



UNIVERSIDAD DE JAÉN
Centro de Estudios de Postgrado

APORTACIONES DE LOS SIMULADORES DE TRÁFICO EN LINARES

Alumno/a: García Ortega, Ana María

Tutor/a: Francisco Ramón Feito Higuera

Dpto: Informática



-



SOFTWARE PARA LA SIMULACIÓN DE TRÁFICO, SUMO

Autor: Ana María García Ortega

Tutor: Francisco Feito

Departamento: Departamento de Ingeniería Informática

Titulación: Máster en Ingeniería del Transporte Terrestre y Logística

Palabras clave: simulador, tráfico, SUMO, vías, software, zona urbana, zona interurbana.

Resumen

El trabajo está dedicado al estudio de los simuladores de tráfico en concreto a destinado para ver los posibles casos que se puedan dar en un entorno urbano o interurbano. El trabajo está basado en la asignatura Sistemas de Información y centrado en el estudio de un caso que se pudiera dar en Linares. El software ofrece un interface grafico de la situación dada a estudiar, ya sea un evento en ciudad, zona específica o funcionamiento de los semáforos, como por ejemplo la aglomeración de coches que se pudiera dar si hubiera una corrida de toros o un concierto.





SIMULATION SOFTWARE FOR TRAFFIC, SUMO

Author: Ana Maria Garcia Ortega

Tutor: Francisco Feito

Department: Department of Computer Engineering

Qualification: Masters in Engineering Land Transport and Logistics

Keywords: simulation, traffic, SUMO, roads, software, urban, interurban zone

Abstract

This project is devoted to the study of traffic simulators specifically designed for possible cases that may occur in an urban or suburban environment. The research is based on the course on Information Systems and it is focused on a case that could happen in Linares. The software provides a graphical interface for the possible situation to study either an event in town, a specific area or the working system of traffic lights, such as an agglomeration of cars that may happen if there were a bullfight or a concert.





INDICE

CAPÍTULO I: INTRODUCCIÓN.....	10
1.1. PRESENTACIÓN	10
1.2. OBJETIVOS.....	11
1.3. ESTADO ACTUAL DE LOS SIMULADORES	11
1.3.1. Simulador de Tráfico con AVENUE	11
1.3.2. Simulación de tráfico con AIMSUN	13
1.3.3. Simulación de tráfico con DRÁCULA	15
1.3.4. Simulación de tráfico con SUMO	18
1.3.5. Simulación de tráfico con SUMO VEINS	25
CAPÍTULO II: MÉTODO, SIMULADOR DE MOVILIDAD URBANA, SUMO	32
2.1 DESARROLLO DEL SOFTWARE Y ASPECTOS FUNDAMENTALES.	32
2.2 PRINCIPIOS DE CONSTRUCCIÓN EN SUMO.....	34
2.2.1. Preparación de una red de carreteras a simular.....	35
2.3 CALIBRACIÓN Y VALIDACIÓN.	37
CAPÍTULO III: DISCUSIÓN SOBRE UNA SIMULACIÓN EN LINARES.....	40
3.1. ÁMBITO TERRITORIAL Y ESTADO ACTUAL DE LINARES.	40
3.2. GEOGRAFÍA	41
3.3. OROGRAFÍA	41
3.4. HISTORIA	42
3.5. DEMOGRAFÍA	42
3.6. TRANSPORTES Y MEDIOS DE COMUNICACIÓN.	43
3.6.1. Red de carreteras del Estado.	43
3.6.2. Red Ferroviaria	43
3.6.3. Taxis	44
3.6.4. Carril-bici	44
3.7. RUTAS DE AUTOBUSES URBANOS.....	44
3.7.1. Líneas de urbanos	44
3.7.2. Rutas de urbanos.....	45
3.8. SIMULACIÓN DE URBANOS CON SUMO.....	48
3.8.1. Presentación.....	48
3.8.2. Datos de partida.	48
3.9. PROCESAMIENTO Y GESTIÓN DE DATOS.....	52



3.9.1.	Obtención de escenario de actuación.....	52
3.9.2.	Añadir polígonos, mapa de características.	55
3.9.3.	Creación de rutas	56
3.9.4.	Creación de semáforos.....	57
3.9.5.	Creación e inserción de vehículos	60
3.10.	Visualización en Sumo-GUI	63
CAPÍTULO IV: CONCLUSIONES.....		65
Bibliografía		67





CAPÍTULO I: INTRODUCCIÓN

1.1. PRESENTACIÓN

Gracias a las nuevas tecnologías en informática, nuevos software y algoritmos o como también gracias a los Sistemas Inteligentes del Transporte podemos realizar simulaciones de los flujos de tráfico, y diversas situaciones que nos podemos encontrar en redes tanto urbanas como interurbanas y de manera multimodal. Los simuladores nos dan una vista “microscópica” de lo que bien puede suceder o no según unas determinadas situaciones a las cuales les afectan unas u otras causas.

Las funciones de los simuladores tratan de reproducir un sistema, en el cual se reproducen las sensaciones y experiencias que podrían llegar a suceder. Se encargan de reproducir tanto sensaciones como aceleración como velocidad las cuales son sensaciones físicas que se calculan y reproducen mediante modelos matemáticos complejos adaptados a software.

También reproducen el exterior del terreno, en el cual se emplean cartografía o como también proyecciones de base de datos.

En las últimas décadas ha incrementado el número de vehículos en Linares con lo que sería viable hacer un estudio del tráfico y ver los puntos de mayor conflicto por acumulación de tráfico según un horario determinado, zona o uso de la misma. Se realizará por medio de un simulador de tráfico, en nuestro caso con Sumo Simulator. Todo ello nos da una serie de ventajas tales como:

- Conseguir una aproximación de la realidad y comportamiento de las redes de tráfico y poder actuar como mejor sea conveniente.
- Permiten hacer pruebas con el cambio de señales de tráfico, semáforos...
- Poder simular casos a tiempo continuo o en días o casos eventuales.
- Los simuladores suelen ser de bajo coste, si usamos de libre licencia.
- El margen de error a la hora de simular cualquier escena es bajo, ya que pueden haber algunos factores externos inesperados en el flujo de tráfico.
- Previsiones para futuros proyectos múltiples.



1.2. OBJETIVOS

Este trabajo ofrece como los simuladores analizan y estudian las acciones básicas del día a día o con eventos, estudian su respuesta ante los diversos factores que pueden afectar tanto al tráfico como a el mismo peatón, su tiempo de respuesta ante dicha situación y sus parámetros característicos, con los cuales se modifica y ve como resultaría viéndolo desde otra perspectiva gracias a los simuladores.

1.3. ESTADO ACTUAL DE LOS SIMULADORES

1.3.1. *Simulador de Tráfico con AVENUE*

❖ Introducción

En Japón se inicia sobre 1970 los modelos de simulación de tráfico, como sabemos hay gran densidad de tráfico en dicho país. Los métodos pioneros en la Universidad de Tokio son el método de entrada-salida y el método de densidad de bloque en el cual una sección de carretera se divide en bloques, los cuales tienen unos cientos de metros y también se tienen los flujos de vehículos con su propagación. Para el control de estos bloques se usa el diagrama fundamental de la carretera. (Barceló, 2010)

También había otros modelos que no tenían elección de rutas o y se aplicaban a zonas pequeñas. Otros modelos anteriores en Japón como Macstran y Mistran por el Instituto Nacional de Investigación de Ciencias Policiales estaban dirigidos para la evaluación de sistemas de gestión de tráfico incluyendo también control de señales de tráfico. (Barceló, 2010)

En el comienzo de los 90, se desarrolló SONIDO (Simulación en redes viales urbanas con guía dinámica de ruta). Este simulador para hacer su elección de ruta usa una función de elección de la misma para una aplicación a gran escala. A causa de ser a gran escala esta aplicación, los modelos de flujo para este simulador son más simplificados, por ejemplo podríamos nombrar que no hay cambio de carril, un vehículo se basa en la relación espacio/velocidad. Sonido es muy práctico para evaluar planificaciones de toda una red pero en áreas más pequeñas está más limitado. (Barceló, 2010)

A causas de estas limitaciones que suponían estos simuladores se desarrollo AVENUE (Avanzado Evaluador Visual para redes viales en zonas urbanas) con propósito de gestionar el tráfico de ámbito más local. (Barceló, 2010)



❖ Principios de modelado

- ***Enfoque Híbrido del Modelado del flujo de tráfico***

Avenue emplea el “Bloque híbrido Método Densidad” (HBDM) en el cual se combina las imágenes de vehículos discretos y la aproximación del flujo de tráfico. (Horiguchi et al., 1994). En general se requiera una evaluación rigurosa y simulaciones de tráfico para evitar o reducir en lo máximo posible los cuellos de botella y mejorar la dinámica del flujo de tráfico. (Barceló, 2010)

Avenue trata a los vehículos representa los atributos individuales a través de imágenes discretas, así como destino, origen o tipo de vehículo. Un vehículo fluye en la carretera de acuerdo con la dinámica de fluidos, pero como sabemos puede elegir carriles y rutas de acuerdo a las preferencias individuales. (Barceló, 2010)

- ***Ruta Dinámica Elección Modelo***

Son necesarios para saber la “preferencia” de elección de rutas de los conductores. Avenue implementa muchas clases de rutas modelo y con diversos pilotos de acuerdo a (óptima dinámica de usuario) y DSUO (óptimos de usuario dinámico y estocástico) principalmente. Todo va ligado a un coste genérico y las clases se encuentran parametrizadas, considera los peajes número de vueltas, tiempo estimado del viaje etc. Avenue también tiene en cuenta rutas fijas para tipos de vehículos especiales como autobuses o metro ligero. (Barceló, 2010)

- ***Marco Común de Modelos de Simulación del tráfico de red***

Avenue, Sonido y otros simuladores son implementados en un marco común de simulación de datos para así ser compartida entre Avenue y otros simuladores. La ejecución de programación del marco está orientada a objetos y proporciona clases fundamentales en términos de vehículo generación, la acumulación de datos de resultados, la guía de ruta, señales de tráfico, la guía de ruta... (Barceló, 2010)

Gracias a la adición de las subclases los usuarios pueden ampliar la capacidad del modelo e implementar funciones deseadas. (Barceló, 2010)

❖ Todo en Uno Software Package

Es una política de diseño de software, no un tipo de modelado. Algunos paquetes del software contienen varios sub-programas, como "constructor de la red ". Quizás puede ser molesta para algunos usuarios las ventanas de conmutación, apertura o cierre de la aplicación. (Barceló, 2010)

En el caso de Avenue los paquetes van todos en un mismo software. Así los usuarios pueden construir los datos de simulación con GUI y verificar continuamente los datos.

La siguiente figura muestra incluyendo el constructor de datos, el gestor de capa de datos, y la 3-D. (Barceló, 2010)

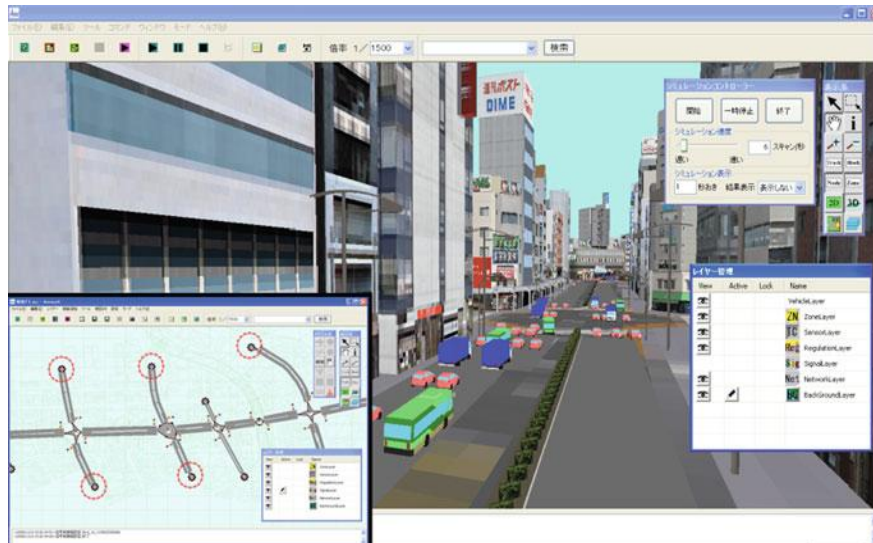


Fig 1. Gestor de capas y 3-D (Barceló, 2010)

1.3.2. Simulación de tráfico con AIMSUN

❖ Introducción

▪ Antecedentes y visión general

Es un software de modelización del transporte de un programa de investigación a largo plazo de la Universidad de Cataluña, hoy en día se encuentra en su sexta versión comercial. Después de haber superado el objetivo declarado de la AIMSUN originales “avanzado simulador microscópico interactivo para redes urbanas y no urbanas” acrónimo (Ferrer y Barceló., 1993; Barceló et al, 1994, 1998a), hoy día el software incluye modelos macroscópicos, mesoscópicos y microscópicas y modelos dinámicos de asignación de tráfico basado en el equilibrio, ya sea de usuario o la elección de ruta estocástica. (Barceló, 2010)

La ingeniería de tráfico y fuera de línea son las ramas principales en las que trabaja Aimsun, más recientemente, en línea (tiempo real) como decisiones a la gestión de tráfico. Los objetivos que tiene Aimsun principalmente son proporcionar soluciones a corto y medio plazo. Hay importantes interfaces entre esos dos dominios a nivel de metodología y tecnología. (Barceló, 2010)



▪ Principios de Desarrollo

El ritmo rápido de la evolución de Aimsun, casi garantiza que algunos aspectos de nuestra descripción sean obsoletos. Veamos los principios de desarrollo de Aimsun.

Integración: El volumen constante mejora y la calidad de los datos de tráfico, la disponibilidad de los recursos informáticos todo da lugar a mejora en la experiencia profesional. Los modelos están en constante aumento en el tamaño, complejidad y detalle, se reconocen explícitamente los estudios de modelización. (Barceló, 2010)

En el caso de Aimsun, la información compartida por todos los modelos es topología de la red, la demanda y en función del tiempo caminos más corto (más baratos) y sus respectivos tiempos de viaje (o costos). Las entradas de rutas, caminos o demandas no son simplemente entradas compartidas, más bien es el caso de que los modelos puedan permitir ser explotados por otro modelo de Aimsun. (Barceló, 2010)

El principio que se toma es afiliar todo lo que puede y debe ser compartida entre todos los modelos que permitan a los flujos de trabajo sofisticados que implican secuencia, interactividad e incluso aplicación simultánea de dos o más modelos. Otro objetivo es la integridad, es decir, integrar todo lo que se necesario para el fin del modelado. (Barceló, 2010)

❖ *Los objetivos del estudio integral:*

▪ Modularidad:

Trata de la ruptura de los procesos, lo que permite una reutilización consistente y más fácil dentro de los procesos de mayor tamaño. La modularidad permite a los usuarios desacoplar los modelos de dinámica de enlace de los procesos por los cuales el tráfico se asigna dinámicamente a varias rutas. Con esto se consigue dos nuevas combinaciones con lo que se obtienen aplicaciones más fáciles y útiles. Por un lado está la combinación de la simulación mesoscópica con la asignación dinámica del tráfico por mediante la elección de ruta estocástica, puede ser una herramienta para la ruta más rápida. (Barceló, 2010)

También puede especificar a los conductores la distancia de la zona de influencia del incidente y seguir siguiendo las rutas establecidas (los resultados de un equilibrio dinámico anterior asignación). (Barceló, 2010)

En segundo lugar al combinar dinámica de equilibrio de usuario y simulación microscópica hace que la transferencia modal, peatonalización, carriles exclusivos de autobús, sufra cambios duraderos en la red de utilización y en la que peatones, transporte público y vehículos privados interactúan en paradas y cruces creando dependencias mutuas que afectan a la capacidad en el complejo de diversas maneras. El principio subyacente es hacer que los “ingredientes” básicos se rompan para volver a combinar y crear otros métodos más prácticos. (Barceló, 2010)



- **Escalabilidad:**

Anualmente la informática de hardware mejora con lo cual hay una mejora de la disponibilidad de los datos y su calidad. (Barceló, 2010)

Al combinar la calidad y disponibilidad conseguimos un deseable modelo dinámico, sería aplicable en zonas como Montreal. También es posible con simulación mesoscópica o implementaciones multi-threaded. En los próximos años se desafía el seguimiento del cambio del ritmo de los paradigmas de cálculo, también sabiendo que los requisitos de los usuarios son cada vez más exigentes y las respuestas de rendimiento esperadas cada vez más estrictas. (Barceló, 2010)

- **Interoperabilidad y extensibilidad:**

Con la abundancia de ITS junto a tecnologías como SIG, cartografía y 3D significa una creación de un modelo de redes de carreteras más preciso y difícil aunque lo sea es a la vez más fácil. Hay una serie de novedades que se necesitan para ser capturados por los modelos con precisión, son novedades tales como: misiones carril dinámico, control de la señal de adaptación o como control de velocidad variable. (Barceló, 2010)

En el caso del uso de los mapas de alta calidad y datos SIG el modelo se simplifica en comparación de potros años atrás, un ejemplo es el de los edificios 3D, usando una perspectiva de la realidad virtual con una escala viable. (Barceló, 2010)

Sea cual sea el caso es necesario la interoperabilidad, que es la capacidad para hacer intercambio de datos con otras aplicaciones en una variedad de formatos, y la extensibilidad, la cual es la capacidad que tienen los usuarios para programar ciertas extensiones con una determinada facilidad. (Barceló, 2010)

1.3.3. Simulación de tráfico con DRÁCULA

❖ Introducción

Este modelo de simulación de tráfico se diseñó con el fin de obtener una herramienta que sirviese para saber la dinámica entre oferta y demanda interacciones en las redes de carreteras. (Barceló, 2010)

Por esto, la importancia de microsimulación, integrando decisiones responsables, viajes individuales, experiencias y aprendizajes. Todo esto se representa por medio de modelos de modelos basados en la asignación dinámica del tráfico microscópico, en modelado explícito de opciones ruta y hora de salida de los individuos del día a día y cómo su experiencia pasada y el conocimiento de la red influye en sus decisiones futuras. (Barceló, 2010)

También junto a esto está la microsimulación de tráfico dentro del día basado en el coche de seguimiento y reglas de cambio de carril. (Barceló, 2010)



