



UNIVERSIDAD MICHOACANA DE SAN NICOLÁS DE HIDALGO

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

Mat. Luis Manuel Rivera Gutiérrez

UNA SIMULACIÓN BASADA EN SUMO DE LA DINÁMICA VEHICULAR DEL CENTRO

HISTÓRICO DE LA CIUDAD DE MORELIA

TESIS

QUE PARA OBTENER EL TÍTULO DE

Licenciado en Ciencias Físico Matemáticas

PRESENTA

Fernando Obed Guillen Reyes

ASESORES:

DR. EN CS. MATEMÁTICAS FRANCISCO JAVIER DOMÍNGUEZ MOTA

DR. EN CS. DESARROLLO REGIONAL JAIME SAAVEDRA ROSALES

-

MORELIA MICHOACÁN, ABRIL 2014



En Memoria de Paulina Guillen Reyes
1991-1993

Agradecimientos

Al pensar a quién agradecer este logro, me di cuenta que todo lo que he logrado se lo debo a mis padres Fernando y Rosa María, a mis hermanos Claudia, Paulina (q.e.p.d.), Misael y Nahum, a mis abuelitos Rodolfo y María (gechu), pues al reflexionar acerca de este asunto, realmente lo único que yo hice fue estudiar y pasar exámenes, todo lo demás lo hicieron ellos. Tal vez hubo épocas en las que tuvimos problemas y tiempos difíciles, pero al final recibí un apoyo incondicional de mis padres, hermanos y abuelos. Nunca terminaré de agradecer a mis abuelitos la confianza, apoyo y cariño que me dieron durante mis estudios en la UMSNH, todas esas mañanas en que me invitaron a almorzar y tomar un café, todas las veces en que mi abuelita gechu cuidó de mí cuando me enferme, y que sin importar lo ocupada que estuviera, siempre se preocupaba por mi salud.

Quiero agradecer especialmente al Dr. Francisco Domínguez Mota, por darme la oportunidad de trabajar con él, le agradezco por todos los conocimientos compartidos, por todo el apoyo académico y moral que me brindó desde el momento que lo conocí, el cual me ayudó a superar obstáculos que yo me ponía a mí mismo, gracias al Dr. Francisco Domínguez Mota por orientarme en este negocio tan maravilloso e impresionante que son las Matemáticas Aplicadas.

Agradezco mucho a mi mejor amigo Jose Manuel Fuerte, por su gran amistad y apoyo moral cuando lo más lo necesite, es como mi tercer hermano, pues él me enseñó muchas cosas, además de que vivimos grandes momentos desde que eramos un par de adolescentes, buenos momentos que son inolvidables. También agradezco a mis amigos a Miguel, Paco e Iliana, pues también viví grandes momentos en mi vida con ellos, que me hicieron adquirir buenos recuerdos que jamás olvidare, a todos ellos los considero amigos de por vida.

También quiero agradecer los profesores José Gerardo Tinoco, Lourdes Guerrero, Karina Figueroa, David Meza, Mario Suárez, Armando Sepulveda, Roberto García, Gabriel Arrollo, José Vega y Adán Cortés por todo el apoyo, paciencia y conocimientos brindados durante mis estudios. Sin Olvidar a mis compañeros de cubículo Erika, Gerry, Lucy, Belem, Geraldine, Yunuen, Michel, Ale y Pablo, pues me orientaron y ayudaron durante mi estancia en este departamento de Matemáticas Aplicadas, un departamento que esta lleno de camaradería y conocimiento. En mi estancia en estas oficinas tuve buenas e inolvidables ex-

periencias, por ejemplo Erika, Lucy, Belem y Geraldine fueron lo mas cercano a tener 4 esposas, pues siempre me ragañaban por fumar y no comer bien, sin olvidar los consejos del Gerry y el liderazgo de Lucy y Erika, la verdad les agradezco mucho, pues durante este corto tiempo aprendí mucho de ustedes.

Resumen

En este trabajo se propone un modelo basado en SUMO aplicado al centro histórico de la ciudad de Morelia, el cual delimitamos en una zona constituida por las calles Madero, Nocupétaro y Morelos Norte, en la que se genera a diario un alto nivel de tráfico que es afectado frecuentemente por inconvenientes impredecibles como las manifestaciones, los accidentes viales o las reparaciones de vialidades o servicios, y algunas veces por eventos predecibles como la celebración de una festividad o un evento deportivo. Esto es lo que motivó el presente estudio, que tiene por objetivo el mejoramiento de niveles de servicio en un día normal a través del diseño a futuro de estrategias de contingencia, de tal suerte que en caso de que la red vial se vea afectada por alguno de los inconvenientes ya mencionados, el impacto en el tráfico sea mínimo.

Primero, mostramos una breve introducción y descripción del problema, así como de nuestro hermoso centro histórico, después señalamos los antecedentes de modelos matemáticos usados anteriormente, y del modelo matemático utilizado en este trabajo para luego entrar en detalle de algunos conceptos e ideas utilizadas en la Ingeniería de Tránsito, las cuales sirven para ponernos en el contexto de esta área.

Posteriormente, se desarrolla toda una metodología diseñada para el estudio del problema planteado, usando los siguientes tres paquetes de software: OSM (OpenStreetMap Creative Commons 2.0 (CC BY-SA)), JOSM (Java OpenStreetMap Editor version 4878) y SUMO (Simulation of Urban Mobility version 0.16.0). Se muestran tres ejemplos de aplicación de dicha metodología, donde la complejidad de las aplicaciones aumenta. El trabajo aquí presentado nos permite, eventualmente, ser capaces del estudio del transporte urbano en todo el centro histórico.

Palabras Clave : SUMO, Simulación, Centro Histórico , Sector República.

Abstract

In this thesis, we propose a SUMO based model to study the mobility of the Morelia's downtown area which is delimited by the Madero, Morelos and Nocupétaro avenues. The traffic in this area is highly crowded and often affected by unpredictable problems such as demonstrations, road accidents or repairs, and sometimes by scheduled events such as the celebration of a holiday or some sport event. This paper aims at improving service levels in a normal day by taking into account all the mentioned scenarios in order to design strategies to increase the traffic flow and mobility and, at the same time, to decrease the impact of the contingencies.

First, we show a little description of the problem and of our beautiful downtown area; second, we discuss the mathematics and modelling schemes used in this work. We also explain some useful concepts and ideas in order to understand the Traffic Engineering context.

Then, we develop a methodology using three software packages: OSM Creative Commons 2.0 (CC BY-SA), JOSM version 4878, and SUMO version 0.16.0. Finally, we show three applications that use the methodology which allow us to study the urban mobility of our downtown of interest.

Key words : SUMO, Simulation, Morelia's downtown, Republica sector.

Índice general

1. Planteamiento del Problema	9
1.1. El Centro Histórico de la Ciudad de Morelia	9
1.2. La Red Vial del Centro Histórico	11
1.3. El Sector República	12
2. Modelado del problema y conceptos de dinámica vehicular	19
2.1. Introducción	19
2.2. Enfoques macroscópico y microscópico	19
2.3. Modelado Matemático Lighthill-Whitham-Richards	20
2.4. Ecuación sin viscosidad de Burguers	23
2.5. Ecuación de Momentum del tipo Navier-Stokes	24
2.6. Solución Numérica de la ecuación de Burguers con el esquema de Lax-Wendroff	25
3. OpenStreetMap (OSM) y JOSM	27
3.1. Introducción	27
3.2. Características de OSM	27
3.3. Recopilación de datos	29
3.4. Formato de datos de OSM	29
3.5. Java OpenStreetMap Editor (JOSM)	29
3.5.1. ¿Cómo comprender un mapa en JOSM para su construc- ción o edición?	30
3.5.2. Consejos para la construcción del mapa	32
4. Simulation of Urban Mobility (SUMO)	43
4.1. Introducción a SUMO	43
4.1.1. Uso de SUMO	43
4.2. Creación básica de un modelo usando SUMO	46
4.2.1. Parte geométrica del modelo (mapa)	47
4.3. Creación óptima de un modelo de transporte urbano utilizando OSM,JOSM y SUMO	50
4.3.1. Creando el mapa de nuestro modelo usando OSM y JOSM	50
4.3.2. La demanda de transporte urbano	53
4.3.3. Archivos de configuración	55

4.3.4. Datos de salida de una simulación en SUMO	55
5. Aplicaciones	67
5.1. Introducción	67
5.1.1. Primera aplicación	67
5.1.2. Segunda aplicación	70
5.1.3. Tercera aplicación	72
6. Conclusiones	85
7. Bibliografía	87

Índice de figuras

1.1. Delimitación de la Zona del Centro Histórico Fuente: Facultad de Ingeniería Civil. UMSNH	10
1.2. Sector República	13
1.3. Clasificación funcional de un sistema vial	15
3.1. Modo estandar.	33
3.2. Modo ciclista.	34
3.3. Modo humanitario.	35
3.4. Modo mapquest.	36
3.5. Modo transporte.	37
3.6. Interfaz de JOSM.	38
3.7. Zona de estudio en JOSM.	39
3.8. Ejemplos de grafos conexos	40
3.9. Ejemplos de grafos no conexos	40
3.10. Ejemplo de un mapa de una intersección construida en JOSM . .	41
3.11. Ejemplo de una intersección real en JOSM, los nodos verdes no representan intersecciones, solo sirven para darle forma a las aristas.	41
4.1. Visualizando una intersección como un grafo en SUMO	47
4.2. Archivo ejemplo_sin_osm_josm.nod.xml	48
4.3. Archivo ejemplo_sin_osm_josm.edg.xml	48
4.4. Vista Completa del mapa creado solo con SUMO	57
4.5. Vista de cercana del mapa creado solo con SUMO	58
4.6. Archivo ejemplo_sin_josm.rou.xml	59
4.7. Archivo ejemplo_sin_josm.sumo.xml	60
4.8. Simulación en SUMO del ejemplo descrito en esta sección	61
4.9. Interfaz de JOSM seleccionando el mapa a descargar	62
4.10. Mapa inicial del Sector República descargado y guardarlo usando JOSM	63
4.11. Mapa del Sector República editado y corregido en JOSM	64
4.12. Conexiones de la intersección, del ejemplo visto en la sección anterior.	65
4.13. Archivo simulacion_centro.typ.xml	66
4.14. Archivo de configuración simulacion_centro.sumo.cfg	66

4.15. Archivo simulacion.centro.sumo.xml , donde agregamos el archivo de configuracion automovil.xml para obtener resultados de la simulación	66
5.1. Planteamiento del recorrido en la primera aplicación	76
5.2. Archivo primera_plicacion_4.flows.xml	77
5.3. Intersecciones del problema, la intersección encerrada por el círculo es en la cual mostramos como definir las probabilidades . . .	78
5.4. Archivo simulacion.centro.turn.xml que contiene la definición de probabilidades en las intersecciones	79
5.5. Grafica de los tiempos de recorrido	80
5.6. Descripción del problema de la segunda aplicación	81
5.7. Descripción del mapa y zona de estudio en la Tercera Aplicación	82
5.8. Rutas de recorrido en la Tercera Aplicación	83
5.9. Glorieta en la intersección de la Av. Madero y la Av. Michelena .	84

Capítulo 1

Planteamiento del Problema

1.1. El Centro Histórico de la Ciudad de Morelia

El centro histórico representa el corazón de la ciudad, testimonio vivo de la historia local, sitio de grandes acontecimientos sociales y políticos, ámbito que se ve enriquecido por la relación física y visual con su patrimonio edificado, su entorno natural y la presencia de tradiciones particulares de Morelia. Esta zona de Monumentos Históricos comprende una arquitectura civil y religiosa que destaca por su monumentalidad y relevancia, construidas entre los siglos XVII y XIX.

Debido a su importancia, gran parte de la actividad económica, social y gubernamental se lleva a cabo en el centro histórico. En términos prácticos, esto implica desplazar gran cantidad de gente desde y hacia el centro, además de los vehículos que sólo atraviesan el mismo.

Para estudiar la zona del centro histórico de la ciudad de Morelia empezamos con lo siguiente, el centro histórico comprende 482,30 hectáreas, en la figura (1.1) se puede observar. Esta zona se divide en los siguientes sectores:

- Sector República
- Sector Revolución
- Sector Nueva España
- Sector Independencia

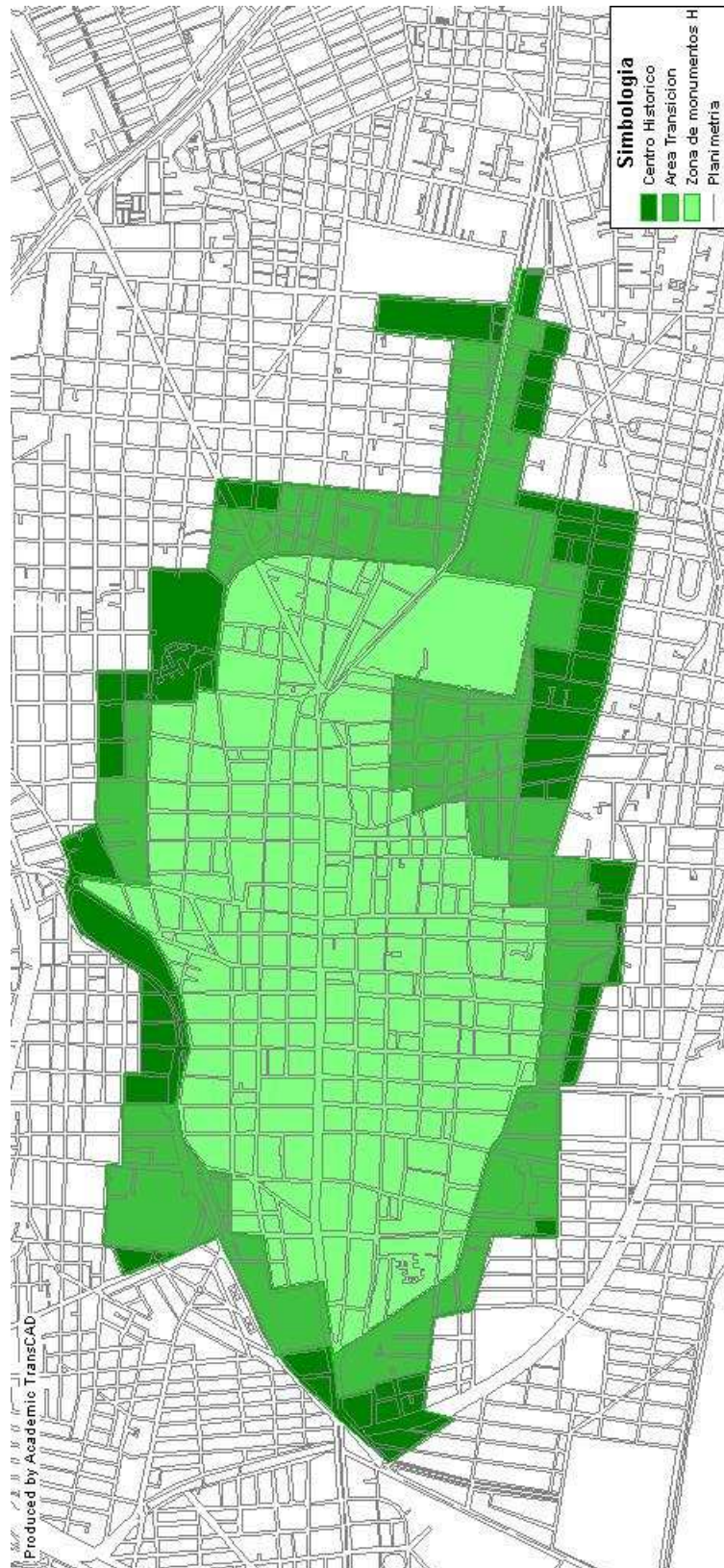


Figura 1.1: Delimitación de la Zona del Centro Histórico
Fuente: **Facultad de Ingeniería Civil. UMSNH**

1.2. La Red Vial del Centro Histórico

En esta sección introducimos los conceptos de la ingeniería de tránsito que emplearemos en este trabajo. Para poder hablar de una vialidad en ese contexto, es necesario definir el concepto de camino, entendiéndose como tal aquella faja de terreno acondicionada para el tránsito de vehículos. La denominación de camino incluye a nivel rural las llamadas carreteras, y a nivel urbano las calles de la ciudad. Ciertamente, uno de los patrimonios más valiosos con los que cuenta cualquier país es la infraestructura de su red vial, por lo que su magnitud y calidad representan uno de los indicadores del grado de desarrollo del mismo. Se encontrará siempre que un país de un alto nivel de vida tendrá un excelente sistema vial, un país atrasado tendrá una red deficiente. El diseño geométrico de las carreteras y calles, incluye todos aquellos elementos relacionados con el alineamiento horizontal, el alineamiento vertical y los diversos componentes de la sección transversal, como se mencionará enseguida.

La vialidad urbana del Centro Histórico de la ciudad de Morelia, corresponde a un sistema ortogonal de calles, en el sentido urbanístico, más o menos rectas, con anchos que oscilan de los 3 a los 20 metros y con discontinuidades producto de los remates arquitectónicos, usadas para realzar la importancia de determinadas edificaciones pero que dificultan el flujo vehicular (Programa Parcial del Centro Histórico. 1997. Instituto de Desarrollo Municipal, Morelia, Michoacán.)

Este estilo se reprodujo imperfectamente en la medida que la ciudad fue creciendo, sufriendo adaptaciones derivadas de los accidentes topográficos como los ríos o las pendientes de las lomas. De esta manera, los alineamientos de las calles actuales presentan frecuentes cambios de dirección, los anchos de calzada de sus calles y avenidas no son constantes y comúnmente también existen discontinuidades en las arterias, derivadas de estilos arquitectónicos como los llamados remates arquitectónicos y topográficos y, más recientemente sobre todo en la periferia de las manchas urbanas, son manifiestas las discontinuidades con el resto de la ciudad, debido a la anarquía de los nuevos asentamientos. Conjuntamente, un importante flujo vehicular circula diariamente en el Centro Histórico, esta concentración vehicular está estrechamente vinculada con los motivos de viaje, generados por los usos del suelo y los equipamientos existentes en el Centro Histórico.

Ante el crecimiento acelerado y explosivo de vehículos particulares y públicos, tanto las calles locales como las colectoras y parte de las calles principales han sido ocupadas mayoritariamente por vehículos estacionados que dificultan tanto la circulación vehicular, como los cruces peatonales, convirtiendo al Centro Histórico en una **isla** de difícil acceso y hasta en un lugar peligroso para los transeúntes. Dada la demanda social existente, es necesario recuperar la esencia de este espacio como un elemento de recreación y de conexión urbana. Adicionalmente, es factible y deseable diversificar su uso sin desvirtuar su esencia, aprovechándolo para generar otro tipo de equipamiento urbano.

Es por lo anterior que consideramos que es necesario proveer al Centro Histórico de accesos continuos, rápidos y seguros, no sólo para los vehículos sino también para los peatones y personas discapacitadas.

1.3. El Sector República

Enseguida describiremos la zona de nuestro interés, que es el **Sector República**, esta zona del centro histórico está comprendida entre la **Av. Madero**, **Av. Morelos** y la **Av. Nocupétaro**, la figura (1.2) muestra esta zona.

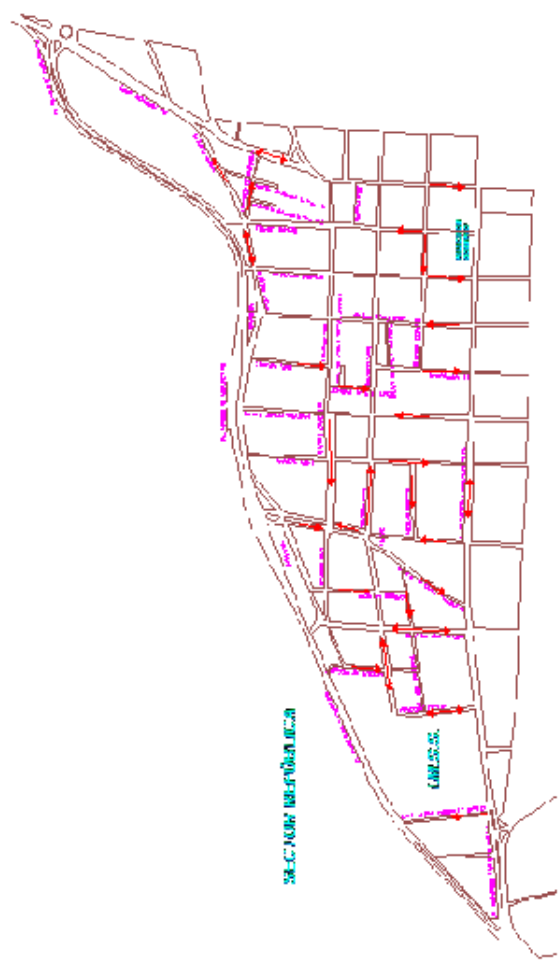


Figura 1.2: Sector República

Este sector esta formado por 23 calles, y son las siguientes:

- Melchor Ocampo
- Nicolás Romero
- Santiago Tapia
- Juan José Codallos
- Eduardo Ruiz
- Mártires de Tacubaya
- General García Pueblita
- Avenida Héroes de Nocupétaro
- Benito Juárez
- Ignacio Zaragoza
- Guillermo Prieto
- El Nigromante
- Juan Álvarez
- Miguel Bernal Jiménez
- Valentín Gómez Farías
- León Guzmán
- Guadalupe Victoria
- Lerdo de Tejada
- Carlos Salazar
- Vicente Rivapalacio
- Nicolás de Regules
- Francisco Zarco
- Jesús Gonzlez Ortega

Dentro de un criterio amplio de planeación, la red vial, tanto rural como urbana, se debe de clasificar de tal manera que se puedan fijar funciones específicas a las diferentes carreteras y calles, para así atender las necesidades de movilidad de personas y mercancías, de una manera rápida, confortable y segura, y a las necesidades de accesibilidad a las distintas propiedades o usos del área colindante.

Para facilitar la movilidad es necesario disponer de carreteras y calles rápidas, y para tener acceso es indispensable contar con carreteras y calles lentas. Naturalmente entre estos dos extremos aparece todo el sistema de carreteras y calles. En términos generales, las carreteras y las calles urbanas, pueden clasificarse funcionalmente en tres grandes grupos: principales (arterias), secundarias (colectoras) y locales, que es precisamente la **clasificación vial**. La figura (1.3) presenta en forma gráfica los grados de movilidad y acceso de un sistema vial. En un extremo, las carreteras y calles principales son de accesos controlados destinados a proveer alta movilidad y poco o nulo acceso a la propiedad lateral, mientras que, en el otro extremo, las carreteras y calles locales son de accesos no controlados que proveen fácil acceso a la propiedad lateral, pero raramente las utiliza el tránsito de paso. Como vemos, a mayor movilidad hay menor accesibilidad, y a mayor accesibilidad menor movilidad. La clasificación funcional

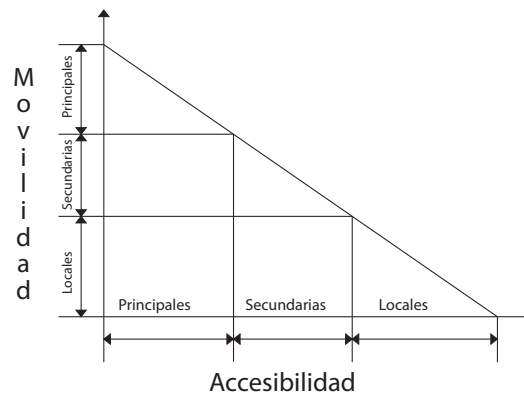


Figura 1.3: Clasificación funcional de un sistema vial

es clave en el proceso de planeación del transporte, ya que agrupa las distintas carreteras y calles en clases o sistemas de acuerdo al servicio que se espera que presten, y contribuye a la solución de muchos problemas mediante:

- La determinación de la importancia relativa de las distintas carreteras y calles.
- El establecimiento de las bases para la asignación de niveles de servicio o especificaciones de proyecto.
- La evaluación de deficiencias, comparando la geometría actual o los niveles de servicio con las especificaciones.
- La determinación de las necesidades resultantes.
- La estimación de los costos de las mejoras.

