

Universidad Tecnológica Nacional

Proyecto Final

Gestión de redes satelitales para estaciones
remotas móviles

Autor:

- Santiago Fabián Miño

Director:

*Proyecto final presentado para cumplimentar los requisitos académicos
para acceder al título de Ingeniero Electrónico*

en la

Facultad Regional Paraná

Fecha: octubre de 2018

Declaración de autoría:

Yo declaro que el Proyecto Final “Gestión de redes satelitales para estaciones remotas móviles” y el trabajo realizado son propios.

Declaro:

- Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero Electrónico, en la Universidad Tecnológica Nacional, Regional Paraná.
- Se establece claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pre-grado.
- Siempre que se ha utilizado trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
- Se ha indicado y agradecido correctamente a todos aquellos que han colaborado con el presente trabajo.
- Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, se ha indicado claramente el alcance del trabajo realizado.

Firma:

•

Fecha:

Agradecimientos:

Especialmente a mis padres, a mi hermana y a Florencia por su apoyo y contención a lo largo de esta larga carrera. A todos los profesores, compañeros y amigos ligados al ambiente universitario que tuve la dicha de conocer en todo este tiempo. Por último, pero no menos importante, a la universidad pública, con el deseo cabal de que continúe siempre con su carácter abierto, inclusivo y extensivo a toda la población.

Santiago F. Miño

Universidad Tecnológica Nacional

Abstract

Facultad Regional Paraná

Ingeniero en Electrónica

Gestión de redes satelitales para estaciones remotas móviles

Santiago Fabián Miño

Abstract:

In order to control and evaluate some of the different parameters that intervene in any satellite network and, in this way, to be able to determine which is the best network for a particular remote station, we have developed a program through which we can know and work with those parameters in a centralized way. The program must be installed in the workstation of the technicians or operators and it will communicate with a database that contains information of both the networks and the remote mobile stations.

It was developed in Python language and interacts with the database in a MySQL environment. From there it obtains, among other things, information such as geographical position, hardware elements and contracted bandwidth with which it will perform the necessary calculations to finally get the information that will be shown to the user through a graphical interface that works both in Windows platforms such as Linux.

We managed to develop a program that allows the user to interact with the database and allows both the visualization of up to four satellite parameters and the simulation of different scenarios that may arise

Keywords:

Blockage chart, databases, link budget, Python, satellite footprints, subscription.

Resumen:

Con el objeto de controlar y evaluar algunos de los diferentes parámetros que intervienen en toda red satelital y, de esta manera, poder determinar cuál es la mejor red para una estación remota en particular, hemos desarrollado un programa mediante el cual podemos conocer y trabajar con esos parámetros en forma centralizada. El programa debe ser instalado en la estación de trabajo de los técnicos u operadores y se comunicará con una base de datos que contiene información tanto de las redes como de las estaciones remotas móviles.

El mismo fue desarrollado en lenguaje Python e interactúa con la base de datos en un entorno MySQL. Desde allí obtiene, entre otras cosas, información tal como la posición geográfica, elementos del hardware y ancho de banda contratado con la que realizará los cálculos necesarios para finalmente conseguir la información que será mostrada al usuario a través de una interfaz gráfica que funciona tanto en plataformas Windows como Linux.

Conseguimos desarrollar un programa que posibilita la interacción del usuario con la base de datos y permite tanto la visualización de hasta cuatro parámetros satelitales, como la simulación de diferentes escenarios que puedan presentarse.

Palabras Clave:

Bases de datos, cálculo de enlace, carta de bloqueo, huella satelital, Python, suscripción.

Reconocimientos:

Quisiera reconocer la colaboración de aquellas personas, muchas veces anónimas, a las que no puedo agradecerles personalmente pero que de una u otra manera, aunque siempre de forma desinteresada, comparten sus conocimientos y experiencias a través de la aldea global y de las herramientas que de ella nacen. Sin su ayuda, sin dudas la realización de este trabajo hubiese sido menos enriquecedora y llevadera.