Algunas de las funciones que tiene el software son como en el modelado, respecto a los caminos rurales de dos carriles, el comportamiento del adelantamiento en dichos caminos y la integración plena con un modelo microscópico de las operaciones de transporte público y la demanda de este transporte, ambos son representados. (Barceló, 2010)

❖ Principio de modelados de construcción con DRÁCULA

El modelo de microsimulación red dinámica Drácula (Ruta Dinámica Asignación Combinando el aprendizaje de usuario y microsimulación) se desarrolla en la Universidad de Leeds desde 1993. El enfoque que se le dio fue único para el modelado de redes de tráfico, en el cual se mas hace énfasis en la “microsimulación” de opciones y vehículos de los individuos responsables del movimiento del viaje. (Barceló, 2010)

Este modelo intenta hacer la representación directa a tiempo real, de los conductores pues este comportamiento varía día a día. El “withinday” se le suma a todo esto, que consiste en una simulación detallada del tráfico con sus movimientos continuos de vehículos individuales de acuerdo para el coche de seguimiento y las reglas de cambio de carril y controles de tráfico. (Barceló, 2010)

También se combina la evolución del sistema de tráfico a través de un cierto número representativo de días para que dentro de una variabilidad de días se incluya la interacción entre la oferta y la demanda. (Barceló, 2010)

Drácula está compuesto por una serie de sub-modelos. El modelo representa la variabilidad de en el día a día de la demanda total dentro de un periodo de tiempo. Simula para cada viajero en función de los conductores individuales conocimientos de la red como la ruta seguida partida de tiempo preferida. (Barceló, 2010)

Cuando se tiene la información pasa a la simulación de tráfico en dicho modelo se representa la variabilidad dentro de los días establecidos y condiciones de la red. Al introducir los datos simula la ruta elegida por el vehículo individual y su rendimiento es registrado. (Barceló, 2010)

Tras el periodo de estudio, las experiencias de cada individuo son almacenadas con su información de la historia del viaje, que pueden o no a mayor o menor medida, influenciar en las decisiones del día siguiente. (Barceló, 2010)

❖ Estructura Modelo DRÁCULA

Los sub-modelos y la evolución dinámica de las interacciones de demanda-oferta que representa son los siguientes: (Barceló, 2010)

- I. Un sub-modelo de la población, genera todos los conductores potenciales sintetizados de una población en un área de estudio.(de un punto “a” a un punto “b”)
- II. Un sub-modelo de demanda representa la variabilidad de la demanda total del día a día. Con ello se puede predecir la demanda por individuo de una población particular con potencial de conductores.
- III. Un sub-modelo de asignación de tráfico determina las rutas, salidas y los tiempos de los conductores individuales, en función de la experiencia del viaje realizado y el conocimiento percibido en dichas condiciones de la red.
El resultado que da es el viaje realizado desde su salida en el origen de partida hasta su destino final a través de la trayectoria realizada en su tiempo de salida.
- IV. Sub-modelo de microsimulación de tráfico, los vehículos se desplazan a través de una red siguiendo las rutas elegidas por ellas y esto es representado por un sub-modelo de simulación, todo siguiendo las reglas de cambio de carril.
- V. Costos experimentados por los conductores en el día y cada uno de los enlaces por los que pasan. Después paso a ser base de sus conocimientos. Con el modelo de aprendizaje se actualiza la percepción de del conductor de las condiciones de la red que a la misma vez afectaría a su decisión en el día siguiente.
- VI. Un sub-módulo de recogida de datos como : tiempo epleado en el viaje, incidencias, emisión ...
El sistema evoluciona cada día continuamente hasta alcanzar un estado ciertamente equilibrado entre la oferta y la demanda. Los resultados son obtenidos a lo largo de la evolución, resultados de las varianzas, distribuciones de probabilidad.... Tanto como dentro de los días o entre ellos.

La simulación es animada por medio de una interface grafica de usuario (GUI) que sirve tanto para examinar tráfico como depurar.

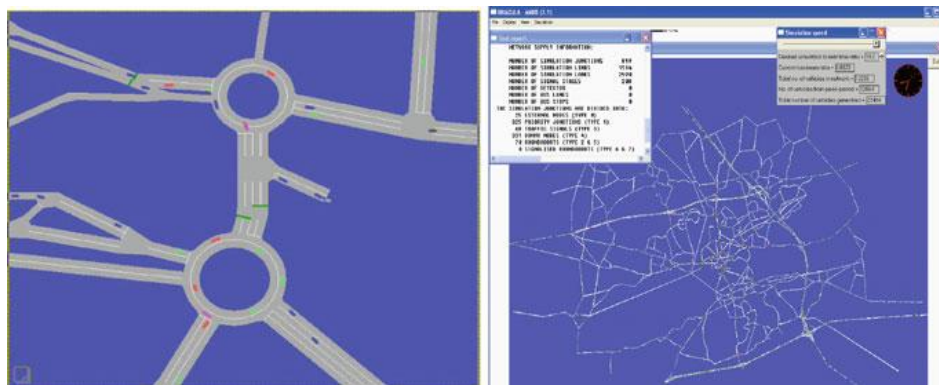


Fig 2. Instantánea de la animación vehículo (Barceló, 2010)

❖ Datos de origen y vinculación con los modelos convencionales de red

Dracula, que es un modelo de microsimulación requiere de los datos necesarios como los modelos de red tradicionales como lo es SATURNO, necesita sus nodos, enlaces número de carriles, operaciones de señal, marcas de carril...



Hay una relación directa entre Drácula y Saturno en que la primera tiene las redes y demanda de datos básicos de este último. Los beneficios son los twofolds. Se aprovecha el banco de datos de las redes de Saturno desarrollado en todo el mundo por Drácula, por lo que reducir el tiempo que se necesita para codificar una red desde cero. (Barceló, 2010)

1.3.4. Simulación de tráfico con SUMO

❖ Introducción

Sus siglas SUMO significan “Simulación de Movilidad Urbana” es un simulador multimodal e intermodal en una simulación de flujo de tráfico de espacio discreto y tiempo continuo. Dicho software, en el cual profundizaremos más en este trabajo, fue iniciado en 2001 y su primera apertura fue en 2002. Es software de código abierto y esto es debido a dos razones, para hacer el trabajo disponible como código abierto bajo la GNU Public License (GPL). (Daniel Krajzewicz, 2012)

La primera razón fue con el deseo de apoyar el tráfico a la comunidad con una herramienta gratuita la cual posee algoritmos que se pueden implementar. Otros simuladores de tráfico de código abierto también se encontraban disponibles pero se encuentran en tesis de alumnos.

La segunda razón para hacer un código abierto fue para obtener el apoyo de otras instituciones.

En la última década Sumo ha destacado en los servicios públicos de modelado de tráfico destacando en los servicios públicos de modelado de tráfico que incluyen un camino importado de redes capaces de leer en distintos formatos, utilidades de enrutamiento y generación de demanda que utilizan grandes y altos números de fuentes de entrada (matrices de destino de origen, conteos de tráfico, etc.), una simulación de alto rendimiento utilizable para uniones individuales, o como ciudades enteras incluyendo un control remoto de interfaz. (Daniel Krajzewicz, 2012)

Se puede hacer la adaptación en línea y para ello costa de muchas herramientas adicionales y scripts. Su mayor parte de desarrollo se lleva a cabo por el Instituto de Sistemas de Transporte en el alemán Centro Aeroespacial (Deutsches Zentrum für Luft- und Raumfahrt, DLR). Las partes externas son soportadas en diferentes ampliaciones de la suite simulación. (Daniel Krajzewicz, 2012)

❖ Suite de Sumo

Sumo es más bien una suite de aplicaciones y no solo un simulador de tráfico el cual ayuda a simular el escenario. Conforme Sumo se va transformando en una suite va

incluyendo formatos propios tales como para demanda de tráfico y redes de carreteras y estos son importados a partir de una fuente ya existente. La principal aplicación que tenemos en mente que realiza Sumo es la de simular zonas a gran escala, pero a esta le faltan atributos. (Daniel Krajzewicz, 2012)

❖ Aplicaciones en la suite de Sumo

a) Generación de red de caminos

Sumo representa las redes de carreteras del mundo como gráficos en el cual las intersecciones se representan como nodos y las carreteras por líneas o bordes. Las intersecciones consisten en una posición, una la forma y la derecha el paso de normas en el que se pueden sobreescribir los semáforos. Los bordes son conexiones unidireccionales entre los nodos y tienen un número no variable de carriles. Los carriles tienen la geometría, información sobre los vehículos que pueden circular por ellos y velocidad máxima en ellos. Así pues en los carriles se representan los cambios a lo largo de un camino por múltiples aristas. Hay puntos de vista desde distintos enfoques como desde VisSim o Formato de red o el OpenDrive son formatos existentes.

Además Sumo también podemos decir que incluye en la red de carreteras los planes de semáforos, conexiones entre carriles a través de unas intersecciones que describen qué carriles pueden ser utilizados para llegar a un carril posterior.

Para generar la red de carreteras desde Sumo se puede hacer desde la aplicación "netgenerate" o mediante la importación de una cámara digital hoja de ruta utilizando "netCONVERT". Esta ultima aplicación construye tres tipos de redes de carreteras: las "manhattan", como lo son las redes grid, las redes circulares "araña neta", y redes al azar. (Daniel Krajzewicz, 2012)

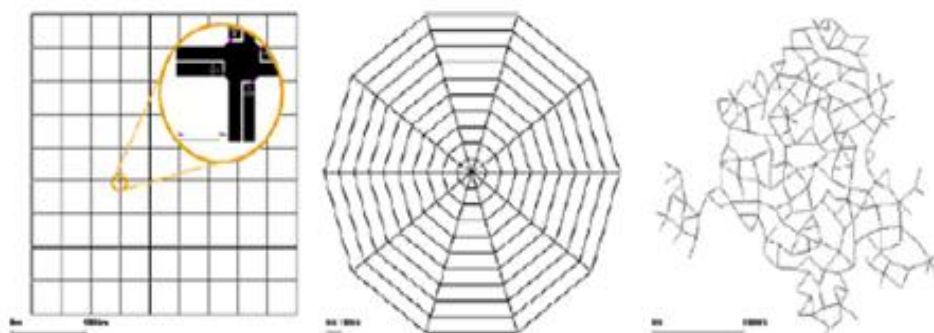


Fig. 3. Muestra ejemplos de las redes generadas. (Daniel Krajzewicz, 2012)

NetConvert (importador de redes) también convierte redes de otros simuladores de tráfico como los de Visium, MatSim o VisSim. También puede leer otros formatos

digitales de red como son OpenStreetMap o shapefiles o menos conocidos como RoboCup o OpenDrive.

Las siguientes figuras muestran las capacidades de como se importan carreteras en OpenStreetMap web, y es comparada con una representación de Sumo. (Daniel Krajzewicz, 2012)



Fig 4. Marco de trabajo Original de OpenStreetMap de Gothenborg (Daniel Krajzewicz, 2012)



Fig 5. Marco de trabajo importado en Sumo de Gothenborg (Daniel Krajzewicz, 2012)

La mayoría de las redes de carreteras digitales que se encuentra disponibles son destinadas a ser utilizadas en propósitos de navegación. Normalmente como tales carecen de atributos suficientes para el detalle necesario para una simulación microscópica de tráfico, a falta de detalles en tales como posiciones de semáforos, intersecciones, información sobre carriles....etc. (Daniel Krajzewicz, 2012)

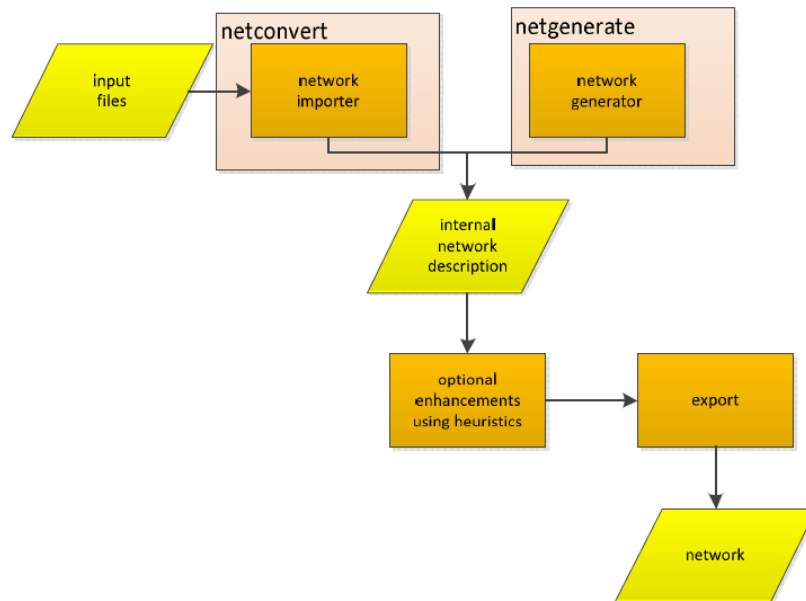


Fig 6. Procedimiento de preparación de la red común en NetConvert y netgenerate. (Daniel Krajzewicz, 2012)

También hay una nueva herramienta llamada NetEdit la cual permite editar las redes de carreteras gráficamente. Suele ser más simple y rápido que en la preparación de los archivos de entrada XML. También combina los pasos separado de la generación de la red de inspección con NetConvert y la GUI simulación. Esta herramienta aun no se encuentra disponible al público. (Daniel Krajzewicz, 2012)

b) Vehículos y rutas

Como ya se ha comentado anteriormente Sumo es un simulador microscópico en el cual cada vehículo está dado explícitamente y definido al menos por un identificador, la hora de salida y la ruta que este lleva. Cuando hablamos de ruta nos referimos a la lista completa de bordes conectados entre el inicio y el destino de un vehículo. Si fuera necesario también se puede describir un vehículo con más precisión determinando su hora de salida y llegada, también como que carriles ha de usar, la posición exacta en un borde o la velocidad. A los vehículos se le puede poner un tipo asignado definiéndoles sus variables del modelo y propiedades tales como sus emisiones o contaminantes o contaminación acústica, ruido. Las variables adicionales nos dan la información de cómo es en sí dicho vehículo en la interfaz gráfica dentro de una simulación. (Daniel Krajzewicz, 2012)

Una gran ciudad tiene un gran escenario de simulación en el que fácilmente circulan millones de vehículos con sus respectivas rutas. También resulta difícil en un área pequeña definir la demanda de tráfico de forma manual.



Las suites de Sumo tienen aplicaciones que utilizan diversas fuentes para crear la demanda. Para escenarios a gran escala, por lo general se utilizan matrices de destino (matrices O / D). Describen el movimiento entre las zonas llamadas de análisis de tráfico (TAZ) en el número de vehículos por hora. Estas matrices para ser usadas en Sumo se desglosan en viajes de vehículos individuales con los tiempos de ida repartidos por el lapso del tiempo descrito. (Daniel Krajzewicz, 2012)

Al día solo se da una sola matriz y esto hace que sea impreciso puesto que es un tráfico de simulación microscópica y los flujos entre dos TAZ varían fuertemente en un día. Como por ejemplo las personas que se mueven diariamente en los centros de la ciudad para ir a trabajar o salir por la tarde o noche. Estos cambios de dirección no se pueden agregar a una matriz que sea de veinticuatro horas, pero las suites de Sumo también disponen de matrices de una sola hora más útiles pero no siempre disponibles. La suite de Sumo "od2trips", sirve para conservar matrices O/D solo para viajes de vehículos. (Daniel Krajzewicz, 2012)

También hay una aplicación que asigna opcionalmente un borde de la carretera la red como posición de salida y llegada respectivamente. Las zonas de asignación a otros bordes han de ser suministrados como otra entrada. Los viajes obtenidos de od2trips se basan en comenzar en una calle y a una hora de salida, pero esta simulación ha de estar completa de los bordes por donde vaya a pasar. Las rutas se calculan generalmente mediante la realización de una asignación dinámica de usuario (DUA). Es un proceso iterativo donde se ve enrutamiento, camino más corto y diferentes costos. (Daniel Krajzewicz, 2012)

En Sumo también encontramos aplicaciones de cómputo de ruta. La primera es "jtrouter", utiliza las definiciones de turnos para calcular porcentajes en las intersecciones y calcular rutas a través de la red con estos porcentajes. Puede ser utilizado para establecer la demanda de una parte de la red vial con pocos nodos de una ciudad. La segunda aplicación que encontramos es "dfrouter", que calcula rutas mediante la información de un bucle inductivo u otra detección de sección transversal. Al ser detector transversal es muy efectivo en zonas de autopistas o carreteras cubiertas con detectores. Cuando los detectores son bajos como es obvio se producen ciertos errores pues hay cobertura baja. (Daniel Krajzewicz, 2012)

Dos herramientas incluidas en Sumo, Traffic Modeler y Activitygen ofrecen una interfaz gráfica de usuario que permite que el usuario pueda configurar las fuentes de demanda y sumideros gráficamente. Son incluidas en esta suite y utilizan modelos propios para la creación de movilidad que se desea para un área investigada, requiriendo diferentes datos. (Daniel Krajzewicz, 2012)

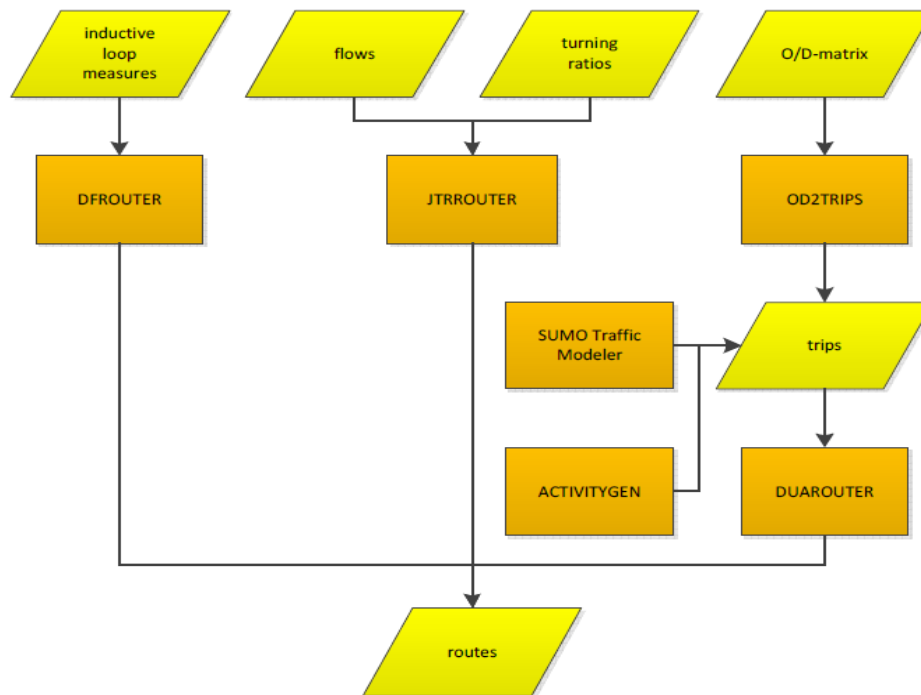


Fig 7. Métodos admitidos para la generación de demanda. (Daniel Krajzewicz, 2012)

