



## **Proyecto Final de Carrera**

### **Ingeniería en Informática**

**Curso 2010/2011**

## **Anexos**

**Eduardo López García**

Director: Sergio Ilarri Artigas  
Departamento de Informática e Ingeniería de Sistemas  
Centro Politécnico Superior  
Universidad de Zaragoza

Diciembre de 2010



## **ANEXO A – MANUAL DE USUARIO**

---

Este anexo contiene un manual de usuario del simulador desarrollado. En él se explicarán todas las acciones que se pueden realizar. Junto a estas explicaciones aparecerán imágenes para facilitar su comprensión.

### **A.1. Introducción**

---

La aplicación es, de forma resumida, una herramienta de simulación para facilitar la evaluación de diferentes técnicas de gestión de datos en redes de vehículos. En la actualidad se usa para evaluar diferentes técnicas basadas en agentes móviles para monitorizar áreas geográficas usando vehículos, pero el simulador puede utilizarse para otros propósitos y extenderse. Los vehículos poseen diferentes estrategias de movilidad con las que se mueven por el mapa de simulación. Además, el simulador permite modificar distintos parámetros de configuración de la simulación, los cuales se pueden guardar para su posterior uso con una herramienta específica para esta función. Existe una herramienta que permite la descarga de nuevos mapas desde diversas fuentes para incluirlos en las simulaciones. También cuenta con una herramienta que posibilita la importación de trazas GPS, en la que se pueden realizar diversos filtros para centrarse en las trazas que más interesen. Además, existe un sistema que permite añadir a las simulaciones elementos fijos de soporte a la VANET. Y por último, todo esto se controla utilizando una interfaz de usuario fácil de usar y amigable.

En los sucesivos apartados de este manual de usuario se irán explicando desde la “instalación” de la aplicación hasta las diversas funcionalidades que forman parte del simulador.

### **A.2. Ejecución de la aplicación**

---

En esta sección se va a explicar todo lo referente a la ejecución de la aplicación. Para ello se comenzará por contar qué hace falta tener instalado para que el simulador funcione y cómo instalar el simulador.

- Se debe instalar un JDK de Java. En un sistema operativo Linux se debe instalar el JDK de Java 1.6 update 11 o superior. En un sistema operativo Windows se debe instalar el JDK de Java 1.6 update 18 o superior. Se puede descargar desde la web oficial<sup>1</sup>.
- Si se quiere ejecutar este simulador o cualquier aplicación programada en el lenguaje de programación Java desde cualquier ruta del sistema es necesario realizar unas modificaciones de las variables de entorno en Windows<sup>2</sup> y en Linux<sup>3</sup>.

---

<sup>1</sup> <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

<sup>2</sup> <http://www.webtutoriales.com/tutoriales/programacion/java/descargar-instalar-java.17.html>

<sup>3</sup> <http://www.pcdigital.org/instalar-java-jdk6-debian-ubuntu/>

- Copiar la carpeta del simulador en el disco duro de nuestro ordenador. No hace falta copiarla en una ubicación especial, se puede situar donde el usuario desee.
- Para instalar la aplicación, se debe entrar dentro de la carpeta y acceder a la carpeta “SCRIPTS”. Dependiendo del sistema operativo en el que nos encontremos entraremos en la carpeta “Unix” o “Windows”. A continuación, se ejecutará el script de compilación: “compil.sh” para Unix y “win\_compilacion.bat” para Windows.
- Para ejecutar la aplicación sólo hace falta ejecutar el script de Unix “simul.sh” o el de Windows “win\_simulacion.bat” dependiendo del sistema operativo en el que nos encontremos.

### A.2.1. Ejecución en modo gráfico

La aplicación permite la ejecución de la misma de dos formas: en modo gráfico y en modo línea de comandos (modo *batch*). En este apartado se va a explicar lo necesario para realizar la ejecución en modo gráfico.

Si nos encontramos en un sistema operativo Unix, hay que utilizar el script de ejecución (simul\_graphic.sh). La línea del script que ejecuta el simulador en modo gráfico es la siguiente:

```
java -cp ../../BIN/../../LIBS/args4j.jar simulator $*
```

Si nos encontramos en un sistema operativo Windows, hay que utilizar el script de ejecución (win\_simulacion\_graphic.bat.) La línea del script que ejecuta el simulador en modo gráfico es la siguiente:

```
java simulator
```

### A.2.2. Ejecución modo en batch o línea de comandos

A continuación, se va a explicar lo necesario para realizar la ejecución en modo *batch*. Además, se explicarán los distintos parámetros que se pueden añadir a la línea de comandos para modificar la simulación.

Si nos encontramos en un sistema operativo Unix, hay que utilizar el script de ejecución (simul\_batch.sh). La línea del script que ejecuta el simulador en modo *batch* es la siguiente:

```
java -cp ../../BIN/../../LIBS/args4j.jar simulator -batch
```

Si nos encontramos en un sistema operativo Windows, hay que utilizar el script de ejecución (win\_simulacion\_batch.bat.) La línea del script que ejecuta el simulador en modo *batch* es la siguiente:

```
java Simulator -batch
```

A continuación, se van a explicar los diferentes parámetros de configuración de la simulación que existen tanto para el modo *batch* como para el modo gráfico. Además se incluirán varios ejemplos.

Todos los siguientes parámetros de simulación deben ir precedidos de un guión (“-”).

- **batch**: ejecución en modo *batch* (modo no gráfico).
- **view**: con este parámetro se conseguiría sólo visualizar el grafo y no se realizaría ninguna simulación. Está pensado para depuración y para tomar capturas de pantalla del grafo.
- **sni**: se muestran los identificadores de los nodos del grafo.
- **p**: no se muestran las *tiles* del mapa
- **t**: con este parámetro seguido del número que se desee se modifica el tiempo de iteración. Por ejemplo, -t 350 se modificaría el tiempo de iteración a 350 (3,5 ms).
- **v**: con este parámetro seguido del número que se desee se indica el número de vehículos de la simulación. Por ejemplo, -v 35, habrá 35 vehículos en la simulación.
- **f**: con este parámetro seguido del número que se desee se modifica el número de dispositivos fijos.
- **j**: con este parámetro seguido de un número comprendido entre 1 y 4 se selecciona la estrategia de salto para los agentes móviles. 1 = Random, 2 = BEP (Basic Encounter Probability), 3 = Min distance, 4 = Front angle.
- **mob**: con este parámetro seguido de un número comprendido entre 1 y 6 se selecciona el tipo de estrategia de movilidad para los vehículos. 1 = Random, 2 = Routes, 3 = Straight, 4 = Heur&Dest, 5 = GaussMarkov, 6 = ManhattanDynamic.
- **e**: con este parámetro seguido de un número comprendido entre 1 y 6 se selecciona el tipo de experimento que se quiere realizar en la simulación. 1 = Monitoring, 2 = Query from fixed origin, 3 = Query from moving origin, 4 = Query flooding from fixed origin, 5 = Query from moving origin within limited area, 6 = Query from fixed origin with limited area.
- **map**: con este parámetro seguido de un nombre (indica el nombre mapa) se cargará dicho mapa en la simulación, si existe. Por ejemplo, -map Zaragoza, cargará el mapa con el nombre de Zaragoza.

- **m:** con este parámetro seguido del número que se desee se modifica el número de iteraciones de medición que se realizarán en el área geográfica a monitorizar. Sólo es aplicable cuando se quiera simular un experimento de monitorización.
- **prd:** con este parámetro seguido de un número comprendido entre 0 y 100 se modifica la probabilidad de un dispositivo de contener información relevante.
- **ant\_on:** con este parámetro se indica al simulador que se desea incluir en la simulación los elementos fijos de soporte en las comunicaciones si es posible.

A continuación, se muestran algunos ejemplos usando algunos de los parámetros de simulación anteriores en un sistema operativo Windows.

```
java Simulator -batch -v 125 -map Sevilla -mob 5
```

En el ejemplo anterior, se ejecuta en modo *batch*, con 125 vehículos, en el mapa de Sevilla y con la estrategia de movilidad 5 o Gauss-Markov.

```
java Simulator -ant_on -m 2500 -map Dublin -e 6 -j 2
```

En el ejemplo anterior, se ejecuta en modo gráfico, se han añadido los elementos fijos de soporte en las comunicaciones, se ha ajustado el número de iteraciones de medición a 2500, la simulación se realizará en el mapa de Dublín, el tipo de experimento será “Query from fixed origin with limited area” y la estrategia de salto de los agentes móviles será “BEP (Basic Encounter Probability)”.

### A.3. Pantalla inicial

---

Al ejecutar el simulador en modo gráfico aparece la pantalla inicial, que se diferencia de las posteriores pantallas de simulación en que en el área central aparece el logo de VESPA (Vehicular Event Sharing with a mobile P2P Architecture) [15], que es un proyecto que propone un sistema de comunicación ad hoc entre redes de vehículos, describe el intercambio de información en forma de estructuras denominadas eventos (aviso de accidentes, aviso de plazas libres de aparcamiento, etc.) y ofrece un método para estimar la relevancia que estos tienen para un vehículo receptor (Encounter Probability). La razón de que aparezca el logo de VESPA es que este simulador se va a utilizar para simular los beneficios que pueden aportar los agentes móviles en el contexto del proyecto de VESPA.

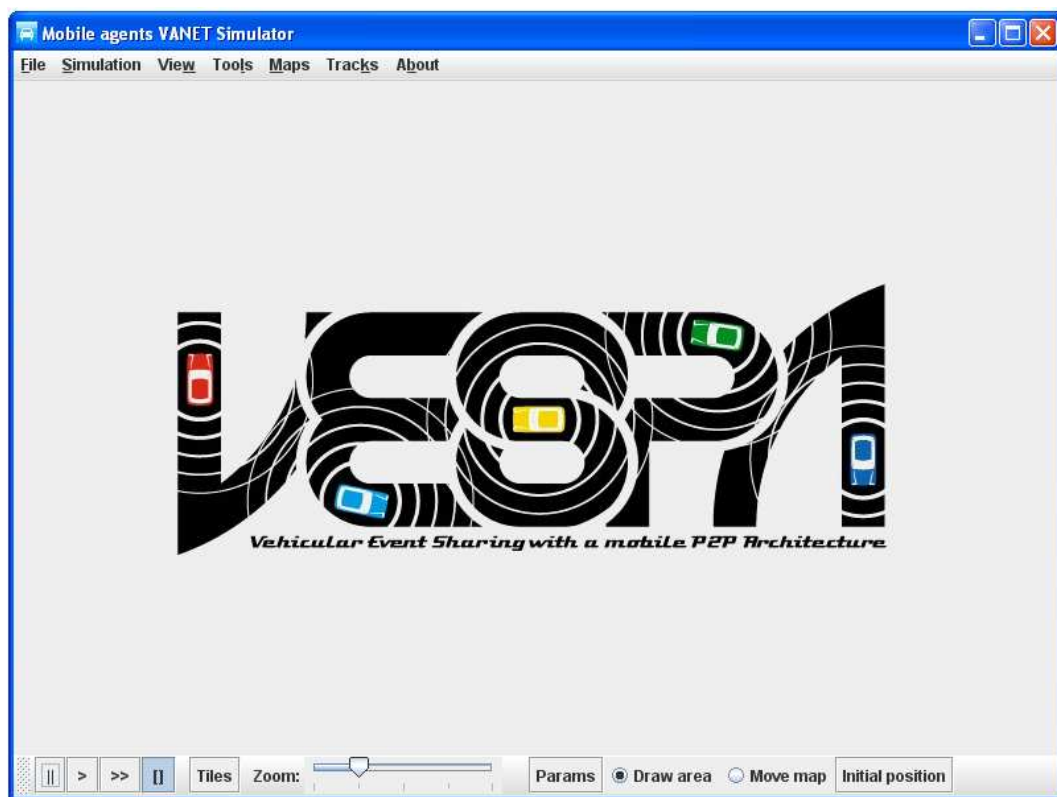


Figura A.1 – Pantalla inicial del simulador

En la figura A.1 se puede ver que el simulador consta de varias partes diferenciadas:

- Una barra superior de menú.
- Una parte central en la que aparecerá la simulación.
- Una barra inferior de control de la simulación.

Esta pantalla sólo aparece la primera vez que se ejecuta el simulador ya que en otros estados el simulador mostrará la simulación que se esté realizando en el momento.

## A.4. Menú superior

En esta sección se va a hablar del menú superior de la aplicación. Véase la Figura A.2 en la que se ha marcado en rojo el menú superior del que se va a hablar.

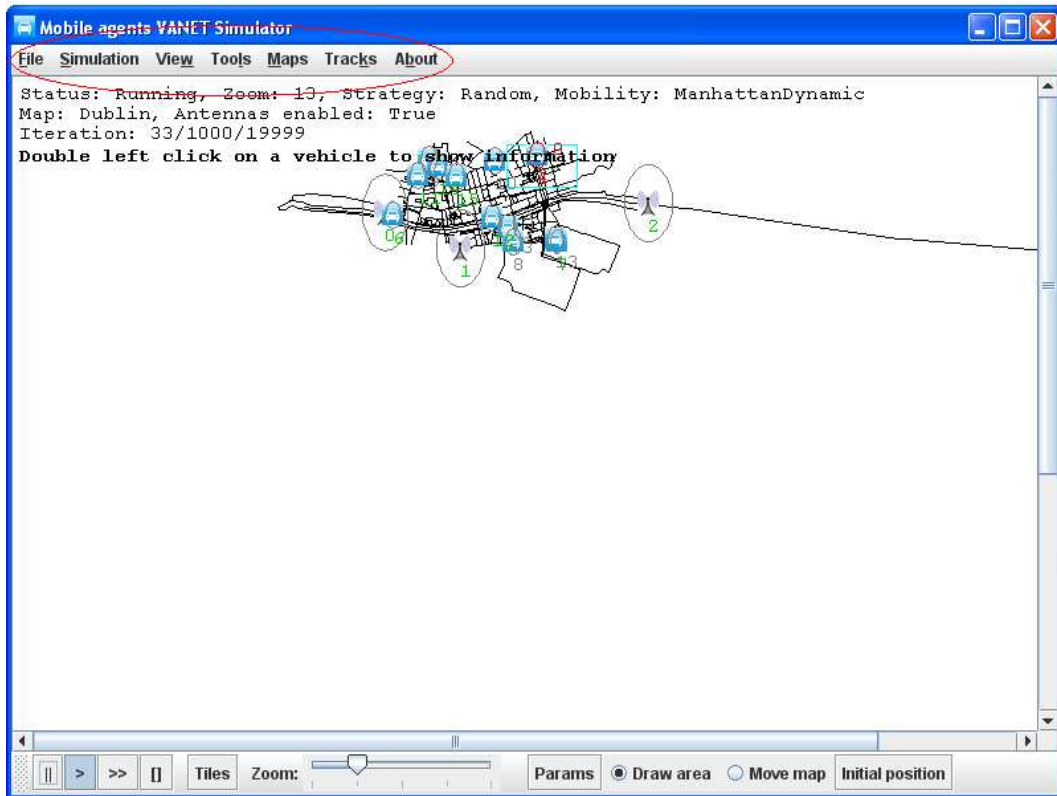


Figura A.2 – Pantalla del simulador con barra de menú remarcada

Desde este menú se puede controlar la simulación de una forma sencilla y acceder a todas las herramientas que conforman el simulador. La barra del menú consta de 7 opciones:

- *File*
- *Simulation*
- *View*
- *Tools*
- *Maps*
- *Tracks*
- *About*

### Menú File

En este menú aparecen 2 submenús: “New” y “Exit” (ver Figura A.3). Con la opción “New” se para la simulación actual de forma que el usuario pueda comenzar una nueva simulación pulsando el botón “play”. Para acceder de forma rápida a la opción “New” se puede usar la combinación de teclas de acceso rápido “Control + N” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.



Con la opción “Exit” se sale del simulador. Para acceder de forma rápida a la opción “Exit” se puede usar la combinación de teclas de acceso rápido “Control + E” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.



Figura A.3 – Menú *File* de la barra de menú

### **Menú Simulation**

En este menú aparecen 5 submenús: “Run”, “Pause”, “Stop”, “Fast” y “Simulation parameters” (ver Figura A.4). Con la opción “Run” se inicia la simulación. Para acceder de forma rápida a la opción “Run” se puede usar la combinación de teclas de acceso rápido “Control + R” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Pause” la simulación permanece pausada. Para acceder de forma rápida a la opción “Pause” se puede usar la combinación de teclas de acceso rápido “Control + P” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Stop” la simulación se para y se eliminan todos los elementos de la simulación. Para acceder de forma rápida a la opción “Stop” se puede usar la combinación de teclas de acceso rápido “Control + O” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Fast” la simulación se realiza de una forma rápida en comparación con la velocidad de la simulación en modo “Run”. Es decir, los vehículos se moverán mucho más rápido que si se simula en modo “Run”. Para acceder de forma rápida a la opción “Fast” se puede usar la combinación de teclas de acceso rápido “Control + A” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Simulation parameters” se abre un menú en el que se pueden modificar diversos parámetros de la simulación (ver Figura A.5). Si había una simulación funcionando, ésta permanecerá en estado de pausa hasta que se salga de dicho menú. Para acceder de forma rápida a la opción “Simulation parameters” se puede usar la combinación de teclas de acceso rápido “Control + U” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”. Para más información sobre el menú de configuración de parámetros de simulación, consúltese la Sección A.7 “Menú de configuración de los parámetros de simulación”.

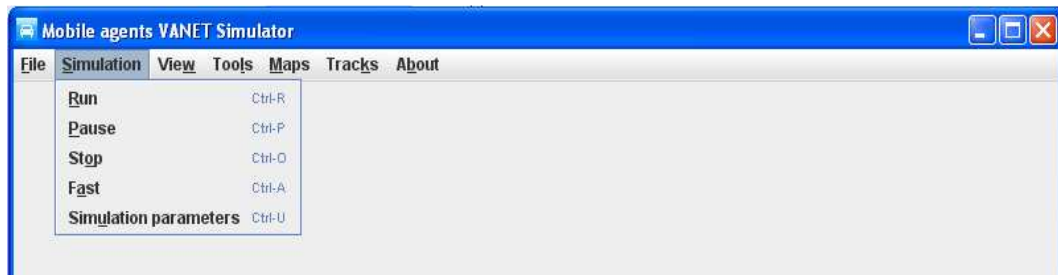
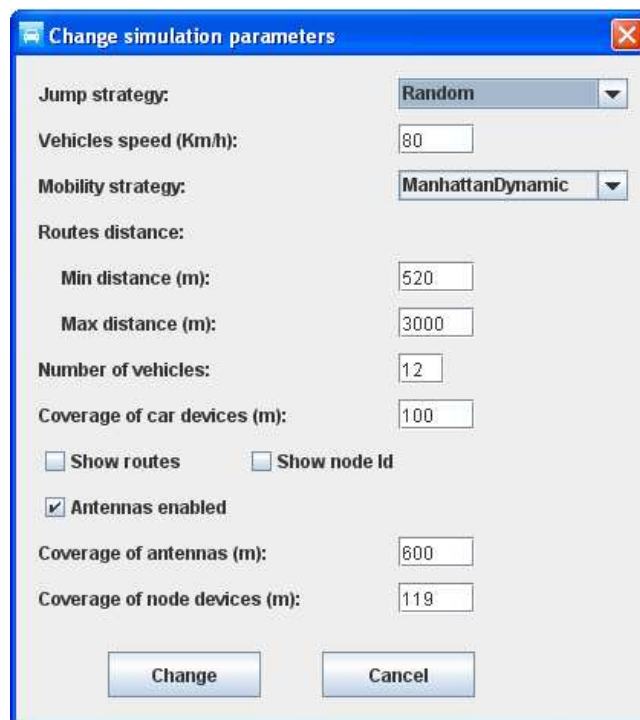
Figura A.4 – Menú *Simulation* de la barra de menú

Figura A.5 – Ventana de configuración de los parámetros de la simulación

### Menú View

En este menú aparecen 5 submenús: “Tiles”, “Zoom +”, “Zoom -”, “Initial Position” y “Info Vehicle options” (ver Figura A.6). Con la opción “Tiles” se muestran las *tiles*, o lo que es lo mismo el mapa sobre el que se está simulando. Para acceder de forma rápida a la opción “Tiles” se puede usar la combinación de teclas de acceso rápido “Control + I” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Zoom +” se acerca la vista de la zona en la que se está simulando. Para acceder de forma rápida a la opción “Zoom +” se puede usar la combinación de teclas de acceso rápido “Control + +” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Zoom -” se aleja la vista de la zona en la que se está simulando. Para acceder de forma rápida a la opción “Zoom -” se puede usar la combinación de teclas de acceso rápido “Control + ‘-’” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Initial position” la vista se sitúa donde estaba al iniciar la simulación. Normalmente la vista se encuentra centrada sobre el grafo del mapa de la simulación. Para acceder de forma rápida a la opción “Initial position” se puede usar la combinación de teclas de acceso rápido “Control + Z” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Info vehicle options” se abre una pequeña ventana en la que se puede configurar la información que se quiere visualizar cuando se haga doble clic sobre un vehículo (ver Figura A.7). Para acceder de forma rápida a la opción “Info Vehicle options” se puede usar la combinación de teclas de acceso rápido “Control + C” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”. Para más información sobre el menú de configuración la información a visualizar cuando se hace doble clic sobre un vehículo, consúltese la Sección A.8 “Menú de configuración de la información de un vehículo”.



Figura A.6 – Menú View de la barra de menú

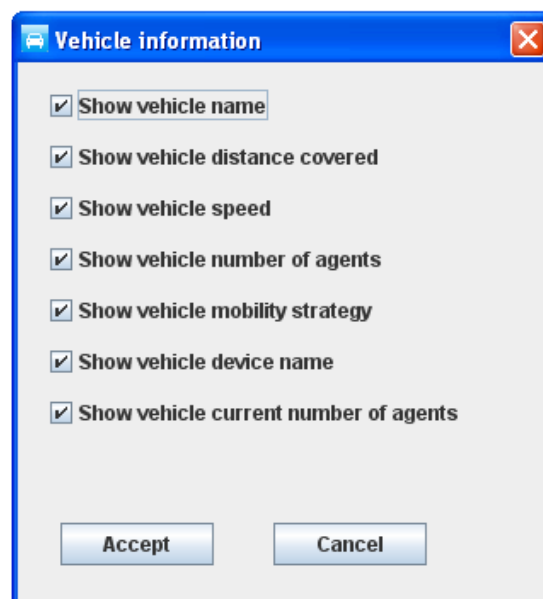


Figura A.7 – Ventana de configuración de la información a visualizar de un vehículo

### **Menú Tools**

En este menú aparecen 3 submenús: “ToolBar”, “Draw area” y “Move map” (ver Figura A.8). Con la opción “ToolBar” se muestra/esconde la barra de control de la simulación que aparece en la parte inferior de la ventana del simulador. Para acceder de forma rápida a la opción “ToolBar” se puede usar la combinación de teclas de acceso rápido “Control + T” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Draw area” se selecciona la función del cursor en el simulador de dibujar un área de monitorización. Esta función es excluyente de la función de moverse por el mapa en la cual aparece otro cursor diferente. Para acceder de forma rápida a la opción “Draw area” se puede usar la combinación de teclas de acceso rápido “Control + Y” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.

Con la opción “Move map” se selecciona la función del cursor en el simulador de moverse por el mapa. Esta función es excluyente de la función de dibujar el área de monitorización. Cuando se selecciona esta función del cursor, éste cambia su apariencia para indicar la nueva función. Para acceder de forma rápida a la opción “Move mapa” se puede usar la combinación de teclas de acceso rápido “Control + X” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.



**Figura A.8 – Menú Tools de la barra de menú**

### **Menú Maps**

En este menú aparecen 2 submenús: “Download” y “Open” (ver Figura A.9). Con la opción “Download” se abre una ventana con todas las opciones para la descarga de nuevos mapas (ver Figura A.10). Para acceder de forma rápida a la opción “Download” se puede usar la combinación de teclas de acceso rápido “Control + D” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”. Para más información sobre cómo descargar un mapa, consúltase la Sección A.10 “Cómo descargar un mapa”.

Con la opción “Open” se abre una ventana con todas las opciones para la carga de un mapa (ver Figura A.11). Para acceder de forma rápida a la opción “Open” se puede usar la combinación de teclas de acceso rápido “Control + H” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”. Para más información sobre cómo abrir un mapa, consúltase la Sección A.11 “Cómo abrir un mapa”.



Figura A.9 – Menú *Maps* de la barra de menú

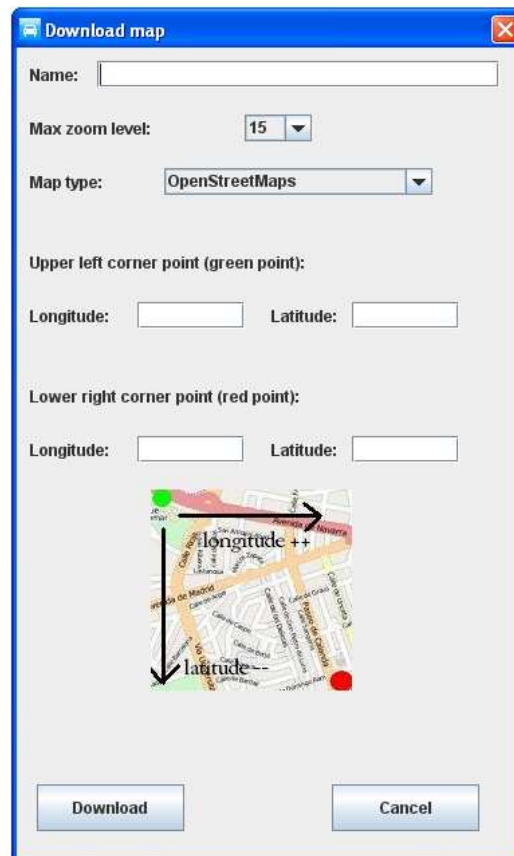


Figura A.10 – Ventana para descargar mapas

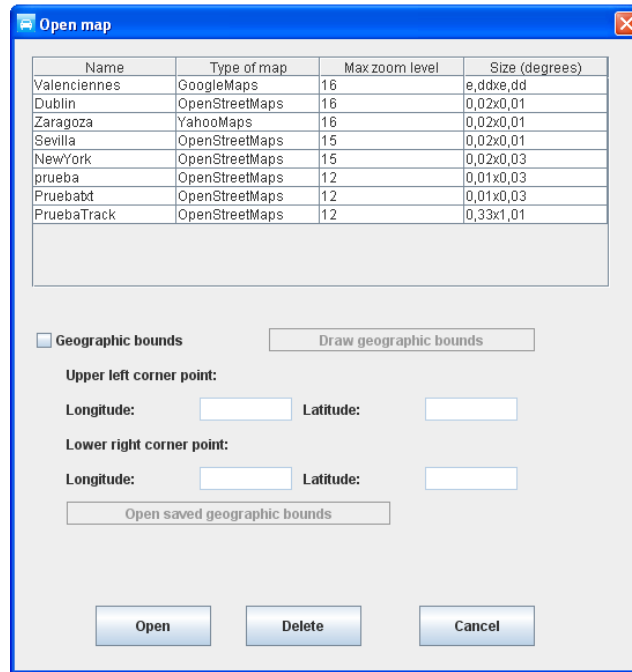


Figura A.11 – Ventana para abrir un mapa

### Menú Tracks

En este menú aparecen un submenú: “Import” (ver Figura A.12). Con la opción “Import” se abre una ventana con todas las opciones para la importación de trazas GPS (ver Figura A.13). Para acceder de forma rápida a la opción “Import” se puede usar la combinación de teclas de acceso rápido “Control + Q” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”. Para más información sobre cómo importar trazas GPS, consúltese la Sección A.14 “Cómo importar trazas GPS”.



Figura A.12 – Menú Tracks de la barra de menú

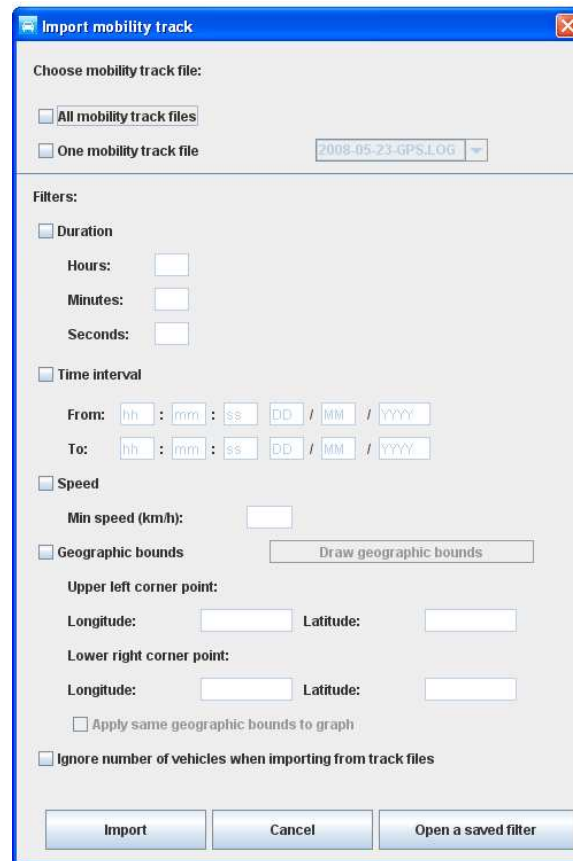


Figura A.13 – Ventana para importar trazas GPS

### Menú About

En este menú aparece un sólo submenú: “Version” (ver Figura A.14). Con la opción “Version” se abre una ventana con la información acerca de la versión del simulador (ver Figura A.15). Para acceder de forma rápida a la opción “Version” se puede usar la combinación de teclas de acceso rápido “Control + V” explicada en la Sección A.6 “Combinación de teclas de acceso rápido”.



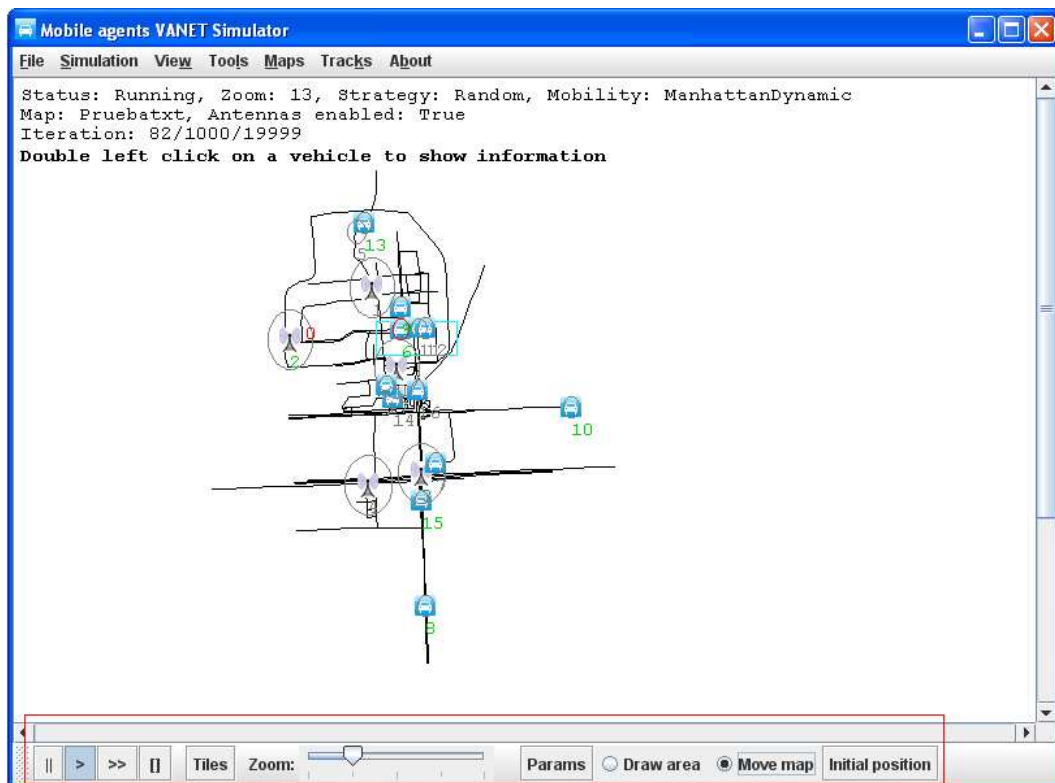
Figura A.14 – Menú About de la barra de menú



Figura A.15 – Ventana de la versión del simulador

### A.5. Barra inferior de control de la simulación

En la parte inferior de la ventana del simulador aparece la barra de control de la simulación (ver Figura A.16). Como se ha dicho en la sección anterior, esta barra puede mostrarse o esconderse a través de la opción “ToolBar” del menú “Tools” de la barra superior de menú.





La barra de control de la simulación consta de varios elementos (en orden de izquierda a derecha):

- Botón “pause”
- Botón “play”
- Botón “fast”
- Botón “stop”
- Botón “Tiles”
- Barra de zoom
- Botón “Params”
- Botones excluyentes “Draw area”/ ”Move map”
- Botón “Initial position”

Los botones “play”, “pause”, “stop”, “fast” y “Params” realizan la misma función que seleccionar la pertinente opción del menú “Simulation” de la barra superior de menú. El botón “Tiles”, la barra de zoom y el botón “Initial position” realizan la misma función que seleccionar la pertinente opción del menú “View”. Finalmente los botones excluyentes “Draw area”/”Move map” realizan la misma función que seleccionar la pertinente opción de menú “Tools”.

Si se quiere obtener información más detallada sobre la función que realiza cada botón de la barra de control de la simulación, consúltase la Sección A.4 “Menú superior” de este anexo.

## **A.6. Combinación de teclas de acceso rápido**

---

En esta sección se van a resumir las combinaciones de teclas de acceso rápido para acceder a diversas funciones del simulador. A continuación, se muestran estas combinaciones:

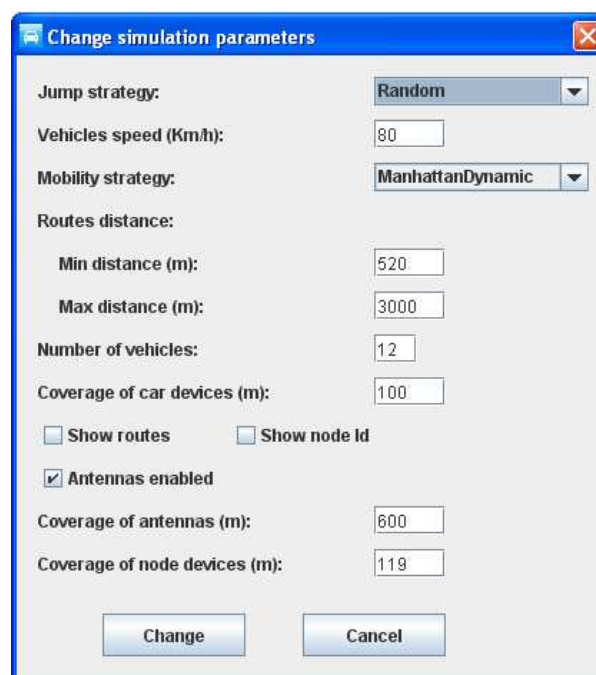
- Ctrl izq + N: función “New”
- Ctrl izq + E: función “Exit”
- Ctrl izq + R: función “Run” o botón “play”
- Ctrl izq + P: función “Pause” o botón “pause”
- Ctrl izq + O: función “Stop” o botón “stop”
- Ctrl izq + A: función “Fast” o botón “fast”
- Ctrl izq + U: función “Simulation parameters” o botón “Params”
- Ctrl izq + I: función “Tiles” o botón “Tiles”
- Ctrl izq + ‘+’: función “Zoom +” o mover la barra de zoom hacia la derecha
- Ctrl izq + ‘-’: función “Zoom -” o mover la barra de zoom hacia la izquierda
- Ctrl izq + Z: función “Initial position” o botón “Initial position”
- Ctrl izq + C: función “Info vehicle options”
- Ctrl izq + T: función “ToolBar”
- Ctrl izq + Y: función “Draw area” o botón “Draw area”

- Ctrl izq + X: función “Move map” o botón “Move map”
- Ctrl izq + D: función “Download”
- Ctrl izq + H: función “Open”
- Ctrl izq + Q: función “Import”
- Ctrl izq + V: función “Version”

### **A.7. Menú de configuración de los parámetros de la simulación**

A través de la ventana de configuración de los parámetros de la simulación se puede modificar el entorno de simulación de una forma rápida y sencilla, y además que tenga efecto sobre la simulación actual. Esta ventana se abre con la opción “Simulator parameters” del menú “Simulation” o con el botón “Params” de la barra de control de la simulación.