Con la clasificación funcional es posible lo siguiente:

- Establecer sistemas integrados de una manera lógica, agrupando todas las carreteras y calles que deben estar bajo una misma jurisdicción debido al tipo de servicio que ofrecen.
- Asignar responsabilidades para cada clase de camino en el ámbito gubernamental.
- Agrupar las carreteras y las calles que requieren el mismo grado de ingeniería y competencia administrativa.
- Relacionar las especificaciones geométricas del proyecto con cada tipo de carretera o calle.
- Establecer las bases para programas a largo plazo, implementación de prioridades y planeación fiscal.

Sistema vial urbano En la figura anterior también se ilustra, en términos de movilidad y accesibilidad, la clasificación de un sistema vial urbano. Con el propósito de unificar y simplificar la nomenclatura, se sugiere la siguiente clasificación:

- *Autopistas y vías rápidas:* Las autopistas son las que facilitan el movimiento expedito de grandes volúmenes de tránsito entre áreas, a través o alrededor de la ciudad o área urbana. Son divididas, con control total de sus accesos y sin comunicación directa con las propiedades colindantes. Una autopista tiene separación total de los flujos conflictivos, en tanto que una vía rápida puede o no tener algunas intersecciones a desnivel, pero puede ser la etapa anterior de una autopista. Estos dos tipos de arterias forman parte del sistema o red vial primaria de un área urbana.
- *Calles principales:* son las que permiten el movimiento del tránsito entre áreas o partes de la ciudad. Dan servicio directo a los generadores principales de tránsito, y se conectan con el sistema de autopistas y vías rápidas. Con frecuencia son divididas y pueden tener control parcial de sus accesos. Las calles principales se combinan entre sí para formar un sistema que mueve el tránsito en toda la ciudad, en todas las direcciones.
- *Calles colectoras:* son las que ligan las calles principales con las calles locales, proporcionando a su vez acceso a las propiedades colindantes.
- *Calles locales:* Proporcionan acceso directo a las propiedades, sean estas residenciales, comerciales, industriales o de algún otro uso; además de facilitar el tránsito local. Se conectan directamente con las calles colectoras y /o con las calles principales.

Conceptos principales de la ingeniería de tránsito

Además para estar más centrados en la **Ingeniería de Tránsito**, dos conceptos básicos: Primero definimos los conceptos que son básicos:

AUTOPISTAS Y VÍAS RÁPIDAS	No existen en Morelia
AVENIDAS	Avenida Héroes de Nocupétaro
CALLES PRINCIPALES	Avenida Francisco I. Madero Poniente
CALLES COLECTORAS	Cuautla Vicente Riva Palacio Santiago Tapia
CALLES LOCALES	Melchor Ocampo Eduardo Ruiz Nicolás Romero Jose Codallos Sebastián Lerdo de Tejada Benito Juárez Ignacio Zaragoza Guillermo Prieto Valentín Gómez Farías León Guzmán Carlos Salazar Nicolás de Regules Francisco Zarco J. González Ortega El Nigromante Juan Álvarez Miguel Bernal Jiménez García Pueblita Mártires de Tacubaya

Tabla 1.1: Propuesta

- **Aforo** : Cantidad de vehículos que pasa por un punto específico.
- **Nivel de Servicio**: Medida cualitativa que representa el funcionamiento de una vialidad, tomando en cuenta factores como velocidad, libertad de maniobra, comodidad y seguridad.

donde agregamos el hecho de que existen 6 niveles de servicio, los cuales son **Nivel A**, **Nivel B**, **Nivel C**, **Nivel D**, **Nivel E** y **Nivel F**, donde la calidad de nivel de servicio va decreciendo, siendo el nivel A el mejor y el F el peor.

Conociendo ya los diferentes criterios disponibles en la **Ingeniería de Tránsito** para la clasificación de una red vial urbana se formuló la siguiente propuesta para la red vial en el Sector República del Centro Histórico, la cual se muestra en la tabla 81.1).

Las avenidas, calles principales y colectoras son las arterias más importantes porque facilitan la movilidad del tráfico y necesitan mayor libertad de los carriles de circulación para evitar congestionamientos. En contraparte las calles locales son importantes solamente para propiciar mayor accesibilidad.

Capítulo 2

Modelado del problema y conceptos de dinámica vehicular

2.1. Introducción

En este capítulo entramos de lleno a cómo se modela un problema de tráfico vehicular, así como la teoría manejada, relacionándonos con la ingeniería de tránsito; por tanto, la importancia de este capítulo es que entraremos en contexto a lo que se aborda desde la perspectiva matemática y también desde la perspectiva de ingeniería de tránsito. Al hacer la modelación matemática de un problema o fenómeno real, lo primero que hacemos es observar las variables existentes y más relevantes del problema, para después elegir algunas de ellas para modelar el problema, en este capítulo abordamos las distintas perspectivas que hay para observar en el problema de tránsito vehicular, así como el modelado matemático.

2.2. Enfoques macroscópico y microscópico

Al comenzar el análisis del problema de transporte urbano de nuestra bella ciudad de Morelia, nos podemos dar cuenta que lo podemos analizar principalmente de dos maneras, podemos observar el problema fijándonos solamente en una avenida o en la red vial de toda la nuestra ciudad, que incluiría a miles de intersecciones, a esto lo llamamos analizar el problema de manera microscópica y de manera macroscópica, respectivamente; a continuación explicamos esto con más detalle.

Enfoque Macroscópico

En el enfoque macroscópico se aborda el problema de tránsito vehicular de manera más global, es decir, se aborda no como el análisis de un grupo pequeño de calles y avenidas, sino como el análisis de toda una ciudad como nuestra bella ciudad de Morelia. Al analizar el problema de transporte urbano de manera macroscópica se hace de una manera parecida al comportamiento de un flujo compresible en un régimen. Este enfoque macroscópico tiene algunas ventajas como:

- Resultan ser menos costosos computacionalmente hablando a comparación de los modelos con enfoque microscópico.
- Generalmente los resultados obtenidos muestran una buena relación con el comportamiento empírico que se observa en el tránsito.

Como es de esperarse, al analizar el problema de transporte urbano de manera macroscópica, se utilizan variables de contexto más global, es decir, variables de carácter promedio (de cantidades promediadas). Las variables que son consideradas como relevantes no son generales en todos los modelos, esto es porque no se estudia lo mismo en todos los modelos de transporte urbano, por tanto, las variables relevantes en un modelo en específico dependen de que se va a estudiar. Algunos ejemplos de variables en el enfoque macroscópico es la densidad $\rho(x, t)$, la velocidad promedio $v(x, t)$, etcétera.

Enfoque microscópico

Este enfoque como el nombre lo dice, es más específico y focalizado que el macroscópico. En los modelos microscópicos de transporte urbano se estudia una sección pequeña de una ciudad, que puede constar a lo mas de algunas decenas de calles y avenidas, un ejemplo de esto puede ser el estudio en una zona como el **Sector República**. La razón por la que no es conveniente analizar alguna zona grande desde el punto de vista microscópico, es que las variables serían de caracter microscópico, y el estudio de este tipo de variables es costoso computacionalmente hablando, por lo que entre más grande sea la zona a estudiar desde este enfoque, más sera el costo computacional. Por lo general al usar este enfoque, solamente se estudia zonas consistentes de un grupo muy pequeño de intersecciones viales.

2.3. Modelado Matemático Lighthill-Whitham-Richards

Para empezar el modelado del problema de tránsito, aclaramos que esto lo haremos desde la perspectiva macroscópica, por lo que primero definimos algunas variables que permiten aclarar el contexto desde el cual analizamos el problema,

como lo son la densidad, la velocidad y el flujo vehicular, las cuales se denotan por ρ , v y F respectivamente. Como vimos en la sección anterior, al analizar el problema de transporte de manera macroscópica, lo hacemos siguiendo la analogía con a un problema de un fluido compresible, pues si viéramos el transporte urbano por medio de imágenes de satélite en tiempo real, podríamos observar que el comportamiento del transporte urbano es similar.

Sean $\rho(x, t)$ y $F(x, t)$ las funciones de **densidad** y **flujo** en un lugar y tiempo arbitrarios x y t , la función de densidad denota la cantidad de vehículos por unidad de longitud (por ejemplo km) en $x \in \mathbb{R}$ y en un tiempo determinado $t \geq 0$, por otra parte el flujo denota la cantidad de vehículos que pasan por $x \in \mathbb{R}$ en un tiempo $t \geq 0$. La ecuación de continuidad que representa a un fluido es:

$$\frac{\partial}{\partial t} \rho(x, t) + \frac{\partial}{\partial x} F(x, t) = \sum_{i=0}^{i=F_{entradas}} \alpha_i(x - x_i, t) + \sum_{j=0}^{j=F_{salidas}} \beta_j(x - x_j, t),$$

en donde el par de miembros del lado derecho de la ecuación describen las entradas y salidas de la red vial modelada, las cuales tienen lugar en x_i y x_j , respectivamente.

La ecuación anterior la podemos escribir en la siguiente forma:

$$\frac{\partial}{\partial t} \rho(x, t) + \frac{\partial}{\partial x} F(x, t) = \sum_{i=0}^{i=F_{entradas}} \alpha_i(t) \phi_i(x - x_i) + \sum_{j=0}^{j=F_{salidas}} \beta_j(t) \phi_j(x - x_j),$$

tal que:

$$\begin{aligned} \alpha_i(x - x_i, t) &= \alpha_{0,i}(t) \phi_i(x - x_i) \\ \beta_j(x - x_j, t) &= \beta_{0,j}(t) \phi_j(x - x_j), \end{aligned}$$

en donde las funciones $\phi_i(x - x_i)$ y $\phi_j(x - x_j)$ representan la distribución en el espacio, las funciones $\alpha_{0,i}(t)$ y $\alpha_{0,j}(t)$ representan las variaciones en el tiempo.

Si además planteamos la función $v(x, t)$, que denota la **velocidad** de los vehículos en x y t , entonces podemos plantear una relación entre la densidad y el flujo, la cual es:

$$F(x, t) = v(x, t) \rho(x, t).$$

Por tanto, la ecuación de continuidad queda de la siguiente manera:

$$\frac{\partial}{\partial t} \rho(x, t) + \frac{\partial}{\partial x} (v(x, t) \rho(x, t)) = \sum_{i=0}^{i=F_{entradas}} \alpha_i(t) \phi_i(x - x_i) + \sum_{j=0}^{j=F_{salidas}} \beta_j(t) \phi_j(x - x_j).$$

También podemos plantear otras variables tales como, **el número de vehículos** N en el intervalo (x_1, x_2) y la **variación de vehículos** A en el intervalo (x_1, x_2) y tiempo t como.

$$N = \int_{x_1}^{x_2} \rho(x, t) dx,$$

$$A = \frac{d}{dt} \int_{x_1}^{x_2} \rho(x, t) dx.$$

Por simplicidad imaginamos que en nuestro modelo no hay entradas ni salidas de flujo, es decir, por simplicidad supondremos que:

$$\sum_{i=0}^{i=F_{entradas}} \alpha_i(x - x_i, t) + \sum_{j=0}^{j=F_{salidas}} \beta_j(x - x_j, t) = 0,$$

por lo que la Ecuación Diferencial Parcial (EDP) de continuidad es:

$$\frac{\partial}{\partial t} \rho(x, t) + \frac{\partial}{\partial x} (v(x, t) \rho(x, t)) = 0.$$

Al plantear esta última idea, es como si imaginamos una carretera de un sólo carril como una línea recta, por ejemplo el eje x en el espacio y que los vehículos deben conservarse, es decir, el numero de vehículos que entra al intervalo (x_1, x_2) debe ser igual al numero de vehículos que lo deja, entonces tenemos la siguiente igualdad, que a veces es llamada ecuación de conservación:

$$\frac{d}{dt} \int_{x_1}^{x_2} \rho(x, t) dx = \rho(x_2, t) v(x_2, t) - \rho(x_1, t) v(x_1, t)$$

dado que

$$\frac{d}{dt} \int_{x_1}^{x_2} \rho(x, t) dx = \int_{x_1}^{x_2} \partial_t \rho(x, t) dx$$

y también como:

$$\rho(x_2, t) v(x_2, t) - \rho(x_1, t) v(x_1, t) = - \int_{x_1}^{x_2} \partial_x (\rho(x, t) v(x, t)) dx$$

entonces sustituyendo en la ecuación de conservación, tenemos

$$\int_{x_1}^{x_2} \partial_t \rho(x, t) dx = - \int_{x_1}^{x_2} \partial_x (\rho(x, t) v(x, t)) dx$$

integrando en un intervalo de tiempo (t_1, t_2) la ecuación anterior, tenemos que

$$\int_{t_1}^{t_2} \int_{x_1}^{x_2} \partial_t (\rho(x, t)) dx dt = - \int_{t_1}^{t_2} \int_{x_1}^{x_2} \partial_x (\rho(x, t) v(x, t)) dx dt.$$

Los valores x_1, x_2, t_1 y t_2 son arbitrarios, por tanto podemos simplificar la ecuación anterior como:

$$\rho_t - (\rho v)_x = 0$$

la cual es EDP que debe ser complementada con la condición inicial $\rho(x, 0) = \rho_0(x)$ $x \in \mathbb{R}$.

Es importante señalar que la densidad ρ y velocidad v no pueden ser del todo arbitrarias, es decir, $0 \leq \rho \leq \rho_{max}$ y $0 \leq v \leq v_{max}$, donde ρ_{max} y v_{max} son la densidad máxima y velocidad máxima respectivamente, pues esto aterriza el modelo un poco mas a la realidad, además podemos señalar que hay una relación intuitiva entre ρ y v , pues si $\rho = 0$, entonces no hay vehículos, por lo que podemos movernos con una velocidad $v = v_{max}$, análogamente, si $\rho = \rho_{max}$, entonces no podemos movernos, pues la vialidad estaría a su máxima capacidad, por tanto $v = 0$, la relación lineal mas simple entre ρ y v es:

$$v(\rho) = v_{max}\left(1 - \frac{\rho}{\rho_{max}}\right).$$

Sustituyendo en la ecuación diferencial parcial obtenida anteriormente, y resumiendo todos los conceptos formulados, tenemos el siguiente modelo:

$$\rho_t - [\rho v_{max}\left(1 - \frac{\rho}{\rho_{max}}\right)]_x = 0,$$

en donde

$$v(\rho) = v_{max}\left(1 - \frac{\rho}{\rho_{max}}\right) \quad , \quad 0 \leq \rho \leq \rho_{max},$$

el cual es llamado **Modelo de Lighthill-Whitham-Richards**.

2.4. Ecuación sin viscosidad de Burgers

En la sección anterior vimos el planteamiento del modelado matemático utilizado en este trabajo, el cual es el modelo de Lighthill-Whitham-Richards, ahora veremos una simplificación del este modelo para empezar a indagar en la solución numérica.

El modelo consistía en

$$\rho_t - [\rho v_{max}\left(1 - \frac{\rho}{\rho_{max}}\right)]_x = 0,$$

donde

$$v(\rho) = v_{max}\left(1 - \frac{\rho}{\rho_{max}}\right) \quad y \quad 0 \leq \rho \leq \rho_{max}.$$

Haciendo los siguientes cambios de variables en la EDP del modelo de Lighthill-Whitham-Richards:

$$v_{max} = \frac{L}{\tau} \quad , \quad x_s = \frac{x}{L} \quad , \quad t_s = \frac{t}{\tau} \quad , \quad u = 1 - \frac{2\rho}{\rho_{max}}$$

tenemos la siguiente ecuación con condición inicial:

$$u_t + \left(\frac{u^2}{2}\right)_x = 0 \quad , \quad x \in \mathbb{R} \quad , \quad t \geq 0$$

$$u(x, 0) = u_0(x) \quad , \quad x \in \mathbb{R} \quad , \quad u_0 = 1 - \frac{2\rho_0}{\rho_{max}}.$$

El problema de EDP con condición inicial anterior es conocido como **ecuación**

sin viscosidad de Burguers, y es una simplificación del modelo matemático de Lighthill-Whitham-Richards.

2.5. Ecuación de Momentum del tipo Navier-Stokes

Ahora haremos una mejora al modelos de Lighthill-Whitham-Richards, la cual consiste en plantear una relación distinta entre la densidad ρ y el flujo F , anteriormente se plateó al flujo en términos de la densidad, es decir, $F(\rho) = v\rho$, pero en la nueva relación consideramos también a ρ_x de la densidad, por tanto planteamos la nueva igualdad:

$$F(\rho) = f(\rho) - D\rho_x$$

en donde D es una constante positiva, notemos además que con esta nueva relación, si ρ es constante, entonces F también lo será, algunas observaciones adicionales con esta nueva ecuación son, por ejemplo, si $\rho_x > 0$ ($\rho_x < 0$), implica que F es menor (mayor), y por tanto la velocidad de los vehículos se reducirá (aumentará).

Si sustituimos la nueva relación entre F y ρ en la ecuación de continuidad vista en la sección anterior, tenemos lo siguiente:

$$\begin{aligned}\frac{\partial}{\partial t}\rho(x,t) + \frac{\partial}{\partial x}F(x,t) &= 0 \\ \frac{\partial}{\partial t}\rho(x,t) + \frac{\partial}{\partial x}(f(\rho) - D\rho_x) &= 0 \\ \frac{\partial}{\partial t}\rho(x,t) + v_g\frac{\partial}{\partial x}\rho(x,t) &= D\frac{\partial^2}{\partial x^2}\end{aligned}$$

en donde

$$v_g = \frac{d}{d\rho}f(\rho).$$

La no linealidad y difusión de la ecuación anterior tiene efectos opuestos, por ejemplo, el término v_g implica el **pronunciamiento** y la **ruptura** de la onda, mientras que el término del lado derecho **suaviza** el perfil. Si $D = 0$, entonces se tiene un ancho distinto de cero en la onda de choque.

Siguiendo el mismo razonamiento que se utilizó con el flujo, también podemos hacerlo con la velocidad, en otras palabras se propone que

$$v(x,t) = v_0(\rho) - \frac{D}{\rho}\frac{\partial\rho(x,t)}{\partial x},$$

en donde

$$v_0(\rho) = \frac{f(x,t)}{\rho(x,t)}.$$

2.6. Solución Numérica de la ecuación de Burgers con el esquema de Lax-Wendroff

En esta sección abordamos la forma de obtener una solución numérica de una EDP no lineal, como la ecuación de **Burgers**. El analizar una EDP no lineal numéricamente es en ocasiones complejo, la ecuación de Burgers no es la excepción.

Para resolver numéricamente una EDP en una región Ω del plano xt , primero discretizamos Ω , si nuestra región es

$$\Omega = \{(x, t) : 0 \leq x \leq 1, \quad 0 \leq t \leq 1\}$$

entonces podemos discretizarla de la manera siguiente

$$L = \{u_i^j = (x_i, t_j) : 0 \leq i \leq n, 0 \leq j \leq m, x_i = ih, t_j = jk, h = \frac{1}{n}, k = \frac{1}{m}, n, m \in \mathbb{N}\}.$$

Es en cada punto $u_i^j \in L$ en donde obtenemos la aproximación, es decir, u_i^j será la aproximación de $u(x_i, t_j)$.

Para realizar esta solución numérica, se utilizan métodos numéricos llamados esquemas, hay esquemas de primer y segundo orden, que tienen diferencia en la precisión. Para la ecuación sin viscosidad de Burgers se utiliza el esquema de segundo orden de **Lax-Wendroff**, el cual consiste en

$$\begin{aligned} u_i^{j+1} &= u_{i+1}^j - \frac{k}{4h}((u_{i+1}^j)^2 - (u_{i-1}^j)^2) + \frac{k^2}{2h^2}(\frac{1}{4}(u_i^j + u_{i+1}^j)((u_{i+1}^j)^2 - (u_i^j)^2) \\ &\quad - \frac{1}{4}(u_i^j + u_{i-1}^j)((u_i^j)^2 - (u_{i-1}^j)^2)) \end{aligned}$$

Capítulo 3

OpenStreetMap (OSM) y JOSM

3.1. Introducción

Entre los software que utilizamos en el desarrollo de este trabajo, uno de los más importantes fue OpenStreetMap, pues es donde se empieza a crear nuestro modelo, OSM es la abreviación de **OpenStreeMap**, que es un proyecto de colaboración en el que participan miles de usuarios de todo el mundo inscritos en el proyecto OSM. Este proyecto fue fundado por Steve Coast y tiene como propósito construir la información geográfica de todo el mundo, y que esta sea gratuita al descargarla, o en otras palabras, que seamos capaces de obtener el mapa de casi cualquier ciudad del planeta.

El Proyecto comenzó el 4 de junio de 2004 y la cantidad de usuarios inscritos al proyecto OSM crece en promedio un 10 % al mes, a tal grado que en 2013 ya se contaba con 1 000 000 de usuarios registrados en el proyecto OSM. Debemos destacar que Alemania y el Reino Unido son los países con mayor concentración de usuarios, pero hay usuarios de todo el mundo.

Los usuarios contribuyen al menos una vez, ya sea editando o construyendo los mapas, no siempre lo hacen a mano, la información para construir los mapas se recopila usando dispositivos GPS, por lo que el trabajo de los usuarios es más que nada editar, en una de las siguientes secciones abordaremos la parte de edición de los mapas obtenidos en OSM.

3.2. Características de OSM

Al hablar de características de OSM, nos referimos principalmente a la interfaz que encuentra en la página www.openstreetmap.org , no abordaremos en cómo descargar los datos, pues la interfaz de OSM es muy intuitiva, pues simplemente

se selecciona el paquete a descargar.

Por tanto, en lo que a características se refiere, mencionaremos que el manejo de la interfaz de OSM es casi análogo a otras páginas que proporcionan información semejante, pero existen algunas características peculiares, por ejemplo, se puede visualizar el mapa de todo el mundo. Además hay varias formas de visualizar las figuras (3.1 , 3.2 , 3.3 , 3.4 , 3.5):

El modo que usamos fue el estándar, pues es simple y se ajustaba a nuestros propósitos, además debemos resaltar que en la interfaz existen links que nos lleva a páginas de OSM más complejas en las cuales podemos descargar incluso el mapa de todo un país (de forma comprimida); existen varios formatos, ya depende de nuestros propósitos el como descargar la información, de nuevo resalto que en cuanto a características, OSM es bastante intuitivo.

3.3. Recopilación de datos

Anteriormente mencionamos que la recopilación de datos es principalmente por medio de dispositivos **GPS** en bicicletas, vehículos, etcétera y los datos son recopilados en campo por usuarios inscritos en el proyecto.

A veces son indexadas algunas imágenes de puntos especiales (turismo principalmente). Hay usuarios que cartografían la zona donde residen hasta completarla, y aunque suene extraño, suelen organizarse fiestas entre usuarios para recopilar información. Es por esto que el proyecto ha avanzado mucho, pues por las contribuciones realizadas por miles de usuarios de todo el mundo, es probable que en una década el proyecto esté casi completo.

3.4. Formato de datos de OSM

El formato de datos utilizados en OSM es similar al que se usa en el lenguaje **XML**, donde **XML** significa **Extensible Markup Language**, este lenguaje es un lenguaje de banderas, fue desarrollado por **World Wide Web Consortium**. Este lenguaje es utilizado para almacenar datos de forma legible (parecido a **HTML**), **XML** permite estructurar documentos pesados y a diferencia de otros lenguajes **XML** da soporte a bases de datos, una de las principales ventajas de **XML** es que permite la comunicación entre aplicaciones.

El lenguaje **XML** se propone como el estándar para la comunicación entre distintas plataformas, incluso **XML** ha sido utilizado para crear otros lenguajes. Algunas de las desventajas de **XML** es que han sido criticadas sus extensiones por su nivel de detalle y la complejidad.

3.5. Java OpenStreetMap Editor (JOSM)

Ahora empezamos a hablar del segundo software utilizado, que fue de gran importancia por su practicidad, el cual es Java OpenStreetMap Editor (JOSM), que consiste en un editor de OSM. En este software podemos construir y editar los mapas de OSM, de hecho es el editor que los usuarios de OSM utilizan para editar los mapas de OSM.

Al igual que OSM, JOSM tiene una interfaz que es bastante intuitiva, con una breve revisión de las características de la interfaz basta para saber como manejarlo. Este software es descargado gratuitamente de internet. Podemos pregun-

tarnos: ¿para qué utilizar un editor de OSM si el mapa ya está construido y lo podemos descargar directamente de OSM?. Pues el detalle es que aunque OSM es un gran recurso de información, los mapas están en forma de paquetes, es decir se descargan determinadas zonas completas de alguna región y no de alguna región pequeña en específico, en cambio, utilizando JOSM podemos seleccionar cualquier zona en específico del planeta, por pequeña que sea, nuestra zona de estudio es solo una parte del centro de nuestra ciudad, por esto es que fue muy práctico el uso de JOSM en este proyecto.

