
ISAC-NAV

Por
Pablo Saro Buendía



UNIVERSIDAD COMPLUTENSE MADRID

Grado en Ingeniería de Computadores
FACULTAD DE INFORMÁTICA

Dirigido por
José Luis Vázquez Poletti y Juan Carlos Fabero
Jiménez

MADRID, 2021–2022

ISAC-Nav

Memoria que se presenta para el Trabajo de Fin de Grado

Pablo Saro Buendía

Dirigido por

José Luis Vázquez Poletti y Juan Carlos Fabero Jiménez

**Departamento de Arquitectura de Computadores y Automática
Facultad de Informática
Universidad Complutense de Madrid**

Madrid, 2022

Resumen

En los últimos años, muchas aplicaciones y sistemas requieren de una herramienta de geolocalización para su correcto funcionamiento. Actualmente, el GPS es el sistema de posicionamiento global de referencia, pero no el único, ya que cada vez hay más sistemas alternativos en el mercado.

Los sistemas de posicionamiento que más han evolucionado en las últimas décadas son los basados en el uso de las redes móviles, debido al desarrollo masivo que ha experimentado la telefonía y que su infraestructura es prácticamente global.

El proyecto “ISAC-Nav” pretende crear un sistema de posicionamiento funcional que haga uso exclusivamente de la infraestructura de redes móviles. Para ello, partiendo de las actuales técnicas de posicionamiento basadas en las redes móviles, se implementa un algoritmo cuyo propósito es localizar el centro del polígono generado por la unión de las antenas captadas en ese instante y la potencia recibida de cada una de ellas.

Tras detallar la arquitectura del proyecto y justificar el por qué se han implementado o descartado distintas tecnologías, se presentan dos casos de uso reales. A continuación, se expone un estudio donde, realizando una simulación en un entorno urbano, se mide la eficiencia del sistema creado. Luego, los resultados se comparan con los de dos sistemas de posicionamiento contrastados, en cuanto a fiabilidad y validez, para demostrar la viabilidad del proyecto “ISAC-Nav”.

Por último, se exponen las conclusiones por las cuales se consideran alcanzados los objetivos del proyecto “ISAC-Nav” entre los que destaca la creación de un sistema de posicionamiento funcional, el cual podrá desarrollarse a través de futuros proyectos.

Palabras Claves

- GSM
- Trilateración
- Redes móviles
- Radiofrecuencia
- Posicionamiento
- Telecomunicaciones

Abstract

In recent years, many applications and systems require a geolocation tool for their correct operation. Currently, GPS is the global positioning system of reference, but not the only one, as there are more and more alternative systems on the market.

The positioning systems that have evolved the most in recent decades are those based on the use of mobile networks, due to the massive development of telephony and the fact that their infrastructure is practically global.

The “ISAC-Nav” project aims to create a functional positioning system that makes exclusive use of the mobile network infrastructure. To this end, based on current positioning techniques based on mobile networks, an algorithm is implemented to locate the center of the polygon generated by the union of the antennas captured at that moment and the power received from each one of them.

After detailing the project architecture and justifying why different technologies have been implemented or discarded, two real use cases are presented. Next, a study is presented in which, by performing a simulation in an urban environment, the efficiency of the created system is measured. The results are then compared with those of two proven positioning systems, in terms of reliability and validity, to demonstrate the feasibility of the “ISAC-Nav” project.

Finally, conclusions are drawn as to why the objectives of the “ISAC-Nav” project are considered to have been achieved, among which the creation of a functional positioning system, which can be developed through future projects.

Key words

- GSM
- Trilateration
- Mobile networks
- Radiofrequency
- Positioning
- Telecommunications
- Open Source

Índice general

	Página
1. Introducción	1
1.1. Objetivos	2
1.2. Plan de trabajo	2
2. Introduction	5
2.1. Objectives	6
2.2. Work plan	6
3. Estado del arte	9
3.1. Estaciones base	10
3.2. Red móvil	10
3.2.1. Handover	11
3.2.2. RSSI	12
3.2.3. Generaciones Móviles	13
3.3. Sistemas de posicionamiento	14
3.3.1. Servicio de posicionamiento Global (GPS)	14
3.3.2. Posicionamiento redes móviles	15
3.3.3. Sistemas híbridos	20
3.4. GSM	21
3.4.1. Arquitectura	21
3.5. Trilateración	22
3.5.1. Centroide	23
3.5.2. Centro de masas	23
3.6. Cuestiones Legales	24
4. Arquitectura	25
4.1. Módulo de captación	25
4.1.1. Dongle RTL-SDR	26
4.1.2. Kalibrate-rtl	26
4.1.3. GNU-Radio	27
4.1.4. Grgsm	27
4.1.5. “Antenas.txt”	28
4.1.6. GQRX	28
4.2. Módulo de procesamiento	29
4.2.1. BBDD	29
4.2.2. “ISAC-NAV”	30
4.2.3. Makefile	31

4.3. Módulo de representación	32
4.3.1. OSM	32
4.3.2. Leaflet	33
4.3.3. Web	33
4.3.4. Firefox	33
4.4. Script	33
4.5. Sistema de ficheros	34
4.6. Tecnologías descartadas	35
4.6.1. Docker	35
4.6.2. BBDD	35
4.6.3. Node-Js	39
4.6.4. Xleaflet	40
5. Casos de uso	41
5.1. Ajuste de “ISAC-Nav”	41
5.2. Caso de uso en un entorno urbano	43
5.3. Caso de uso en el campus universitario	46
6. Estudio “ISAC-Nav”	48
6.1. Análisis RSSI	48
6.2. Análisis de la precisión	49
7. Conclusiones	50
7.1. Conclusiones	50
8. Conclusions	52
8.1. Conclusions	52
9. Trabajos futuros	54
Glosario	56
Bibliografía y enlaces de referencia	60
9.1. instalación	61
9.1.1. Gnuradio	61
9.1.2. Grgsm	61
9.1.3. Kalibrate-rtl	61
9.2. Configuración	61

Capítulo 1

Introducción

Con la aparición de tecnologías cada vez más avanzadas, surge la necesidad de hacer uso de sistemas de posicionamiento en tiempo real, cuyo objetivo es estimar la posición del usuario. Cada vez hay más dispositivos multifuncionales, destacando los teléfonos tipo *smartphone*, que pueden trabajar con aplicaciones que necesitan hacer uso de su ubicación para poder ofrecer su servicio.

A estos servicios se les conoce como servicios basados en localización (LBS, *Location Based Services*) y están presentes en una gran variedad de áreas: turismo, entretenimiento salud, etc. El elevado uso de estos sistemas se constató en el estudio realizado en 2012 por *Pew Internet & American Life Project*, donde el 74 % de los usuarios de *smartphones* utilizaban los servicios de localización del dispositivo [1].

El sistema de posicionamiento más frecuente si se quiere conocer la ubicación de un dispositivo es el GPS (*Global Positioning System*), por ello, normalmente los dispositivos tienen implementado un módulo GPS que les permite decodificar la señal de radio transmitida por los satélites que componen este sistema.

La señal GPS viaja por línea de visión, la cual es un tipo de propagación donde solo se pueden enviar datos cuando la estación de transmisión y recepción están a la vista, esto conlleva que este sistema tenga problemas de cobertura. Como consecuencia el sistema GPS suele fallar en zonas montañosas abiertas, túneles e incluso en espacios cerrados donde el grosor de las paredes atenúa la señal.

Pero el GPS no es el único sistema de posicionamiento y teniendo presentes sus carencias, el presente proyecto tiene como objetivo aportar una alternativa. La primera fase, comprende un estudio de las distintas técnicas de localización y posicionamiento, profundizando en los métodos más comunes existentes para crear sistemas de posicionamiento basados en la red de telefonía móvil.

La red de telefonía móvil tiene como objetivo principal la transmisión de datos. Para poder transmitir los datos, de forma inalámbrica, se sirve de señales con carácter electromagnético las cuales enlazan a distancia los dispositivos a la red. Estas comunicaciones inalámbricas utilizan estándares y protocolos concretos que permiten la comunicación de los dispositivos con las antenas transceptoras que dan acceso a la red.

Dentro del espectro electromagnético, la red de telefonía móvil hace uso de unas bandas reservadas en el estándar GSM (*Global System for Mobile*), las cuales pueden ser

identificadas mediante la longitud de onda, frecuencia e intensidad.

Actualmente existen numerosos sistemas de posicionamiento basados en la red de telefonía, como el AGPS que combina la infraestructura de la red móvil con el sistema GPS. Sin embargo, el presente proyecto trata de crear un sistema totalmente independiente de la señal GPS.

1.1. Objetivos

El objetivo principal del proyecto “ISAC-Nav” es crear un sistema de posicionamiento que permita geolocalizar al usuario a través de la red móvil, a partir de la señal de al menos dos antenas.

Para la correcta implementación del sistema de posicionamiento es necesaria la creación de un sistema de código libre con tres objetivos secundarios. En primer lugar debe ser capaz de captar la información en tiempo real de las antenas que están emitiendo en las bandas reservadas para el estándar GSM. En segundo lugar debe procesar la información captada, utilizando un algoritmo de trilateración que permita optimizar dicho procesamiento. Y por último, el sistema de código libre debe ser capaz de representar visualmente la información obtenida en un mapa para facilitar su comprensión.

El objetivo final, tras el correcto funcionamiento del sistema, se centraría en realizar un estudio para comprobar la precisión del mismo y conocer su fiabilidad en función a los datos aportados por sistemas de posicionamiento ya consolidados.

1.2. Plan de trabajo

Para alcanzar el objetivo principal anteriormente mencionado, se han seguido los pasos que están plasmados en esta sección, a través de un Diagrama de Gantt donde se representa el tiempo dedicado a cada una de las tareas.

- Estudio previo a la realización del TFG: se compone de las reuniones con los tutores donde se planteó el tema a desarrollar y el enfoque para crear un sistema de geoposicionamiento basado en las antenas de telefonía móvil; del acuerdo sobre los requisitos que debía tener el proyecto (tenía que ser un sistema *open source* desarrollado sobre Linux) y el *hardware* necesario (Dongle RTL_SDR, definido en el punto 4.1.1) para la captación de las señales electromagnéticas.



Figura 1.1: Dongle RTL_SDR

- Estudio previo de la materia: realización de la investigación sobre el tema a tratar y los proyectos existentes con el fin de conocer en profundidad la materia, así como sus diseños y la precisión de los mismos. Dicha investigación resultó en la existencia de varios proyectos similares, la mayoría basados en terminales Android, pero a pesar del gran desarrollo de los sistemas de posicionamiento que hacen uso de las redes móviles no se encontró ningún proyecto con implementación semejante a la de “ISAC-Nav”.
- Planificación del proyecto: antes de empezar a desarrollar el sistema, se definieron los programas necesarios para la captación de las señales que tenían que ser compatibles con el *hardware*, como se procesaría la información obtenida y la representación de esta. Por ello, el proyecto se estructura en tres módulos: captación, procesamiento y representación.
- Desarrollo de los módulos: al ser módulos independientes, se pueden desarrollar y probar uno a uno. Estos módulos se comunican a través del sistema de ficheros, y un *script* es el encargado de ir invocándolos durante la ejecución.
- Unificación y solución de Errores: en esta etapa se creó el *script* y se solucionaron los errores del desarrollo.
- Pruebas del sistema: cuando el proyecto ya era funcional, se realizó un pequeño estudio para ver la eficiencia del sistema y se comparó la precisión del proyecto con otros dos sistemas de posicionamiento ya existentes.
- Redacción de la memoria: en primer lugar se configuraron los objetivos y se plasmó el resultado del trabajo de investigación previo de la materia. En segundo lugar se fueron realizando los trabajos de implementación del sistema, durante el que se fueron tomando apuntes de los resultados y una vez que el sistema fue viable, se inició la redacción de la presente memoria. En último lugar, se ha trabajado y modificado los errores presentes en el sistema y en su parte documental, la memoria.

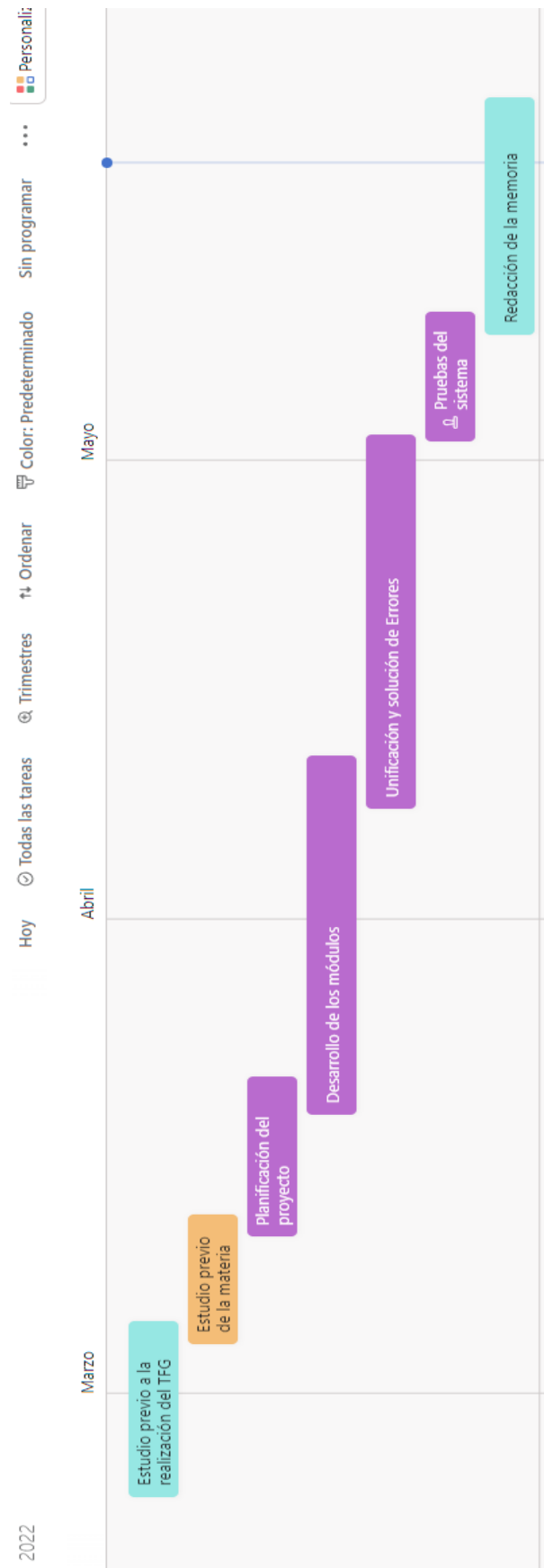


Figura 1.2: Diagrama de Gantt de “ISAC-Nav”

Capítulo 2

Introduction

With the emergence of more and more advanced technologies, the need arises to make use of real time positioning systems, whose objective is to estimate the user's position. There are more and more multifunctional devices, especially smartphones, which can work with applications that need to make use of their location to be able to offer their service.

These services are known as Location Based Services (LBS) and are present in a wide variety of areas: tourism, entertainment, health, etc. The high use of these systems was confirmed in the 2012 study conducted by the Pew Internet American Life Project, where 74% of smartphone users used the device's location services [1].

The most common positioning system if you want to know the location of a device is the GPS (Global Positioning System), therefore, usually the devices have a GPS module implemented that allows them to decode the radio signal transmitted by the satellites that make up this system.

The GPS signal travels through line of sight, which is a type of propagation where data can only be sent when the transmitting and receiving station is in sight, which leads to coverage problems. As a consequence the system GPS system often fails in open mountainous areas, tunnels and even in enclosed spaces where the thickness of the walls attenuates the signal.

But GPS is not the only positioning system and, bearing in mind its weaknesses, this project aims to provide an alternative. The first phase includes a study of the different localization and positioning techniques, going deeper into the most common existing methods to create positioning systems based on the mobile network.

The main purpose of the mobile telephone network is the transmission of data. For wireless data transmission, it uses electromagnetic signals that link the devices to the network remotely. These wireless communications use specific standards and protocols that allow the devices to communicate with the transceiver antennas that provide access to the network.

Within the electromagnetic spectrum, the mobile telephone network makes use of bands reserved in the GSM (Global System for Mobile) standard, which can be identified by wavelength, frequency and intensity.

Currently, there are numerous positioning systems based on the telephone network, such

as AGPS, which combines the mobile network infrastructure with the GPS system. However, the present project tries to create a system totally independent of the GPS signal. the GPS signal.