A continuación, se muestra la ventana de configuración de los parámetros de la simulación:



**Figura A.17 – Ventana de configuración de los parámetros de la simulación**

Como se ve en la Figura A.17 se pueden modificar diversos parámetros de la simulación. Algunas modificaciones tienen efecto en la simulación actual, si hay una simulación funcionando en el momento que se abre dicha ventana, y otros tienen efecto en la siguiente simulación que se realice. A continuación, se enumeran todos los parámetros de configuración de la ventana y se explica para qué sirven.

- **Jump strategy (estrategia de salto):** con este menú desplegable se puede modificar la estrategia de salto de los agentes móviles. La modificación tiene efecto en la simulación actual.

- **Vehicle speed (velocidad de los vehículos):** se modifica la velocidad de todos los vehículos. La velocidad se indica en km/h. La modificación tiene efecto en la simulación actual.
- **Mobility strategy (estrategia de movilidad):** con este menú desplegable se puede modificar la estrategia de movilidad de todos los vehículos. La modificación tiene efecto en la simulación actual.
- **Min distance (mínima distancia):** se modifica la mínima distancia a la que se puede encontrar el nodo destino en una estrategia basada en rutas. La distancia se indica en metros. La modificación tiene efecto en la simulación actual.
- **Max distance (máxima distancia):** se modifica la máxima distancia a la que se puede encontrar el nodo destino en una estrategia basada en rutas. La distancia se indica en metros. La modificación tiene efecto en la simulación actual.
- **Number of vehicles (número de vehículos):** se modifica el número de vehículos de una simulación. La modificación tiene efecto en la siguiente simulación.
- **Coverage of car devices (cobertura del dispositivo de los coches):** se modifica la cobertura (rango de alcance de comunicaciones) del dispositivo integrado en cada vehículo que permite la comunicación con otros vehículos. La cobertura se indica en metros. La modificación tiene efecto en la simulación actual.
- **Show routes (mostrar las rutas):** se muestran/esconden las rutas de los vehículos cuando siguen una estrategia de movilidad basada en rutas. La modificación tiene efecto en la simulación actual.
- **Show node id (mostrar el identificador de los nodos):** se muestra/esconde el identificador de los nodos que conforman el grafo. La modificación tiene efecto en la simulación actual.
- **Antennas enabled (activar antenas):** se usan los elementos fijos de soporte para las comunicaciones, si existen para ese mapa. La modificación de este parámetro sólo se puede realizar si el simulador no tiene ninguna simulación funcionando, por lo que tendrá efecto en la siguiente simulación.
- **Coverage of antennas (cobertura de las antenas):** se modifica la cobertura de los elementos fijos de soporte de las comunicaciones. La cobertura se indica en metros. La modificación tiene efecto en la simulación actual.

- **Coverage of node devices (cobertura de los dispositivos de los nodos):** se modifica la cobertura (rango de alcance de comunicaciones) de los dispositivos que se encuentran en los nodos del grafo utilizado para representar las carreteras del mapa. La cobertura se indica en metros. La modificación tiene efecto en la simulación actual.

Una vez que se han modificado los parámetros hay que pulsar el botón “Change” para que los cambios se produzcan. Si se quiere salir de esta ventana se puede hacer pulsando el botón “Cancel” o con la cruz de la parte superior derecha de la ventana.

## A.8. Menú de configuración de la información de un vehículo

---

A través de la ventana de configuración de la información de un vehículo se puede modificar la información que se visualizará cuando se haga doble clic sobre un vehículo. Esta ventana se abre con la opción “Info vehicle options” del menú “View”. A continuación, se muestra la ventana (Figura A.18).

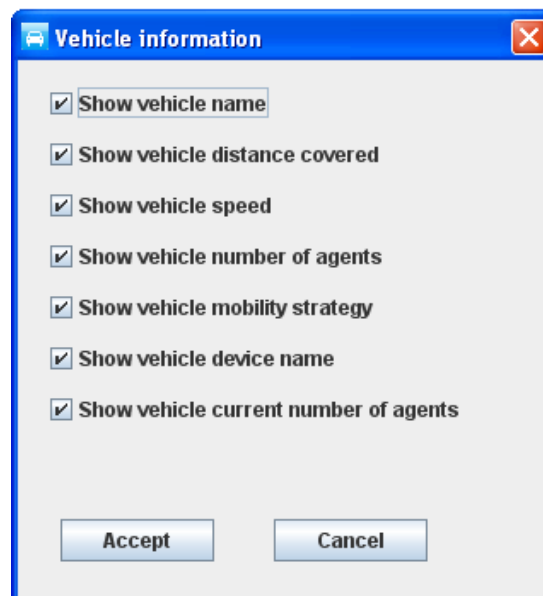


Figura A.18 – Ventana de configuración de la información de un vehículo a visualizar

Si aparece el signo “tick” quiere decir que dicha información aparecerá cuando se haga doble clic sobre un vehículo.

Por tanto, en la Figura A.19 aparecerá toda la información cuando se haga doble clic sobre un vehículo. Si no se quiere mostrar algún atributo, simplemente hay que hacer clic sobre dicho atributo y entonces el “tick” desaparecerá y no se mostrará dicha información del vehículo.

Para que los cambios se produzcan hay que pulsar el botón “Accept”, o si no se quiere que los cambios se produzcan hay que pulsar el botón “Cancel”.

## A.9. Cómo visualizar información de un vehículo

---

Para visualizar la información de un vehículo durante una simulación simplemente hay que realizar doble clic sobre el vehículo sobre el que se esté interesado. La única restricción tiene que ver con el cursor. Solo se muestra la información del vehículo si la función del cursor es “Draw area”, es decir, el cursor tiene la forma habitual de flecha. A continuación, se muestra la ventana que aparecerá (Figura A.19).



Figura A.19 – Ventana con información de un vehículo

## A.10. Cómo descargar un mapa

Al acceder a la opción “Download” del menú “Maps” se abre la ventana para descargar un mapa de la zona que se desee (ver Figura A.20). A continuación, se va a explicar todo acerca de cómo descargar un mapa.

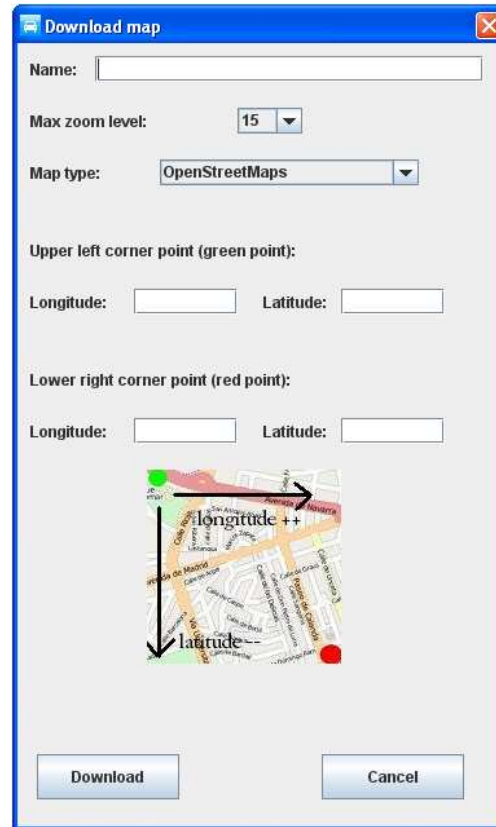


Figura A.20 – Ventana de descargar mapas

El primer elemento que nos encontramos en la ventana es el campo de texto “Name”. Aquí hay que escribir el nombre que se quiera para el mapa que se va a descargar. De esta forma cuando busquemos el mapa para abrirlo para una simulación, éste será el nombre con el que aparecerá.

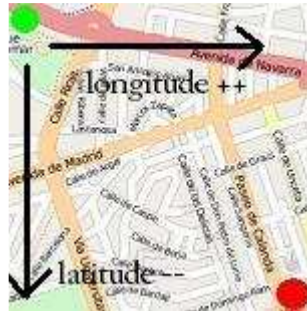
El segundo elemento que aparece es el menú desplegable “Max zoom level” con el podemos seleccionar el nivel máximo de zoom con el que queremos descargar del mapa. De esta forma, cuando usemos ese mapa en una simulación, si se va a usar un nivel de zoom que no se ha descargado previamente el simulador los descargará automáticamente. El inconveniente será que tardará más que si ya se hubiera descargado previamente, ya que en este caso estaría almacenado en el disco duro. El nivel máximo de zoom que se permite seleccionar con este menú desplegable es 16, ya que se considera suficiente nivel de zoom para observar un mapa con detalle.

El siguiente elemento que aparece en la ventana es “Map type” o lo que es lo mismo, el tipo de mapa que se va a descargar. Este menú permite seleccionar entre 5 tipos de mapa (GoogleMaps, OpenStreetMap, YahooMaps, MicrosoftMaps y MicrosoftMapsHybrid). Con este menú se elige el tipo de servidor desde el que se desea descargar el mapa, y por tanto la apariencia del mismo será diferente dependiendo de la fuente de la que se descargue.

Los cuatro siguientes y últimos campos de texto son los encargados de delimitar la zona que se quiere descargar. Por un lado, encontramos “Upper Left corner point (green point)” y por otro “Lower right corner (red point)”. Es decir, se indica el mapa que se quiere descargar con 2 puntos, uno situado en la posición superior izquierda y el otro en la posición inferior derecha.

Hay que tener en cuenta que la latitud aumenta cuando nos acercamos hacia el polo norte. Por tanto, como bien indica la imagen explicativa de la ventana de descarga de mapas, la latitud del punto superior izquierdo es mayor que la latitud del punto inferior derecho. Hay que tener en cuenta que un mapa de  $1^\circ \times 1^\circ$  es muy grande y dependiendo de la zona puede descargarse un fichero OSM muy extenso y una gran cantidad de imágenes que conformen el mapa, por lo que no se aconseja descargar zonas tan extensas. Para que nos hagamos una idea, Zaragoza está separada de los Pirineos por casi  $1^\circ$  de latitud y separada de Monzón por aproximadamente  $1^\circ$  de longitud (información extraída a través de [14]).

Una vez que todos los campos están rellenos, hay que pulsar el botón “Download” para iniciar la descarga. Si algún campo es incorrecto o falta por completarse, se avisará de ello.



**Figura A.21 – Definición área a descargar con 2 puntos**

A continuación, se muestra un ejemplo (ver Figura A.22). En el ejemplo el nombre del mapa es “Ejemplo”, El nivel máximo de zoom es 13, el tipo de mapa que se está descargando es “GoogleMaps”. El punto que delimita el mapa a descargar y está situado en la parte superior izquierda tiene longitud = 30 y latitud = 50. Por otra parte el punto situado en la parte inferior derecha tiene longitud = 30.28 y latitud = 49.86. Para obtener las coordenadas geográficas de un punto que interese y después proceder a la descarga de ese mapa, se aconseja el uso de la siguiente web [14].

Como se ve en la Figura A.22, una vez pulsado el botón “Download” se mostrará una barra de progreso que indicará el estado de la descarga.

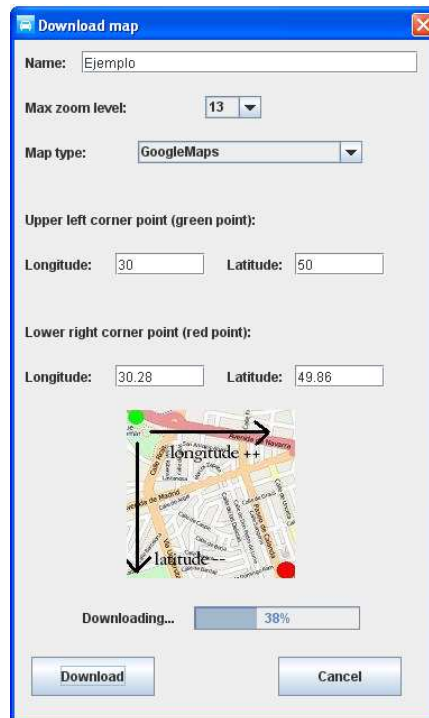


Figura A.22– Ventana de descarga en progreso

Cuando se habla de seleccionar el tipo de mapa que se quiere descargar hay que tener en cuenta que lo que se selecciona es el servidor de imágenes de mapas, ya que los datos del mapa (información sobre las calles y carreteras utilizada para componer el grafo) se descarga automáticamente de OpenStreetMap.



## A.11. Cómo abrir un mapa

Una vez que un mapa se ha descargado ya se puede abrir para realizar simulaciones sobre él. Para que aparezca la ventana desde la que se puede abrir un mapa es necesario seleccionar la opción “Open” del menú “Maps”, y se mostrará la siguiente ventana (ver Figura A.23).

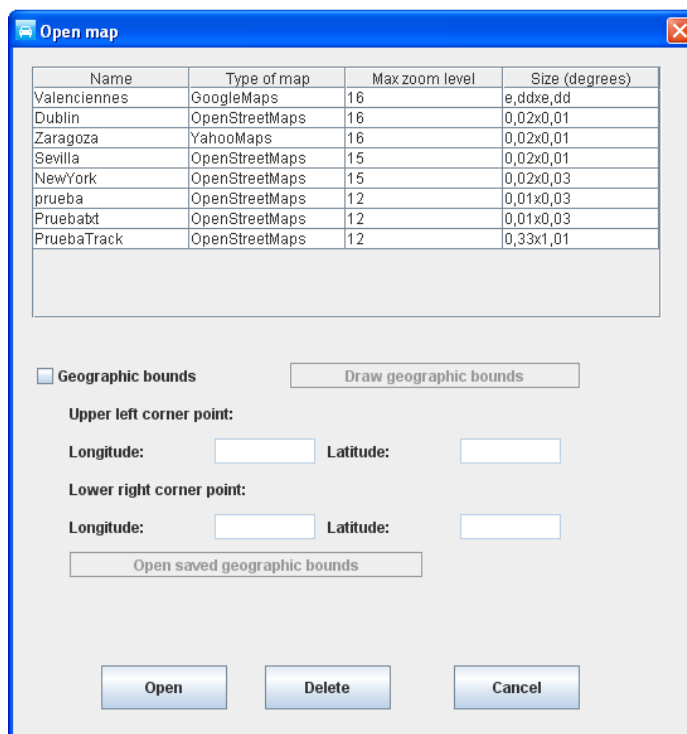


Figura A.23 – Ventana para abrir un mapa

La ventana “Open map” consta de 2 partes en general: la tabla en la que aparecen los mapas disponibles para realizar simulaciones y los elementos necesarios para realizar un filtro geográfico sobre un mapa (ver Sección A.12 “Aplicar un filtro geográfico a un mapa”). En esta sección nos centraremos en abrir un mapa de forma rápida sin aplicar un filtro.

### A.11.1. Abrir un mapa

En la tabla en la que aparecen los mapas disponibles para usar en las simulaciones aparecen varias filas y varias columnas. Cada fila pertenece a un mapa, y cada columna muestra diferente información. La primera columna muestra el nombre del mapa. Este nombre fue el que el usuario le puso cuando lo descargó con la herramienta de descarga de mapas. La siguiente columna muestra el tipo de mapa (GoogleMaps, OpenStreetMaps, etc.). La siguiente columna muestra el nivel máximo de zoom que se descargó con la herramienta de descarga de mapas. Finalmente, la última columna muestra el tamaño del mapa en grados x grados.

Por ejemplo, la tercera fila indica el mapa llamado Zaragoza, el tipo de mapa es YahooMaps, el nivel máximo de zoom es 16 y su tamaño es 0,02 ° x 0,01°.

Para abrir un mapa, lo único que hay que hacer es seleccionar el mapa que se desee y pulsar el botón “Open”. Si existe una simulación en el momento en el que se quiere abrir el mapa aparecerá la siguiente ventana.

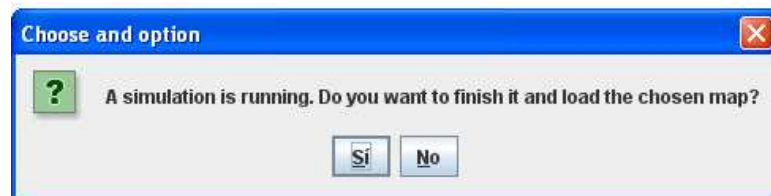


Figura A.24 – Ventana para iniciar una nueva simulación existiendo una ya

A continuación, se abrirá el mapa y comenzará la simulación.

### A.11.2. Eliminar un mapa

---

Para eliminar un mapa, simplemente hay que seleccionar el mapa de la tabla que se desea eliminar y pulsar el botón “Delete”. A continuación, se actualizará la tabla de los mapas disponibles y dicho mapa ya no aparecerá.

## A.12. Aplicar un filtro geográfico a un mapa

---

Puede resultar interesante centrar la simulación en una determinada zona de un mapa. Para ello existen los filtros geográficos, que omiten aquellos puntos o nodos del mapa que se encuentran fuera del área seleccionada.

Aplicar un filtro geográfico a un mapa es una tarea muy sencilla. Lo primero que hay que hacer es abrir la ventana a partir de la cual se abren los mapas (consultar la Sección A.11 “Cómo abrir un mapa”).

En esta ventana, se selecciona el mapa al que se desee aplicar el filtro geográfico, después se selecciona la opción “Geographic bounds” y ya podremos indicar el filtro geográfico que se desee.

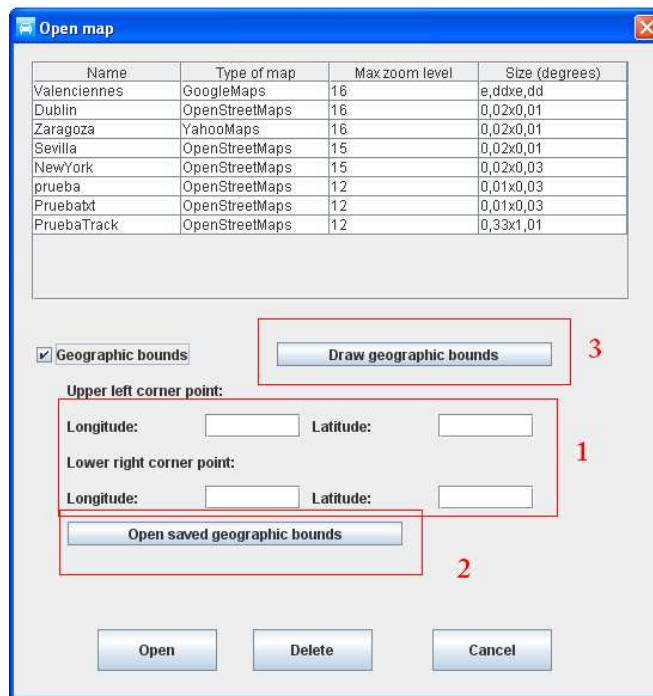


Figura A.25 – Ventana abrir mapa con filtro geográfico remarcado

Existen 3 formas de indicar un filtro geográfico:

- Escribiendo la latitud y longitud de los 2 puntos que delimitan el área (número 1 de la Figura A.25).
- Cargando o abriendo un filtro guardado (número 2 de la Figura A.25).
- Dibujando sobre el mapa el área que determinará el filtro geográfico (número 3 de la Figura A.25).

Una vez que se hayan determinado los valores del filtro de alguna de las formas anteriores, y pulsado el botón “Open” se mostrará la siguiente ventana preguntando si se desea guardar el filtro seleccionado.

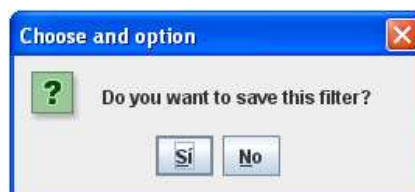


Figura A.26 – Ventana pregunta para guardar el filtro

Si se quiere guardar el filtro indicado aparecerá la siguiente ventana para que se le indique el nombre del filtro.

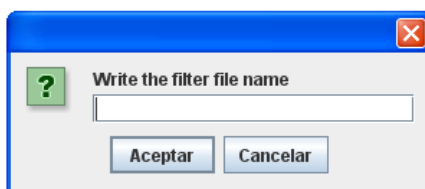


Figura A.27 – Ventana para introducir el nombre del filtro

A continuación, se muestran las distintas formas de indicar un filtro geográfico para un mapa.

### A.12.1. Filtro geográfico: escribir filtro

En la Figura A.25 la parte de la ventana recuadrada e indicada con el número 1 es la encargada del filtro manual.

Los cuatro campos de texto son los encargados de delimitar la zona que se quiere filtrar, es decir, la zona que nos interesa del mapa. Por un lado encontramos “Upper Left corner point” y por otro “Lower right corner”. Es decir, se indica la zona que nos interesa con 2 puntos, uno situado en la posición superior izquierda y el otro en la posición inferior derecha.

Hay que tener en cuenta que la latitud crece hacia el polo norte. Por tanto, la latitud del punto superior izquierdo es mayor que la latitud del punto inferior derecho. Una buena herramienta para conocer las coordenadas geográficas de un lugar es [14].

### A.12.2. Filtro geográfico: abrir filtro

En la Figura A.25 la parte de la ventana recuadrada e indicada con el número 2 es la encargada de abrir un filtro guardado. Si se pulsa el botón “Open saved geographic bounds” se abrirá la siguiente ventana.

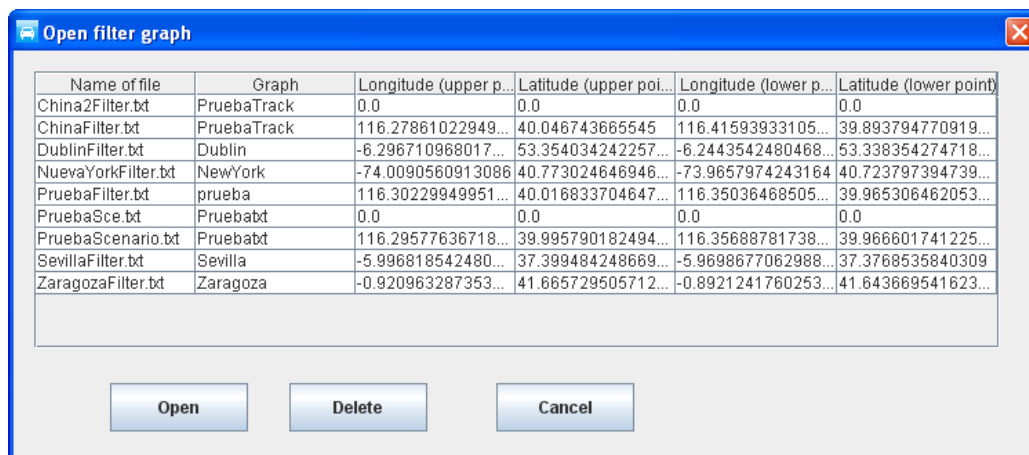


Figura A.28– Ventana para abrir un filtro geográfico para un mapa

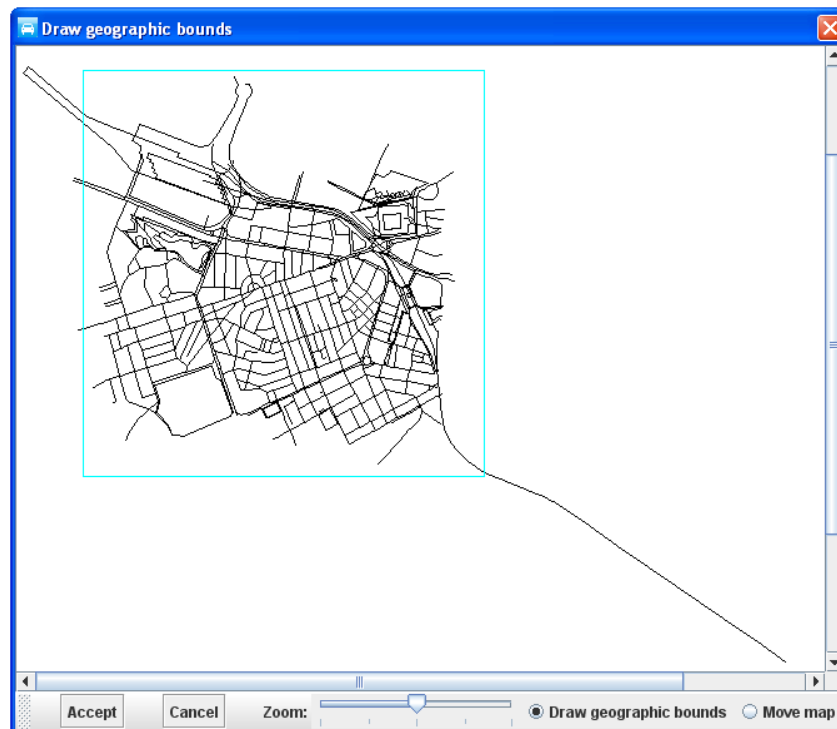
En la tabla que aparece hay filas y columnas. Las filas indican cada filtro guardado que existe. La primera columna indica el nombre del fichero donde se guarda el filtro. La segunda columna indica el nombre del grafo al que se refiere dicho filtro. Las cuatro columnas siguientes indican las coordenadas de los puntos que delimitan la zona a filtrar. La primera de estas 4 columnas indica la longitud geográfica del punto situado en la parte superior izquierda. La siguiente columna indica la latitud geográfica del punto situado en la parte superior izquierda. Las 2 siguientes columnas indican respectivamente la longitud y latitud geográfica del punto situado en la parte inferior derecha.

Para abrir un filtro determinado, hay que seleccionarlo en la tabla y después pulsar el botón “Open”. A continuación, se rellenarán los campos de texto de la ventana de Figura A.25 con los valores del filtro seleccionado.

Si lo que se quiere es eliminar un filtro de la lista, sólo hay que seleccionarlo y después pulsar el botón “Delete”. El filtro se eliminará y desaparecerá de la lista.

### A.12.3. Filtro geográfico: dibujar filtro

En la Figura A.25 la parte de la ventana recuadrada e indicada con el número 3 es la encargada de abrir una ventana con el mapa seleccionado para dibujar el área a filtrar. Si se pulsa el botón “Draw geographic bounds” se abrirá la siguiente ventana.



**Figura A.29 – Ventana para dibujar un filtro geográfico**

En esta ventana aparece dibujado el grafo del mapa seleccionado previamente. Existe una pequeña barra de herramientas que permite aceptar el filtro dibujado, cancelarlo y salir de la ventana, hacer zoom sobre el mapa, dibujar el área que indica el filtro geográfico o moverse con el mapa con el cursor.

Una vez dibujada el área que indica el filtro geográfico sólo hay que pulsar el botón “Accept” y se rellenarán los campos de texto referentes al filtro geográfico, indicados en la Figura A.25 con el número 1.

### **A.13. Cómo configurar una simulación a través del fichero XML**

Para configurar una simulación ya sea en modo gráfico o en modo *batch* existe la posibilidad de hacerlo a través de un fichero de configuración llamado “config.xml” que está en formato XML.

A continuación vamos a explicar, paso por paso, las acciones que hay que realizar para configurar una simulación a través del fichero de configuración “config.xml”:

1. **Acceder al fichero de configuración.** Supongamos que la carpeta del simulador se llama “Simulador”. El fichero de configuración “config.xml” está en la ruta “Simulador/CONFIG”.
2. **Modificar el fichero de configuración.** Abrir el fichero con un editor y modificar los valores de los campos que se quiera.
3. **Guardar el fichero.** Guardar los cambios realizados.
4. **Iniciar la simulación.** Ejecutar el simulador tanto en modo *batch* como en modo gráfico y los cambios se verá reflejados.

A continuación se van a explicar los parámetros de configuración que aparecen en el fichero de configuración “config.xml”

- **numberOfVehicles:** número de vehículos de la simulación.
- **mobilityStrategy:** estrategia de movilidad de los vehículos.
- **mediumDistanceRoute:** mínima distancia a la que se tiene que encontrar el punto destino del punto origen para considerarse válido en una estrategia de movilidad basada en rutas.
- **longDistanceRoute:** máxima distancia a la que se puede encontrar el punto destino del punto origen para considerarse válido en una estrategia de movilidad basada en rutas.
- **speedCar:** velocidad de los vehículos.
- **jumpStrategy:** estrategia de salto de los agentes móviles.
- **MobileCommunicationRange:** cobertura de los dispositivos móviles o lo que es lo mismo de los dispositivos integrados en los vehículos. Se indica en metros.
- **programName:** nombre del simulador.
- **version:** versión del simulador.

- **batch:** modo *batch* o modo gráfico.
- **simStatus:** estado de la simulación cuando se guardó el fichero de configuración.
- **viewGraphOnly:** visualizar solamente el grafo del mapa.
- **showNodeId:** mostrar o no el identificador de cada nodo del grafo del mapa.
- **showRoutes:** mostrar o no las rutas creadas por los vehículos.
- **noTiles:** mostrar las imágenes (*tiles*) del mapa.
- **zoomLevel:** nivel de zoom de la simulación.
- **logFileName:** nombre del fichero de registro o log.
- **logAgents:** nivel de configuración de log para los agentes (0 = ninguno, 1 = alguno, 2 = detallado).
- **logDevices:** nivel de configuración de log para los dispositivos (0 = ninguno, 1 = alguno, 2 = detallado).
- **logVehicles:** nivel de configuración de log para los vehículos (0 = ninguno, 1 = alguno, 2 = detallado).
- **iterTime:** tiempo entre cada iteración de la simulación.
- **numFixedDevices:** número de dispositivos fijos.
- **maxAgsFixedDev:** número máximo de agentes posibles en un dispositivo fijo.
- **maxAgsMobileDev:** número máximo de agentes posibles en un dispositivo móvil (de los vehículos).
- **initialAgents:** número inicial de agentes móviles.
- **itdelayAgentMove:** número de iteraciones de delay de los agentes.
- **itdelayAgentLocalClone:** número de iteraciones de delay de los agentes clonados utilizados en algunos tipos de simulaciones.
- **experiment:** tipo de experimento a realizar en la simulación. Es un número comprendido entre 1 y 6 se selecciona el tipo de experimento que se quiere realizar en la simulación. 1 = Monitoring, 2 = Query from fixed origin, 3 = Query from moving



origin, 4 = Query flooding from fixed origin, 5 = Query from moving origin within limited area, 6 = Query from fixed origin with limited area.

- **samplesX**: el área de monitorización se puede dividir en subáreas o celdas. Este parámetro indica el número de subáreas horizontales que habrá en la subdivisión.
- **samplesY**: el área de monitorización se puede dividir en subáreas o celdas. Este parámetro indica el número de subáreas verticales que habrá en la subdivisión.
- **maxMeasureIter**: número máximo de iteraciones que se monitorizará el área a monitorizar.
- **maxSimIter**: número máximo de iteraciones que se simulará.
- **probRelData**: número comprendido entre 0 y 100, que representa la probabilidad de un dispositivo de contener información relevante para una consulta.
- **processingTime**: iteraciones por vehículo con información relevante.
- **indexDevOrigin**: identificador del dispositivo en el que comienza el agente móvil.
- **iterStartReturn**: iteración en la que comienza la simulación.
- **fixedCommRange**: cobertura del dispositivo de comunicaciones de un elemento fijo. El valor se indica en metros.
- **maxData**: indica el tamaño máximo de la estructura en la que se guarda el grafo y las estructuras referentes a las rutas.
- **antennasEnabled**: usar o no los elementos fijos de soporte en las comunicaciones dentro de la simulación.

A continuación, se muestra un ejemplo de fichero de configuración.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<config>
  <numberOfVehicles>15</numberOfVehicles>
  <mobilityStrategy>Route</mobilityStrategy>
  <mediumDistanceRoute>520</mediumDistanceRoute>
  <longDistanceRoute>3000</longDistanceRoute>
  <speedCar>80</speedCar>
  <jumpStrategy>Random</jumpStrategy>
  <MobileCommunicationRange>100.0</MobileCommunicationRange>
  <programName>Mobile agents VANET Simulator</programName>
  <version> 0.5.10 (Dec 2009)</version>
  <batch>no</batch>
  <simStatus>Paused</simStatus>
  <viewGraphOnly>no</viewGraphOnly>
  <showNodeId>no</showNodeId>
  <showRoutes>yes</showRoutes>
  <noTiles>yes</noTiles>
  <zoomLevel>13</zoomLevel>
  <logFileName>vanetsim.log</logFileName>
  <logAgents>0</logAgents>
  <logDevices>0</logDevices>
  <logVehicles>0</logVehicles>
  <iterTime>300</iterTime>
  <numFixedDevices>1</numFixedDevices>
  <maxAgsFixedDev>100</maxAgsFixedDev>
  <maxAgsMobileDev>2</maxAgsMobileDev>
  <initialAgents>1</initialAgents>
  <itdelayAgentMove>2</itdelayAgentMove>
  <itdelayAgentLocalClone>0</itdelayAgentLocalClone>
  <experiment>5</experiment>
  <samplesX>1</samplesX>
  <samplesY>1</samplesY>
  <maxMeasureIter>1000</maxMeasureIter>
  <maxSimIter>19999</maxSimIter>
  <probRelData>50</probRelData>
  <processingTime>5</processingTime>
  <indexDevOrigin>1</indexDevOrigin>
  <iterStartReturn>165</iterStartReturn>
  <fixedCommRange>119.00</fixedCommRange>
  <maxData>200000</maxData>
  <antennasEnabled>no</antennasEnabled>
</config>
```

## A.14. Cómo importar trazas GPS

---

Una herramienta importante del simulador es la que permite importar trazas GPS obtenidas por otros medios ajenos al simulador y permite usarlas dentro del simulador. Esta herramienta es importante, ya que gracias a ella el simulador no es ajeno a otras herramientas y por ello está abierto hacia el exterior.

En esta sección se va a explicar cómo importar trazas GPS desde cero ya sea de formatos conocidos como desconocidos para el simulador. Para las trazas GPS de formato desconocido para el simulador se explicará lo que es necesario realizar para adaptar el simulador a otros tipos de formato.

### A.14.1. Importar trazas GPS de formato conocido

---

Como ya se ha comentado, el simulador es capaz de importar trazas GPS de los formatos .txt, .log y .plt. En el proyecto *GeoLife GPS Trajectories* [27, 28] se pueden encontrar una gran cantidad de ficheros de trazas GPS que fueron registradas durante 2 años en varios países del mundo, aunque la gran mayoría de ellas fueron registradas en Beijing (China).

Lo primero que es necesario hacer para importar trazas GPS es copiar aquellos ficheros de trazas GPS que se quieran importar en la carpeta "TRACKS". Supongamos que la carpeta del simulador se llama "Simulador", por lo tanto la ruta donde habrá que copiar los ficheros será "Simulador/TRACKS/".

A continuación, se ejecutará el simulador si no se había arrancado todavía y se abrirá la herramienta de importar trazas GPS a través del submenú "Import" que está dentro del menú "Tracks". Aparecerá la ventana de la Figura A.30.

En el apartado "Choose mobility track file" se deberá elegir una de las 2 opciones "All mobility track file" o "One mobility track file" dependiendo de si se quieren importar todas las trazas de la carpeta "TRACKS" o solamente un fichero de trazas. Si se quiere importar un único fichero de trazas se elegirá "One mobility track file" y a continuación se elegirá el fichero que se desee a través del menú desplegable.

Si se considera oportuno se puede realizar algún tipo de filtro con las opciones que aparecen en la sección "Filter" de esta misma ventana (ver Sección A.15 "Cómo definir y aplicar filtros a las trazas importadas"). Por último, para importar las trazas seleccionadas pulsar el botón "Import".

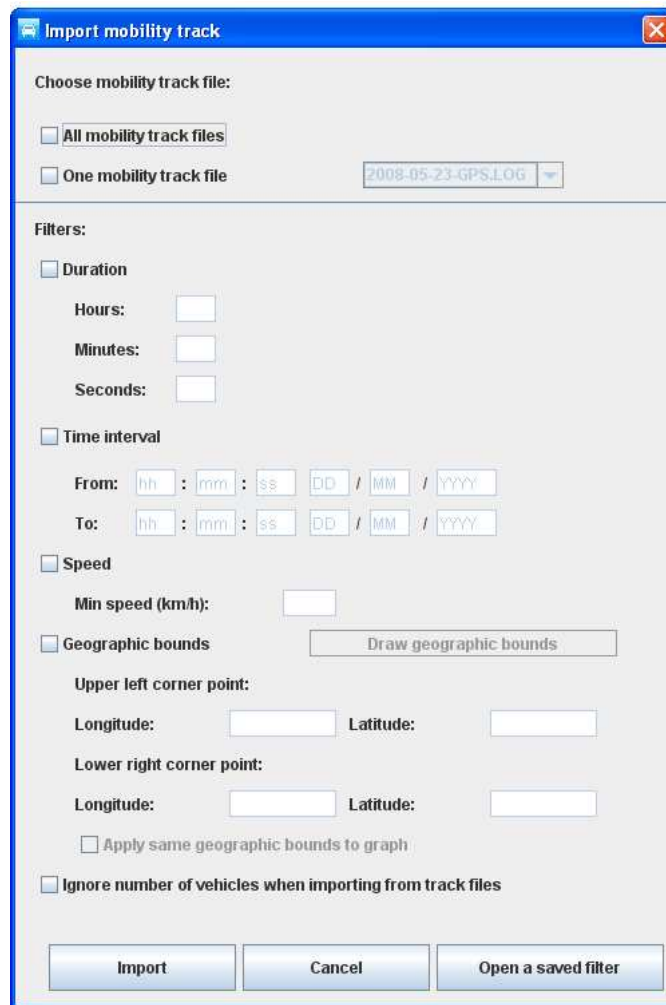


Figura A.30 – Ventana para importar trazas

Si las trazas importadas fueran de una zona geográfica diferente a la del mapa o tuvieran ciertos puntos de la traza fuera del mapa, el simulador avisaría de ello (ver Figura A.31).



Figura A.31 – Ventanas de aviso sobre los límites de las trazas

#### A.14.2. Importar trazas GPS de formato desconocido

Lo primero que es necesario hacer para importar trazas GPS es copiar aquellos ficheros de trazas GPS que se quieran importar en la carpeta “TRACKS”. Supongamos que la carpeta del simulador se llama “Simulador”, por lo tanto la ruta donde habrá que copiar los ficheros será “Simulador/TRACKS/”.

Después, debido a que el formato del fichero de las trazas es desconocido para el simulador, es necesario implementar la clase correspondiente que lea el fichero de trazas y las incorpore al simulador. Al final de esta sección se muestran dos ejemplos sobre cómo importar trazas de formatos desconocidos.

A continuación, se compilará el simulador y se ejecutará. Los pasos a seguir son los mismos que los explicados en la Sección A.14.1 “Importar trazas GPS de formato conocido”.

A continuación se muestran 2 ejemplos sobre formatos de ficheros de trazas desconocidos para el simulador y lo que habría que hacer para poder importarlos. El primero es un formato de trazas GPS reales (GPX) y el otro es un formato de salida de algunos simuladores que sirve de entrada para otros simuladores (tcl).

#### **GPX o GPS eXhange format**

El formato GPX [42, 43 y 44] está en XML y está pensado para transferir datos GPS entre aplicaciones. Se puede usar para describir puntos, recorridos y rutas. Si se tiene un terminal móvil con “Android,” se puede descargar del “Market” (tienda de aplicaciones para “Android”) la aplicación “My Tracks” que es capaz de registrar rutas en diversos formatos entre ellos GPX.

La estructura de un fichero de formato GPX para una traza es la siguiente:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<gpx ...>
/*Metadatos
<metadata> ... </metadata>
/*Datos
/*Ejemplos: Track
<trk>
<trkseg>
<trkpt lat="#" lon="#">
<ele>#</ele>
</trkpt>
<trkpt ...>
...
</trkpt>
</trkseg>
<trkseg>
<trkpt ...>
...
</trkpt>
</trkseg>
...
</trk>
/*Fin del fichero
</gpx>
```

La información que realmente nos interesa para la importación es la comprendida entre `<trk>` y `</trk>`, ya que indica un track o traza.