Índice:

1. Introducción.....	1
2. Desarrollo.....	3
2.1. Conceptos preliminares.....	3
2.1.1 Redes satelitales.....	3
2.1.2 Suscripción. CIR y MIR.....	5
2.1.3 Coordenadas polares. Elevación, azimut y ángulo relativo.....	6
2.2 Desarrollo.....	6
2.2.1 Base de datos.....	7
2.2.2 Aplicación principal. Contenido y prestaciones.....	10
I. Posición geográfica.....	11
II. Capacidad disponible.....	15
III. Potencia.....	18
IV. Bloqueo.....	26
2.2.3 Interfaz con el usuario.....	29
2.3 Consideraciones acerca de la implementación.....	38
2.3.1 Consideraciones sobre la base de datos.....	38
2.3.2 Consideraciones sobre la aplicación principal.....	39
3. Resultados.....	41
4. Análisis de Costos.....	42
5. Discusión y Conclusión.....	44
6. Literatura citada.....	45

Lista de Figuras:

Figura 1.1: Antena parabólica fija	1
Figura 1.2: Ejemplos de antenas parabólicas móviles. En el caso de la imagen de la izquierda, las antenas se encuentran dentro de los domos (radomos) para protegerlas de las inclemencias climáticas.....	1
Figura 2.1: Esquema básico de un enlace satelital.....	3
Figura 2.2: Valores de PIRE para el satélite SES-4 (Haz para el Cono Sur en banda Ku). La unidad de cada valor es el Decibel Watt [dBW]	4
Figura 2.3: Configuración básica de una antena parabólica terrestre.....	5
Figura 2.4: Coordenadas polares. Elevación y Azimut.....	6
Figura 2.5: Esquema general de todo el sistema.....	7
Figura 2.6: Esquema básico de una base de datos.....	8
Figura 2.7: Servidor de base de datos alojado en Raspberry Pi.....	8
Figura 2.8: Esquema de la Aplicación principal.....	11
Figura 2.9: Coberturas o “footprints” de algunos satélites. Los puntos indican las estaciones remotas; cómo se observa, cada una puede estar cubierto por uno o más satélites.....	12
Figura 2.10: Esquema del bloque Posición geográfica.....	13
Figura 2.11: Diagrama de flujo del bloque Posición geográfica.....	14
Figura 2.12: Captura de pantalla para el ejemplo de una remota ubicada en Caracas, Venezuela.....	15
Figura 2.13: Esquema del bloque Capacidad.....	16
Figura 2.14: Diagrama de flujo del bloque Capacidad.....	17
Figura 2.15: Elementos presentes en un enlace satelital típico.....	19
Figura 2.16: Ejemplo de huella satelital.....	20
Figura 2.17: Zonas de cobertura en función del PIRE.....	21
Figura 2.18: Esquema del bloque Potencia.....	24
Figura 2.19: Diagrama de flujo del bloque Potencia.....	25

Figura 2.20: Ejemplo de Carta de Bloqueo.....	26
Figura 2.21: Esquema del bloque Bloqueo.....	27
Figura 2.22: Diagrama de flujo del bloque Bloqueo.....	28
Figura 2.23: Captura de pantalla de la ventana principal.....	29
Figura 2.24: Información mostrada para el ejemplo de la Remota 10.....	30
Figura 2.25: Ubicación del punto de intersección de los ángulos de elevación y azimut relativo para el ejemplo de la Remota 10.....	31
Figura 2.26: Conjunto de botones para acceder a nuevas ventanas.....	31
Figura 2.27: Ventana para el cálculo de enlace.....	32
Figura 2.28: Remota dentro de Zona 1.....	33
Figura 2.29: Remota dentro de Zona 2.....	33
Figura 2.30: Remota fuera de la zona de cobertura.....	33
Figura 2.31: Remota con excelente recepción, pero potencia de BUC insuficiente.....	34
Figura 2.32: Remota con excelente recepción y potencia de BUC adecuada.....	35
Figura 2.33: Ventana Capacidad.....	36
Figura 2.34: Ejemplo de simulación para la Remota 10.....	37
Figura 2.35: Resultado de la simulación de la Remota 10.....	38
Figura 2.36: Esquema general del desarrollo completo.....	38
Figura 2.37: Fotografía de la Raspberry Pi utilizada en este trabajo.....	39
Figura 2.38: Contenido de la carpeta de la aplicación para la plataforma Windows.....	40
Figura 2.39: Marcador de Google Maps.....	40

Lista de Tablas

Tabla 2.1: Tabla "remotas"	9
Tabla 2.2: Tabla "redes"	10
Tabla 2.3: Tamaños mínimos y máximos del reflector en función del PIRE del satélite	21
Tabla 4.1: Análisis de Costos	43

Lista de Abreviaciones

API: Application Programming Interface (Interfaz de Programación de Aplicaciones)

ARS: Código internacional para identificar al peso argentino

BUC: Block-Up Converter

CIR: Committed Information Rate

DBMS: Data Base Management System

FEC: Forward Error Correction

GEO: Geostationary Orbit

IBO: Input Back Off

LEO: Low Earth Orbit

LNB: Low Noise Block

MEO: Medium Earth Orbit

MIR: Maximum Information Rate

MODCOD: Acrónimo de Modulation y Coding

OBO: Output Back off

PIRE: Potencia Isotrópica Radiada Efectiva

SBC: Single Board Computer

TDMA: Time Division Multiple Access

Lista de Símbolos

C/N: Relación Señal a Ruido

C/N₀: Relación Señal a Densidad de Ruido

dBW: Decibel Watt

G/T: Factor de mérito

Mbps: Mega bits por segundo

Dedicado a:

Santiago Maffei, quien, de estar aún en este mundo, sería sin dudas un gran ingeniero.