2.1. Objectives

The main objective of the “ISAC-Nav” project is to create a positioning system that allows geolocation of the user through the mobile network, based on the signal from at least two antennas.

For the correct implementation of the positioning system, it is necessary to create an open source system with three secondary objectives. Firstly, it must be able to capture real-time information from the antennas that are transmitting in the bands reserved for the GSM standard. Secondly, it must process the captured information, using a trilateration algorithm to optimize the processing. And finally, the open-source system must be able to visually represent the information obtained on a map to facilitate its understanding.

The final objective, after the correct operation of the system, would be to carry out a study to check its accuracy and to know its reliability based on the data provided by already consolidated positioning systems.

2.2. Work plan

In order to achieve the main objective mentioned above, we have followed the steps shown in this section, through a Gantt Chart where the time dedicated to each task is represented.

- Study prior to the completion of the TFG: the project consists of the meetings with the tutors where the theme to be developed and the approach to create a geopositioning system based on mobile phone antennas were discussed. The agreement on the requirements that the the project needs (it had to be an open source system developed on Linux) and the necessary hardware system developed on Linux) and the necessary hardware (RTL_SDR dongle, defined in 4.1.1) to capture the electromagnetic signals.



Figura 2.1: Dongle RTL_SDR

- Previous study of the TFG: we investigated if there was a project similar to the one proposed in this memory, we saw similar projects (most of them are developed for Android) and that the subject was already quite developed, even so, we did not find projects with the same implementation and approach.
- Previous study of the subject: research on the subject to be dealt with and the existing projects in order to know in depth the subject matter, as well as their designs and their accuracy. This research resulted in the existence of several similar projects, most of them based on Android terminals, but in spite of the great development of positioning systems that make use of mobile networks, no project with an implementation similar to “ISAC-Nav” was found.
- Project planning: before starting to develop the system, we defined the programs needed to capture the signals, which had to be compatible with the hardware, how the information obtained would be processed and how it would be represented. Therefore, the project is structured in three modules: acquisition, processing and representation.
- Development of the modules: being independent modules, they can be developed and tested one by one. These modules communicate through the file system, and a script is in charge of invoking them during execution.
- Unification and Bug Fixing: at this stage the script was created and the development bugs were fixed.
- System testing: when the project was already functional, a small study was conducted to see the efficiency of the system and the accuracy of the project was compared with two other existing positioning systems.
- Drafting of the report: firstly, the objectives were set and the results of the previous research work on the subject were set out. Secondly, the system implementation work was carried out, during which notes were taken of the results and once the system was viable, the writing of this report was started. Lastly, the errors present in the system and in its documentary part were worked on and modified.

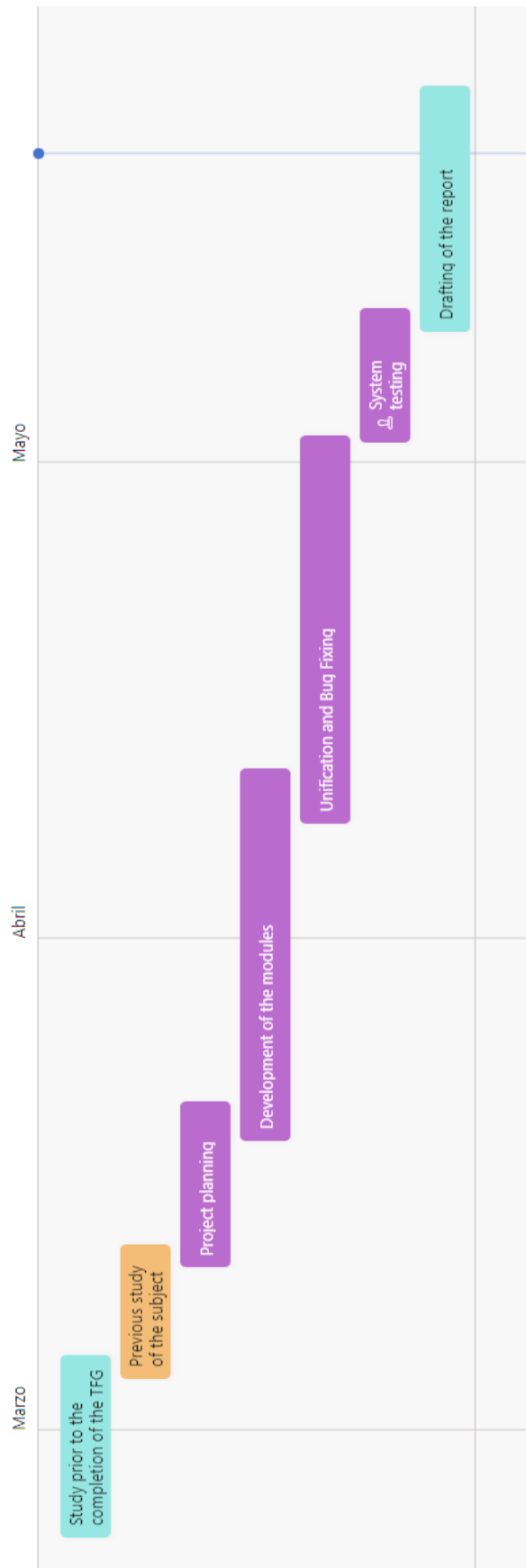


Figura 2.2: "ISAC-Nav" Gantt Chart

Capítulo 3

Estado del arte

El ser humano siempre ha tratado de comunicarse a larga distancia, las primeras técnicas utilizaban medios como sonidos, almenaras o señales de humo. Estas comunicaciones se fueron modernizando hasta conseguir transmitir información a través de microondas electromagnéticas, las cuales viajan a una mayor distancia y aportan información más detallada, dando origen al sector de las telecomunicaciones.

La primera comunicación inalámbrica fue en 1895. Guglieno Marconi consiguió realizar la primera transmisión telegráfica utilizando ondas de radio (mediante radiaciones electromagnéticas) [2].

Estos sistemas fueron evolucionando hasta dar pie a los radioteléfonos que son los predecesores de los teléfonos móviles. Los radioteléfonos se componen de un dispositivo de captación de voz que conectado a una antena permitía transmitir el mensaje en una frecuencia concreta. La comunicación era *simplex* es decir, unidireccional, para poder recibir el mensaje era necesario tener un receptor dentro del rango de emisión de la antena.

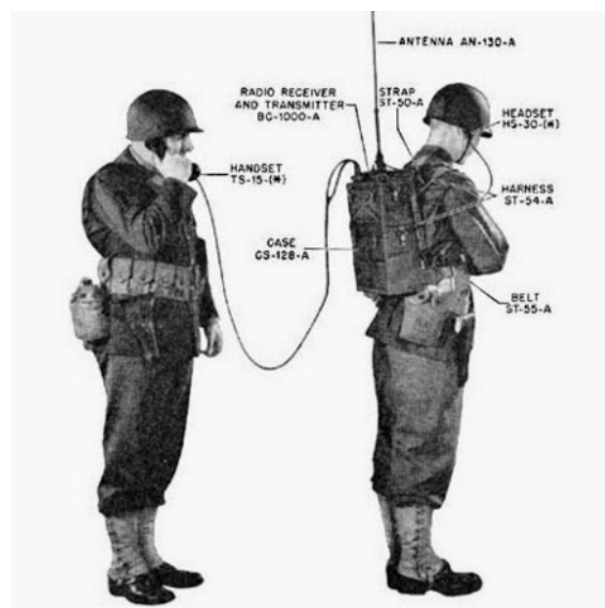


Figura 3.1: Radioteléfono. Fuente: <https://es.wikipedia.org/wiki/SCR-300>

En 1947, la empresa AT&T, estaba tratando de implementar estos dispositivos en sus vehículos y propusieron una serie de cambios a la Comisión Federal de Comunicaciones (FCC, *Federal Communications Commission*) para que la comunicación no se detuviera si estos salían del rango de las antenas. Estos cambios consistían en reservar un rango de frecuencias concretos para las telecomunicaciones, este sistema se convertiría más tarde, en las redes móviles que conocemos hoy en día [3].

3.1. Estaciones base

Las estaciones base (BS, *Base Station*) son antenas fijas que sirven para conectar de manera bidireccional dispositivos de baja frecuencia. Son el punto de acceso a la red móvil y son capaces tanto de transmitir como de recibir información. Generalmente están ubicadas en una posición elevada y su alcance varía en función de donde esté ubicada, pudiendo ir desde unos cientos de metros en zonas urbanas hasta varios kilómetros en entornos rurales.

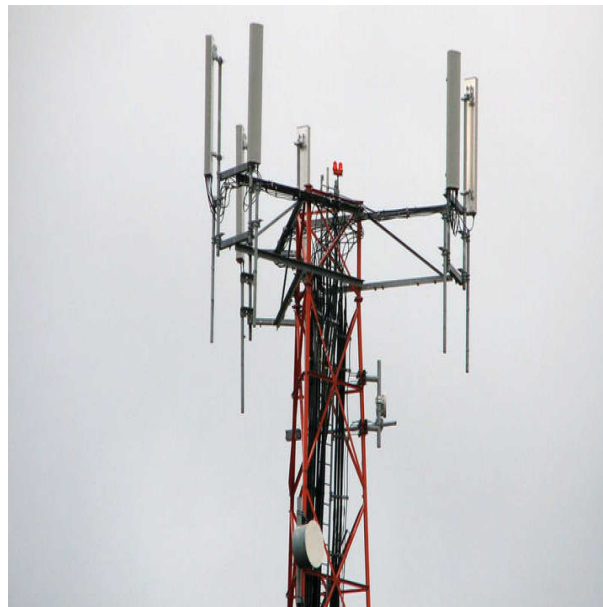


Figura 3.2: Estación base. Fuente: <https://www.rtve.es/noticias/20120227/estacion-base-telefonía-movil-chip/501966.shtml>

3.2. Red móvil

Una red móvil es el sistema necesario para que se pueda establecer una comunicación entre la central (propiedad de la operadora) y los móviles que están conectados a la red.

Está formada por un conjunto de estaciones base que se encargan de dar soporte a un área delimitada. Esta área son celdas, el tamaño de las celdas está condicionado por varios factores como pueden ser el tipo de BS o la ubicación de estas y están intercomunicadas mediante ondas de radio.

Estas redes tienen como objetivo conectar a los usuarios a través de un sistema perteneciente a un operador, que es el encargado de la gestión de la red. La comunicación a través de las radiofrecuencias, es el enlace final con el usuario.

En las redes móviles, el dispositivo inalámbrico se conecta a través del aire emitiendo ondas que alcanzan una BS, esta BS se comunica con el sistema del operador que se encarga de encaminar la comunicación a través de su red hasta llegar a su destino. Para que la comunicación se pueda dar, es necesario que los dispositivos estén dentro del rango de alguna BS.

En la actualidad, las redes móviles tienen una infraestructura con una cobertura prácticamente global y una gran parte de la población cuenta con acceso a esta red (en 2016 se estimó que un 76 % de la población mundial hace uso de la infraestructura de telefonía móvil [4]).

En este punto la red móvil no se diferencia en gran medida del radioteléfono. La propuesta de la empresa AT&T era que los usuarios pudieran mantener una conversación desde el coche y para ello, era necesario que el dispositivo pudiera estar en movimiento sin que se cortase la conexión. Para que se cumpliera esta condición era necesario un protocolo que permitiera ir cambiando de BS y de celdas antes de que el dispositivo se saliera de rango.

3.2.1. Handover

Es un protocolo tanto de conexión como de traspaso de celda para dispositivos en movimiento que tengan una comunicación activa, es transparente para el usuario y es un protocolo clave para conectar el dispositivo a una BS.

Una vez queda establecida la comunicación, es posible que el dispositivo esté en movimiento y empiece a alejarse de la BS con la que está intercambiando mensajes, cuando esto sucede, el receptor necesita más potencia para poder mantener la comunicación con la BS, si continúa alejándose la comunicación empeora hasta llegar al punto donde se pierde la conexión. Afortunadamente esta situación no ocurre puesto que cuando el dispositivo empieza a perder cobertura, activa un protocolo para actualizar la BS a la que está conectado.

Este protocolo tiene varias fases

- **Iniciación:** en esta fase, el dispositivo empieza a comunicarse con las diferentes BS que están dentro de su rango, empiezan a intercambiar mensajes y mide la intensidad y la calidad de la señal recibida. Una vez que las ha evaluado, envía un reporte de vuelta.
- **Selección de red:** las BSs comunican los resultados a la unidad de control de la red, esta procesa la información y en función de los recursos que estén disponibles y de la carga de la red tomará la decisión de asignar, mantener o cambiar la BS del dispositivo. En caso de traspaso, le envía la información al dispositivo de la nueva BS que se le ha asignado para que empiece el traspaso.
- **Ejecución:** el dispositivo empieza a intercambiar mensajes con la nueva BS para ejecutar los protocolos de conexión necesarios.

- Sincronización: una vez que la conexión ha sido aceptada, el dispositivo reconfigura su información para empezar a transmitir a través de la nueva BS y termina enviándole un mensaje de que el traspaso ha finalizado.

Este protocolo tiene como fin disfrutar de los mejores servicios disponibles de la red en ese momento, normalmente es invisible para el usuario, ya que este no suele notar cuando ha sucedido un intercambio de BS, aun así, no es instantáneo y conlleva una serie de mensajes que pueden generar algunos cortes en la comunicación [5].

3.2.2. RSSI

El *handover* está basado en el RSSI (*Received Signal Strength Indicator*), una escala de referencia (relacionada con los milivatios) que sirve para medir los niveles de potencia recibida en las redes inalámbricas, como pueden ser las redes Wifi o de telefonía móvil, junto con el ruido y las interferencias.

La escala tiende a dar valores negativos (puede dar valores positivos, pero no son muy comunes), se utiliza el 0 como el valor de referencia, 0 RSSI o 0 dBm ($0 \text{ dBm} = 1 \text{ mW}$) y la calidad de la señal irá empeorando según vaya aumentando el valor negativo de la señal.

Esta escala no sirve como indicativo para medir la calidad de la señal que recibe el dispositivo, para ver la calidad se tiene que contrastar con el ruido que acarrea la señal.

A continuación se presenta una tabla de la escala RSSI, cuyos valores no están estandarizados y no tienen por qué tener el mismo significado en otros dispositivos.

Valor	Significado
0	Valor de referencia
-40 a -60	Tasa idónea, con buenas tasas de transferencia
-60	Buena, ajustando la transmisión se pueden conseguir conexiones estables
-70	Con estos niveles la señal ya tiene poca intensidad, sigue siendo bastante buena pero puede tener problemas si hay lluvia o viento
-80	Señal mínima aceptable para poder establecer una conexión

Cuadro 3.1: Tabla RSSI

La tabla de equivalencia aproximada en dBm sería:

Valor (dBm)	Significado
-76	Perfecta
-89 y -77	Buena
-97 y -90	Decente
-103 y -98	Baja
-112 y -104	Poca cobertura
-113 y -132	Muy poca cobertura
-135	Sin cobertura

Cuadro 3.2: Tabla dBm

3.2.3. Generaciones Móviles

La tecnología relacionada con las telecomunicaciones ha progresado y se ha ido extendiendo de forma masiva durante las últimas décadas. Han sido numerosos los cambios significativos que han propiciado una revolución en el mundo de la telefonía móvil, y por consecuencia, han provocado que la evolución de las redes móviles se pueda encasillar en generaciones móviles.

- 1 generación: GPRS (*General Purpose Radio Service*), es el tipo de red más lenta y básica, tiene una velocidad de 80 Kbps de velocidad teórica. Sirve para enviar SMS y establecer llamadas, se puede navegar con una velocidad muy baja y da servicio a aplicaciones que requieren poco uso de red.
- 2 generación: EDGE (*Enhanced Data Rates for GSM Evolution*), es el hermano mayor del GPRS, tiene una velocidad teórica de 236 kbps, puede utilizar aplicaciones más pesadas y en esta red tienen preferencia la voz sobre los datos, no se pueden utilizar dos servicios a la vez.
- 3 generación: UMTS (*Universal Mobile Telecommunications System*), basa su funcionamiento en antenas UMTS, tiene un pico teórico de 2 Mbps, esta generación supuso un gran salto en la evolución de las redes, las principales características que incorporaron fueron:
 - a). Transmisiones simétricas/asimétrica.
 - b). 384 kbps en espacios abiertos y 2Mbps con baja movilidad.
 - c). Ancho de banda dinámico.
 - d). Soporte de conmutación de paquetes.
 - e). Acceso a Internet.
 - f). Servicios simultáneos en una conexión.
 - g). Mejora en la calidad de la voz.
 - h). Mayor capacidad y anchos de bandas más amplios.
 - i). Servicios dependientes de la posición.
 - j). *Roaming* (tanto internacional como entre distintas operadoras).