Esta figura resume las posibilidades para la creación de una demanda para hacer una simulación utilizando estas herramientas incluidas en SUMO.

c) Simulación

La aplicación del simulador se realiza en un tiempo discreto. El tiempo suele ser de un segundo pero se puede acortar a un microsegundo. Internamente, el tiempo está representado en microsegundos, almacenados como valores enteros. La máxima duración de un escenario está ligada a 49 días. La simulación interna del modelo espacio continuo está descrito por el carril del vehículo y la distancia recorrida desde el inicio del carril. Para calcular la velocidad del vehículo en la red se calcula utilizando el llamado modelo coche siguiente. Los modelos de coches de seguimiento suelen computar la velocidad de un vehículo, mirando la velocidad de este y la de su vehículo líder, Sumo utiliza una extensión del modelo de coche siguiente estocástico desarrollado Stefan Krauß, por defecto. Este fue elegido por su gran velocidad de ejecución y sencillez también ha sido demostrado que es válido dentro de un conjunto de comparaciones de modelos de automóviles de seguimiento de las realizadas. Sin embargo tiene alguna deficiencia como que produce una baja aceptación durante el cambio de carril y el modelo tampoco escala bien cuando se pasa la longitud del paso del tiempo. (Daniel Krajzewicz, 2012)

Actualmente entre otros modelos de seguimiento, aparte de Sumo o Lidiar encontramos modelos tales como: modelo de controlador inteligente (IDM), Kerner de modelo de tres fases y el modelo Wiedemann. Pero también se encontraban caminos complejos de

las redes en estos modelos, y con posibles restricciones indefinidas con lo que el uso de diferentes modelos de automóviles de seguimiento debe ser declarado como experimental solamente, en el momento actual. (Daniel Krajzewicz, 2012)

En una simulación hay posibilidades a limitar el comportamiento de propio de un conductor como también se le puede dar a cada vehículo su propio conjunto de parámetros suelen ser desde longitud del vehículo para modelar parámetros como avanzar como se prefiera en el tiempo e incluso dejar que los diferentes modelos funcionan juntos. Para calcular el cambio de carril se realiza utilizando un modelo que se desarrolla mediante la implementación de Sumo. (Daniel Krajzewicz, 2012)

Encontramos dos versiones de simulación de tráfico los de Sumo que es una aplicación en línea de comandos para simular de manera eficiente. Los Sum-gui con interfaz gráfica de usuario (GUI) prestados de la simulación de la red y los vehículos que utilizan OpenGL. La visualización se puede hacer al gusto con lo que se quiera representar tanto como velocidades, seguimiento a vehículos particulares o tiempos de espera. (Daniel Krajzewicz, 2012)

También complementariamente hay elementos gráficos, como polígonos, imágenes o puntos de interés que permiten tener mejor visual del escenario simulado. La GUI también da la opción de interactuar con el escenario como cambiando rutas o redireccionando, cambiar velocidades... (Daniel Krajzewicz, 2012)

A continuación vean una imagen con una intersección simulada en Sumo-gui. Este ofrece toda la línea de comandos y características propias. (Daniel Krajzewicz, 2012)

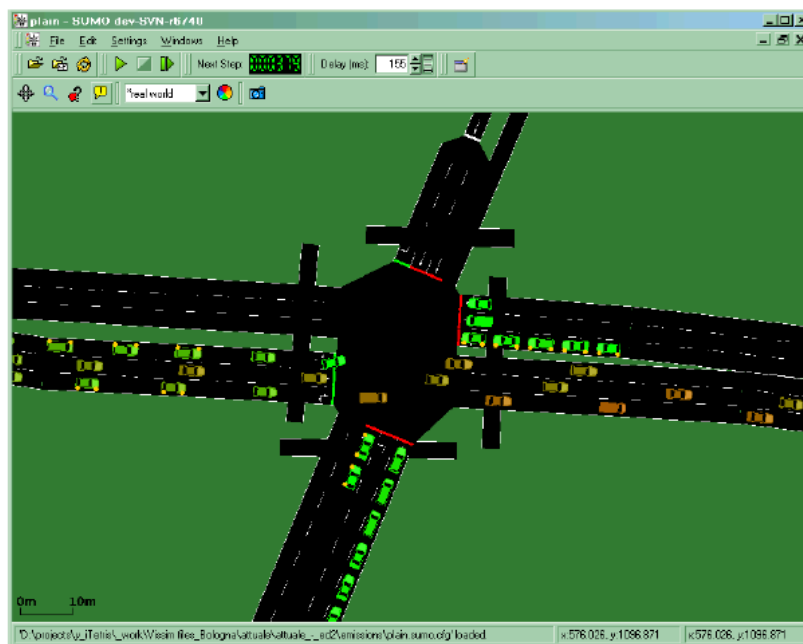


Fig 8. Captura de pantalla de los vehículos, (Daniel Krajzewicz, 2012)

Sus emisiones de CO2 por color según emisión.



Sumo ofrece poder generar distintas salidas para cada ejecución que se realice en la simulación. También añade a parte de las medidas convencionales de tráfico las emisiones de contaminantes, emisión de ruido así como el consumo de combustible. Los archivos generados por Sumo están en formato .XML. (Daniel Krajzewicz, 2012)

d) Interacción On-line

A partir de 2006, la aplicación se amplió la aplicación y así actuar, interactuar con una aplicación externa a través de conexión socket. Esta es la aplicación es Traci para Interface Traffic Control por Axel Wegener y sus compañeros y la puso a disposición en el lanzamiento de Sumo. (Daniel Krajzewicz, 2012)

Para comenzar a usar Sumo en línea tiene que ser iniciado con una opción adicional para tener el número de puertos. Cuando la simulación ha cargado Sumo tiene en cuenta el número de puertos cargados para la conexión entrante. Cuando ya ha sido conectado, el cliente es responsable de desencadenar los pasos de la simulación en Sumo, como cuando se cierra o pierde la conexión que obliga al simulador a dejar de funcionar. El cliente que entra en línea puede acceder a todos los parámetros y valores como vehículos, bordes, intersecciones, semáforos etc... Pero también el cliente puede manejar a su manera o deseo los valores, como cambiar el programa de semáforos. Con esto se puede realizar una interacción compleja como la sincronización en línea del tráfico luces o el modelado de comportamiento especial de vehículos individuales. (Daniel Krajzewicz, 2012)

Mientras en DLR se utiliza principalmente la biblioteca de datos de Python al interactuar con la simulación, el cliente puede escribir en cualquier lenguaje de programación siempre y cuando el sockets TCO sea compatible. (Daniel Krajzewicz, 2012)

1.3.5. Simulación de tráfico con SUMO VEINS

❖ Introducción

A hoy día la comunicación inalámbrica es un aspecto fundamental para los avances que se han logrado los cuales hacen de nuestro día a día más ameno, sencillo, práctico, fácil... La comunicación inalámbrica afecta también a la comunicación para los coches (C2C). Las redes ad hoc son las Vanet son las denominadas para la comunicación entre coches. Estas redes ofrecen el provecho de disminuir atascos o accidentes, acortar rutas o acortar las emisiones de gases. Hay un inconveniente a la hora de simular con simuladores y es que se puede simular una hipótesis pero no la cantidad de vehículos



circulantes de la ciudad o zona sin tener los datos más exactos posibles. Para este problema lo que se realiza es estudiar el marco donde se encuentra el tráfico a simular con SUMO, y una red OMNET++, simulador en paralelo y así poder simular los tráfico reales. (Noori, 2012)

A) Redes Vehiculares ad-hoc.

Las redes VANET representen las redes ad-hoc Mobile Networks, lo que hacen es unir, comunicar de manera económica los vehículos entre sí y las carreteras y así recibir y enviar información. Los conceptos básicos y principales que nos encontramos en las redes VANETs son nodos, coches. Los nodos se pueden mover muy rápido con lo que se da que la red sea bastante dinámica, al igual que su densidad de vehículos es muy variable. (Noori, 2012)

B) Simulación de redes VANETs.

Dado a la nueva tecnología inalámbrica, complejidad de gastos y nuevos protocolos, no se pueden llevar a pruebas grandes cantidades de datos para su simulación con lo que se ha de buscar distintas maneras de enfocar las VANETs. La simulación de VANETs necesita fundamentalmente la movilidad y la red de estudio. Hay varios software que realizan la simulación de dichas redes: los que simulan red, los generadores de movilidad y el que combina ambos o puede simular tanto uno como otro. (Noori, 2012)

A la hora de simular esto ha de ser realista porque si no, no se podría poner como una red de entrada.

Los simuladores crean y calculan los componentes necesarios para una red inalámbrica, tal y como sería una red detallada con sus coches, los cuales son los nodos, transmisión de tráfico de datos.... (Noori, 2012)

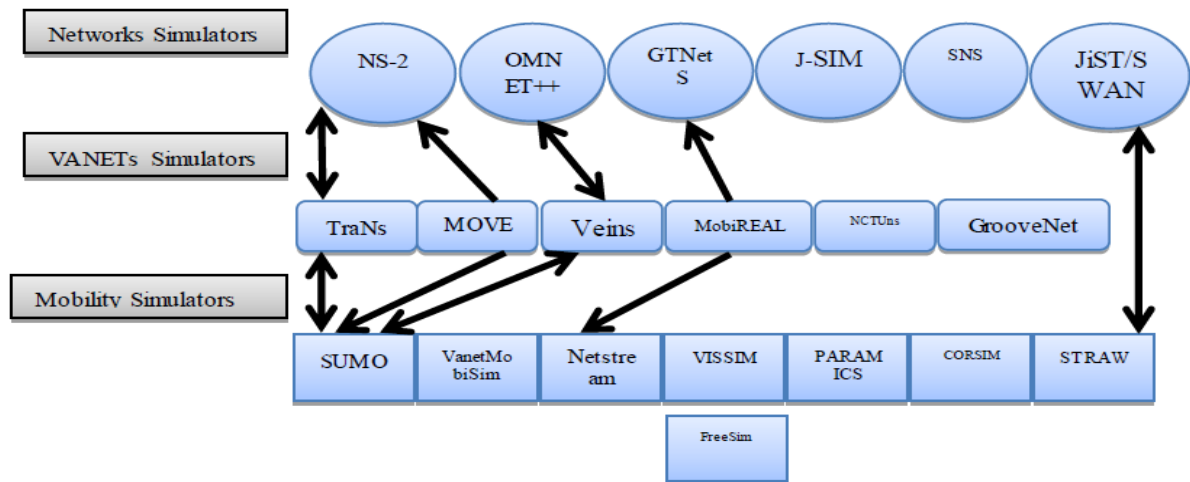


Fig 9. Clasificación de simulación de VANETs. (Noori, 2012)

1) *Generadores de movilidad:*

Suelen clasificarse en:

*Macroscópicas: Estas solo se centra en la movilidad del tráfico y el flujo del mismo y no individualmente. Aquí podemos encontrar los términos densidad de tránsito o de tráfico. (Noori, 2012)

*Microscópicas: son las que están enfocadas al estudio del movimiento individual de cada vehículo para estas es lo más importante. En este modelo los parámetros usados para la movilidad suelen ser mapas de carreteras, velocidad de los coches, salidas, limitación de tiempo...cosas en si más puntuales e individuales. La salida puede ser la coordenada principal del vehículo en cada tiempo y obviamente los parámetros relacionados con el movimiento del vehículo como lo son la aceleración o velocidad... (Noori, 2012)

2) *Simulador de redes.*

Este simulador de red se caracteriza por ser utilizado para la simulación de redes informáticas. Simulan los VANETs evaluando los protocolos de red para la movilidad de nodos y otra técnica requerida. (Noori, 2012)

Los MANETs necesitan extensiones de las VANETs, y son por ahora los más utilizados, como los generadores de movilidad vehicular. Hay ejemplos tales como: Jist / CISNES y GTNetS, omnet ++, NS-2 y NS-3. (Noori, 2012)



3) **Simulador VANETs.**

Dan una simulación del tráfico y de la red y también pueden ser combinados tráfico y red. Entre ellos nos encontramos con VEINS, MOVE, MobiREAL... (Noori, 2012)

❖ **Marco de trabajo de VEINS**

A) Simulador aceptable

En primer lugar se debe considerar el simulador de redes, los más populares son NS-2, GlomoSim y omnet ++. Respecto a los estándares de tecnología y los servicios prestados omnet ++ y NS-2 son softwares adecuados para la simulación de redes inalámbricas. De acuerdo con NS-3, Jist y omnet++ se pueden realizar simulaciones eficientes a gran escala. Pero siempre hay algunos inconvenientes como en el caso de MOVER que no puede crear una comunicación recíproca entre la red y el simulador de tráfico. (Noori, 2012)

Actualmente un marco de simulación con integración entre el simulador de tráfico microscópico DIVERT se presenta y luego se extiende. Un problema principal es que en la mayor parte de los simuladores integrados es que en el comportamiento de los vehículos no puede influir el simulador de red en el simulador de movilidad. (Noori, 2012)

En el marco de trabajo de VEINS su encuentran unidos SUMO y omnet++ pues la re-configuración en línea y re-enrutamiento de coches son características destacables de ello. Estos simuladores de red se pueden encontrar detalladamente en IEEE 802.11p e IEEE 1609.4 DSRC / capas de red WAVE, así se apoya el mapa realista y tráfico. VEINS permite conectar dos simuladores paralelamente conectados a través de un socket TCP. VEINS está desarrollado en base a MIXIM. OMNeT ++ que proporcionan que hacen que sea una muy buena simulación de redes pero tiene un inconveniente y es que se encuentra una deficiencia en el modelado de la comunicación inalámbrica. Se le añade SUMO que es un código abierto, es un paquete de simulación del tráfico rodado microscópico y continuo diseñado para manejar grandes redes de carreteras. (Noori, 2012)

B) Balizamiento

El concepto se basa en C2C, la emisión continua de todos los coches, lo que permite averiguar todos los coches vecinos en tiempo real. Dicha información se transmite como mensajes periódicos conocidos como balizas. Las balizas generan continuamente información sobre el vehículo y lo envían a los vehículos vecinos. Las balizas contienen información como: velocidad, aceleración, etc. Como se ha mencionado el protocolo IEEE 802.11p se utiliza para comunicaciones en el VANETs. IEEE 802.11p utiliza un protocolo de Media Access Control (MAC) que se basa en un protocolo de acceso



múltiple por detección de portadora con evitación de colisión (CSM + -A / CA). Las balizas envían información cada un intervalo fijo de tiempo en VANETs. (Noori, 2012)

❖ Escenario de simulación y resultado

Este apartado se basa en describir los pasos de una simulación realista de una ciudad con VANETs. En esta simulación se ha usado SUMO, omnet++y VEINS para simular el escenario. (Noori, 2012)

A) Mapa realista y Tráfico Realista

Los escenarios en simulador de SUMO están formados por dos partes: la Red Vial que son los mapas en el cual se incluyen calles, carreteras, semáforos, cruces etc. y la demanda de tráfico, lo que significa detalles de los coches que viajan como la velocidad coches, algunas propiedades físicas de cada vehículo, la dirección, de salida y de destino el tiempo y posiciones, etc. Para generar la red de carreteras de SUMO se puede hacer con una aplicación llamada “NetGen” que lo proporciona con el paquete SUMO también por la importación de un mapa digital de carreteras. Con respecto a la demanda de tráfico puede ser definida por diferentes fuentes. Se utilizan matrices para escenarios a gran escala que describen el movimiento entre las zonas de asignación de tráfico en el número de vehículos por tiempo. (Noori, 2012)

Este ejemplo usa los datos de (Uppoor & Fiore, 2011) para la red de carreteras y la demanda de tráfico. En (Uppoor & Fiore, 2011) la demanda realista para la ciudad de Colonia en Alemania con el mapa realista presentado. El conjunto de datos de coches se basa en basa principalmente en los datos realizados por el proyecto TAPAS-Colonia, este, proporcionado por el Instituto de Sistemas de Transporte en el Centro Aeroespacial Alemán (ITS-DLR), determina el tráfico de coches realistas en la ciudad de Colonia. Por medio de OpenStreetMap se importan la red de calles. El mapa de este ejemplo ocupa unos 400km² Estos dos datos se exportan a SUMO donde en la figura A vemos el mapa de Colonia y en la B se muestra el tráfico en los detalles. (Noori, 2012)

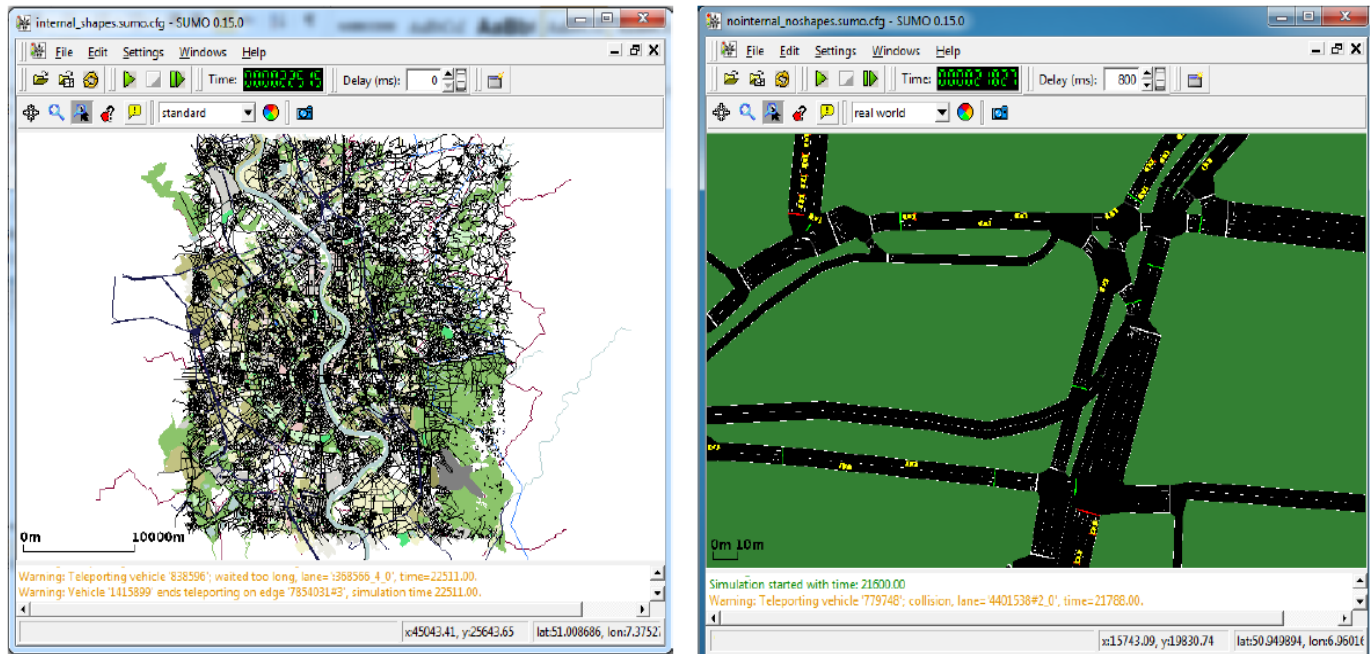
B) Configuración de Simulación de Red

Se prepara el tráfico realista y el mapa de SUMO. A continuación se configura el simulador de red y así comenzar con la simulación VANET. El marco de trabajo de VEINS se implementa en el 802.11p en omnet++con todos los parámetros correspondientes por defecto. Estos parámetros los veremos en la Tabla I. (Noori, 2012)

C) Resultado de la Simulación

VEINS conecta con Sumo y omnet++. Primero los coches se generan en SUMO y después son exportados al simulador en red omnet++ y este los considera todos como nodos y así el escenario es simulado. Si sufriera algún cambio den red, VEINS puede cambiar el escenario de coches en SUMO. (Noori, 2012)

En la figura 10 podremos ver cómo funciona VEINS con más detalle.