Además y más importante, OSM puede tener errores, pues a veces los mapas de una región pueden haber sido creados y editados por personas de una región totalmente distinta, personalmente en el mapa que nosotros descargamos y utilizamos en este proyecto, tenía bastante información equivocada, por lo que JOSM fue una gran herramienta, pues directamente corregimos errores como por ejemplo: el que falte una calle, falta de semáforos, sentidos de las calles equivocados y señalización vial incorrecta, si no contáramos con JOSM, hubiéramos tenido que aprender la estructura de OSM dentro de **XML** y corregir un archivo de cerca de 6 000 líneas de código a mano y hubiéramos tardado meses en terminar tan solo la edición del mapa, en cambio utilizando JOSM llevamos a cabo la edición en 4 días, pues fue necesario aprender primero a usar JOSM y recopilar la información vial correcta, una herramienta de JOSM que puede llegar a ayudar o ser útil para la visualización es que podemos poner un a OSM como fondo de pantalla de la interfaz JOSM, o si queremos construir un mapa que no esté en OSM pero que tengamos una imagen del mapa que queramos, simplemente configuramos el fondo de pantalla de JOSM con la imagen del mapa y que aparezca el mapa en la escala correcta, además de que cada segmento creado en JOSM tiene una medida en metros, la cual se ve en la parte inferior de JOSM, así como las coordenadas geográficas. La interfaz de JOSM se visualiza en la figura (3.6).

3.5.1. ¿Cómo comprender un mapa en JOSM para su construcción o edición?

Podemos tener dos escenarios: el primero sería tener ya el mapa descargado de OSM utilizando JOSM o construir un mapa desde cero, después de descargar el mapa usando JOSM, solo vemos una línea y puntos, como si fuera una especie de rejilla; por ejemplo, en la figura (3.7) se muestra nuestra zona de estudio en JOSM, por tanto, la pregunta inicial al usar JOSM es ¿cómo comprender la representación de un mapa en JOSM, para así construirlo o editarlo?; es por este motivo que escribimos esta sección.

Observemos que existe una analogía entre un mapa en JOSM y el concepto matemático **Grafo**, naturalmente con algunas ideas adicionales. Empleando la a la teoría de grafos tenemos varias definiciones:

- **Grafo** : Un grafo **G** es una estructura matemática formada por los conjuntos $A = \{a_1, a_2, a_3, a_4, \dots, a_m\}$ y $E = \{e_1, e_2, e_3, e_4, \dots, e_n\}$, que contienen

las aristas y los nodos respectivamente.

- **Nodo** : Un nodo es la estructura o unidad fundamental de un grafo (podemos visualizarlo como un punto).
- **Arista** : Una arista de grafo es una relación entre dos nodos (podemos visualizarla como una línea entre un par de nodos).
- **Trayectoria** : Una trayectoria entre un par de nodos e_i y e_j es una sucesión de nodos (pueden estar repetidos los nodos) donde el primer nodo de la sucesión es e_i y el último es e_j .
- **Trayectoria Simple** : Una trayectoria simple es una trayectoria en la que no se repiten los nodos.
- **Grafo conexo** : Un grafo es conexo si existe una trayectoria simple entre cualquier par de nodos del grafo.

Cualquier grafo lo podemos visualizar como un grupo de puntos (nodos), donde algunos de ellos están unidos con líneas (aristas). Para entender los grafos conexos mostramos en las figuras (3.8) y (3.9), algunos ejemplos de grafos conexos y de grafos no conexos, en estas imágenes se presentan primero dos grafos conexos, y vemos que basta con quitar una arista para que un grafo deje de ser conexo:

Vemos que podemos entender a los grafos conexos como grafos que **constan de una sola pieza**, y a los grafos no conexos como grafos que **vienen en varias piezas**, o vista esta idea de otra forma, también podemos visualizar a los grafos no conexos como grafos formados por uno o más grafos conexos.

A un mapa en JOSM podemos entenderlo como un grafo conexo, en el cual algunas aristas serán dirigidas (podemos imaginarlas como flechas) y algunas serán simples (una línea simple), las aristas representan las calles y los nodos las intersecciones entre estas, es por esto que a las calles con un solo sentido se representan con aristas dirigidas (flechas), y las calles que sean de ambos sentidos se representan con aristas simples, en la figura (3.10) se muestra un ejemplo de una intersección simple, donde las calles son de un solo sentido y además se tiene un par de accesos adicionales:

Sabemos que en la realidad las calles no son siempre rectas, por ejemplo hay curvas, por lo que habrá nodos que no serán la representación física de una intersección, sino la lógica, estos nodos solo tienen como propósito darle forma a las calles, en JOSM podemos agregar tantos de estos nodos a una arista (calle) para darle forma real, en la figura (3.11) se muestra la diferencia, y este tipo nuevo de nodos en color verde para darle forma a los accesos:

Estas son las ideas que complementan la analogía entre grafos y mapas en JOSM. Teniendo esto claro, estamos listos para construir o editar mapas en JOSM, que serán posteriormente utilizados por el tercer y último software (SUMO) que se describe en el siguiente capítulo.

3.5.2. Consejos para la construcción del mapa

En la creación de cualquier mapa de una red vial aconsejamos usar JOSM, pues solo es necesario descargar de OSM el mapa (usando JOSM) y editarlo por la practicidad de JOSM, tal vez hasta este capítulo no se vea cual es la practicidad de JOSM en este proyecto, pero en el capítulo donde veamos el software que utilizamos para las simulaciones, es cuando se comprenderá lo útil y práctico que es JOSM e implícitamente OSM. Otro consejo es que se debe asegurar que la parte geométrica de nuestro modelo sea correcta, pues como lo mencionamos, pueden existir errores en nuestro mapa, también hay que asegurar que el mapa en JOSM siempre se debe guardar con el formato

```
<nombre\_del\_mapa>.osm.xml
```

Por último hay que tener cuidado, pues siempre que cerremos JOSM (depués de trabajar) preguntará si **guardamos y subimos, solo guardamos o cancelamos**, recomiendo solo guardar el mapa, pues el modelo es nuestro, además de que si no eres usuario del proyecto, no podrás subir ninguna alteración al mapa.

Es de gran importancia señalar que al construir o editar un mapa en JOSM, a cada objeto del mapa(líneas o puntos) JOSM le asignará un número de identificación único, incluso es único en OSM, este número de identificación es creado en parte con las coordenadas geográficas del objeto; este número de identificación es abreviado como **id** y no puede ser modificado.

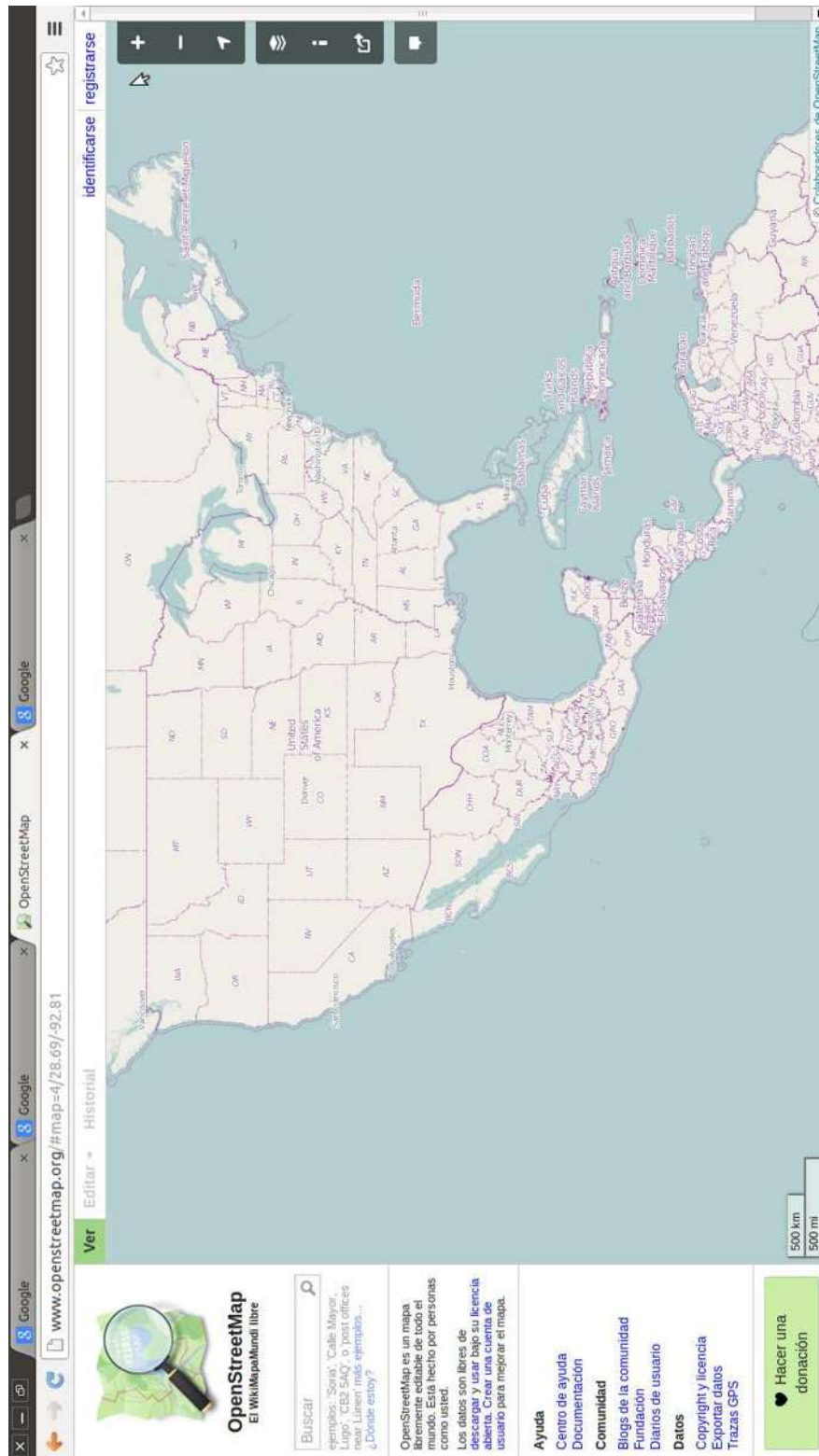


Figura 3.1: Modo estandar.

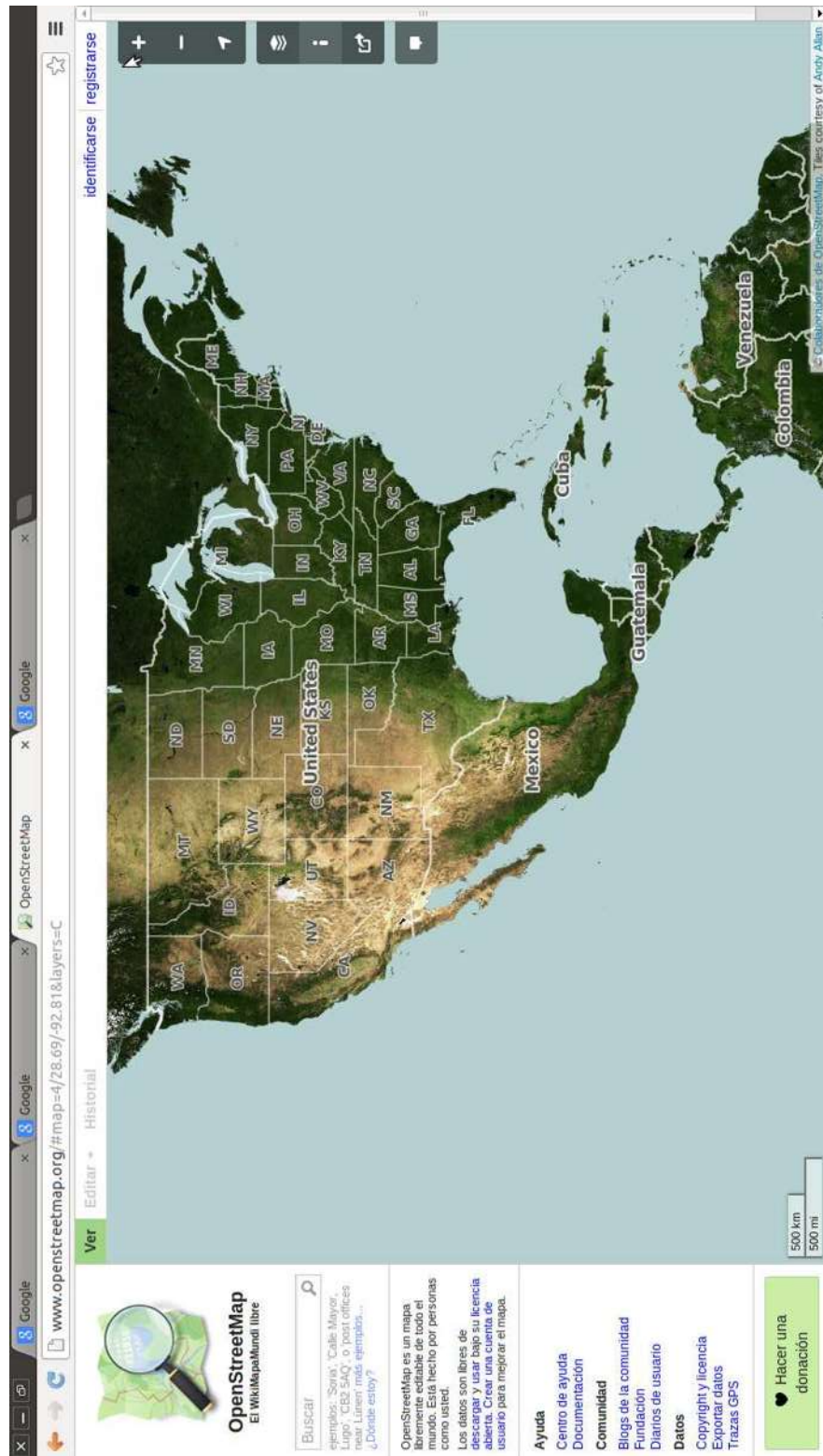


Figura 3.2: Modo ciclista.

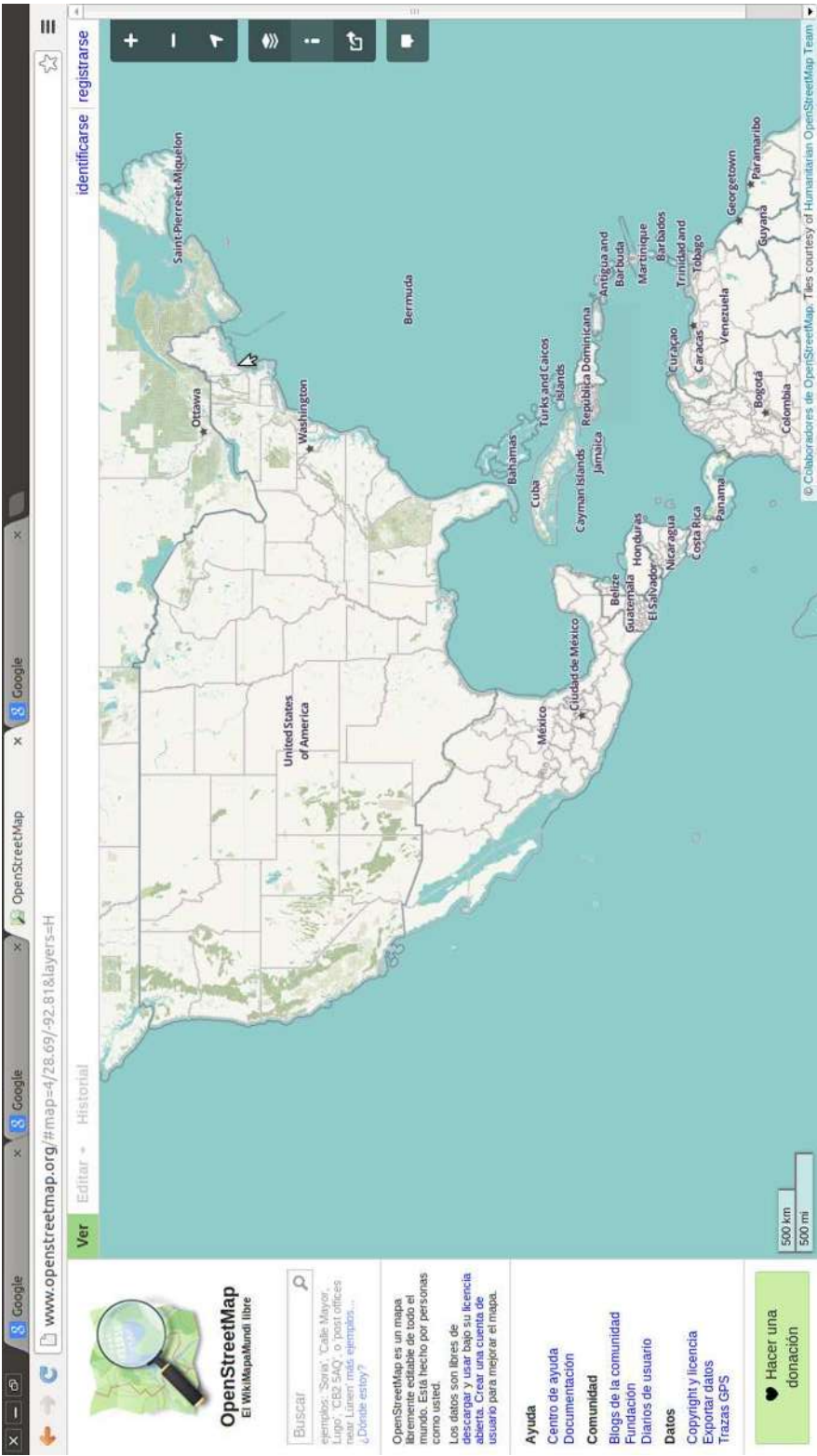


Figura 3.3: Modo humanitario.

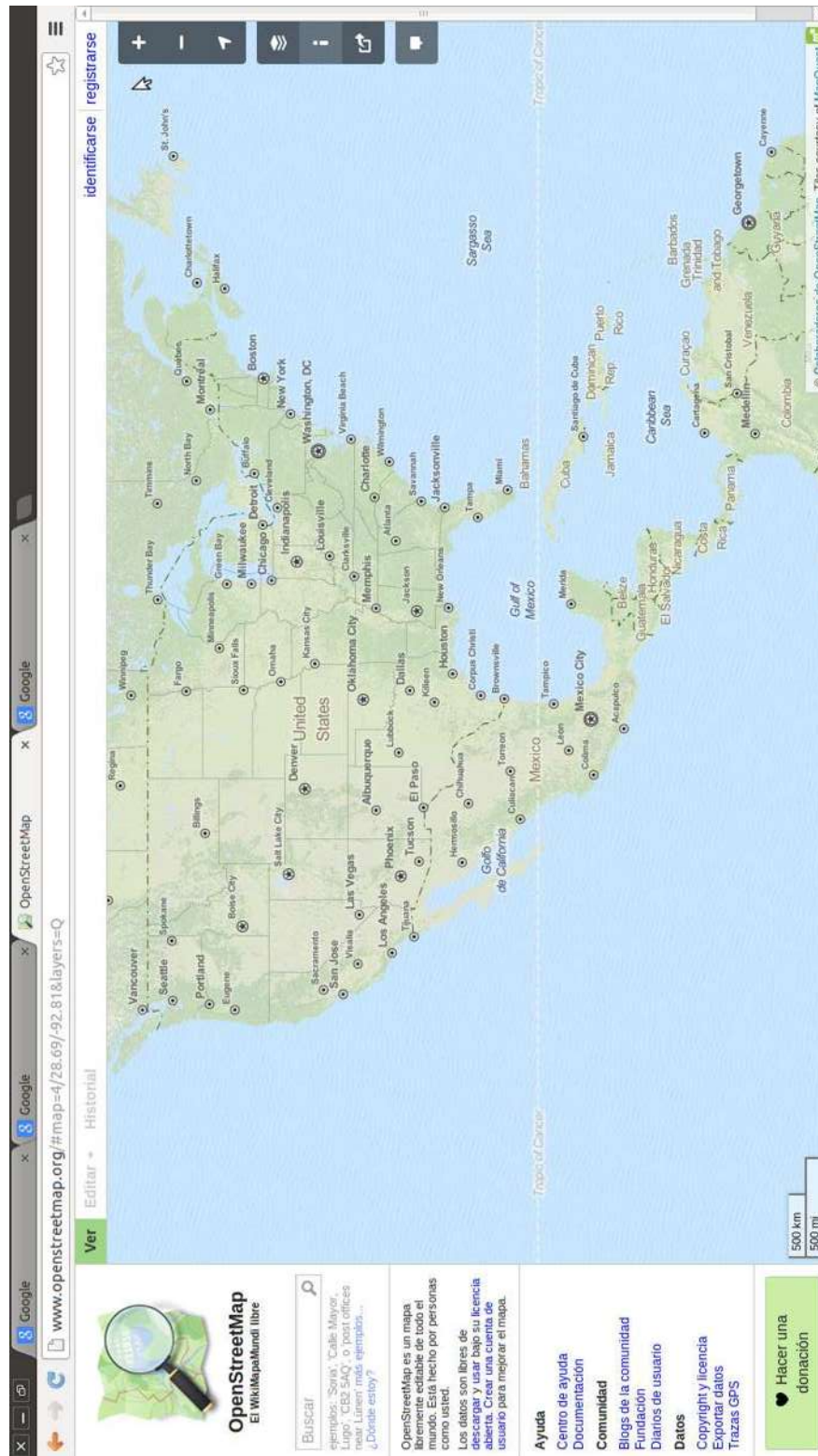


Figura 3.4: Modo mapquest.

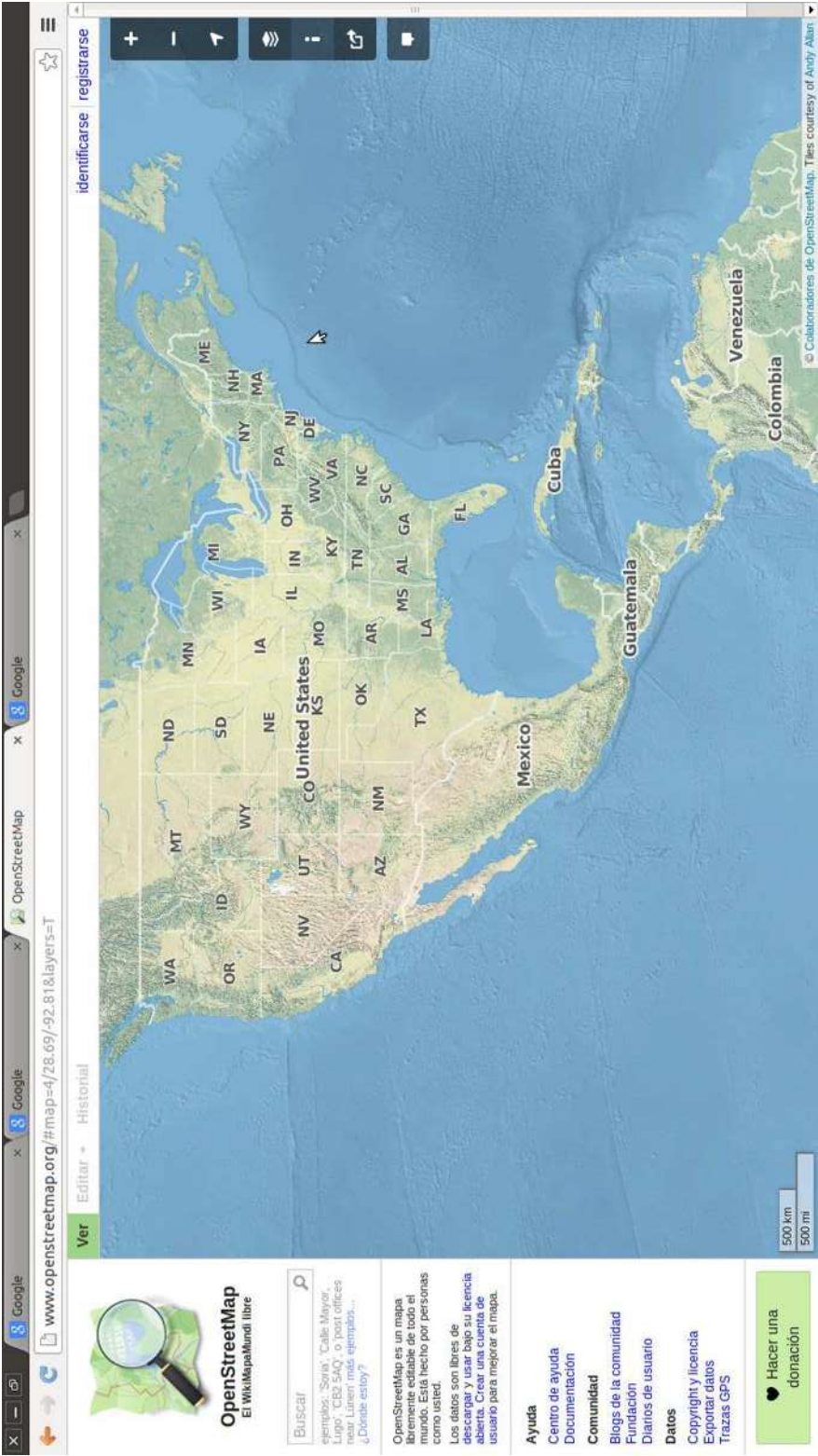


Figura 3.5: Modo transporte.