Capítulo 1: Introducción

Con cierta frecuencia solemos encontrar por la ciudad diversos tipos de antenas parabólicas similares a las de la Figura 1.1. Este tipo de antenas se encuentran apuntando a un satélite geoestacionario y, salvo que el viento u otro evento desafortunado actúe sobre ellas, no cambiarán esa posición; decimos entonces que son antenas fijas y que, junto a un determinado y único satélite, forman parte de una *red satelital*.



Figura 1.1: Antena parabólica fija. Recuperado de <https://techlandia.com/>

Por otro lado, también existen antenas que deben cambiar de satélite (y por lo tanto de red) en determinado momento dependiendo de su ubicación geográfica y/o de la orientación del objeto sobre la que estén montadas; la Figura 1.2 muestra algunos ejemplos de este último caso: antenas parabólicas montadas sobre embarcaciones o camionetas como las utilizadas para transmitir eventos deportivos o noticias en tiempo real.



Figura 1.2: Ejemplos de antenas parabólicas móviles. En el caso de la imagen de la izquierda, las antenas se encuentran dentro de los domos (radomos) para protegerlas de las inclemencias climáticas. Recuperado de: <https://es.123rf.com/>

La conexión a uno u otro satélite (en el caso de que haya más de uno) requiere de la evaluación de múltiples variables y ciertas condiciones deben cumplirse para que determinada antena pueda conectarse o mantenerse en una red. Esas condiciones son las que analizaremos y estudiaremos a lo largo del presente trabajo.

Actualmente, la asignación de una antena (que de ahora en más llamaremos *estación remota* o solo *remota*) a un satélite (o red) es realizada por una o varias personas que, analizando todas esas condiciones, son las encargadas de mantener la red en buenas condiciones. El operador encargado de decidir cuál es la mejor red para una estación remota debe trabajar con muchas redes y antenas simultáneamente, analizando todas esas variables por separado, lo que dificulta su trabajo y reduce tanto su productividad como las probabilidades de tomar una decisión acertada.

El sistema presentado en este trabajo, es capaz de evaluar rápida y eficazmente las condiciones antes mencionadas para finalmente determinar, entre otras cosas, cuál es la mejor red a utilizar, minimizando así pérdidas de tiempo y evitando omitir algunos detalles por parte del operador quien será, en última instancia, quien decida si la red sugerida por el sistema es adecuada o no.

La interfaz con el usuario será una aplicación que el operador tendrá instalada en su estación de trabajo (que puede ser una PC de escritorio o laptop con un sistema operativo Windows o Linux). A través de esa aplicación se accederá a una base de datos centralizada que contendrá la información necesaria de cada red y estación remota para realizar los cálculos que finalmente determinarán cual es la red que mejor se adapta a las necesidades de cada estación remota.

Capítulo 2: Desarrollo

2.1. Conceptos preliminares

2.1.1 Redes satelitales

Los satélites forman una parte esencial del sistema mundial de telecomunicaciones, permitiendo transportar grandes cantidades de información y tráfico entre diversas partes del globo. Los satélites poseen ciertas características que los hacen únicos frente a otros medios de comunicación. La principal ventaja, debida principalmente a la distancia que los separa de la tierra (unos 38.000 km para satélites geoestacionarios), es la posibilidad de cubrir vastas regiones del globo comunicando puntos muy distantes entre sí con una infraestructura relativamente sencilla y formando el punto central de un sistema que permite comunicar a muchos usuarios simultáneamente.

En su esquema más simple, un enlace satelital está formada por dos antenas parabólicas que se comunican entre sí a través de un satélite que es el encargado de retransmitir o replicar la información enviada por ellos. Una de esas antenas es la que provee los servicios (Internet, TV, telefonía, etc.) y una o varias antenas remotas reciben esa información, pudiendo ser el flujo del tráfico uni o bidireccional. Al flujo de información desde la antena terrestre hacia el satélite se la conoce como *Uplink*, mientras que al que fluye desde el satélite hacia la estación terrena la conocemos como *Downlink*. La figura 2.1 muestra este esquema.