(a) Fig.10. Mapa de Colonia y el tráfico durante la simulación en SUMO (Noori, 2012) (b)

TABLA I

PARÁMETROS DE SIMULACIÓN DE RED

(Noori, 2012)

Parámetros	Valor
Tasa de balizamiento	10Hz
Tamaño de baliza	256bytes
Potencia de transmisión máxima	20mW
Umbral de atenuación de la señal mínimo	-89dBm
Tasa de bits canal	11Mbits ⁻¹

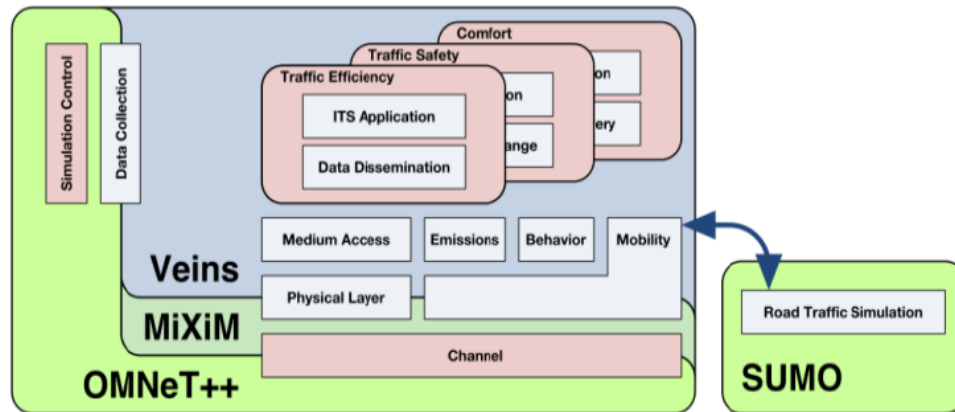


Fig.11. integración de SUMO y omnet ++ por VEINS (Noori, 2012)

El escenario de simulación se utiliza utilizando la configuración, el conjunto de datos y parámetros obtenidos en la sección anterior. Este escenario se ejecuta la simulación 12 veces desde cero. Cada 30 segundos se añaden a la hora de simulación hasta 360 segundos. En este escenario los vehículos mandan 10 balizas por segundo y todos los vecinos pueden recibirlas. En la salida de la simulación se obtienen datos de información de cada vehículo como los paquetes transmitidos, recibidos, paquetes perdidos, velocidad, posición y etc. (Noori, 2012)

Con el fin de analizar los resultados se seleccionan varios coches arbitrariamente. Con esto podremos calcular la probabilidad de balizas de entrega de cada vehículo mediante las estadísticas de salida. (Noori, 2012)

En la figura 4 se muestra el resultado para dos coches en la zona diferente frente a número de todos coches vecino durante la simulación. En el resultado se ve que la entrega de cada baliza en cada coche es distinta. Esto puede ser por el cambio de número de coches o densidad de los mismos, balizas recibidas en cada periodo... (Noori, 2012)

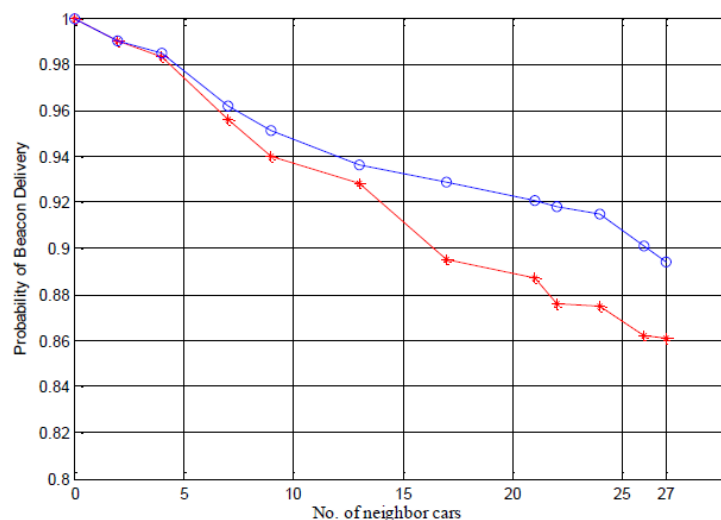


Fig.12. Probabilidad de baliza de entrega (Noori, 2012)



CAPÍTULO II: MÉTODO, SIMULADOR DE MOVILIDAD URBANA, SUMO

2.1 DESARROLLO DEL SOFTWARE Y ASPECTOS FUNDAMENTALES.

INTRODUCCIÓN

Sumo es un software de código abierto que simula un escenario de una ciudad, zona... Es el principal simulador de este trabajo fin de máster y sobre el cual nos basaremos para nuestra demostración.

Como ya se comentó anteriormente Sumo es un simulador para tráfico microscópico, multimodal con gran cantidad de aplicaciones, simula el comportamiento de los vehículos al hacer cambios de carril tanto en uno como en más carriles. Describe un modelo jerárquico de cuatro capas para estudiar el movimiento del vehículo durante la simulación y ver si se necesita una mejora (Barceló, 2010). La dinámica de la conducción microscópica de los vehículos tienen los siguientes modelos de interacción:

- Modelo del siguiente coche: halla la velocidad de un vehículo en relación con la vehículo delante de él.
- Modelo de Intersección: determina el comportamiento de los vehículos en los diferentes tipos de intersecciones en lo que se refiere a las normas de derecho de vía, la aceptación brecha y evitar obstrucción de unión.
- Lane-cambio de modelo: determina la elección de carril en las carreteras de varios carriles y la velocidad ajustes relacionados con el cambio de carril.

Sumo comienza a diseñarse en 2000 y en 2001 tuvo su primera aplicación. Sumo comenzó a desarrollarse con la colaboración de Zaik que es el Centro de Informática Aplicada de Colonia y el Instituto de de Sistemas de Transporte, ITS en el Centro Aeroespacial Alemán DLR. Desde 2004 solo se desarrolla en DLR con ayuda de alguna organización o personal cualificado. (Erdmann, 2014)

La licencia a la que está sometido Sumo es la Licencia Pública General GNU (GPL, 2009), tanto como código fuente y compilado, formato ejecutable para múltiples



Plataformas Windows y Linux. Como comentamos anteriormente la razón de que sea código abierto es doble pues por una parte había muchas tesis doctorales dedicadas a pequeñas simulaciones y muchas veces estaban incompletas. Las simulaciones no fueron nunca publicadas y estas desaparecieron. Los resultados de estas eran difíciles de comparar pues tenían distintos modelos y diversos detalles de implementación. (Barceló, 2010)

El Instituto de Sistemas de Transporte investigó para evaluar los diferentes modelos y algoritmos relacionados con el tráfico y la gestión del tráfico, la idea del fin era desarrollar una simulación de tráfico extensible y libre para que las simulaciones fueran más comparables y que diferentes personas puedan utilizarlo como banco de pruebas para sus aplicaciones. (Barceló, 2010)

Los principales criterios de desarrollo de Sumo son:

*Portabilidad: Es decir que la simulación fuera capaz de funcionar en cualquier entorno común ya que se usan diversos sistemas operativos y aparte de que Sumo empezó en el sistema operativo Solaris. (Barceló, 2010)

*Extensibilidad: Es decir ha de ser fácil para alguien que no esté relacionado se pueda adaptar fácilmente a él. (Barceló, 2010)

En un principio Sumo fue creado para la simulación de la ciudad de Colonia. Los datos principales para esta fueron tomados durante el proyecto "Zeitbudgeterhebung 1992" (Blanke et al., 1996). En el modelo de investigación puso como un primer ejemplo a una persona, la cual tiene unos deseos de movilidad bien si es por la mañana o por la tarde si quiere evitar un atasco... Para la aplicación de Sumo dentro de esta investigación tiene las siguientes características:

- *Memoria de huella pequeña:* Es decir, la simulación debe ser capaz de abarcar grandes áreas de la ciudad en trabajo estándar.
- *Velocidad de ejecución:* Realizar la simulación lo más rápido posible.

Éstas fueron las principales características pero aun siguen siendo importantes en los diseños de Sumo, son las que le dan forma el desarrollo. (Barceló, 2010)

La segunda es que Sumo se ha desarrollado continuamente durante trece años y solo el Instituto de Sistemas de Transporte o externos han colaborado o están trabajando. En este tiempo Sumo mejoro significativamente pero solo a lo largo de las características necesarias con lo que había como resultado durante los primeros años eran unas simulaciones de tráfico incompletas también porque algunas de las características esperadas no eran necesarias para la investigación actual. (Barceló, 2010)



La tercera influencia es que la mayoría de los usuarios no son científicos de tráfico sino científicos de computación. Hay razones para esto como lo que en un primer momento no se establece como simulador de tráfico, añadiendo que el usuario debe tener afinidad por los sistemas informáticos es decir, que sepa manejarse en línea sin gráficos o escribir programas y evaluar los resultados. (Barceló, 2010)

Aún así los criterios buscados se lograron y se pudo utilizar Sumo con éxito en varios proyectos en los últimos años. Los temas oscilaban en adaptaciones para semáforos para la revisión del tráfico en aéreas a gran escala y llevar un seguimiento. El tema de los semáforos también había sido investigado dentro de del proyecto “Orinoko” un proyecto realizado dentro de la iniciativa de investigación alemana “Verkehrsmanagement2010” (Gestión del tráfico 2010, 2009). En este, Sumo fue utilizado para la evaluación el nuevo plan del programa semanal de señal desarrollado por uno de los socios del proyecto. (Barceló, 2010)

En este plan se utiliza dentro de un área grande alrededor del recinto ferial de Nuremberg, en la zona “VLS”. Para predecir el desempeño del nuevo plan había que hacer una representación muy detallada del área y se incluía una red muy detallada y la representación de la demanda. La red se basó en utilizar datos de Navteq y se complemento con el número de calles, carriles, intersecciones, semáforos... Los datos de calibración fueron calculados por Validar. Sumo también fue extendido para simular planes de señal programa semanal, incluía la aplicación de dos métodos para cambiar entre los diferentes programas, de nombre SGP y estiramiento. (Barceló, 2010)

2.2 PRINCIPIOS DE CONSTRUCCIÓN EN SUMO.

Sumo no es una suite de software establecida, utiliza redes y demandas por lo general de otros paquetes de simulación. Esto causó que se dedicara mucho trabajo al desarrollo de importación de carretera y los datos de la demanda.

Se tomó la filosofía en lo largo del tiempo esto es, su principal idea es preparar las redes de carreteras o demandas para simular a partir de descripciones digitales y enriquecer con esto como datos de entrada a la simulación. (Barceló, 2010)



2.2.1. Preparación de una red de carreteras a simular.

En primer lugar se importa a Sumo una red de carreteras digital ya disponible. Para importar se utiliza una herramienta de Sumo que se llama “NetConvert” , esta lee formatos Visum, Tiger, archivos de ArcView, VisSim, carpetas Robocup Rescue League, Open Street Map y XML. Estas redes no están diseñadas para una simulación microscópica de tráfico. (Barceló, 2010)

En las redes de tráfico para estas representaciones se deben añadir posiciones de semáforos, carriles, normas de intersección. (Barceló, 2010)

Con NetConvert se calculan valores desaparecidos de la heurística. Algunas de las heurísticas se aplican solo si cierta información no es dada. Como puede ser el caso de los archivos ArcView, las conexiones de los bordes. (Barceló, 2010)

La heurística de NetConvert incluye:

- Cálculo de la dirección de giro (para cálculo de conexiones entre carriles)
- Cálculo de tipo de intersección.
- Cálculo de las normas de las vías que hay en una intersección.
- Cálculo de carretera y de intersección geométrica.

Con esta herramienta también se determinan posiciones de semáforos, carriles adicionales para carreteras, que rampas se han de añadir... Pero aún así debe hacerse una inspección manual redes. Ahora veamos el procedimiento para preparar la simulación de una red. (Barceló, 2010)

- ✓ Descripción de la red importándola y escribiendo en archivos XML/Sumo que describe la red de carreteras en nodos que son intersecciones, los bordes que son las carreteras y conexiones que hay entre carriles.
- ✓ Los archivos XML pueden utilizarse dentro de la simulación convirtiéndolos en una red de carreteras.
- ✓ Se inspecciona la red resultante.
- ✓ Si esta es errónea se corrige en los archivos XML, utilizados para su construcción.
- ✓ Iteración en el proceso para conseguir una buena calidad.
- ✓ Cuando se tiene una topología fija de la red, la información de los semáforos ha de ser mapeada y sustituir con NetConvert las definiciones reales. (Barceló, 2010)

Dentro de los proyectos más fuertes de Sumo, las redes son importadas desde Visum y Navteq. Algunas redes vienen importadas desde VisSim pero se ha puesto más empeño en importar desde Open Street Map, ya que son ejemplos del mundo real y legalmente hechos al público. Todas las demás instalaciones de importación fueron para facilitar su



uso pero sin ser probadas a fondo. Como es obvio esta afecta en algunos casos a no importar información requerida. También mencionar que el importador de redes Tiger, muy popular en comunicación vehículo a vehículo no es admitido. (Barceló, 2010)

Para elegir un método debemos ver el tema de investigación, la disponibilidad de datos sobre el tráfico de esa zona.

Sumo fue diseñado para simular una población sintética con personas individuales y horarios distintos.

El término “nativo” en Sumo se refiere a la lista de horarios de salida de los vehículos, sus viajes, el camino y donde empieza y terminará. (Barceló, 2010)

Esto se soluciona con un algoritmo que calcule el camino más corto creando una serie de rutas. Las rutas junto a la hora de inicio y alguna información adicional son las entradas posibles requeridas. Los sistemas basados en poblaciones sintéticas o agentes son capaces de entregar la hora de salida y el viaje origen de un vehículo. Para una demanda de un sitio en concreto es lento ya que para ello se deben de recopilar datos sociológicos, esto conlleva a que a los propios proyectos se recurra a unas descripciones comunes y establecidas por la demanda de tráfico por carretera. (Barceló, 2010)

Cuando se estudia ciudades a gran escala y completan son importadas matrices O/D de Visum u otros formatos dan buenos resultados. La importación consiste, desde Sumo suite:

- Convertir la matriz o matrices O/D en información de un vehículo y viaje que lleva cuando sale desde el origen y calle de destino.
- Con el fin de obtener un conjunto realista de rutas de carreteras se hace una asignación dinámica.