Por otra parte, lo que se encuentre entre `<trkseg>` y `</trkseg>` será la información relativa a un segmento de la traza. Toda la traza puede estar formada por un único segmento o por diversos segmentos. A un único vehículo se le puede asignar toda la traza con sus diversos segmentos o asignar cada segmento a un vehículo diferente.

A continuación se muestra un ejemplo de traza en formato GPX extraída de [43]. Dicha traza fue registrada por un dispositivo GPS modelo Garmin Oregon 400t.

```

<?xml version="1.0" encoding="UTF-8" standalone="no" ?>

<gpx xmlns="http://www.topografix.com/GPX/1/1" xmlns:gpxx="http://www.garmin.com/
xmlschemas/GpxExtensions/v3" xmlns:gpxtpx="http://www.garmin.com/xmlschemas/Track
PointExtension/v1" creator="Oregon
400t" version="1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.topografix.com/GPX/1/1
http://www.topografix.com/GPX/1/1/gpx.xsd
http://www.garmin.com/xmlschemas/GpxExtensions/v3
http://www.garmin.com/xmlschemas/GpxExtensionsv3.xsd
http://www.garmin.com/xmlschemas/TrackPointExtension/v1
http://www.garmin.com/xmlschemas/TrackPointExtensionv1.xsd">
  <metadata>
    <link href="http://www.garmin.com">
      <text>Garmin International</text>
    </link>
    <time>2009-10-17T22:58:43Z</time>
  </metadata>
  <trk>
    <name>Example GPX Document</name>
    <trkseg>
      <trkpt lat="47.644548" lon="-122.326897">
        <ele>4.46</ele>
        <time>2009-10-17T18:37:26Z</time>
      </trkpt>
      <trkpt lat="47.644548" lon="-122.326897">
        <ele>4.94</ele>
        <time>2009-10-17T18:37:31Z</time>
      </trkpt>
      <trkpt lat="47.644548" lon="-122.326897">
        <ele>6.87</ele>
        <time>2009-10-17T18:37:34Z</time>
      </trkpt>
    </trkseg>
  </trk>
</gpx>

```

La traza del ejemplo tiene el nombre de “Example GPX Document” y consta de un solo segmento. El segmento consta de 3 puntos los cuales contienen información acerca de las coordenadas geográficas (latitud y longitud) donde fueron registrados, altura o elevación y fecha y hora a la que fue registrado el punto.

En el ejemplo siguiente, el primer punto tiene la siguientes coordenadas geográficas: latitud = 47.644548, longitud = -122.326897, elevación = 4.94, fecha = 17/10/2009 y hora = 18:37:31. La fecha y hora está en UTC (Universal Coordinated Time). El formato de la fecha es YYYY-MM-DD y el de la hora hh:mm:ss. La ISO 8601 sugiere separar la fecha de la hora con el carácter de la letra ‘T’ cuando van seguidos. Al final de la hora se concatena el carácter de la letra ‘Z’ para indicar que la hora es en UTC

```
<trkpt lat="47.644548" lon="-122.326897">  
  <ele>4.94</ele>  
  <time>2009-10-17T18:37:31Z</time>  
</trkpt>
```

Una vez explicado el formato GPX se pasa a explicar el proceso que habría que realizar para que el simulador pudiera importar este formato.

Existe una clase abstracta llamada “TrackReader” de la que derivan otras clases que son las encargadas de leer los ficheros de trazas de diferentes formatos (TrackReader\_plt, TrackReader\_txt y TrackReader\_log). La clase abstracta “TrackReader” tiene tres métodos que heredan sus clases hijas y un método abstracto que cada clase hija debe implementar.

El método abstracto se llama “trackFileReader” y es el encargado de recorrer el fichero de trazas leyendo sus datos para conformar la estructura que necesita el vehículo para seguir la traza de movimiento que se importe.

Los otros métodos que se heredan son:

- “degreesToDecimal”: este método convierte las coordenadas geográficas en valores decimales. Para ello hay que introducir la coordenada en grados y segundos e indicar su orientación.
- “restartTrackData”: este método inicializa los atributos de la traza.
- “testFilterAndAddNode”: este método analiza cada nodo de la traza respecto al filtro seleccionado si existe, y si pasa el filtro lo añade a la estructura donde se guarda la traza que utilizará el vehículo. Además, es capaz de calcular la velocidad a partir del nodo siguiente ya que en algunos formatos no se indica la velocidad.

Por lo tanto, habría que implementar una clase que, por ejemplo, se llamara “TrackReader\_gpx” que fuera hija de la clase abstracta “TrackReader” y que implementara el método abstracto “trackFileReader”. Este método debería recorrer el fichero XML con la información de las trazas GPS y extraer de cada punto la información de sus coordenadas geográficas y su fecha y hora. La fecha y hora se necesitan para calcular la velocidad y para poder aplicar los filtros temporales. Después, en la clase “Track”, habría que añadir el código necesario para indicar que cuando el simulador intente importar un fichero de trazas de formato GPX use la clase “TrackReader\_gpx”.



**tcl**

El formato tcl [45, 46, 47] es el usado por el simulador de red Ns-2 para importar movimientos de simuladores de movilidad. Existen simuladores de movilidad que son capaces de exportar sus trazas de movilidad en el formato tcl para que Ns-2 sea capaz de importarlas. Algunos de los simuladores que exportan sus trazas de movimiento en este formato son TraNSLite, VanetMobiSim y MobiSim. A continuación, se explica dicho formato:

Al principio aparece una gran cantidad de información destinada al simulador Ns-2, por lo que no viene al caso entrar a explicarla. De toda esta información inicial la única que resulta interesante para importar las trazas es el tamaño de la pantalla ya que la posición de los vehículos viene dada en coordenadas de pantalla y el número de nodos móviles o vehículos. A continuación, usaremos nodos móviles o vehículos indistintamente, no hay que confundirlos con los nodos o puntos de la traza.

```
set opt(x) 602      ;# x coordinate of topology
set opt(y) 702      ;# y coordinate of topology

set val(nn) 25      ;# number of mobilenodes
```

La siguiente información que aparece que nos incumbe es la referente a la posición inicial de cada vehículo. Aquí aparece el identificador de cada vehículo seguido de las coordenadas de pantalla (x,y,z) que ocupa al iniciar la simulación.

```
$node_(0) set X_ 0.0050
$node_(0) set Y_ 695.0
$node_(0) set Z_ 0.0
$node_(0) setdest 0.0050 695.0 1
```

En el ejemplo anterior el nodo 0 comenzará su movimiento en la posición (0.0050, 695.0, 0.0). La última línea indica de otra forma la posición inicial e inicializa la velocidad a 1m/s.

Una vez que se ha indicado la posición inicial de todos los nodos móviles, comienza la traza de movimiento para cada uno. En cada línea aparece la siguiente posición a ocupar por un nodo móvil. Además aparece la velocidad del nodo, el identificador del nodo y el tiempo en el que se produce este movimiento.

```
$ns_ at 1.0 "$node_(0) setdest 0.0050 693.0 1.96"
$ns_ at 2.0 "$node_(0) setdest 1.0E-6 688.000001 4.11"
$ns_ at 3.0 "$node_(0) setdest 2.0E-6 683.000002 5.81"
$ns_ at 4.0 "$node_(0) setdest 3.0E-6 676.000003 7.12"
$ns_ at 5.0 "$node_(0) setdest 4.0E-6 667.000004 8.53"
```

En el ejemplo anterior, el nodo 0, en el intervalo de tiempo 1, se moverá hacia la posición (0.0050, 693) a una velocidad de 1.96 m/s. Este mismo nodo, en el intervalo de tiempo 2 se moverá hacia la posición (1.0E-6 688.000001) a una velocidad de 4.11 m/s etc.

A continuación, se muestra un breve ejemplo de una traza de movimiento de 3 nodos móviles.

```

set val(nn) 3      ;# number of mobilenodes
set opt(x) 650     ;# x coordinate of topology
set opt(y) 760     ;# y coordinate of topology

$node_(0) set X_ 0.0050
$node_(0) set Y_ 695.0
$node_(0) set Z_ 0.0
$node_(0) setdest 0.0050 695.0 1
$node_(1) set X_ 1.0E-6
$node_(1) set Y_ 695.000001
$node_(1) set Z_ 0.0
$node_(1) setdest 1.0E-6 695.000001 1
$node_(2) set X_ 2.0E-6
$node_(2) set Y_ 695.000002
$node_(2) set Z_ 0.0
$node_(2) setdest 2.0E-6 695.000002 1

$ns_ at 1.0 "$node_(0) setdest 0.0050 693.0 1.96"
$ns_ at 2.0 "$node_(0) setdest 1.0E-6 688.000001 4.11"
$ns_ at 3.0 "$node_(0) setdest 2.0E-6 683.000002 5.81"
$ns_ at 4.0 "$node_(0) setdest 3.0E-6 676.000003 7.12"
$ns_ at 5.0 "$node_(0) setdest 4.0E-6 667.000004 8.53"
$ns_ at 6.0 "$node_(0) setdest 4.9999999999999996E-6 657.000005 9.84"
$ns_ at 7.0 "$node_(0) setdest 6.0E-6 645.000006 11.64"
$ns_ at 8.0 "$node_(0) setdest 7.0E-6 632.000007 13.24"
$ns_ at 9.0 "$node_(0) setdest 8.0E-6 617.000008 15.11"
$ns_ at 10.0 "$node_(0) setdest 9.0E-6 600.000009 17.12"
$ns_ at 11.0 "$node_(1) setdest 9.999999999999999E-6 692.00001 2.22"
$ns_ at 11.0 "$node_(0) setdest 1.1E-5 582.000011 18.36"
$ns_ at 12.0 "$node_(1) setdest 1.2E-5 688.000012 4.55"
$ns_ at 12.0 "$node_(0) setdest 1.3E-5 562.000013 19.39"
$ns_ at 13.0 "$node_(1) setdest 1.4E-5 681.000014 6.94"
$ns_ at 13.0 "$node_(0) setdest 1.4999999999999999E-5 542.000015 20.65"
$ns_ at 14.0 "$node_(1) setdest 1.6E-5 672.000016 8.91"
$ns_ at 14.0 "$node_(0) setdest 1.7E-5 526.000017 16.02"
$ns_ at 15.0 "$node_(1) setdest 1.8E-5 661.000018 11.04"
$ns_ at 15.0 "$node_(0) setdest 1.8999999999999998E-5 515.000019 11.08"
$ns_ at 16.0 "$node_(1) setdest 1.9999999999999998E-5 649.00002 12.22"
$ns_ at 16.0 "$node_(0) setdest 2.1E-5 502.000021 12.99"
$ns_ at 17.0 "$node_(1) setdest 2.2E-5 634.000022 14.13"
$ns_ at 17.0 "$node_(0) setdest 6.000023 500.000023 8.30"
$ns_ at 18.0 "$node_(1) setdest 2.4E-5 619.000024 15.64"
$ns_ at 18.0 "$node_(0) setdest 16.000025 500.000025 10.32"
$ns_ at 19.0 "$node_(1) setdest 2.6E-5 602.000026 16.87"
$ns_ at 19.0 "$node_(0) setdest 29.000027 500.000027 12.44"
$ns_ at 20.0 "$node_(1) setdest 2.8E-5 584.000028 18.21"
$ns_ at 20.0 "$node_(0) setdest 42.000029 500.000029 13.84"
$ns_ at 21.0 "$node_(2) setdest 2.9999999999999997E-5 693.00003 1.85"
$ns_ at 21.0 "$node_(1) setdest 3.1E-5 564.000031 19.49"
$ns_ at 21.0 "$node_(0) setdest 57.000032 500.000032 14.99"
$ns_ at 22.0 "$node_(2) setdest 3.2999999999999996E-5 689.000033 3.22"

```

Una vez explicado el formato del fichero habría que adaptar el simulador para que sea capaz de importar este tipo de fichero. La adaptación a este formato es algo más compleja que la adaptación al formato GPX debido a los siguientes elementos especiales a considerar:

- La posición inicial viene indicada en el fichero con 3 coordenadas a pesar de que después sólo se usan 2 coordenadas. El simulador del PFC usa 2 coordenadas (x,y), por lo tanto en la posición inicial habría que omitir la coordenada z.
- En las trazas aparece un intervalo de tiempo junto a cada movimiento. En el simulador del PFC los vehículos se rigen por posiciones y velocidad, y por lo tanto esta información, en principio, no se utilizaría.
- Aparecen las trazas de diferentes vehículos mezcladas. En los formatos que reconoce el simulador pueden aparecer varias trazas en un mismo fichero, pero éstas aparecen separadas, es decir, primero aparecen todos los puntos de una traza a continuación aparece un indicador que avisa el comienzo de una nueva traza y después comienzan los puntos de la siguiente traza. En el formato tcl, como se puede ver en el ejemplo, aparecen los puntos de las diferentes trazas mezclados. Por lo tanto habría que ir almacenando los puntos de cada traza en una estructura diferente o hacer tantas lecturas del fichero como trazas diferentes existan y en cada lectura centrarse en una traza. El número de trazas es el mismo que el de nodos móviles ya que cada traza corresponde a un nodo móvil. Como se ha explicado anteriormente, cada punto de la traza está identificado con un nodo móvil.
- La velocidad viene indicada en m/s y habría que convertirla a km/h ya que es la unidad de la velocidad del simulador.
- Las posiciones de los puntos de la traza vienen indicadas en coordenadas de pantalla. En el simulador del PFC se trabaja sobre coordenadas geográficas (latitud y longitud) por lo que sería necesario adaptar las coordenadas de pantalla a coordenadas geográficas. Para realizar esta adaptación habría que usar el tamaño de la pantalla que viene indicado en el fichero tcl, el nivel de zoom y el mapa sobre el que se quieren adaptar. Lo malo que tiene usar coordenadas de pantalla es que los movimientos de los nodos móviles sólo tienen sentido sobre el entorno que se han generado. Por otra parte, tienen la ventaja de que se pueden adaptar a cualquier mapa aunque no respeten la topología (calles, carreteras...) de ese mapa.

- Para poder aplicar el filtro temporal de intervalo de tiempo sería necesario proveer a cada punto de la traza con una fecha y hora. Sucede lo mismo para poder aplicar el filtro temporal de duración. Estos filtros se podrían aplicar ya que se dispone de la velocidad de un punto hasta otro y el simulador ha convertido las coordenadas de pantalla en coordenadas geográficas. Para ello habría que asignar una fecha y hora inicial a la traza.

Para reconocer este formato habría que adaptar el simulador de una forma similar a la explicada en el formato GPX.

### **A.15. Cómo definir y aplicar filtros a las trazas importadas**

A través de “Import”, situado en el menú “Tracks”, se abre la herramienta encargada de importar trazas de movimiento. Esta ventana consta de 2 partes: la destinada a seleccionar los ficheros de trazas para importar y la parte en la que se pueden definir los filtros que se aplicarán a las trazas importadas.

En la parte destinada a definir los filtros aparece en la parte inferior la opción “Ignore number of vehicles when importing from track files” que gestiona la posible restricción a la hora de importar las trazas dependiendo del número de vehículos.

Pueden suceder los siguientes casos:

- Número de vehículos ( $V$ ) > número de trazas obtenidas ( $T$ ).
- $V < T$ .

Dependiendo de la configuración que haya elegido el usuario se actuará de una forma u otra. Si sucede el primer caso ( $V > T$ ), se avisará al usuario de ello. Si sucede el segundo caso ( $V < T$ ), se avisará al usuario de ello y si el usuario ha decidido ignorar el número de vehículos, se crearán tantos vehículos con su respectiva traza como trazas obtenidas ( $T$ ). Mientras que si el usuario no ha decidido ignorar el número de vehículos, se usarán  $V$  vehículos y  $V$  trazas (las  $V$  primeras trazas leídas).

Además de estos casos especiales, existen varios filtros que criban las trazas leídas. Los filtros que se pueden aplicar a las trazas son los siguientes:

- **Filtro temporal:** se puede elegir el filtro temporal de duración o el filtro temporal de intervalo de tiempo. El filtro de duración admite puntos de la traza comprendidos entre la fecha inicial ( $f1$ ) del primer punto y la fecha final determinada por la suma de la fecha inicial y la duración elegida ( $f2 = f1 + \text{duración}$ ). La duración la indica el usuario en horas, minutos y segundos (ver

Figura A.32, cuadrado número 1) El filtro de intervalo de tiempo admite puntos de la traza comprendidos entre la fecha inicial (“From”) y la fecha final (“To”) determinada por el usuario. (Figura A.32, cuadrado número 2).

- **Filtro de velocidad:** con este filtro se indica la velocidad mínima que debe tener un punto de la traza para ser aceptado e importado. (Figura A.32, cuadrado número 3)
- **Filtro geográfico:** este filtro es similar al filtro geográfico aplicado a los mapas. Para aplicar este filtro hay que indicar el área geográfica que se desea que vendrá determinada por dos puntos (uno en la posición superior izquierda y otro en la inferior derecha). Los puntos de la traza que estén situados dentro de esa área serán aceptados e importados. (Figura A.32, cuadrado número 4).

Import mobility track

Choose mobility track file:

☐ All mobility track files

☐ One mobility track file 2008-05-23-GPS.LOG

Filters:

☐ Duration

Hours:

Minutes:

Seconds:

☐ Time interval

From:  :  :   /  /

To:  :  :   /  /

☐ Speed

Min speed (km/h):

☐ Geographic bounds Draw geographic bounds

Upper left corner point:

Longitude:  Latitude:

Lower right corner point:

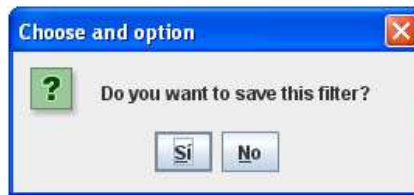
Longitude:  Latitude:

☐ Apply same geographic bounds to graph

☐ Ignore number of vehicles when importing from track files

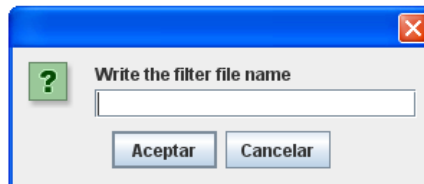
Figura A.32 – Ventana para importar trazas con filtros remarcados

Una vez que se haya indicado el filtro y pulsado el botón “Import” se mostrará la siguiente ventana preguntando si se desea guardar el filtro seleccionado.



**Figura A.33 – Ventana pregunta para guardar filtro**

Si se quiere guardar el filtro indicado, aparecerá la siguiente ventana para que se le indique el nombre del filtro.

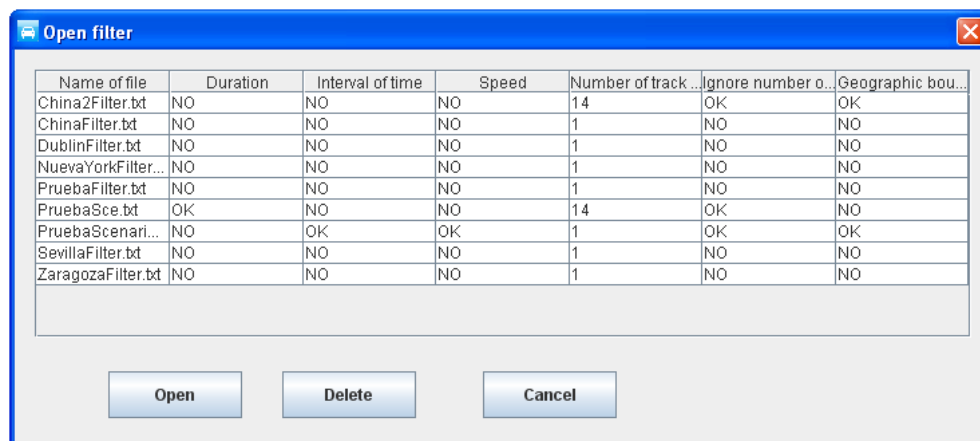


**Figura A.34 – Ventana introducir nombre del filtro**

A continuación se muestra cómo abrir un filtro guardado.

#### A.15.1. Abrir filtro

En la Figura A.32 el botón “Open a saved filter” situado en la parte inferior derecha de la ventana es el encargado de abrir un filtro guardado. Si se pulsa el botón se abrirá la siguiente ventana.



**Figura A.35 – Ventana con los filtros disponibles de las trazas**

En la tabla que aparece hay filas y columnas. Las filas indican cada filtro guardado que existe. La primera columna indica el nombre del fichero donde se guarda el filtro. La segunda columna indica si en el fichero hay un filtro de duración. La tercera columna indica si en el fichero hay un filtro de intervalo de tiempo. La cuarta columna indica si en el fichero hay un filtro de velocidad. La quinta columna indica el número de ficheros de trazas al que afecta el filtro del fichero cuando fue

guardado. Si el número que aparece en esta columna es mayor que uno quiere decir que el filtro afectó a todos los ficheros de trazas que existían en la carpeta “TRACKS”. Esto se debe a que, dependiendo de la opción seleccionada por el usuario, un filtro puede afectar a un fichero de trazas GPS o a todos los ficheros de la carpeta “TRACKS”. La sexta columna indica si se ignora el número de vehículos a la hora de importar las trazas. Por último, la última columna indica si existe en el fichero un filtro geográfico.

Para abrir un filtro determinado, hay que seleccionarlo en la tabla y después pulsar el botón “Open”. A continuación, se rellenarán automáticamente los campos de texto de la ventana de la Figura A.32 con los valores del filtro seleccionado.

Si lo que se quiere es eliminar un filtro de la lista, sólo hay que seleccionarlo y después pulsar el botón “Delete”. El filtro se eliminará y desaparecerá de la lista.

#### A.15.2. Filtros geográficos

---

Existen 2 formas de indicar un filtro geográfico:

- Escribiendo la latitud y longitud de los 2 puntos que delimitan el área que será el filtro geográfico.
- Dibujando el área que será el filtro geográfico.

##### **Definir filtro geográfico de forma manual**

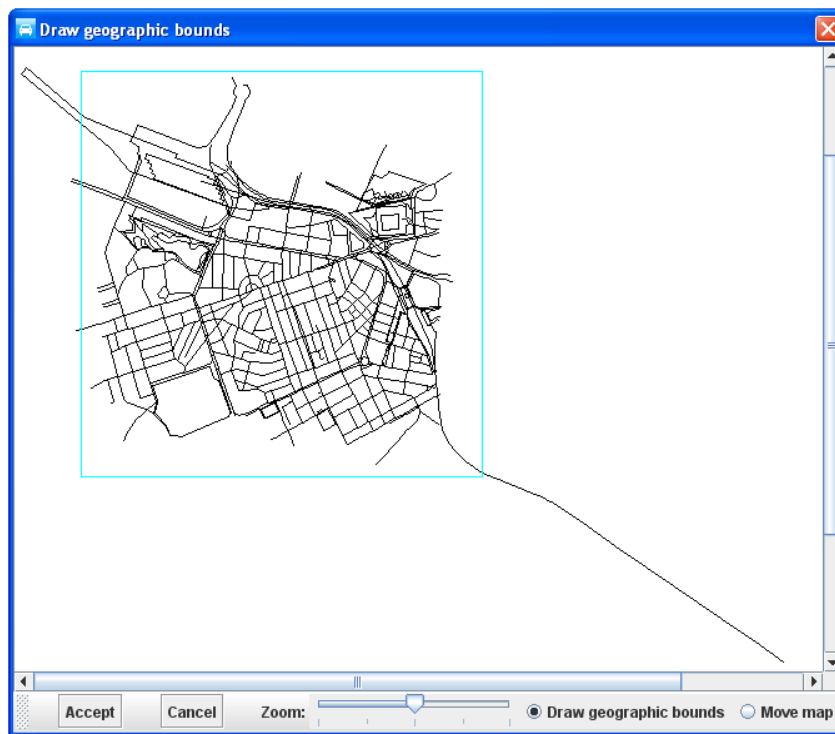
En la Figura A.32 la parte de la ventana recuadrada e indicada con el número 4 es la encargada del filtro geográfico.

Los cuatro campos de texto son los encargados de definir el área que se va a filtrar. Por un lado encontramos “Upper Left corner point” y por otro “Lower right corner”. Es decir, se indica la zona que nos interesa con 2 puntos, uno situado en la posición superior izquierda y el otro en la posición inferior derecha.

Hay que tener en cuenta que la latitud crece hacia el polo norte. Por tanto, la latitud del punto superior izquierdo es mayor que la latitud del punto inferior derecho. Una buena herramienta para conocer las coordenadas geográficas de un lugar es [14].

### **Definir filtro geográfico dibujando**

En la Figura A.32 el botón “Draw geographic bounds” de la parte superior derecha del recuadrado rojo con el número 4 es el encargado de abrir una ventana con el mapa indicado del filtro seleccionado para dibujar el área a filtrar. Si se pulsa el botón “Draw geographic bounds” se abrirá la siguiente ventana.



**Figura A.36 – Ventana para dibujar filtro geográfico**

En esta ventana aparece dibujado el grafo del mapa cargado para importar las trazas. Existe una pequeña barra de herramientas que permite aceptar el filtro dibujado, cancelarlo y salir de la ventana, hacer zoom sobre el mapa, dibujar el área que indica el filtro geográfico o moverse por el mapa con el cursor.

Una vez dibujada el área que indica el filtro geográfico sólo hay que pulsar el botón “Accept” y se rellenarán los campos de texto referentes al filtro geográfico.



## A.16. Cómo añadir nodos fijos de soporte de comunicaciones

En esta sección se va a explicar de forma detallada cómo añadir elementos fijos de soporte de las comunicaciones entre los vehículos.

Lo primero que hay que hacer es conocer el formato de un fichero de nodos fijos de soporte de comunicaciones. En el Anexo B.4 “Fichero de los elementos fijos de soporte” se detalla el formato de un fichero de este tipo. Una vez que se conoce cómo es el fichero, se puede modificar uno ya existente o crear uno nuevo.

Para ello, necesitamos saber dónde encontrar un fichero de este tipo o dónde guardarlo. Supongamos que se quiere modificar un fichero ya existente para el mapa llamado “Zaragoza”. Para ello deberemos ir hasta la carpeta del mapa de Zaragoza y modificar el fichero ya existente. La ruta es la siguiente, si consideramos que la carpeta del simulador se llama “Simulador”, “Simulador/GRAPH/Zaragoza”. Y el fichero se llama siempre “Antennas.txt”, ya que si se modifica el nombre, el simulador no lo encontrará.

Una vez modificado el fichero, se deberá activar la opción que indica al simulador que debe añadir los nodos fijos de soporte en la simulación. Esto se puede hacer a través del fichero de configuración “config.xml” modificando el parámetro “antennasEnabled” y después arrancado el simulador. Con el simulador arrancado, también se puede parar la simulación actual y activar el parámetro “Antennas enabled” del menú de configuración de simulación (ver Figura A.37); después se debe arrancar la simulación y el cambio tendrá efecto.

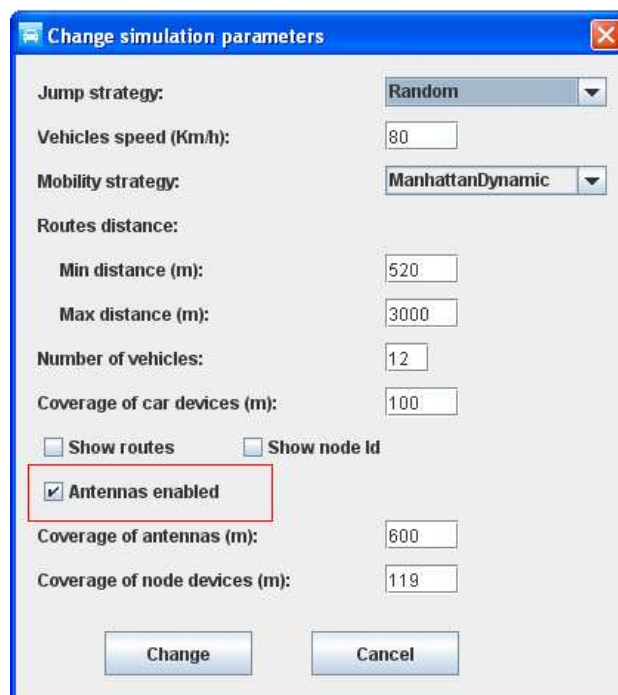


Figura A.37 – Ventana para añadir las “antenas”



## ANEXO B – FORMATOS DE LOS FICHEROS

---

Debido a la existencia de diversos ficheros con sus respectivos formatos se ha optado por explicar el formato de cada uno de ellos para que se entiendan mejor. Además, se ha optado por organizarlo en un anexo para tener toda esta información agrupada en una misma sección. El sistema gestiona 8 tipos de ficheros, de los que 2 son en formato XML:

- **Fichero de configuración:** explicado en la Memoria en la Sección 4.2.2 “Sistema de carga y almacenamiento de parámetros”. En este fichero se guardan los parámetros de simulación en formato XML. Este fichero se llama “config.xml”. Para más información consúltase la Sección B.1 “Fichero de configuración”.
- **Fichero de información OSM:** está en formato XML y contiene la información relativa a las vías de circulación de la zona seleccionada (ver Sección 3.5 “OpenStreetMap” de la Memoria). El fichero tendrá el nombre que el usuario indicó cuando descargó el mapa. Para más información consúltase la Sección B.2 “Fichero de información OSM (OpenStreetMap)”.
- **Fichero de área de monitorización:** éste es un fichero de texto en el que se almacena la información relativa al área de monitorización de un mapa determinado. Cada mapa descargado tiene su propio fichero de área de monitorización. De esta forma no es necesario dibujar el área de monitorización cada vez que se realiza una simulación, sino que el simulador carga automática el área guardada en dicho fichero (consultar la Sección B.3 “Fichero de las áreas de monitorización”). Este fichero se llama “MonitoredArea.txt”.
- **Fichero de dispositivos fijos de soporte:** este fichero de texto contiene la información sobre los dispositivos fijos de soporte para las comunicaciones entre vehículos explicados en la Memoria en la Sección 4.2.6 “Nodos de soporte para las comunicaciones”. Cada mapa tiene su propio fichero de dispositivos fijos de soporte. Este fichero se llama “Antennas.txt”. Para ver el formato de este fichero consúltase la Sección B.4 “Fichero de los elementos fijos de soporte”.
- **Fichero de información de los mapas descargados:** este fichero de texto almacena información sobre los mapas descargados. Contiene información sobre el nombre del mapa, tipo de mapa, tamaño y nivel de zoom máximo del mismo. Para ver el formato de este fichero consúltase la Sección B.5 “Fichero de información de los mapas descargados”. Este fichero se llama “downloadedMaps.txt”.
- **Fichero de trazas GPS:** existen 3 tipos diferentes de formato de fichero de trazas GPS (.txt, .log y .plt) que reconoce el simulador de momento. Para más información consúltase la Sección B.7 “Fichero de trazas GPS reales”.
- **Fichero de registro:** este fichero de texto contiene la información sobre lo hechos que suceden en el simulador. Lo referente a este fichero ya estaba desarrollado en el prototipo anterior. El fichero se llama “vanetsim.log”. Para más información consúltase la Sección B.6 “Fichero de registro (log)”.
- **Fichero para guardar los filtros:** en estos ficheros se guardan los filtros geográficos aplicados a los mapas y los filtros temporales (de duración y

de intervalo de tiempo) de velocidad y geográficos aplicados a las trazas GPS. Para más información consúltase la Sección B.8 “Ficheros de filtros”.

Por lo tanto, a continuación se explica el formato de cada fichero usado en este simulador.