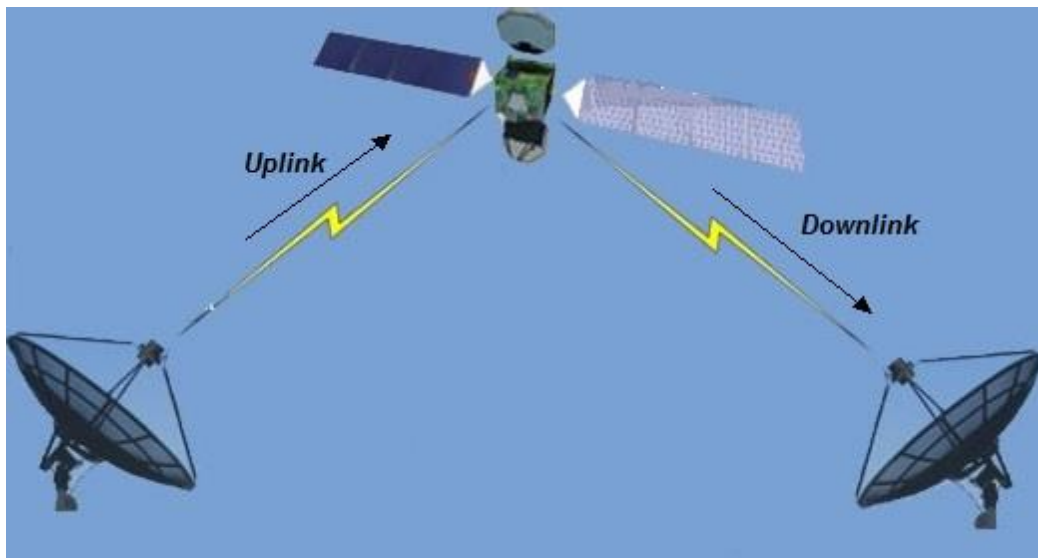


Figura 2.1: Esquema básico de un enlace satelital

Por supuesto, una red satelital es mucho más amplia y posee una estructura más compleja más allá de las antenas que permite brindar los servicios mencionados anteriormente, pero para los fines del presente trabajo, el esquema de la figura anterior es suficiente.

Los satélites pueden ser colocados en diversas órbitas más o menos cercanas a la Tierra dependiendo de la necesidad y así tenemos, entre otros, a los LEO (por sus siglas en inglés *Low Earth Orbit* – Órbita Terrestre Baja) utilizados principalmente en el campo meteorológico, los MEO (*Medium Earth Orbit* – Órbita Terrestre Media) con el Sistema de Posicionamiento Global (GPS) como su principal actor y finalmente a los GEO (*Geostationary Orbit* – Órbita Geoestacionaria) que son los más utilizados en el área de las telecomunicaciones.

La zona de cobertura o “huella” (*footprint*, en inglés) de un satélite, varía de uno a otro. Mas aún, dentro de una misma zona de cobertura, la intensidad de la señal recibida varía dentro de la huella. En la figura 2.2 podemos ver un ejemplo de ello para el satélite SES-4