Por matrices O/D consisten en la demanda y llevan un periodo de tiempo de solo 1 hora. Para matrices de un día se permite aplicar una línea de tiempo que el usuario es quien lo define a base de porcentajes durante el día. (Barceló, 2010)

Las matrices O/D pueden variar mucho como las conexiones entre los distritos y la red de carreteras subyacentes. Estas matrices deben usar carreteras secundarias de la red como la conexión a los distritos ya que se pueden mapear el área y bordes de distritos y se pueden usar como orígenes y destinos. En cambio al referirse en conexiones adicionales estos no tienen una partida real en la red de carreteras, se deben señalar las principales intersecciones y controladas por semáforos. En esto se debe evitar los atributos ya que diverge de la realidad. (Barceló, 2010)

Para representar el tráfico en las autopistas el tráfico se usan router lo que hacen es leer en los puntos de inflexión en función del tiempo a proporción en las intersecciones y los flujos de tráfico. Los flujos se describen por su borde de partida y cantidad. (Barceló, 2010)



Con los datos proporcionados con esta herramienta se puede calcular la demanda por vehículo así como hacer una lista con las horas de salida, rutas, cada vehículo pasa de acuerdo a las relaciones dadas. Estos resultados son simples y directos por lo que hace fácil su evaluación. Cuando esta herramienta se usa en redes más grandes puede causar defectos ya que la herramienta no es capaz de resolver problemas con vías más complejas. (Barceló, 2010)

La función de la segunda herramienta utiliza flujos temporales de tráfico desde la ubicación de la observación. Esta herramienta marca dos tipos de puntos los puntos llamados “fuente” que son los que no tienen observación delante y los “sumidero” que son los que no tienen observación detrás. Después las corrientes de tráfico se propagan por la red situándose detrás de los puntos fuente hasta encontrar un punto sumidero. En las uniones cuando hay más de una continuidad posible, estas se distribuyen con relación a los puntos de observación medidos. Este principio es muy simple pero tiene la desventaja de que solo computan las distribuciones de flujos a través de la red, no las rutas reales de los vehículos. (Barceló, 2010)

El problema se presenta cuando se tiene la necesidad de tener todas las entradas y salidas cubiertas para la red, pero no suele darse este caso ya que este enfoque se utiliza solo para redes donde no hay bucles en la ruta de vehículos. (Barceló, 2010)

2.3 CALIBRACIÓN Y VALIDACIÓN.

La simulación de tráfico como ya sabemos es una aplicación informática por lo cual la validación se hace a diferentes niveles (Barceló, 2010). Sumo consta de los siguientes niveles:

- Pruebas unitarias

Estas son comprobaciones pequeñas y se comportan como lo debería haber hecho algún parámetro en un supuesto problema interno. (Barceló, 2010)

- Pruebas de Aceptación

Esta compara la salida con sus comandos de línea y archivos generados y la compara con una ejecución completa, si la salida actual y completa coinciden, el software devolverá un mensaje de todo correcto, por el contrario se informaría igualmente de



error. La suite Sumo se verifica todas las noches con más de dos mil pruebas de este tipo y unas ochocientas son las mismas pruebas. (Barceló, 2010)

La verificación de pruebas grandes se basa en incluir todas las pruebas pero entre muchos vehículos. Las pruebas son más sensibles a pequeños cambios de modelo y más complicados. Esto es debido a que los archivos que se generan van comparándose directamente unos con otros y la salida tiene un cambio menor de tiempo y crea error con lo que debe de devolver a realizarse, esto se puede evitar definiendo todo correctamente desde un principio. (Barceló, 2010)

Con lo cual concluimos que esta prueba es buena para asegurar un correcto funcionamiento pero no unas simulaciones complejas. (Barceló, 2010)

- Pruebas Modelo

Al lado de la comparación del diagrama de Krau del modelo de seguimiento de coche se comparó con modelos microscópicos de flujo de tráfico, el resultado daba la representación entre el flujo real y el conductor real. (Barceló, 2010)

En general no va a haber unos modelos de verificación que cubran eficazmente las aplicaciones que suite ofrece ya que hay infinitos números de ajustes, con lo que nos lleva a pensar que la verificación en un escenario es siempre necesaria. La verificación dependerá de los datos que se posean y estén disponibles. (Barceló, 2010)

2.4. TRABAJAR CON APLICACIONES EXTERNAS.

Nos encontramos con Traci que es una aplicación externa con la cual Sumo más se beneficia ya que esta hace que Sumo opte a más posibilidades de interactuar con una simulación en línea que se ejecuta mediante la conexión de una aplicación externa para Sumo utilizando sockets. Traci fue ejecutado por miembros de la universidad de Lübeck. En Traci cada paso tiene que ser llamado explícitamente a una aplicación externa, el resultado hace que la aplicación y simulación se ejecuten de manera síncrona. Con Traci se pueden simular objetos e incluir semáforos, vehículos, carreteras... El uso de carriles o funcionamiento de semáforos, cambiar velocidad de los coches, o frenado se pueden combinar con estas interfaz. (Barceló, 2010)

El principal objetivo de Traci es conectar a Sumo con la red de comunicaciones ns2 directamente o mediante aplicaciones como Trans que da la opción de definir aplicaciones V2V para simulación utilizando ns2 y Sumo. (Barceló, 2010)



2.5. PROYECTOS SELECCIONADOS.

Los dos siguientes proyectos son los dos más elaborados profundamente de DLR:

- *DELPHI*: es un portal accesible a las autoridades para la gestión propia a fuerza de reacción ante catástrofes o grandes eventos. (Barceló, 2010)
- *iTETRIS*: está fundada por la Comunidad Europea y tiene como objetivo evaluar las estrategias de gestión en V2V y V2I (Barceló, 2010)

CAPÍTULO III: DISCUSIÓN SOBRE UNA SIMULACIÓN EN LINARES.

3.1. ÁMBITO TERRITORIAL Y ESTADO ACTUAL DE LINARES.

INTRODUCCIÓN

El ámbito territorial en el que se desarrolla el trabajo es en el municipio de Linares, el cual es segunda ciudad de Jaén con 60.740 habitantes (INE 2013).

Se encuentra situada al noroeste de Jaén con coordenadas geográficas 38°06'00" Norte 3°38'00" Oeste y altitud media de 419 metros, entre el sector oriental de Sierra Morena y la parte alta de la depresión del río Guadalquivir, por lo que se pueden apreciar dos zonas bien diferenciadas: la sierra y la campiña. Su principal industria hasta finales del siglo XX fue la extracción minera la cual dejó ya de funcionar. (Wikipedia, 2015)

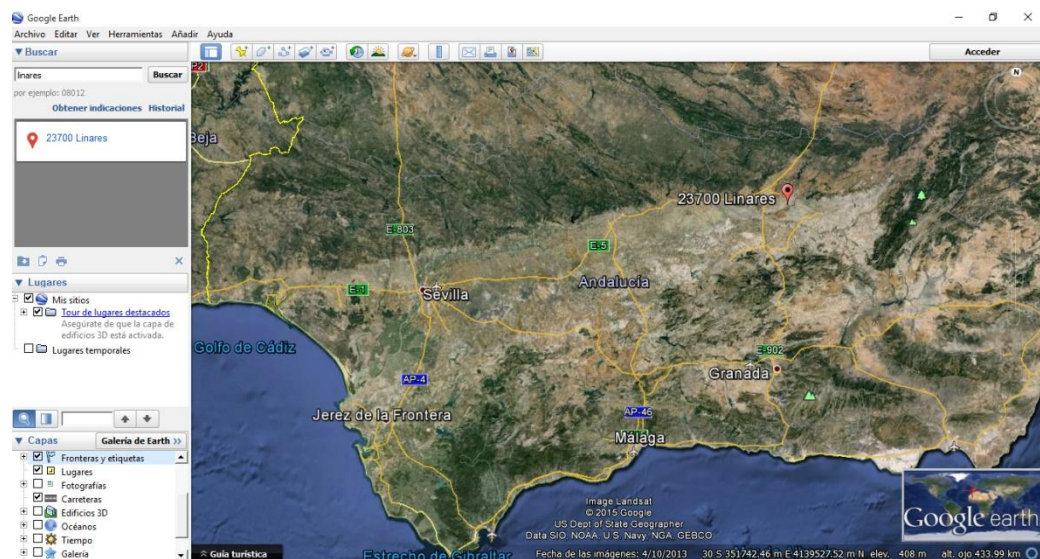


Fig 13. Situación de Linares en Andalucía. (Google Earth)

En la figura anterior se puede ver la situación de Linares en Andalucía, proporcionada con una captura de pantalla realizada al programa Google Earth. A continuación veremos el emplazamiento de Linares, obteniendo también la figura de la misma manera a la anterior.

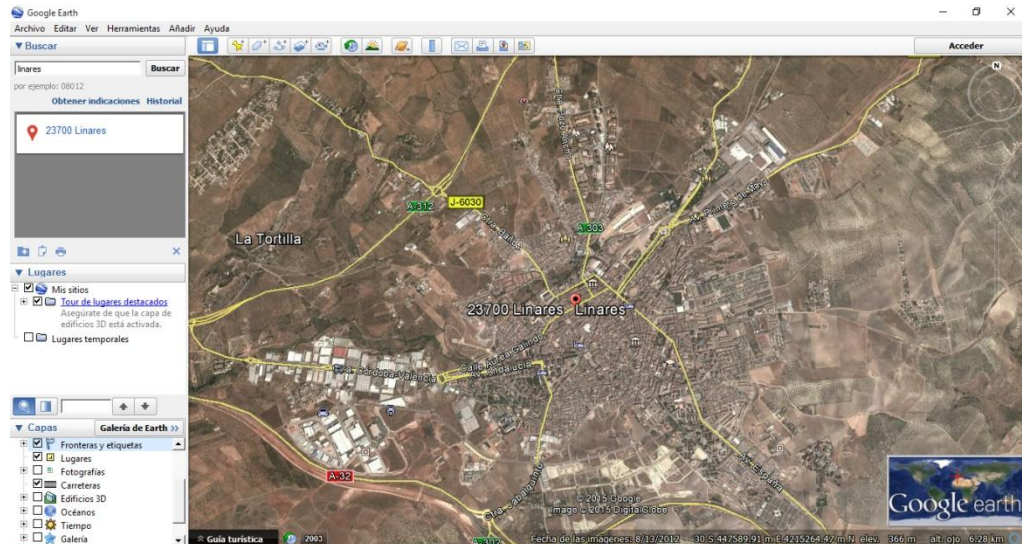


Fig 14. Emplazamiento de Linares. Google Earth

3.2. GEOGRAFÍA

Linares ocupa una extensión de 197,92 kilómetros cuadrados en la cual forman parte de esta extensión su seis pedanías, las cuales son: Estación Linares-Baeza, La Cruz, La Tortilla, Liñarejos, San Roque y Tobaruela. Es situado en un altiplano de desde la zona norte del término, el cual es perteneciente a Sierra Morena y desde aquí se abre hacia el valle del Guadalquivir. (Wikipedia, 2015)

3.3. OROGRAFÍA

Linares no presenta un terreno quebrado sino todo lo contrario, un terreno suave con poco desnivel presentando el pico más alto, fuera de la ciudad con una altitud de 552 metros, que es el Paño Pico (Wikipedia, 2015) y en la ciudad tiene una altitud media de 420 metros, la cual varía entre más-menos diez metros, en general para su extensión.



3.4. HISTORIA

Los primeros datos sobre Linares se remontan a antes de la Reconquista. También es conocido cerca de la ciudad una ciudad histórica, a cinco kilómetros del casco urbano conocido como Cástulo de época de los íberos y es uno de los yacimientos más importantes de Andalucía. (Wikipedia, 2015)

Respecto a la industria de Linares se basaba principalmente a explotar en minería su riqueza en plomo, con comienzo a mediados del siglo XIX. Con esta industria hace que la población aumente por seis obteniendo así el título de ciudad, otorgado a Linares por Alfonso XII. (Linares, 2009)

Con la llegada del ferrocarril se abrió un gran signo de progreso dotando a Linares de un hospital, casino, Caja propia de ahorros bajo el patronazgo de los Marqueses de Linares como también se abrió un asilo. Linares disfrutó con más de un siglo de duración con una sucursal del Banco de España. En esta ciudad es donde primero se creó la primera emisión radiofónica de la provincia, la primera escuela de capataces de minas y también una cámara de comercio. (Linares, 2009)

Las minas de Linares fueron cerrando y sus tranvías desapareciendo. Todo fue debido a que a mediados del siglo XX se comprueba que el plomo es malo para la fabricación de cañerías de agua lo que hace que su precio baje mucho en el mercado muy considerablemente con lo que finalmente se cierran todas las minas siendo la última en cerrarse la de la Cruz, en 1991. (Linares, 2009)

Pero la industria de Linares no quedó ahí, pues sobre los años cincuenta comienza la industria automovilística con la llegada de Santana, estuvo abierta hasta 2011, ahora reconvertida en un parque empresarial destinado al asentamiento y ejecución de ciertas empresas.

Por último, hoy en día la ciudad de Linares ofrece un valor añadido más junto a Cástulo, y es un complejo deportivo conocido como “La Garza” emplazado en un bosque de encinas, y es uno de los mayores de Andalucía. (Linares, 2009)

3.5. DEMOGRAFÍA

A continuación veremos unas tablas con la evolución demográfica de Linares desde 1900 hasta 2013 usando como fuente de información (Wikipedia, 2015) que usa los datos de la página del Instituto Nacional de Estadística, INE.



En el siglo XX:

1900	1910	1920	1930	1940	1950	1960	1970	1980	1990	2000
38,245	37,039	40,010	42,170	47,562	52,811	60,068	50,516	54,547	58,417	58,034

En los últimos trece años:

2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
57,796	57,800	58,257	59,096	60,807	61,452	61,262	61,340	61,338	61,306	61,116	60,950	60,740

Como conclusión a principios del siglo XX el crecimiento de Linares aumento moderadamente hasta que a mediados, entre 1950 y 1960 con el tema de la minería y Santana empezó a incrementar, posteriormente por el contrario descendió como consecuencia de empezar a cerrar las primeras minas.

En los últimos trece años ha tenido un crecimiento positivo pero en pequeña proporción aumentando unos 3000 habitantes durante estos trece años. (Wikipedia, 2015)

3.6. TRANSPORTES Y MEDIOS DE COMUNICACIÓN.

3.6.1. Red de carreteras del Estado.

Linares está comunicado con Bailén por el oeste y al este con Úbeda por medio de la autovía A-32. La conexión enlaza tanto con la autovía A-4 comunicando con el norte de España pasando por Madrid y al oeste hacia Sevilla como también la A-44 que comunica el sureste de Andalucía hacia Granada.

Linares también se comunica con los términos municipales por medio de carreteras autonómicas. También dota del cinturón sur y la rotonda oeste-norte. Por medio de la autovía A-32 se comunican Linares y Estación Linares-Baeza. (Wikipedia, 2015)

3.6.2. Red Ferroviaria



La estación de Linares ferroviaria es la estación Linares-Baeza y se encuentra a 6 kilómetros de la ciudad.

Desde comienzos del siglo XX Linares constaba de una extensa red de tranvías que comunicaba Linares con distintas localidades y las minas cercanas hasta los años sesenta. (Wikipedia, 2015)

3.6.3. Taxis

Linares contaba con cuarenta taxis en 2008 para toda la ciudad. Estos son blancos y en sus lados llevan la bandera y escudo de Linares. Consta de dos compañías de taxi Teletaxi Provincial y Radio Taxi Linares. (Wikipedia, 2015)

3.6.4. Carril-bici

A hoy día Linares cuenta con solo 220 metros de carril-bici y en construcción hay más kilómetros, son un total de 3 kilómetros cuyo objetivo es unir el cinturón sur con la futura rotonda norte.

3.7. RUTAS DE AUTOBUSES URBANOS.

3.7.1. Líneas de urbanos

Los urbanos de Linares están gestionados por la empresa Transporte urbanos de Linares S.L. y constan de nueve vehículos.

Los autobuses urbanos están dotados de sistema electrónico multiplex, SAE, canceladoras por tarjeta sin contacto por poste con indicación del tiempo restante para el paso por parada del siguiente autobús, etc. (Wikipedia, 2015)

Las líneas regulares son las siguientes: (Tulinares, 2006)

- Línea 1: Estación - Hospital San Agustín, Residencial - Estación

- Línea 2: Estación - Polígono Industrial - Estación

- Línea 3: Bda. Santa Ana - Hospital San Agustín, Residencial – Bda. Santa Ana

- Línea 4: Villalonga - Hospital San Agustín, Residencial - Villalonga

- Línea 6: Villalonga - Geriátrico - Villalonga

- Línea 7: Villalonga - Estación - Villalonga

- Línea 8: Centro - La Garza - Centro (solo operativa entre el inicio de verano y el 27 de agosto)

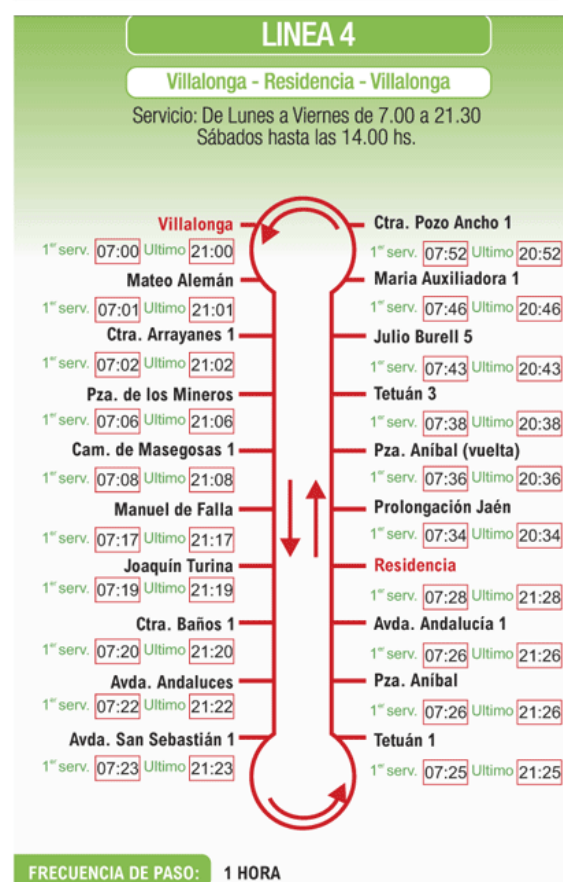
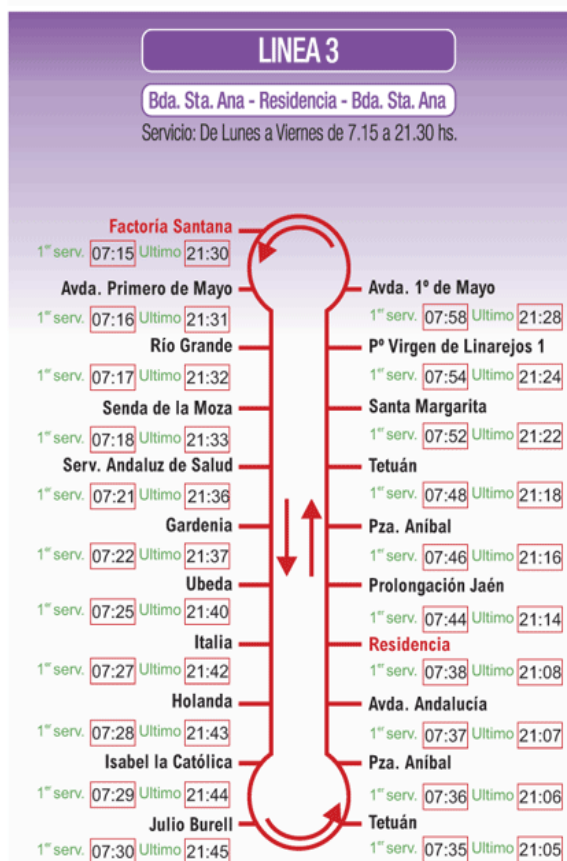
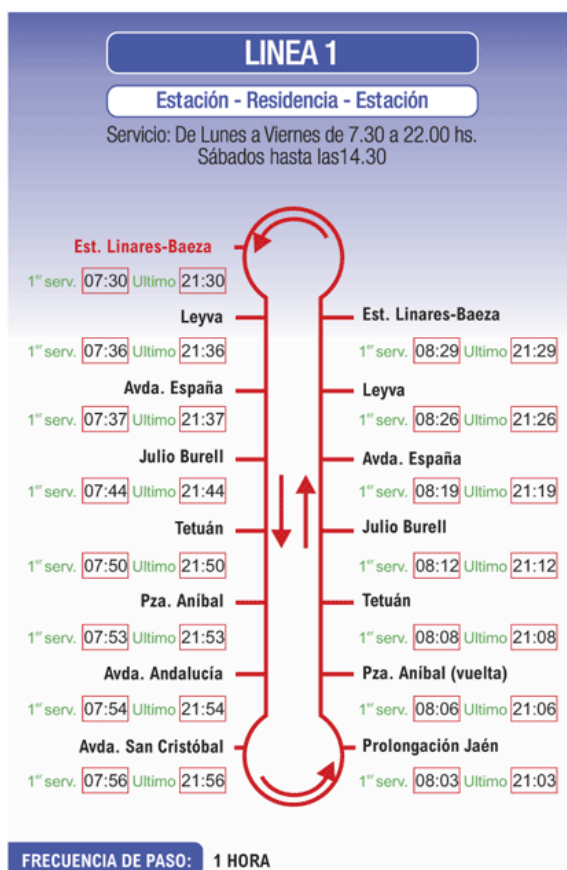
(Wikipedia, 2015) (Tulinares, 2006)