## B.1. Fichero de configuración

---

Como ya se ha comentado en numerosas ocasiones, existe un fichero de configuración que permite modificar los parámetros de configuración de una simulación. Este fichero se llama “config.xml” y está en formato XML. Se puede encontrar en la carpeta “CONFIG” del simulador.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<config>
  <numberOfVehicles>15</numberOfVehicles>
  <mobilityStrategy>Route</mobilityStrategy>
  <mediumDistanceRoute>520</mediumDistanceRoute>
  <longDistanceRoute>3000</longDistanceRoute>
  <speedCar>80</speedCar>
  <jumpStrategy>Random</jumpStrategy>
  <MobileCommunicationRange>100.0</MobileCommunicationRange>
  <programName>Mobile agents VANET Simulator</programName>
  <version> 0.5.10 (Dec 2009)</version>
  <batch>no</batch>
  <simStatus>Paused</simStatus>
  <viewGraphOnly>no</viewGraphOnly>
  <showNodeId>no</showNodeId>
  <showRoutes>yes</showRoutes>
  <noTiles>yes</noTiles>
  <zoomLevel>13</zoomLevel>
  <logFileName>vanetsim.log</logFileName>
  <logAgents>0</logAgents>
  <logDevices>0</logDevices>
  <logVehicles>0</logVehicles>
  <iterTime>300</iterTime>
  <numFixedDevices>1</numFixedDevices>
  <maxAgsFixedDev>100</maxAgsFixedDev>
  <maxAgsMobileDev>2</maxAgsMobileDev>
  <initialAgents>1</initialAgents>
  <itdelayAgentMove>2</itdelayAgentMove>
  <itdelayAgentLocalClone>0</itdelayAgentLocalClone>
  <experiment>5</experiment>
  <samplesX>1</samplesX>
  <samplesY>1</samplesY>
  <maxMeasureIter>1000</maxMeasureIter>
  <maxSimIter>19999</maxSimIter>
  <probRelData>50</probRelData>
  <processingTime>5</processingTime>
  <indexDevOrigin>1</indexDevOrigin>
  <iterStartReturn>165</iterStartReturn>
  <fixedCommRange>119.00</fixedCommRange>
  <maxData>200000</maxData>
  <antennasEnabled>no</antennasEnabled>
</config>
```

A continuación, se explican todos los elementos del fichero de configuración:

- **numberOfVehicles**: número de vehículos de la simulación.
- **mobilityStrategy**: estrategia de movilidad de los vehículos.
- **mediumDistanceRoute**: mínima distancia a la que se tiene que encontrar el punto destino del punto origen para considerarse válido en una estrategia de movilidad basada en rutas.
- **longDistanceRoute**: máxima distancia a la que se puede encontrar el punto destino del punto origen para considerarse válido en una estrategia de movilidad basada en rutas.
- **speedCar**: velocidad de los vehículos.
- **jumpStrategy**: estrategia de salto de los agentes móviles.
- **mobileCommunicationRange**: cobertura de los dispositivos móviles, o lo que es lo mismo, de los dispositivos integrados en los vehículos. Se indica en metros.
- **programName**: nombre del simulador.
- **version**: versión del simulador.
- **batch**: modo *batch* o modo gráfico.
- **simStatus**: estado de la simulación cuando se guardó el fichero de configuración.
- **viewGraphOnly**: visualizar solamente el grafo del mapa.
- **showNodeId**: mostrar o no el identificador de cada nodo del grafo del mapa.
- **showRoutes**: mostrar o no las rutas creadas por los vehículos.
- **noTiles**: mostrar las imágenes (*tiles*) del mapa.
- **zoomLevel**: nivel de zoom de la simulación.
- **logFileName**: nombre del fichero de registro o log.
- **logAgents**: nivel de configuración de log para los agentes (0 = ninguno, 1 = alguno, 2 = detallado).

- **logDevices:** nivel de configuración de log para los dispositivos (0 = ninguno, 1 = alguno, 2 = detallado).
- **logVehicles:** nivel de configuración de log para los vehículos (0 = ninguno, 1 = alguno, 2 = detallado).
- **iterTime:** tiempo entre cada iteración de la simulación.
- **numFixedDevices:** número de dispositivos fijos.
- **maxAgsFixedDev:** número máximo de agentes posibles en un dispositivo fijo.
- **maxAgsMobileDev:** número máximo de agentes posibles en un dispositivo móvil (de los vehículos).
- **initialAgents:** número inicial de agentes móviles.
- **itdelayAgentMove:** número de iteraciones de delay de los agentes.
- **itdelayAgentLocalClone:** número de iteraciones de delay de los agentes clonados utilizados en algunos tipos de simulaciones.
- **experiment:** tipo de experimento a realizar en la simulación. Es un número comprendido entre 1 y 6 se selecciona el tipo de experimento que se quiere realizar en la simulación. 1 = Monitoring, 2 = Query from fixed origin, 3 = Query from moving origin, 4 = Query flooding from fixed origin, 5 = Query from moving origin within limited area, 6 = Query from fixed origin with limited area.
- **samplesX:** el área de monitorización se puede dividir en subáreas o celdas. Este parámetro indica el número de subáreas horizontales que habrá en la subdivisión.
- **samplesY:** el área de monitorización se puede dividir en subáreas o celdas. Este parámetro indica el número de subáreas verticales que habrá en la subdivisión.
- **maxMeasureIter:** número máximo de iteraciones que se monitorizará el área a monitorizar.
- **maxSimIter:** número máximo de iteraciones que se simulará.
- **probRelData:** número comprendido entre 0 y 100, que representa la probabilidad de un dispositivo de contener información relevante para una consulta.

- **processingTime**: iteraciones por vehículo con información relevante.
- **indexDevOrigin**: identificador del dispositivo en el comienza el agente móvil.
- **iterStartReturn**: iteración en la que comienza la simulación.
- **fixedCommRange**: cobertura del dispositivo de comunicaciones de un elemento fijo. El valor se indica en metros.
- **maxData**: indica el tamaño máximo de la estructura en la que se guarda el grafo y las estructuras referentes a las rutas.
- **antennasEnabled**: usar o no los elementos fijos de soporte en las comunicaciones dentro de la simulación.

## B.2. Fichero de información OSM (OpenStreetMap)

---

La información de las carreteras y calles de los mapas se extrae del fichero OSM en formato XML que se obtiene cuando se descarga un mapa. Esta información es la que se usa para componer el grafo del mapa sobre el que se mueven los vehículos. Por tanto, esta información se utiliza para las estrategias de movilidad de los vehículos, ya que sin ella no sería posible, por ejemplo, calcular una ruta desde un punto origen hasta otro destino. A continuación, se muestran los elementos importantes de un fichero en formato OSM. Estos ficheros se almacenan en la carpeta con el nombre que se eligió al descargar el mapa dentro de la carpeta “GRAPH”.

### Nodos

Los nodos (*nodes*) son los elementos que conforman las vías (carreteras, calles, etc.). Estos elementos contienen una posición geográfica fija. Se usan para indicar posiciones de interés o formar vías. Tienen la siguiente apariencia:

```
<node id="156804" lat="61.8083953857422" lon="10.8497076034546" visible="true"
timestamp="2005-07-30T14:27:12+01:00"/>
```

Los nodos pueden tener etiquetas:

```
<node id="156804" lat="61.8083953857422" lon="10.8497076034546" visible="true"
timestamp="2005-07-30T14:27:12+01:00"> <tag k="tourism" v="hotel" />
<tag k="name" v="Cockroach Inn" />
</node>
```

## Vías

Las vías (*ways*) representan una lista ordenada de nodos. Tienen que tener como mínimo dos nodos, y normalmente tienen etiquetas (*tags*) para especificar el significado de la vía (carretera, río, autopista, etc.).

```
<way id="35" visible="true" timestamp="2006-03-14T10:07:23+00:00" user="johnz">
  <nd ref="156804"/>
  <nd ref="156805"/>
  <nd ref="156806"/>
  <tag k="highway" v="secondary"/>
</way>
```

## Relaciones

Las relaciones (*relations*) se usan para modelar un tipo de relación entre objetos, y también pueden representar un objeto ellas mismas. Las relaciones, normalmente, tienen uno o más miembros y varias etiquetas, de las cuales una designada con “type” especifica el tipo de relación. Los miembros pueden ser nodos, vías u otras relaciones. Las relaciones sin miembros o etiquetas también están permitidas. La lista está desordenada.

```
<relation id="77" visible="true" timestamp="2006-03-14T10:07:23+00:00" user="fred">
  <member type="way" ref="343" role="from" />
  <member type="node" ref="911" role="at" />
  <member type="way" ref="227" role="to" />
  <tag k="type" v="turn_restriction"/>
</relation>
```

El atributo “ref” indica el identificador del objeto. Con el atributo “type” y el atributo “ref” se identifica el objeto referenciado. El atributo “role” se usa opcionalmente para especificar la función que desempeña en la relación.

Con estas relaciones y con la información de las vías se establecen restricciones en el tráfico. Por ejemplo, se puede indicar si una vía es de sentido único o de doble sentido. Toda esta información se podría usar para modelar una estrategia de movilidad que respetara ciertas normas de tráfico y por lo tanto que fuera más realista.

En el fichero primero aparecen los nodos con toda su información. A continuación aparecen las vías que están formadas por nodos. Cuando se leen las vías se establecen las uniones entre los diversos nodos. A continuación, se muestra un ejemplo de fichero de este tipo.



```

<?xml version="1.0" encoding="UTF-8"?>
<osm version="0.6" generator="CGImap 0.0.2">
  <bounds minlat="41.646749" minlon="-0.916157" maxlat="41.657972" maxlon="-0.89427"/>
  <node id="30389454" lat="41.6590548" lon="-0.8928306" user="Emilio Gomez" uid="2904"
visible="true" version="3" changeset="621276" timestamp="2008-09-13T15:24:14Z">
    <tag k="created_by" v="JOSM"/>
  </node>
  <node id="30389501" lat="41.6534786" lon="-0.892797" user="jynus" uid="7406" visible="true"
version="5" changeset="686728" timestamp="2008-11-05T15:45:15Z">
    <tag k="created_by" v="JOSM"/>
  </node>
  <node id="30438349" lat="41.6579256" lon="-0.9085487" user="jynus" uid="7406"
visible="true" version="5" changeset="1296522" timestamp="2009-05-23T17:24:47Z"/>
  <node id="30438352" lat="41.6556435" lon="-0.8978763" user="joanjim" uid="101525"
visible="true" version="7" changeset="623050" timestamp="2009-02-22T17:57:21Z"/>
  <node id="30438353" lat="41.6495639" lon="-0.8986742" user="jynus" uid="7406"
visible="true" version="3" changeset="2083597" timestamp="2009-08-09T11:10:37Z"/>
  <node id="30438361" lat="41.6556157" lon="-0.908609" user="Emilio Gomez" uid="2904"
visible="true" version="3" changeset="621276" timestamp="2008-09-13T17:01:07Z">
    <tag k="created_by" v="JOSM"/>
  </node>
  <node id="30438365" lat="41.6554583" lon="-0.8983497" user="joanjim" uid="101525"
visible="true" version="6" changeset="623050" timestamp="2009-02-22T17:57:22Z"/>
  <node id="30438367" lat="41.6556243" lon="-0.8980773" user="joanjim" uid="101525"
visible="true" version="6" changeset="623050" timestamp="2009-02-22T17:57:22Z"/>
  <node id="30438368" lat="41.6492826" lon="-0.8990635" user="jynus" uid="7406"
visible="true" version="4" changeset="2083597" timestamp="2009-08-09T11:10:37Z"/>
  <node id="30438377" lat="41.6537742" lon="-0.9100585" user="joanjim" uid="101525"
visible="true" version="5" changeset="623050" timestamp="2009-02-22T17:52:35Z"/>
  <node id="30438380" lat="41.6492916" lon="-0.899298" user="jynus" uid="7406" visible="true"
version="2" changeset="2083597" timestamp="2009-08-09T11:10:37Z"/>
  <node id="30438382" lat="41.6555607" lon="-0.9082042" user="jynus" uid="7406"
visible="true" version="4" changeset="137055" timestamp="2008-10-07T18:33:17Z">
    <tag k="created_by" v="JOSM"/>
  </node>
  <way id="27064967" user="Jesús Gámez" uid="6389" visible="true" version="1"
changeset="632181" timestamp="2008-09-14T15:39:15Z">
    <nd ref="263695626"/>
    <nd ref="263695623"/>
    <tag k="highway" v="footway"/>
    <tag k="name" v="Calle de Mariano Carderera"/>
    <tag k="source" v="yahoo"/>
  </way>
  <way id="27064968" user="Jesús Gámez" uid="6389" visible="true" version="1"
changeset="632181" timestamp="2008-09-14T15:39:16Z">
    <nd ref="296947074"/>
    <nd ref="296947077"/>
    <tag k="highway" v="footway"/>
  </way>
  <way id="27064969" user="Jesús Gámez" uid="6389" visible="true" version="1"
changeset="632181" timestamp="2008-09-14T15:39:16Z">
    <nd ref="296947082"/>
    <nd ref="296947085"/>
    <tag k="highway" v="footway"/>
  </way>
</osm>

```

### B.3. Fichero de las áreas de monitorización

Cada mapa posee un área de monitorización para dar la posibilidad de realizar simulaciones de este tipo (monitorización de áreas). Esta área de monitorización está indicada a través de la información que se guarda en un fichero de texto junto a cada fichero OSM (“GRAPH/Nombre de mapa/”). Por tanto, existe un fichero que indica el área de monitorización por cada mapa existente en el simulador. Este fichero se llama “MonitoredArea.txt”.

La información necesaria para cargar un área de monitorización se guarda en una sola línea del fichero de texto. Esta línea contiene 5 elementos: la palabra clave “MonitoredArea” y 4 números. Cada uno de estos números indica una coordenada geográfica. Estos se agrupan de dos en dos. Es decir, los 2 primeros indican un punto y los 2 segundos indican otro punto, de tal forma que el primero indica la longitud geográfica del Punto1, el segundo indica la latitud geográfica del Punto1, el tercero indica la longitud geográfica del Punto2 y el cuarto indica la latitud geográfica del Punto2.

Por tanto, sólo son necesarios 2 puntos para indicar el área de monitorización. El Punto1 es el correspondiente al vértice superior izquierdo del rectángulo que indica el área y el Punto2 es el correspondiente al vértice inferior derecho del rectángulo que indica el área. A continuación, se muestra un esquema en la Figura B.1.

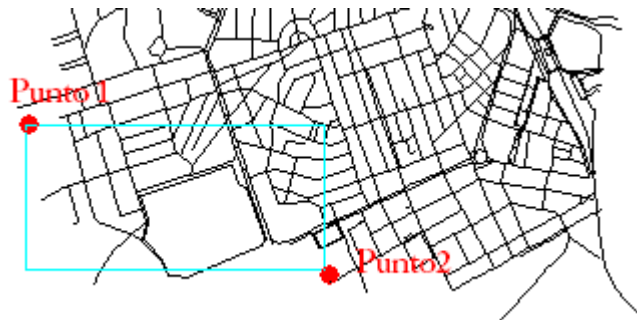


Figura B.1 – Explicación del área de monitorización

Hay que tener en cuenta que la latitud del Punto2 es menor que la latitud del Punto1. A continuación se muestra un ejemplo de fichero.

```
MonitoredArea -0.9183025360107422 41.645725096630095 0.012788772583007812 0.004616613334867736
```

## B.4. Fichero de los elementos fijos de soporte

---

Cada mapa puede poseer elementos fijos de soporte para las comunicaciones. Toda la información que se necesita para cargar estos elementos en un mapa se encuentra almacenada en un fichero de texto por cada mapa. Este fichero se llama “Antennas.txt”. Por lo tanto, este fichero se encontrará junto al fichero OSM y al fichero que indica el área de monitorización. Lo podemos encontrar en “GRAPH/Nombre del mapa/”.

Cada línea del fichero indica un dispositivo fijo de soporte, exceptuando la primera que comienza por “//” e indica un comentario que explica el significado de cada elemento de cada línea del fichero.

El primer y segundo elemento de la línea indica la longitud y latitud geográfica respectivamente, en la que irá situado el dispositivo fijo de soporte para las comunicaciones. El tercer y último elemento de la línea indica la cobertura en metros.

```
// Longitude, Latitude, Coverage
-73.9978, 40.7361, 300
-73.9963, 40.7434, 300
-73.9876, 40.7497, 300
-73.9754, 40.7479, 300
-73.9892, 40.7579, 300
-73.9700, 40.7577, 300
-73.9879, 40.7654, 300
-73.9678, 40.7652, 300
-74.0037, 40.7523, 300
-73.9826, 40.7302, 300
```

En el ejemplo anterior, hay 10 dispositivos fijos de soporte (uno por cada línea) los cuales tienen todos 300 metros de cobertura (distancia de alcance de comunicaciones). De momento, cada fichero de este tipo incluye esta información, aunque puede ser ampliada más adelante: existencia de acceso a Internet de cada elemento fijo, conexión directa entre elementos fijos de soporte, etc. Para ello habría que modificar las clases “Antenna” y “Antennas\_file” para poder incorporar la nueva información.

### **B.5. Fichero de información de los mapas descargados**

---

Cada vez que se descarga o elimina un mapa se actualiza un fichero de texto en el que se guarda toda la información necesaria acerca de los mapas disponibles en el simulador. Este fichero se puede encontrar en “GRAPH/”. El fichero se llama “downloadedMaps.txt”.

En este fichero se guarda el nombre del mapa que se indicó cuando se descargó el mapa, el tipo de mapa, o lo que es lo mismo el servidor del que se descargaron las imágenes del mapa, el nivel del zoom máximo y el tamaño del mapa en grados x grados.

A continuación se muestra un ejemplo:

Valenciennes GoogleMaps 16 0,01x0,015
Dublin OpenStreetMaps 16 0,02x0,01
Zaragoza YahooMaps 16 0,02x0,01
Sevilla OpenStreetMaps 15 0,02x0,01
NewYork OpenStreetMaps 15 0,02x0,03

Como se puede ver en el ejemplo anterior, hay 5 mapas ya que cada línea del fichero corresponde a un mapa. Nos centramos en la primera línea del fichero. El primer atributo es el nombre del mapa (Valenciennes), el segundo indica el tipo de mapa (GoogleMaps), el tercero indica el nivel máximo de zoom (16) y el último indica el tamaño del mapa (0,01° x 0,015°)

### **B.6. Fichero de registro (*log*)**

---

Todo lo necesario para la gestión del fichero de registro ya estaba implementado en el prototipo anterior. De todas formas, se va a explicar el formato de un fichero de registro.

Cada línea refleja un evento o suceso que ha sucedido en el simulador y es necesario registrar. Cada línea consta de 2 atributos: atributo temporal y atributo de información. El atributo temporal indica cuándo se ha producido el evento. Esto viene indicado con una fecha y una hora. El formato del atributo temporal es [MM/DD/YY hh:mm:ss].

Después de este atributo temporal aparece la información acerca del suceso acontecido.

El fichero de registro comienza con información sobre el nombre y versión del simulador y después aparece “Logging started”, que indica el comienzo del registro de sucesos. A continuación, aparece información acerca de la simulación: vehículos generados, dispositivos fijos, iteraciones de medición, nodos del mapa cargado, etc. A continuación, se muestra un ejemplo:

```
[09/27/10 13:09:57] Mobile agents VANET Simulator. Version 0.5.10 (May 2010)
[09/27/10 13:09:57] Logging started
[09/27/10 13:09:57] 12 vehicles, 1 fixed devices
[09/27/10 13:09:57] Maximum measurement iterations = 1000
[09/27/10 13:10:11] Loaded 1563 nodes. Map: Zaragoza
[09/27/10 13:10:12] Loading monitored area from file
[09/27/10 13:10:12] Created device 0
[09/27/10 13:10:12] Created device 1
[09/27/10 13:10:12] Created vehicle Car0
[09/27/10 13:10:12] Created device 2
[09/27/10 13:10:12] Created vehicle Car1
[09/27/10 13:10:12] Created device 3
[09/27/10 13:10:12] Created vehicle Car2
[09/27/10 13:10:12] Created device 4
[09/27/10 13:10:12] Created vehicle Car3
[09/27/10 13:10:12] Created device 5
[09/27/10 13:10:12] Created vehicle Car4
[09/27/10 13:10:12] Created device 6
[09/27/10 13:10:12] Created vehicle Car5
[09/27/10 13:10:12] Created device 7
[09/27/10 13:10:12] Created vehicle Car6
[09/27/10 13:10:12] Created device 8
[09/27/10 13:10:12] Created vehicle Car7
[09/27/10 13:10:12] Created device 9
[09/27/10 13:10:12] Created vehicle Car8
[09/27/10 13:10:12] Created device 10
[09/27/10 13:10:12] Created vehicle Car9
[09/27/10 13:10:12] Created device 11
[09/27/10 13:10:12] Created vehicle Car10
[09/27/10 13:10:12] Created device 12
[09/27/10 13:10:12] Created vehicle Car11
[09/27/10 13:10:12] Created agent 0
[09/27/10 13:10:16] 8: Agent 0 moves to device Device10 (Car9)
[09/27/10 13:10:19] 16: Agent 0 moves to device Device0 (Node 72)
[09/27/10 13:10:22] 53: Agent 0 moves to device Device8 (Car7)
[09/27/10 13:10:23] 105: Agent 0 moves to device Device2 (Car1)
[09/27/10 13:10:23] 133: Agent 0 moves to device Device12 (Car11)
[09/27/10 13:10:23] 134: Agent 0 moves to device Device0 (Node 72)
[09/27/10 13:10:23] 138: Agent 0 moves to device Device2 (Car1)
[09/27/10 13:10:24] 142: Agent 0 moves to device Device11 (Car10)
[09/27/10 13:10:24] 143: Agent 0 moves to device Device0 (Node 72)
[09/27/10 13:10:24] 169: Agent 0 moves to device Device12 (Car11)
[09/27/10 13:10:24] 185: Agent 0 moves to device Device4 (Car3)
[09/27/10 13:10:25] 289: Agent 0 moves to device Device1 (Car0)
[09/27/10 13:10:26] 291: Agent 0 moves to device Device10 (Car9)
[09/27/10 13:10:26] 293: Agent 0 moves to device Device4 (Car3)
[09/27/10 13:10:26] 295: Agent 0 moves to device Device1 (Car0)
[09/27/10 13:10:26] 314: Agent 0 moves to device Device12 (Car11)
[09/27/10 13:10:26] 326: Agent 0 moves to device Device2 (Car1)
[09/27/10 13:10:26] 358: Agent 0 moves to device Device0 (Node 72)
[09/27/10 13:10:26] 362: Agent 0 moves to device Device1 (Car0)
[09/27/10 13:10:27] 367: Agent 0 moves to device Device2 (Car1)
[09/27/10 13:10:28] 463: Agent 0 moves to device Device3 (Car2)
[09/27/10 13:10:29] 549: Agent 0 moves to device Device0 (Node 72)
[09/27/10 13:10:29] 552: Agent 0 moves to device Device5 (Car4)
[09/27/10 13:10:30] 615: Agent 0 moves to device Device3 (Car2)
[09/27/10 13:10:30] 619: Agent 0 moves to device Device10 (Car9)
[09/27/10 13:10:31] 717: Agent 0 moves to device Device12 (Car11)
[09/27/10 13:10:31] 718: Agent 0 moves to device Device8 (Car7)
[09/27/10 13:10:31] 720: Agent 0 moves to device Device10 (Car9)
[09/27/10 13:10:31] 721: Agent 0 moves to device Device12 (Car11)
[09/27/10 13:10:40] Terminated agent 0
```

## B.7. Ficheros de trazas GPS reales

---

El simulador trata diferentes formatos de ficheros de trazas GPS reales: plt, txt y log. Cada uno tiene unas particularidades que se van a explicar a continuación.

### B.7.1. Formato log

---

El formato log [30, 31, 50] es un formato propio de trazas GPS del proyecto GeoLife Trajectories. Cada fichero .log puede contener trazas de diferentes días y por tanto de diferentes lugares, es decir, que el punto donde acaba una trayectoria un día no tiene por qué ser el mismo que el punto donde empiece otra trayectoria de otro día.

A continuación se muestran varios ejemplos de puntos pertenecientes a una trayectoria registrada en un fichero de formato log.

Y 15-05-10 01:00:20 11626.686933 E 3959.348887 N 33.0 19.446 308.1 06
Y 15-05-10 01:00:24 11626.672117 E 3959.356818 N 32.5 26.2984 309.2 06

El significado de los atributos es el siguiente (ordenados según su aparición):

- **Validez:** indica si el nodo registrado es válido o no. “Y” significa que es válido.
- **Fecha:** indica la fecha en que fue registrada la posición. El formato de la fecha es DD/MM/YY (día/mes/año).
- **Hora:** indica la hora en la que fue registrada la posición. El formato (UTC) de la hora es horas:minutos:segundos (hh:mm:ss).
- **Longitud:** longitud geográfica de la posición registrada.
- **Sentido de la longitud:** sentido de la longitud registrada indicada en el campo anterior. “E” significa este y “W” significa oeste.
- **Latitud:** latitud geográfica de la posición registrada.
- **Sentido de la latitud:** sentido de la latitud registrada indicada en el campo anterior. “N” significa norte y “S” significa sur.
- **Altura:** altura de la posición registrada. Las unidades de este valor son metros.
- **Velocidad:** velocidad a la que se movía el elemento que registró dicha posición. La unidad de este valor es Km/h.
- **Dirección:** dirección que seguía el dispositivo que registró la posición. La unidad de este valor son los grados. Se usa el norte como base y se sigue el sentido de las agujas del reloj.
- **Número de satélites:** número de satélites utilizados para registrar la posición.

Y 15-05-10 01:00:22 11626.680003 E 3959.352531 N 32.0 23.15 305.0 06
--

El valor 11626.686933 significa 116 grados y 26.686933 segundos.

A continuación, se muestra un ejemplo de fichero .log de trazas GPS reales extraído del proyecto *GeoLife GPS Trajectories*.

Y 25-05-08 00:57:34 11619.2151 E 4000.3676 N 21.6 0.14816 108.15 04
Y 25-05-08 00:57:37 11619.2126 E 4000.3672 N 11.2 0.16668 98.05 04
Y 25-05-08 00:57:39 11619.2127 E 4000.3673 N 11.4 0 98.05 04
Y 25-05-08 00:57:47 11619.2133 E 4000.3679 N 2.7 1.74088 167.23 04
N 25-05-08 00:57:53 11619.2133 E 4000.3679 N 2.7 3.22410976 167.23 00
Y 25-05-08 00:57:55 11619.2338 E 4000.3752 N 81.9 4.48184 86.67 04
Y 25-05-08 00:57:58 11619.2283 E 4000.3740 N 77.0 0 86.67 04
Y 25-05-08 00:58:00 11619.2297 E 4000.3750 N 65.7 3.20396 89.90 04
N 25-05-08 00:58:01 11619.2297 E 4000.3750 N 65.7 5.93373392 89.90 00
Y 25-05-08 00:58:03 11619.2251 E 4000.3707 N 49.7 0 89.90 04
Y 25-05-08 00:58:06 11619.2332 E 4000.3720 N 64.4 12.46396 70.08 04
Y 25-05-08 00:58:08 11619.2421 E 4000.3739 N 57.1 13.7048 84.20 04
N 25-05-08 01:00:05 11619.2421 E 4000.3739 N 57.1 25.3812896 84.20 00
N 25-05-08 01:00:07 11619.2421 E 4000.3739 N 57.1 47.0061483392 84.20 00
N 25-05-08 01:00:09 11619.2421 E 4000.3739 N 57.1 87.0553867241984 84.20 00
N 25-05-08 01:00:11 11619.2421 E 4000.3739 N 57.1 161.226576213215 84.20 00
N 25-05-08 01:00:30 11619.2421 E 4000.3739 N 57.1 298.591619146874 84.20 00
Y 25-05-08 01:03:37 11619.3596 E 4000.2739 N 49.5 29.9098 258.58 03
Y 25-05-08 01:03:39 11619.3409 E 4000.2736 N 49.5 121.06524 262.25 03
N 25-05-08 01:07:07 11619.3409 E 4000.2736 N 49.5 224.21282448 262.25 00
Y 25-05-08 01:07:09 11619.4256 E 4000.0704 N 49.6 26.7614 94.65 03
Y 25-05-08 01:07:11 11619.4709 E 4000.0800 N 66.5 17.57548 188.32 04
Y 25-05-08 01:07:21 11619.4864 E 4000.0551 N 49.4 0 195.36 03
Y 25-05-08 01:07:23 11619.4672 E 4000.0481 N 52.8 18.00144 205.97 04
Y 25-05-08 01:07:25 11619.4667 E 4000.0454 N 57.6 17.29768 197.57 04
Y 25-05-08 01:07:27 11619.4739 E 4000.0440 N 62.4 11.07496 167.70 04
N 25-05-08 01:07:55 11619.4739 E 4000.0440 N 62.4 20.51082592 167.70 00
N 25-05-08 01:07:56 11619.4739 E 4000.0440 N 62.4 37.98604960384 167.70 00
Y 25-05-08 01:07:58 11619.5086 E 3959.9885 N 81.3 16.00128 79.82 04
Y 25-05-08 01:08:07 11619.5308 E 3959.9869 N 85.9 12.48248 92.47 03
Y 25-05-08 01:08:08 11619.5320 E 3959.9877 N 85.9 13.64924 103.00 03
Y 25-05-08 01:08:10 11619.5336 E 3959.9829 N 43.0 13.96408 141.34 04
Y 25-05-08 01:08:13 11619.5379 E 3959.9792 N 57.4 15.07528 161.76 04
Y 25-05-08 01:08:46 11619.6059 E 3959.9156 N 86.2 0 161.76 03
Y 25-05-08 01:08:48 11619.5859 E 3959.9038 N 9.3 16.09388 171.13 04
Y 25-05-08 01:08:50 11619.5761 E 3959.8979 N 57.1 16.98284 187.51 04
Y 25-05-08 01:08:52 11619.5754 E 3959.8931 N 59.5 16.22352 178.46 04
N 25-05-08 01:11:07 11619.5754 E 3959.8931 N 59.5 30.04595904 178.46 00
Y 25-05-08 01:11:07 11619.5384 E 3959.5371 N 134.5 22.5018 183.27 04
Y 25-05-08 01:11:10 11619.5764 E 3959.5599 N 67.9 16.9458 183.15 04
Y 25-05-08 01:12:01 11619.5336 E 3959.4681 N 134.9 0 182.50 03
Y 25-05-08 01:12:04 11619.4970 E 3959.4511 N 430.5 35.22504 220.54 04
Y 25-05-08 01:12:06 11619.4956 E 3959.4466 N 413.0 17.81624 190.94 04
Y 25-05-08 01:12:30 11619.5738 E 3959.3897 N 25.3 0 190.94 04

### B.7.2. Formato txt

---

El formato “.txt” [49] es un formato bastante simple de trazas GPS reales. Cada fichero “.txt” puede contener varias trazas independientes, es decir, de diferentes días, y por lo tanto de diferentes lugares. Es decir, que el punto donde acaba una trayectoria un día no tiene por qué ser el mismo que el punto donde empieza otra trayectoria de otro día.

En un fichero de trazas GPS “.txt” cada línea representa un punto diferente registrado. Cada línea tiene 7 atributos con información sobre el punto registrado. A continuación, se explican los 7 atributos en orden de izquierda a derecha según su aparición en la línea, separados por una coma:

- Atributo 1: indica la característica del punto. TP significa track point y WP significa way point.
- Atributo 2: D significa datum, lo que quiere decir que los datos que aparecen a continuación pertenecen a un punto registrado.
- Atributo 3: indica la latitud geográfica en la que se registró el punto.
- Atributo 4: indica la longitud geográfica en la que se registró el punto.
- Atributo 5: indica la fecha en la que fue registrado el punto. El formato de la fecha es YYYY/MM/DD (año/mes/día).
- Atributo 6: indica la hora en la que fue registrado el punto. El formato de la hora es hh:mm:ss (horas:minutos:segundos).
- Atributo 7: indica si el punto pertenece a la misma traza que el punto anterior o es el primer punto de una nueva traza. 0 indica que el punto pertenece a la misma traza y 1 indica que comienza una nueva traza.



A continuación se muestra un ejemplo de traza registrada en un fichero de formato “.txt”.

```
Datum,WGS84,WGS84,0,0,0,0,0
TP,D,39.9776850, 116.3276249, 2008/04/01,03:25:46,1
TP,D,39.9777233, 116.3276216, 2008/04/01,03:25:47,0
TP,D,39.9778499, 116.3276266, 2008/04/01,03:25:50,0
TP,D,39.9779866, 116.3276249, 2008/04/01,03:25:53,0
TP,D,39.9781200, 116.3276133, 2008/04/01,03:25:56,0
TP,D,39.9782500, 116.3275999, 2008/04/01,03:25:59,0
TP,D,39.9783850, 116.3275983, 2008/04/01,03:26:02,0
TP,D,39.9785150, 116.3275966, 2008/04/01,03:26:05,0
TP,D,39.9786449, 116.3275966, 2008/04/01,03:26:08,0
TP,D,39.9787699, 116.3275933, 2008/04/01,03:26:11,0
TP,D,39.9788916, 116.3275899, 2008/04/01,03:26:14,0
TP,D,39.9790099, 116.3275700, 2008/04/01,03:26:17,0
TP,D,39.9791283, 116.3275500, 2008/04/01,03:26:20,0
TP,D,39.9792466, 116.3275399, 2008/04/01,03:26:23,0
TP,D,39.9793733, 116.3275350, 2008/04/01,03:26:26,0
TP,D,39.9794999, 116.3275283, 2008/04/01,03:26:29,0
TP,D,39.9796383, 116.3275099, 2008/04/01,03:26:32,0
TP,D,39.9797750, 116.3275033, 2008/04/01,03:26:35,0
TP,D,39.9799133, 116.3274899, 2008/04/01,03:26:38,0
TP,D,39.9800499, 116.3274816, 2008/04/01,03:26:41,0
```

La primera línea indica el sistema de coordenadas que se usa, es el WGS84 [51]. El WGS84 es un sistema de coordenadas cartográficas mundial que permite localizar cualquier punto de la Tierra (sin necesitar otro de referencia) por medio de tres unidades dadas. WGS84 son las siglas en inglés de World Geodetic System 84 (que significa Sistema Geodésico Mundial 1984).

### B.7.3. Formato plt

---

El formato “.plt” [48] es un formato de fichero de trazas GPS más simple que el “.txt”. Este formato es el que usa para exportar las trazas OziExplorer [52]. OziExplorer es un programa interactivo que usa mapas de imágenes convencionales y algunos en formatos vectoriales, y que permite planificar viajes, crear y añadir *waypoints*, rutas y *tracks*. Además permite intercambiar este tipo de información con receptores GPS y un ordenador PC.

- Línea 1: indica el tipo de fichero y la versión del mismo.
- Línea 2: sistema de coordenadas en el que se expresan las coordenadas geográficas de cada punto de la traza o *track*.
- Línea 3: indica que la altitud se expresa en pies.

- Línea 4: reservado para un uso futuro.
- Línea 5: esta línea tiene diversa información.
  - Campo 1: siempre cero.
  - Campo 2: anchura de la línea de la traza en la pantalla indicada en píxeles.
  - Campo 3: color de la traza (codificación RGB).
  - Campo 4: descripción o comentario de la traza (no permite comas).
  - Campo 5: valor para saltar puntos de la traza. Reduce el número de puntos de la traza que se dibujan, usualmente tiene el valor 1.
  - Campo 6: tipo de traza. 0 = normal, 10 = polígono cerrado, 20 = zona de alarma.
  - Campo 7: estilo para rellenar la traza. 0 =bsSolid; 1 =bsClear; 2 =bsBdiagonal; 3 =bsFdiagonal; 4 =bsCross; 5 =bsDiagCross; 6 =bsHorizontal; 7 =bsVertical
  - Campo 8: número de puntos que constituyen la traza.
- Línea 6: la traza o *track*. Hay que tener en cuenta que cada línea está dedicada a cada punto de la traza, que cada campo está separado por una coma, y que si un campo no es necesario o no se dispone de información se colocan varias comas seguidas.

Una vez explicadas las primeras líneas del fichero se procede a explicar cada campo de las líneas de los puntos de la traza registrados.

- Campo 1: latitud geográfica expresada en grados decimales. Si es negativa, está en el hemisferio sur.
- Campo 2: longitud geográfica expresada en grados decimales. Si es negativa, está hacia el oeste del meridiano de Greenwich.
- Campo 3: es un indicador que si vale 1 representa una discontinuidad en la línea de *track* y 0 si es normal.
- Campo 4: altitud en pies. -777 si no es válido.
- Campo 5: fecha y hora codificada.
- Campo 6: fecha expresada en el formato YYYY/MM/DD.
- Campo 7: hora expresada en el formato hh:mm:ss.

En la importación de las trazas de este formato, para conocer la fecha y hora en la que fue registrado un punto se usan los campos 6 y 7 en vez del campo 5.

A continuación se muestra un pequeño ejemplo de un fichero en formato “.plt” de una traza.

```
iTravel-Tech Track Point File Version 1.1.8
WGS 84
Altitude is in Feet
Reserved 3
0,2,255,My Track,0,0,2,8421376
39.976901,116.329887,0,285,39573.654433,2008/05/05,15:42:23
39.976487,116.329889,0,160,39573.654444,2008/05/05,15:42:24
39.976195,116.329918,0,18,39573.654456,2008/05/05,15:42:25
39.976590,116.329900,0,173,39573.654468,2008/05/05,15:42:26
39.976776,116.329879,0,238,39573.654479,2008/05/05,15:42:27
39.977605,116.329813,0,564,39573.654491,2008/05/05,15:42:28
39.977748,116.329823,0,606,39573.654514,2008/05/05,15:42:30
39.977752,116.329812,0,606,39573.654537,2008/05/05,15:42:32
39.977799,116.329783,0,634,39573.654560,2008/05/05,15:42:34
39.977135,116.329825,0,359,39573.654572,2008/05/05,15:42:35
39.977122,116.329790,0,354,39573.654595,2008/05/05,15:42:37
39.977088,116.329747,0,337,39573.654618,2008/05/05,15:42:39
39.977035,116.329704,0,305,39573.654641,2008/05/05,15:42:41
39.977005,116.329716,0,273,39573.654664,2008/05/05,15:42:43
39.977020,116.329674,0,277,39573.654687,2008/05/05,15:42:45
39.977011,116.329639,0,276,39573.654711,2008/05/05,15:42:47
39.977001,116.329605,0,276,39573.654734,2008/05/05,15:42:49
39.976984,116.329566,0,270,39573.654757,2008/05/05,15:42:51
39.976973,116.329526,0,267,39573.654780,2008/05/05,15:42:53
39.976976,116.329506,0,270,39573.654803,2008/05/05,15:42:55
39.976976,116.329485,0,273,39573.654826,2008/05/05,15:42:57
39.976980,116.329467,0,268,39573.654850,2008/05/05,15:42:59
```

## B.8. Ficheros de filtros

---

Los filtros que se aplican en el simulador (a los mapas y a las trazas) pueden ser guardados en el sistema. Estos filtros se guardan en ficheros de texto en la carpeta del simulador “SCENARIOS” todos al mismo nivel y con el nombre que el usuario ha indicado previamente. A continuación, se explica el formato de un fichero de este tipo y se muestra un ejemplo.

Cada línea del fichero indica un atributo del filtro. Cada línea comienza con un identificador del atributo del filtro y a continuación aparecen los elementos que definen el filtro. Todos los elementos de la línea van separados por comas.

En la primera línea aparece la palabra clave “Duration” que hace referencia al filtro temporal de duración que se puede definir para las trazas. Si existe dicho filtro, aparecerá a continuación la información referente a este filtro (horas, minutos, segundos).

En la segunda línea aparece la palabra clave “TimeInterval” que hace referencia al filtro temporal de intervalo de tiempo que se puede definir para las trazas. Hay que tener en cuenta que este filtro y el anterior son excluyentes, es decir, que si aparece un filtro el otro nunca podrá aparecer. Si existe dicho filtro, aparecerá a continuación la información referente a este filtro (fecha de inicio, hora de inicio, fecha de fin, hora de fin). El formato de la fecha es DD/MM/YYYY y el formato de la hora es hh:mm:ss.

En la tercera línea aparece la palabra clave “Speed” que hace referencia al de la velocidad que se puede definir para las trazas. Si existe dicho filtro, aparecerá a continuación la información referente a este filtro (velocidad mínima).

En la cuarta línea aparece la palabra clave “FileName” que hace referencia al fichero de trazas o ficheros de trazas que fueron sometidos a este filtro. Puede aparecer el nombre de un fichero, el nombre de varios ficheros o aparecer “null.null” que indica que este filtro no se ha aplicado a ninguna traza sino a un mapa.

En la quinta línea aparece la palabra clave “IgnoreNumVehicles” que hace referencia a ignorar el número de vehículos de la simulación al importar trazas. Si a continuación aparece “true” quiere decir que se ignorará esta restricción, en caso contrario aparecerá “false”.

En la sexta línea aparece la palabra clave “TrackGeographicBounds” que hace referencia al filtro geográfico que se puede aplicar a las trazas. Si existe dicho filtro, aparecerán a continuación las coordenadas geográficas del punto situado en la posición superior izquierda (longitud,latitud) y las coordenadas del punto situado en la posición inferior derecha (longitud, latitud). Hay que tener en cuenta que la latitud del punto situado en la posición inferior derecha es menor que la del punto situado en la posición superior izquierda.

En la séptima línea aparece la palabra clave “GraphName” que hace referencia al mapa que se usó para importar esas trazas. A continuación aparece el nombre del mapa.

En la octava y última línea aparece la palabra clave “GraphGeographicBounds”, que hace referencia al filtro geográfico que se puede aplicar a un mapa. Si existe dicho filtro, aparecerán a continuación las

coordenadas geográficas del punto situado en la posición superior izquierda (longitud,latitud) y las coordenadas del punto situado en la posición inferior derecha (longitud, latitud). Hay que tener en cuenta que la latitud del punto situado en la posición inferior derecha es menor que la del punto situado en la posición superior izquierda.

A continuación, se muestra un ejemplo de filtro.