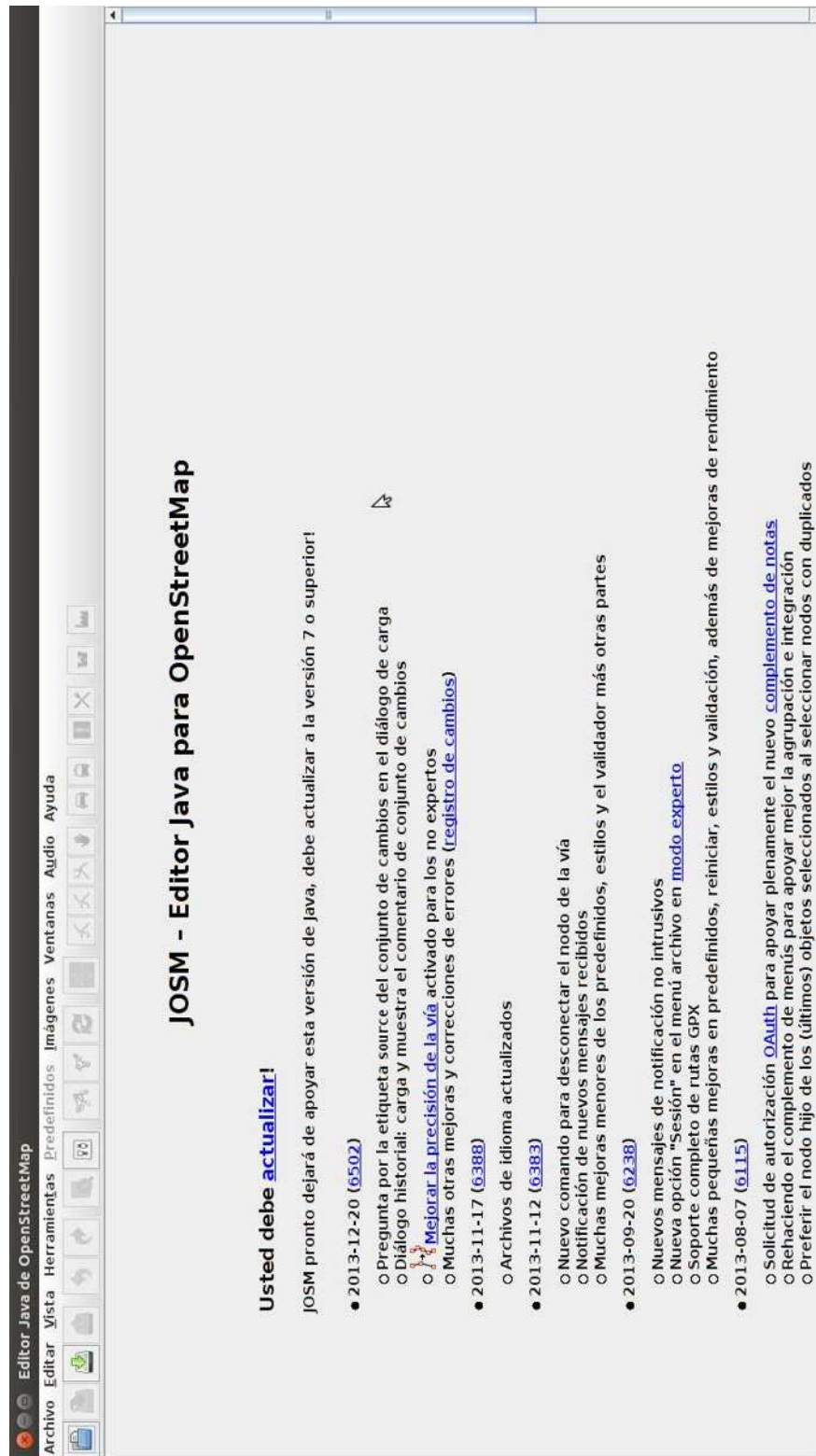


Figura 3.6: Interfaz de JOSM.

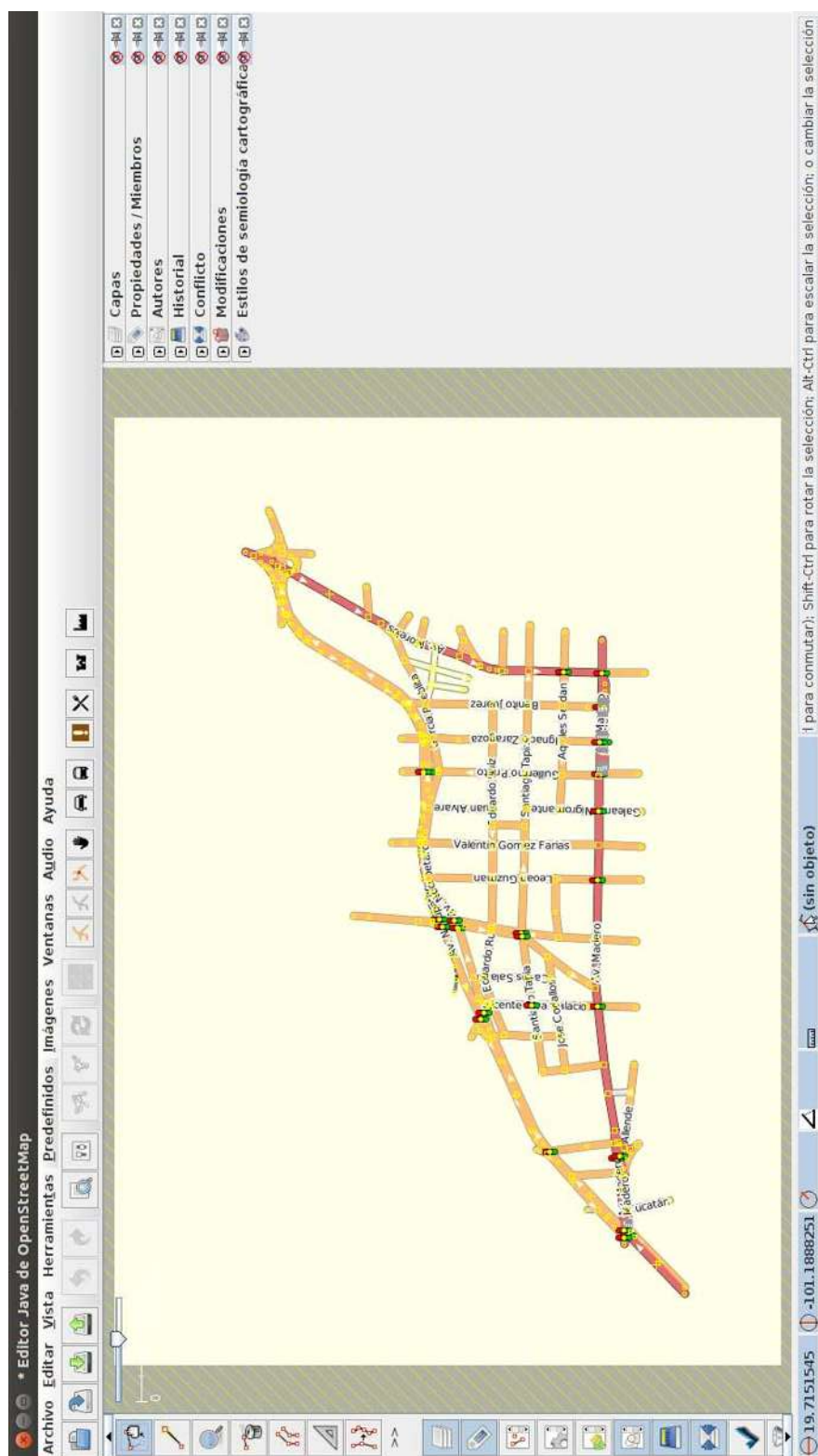


Figura 3.7: Zona de estudio en JOSM.

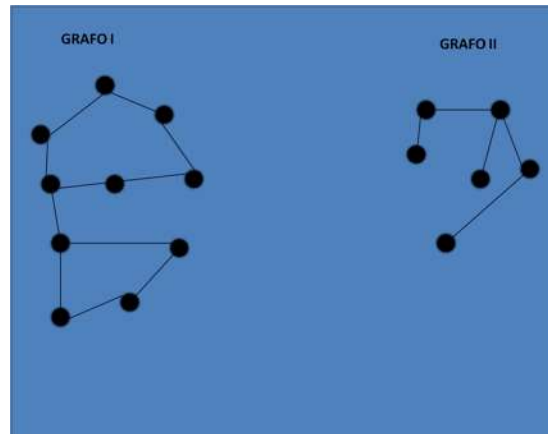


Figura 3.8: Ejemplos de grafos conexos

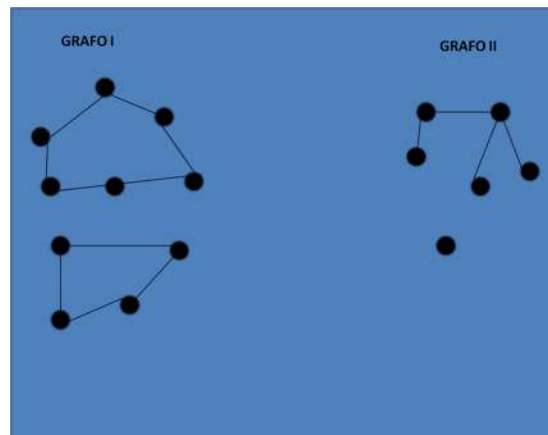


Figura 3.9: Ejemplos de grafos no conexos

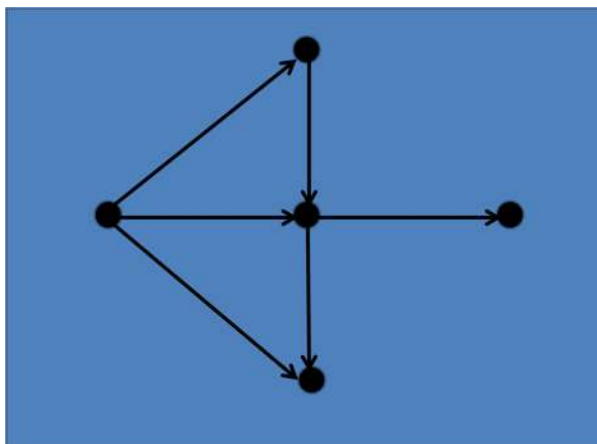


Figura 3.10: Ejemplo de un mapa de una intersección construida en JOSM

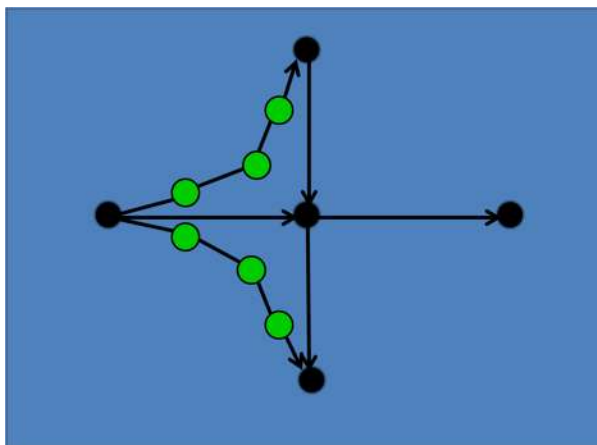


Figura 3.11: Ejemplo de una intersección real en JOSM, los nodos verdes no representan intersecciones, solo sirven para darle forma a las aristas.

Capítulo 4

Simulation of Urban Mobility (SUMO)

4.1. Introducción a SUMO

SUMO es la abreviación de **Simulation of Urban Mobility**. Este es el software donde se realizan las simulaciones de transporte urbano, fue creado y es desarrollado por el **Instituto de Sistemas de Transporte del Centro Aeroespacial Alemán** (ISTCAA), es un software gratuito que cuenta con muchas herramientas muy prácticas, es altamente portable, puede ser utilizado en Windows y Ubuntu. Está diseñado para microsimulaciones de tráfico, pero si se cuenta con grandes recursos computacionales se pueden llevar a cabo simulaciones de zonas mas grandes, esto es porque el problema de optimizar el transporte urbano es un problema altamente complejo, y si se quiere simular pues será también altamente complejo computacionalmente hablando. La única desventaja de SUMO es que cuenta con poca documentación, y la que tiene es un poco breve, es decir, a veces no se explica a detalle en la documentación de SUMO como usarlo, pero en nuestra opinión esto era de esperarse pues SUMO fue creado en 2011, en otras palabras, es un software relativamente nuevo que ha sido desarrollado rápidamente. Existen otros tipos de software para realizar simulaciones de transporte urbano, pero a diferencia de SUMO, estos no son gratuitos.

4.1.1. Uso de SUMO

Entrando de lleno a como usar SUMO, podemos decir que es un software que tiene varios modos de operación, una manera podría ser que SUMO es un paquete que se compone de comandos (rutinas, subprogramas o como se prefiera entender) que se utilizan desde una terminal, algunos de estos comandos generan datos y otros abren aplicaciones que utilizan estos datos, por lo que podemos decir que se complementan entre si, la tabla (4.1) muestra los principales co-

mandos en SUMO:

Todos los archivos que necesitamos proporcionar a SUMO para hacer una

Principales comandos de SUMO	Descripción
sumo	Inicializa la simulación de una forma no visual en la terminal
sumo-gui	Inicializa la simulación de forma gráfica, es decir, abre una interfaz donde se puede manipular la simulación
netconvert	Importa o convierte modelos (mapas) de otros formatos y los transforma a formatos que SUMO pueda manejar
netgenerate	Genera modelos (mapas) aleatorios que pueden ser usados directamente en SUMO
duarouter	Genera rutas de vehículos utilizando definiciones de demanda de transporte y el modelo (mapa)
jtrrouter	Genera rutas para vehículos utilizando definición de probabilidades y de flujos (aforos)

Tabla 4.1: comandos en SUMO

simulación tienen formato **XML**, la única diferencia entre los archivos es la extensión, que para SUMO es muy importante, pues al leer la extensión, SUMO sabrá que esperar en el archivo. Por poner un ejemplo, entre los archivos que SUMO necesita para una simulación está el archivo que contiene las rutas de los vehículos que aparecen en la simulación, este archivo debe tener una extensión ***.rou** y como está en formato **XML**, entonces la estructura del nombre del archivo es:

```
<nombre_del_archivo>.rou.xml
```

Aquí es donde resaltamos la importancia de las extensiones en los archivos que utiliza SUMO, pues además de ayudar a SUMO a saber que esperar, nos ayuda mentalmente a visualizar lo que necesitamos para lograr una simulación en particular; la tabla (4.2) muestra las principales extensiones para los archivos en SUMO(4.2):

Todo el trabajo aquí presentado lo hicimos en **UBUNTU versión 12.04, en LINUX**, pues es más cómodo trabajar en la terminal (shell) de Ubuntu, además de que podemos usar SUMO desde cualquier directorio, debemos aclarar que sólo se necesita un conocimiento básico de la terminal en Ubuntu, así como saber manejar desde la misma archivos, programas y comandos. Para hacer todo

Extensiones	Contenido
*.net.xml	Los archivos que tengan esta extensión contienen la parte geométrica del modelo, o en otras palabras, el mapa del modelo
*.nod.xml	Esta extensión es para el archivo que contenga la definición de los nodos
*.edg.xml	Esta extensión es para el archivo que contenga las aristas del mapa
*.con.xml	Existe un archivo en el cual se definen las maniobras permitidas en una intersección, es decir, las posibles formas de avanzar en una intersección, propias de los carriles
*.rou.xml	El archivo con esta extensión contiene la definición de las clases de vehículos, de las rutas y de cada vehículo que aparezca en la simulación
*.turn.xml	El archivo con esta extensión contiene las probabilidades de cada intersección en el sentido que, al llegar un vehículo a una intersección determinada, que probabilidad (en la intersección) hay que el vehículo gira hacia la derecha, que probabilidad de que siga de frente y que probabilidad hay de que el vehículo gire a la izquierda. No hay que olvidar que la suma de todas las probabilidades en una intersección debe ser 1, como en la teoría de la probabilidad
*.flows.xml	El archivo con esta intersección contiene los aforos de las intersecciones del mapa
*.det.xml	Para definir detectores en el mapa se crea un archivo con esta extensión
*.stat.xml	El archivo con esta extensión, tendrá información estadística que puede ser utilizada para simular demanda vehicular
*.sumo.cfg	El archivo con est extesión configura la simulación, este archivo le dirá a SUMO que mapa utilizar y que archivo de demanda de transporte urbano usar, así como el tiempo de duración de la simulación

Tabla 4.2: Extensiones de los archivos usados en SUMO

desde Windows, se necesita lo mismo, y señalar que la instalación y ejecución en SUMO es distinta, este es el motivo por el cual preferimos UBUNTU, pues de nuevo es importante destacar que a diferencia de Windows, en Ubuntu podemos

hacer simulaciones desde cualquier directorio, en Windows esto no es posible. Hay que poner mucha atención, que nosotros por practicidad, daremos por hecho que nos establecemos en un directorio en específico, y desde el cual ejecutamos todos los comandos que se señalarán en las próximas secciones, daremos por hecho también que se deja en claro que los archivos creados en las próximas secciones se guardaran en el mismo directorio, es decir, a menos que se diga lo contrario dejamos en claro que nos establecemos en un directorio y en el trabajaremos y guardaremos todo.

Por último hay que decir que SUMO requiere de ciertos archivos que debemos crear antes de llevar a cabo la simulación, por ejemplo debemos tener toda la información que utilizará la simulación como lo son el mapa (parte geométrica) y la demanda de transporte (los vehículos que aparecerán en la simulación). Para dejar esta idea lo más clara posible, podemos hablar de una acerca de los vehículos que aparecen en la simulación: cada vehículo que aparezca en la simulación debe tener definida la ruta que debe seguir en el mapa, la información del tipo de vehículos. Las rutas que debe seguir (más adelante entraremos en detalle de en que consiste la ruta de un vehículo) deben ser definidas en un archivo específico (extensión específica) antes de llevar a cabo la simulación, ya que la ruta que seguirá cada vehículo no es creada durante la simulación, y por tanto el vehículo no sabría hacia dónde dirigirse. En forma general nosotros clasificamos los archivos que utiliza SUMO en tres clases:

- Archivos estáticos
- Archivos dinámicos
- Archivos de Configuración

Explicamos brevemente cada clase. Los archivos estáticos son archivos que tienen información del mapa a utilizar, o en otras palabras la geometría de nuestra simulación, es por eso que los llamamos archivos estáticos, pues no cambian al hacer la simulación; por otro lado los archivos dinámicos son todo lo contrario, pueden cambiar cada vez que se haga la simulación, pero el nombre del dinámico también se refiere a que estos archivos contienen datos que componen la parte que se mueve en la simulación. Por último, los archivos de configuración son los archivos que configuran principalmente la simulación y los datos de salida.

4.2. Creación básica de un modelo usando SUMO

Para hacer un mapa y la simulación de tránsito hay dos formas, con OSM, JOSM y SUMO, o usando sólo SUMO. A continuación vamos a mostrar como construir un mapa usando sólo SUMO, no utilizaremos otro recurso en esta sección, esto es posible, pero nada práctico, a continuación veremos por qué. En este ejemplo, construimos una simple intersección en forma de cruz, donde cada calle es de ambos sentidos y de 2 carriles, que es el más básico, cualquier ejemplo más amplio es simplemente una extensión.

En secciones anteriores decíamos que podíamos visualizar la construcción de un mapa en JOSM como si fuera un grafo conexo, pues en SUMO es casi lo mismo, pues a diferencia de los grafos conexos (donde algunas aristas podían ser dirigidas) vistos en el capítulo anterior, para establecer una analogía entre un mapa creado en SUMO y un grafo (como los ya vistos) solo tenemos que agregar la característica de que en SUMO, al hacer un mapa, todas las aristas serán dirigidas y puede haber más de una arista entre cualquier par de nodos, algunas personas llaman esta característica en un grafo como **un grafo con multi-aristas**, por lo que sería igual al diseñar el mapa de la intersección planteado en esta sección podemos imaginarlo como un grafo con 5 nodos (4 extremos y el centro) y ocho aristas dirigidas, pues las calles son de ambos sentidos, si las calles fueran de un solo sentido, tendríamos que definir solo cuatro aristas dirigidas. La figura (4.1) muestra esta descripción:

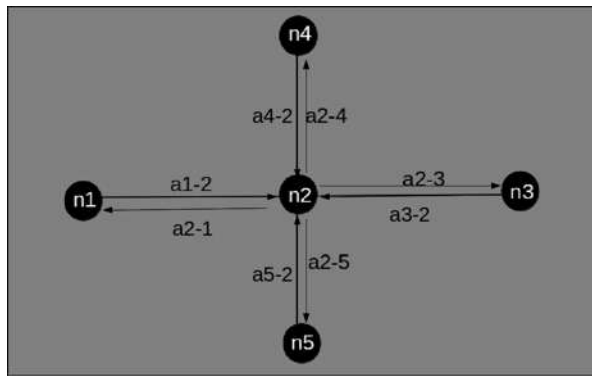


Figura 4.1: Visualizando una intersección como un grafo en SUMO

4.2.1. Parte geométrica del modelo (mapa)

Para construir nuestro mapa sin OSM y JOSM, nos basaremos en el grafo planteado en la sección anterior, es decir, vamos a hacer una intersección donde las 2 calles son de dos sentidos, pero puede haber el caso que una calle tenga “**prioridad**” sobre otra. Esta “**prioridad**” se establece ya sea porque una de las calles es principal y la otra secundaria o por algún otro motivo similar. Además imaginemos que el límite de velocidad en las calles es de 40 km/h, a continuación vemos como establecer toda esta información en nuestro mapa usando solo SUMO, debemos definir 2 archivos principalmente, los cuales llamamos:

- ejemplo_sin_osm_josm.nod.xml
- ejemplo_sin_osm_josm.edg.xml

En el archivo `ejemplo_sin_osm_josm.nod.xml` es en el que se definen los nodos de la intersección, y en el archivo `ejemplo_sin_osm_josm.edg.xml` es el que contiene las aristas, en las figuras (4.2) y (4.3) se muestran tal cuales estos archivos:

```
<nodes>

  <node id="n1" x="-400.0" y="0.0" />
  <node id="n2" x="0.0" y="0.0"/>
  <node id="n3" x="+400.0" y="0.0"/>
  <node id="n5" x="0.0" y="-400.0" />
  <node id="n4" x="0.0" y="+400.0"/>

</nodes>
```

Figura 4.2: Archivo `ejemplo_sin_osm_josm.nod.xml`

```
<edges>

  <edge id="a1-2" from="n1" to="n2" priority="1" numLanes="1" speed="40"/>
  <edge id="a2-1" from="n2" to="n1" priority="1" numLanes="1" speed="40"/>
  <edge id="a2-3" from="n2" to="n3" priority="2" numLanes="1" speed="40"/>
  <edge id="a3-2" from="n3" to="n2" priority="3" numLanes="1" speed="40"/>
  <edge id="a2-5" from="n2" to="n5" priority="4" numLanes="1" speed="40"/>
  <edge id="a5-2" from="n5" to="n2" priority="4" numLanes="1" speed="40"/>
  <edge id="a2-4" from="n2" to="n4" priority="4" numLanes="1" speed="40"/>
  <edge id="a4-2" from="n4" to="n2" priority="4" numLanes="1" speed="40"/>

</edges>
```

Figura 4.3: Archivo `ejemplo_sin_osm_josm.edg.xml`

En los archivos mostrados en las imágenes vemos que tienen formato **XML**. No hay que olvidar que los archivos que le proporcionamos a SUMO tienen este formato; vemos que en el archivo `ejemplo_sin_osm_josm.edg.xml` establecimos la prioridad y límite de velocidad que queríamos en las calles, esto se ve por ejemplo en `priority="4"` y `speed="40"`, respectivamente en la arista con `id="a2-5"`, pues cada objeto que aparezca en la simulación debe tener un nombre (`id`). Para producir nuestro mapa con estas características, debemos ejecutar el siguiente comando:

```
$netconvert - -node-files=ejemplo_sin_josm.nod.xml - -edge-files=ejemplo_sin_josm.edg.xml
- -output-file=ejemplo_sin_josm.net.xml
```

Al ejecutar el comando anterior se genera el archivo **ejemplo_sin_josm.net.xml**, el cual es nuestro mapa en SUMO. Aquí es donde vemos que los archivos escritos y generados hasta el momento en esta sección son archivos estáticos, pues son la parte geométrica de la simulación.