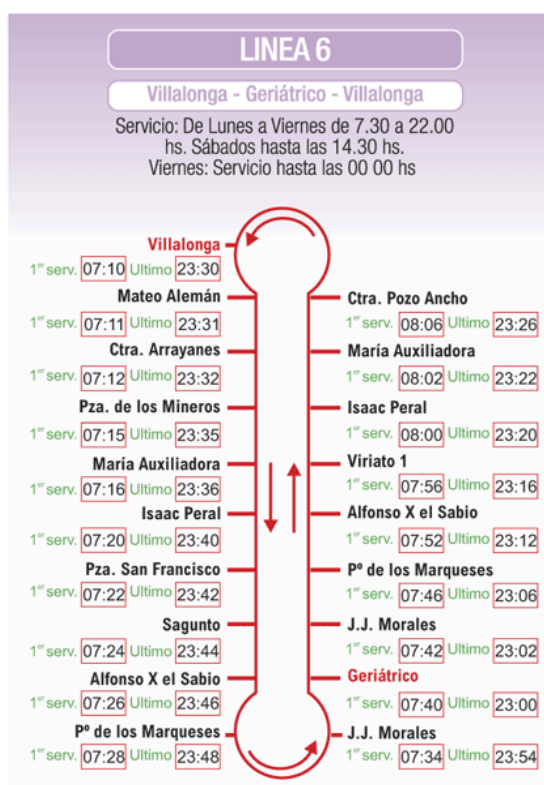


Fig. 14 Página oficial de urbanos de Linares. (Tulinares, 2006)

3.7.2. Rutas de urbanos

A continuación se presentan las imágenes con los trayectos de las líneas de urbanos de Linares, obtenidas desde la página oficial del ayuntamiento de Linares.





(Tulinares, 2006)



(Wikipedia, 2015)



3.8. SIMULACIÓN DE URBANOS CON SUMO.

PRESENTACIÓN Y DATOS DE PARTIDA.

3.8.1. Presentación

Este trabajo tiene como objetivo realizar una simulación de autobuses urbanos con unas rutas alternativas, con su posterior simulación en el simulador Sumo.

3.8.2. Datos de partida.

- Semáforos.

En primer lugar se obtuvieron los semáforos de la ciudad. Estos datos fueron proporcionados por el subinspector de policía local de Linares, quien facilitó los cruces con los semáforos, que es donde principalmente se encuentra y algunos otros, en avenidas.

La situación de los semáforos será posteriormente relacionada con los cruces y coordenadas del software, por lo que esto son simplemente datos de partida para la orientación de la situación de los mismos.

A continuación se muestra el plano escaneado pues los datos fueron facilitados en formato papel, en concreto en un formato A3.

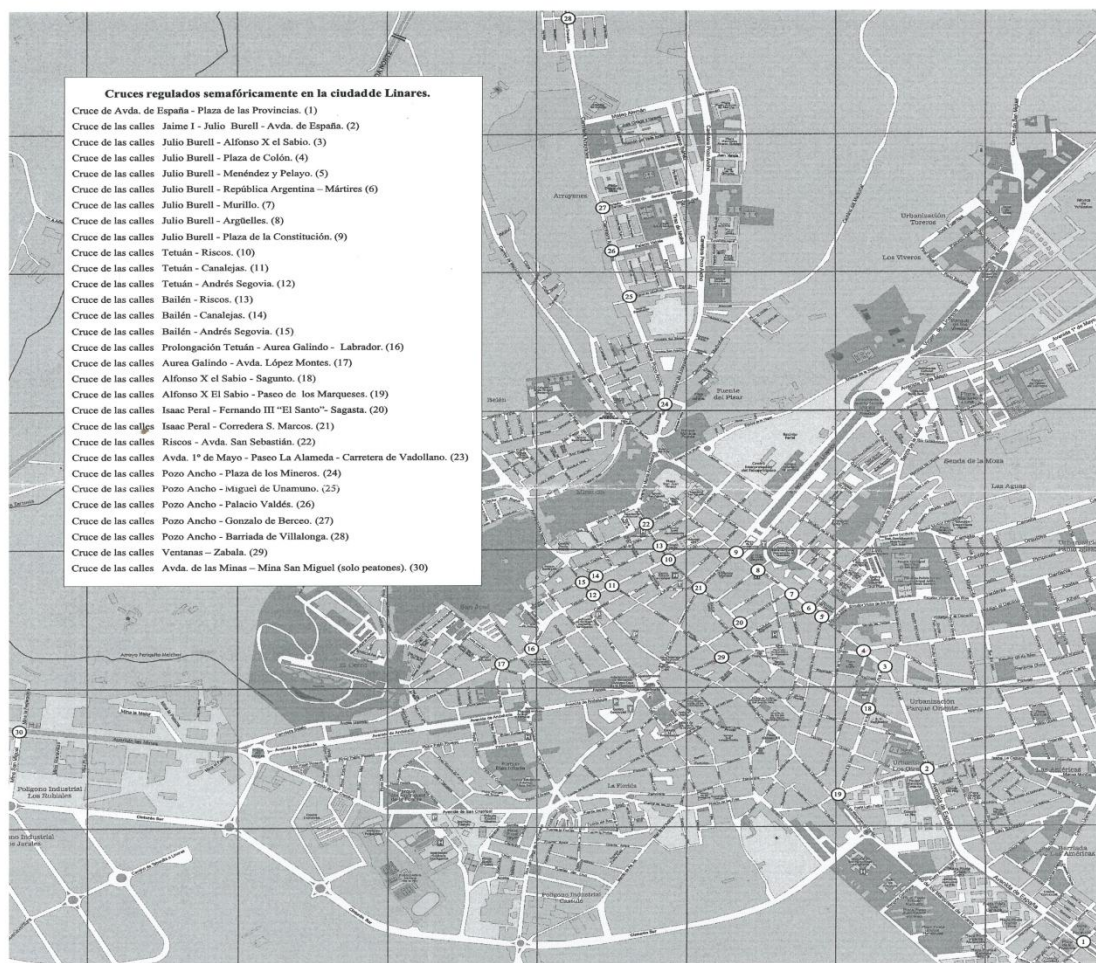


Fig. 15 Mapa con los principales cruces y sus semáforos.

Una vez localizados los semáforos se procede a la localización por coordenadas UTM (Universal Transversal Mercator). Para obtener las coordenadas se realizó su localización con el programa Google Earth, el cual te las proporciona tanto en UTM como en coordenadas geográficas, pero para este caso y su posterior tratamiento de datos se optó por UTM que nos facilita el trabajo.

El siguiente paso con los semáforos fue pasar las localizaciones a Excel y tratarlas en MapInfo para pasar las coordenadas a formato shape, es uno de los formatos con los que trabaja Sumo.

- Escenario, campo de trabajo.

El escenario de trabajo se obtuvo con un formato “.osm” ofrecido por el programa Open Street Maps que es “es un proyecto colaborativo para crear mapas libres y editables”. (Namdre, 2015)



Fig. 16 Página oficial de Open Street Map

Veamos los pasos a seguir para obtener el archivo .osm

- ✓ Primero se busca la zona de trabajo:

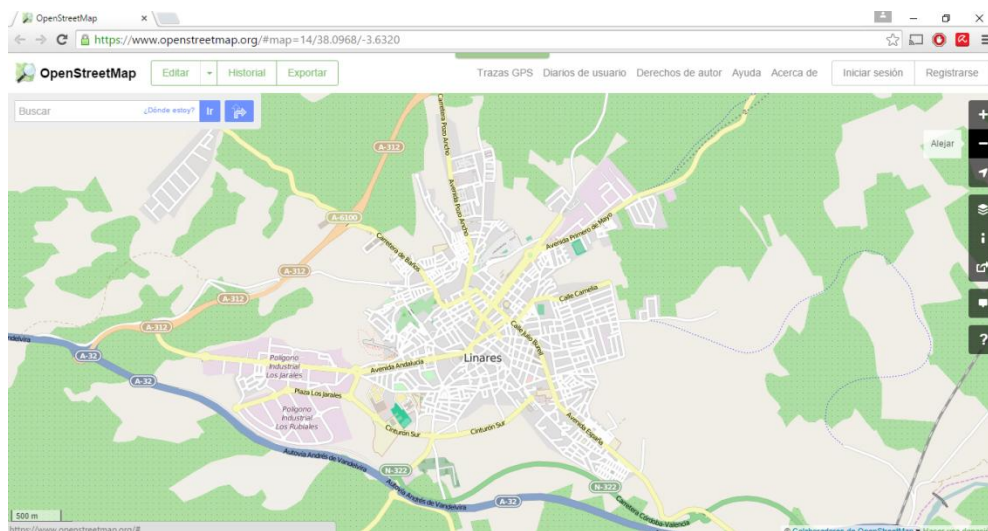


Fig. 17 Localización de Linares en OpenStreetMap

- ✓ Segundo, exportamos la imagen

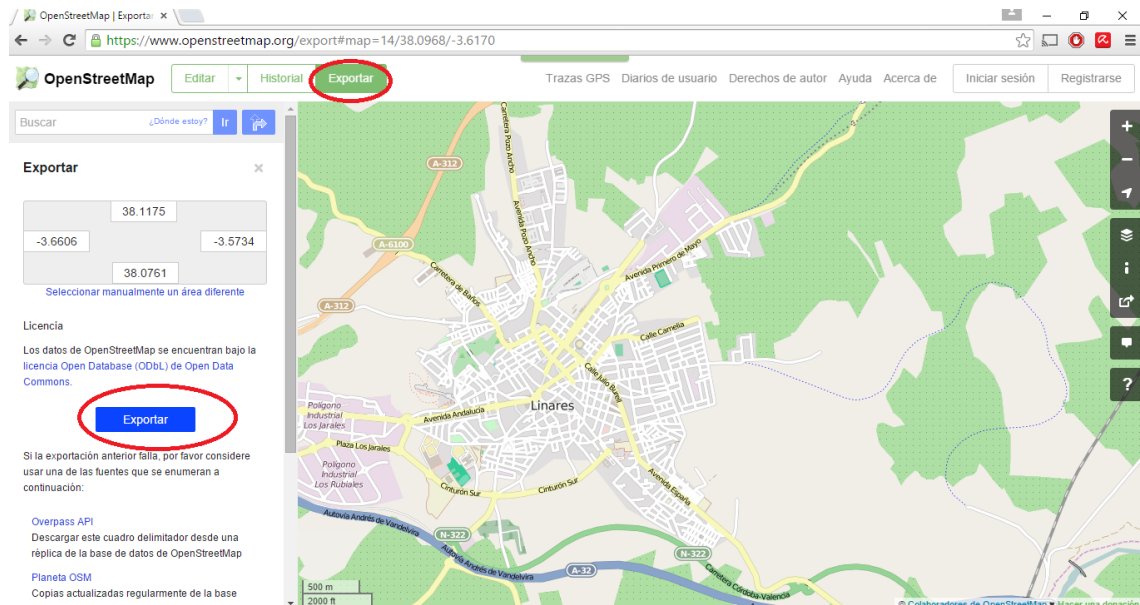


Fig.18 Como exportar fichero .osm

Así obtendremos nuestro fichero “.osm”, que por defecto se llamará “map.osm”.

- Rutas

En nuestro caso no estableceremos unas rutas determinadas, lo que se hará es crear una simulación con un comando de rutas alternativas que nos proporciona Sumo dentro de sus herramientas, la cual es randomTrpis.py y la pondremos en Duarouter para que haga rutas alternativas bidireccionales.

- Tipos de vehículos.

Los tipos de vehículos que se van a definir son los principales en una ciudad, es decir:

- ✓ Vehículos de emergencia
- ✓ Autoridad
- ✓ Privados
- ✓ Taxis
- ✓ Autobuses urbanos.

Todos ellos creados en formato .xml y posteriormente tratados en Sumo.

3.9. PROCESAMIENTO Y GESTIÓN DE DATOS.

En este apartado se explica detalladamente cómo se realizó la simulación paso por paso y las herramientas y extensiones necesarias para su creación.

3.9.1. Obtención de escenario de actuación.

En primer lugar, como se ha dicho anteriormente, obtenemos nuestro escenario elegido descargándolo de OpenStreetMap. La extensión de este archivo es .osm que por defecto se descargará como map.osm.

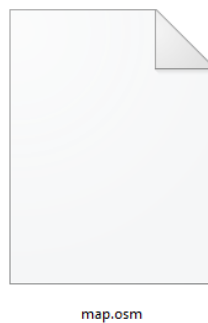


Fig.19 Imagen de archivo exportado de OpenStreetMap

Posteriormente lo tratamos en el software de comandos de Sumo, “start-command-line”, pues Sumo trabaja desde MS DOS con líneas de comando. Se usan sus herramientas para tratar los archivos y posteriormente se visualizan de Sumo-GUI el cual se explicará más adelante.

La herramienta que usamos en start-command-line es NetConvert, que procesa este archivo y lo convierte en un archivo “net.xml” para que Sumo pueda procesarlo para sus distintas funciones.

Exactamente cuándo ejecutamos “start-command-line” debemos dirigirnos al directorio donde se encuentra nuestro archivo y desde ahí ejecutar lo deseado. Vean un ejemplo en la imagen de a continuación.

```
C:\WINDOWS\system32\cmd.exe
info: added location of sumo executables to the search path
info: variables SUMO_HOME, SUMO_BINDIR are set

use the 'cd /d' command to change directory
example usage:
cd /d c:\foo\bar

C:\>cd users

C:\Users>cd ana

C:\Users\Ana>cd desktop

C:\Users\Ana\Desktop>cd sumo

C:\Users\Ana\Desktop\sumo>cd sumo programa

C:\Users\Ana\Desktop\sumo\sumo programa>cd sumo-0.24.0

C:\Users\Ana\Desktop\sumo\sumo programa\sumo-0.24.0>_
```

Fig.20 Directorio donde se encuentran los archivos a tratar

Cuando se está en el directorio se continúa escribiendo el comando/herramienta a utilizar, ahora Netconvert.

El comando finalmente quedaría escrito definitivamente para realizar la conversión de la siguiente manera:

“netconvert -- osm-files map.osm”

Con esto obtenemos un fichero net.net.xml en el cual tenemos toda la información del escenario descargado de OpenStreetMap.



Fig.21 Imagen de archivo obtenido con Netconvert

Para poder ver toda la información y obtener la necesaria o específica de este fichero se abrirá bien con un programa de texto simple, como Bloc de notas o con un programa específico de .xml en este caso se usó “AltovaXMLSpy 2016”.

Al abrirlo obtenemos gran cantidad de información como puede ser desde donde se encuentran las intersecciones, las direcciones de calles, los polígonos que forman la

ciudad o incluso semáforos, aunque en este caso, los semáforos no vienen como información. Vean a continuación como es abierto este archivo con “AltovaXMLSpy 2016”.