```
Duration,  
TimeInterval,04/05/1986,01:02:03,09/10/1987,06:07:08  
Speed,13  
FileName,4.14.txt  
IgnoreNumVehicles,true  
TrackGeographicBounds,116.2957763671875,39.99579018249429,116.35688781738281  
,39.96660174122572  
GraphName,Pruebatxt  
GraphGeographicBounds,116.2957763671875,39.99579018249429,116.35688781738281  
,39.96660174122572
```

El filtro anterior tiene un filtro temporal de intervalo de duración (desde 04/05/1986 01:02/03 hasta 09/10/1987 06/07/08), un filtro de velocidad cuya velocidad mínima es de 13 km/h, el fichero de trazas al que se le aplicó el filtro se llama 4.14.txt. Además se ignora el número de vehículos de la simulación a la hora de importar las trazas, el filtro geográfico es el que aparece y además se guardó ese mismo filtro geográfico para el mapa “Pruebatxt”.



## ANEXO C – ANÁLISIS DEL SISTEMA

---

En este anexo se van a enumerar los distintos requisitos y casos de uso identificados a lo largo de la fase de análisis. Con los requisitos y casos de uso se pretende describir las funcionalidades de la aplicación. Los requisitos se explicarán de una forma breve y se clasificarán como funcionales y no funcionales.

Se considera que el simulador será controlado por un usuario normal (investigador o similar) y por lo tanto habrá un solo tipo de actor.

### C.1. Requisitos

---

<b>Identificador:</b> 1
<b>Descripción:</b> El sistema permitirá diferentes estrategias de movilidad para los vehículos. Estas estrategias de movilidad son Gauss-Markov, Manhattan y dos estrategias de movilidad basadas en rutas (rutas calculadas con el algoritmo del camino más corto de Dijkstra y rutas calculadas de forma dinámica con una heurística). Las estrategias de movilidad Gauss-Markov y Manhattan no son puramente esas políticas de movilidad, sino que están adaptadas a la circulación por vías cuya movilidad está restringida por la topología de la ciudad.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 2
<b>Descripción:</b> El sistema permitirá controlar la simulación a través de una interfaz amigable de forma que se pueda definir un escenario de simulación y controlarlo de forma sencilla.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 3
<b>Descripción:</b> El sistema permitirá acceder a las diferentes herramientas del mismo a través de una barra de menú. Desde esta barra de menú se permite: salir del simulador, parar la simulación actual, iniciar una simulación, pausar la simulación, simular de forma rápida (más rápida que en modo normal), abrir una ventana en la que se pueden modificar parámetros de la simulación, mostrar las imágenes del mapa de la simulación ( <i>tiles</i> ), acercar el zoom, alejar el zoom, centrar la vista en el mapa, abrir una ventana en la que se puede modificar la información que se quiere mostrar de un vehículo cuando se haga doble clic sobre él, mostrar o esconder la barra de herramientas con botones para controlar el simulador, seleccionar si se quiere dibujar el área de monitorización o desplazarse por el mapa, descargar un mapa, abrir un mapa, importar trazas GPS y mostrar información acerca del simulador.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 4
<b>Descripción:</b> El sistema permitirá controlar la simulación a través de una barra de control con diversos botones. Desde esta barra se permite: parar la simulación actual, iniciar una simulación, pausar la simulación, simular de forma rápida, abrir una ventana en la que se pueden modificar parámetros de la simulación, mostrar las imágenes del mapa de la simulación, acercar el zoom, alejar el zoom, centrar la vista en el mapa y seleccionar si se quiere dibujar el área de monitorización o desplazarse por el mapa.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 5
<b>Descripción:</b> El sistema permitirá cambiar la zona de visualización del mapa a través de una barra de desplazamiento o través del cursor.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 6
<b>Descripción:</b> El sistema permitirá visualizar información sobre un vehículo haciendo doble clic sobre el vehículo en cuestión. La información que se mostrará será: nombre del vehículo, distancia recorrida por el mismo, velocidad actual, número de agentes que han pasado por el vehículo, estrategia de movilidad, nombre del dispositivo del vehículo y número actual de agentes que tiene.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 7
<b>Descripción:</b> El sistema permitirá elegir la información que se desea ver de un vehículo cuando se haga doble clic sobre él.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 8
<b>Descripción:</b> La ventana que muestra la información del vehículo se adapta (su tamaño) a la cantidad de información que se va a mostrar.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 9
<b>Descripción:</b> El sistema permitirá modificar parámetros de simulación mientras ésta se está produciendo o cuando no existe ninguna simulación en marcha. Se permitirá modificar los siguientes parámetros de simulación: estrategia de salto de los agentes, velocidad de los vehículos en km/h, estrategia de movilidad de los vehículos, mínima y máxima distancia a la que se tiene que encontrar el destino cuando se tiene una estrategia de movilidad basada en rutas, número de vehículos, cobertura de los dispositivos de los vehículos, mostrar las rutas que siguen los vehículos



o no, mostrar información acerca de los nodos o no, incluir antenas (nodos de soporte en las comunicaciones) o no, cobertura de los dispositivos de las antenas y cobertura de los dispositivos asociados a un nodo.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 10
<b>Descripción:</b> El sistema permitirá descargar mapas. La herramienta de descarga de mapas permitirá elegir el nombre con el que se quiera guardar el mapa, el nivel de zoom máximo y el tipo de mapa que se quiere descargar. Los 5 tipos de mapas son: YahooMaps, GoogleMaps, OpenStreetMap, MicrosoftHybrid y MicrosoftMap. Cuando se descarga un mapa, realmente, se producen dos descargas diferentes: una descarga el mapa dividido en pequeñas imágenes de 256 x 256 pixeles, y otra descarga el fichero XML con la información sobre las vías (carreteras, calles, etc.) desde el servido de OpenStreetMap.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 11
<b>Descripción:</b> El sistema permitirá aplicar filtros geográficos cuando se abra un mapa previamente descargado.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 12
<b>Descripción:</b> La forma de definir los filtros geográficos sobre un mapa será de 2 formas: escribiendo los límites (coordenadas geográficas de 2 puntos que limitan el mapa) en unos campos de texto o través de una herramienta gráfica que permite dibujar el área a filtrar.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 13
<b>Descripción:</b> La herramienta que permitirá dibujar el área a filtrar al abrir un mapa dispondrá de varios botones. Se abrirá en otra ventana el mapa a filtrar.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 14
<b>Descripción:</b> La herramienta que permite dibujar el área a filtrar al abrir un mapa permitirá aceptar el filtro, cancelarlo, hacer zoom sobre el mapa, dibujar el filtro con el cursor y moverse por el mapa con el cursor y a través de unas barras de desplazamiento.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 15
<b>Descripción:</b> El sistema permitirá ver un listado de los mapas previamente descargados para que el usuario elija el que quiera cargar. En esta lista aparecerá información acerca de cada mapa (nombre, máximo zoom descargado, tipo de mapa y tamaño).
<b>Tipo:</b> Funcional

<b>Identificador:</b> 16
<b>Descripción:</b> El sistema descargará automáticamente las imágenes de un mapa que no se hayan descargado si al simular sobre él se modifica el zoom y dichas imágenes no habían sido previamente descargadas.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 17
<b>Descripción:</b> El sistema permitirá eliminar un mapa descargado a través de la lista en la que se muestran los mapas descargados. Se eliminará el fichero correspondiente, las imágenes que componen el mapa y todas las carpetas creadas para almacenar dicho mapa.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 18
<b>Descripción:</b> El sistema permitirá guardar un filtro geográfico aplicado a un mapa. Para ello se podrá elegir el nombre para dicho filtro. Además se podrá elegir un filtro previamente usado y guardado para abrir un mapa. También se podrá eliminar un filtro previamente guardado.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 19
<b>Descripción:</b> El sistema permitirá ver un listado de los filtros guardados para los mapas. En esta lista aparecerá información acerca de cada filtro (nombre del filtro, nombre del mapa al que afecta y valores del filtro (coordenadas geográficas)).
<b>Tipo:</b> Funcional

<b>Identificador:</b> 20
<b>Descripción:</b> El sistema permitirá cargar y guardar los parámetros de simulación.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 21
<b>Descripción:</b> Los parámetros de simulación se guardarán en un fichero XML.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 22
<b>Descripción:</b> El sistema permitirá importar trazas GPS reales. Los formatos de los ficheros de las trazas GPS que se podrán importar serán: txt, plt y log.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 23
<b>Descripción:</b> El sistema permitirá aplicar filtros a las trazas GPS importadas. Estos filtros serán temporales (de intervalo de tiempo y duración), de velocidad y geográficos. Además, a la hora de importar las trazas se podrá ignorar el número de vehículos que existen en la simulación.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 24
<b>Descripción:</b> El sistema permitirá guardar un filtro aplicado a una traza. Para ello podrá elegir el nombre para dicho filtro. Además podrá elegir un filtro previamente usado y guardado. También se podrá eliminar un filtro previamente guardado.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 25
<b>Descripción:</b> El sistema permitirá ver un listado de los filtros guardados para las trazas. En esta lista aparecerá información acerca de cada filtro (nombre del filtro e información acerca del filtro).
<b>Tipo:</b> Funcional

<b>Identificador:</b> 26
<b>Descripción:</b> El sistema permitirá simular la existencia de elementos fijos de soporte de la VANET (nodos de soporte para las comunicaciones en las carreteras).
<b>Tipo:</b> Funcional

<b>Identificador:</b> 27
<b>Descripción:</b> El sistema permitirá simular en modo <i>batch</i> (por lotes) y en modo gráfico.
<b>Tipo:</b> Funcional

<b>Identificador:</b> 28
<b>Descripción:</b> El sistema almacenará los mapas en una carpeta llamada "GRAPH".

Dentro de esta carpeta, cada mapa se guarda en una subcarpeta con el nombre que se le asignó.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 29
<b>Descripción:</b> El sistema tendrá un fichero con nombre “downlaodedMaps.txt” en la carpeta “GRAPH”. En este fichero se almacenará toda la información relativa a los mapas disponibles en el simulador. Cada vez que se elimine o descargue un mapa, se modificará la información de este fichero. Se almacenará la siguiente información de cada mapa: nombre del mapa, tipo de mapa, nivel máximo de zoom y tamaño del mapa.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 30
<b>Descripción:</b> El sistema almacenará los filtros en una carpeta llamada “SCENARIOS”.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 31
<b>Descripción:</b> El sistema obtendrá los ficheros de trazas para importar en la carpeta “TRACKS”. Por tanto, el usuario deberá depositar los ficheros en esta carpeta para que sean visibles para el simulador.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 32
<b>Descripción:</b> El sistema funcionará tanto en Unix como en Windows. Será implementado en Java.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 33
<b>Descripción:</b> El fichero de configuración “config.xml” se guardará en la carpeta “CONFIG”.
<b>Tipo:</b> No funcional

<b>Identificador:</b> 34
<b>Descripción:</b> El fichero de los elementos fijos de soporte “Antennas.txt” y el fichero que guardará el área de monitorización se guardará en la subcarpeta con el nombre del mapa dentro de la carpeta “GRAPH”.
<b>Tipo:</b> No funcional

## C.2. Casos de uso

---

A continuación se presentan los casos de uso.

### C.2.1. Gestionar mapas

---

<b>Nombre:</b> Visualizar información de mapa
<b>Descripción:</b> Se puede elegir entre los mapas almacenados en el simulador y visualizar información acerca de ellos (nombre del mapa, tipo de mapa, nivel máximo de zoom y tamaño).

<b>Nombre:</b> Listar mapas
<b>Descripción:</b> Se puede listar los mapas almacenados en el simulador para que el usuario los visualice.

<b>Nombre:</b> Cargar mapa
<b>Descripción:</b> Se puede cargar un mapa para realizar simulaciones sobre él o para verlo simplemente.

<b>Nombre:</b> Eliminar mapa
<b>Descripción:</b> Se puede eliminar un mapa almacenado en el simulador de forma que ya no se podrá utilizar en futuras simulaciones.

### C.2.2. Descargar mapas

---

<b>Nombre:</b> Descargar mapa
<b>Descripción:</b> Se puede descargar el mapa y asignarle el nombre que el usuario desee. En la descarga se indica además el nivel de zoom máximo, el servidor y las coordenadas geográficas de la zona que se quiere descargar.

<b>Nombre:</b> Seleccionar servidor de descarga de mapa
<b>Descripción:</b> Se puede descargar el mapa desde distintos servidores: OpenStreetMap, GoogleMaps y MicrosoftMaps.

### C.2.3. Controlar mapas

---

<b>Nombre:</b> Visualizar mapa
<b>Descripción:</b> Se puede visualizar un mapa.

<b>Nombre:</b> Moverse por el mapa
<b>Descripción:</b> Se puede mover el mapa a través del cursor o a través de las barras de desplazamiento.

<b>Nombre:</b> Acercar zoom
<b>Descripción:</b> Se puede aumentar el zoom para visualizar el mapa desde más cerca.

<b>Nombre:</b> Alejar zoom
<b>Descripción:</b> Se puede reducir el zoom para visualizar el mapa desde una distancia más lejana.

<b>Nombre:</b> Mostrar posición inicial mapa
<b>Descripción:</b> Se puede volver a mostrar el mapa según se había mostrado al comienzo a pesar de haberlo movido posteriormente.

<b>Nombre:</b> Mostrar grafo y mapa
<b>Descripción:</b> Se puede mostrar el grafo del mapa junto con el mapa

<b>Nombre:</b> Mostrar rutas
<b>Descripción:</b> Se puede mostrar la ruta que sigue cada vehículo

<b>Nombre:</b> Mostrar identificador de los nodos
<b>Descripción:</b> Se puede mostrar el identificador de los nodos que forman el grafo del mapa.

<b>Nombre:</b> Mostrar información de vehículo
<b>Descripción:</b> Se puede mostrar información de un vehículo

#### C.2.4. Filtros en mapas

<b>Nombre:</b> Aplicar filtro
<b>Descripción:</b> Se puede aplicar un filtro geográfico

<b>Nombre:</b> Indicar filtro de forma manual
<b>Descripción:</b> Se puede definir el filtro escribiendo las coordenadas

<b>Nombre:</b> Indicar filtro de forma gráfica
<b>Descripción:</b> Se puede definir el filtro dibujando el área que interesa.

<b>Nombre:</b> Listar filtros guardados
<b>Descripción:</b> Se puede mostrar una lista con los filtros guardados con información sobre ellos (nombre del filtro, mapa al que afecta, coordenadas del filtro).

<b>Nombre:</b> Guardar filtro
<b>Descripción:</b> Se puede guardar un filtro utilizado.

<b>Nombre:</b> Cargar filtro
<b>Descripción:</b> Se puede cargar un filtro almacenado.

<b>Nombre:</b> Eliminar filtro
<b>Descripción:</b> Se puede eliminar un filtro almacenado.

<b>Nombre:</b> Moverse por el mapa de la herramienta gráfica de filtro
<b>Descripción:</b> Se puede mover el mapa a través del cursor o a través de las barras de desplazamiento.

<b>Nombre:</b> Acercar zoom de la herramienta gráfica de filtro
<b>Descripción:</b> Se puede aumentar el zoom para visualizar el mapa desde más cerca.

<b>Nombre:</b> Alejar zoom de la herramienta gráfica de filtro
<b>Descripción:</b> Se puede reducir el zoom para visualizar el mapa desde una distancia más lejana.

### C.2.5. Importar trazas

<b>Nombre:</b> Importar trazas
<b>Descripción:</b> Se puede importar trazas GPS reales de 3 formatos de ficheros (plt, txt y log).

<b>Nombre:</b> Elegir fichero a importar
<b>Descripción:</b> Se puede elegir el fichero o ficheros de los que importar las trazas.

### C.2.6. Filtros en trazas

---

<b>Nombre:</b> Aplicar filtro
<b>Descripción:</b> Se puede aplicar un filtro o una combinación de ellos: temporales (duración o intervalo de tiempo), de velocidad y geográfico.

<b>Nombre:</b> Indicar filtro geográfico de forma manual
<b>Descripción:</b> Se puede definir el filtro escribiendo las coordenadas

<b>Nombre:</b> Indicar filtro geográfico de forma gráfica
<b>Descripción:</b> Se puede definir el filtro dibujando el área que interesa.

<b>Nombre:</b> Listar filtros guardados
<b>Descripción:</b> Se puede mostrar una lista con los filtros guardados con información sobre ellos (nombre del filtro, características del filtro).

<b>Nombre:</b> Guardar filtro
<b>Descripción:</b> Se puede guardar un filtro utilizado.

<b>Nombre:</b> Cargar filtro
<b>Descripción:</b> Se puede cargar un filtro almacenado.

<b>Nombre:</b> Eliminar filtro
<b>Descripción:</b> Se puede eliminar un filtro almacenado.

<b>Nombre:</b> Moverse por el mapa de la herramienta gráfica de filtro geográfico
<b>Descripción:</b> Se puede mover el mapa a través del cursor o a través de las barras de desplazamiento.

<b>Nombre:</b> Acercar zoom de la herramienta gráfica de filtro geográfico
<b>Descripción:</b> Se puede aumentar el zoom para visualizar el mapa desde más cerca.

<b>Nombre:</b> Alejar zoom de la herramienta gráfica de filtro geográfico
<b>Descripción:</b> Se puede reducir el zoom para visualizar el mapa desde una distancia más lejana.



<b>Nombre:</b> Definir filtro temporal de duración
<b>Descripción:</b> Se puede definir un filtro temporal de duración indicando las horas, minutos y segundos.

<b>Nombre:</b> Definir filtro temporal de intervalo de tiempo
<b>Descripción:</b> Se puede definir un filtro temporal de intervalo de tiempo indicando la fecha y hora de inicio y la fecha y hora de fin.

<b>Nombre:</b> Definir filtro de velocidad
<b>Descripción:</b> Se puede definir un filtro de velocidad indicando el valor de la velocidad mínima.

### C.2.7. Simulación

<b>Nombre:</b> Iniciar simulación
<b>Descripción:</b> Se puede iniciar una simulación.

<b>Nombre:</b> Parar simulación
<b>Descripción:</b> Se puede parar una simulación.

<b>Nombre:</b> Pausar simulación
<b>Descripción:</b> Se puede pausar una simulación.

<b>Nombre:</b> Simular rápido
<b>Descripción:</b> Se puede simular de forma rápida.

<b>Nombre:</b> Dibujar área de monitorización
<b>Descripción:</b> Se puede dibujar el área de monitorización.

<b>Nombre:</b> Cambiar estrategia de salto de los agentes
<b>Descripción:</b> Se puede modificar la estrategia de salto de los agentes móviles.

<b>Nombre:</b> Cambiar velocidad
<b>Descripción:</b> Se puede modificar la velocidad de los vehículos.

<b>Nombre:</b> Cambiar estrategia de movilidad
<b>Descripción:</b> Se puede modificar la estrategia de movilidad de los vehículos.

<b>Nombre:</b> Modificar distancias rutas
<b>Descripción:</b> Se puede modificar la distancia mínima y máxima a la que se debe encontrar el nodo destino del nodo origen cuando se genera de forma aleatoria para una estrategia de movilidad basada en rutas.

<b>Nombre:</b> Modificar el número de vehículos
<b>Descripción:</b> Se puede modificar el número de vehículos de la simulación.

<b>Nombre:</b> Modificar la cobertura de los dispositivos móviles
<b>Descripción:</b> Se puede modificar la cobertura de los dispositivos integrados en los vehículos.

<b>Nombre:</b> Modificar la cobertura de los dispositivos fijos
<b>Descripción:</b> Se puede modificar la cobertura de los dispositivos integrados en los elementos fijos de soporte para las comunicaciones.

<b>Nombre:</b> Modificar la cobertura de los dispositivos de los nodos
<b>Descripción:</b> Se puede modificar la cobertura de los dispositivos integrados en los nodos.

<b>Nombre:</b> Añadir elementos fijos de soporte a la comunicaciones
<b>Descripción:</b> Se puede añadir a la simulación elementos fijos de soporte en las comunicaciones de la VANET.

<b>Nombre:</b> Modificar parámetros de simulación con fichero
<b>Descripción:</b> Se puede modificar los parámetros de la simulación a través de un fichero de configuración.

