1	L1	TL	TREN_LIGERC	0	0	2	A	2	25	26
25	17	0.48	1	L1	TL	TREN_LIGERC	0	0	2	A	2	26	25

Figura 5.8. Homologación del campo dirección en la tabla asociada a los arcos

Fuente: Propia

5.4. Configuración del Servidor Web

El siguiente paso es instalar un servidor web (Apache Tomcat 7.0.42) que permita manejar las clases programadas en java asociadas al algoritmo SVHP (servlets) y desplegar las páginas web: para introducir los datos (jsp) y para presentar los resultados (html).

El sistema programado funciona de la siguiente manera:

1. El cliente pide, a través de un browser, una página a Apache Tomcat.
2. Tomcat selecciona al servlet indicado para realizar la tarea del usuario.
3. El servlet (que es un objeto de java), genera la página web.
4. Se devuelve al browser la página web generada.

La estructura general del sistema se muestra en la Figura 5.9.

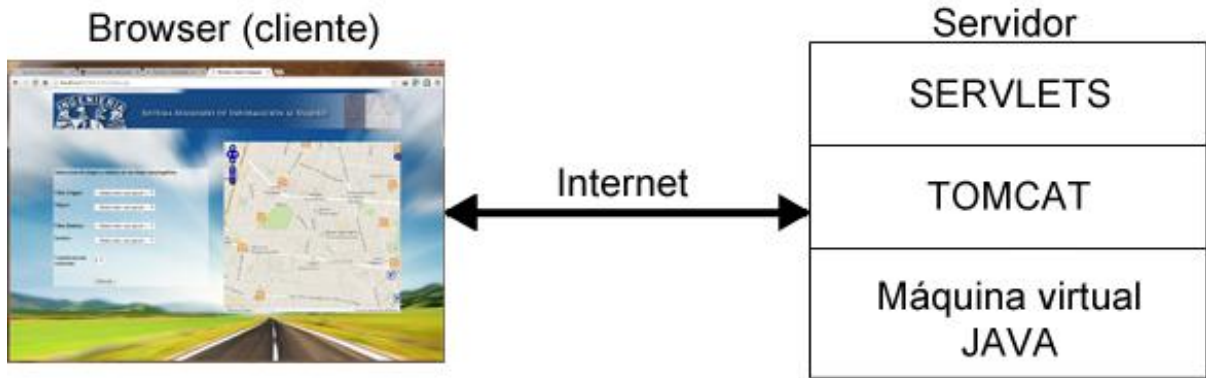


Figura 5.9. Estructura general del sistema web

Fuente: Propia


Al instalar Apache Tomcat, se genera la siguiente estructura de directorios:

- bin
- common
- conf
- logs
- server
- shared
- webapps
- work


De los directorios instalados por el servidor web, se utilizaron únicamente dos directorios (bin y webapps) para el funcionamiento del SAIV, los cuales se explican a continuación:

- **bin.** Aquí se encuentran los scripts que permiten arrancar/parar al servidor web. Si se ha logrado "arrancar" (iniciar) el servidor Apache Tomcat de manera correcta se mostrará una pantalla similar a la de la Figura 5.10.
- **webapps.** En dicho directorio se copiarán los archivos del sistema programado, el cual incluye las páginas web en formato html y jsp, y los servlets asociados al algoritmo SVHP.

Home Documentation Configuration Examples Wiki Mailing Lists Find Help

Apache Tomcat/7.0.42  <http://www.apache.org/>

If you're seeing this, you've successfully installed Tomcat. Congratulations!

 **Recommended Reading:**

- [Security Considerations HOW-TO](#)
- [Manager Application HOW-TO](#)
- [Clustering/Session Replication HOW-TO](#)

Server Status
Manager App
Host Manager

Developer Quick Start

- [Tomcat Setup](#)
- [Realms & AAA](#)
- [Examples](#)
- [Servlet Specifications](#)
- [First Web Application](#)
- [JDBC Data Sources](#)
- [Tomcat Versions](#)

Managing Tomcat

For security, access to the [manager.webapp](#) is restricted. Users are defined in:

```
#CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 7.0 access to the manager application is split between different users. [Read more...](#)

[Release Notes](#)

[Changelog](#)

[Migration Guide](#)

[Security Notices](#)

Documentation

[Tomcat 7.0 Documentation](#)

[Tomcat 7.0 Configuration](#)

[Tomcat Wiki](#)

Find additional important configuration information in:

```
#CATALINA_HOME/RUNNING.txt
```

Developers may be interested in:

- [Tomcat 7.0 Bug Database](#)
- [Tomcat 7.0 JavaDocs](#)
- [Tomcat 7.0 SVN Repository](#)

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

- announce@tomcat.apache.org
important announcements, releases, security vulnerability notifications. (Low volume).
- users@tomcat.apache.org
User support and discussion
- taglibs-user@tomcat.apache.org
User support and discussion for [Apache Taglibs](#)
- dev@tomcat.apache.org
Development mailing list, including commit messages

Other Downloads: [Tomcat Connectors](#), [Tomcat Native](#) | Other Documentation: [Tomcat Connectors](#), [mod_ik Documentation](#) | Get Involved: [Overview](#), [SVN Repositories](#) | Miscellaneous: [Contact](#), [Legal](#) | Apache Software Foundation: [Who We Are](#)

Figura 5.10. Apache Tomcat

Fuente: Propia

5.5. Programación del sistema en Java

Habiendo instalado el servidor Web, se pueden realizar pruebas sobre las clases programadas y simular la instalación del sistema en un servidor real.

El sistema fue programado en Java, debido a que el sistema es portable, no depende del sistema operativo donde se instale (ya que se instala una maquina virtual para su ejecución), es un lenguaje orientado a objetos y basado en clases, el código no necesita ser recompilado y es uno de los lenguajes más usados para generar aplicaciones cliente-servidor, debido a esto existen muchos recursos y material didáctico para aprender dicho lenguaje.

Para la codificación se utilizó el compilador Eclipse Java JEE for Web Developers, versión Índigo Service Release 2, debido a la facilidad que ofrece para configurar dentro del entorno el servidor web y poder compilar el código y simular su ejecución en un navegador web sin tener que salir del compilador.

5.5.1. Configuración de Eclipse JEE

Lo primero que se debe hacer es registrar el servidor apache Tomcat 7.0, que fue el que se utilizó para la implementación del sistema SAIV.

Presionando el botón derecho del mouse sobre el *explorador del proyecto (Project Explorer)*, se elige la opción *New* → *Other* → *Server*, esto abre la ventana *New Server* que se muestra en la Figura 5.11.

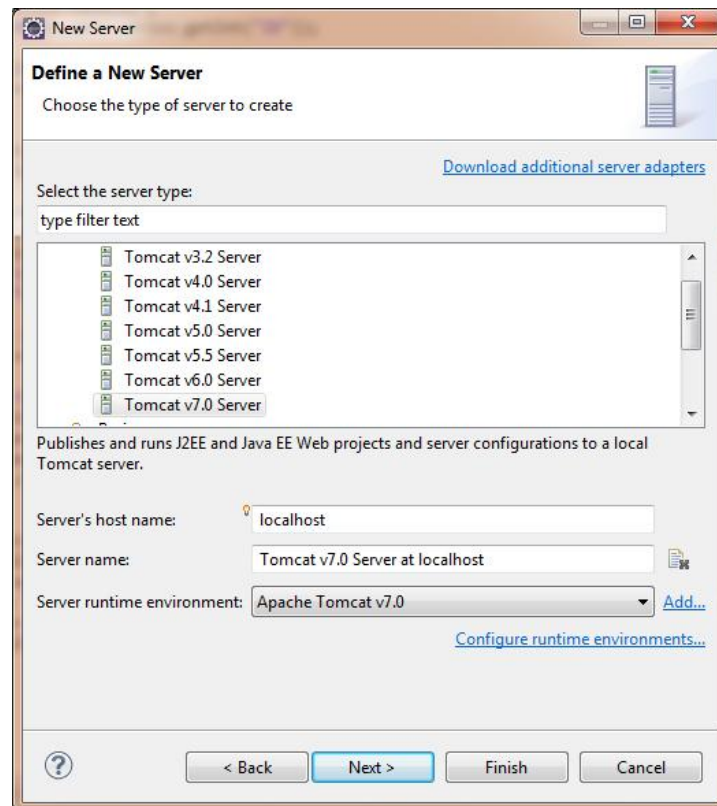


Figura 5.11. Registro del nuevo servidor en Eclipse

Fuente: Propia

Al presionar el botón *Next* (Figura 5.11), se abre una nueva ventana (*New Server Runtime Environment*), en ella se selecciona el servidor en el cual se copiarán los archivos del sistema SAIV y la ruta donde se ubica, como se muestra en la Figura 5.12. En este caso se eligió el servidor *Apache Tomcat versión 7.0*.

Presionando el botón *Finish*, el servidor se mostrará en el *explorador del proyecto* (Project Explorer), como se muestra en la Figura 5.13.

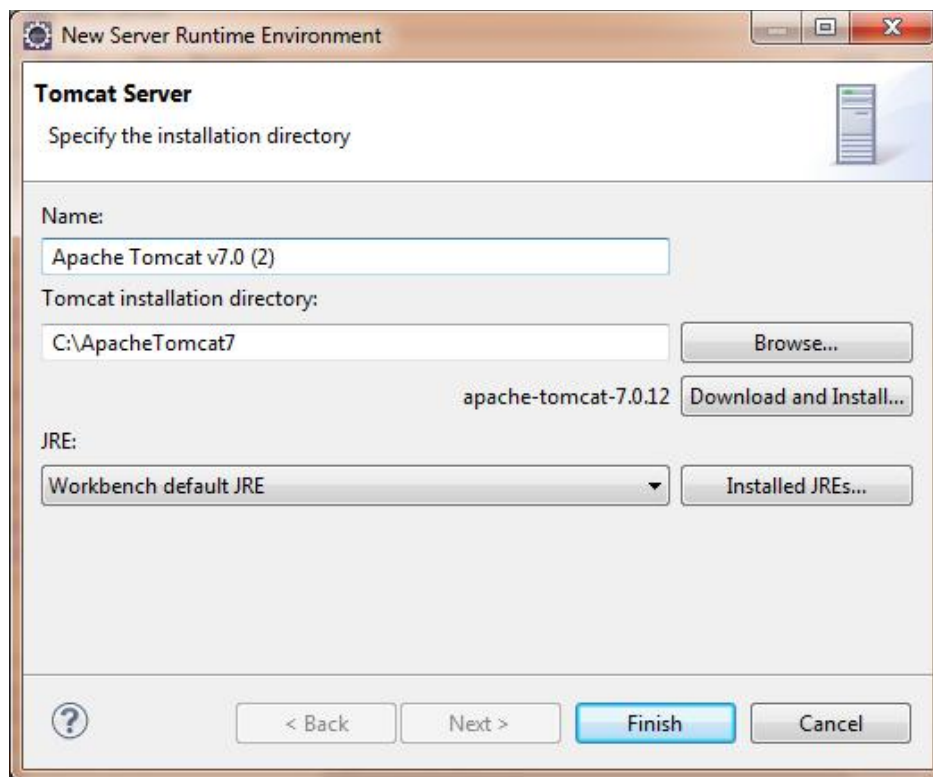


Figura 5.12. Selección del tipo de servidor y su ubicación

Fuente: Propia

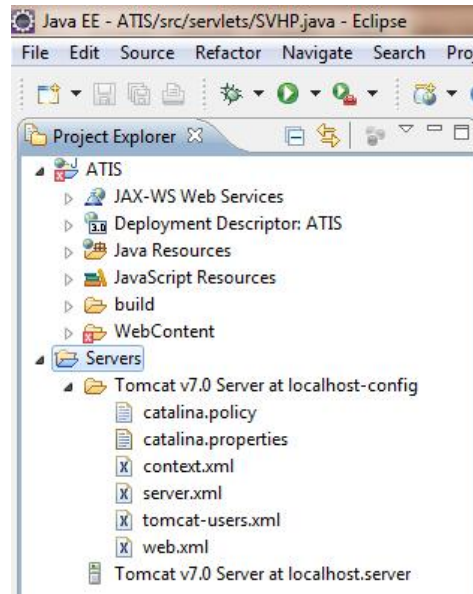


Figura 5.13. Visualización del servidor en el explorador

Fuente: Propia

Después de registrar el servidor en el entorno de Eclipse, se crea un proyecto nuevo, en este caso como se van a crear páginas web dinámicas haciendo uso de jsp y servlets, se selecciona la opción New → Project → Dynamic Web Project, como se muestra en la Figura 5.14.

Finalmente, se escribe el nombre del proyecto, su ubicación, el entorno de ejecución y el servidor web que se utilizará en el proyecto, como se muestra en Figura 5.15. Para finalizar, se presiona el botón *Finish* para crear el proyecto.

En la Figura 5.16, se muestra la estructura del proyecto creado, existen dos carpetas de interés donde se escribirán las clases (servlets cuya extensión es *.java*) y las páginas *jsp*.

- *Java Resources*: Aquí estarán todos los archivos *.java*. A partir de estos archivos, Eclipse generará los archivos de clase (*.class*).
- *WebContent*: Aquí se ubicarán los archivos *jsp* y *html* del proyecto.

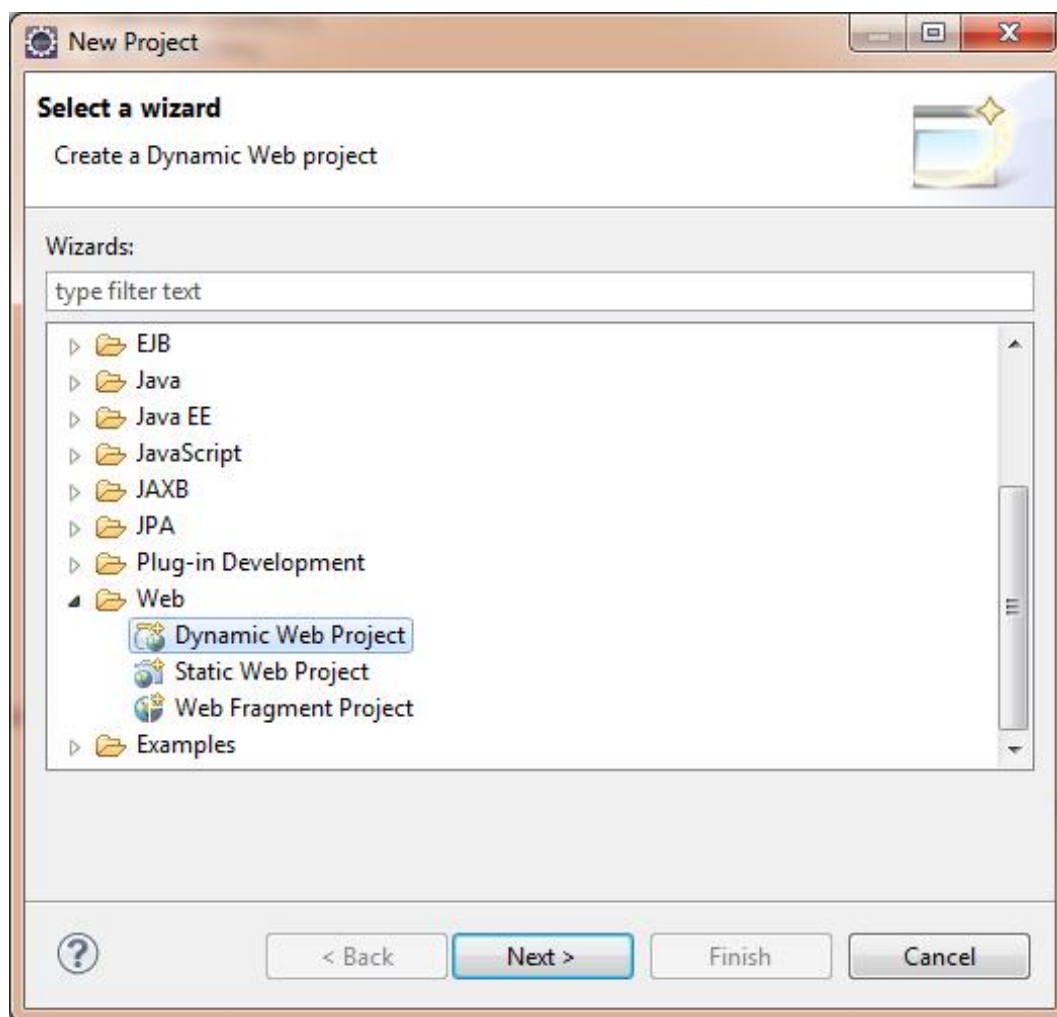


Figura 5.14. Proyecto web dinámico en Eclipse

Fuente: Propia

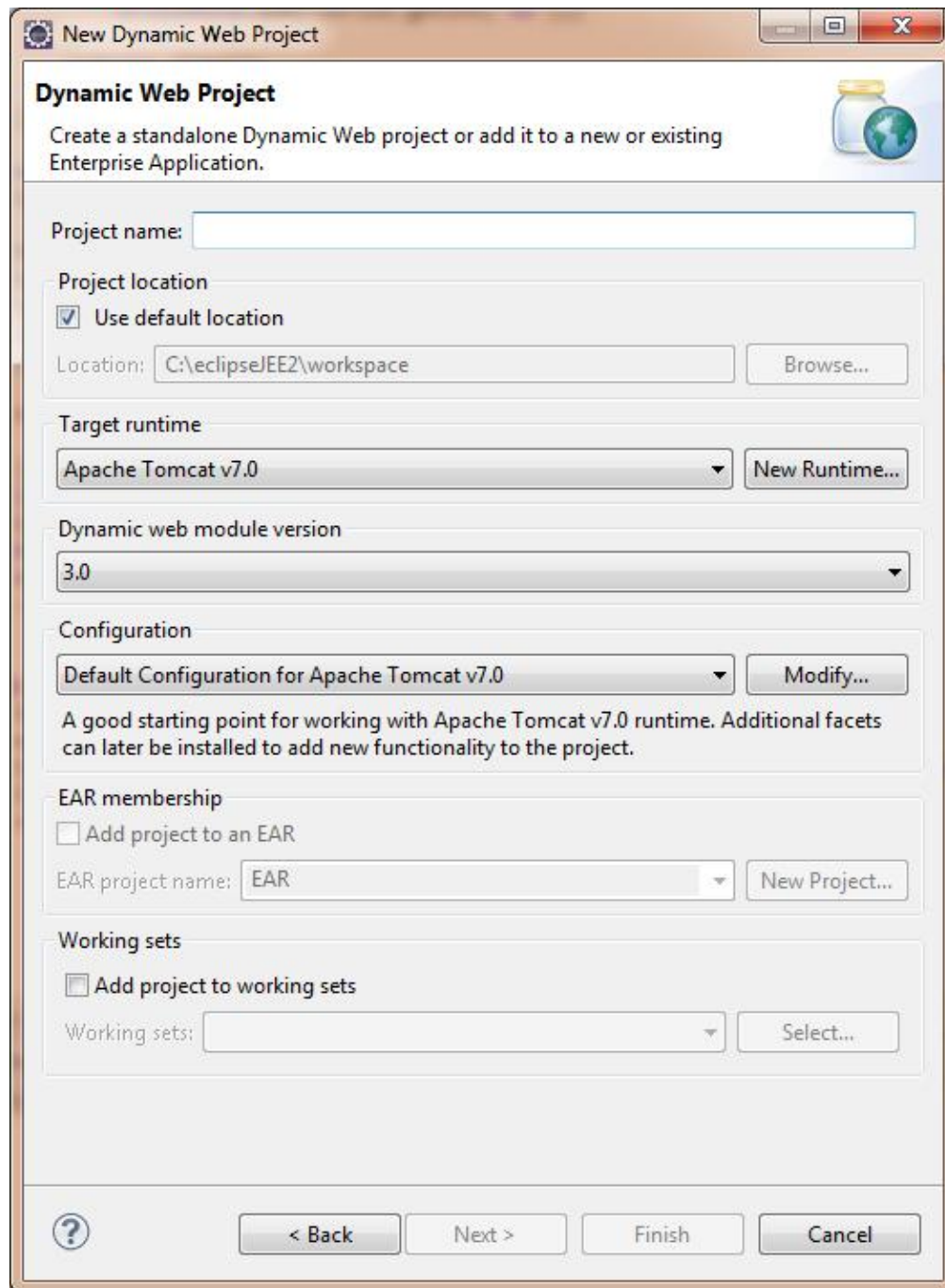


Figura 5.15. Registro del proyecto dinámico
Fuente: Propia

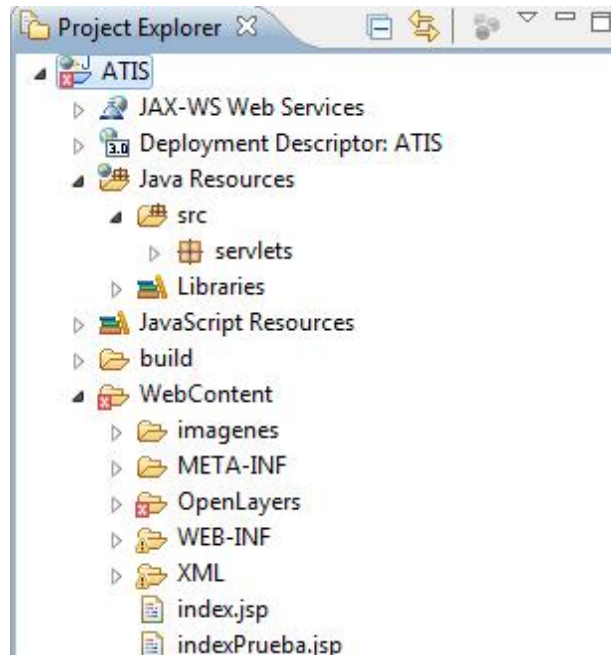


Figura 5.16. Estructura del proyecto java

Fuente: Propia

5.6. Clases programadas

El proyecto implementado, está pensado para usarse desde un browser (Chrome, FireFox, Explorer, Zafari, etc.), sin importar el sistema operativo que se tenga.

La página principal, desde donde se introducen los datos (origen, destino y número de transferencias máximo) es el archivo de nombre *index.jsp*, el cual manda llamar al archivo *SVHP_Servlet*, que a su vez transfiere los datos capturados hacia la clase principal, de nombre *SVHP*. El proyecto se divide en las siguientes clases:

- *SVHP_Servlet*
- *SVHP*
- *Nodo*
- *escribeNodo*
- *escribeltinerario*

La clase *SVHP_Servlet* se encarga de enviar los datos capturados en la página *index.jsp* y transferirlos hacia el algoritmo *SVHP*, asimismo, se encarga de crear la página web dinámica con los resultados entregados por la clase *SVHP* (para revisar el código correspondiente ver el Anexo D).

La clase *SVHP* está asociada al algoritmo del mismo nombre, el cual calcula las hiper rutas mínimas viables y multimodales, dados un origen y un destino. Como resultado entrega una lista de nodos y otra de arcos correspondientes a las hiper rutas calculadas para cada número de transferencias modales hasta llegar al máximo permitido, introducido por el usuario como dato inicial (para revisar el código correspondiente ver el Anexo E).

La clase *Nodo*, se encarga de generar objetos del mismo nombre y asociar a dicho objeto características como el identificador (ID), el nombre del nodo (nombre), costo del nodo para un cierto número de transferencias (costo), el número de transferencias acumuladas hasta la iteración actual (transferencias), una lista de nodos sucesores para poder recorrer todos los nodos que pertenecen a la hiper ruta desde el origen hasta el destino (sucesor), el estado del nodo para verificar si es viable la hiper ruta calculada (estado), etc. (para revisar el código correspondiente ver el Anexo F).

La clase *escribeNodo*, se utiliza para escribir en la página de resultados, el nombre de los puntos origen y destino. El dato que recibe la clase es el identificador numérico del nodo (ya sea el origen o el destino) y a cambio devuelve el nombre asociado a dicho nodo (para revisar el código correspondiente ver el Anexo G).

La clase *escribeltinerario*, se usa para escribir el itinerario a seguir para la hiper ruta calculada. Donde abordar/descender, transbordos, transferencias, nombres de las estaciones, modos de transporte empleados, líneas del modo abordado y direcciones de viaje dentro de la línea son escritas por esta clase para integrarse con el resto de la información a la página de resultados (para revisar el código correspondiente ver el Anexo H).

La clase *escribeLonLat*, se usa para dibujar en el mapa el origen y destino, dadas la longitud y latitud de ambos puntos (Anexo I).

Para realizar las pruebas iniciales del sistema y poder detectar los errores de codificación, se utilizó una hiperred de prueba. Con ello se pudieron diferenciar los errores de codificación, con respecto a los errores de construcción de la hiperred.

5.7. Incorporación de un servidor de mapas

Para presentar la ubicación de las estaciones de los distintos modos de transporte y la información asociada a las mismas, se utilizó la librería OpenLayers. Este conjunto de bibliotecas es gratuito, de código abierto, está escrito en Java Script y permite incorporar sobre los mapas, distintas capas hechas en formato kml (Figura 5.17). Los resultados también se despliegan de forma gráfica haciendo uso de esta biblioteca. Este proceso de despliegue ocurre en su totalidad en la computadora del usuario, ayudando a que las tareas realizadas en el servidor sean menores.

5.8. Caso de estudio del sistema SAIV

El sistema SAIV, abarca la zona de estudio descrita en la sección 1.4 y considera los sistemas de transporte público descritos en la sección 1.4.2, así como la red de transporte privado descrita en la sección 4.2 y se incorporaron al sistema los estacionamientos mostrados en la Figura 4.7.

La hiperred consta de 2151 arcos y 1244 nodos, considerando tanto al transporte público como a las vialidades para el automóvil.

La base de datos contiene las tablas de Nodos y Arcos, con 1244 y 2598 registros respectivamente. La tabla de Nodos contiene los campos ID, longitud y latitud, mientras que la tabla de Arcos, contiene los campos ID, length, dir, línea, modo, ruta, frec_h_pic, frec_h_val, t_d_rec, tipo, tiempoapro, form_id y to_id.

5.9. Manejo del SAIV por el usuario final

Para que el usuario pueda hacer uso del sistema, debe acceder desde una computadora con cualquier navegador Web, sin importar el sistema operativo.

En la página principal del sistema, el usuario debe introducir el origen, destino y el número de transferencias máximas, además puede conocer la ubicación de las estaciones de cada uno de los modos de transporte a través del mapa que se incluye en la misma página, como se muestra en la Figura 5.17.

5.10. Análisis de resultados

En esta sección se presentan una serie de hiperrutas mínimas viables multimodales obtenidas con el algoritmo SVHP (sección 3.6) para la zona de estudio descrita en la sección 1.4 y haciendo uso de los modos de transporte descritos en la sección 1.4.2.

Los resultados obtenidos para cada par origen-destino se obtienen hasta llegar al número de transferencias máximas establecidas de inicio por el usuario, obteniéndose un conjunto de hiperrutas mínimas conocidas como conjunto Pareto Optimal (conjunto de soluciones no dominadas, en función del tiempo y el número de transferencias). Dichos resultados se expresan de forma escrita y de forma gráfica a través de OpenLayers.

Los tiempos de las hiperredes de transporte público fueron tomados del trabajo de tesis realizado por Soto (2013), mientras que los tiempos de recorrido de la red vial para el automóvil fueron obtenidos utilizando la longitud de cada arco del archivo DBD de TransCad y suponiendo una velocidad promedio de 15 km/hr.

El tiempo de ejecución para un par origen-destino dado, es variable, alcanzando un máximo de diez segundos, lo cual es bastante bueno si se considera que el área de estudio es muy grande y que la ejecución del programa se realiza dentro del entorno de desarrollo de Eclipse. Las corridas fueron hechas en un par de computadoras, la primera con procesador Intel Core i3 a 3.1 GHz y 8 GB de RAM, las segunda con un procesador

Intel Core i5 a 2.3 GHz y la misma memoria RAM, obteniéndose tiempos similares de ejecución en ambas computadoras.

A continuación se presentan algunos ejemplos de resultados generados por el SAIV, cada ejemplo consta de una o más hiperrutas mínimas viables multimodales. Los ejemplos 5.10.1 y 5.10.2 incluyen únicamente transporte público, los ejemplos 5.10.3 y 5.10.4 incluyen al automóvil al inicio del recorrido y los ejemplos 5.10.5 y 5.10.6 incluyen al automóvil al final del recorrido.

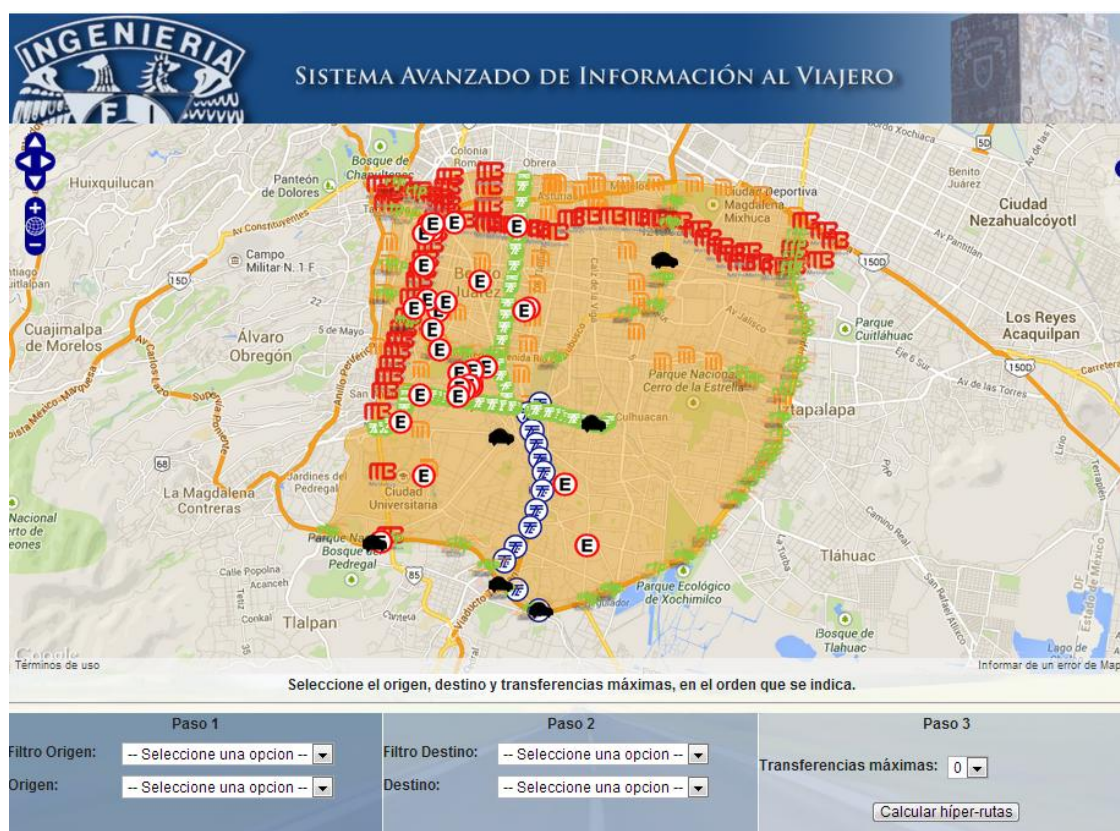


Figura 5.17. Página principal del sistema web

Fuente: Propia

5.10.1. Ejemplo 1. Metro Iztapalapa-Metrobús Poliforum

Para ir del Metro Iztapalapa al Metrobús Polifórum, el mejor tiempo es de 38 minutos realizando un solo cambio de modo; se obtiene una hiperruta considerando las 3 rutas del

Metrobús que pasan por la Línea 1 en dirección al Metrobús Polifórum y que se muestran como opciones escritas en color rojo en la Figura 5.18.

Las hiperrutas con dos y tres transferencias modales obtuvieron un tiempo mayor al ofrecido por la hiperruta con una transferencia, por lo tanto, no forman parte del conjunto Pareto Optimal y no se le presentan al usuario.



Figura 5.18. Hiperruta Metro Iztapalapa - Metrobús Polifórum

Fuente: Propia

5.10.2. Ejemplo 2. Metrobús Río frío-Tren ligero Periférico

Entre el origen Metro Río Frío y el destino Tren Ligero Periférico, se obtiene una hiperruta con dos transferencias y 73 minutos de recorrido, considerando las 3 rutas del Metrobús (Tepalcates-Del valle, Tepalcates-Etiopía y Tepalcates-Tacubaya) que pasan por la Línea 2 en dirección al Metrobús Xola, donde el usuario aborda el Metro en su Línea 2, para dirigirse a Tasqueña y finalmente cambiar al Tren Ligero para llegar a la estación Periférico, como se muestra en la Figura 5.19.

La hiperruta con tres transferencias modales obtuvo un tiempo mayor al ofrecido por la hiperruta con dos transferencias, por lo tanto, no forma parte del conjunto Pareto Optimal y no se le presenta al usuario.



Figura 5.19. Hiperruta Metrobús Río Frío - Tren Ligero Periférico

Fuente: Propia



Figura 5.20. Hiperruta Metrobús Río Frío - Tren Ligero Periférico (continuación)

Fuente: Propia

Los siguientes ejemplos muestran la combinación de automóvil con transporte público, cuando el auto es abordado al inicio del recorrido.

5.10.3. Ejemplo 3. Automóvil Periférico/Picacho-Metro Tepalcates

Entre el origen Periférico/Picacho (comenzando el recorrido en automóvil) y el destino Metro Tepalcates, se obtiene un conjunto Pareto Optimal con una y tres transferencias modales y con tiempos de 91 y 88 minutos respectivamente (Figura 5.20). La hiperruta con una transferencia modal, se muestra en color azul (tanto en el texto como en el mapa), mientras que la hiperruta con tres transferencias modales se muestra en color verde (en el texto y en el mapa).