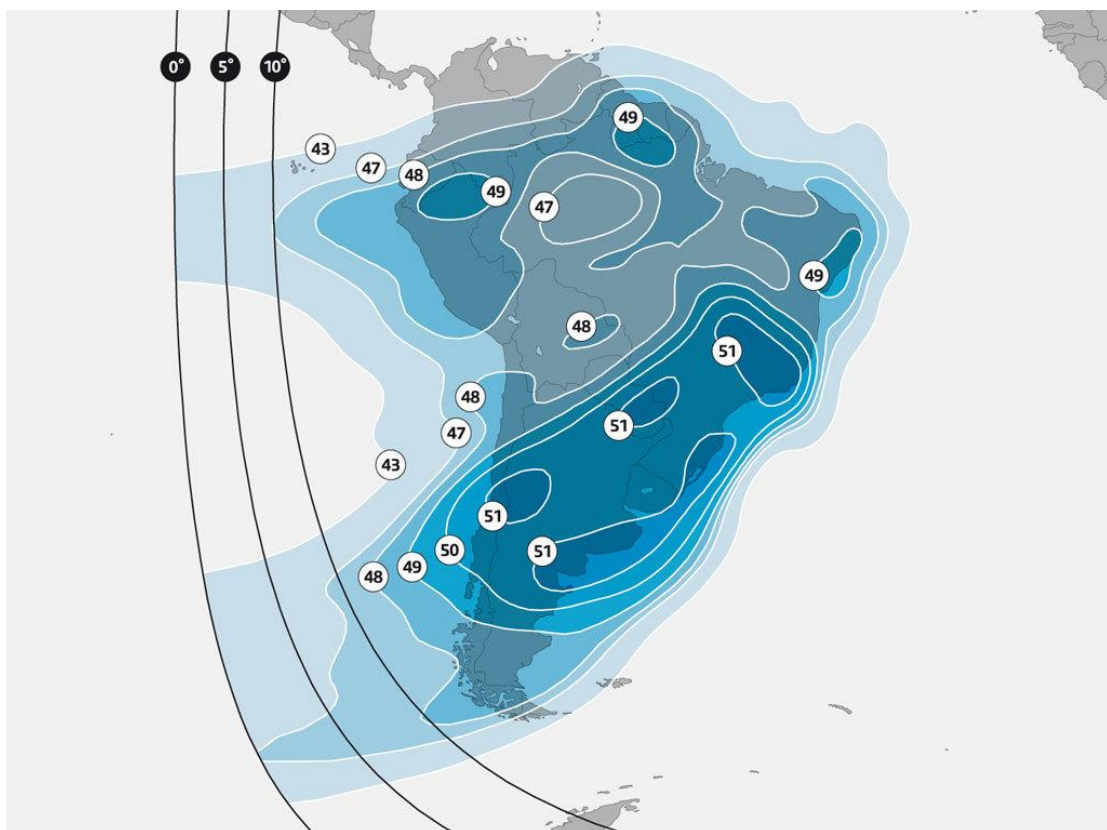


Figura 2.2: Valores de PIRE para el satélite SES-4 (Haz para el Cono Sur en banda Ku). La unidad de cada valor es el Decibel Watt [dBW]. Recuperado de: <https://www.ses.com/our-coverage/satellites/325>

Estas diferentes zonas van a depender de la configuración de los transpondedores (*transponders*, en inglés) del satélite y el nivel de transmisión o recepción del satélite son medidos a través del PIRE (Potencia Isotrópica Radiada Equivalente, medida en decibelios-Watt [dBW]) y el G/T (figura de mérito, medido en decibelios-Kelvin dBK) respectivamente. Los transpondedores son básicamente amplificadores cuya tarea es la de recibir la señal proveniente de la Tierra, procesarla (amplificarla y desplazarla en frecuencia) y enviarla nuevamente a la Tierra. En general, los satélites cuentan con varios de ellos, cubriendo toda la gama de frecuencias de operación de los mismos.

En lo que respecta a las estaciones terrestres y especialmente a las antenas más pequeñas o VSAT (del inglés *Very Small Aperture Terminal* o Terminal de Apertura muy Pequeña) la configuración básica de los elementos que las componen es como la que se muestra en la figura 2.3, formada por la antena propiamente dicha o Reflector, un Receptor (LNB) y un Transmisor (BUC).

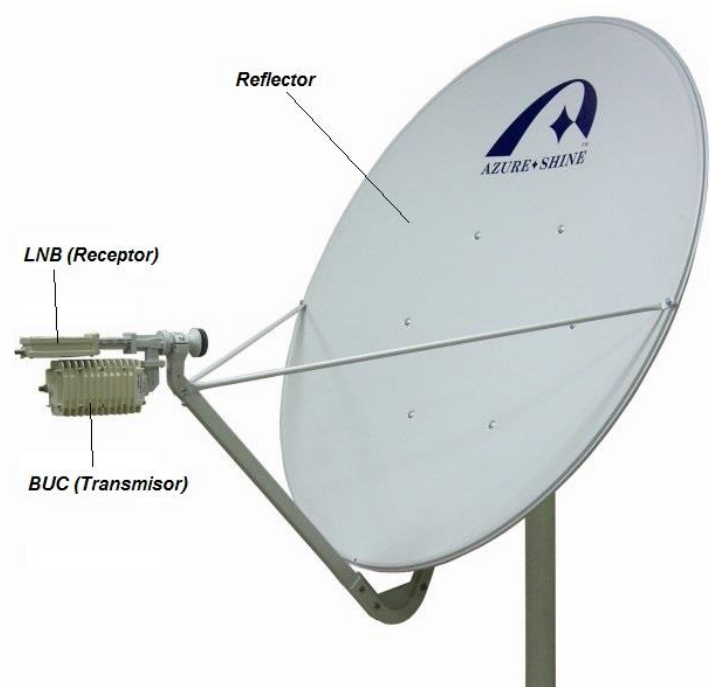


Figura 2.3: Configuración básica de una antena parabólica terrestre. Recuperado de https://www.alibaba.com/product-detail/120cm-VSAT-Antenna_11749858.html