<b>Nombre:</b> Modificar información a visualizar del vehículo
<b>Descripción:</b> Se puede modificar la información del vehículo a visualizar.

<b>Nombre:</b> Mostrar la barra de control de la simulación
<b>Descripción:</b> Se puede mostrar la barra de control de la simulación.

<b>Nombre:</b> Ocultar la barra de control de la simulación
<b>Descripción:</b> Se puede ocultar la barra de control de la simulación.

<b>Nombre:</b> Salir del simulador
<b>Descripción:</b> Se puede cerrar el simulador.

<b>Nombre:</b> Acceder con combinación de teclas
<b>Descripción:</b> Se puede acceder a diversas funcionalidades del simulador a través de combinaciones de teclas.

<b>Nombre:</b> Mostrar información del simulador
<b>Descripción:</b> Se puede mostrar información acerca del simulador (creadores y versión).

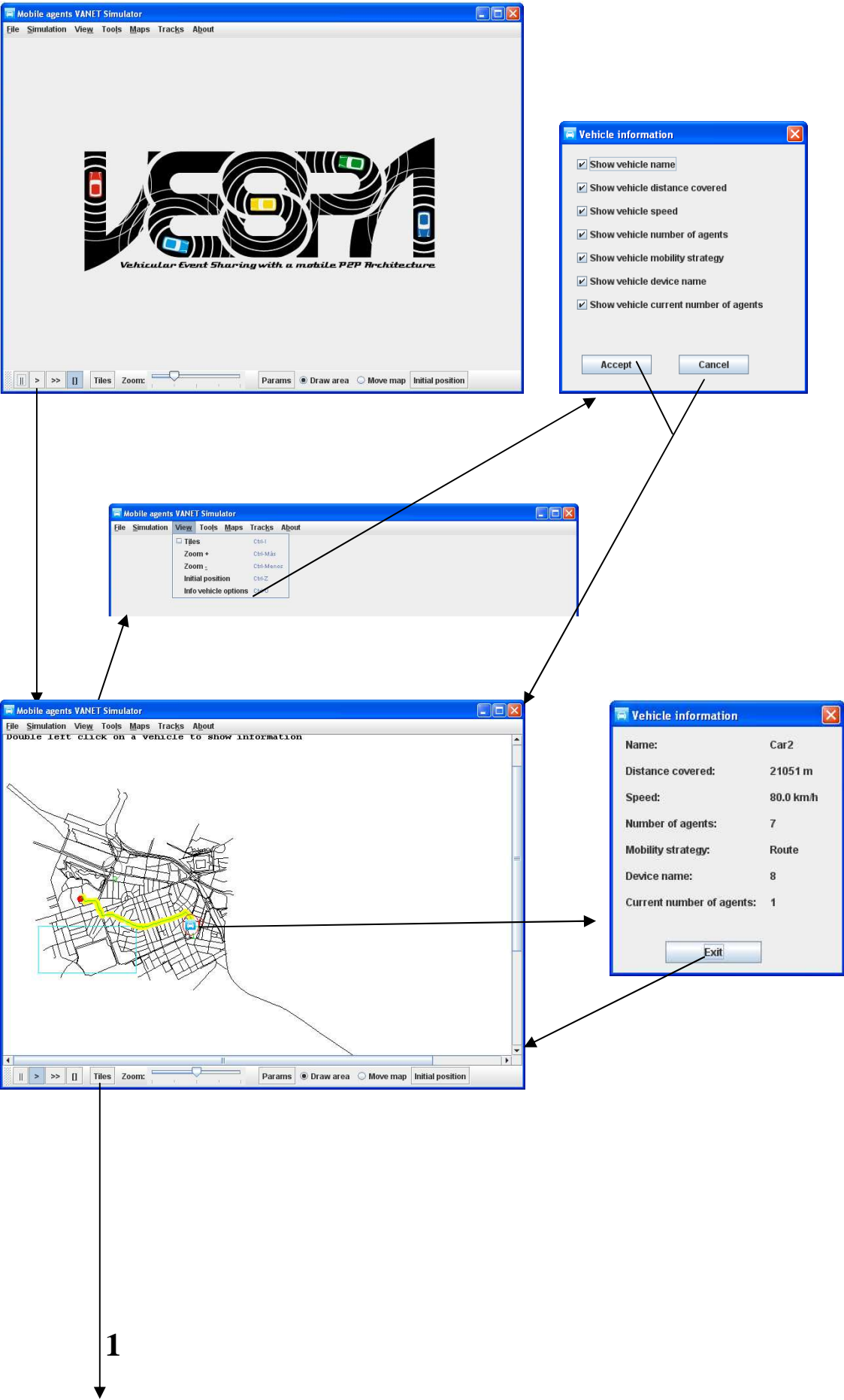
<b>Nombre:</b> Simular en modo línea de comandos ( <i>batch</i> )
<b>Descripción:</b> Se puede ejecutar el simulador en modo <i>batch</i> para realizar simulaciones por lotes.

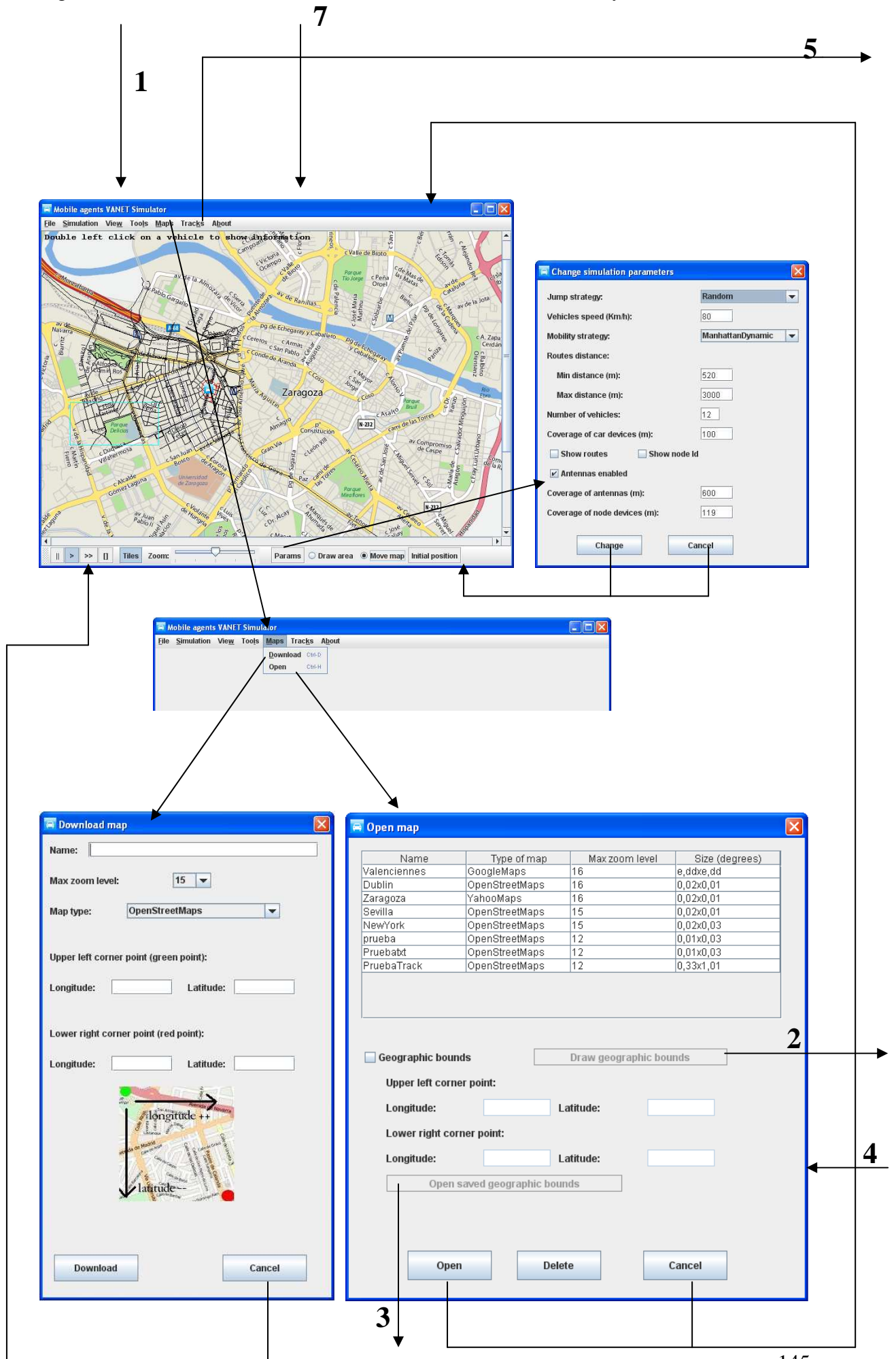
### C.3. Prototipado de ventanas

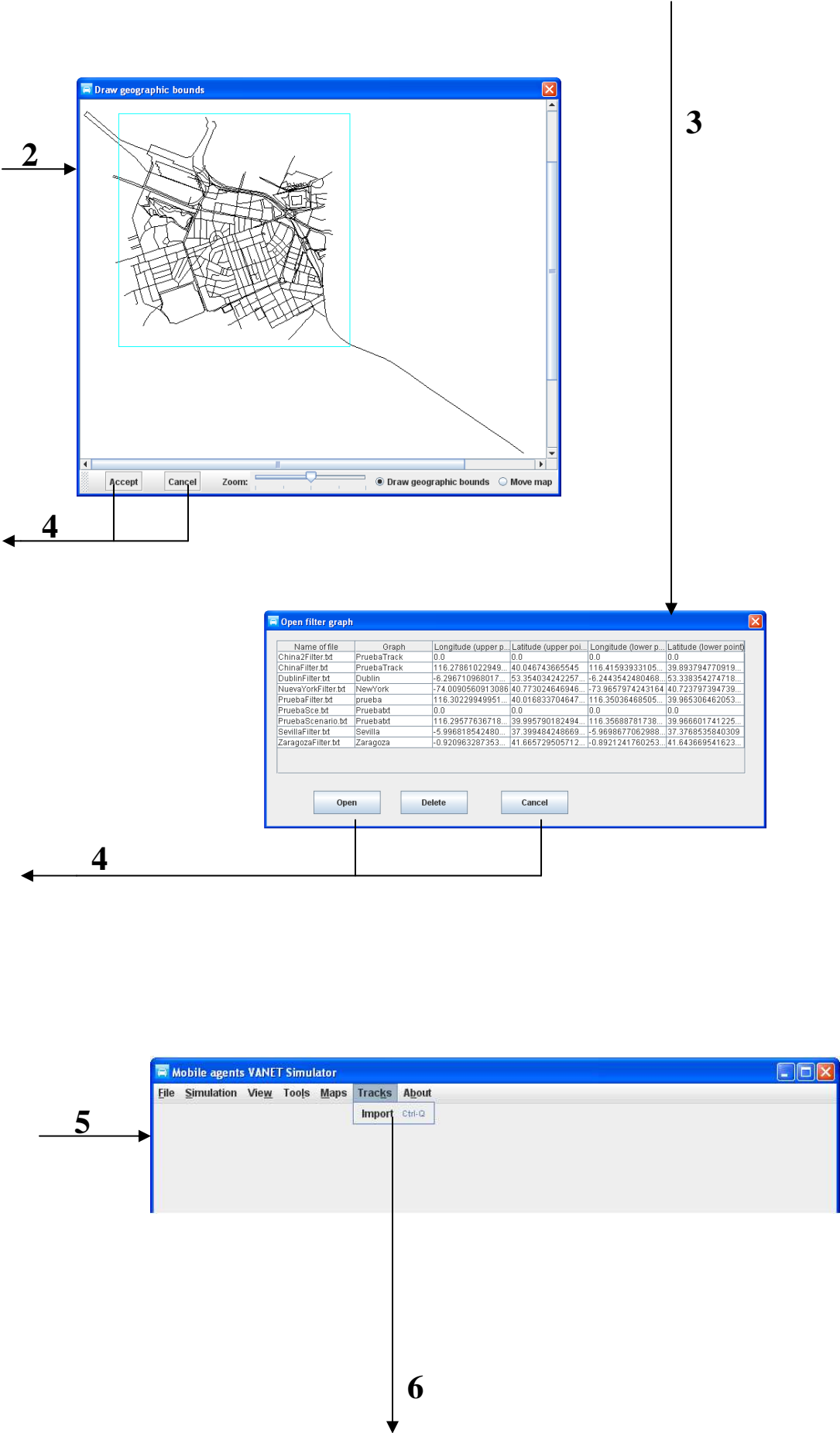
---

En esta sección se trata el prototipado de las ventanas. El prototipo (simulador) inicial contaba solamente con la ventana de simulación y una pequeña barra de control de simulación. Debido a que se añadieron herramientas y diversas funcionalidades era necesario implementar una interfaz que no tuviera un número excesivo de ventanas pero sí el suficiente. Para el diseño de las diversas ventanas se realizaron bocetos de las mismas para facilitar la posterior implementación.

A continuación, se muestra un diagrama general de navegación.







6

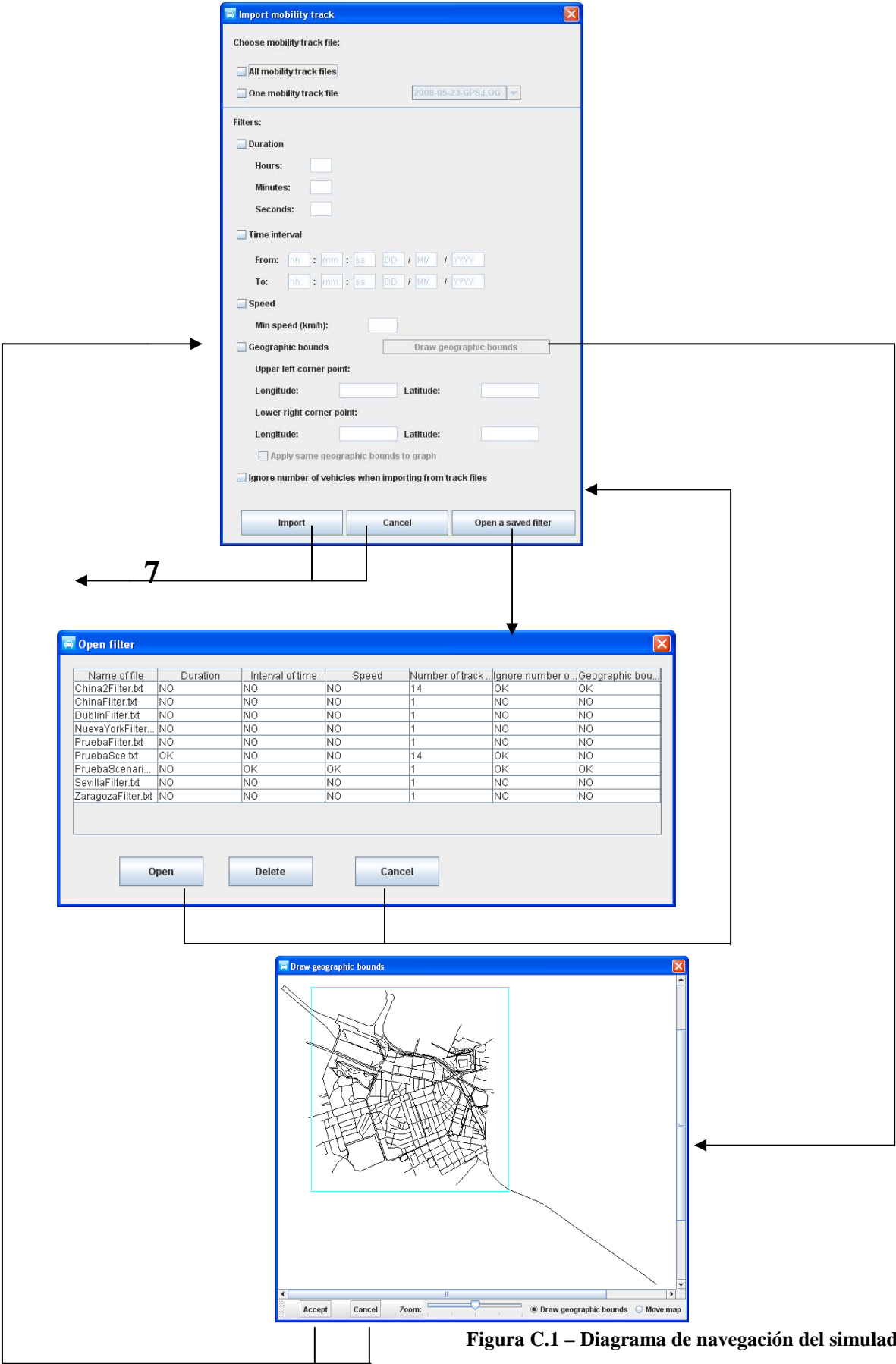


Figura C.1 – Diagrama de navegación del simulador





## ANEXO D – DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

---

En este anexo se van a explicar de forma breve las clases que forman parte de esta aplicación, cómo están estructuradas las diferentes carpetas que conforman el simulador y la estructura de las imágenes que conforman los mapas almacenados en el simulador.

### D.1. Clases del sistema

---

El prototipo inicial sobre el que se partió tenía todas las clases al mismo nivel y se consideró oportuno organizarlas en “packages” para su mejor gestión y organización.

Las clases que componen el sistema están agrupadas en “packages” dependiendo de su funcionalidad para una mejor gestión del código fuente. Todos estos “packages” se pueden encontrar dentro de la carpeta “SRC” que es donde se guarda el código fuente de la aplicación.

El simulador tiene los siguientes módulos o “packages”:

- **Agent:** módulo que engloba las clases que gestionan los agentes.
- **Antenna:** módulo en el que se encuentran las clases que gestionan los elementos fijos de soporte en las comunicaciones de la VANET.
- **Area:** módulo en el que se encuentran las clases que gestionan las áreas de monitorización.
- **Device:** módulo que engloba las clases que gestionan todo lo referente a los dispositivos.
- **File:** módulo que engloba las clases encargadas del tratamiento de los diferentes ficheros de la aplicación.
- **Global:** módulo en el que se encuentra la clase encargada de la gestión de las variables globales.
- **Graph:** módulo en el que se encuentran las clases que gestionan todo lo referente al grafo del mapa.
- **Graphics:** módulo que engloba las clases que gestionan la interfaz.
- **Mobility:** módulo que engloba las clases que gestionan todo lo referente a las estrategias de movilidad.
- **Track:** módulo en el que se encuentran las clases encargadas de gestionar las trazas y su importación.

- **Vehicle:** módulo en el que se encuentran las clases que gestionan todo lo referente a los vehículos.

Al mismo nivel que los módulos anteriores se encuentra la clase “Simulator” que es la clase principal de la aplicación.

#### D.1.1. Agent

---

Las clases que componen el paquete “Agent” fueron implementadas en su totalidad en el prototipo anterior. Sólo hubo que realizar unas ligeras modificaciones de adaptación. Las clases que pertenecen a este módulo son las siguientes:

- Agent
- AgentFlood
- AgentMonitor
- AgentQuery2
- AgentQuery
- AgentTest
- IAgent.

#### **Agent**

Esta clase es una clase abstracta que implementa la interfaz “IAgent”. Aquí están implementados todos los métodos fundamentales del agente que le permiten moverse, o lo que es lo mismo saltar de un lugar a otro.

Esta clase fue implementada en el prototipo anterior, sólo se realizaron unas ligeras modificaciones sobre ella.

#### **AgentFlood**

Esta clase es hija de la clase “Agent”. En ella se implementan los métodos que la diferencian de las otras clases hijas de “Agent” para realizar una funcionalidad determinada. Este agente tiene de especial que es un agente no móvil que simula el envío y recepción de mensajes a otros agentes.

Esta clase fue implementada en el prototipo anterior, y no se realizó ninguna modificación sobre ella.

### **AgentMonitor**

Esta clase es hija de la clase “Agent”. En ella se implementan los métodos que la diferencian de las otras clases hijas de “Agent” para realizar una funcionalidad determinada. Este agente tiene de especial que es un agente móvil que se mueve hacia el área de monitorización para tomar muestras hasta que el tiempo de tomar muestras expira.

Esta clase fue implementada en el prototipo anterior, y no se realizó ninguna modificación sobre ella.

### **AgentQuery2**

Esta clase es hija de la clase “Agent”. En ella se implementan los métodos que la diferencian de las otras clases hijas de “Agent” para realizar una funcionalidad determinada. Este agente simula un agente móvil con un comportamiento especial. Primero se dirige hacia el área de monitorización en la que debe estar durante un número de iteraciones para después comenzar a recoger toda la información. Para ello debe visitar un número de vehículos con información relevante y una vez que los ha visitado puede volver al lugar de origen que puede ser un dispositivo fijo integrado en un nodo o uno móvil integrado en un vehículo.

Esta clase fue implementada en el prototipo anterior, y no se realizó ninguna modificación sobre ella.

### **AgentQuery**

Esta clase es hija de la clase “Agent”. En ella se implementan los métodos que la diferencian de las otras clases hijas de “Agent” para realizar una funcionalidad determinada. Este agente simula un agente móvil con un comportamiento especial prácticamente igual que el del “AgentQuery2”. La principal diferencia en el comportamiento de ambos es que éste sólo debe llegar hasta el área de monitorización y no tiene que permanecer allí un número de iteraciones.

Esta clase fue implementada en el prototipo anterior, y no se realizó ninguna modificación sobre ella.

### **AgentTest**

Esta clase es hija de la clase “Agent”. En ella se implementan métodos básicos que pueden servir de plantilla o ejemplo para otros agentes.

Esta clase fue implementada en el prototipo anterior, y no se realizó ninguna modificación sobre ella.

## IAgent

Esta clase implementa una interfaz para las clases de los agentes. La interfaz contiene 2 métodos: “agentInitializationCode” y “agentRunCodeIteration”.

Esta clase fue implementada en el prototipo anterior, y no se realizó ninguna modificación sobre ella.

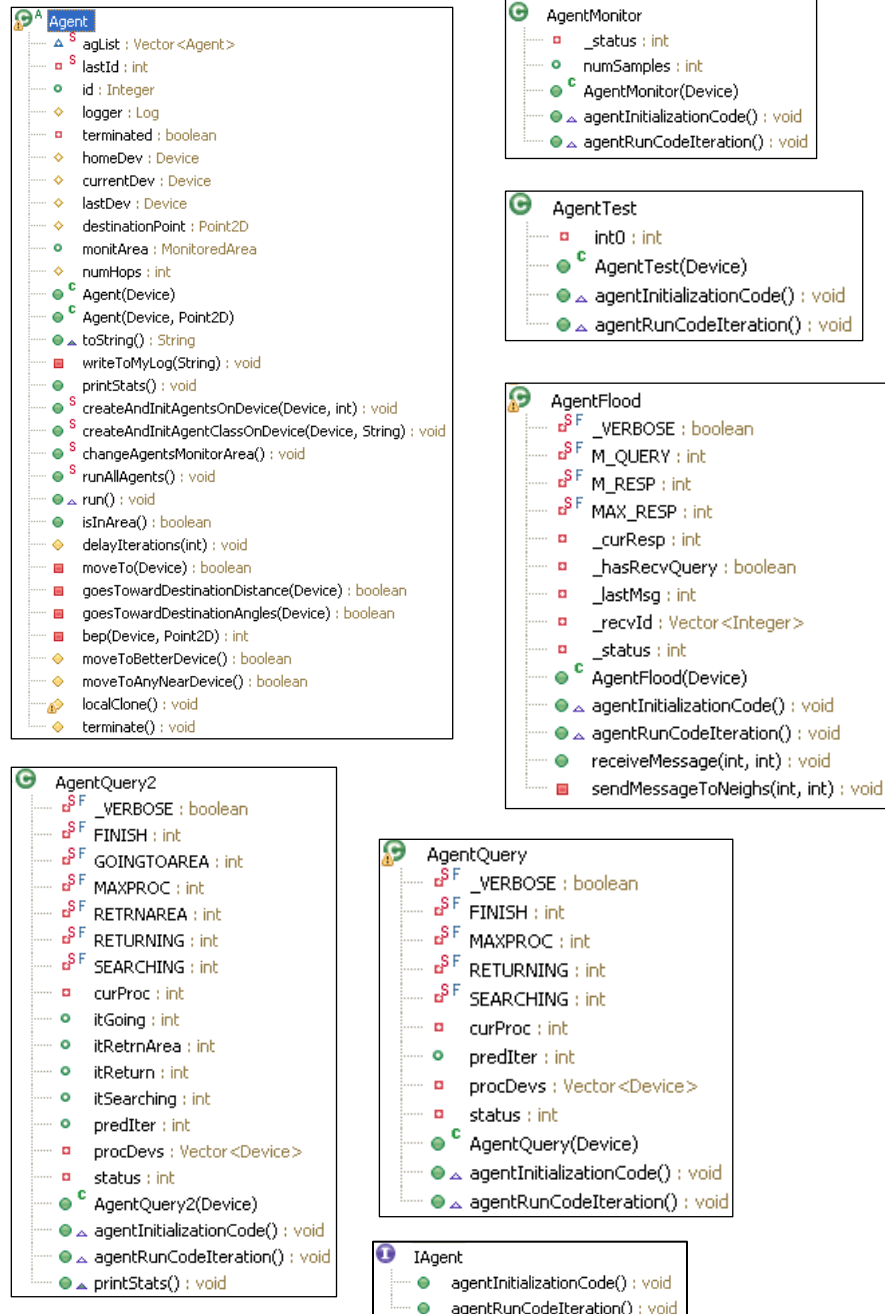


Figura D.1 – Clases del módulo *Agent*

### D.1.2. Antenna

---

Las clases que componen el paquete “Antenna” son las encargadas de gestionar lo referente a los dispositivos fijos de soporte para las comunicaciones en la VANET. Las clases que pertenecen a este módulo son las siguientes:

- AllAntennas
- Antenna

#### AllAntennas

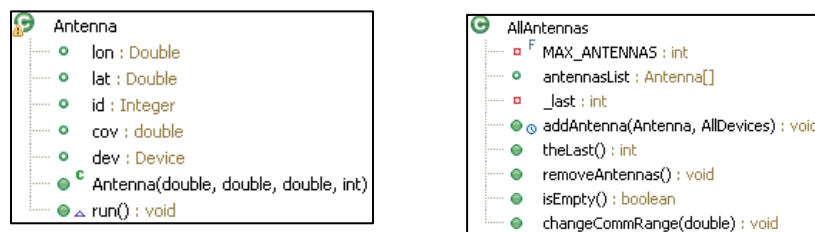
Esta clase es la encargada de la gestión básica de las antenas o dispositivos fijos de soporte en las comunicaciones. Tiene los métodos encargados de añadir y eliminar antenas y cambiar la cobertura de las mismas, entre otros métodos.

Esta clase fue implementada íntegramente para este PFC.

#### Antenna

Esta clase es la encargada de simular el funcionamiento de cada antena. Tiene diversos atributos que la sitúan dentro de un mapa (longitud y latitud) y le otorga una cobertura al dispositivo (“device”) que tiene integrado, que es el encargado de realizar las comunicaciones con otros elementos de la VANET que tienen su dispositivo integrado.

Esta clase fue implementada íntegramente para este PFC.



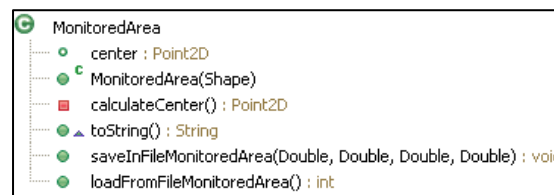
**Figura D.2 – Clases del módulo *Antenna***

### D.1.3. Area

---

La clase que compone el paquete “Area” se llama “MonitoredArea”. Esta clase gestiona lo referente al área de monitorización. Tiene los métodos necesarios para cargar un área de monitorización de un fichero, guardarla en un fichero...

Esta clase fue implementada en el prototipo anterior, y se añadieron los métodos necesarios para cargar un área de monitorización desde un fichero y para guardarla en un fichero.



**Figura D.3 – Clase del módulo Area**

### D.1.4. Device

---

Las clases que componen el paquete “Device” son las encargadas de gestionar lo referente a los dispositivos que van “integrados” en los vehículos, en los nodos y en las antenas. Las clases que pertenecen a este módulo son las siguientes:

- AllDevices
- Device

#### AllDevices

Esta clase es la encargada de la gestión básica de los dispositivos que van a estar en los nodos, vehículos y antenas. Tiene métodos para añadir dispositivos a nodos aleatorios, a nodos fijos, a vehículos, cambiar la cobertura, calcular la proximidad con otros dispositivos, etc.

Esta clase fue implementada en el prototipo anterior, y se añadieron los métodos necesarios para cambiar la cobertura.

## Device

Esta clase es la encargada del funcionamiento de un dispositivo. Puede estar integrado en un vehículo o en un elemento fijo. Sobre los dispositivos están los agentes y por lo tanto, deben gestionar lo referente a las llegadas y salidas de los agentes dentro del dispositivo.

Esta clase fue implementada en el prototipo anterior, y se añadieron los métodos necesarios para posibilitar los dispositivos en las antenas.



Figura D.4 – Clases del módulo *Device*

### D.1.5. File

Las clases que componen el paquete “File” son las encargadas de gestionar el tratamiento de los ficheros que se usan en el simulador. Las clases que pertenecen a este módulo son las siguientes:

- Antenna\_file
- DownloadedMaps\_file
- Folder
- Log
- Scenario\_file
- Xml\_file

**Antenna\_file**

Esta clase es la encargada de leer el fichero correspondiente con la información sobre las antenas para que el simulador las pueda añadir a la simulación.

Esta clase fue implementada íntegramente para este PFC.

**DownloadedMaps\_file**

Esta clase es la encargada de leer el fichero “downloadedMaps.txt” y extraer la información necesaria para listar los mapas disponibles junto con sus características para realizar simulaciones.

Esta clase fue implementada íntegramente para este PFC.

**Folder**

Esta clase es la encargada de la gestión de todo lo referente a las carpetas de la aplicación. Sus métodos son los encargados de crear el árbol de directorios donde se almacenarán las imágenes de los mapas, eliminar todos los archivos de un mapa cuando el usuario lo decide desde la interfaz, etc.

Esta clase fue implementada íntegramente para este PFC.

**Log**

Esta clase es la encargada de crear el fichero de registro o log.

Esta clase fue implementada en el prototipo anterior y se realizaron unas ligeras modificaciones para adaptarla a la estructura de carpetas de este PFC.

**Scenario\_file**

Esta clase es la encargada de leer los ficheros donde se almacenan los filtros para los mapas y las trazas. De ellos extrae la información necesaria para listar todos los filtros existentes junto con sus características.

Esta clase fue implementada íntegramente para este PFC.

**Xml\_file**

Esta clase es la encargada de leer el fichero de configuración “config.xml”, crearlo y modificarlo.

Esta clase fue implementada íntegramente para este PFC.

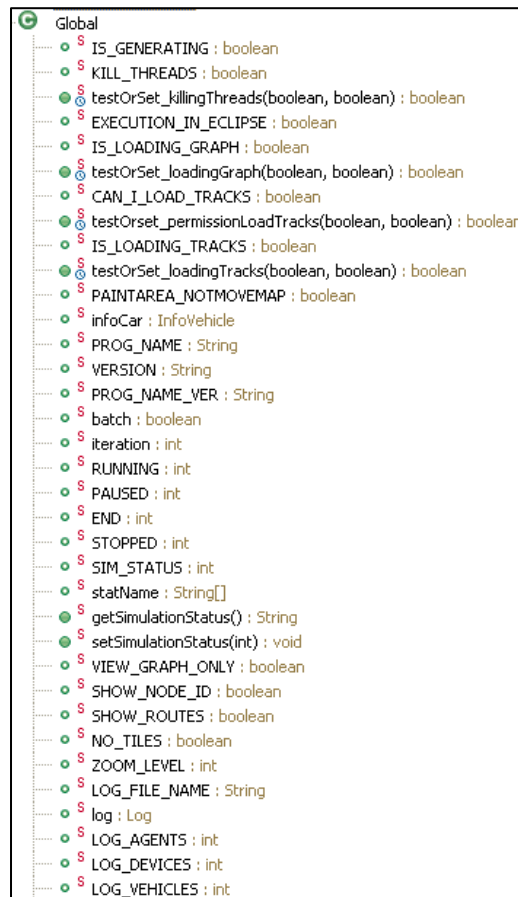


Figura D.5 – Clases del módulo *File*

### D.1.6. Global

La clase que compone el paquete “Global” es la encargada de almacenar todas las variables (atributos) globales. Esta clase se llama “Global” y en ella podemos encontrar desde una cadena donde se guarda la versión del simulador hasta el número máximo de iteraciones de cada simulación.

Esta clase fue implementada en el prototipo anterior, y se añadieron métodos y atributos necesarios para diversas funcionalidades, como por ejemplo sincronizar los *threads* cuando se quieren matar o sincronizar la carga de un nuevo mapa.



**Figura D.6 – Clase del módulo *Global***

En la figura anterior no aparecen todos los atributos de la clase *Global* debido a su gran extensión.

### D.1.7. Graph

---

Las clases que componen el paquete “Graph” son las encargadas de gestionar lo referente al grafo donde se almacena la información de las calles y carreteras del mapa. Las clases que pertenecen a este módulo son las siguientes:

- Graph
- GraphFilter
- GraphEdge
- GraphNode

#### **Graph**

Esta clase es la encargada de leer el fichero que contiene la información de las carreteras y calles de un mapa y almacenarlo en las estructuras necesarias para utilizar dicha información en el simulador.

Esta clase fue implementada en el prototipo anterior, y se añadieron los métodos necesarios para posibilitar la lectura de los mapas OSM descargados y para poder aplicar los filtros geográficos sobre los mapas.

#### **GraphFilter**

Esta clase es la encargada de almacenar la información necesaria para aplicar los filtros geográficos a los mapas.

Esta clase fue implementada íntegramente para este PFC.

#### **GraphEdge**

Esta clase es la encargada de almacenar la información necesaria sobre las relaciones entre los nodos del grafo. Es como si se guardara el segmento que une 2 nodos.

Esta clase fue implementada en el prototipo anterior y no se modificó.

## GraphNode

Esta clase es la encargada de almacenar la información necesaria sobre los nodos del grafo. Contiene un método que se encarga de calcular la distancia entre el nodo almacenado en la clase y otro nodo que se indica.

Esta clase fue implementada en el prototipo anterior y se le añadió el método capaz de calcular la distancia entre 2 nodos.



**Figura D.7 – Clases del módulo *Graph***

### D.1.8. Graphics

---

Las clases que componen el paquete “Graphics” son las encargadas de gestionar lo referente a la interfaz de la aplicación. Las clases que pertenecen a este módulo son las siguientes:

- DownloaderThread
- MapTiles
- MenuBar
- SwGraphics
- ToolBar
- ToolBar\_DrawGeoBounds
- WindowDownloadMap
- WindowDrawGeoBounds
- WindowImportTrack
- WindowInfoCar
- WindowInfoCarOptions
- WindowOpenFilter
- WindowOpenFilterGraph
- WindowOpenMap
- WindowVersion

#### **DownloaderThread**

Esta clase es la encargada de la descarga de las imágenes o *tiles* que componen un mapa. Se le pasan las características del mapa que se quiere descargar (tipo de mapa, nivel de zoom máximo, longitudes y latitudes), el nombre con el que se quiere guardar y la ruta, y esta clase se encarga de la descarga y almacenamiento de las imágenes. Además, se encarga de mostrar una barra de progreso que indica en qué estado se encuentra la descarga.

Esta clase fue implementada íntegramente para este PFC.

### **MapTiles**

Esta clase es la encargada de la descarga de las imágenes o *tiles* que componen un mapa desde el servidor correspondiente. Se le pasan las características de la *tile* que se quiere descargar y esta clase la descarga. Además, tiene un método para calcular la *tile* a partir de la longitud o latitud y el nivel de zoom y viceversa.

Esta clase fue implementada en el prototipo anterior y se le añadieron los métodos necesarios para descargar cualquier tipo de mapa de los soportados (OpenStreetMap, GoogleMaps y MicrosoftMaps).

### **MenuBar**

Esta clase es la encargada de la gestión de la barra de menú situada en la parte superior de la ventana de la aplicación. En algunas acciones debe estar sincronizada con la barra de control de simulación situada en la parte inferior de la ventana de la aplicación.

Esta clase fue implementada íntegramente para este PFC.

### **SwGraphics**

Esta clase es la clase principal de la interfaz. Es la encargada de crear la ventana de la aplicación sobre la que se muestra la simulación (grafo, vehículos, etc.) y todos los menús. Usa la librería “Swing” para la creación de todos los gráficos.

Esta clase fue implementada en el prototipo anterior y se le añadieron los métodos y atributos necesarios para implementar las nuevas funcionalidades. Además, se tuvieron que modificar una gran cantidad de métodos ya implementados para adaptarlos a la nueva interfaz.

### **ToolBar**

Esta clase es la encargada de la gestión de la barra de control de la simulación, es decir, la barra situada en la parte inferior de la ventana de la aplicación. Esta barra debe estar sincronizada en algunas de sus funciones con la barra de menú comentada anteriormente. Además, es la encargada de la ventana en la se pueden modificar los parámetros de simulación.

Esta clase fue implementada en el prototipo anterior y se le añadieron los métodos y atributos necesarios para implementar las nuevas funcionalidades. Además, se tuvieron que modificar bastantes métodos ya implementados para adaptarlos a la nueva interfaz.

**ToolBar DrawGeoBounds**

Esta clase es la encargada de la barra de control que aparece en la parte inferior de la ventana que permite dibujar el filtro geográfico tanto para los mapas como para las trazas.

Esta clase fue implementada íntegramente para este PFC.

**WindowDownloadMap**

Esta clase es la encargada de la ventana desde la que se descargan los mapas. Además de encargarse de los elementos de la ventana, se encarga del control de errores de los datos introducidos por el usuario. Por ejemplo, comprueba que la latitud del punto situado en la posición inferior derecha sea menor que la del punto situado en la posición superior izquierda; de asegurarse que se introduzcan números en las coordenadas geográficas en vez de letras, etc.

Esta clase fue implementada íntegramente para este PFC.

**WindowDrawGeoBounds**

Esta clase es la encargada de la ventana que permite dibujar el filtro geográfico para los mapas y las trazas.

Esta clase fue implementada íntegramente para este PFC.

**WindowImportTrack**

Esta clase es la encargada de la importación de las trazas. Además de encargarse de todos los elementos que aparecen en esta ventana, se encarga del control de errores que pueden surgir cuando el usuario utiliza esta ventana. Por ejemplo, comprueba que si el usuario ha seleccionado usar un tipo de filtro, realmente introduzca valores para ese filtro y que sean válidos.

Esta clase fue implementada íntegramente para este PFC.

**WindowInfoCar**

Esta clase es la encargada de mostrar la información, que previamente se haya configurado, de un vehículo. Esta ventana tiene de especial que se adapta a la cantidad de información que se va a mostrar, es decir, que ajusta su tamaño dependiendo del número de atributos que se van a mostrar de un vehículo cuando se haga doble clic sobre uno.

Esta clase fue implementada íntegramente para este PFC.

**WindowInfoCarOptions**

Esta clase es la encargada de la configuración de la información que se visualizará cuando se realice doble clic sobre un vehículo. Cuando se habla de la configuración de la información se quiere decir que el usuario decide qué información mostrar de los vehículos.

Esta clase fue implementada íntegramente para este PFC.

**WindowOpenFilter**

Esta clase es la encargada de la ventana donde se listan todos los filtros existentes para aplicar a las trazas. En ella se muestra una lista con todos los filtros y sus características y además unos botones para la gestión de estos filtros.

Esta clase fue implementada íntegramente para este PFC.

**WindowOpenFilterGraph**

Esta clase es la encargada de la ventana donde se listan todos los filtros existentes para aplicar a los mapas. En ella se muestra una lista con todos los filtros y sus características y además unos botones para la gestión de estos filtros.

Esta clase fue implementada íntegramente para este PFC.