La hiperruta con una transferencia modal y tiempo esperado de recorrido de 91 minutos, está compuesta por una sola ruta, formada por una subruta realizada en automóvil, comenzando el recorrido en la lateral de Periférico Sur en dirección de norte a sur, a la altura de la Carretera Picacho Ajusco, hasta el estacionamiento de la Tienda UNAM

(dibujado en color verde claro, Figura 5.20), la siguiente subruta, del Metro Universidad al Metro Puebla (dibujado el primer tramo en verde claro y el segundo en azul, Figura 5.20) y finalmente del Metro Puebla se debe caminar al Metro Agrícola Oriental (debido a que la estación Pantitlán está fuera del área de estudio y esto impide el transbordo de la Línea 9 a la Línea A) para terminar el recorrido en Metro Tepalcates (color verde oscuro).

La hiperruta con tres transferencias modales y tiempo esperado de recorrido de 88 minutos, está compuesta por una hiperruta que considera tres opciones (Del Valle-Tepalcates, Tacubaya-Tepalcates y Etiopía-Tepalcates, mostradas en color verde oscuro en la Figura 5.20). Se realiza un viaje mixto comenzando en automóvil desde la lateral de Periférico Sur en dirección de norte a sur, a la altura de la Carretera Picacho Ajusco (verde claro, Figura 5.20), el trayecto en auto culmina cuando el usuario deja su automóvil en el estacionamiento de la Tiena UNAM, posteriormente continua el recorrido en Metro (verde claro, Figura 5.20), iniciando en Metro Universidad y descendiendo en Metro Etiopía, después hay otro cambio de modo al abordar el Metrobús en la Línea 2, estación Etiopía y culmina el trayecto en Metrobús al descender en la estación Tepalcates (verde oscuro, Figura 5.20). El último tramo de viaje es un arco peatonal que lleva al usuario del Metrobús Tepalcates a la estación del Metro del mismo nombre.

En ambas hiperrutas (una y tres transferencias modales), se verifica la viabilidad para incluir al automóvil (modo restringido).



Origen: Periférico Sur N/S, Destino: Tepalcates, Transferencias máximas: 3

Con 1 transferencia (cambio de modo).

Tiempo estimado: 91.0 minutos.

Itinerario:

En el automóvil, iniciar el recorrido en periférico sur dirección N/S.

Posteriormente circular por Insurgentes en dirección Norte.

Tomar el Circuito Mario de la Cueva.

Dejar el coche en la Tienda UNAM.

CAMBIAR DE MODO DE TRANSPORTE PRIVADO A PÚBLICO.

En Universidad, tomar la línea 3 en dirección a Indios Verdes.

Descender en metro Centro Médico.

TRANSBORDAR de la línea 3 a la línea 9 del metro.

En Centro Médico, tomar la línea 9 en dirección a Pantitlán.

Descender en metro Puebla.

CAMINAR.

En Agrícola Oriental, tomar la línea A en dirección a La Paz.

Descender en metro Tepalcates.

Con 3 transferencia (cambio de modo).

Tiempo estimado: 88.0 minutos.

Itinerario:

En el automóvil, iniciar el recorrido en periférico sur dirección N/S.

Posteriormente circular por Insurgentes en dirección Norte.

Tomar el Circuito Mario de la Cueva.

Dejar el coche en la Tienda UNAM.

CAMBIAR DE MODO DE TRANSPORTE PRIVADO A PÚBLICO.

En Universidad, tomar la línea 3 en dirección a Indios Verdes.

Descender en metro Etiopía.

CAMBIAR DE MODO DE TRANSPORTE.

Abordar el metrobús en la estación Etiopía. Línea 2 Col. Del Valle-Tepalcates.

Descender en la estación del metrobús Tepalcates.

CAMBIAR DE MODO DE TRANSPORTE.

Otra opción:

Abordar el metrobús en la estación Etiopía. Línea 2 Tacubaya-Tepalcates.

Descender en la estación del metrobús Tepalcates.

CAMBIAR DE MODO DE TRANSPORTE.

Otra opción:

Abordar el metrobús en la estación Etiopía. Línea 2 Etiopía-Tepalcates.

Descender en la estación del metrobús Tepalcates.

CAMBIAR DE MODO DE TRANSPORTE.

Figura 5.21. Hiperruta Automóvil Periférico Sur N/S - Metro Tepalcates

Fuente: Propia

5.10.4. Ejemplo 4. Automóvil Eje 10Sur-RTP Centenario

Para ir del Eje 10 Sur y Av. Aztecas en automóvil a la estación Centenario del sistema de transporte RTP en dirección Oriente/Poniente, se obtiene una hiperruta con 2 transferencias modales y 39 minutos de recorrido (Figura 5.21). El automóvil se usa al inicio del trayecto y se estaciona en la Tienda UNAM para abordar el metro en la estación Universidad en dirección a Indios Verdes. Posteriormente se hace un nuevo cambio de modo al descender en la estación Coyoacán del Metro y abordar el RTP en Avenida Universidad en dirección O/P, finalmente continúa el recorrido hasta llegar a la estación Centenario O/P.

La hiperruta con tres transferencias modales obtuvo un tiempo mayor al ofrecido por la hiperruta con dos transferencias, por lo tanto, no forma parte del conjunto Pareto Optimal y no se le presenta al usuario.



Figura 5.22. Hiperruta Automóvil Eje 10 Sur E/O - RTP Centenario O/P

Fuente: Propia

Con 2 transferencia (cambio de modo).
Tiempo estimado: 39.0 minutos.

Itinerario:

En el automóvil, iniciar el recorrido en Eje 10 Sur.
Continuar sobre eje 10 Sur y dar vuelta a la izquierda en Delfín Madrigal.
Dejar el coche en la Tienda UNAM.
CAMBIAR DE MODO DE TRANSPORTE PRIVADO A PÚBLICO.
En Universidad, tomar la línea 3 en dirección a Indios Verdes.
Descender en metro Coyoacán.
CAMBIAR DE MODO DE TRANSPORTE.
Abordar el RTP en la estación Av. Universidad O/P.
Descender en la estación Centenario O/P.

Figura 5.23. Hiperruta Automóvil Eje 10 Sur E/O - RTP Centenario O/P (continuación)

Fuente: Propia

En las secciones 5.9.5 y 5.9.6, se considera el uso del automóvil al final del recorrido, iniciando éste en transporte público.

5.10.5. Ejemplo 5. Metro Lázaro Cárdenas-Automóvil Periférico Sur

Entre el origen Metro Lázaro-Cárdenas y el destino en Periférico Sur, comienza el recorrido en transporte público (Línea 9 del Metro, estación Lázaro Cárdenas) y termina en automóvil en la lateral de Periférico en dirección S/N (Figura 5.22). Al ser un viaje mixto e incorporar el modo con restricciones (automóvil), también se verifica la viabilidad, como se mostró en el diagrama de estados de la sección 4.3. En la figura 5.22, se muestra el recorrido completo para el viaje mixto, donde la última parte del viaje se realiza en automóvil, partiendo del estacionamiento de la tienda UNAM, para llegar a la lateral de Periférico Sur en sentido S/N.

Las hiperrutas con dos y tres transferencias modales obtuvieron un tiempo mayor al ofrecido por la hiperruta con una transferencia, por lo tanto, no forman parte del conjunto Pareto Optimal y no se le presentan al usuario.



Figura 5.24. Hiperruta Metro Lázaro Cárdenas – Automóvil Periférico Sur S/N

Fuente: Propia

5.10.6. Ejemplo 6. Metrobús Amores-Automóvil Circuito Interior

Entre el Metrobús Amores y el Circuito Interior, se obtienen dos hiperrutas iniciales para el Metrobús, en la estación Amores de la Línea 2, el usuario puede abordar la ruta Del Valle-Tepalcates o Tacubaya-Tepalcates, si el usuario toma la primera ruta llega directamente a la estación Xola del Metrobús, en caso de haber elegido la segunda ruta, el viajero tiene que descender en Etiopía y en esta parada esperar a que pasen las rutas Del valle-Tepalcates o Etiopía-Tepalcates. Para este trayecto en Metrobús, se presentan dos hiperrutas iniciales y de una de esas hiperrutas, se desprenden otras dos (Figura 5.23).

Posteriormente el camino es único y consiste en abordar el Metro en la estación Xola y dirigirse a Tasqueña, para descender en Portales, después hay que caminar para recoger el automóvil del estacionamiento ubicado en Balboa 202 y poder culminar el trayecto en coche, hasta llegar al destino localizado en la intersección de Circuito Interior y Av. Canal de Tezontle (Figura 5.23).

La hiperruta con tres transferencias modales obtuvo un tiempo mayor al ofrecido por la hiperruta con dos transferencias, por lo tanto, no forma parte del conjunto Pareto Optimal y no se le presenta al usuario.



Origen: Amores, Destino: Circuito Interior S/N, Transferencias máximas: 3

Con 2 transferencia (cambio de modo).
 Tiempo estimado: 54.0 minutos.

Itinerario:

Abordar el metrobús en la estación Amores. Línea 2 Col. Del Valle-Tepalcates.

Descender en la estación del metrobús Xola.

CAMBIAR DE MODO DE TRANSPORTE.

En Xola, tomar la línea 2 en dirección a Tasqueña.

Descender en metro Portales.

CAMBIAR DE MODO DE TRANSPORTE PÚBLICO A PRIVADO.

Abordar el automóvil en el estacionamiento de Balboa 202, Colonia Portales.

Incorporarse a Calzada de Tlalpan en dirección N/S

Incorporarse a Circuito interior en dirección S/N Descender en Circuito Interior dirección S/N.

Otra opción:

Abordar el metrobús en la estación Amores. Línea 2 Tacubaya-Tepalcates.

Descender en la estación del metrobús Etiopía.

Abordar el metrobús en la estación Etiopía. Línea 2 Col. Del Valle-Tepalcates.

Descender en la estación del metrobús Xola.

CAMBIAR DE MODO DE TRANSPORTE.

En Xola, tomar la línea 2 en dirección a Tasqueña.

Descender en metro Portales.

CAMBIAR DE MODO DE TRANSPORTE PÚBLICO A PRIVADO.

Abordar el automóvil en el estacionamiento de Balboa 202, Colonia Portales.

Incorporarse a Calzada de Tlalpan en dirección N/S

Incorporarse a Circuito interior en dirección S/N Descender en Circuito Interior dirección S/N.

Otra opción:

Abordar el metrobús en la estación Etiopía. Línea 2 Etiopía-Tepalcates.

Descender en la estación del metrobús Xola.

CAMBIAR DE MODO DE TRANSPORTE.

En Xola, tomar la línea 2 en dirección a Tasqueña.

Descender en metro Portales.

CAMBIAR DE MODO DE TRANSPORTE PÚBLICO A PRIVADO.

Abordar el automóvil en el estacionamiento de Balboa 202, Colonia Portales.

Incorporarse a Calzada de Tlalpan en dirección N/S

Incorporarse a Circuito interior en dirección S/N Descender en Circuito Interior dirección S/N.

Figura 5.25. Hiperruta Metrobús Amores – Automóvil Circuito Interior S/N

Fuente: Propia

A través de los ejemplos anteriormente descritos, se observa que es factible la realización de un Sistema Web multiusuario, que utilice como base el algoritmo SVHP modificado para la zona de estudio, que permita al usuario observar los resultados a través de texto y de forma gráfica, siendo cercanos a la realidad los tiempos calculados y ofreciendo un conjunto de posibilidades al usuario, en una red multimodal, en lugar de una sola ruta monomodal, como lo ofrecen otros programas.

Conclusiones y recomendaciones

A través del SAIV diseñado, desarrollado e implementado, los viajeros pueden conocer las distintas maneras de llegar a su destino, haciendo uso del transporte público y del automóvil. Dado que el transporte público en la Zona Metropolitana del Valle de México (ZMVM) carece de horarios fijos de salida y llegada a cada estación, es necesario que se consideren los tiempos de espera de los usuarios en la parada donde se ubican, razón por la cual fueron utilizadas hiperrutas mínimas en lugar de hacer un simple cálculo de ruta mínima.

Hacer un sistema de este tipo es relativamente económico, en comparación con los recursos requeridos para otros SIT, donde se requiere la instalación de equipo costoso. Sin embargo, si se combina este SAIV con cámaras, sensores, etc. se podría tener la información de los tiempos de viaje y espera en tiempo real, lo que representaría un gran beneficio para los viajeros.

La realización de trabajos de tesis como éste es bastante laboriosa, sobre todo si consideramos la zona de estudio utilizada para el presente trabajo, al ser una zona grande, se dificulta la detección de errores, ya sea por codificación o en la construcción de la hiperrred. Si se generan estándares para la construcción de las hiperrredes y de las bases de datos asociadas a las mismas, podría desarrollarse en un futuro próximo un sistema SAIV para la ZMVM.

La información referente a los tiempos de viaje para los distintos modos de transporte público es la información oficial, sin embargo, no es muy exacta y esto impacta obviamente en los resultados obtenidos con el algoritmo SVHP.

Los tiempos de procesamiento del algoritmo SVHP son relativamente pequeños, debido al tipo de estructuras de datos utilizadas, con ello, se requiere un número menor de

conexiones a la base de datos y se reduce el tiempo de ejecución del algoritmo, no obstante se tendrán que mejorar otros procesos para que se disminuyan los tiempos totales en que el sistema entrega la información al usuario, ya que el sistema atenderá múltiples peticiones de manera simultánea.

A futuro deberá considerarse la interacción con el mapa para poder seleccionar el origen y el destino, de esa manera, se mejorará el sistema, que actualmente presenta los resultados de forma gráfica en el mapa y de forma escrita para indicar el itinerario de viaje. La manera en la que se dibujan los resultados, haciendo uso únicamente de OpenLayers, presenta una mejora significativa en los tiempos de procesamiento en el servidor, ya que no es necesario realizar una nueva petición de datos para que los resultados sean dibujados.

Como trabajo a futuro, se deberán verificar los requerimientos de hardware para el sistema en producción, tomando en cuenta el tamaño de la red, el número de nodos, y su posible expansión. Con ello se podrá estimar el número de usuarios que el SAIV atenderá de manera simultánea y se podrán identificar cuellos de botella y oportunidades de mejorar, mediante pruebas técnicas de estrés, carga y rendimiento (Stress, load and performance testing).

También se deberán validar los datos de entrada, para verificar que la información introducida sea coherente (evitar que el origen y el destino sean iguales, deshabilitar el botón 'Calcular' cuando no se han introducido datos, etc.) y se deberán corregir los problemas detectados en la hipertexto construida y los de codificación en el sistema, que resulten de las pruebas correspondientes.