Luego se implementó el HSDPA (*High Speed Downlink Packet Acces*), que supuso una mejora de la velocidad teórica con un pico de 14 Mbps, y a continuación del HSPA+ (Evolved HSPA) que es una mejora de la red UMTS, alcanzando picos de 84 Mbps.

- 4 generación: LTE (*Long Term Evolution*), tiene una velocidad teórica de 150 Mbps, es la red que esta más extendida, no es una tecnología o un estándar definido como tal, es más bien un conjunto de tecnologías y protocolos, tiene una velocidad teórica de 1 Gbps en estático y 100 Mbps en movimiento.
- 5 generación: está empezando a implementarse y, de acuerdo con la asociación GSM, se prevé que cuente con 1.7 millones de subscriptores en 2025, la ventaja más destacable frente a sus predecesoras es que soportará un mayor ancho de banda,

lo que se traduce en mucha más velocidad de descarga llegando a competir con la velocidad por cable, actualmente ha alcanzado velocidades de hasta 3.6 Gbps.

1 generación	80 kbps
2 generación	236 kbps
3 generación	2 Mbps
3.5 generación	14 Mbps
4 generación	100 Mbps
5 generación	3.6 Gbps

Cuadro 3.3: Evolución de la velocidad de transmisión en las distintas generaciones

La infraestructura necesaria para dar soporte a las distintas generaciones es distinta, las nuevas generaciones se han ido implementando en paralelo a las generaciones pasadas, por lo tanto, es común que haya zonas que tengan conexiones más lentas, ya que las BSs pertenecen a generaciones pasadas. Esto es bastante común en zonas rurales donde no suele interesar instalar BSs que tengan un menor alcance.

3.3. Sistemas de posicionamiento

Los sistemas de posicionamiento tienen como fin determinar la posición, suelen estar basados en algoritmos que utilizan como datos de entrada la distancia del dispositivo respecto a unos puntos ya conocidos. El GPS es el sistema más conocido, seguido de los sistemas basados en telefonía móvil y Wifi. En este apartado se va a dar una breve explicación de los sistemas GPS y se va a profundizar en las distintas técnicas de posicionamiento que utilizan la infraestructura de las redes móviles.

3.3.1. Servicio de posicionamiento Global (GPS)

El GPS (*Global Positioning System*) es un sistema desarrollado por el departamento de defensa de Estados Unidos en los años 70, basado en un conjunto de satélites (originalmente era una constelación de 24 satélites, actualmente consta de 31 satélites activos), que están transmitiendo constantemente. Esta información puede ser captada por el receptor para localizar su posición. Suele ser necesario la información de 4 satélites y una serie de algoritmos de trilateración para ser capaz de calcular la posición con precisión.

El GPS tiene 3 componentes principales [6].

- Segmento espacial: compuesto por los satélites orbitando la tierra encargados de transmitir mensajes con su posición y el tiempo exacto en el que se emite el mensaje.
- Segmento de control: son los elementos que regulan el sistema, revisa que la información de los satélites sea correcta y de hacer alguna corrección si es necesario.
- Segmento de usuario: son los dispositivos que cuentan con un receptor GPS.

Al sistema GPS se le pueden sacar 2 grandes inconvenientes, uno es la dependencia de estar en el rango de cobertura de los satélites para ser capaz de calcular la posición del

dispositivo, esto hace que sea un método menos preciso en edificios y túneles y el otro es el alto consumo de batería por parte del módulo GPS.



Figura 3.3: Sistema GPS

3.3.2. Posicionamiento redes móviles

Estos sistemas de posicionamiento parten del conocimiento. Normalmente, necesitan co-tejar la información obtenida de las BSs (normalmente el identificador) que están al alcance del dispositivo, con una base de datos (BBDD) que tiene las direcciones de las distintas BSs, posteriormente se aplican distintos métodos para estimar la posición final del dispositivo. En este apartado se explican los principales métodos para estimar la posición que usan estos sistemas. Hay que tener en cuenta que la precisión de los métodos basados en la intensidad de la señal (RSSI), está muy condicionada por el entorno, no es lo mismo un entorno rural donde las BSs tienen más rango y las celdas son más grandes con un entorno urbano donde hay más edificios que condicionan la señal.

Celda de origen (COO, *Cell Of Origin*)

Es el método más sencillo, se basa en conocer el identificador de la celda de origen (CID, *Cell ID*), cada BS está asociada a una celda y cada celda tiene un identificador único, por lo tanto, se puede deducir que la estación móvil (MS, *mobile station*) se encuentra en el interior de esa celda. No es muy preciso, pero es un método muy rápido y ligero, este método es óptimo en ciudades donde la densidad de BS es mayor y las celdas son más pequeñas, sin embargo, es muy poco preciso en entornos rurales donde las BS pueden dar cobertura hasta a 35 Km de distancia.

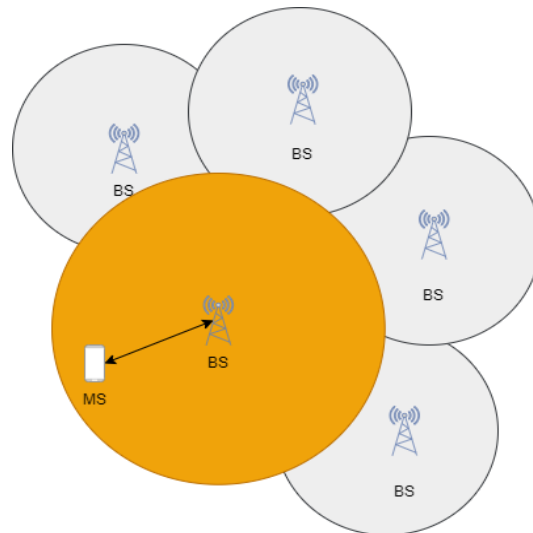


Figura 3.4: COO

Celda de origen con sectores (COO CS, *Cell Of Origin with Cell Sector*)

Es una variación del método anterior en el que se tienen en cuenta varias BS, se sabe que las BSs tienen 3 antenas y que cada antena transmite en un sector de 120 grados, si se conoce la antena de las que se está recibiendo información, se puede saber en qué sector alrededor de la BS se encuentra la MS actualmente, e intentar calcular su posición mediante algoritmos de trilateración.

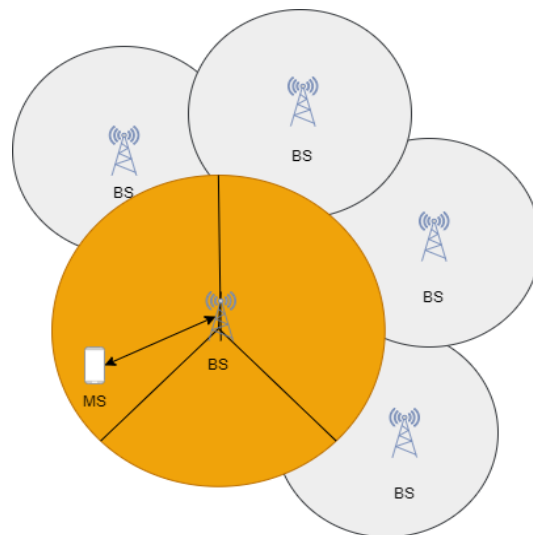


Figura 3.5: COO CS

Celda de origen con sectores y RSSI (COO CS with RSSI)

Este método es una optimización del COO CS, intenta ajustar la posición teniendo en cuenta el RSSI de las señales obtenidas para ajustar las distancias a la que se encuentran la MS de las BSs y posteriormente aplicando algún algoritmo de trilateración, este método

está siendo el más implementado ya que la información está disponible prácticamente para todos los usuarios con teléfonos compatibles con GSM.

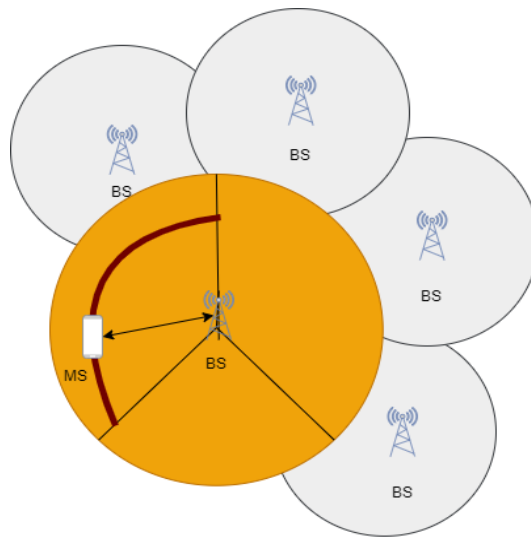


Figura 3.6: COO CS with RSSI

Avance de tiempo (TA, *Timing Advance*)

En el avance de tiempo se aprovecha una característica de las BSs, utilizan una tecnología llamada TMDA (*Time Division Multiple Access*), es un método que permite multiplexar los canales de entrada a las BSs, esto permite aumentar la cantidad de dispositivos que pueden acceder a la misma BS ya que a cada banda se le asigna una fracción de tiempo. Utilizando las rendijas de tiempo asignadas a la banda y una variable de la BS llamada Time Advance (TA), que indica el retardo que hay entre la BS y la MS, se puede calcular la distancia en función del tiempo de transmisión.

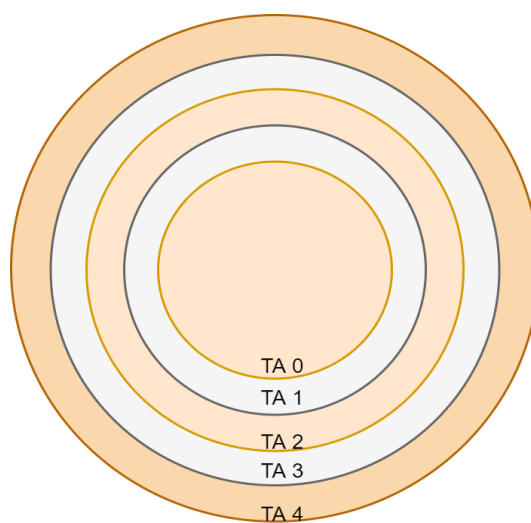


Figura 3.7: TA

Angulo de llegada (AOA, *Angle Of Arrival*)

Los sistemas AOA se basan en el ángulo respecto a 2 o 3 BS, conociendo las posiciones de las BSs y la distancia entre estas, se puede triangular la posición si se es capaz de calcular el ángulo con el que incide la señal en la MS. Estos sistemas requieren un *hardware* específico que sea capaz de medir el ángulo de incidencia de las señales recibidas, por eso este método es menos atractivo para los teléfonos móviles.

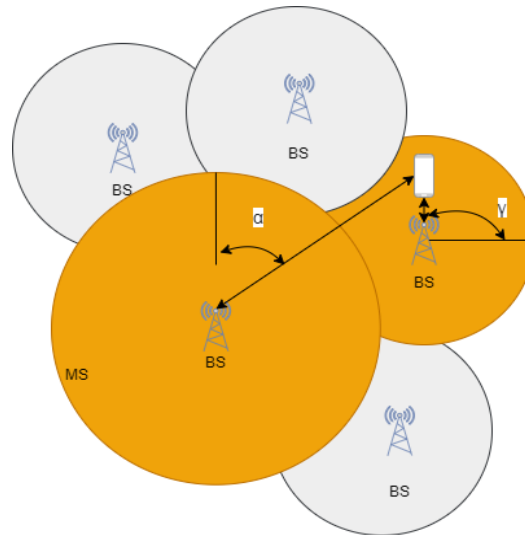


Figura 3.8: AOA

Tiempo de ida y vuelta (RTof, *Round Trip Of Flight*)

Este método se basa en calcular la diferencia de tiempos de llegada de una señal. Esta señal se envía a varias BS y estas reenvían el mensaje a la MS, basándose en la diferencia del tiempo de llegada de las señales se puede estimar la distancia a las BSs, con esta información se puede aplicar algún algoritmo.

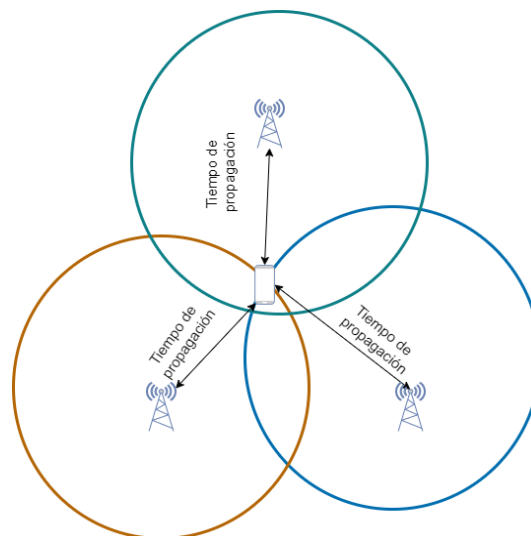


Figura 3.9: RTof

Diferencia de tiempo de llegada (TDOA, *Time Difference Of Arrival*)

Parecido a RToF, este método se basa en enviar simultáneamente una señal de las BSs a la MS, la ubicación también puede estar estimada basándose en el punto de intersección de las hipérbolas que se forman al juntar la señal de 2 o 3 BSs. Este es el método empleado por el GPS.

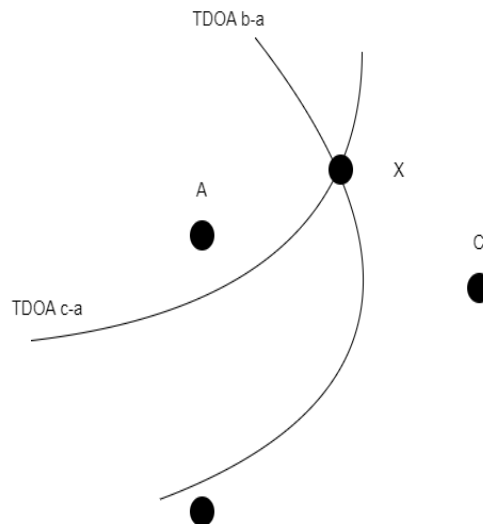


Figura 3.10: TDOA

Diferencia de tiempo observada mejorada (EOTD, *Enhanced Observed Time Difference*)

Es una versión de TDOA mejorada donde varias BSs mandan una señal simultánea a la MS y calcula la posición en función de la variación del tiempo de llegada. La precisión del reloj en estos métodos es muy importante ya que, cualquier mínimo error en la medición produce una gran variación del resultado, una desviación de 1 microsegundo conlleva un error de 150 m [7].

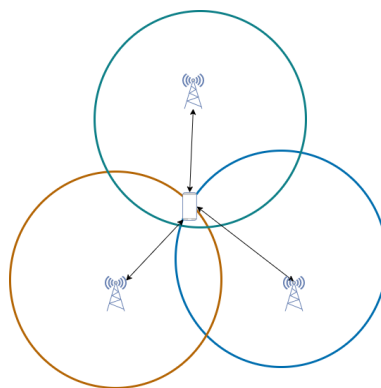


Figura 3.11: EOTD

Método	Precisión
Celda de origen - COO	100 m-35 Km
Celda de origen con sectores - COO CS	100 m-35 Km
COO CS with RSSI	10 0m-20 Km
Angulo de llegada - AOA	50 m-150 m
Tiempo de ida y vuelta - RToF	50 m-150 m
Diferencia de tiempo de llegada - TDOA	50 m-150 m
Diferencia de tiempo observada mejorada - EOTD	50 m-150 m

Cuadro 3.4: Precisión de los sistemas basados en GSM

3.3.3. Sistemas híbridos

Son sistemas que mezclan el GPS con la red de telefonía móvil, se denomina GPS asistido (AGPS, *Assisted Global Positioning System*), este sistema ofrece una cobertura y una precisión mayor a los sistemas basados únicamente en redes móviles, el AGPS está conformado por:

- Una MS con un receptor GPS.
- Un servidor de referencia que ve de manera simultánea los mismos satélites que ve la MS.
- Una red móvil.
- Sistema GPS.