2.1.2 Suscripción. CIR y MIR

Tanto el CIR (*Committed Information Rate* o Tasa de Información Comprometida) como el MIR (*Maximum Information Rate* o Tasa de Información Máxima) son conceptos bien conocidos y utilizados en el mundo de las telecomunicaciones cuando hablamos de prestación de servicios. El primero, como su nombre lo indica, hace referencia a la tasa de información o ancho de banda que se debe garantizar a los clientes, mientras que el segundo indica el valor máximo al que un determinado servicio podría eventualmente llegar. Estos valores se suelen especificar tanto para el Uplink como para el Downlink. Así, podemos tener, por ejemplo, un servicio de internet con un CIR(Down/Up): 10/5 [Mbps] y MIR(Down/Up): 15/7 [Mbps]

La suscripción nos da una idea del nivel de utilización de una red, generalmente en términos porcentuales. Por ejemplo, si una red tiene la capacidad de proveer 20 [Mbps] y la suma del CIR de las remotas registradas en esa red es de 10 [Mbps], decimos que la red tiene una suscripción del 50%. Por supuesto, la suma del CIR puede llegar a superar la capacidad de la red y entonces diremos que la red está *sobre suscripta*. En general, por cuestiones de rentabilidad financiera que escapan al análisis meramente técnico, se suelen manejar niveles de suscripción que superen levemente la capacidad de la red,

pero es importante saber que un excesivo nivel de sobresuscripción puede ser perjudicial ya que no seremos capaces de garantizar el CIR a todos los clientes registrados en esa red.

2.1.3 Coordenadas polares. Elevación, azimut y ángulo relativo

A la hora de apuntar cualquier antena parabólica a un satélite determinado, nos encontraremos con dos ángulos que debemos conocer para lograr nuestro cometido. Por un lado, tenemos la elevación, que es el ángulo medido desde el horizonte (mínimo ángulo: 0°) hacia el cenit (máximo ángulo: 90°). Por otro lado, tenemos al azimut, que es un ángulo que varía desde 0 hasta 360, estando el mínimo ubicado en el punto cardinal Norte y aumentando en dirección al Este. La figura 2.4 nos ayudará a visualizar y comprender mejor estos ángulos.

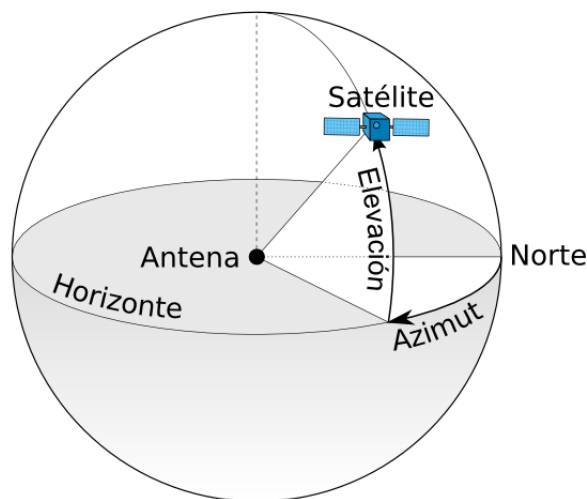


Figura 2.4: Coordenadas polares. Elevación y Azimut. Recuperado de: <https://linuxmanr4.com/2010/03/13/como-orientar-una-antena-de-dish-en-mexico/>

Cuando nos encontramos con antenas montadas sobre objetos móviles, y más específicamente barcos, debemos tener en cuenta otro ángulo: el relativo. El ángulo relativo es el que realmente nos interesa en este trabajo y se define como la diferencia entre el azimut y el *heading* o dirección del barco.

$$(1) \quad \textit{relativo} = \textit{azimut} - \textit{heading}$$

Como se puede deducir de la ecuación (1), el heading es también un ángulo y está referido a la proa del barco. Así, si la proa apunta exactamente hacia el Norte, el heading será de 0°; si apunta al Este será de 90° y así con cualquier ángulo hasta completar los 360°. De esta misma ecuación, podemos también deducir que, siendo el azimut de un valor constante para un satélite determinado, no lo es así el heading (ya que el barco se mueve) por lo que el ángulo relativo irá cambiando a medida que el barco se desplaza o rota sobre su eje.

2.2 Desarrollo

Como mencionamos en la Introducción del presente trabajo, nos encontramos ante la necesidad de evaluar algunas condiciones para determinar la elección de uno u otro satélite para una remota determinada. En este capítulo veremos cuáles son esas condiciones, analizaremos su funcionamiento y su interacción con el usuario.

Primero, veamos en la figura 2.5 un esquema general del sistema para ubicarnos mejor sobre qué elementos trabajaremos. La base de datos (1) es externa a la aplicación desarrollada y, en general, es parte de la infraestructura del Hub. Para el presente trabajo hemos simulado una de similares características que contiene la información que nos es útil. La misma puede ser accedida bien desde internet o desde un entorno local. En el otro extremo, tenemos la aplicación desarrollada en el presente trabajo (2) que será instalada en la máquina o estación de trabajo de los operadores.