Una de las grandes ventajas de planear el viaje, es saber cuáles caminos son más convenientes antes de iniciar el recorrido, con ello el usuario ahorra tiempo, conoce el número de cambios de modo que debe realizar y la manera en que éstos se llevan a

cabo, facilita la realización de viajes mixtos, evita que el automóvil se utilice de forma irracional y como consecuencia ayuda a disminuir los daños ambientales.

El número de viajes realizados en automóvil es muy elevado y el número de personas por vehículo es muy bajo, recorriendo distancias muy largas y con tiempos de viaje muy elevados; ante este problema el uso del SAIV permitirá a los usuarios disminuir las distancias recorridas en automóvil, ya que el usuario podrá saber cómo combinar su viaje en auto con el transporte público.

Varios motivos influyen para que actualmente este tipo de viajes mixtos (público-privado) no se lleven a cabo con la frecuencia que debieran realizarse, principalmente esto se debe a que los estacionamientos en la ZMVM no son suficientes, tampoco cuentan con la capacidad adecuada, ni con costos adecuados (reducción de costo por estancias largas de todo el día), además el personal que labora en ellos tampoco es confiable, y las empresas encargadas del manejo de los estacionamientos no se responsabilizan por el robo de pertenencias dentro de los vehículos, o por daños en el exterior del mismo.

Si se habilitaran los estacionamientos actualmente subutilizados como el ubicado en la cercanía de Metro Universidad (que se utiliza como corralón para taxis), y se ampliaran y remodelaran otros estacionamientos existentes, se incentivaría la realización de viajes público-privados, con los beneficios ya mencionados.

Será labor del gobierno del Distrito Federal y del Estado de México, habilitar los estacionamientos bajo su administración así como, construir nuevos estacionamientos, y motivar a los particulares que sean dueños de estacionamientos para que el usuario de automóvil pueda dejar su coche en un estacionamiento seguro y continuar su recorrido en transporte público.

El objetivo de la presente tesis fue alcanzado con éxito, ya que el sistema web puede obtener las hiper rutas mínimas multimodales y además verificar la viabilidad de las hiper rutas obtenidas al incorporar un modo de transporte restringido, como lo es el automóvil, sin embargo, muchas mejoras podrán ser implementadas en un futuro cercano para mejorar el sistema y para que éste sea publicado y utilizado por la gente.

Referencias

Acha, J. A. & Espinoza, J. C. 2004. 'Hacia una arquitectura nacional para los sistemas inteligentes de transporte'. Secretaría de Comunicaciones y Transportes. Instituto Mexicano del Transporte, Publicación técnica No. 251.

Adler, J. L. & Blue, V. J. 1998. Toward the design of intelligent traveler information systems. Elsevier Science Ltd., Transportation Research Part C (6), 157-172.

Ahuja, R., Magnanti, T. & Orlin, J. 1993. Network Flows: Theory, Algorithms, and Applications, Prentice Hall, United States.

Cisnéros, O. 2009. 'Los sistemas anticolidión (CAS, Collision Avoidance Systems)'. E-Safety: Nuevas tecnologías al servicio de la seguridad vial, No 41. Julio-Septiembre 2009, consultado: 2012, http://www.centro-zaragoza.com:8080/web/sala_prensa/revista_tecnica/hemeroteca/articulos/R41_A7.pdf.

Daza, J. A. & De la Torre, M. E. 2006. 'Identificación del conocimiento y necesidades de los servicios de ITS por parte de las empresas transportistas y con flota propia en México'. Secretaría de Comunicaciones y Transportes. Instituto Mexicano del Transporte. Publicación técnica No. 289.

Departamento de Transporte, Reino Unido 2010, *Passenger Information systems* [Página web], consultado en 2012, <http://www.dft.gov.uk/itstoolkit/Tools/T20.php>

El Sol de México 2012. 'Obsoletas mas de 2º mil unidades del transporte público' [en línea], *El Sol de México*, 8 de mayo de 2012, consultado en 2013, <http://www.oem.com.mx/eloccidental/notas/n2533319.htm>

ERTICO, ITS Europe. *ITS can improve our daily lives* [en línea], consultado en 2012, <http://www.ertico.com/about-ertico-its/>

Federal Signal Corporation, 2009. Case studies, Intelligent Transportation Systems.

http://pipstechnology.com/images/uploads/library/Case_Study_-_Intelligent_Transportation.pdf

Fideicomiso para el Mejoramiento de las Vías de Comunicación del Distrito Federal (FIMEVIC). *Diagnóstico de la movilidad en la Ciudad de México* [en línea], consultado en 2011, <http://www.fimevic.df.gob.mx/problemas/1diagnostico.htm>

Gandy, R. & Meitner, M. J. 2008. 'Advanced traveler information systems: Bringing environmental information to recreational driving'. *Elsevier Science Ltd.*, Landscape and urban planning, 185-194.

Instituto Nacional de Estadística Geografía e Informática (INEGI) 2005. *Densidad de población* [en línea], consultado en 2012, <http://cuentame.inegi.gob.mx/poblacion/densidad.aspx?tema=p>

Instituto Nacional de Estadística Geografía e Informática (INEGI) 2010. *Vehículos de motor registrados en circulación* [en línea], consultado en 2011, http://www.inegi.org.mx/est/lista_cubos/consulta.aspx?p=adm&c=8

Instituto Nacional de Estadística Geografía e Informática (INEGI) 2010. *Censo de población y vivienda 2010* [en línea], consultado en 2011, <http://www.inegi.org.mx/est/contenidos/proyectos/ccpv/cpv2010/Default.aspx>

Instituto Nacional de Estadística Geografía e Informática, Gobierno del Distrito Federal, Gobierno del Estado de México 2007. *Encuesta 2007 Origen-Destino (EOD)* [en línea], consultado en 2011, http://inigo.bicitekas.org/wp-content/uploads/2013/07/2007_Encuesta_Origen_Destino_INEGI.pdf

Intelligent Transportation Systems Society of Canada (ITS Canada) 2012. *Intelligent transportation* [en línea], consultado en 2012, <http://www.itscanada.ca/it/index.html>

Instituto de Políticas para el Transporte y el Desarrollo (ITDP) 2012. *Reforma urbana* [en línea], consultado en 2012, <http://mexico.itdp.org/noticias/reformaurbana/>

ITS México Sistemas Inteligentes de Transporte, A.C. [en línea], consultado en 2012, <http://www.itsmexico.org.mx/page1.aspx>

ITS World Congress [en línea], consultado en 2012, <http://itsworldcongress.org/>,

Lobo, A. Transporte público sustentable: Movilidad con calidad de vida [en línea], consultado en 2012, [http://www.circuloverde.com.mx/es/uploads/1/8._transporte_sustentable.pdf](http://www.circuloverde.com.mx/es/uploads/1/8/_transporte_sustentable.pdf)

López, D. 2010. 'Híper rutas más cortas con respecto del tiempo entre las instalaciones dentro de Ciudad Universitaria: Un modelo de transporte multimodal' Maestría en Ingeniería. Universidad Nacional Autónoma de México.

Lupano, J. A. & Sánchez, R. J. 2009. *Políticas de movilidad urbana e infraestructura urbana de transporte* [en línea], consultado en 2011, <http://www.eclac.org/publicaciones/xml/2/35492/lcw230e.pdf>

Lozano, A. & Storchi, G. 2001. 'Shortest viable path in multimodal networks'. *Elsevier Science Ltd.*, Transportation research A 35 (3), 225-241.

Lozano, A. & Storchi, G. 2002. Shortest viable hyperpath in multimodal networks. *Elsevier Science Ltd.*, Transportation Research Part B 36, 853-874.

Lozano A., Antún J.P., Granados F. 2005. *El transporte particular y los problemas de congestión. En El reto del transporte en la ciudad de México: voces, ideas y propuestas.* Edamex, México.

Maps of world 2014. *Top ten largest cities map* [en línea], consultado en 2014, <http://www.mapsofworld.com/world-top-ten/world-top-ten-largest-cities-map.html>

Mercado, L. E. 2006. 'Motor para toma de acciones preventivas a partir de recolección de datos simulados para el control automatizado de una autopista' Universidad de las Américas Puebla.

Metrobús. *Rutas de servicio* [en línea], consultado en 2012, <http://www.metrobus.df.gob.mx/rutas.html>

Mora, K. 2012. 'Urgen especialistas cambio de imagen de tren ligero' [en línea], *El Universal*, 22 de mayo de 2012, consultado en 2014, <http://www.eluniversaldf.mx/home/nota46243.html>

Munich Airport [en línea], consultado en 2014, <http://www.munich-airport.de/en/consumer/erlebnis/airporttour/index.jsp>

Nelson, Donna C. 2000. *Intelligent Transportation Primer*. Institute of Transportation Engineers, Washington, D.C. pp. 1-9.

Programa Universitario de Estudios sobre la Ciudad - Universidad Nacional Autónoma de México 2011, *Evaluación del diseño e instrumentación de la política de transporte público colectivo de pasajeros en el Distrito Federal* [en línea], consultado en 2012, http://www.evalua.df.gob.mx/files/recomendaciones/evaluaciones_finales/ev_transp.pdf

Red de Transporte de Pasajeros del D.F. (RTP), *Red de rutas* [en línea], consultado en 2012, http://www.rtp.gob.mx/red_rutas.html

Secretaría de Desarrollo Urbano y Vivienda. *Programas Delegacionales de Desarrollo Urbano* [en línea], consultado en 2012, <http://www.seduvi.df.gob.mx/portal/index.php/programas-de-desarrollo/programas-delegacionales>

Secretaría de Estado de Transportes del gobierno de España 2010. Los Sistemas Inteligentes de Transporte. Su aplicación a los modos terrestre, marítimo y aéreo. Vol. 2 Trabajos de prospectiva y tecnología del transporte.

Secretaría de Transporte y Vialidad del Distrito Federal (SETRAVI). *Estadísticas* [en línea], consultado en 2012, <http://www.semovi.df.gob.mx/wb/stv/estadisticas.html>

Secretaría del Medio Ambiente (SMA). *Proyecto de introducción de medidas ambientales amigables al transporte del año 2000 al 2007* [en línea], consultado en 2012, <http://www.sma.df.gob.mx/transportesustentable/>

Siemens AG, 2007. *SIPARK SSD car park guidance system. A professional solution designed to industrial standard* [en línea], consultado en 2012, <http://www.mobility.siemens.com/mobility/global/Documents/en/road-solutions/parking-management/sipark-ssd-car-park-guidance-system-en.pdf>

Sistema de Monitoreo Atmosférico 2014. *Gráficos. Cumplimiento de las Normas Oficiales Mexicanas para O3* [en línea], consultado en 2014, <http://www.aire.df.gob.mx/default.php?opc=%27aqBhnmOkZA==%27>

Sistema de Monitoreo Atmosférico 2014. *Gráficos. Frecuencia anual de días con concentraciones por arriba del valor límite de O3* [en línea], consultado en 2014, <http://www.aire.df.gob.mx/default.php?opc=%27aqBhnmOkZA==%27>

Sistema de Transporte Colectivo Metro (STC). *Datos de operación* [en línea], consultado en 2012, <http://www.metro.df.gob.mx/operacion/index.html>

Sistema de Transporte Eléctrico (STE), *Servicios* [en línea], consultado en 2012, <http://www.ste.df.gob.mx/servicios/>

Soto, M. 2013. 'Modelación de hiper rutas mínimas multimodales del transporte público de pasajeros: zona centro sur del DF' Maestría en Ingeniería. Universidad Nacional Autónoma de México.

SUSSMAN, J. 2000. *Introduction to Transportation Systems*, Norwood, MA, Artech House Publisher.

Taylor, M.A.P. 2001, 'Intelligent Transport Systems', en K.J. Button y D.A. Hensher, *Handbook of Transport Systems and Traffic Control*, Elsevier Science Ltd.

Torsten H. 2011, *In time NYC now displays real-time data for MTA's bus line B63* [página web], consultado en 2012,
http://www.tntbrain.com/Home/Intime_B63_%28B63_bus_route_in_Brooklyn%29.html

Toyota. *History of Toyota's ITS activities* [página web], consultado en 2012.
http://www.toyota-global.com/innovation/intelligent_transport_systems/toyota_ap/history/

Unidad operativa de control de tránsito, Santiago de Chile (UOCT), Ministerio de Transporte y Telecomunicaciones, 2010. *Sistemas complementarios y de apoyo a la gestión de tránsito* [en línea], consultado en 2012,
<http://www.uoct.cl/sistemas-complementarios-y-de-apoyo-a-la-gestion-de-transito/>

U.S. Department of Transportation. Federal Highway Administration, 1996. *Development of Human Factors Guidelines for Advanced Traveler Information Systems and Commercial Vehicle Operations: Comparable Systems Analysis* [página web], consultado en 2012,
http://www.fhwa.dot.gov/publications/research/safety/95197/sec1/sec1_01.cfm

Anexos

A. Estacionamientos en la zona de estudio

Estacionamientos públicos en Coyoacán

1. Av. Miguel Hidalgo No. 102
2. Xicoténcatl No. 195 Esq. Corina
3. Xicoténcatl 287. Del Carmen, Coyoacán
4. Xicoténcatl 401. Del Carmen, Coyoacán
5. Higuera No. 16
6. Carranza No. 106
7. Tres cruces No. 105
8. Calzada del hueso, Ex hacienda Coapa. Galerías Coapa
9. Periférico Sur 4690. (Centro comercial PeriSur) *
10. Av. Universidad 1992. Estacionamiento Boliche y Hermanos Vázquez
11. Eje 1 Oriente, Canal de Miramontes. (Centro comercial Zapamundi)
12. Miguel Ángel de Quevedo 282
13. Estacionamientos en Ciudad Universitaria *
14. Estacionamiento Walmart/Suburbia/Vips Miguel Ángel de Quevedo *
15. Estacionamiento Toks Tasqueña *

Estacionamientos públicos en Benito Juárez

1. Av. Coyoacán 1434 (esq. Félix Cuevas)
2. Insurgentes Sur 975 (esq. California), Cd. de los deportes
3. Reforma 44, San Simón Ticumac
4. Rodríguez Saro 431, Del Valle
5. Patricio Sanz 1523, Tlacoquemecatl
6. Dakota esq. Altadena, Col. Nápoles
7. Balboa 202, Portales *
8. Amores 1402, Del Valle
9. Parroquia, Galerías Insurgentes
10. Casas grandes 143, Narvarte
11. Adolfo Prieto 216, Del Valle
12. Centro Coyoacán. Av. Coyoacán 2000, Xoco
13. Montecito 35. World Trade Center México
14. Dakota y Filadelfia. World Trade Center, Nápoles
15. Tenayuca 42, Letrán Valle, Benito Juárez

Estacionamientos Públicos en Iztacalco.

1. Estacionamientos foro Sol/Palacio de los Deportes

Estacionamientos públicos en Iztapalapa.

1. Canal de Tezontle. Plaza Oriente
2. Eje 4 oriente Av. Río Churubusco. Plaza Central

* Estacionamientos utilizados en el SAIV

B. JDBC para estructuras de datos