Esta red puede predecir los satélites que envían la señal GPS que recibe la MS, realizar los cálculos necesarios y transmitirlos a la MS, con esto se puede hacer que el módulo GPS de la MS sea menos potente, ya que no necesitara tanta potencia para detectar y demodular las ondas del GPS. Este sistema trata de suplir los puntos débiles del GPS al distribuir los recursos necesarios. Tiene unos márgenes de error de 50m [8].

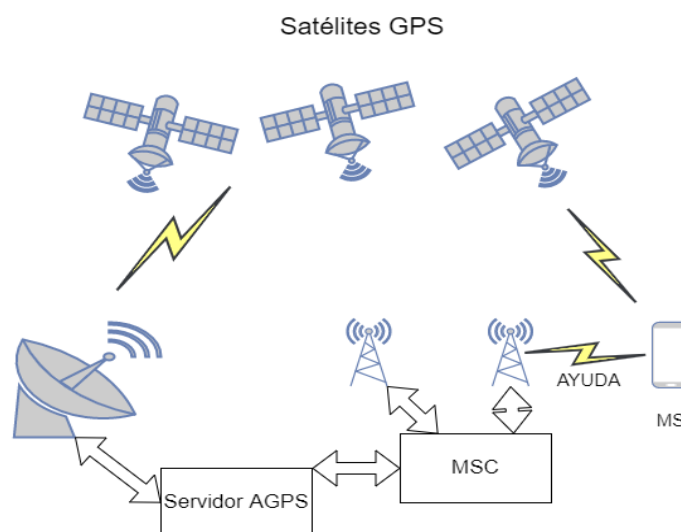


Figura 3.12: Esquema AGPS

3.4. GSM

GSM es un sistema de estándares de telefonía digital, está diseñado para crear un sistema público de redes móviles con frecuencias reservadas con el fin de reemplazar los sistemas anteriores, que eran incompatibles entre sí.

Está desarrollado y regulado por el ETSI (*European Telecommunications Standards Institute*), una organización de normalización independiente. En el 2011 ya lo implementaban entre un 80-85 % de los dispositivos móviles [7].

Banda	Nombre	Uplink(MHz)	Downlink(MHz)	Notas
GSM 850	GSM 850	824,0 - 849,0	869,0 - 894,0	EEUU, Sudamérica y Asia
GSM 900	P-GSM 900	890,0 - 915,0	935,0 - 960,0	La más extendida
GSM 900	E-GSM 900	880,0 - 890,0	925,0 - 935,0	Extensión de GSM 900
GSM 900	R-GSM 900	876,0 - 880,0	921,0 - 925,0	GSM ferroviario
GSM 1800	GSM 1800	1710,0 - 1785,0	1805,0 - 1880,0	Incompatible con GSM 1900
GSM 1900	R-GSM 900	1850,0 - 1910,0	1930,0 - 1990,0	Usada en Norteamérica

Cuadro 3.5: Frecuencias GSM

3.4.1. Arquitectura

La arquitectura del GSM se basa 3 subsistemas:

- El subsistema de estaciones base (BSS, *Base Station Subsystem*), conformado por el BTS (*Base Transceiver System*), que agrupa las BS y el controlador del sistema, (BSC, *Base Station Controller*), que es el encargado de encaminar la información de las MSs que están conectadas a la red.
- El subsistema de intercambio de red (NSS, *Network Switching Subsystem*), se encarga de conectar el centro de conmutación móvil (MSC, *Mobile Switching Center*) con el BSC.
- El subsistema de soporte a las operaciones (OSS, *Operation Support Subsystem*), es el encargado de mantener y dar apoyo a los otros subsistemas.

Los canales GSM son los encargados de establecer las distintas conexiones utilizando el espectro de frecuencias que tienen reservadas, utilizan 2 canales, el canal de tráfico, que se encarga del flujo de los datos y el canal de control, que supervisa la sincronización y la señalización de los distintos elementos del sistema [9].

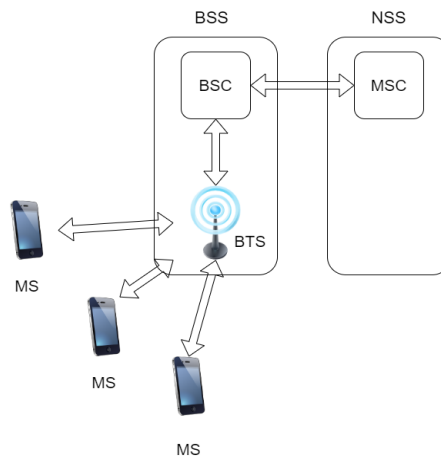


Figura 3.13: Red GSM

3.5. Trilateración

Cuando se trata de estimar la distancia, lo más intuitivo es pensar en un sistema basado en antenas isotrópicas, usar un algoritmo que calcule el tiempo de vuelo y aplicar una fórmula semejante a: $d=t*v$; donde “d” es la distancia, “t” es el tiempo y “v” es la velocidad de propagación de la onda.

Estos sistemas requieren un *hardware* específico y elementos de sincronización que permitan calcular el tiempo de vuelo de la onda. Son sistemas que requieren una gran precisión, por eso utilizan relojes y protocolos que permitan la sincronización entre la BSs y la MS.

No se va a emplear esta tecnología a la hora de desarrollar “ISAC-Nav”, por lo que a continuación se van a exponer técnicas de posicionamiento para mejorar la precisión en métodos que utilizan la información intercambiada entre la MS y las BSs para intentar calcular la distancia entre estas.

Un dato que nos puede ayudar para aumentar la precisión es el RSSI, gracias a una BBDD se sabe que las antenas en un entorno urbano emiten con la misma potencia y que tienen el mismo rango de alcance, aun así, el RSSI está influenciado en gran medida por el entorno, esto hace que los resultados en las medidas sean muy variados y poco uniformes, es por esto que se suele usar como referencia y normalmente los métodos basados en el RSSI suelen estar asistidos por acelerómetros, giroscopios o balizas *bluetooth* que mejoran su precisión.

A continuación, se van a explicar una serie de algoritmos que se pueden emplear para estimar la ubicación final de la MS. Estos algoritmos crean un polígono utilizando como vértices las posiciones de las BSs que es capaz de identificar la MS, y para la distancia entre la MS y las BSs usan la información del RSSI recopilada durante la ejecución por un dongle.

3.5.1. Centroide

El centroide o baricentro de una figura geométrica plana es un punto con la propiedad de que, cualquier recta que pase por él, divide al polígono en dos segmentos del mismo tamaño, también se puede definir como la media aritmética de todos los puntos que forman un polígono. Se calcula con la siguiente formula:

$$C = \frac{\sum x_i}{i} \quad (3.1)$$

En este proyecto la “x” estaría conformada por las coordenadas de las estaciones bases e “i” es el número de nodos del polígono, iría en función del número de BS detectadas en la última medición, se puede optimizar este método si se aplica el RSSI a los nodos, en ese caso se estaría hablando del centro de masas.

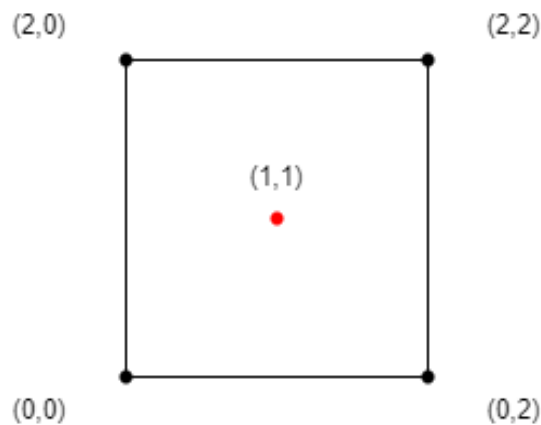


Figura 3.14: Centroide de un polígono

3.5.2. Centro de masas

El centro de masas es el punto geométrico donde la posición relativa y ponderada de la masa resultante de aplicar las fuerzas suma cero. También es conocido como el centro de gravedad, en nuestro caso las fuerzas están representadas por el RSSI. Usaremos un sistema discreto formado por un conjunto de masas puntuales.

$$r_m = \frac{\sum_i m_i r_i}{\sum_i m_i} = \frac{1}{M} \sum_i m_i r_i \quad (3.2)$$

Para poder aplicar el centro de masas es necesario darle un valor positivo al RSSI. Al obtener las mediciones el valor del RSSI esta en dBm (negativo), por lo que es imprescindible convertirlo a vatios con la siguiente fórmula.

$$XdBm = 10^{\left(\frac{x}{10} - 3\right)W} \quad (3.3)$$

Se emplea este algoritmo, el cual con el conocimiento de la posición de las BS y su RSSI es capaz de calcular el centro de masa del polígono resultante.

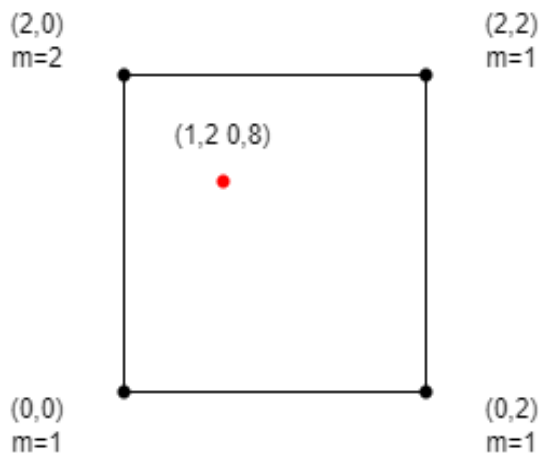


Figura 3.15: Centro de masas de un polígono

3.6. Cuestiones Legales

Tras el estudio técnico es interesante abordar el marco legal que regula el posicionamiento basado en tecnologías móviles.

Al tratarse de un fenómeno reciente aún no ha sido definido un marco legal y los estados miembros de la UE, incluido España, no prevén una regulación expresa de los sistemas de geolocalización a través de *Smartphones*. Estos sistemas por razones obvias presentan diversas cuestiones que los proveedores deben conocer y respetar para garantizar el derecho a la intimidad. También se tienen que encargar de la protección de los datos personales de los usuarios del servicio.

Actualmente la regulación de la geolocalización en el sector de los móviles se limita principalmente a la Directiva Europea de Protección de Datos 95/46/CE. A efectos de la aplicabilidad de esta directiva se considera que los datos de localización de los *Smartphones* y la combinación de la dirección única MAC junto con la localización calculada de un punto de acceso *WiFi* son datos personales.

Respecto a las BS y los datos obtenidos por los operadores de telecomunicaciones se aplica la Directiva Europea de Privacidad 2002/58/CE¹.

Estas directivas establecen el régimen general de la protección de datos sin llegar a resolver cuestiones específicas de la geolocalización a través de los dispositivos móviles, es decir, son meras directrices. No obstante, este instrumento conlleva una serie de riesgos propios[10].

¹<https://www.boe.es/doue/2002/201/L00037-00047.pdf>

Capítulo 4

Arquitectura

“ISAC-Nav” está compuesto de tres módulos independientes, estos módulos son: módulo de captación, módulo de procesamiento y módulo de representación. Los módulos se comunican a través del sistema de ficheros, cuya arquitectura se detalla a lo largo del presente capítulo junto a los programas y sus ficheros empleados. Además, se analizan las tecnologías propuestas para la consecución del proyecto y los motivos por los que fueron descartadas, modificadas o transformadas.

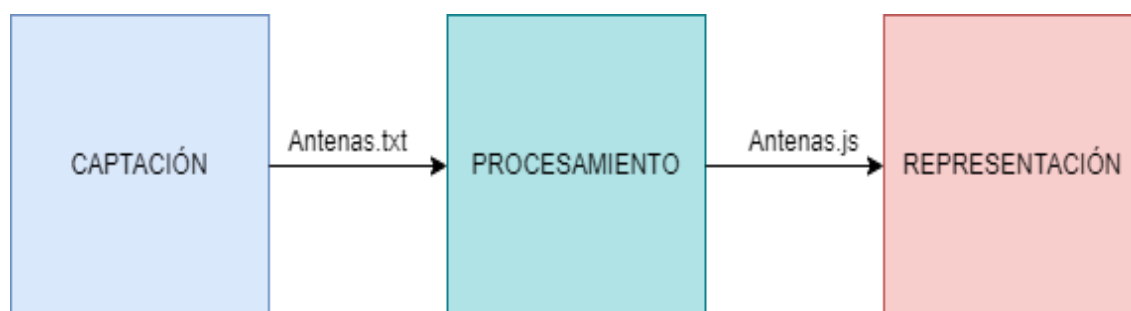


Figura 4.1: Esquema de la comunicación entre los módulos

Antes de empezar con el desglose de la arquitectura, es importante destacar que todo el contenido de este proyecto es *open source* y está pensado para que el código pueda ser visto, modificado y distribuido en la forma que se considere conveniente. Se puede acceder al código a través de mi Github: <https://github.com/PabloSaro>.

Para poder realizar ejecuciones locales de este proyecto, es necesario instalar previamente algunos programas y librerías en versiones concretas, también es necesario modificar las direcciones de los directorios en el *script* ajustándolas a las del nuevo sistema. Hay un manual de instalación en el anexo.

4.1. Módulo de captación

El módulo de captación, hace uso del dongle RTL-SDR (*Real Time Logic - Software Definition Radio*). Es el encargado de calcular el error del dispositivo, captar la información

de las antenas y dejar la información plasmada en ficheros para que otro módulo haga uso de ella.

4.1.1. Dongle RTL-SDR

Para ser capaz de recibir información de las antenas gsm, “ISAC-Nav” hace uso de un dispositivo *hardware*, el dongle RTL_SDR. El dongle empleado en este proyecto tiene un coste inferior a 15€ y cuenta con un chip Realtek RTL2832U, una antena y una conexión USB a través de la cual se conecta al sistema. Este dispositivo, funciona como receptor de radiofrecuencias para los programas que utilizan tecnología SDR, que tienen la peculiaridad de necesitar un *hardware* específico para funcionar correctamente.

El siguiente diagrama de bloques es una posible representación de cómo funciona un transceptor SDR.

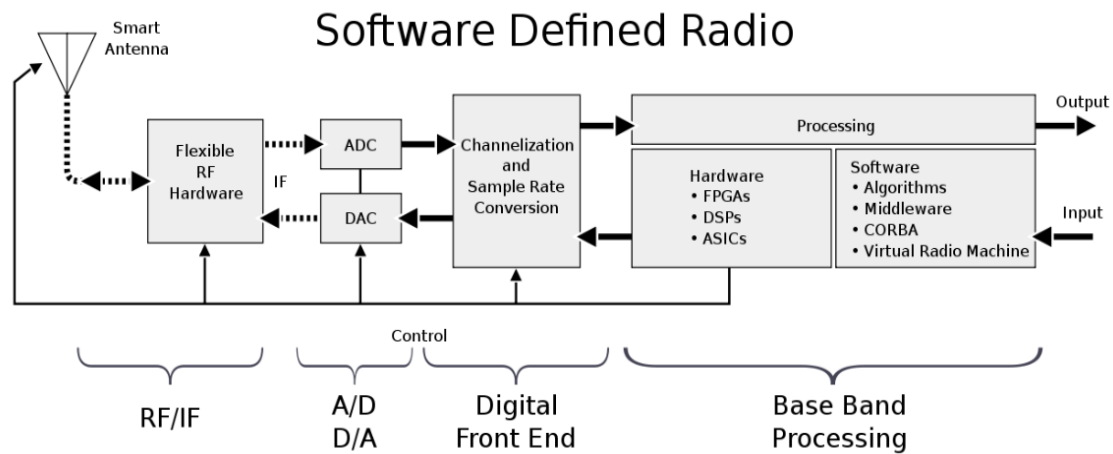


Figura 4.2: Diagrama de bloques conceptual de un transceptor SDR ideal. Fuente: https://es.wikipedia.org/wiki/Radio_definida_por_software.