**WindowOpenMap**

Esta clase es la encargada de la ventana que permite abrir nuevos mapas para usar en las simulaciones. Esta ventana lista todos los mapas existentes en el simulador junto con las características de cada uno. Además, añade la posibilidad de aplicar los filtros geográficos a los mapas y comprobar los errores que el usuario pueda cometer al moverse por esta ventana.

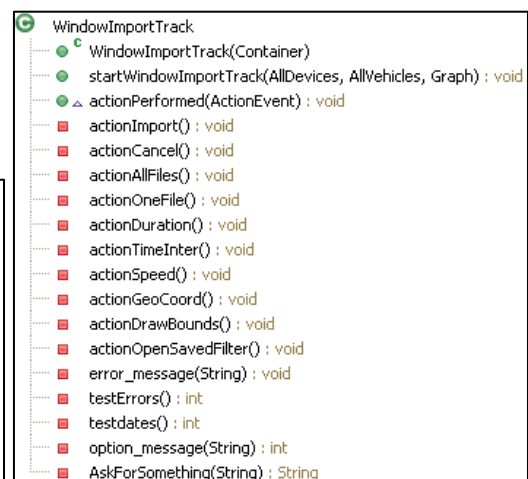
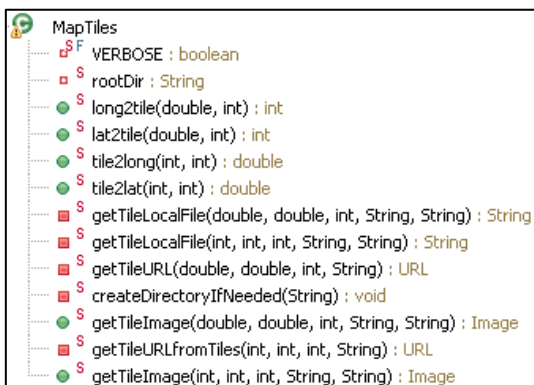
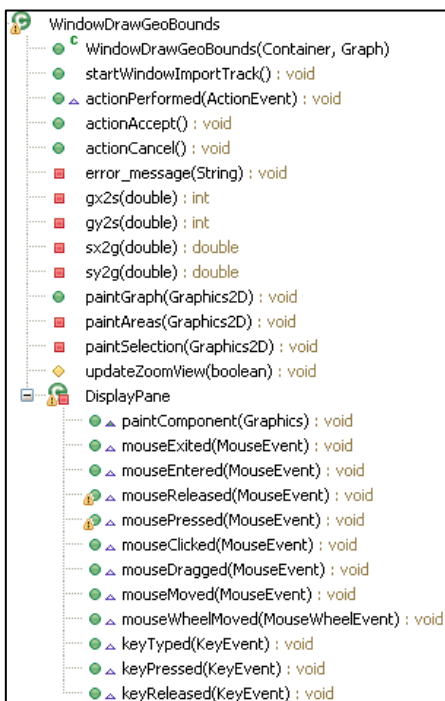
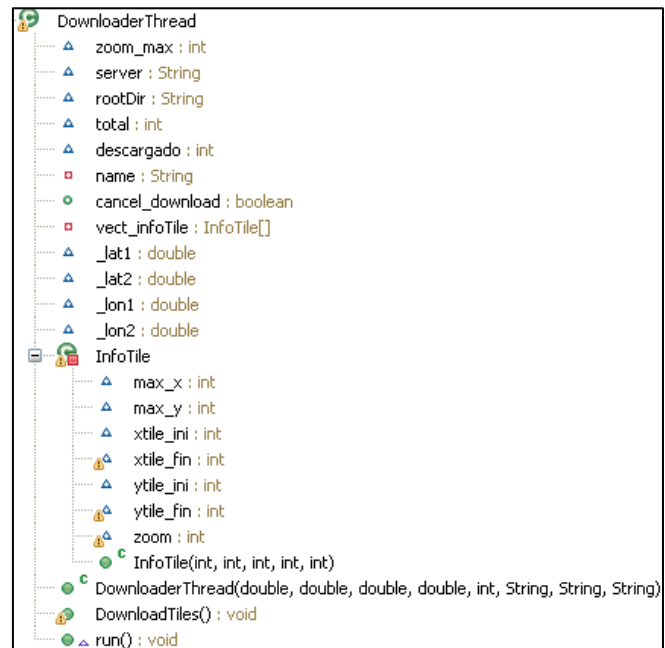
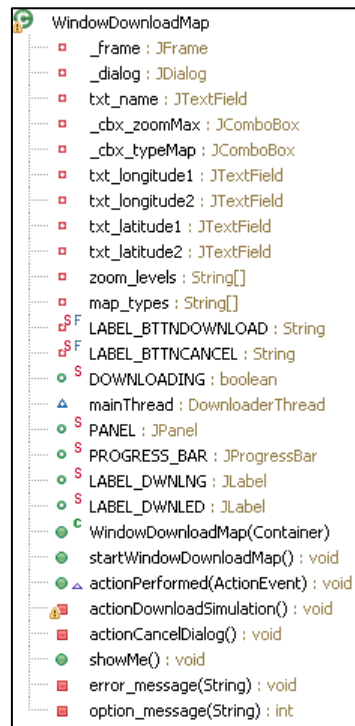
Esta clase fue implementada íntegramente para este PFC.

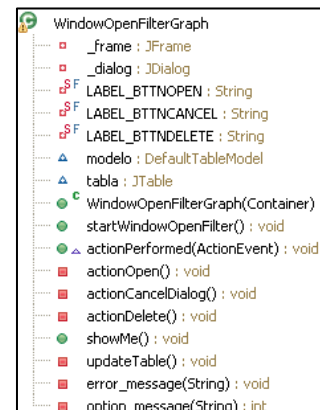
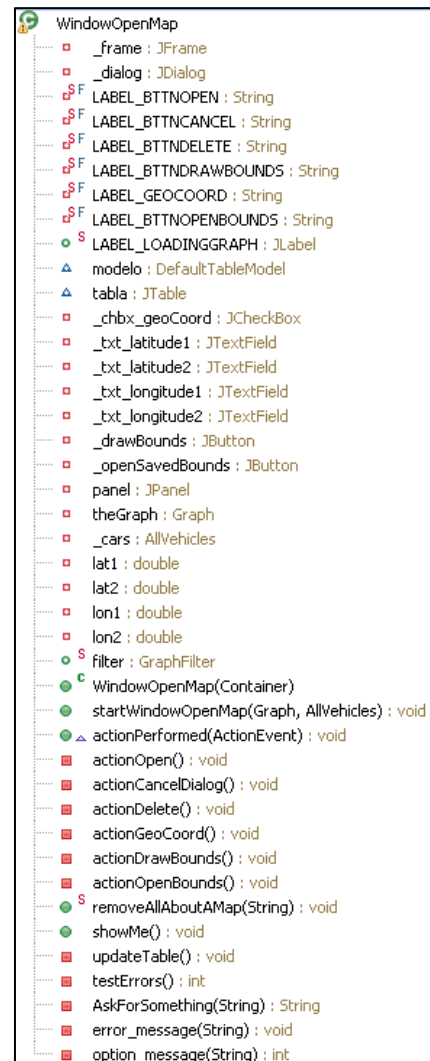
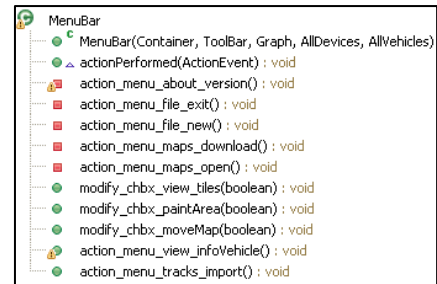
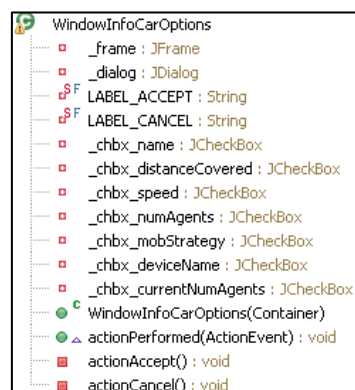
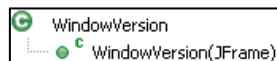
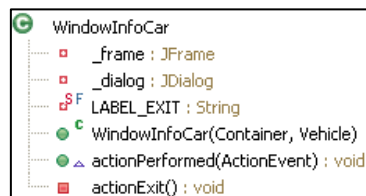
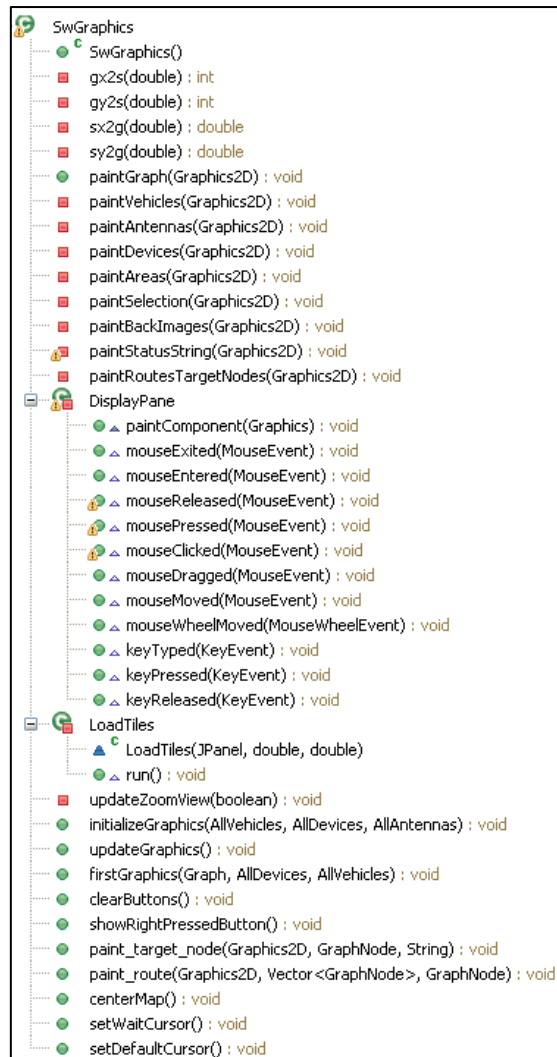
**WindowVersion**

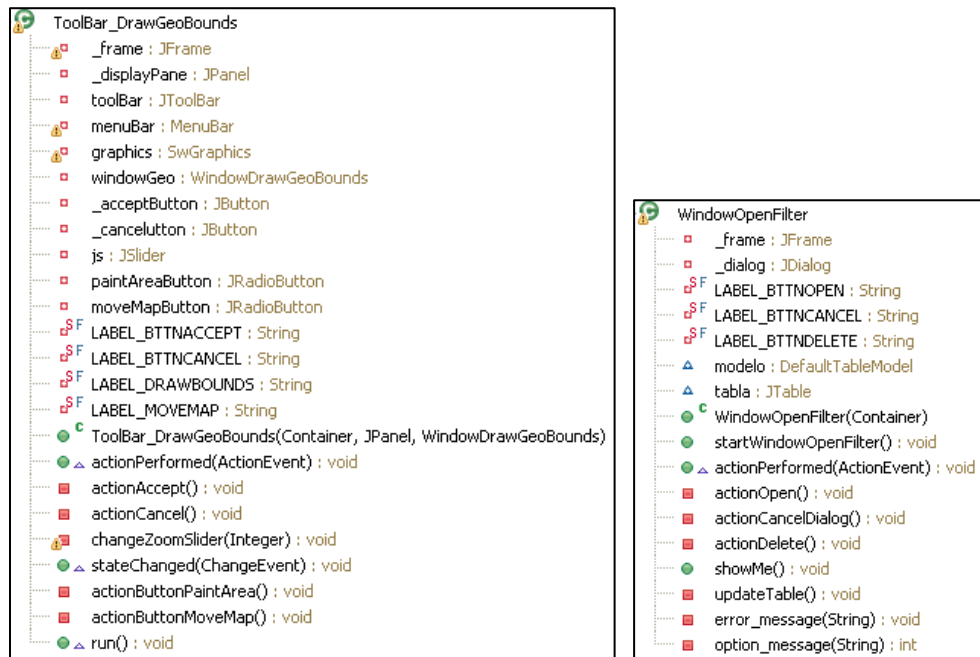
Esta clase es la encargada de la ventana que muestra la información sobre la aplicación (creadores y versión).

Esta clase fue implementada íntegramente para este PFC.









**Figura D.8 – Clases del módulo *Graphics***

En algunas de las clases anteriores se ha optado por omitir los atributos de las mismas debido a su gran cantidad.

### D.1.9. Mobility

---

Las clases que componen el paquete “Mobility” son las encargadas de gestionar lo referente a las estrategias de movilidad de los vehículos. Las clases que pertenecen a este módulo son las siguientes:

- Mob\_GaussMarkov
- Mob\_heuristicAndDestination
- Mob\_ManhattanDynamic
- Mob\_randomAdjacentNode
- Mob\_routes
- Mob\_straightAdjacentNode
- Mob\_track
- Mobility
- Route

#### **Mob\_GaussMarkov**

Esta clase es la encargada de la estrategia de movilidad “Gauss-Markov”. Esta clase es una clase hija de la clase “Mobility”.

Esta clase fue implementada íntegramente para este PFC.

#### **Mob\_heuristicAndDestination**

Esta clase es la encargada de la estrategia de movilidad que se basa en una heurística para alcanzar un nodo destino. Dota a los vehículos de un movimiento similar al de la clase “Mob\_routes. Esta clase es una clase hija de la clase “Mobility”.

Esta clase fue implementada íntegramente para este PFC.

#### **Mob\_ManhattanDynamic**

Esta clase es la encargada de la estrategia de movilidad “Manhattan”. Esta clase es una clase hija de la clase “Mobility”.

Esta clase fue implementada íntegramente para este PFC.

**Mob\_randomAdjacentNode**

Esta clase es la encargada de la estrategia de movilidad “Random”. Esta clase es una clase hija de la clase “Mobility”.

Esta clase fue implementada en el prototipo anterior, y se adaptó para que fuera una clase hija de la clase “Mobility”.

**Mob\_routes**

Esta clase es la encargada de la estrategia de movilidad basada en rutas aplicando el algoritmo del camino más corto de Dijkstra. Esta clase es una clase hija de la clase “Mobility”.

Esta clase fue implementada íntegramente para este PFC.

**Mob\_straightAdjacentNode**

Esta clase es la encargada de la estrategia de movilidad “Straight”. Esta clase es una clase hija de la clase “Mobility”.

Esta clase fue implementada en el prototipo anterior, y se adaptó para que fuera una clase hija de la clase “Mobility”.

**Mob\_track**

Esta clase es la encargada de que los vehículos puedan usar las trazas importadas desde los ficheros. Con esta clase, los vehículos siguen las trazas como si fueran rutas precalculadas. Esta clase es una clase hija de la clase “Mobility”.

Esta clase fue implementada íntegramente para este PFC.

**Mobility**

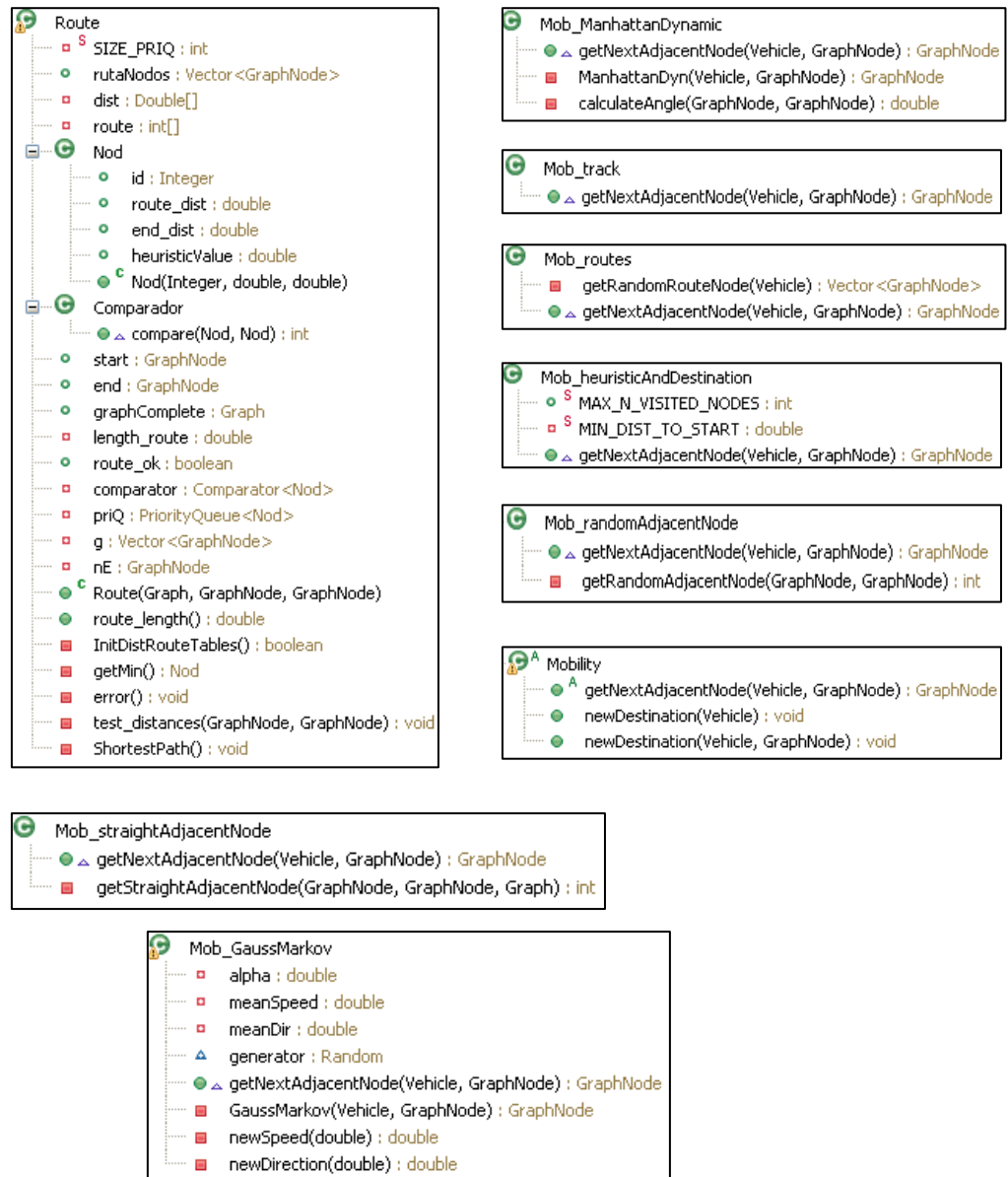
Esta clase es una clase abstracta de la que todas las clases que implementan una estrategia de movilidad heredan sus métodos.

Esta clase fue implementada íntegramente para este PFC.

**Route**

Esta clase es la encargada de calcular la ruta con el algoritmo de Dijkstra del camino más corto. Posee los atributos y métodos necesarios para aplicar el algoritmo de Dijkstra (tabla de rutas, tabla de distancias...).

Esta clase fue implementada íntegramente para este PFC.

Figura D.9 – Clases del módulo *Mobility*

### D.1.10. Track

---

Las clases que componen el paquete “Track” son las encargadas de gestionar lo referente a la importación de las trazas. Las clases que pertenecen a este módulo son las siguientes:

- Track
- TrackFilter
- TrackNode
- TrackReader
- TrackReader\_log
- TrackReader\_plt
- TrackReader\_txt
- TrackVehicle

#### **Track**

Esta clase es la encargada de la gestión principal de las trazas. Detecta el formato del fichero de trazas y usa la clase respectiva para que se encargue de la lectura de las trazas y la importación de las mismas. Además, es la encargada de asignar la traza de movimiento leída en cada fichero a cada vehículo, aunque cabe la posibilidad de que la clase encargada de leer el fichero de trazas con el formato que sea también pueda asignar la traza de movimiento al vehículo.

Esta clase fue implementada íntegramente para este PFC.

#### **TrackFilter**

Esta clase es la encargada del almacenamiento de cualquier combinación de los filtros posibles para las trazas.

Esta clase fue implementada íntegramente para este PFC.

#### **TrackNode**

Esta clase es la encargada de almacenar la información necesaria de cada nodo que compone la traza de movimiento que posteriormente usará el vehículo para seguir la traza. Los atributos más importantes de esta clase son “node”, “speed” y “dateAndHour”, que contienen

información sobre las coordenadas geográficas de esa posición, la velocidad y la fecha y hora respectivamente.

Esta clase fue implementada íntegramente para este PFC.

### **TrackReader**

Esta clase es una clase abstracta que contiene los métodos y atributos necesarios para aquellas clases hijas que los hereden. Las clases hijas serán las encargadas de leer cada formato de fichero de trazas.

Esta clase fue diseñada íntegramente para este PFC.

### **TrackReader log**

Esta clase es una clase hija de “TrackReader”. Está especializada en la lectura de los ficheros de trazas de formato “.log”.

Esta clase fue implementada íntegramente para este PFC.

### **TrackReader plt**

Esta clase es una clase hija de “TrackReader”. Está especializada en la lectura de los ficheros de trazas de formato “.plt”.

Esta clase fue implementada íntegramente para este PFC.

### **TrackReader txt**

Esta clase es una clase hija de “TrackReader”. Está especializada en la lectura de los ficheros de trazas de formato “.txt”. Debido a que en los ficheros de este formato aparece más de una traza independiente de las otras, esta clase es capaz de asignar la traza a un vehículo.

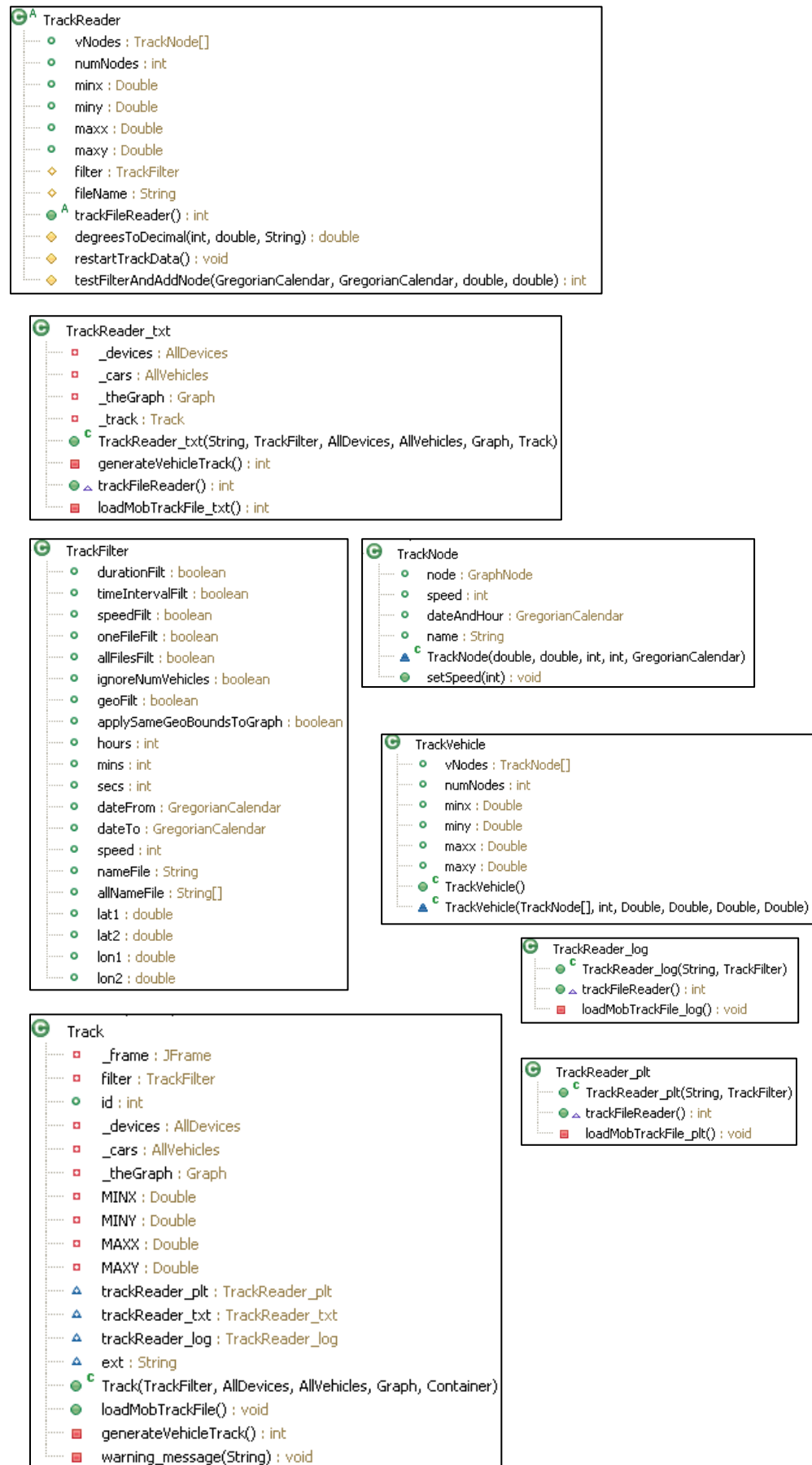
Esta clase fue implementada íntegramente para este PFC.

### **TrackVehicle**

Esta clase es la encargada de almacenar la información de la traza del fichero para que sea utilizable por el simulador y más en concreto por el vehículo.

Esta clase fue implementada íntegramente para este PFC.



Figura D.10 – Clases del módulo *Track*

### D.1.11. Vehicle

---

Las clases que componen el paquete “Vehicle” son las encargadas de gestionar lo referente a los vehículos. Las clases que pertenecen a este módulo son las siguientes:

- AllVehicles
- Vehicle
- InfoVehicle

#### **AllVehicles**

Esta clase es la encargada de la gestión básica de los vehículos. Tiene métodos para añadir vehículos a la simulación y asignarles una posición inicial aleatoria, asignarle a cada vehículo un hilo de ejecución, modificar su estrategia de movilidad, etc.

Esta clase fue implementada en el prototipo anterior, y se añadieron los métodos necesarios para poder generar vehículos con trazas, modificar su estrategia de movilidad, eliminar los vehículos de la simulación y un método de sincronización que permite al simulador iniciar la simulación automáticamente al importar trazas GPS.

#### **Vehicle**

Esta clase es la encargada del funcionamiento de un vehículo. Tiene los métodos necesarios para utilizar la estrategia de movilidad asignada, moverse a una determinada velocidad, moverse hacia un determinado lugar, etc.

Esta clase fue implementada en el prototipo anterior, y se añadieron los métodos y atributos necesarios para que los vehículos fueran capaces de seguir rutas, utilizar diferentes estrategias de movilidad, etc.

#### **InfoVehicle**

Esta clase es la encargada de la gestión de la información de cada vehículo que será visualizada cuando el usuario haga doble clic sobre un vehículo determinado.

Esta clase fue implementada íntegramente para este PFC.

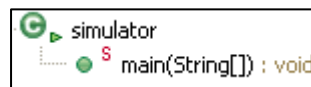
Figura D.11 – Clases del módulo *Vehicle*

### D.1.12. Clase principal: Simulator

---

La clase principal de la aplicación es la clase “Simulator” que es la que contiene el método “main”. En esta clase se crean todas las clases principales que se van a usar en el simulador, como por ejemplo: “AllDevices”, “AllVehicles”, “Global”, “SwGraphics”, “Xml\_file”, etc.

La estructura de esta clase consta principalmente de un bucle del que sólo se sale cuando el usuario decide salir de la aplicación. Dentro de este bucle se gestiona todo lo referente a la propia simulación, como por ejemplo: el tipo de experimento que se va a realizar durante la simulación, la creación de los vehículos, gestionar las iteraciones de la simulación, crear los dispositivos, eliminar todo lo necesario cuando finaliza una simulación etc.



**Figura D.12 – Clase principal *Simulator***

## D.2. Estructura de las carpetas

---

La aplicación consta de diversas carpetas donde se guardan el código fuente, las imágenes de los mapas, los ficheros de los mapas, los ficheros de las trazas, etc.

Las carpetas que componen el sistema son las siguientes:

- BIN: en esta carpeta se guardan las clases compiladas (.class).
- CACHE: en esta carpeta se almacenan las imágenes (*tiles*) que se descargan junto con los mapas. En la siguiente sección de este anexo se explicará cómo se organizan las carpetas que guardan estas imágenes.
- CONFIG: esta carpeta contiene el fichero de configuración “config.xml”.
- GRAPH: en esta carpeta se guardan los ficheros que almacenan la información de la topología (carreteras, calles, etc.) de un mapa. Dentro de la carpeta “GRAPH” se encuentran al mismo nivel las carpetas cuyos nombres son los asignados por el usuario cuando se descargó el mapa, y el fichero de texto “downloadedMaps.txt” que contiene información sobre los mapas descargados. Dentro de cada carpeta se guarda el fichero de los elementos fijos de soporte para las comunicaciones, el fichero del

área de monitorización y el fichero con la información geográfica del mapa (.osm).

- **LIBS:** en esta carpeta se guardan las librerías que se usan. En esta aplicación se utiliza la librería “args4j.jar” encargada del paso de parámetros a través de la línea de comandos o modo *batch*.
- **LOG:** en esta carpeta se guarda el fichero de registro de los eventos que suceden en las simulaciones.
- **PICS:** esta carpeta contiene todas las imágenes que el simulador utiliza. Por ejemplo, podemos encontrar las imágenes que se usan para representar los vehículos, las antenas, etc.
- **SCENARIOS:** esta carpeta contiene los ficheros donde se guardan los filtros que se aplican a las trazas y a los mapas.
- **SCRIPTS:** en esta carpeta podemos encontrar otras 2 carpetas: “UNIX” y “WINDOWS”. En cada una de las carpetas hay 3 scripts (compilación, ejecución en modo gráfico y en modo *batch*) que es necesario usar para ejecutar la aplicación.
- **SRC:** en esta carpeta se guarda el código fuente. Dentro de esta carpeta se puede encontrar al mismo nivel el fichero “Simulator.java”, que es la clase principal de la aplicación, y las carpetas que componen los diferentes “packages” que se han explicado anteriormente.
- **TRACKS:** esta carpeta contiene los ficheros de las trazas GPS que se quieren importar.
- **VERSIONS:** en esta carpeta podemos encontrar un fichero de texto llamado “Version.txt” en el que se indica el progreso de las funcionalidades y versiones de la aplicación.

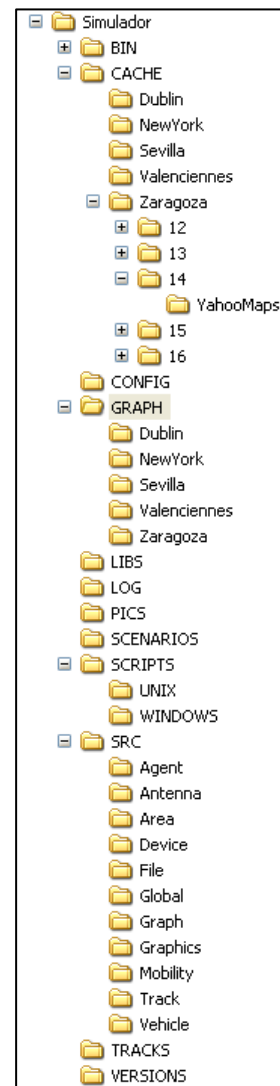


Figura D.13 – Árbol de directorios

### D.3. Estructura del almacenamiento de las tiles

Debido a que la estructura de los directorios que almacenan las imágenes de los mapas no es trivial se ha decidido dedicar esta sección a explicarla.

La carpeta que contiene toda la estructura de las imágenes es la carpeta “CACHE”. Dentro de esta carpeta existen tantas carpetas como mapas se han descargado. Cada una de ellas lleva el nombre que el usuario eligió cuando descargó el mapa. Dentro de la carpeta de cada mapa habrá tantas carpetas como niveles de zoom tenga ese mapa. La cantidad de niveles de zoom que tiene un mapa depende del nivel máximo de zoom que el usuario eligió cuando descargó el mapa. Hay que tener en cuenta que el nivel mínimo que se descarga es el 12 ya que se considera suficientemente lejano para visualizar el mapa. Por tanto si se eligió un nivel máximo de zoom de 15, habrá 4 carpetas (12, 13, 14 y 15).

Dentro de cada una de estas carpetas del nivel de zoom habrá otra carpeta que llevará el nombre del tipo de mapa, por ejemplo “YahooMaps”. Y ya dentro de esta última carpeta se guardarán todas las imágenes correspondientes.

Si durante la simulación se necesitan imágenes de un determinado nivel de zoom que no fue descargado, la aplicación creará las carpetas necesarias en el lugar que le corresponda según lo explicado anteriormente y guardará en esas carpetas las imágenes que se descarguen. A continuación, se muestra una figura para comprender mejor lo explicado.

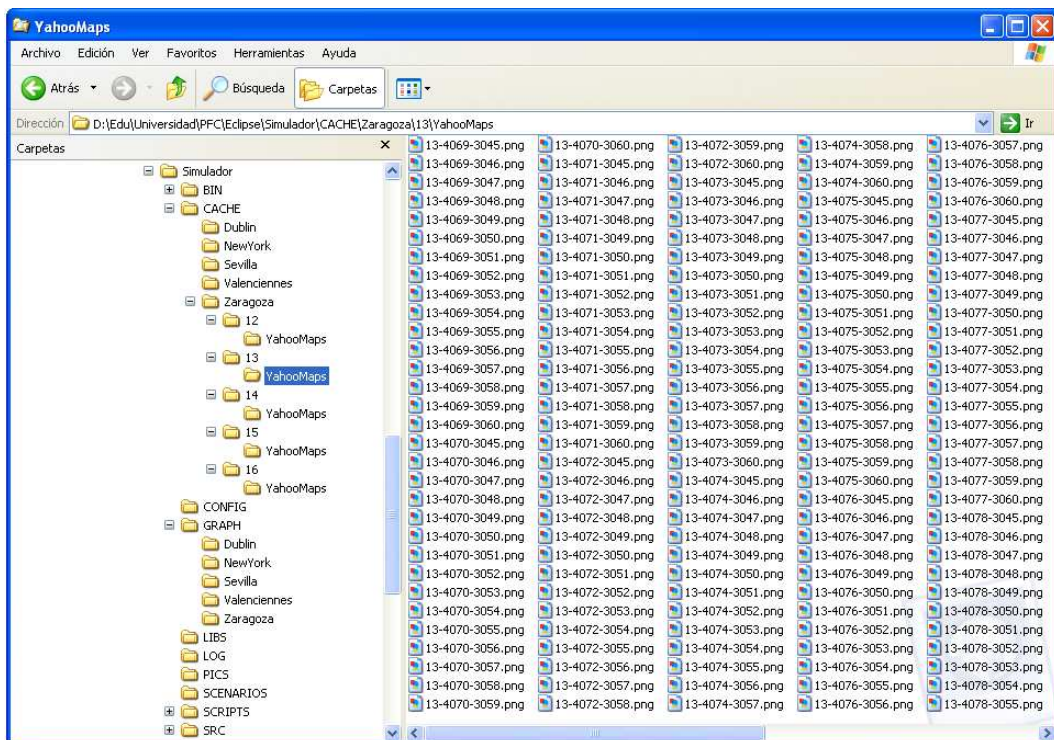


Figura D.14 – Árbol de directorios para las imágenes de los mapas

## D.4. Diagrama de clases

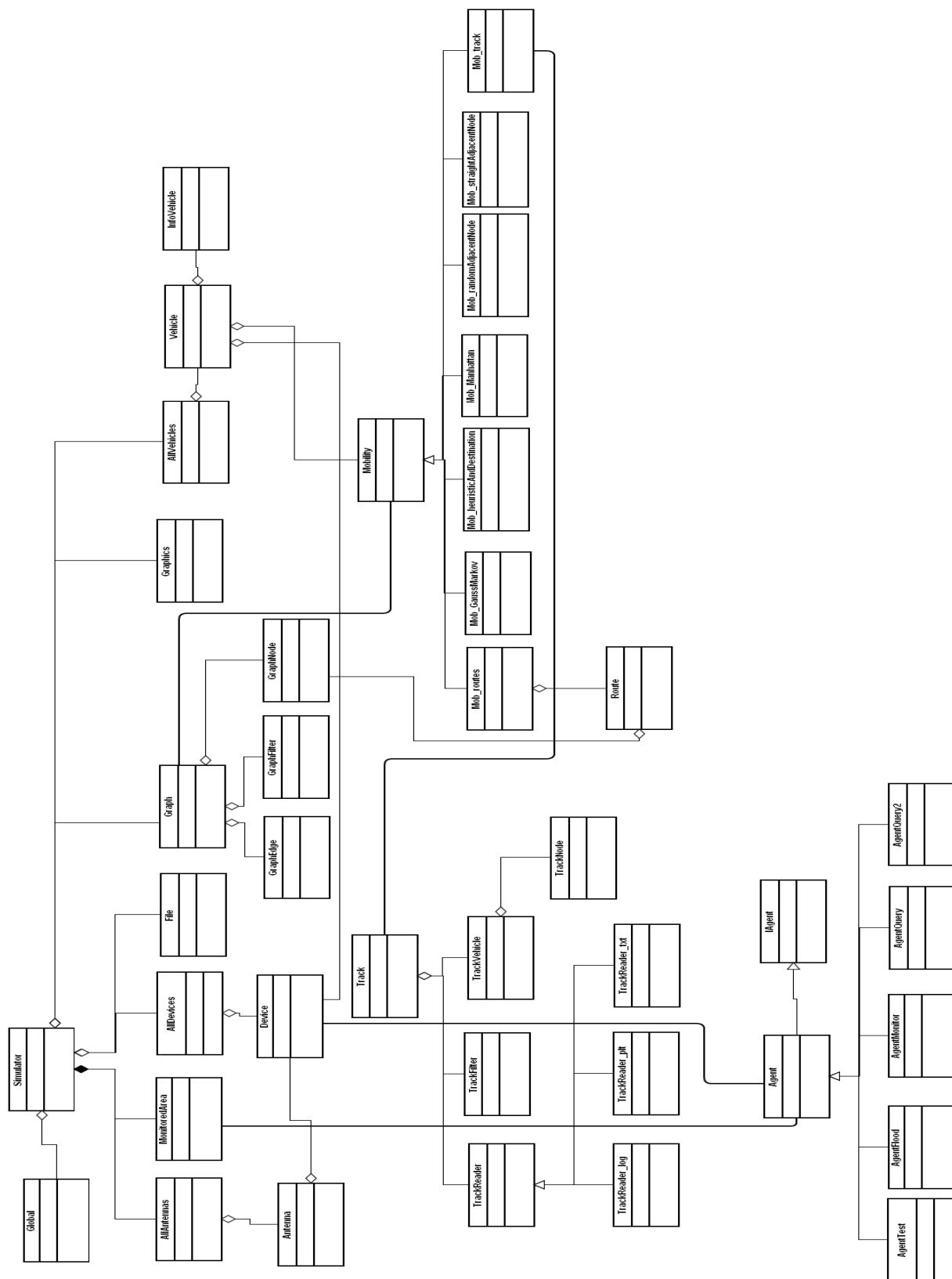


Figura D.15 – Diagrama de clases esquemático





## ANEXO E – MATERIAL RELACIONADO

---

En este anexo se va a explicar de una forma algo más extensa algunos de los apartados sobre materia relacionado con el proyecto que fueron brevemente comentados en la memoria.

### E.1. Prototipo de simulador del SID

---

Además de las limitaciones del prototipo previo del SID, comentadas en la memoria poseía otras.

Otras limitaciones que tenía este prototipo eran que sólo se podían realizar las simulaciones en el mismo mapa (mapa de la ciudad de Valenciennes en Francia) y que para realizar cambios en la configuración de la simulación había que modificar el código fuente. Por ejemplo, para cambiar el número de vehículos de la simulación o la estrategia de movilidad de los vehículos había que realizar el cambio pertinente en el código fuente. Además, el simulador informaba de los parámetros de configuración de una forma reducida, ya que sólo decía lo siguiente:

- **Status (estado de la simulación):** *running/paused/fast*
- **Zoom (nivel de zoom):** indica con un número el nivel de zoom
- **Strategy (estrategia):** estrategia de salto de los agentes móviles
- **Iteration (iteración):** el primer valor indica la iteración actual de la simulación, el segundo valor indica el número de iteraciones durante las cuales el agente móvil debe realizar mediciones sobre el área de monitorización, y el último valor indica el número máximo de iteraciones de la simulación.

Al hacer clic sobre el botón “Params” se abría un menú de configuración que permitía modificar la estrategia de salto de los agentes móviles y la velocidad de los vehículos. Se muestra dicho menú en la Figura E.1:

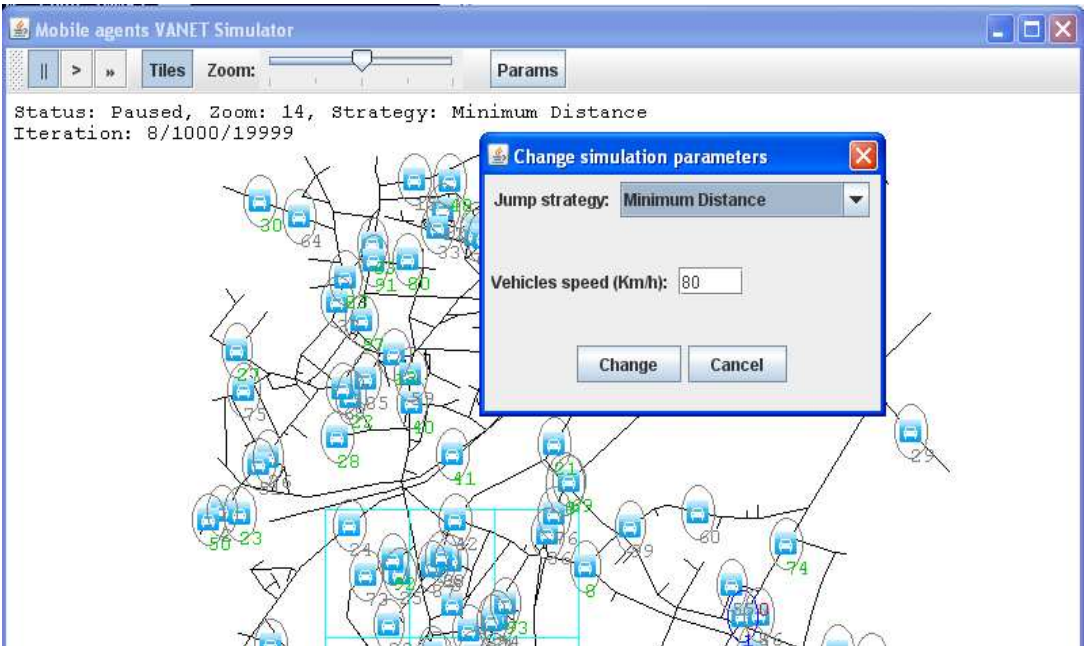


Figura E.1 – Imagen del prototipo previo con ventana de modificación de parámetros

E.2. Agentes móviles

Para la monitorización de áreas geográficas y otras tareas se usan agentes móviles. Los agentes móviles son componentes software que tienen la capacidad de moverse de un nodo a otro de una red de forma autónoma y continuar su ejecución en su nuevo emplazamiento [13].

Por ejemplo, cuando se quiera monitorizar un área determinada se “soltará” uno o varios agentes móviles desde un lugar determinado y este o estos agentes móviles irán “saltando” de vehículo en vehículo hasta alcanzar el área a monitorizar. Una vez realizadas todas las mediciones, volverán con la información obtenida “saltando” de vehículo en vehículo hasta alcanzar el centro de procesamiento de la información donde su ejecución finalizará a no ser que se quieran tomar más mediciones [1] (ver Figura E.2).

La toma de mediciones en el área geográfica tiene cierta dificultad, ya que el agente tiene que hacer todo lo posible (moverse de vehículo en vehículo) para mantenerse en esa área.

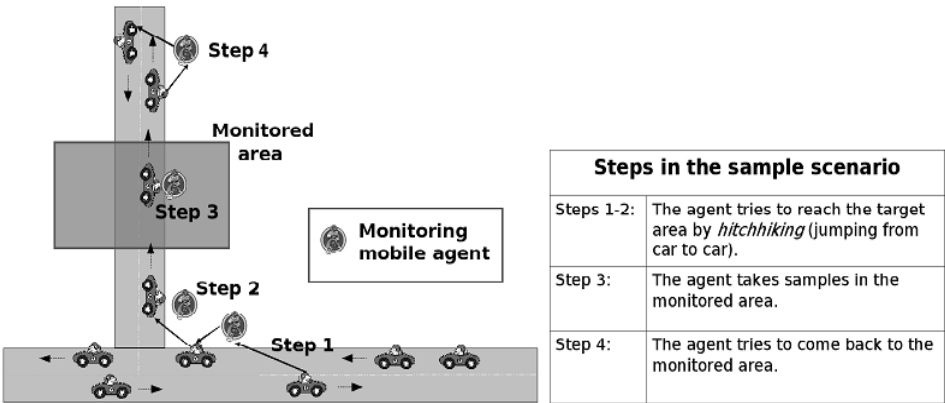


Figura E.2 – Imagen explicativa de un agente móvil saltando “Extraído de [1]”

A continuación se muestra un ejemplo de utilización de agentes móviles y monitorización de áreas geográficas.

Supongamos que tenemos el centro de procesamiento de información en el CPS (Centro Politécnico Superior), que existen vehículos dotados con dispositivos que les permiten formar una VANET, y que los vehículos están equipados con sensores capaces de medir el nivel de polen en el aire. Imaginemos que tenemos el objetivo de medir la concentración de polen en distintas áreas de Zaragoza en las que no existe un centro de medición fijo.

Soltamos el agente móvil encargado de realizar las mediciones desde el CPS y cuando pase un vehículo de la VANET, el agente móvil se moverá hasta el vehículo. El agente móvil irá saltando entre los vehículos de la VANET hasta alcanzar la zona que se quiere monitorizar. Realizará las mediciones oportunas y emprenderá el viaje de vuelta hasta el CPS saltando de vehículo en vehículo. Una vez que llegue al CPS se procesará la información que haya obtenido y habremos realizado las mediciones sin movernos del CPS y pudiendo haber aprovechado el tiempo que ha estado el agente móvil monitorizando el área desempeñando otras tareas. Otra ventaja muy importante es el ahorro económico debido a que no es necesario realizar un despliegue de sensores en las zonas que se quieren monitorizar.

### **E.3. Monitorización de áreas geográficas**

---

Hoy en día existe un gran interés en las redes VANET (Vehicular Ad hoc Network) por los servicios que pueden ofrecer.

Una red VANET es una red ad hoc formada por vehículos en la que se intercambian información con un determinado objetivo. Los vehículos usan comunicaciones de corto alcance, como IEEE 802.11 para el intercambio dinámico de información con otros vehículos que se encuentren cercanos (100 – 200 metros).

Las redes VANET tienen diversos usos como puede ser: aviso de accidentes en carretera, información acerca de plazas de aparcamiento [2], información sobre la gasolinera más cercana, monitorización de áreas geográficas, etc.

Sobre los diversos usos de una VANET, en esta sección nos vamos a centrar en la monitorización de áreas geográficas [1]. Los vehículos pueden medir ciertos parámetros del ambiente del lugar en el que se encuentran, gracias a diversos sensores que llevan equipados, por ejemplo la temperatura. Por lo tanto, si se quiere monitorizar la temperatura de una determinada zona geográfica, no sería necesario acercarse hasta esa zona para medir la temperatura sino que con una VANET se podría hacer sin la necesidad de moverse y sin tener que invertir dinero en el despliegue manual de los sensores.

Usar una VANET para monitorizar un área geográfica consta de los siguientes pasos:

- Determinar el área de monitorización y el tiempo que se quiere monitorizar.
- Asignar los vehículos a la tarea de monitorización: los vehículos tienen que estar equipados con los sensores necesarios para realizar la monitorización (sensores de CO<sub>2</sub>, sensores para medir el porcentaje de polen en el aire, sensores de humedad...).

- Realizar las mediciones en el área de monitorización mediante los sensores que llevan equipados los vehículos.
- Recolectar las mediciones realizadas. Una vez se han realizado las mediciones sobre el área de monitorización es necesario enviar toda esta información de vuelta al lugar donde se tiene que procesar.
- Procesar la información obtenida. En el lugar de procesamiento se analiza toda la información obtenida.

El proceso de monitorización de áreas geográficas se realiza con agentes móviles [12]. Un agente móvil es un componente software que tiene la capacidad de moverse de un nodo a otro de una red de forma autónoma y continuar su ejecución en su nuevo emplazamiento [13].