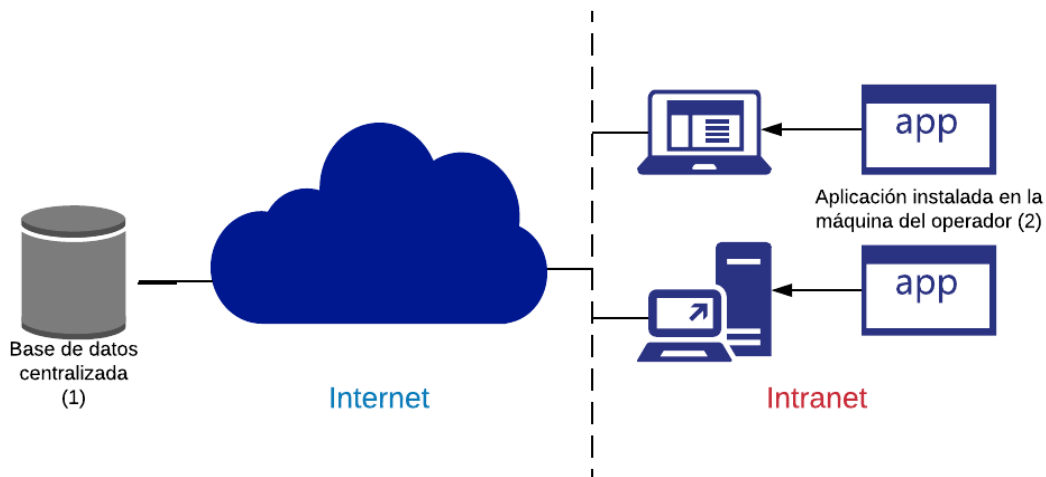


Figura 2.5: Esquema general de todo el sistema

Analizaremos a continuación cada uno de estos dos bloques (base de datos y aplicación) y añadiremos también la interfaz con el usuario para ver cómo interactúa ésta con los bloques anteriores.

2.2.1 Base de datos

Podemos pensar a una base de datos como un conjunto de tablas con columnas (campos) y filas (registros) similar a una hoja de cálculo, con información perteneciente a un mismo contexto.

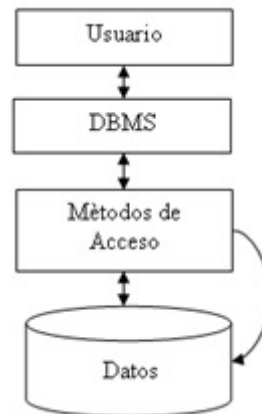


Figura 2.6: Esquema básico de una base de datos. Recuperado de https://es.wikipedia.org/wiki/Base_de_datos

Como vemos en la figura 2.6, el *DBMS* (por sus siglas en inglés *Database Management System* o Sistema de Administración de Base de Datos) es independiente del tipo de datos o información almacenada y en este trabajo está desarrollada en un entorno *MySQL* y montada físicamente en una computadora *Raspberry Pi 2*. En un principio se pensó en utilizar otros *DBMS* (como por ejemplo *SQLite*), pero finalmente se optó por *MySQL* debido a su rendimiento y confiabilidad. Además, debido a su popularidad, es el que seguramente encontraremos a la hora de una eventual integración de este sistema al mundo real.

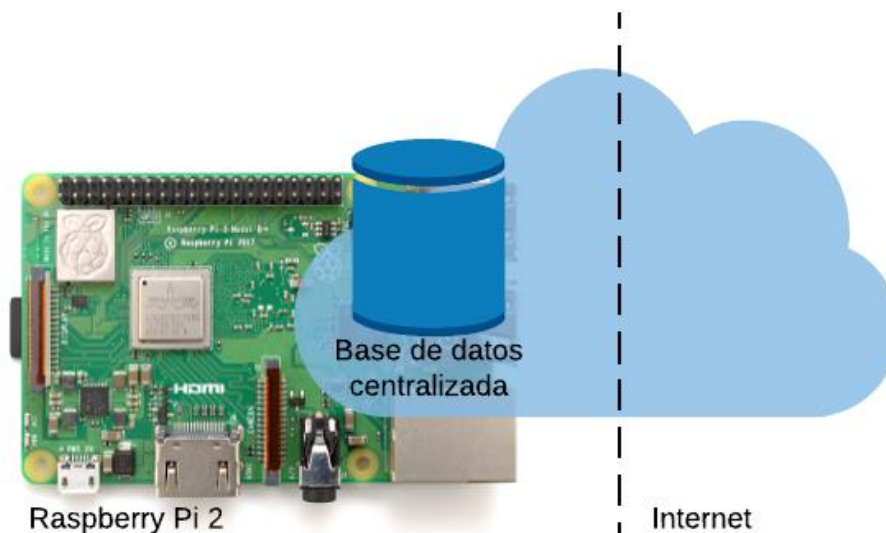


Figura 2.7: Servidor de base de datos alojado en Raspberry Pi. Recuperado y modificado de <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>