Podemos visualizar este mapa ejecutando el siguiente comando desde la terminal, el cual abre la interfaz gráfica de SUMO:

```
$sumo-gui -n ejemplo_sin_josm.net.xml
```

Las figuras (4.4) y (4.5) muestran nuestro mapa en la interfaz de SUMO (sumo-gui):

Ahora escribamos el único archivo dinámico, el cual es el archivo donde definiremos nuestros distintos tipos de vehículos, las rutas que seguirán y otras características. Por practicidad llamamos a este archivo **ejemplo_sin_josm.rou.xml**, la figura 4.6 muestra el archivo:

En el archivo **ejemplo_sin_josm.rou.xml** lo que hicimos fue definir los tipos de vehículos (que características tienen), las rutas posibles a seguir y cada vehículo que aparecerá en la simulación, no abordaremos en el código de este archivo, es decir, no explicaremos aquí las características de los vehículos y la definición de cada vehículo, pues esto es fácil de entender directamente de la documentación de SUMO; lo único que tal vez tenga un poco de dificultad entender en este archivo, es que se utiliza el sistema RGB(Red-Green-Blue) para darle color a un cada tipo de vehículo, y que las rutas se forman simplemente enlistando las aristas (edges) definidas.

Observemos que este método no es práctico, porque si queremos una simulación de 10 000 vehículos, tendríamos que escribir 10 000 líneas, si queremos una red vial que conste de 11 intersecciones se complica definir los nodos y las aristas. Alguien podría preguntar, ¿entonces para qué leer esta sección?, la respuesta es que en esta sección la intención es sentar las bases de entendimiento de los mapas, simulaciones y funcionamiento básico de SUMO.

Ahora vamos con el único archivo de configuración, el cual aparece en la figura 4.7 al cual llamamos **ejemplo_sin_osm_josm.sumo.cfg**, en el cual se configura el tiempo que durará la simulación (en este caso 400 segundos), los archivos invocados que son necesarios para la simulación, que en este caso son los archivos **ejemplo_sin_osm_josm.net.xml** y **ejemplo_sin_osm_josm.rou.xml**, así como una característica adicional, la cual es **time-to-teleport value=1**, que simplemente es para evitar que se produzcan tele-transportaciones en los vehículos, cosa que a veces llega a pasar en SUMO por limitación de recursos, pero que no se aborda como un error, sino como un contratiempo que se soluciona durante la simulación.

Por último, si se quiere hacer la simulación descrita en esta sección, solo basta copiar todos los archivos descritos, ponerle los nombres señalados, ejecutar los comandos (todo sobre el mismo directorio), y por último, solo ejecutar el siguiente comando desde la terminal, donde la parte - **-error-log Error_del_ejemplo** produce un archivo de texto llamado **Error_del_ejemplo** con los errores de la simulación, si es que llega a ocurrir alguno.

```
$sumo-gui -c ejemplo_sin_josm.sumo.cfg - -error-log Error_del_ejemplo
```

La figura 4.8 muestra la simulación en SUMO:

Este comando inicializa la simulación de forma gráfica, pero si se quiere solamente hacer el proceso de la simulación de forma no gráfica, entonces ejecutamos el comando:

```
$sumo -c ejemplo_sin_josm.sumo.cfg - -error-log error
```

4.3. Creación óptima de un modelo de transporte urbano utilizando OSM,JOSM y SUMO

4.3.1. Creando el mapa de nuestro modelo usando OSM y JOSM

Ya vimos en la sección anterior lo poco práctico que es crear un mapa usando solo SUMO, vimos también las dificultades de crear la demanda de tránsito escribiéndola a mano, es por eso que en esta sección presentamos una forma de hacer el mismo trabajo (construcción del mapa y simulación) pero de forma óptima, por lo que empezamos con el mapa, y esta vez si usamos JOSM, y por tanto implícitamente también OSM.

Para crear el mapa utilizamos JOSM (y OSM en conjunto), primero abrimos JOSM y aparecerá la clásica página de bienvenida de JOSM, ahora le damos click en la opción del menú donde podemos descargar el mapa que queramos directamente del servidor de OSM, donde podemos enfocar cualquier parte del mundo y señalar con un rectángulo la zona de mapa que queremos descargar, la siguiente figura 4.9 muestra este proceso.

Solo queda descargar la zona seleccionada y guardarla con el siguiente formato:

```
<nombre_del_archivo>.osm.xml
```

Por ejemplo, al descargar la zona seleccionada, nosotros llamamos al archivo **mapa.osm.xml** (y no hay que olvidar guardarlo en el directorio donde estemos trabajando). Después de descargar y guardar ya tenemos un mapa previo, pues como vemos a veces al abrir mapa inicial (recién descargado) usando JOSM,

4.3. CREACIÓN ÓPTIMA DE UN MODELO DE TRANSPORTE URBANO UTILIZANDO OSM, JOSM Y SUMO

en este pueden aparecerá más calles, y con tal vez más cosas que nosotros hubiéramos querido descargar, esto no es problema, pues podemos editar la zona usando las herramientas de JOSM, hasta dejar el mapa como nosotros queramos y arreglar cualquier posible error que aparezca en el mapa, así como poner semáforos, por ejemplo, nuestro mapa es solo el Sector República, podemos ver que al tratar de enfocar lo más posible la zona, siempre aparecen más calles como se muestra en la figura 4.10.

Después de editar y corregir el mapa con JOSM, nuestro mapa queda como se muestra en la figura 4.11 imagen:

La edición y corrección del mapa no es difícil, para una edición y corrección óptima primero se recomienda leer el tutorial de JOSM y darle una revisión a las herramientas disponibles en JOSM, además de conocer el mapa para poder corregirlo. Al empezar la edición es conveniente hacerlo en orden, es decir, de calle por calle (que estén juntas o cerca) y determinar una forma estándar que sea fácil de recordar para nombrar las calles o detalles del mapa que queramos establecer.

Ya con el mapa editado, corregido y guardado por ejemplo con el nombre **mapa.osm.xml**, es **muy importante asegurarnos que ya no hay nada por modificar desde JOSM**, más adelante hablaremos el porque de este señalamiento. Se puede pensar que ya tenemos el mapa para usar en SUMO pero en realidad es que no es así, pues a pesar que el mapa esta en el lenguaje **XML**, el mapa tiene el formato que utiliza OSM (**mapa.osm.xml**), por lo que necesitamos convertirlo en un archivo que SUMO pueda entender.

Si recordamos, en la sección de herramientas de SUMO, esta el comando **netconvert** que sirve para adaptar archivos de formatos ajenos a un formato que SUMO pueda entender, por lo que tan solo con ejecutar el el siguiente comando desde la terminal:

```
$ netconvert - -osm-files mapa.osm.xml - -no-turnarounds true -o simulacion.centro.net.xml - -plain-output-prefix simulacion.centro
```

El comando anterior produce los siguientes archivos:

- **simulacion.centro.net.xml**
- **simulacion.centro.nod.xml**
- **simulacion.centro.edg.xml**
- **simulacion.centro.con.xml**

Hasta ahora sabemos que contienen los primeros tres archivos de la lista anterior. El archivo **simulacion.centro.con.xml** contiene la definición de las conexiones de cada intersección del mapa, es decir, estas conexiones definen en una

intersección la configuración de, dado un determinado carril, si un vehículo se mueve a través de este carril, a cuáles carriles puede cruzar solamente, a continuación la figura 4.12 muestra gráficamente con flechas (las que se encuentran sobre la intersección y no sobre los carriles), las conexiones definidas en la intersección del ejemplo de la sección anterior.

Puede llegar a pasar que tras ejecutar el último comando, se generen errores que mencionen por ejemplo, que no se sabe conoce el tipo de calle **highway.primary**. Este error es producido porque en OSM existen ciertos tipos de definiciones de una calle, que en SUMO no están definidas, por lo que hay que definirlos en SUMO creando el archivo **simulacion.centro.typ.xml**, en la figura 4.13 vemos un ejemplo de como crear este archivo, dependiendo de los tipos de calle que se necesiten definir (estos apareceran como error al ejecutar el último comando mencionado), y además agregar otras características como lo son la prioridad entre los distintos tipos de calles:

Al ejecutar el último comando mencionado, como ya vimos, se genera el mapa que puede ser usado directamente en SUMO. Uno puede preguntarse, ¿entonces para que necesitamos los archivos **simulacion.centro.nod.xml**, **simulacion.centro.edg.xml** y **simulacion.centro.con.xml**?, la respuesta es porque el mapa generado tiene muchas cosas determinadas por default en SUMO que tal vez debamos editar, como lo son las conexiones en cada intersección. A veces en la realidad no todas las posibles conexiones están permitidas en una intersección, por lo que si queremos eliminar alguna conexión de alguna intersección del mapa en específico, podemos hacerlo directamente del archivo **simulacion.centro.con.xml**.

Si queremos hacer alguna modificación de cualquier tipo al mapa, podemos hacerla directamente desde los archivos **simulacion.centro.nod.xml** y **simulacion.centro.edg.xml**, pues es muy **importante** volver a mencionar que no debemos modificar el mapa desde JOSM (más adelante se mencionará la razón).

Después de cualquier modificación a los archivos estáticos mencionados, solo tenemos que ejecutar el siguiente comando desde la terminal, y así obtener el archivo **simulacion.centro.net.xml** (nuestro mapa) modificado:

```
$ netconvert - -node-files=simulacion.centro.nod.xml
- -edge-files=simulacion.centro.edg.xml
- -connection-files=simulacion.centro.con.xml
- -type-files=simulacion.centro.typ.xml
- -no-turnarounds - -output-file=simulacion.centro.net.xml
```

No olvidemos que el archivo estático más importante es **simulacion.centro.net.xml** pues contiene todo el mapa y es el que usa SUMO, a continuación mencionamos las ventajas que obtuvimos al hacer nuestro mapa de la forma descrita en esta sección, pues por ejemplo:

4.3. CREACIÓN ÓPTIMA DE UN MODELO DE TRANSPORTE URBANO UTILIZANDO OSM, JOSM Y SUMO5

- No tuvimos que escribir ningún archivo como los escritos en la sección anterior, hubiera sido muy complejo escribir dichos archivos para un mapa como el Sector República.
- La edición del mapa en JOSM no es difícil.
- El mapa es correcto geográficamente hablando.
- Y tan solo ejecutando un comando obtuvimos todos los archivos estáticos de nuestro modelo
- En caso de tener que modificar algún error no detectado, podemos hacerlo directo de los archivos que tenemos y no desde JOSM, la razón es porque si editamos el mapa de nuevo en JOSM, es decir, si modificamos el archivo **mapa.osm.xml**, y de este archivo generamos el archivo **simulacion_centro.net.xml** (con uno de los comandos ya mencionados), puede pasar que algunos **id's** de los objetos del mapa (principalmente aristas) cambien, y esto puede llegar a arruinar todo el trabajo realizado previamente (edición de las características reales del mapa), y tener que empezar de nuevo.

Son claras las ventajas de hacer el mapa que conforma nuestro modelo en la forma descrita en esta sección, en lo siguiente se comprenderá lo que es SUMO, pues continuamos describiéndolo, en la siguiente sección se explicará cómo crear la parte dinámica de nuestra simulación en una forma **óptima**, es decir, anteriormente se mencionó que para tener el archivo **ejemplo_sin_osm_josm.rou.xml** escribimos unas 17 líneas de código, que tan solo generaban menos de diez vehículos, es decir, teníamos que escribir un archivo no muy pequeño para tan solo generar menos de una decena de vehículos en la simulación, ahora imaginemos el tamaño del archivo que tendríamos que escribir si hacemos una simulación con un mapa del tamaño del sector República, necesitaríamos definir por lo menos unos 100 000 vehículos para la una simulación de un par de horas, lo que llevaría mucho tiempo, además del hecho de que seríamos propensos a cometer muchos errores.

4.3.2. La demanda de transporte urbano

Ya vimos secciones anteriores que no es nada práctico escribir a mano la demanda de transporte, pues en un mapa como el sector república, necesitaríamos escribir cientos de rutas y miles de líneas (una para cada automóvil), fácilmente caeríamos en errores de código constantemente. En esta sección mostramos que no tiene que ser así, pues SUMO tiene varios recursos para generar la demanda de transporte de forma rápida y práctica, tan solo necesitamos decirle a SUMO algunas especificaciones de como queremos que sea esa demanda de transporte.

Durante la realización de este proyecto considerábamos solo algunas de las formas en que se puede generar transporte urbano, a continuación se muestran algunas maneras de generar transporte urbano.

- En SUMO, después de hacer un mapa, lo primero que se hace es ver automóviles recorriendo el mapa, y la forma más inmediata de generar transporte es usando los comandos en conjunto **duarouter** y **randomTrips.py**, siendo **duarouter** uno de los comandos mas importantes para generar transporte, **randomTrips.py** genera un archivo de viajes aleatorios y con extensión ***.trip.xml** que después es utilizado por **duarouter** generando así la demanda de transporte aleatoria, esta manera no es práctica para generar una demanda de transporte real, pues es muy irrealista, pueden aparecer vehículos dándole la vuelta a una cuadra o vehículos dando vueltas sin sentido en el mapa por ejemplo, que es algo no muy verosímil.
- También podemos generar demanda de transporte usando **Matrices OD**, OD significa origen y destino, esta manera es utilizada comúnmente por las autoridades, pero hay que tener en cuenta que las matrices deben tener un formato especial.
- El tercer recurso que mencionamos es de carácter estadístico, pues debemos crear todo un archivo con información estadística, que debemos guardar en un archivo con extensión ***.stat.xml**, entre la información estadística necesaria esta la edad en que se puede conducir, el número total de habitantes y de hogares, la edad en que las personas son consideradas niños, la probabilidad de que un adulto tenga un vehículo y la probabilidad de que un adulto tenga empleo, hay más información que es necesaria, pero no la mencionaremos toda debido a que esa información esta disponible en la página de SUMO.
- La cuarta y última forma que mencionaremos es la de usar probabilidades y aforos, que es la manera que consideramos más práctica, pues esta información puede ser obtenida en trabajo de campo. El método consiste en escribir un archivo que contenga las probabilidades de cada intersección, en el mapa, y que son propias de cada intersección, es decir, imaginemos una intersección, al llegar un vehículo desde alguna calle (arista) de la intersección tenemos que definir la probabilidad de que el vehículo gire a la derecha, la probabilidad de que gire a la izquierda y finalmente la probabilidad de que siga en línea recta, como se establece en la teoría de la probabilidad, la suma de estas 3 probabilidades debe ser igual a 1, de igual forma definimos las probabilidades en cada calle de la intersección.

Hay que tener claro que estas probabilidades son propias de las intersecciones y no de los vehículos, este archivo debe tener extensión ***.turn.xml**. También necesitamos definir un archivo con los aforos de la zona que estamos analizando, los cuales también se obtienen con estudios de campo,

4.3. CREACIÓN ÓPTIMA DE UN MODELO DE TRANSPORTE URBANO UTILIZANDO OSM, JOSM Y SUMO

estos aforos son fáciles de definir y se deben guardar en un archivo con extensión ***.flows.xml**, recordando que un aforo es al numero de vehículos que pasan en cierto tiempo por un punto en específico. Después de crear los archivos (que por practicidad los llamamos con el prefijo **simulacion_centro**) **simulacion_centro.turn.xml** y **simulacion_centro.flows.xml** (en el siguiente capítulo mostraremos ejemplos de estos archivos) tan solo ejecutamos el siguiente comando, para generar el archivo **simulacion_centro.rou.xml**, que contiene precisamente la demanda de transporte:

```
$jtrrouter - -flow-files=simulacion_centro.flows.xml - -turn-ratio-files=simulacion_centro.turn.xml - -net-file=simulacion_centro.net.xml - -output-file=simulacion_centro.rou.xml
```

La cuarta opción fue la que usamos en este proyecto, pues tuvimos ciertos aforos, que nos sirvieron para plantear algunos ejemplos de aplicaciones,

4.3.3. Archivos de configuración

Anteriormente en la clasificación que hicimos en los archivos que utiliza SUMO están los archivos de configuración. En este caso, mostramos el archivo de configuración mas importante, que debe tener por extensión ***.sumo.cfg**, este archivo configura como será la simulación, la figura 4.14 muestra el archivo **simulacion_centro.sumo.cfg**:

En el cual le decimos a SUMO que mapa y demanda de transporte usar, cuánto debe durar la simulación, que en este caso son 4000 segundos, usando el archivo **simulacion_centro.sumo.cfg** y el comando **sumo-gui** que se lleva a cabo la simulación, como se vió en secciones anteriores.

Para terminar esta sección, podemos preguntarnos ¿Para qué hacer una simulación si no estamos obteniendo datos o resultados de la simulación?, esta pregunta se contesta precisamente en la siguiente sección.

4.3.4. Datos de salida de una simulación en SUMO

Por último hablemos de como obtener datos de la simulación, pues después de cada experimento o simulación, el objetivo es obtener datos, por lo que mencionamos algunas clases de datos de salida que podemos obtener SUMO:

- Podemos obtener una aproximación del impacto ambiental, o en otras palabras, la contaminación causada por el transporte urbano.
- Se puede obtener un registro de la posición de todos los vehículos que sean de una clase en especial y que están en la simulación en cierto paso de la simulación.

- Podemos generar un registro de todos los vehículos que pasan por alguna calle en especial, o por algún punto en especial.
- Podemos generar la historia de un solo vehículo, es decir, registrar en un cierto paso de la simulación la velocidad que tiene el vehículo en ese instante, así como las coordenadas geográficas del vehículo, en que calle se encuentra y la posición en el mapa de SUMO. Para generar esta salida de datos, necesitamos un archivo de configuración, al cual llamaremos **automovil_4.xml** y mostramos a continuación:

Todos estos posibles resultados son interesantes, la información de cómo obtener estos resultados esta en la documentación de SUMO. Por dar un ejemplo, mostraremos como obtener la cuarta opción, es decir la historia de un vehículo de entre miles en la simulación, y al hablar de la historia de un vehículo, nos referimos a cierta información registrada del vehículo en cada paso de la simulación mientras aparece en ella, como lo son las coordenadas geográficas, el instante de tiempo de la simulación, en que calle se encuentra y que velocidad lleva en ese instante.

Para obtener esta historia solo debemos crear un archivo adicional que también es un archivo de configuración, pues configura en cierta forma la salida de datos de la simulación, nosotros lo llamamos **automovil.xml**, y el contenido de este archivo es el siguiente:

```
<additional>
  <vTypeProbe id="automovil_prueba"
    type="prueba" freq="60" file="automovil_prueba.xml" />
</additional>
```

Una breve descripción del contenido del archivo **automovil.xml** es que le estamos diciendo a SUMO que queremos la historia del automóvil de tipo prueba (< vType id="prueba".... >), que obtenga los datos de la historia cada 60 segundos (freq="60") y que toda la historia la guarde en un archivo llamado **automovil_prueba.xml**. Por último, solo basta agregar al archivo de configuración **simulacion_centro.sumo.cfg** que queremos este obtener este tipo de datos, solo basta agregar **value="automovil.xml"**, como se muestra en la figura 4.15:

Finalmente, para llevar a cabo la simulación, de nuevo solo ejecutamos el siguiente comando:

```
$ sumo-gui - -configuration-file simulacion_centro.sumo.cfg - -error-log Error_de_la_simulacion
```