```

package connpostgresql;
import java.sql.*;

public class ConnPostgreSQL {
    static int i=1; //Indice de la tabla original
    static int j=1; //Indice de la "tabla copia"
    static int numElementos = 5000; //Número de registros de la base de datos
    //Declaracion de arreglos para hacer la copia de la tabla
    static int identificador[] = new int[numElementos]; //...de tamaño 1657...
    static float longitud[] = new float[numElementos]; //De tamaño 1657
    static int direccion[] = new int[numElementos];
    static String linea[] = new String[numElementos];
    static String modo[] = new String[numElementos];
    static String ruta[] = new String[numElementos];
    static float frecHpico[] = new float[numElementos];
    static float frecHvalle[] = new float[numElementos];
    static float tiempor[] = new float[numElementos];
    static String tipo[] = new String[numElementos];
    static float tiempoa[] = new float[numElementos];
    static int ccstyle[] = new int[numElementos];
    static int fromID[] = new int[numElementos];
    static int toID[] = new int[numElementos];
    static int formula[] = new int[numElementos];
    //Variables para recuperar los datos almacenados en la BD
    static int ID;
    static float LONGITUD;
    static int DIR;
    static String LINEA; //Id de linea
    static String MODO;
    static String RUTA; //Id de ruta
    static float FrechPico;
    static float FrechVal;
    static float TRec;
    static String TIPO; //ARCO O H-ARCO A, H, P (arco id conex peatonales)
    static float TApprox; //Calculo conectividad en Transcad
    static int CCSTYLE; //Colorcitos
    static int FROM_ID; //Nodo cola
    static int TO_ID; //Nodo cabeza
    static int FORMULA; //Campo de Minerva

    public static void main(String[] argv) throws SQLException {
        System.out.println("----- PostgreSQL "
            + "Prueba de conexion con JDBC -----");
        try {
            Class.forName("org.postgresql.Driver");
        }
        catch (ClassNotFoundException e)
        {
            System.out.println("Donde esta el driver JDBC para PostgreSQL? "
                + "Incluir en la libreria");
            return;
        }
    }
}

```

```

        System.out.println("PostgreSQL JDBC driver registrado!");
    Connection con = null;

    try {
        con = DriverManager.getConnection(
            "jdbc:postgresql://localhost:5432/postgres", "postgres",
            "admin");
    }
    catch (SQLException e) {
        System.out.println("Falla en la conexion! Verificar consola de salida");
        return;
    }
    if (con != null) {
        System.out.println("Conexion realizada con exito!");
        //Despues de conectarse borra los datos pre existentes en las tablas
        //para garantizar que al ejecutarse una vez, se tengan los datos y estructuras correctos en la BD
        PreparedStatement borra = con.prepareStatement("DELETE FROM hrmm2;");
        borra.executeUpdate();
        borra = con.prepareStatement("DELETE FROM entrante;");
        borra.executeUpdate();
        borra = con.prepareStatement("DELETE FROM saliente;");
        borra.executeUpdate();
        borra = con.prepareStatement("DELETE FROM apuntador;");
        borra.executeUpdate();
        borra = con.prepareStatement("DELETE FROM rapuntador;");
        borra.executeUpdate();
    }
    else {
        System.out.println("Falla en la conexion!");
    }

    /*******
try(Statement stmt = con.createStatement()){
    ResultSet rs = stmt.executeQuery("SELECT * FROM hrmm;");
    while (rs.next()){
        //Obtiene los valores de la BD y los asigna a las correspondientes variables
        ID = rs.getInt("ID");
        LONGITUD = rs.getFloat("LENGTH");
        DIR = rs.getInt("Dir");
        LINEA = rs.getString("LINEA");
        MODO = rs.getString("MODO");
        RUTA = rs.getString("RUTA");
        FrechHPico = rs.getFloat("FrechHPico");
        FrechHVal = rs.getFloat("FrechHVal");
        TRec = rs.getFloat("TRec");
        TIPO = rs.getString("TIPO");
        TAprox = rs.getFloat("TAprox");
        FROM_ID = rs.getInt("FROM_ID");
        TO_ID = rs.getInt("TO_ID");

        //Asignacion de los valores de la BD a los arreglos
        switch(DIR) {
            case 1: registros2arreglos(j);

```

```

        j=j+1;
        break;
    case -1: registros2arreglos(j);
        j=j+1;
        break;
    case 0: registros2arreglos(j);
        j=j+2;
        break;
}
i= i + 1; //Aumento en el indice de la BD
}
stmt.close();
}
catch ( Exception e ){
    System.out.println(e.getMessage());
}

//*****
try{ //Escritura en la tabla copia "hrmm2" ya sin los valores -1 y 0 para el campo DIR
    PreparedStatement escribe = con.prepareStatement("INSERT INTO hrmm2 VALUES
(?,?,?,?,?,?,?,?,?,?,?,?,?);");
    int k = 1;
    for (k=1; k<j; k++) {
        escribe.setInt(1, identificador[k]);
        escribe.setFloat(2, longitud[k]);
        escribe.setInt(3, direccion[k]);
        escribe.setString(4, linea[k]);
        escribe.setString(5, modo[k]);
        escribe.setString(6, ruta[k]);
        escribe.setFloat(7, frecHpico[k]);
        escribe.setFloat(8, frecHvalla[k]);
        escribe.setFloat(9, tiempo[k]);
        escribe.setString(10, tipo[k]);
        escribe.setFloat(11, tiempoa[k]);
        //escribe.setInt(12, ccstyle[k]);
        escribe.setInt(12, fromID[k]);
        escribe.setInt(13, toID[k]);
        //escribe.setInt(15, formula[k]);
        escribe.executeUpdate();
    }
}
catch(Exception e) {
    System.out.println(e.getMessage());
}

//*****
try(Statement stmt2 = con.createStatement()){
    j=1;
    ResultSet ee = stmt2.executeQuery("SELECT * FROM hrmm2 order by \"TO_ID\";"); //Estrella
entrante
    while (ee.next()){

```

```

//Obtiene los valores de la BD y los asigna a las correspondientes variables
ID = ee.getInt("ID");
LONGITUD = ee.getFloat("LENGTH");
DIR = ee.getInt("Dir");
LINEA = ee.getString("LINEA");
MODO = ee.getString("MODO");
RUTA = ee.getString("RUTA");
FrecHPico = ee.getFloat("FrecHPico");
FrecHVal = ee.getFloat("FrecHVal");
TRec = ee.getFloat("TRec");
TIPO = ee.getString("TIPO");
TAprox = ee.getFloat("TAprox");
FROM_ID = ee.getInt("FROM_ID");
TO_ID = ee.getInt("TO_ID");
registros2arreglos(j);
j= j + 1;
}
stmt2.close();
}
catch ( Exception e){
System.out.println(e.getMessage());
}

//*****
try{ //Escritura en la tabla Estrella entrante
PreparedStatement entrante = con.prepareStatement("INSERT INTO entrante VALUES
(?,?,?,?,?,?,?,?,?,?,?,?,?)");
PreparedStatement rap = con.prepareStatement("INSERT INTO rapuntador VALUES (?,?)");
int k = 1;
int regAnt = -1; //Garantizo la NO existencia de un nodo negativo
for (k=1; k<j; k++) {
entrante.setInt(1, identificador[k]);
entrante.setFloat(2, longitud[k]);
entrante.setInt(3, direccion[k]);
entrante.setString(4, linea[k]);
entrante.setString(5, modo[k]);
entrante.setString(6, ruta[k]);
entrante.setFloat(7, frecHpico[k]);
entrante.setFloat(8, frecHvalle[k]);
entrante.setFloat(9, tiempor[k]);
entrante.setString(10, tipo[k]);
entrante.setFloat(11, tiempoa[k]);
entrante.setInt(12, fromID[k]);
entrante.setInt(13, toID[k]);
entrante.executeUpdate();

if (regAnt != toID[k]) { //Guarda los datos en la estructura R-apuntador
rap.setInt(1, k);
rap.setInt(2, toID[k]);
rap.executeUpdate();
}
regAnt = toID[k];
}
}

```

```

    }
    catch(Exception e) {
        System.out.println(e.getMessage());
    }

    try(Statement stmt3 = con.createStatement()){
        j=1;
        ResultSet es = stmt3.executeQuery("SELECT * FROM hrmm2 order by \"FROM_ID\";"); //Estrella
saliente
        while (es.next()){
            //Obtiene los valores de la BD y los asigna a las correspondientes variables
            ID = es.getInt("ID");
            LONGITUD = es.getFloat("LENGTH");
            DIR = es.getInt("Dir");
            LINEA = es.getString("LINEA");
            MODO = es.getString("MODO");
            RUTA = es.getString("RUTA");
            FrechHPico = es.getFloat("FrechHPico");
            FrechHVal = es.getFloat("FrechHVal");
            TRec = es.getFloat("TRec");
            TIPO = es.getString("TIPO");
            TAprox = es.getFloat("TAprox");
            FROM_ID = es.getInt("FROM_ID");
            TO_ID = es.getInt("TO_ID");
            registros2arreglos(j);
            j= j + 1;
        }
        stmt3.close();
    }
    catch ( Exception e ){
        System.out.println(e.getMessage());
    }

    //*****
    try{ //Escritura en la tabla Estrella entrante
        PreparedStatement saliente = con.prepareStatement("INSERT INTO saliente VALUES
(? ,? ,? ,? ,? ,? ,? ,? ,? ,? ,? ,? );");
        PreparedStatement ap = con.prepareStatement("INSERT INTO apuntador VALUES (?,?);");
        int k = 1;
        int regAnt = -1; //Garantizo la NO existencia de un nodo negativo
        for (k=1; k<j; k++) {
            saliente.setInt(1, identificador[k]);
            saliente.setFloat(2, longitud[k]);
            saliente.setInt(3, direccion[k]);
            saliente.setString(4, linea[k]);
            saliente.setString(5, modo[k]);
            saliente.setString(6, ruta[k]);
            saliente.setFloat(7, frechHpico[k]);
            saliente.setFloat(8, frechHvalle[k]);
            saliente.setFloat(9, tiempo[k]);
            saliente.setString(10, tipo[k]);
            saliente.setFloat(11, tiempoa[k]);
            saliente.setInt(12, fromID[k]);

```

```

    saliente.setInt(13, toID[k]);
    saliente.executeUpdate();

    if (regAnt != fromID[k]) { //Guarda los datos en la estructura apuntador
        ap.setInt(1, k);
        ap.setInt(2, fromID[k]);
        ap.executeUpdate();
    }
    regAnt = fromID[k];
}
}
catch(Exception e) {
    System.out.println(e.getMessage());
}

con.close();
} //Fin Main

public static void registros2arreglos(int k) {

    if (DIR == 1) {
        fromID[k] = FROM_ID;
        toID[k] = TO_ID;
        direccion[k] = 1;
        identificador[k] = ID;
        longitud[k] = LONGITUD;
        linea[k] = LINEA;
        modo[k] = MODO;
        ruta[k] = RUTA;
        frecHpico[k] = FrecHPico;
        frecHvalle[k] = FrecHVal;
        tiempor[k] = TRec;
        tipo[k] = TIPO;
        tiempoa[k] = TAprox;
    } else {
        if (DIR == -1) {
            fromID[k] = TO_ID;
            toID[k] = FROM_ID;
            direccion[k] = 1;
            identificador[k] = ID;
            longitud[k] = LONGITUD;
            linea[k] = LINEA;
            modo[k] = MODO;
            ruta[k] = RUTA;
            frecHpico[k] = FrecHPico;
            frecHvalle[k] = FrecHVal;
            tiempor[k] = TRec;
            tipo[k] = TIPO;
            tiempoa[k] = TAprox;
        }
        else {

```

```
    fromID[k] = FROM_ID;
    toID[k] = TO_ID;
    direccion[k] = 1;
    identificador[k] = ID;
    longitud[k] = LONGITUD;
    linea[k] = LINEA;
    modo[k] = MODO;
    ruta[k] = RUTA;
    frechPico[k] = FrechPico;
    frechValle[k] = FrechVale;
    tiempor[k] = TRec;
    tipo[k] = TIPO;
    tiempoa[k] = TAprox;

    fromID[k+1] = TO_ID;
    toID[k+1] = FROM_ID;
    direccion[k+1] = 1;
    identificador[k+1] = ID;
    longitud[k+1] = LONGITUD;
    linea[k+1] = LINEA;
    modo[k+1] = MODO;
    ruta[k+1] = RUTA;
    frechPico[k+1] = FrechPico;
    frechValle[k+1] = FrechVale;
    tiempor[k+1] = TRec;
    tipo[k+1] = TIPO;
    tiempoa[k+1] = TAprox;
}
}

} //Fin funcion
} //Fin Clase
```

C. Creación de tablas en SQL

```
CREATE TABLE hrmm2
(
  "ID" integer,
  "LENGTH" real,
  "Dir" integer,
  "LINEA" text,
  "MODO" text,
  "RUTA" text,
  "FrechPico" real,
  "FrechVal" real,
  "TRec" real,
  "TIPO" text,
  "TAprox" real,
  "FROM_ID" integer,
  "TO_ID" integer,
)
WITH (
  OIDS=FALSE
);
ALTER TABLE hrmm2 OWNER TO postgres;
```


D. Clase SVHP_Servlet

```

package servlets;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.SQLException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class SVHP_Servlet extends HttpServlet {
    private static final long serialVersionUID = 1L;
    // Datos introducidos por el usuario
    public static int ORIGEN;
    public static int DESTINO;
    public static int transferenciasMaximas;
    public static String S_origen;
    public static String S_destino;
    public static String S_transferencias;
    public static String S_ResultadoFinal;
    public static PrintWriter out;