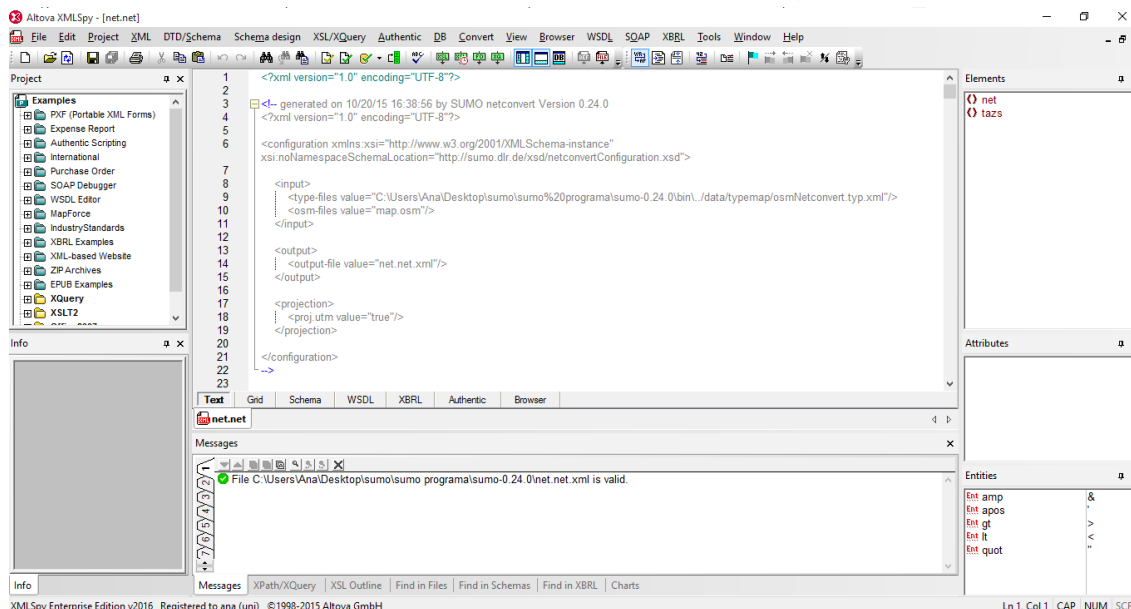


Fig.22 Parte programada al realizar Netconvert

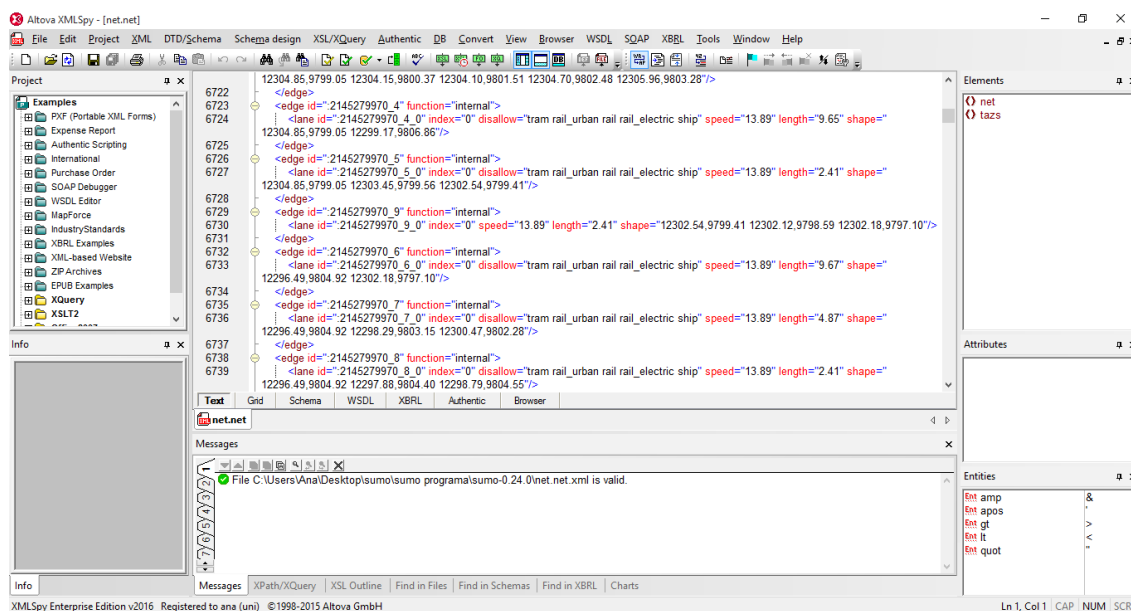


Fig.23 Definición de bordes

En la imagen 22 se observa cómo se definen los bordes (edge) de las líneas definidas en el mapa, como se puede observar definen en esta parte en concreto, que es un rail

ferroviario, como también especifica por ejemplo la velocidad a la que se circula, longitud...

3.9.2. Añadir polígonos, mapa de características.

Para realizar este paso lo que se realiza es lo siguiente:

Entramos en el archivo net.net.xml donde se hallan todos los datos obtenidos desde OpenStreetMap y copiamos la información de los polígonos y pegamos en el programa editor de .xml para guardarlo con esa extensión y así Sumo pueda realizar su procesamiento.

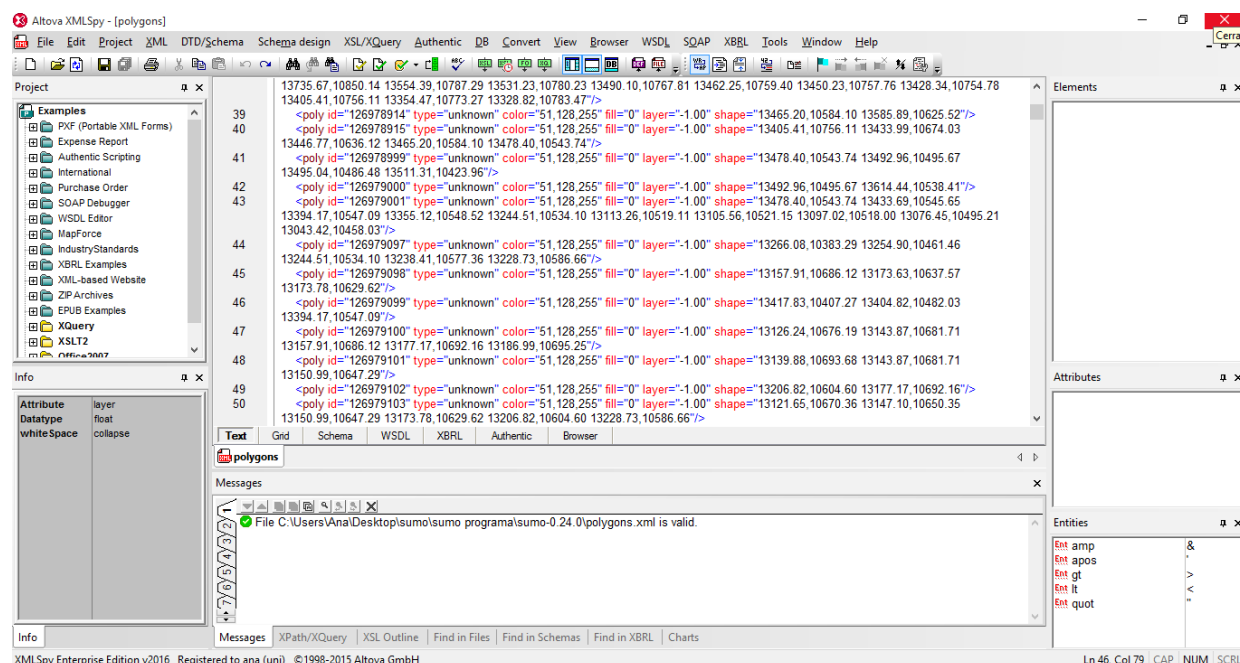


Fig.24 Polígonos definidos por OpenStreetMap.

Como vemos aquí se definen los polígonos, pero en este caso, para Linares no se obtiene una gran cantidad de información definida de los polígonos por lo que serán más bien homogéneos y todos de igual color.

Una vez obtenido el archivo .xml lo llevamos a start-command-line y continuamos su procesamiento con una nueva herramienta, “Polyconvert” la cual trabaja y se ejecuta como se muestra a continuación:

```
<SUMO_HOME>polyconvert --net-files net.net.xml --osm-files map.osm --type-file
polygons.xml - o polygonos.poly.xml
```

Para definir la salida pondremos “-o” (output) como queremos que se llame el archivo de salida y su extensión, la cual ahora es .poly.xml

Estos polígonos serán mostrados en Sumo-GUI.

3.9.3. Creación de rutas

Para crear rutas en este caso se ha utilizado la herramienta “randomTrips.py” que crea rutas alternativas según sus datos de direcciones de portados del archivo net.net.xml.

Esta herramienta para poder procesar en start-command-line necesita otro programa, Python pues son herramientas con extensión .py. Para que start-command-line ejecute la herramienta de rutas llama a Python y deben sincronizarse a la vez por lo que se debe instalar en nuestro ordenador. Una vez instalado hay que hacer un ajuste en la configuración del sistema para que se puedan sincronizar y Sumo haga la llamada correctamente.

Veamos ahora como se escribe esta herramienta para procesar posteriormente:

```
<SUMO_HOME>python \tools\randomtrips.py -net.net.xml -r rutas.rou.xml -e 50 -l
```

Con esto se crean tres archivos más, trips.trips.xml, rutas.alt.xml y rutas.rou.xml. En el comando de ejecución, “50” es un ejemplo del número de rutas que queremos en el escenario y “-r” es el termino para indicar Duarouter, rutas bidireccionales y también es para indicar que es el archivo de salida.

Se muestra a continuación como queda cuando se abre el archivo en formato .xml

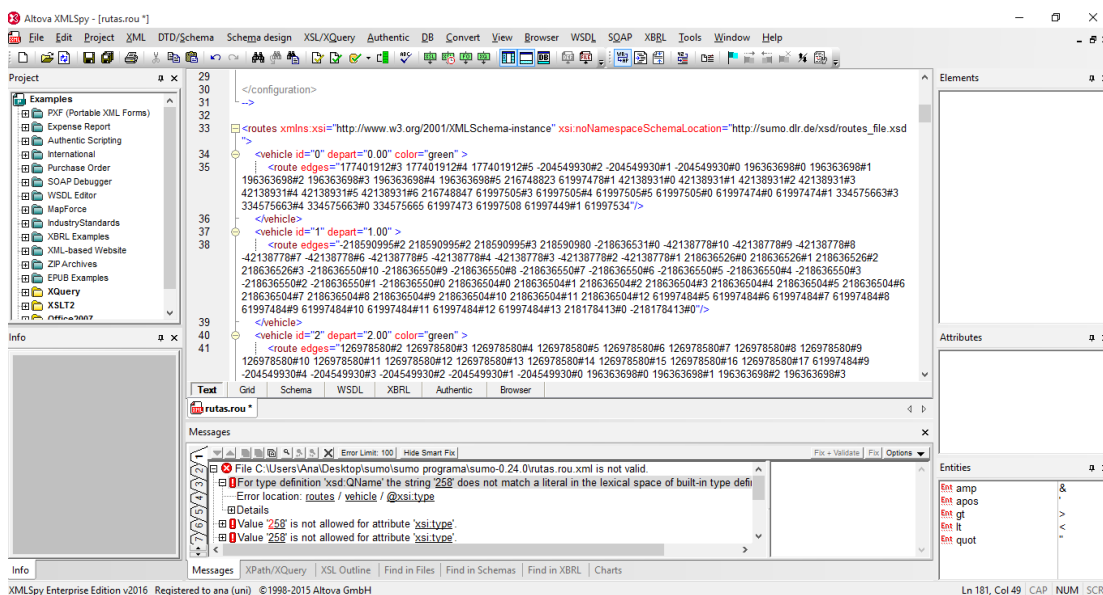


Fig.25 Fichero .rou.xml con rutas y vehículos.

Al igual que en la definición de bordes por ejemplo, se podrán añadir atributos como color, tipo de vehículo peatones etc.

3.9.4. Creación de semáforos

Los semáforos se pueden crear de diversas maneras y con distintas herramientas veámoslo a continuación.

- Una manera de hacerlo es localizando las intersecciones o “junction” en Sumo-GUI o net.net.xml que se sepan que tienen semáforos y definirlos como tal. En este trabajo el intento es así, pues que con los datos de partida en coordenadas UTM, los semáforos no los podemos meter pues no se relacionan ambos datos de posición, por un distinto sistema de posicionamiento para ambos.

Al acceder al net.net.xml se cambia el “type” de la intersección o “junction” y lo ponemos como “traffic_light”. Pero hay que tener en cuenta que no todos los semáforos se pueden poner donde se quiera pues net.net.xml tiene en escrita su lógica la cual si la cambias puedes crear un error en ella y no compilara el simulador y deberemos de restablecer como se encontraba a principio para que tenga su lógica.

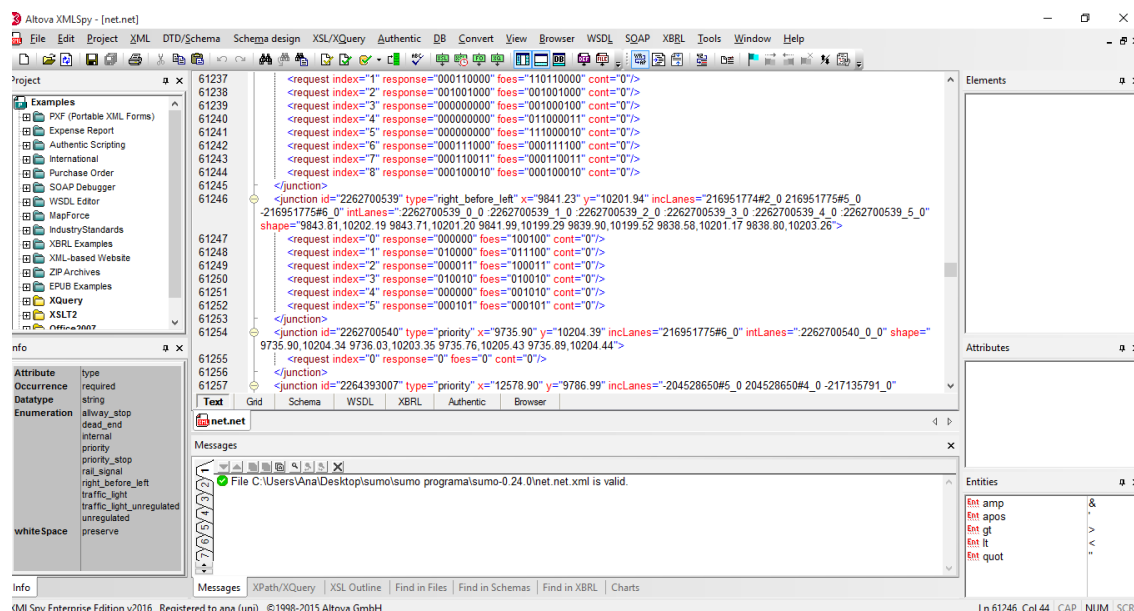


Fig.26 Programación de junction y sus atributos.

Como se puede observar vemos como se define y la secuencia que lleva. En la esquina inferior izquierda en el recuadro gris se puede observar el tipo de atributo seleccionado, en este caso “type” y la variable que se le puede asignar según función deseada.



- Con las herramientas de Sumo también se pueden definir los semáforos. Las herramientas para ello son “tls_cvs2sumo.py” y “tls_check.py”.

✓ Funcionamiento de “tls_cvs2sumo.py”:

Esta herramienta convierte una descripción tls_csv dentro de Sumo, para que pueda ser leída como un archivo adicional. La estructura de este archivo es la siguiente y se encontrará al principio. [www004]

key; <KEY>

subkey; <subclave>

offset; <OFFSET>

Finalmente la descripción de tls-csv se verá:

key;105

subkey;utopia

offset;0

(link;<LINK_NUMBER>;<FROMEDGE>|<FROMLANE>;[<TOEDGE>|<TOLANE>];0)+

(param;<NAME>;<VALUE>)+

(<LINK_NUMBER>;<PHASE_LIGHT>[;<PHASE_LIGHT>]+)+

(min;<PHASE_LENGTH>[;<PHASE_LENGTH>]+)*

time;<PHASE_LENGTH>[;<PHASE_LENGTH>]+

(max;<PHASE_LENGTH>[;<PHASE_LENGTH>]+)*

(Namdre, DLR, 2014)

Después se define que señal en el programa es responsable para cada enlace. El enlace se puede describir ya sea usando o el carril de entrada solamente, o también incorpora el saliente del borde / carril. (Namdre, DLR, 2009) es decir que los enlaces correctos para describir serían de la siguiente estructura (Namdre, DLR, 2009)

link;<LINK_NUMBER>;<FROM_EDGE>;;0

link;<LINK_NUMBER>;<FROM_LANE>;;0



link;<LINK_NUMBER>;<FROM_EDGE>;<TO_EDGE>;0

link;<LINK_NUMBER>;<FROM_EDGE>;<TO_LANE>;0

link;<LINK_NUMBER>;<FROM_LANE>;<TO_EDGE>;0

link;<LINK_NUMBER>;<FROM_LANE>;<TO_LANE>;0

Esta configuración de enlace “link” ha sido proporcionada desde (Namdre, DLR, 2009)

Es posible dar también más de un estado a una sola señal viene dado por:

<LINK_NUMBER>;<STATES>

Los estados de los semáforos se codifican por colores asignando ‘g’ ‘r’ ‘y’ y ha de hacerse en minúscula. Un ejemplo será:

1;g;g;g;y;r;r;r;r (Namdre, DLR, 2009)

Después se definen las fases de tiempos:

time;<TIMES>

Los tiempos vienen dados en segundos y separados por punto y coma. Vean un ejemplo:

time;18;33;3;6;3;3;9;15;90

(Namdre, DLR, 2009)

Finalmente veamos un ejemplo de cómo sería una descripción CSV:

key;102

subkey;utopia

offset;0

link;1;4643;;0

link;1;3078;;0

link;2;3074;;0

link;2;-6494;;0

1;g;g;y;r;r;r;r

2;r;r;r;r;g;g;y;r

3;r;r;r;r;g;y;y;r

4;g;y;r;r;r;r;r

min;21;3;3;2;20;9;3;2

time;45;3;3;2;36;9;3;2

max;78;3;3;2;62;9;3;2

Estructura de csv (Namdre, DLR, 2009)

Finalmente se convertirá utilizando:

tools/tls/tls_csv2SUMO.py lsa_def1.csv,lsa_def2.csv input_net.net.xml [www005]

3.9.5. Creación e inserción de vehículos

Para poder insertar vehículos primero debemos crear las rutas en nuestro escenario como se explicó en el apartado 4.2.3. Una vez obtenidas nuestras rutas podremos cambiar solo el color en este archivo “rutas.rou.xml”, refiriéndome a “solo” porque también nos interesa cambiar el tipo de vehículo que es, aunque también se lo pueden añadir otros atributos más.