4.1.2. Kalibrate-rtl

Kalibrate es un *software* que sirve para escanear las BSs que están emitiendo en la frecuencia de una banda dada y calcular la desviación local, en este proyecto se usa para medir el error del dispositivo (dongle RTL-SDR).

```

blanzas@blanzas-VirtualBox:~/kalibrate-rtl-master/src$ ./kal -h
kalibrate v0.4.1-rtl, Copyright (c) 2010, Joshua Lackey
modified for use with rtl-sdr devices, Copyright (c) 2012, Steve Markgraf
Usage:
  GSM Base Station Scan:
    kal <-s band indicator> [options]

  Clock Offset Calculation:
    kal <-f frequency | -c channel> [options]

where options are:
-s band to scan (GSM850, GSM-R, GSM900, EGSM, DCS, PCS)
-f frequency of nearby GSM base station
-c channel of nearby GSM base station
-b band indicator (GSM850, GSM-R, GSM900, EGSM, DCS, PCS)
-g gain in dB
-d rtl-sdr device index
-e initial frequency error in ppm
-E manual frequency offset in hz
-v verbose
-D enable debug messages
-h help

```

Figura 4.3: Parámetros de entrada de Kal

El error de los relojes de cuarzo se calcula en ppm (*Parts Per Million*). Normalmente, los relojes de cuarzo del microcontrolador del dongle tendrían que estar calibrados de forma externa por un osciloscopio que determina la precisión de una forma más exacta, sin embargo, no se cuenta con uno, razón por la cual será necesario usar un *software* para realizar el ajuste.

4.1.3. GNU-Radio

Es un kit *free* y *open source* de herramientas de desarrollo *software*, este kit, proporciona los bloques de procesamiento de señales necesarios para desarrollar aplicaciones basadas en radiofrecuencias. Grgsm necesita este software para funcionar.



Figura 4.4: GNU-Radio logo

4.1.4. Grgsm

Grgsm son una serie de programas en python diseñados para el procesamiento de señales GSM. Estos programas fueron escritos por Potr Krysik y son una continuación del proyecto *Aiprobe*. Los programas de Grgsm son:

- Grgsm_decode: programa diseñado para decodificar un canal GSM. Tiene la capacidad de decodificar canales de señalización y canales de tráfico con voz (el análisis de

los datos puede ser realizado en Wireshark y el sonido decodificado es almacenado en un archivo de audio).

- Grgsm_livemon: es un monitor interactivo de un solo canal C0, analiza este canal utilizando Wireshark.
- Grgsm_scanner: aplicación que escanea las bandas GSM e imprime la información sobre las BSs que es capaz de percibir que están emitiendo en esas bandas.

“ISAC-Nav” usa Grgsm_scanner para volcar, sobre un fichero (“Antenas.txt”), la información de las BSs que el dongle es capaz de captar en las bandas GSM, en el momento de la ejecución.

4.1.5. “Antenas.txt”

Es el fichero de salida del módulo de captación y el de entrada para el módulo de procesamiento. En este fichero, cada línea representa una banda de frecuencia diferente, si no ha encontrado una antena emitiendo en esa banda pondrá Args=, y si encuentra una BS pondrá una tupla con los siguientes argumentos.

Nombre	Definición
ARFCN	Frecuencia absoluta del canal (<i>Absolute Radio Frequency Channel</i>)
Freq	Frecuencia
CID	Identificador de celda
LAC	Identificador de área (<i>Location AreaCode</i>)
MCC	Código del país (<i>Mobile Country Code</i>)
MNC	Código de la red (<i>Mobile Network Code</i>)
Pwr	Valor del RSSI

Cuadro 4.1: Parámetros “Antenas.txt”

4.1.6. GQRX

Es un programa diseñado para explorar el espectro de las radiofrecuencias. Tiene muchas funcionalidades como observar, grabar, escuchar, etc., se puede utilizar para múltiples usos como puede ser sintonizar cadenas de radios, explorar los canales de voz en la banda de las comunicaciones aéreas o en el caso de “ISAC-Nav” para ver la información que pasa por las bandas que se han detectado con el Grgsm_scanner, estos canales no son de voz, así que, no se pueden escuchar las conversaciones que se transmiten por él.



Figura 4.5: GQRX logo

4.2. Módulo de procesamiento

Este módulo es capaz de estimar una posición a partir de los datos obtenidos por el Grgsm_scanner. Recibe como entrada el fichero “Antenas.txt”, y con la información de este, es capaz de localizar las BSs captadas por el módulo de captación, a continuación, hace una aproximación de la posición de la MS desde la que se han tomado las medidas. Para dicho cálculo se basa en la localización y la potencia que recibe de cada BS, después, aplica el algoritmo de centro de masas y para terminar, genera un fichero (“Antenas.js”) que será la salida de este módulo y la entrada del siguiente.

4.2.1. BBDD

La implementación de la base de datos ha sido uno de los mayores inconvenientes de este proyecto, se exploraron múltiples opciones (definidas en el punto 4.6.2) hasta llegar a OpenCellid ¹, la BBDD implementada en la versión final de “ISAC-Nav”.



Figura 4.6: OpenCellid logo

OpenCellid es una BBDD comunitaria y de libre acceso, creada para recopilar datos sobre las BS de los móviles, uno de los datos es la localización aproximada de las BSs. Esta BBDD se va actualizando por los datos aportados por los usuarios.

Los datos de OpenCellid se pueden utilizar con fines privados o comerciales por parte de los usuarios de todo el mundo. Estos son accesibles a través de una API (*Application Programming Interface*) o se pueden descargar como ficheros “.csv”.

“ISAC-Nav” cuenta con una copia local de la BBDD (“BBDD.csv”) con la información de las antenas de España, ya que no es un fichero muy pesado (78 Kb) y reduce la complejidad de la implementación. Los campos que usa son:

¹OpenCellid web: <https://opencellid.org>

Nombre	Definición
Radio	Tipo de la antena (GSM, UMT, ...)
Freq	Frecuencia
Cell	Identificador de celda (CID)
Area	Código del área (LAC)
MCC	Código del país
Net	Código de la red
Lat	Latitud
Lon	Longitud
Range	Rango de la antena

Cuadro 4.2: Parámetros usados de “BBDD.csv”

4.2.2. “ISAC-NAV”

Es el nombre que se le ha dado al programa principal del proyecto “Isac-Nav”. Es un programa modular escrito en C++ y está compuesto principalmente por 3 módulos (Datos, I/O, Radio) y un controlador.

Datos

Solo tiene un fichero, “Data.H”, con dos estructuras donde se almacenan los datos de las antenas y los resultados. Es necesario que este incluido en el resto de los módulos del programa para que puedan operar con las antenas.

Controlador

Está implementado en el “Main.cpp”. Tiene el control de lo que sucede en el programa, del orden en el que se llaman a las funciones y crea el vector de antenas del tipo definido en “Data.h”. Este vector se envía a los módulos y estos lo modifican.

También revisa que después de la lectura se mantengan dos o más antenas, en caso contrario detiene la ejecución e imprime un mensaje de error indicando que el número de antenas es insuficiente.

I/O

Está formado por dos ficheros, uno de cabecera (“Lector.h”) y otro con la implementación de las funciones declaradas en el fichero de cabeceras (“Lector.cpp”), este módulo tiene dos funciones en una ejecución que no tenga fallos (más de una antena).

- Comprobar las antenas: abre el fichero “Antenas.txt” y comprueba que cada antena está en la BBDD, en caso afirmativo se añade al vector de antenas y en caso negativo la descarta.

- Guardar las antenas: una vez que se han realizado los cálculos del programa (los ejecuta el módulo radio), almacena los resultados en un fichero (“Antenas.js”), que será el fichero que conecta el procesamiento con la representación.

Radio

Igual que el módulo anterior tiene dos ficheros, uno de cabecera (“Radio.h”) y otro con la implementación de las cabeceras (“Radio.cpp”), este recibe el vector de antenas, pasa el valor de RSSI a Watts, aplica el algoritmo del centro de masa y almacena los resultados en la estructura de los resultados.

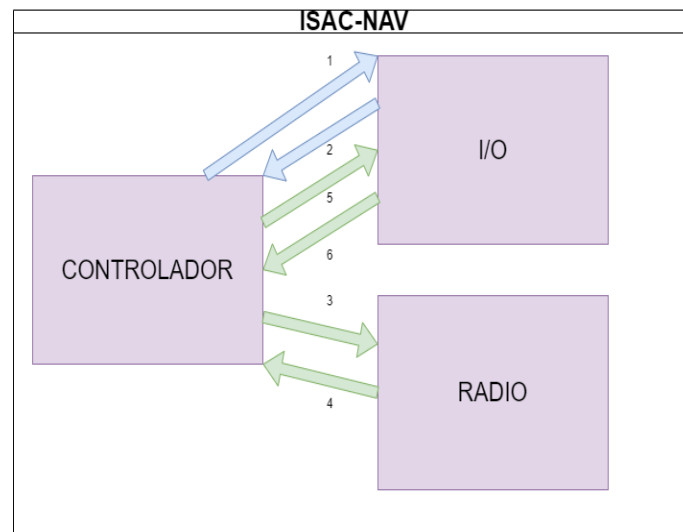


Figura 4.7: Flujo de ejecución de “ISAC-Nav”

4.2.3. Makefile

Es un fichero de texto plano con una serie de instrucciones en *bash*. Estas instrucciones tienen que respetar el árbol de dependencias (Figura 4.8) para poder generar los archivos intermedios de los módulos (“*.o”) y un ejecutable (“ISAC-NAV”).

Para compilar el programa, solo hay que ejecutar el comando *make* y para limpiar una compilación anterior usar el comando *make clean*.

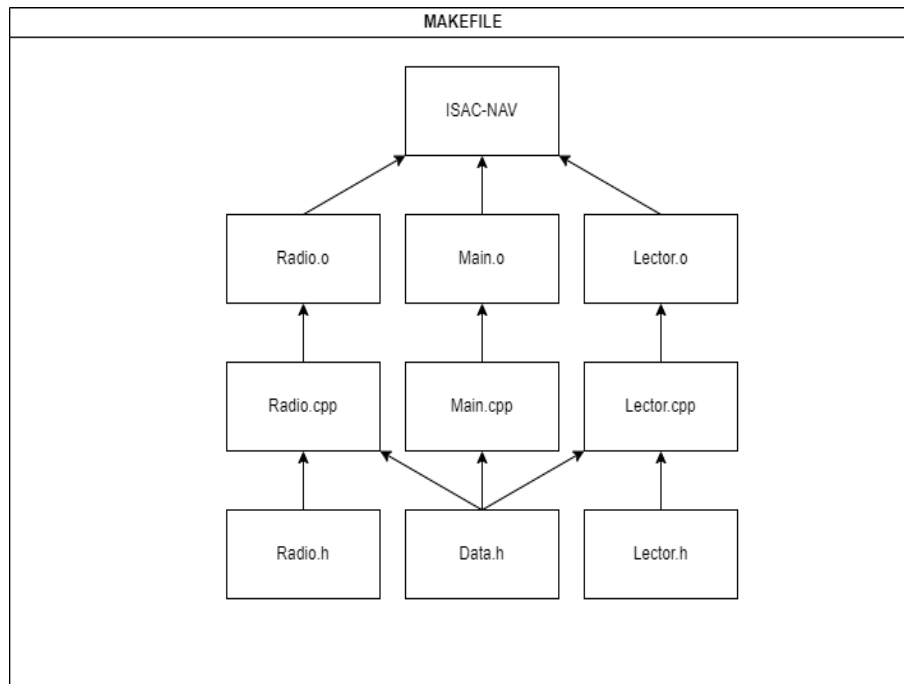


Figura 4.8: Diagrama Makefile

4.3. Módulo de representación

El objetivo de este módulo es darle una representación visual a los resultados sobre un mapa dinámico que se actualiza periódicamente. La representación está implementada sobre una página web que se ejecuta de manera local. No está subida a un servidor pero si hace peticiones a otros servidores, por ejemplo, para cargar la sección del mapa.

4.3.1. OSM

Para la representación del mapa, se hace uso de OSM (*Open Street Map*), un proyecto *free and open source*, que cuenta con más de 7.541.146 de usuarios registrados, estos han ido creando los mapas a partir de la información geográfica aportada por los mismos.



Figura 4.9: Open Street Map logo

4.3.2. Leaflet

Leaflet es una biblioteca desarrollada en java script, está diseñada por Vladimir Agafonkin y es un proyecto de código abierto centrado en crear mapas interactivos con OSM, cuenta con todas las características de mapeo necesarias para poder representar los resultados sobre un mapa interactivo.



Figura 4.10: Leaflet logo

4.3.3. Web

Implementada por 3 Ficheros:

- “Index.html”: es el fichero principal, tiene las llamadas a la API de Leaflet y Google e incluye a los otros ficheros.
- “AntenasFijo.js”: en este fichero está la inicialización del mapa, también está implementada la lógica de un botón que crea una llamada a una API de Google, esta API da la ubicación del dispositivo basándose en los datos de navegación de la MS y se puede usar para equiparar el nivel de precisión del proyecto en comparación con otro sistema.
- “Antenas.js”: es el fichero de salida del módulo de procesamiento, tiene la información en tuplas de la ubicación de la MS y de las distintas antenas que ha sido capaz de localizar.

Los ficheros JavaScript (“.js”), están agrupados en una carpeta llamada “js”.

4.3.4. Firefox

Firefox es el navegador que va a interpretar los ficheros web, es decir, donde se van a representar los resultados. Se ha seleccionado Firefox ya que es el navegador que viene por defecto en Linux, sistema operativo en el cual se ha desarrollado el proyecto.

4.4. Script

Se ha creado un *script* llamado “ISAC-NAV.sh.”^{en}cargado de lanzar el proyecto y ejecutar todos los módulos. Como se puede ver en el diagrama de flujo del *script* (Figura 4.13) comienza calculando el error del dispositivo, después lanza la web, llama al makefile para actualizar los ficheros a la versión más reciente y por último entra en un bucle donde

rastrea y procesa los datos que es capaz de encontrar. Al terminar la iteración del bucle habrá escrito el resultado en un fichero con los datos del último muestreo, la representación cambiará cuando el navegador se actualice.

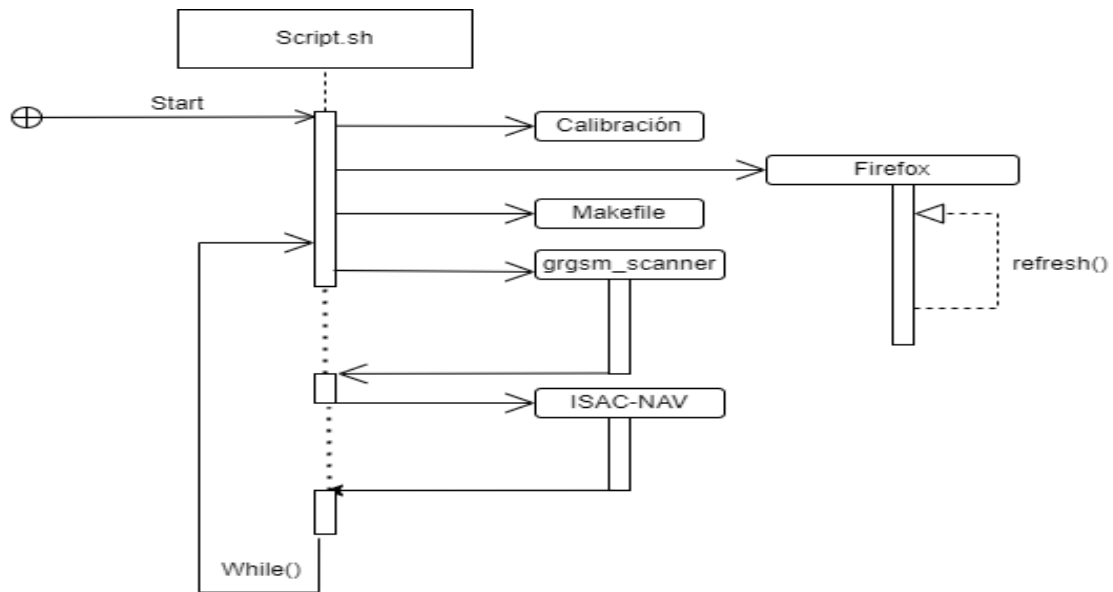


Figura 4.11: Diagrama de flujo del *Script*

4.5. Sistema de ficheros

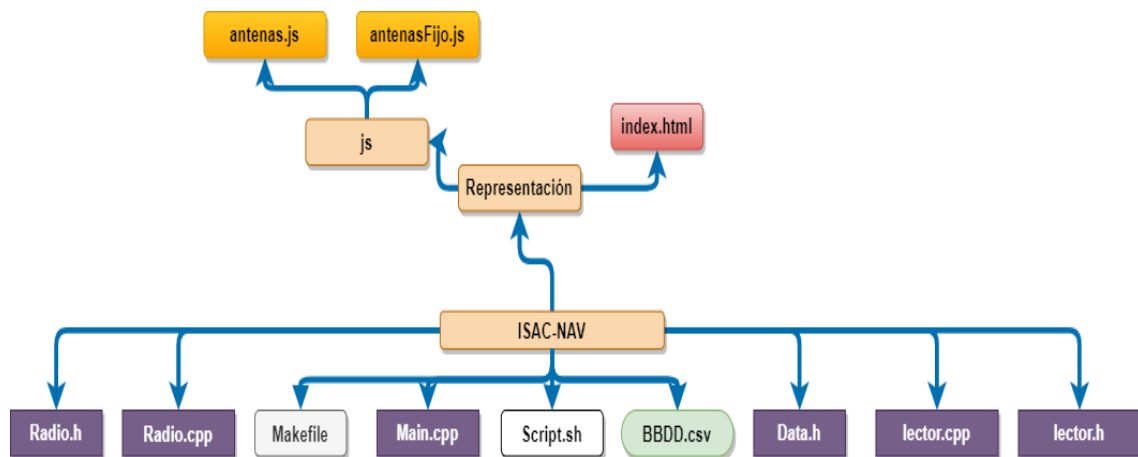


Figura 4.12: Jerarquía sistema de ficheros

Este diagrama es una foto de la jerarquía del sistema de ficheros antes de ejecutar el *script*.