4.3. CREACIÓN ÓPTIMA DE UN MODELO DE TRANSPORTE URBANO UTILIZANDO OSM,JOSM Y SUMO5

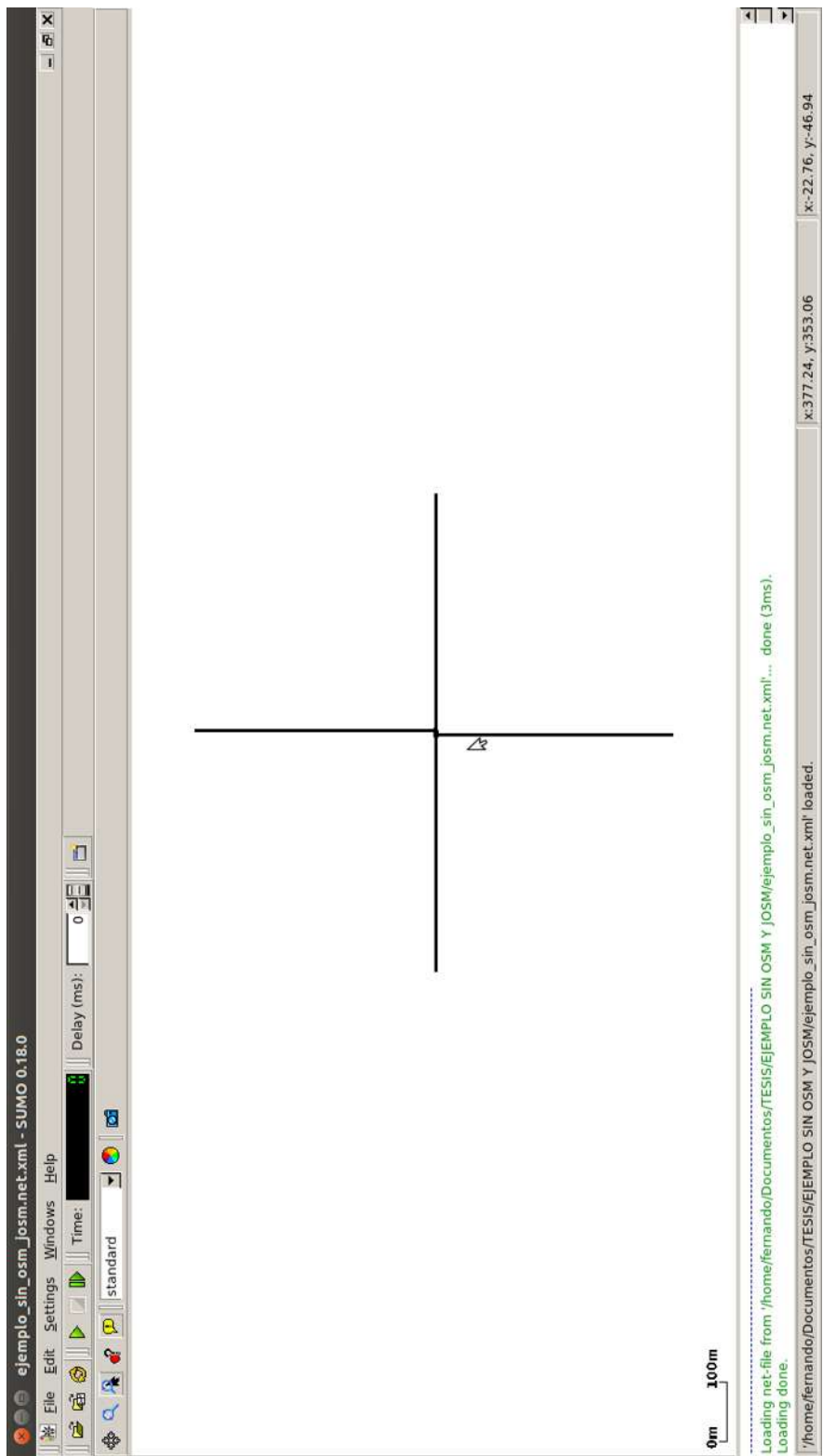


Figura 4.4: Vista Completa del mapa creado solo con SUMO

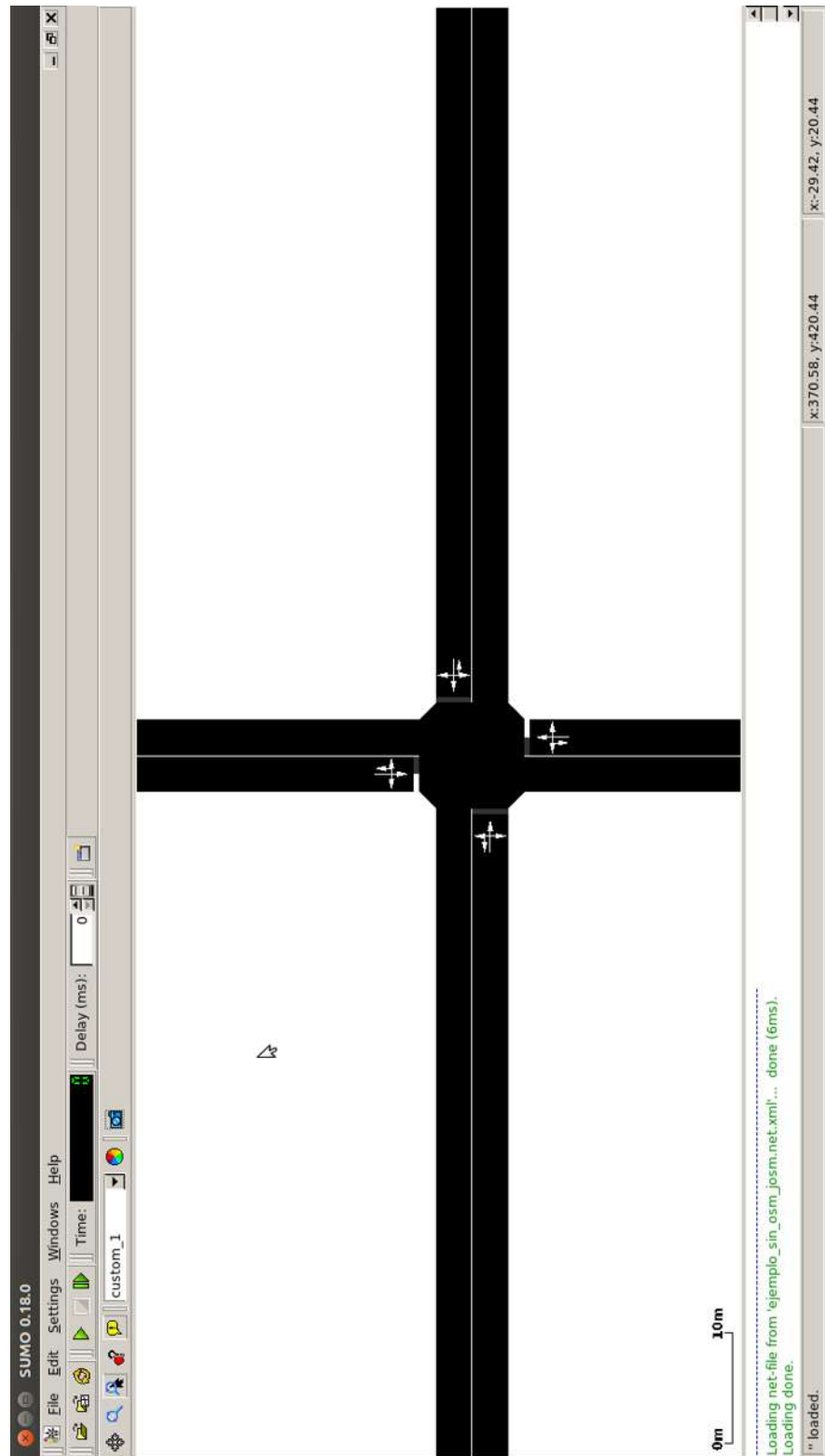


Figura 4.5: Vista de cercana del mapa creado solo con SUMO

```

</routes>

<!-- a continuacion definimos los tipos de vehiculos posibles y sus caracteristicas -->

<vType id="vehiculoA" accel="0.8" decel="4.5" sigma="0.4" length="4" maxSpeed="60" color="1,0,1"/>
<vType id="vehiculoB" accel="0.6" decel="4.5" sigma="0.6" length="10" maxSpeed="40" color="0,1,0"/>

<!-- a continuacion definimos las rutas posibles enlistando aristas -->

<route id="ruta-e-o" edges="a1-2 a2-3"/>
<route id="ruta-o-e" edges="a3-2 a2-1"/>
<route id="ruta-s-n" edges="a5-2 a2-4"/>
<route id="ruta-n-s" edges="a4-2 a2-5"/>

<!-- A continuacion definimos a cada vehiculo que aparecera en la simulacion, uno por uno -->

<vehicle id="A1" type="vehiculoA" route="ruta-e-o" depart="0" />
<vehicle id="A2" type="vehiculoA" route="ruta-o-e" depart="0" />
<vehicle id="A3" type="vehiculoA" route="ruta-s-n" depart="0" />
<vehicle id="A4" type="vehiculoA" route="ruta-n-s" depart="0" />

<vehicle id="B1" type="vehiculoB" route="ruta-e-o" depart="4" />
<vehicle id="B2" type="vehiculoB" route="ruta-o-e" depart="4" />
<vehicle id="B3" type="vehiculoB" route="ruta-s-n" depart="4" />
<vehicle id="B4" type="vehiculoB" route="ruta-n-s" depart="4" />
</routes>

```

Figura 4.6: Archivo ejemplo_sin_josm.rou.xml

```
<configuration >
  <input>
    <n value="ejemplo_sin_osm_josm.net.xml"/>
    <r value="ejemplo_sin_osm_josm.rou.xml"/>
  </input>
  <time>
    <begin value="0" />
    <end value="400" />
  </time>
  <time-to-teleport value="-1"/>
</configuration>
```

Figura 4.7: Archivo ejemplo_sin_josm.sumo.xml

4.3. CREACIÓN ÓPTIMA DE UN MODELO DE TRANSPORTE URBANO UTILIZANDO OSM,JOSM Y SUMO

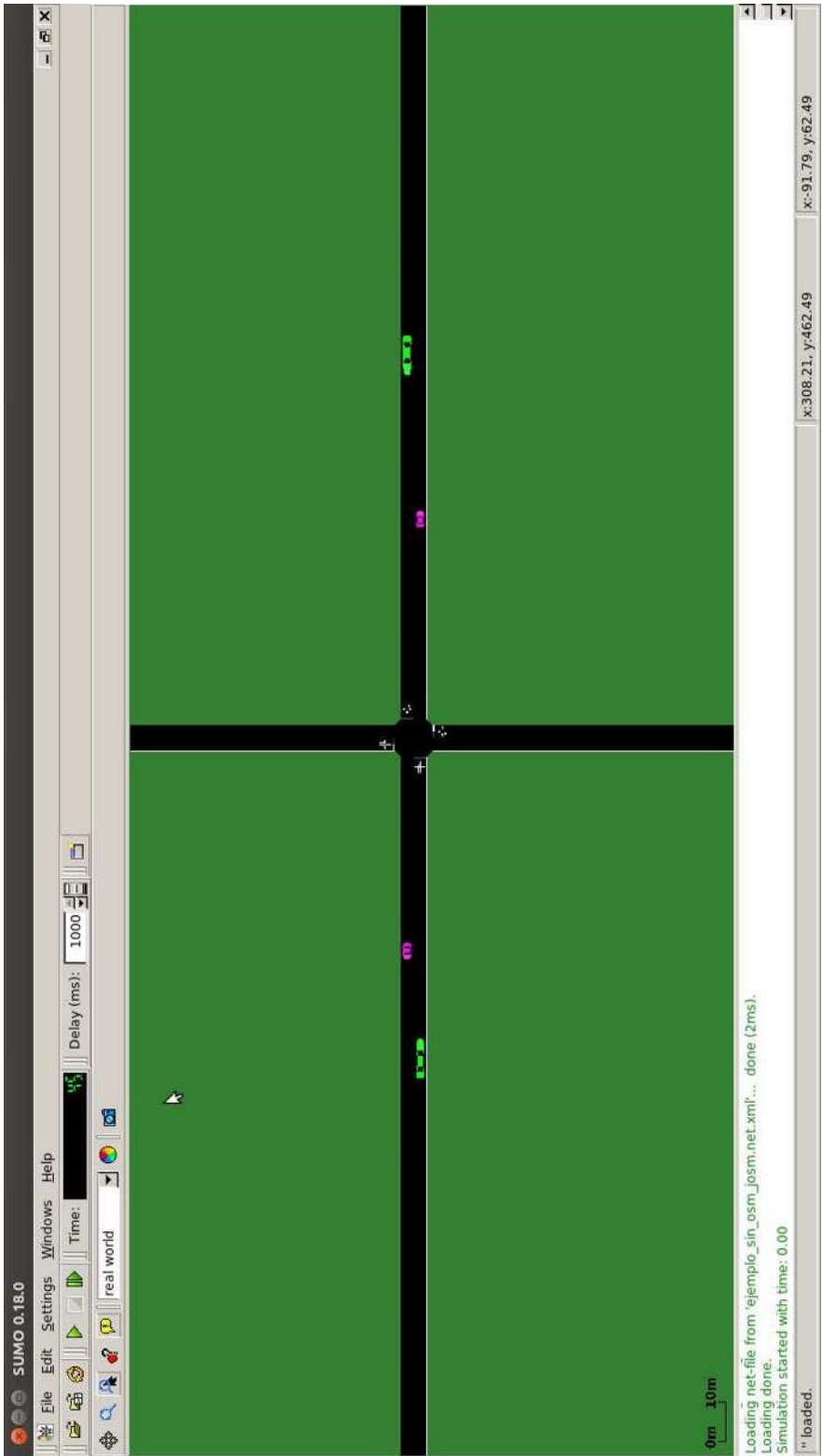


Figura 4.8: Simulación en SUMO del ejemplo descrito en esta sección

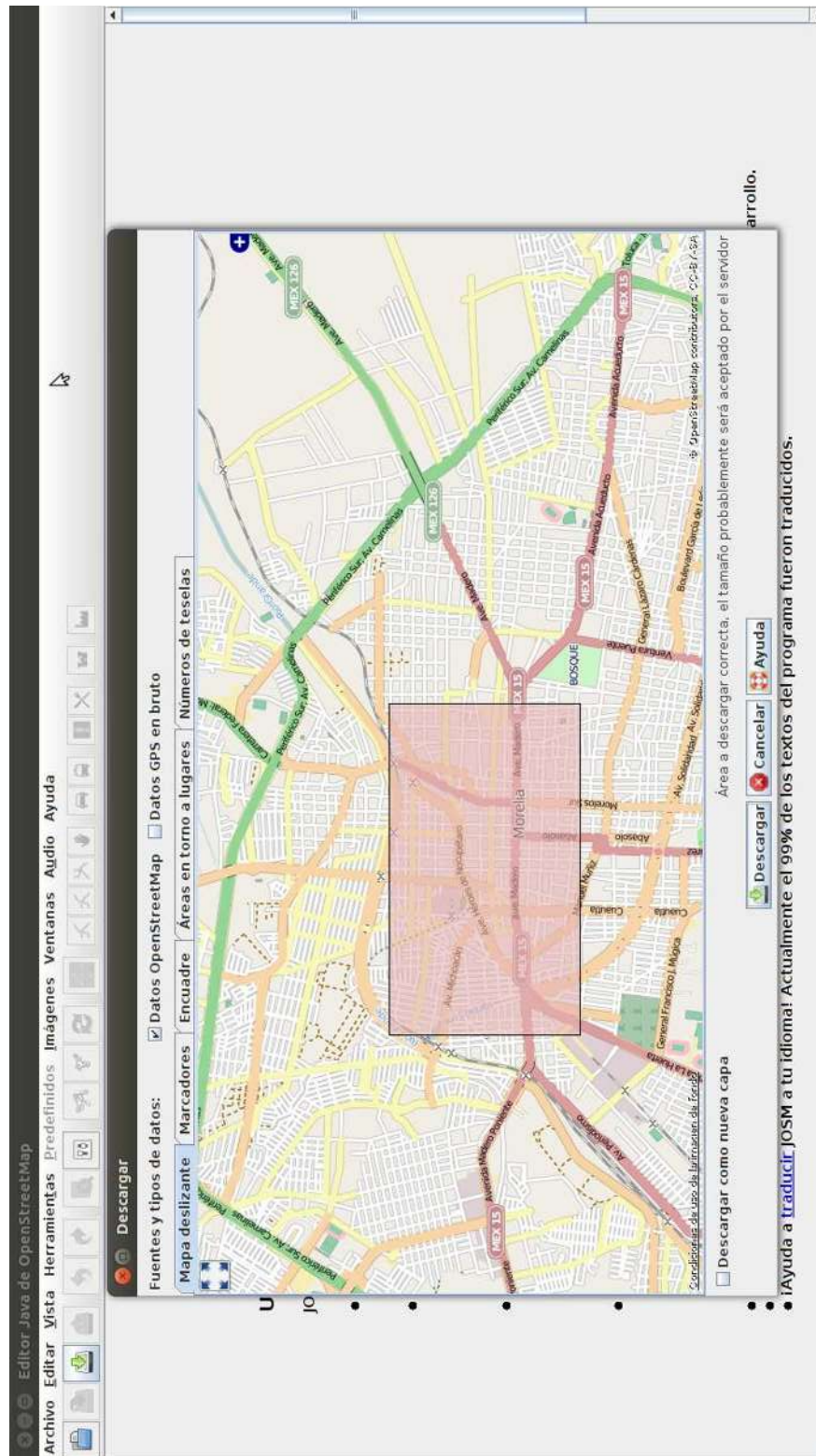


Figura 4.9: Interfaz de JOSM seleccionando el mapa a descargar

4.3. CREACIÓN ÓPTIMA DE UN MODELO DE TRANSPORTE URBANO UTILIZANDO OSM,JOSM Y SUMO6

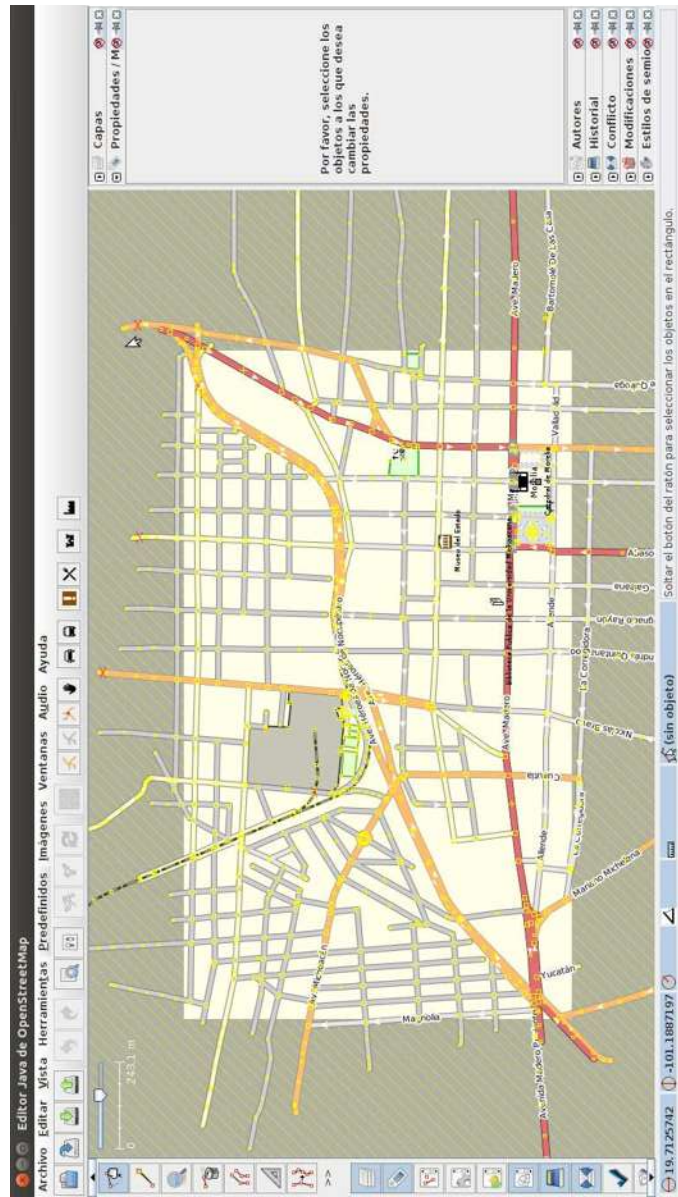


Figura 4.10: Mapa inicial del Sector República descargado y guardarlo usando JOSM

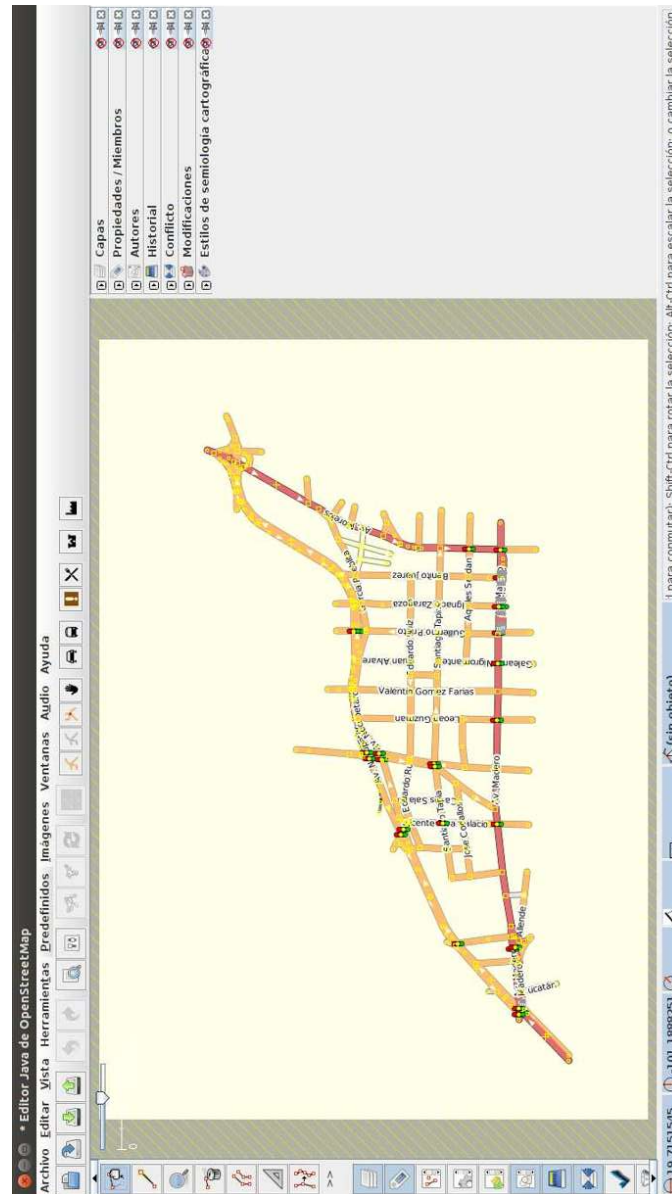


Figura 4.11: Mapa del Sector República editado y corregido en JOSM

4.3. CREACIÓN ÓPTIMA DE UN MODELO DE TRANSPORTE URBANO UTILIZANDO OSM,JOSM Y SUMO

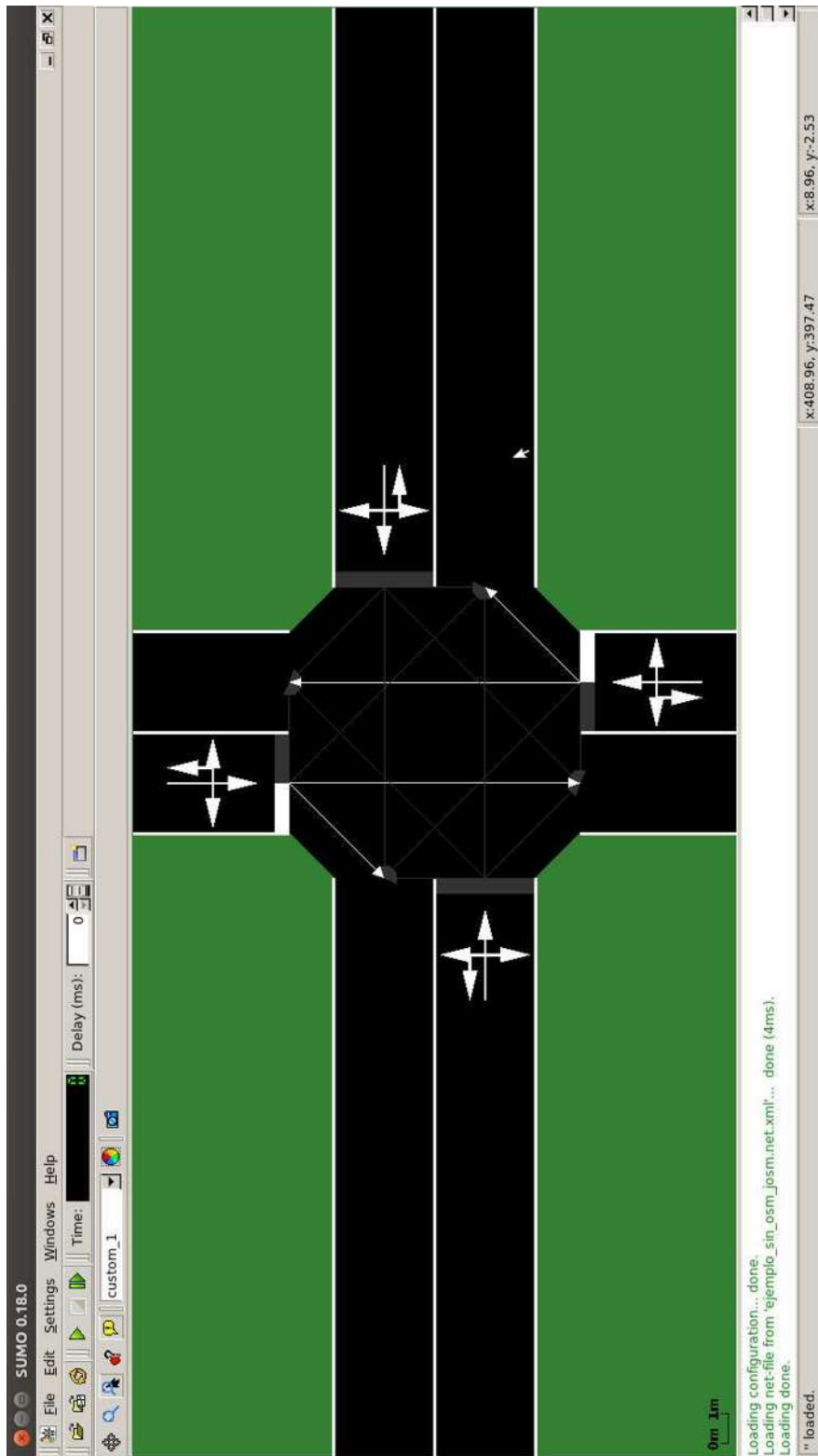


Figura 4.12: Conexiones de la intersección, del ejemplo visto en la sección anterior.

```

<types>
<type id="highway.primary" priority="11" speed="40" />
<type id="highway.primary_link" priority="11" speed="40" />
<type id="highway.secondary" priority="11" speed="30" />
<type id="highway.tertiary" priority="11" speed="30" />
<type id="highway.residential" priority="11" speed="30" />
<type id="highway.unclassified" priority="11" speed="30" />
<type id="highway.trunk" priority="11" speed="30" />
</types>

```

Figura 4.13: Archivo `simulacion_centro.typ.xml`

```

<configuration >
  <input>
    <n value="simulacion_centro.net.xml"/>
    <r value="simulacion_centro.rou.xml" />
  </input>
  <time>
    <begin value="0" />
    <end value="4000" />
  </time>
  <time-to-teleport value="-1"/>
</configuration>

```

Figura 4.14: Archivo de configuración `simulacion_centro.sumo.cfg`

```

<configuration >
  <input>
    <n value="simulacion_centro.net.xml"/>
    <r value="simulacion_centro.rou.xml" />
    <a value="automovil.xml" />
  </input>
  <time>
    <begin value="0" />
    <end value="4000" />
  </time>
  <time-to-teleport value="-1"/>
</configuration>

```

Figura 4.15: Archivo `simulacion_centro.sumo.xml`, donde agregamos el archivo de configuración `automovil.xml` para obtener resultados de la simulación

Capítulo 5

Aplicaciones

5.1. Introducción

Es este capítulo vamos a mostrar tres ejemplos de aplicación de nuestra metodología, usaremos la forma óptima de crear nuestro mapa y simulación, visto en el capítulo anterior, además de generar la demanda de transporte urbano utilizando aforos y probabilidades (opción cuatro en la sección de **Demanda de transporte** del capítulo anterior) en las tres aplicaciones. Los tres ejemplos de aplicación serán en áreas específicas del sector República, que cada vez son más amplias y complejas.