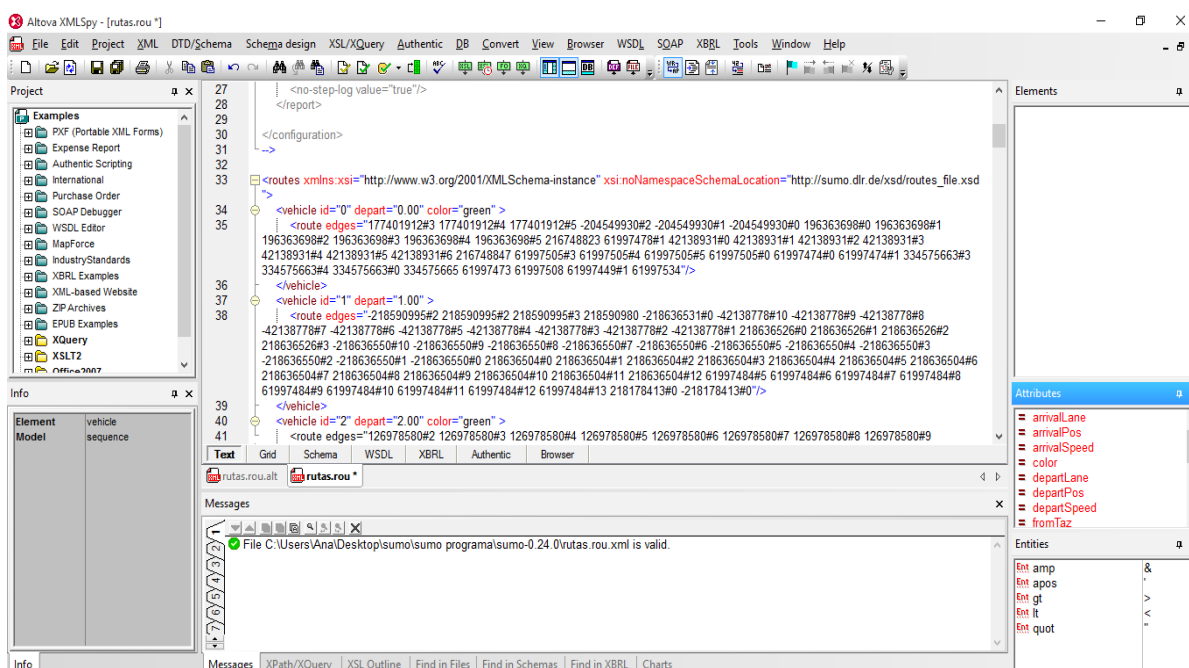


Fig.27 Atributos de rutas.rou para los coches

Como se puede ver a la derecha, en azul, están señalados los atributos que se le pueden añadir a las rutas y vehículos. Cualquier atributo que no se encuentre ahí no se podrá forzar a ponerlo y habría que crear otro fichero rutas.rou.xml para poder añadir el atributo vClass que nos permite definir la clase deseada de vehículo.

Para definir bien accedemos a un archivo llamado “hello.rou” que nos lo proporciona Sumo como plantilla editable para poder usarla para editar vehículos y rutas y ya desde el poder añadir como atributo vClass y definir el tipo de vehículo.

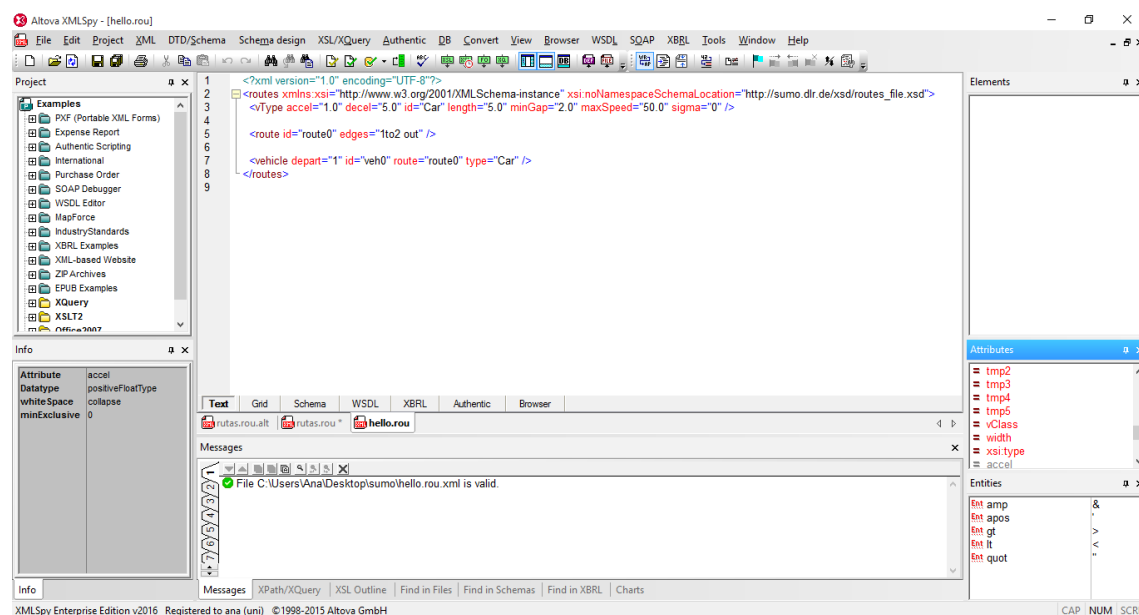


Fig.28 Archivo hello.rou.xml editable con atributo vClass

Ahora cómo podemos observar en los atributos podemos poner la clase de vehículo. Veamos cómo debemos formarlo.

Al tener las rutas creadas de anteriormente lo que se debe hacer es pasarlas y adaptarlas según la secuencia del formato, al archivo hello.rou, proporcionado por Sumo al instalar el programa.

Al cambiar el archivo de rutas no debemos olvidar que para que el Simulador-GUI funcione correctamente. Para ello debemos de cargar como rutas el nuevo archivo creado (hello.rou.xml) sino hacemos el cambio de archivo cargado para simulación seguirá mostrando el antiguo.

A continuación se muestra una tabla con los distintos vClass de vehículos.



vClass	bitmask integer	comment
ignoring	0	may drive on all lanes regardless of set permissions. This is the default class.
private	1	
emergency	2	
authority	4	
army	8	
vip	16	
passenger	32	
hov	64	
taxi	128	
bus	256	urban line traffic
coach	512	overland transport
delivery	1024	
truck	2048	
trailer	4096	truck with trailer
tram	8192	
rail_urban	16384	heavier than 'tram' but distinct from 'rail'. Encompasses Light Rail and S-Bahn
rail	32768	heavy rail
rail_electric	65536	
motorcycle	131072	
moped	262144	motorized 2-wheeler which may not drive on motorways
bicycle	524288	
pedestrian	1048576	lanes which only allow this class are considered to be 'sidewalks' in NETCONVERT
evEHICLE	2097152	future mobility concepts such as electric vehicles which may get special access rights
ship	4194304	basic class for navigating waterways
custom1	8388608	reserved for user-defined semantics
custom2	16777216	reserved for user-defined semantics

Fig.29 Tipos de vClass (Namdre, DLR, 2009)

Para una visualización más agradable del tráfico, la aparición de los vehículos de un tipo de vehículo puede ser cambiado asignándoles una forma determinada mediante el atributo guiShape. Estas formas se utilizan cuando se ajusta el modo de dibujo para los vehículos a formas simples. (Namdre, DLR, 2009)

Formas conocidas:

pedestrian	passenger/wagon	bus	rail/city
bicycle	passenger/van	bus/city	rail/slow
motorcycle	delivery	bus/flexible"	rail/fast
passenger	truck	bus/overland"	rail/cargo
passenger/sedan	truck/semitrailer	rail	vehicle
passenger/hatchback	truck/trailer	rail/light	ship

(Namdre, DLR, 2009)

3.10. Visualización en Sumo-GUI

Por último queda realizar la visualización de nuestro trabajo realizado para ello debemos tener instalado Sumo-GUI que es el software de visualización de Sumo.

En primer lugar debemos tener localizados nuestros archivos net.net.xml, rutas.rou.xml (o hello.rou.xml. si hemos cambiado tipos de vehículos y demás) y polígonos.poly.xml.

Después buscamos el archivo “sumo.cfg” que se encuentra en la carpeta donde se instala el programa. Este también es un archivo editable en .xml, abrimos y cargamos nuestros archivos, veámoslo en la figura siguiente.

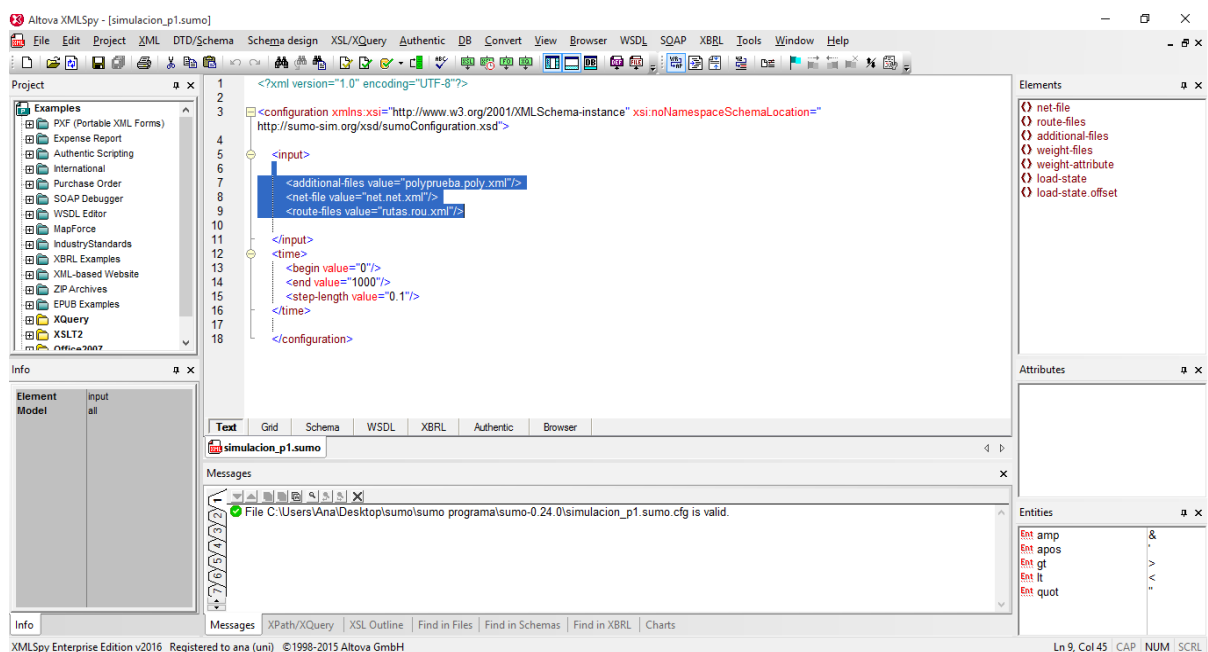


Fig.30 Archivo editable sumo.cfg para visualización

En azul se observan los archivos cargados para su representación. Una vez editado lo guardamos para su posterior ejecución y visualización.

Para visualizar lo podemos hacer desde:

- Start-command-line : Iniciamos y nos dirigimos hasta donde se encuentran el archivo simulación_p1.cfg y seguidamente escribimos:

Sumo-gui -c simulación_p1.cfg + ENTER

Y si todo está realmente correcto Sumo-GUI se inicia sin fallo y vemos la simulación.



- Desde el propio archivo: Simplemente buscamos el archivo simulación_p1.cfg y lo abrimos con Sumo-GUI.
- Sumo-GUI: abrir directamente el archivo desde el programa en el icono de la carpeta.

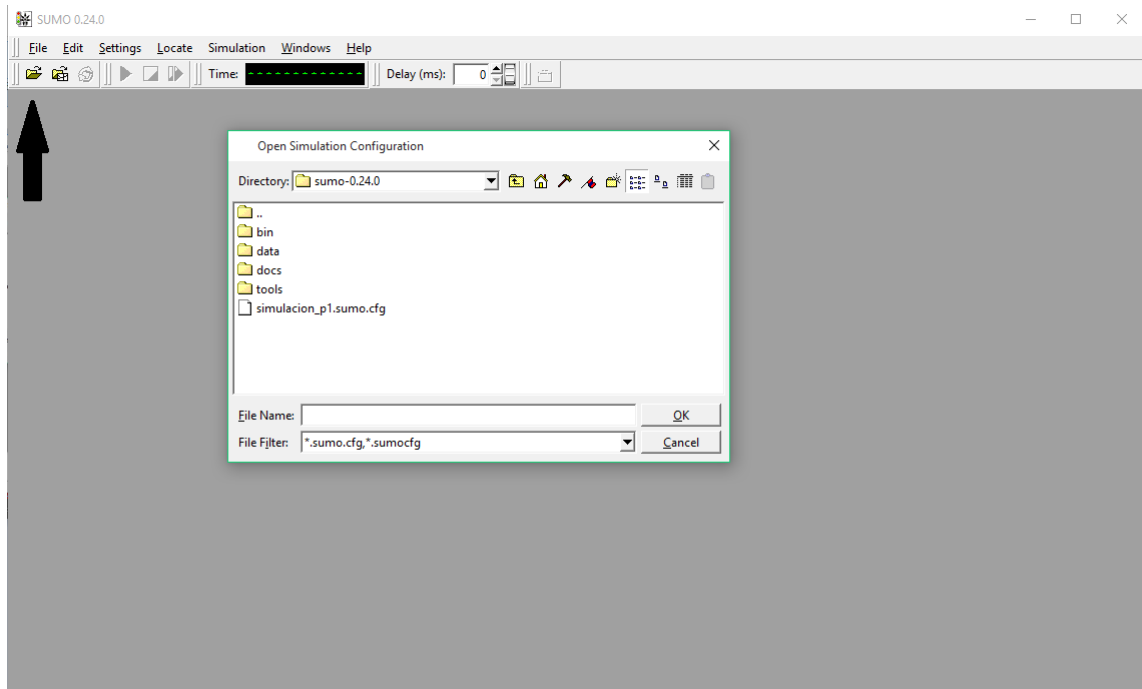


Fig.31 Abrir visualización desde Sumo-GUI.

La otra carpeta visualiza el archivo net.net.xml.



CAPÍTULO IV: CONCLUSIONES

Este trabajo fin de máster tenía como objetivo explicar cómo influyen los simuladores de tráfico hoy en día en la vida diaria, como pueden ayudar o que pueden aportar. Los simuladores pueden ayudar a ver como se encuentra una ciudad, entorno urbano en diferentes eventos que ocurran en un ella o también como pueden afectar por ejemplo radares en tramos de carreteras, autovías, como afecta al tráfico un túnel de nueva construcción limitando el simulador a las condiciones de la vía como velocidad o tipo de tráfico que transite por ella.

Los objetivos en el presente caso eran ver que nos pueden aportar en la ciudad de Linares. Tras ilustrar cuatro simuladores se tomo la decisión de realizarlo con Sumo y ofrecer una visualización del caso de Linares.

De las aportaciones de Sumo no se han podido realizar algo tan extenso como la simulación de una aglomeración de tráfico por un evento en particular como un concierto, corrida de toros o unas rutas claras de urbanos, lo que se ha hecho es una simulación más general de la realidad creando rutas alternativas al azar creadas por una herramienta específica de Sumo, “randomTrips.py”. Todo esto en parte no fue más específico por no tener datos más reales, que los que nos ofrecía OpenStreetMap, además que el archivo .osm no fue del todo completo como a lo mejor lo sería de Madrid.

Las aportaciones de este trabajo son positivas para una persona que se inicie con Sumo por primera vez, pues comenzar a iniciar Sumo es un poco lioso si no se tienen puntos claves para su uso, como el tener instalado Python para que Sumo pueda usar sus herramientas, con lo que este trabajo será de gran ayuda para usuarios iniciales en Sumo.

Los beneficios potenciales del tema de trabajo, simuladores en general son bastantes positivos a la hora de realizar un estudio de carreteras sobre todo desde mi punto de vista, para un estudio de optimización de recursos a realizar una carretera y así también ver el estudio económico viendo una simulación de “lo que se busca” para un futuro proyecto, por ejemplo.

Como valoración personal veo el tema muy actual y en progreso, con futuro pero también decir que debería de tener algo más de estudio detallado del terreno como más detalle en elevaciones, coordenadas UTM todo usado más referido a la realidad no



simplemente buscar puntos o semáforos por ejemplo por numero de intersecciones, para mi debería de tener las dos opciones el software Sumo. Pero en general es bastante completo además de libre y está progresando y facilitando cada vez más su uso e interface como por ejemplo con Sumo-Veins que es una extensión de Sumo que mejora la interface simulador-usuario.



Bibliografía

Barceló, J. (2010). *Fundamental of traffic simulation*. Barcelona: Springer.

Daniel Krajzewicz, J. E. (Abril de 2012). Recent Development and Applications of SUMO – Simulation of Urban MObility. *International Journal on Advances in Systems and Measurements* , págs. 128-138.

Erdmann, J. (2014). SUMO's Lane-Changing Model. En M. W. Michael Behrisch, *Modeling Mobility with Open Data* (págs. 105-123). Berlín: Springer.

Linares, A. d. (27 de 7 de 2009). *Ciudad de Linares*. Recuperado el 10 de octubre de 2015, de http://www.ciudaddelinares.es/portalLinares/p_20_contenedor1.jsp?seccion=s_fdes_d4_v2.jsp&codbusqueda=194&language=es&codResi=1&codMenuPN=190&codMenu=213&layout=p_20_contenedor1.jsp&layout=p_20_contenedor1.jsp

Namdre. (8 de abril de 2009). *DLR*. Recuperado el 12 de octubre de 2015, de http://sumo.dlr.de/wiki/Simulation/Traffic_Lights

Namdre. (abril de 5 de 2009). *DLR*. Recuperado el 12 de octubre de 2015, de http://sumo.dlr.de/wiki/Definition_of_Vehicles,_Vehicle_Types,_and_Routes#Vehicle_Types

Namdre. (11 de julio de 2014). *DLR*. Recuperado el 12 de octubre de 2015, de <http://sumo.dlr.de/wiki/Tools/tls>

Namdre. (2015 de septiembre de 2015). *DLR*. Recuperado el 10 de octubre de 2015, de <http://sumo.dlr.de/wiki/Tools/tls>

Noori, H. (2012). Realistic Urban Traffic Simulation as Vehicular Ad-Hoc Network (VANET) via Veins Framework. *12th Conference of Open Innovations Framework Prgramm. FRUCT*, (págs. 100-105). Tampere.

Tulinares. (2006). *Tulinares*. Recuperado el 10 de octubre de 2015, de <http://www.tulinares.com/lineas.php>

Uppoor, S., & Fiore, M. (2011). Uppoor, S.; Fiore, M.; “Large-scale urban vehicular mobility for networking research,” Vehicular Networking Conference (VNC). *Large scale urban vehicular mobility for networking research*, (págs. 62-69).

Wikipedia. (s.f.). Obtenido de [https://es.wikipedia.org/wiki/Linares_\(Ja%C3%A9n\)](https://es.wikipedia.org/wiki/Linares_(Ja%C3%A9n))

Wikipedia, C. (19 de octubre de 2015). *Wikipedia, la enciclopedia libre*. Recuperado el 10 de octubre de 2015, de [https://es.wikipedia.org/w/index.php?title=Linares_\(Ja%C3%A9n\)](https://es.wikipedia.org/w/index.php?title=Linares_(Ja%C3%A9n))