4.6. Tecnologías descartadas

Durante el desarrollo del proyecto, se ha ido adaptando la arquitectura hasta llegar a la foto del proyecto que se ha plasmado en esta memoria. En esta sección, se hace un breve repaso de algunas tecnologías que han formado parte del proyecto y se justifica por qué han sido descartadas.

4.6.1. Docker

Docker es una aplicación que permite crear, probar e implementar aplicaciones de una manera muy rápida y sencilla. Se planteó el uso de un contenedor de Docker que incluía el programa Grgsm ya que la instalación de este es compleja. Grgsm no funciona con las versiones más recientes de gnu-radio y grosmosdr (una librería de gnuradio), esto significa que hay que hacer una instalación personalizada con versiones antiguas de estos programas, en el anexo están las instrucciones para realizar una correcta instalación. Finalmente, Docker fue descartado para no añadir complejidad a la implementación proyecto.



Figura 4.13: Logo de Docker

4.6.2. BBDD

El planteamiento respecto a la BBDD ha sufrido modificaciones, en este punto vamos a ver algunas de las opciones que se plantearon.

- Infoantenas ²: es una web que proporciona el gobierno de España, es un registro nacional de frecuencias en el que se encuentra toda la información, entre otras, relativa a las estaciones de telefonía móvil en todo el territorio nacional. Esta información está representada sobre un mapa dinámico en donde se puede hacer *zoom*, y cuando ya estás en un plano bastante cerrado, muestra las ubicaciones de las BSs. La web tiene un foro donde varios usuarios han preguntado por el acceso a la BBDD y les responden que el acceso es restringido con el fin de garantizar la protección de datos oficiales.

²Web Infoantenas: <https://geoportal.minetur.gob.es/VCTEL/vcne.do>

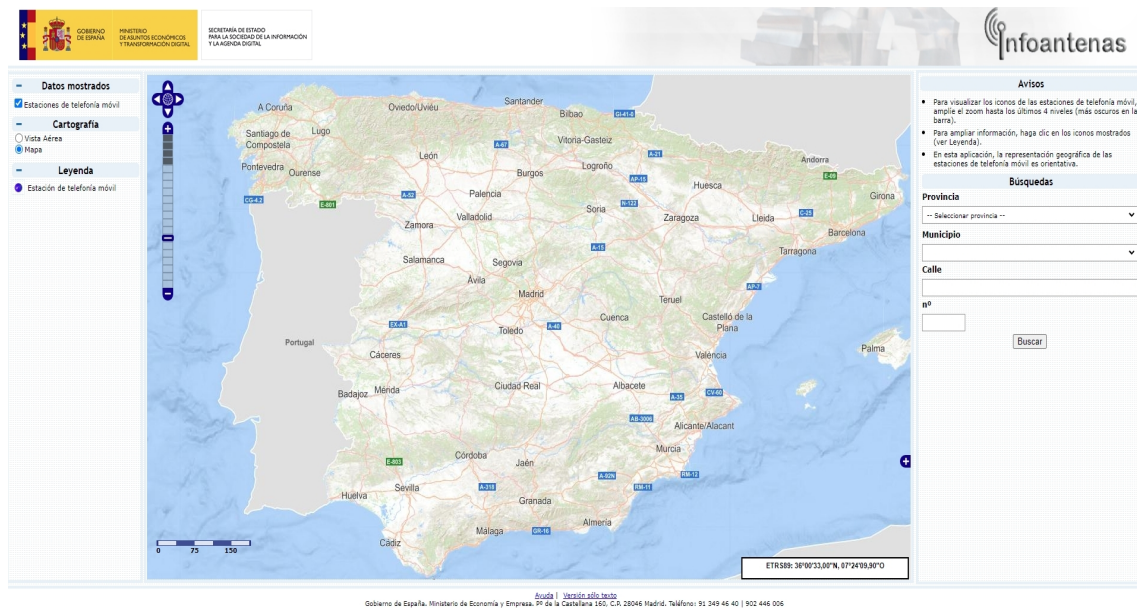


Figura 4.14: Vista InfoAntenas

- **AntenasGSM³**: esta web saca sus datos directamente de la web del gobierno, permite ver la información de cada antena sobre un mapa de Google maps. Los datos de su web son privados. Se le pregunto al administrador por los datos y puso a disposición del proyecto una porción de su BBDD con la información de las antenas de la comunidad de Madrid. En la BBDD no estaban los campos necesarios para poder identificar las BSs registradas por Grgsm, así que fue descartada.

Una vez desechadas estas opciones, se creo una BBDD mediante el uso de apps. Estas apps sirven para la monitorización de las señales GSM y van guardando la información en un registro que posteriormente se puede exportar, se plantearon dos apps.

- **NetMonster**: es una app que recoge, muestra y almacena la información sobre la BS a la que está conectado la MS e intenta aproximar la ubicación de la antena. El problema era conseguir una BBDD con suficientes antenas.
- **NetMonitor**: una app muy similar a NetMonster, con la diferencia de que era capaz de intentar ubicar varias BSs simultáneamente. Tenía una opción que permitía exportar los resultados en un formato que posteriormente se podían importar en Google Earth.

³Web AntenasGSM: <https://antenasgsm.com/>



Figura 4.15: Interfaz NetMonster

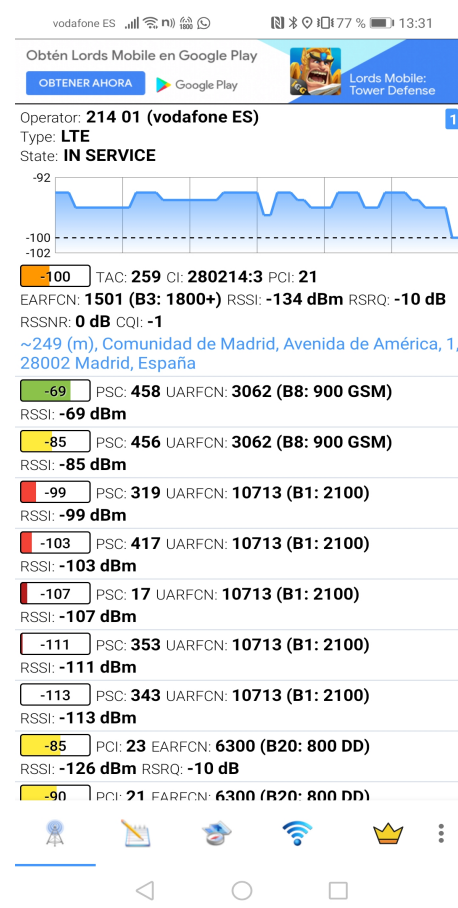


Figura 4.16: Interfaz Net Monitor

Este método era muy lento, así que se exploraron distintas alternativas, más concretamente, buscamos BBDD que acumularan las mediciones de los usuarios y ahí es donde encontramos OpenCelliD. Se compararon los resultados de la BBDD creadas a través de las apps y los datos de OpenCelliD con los datos del Gobierno para ver cuál tenía una mayor fiabilidad.

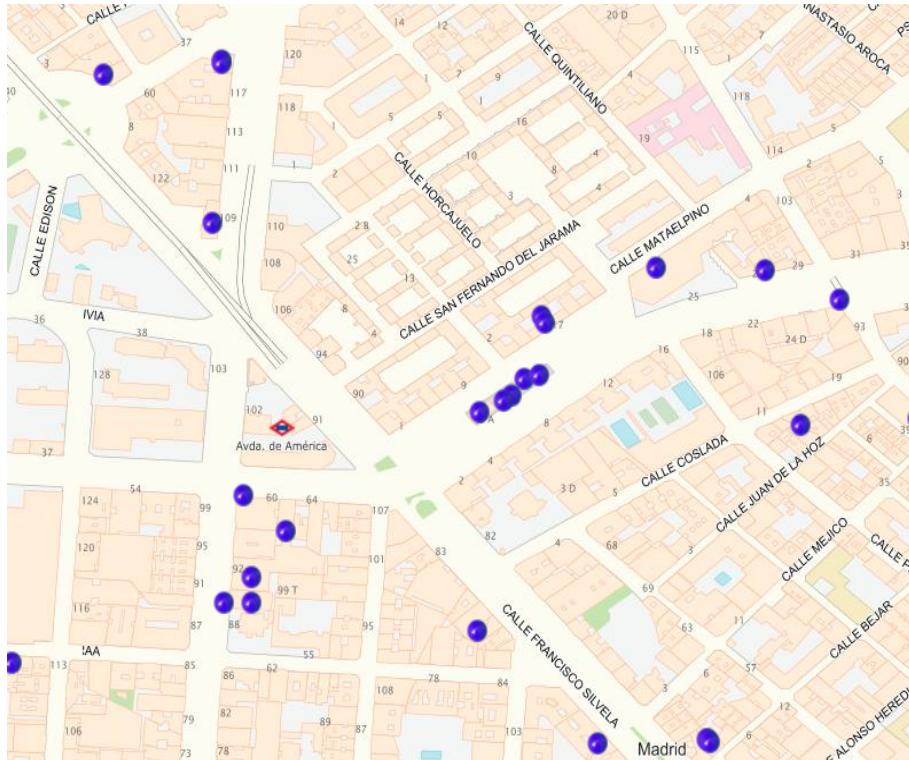


Figura 4.17: Antenas registradas en Infofanten (portal del gobierno)

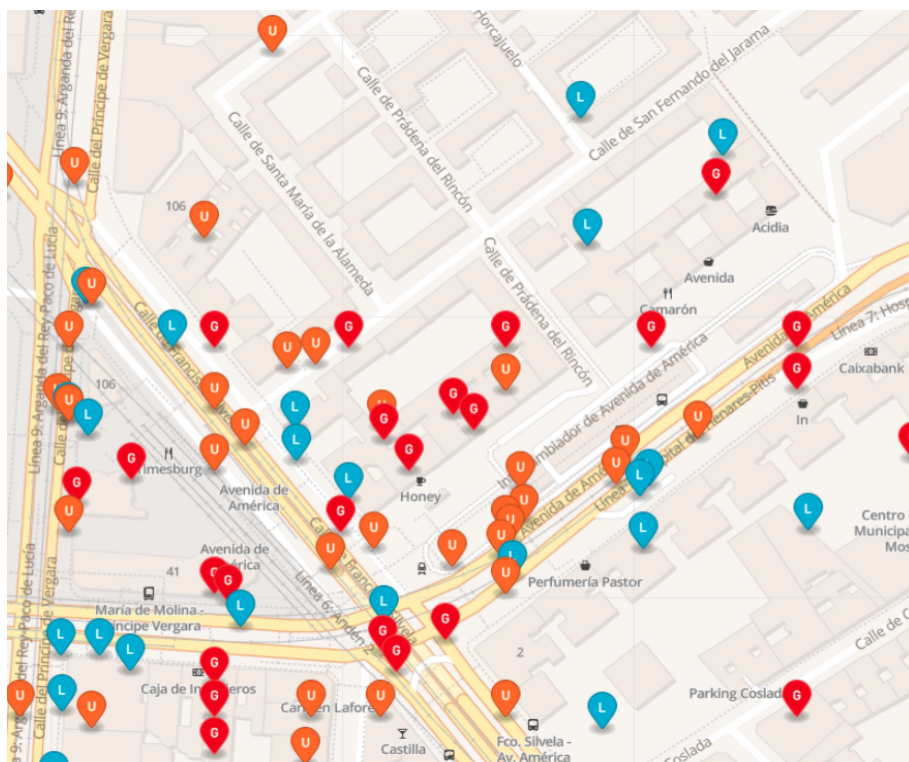


Figura 4.18: Antenas en la BBDD OpenCellID



Figura 4.19: Antenas captadas por Net Monitor

El resultado fue que las BBDDs comunitarias tenían un error menor que las BBDD creadas a partir de las apps.

4.6.3. Node-Js

Node.js es un entorno en tiempo de ejecución de JavaScript. Este entorno de tiempo de ejecución en tiempo real incluye todo lo que se necesita para ejecutar un programa escrito en JavaScript. Utiliza un modelo de entrada y salida sin bloqueo controlado por eventos que lo hacen ligero y eficiente. Con node.js se quería acceder al sistema de ficheros local sin necesidad de hacer una petición al usuario para poder cargar un fichero con los resultados de la ejecución. Se descartó ya que añadía complejidad al necesitar un entorno ejecutándose en tiempo real para poder cargar los ficheros. En su lugar se cambió la salida del programa “ISAC-NAV” para que generase un fichero JavaScript que la web actualizara al recargarse.



Figura 4.20: Node-js Logo

4.6.4. Xleaflet

Es una librería que implementa la API de Leaflet para aplicaciones en C++, esta librería requiere de una instalación previa que añadiría complejidad a la solución.

Capítulo 5

Casos de uso

En este capítulo se muestra como “ISAC-Nav” ajusta el dongle, explicando detalladamente los comandos que ejecuta el *script* para calcular el error del dispositivo y los ficheros intermedios que genera durante el proceso, seguido de dos casos de uso reales del sistema, el primero en un entorno urbano y el segundo en entorno urbano un poco más apartado.

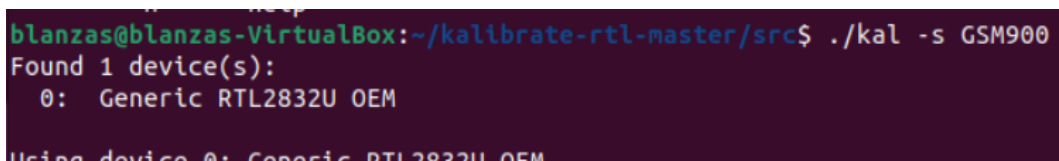
5.1. Ajuste de “ISAC-Nav”

Las antenas suelen tener un error de desviación, este error hay que indicárselo a Grgsm_scanner para que sea más preciso con los resultados. En este apartado se explica cómo “ISAC-Nav” calcula el error con Kalibrate-rtl (explicado en el punto 4.1.2) paso a paso.