5.1.1. Primera aplicación

En esta primera aplicación, estamos interesados en medir el tiempo de recorrido de un automóvil que recorre la sección de la Av. Madero comprendida en el sector República.

La dirección del recorrido es de poniente a oriente y planteamos 4 escenarios distintos, donde en el primer escenario registramos el tiempo de recorrido de nuestro vehículo sin ningún otro en la simulación, es decir, nuestro vehículo circulará libremente, con una única restricción, el límite de velocidad que establecimos es de $40km/h$. En los tres escenarios restantes si habrá tráfico, e incrementará, de manera que en el cuarto escenario intentamos llevar al límite el transporte urbano.

A continuación la figura 5.1 muestra la imagen del problema, donde se señala el recorrido y orientación, además las tablas 5.1 y 5.2 describen los aforos (flows) de cada escenario:

Estos aforos los definimos en 4 archivos, respectivamente dependiendo del escenario:

- `primera_plicacion_1.flows.xml`

Escenario	Flujo de Vehículos en el intervalo de la simulación (aprox. 1h 11min)
Escenario 1	Sin tráfico y recordando que el límite de velocidad es de 40 km/h
Escenario 2	Av. Madero Poniente 5000 Av. Madero Poniente Acceso 3000 Av. Madero Oriente 5000 Av. Morelos 5000 Ignacio Zaragoza 1500 Abasolo 2000 Nigromante 2000 Rayón 2000 León Guzmán 2000 Nicolás Bravo 2000 Sebastián Lerdo Tejada 300 Rivapalacio 2000 Cuautla 2000 Francisco Zarco 2000 Desviación Allende 500 Yucatán 300

Tabla 5.1: Aforos usados en los escenarios 1 y 2

- **primera_plicacion_2.flows.xml**
- **primera_plicacion_3.flows.xml**
- **primera_plicacion_4.flows.xml**

La figura 5.2 muestra el archivo **primera_plicacion_4.flows.xml**, para tener en claro como definir los flujos. En cuanto a la descripción del contenido de este archivo, podemos decir que, debemos indicarle de que calle (edge, o en español arista) saldrán los vehículos, esto se hace en la parte **from="edge"**, la parte **begin="0" end="4000"** sólo es el intervalo de tiempo en el que se distribuye el numero se vehículos que salen de la respectiva calle, claro esta que es conveniente establecer el tiempo de la simulación como intervalo; por último para indicar el numero de vehículos del aforo, lo indicamos en la parte **number="<aforo>"**.

Como sabemos además de definir el archivo con los aforos, también debemos definir las probabilidades en cada intersección de la zona de estudio, en el archivo **simulacion.centro.turn.xml**; la figura 5.3 muestra la imagen de nuestra primera intersección que tenemos en el recorrido, que es la intersección entre la Av. Nocupétaro y la Av. Madero, es importante recordar que definimos estas probabilidades en cada intersección de la Av. Madero comprendida en el sector República, es por este motivo que no mostramos el archivo **simulacion.centro.turn.xml** completo, pues es algo extenso, pero ejemplificamos

Escenario	Flujo de Vehículos en el intervalo de la simulación (aprox. 1h 11min)
Escenario 3	Av. Madero Poniente 10000 Av. Madero Poniente Acceso 6000 Av. Madero Oriente 10000 Av. Morelos 10000 Ignacio Zaragoza 3000 Abasolo 2000 Nigromante 4000 Rayón 4000 León Guzmán 4000 Nicolás Bravo 4000 Sebastián Lerdo Tejada 600 Rivapalacio 4000 Cuautla 4000 Francisco Zarco 4000 Desviación Allende 1000 Yucatán 600
Escenario 4	Av. Madero Poniente 20000 Av. Madero Poniente Acceso 12000 Av. Madero Oriente 20000 Av. Morelos 20000 Ignacio Zaragoza 6000 Abasolo 2000 Nigromante 8000 Rayón 8000 León Guzmán 8000 Nicolás Bravo 8000 Sebastián Lerdo Tejada 1200 Rivapalacio 8000 Cuautla 8000 Francisco Zarco 8000 Desviación Allende 2000 Yucatán 1200

Tabla 5.2: Aforos usados en los escenarios 3 y 4

Escenario	Tiempo de recorrido
Escenario 1	5.06 minutos
Escenario 2	15.03 minutos
Escenario 3	16.95 minutos
Escenario 4	17 minutos

Tabla 5.3: Tiempos de recorrido

como definir las probabilidades en la intersección en la figura 5.4 .

Vemos que en el archivo **simulacion_centro.turn.xml** también definimos las llamadas sumideros (**sink's**), que simplemente consisten en las aristas en donde desaparecerán los vehículos de la simulación.

Para llevar a cabo la simulación de cada escenario, necesitamos la demanda de transporte de cada escenario, ya que los escenarios son distintos, por lo que ejecutamos el siguiente comando para obtener; por ejemplo, la demanda de transporte del escenario 4:

```
jtrrouter -flow-files=primera_aplicacion_4.flows.xml
-turn-ratio-files=simulacion_centro.turn.xml
-net-file=simulacion_centro.net.xml -error-log error_aplicaion_4
-output-file=simulacion_centro.rou.xml -b 0 -e 4000
```

Como el mapa es el mismo en todos los escenarios, el archivo **primera_plicacion.net.xml** no cambia (por eso lo clasificábamos como un archivo estático), de manera que para llevar a cabo, por ejemplo, el escenario tres, solo cambiamos en el comando el archivo **primera_plicacion_4.flows.xml** por el archivo **primera_plicacion_3.flows.xml**, pues también las probabilidades son las mismas en los 4 escenarios.

Ya con el mapa y demanda de transporte generados, llevamos a cabo las simulaciones, en donde obtuvimos respectivos tiempos de recorrido que se muestran en la tabla 5.3 y en la figura 5.5.

5.1.2. Segunda aplicación

En la primera aplicación nuestra simulación giró sólo en torno a la Av. Madero, con sus respectivas entradas y salidas de vehículos, en esta segunda aplicación no solo hacemos una simulación para hacer un estudio, sino que hacemos una propuesta de la configuración vial del sector República, en la cual proponemos principalmente dividir la Av. Madero en dos partes, la parte izquierda que comienza desde la intersección de la Av. Madero y la Av. Nocupétaro, hasta la intersección de la Av. Madero y la Calle Francisco Zarco, permanece igual que siempre, es decir, es de dos sentidos. Por el contrario, el lado derecho de esta división de la

Av. Madero constará de un solo sentido, el cual es en dirección poniente-oriente, por lo que este tramo de la Av. Madero será de 4 carriles, nuestra propuesta es que esto sea hasta la intersección de la Av. Madero y la Av. Acueducto, y no solo hasta la Av. Morelos.

Como nuestra zona de estudio es el sector República, la simulación será igual, en esta aplicación no solo se abarca la Av. madero, también incluimos las calles Santiago Tapia y Allende (paralelas a la Av. Madero), donde estas tendrán dirección oriente-poniente.

Nuestro objetivo en este ejemplo de aplicación es el mismo, es decir, obtuvimos los tiempos de recorrido en la Av. Madero, solo que esta vez registramos la historia de 4 vehículos (en lugar de solo uno como en la aplicación anterior), debido a que en la parte de la Av. Madero comprendida entre las calles Abasolo y Morelos hay una división en la Av. Madero, pues en esta parte la Av. Madero tiene una especie de camellón, por lo que obtenemos los tiempos de recorrido en los escenarios 1 y 4 solamente, pero de dos vehículos, donde uno de ellos recorre la parte superior de la Av. Madero donde esta el camellón y el otro vehículo hace el recorrido en la parte inferior de esta sección, esto lo hacemos con el fin de ver si hay alguna diferencia de tiempo de recorrido, y es de esperarse que sí, pues en la parte superior de la Av. Madero de la sección mencionada, se encuentran un par de entradas y salidas, a diferencia de la parte inferior donde no hay ninguna. Además registramos el tiempo de recorrido de los dos vehículos restantes, que recorren las calles Santiago Tapia y Allende.

A continuación se muestran la figura 5.6 que describen lo que mencionamos, pero daremos una descripción adicional para que el problema y los objetivos queden lo mas claros posible.

En la figura 5.6 vemos que la zona de nuestro interés en este ejemplo de aplicación es la zona delimitada por la línea punteada, en todas las intersecciones de esta zona son donde debemos definir las probabilidades, pues seguimos la misma metodología. Sobre la Av. Madero vemos que aparece un punto negro, esto nos indica lo siguiente, desde el punto **A** hasta el punto negro, la Av. Madero tiene dos sentidos, pero del punto negro al punto B, la Av. Madero consiste de un solo sentido, que es poniente-oriente, por lo que tenemos una vialidad de 4 carriles.

Las flechas negras indican el recorrido de nuestros dos vehículos, donde vemos que hay un área dentro del cuadro negro, es esta área donde se hablaba que la Av. Madero tiene un espacio en medio (camellón), y que se separa en dos calles de dos carriles cada una, un vehículo recorrerá esa sección de la Av. Madero por la parte superior y el otro por la parte inferior, para ver si hay diferencia en el tiempo de recorrido, lo que el sentido común nos dice que tiene que haber, pues si vemos en la imagen, en la parte superior de la Av. Madero en esa sección hay dos entradas y una salida, que a diferencia de la parte inferior no lo hay, pues quitamos los semáforos de esta zona y en la parte inferior no hay nada que perturbe la circulación de vehículos; al vehículo que circula en la parte superior de esta sección lo llamamos **Vehículo A**, al vehículo que circula en la parte

Escenario	Tiempo de recorrido del carro
Escenario 1 en la Av. Madero en el vehículo A	5.04 minutos
Escenario 1 en la Av. Madero en el vehículo B	5.04 minutos
Escenario 1 en la calle Santiago Tapia	6.13 minutos
Escenario 1 en la calle Allende	4.83 minutos
Escenario 4 en la Av. Madero del vehículo A	12.78 minutos
Escenario 4 en la Av. Madero del vehículo B	7.33 minutos
Escenario 4 en la calle Santiago Tapia	1 hora 11 minutos
Escenario 4 en la calle Allende	8.93 minutos

Tabla 5.4: Tiempos de recorrido

inferior lo llamamos **Vehículo B**.

Siguiendo la misma metodología mostrados en la Primera Aplicación (de la sección anterior), obtuvimos las respectivas demandas de transporte, usando los escenarios 1 y 4 y. Tras ejecutar las simulaciones, obtuvimos los siguientes resultados, que se muestran en la tabla 5.4.

Y como era de esperar obtuvimos diferencia en los tiempos de recorrido. Y como era de esperar, obtuvimos una diferencia en los tiempos de recorrido entre el vehículo A y B, que recordando, circulan por la Av. Madero, debemos mencionar que el vehículo que circuló en la calle Santiago Tapia que quedó atrapado en una congestión vial al inicio de su recorrido, y no logró avanzar en toda la simulación.

5.1.3. Tercera aplicación

Por último mostramos la tercera aplicación, que fue la más ambiciosa de todas, pues proponemos un escenario que durante un período de tiempo fue polémico y algunos consideraban como descabellado, como lo es una zona exclusivamente peatonal en la zona más transitada del centro histórico, mas específicamente planteamos como zona exclusivamente peatonal el área comprendida entre las calles Abasolo, Allende, Morelos y Aquiles Serdán, Esto se muestra en la figura 5.7.

Esta zona es la mas afectada en cuanto inconvenientes impredecibles como las manifestaciones y plantones, así como incidentes predecibles como eventos deportivos, celebraciones de días festivos y eventos culturales.

Por tanto, nuestra propuesta consiste en lo siguiente, como ya dijimos, proponemos que el área del sector república comprendida entre las calles ya mencionadas sea de uso exclusivamente peatonal. Además incluimos una zona más amplia que en las aplicaciones anteriores, pues incluimos las siguientes calles paralelas a la Av. Madero y en el lado norte, las cuales son Aquiles Serdán, Santiago Tapia y Eduardo Ruiz, aunque la calle Aquiles Serdán es relativamente ignorada por su tamaño, las cuales tendrán dirección oriente-poniente, en cuanto a las dos calles que quedan al sur, están las calles Allende y Corregidora que las establecemos con dirección poniente-oriente.

También debemos mencionar que hicimos más modificaciones a la configuración vial del sector República, como por ejemplo en la sección de la calle Abasolo en el sector república establecemos la dirección norte-sur, además conservamos la propuesta de hacer la misma sección (como en la sección anterior) de la Av. Madero de un solo sentido, también propusimos una glorieta en la intersección de la Av. Madero y la Av. Michelena, toda esta descripción la mostramos en la figura 5.9, por último mencionamos que nuestro objetivo sigue siendo el mismo, es decir, obtener los tiempos de recorrido de ocho vehículos los cuales recorren las calles y avenidas Madero, Allende, Corregidora, y Eduardo Ruiz, donde dos vehículos recorren respectivamente las calles y avenidas mencionadas, pero con una pequeña diferencia de recorrido en cada par en una sección de su recorrido, por poner un ejemplo para explicar más a detalle a que nos referimos vamos a decir uno en específico. En la Av. Madero como se ha visto en las aplicaciones anteriores, comienzan su recorrido en la parte poniente de nuestra zona de estudio (sector República), es esta tercera aplicación es igual, solamente que ahora una será un par de vehículos que comenzarán el recorrido (en vez de uno como en la primera aplicación); estos vehículos recorren la Av. Madero hasta llegar a la intersección con la calle Abasolo, después recorren Abasolo hasta llegar a la intersección con la calle Allende, luego recorren la calle Allende, pero un vehículo llega hasta la intersección con la calle Valladolid, la cual recorre, incorporándose a la Av. Madero, en cambio, el otro vehículo llega hasta la calle Vasco de Quiroga, la cual recorre para también incorporarse de nuevo a la Av. Madero. Vemos que solo hay una pequeña diferencia de recorrido de una cuadra, pero estamos interesados en ver si existe alguna diferencia de tiempo de recorrido, lo mismo pasa con los tres pares de vehículos en las calles y avenidas Allende, Corregidora y Eduardo Ruiz (un par para cada calle o avenida respectivamente), En cuanto a la calle Santiago Tapia, solo utilizamos un vehículo para medir su tiempo de recorrido, las rutas de esta tercera aplicación se muestran en la figura 5.8. Usaremos los escenarios (aforos) 1 y 4 planteados en la Primera Aplicación.

Por último después de seguir la metodología descrita en este trabajo y hacer las simulaciones, obtuvimos los siguientes que se muestran en las tablas 5.5 y 5.6.

Ruta	Tiempo de recorrido
Av. Madero, Abasolo, Allende, Valladolid y Av. Madero	5.81 minutos
Av. Madero, Abasolo, Allende, Vasco de Quiroga y Av. Madero	5.53 minutos
Av. Madero, Allende, Valladolid y Av. Madero	4.81 minutos
Av. Madero, Allende, Vasco de Quiroga y Av. Madero	4.93 minutos
Av. Madero, Michelena, Corregidora, Valladolid y Av. Madero	4.5 minutos
Av. Madero, Michelena, Corregidora, Vasco de Quiroga y Av. Madero	5.81 minutos
Santiago Tapia, Francisco Zarco y Av. Madero	5.55 minutos
Eduardo Ruiz, Riva Palacio, Santiago Tapia, Francisco Zarco y Av. Madero	6.94 minutos
Eduardo Ruiz, Guadalupe Victoria, Santiago Tapia, Francisco Zarco y Av. Madero	6.91 minutos

Tabla 5.5: Tiempos de recorrido en el escenario 1

Ruta	Tiempo de recorrido
Av. Madero, Abasolo, Allende, Valladolid y Av. Madero	15.28 minutos
Av. Madero, Abasolo, Allende, Vasco de Quiroga y Av. Madero	12.03 minutos
Av. Madero, Allende, Valladolid y Av. Madero	7.61 minutos
Av. Madero, Allende, Vasco de Quiroga y Av. Madero	8.45 minutos
Av. Madero, Michelena, Corregidora, Valladolid y Av. Madero	6.6 minutos
Av. Madero, Michelena, Corregidora, Vasco de Quiroga y Av. Madero	7.4 minutos
Santiago Tapia, Francisco Zarco y Av. Madero	5.25 minutos
Eduardo Ruiz, Riva Palacio, Santiago Tapia, Francisco Zarco y Av. Madero	9.04 minutos
Eduardo Ruiz, Guadalupe Victoria, Santiago Tapia, Francisco Zarco y Av. Madero	8.85 minutos

Tabla 5.6: Tiempos de recorrido en el escenario 4

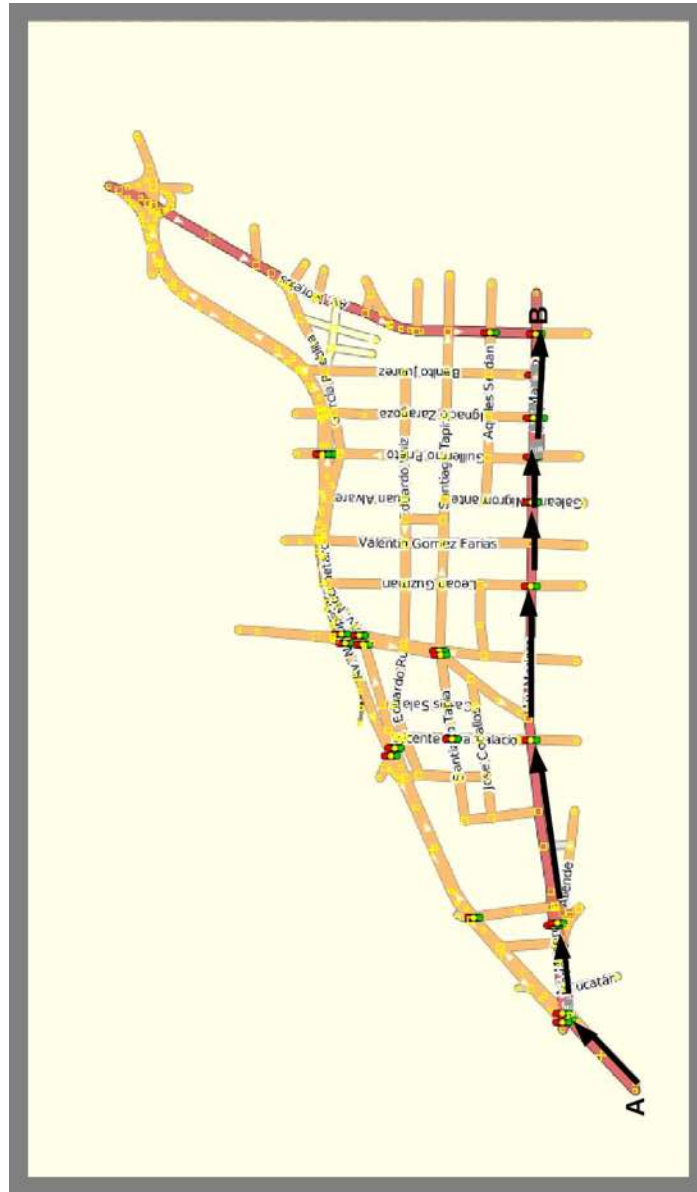


Figura 5.1: Planteamiento del recorrido en la primera aplicación

```

<flowdefs>
  <flow id="madero_poniente" from="182915012" begin="0" end="4000" number="20000"/>
  <flow id="madero_poniente_acceso" from="180968837" begin="0" end="4000" number="12000"/>
  <flow id="madero_orientado" from="-3722" begin="0" end="4000" number="20000"/>
  <flow id="morelos" from="105482071#14" begin="0" end="4000" number="20000"/>
  <flow id="ignacio_zaragoza" from="183747290#4" begin="0" end="4000" number="6000"/>
  <flow id="abasolo" from="190111677" begin="0" end="4000" number="8000"/>
  <flow id="nigromante" from="183747285#1" begin="0" end="4000" number="8000"/>
  <flow id="rayon" from="169184440" begin="0" end="4000" number="8000"/>
  <flow id="leon_guzman" from="183747287#3" begin="0" end="4000" number="8000"/>
  <flow id="nicolas_bravo" from="169184314" begin="0" end="4000" number="8000"/>
  <flow id="sebastian_lerdo_tejada" from="183747286#1" begin="0" end="4000" number="1200"/>
  <flow id="rivapalacio" from="-183747281#0" begin="0" end="4000" number="8000"/>
  <flow id="cuautla" from="-169185335" begin="0" end="4000" number="8000"/>
  <flow id="francisco_zarco" from="-180682627#0" begin="0" end="4000" number="8000"/>
  <flow id="allende_desviacion" from="164570787" begin="0" end="4000" number="2000"/>
  <flow id="yucatan" from="-169860459" begin="0" end="10000" number="1200"/>
</flowdefs>

```

Figura 5.2: Archivo `primera_plicacion_4.flows.xml`

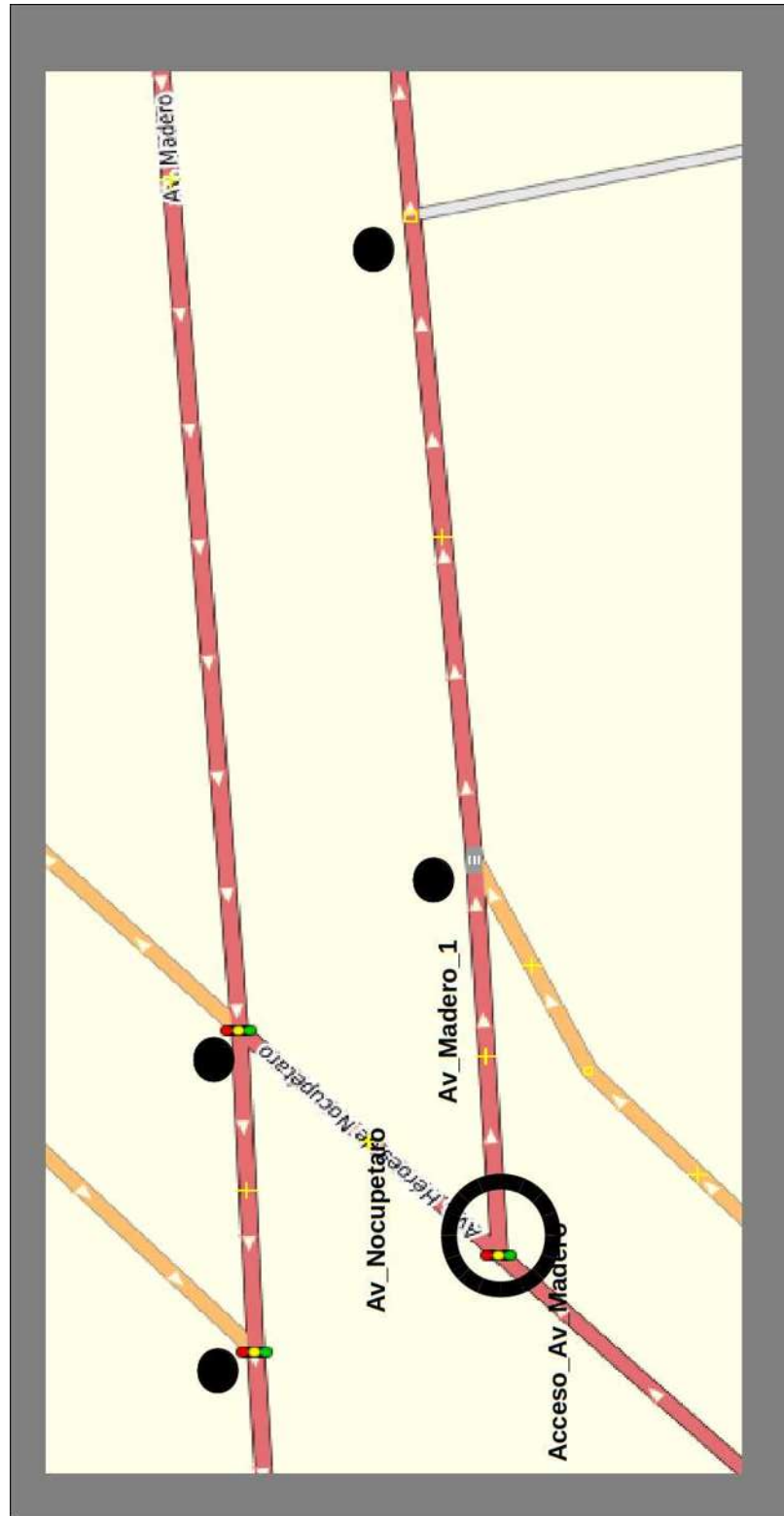


Figura 5.3: Intersecciones del problema, la intersección encerrada por el círculo es en la cual mostramos como definir las probabilidades

```

<turns>
<interval begin="0" end="4000">
  <fromEdge id="Acceso_Av_Madero">
    <toEdge id="Av_Madero_1" probability="0.25" />
    <toEdge id="Av_Nocupetaro" probability="0.75" />
  </fromEdge>
  .
  .
  .
  <!-- Segimos definiendo probabilidades en cada interseccion -->
  .
  .
  .
</interval>

<sink edges="edge_de_salida_1" />
<sink edges="edge_de_salida_2" />
<sink edges="edge_de_salida_3" />
<sink edges="edge_de_salida_4" />
.
.
.
<!-- Definimos mas "sink" si es necesario -->
</turns>