    @Override
    protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws
    ServletException, IOException {
        S_origen = req.getParameter("origen");
        S_destino = req.getParameter("destino");
        S_transferencias = req.getParameter("transferencias");
        ORIGEN = Integer.parseInt(S_origen);
        DESTINO = Integer.parseInt(S_destino);
        transferenciasMaximas = Integer.parseInt(S_transferencias);

        try {
            SVHP ObjSVHP = new SVHP(ORIGEN, DESTINO, transferenciasMaximas);
            S_ResultadoFinal = ObjSVHP.impression();
            response(resp, S_ResultadoFinal);
        }
        catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
        catch (SQLException e) {
            e.printStackTrace();
        }
    }

    private static void response(HttpServletResponse resp, String msg) throws IOException {
        out = resp.getWriter();
        out.println("<html>");
        out.println("<head><title>Resultado SVHP</title>");
        out.println("<link rel='stylesheet' href='OpenLayers/examples/style.css' type='text/css'>
        </head>");
        out.println("<body>");
    }
}

```

```
out.println("<table border='0' width='1280' align='center'>");
out.println("    <tr>");
out.println("        <td colspan='2'><div align='center'><h1>Advanced Traveller
Information System (ATIS).</h1></div></td>");
out.println("    </tr>");
out.println("    <tr>");
out.println("        <td width='100%' valign='top'>");
out.println("            <t1>" + msg + "</t1> </td>");
out.println("    </tr></table>");
out.println("</body>");
out.println("</html>");
    }
}
```

E. Clase SVHP

```

package servlets;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Map;
import java.sql.*;

public class SVHP {
    public static ArrayList<Integer> qNow = new ArrayList<Integer>();
    public static ArrayList<Integer> qNext = new ArrayList<Integer>();
    public static ArrayList<Float> CostosqNext = new ArrayList<Float>();
    //Arreglo temporal para guardar los costos calculados en la cola de un arco TF, pero que no pueden ser
    //escritos, ya que no se puede calcular el valor del nodo siendo transferencia.
    final static float INFINITO = 7777777;
    public static int contadorTransferencias = 0;
    // Variables asociadas a las columnas de la tabla arcos
    public static int FROM_ID;
    public static int TO_ID = -1;
    public static String TIPO;
    public static String MODO;
    public static float T_D_REC;
    // Datos introducidos por el usuario
    public static int ORIGEN;
    public static int DESTINO;
    public static int transferenciasMaximas = 3; //Temporalmente en 3 transferencias maximas

    public static ArrayList<Integer> cabezas = new ArrayList<Integer>(); //Cabezas de un H-arco
    public static float CostoEnCola = 0; //Costo en la cola de un arco, usado en ConcatenaArco
    public static float tiempoEsperaCola = 0; //Usado en concatenaArco y concatenaHArco

    public static ArrayList<Nodo> NodosVisitados = new ArrayList<Nodo>(); //Arreglo donde se guardan todos
    los nodos visitados
    public ArrayList<Integer> NodosHRuta= new ArrayList<Integer>(); //Nodos que conforman c/u de las h-rutas
    public ArrayList<Integer> PendientesHRuta= new ArrayList<Integer>(); //Cuando se presentan h-arcos...
    public ArrayList<Integer> ArcosHRuta = new ArrayList<Integer>(); //Arcos que conforman c/u de las h-rutas
    public static int calcularHiperRuta = 0;
    public static Map<Integer,String> mapaPareto = new HashMap<Integer,String>();
    public static String S_pareto="";
    public static int numeroRutasParetoptimales = 0;
    public static float tiempoMinOrigen = INFINITO;
    public static int arcoActual=-1;
    public static Connection conexion;

    // Constructor de la clase SVHP (Shortest Viable Hyperpath Problem)
    //*****
    public SVHP(int o, int d, int t) throws ClassNotFoundException, SQLException {
        SVHP.ORIGEN = o;
        SVHP.DESTINO = d;
        SVHP.transferenciasMaximas = t;
        //Conexión a la Base de Datos
        String url = "jdbc:postgresql://127.0.0.1:5432/postgres";
        Class.forName("org.postgresql.Driver");
        conexion = DriverManager.getConnection(url,"postgres","admin");
    }

```

```

//C.I. Para Qnow y Vod
qNow.add(DESTINO); //Inserta en qnow el destino (nodo 22), seria (22,0) --> (nodo,estado)
//Vod = 0;
Nodo N = new Nodo();
N.Destino(DESTINO);
NodosVisitados.add(N);

while(contadorTransferencias <= SVHP.transferenciasMaximas && qNow.isEmpty() == false){
//repeat until h > k or Qnow = vacío
while (qNow.isEmpty() == false) {
    Statement stmt =
    conexion.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDA
    TABLE);
    ResultSet rs = stmt.executeQuery("SELECT * FROM arcos where \"TO_ID\" = \" +
    qNow.get(0) + ";"); //select [j,sx] from Qnow
    int PNV; //Posicion en la que se encontro al elemento "Nodo Visitado"
    PNV = PosicionNodoVisitado(qNow.get(0));

    if ((NodosVisitados.get(PNV).costo < NodosVisitados.get(PNV).lastlabel) &&
    (NodosVisitados.get(PNV).costo < tiempoMinOrigen)) {
        NodosVisitados.get(PNV).lastlabel = NodosVisitados.get(PNV).costo;
        //System.out.println("ID:" + NodosVisitados.get(PNV).id + " t:" +
        NodosVisitados.get(PNV).costo);

        if (NodosVisitados.get(PNV).id != SVHP.ORIGEN) {
            //No debe obtener la estrella entrante del origen
            while (rs.next()) { //for each e E B(j) do {
                FROM_ID = rs.getInt("FROM_ID");
                TO_ID = rs.getInt("TO_ID");
                TIPO = rs.getString("TIPO");
                MODO = rs.getString("MODO");
                T_D_REC = rs.getFloat("T_D_REC");

                if (TIPO.equalsIgnoreCase("H")) { //Si es hiper arco...
                    concatenaHiperArco(conexion);
                }
                else {
                    concatenaArco();
                }
            } // Fin While
        } // Fin If(NodosVisitados.get(PNV).id != SVHP.ORIGEN)
    }

    if (qNow.get(0) == ORIGEN) {
        PNV = PosicionNodoVisitado(ORIGEN);
        if (NodosVisitados.get(PNV).costo < tiempoMinOrigen) {
            tiempoMinOrigen = NodosVisitados.get(PNV).costo;
            System.out.println("Entró el origen con un costo de: " + tiempoMinOrigen
            + " Y CON " + contadorTransferencias + " TRANSFERENCIAS");
            calcularHiperRuta = 1;
        }
    }
}
qNow.remove(0); //Elimina de la lista al nodo del cual se extrajo su estrella entrante

```

```

    } //FIN while qnow diferente de vacio

    if (calcularHiperRuta == 1) {
        System.out.println("AGREGANDO LOS NODOS VISITADOS A LA HIPER RUTA...");
        NodosEnHiperRuta();
    }
    calcularHiperRuta = 0;
    //Qnow = Qnext ; Qnext =vacio; h = h + 1
    qNow.addAll(qNext);
    qNext.clear();
    contadorTransferencias = contadorTransferencias + 1;
    //Agrega los costos de los nodos cola de los arcos de transferencia
    //System.out.println("TRANSFIRIENDO TIEMPOS A LOS NODOS DE Qnext...");
    for (int cont=0;cont<CostosqNext.size();cont++) {
        NodosVisitados.get(PosicionNodoVisitado(qNow.get(cont))).costo = CostosqNext.get(cont);
    }
    CostosqNext.clear();

    System.out.println("Qnow: " + qNow + " Qnext: " + qNext + " Transferencias: " +
        contadorTransferencias);

    } //Fin while principal

} //FIN SVHP

//PROCEDIMIENTO CONCATENA ARCO
//*****
public static void concatenaArco() {
    int wt = 0;
    int con = 0;
    int s = 0;
    int PNV_Cabeza = -1; //Posición fuera del arreglo
    int PNV_Cola = -1; //Posición fuera del arreglo

    if (MOD0.equalsIgnoreCase("TF")) { //Si es tranferencia modal --> wt = 1.
        wt=1;
    }
    else {
        wt=0;
    }
    if (PosicionNodoVisitado(FROM_ID) == -1) { //No se ha agregado el nodo a NodosVisitados y el arco NO es TF
        Nodo N_A = new Nodo(FROM_ID);
        NodosVisitados.add(N_A); //Se puede duplicar el nodo (N_A) en el arreglo NodosVisitados, se debe
        verificar si ya existe
    }
    //Cálculo del costo para los nodos, cuando se conectan a través de ARCOS
    PNV_Cabeza = PosicionNodoVisitado(TO_ID); //Posicion del nodo CABEZA en NodosVisitados
    PNV_Cola = PosicionNodoVisitado(FROM_ID); //Posicion del nodo COLA en NodosVisitados
    tiempoEsperaCola = NodosVisitados.get(PNV_Cola).costo;

    while (con == 0) {
        CostoEnCola = T_D_REC + NodosVisitados.get(PNV_Cabeza).costo; //Tiempo del
        arco + tiempo acumulado hasta el nodo cabeza
    }
}

```

```

//System.out.println("Nuevo Costo: " + CostoEnCola + " Costo guardado: " +
tiempoEsperaCola);
    if ((CostoEnCola < tiempoEsperaCola) && (CostoEnCola < tiempoMinOrigen)) { //Si
el costo calculado es menor al almacenado... graba el dato en el vector de Objs.

        //DIAGRAMA E ESTADOS
        //MODO del arco: TRANSPORTE PUBLICO, TLB, TL, RTP, ME, MB
        if (MODO.equalsIgnoreCase("TLB ") || MODO.equalsIgnoreCase("TL
") || MODO.equalsIgnoreCase("RTP ") || MODO.equalsIgnoreCase("ME ")
            || MODO.equalsIgnoreCase("MB ")) {

                if(NodosVisitados.get(PNV_Cabeza).estado==0) {
                    s = 1;
                }
                if(NodosVisitados.get(PNV_Cabeza).estado==1 ||
NodosVisitados.get(PNV_Cabeza).estado==2) {
                    s = 1;
                }
                if(NodosVisitados.get(PNV_Cabeza).estado==4) {
                    s = 4;
                }

                if(NodosVisitados.get(PNV_Cabeza).estado==3 || NodosVisitados.get(PNV_Cabeza).estado==5) {
                    s = -1;
                    System.out.println("NO VIABLE PARA CONCATENAR
ARCO TP: " + NodosVisitados.get(PNV_Cabeza).id);
                }
            }
        //MODO del arco: PEATONAL, PE y PU
        if (MODO.equalsIgnoreCase("PE ") || MODO.equalsIgnoreCase("PU ")) {

                if(NodosVisitados.get(PNV_Cabeza).estado==0) {
                    s = 2;
                }
                if(NodosVisitados.get(PNV_Cabeza).estado==1) {
                    s = 1;
                }
                if(NodosVisitados.get(PNV_Cabeza).estado==2) {
                    s = 2;
                }
                if(NodosVisitados.get(PNV_Cabeza).estado==4) {
                    s = 4;
                }

                if(NodosVisitados.get(PNV_Cabeza).estado==3 || NodosVisitados.get(PNV_Cabeza).estado==5) {
                    s = -1;
                    System.out.println("NO VIABLE PARA CONCATENAR
ARCO P");
                }
            }
        //MODO del arco: AUTOMOVIL (MODO RESTRINGIDO)

```

```

        if (MOD0.equalsIgnoreCase("AU ")) {

            if(NodosVisitados.get(PNV_Cabeza).estado==0) {
                s = 3;
            }
            if(NodosVisitados.get(PNV_Cabeza).estado==3) {
                s = 3;
            }
            if(NodosVisitados.get(PNV_Cabeza).estado==1) {
                s = 5;
            }
            if(NodosVisitados.get(PNV_Cabeza).estado==5) {
                s = 5;
            }

            if(NodosVisitados.get(PNV_Cabeza).estado==4 || NodosVisitados.get(PNV_Cabeza).estado==2) {
                s = -1;
                System.out.println("NO VIABLE PARA CONCATENAR
ARCO AU");
            }
        }
        //MOD0 del arco: TRANSFERENCIA
        if (MOD0.equalsIgnoreCase("TF ")) {

            if(NodosVisitados.get(PNV_Cabeza).estado==5) {
                s = -1;
                //System.out.println("NO VIABLE PARA CONCATENAR
ARCO TF: " + NodosVisitados.get(PNV_Cabeza).id + " Edo: " + NodosVisitados.get(PNV_Cabeza).estado);
            }

            if(NodosVisitados.get(PNV_Cabeza).estado==0 || NodosVisitados.get(PNV_Cabeza).estado==1) {
                s = 1;
            }
            if(NodosVisitados.get(PNV_Cabeza).estado==2) {
                s = 1;
            }
            if(NodosVisitados.get(PNV_Cabeza).estado==3) {
                s = 4;
            }
            if(NodosVisitados.get(PNV_Cabeza).estado==4) {
                s = 4;
            }
        }

        System.out.println("CABEZA: " + NodosVisitados.get(PNV_Cabeza).id + "
COLA: " + NodosVisitados.get(PNV_Cola).id + " MOD0: " + MOD0 + " ESTADO: " + s);

        if (wt==0 && s!=-1) { //Si el arco NO es una transferencia modal

            if (CostoEnCola < tiempoMinOrigen) { //Si el costo calculado es
menor que el costo en el origen se agrega el nodo a Qnow

```

```

agregado a Qnow
CostoEnCola;
NodosVisitados.get(PNV_Cola).costo =
NodosVisitados.get(PNV_Cola).transferencias =
NodosVisitados.get(PosicionNodoVisitado(TO_ID)).transferencias + wt; // ws(i) = ws(j) + wt
NodosVisitados.get(PNV_Cola).estado = s;
NodosVisitados.get(PNV_Cola).sucesor.clear();
//Borra el contenido del sucesor para garantizar que cada arco solo tenga 1.

NodosVisitados.get(PNV_Cola).sucesor.add(TO_ID); //SAs(i) = e;
qNow.add(FROM_ID); //Agregalo a QNow.
//System.out.println(FROM_ID + ": " +
CostoEnCola + " mejora el tiempo " + tiempoEsperaCola + " se agrega a Qnow");
    }
}

if (wt==1 && s!=-1) { //Transferencia modal

    if (qNext.indexOf(FROM_ID) == -1) { //Si no ha sido agregado el
nodo "FromId" a Qnext
        if (CostoEnCola < tiempoMinOrigen) {
            NodosVisitados.get(PNV_Cola).costo =
CostoEnCola;
            NodosVisitados.get(PNV_Cola).transferencias =
NodosVisitados.get(PosicionNodoVisitado(TO_ID)).transferencias + wt; // ws(i) = ws(j) + wt
            NodosVisitados.get(PNV_Cola).estado = s;
            NodosVisitados.get(PNV_Cola).sucesor.clear();
//Borra el contenido del sucesor para garantizar que cada arco solo tenga 1.