- Hay que ejecutar Kalibrate-rtl con el siguiente comando.

```
./kal -s GSM900 > Rastreo.txt
```

- Con este comando, Kalibrate-rtl busca los canales de los que se recibe señal en la banda GSM900, hace un barrido por las frecuencias reservadas para el GSM. Como se puede ver en la siguiente figura, al ejecutar el comando el programa reconoce el dongle RTL_SDR y, en este caso, dice que se puede acceder a los canales 44, 45 y 48 (también te muestra la frecuencia de estos canales). El resultado de ejecutar este comando se vuelca en el fichero “Rastreo.txt” y sirve para tener un control de las bandas a las que tiene acceso.



```
blanzas@blanzas-VirtualBox:~/kalibrate-rtl-master/src$ ./kal -s GSM900
Found 1 device(s):
 0: Generic RTL2832U OEM
Using device 0: Generic RTL2832U OEM
```

Figura 5.1: ./kal -s GSM900

```
GSM-900:
chan: 44 (943.8MHz - 30.565kHz) power: 368745.50
chan: 45 (944.0MHz - 30.535kHz) power: 163338.03
chan: 48 (944.6MHz - 30.599kHz) power: 180718.96
[FC0012] no valid PLL combination found for 948600000 Hz!
Tuning to 948600000 Hz failed!
[FC0012] no valid PLL combination found for 948600000 Hz!
```

Figura 5.2: Canales encontrados con `./kal -s GSM900`

- Si se han encontrado canales, se almacena el primero en una variable local (“Chanel”) y se ejecuta el siguiente comando para conectarse al canal y obtener la desviación del dongle.

```
./kal -c $Chanel > Error.txt
```

- Este comando genera el fichero de salida “Error.txt” donde se almacena el error del dispositivo, este error se almacena en la variable “Error” que es un parámetro de entrada para `Grgsm_scanner`.

```

Error.txt
1 average [min, max] (range, stddev)
2 - 31.553kHz [-31590, -31523] (67, 18.721979)
3 overruns: 0
4 not found: 0
5 average absolute error: 33.739 ppm
6
```

Figura 5.3: Fichero “Error.txt” tras ejecutar `./kal -c $Chanel >Error.txt`

En este caso nos dice que el dispositivo tiene un error de 33 ppm, que es el valor que entrará a `Grgsm_scanner`.

```
grgsm_scanner -b GSM900 -p $Error > Antenas.txt
```

5.2. Caso de uso en un entorno urbano

Tanto en este punto como en el siguiente se van a presentar casos de uso reales de “ISAC-Nav”, en cada caso de uso se han revisado y analizado los ficheros intermedios (menos los ficheros para calcular el error del dispositivo explicados en el punto anterior ya que se ha utilizado el mismo dispositivo), siguiendo el flujo de ejecución del sistema hasta llegar a la representación.

El primer caso de uso es una medición en un entorno urbano. Para poner en marcha el sistema solo hay que ejecutar el siguiente comando.

```
./ISAC-NAV.sh
```

El programa se ejecuta siguiendo el diagrama de flujo representado en la figura 4.10. Después de la calibración empieza ejecutando el `Grgsm_scanner`, muestra el resultado por la consola y vuelca el resultado en el fichero “Antenas.txt”. En esta iteración el programa ha recopilado los siguientes datos.

MCC	MNC	LAC	CID	Pwr
214	3	1150	177	31
214	3	30	1150	31
214	3	1150	704	34
0	0	0	0	44
214	3	1150	703	26
214	3	1150	776	37
214	3	214	2817	29
214	7	2817	6552	53
214	7	2817	0	60

Cuadro 5.1: “Antenas.txt” medidas en Av de América

Una vez que `Grgsm_scanner` ha terminado de ejecutarse, el *script* lanza “ISAC-NAV” que hace uso del fichero “Antenas.txt” que es procesado durante la ejecución y genera el fichero “Antenas.js” como salida, en este ejemplo, ha generado un fichero con estos datos.

Lat	Lon	RSSI
40.438988	-3.680357	26
40.439758	-3.674240	29
40.442245	-3.672851	31
40.442837	-3.675646	31
40.437412	-3.677502	34
40.440979	-3.679962	37
40.439671	-3.679452	53

Cuadro 5.2: Información de las antenas en “Antenas.js”

Lat	Lon
40.440059	-3.677233

Cuadro 5.3: Información del resultado en “Antenas.js”

Se puede observar que la BBDD no ha reconocido dos antenas, si repasamos el contenido del fichero “Antenas.txt” se puede ver rápidamente que una antena tiene varios de sus campos en blanco y otra tiene el campo CID a cero, esto es debido a un error de comunicación entre la BS y el Grgsm_scanner, aun así, va a calcular el centro de masas con siete nodos.

Estos resultados se pueden representar en cualquier mapa, por ejemplo, en Google Earth.



Figura 5.4: Representación en Google Earth, caso 1

En este ejemplo tenemos un error de unos 135m, se puede ver que las antenas están rodeando el punto desde el que se han tomado las medidas y que el centro de masas está prácticamente en el centro del polígono, esto sucede porque todas las antenas tienen un RSSI muy parecido. Ya solo falta ver la representación de los resultados en la web.

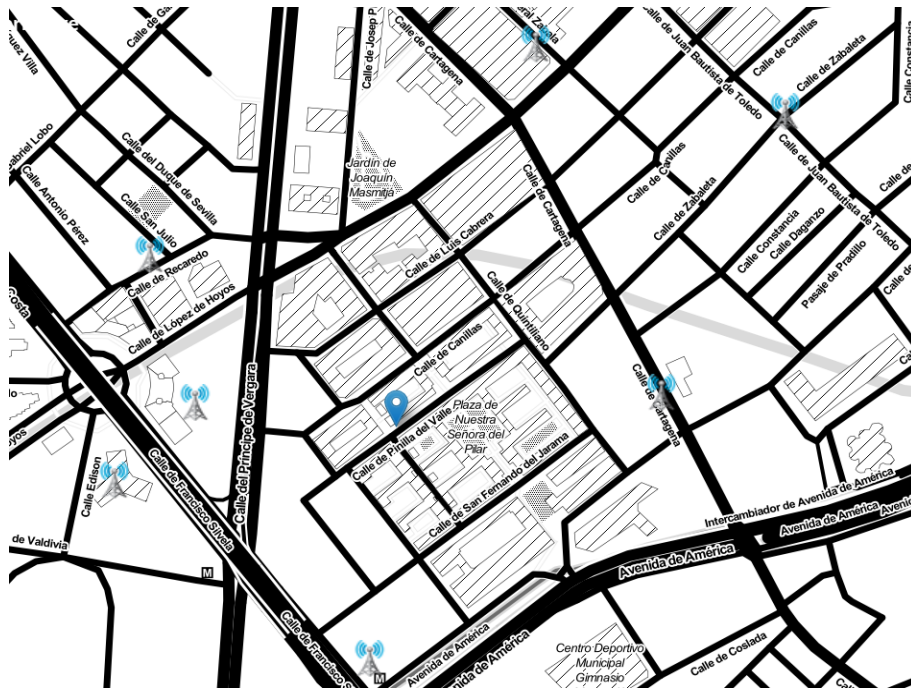


Figura 5.5: Representación en navegador del mapa usando Leaflet, caso 1

Se puede ver que los resultados en Google Earth y los mostrados por Leaflet son muy similares.

Para comparar el resultado de nuestro sistema de posicionamiento con otro similar, la web permite ver la aproximación que hace Google con una API de posicionamiento basado en los datos de navegación.



Figura 5.6: Resultado API Google

Este sistema no es muy preciso, el resultado tiene un margen de error de varios Km.

5.3. Caso de uso en el campus universitario

Este apartado muestra los datos de ejecutar “ISAC-Nav” en la Facultad de Informática de la UCM.

El primer fichero revisado es “Antenas.txt”, el contenido de este es:

MCC	MNC	LAC	CID	Pwr
214	1	13971	9184	46
214	3	1134	1539	50
214	7	2807	1041	51
214	3	1134	1595	56

Cuadro 5.4: “Antenas.txt” medidas en la facultad

En este caso, Grgsm_scanner ha detectado cuatro BSs, se puede observar que los valores del RSSI son un poco más altos que en el caso anterior. Esto se debe a que la facultad está un poco apartada y probablemente esto implique que esté más lejos de las antenas, aun así, hay que recordar que la distancia afecta en gran medida RSSI, aunque, no es el único factor.

El fichero “Antenas.txt” entra al programa “ISAC-Nav” y este genera el fichero “Antenas.js” con los siguientes datos:

Lat	Lon	RSSI
40.456261	-3.735626	46
40.455631	-3.738245	50
40.456722	-3.733978	51
40.456057	-3.736109	52

Cuadro 5.5: Información de las antenas en “Antenas.js”

Lat	Long
40.456057	-3.736109

Cuadro 5.6: Información del resultado en “Antenas.js”

Se puede observar que todas las antenas registradas por Grgsm_scanner están en la BBDD, esto significa que se ha calculado el centro de masas del polígono generado al unir las cuatro BSs, para representar estos datos se vuelve a hacer uso de Google Earth.

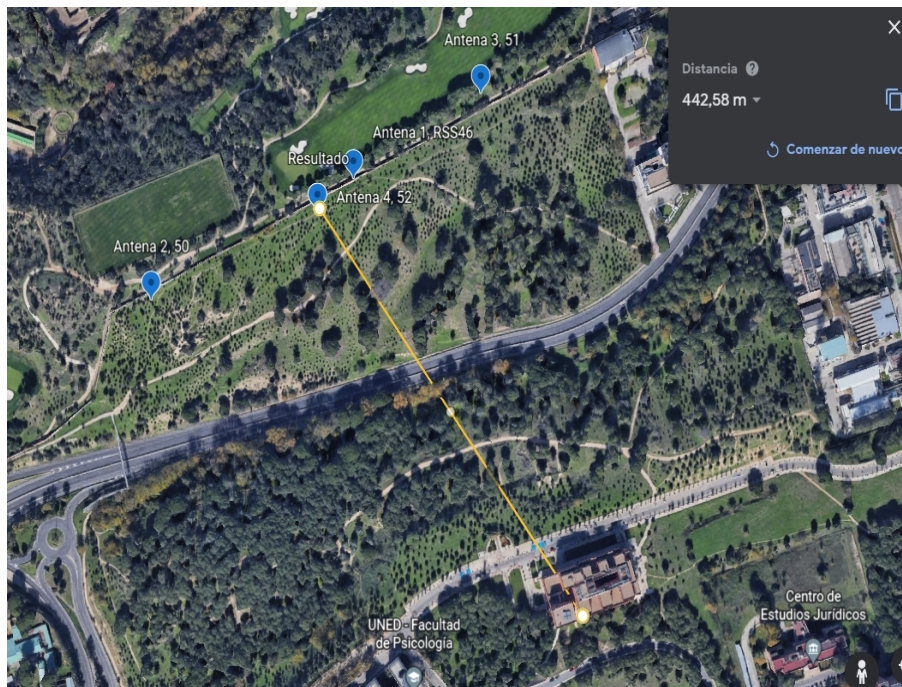


Figura 5.7: Representación en Google Earth, caso 2

En este ejemplo, el sistema tiene un error de unos 442m aproximadamente, para terminar con este caso toca ver la representación.

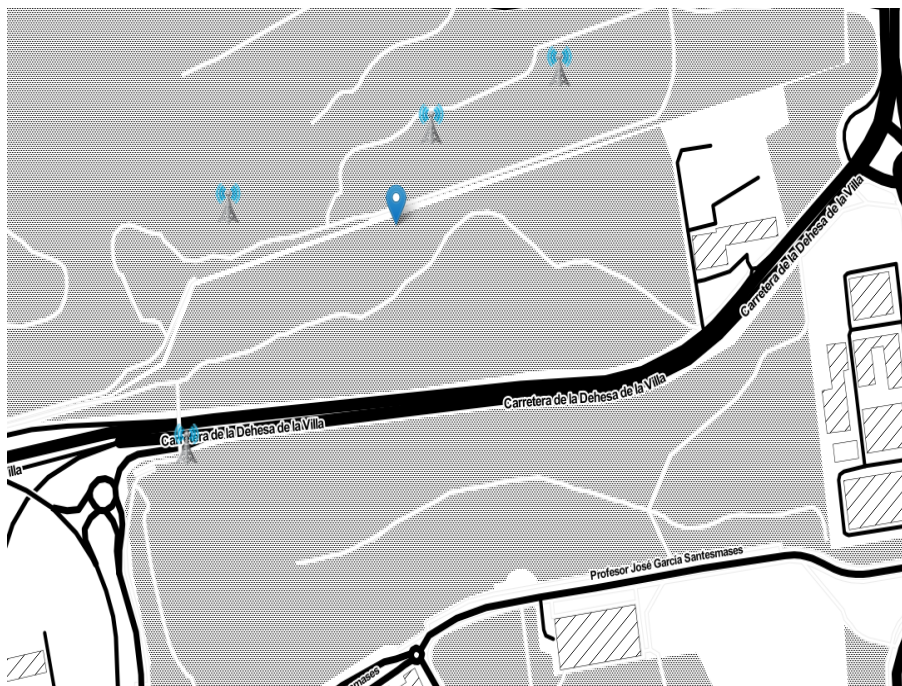


Figura 5.8: Representación en navegador del mapa usando Leaflet, caso 2

Si nos fijamos, los marcadores de las antenas están un poco más abajo de donde deberían estar. Esto se debe a que Leaflet pierde un poco de precisión al poner marcadores personalizados.

Capítulo 6

Estudio “ISAC-Nav”

En este capítulo vamos a analizar la importancia del RSSI y ver la precisión del sistema de posicionamiento creado.

6.1. Análisis RSSI

En la siguiente tabla se pueden ver los resultados de las mediciones de un caso concreto, se ha medido a través de Google Earth la distancia entre cada antena y el punto de captación.

Prueba	Distancia	Pwr
1	471	30
2	200	28
3	306	25
4	256	24
5	153	30
6	556	59

Cuadro 6.1: Mediciones de distancia con su RSSI

Este proceso se ha repetido diez veces y se ha observado que las antenas que están en un rango de 100 a 500 metros, su RSSI oscila entre 20 y 50 dBm, en este rango el valor del RSSI varía mucho, esto se debe a que, en un entorno urbano hay muchos factores que condicionan la señal, como pueden ser los bloques de edificios y que las antenas en el mundo real no son isotrópicas, por lo tanto, no transmiten igual en todas las direcciones.

Cuando la distancia pasa ya de 500 m, el valor empieza a crecer de manera significativa, esto se debe a que la intensidad de la onda va decayendo mientras se propaga (recordar que cuanto más alto es el valor, menor la potencia recibida).

Aplicar el RSSI al algoritmo de centro de masa, ayuda a ajustar el resultado dándole más peso a las antenas más prometedoras, aunque eso signifique que dentro del rango de las antenas más prometedoras no sea muy preciso.

6.2. Análisis de la precisión

Ahora se muestra la exactitud obtenida en una simulación de diez iteraciones de “ISAC-Nav”. Se ha considerado relevante almacenar el número de antenas registradas en cada medición que se han identificado en la BBDD y se ha usado Google Earth para calcular la distancia aproximada (medida en metros) de cada iteración con la posición real (como en el capítulo anterior). Los resultados obtenidos son:

Prueba	Antenas	Distancia
1	8	93
2	4	467
3	4	321
4	6	200
5	1	-
6	7	60
7	4	533
8	8	124
9	6	179
10	5	266

Cuadro 6.2: Resultado de las mediciones

En una ocasión, el sistema no es capaz de detectar antenas, este caso queda descartado.

El cálculo del error medio es:

$$\bar{X} = \frac{93 + 467 + 321 + 350 + 231 + 533 + 124 + 179 + 266}{9} = 271,1m \quad (6.1)$$

En la siguiente tabla se observa que la precisión del sistema creado en comparación con sistemas de posicionamiento.

Nombre	Error
GPS	<10 m
ISAC-Nav	270 m
API Google	1556 m

Cuadro 6.3: Comparación de sistemas de posicionamiento

Si repasamos los datos, es interesante ver que cuando aumenta el número de antenas se suele tener una mayor precisión, esto es debido a que se realizan más ajustes en la medición y los errores se van corrigiendo.

Capítulo 7

Conclusiones

7.1. Conclusiones

Como se nombra al comienzo de esta memoria, el objetivo de este proyecto era crear un sistema de posicionamiento basado en estándares GSM. El requisito principal era crear un sistema funcional que fuera viable en una situación real y para ello impusimos que este sistema tenía que ir actualizándose de forma periódica con la información captada en tiempo real por el usuario.

Para la implementación se decidió crear un sistema modular que fuera capaz de detectar las BSs, operar sobre los datos obtenidos y ser capaz de dar una representación visual.

Una vez que se consiguió un prototipo funcional se realizaron pruebas y ajustes del sistema. Se optó por la implementación de un algoritmo basado en la potencia de la señal recibida.

El RSSI no es muy preciso para medir la distancia a un punto concreto en un entorno urbano, donde la señal tiene muchas interferencias que atenúan la onda, sin embargo, al tener una densidad alta de BSs, tiene bastantes puntos de referencia y el error puntual generado por una antena se va contrarrestando.

El algoritmo implementado es una optimización del centroide, este prioriza las antenas con un buen valor de RSSI y discrimina a las antenas con un mal valor (estas suelen estar más alejadas). Este algoritmo forma un polígono donde los vértices son todas las BS que es capaz de localizar en el momento e intentar calcular su centro de masa, por tanto, para tener una buena medición es necesario detectar antenas en todas las direcciones.