```

Figura 5.4: Archivo **simulacion_centro.turn.xml** que contiene la definición de probabilidades en las intersecciones

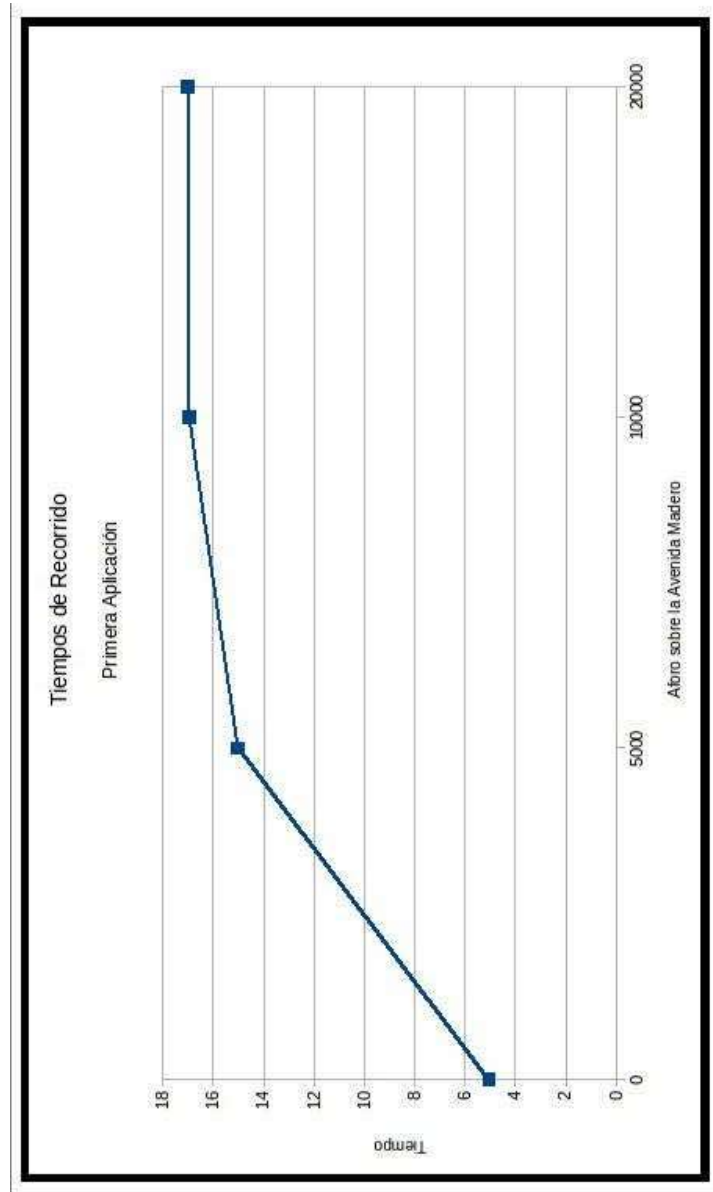


Figura 5.5: Grafica de los tiempos de recorrido

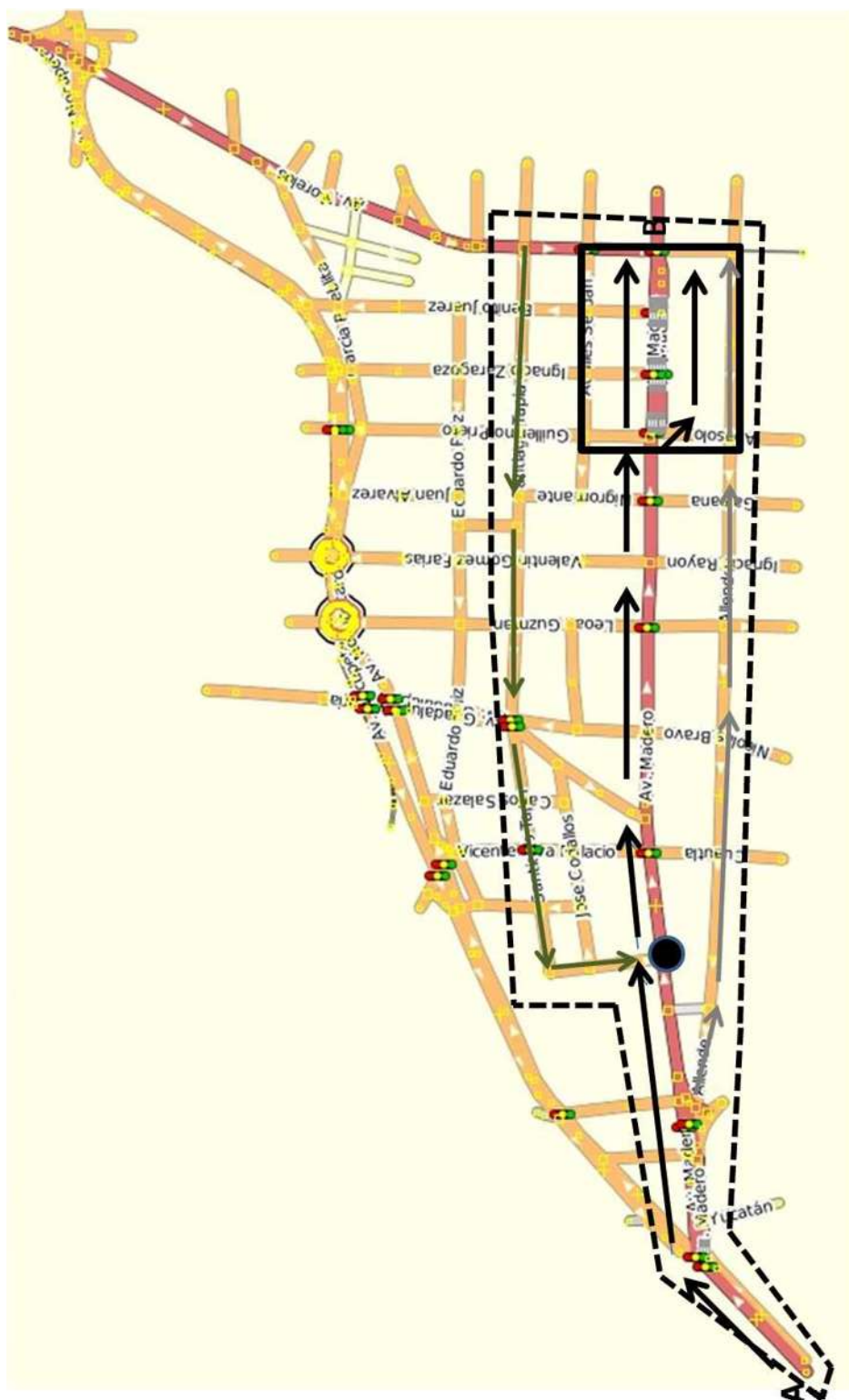


Figura 5.6: Descripción del problema de la segunda aplicación

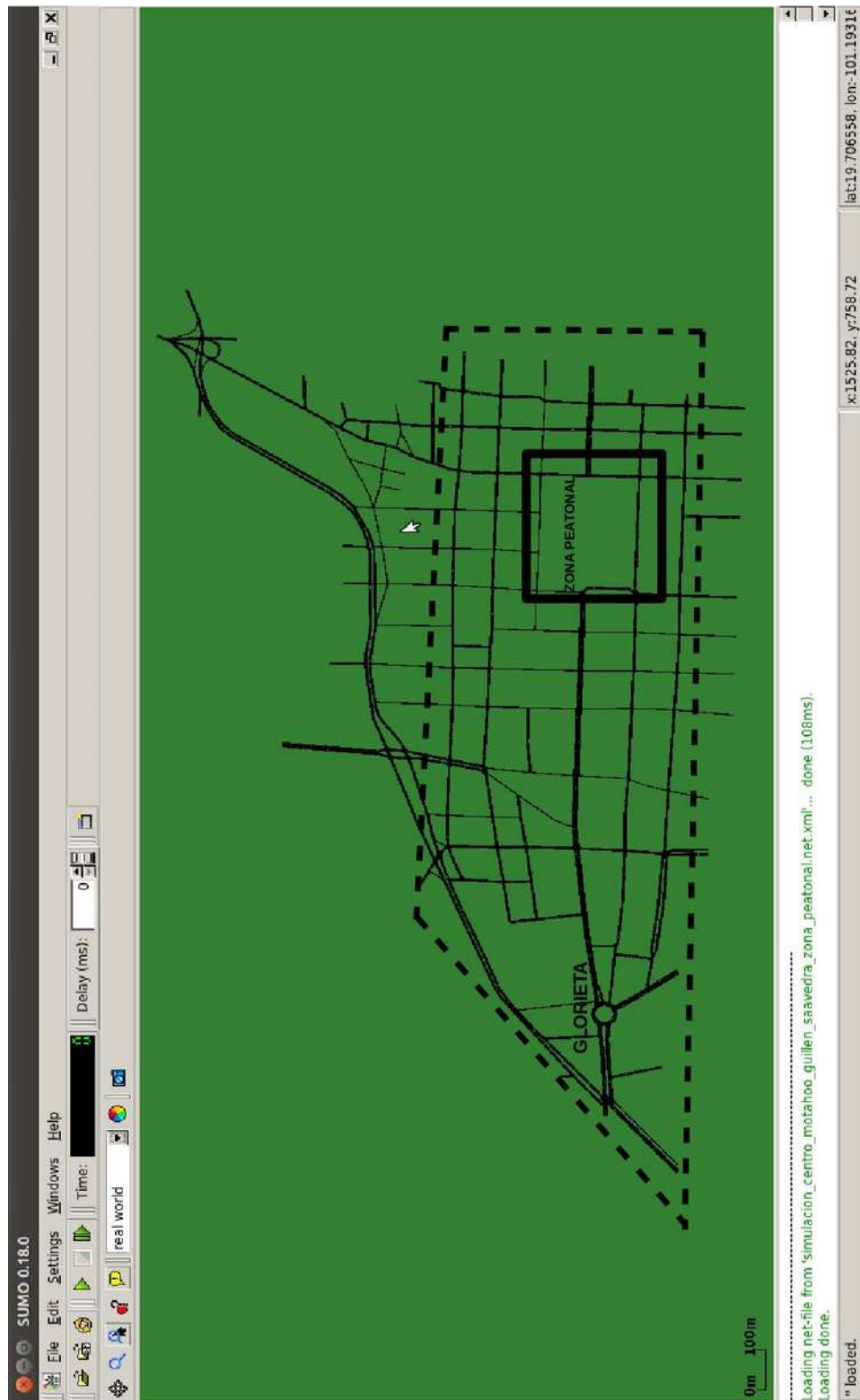


Figura 5.7: Descripción del mapa y zona de estudio en la **Tercera Aplicación**

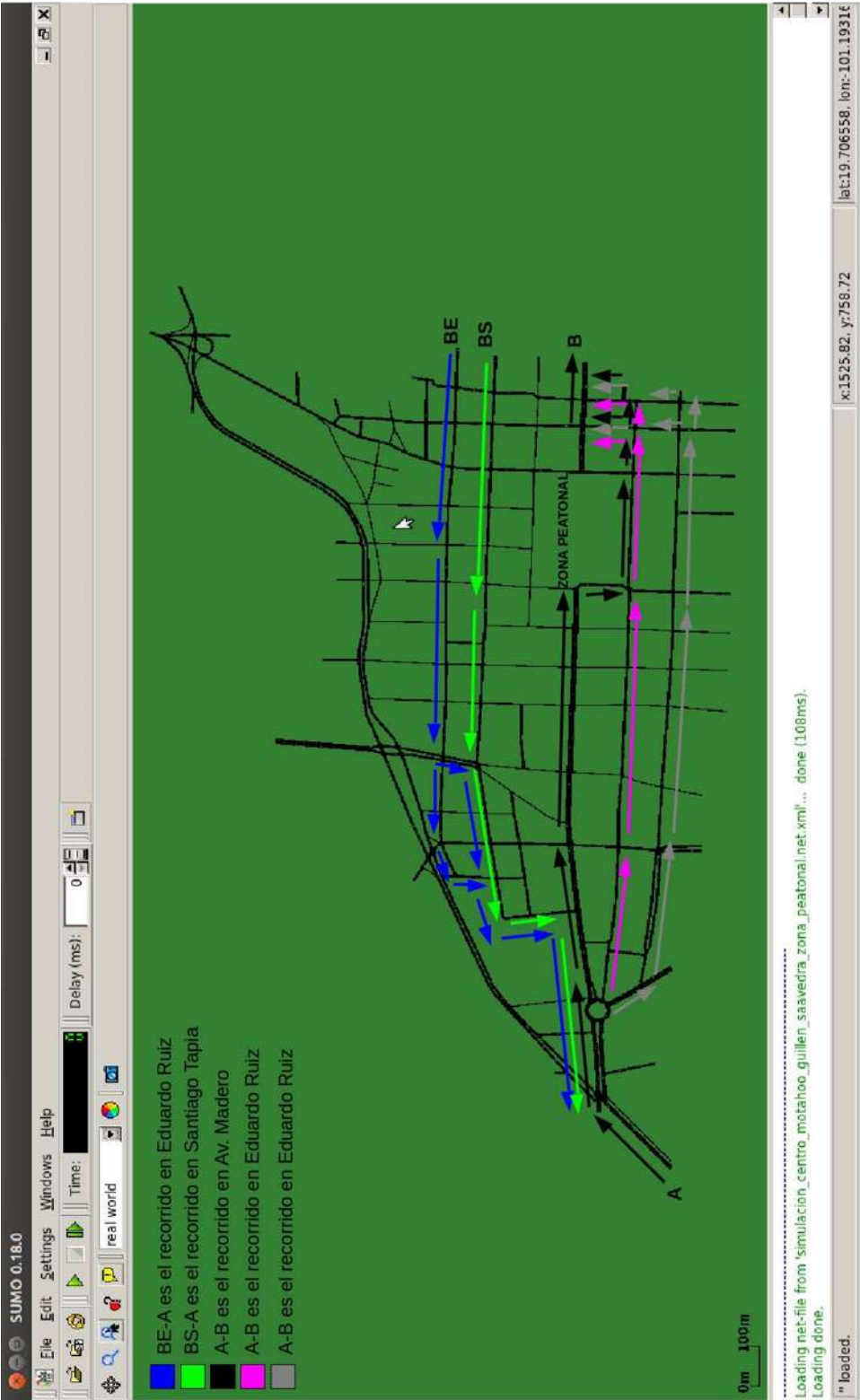


Figura 5.8: Rutas de recorrido en la Tercera Aplicación

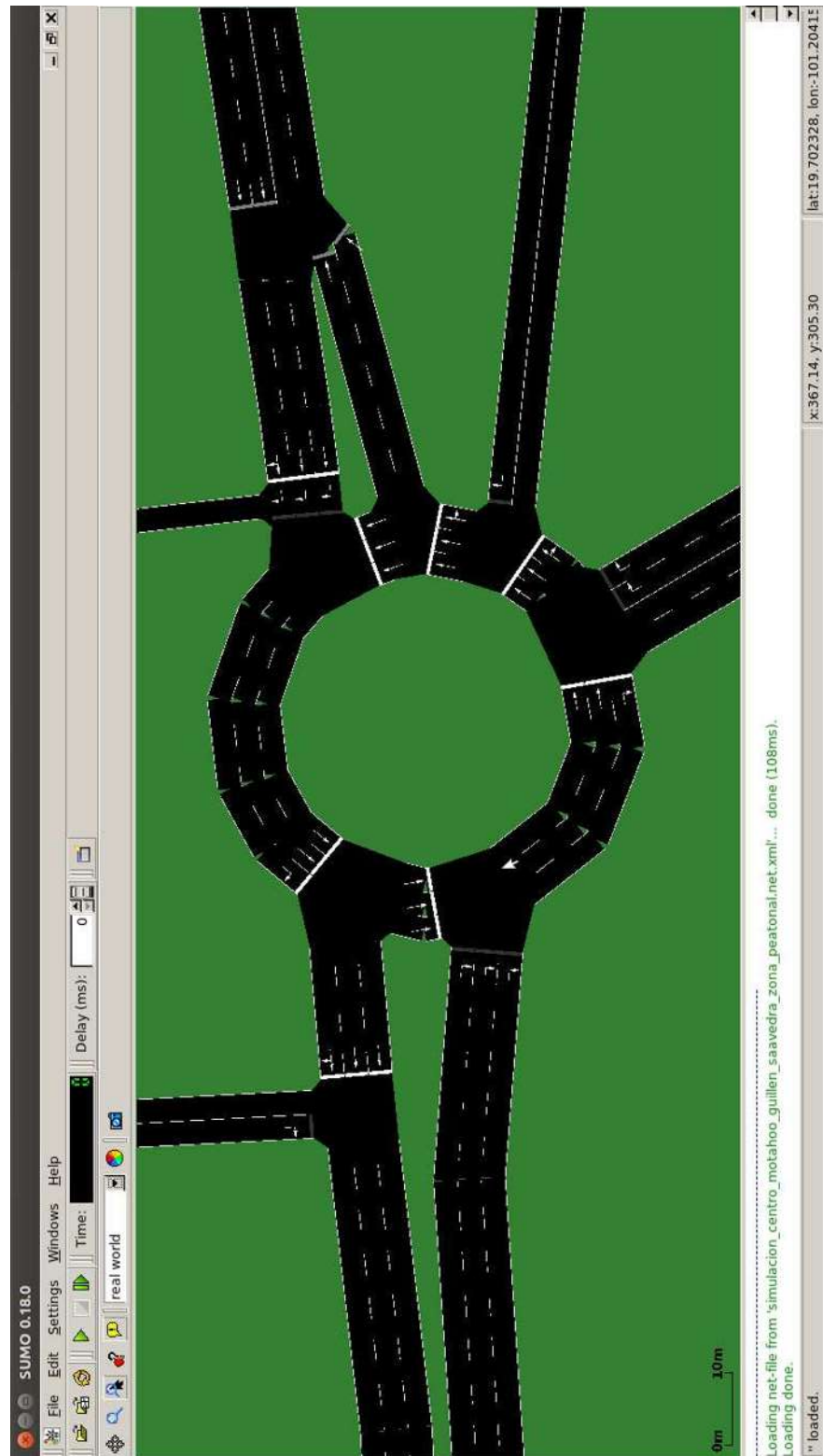


Figura 5.9: Glorieta en la intersección de la Av. Madero y la Av. Michelena

Capítulo 6

Conclusiones

Al terminar este trabajo llegamos a 4 principales conclusiones, después de desarrollar toda esta investigación y metodología. Nos dimos cuenta, como se vió en el capítulo de las **Aplicaciones**, este trabajo no queda sólo en la investigación.

La primera conclusión es que si se quiere desarrollar algún tipo de proyecto, es necesario hacer gran cantidad de trabajo de campo, además de disponer de grandes recursos computacionales dependiendo del área de la zona de estudio.

La segunda conclusión que podemos hacer es que no es difícil desarrollar la metodología mostrada en este trabajo, solo basta documentarse y tener un poco de nociones de programación.

Es de gran importancia darle seguimiento a este trabajo, esta es nuestra tercera conclusión, pues el estudio aquí realizado no solo se puede aplicar a la zona manejada en esta tesis para el estudio, puede ser aplicada desde una intersección hasta el centro histórico, además de que este trabajo no solo es aplicable a la ingeniería de tránsito, pues como se mencionó en capítulos anteriores es posible desarrollar el mismo trabajo para analizar el impacto ambiental.

La cuarta y última conclusión es la más general e importante, pues en pocas palabras, el trabajo realizado es solo una aplicación específica a nuestro bello centro histórico, pero la investigación y metodología mostradas en este trabajo abren un gran panorama de todas las aplicaciones posibles para el mejoramiento del diseño urbano presente, pues toda una gama de propuestas pueden ser investigadas, ahora, siendo orgullosamente Nicolaita, nuestra universidad es capaz de desempeñar el trabajo de mejoramiento de la planeación urbana y de tránsito, cosa que comunmente se dejaba a empresas privadas sin dar buenos resultados y que causan un gran gasto público, además es importante señalar que el trabajo no requirió de gastar ni un solo centavo en compra de software o equipo, por lo que se agrega otra gran ventaja al papel que desempeña la Universidad Michoacana.

Capítulo 7

Bibliografía

Bibliografía

- [1] F. Domínguez-Mota y J. Saavedra Rosales, “El uso de modelos dinámicos de fenómenos de transporte. Caso de estudio: el sector República del centro histórico de la ciudad de Morelia, Michoacán”, Memorias del VI Congreso Estatal de Ciencia y Tecnología, COECyT Michoacán (2010).

- [2] F. Domínguez-Mota y J. Saavedra Rosales, “El uso de modelos dinámicos de fenómenos de transporte II. Caso de estudio: el sector República del centro histórico de la ciudad de Morelia, Michoacán”, Memorias del Congreso Nacional de Ingeniería y Arquitectura 2010, ISBN 978-607-424-195-2, (2010).

- [3] M. Behrisch, L. Bieker, J. Erdmann and D. Krajzewicz, “SUMO - Simulation of Urban MObility: An Overview”, in: SIMUL 2011, The Third International Conference on Advances in System Simulation, (2011).

- [4] Ortúzar and Willumsen, “Modelling Transport”, 4th. edition, (2011).

- [5] J. Arias. Tesis profesional. “La movilidad del tránsito vehicular en la red vial del Centro Histórico de la ciudad de Morelia, Michoacán”. Facultad de Ingeniería Civil. UMSNH. Asesor: M.I. Jaime Saavedra Rosales. (2006).

- [6] Statistical Physics of Vehicular Traffic and Some Related Systems, Debashish Chowdhury, Ludger Santen, Andreas Schadschneider, Institut für Theoretische Physik, Universität zu Köln, Germany, Physics Reports 329, 199 (2000).

- [7] Modeling and Numerical Approximation of Traffic Flow Problems, Dr. Ansgar Jüngel Universität Mainz, Lecture Notes (preliminary version) Winter 2002.
- [8] Burgers equation, Mikel Landajuela, BCAM Internship - Summer 2011.
- [9] Finite Difference Methods for Ordinary and Partial Differential Equations, Steady-State and Time-Dependent Problems, Randall J. LeVeque, University of Washington, Seattle, Washington.
- [10] Mathematical Theories of traffic flow, Mathematics in science and engineering, Richard Bellman, Santa Monica, California.
- [11] Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics, Stefan Krauß, april 1994