            NodosVisitados.get(PNV_Cola).sucesor.add(TO_ID); //SAs(i) = e;
            qNext.add(FROM_ID); //Agregalo a QNext.
            CostosqNext.add(CostoEnCola); //Agrega su
costo calculado al arreglo temporal. DEBO CAMBIARLO CON EL NUEVO VALOR
        }
    }
    con=1; //Se hizo la concatenacion
}
else {
    con=1; //Si el costo calculado es mayor que el almacenado, no hagas
nada, solo salte de ahi!!!
}
}

}
//*****

//PROCEDIMIENTO CONCATENA HIPER ARCO

```



```

//*****
*****
    public static void concatenaHiperArco(Connection conexion) throws SQLException {
        int entrarQnow = 1;           //Bandera para saber si ya se calcularon todos los
costos en las cabezas de los hiper arcos (entra=1, no entra=0)
        int PNV_CabezaHArco = -1; //PosicionNodoVisitado para las cabezas de los hiperArcos, -1=
fuera del arreglo
        ArrayList<Float> VijnodoCabezaHArco = new ArrayList<Float>();    //TIEMPO asignado a
cada cabeza del h-arco
        ArrayList<Float> T_esperaHArco = new ArrayList<Float>();    //TIEMPO de espera de cada
hiper arco
        ArrayList<Integer> AuxiliarCabezas = new ArrayList<Integer>();
        ArrayList<Float> AuxiliarTiempoEspera = new ArrayList<Float>();
        ArrayList<Integer> ArregloPosicion = new ArrayList<Integer>();
        //Calculo del costo y cjto atractivo para el h-arco
        int posicionFinal;
        int k=1; //Algoritmo k=2, pero como el arreglo comenza en el elemento 0, k=1
        float fivk = 0;
        float val = 0;
        ArrayList<Integer> ConjuntoAtractivo = new ArrayList<Integer>();
        float frecCombinada = 0;
        int TF_Max = 0;
        String TIPO_HA;

        String query3 = "SELECT * FROM arcos WHERE \"FROM_ID\" = " + FROM_ID + " ORDER BY
\"TO_ID\";"; //Selecciona las cabezas del h-arco
        Statement stmt3 =
conexion.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
        ResultSet rs3 = stmt3.executeQuery(query3);
        while (rs3.next()) {
            TIPO_HA = rs3.getString("TIPO");
            if (TIPO_HA.equalsIgnoreCase("H")) {
                cabezas.add(rs3.getInt("TO_ID")); //Agrega las cabezas encontradas a un
arreglo llamado "cabezas"
            }
        }

        stmt3.close();

        //Verificar si el H-Arco puede entrar a Qnow
        for (int i=0; i<cabezas.size(); i++) {
            String query4 = "SELECT \"FREC_H_PIC\" FROM arcos WHERE \"FROM_ID\" = " +
FROM_ID + " AND \"TO_ID\" = " + cabezas.get(i) + ";";
            Statement stmt4 =
conexion.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
            ResultSet rs4 = stmt4.executeQuery(query4);
            while (rs4.next()) {
                T_esperaHArco.add(rs4.getFloat("FREC_H_PIC"));
            }

            stmt4.close();
            PNV_CabezaHArco = PosicionNodoVisitado(cabezas.get(i)); //Posicion del nodo
CABEZA de H-Arco en NodosVisitados

```

```

        if (PNV_CabezaHArco != -1) { //Si ya esta almacenado el nodo cabeza en
NodosVisitados...
            VijnodoCabezaHArco.add(NodosVisitados.get(PNV_CabezaHArco).costo);
//Obtiene costo de cada cabeza y lo agrega a VijnodoCabezaHArco
        }
        else {
            VijnodoCabezaHArco.add(INFINITO);
        }
        //System.out.println("NumCabeza: " + cabezas.get(i) + " Costo: " +
VijnodoCabezaHArco + " Tiempo: " + T_esperaHArco);

        if (VijnodoCabezaHArco.get(i) == INFINITO) { //|| VijnodoCabezaHArco.get(i) == 0
Si aun no se ha calculado el costo de algún nodo cabeza... (Vij = INFINITO)

            entrarQnow = entrarQnow * 0; //No debe entrar a Qnow
        }
        else
        {
            entrarQnow = entrarQnow * 1; //Entra a Qnow
        }
    }

    if (entrarQnow == 1) { //Puede entrar a Qnow y se calcula el costo
        if (PosicionNodoVisitado(FROM_ID) == -1) { //Si aun no se ha agregado el nodo a
NodosVisitados
            Nodo N_HA = new Nodo(FROM_ID);

            NodosVisitados.add(N_HA);
        }
        else {
            //System.out.println(FROM_ID + " intenta entrar nuevamente!!!! ");
        }
        tiempoEsperaCola = NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).costo;

        if(cabezas.size()>1) { //No tiene caso ordenar un solo elemento, se ordenan 2 o
mas en un arreglo.

            // ORDENA LOS ARREGLOS DE MANERA CRECIENTE
            ArregloPosicion = ordenaArreglos(VijnodoCabezaHArco.size(),
VijnodoCabezaHArco);

            for (int cont=0;cont<cabezas.size();cont++) {
                posicionFinal = ArregloPosicion.get(cont); // Posiciones
                AuxiliarCabezas.add(cabezas.get(posicionFinal));

                AuxiliarTiempoEspera.add(T_esperaHArco.get(posicionFinal));
                //Calculo de las transferencias MAXIMAS para la cola del hiper
arco

                if
(NodosVisitados.get(PosicionNodoVisitado(cabezas.get(posicionFinal))).transferencias > TF_Max) {
                    TF_Max =
NodosVisitados.get(PosicionNodoVisitado(cabezas.get(posicionFinal))).transferencias;
                }
            }
        }
    }

```

```

        NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).transferencias =
TF_Max; //Actualiza las transferencias de la cola del h-arco
        //System.out.println("Cabezas ordenado: " + AuxiliarCabezas);
        // PROCEDIMIENTO CONJUNTO ATRACTIVO EN UN H-ARCO
        ConjuntoAtractivo.add(AuxiliarCabezas.get(0));
        frecCombinada = 60/AuxiliarTiempoEspera.get(0); //Frecuencia del
primer elemento del h-arco

        val = AuxiliarTiempoEspera.get(0) + VijNodoCabezaHArco.get(0);

        while(k<cabezas.size() && VijNodoCabezaHArco.get(k) < val)
        {
            fivk = 60/AuxiliarTiempoEspera.get(k);

            frecCombinada = frecCombinada + fivk;

            val = val-((val-VijNodoCabezaHArco.get(k))*fivk)/frecCombinada;
            ConjuntoAtractivo.add(AuxiliarCabezas.get(k));
            k++;

        }

        // PREGUNTAR SI ES MENOR AL TIEMPO ALMACENADO EN EL
ARREGLO!!!!!!!!!!!!
        if ((val < tiempoEsperaCola)&&( val < tiempoMinOrigen)) { //Si el costo
calculado es menor al almacenado... graba el dato en el obj correspondiente del ArrayList NodosVisitados
        //System.out.println("ConjuntoAtractivo: " + ConjuntoAtractivo +
" Costo final: " + val);

        NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).sucesor =
ConjuntoAtractivo; //SAs(i) = e;

        NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).costo =
val;

        NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).estado = 1;
        qNow.add(FROM_ID); //Se añade a Qnow
        }
        else {
            //NodosVisitados.remove(PosicionNodoVisitado(FROM_ID));
        }
    }
    else { //Híper arco con una sola cabeza
        AuxiliarCabezas.add(cabezas.get(0));
        ConjuntoAtractivo.add(AuxiliarCabezas.get(0));
        val = VijNodoCabezaHArco.get(0) + T_esperaHArco.get(0);

        // PREGUNTAR SI ES MENOR AL TIEMPO ALMACENADO EN LA
TABLA!!!!!!!!!!!!
        if (val < tiempoEsperaCola)&&( val < tiempoMinOrigen)) { //Si el costo
calculado es menor al almacenado... graba el dato en obj correspondiente en NodosVisitados
        //System.out.println("ConjuntoAtractivo: " + ConjuntoAtractivo +
" Costo final: " + val);

        NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).sucesor =
ConjuntoAtractivo; //SAs(i) = e;

        NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).costo =
val;

```

```

        NodosVisitados.get(PosicionNodoVisitado(FROM_ID)).estado = 1;
        qNow.add(FROM_ID); //Se añade a Qnow
        //System.out.println(FROM_ID + ": " + val + " Mejora el tiempo "
+ tiempoEsperaCola + ", entra a QNow");
    }
    else {
        //System.out.println(FROM_ID + ": " + " NO mejora el tiempo " +
tiempoEsperaCola + ", por lo tanto no entra a QNow");
    }
}

}
else {
    //System.out.println("NO Puede entrar a QNow");
}
cabezas.clear(); //Borra el arreglo temporal que contiene las cabezas del h-arco

}
//*****
*****

// ORDENA LAS CABEZAS DE LOS H-ARCOS PARA LA OBTENCION DEL CJTO ATRACTIVO
//
*****
*****

public static ArrayList<Integer> ordenaArreglos(int dimension, ArrayList<Float> arreglo) {
    ArrayList<Integer> posicion = new ArrayList<Integer>();
    for(int n=0;n<dimension;n++) {
        posicion.add(n);
    }
    for (int i = 0; i < dimension; i++) {
        for (int j = 0; j < dimension - 1; j++) {
            if (arreglo.get(j) > arreglo.get(j+1)) {
                intercambiar(arreglo, j, j+1);
                intercambiar2(posicion, j, j+1);
            }
        }
    }
    return posicion;
}

private static void intercambiar2(ArrayList<Integer> posicion, int a, int b) {
    int tmp = posicion.get(a);
    posicion.set(a,posicion.get(b));
    posicion.set(b,tmp);
}

private static void intercambiar(ArrayList<Float> arreglo, int a, int b) {
    float tmp = arreglo.get(a);
    arreglo.set(a,arreglo.get(b));
    arreglo.set(b,tmp);
}

```

```

// ENCUENTRA LA POSICION DEL NODO QUE SE REQUIERE DENTRO DEL ARREGLO NODOS VISITADOS
//
*****
*****

    public static int PosicionNodoVisitado(int a) {
        int contNV=0; //Puntero para el arreglo NodosVisitados
        int encontradoNV=0; //Bandera para saber si se ha encontrado en NodosVisitados el
elemento referenciado por qNow.get(0).
        int posNV=0; //Posicion en la que se encontro al elemento
        while (contNV<NodosVisitados.size() && encontradoNV==0) {
            if (a == NodosVisitados.get(contNV).id) {
                posNV = contNV;
                encontradoNV = 1;
            }
            contNV++;
        }
        if (encontradoNV == 0) {
            posNV = -1;
        }
        return posNV;
    }

// Escribe en un arraylist los nodos de la hiper-ruta para k transferencias modales
//
*****
*****

    public void NodosEnHiperRuta() throws SQLException {
        int nodoActual = ORIGEN;
        ArrayList<Integer> nodosSucesores; //Sucesores para cada nodo analizado

        NodosHRuta.add(ORIGEN); //NodosHRuta- Array q guarda los nodos de la hruta para k
transferencias

        while (nodoActual != DESTINO || (PendientesHRuta.isEmpty() == false)) {
            if (nodoActual == DESTINO) {
                NodosHRuta.add(PendientesHRuta.get(0)); //Se agrega la cola
                NodosHRuta.add(PendientesHRuta.get(1)); //Se agrega la cabeza
                nodoActual = PendientesHRuta.get(1); //Se reinicia el recorrido a partir
de la cabeza

                PendientesHRuta.remove(0); // Borra el primer elemento
                PendientesHRuta.remove(0); // Borra el segundo elemento (se convierte
en primero despues que se elimina el anterior)
            }
            else {
                nodosSucesores =
NodosVisitados.get(PosicionNodoVisitado(nodoActual)).sucesor; //Sucesores del nodo analizado

                if (nodosSucesores.size() > 1) { //Si tiene mas de un sucesor...
                    NodosHRuta.add(nodosSucesores.get(0)); //Solo el primer
elemento entra en la lista NodosHRuta

                    for (int i=1; i<nodosSucesores.size(); i++) { //El resto entra en el
array PendientesHRuta

```

```

        PendientesHRuta.add(nodoActual);
        PendientesHRuta.add(nodosSucesores.get(i));
    }
    nodoActual = nodosSucesores.get(0); //El elemento agregado
será ahora el nodoActual
    }
    else {
        NodosHRuta.add(nodosSucesores.get(0)); //Agrega el nodo al
array NodosHRuta
        nodoActual = nodosSucesores.get(0); //El elemento agregado
será ahora el nodoActual
    }
}
}
System.out.println("Nodos en h-ruta: " + NodosHRuta);
ArcosEnHiperRuta(conexion);
}

```

```

// ESCRIBE LAS HIPER RUTAS EN TEXTO, ORIGEN, DESTINO Y TRANSFERENCIAS MODALES
*****

```

```

    public void ArcosEnHiperRuta(Connection conexion) throws SQLException {
        int apuntador = 0;
        int colaArcoHR;
        int cabezaArcoHR;
        int incidencias = 0;
        int numArcos=0;
        int PNV;
        escribeltinerario escribeRuta = new escribeltinerario();

        numeroRutasParetoptimales++;
        Statement stmtArcos =
conexion.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_UPDATABLE);
        for (int i=0;i<NodosHRuta.size();i++) {
            if (NodosHRuta.get(i) == DESTINO) {
                incidencias++;
            }
        }
        numArcos = NodosHRuta.size() - incidencias;

        for (int i = 1; i <= numArcos; i++) {
            colaArcoHR = NodosHRuta.get(apuntador);
            if (colaArcoHR != DESTINO) {
                cabezaArcoHR = NodosHRuta.get(apuntador+1);
            }
            else {
                apuntador++;
                colaArcoHR = NodosHRuta.get(apuntador);
                cabezaArcoHR = NodosHRuta.get(apuntador+1);
            }
        }

        ResultSet rsArcos = stmtArcos.executeQuery("SELECT * FROM arcos WHERE
\FROM_ID\" = " + colaArcoHR + " AND \TO_ID\" = " + cabezaArcoHR + ";");
    }
}

```

```

        while (rsArcos.next()) {
            ArcosHRuta.add(rsArcos.getInt("ID"));
        }

        rsArcos.close();
        apuntador++;
    }

    mapaPareto.put(contadorTransferencias, ArcosHRuta.toString());
    System.out.println("ParetoMap: " + mapaPareto);

    PNV = PosicionNodoVisitado(SVHP.ORIGEN);
    System.out.println("PNV ORIGEN: " + PNV);
    System.out.println("ORIGEN EN NODOS VISITADOS: " + NodosVisitados.get(PNV).id + "
COSTO: " + NodosVisitados.get(PNV).costo);
    S_pareto += "<b>Con " + (contadorTransferencias) + " transferencia (cambio de modo).
</b><br>";
    S_pareto += "Tiempo estimado: <b>" + Float.toString(((int)NodosVisitados.get(PNV).costo)
+ " minutos. <br>Itinerario: </b><br>";

    for (int i=0; i<ArcosHRuta.size(); i++) {
        S_pareto += escribeRuta.impresion(ArcosHRuta.get(i));
    }
    S_pareto += "<hr>";
    System.out.println("S_Pareto: "+ S_pareto);
    NodosHRuta.clear();
    ArcosHRuta.clear();
    //System.out.println("S_Pareto despues de borrar arcos: "+ S_pareto);
}

public String impresion() {
    String S_resultado;
    escribeNodos escribeNodo = new escribeNodos();
    S_resultado = "Origen: <b>" + escribeNodo.impresion(ORIGEN) + "</b><br>Destino: <b>" +
    escribeNodo.impresion(DESTINO) + "</b><br>Transferencias máximas: <b>" + transferenciasMaximas +
    "</b><br><hr>";
    S_resultado += S_pareto;
    return S_resultado;
}
}
}

```

F. Clase Nodo

```
package servlets;
import java.util.ArrayList;

public class Nodo {
    int id;
    String nombre;
    float costo;
    int transferencias;
    ArrayList<Integer> sucesor;
    int estado;
    float lastlabel;
    ArrayList<Integer> antecesor;

    public Nodo() {
        this.transferencias = 0;
        this.estado = 0;
        this.lastlabel = 7777777;
        this.sucesor = new ArrayList<Integer>();
        this.antecesor = new ArrayList<Integer>();
    }

    public Nodo(int ID) {
        this.id = ID;
        this.costo = 7777777;
        this.transferencias = 0;
        this.estado = 0;
        this.lastlabel = 7777777;
        this.sucesor = new ArrayList<Integer>();
        this.antecesor = new ArrayList<Integer>();
    }

    public void Destino(int ID) { //Solo para el destino
        this.id = ID;
        this.costo = 0;
    }
}
```


G. Clase escribeNodos

Debido a que el número de líneas codificadas para esta clase es muy grande, se escribe sólo parte de la misma.