Si solo se detectan antenas que están en una dirección, el cálculo ya tendrá un error inicial grande, puesto que, la ubicación real ya está fuera del espacio de las posibles soluciones.

Para formar el polígono se ha fijado que es necesario captar un mínimo de dos antenas.

“ISAC-Nav” detecta las antenas y necesita una BBDD con la posición de las BSs, hay una BBDD oficial del gobierno para esto, pero no es de uso público. La alternativa ha

sido implementar una BBDD comunitaria con la ubicación aproximada de las BSs, esta ubicación está calculada basándose en las mediciones aportadas por los usuarios.

Al no tener el conocimiento exacto de la ubicación de las BSs, todas las medidas parten de un error inicial. Es importante tener una BBDD fiable, puesto que ayuda a reducir el error del sistema.

Se han realizado múltiples mediciones de “ISAC-Nav” para estimar el error del sistema en un entorno urbano real y se ha comparado con el de otros sistemas. De la comparativa se deduce que no es tan preciso como un GPS pero puede ser más fiable que otros sistemas de posicionamiento basados en los datos de navegación.

Pasando a un plano general, los puntos fuertes de “ISAC-Nav” son:

- Ejecución local: esta es la característica más diferencial de “ISAC-Nav”, no necesita tener acceso a internet para estimar la ubicación.
- Es económico: si se cuenta con un dispositivo que pueda correr alguna partición de Linux (en este caso el proyecto se ha implementado sobre Ubuntu 20.04), solo requiere de un dongle, el que se ha usado en este proyecto tiene un coste inferior a 15€.
- Cobertura: este sistema solo necesita estar en el rango de dos estaciones bases, por lo tanto, puede funcionar en lugares donde no llega la cobertura satelital.
- Modularidad: es sencillo implementar cambios en la arquitectura del sistema.
- Bajo consumo: solo necesita alimentar el dongle a través del puerto USB.

Las mayores debilidades de “ISAC-Nav” son:

- El tiempo de carga de Grgsm: el programa tarda 1’ 30” aproximadamente en ejecutarse y la página web se va actualizando cada 40s, esto hace que tarde unos 2 minutos en obtener cada medida, a esto, hay que sumarle la calibración del dispositivo que también tarda un par de minutos.
- Necesidad de *hardware* externo: para la ejecución de “ISAC-Nav” es necesario contar con un dispositivo compatible con las aplicaciones SDR.
- La precisión está condicionada: para la obtención de una buena medida es necesario estar en un entorno con varias BSs, además estas tienen que estar en varias direcciones, para que se genere un polígono que rodee al dispositivo. Funciona mejor en entornos urbanos y de noche.

“ISAC-Nav”, es una opción viable para ser implementado como sistema de posicionamiento de un dispositivo, por esto, se considera que se ha alcanzado el objetivo del proyecto.

Capítulo 8

Conclusions

8.1. Conclusions

As mentioned at the beginning of this memory, the objective of this project was to create a positioning system based on GSM standards. The main requirement was to create a functional system that would be viable in a real situation, and for this purpose we imposed that this system had to be updated periodically with the information captured in real time by the user.

For the implementation it was decided to create a modular system that would be able to detect the BSs, operate on the data obtained and be able to give a visual representation.

Once we had a working prototype we started testing and adjusting the system. We chose to implement an algorithm based on the power of the received signal.

On the one hand, RSSI is not very accurate for measuring the distance to a specific remotely located point in an urban environment, where the signal has many interferences that attenuate the signal. However, by having a high density of BSs, we have enough reference points and the point error generated by one BS is contrasted.

The implemented algorithm is a centroid optimization, which prioritizes antennas with a good RSSI value and discriminates antennas with a high value (these, by definition, have a high RSSI). This algorithm forms a polygon where the vertices are all the BSs that it is able to locate and try to calculate their center of mass, therefore, to have a good measurement it is necessary to detect antennas in all directions.

If only antennas that are in one direction are detected, the calculation will already have a large initial error since the actual location is already outside the possible solution space.

To form the polygon, it has been established that we need to capture a minimum of 2 antennas.

“ISAC-Nav” detects the antennas and needs a BBDD with the position of the BSs, there is an official government BBDD for this, but it is not for public use, the alternative has been to implement a community BBDDD with the approximate location of the BSs. This

location is calculated based on the measurements provided by the users.

Since we do not know the exact location of the BSs, all measurements have an initial error. It is important to have a reliable BBDD, a good BBDD reduces the error of the system.

Multiple measurements of “ISAC-Nav” have been made to estimate the system error in a real urban environment and compared with that of other systems. In the comparison with other systems, it is not as accurate as a GPS but its accuracy can be even better than some other positioning systems that are based on navigation data.

Moving on to a general level, the strongest of “ISAC-Nav” points are:

- Local execution: this is the most differential feature of “ISAC-Nav”, you do not need to internet access to estimate the location.
- It is economical: if you have a device that can run a Linux partition (in this case the project has been implemented on Ubuntu 20.04), it only requires the dongle, the one used in this project has a cost of less than 15€.
- Coverage: this system requires to be in range of the base stations, therefore, it can work in places where satellite coverage does not reach.
- Modularity: it is easy to implement changes in the system.
- Low power consumption: only one dongle needs to be powered through the USB port.

The major weaknesses of “ISAC-Nav” are:

- The loading time of Grgsm: the program takes about 1’ 30”to run, and the web page is updated every 40 seconds, this makes it take about 2 minutes to obtain each measurement, to this must be added the calibration of the device that also takes a couple of minutes.
- Need for external hardware: for the execution of “ISAC-Nav” it is necessary to have a device compatible with the SDR applications.
- Accuracy is conditioned: to obtain a good measurement it is necessary to be in an environment with several BSs, moreover these have to be in several directions, so that the polygon surrounds the device. It works best in urban environments and at night.

“ISAC-Nav” is a viable option to implement a positioning system in a device. Therefore, it is considered that the objective of the project has been achieved.

Capítulo 9

Trabajos futuros

El proyecto “ISAC-NAV” se considera finalizado, aun así, se sopesa que puede servir como punto de partida para futuros nuevos proyectos e iniciativas. Se van a referir una serie proyectos interesantes que se podrán desarrollar y plantear cómo continuación de este.

- Hacer un estudio de la precisión en entorno urbano y rural de varios algoritmos. En este proyecto se ha implementado un algoritmo basado en el RSSI, durante la realización de este, se ha visto como el valor de esta variable variaba en función del entorno, no solo por la distancia, sino también por la interferencia de los edificios. Sería interesante realizar un estudio para analizar más detalladamente el comportamiento de esta variable.
- Otra propuesta de proyecto podría ser, hacer un sistema de posicionamiento aplicando distintos métodos definidos en el estado del arte que utilicen la ubicación de las EBs. También es interesante ver cómo varía el resultado en función del algoritmo aplicado, por ejemplo, ver qué centro del triángulo tiene más precisión (incentro, baricentro o el ortocentro). Otros algoritmos más complejos también son aplicables, como los de *clustering*, es interesante ver el coste que suponen para el sistema frente a la mejora de rendimiento que aportan.
- Y como última propuesta, adaptar “ISAC-NAV” para que trabaje con la infraestructura de la red 5G. Esta red es la más rápida del mercado y, aunque todavía no está muy extendida, su baja latencia en el envío de mensajes (1ms aproximadamente) podría mejorar la velocidad del sistema. Este sistema utilizaría una red distinta y por lo tanto, de igual forma que en los proyectos nombrados anteriormente, se podría ver cómo afecta esto al resultado.

GLOSARIO

AGPS, *Assisted Global Positioning System*
AI, *Artificial Intelligence*
AOA, *Angle Of Arrival*
API, *Application Programming Interface*
BS, *Base Station*
BSC, *Base Station Controller*
BSS, *Base Station Subsystem*
BTS, *Base Transceiver System*
COO, *Cell Of Origin*
COO CS, *Cell Of Origin with Cell Sector*
EDGE, *Enhanced Data Rates for GSM Evolution*
EOTD, *Enhanced Observed Time Difference*
ETSI, *European Telecommunications Standards Institute*
FCC, *Federal Communications Commission*
GSM, *Global System for Mobile*
GPS, *Global Positioning System*
GPRS, *General Purpose Radio Service*
HSDPA, *High Speed Downlink Packet Access*
LTE, *Long Term Evolution*
MCC, *Mobile Country Code*
MNC, *Mobile Network Code*
MSC, *Mobile Switching Center*
NSS, *Network Switching Subsystem*
OSM, *Open Street Map*
OSS, *Operation Support Subsystem*

PPM, *Parts Per Million*

RSSI, *Received Signal Strength Indicator*

RTL, *Real Time Logic*

RToF, *Round Trip of Flight*

SDR, *Software Definition Radio*

TA, *timing advance*

TDOA, *time difference of arrival*

UMTS, *Universal Mobile Telecommunications System*

Índice de figuras

1.1. Dongle RTL_SDR	3
1.2. Diagrama de Gantt de “ISAC-Nav”	4
2.1. Dongle RTL_SDR	7
2.2. “ISAC-Nav” Gantt Chart	8
3.1. Radioteléfono. Fuente: https://es.wikipedia.org/wiki/SCR-300	9
3.2. Estación base. Fuente: https://www.rtve.es/noticias/20120227/estacion-base-telefonía-movil-chip/501966.shtml	10
3.3. Sistema GPS	15
3.4. COO	16
3.5. COO CS	16
3.6. COO CS with RSSI	17
3.7. TA	17
3.8. AOA	18
3.9. RTof	18
3.10. TDOA	19
3.11. EOTD	19
3.12. Esquema AGPS	20
3.13. Red GSM	22
3.14. Centroides de un polígono	23
3.15. Centro de masas de un polígono	24
4.1. Esquema de la comunicación entre los módulos	25
4.2. Diagrama de bloques conceptual de un transceptor SDR ideal. Fuente: https://es.wikipedia.org/wiki/Radio_definida_por_software	26
4.3. Parámetros de entrada de Kal	27
4.4. GNU-Radio logo	27
4.5. GQRX logo	28
4.6. OpenCellID logo	29
4.7. Flujo de ejecución de “ISAC-Nav”	31
4.8. Diagrama Makefile	32
4.9. Open Street Map logo	32
4.10. Leaflet logo	33
4.11. Diagrama de flujo del <i>Script</i>	34
4.12. Jerarquía sistema de ficheros	34
4.13. Logo de Docker	35
4.14. Vista InfoAntenas	36

4.15. Interfaz NetMonster	37
4.16. Interfaz Net Monitor	37
4.17. Antenas registradas en Infoantenas (portal del gobierno)	38
4.18. Antenas en la BBDD OpenCellID	38
4.19. Antenas captadas por Net Monitor	39
4.20. Node-js Logo	39
5.1. ./kal -s GSM900	41
5.2. Canales encontrados con ./kal -s GSM900	42
5.3. Fichero "Error.txt" tras ejecutar ./kal -c \$Chanel >Error.txt	42
5.4. Representación en Google Earth, caso 1	44
5.5. Representación en navegador del mapa usando Leaflet, caso 1	45
5.6. Resultado API Google	45
5.7. Representación en Google Earth, caso 2	47
5.8. Representación en navegador del mapa usando Leaflet, caso 2	47

Índice de cuadros

3.1.	Tabla RSSI	12
3.2.	Tabla dBm	12
3.3.	Evolución de la velocidad de transmisión en las distintas generaciones . .	14
3.4.	Precisión de los sistemas basados en GSM	20
3.5.	Frecuencias GSM	21
4.1.	Parámetros “Antenas.txt”	28
4.2.	Parámetros usados de “BBDD.csv”	30
5.1.	“Antenas.txt” medidas en Av de América	43
5.2.	Información de las antenas en “Antenas.js”	43
5.3.	Información del resultado en “Antenas.js”	44
5.4.	“Antenas.txt” medidas en la facultad	46
5.5.	Información de las antenas en “Antenas.js”	46
5.6.	Información del resultado en “Antenas.js”	46
6.1.	Mediciones de distancia con su RRSI	48
6.2.	Resultado de las mediciones	49
6.3.	Comparación de sistemas de posicionamiento	49

Bibliografía

- [1] L. Descamps-Vila, A. Pérez Navarro, and J. Conesa Caralt, “Integración de un sistema de posicionamiento indoor en aplicaciones sig para dispositivo móvil,” 2013.
- [2] J. Joskowicz, “Breve historia de las telecomunicaciones,” *Instituto de Ingeniería Eléctrica de la republica de Uruguay*, pp. 43–46, 2013.
- [3] J. R. B. Gomez, “Evolucion de las redes móviles hasta hoy en día y el impacto de la red móvil de quinta generación,” *Revista Digital de Tecnologías Informáticas y Sistemas*, vol. 3, no. 3, 2019.
- [4] C. Del Valle, F. J. Moreno, and F. Sierra, *Políticas de comunicación y ciudadanía cultural iberoamericana*, vol. 38. Editorial Gedisa, 2012.
- [5] Y. Zhou and B. Ai, “Handover schemes and algorithms of high-speed mobile environment: A survey,” *Computer Communications*, vol. 47, pp. 1–15, 2014.
- [6] A. El-Rabbany, *Introduction to GPS: the global positioning system*. Artech house, 2002.
- [7] M. Ibrahim and M. Youssef, “Cellsense: An accurate energy-efficient gsm positioning system,” *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 286–296, 2011.
- [8] G. M. Djuknic and R. E. Richton, “Geolocation and assisted gps,” *Computer*, vol. 34, no. 2, pp. 123–125, 2001.
- [9] S. Devra and A. Sharma, “Gsm architecture & channels: review study,” in *Second international conference on innovative trends in electronics engineering (ICITEE2)*, 2016.
- [10] I. de Economía Digital, “Geolocalización, Smartphones y cuestiones legales aún por definir.” <https://www.esic.edu/rethink/marketing-y-comunicacion/geolocalizacion-smartphones-y-cuestiones-legales-aun-por-definir>, Octubre 2013.
- [11] F. Perez, “Redes móviles terrestres: 4g,” *Esc. Técnica Super. Ing. Univ. Pontif. Comillas*, pp. 1–12, 2010.
- [12] C. Drane, M. Macnaughtan, and C. Scott, “Positioning gsm telephones,” *IEEE Communications magazine*, vol. 36, no. 4, pp. 46–54, 1998.
- [13] W. Tomasi, *Sistemas de comunicaciones electrónicas*. Pearson educación, 2003.

Anexo

Este Anexo es un manual para la correcta instalación y configuración local de “ISAC-Nav” en Linux, en este proyecto hemos usado Ubuntu 20.04.

9.1. instalación

Para correr “ISAC-Nav” es necesario tener instalado gnuradio, Grgsm y Kalibrate-rtl.

9.1.1. Gnuradio

Para que Grgsm funcione, es necesario instalar la versión 3.8 de gnuradio, si se actualiza a una versión más moderna no funcionará, por lo tanto, hay que hacer una instalación manual de este programa y de las librerías que utiliza, se recomienda seguir las instrucciones de este tutorial para instalarlo correctamente <https://wiki.gnuradio.org/index.php/InstallingGR>.

9.1.2. Grgsm

Antes de instalar Grgsm, hay que instalar dos librerías (gr-osmosdr libosmocore) de forma manual, en versiones compatibles con gnuradio 3.8, para ello, hay que instalar la rama gr3.8 de gr-osmosdr <https://github.com/osmocom/gr-osmosdr> y los archivos de cabecera de libosmocore. Los comandos para instalar Grgsm se pueden encontrar en esta guía: <https://osmocom.org/projects/gr-gsm/wiki/Installation>.

9.1.3. Kalibrate-rtl

Kalibrate-rtl se puede descargar directamente del github del proyecto <https://github.com/steve-m/kalibrate-rtl>.

9.2. Configuración

Antes de configurar el script es necesario tener el código, se puede descargar a través de mi Github: <https://github.com/PabloSaro> y para poder ejecutar el *script* “ISAC-Nav.sh”,

es necesario configurar las direcciones ajustándolas al nuevo SO.

Pablo Saro Buendía

2021-2022

Ult. actualización 11 de septiembre de 2022

L^AT_EX lic. LPPL & powered by **TEFLON** CC-ZERO

Este documento esta realizado bajo licencia Creative Commons “CC0 1.0 Universal”.