El agente encargado de monitorizar el área geográfica salta de vehículo en vehículo hasta alcanzar el lugar geográfico que desea analizar. Permanece en el lugar en el que tiene que tomar las mediciones el tiempo necesario y una vez obtenida toda la información retoma su camino hacia el lugar de procesamiento de información saltando de vehículo en vehículo [1].

## ANEXO F – ESTRATEGIAS DE MOVILIDAD

---

### F.1. Rutas basadas en Dijkstra

---

En la realidad, todo vehículo parte de un punto origen y quiere dirigirse a un punto destino. Una vez que llega a ese punto destino, se convierte en su nuevo punto origen y vuelve a generar otro destino al que llegar.

Para simular este comportamiento se pensó en calcular la ruta para cada vehículo que le llevara desde el punto origen hasta el punto destino pasando por el menor número de puntos posibles. Esta ruta o sucesión de nodos que se necesita visitar para alcanzar el nodo destino se calcula a partir del algoritmo del camino más corto de Dijkstra [71]. Se utilizó la información de la sección 4.3.2 “Cálculo de rutas” y el Anexo E.3 “Cálculo de rutas con el algoritmo de Dijkstra” de [24] para afrontar la implementación de esta estrategia de movilidad.

Lo primero que hay que hacer es generar el nodo destino al que el vehículo debe dirigirse. Desde la posición inicial que ocupa el vehículo se genera de forma aleatoria un nodo destino que tiene que cumplir una restricción de distancia respecto a la posición inicial del vehículo. La restricción de distancia que debe cumplir el nodo destino es un intervalo de separación, es decir, una distancia mínima y una distancia máxima. Esta restricción existe para asegurarse de generar rutas que no sean excesivamente cortas.

Por ejemplo, si el intervalo de distancia es [500, 1800], el nodo destino debe encontrarse a una distancia del nodo origen mayor de 500 metros y menor de 1800 metros. En la Figura F.1 se muestra un esquema explicativo.

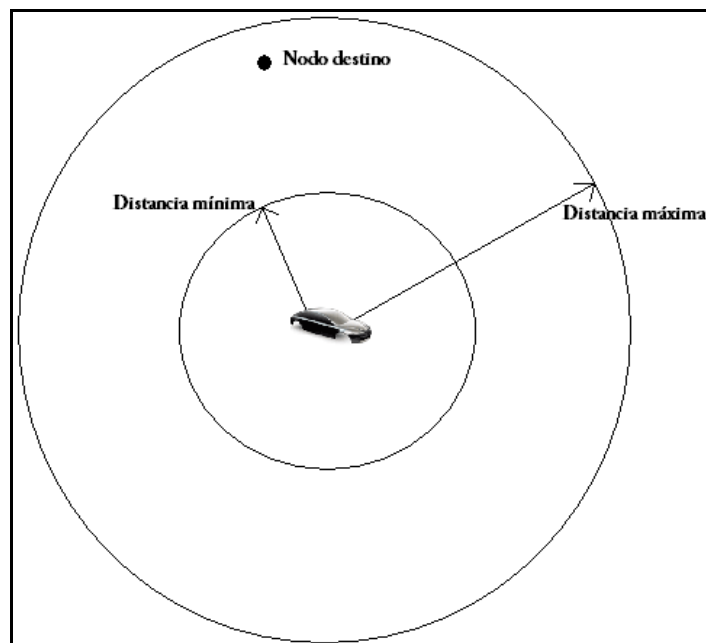


Figura F.1 – Generación del nodo destino para una ruta

Una vez que se ha generado el nodo destino hay que crear la ruta más corta para alcanzar dicho nodo con el algoritmo del camino más corto de Dijkstra. Este algoritmo necesita 2 tablas con un registro para cada nodo del grafo.

- **Tabla de rutas:** para cada nodo  $x$  se indica qué nodo se necesita para alcanzar el nodo  $x$  recorriendo la menor distancia.
- **Tabla de distancias:** para cada nodo  $x$  se indica la distancia mínima que existe entre el nodo origen y el nodo  $x$ .

Inicialmente, la tabla de rutas (Ruta), debe contener en cada registro valores indefinidos o nulos. Y la tabla de distancias (Distancia) debe tener en cada registro el máximo valor representable, excepto el nodo inicial que contendrá un 0. Por otra parte, tenemos un conjunto de nodos ( $Q$ ) en el que se almacenan todos los nodos del grafo. A continuación, se extrae de  $Q$  el nodo inicial de la ruta y se actualiza la tabla de distancias (Distancia) para los nodos vecinos de este nodo inicial de forma que la nueva distancia de cada vecino será igual a la distancia del nodo inicial a ese vecino.

$$\text{Distancia}[\text{vecinoN}] = \text{separación}(\text{nodo inicial}, \text{vecinoN})$$

Después, se actualiza la tabla de rutas (Ruta) de forma que para alcanzar a cada vecino es necesario pasar por el nodo inicial. Es decir, se actualiza el registro de cada vecino con el valor del nodo inicial.

$$\text{Ruta}[\text{vecinoN}] = \text{nodo inicial}$$

A continuación, se extrae el nodo ( $n$ ) de  $Q$  que tenga el menor valor ( $d$ ) en la tabla de distancias. Se analiza este nodo ( $n$ ) de la siguiente forma. Se actualiza la tabla de distancias de manera que para cada vecino de  $n$  se compara si el valor que tiene en la tabla de distancias es mayor que la suma de  $d$  más la distancia entre  $n$  y el vecino. Si lo anterior sucede, entonces se actualiza el registro del vecino de la tabla de distancias con dicha suma y además se modifica el registro de la tabla de rutas del vecino con el del nodo  $n$ .

$$\begin{aligned} d &= \text{Distancia}[n] \\ \text{Si } (\text{Distancia}[\text{vecinoN}] > d + \text{separación}(n, \text{vecinoN})) &\text{ entonces} \\ \text{Distancia}[\text{vecinoN}] &= d + \text{separación}(n, \text{vecinoN}) \\ \text{Ruta}[\text{vecinoN}] &= n \end{aligned}$$

Este proceso se hace para todos los nodos de  $Q$  hasta que se llegue al nodo destino o no queden más nodos en  $Q$ . La ruta que el vehículo debe seguir se obtendrá a partir de la tabla de rutas (Ruta). Se buscará el valor del registro del nodo destino en la tabla de rutas (nodoN) y ese será el último nodo que hay que visitar antes de alcanzar el nodo destino. Después, se obtendrá el valor del registro del nodoN de la tabla de rutas (nodoN-1) y se obtendrá el antepenúltimo nodo que hay que visitar antes de alcanzar el nodo destino. Este proceso se realiza hasta que el valor del registro del nodo nodoN-x sea igual al nodo inicial. Y de esta forma se habrá obtenido la ruta más corta en sentido inverso.

$$\begin{aligned} \text{nodoN} &= \text{Ruta}[\text{nodo destino}] \\ \text{nodoN-1} &= \text{Ruta}[\text{nodoN}] \\ \text{nodoN-2} &= \text{Ruta}[\text{nodoN-1}] \\ &\dots \end{aligned}$$

$$\text{nodoN-x} = \text{Ruta}[\text{nodoN-x-1}] \quad (\text{nodoN-x} = \text{nodo inicio})$$
$$\text{ruta vehículo} = (\text{nodo inicio}, \text{nodoN-x}, \text{nodoN-x-1}, \dots, \text{nodoN-1}, \text{nodoN}, \text{nodo destino})$$

Si se desea entender mejor el funcionamiento de este algoritmo existe un ejemplo realizado paso a paso en el Anexo E.3 de [24].

Una vez implementado el algoritmo, resultó no ser eficiente debido a que puede haber más de un vehículo calculando la ruta óptima para alcanzar su nodo destino y además puede que el grafo sea muy grande en cuanto a número de nodos. Para solucionar este problema se implementaron las mejoras propuestas en la sección “4.3.2 Cálculo de rutas” de [24]. Más concretamente, se aplicaron las soluciones “heurística, árbol y priority queue”. De esta forma, se optimizó el cálculo de rutas y se hizo factible usar este tipo de estrategia de movilidad en las simulaciones. La solución “heurística” se basa en una heurística que utiliza la distancia en línea recta desde cada posible nodo a formar parte de la ruta hasta el nodo destino. Con esta distancia calculada se ordenan los nodos para ser analizados. La solución “priority queue” se basa en tener los nodos a analizar ordenados en una cola de prioridad de forma que el acceso a ellos es más rápido. Y la solución “árbol” se basa en analizar solamente aquellos nodos que sean vecinos de uno visitado y de esta forma no es necesario cargar todo el grafo en memoria.

Una vez que el vehículo alcanza el nodo destino, éste se convierte en su nodo inicial para una nueva ruta. En la Figura F.2 se muestra una imagen del simulador indicando una ruta calculada. El nodo destino aparece en color rojo y la ruta aparece en verde y amarillo.

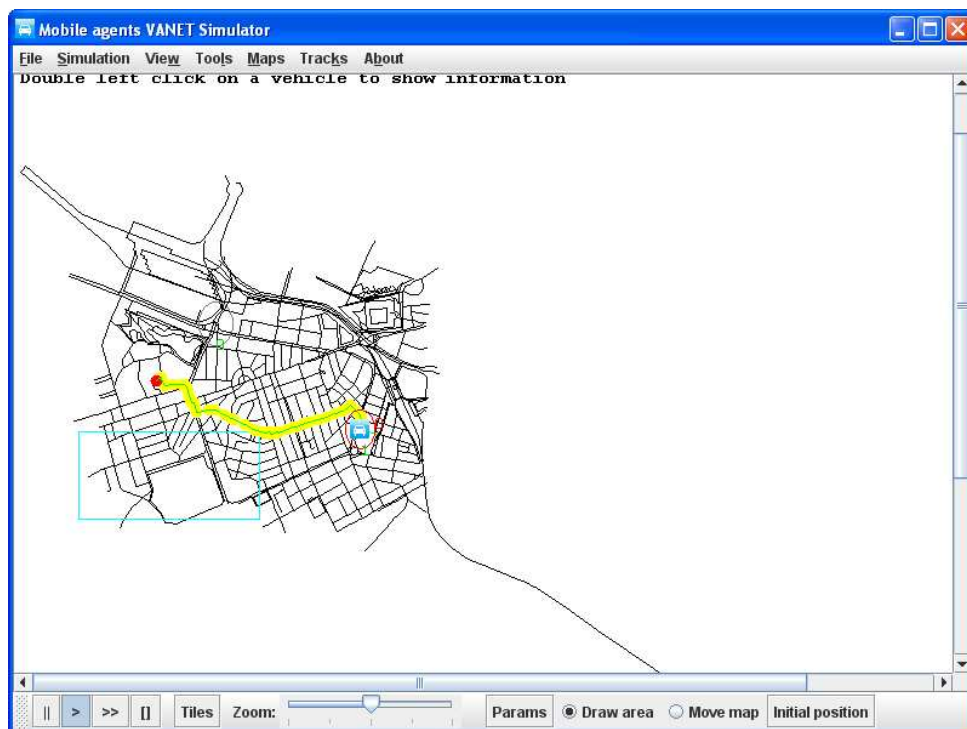


Figura F.2 – Imagen del simulador del PFC indicando una ruta

## F.2. Rutas dinámicas basadas en una heurística

Debido al coste computacional producido por el cálculo de la ruta más corta para cada vehículo al comienzo de ésta, se decidió implementar otro tipo de estrategia de movilidad que fuera más liviano computacionalmente hablando y que dotara a los vehículos de un comportamiento similar.

De esta forma surgió una política de movilidad basada en una heurística que va decidiendo cada vez qué nodo visitar para alcanzar un nodo destino.

Lo primero que se hace es generar un nodo destino de la misma forma que lo obtiene la estrategia de movilidad explicada anteriormente. Una vez que se tiene el nodo destino, se analiza cada nodo vecino del nodo en el que se encuentra el vehículo. El análisis que se realiza mediante una heurística es el siguiente: el nodo vecino que tenga menor distancia en línea recta hasta el nodo destino será el siguiente nodo (*next node*) a visitar. Por lo tanto, la heurística se basa en la distancia de separación en línea recta entre un nodo vecino y el nodo destino. Una vez que el vehículo llega hasta el nodo *next node*, se realiza el análisis de los nodos vecinos del actual nodo en el que se encuentra el vehículo (*next node*) de la forma que se acaba de explicar. La única restricción que existe respecto a la elección del siguiente nodo a visitar (*next node*) es que no puede ser el nodo del que procedemos a no ser que no exista otra opción. En la Figura F.3 se muestra un esquema de lo explicado. Las líneas azules de la figura indican la separación con el nodo destino, no son uniones.

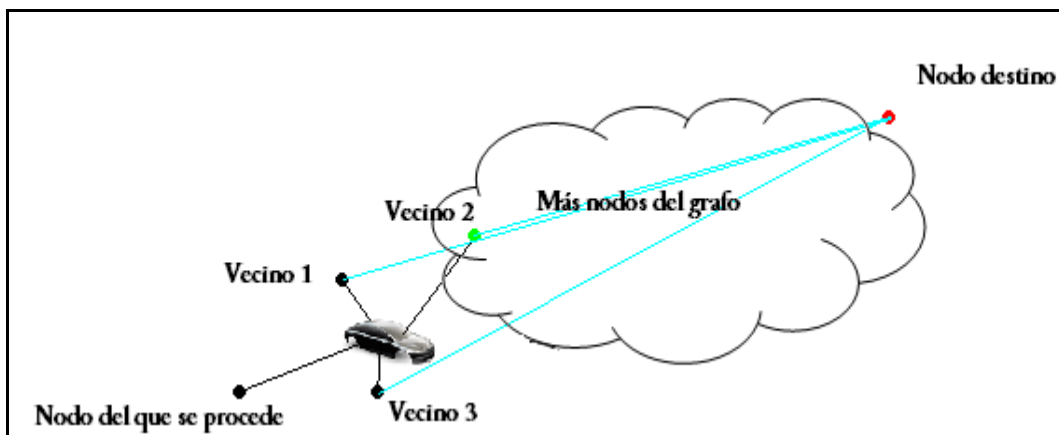


Figura F.3 – Esquema de la heurística

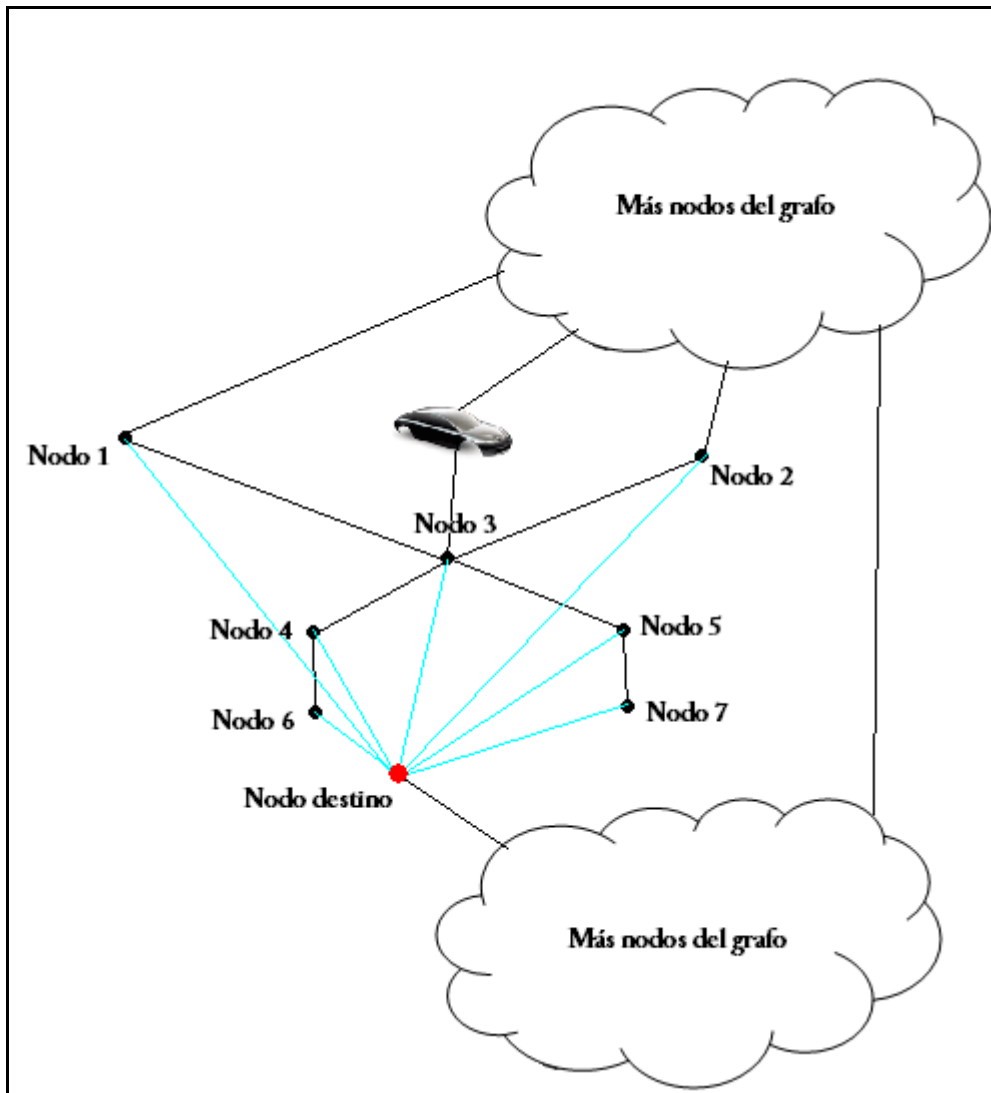
En la Figura F.3, de los tres posibles siguientes nodos (Vecino 1, Vecino 2 y Vecino 3), el siguiente nodo a visitar será el Vecino 2 ya que la distancia de separación en línea recta desde el Vecino 2 hasta el Nodo destino es menor que las distancias del resto. El nodo anterior (Nodo del que se procede) no es candidato a ser el nodo siguiente porque existen otras alternativas.

Cuando se alcanza el nodo destino, se genera otro nodo destino y se comienza el proceso explicado en el párrafo anterior.

Una vez implementada esta estrategia de movilidad se realizaron pruebas para comprobar su funcionamiento y se encontró el siguiente problema. Según la disposición de los nodos, el vehículo puede entrar en bucles, ya que entre varios nodos posibles puede elegir el que se encuentra más cerca del destino en línea



recta, pero éste nodo puede no llevar hasta el nodo destino. Una vez que se tome otro camino se vuelve a meter en otro bucle. En la Figura F.4 se muestra un esquema con el que entender mejor este problema. En la figura las líneas azules indican la separación, no son uniones entre nodos.



**Figura F.4— Esquema del problema**

El vehículo se moverá al Nodo 3. Desde el Nodo 3 se puede mover al nodo 1, Nodo 4 o al Nodo 5. Debido a que la distancia del Nodo 4 al nodo destino es menor que la del Nodo 5 al destino y la del Nodo 1 al nodo destino, el vehículo se moverá al Nodo 4. Del Nodo 4 se moverá al Nodo 6 ya que es el único nodo posible excluyendo el nodo anterior (Nodo 3). Del Nodo 6 se moverá al Nodo 4 porque es la única opción. Del Nodo 4 se moverá al Nodo 3 porque es la única opción excluyendo el nodo del que procede (Nodo 6). Del Nodo 3, se puede mover al Nodo 1 o al Nodo 5 ya que el Nodo 4 se excluye debido a que es el nodo del que procede. Se moverá al Nodo 5. Del Nodo 5 se moverá al 7. Del Nodo 7 volverá al 5 ya que es la única opción. Del 5 volverá al 3. Del Nodo 3 se moverá al 4 ya que está más cerca que el Nodo 1. Y esto se

repetirá infinitamente. De forma que el vehículo se quedará atrapado en un bucle sin salida. La solución a este problema consta de varios elementos.

### **Contador de nodos vistos**

Cada vez que el vehículo se mueve a un nodo se incrementa un contador que cuenta el número de nodos visitados. Cuando el número de nodos visitados llega a un máximo se genera un nuevo nodo destino al que dirigirse. Esto puede provocar que sin entrar en bucles no se llegue nunca a alcanzar el nodo destino, pero si el límite de número de nodos visitados no es muy pequeño este inconveniente no es relevante porque no es necesario que el vehículo llegue hasta el nodo destino.

### **Repetición de nodos visitados**

Si se vuelve a visitar un nodo que previamente se había visitado, el vehículo tenderá a alejarse del nodo destino para evitar el posible bucle y además se penalizará esta visita incrementando el contador de nodos visitados en más de una unidad para provocar que el vehículo genere un nuevo destino antes.

## **F.3. Estrategia basada en Manhattan**

---

La estrategia de movilidad Manhattan [10] toma su nombre de la isla de Manhattan debido a que esta estrategia está especializada en ciudades con la topografía de Manhattan. Las calles de la isla de Manhattan tienen una topografía muy característica y es que tienen forma de malla o cuadrícula, es decir, las calles son paralelas o perpendiculares entre sí.

Por lo tanto, esta estrategia de movilidad no se podría usar en cualquier ciudad del mundo ya que las ciudades no suelen estar construidas en forma de cuadrícula, a no ser que la política de movilidad se adaptara a cualquier tipo de topología.

La política de movilidad Manhattan [10] consiste en que los vehículos se pueden mover a 3 posibles nodos: nodo de la izquierda, nodo de la derecha o nodo de enfrente. Esto es posible a la topología característica de las calles de Manhattan que anteriormente hemos comentado. La probabilidad de que el vehículo siga recto por la misma calle (se mueva al nodo de enfrente) es del 50 %, mientras que la probabilidad de que el vehículo gire a la derecha es del 25 % igual que la probabilidad de que gire a la izquierda, otro 25 %. Véase la Figura F.5 para entender mejor el funcionamiento de esta estrategia de movilidad.

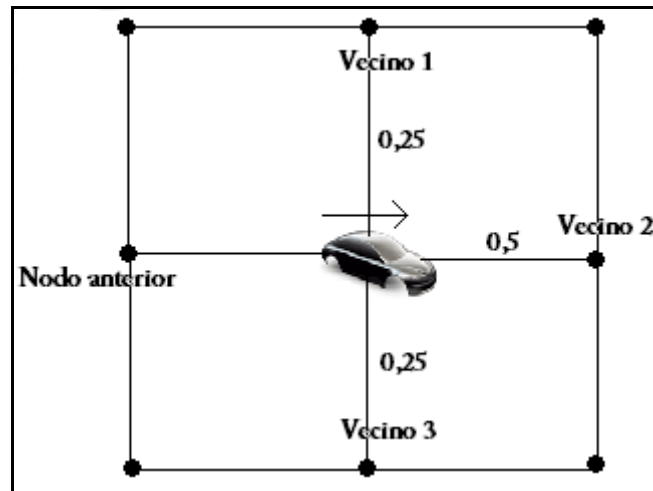


Figura F.5 – Estrategia de movilidad Manhattan

Una vez explicada la estrategia de movilidad Manhattan, se procede a explicar la variación de ésta para permitir usarla en cualquier tipo de mapa, aunque no tenga forma de cuadrícula.

La estrategia de movilidad basada en Manhattan, pero adaptada a cualquier mapa prima el movimiento del vehículo en línea recta al igual que lo hace Manhattan. Por lo tanto, le da una mayor probabilidad de ser el siguiente nodo a visitar al nodo que se encuentre más en línea recta respecto al movimiento del vehículo. Esta política de movilidad tiene varios casos especiales:

- **Un nodo vecino.** El nodo vecino será el último nodo visitado por el vehículo, y como no existe otra opción, será el siguiente nodo a visitar.
- **Dos nodos vecinos.** El vehículo elegirá el nodo que no sea el último nodo visitado.
- **Tres nodos vecinos.** Si se excluye el último nodo visitado, el vehículo tendrá 2 opciones posibles. El nodo que se esté más en línea recta respecto al movimiento del vehículo tendrá una probabilidad de ser elegido del 75% y el nodo restante tendrá una probabilidad del 25%.
- **Más de tres nodos vecinos.** Si se excluye el último nodo visitado, habrá como mínimo 3 posibles nodos para visitar. Al nodo vecino que esté más en línea recta respecto al movimiento del vehículo se le asignará una probabilidad del 50 % de ser elegido. El otro 50 % será repartido en partes iguales con los otros nodos vecinos restantes. Por ejemplo, con 7 posibles nodos a visitar (excluyendo el último nodo visitado), el nodo que esté más en línea recta respecto al movimiento del vehículo tendrá un 50 % y cada uno de los 6 restantes tendrá un 8,33 %. Véase la Figura F.6 para entender mejor esta política.

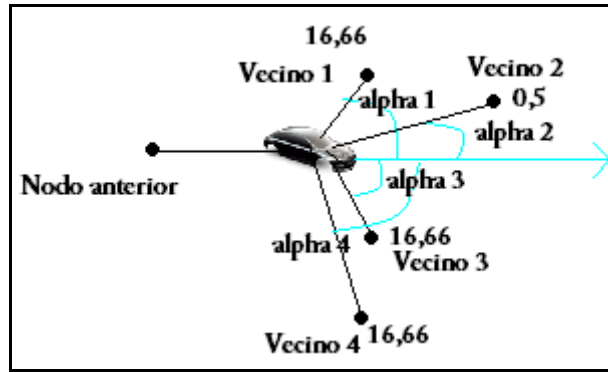


Figura F.6 – Estrategia de movilidad Manhattan adaptada

En la Figura F.6 existen 4 posibles nodos a visitar (Vecino 1, Vecino 2, Vecino 3 y Vecino 4). El nodo que tiene un ángulo más pequeño (alpha 2) respecto a la dirección del movimiento del vehículo es el nodo Vecino 2. Por tanto, Vecino 2 tendrá una probabilidad de ser elegido como siguiente nodo a visitar del 50 % y los nodos restantes (Vecino1, Vecino 3 y Vecino 4) tendrán un 16,66 %.

#### F.4. Estrategia basada en Gauss-Markov

Con la estrategia basada en Gauss-Markov [9] sucede algo similar que con la estrategia Manhattan, ya que la estrategia Gauss-Markov está pensada para que el vehículo se mueva en cualquier dirección sin las restricciones de una ciudad y sus calles.

Gauss-Markov fue diseñado para que tuviera diferentes niveles de aleatoriedad a través de un parámetro. Inicialmente, cada vehículo tiene una velocidad y una dirección. Cada cierta unidad de tiempo  $n$  el vehículo se mueve y se actualiza su velocidad y dirección. El valor de la velocidad y la dirección en la unidad de tiempo  $n$  se calcula a partir de la velocidad y la dirección en la unidad de tiempo  $n-1$  y una variable aleatoria usando las siguientes fórmulas:

$$\begin{aligned} s_n &= \alpha s_{n-1} + (1 - \alpha) \bar{s} + \sqrt{(1 - \alpha^2)} s_{x_{n-1}} \\ d_n &= \alpha d_{n-1} + (1 - \alpha) \bar{d} + \sqrt{(1 - \alpha^2)} d_{x_{n-1}} \end{aligned}$$

Figura F.7 – Fórmulas de velocidad y dirección de Gauss-Markov

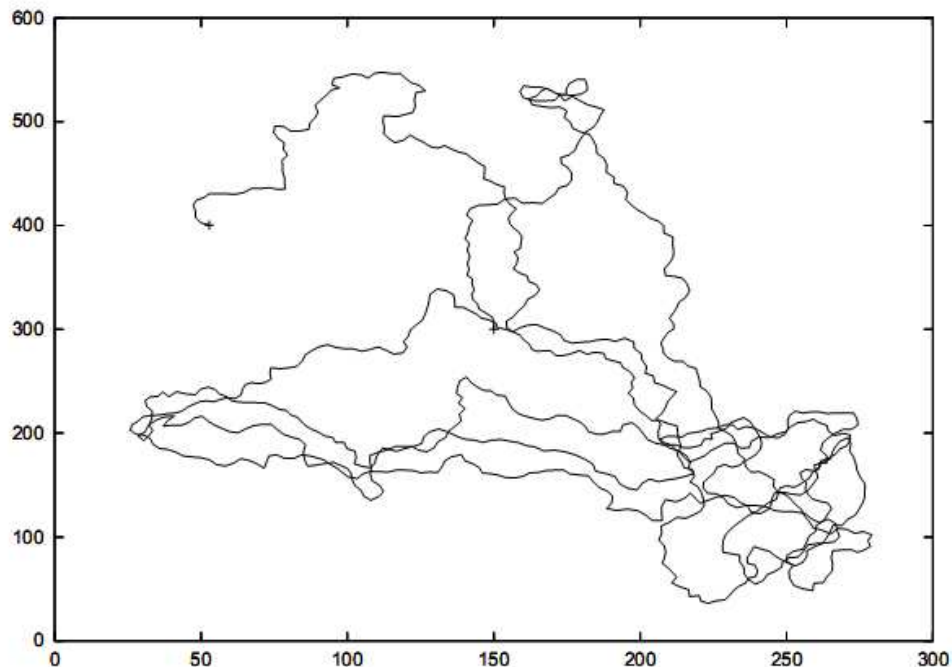
$s_n$  y  $d_n$  indican respectivamente la velocidad y la dirección del vehículo en el intervalo de tiempo  $n$  que se quiere calcular.  $\alpha$  ( $0 \leq \alpha \leq 1$ ) es el parámetro que configura el valor de aleatoriedad.  $\bar{s}$  y  $\bar{d}$  son constantes que representan respectivamente, el valor medio de la velocidad y la dirección cuando  $n \rightarrow \infty$ .  $s_{x_{n-1}}$  y  $d_{x_{n-1}}$  son variables aleatorias de una distribución Gaussiana. Con  $\alpha = 0$  se obtienen valores totalmente aleatorios y con  $\alpha = 1$  se obtienen valores lineales [7]. Valores intermedios de aleatoriedad se obtienen variando el valor  $\alpha$  entre 0 y 1.

En cada intervalo de tiempo la siguiente posición se calcula respecto de la posición actual, la velocidad y la dirección del vehículo. Para el intervalo de tiempo  $n$ , las coordenadas posicionales se obtienen de la siguiente forma:

$$\begin{aligned} x_n &= x_{n-1} + s_{n-1} \cos d_{n-1} \\ y_n &= y_{n-1} + s_{n-1} \sin d_{n-1} \end{aligned}$$

**Figura F.8 – Fórmulas de la posición de Gauss-Markov**

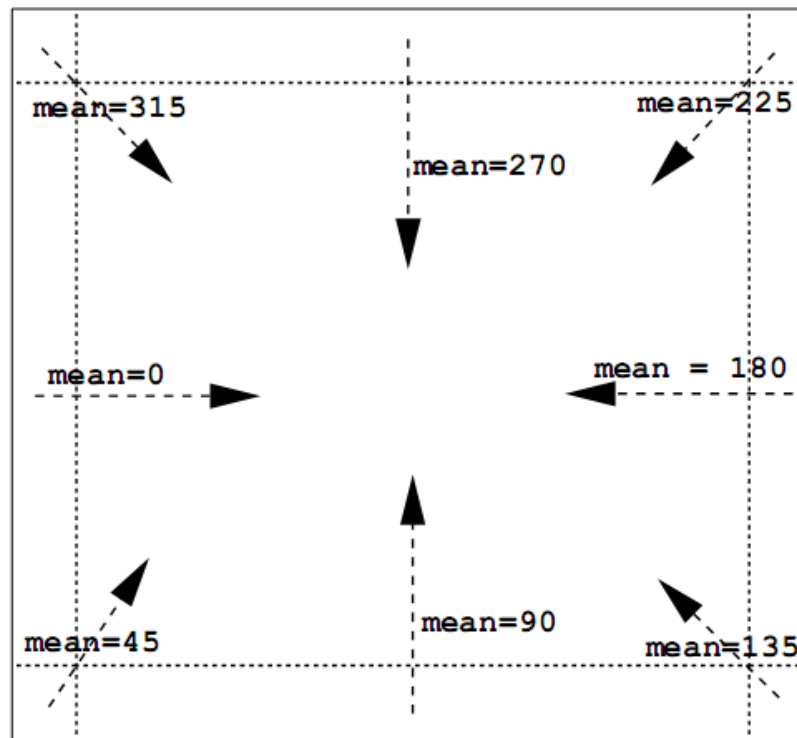
Donde  $(x_n, y_n)$  y  $(x_{n-1}, y_{n-1})$  son las coordenadas  $x$  e  $y$  del vehículo para el intervalo de tiempo  $n$  y  $n-1$ , respectivamente.  $s_{n-1}$  y  $d_{n-1}$  son la velocidad y la dirección para el intervalo  $n-1$ , respectivamente.



**Figura F.9 – Traza de movimiento libre con Gauss-Markov “Extraído de [7]”**

En la gráfica anterior (ver la Figura F.9) se muestra un ejemplo extraído de [7] del camino seguido por un vehículo con la estrategia de movilidad Gauss-Markov. El vehículo comienza su movimiento en el centro del área de simulación y su movimiento dura 1000 segundos.  $n = 1$  segundo,  $\alpha = 0.75$ ,  $s_{x_{n-1}}$  y  $d_{x_{n-1}}$  son elegidos con una distribución Gaussiana con una media igual a cero y una desviación estándar igual a uno. El valor de  $\bar{s}$  es 10 m/s; el valor de  $\bar{d}$  inicialmente es de 90 grados, pero cambia a lo largo del tiempo debido a su aproximación a los extremos del área de simulación. Para asegurar que el vehículo se mueve dentro del área de simulación, cada vez que se acerca a uno de los extremos,  $\bar{d}$  se modifica para que el vehículo se aleje de esa zona. Por ejemplo, cuando se acerca al extremo derecho se le asigna el valor 180 grados.

A continuación se muestra la Figura F.10 que explica las modificaciones de  $\bar{d}$ .



**Figura F.10 – Diagrama explicativo sobre modificación forzada de la dirección**  
“Extraído de [7]”

Una vez explicado el patrón de movilidad Gauss-Markov se va detallar cómo se ha adaptado esta estrategia para que se pueda utilizar en mapas cuyas calles restringen el movimiento de los vehículos.

Debido a que se decidió que en las simulaciones realizadas en el simulador de este PFC los vehículos siempre lleven una velocidad constante, la estrategia de Gauss-Markov sólo afectará a la dirección y no a la velocidad de los vehículos. Por lo tanto, la primera ecuación de la Figura F.7 no fue necesario implementarla.

En esta estrategia de movilidad Gauss-Markov adaptada sólo es necesario calcular las nuevas posiciones ( $x_n$ ,  $y_n$ ) (Figura F.8) a partir de una velocidad constante y una dirección que se calcula como se indica en la Figura F.7. Una vez que se han calculado las coordenadas  $x$  e  $y$ , se buscará qué nodo, de los vecinos del nodo en el que se encuentra el vehículo, está más cerca de la posición indicada por esas coordenadas. La separación entre el nodo a analizar y la posición determinada por las coordenadas se realiza midiendo la distancia en línea recta que separa ambas posiciones.

El nodo vecino más cercano a la posición determinada por dichas coordenadas será el siguiente nodo a visitar. En la Figura F.11 se muestra un esquema explicativo en el que las líneas en azul indican la distancia entre la posición generada hasta cada posible nodo vecino.

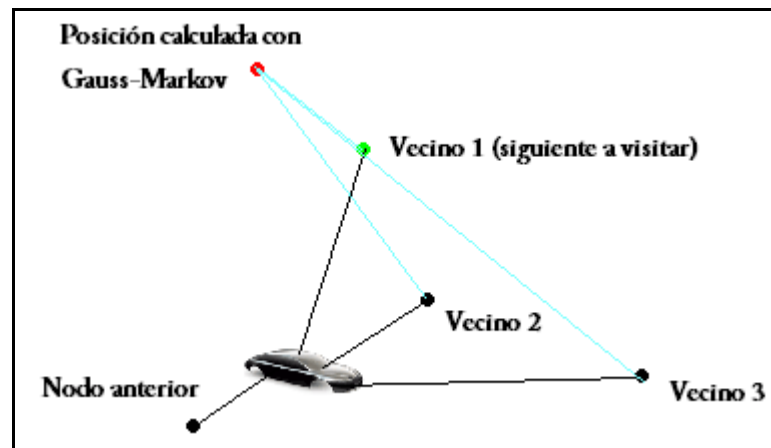


Figura F.11 – Búsqueda siguiente nodo con Gauss-Markov adaptado