```
package servlets;
```

```
public class escribeNodos {
    int numNodo;
    String S_Catalogo;
    String imprime;
```

```
public escribeNodos() {
    this.numNodo=-1;
    this.S_Catalogo="";
}
```

```
public String impresion(int numNodo) { // METRO
    if (numNodo==676) { S_Catalogo = "Tasqueña";}
    if (numNodo==677) { S_Catalogo = "General Anaya";}
    if (numNodo==678) { S_Catalogo = "Ermita";}
    if (numNodo==681) { S_Catalogo = "Portales";}
    if (numNodo==682) { S_Catalogo = "Nativitas";}
    if (numNodo==683) { S_Catalogo = "Villa de Cortés";}
    if (numNodo==1069){ S_Catalogo = "Xola";}
    if (numNodo==712) { S_Catalogo = "Viaducto";}
    if (numNodo==713 || numNodo==714 || numNodo==715) { S_Catalogo = "Chabacano";}
    if (numNodo==648) { S_Catalogo = "Universidad";}
    if (numNodo==657) { S_Catalogo = "Copilco";}
    if (numNodo==656) { S_Catalogo = "Miguel Ángel de Quevedo";}
    if (numNodo==658) { S_Catalogo = "Viveros";}
    if (numNodo==659) { S_Catalogo = "Coyoacán";}
    if (numNodo==666) { S_Catalogo = "Zapata";}
    if (numNodo==667) { S_Catalogo = "División del Norte";}
    if (numNodo==668) { S_Catalogo = "Eugenia";}
    if (numNodo==669) { S_Catalogo = "Etiopía";}
    if (numNodo==698 || numNodo==697) { S_Catalogo = "Centro Médico";}
    if (numNodo==721 || numNodo==722) { S_Catalogo = "Santa Anita";}
    if (numNodo==723 || numNodo==724) { S_Catalogo = "Jamaica";}
    if (numNodo==653) { S_Catalogo = "Barranca del Muerto";}
    if (numNodo==663) { S_Catalogo = "Mixcoac";}
    if (numNodo==664) { S_Catalogo = "San Antonio";}
    if (numNodo==665) { S_Catalogo = "San Pedro de los Pinos";}
    if (numNodo==702 || numNodo==700) { S_Catalogo = "Tacubaya";}
    if (numNodo==732) { S_Catalogo = "Constitución de 1917";}
    if (numNodo==731) { S_Catalogo = "UAM-I";}
    if (numNodo==730) { S_Catalogo = "Cerro de la Estrella";}
    if (numNodo==688) { S_Catalogo = "Iztapalapa";}
    if (numNodo==687) { S_Catalogo = "Atlalilco";}
    if (numNodo==686) { S_Catalogo = "Escuadrón 201";}
    if (numNodo==691) { S_Catalogo = "Aculco";}
    if (numNodo==692) { S_Catalogo = "Apatlaco";}
    if (numNodo==693) { S_Catalogo = "Iztacalco";}
    if (numNodo==694) { S_Catalogo = "Coyuya";}
```

```

if (numNodo==716) { S_Catalogo = "La viga";}
if (numNodo==703) { S_Catalogo = "Patriotismo";}
if (numNodo==704) { S_Catalogo = "Chilpancingo";}
if (numNodo==709) { S_Catalogo = "Lázaro Cárdenas";}
if (numNodo==725) { S_Catalogo = "Mixiuhca";}
if (numNodo==1066) { S_Catalogo = "Velódromo";}
if (numNodo==727) { S_Catalogo = "Ciudad deportiva";}
if (numNodo==734) { S_Catalogo = "Puebla";}
if (numNodo==736) { S_Catalogo = "Agricola oriental";}
if (numNodo==737) { S_Catalogo = "Canal de San Juan";}
if (numNodo==733) { S_Catalogo = "Tepalcates";}
//TREN LIGERO
if (numNodo==214) { S_Catalogo = "Tasqueña";}
if (numNodo==213) { S_Catalogo = "Las Torres";}
if (numNodo==212) { S_Catalogo = "Ciudad Jardín";}
if (numNodo==211) { S_Catalogo = "La Virgen";}
if (numNodo==210) { S_Catalogo = "Xotepingo";}
if (numNodo==209) { S_Catalogo = "Nezahualpilli";}
if (numNodo==208) { S_Catalogo = "Registro Federal";}
if (numNodo==207) { S_Catalogo = "Textitlán";}
if (numNodo==206) { S_Catalogo = "El vergel";}
if (numNodo==205) { S_Catalogo = "Estadio Azteca";}
if (numNodo==204) { S_Catalogo = "Huilpulco";}
if (numNodo==203) { S_Catalogo = "Xomali";}
if (numNodo==202) { S_Catalogo = "Periférico";}
// METROBUS Línea 1
if (numNodo==1007) { S_Catalogo = "Perisur";}
if (numNodo==1072) { S_Catalogo = "Centro Cultural Universitario";}
if (numNodo==1008) { S_Catalogo = "Ciudad Universitaria";}
if (numNodo==1009) { S_Catalogo = "Dr. Gálvez";}
if (numNodo==1010) { S_Catalogo = "La bombilla";}
if (numNodo==1011) { S_Catalogo = "Altavista";}
if (numNodo==1012) { S_Catalogo = "Olivo";}
if (numNodo==1013) { S_Catalogo = "Francia";}
if (numNodo==1014) { S_Catalogo = "José Ma. Velasco";}
if (numNodo==1015) { S_Catalogo = "Teatro Insurgentes";}
if (numNodo==1016) { S_Catalogo = "Río Churubusco";}
if (numNodo==1017) { S_Catalogo = "Felix Cuevas";}
if (numNodo==1018) { S_Catalogo = "Parque hundido";}
if (numNodo==1019) { S_Catalogo = "Ciudad de los deportes";}
if (numNodo==1020) { S_Catalogo = "Colonia del Valle";}
... ..
... ..
... ..
//Automóvil
if (numNodo==3001) { S_Catalogo = "Periférico Sur N/S";}
imprime = S_Catalogo;
S_Catalogo = "";
return imprime;
}
}

```

H. Clase escribeltinerario

Debido a que el número de líneas codificadas para esta clase es muy grande (4861 líneas), se escribe sólo parte de la misma.

```

package servlets;
public class escribeltinerario {
int numArco;
int arcoActual;
String S_CatalogoArcos;
String imprime;

public escribeltinerario() {
    this.numArco=-1;
    this.arcoActual=-1;
    this.S_CatalogoArcos="";
}

public String impresion(int numArco) {
//***** METRO *****
//***** LINEA 2 *****
//***** Chabacano *****
    if(numArco==947) { //Arco de abordaje
        S_CatalogoArcos = "<font color='#cb8600'><b>En Chabacano, tome la línea 2 </b></font>";
        arcoActual=947;
    }
    if(numArco==1792) { //Arco de descenso
        if(arcoActual==906 || arcoActual==908 || arcoActual==909 || arcoActual==912 ||
        arcoActual==913 || arcoActual==915 || arcoActual==916 || arcoActual==946) {
            S_CatalogoArcos = "<font color='#cb8600'><b>en dirección a Cuatro caminos.
            <br>Descender en metro Chabacano. <br></b></font>";
        }
        arcoActual=1792;
    }
//***** Viaducto *****
    if(numArco==946) { //Arco de abordaje
        S_CatalogoArcos = "<font color='#cb8600'><b>En Viaducto, tome la línea 2 </b></font>";
        arcoActual=946;
    }
    if(numArco==1911) { //Arco de descenso
        if(arcoActual==906 || arcoActual==908 || arcoActual==909 || arcoActual==912 ||
        arcoActual==913 || arcoActual==915 || arcoActual==916) {
            S_CatalogoArcos = "<font color='#cb8600'><b>en dirección a Cuatro caminos.
            <br>Descender en metro Viaducto. <br></b></font>";
        }
    }
    if(arcoActual==947) {
        S_CatalogoArcos = "<font color='#cb8600'><b>en dirección a Tasqueña. <br>Descender en
        metro Viaducto. <br></b></font>";
    }
    arcoActual=1911;
}
//***** Xola *****

```

```
if(numArco==916) { //Arco de abordaje
    S_CatalogoArcos = "<font color='#cb8600'><b>En Xola, tome la línea 2 </b></font>";
    arcoActual=916;
}
if(numArco==1912) { //Arco de descenso
    if(arcoActual==906 || arcoActual==908 || arcoActual==909 || arcoActual==912 ||
    arcoActual==913 || arcoActual==915) {
        S_CatalogoArcos = "<font color='#cb8600'><b>en dirección a Cuatro caminos.
        <br>Descender en metro Xola. <br></b></font>";
    }
    if(arcoActual==947 || arcoActual==946) {
        S_CatalogoArcos = "<font color='#cb8600'><b>en dirección a Tasqueña.
        <br>Descender en metro Xola. <br></b></font>";
    }
    arcoActual=1912;
}

... ..
... ..
... ..

imprime = S_CatalogoArcos;
S_CatalogoArcos = "";

return imprime;
}
}
```

I. Clase escribeLonLat

Debido a que el número de líneas codificadas para esta clase es muy grande (4861 líneas), se escribe sólo parte de la misma.

```
package servlets;
```

```
public class escribeLonLat {
    int numNodo;
    String S_LonLat;
    String imprime;

    public escribeLonLat() {
        this.numNodo=-1;
        this.S_LonLat="";
    }

    public String impresion(int numNodo) {
        // METRO
        if (numNodo==676) { S_LonLat = "-99.139737,19.343740";}
        if (numNodo==677) { S_LonLat = "-99.145002,19.3532";}
        if (numNodo==678) { S_LonLat = "-99.142926,19.361844";}
        if (numNodo==681) { S_LonLat = "-99.141558,19.369871";}
        if (numNodo==682) { S_LonLat = "-99.140202,19.379422";}
        if (numNodo==683) { S_LonLat = "-99.138948,19.387467";}
        if (numNodo==1069){ S_LonLat = "-99.137810,19.395191";}
        if (numNodo==712) { S_LonLat = "-99.136893,19.400869";}
        if (numNodo==713 || numNodo==714 || numNodo==715) { S_LonLat = "-
99.135672,19.408980";}
        if (numNodo==648) { S_LonLat = "-99.173927,19.324340";}
        if (numNodo==657) { S_LonLat = "-99.176480,19.336115";}
        if (numNodo==656) { S_LonLat = "-99.181026,19.346824";}
        if (numNodo==658) { S_LonLat = "-99.176035,19.353708";}
        if (numNodo==659) { S_LonLat = "-99.170678,19.361048";}
        if (numNodo==666) { S_LonLat = "-99.164871,19.370608";}
        if (numNodo==667) { S_LonLat = "-99.159095,19.379710";}
        if (numNodo==668) { S_LonLat = "-99.157400,19.385397";}
        if (numNodo==669) { S_LonLat = "-99.156082,19.395842";}
        if (numNodo==698 || numNodo==697) { S_LonLat = "-99.155166,19.406451";}
        if (numNodo==721 || numNodo==722) { S_LonLat = "-99.121707,19.402641";}
        if (numNodo==723 || numNodo==724) { S_LonLat = "-99.121740,19.411024";}
        if (numNodo==653) { S_LonLat = "-99.189407,19.360163";}
        if (numNodo==663) { S_LonLat = "-99.187656,19.375776";}
        if (numNodo==664) { S_LonLat = "-99.186607,19.384523";}
        if (numNodo==665) { S_LonLat = "-99.185899,19.391190";}
        if (numNodo==702 || numNodo==700) { S_LonLat = "-99.187266,19.402964";}
        if (numNodo==732) { S_LonLat = "-99.063830,19.345915";}
        if (numNodo==731) { S_LonLat = "-99.074582,19.350723";}
        if (numNodo==730) { S_LonLat = "-99.085282,19.355554";}
        if (numNodo==688) { S_LonLat = "-99.093518,19.357630";}
        if (numNodo==687) { S_LonLat = "-99.101225,19.355807";}
        if (numNodo==686) { S_LonLat = "-99.109416,19.364580";}
        if (numNodo==691) { S_LonLat = "-99.107677,19.373296";}
    }
}
```

```

if (numNodo==692) { S_LonLat = "-99.109577,19.379125";}
if (numNodo==693) { S_LonLat = "-99.112119,19.388112";}
if (numNodo==694) { S_LonLat = "-99.113448,19.398080";}
if (numNodo==716) { S_LonLat = "-99.126172,19.406512";}
if (numNodo==703) { S_LonLat = "-99.178907,19.406070";}
if (numNodo==704) { S_LonLat = "-99.168513,19.405881";}
if (numNodo==709) { S_LonLat = "-99.144838,19.407068";}
if (numNodo==725) { S_LonLat = "-99.113077,19.408351";}
if (numNodo==1066) { S_LonLat = "-99.103137,19.408495";}
if (numNodo==727) { S_LonLat = "-99.091230,19.408387";}
if (numNodo==734) { S_LonLat = "-99.082458,19.407154";}
if (numNodo==736) { S_LonLat = "-99.069533,19.404433";}
if (numNodo==737) { S_LonLat = "-99.059438,19.398715";}
if (numNodo==733) { S_LonLat = "-99.046309,19.391111";}
//TREN LIGERO
if (numNodo==214) { S_LonLat = "-99.142212,19.343992";}
if (numNodo==213) { S_LonLat = "-99.143502,19.340819";}
if (numNodo==212) { S_LonLat = "-99.141916,19.335774";}
if (numNodo==211) { S_LonLat = "-99.140646,19.331677";}
if (numNodo==210) { S_LonLat = "-99.139359,19.327531";}
if (numNodo==209) { S_LonLat = "-99.138146,19.323728";}
if (numNodo==208) { S_LonLat = "-99.138766,19.317907";}
if (numNodo==207) { S_LonLat = "-99.140547,19.312590";}
if (numNodo==206) { S_LonLat = "-99.143047,19.307170";}
if (numNodo==205) { S_LonLat = "-99.147029,19.301771";}
if (numNodo==204) { S_LonLat = "-99.150697,19.297437";}
if (numNodo==203) { S_LonLat = "-99.146775,19.288664";}
if (numNodo==202) { S_LonLat = "-99.139704,19.282708";}

// METROBUS
//Linea 1
if (numNodo==1007) { S_LonLat = "-99.186150,19.304081";}
if (numNodo==1072) { S_LonLat = "-99.187494,19.314642";}
if (numNodo==1008) { S_LonLat = "-99.188523,19.322916";}
if (numNodo==1009) { S_LonLat = "-99.190068,19.340615";}
if (numNodo==1010) { S_LonLat = "-99.187764,19.346768";}

... ..
... ..
... ..

imprime = S_LonLat;
S_LonLat = "";

return imprime;
}
}

```