El conjunto de la figura 2.7 funciona como servidor de base de datos conteniendo la información necesaria y a ella accederemos, como ya hemos dicho, desde internet o desde un entorno local. En general, estos servidores de base de datos forman parte de una estructura de servidores mucho más compleja y potente, pero para los fines del presente trabajo, la Raspberry cumple perfectamente esa función. Las computadoras de tipo Raspberry forman parte de las llamadas computadoras de placa simple o *SBC* (*Single*

Board Computer) las cuales integran en una sola tarjeta de tamaño reducido la mayoría de los elementos y características de una placa madre o placa base (microprocesador, memoria RAM, periféricos de entrada y salida, etc.)

En el presente trabajo, hemos simulado una base de datos con similares características a la real y está formada por dos tablas, una con información de las remotas llamada “remotas” y la otra con información de las redes llamada “redes”. En las tablas 2.1 y 2.2 se muestran capturas de pantalla de las tablas “remotas” y “redes” respectivamente, que utilizamos en este trabajo y más abajo la descripción de cada campo.

id	lat	long_	head	ciru	cird	red	antena	buc
1	-34.6	-58.38	50	2	5	3	1.2	5
2	10.29	-66.52	124	8	16	3	1.2	5
3	-36.49	-73.02	266	5	10	5	0.75	3
4	55.7	-32.1	175	7	15	2	2.4	8
5	-12.5	90.3	5	7	14	4	0.98	5
6	66.5	-25.3	350	8	20	2	1.2	5
7	2.5	-21.66	52	6	12	3	2.4	10
8	-45.45	-74.2	42	8	15	3	2.4	10
9	23.35	58.24	261	0.5	2	4	0.75	5
10	53.34	10	100	7	14	4	1.2	3
11	9.36	-79.9	55	5	10	1	0.98	3
12	25.5	51.25	25	4	10	4	1.2	3
13	44.17	28.64	24	5	10	4	1.2	3
14	59.33	18.06	162	3	5	1	3.8	10
15	9.59	-83.02	1	8	15	5	1.8	8
16	21.5	-39.13	55	1	3	2	1.2	5

Tabla 2.1: Tabla “remotas”

id: identificador único de cada remota.

lat: latitud.

long_: longitud.

head: heading o dirección del barco.

ciru: CIR de subida o Upstream medido en Mbps.

cird: CIR de bajada o Downstream medido en Mbps.

red: red en la que la remota se encuentra registrada.

antena: tamaño de la antena en metros

buc: potencia del BUC en Watts

id	nombre	pos	cap_down	cap_up
1	IS_903	-31.5	29	9
2	SES_6	-40.5	80	30
3	Anik_G1	-107.3	40	20
4	IS_22	72	22.5	15
5	Telstar_11N	-37.5	35	20

Tabla 2.2: Tabla “redes”

id: identificador único de cada remota.

nombre: nombre dado a la red y por el que se la conoce en el sistema.

pos: posición orbital de -180° a 180° .

cap_down: capacidad de la red en bajada medida en MHz.

cap_up: capacidad de la red en subida medida en MHz.

2.2.2 Aplicación principal. Contenido y prestaciones.

En esta sección veremos cómo fue realizado el análisis y estudio de las diferentes variables para determinar cuál es la mejor red para una remota determinada.

Toda la aplicación fue desarrollada en lenguaje Python y la misma puede instalarse, como una versión portable, en cualquier PC de escritorio o laptop con sistema operativo Windows o Linux (probado con éxito en ambos). La razón por la cual se escogió este lenguaje es debido a la sencillez de su sintaxis, a que puede trabajar en diferentes plataformas (Windows, Linux, Mac, Raspberry Pi, etc.) y además permite tanto la programación secuencial como la orientada a objetos, entre otras cosas.

En la figura 2.8 podemos ver los cuatro bloques que el sistema analiza a través de la aplicación: I) *Posición geográfica*, II) *Capacidad*, III) *Potencia* y IV) *Bloqueo*. Además, al lado de cada bloque se menciona la forma o la manera en la que cada uno de los bloques anteriores fue estudiado o resuelto.

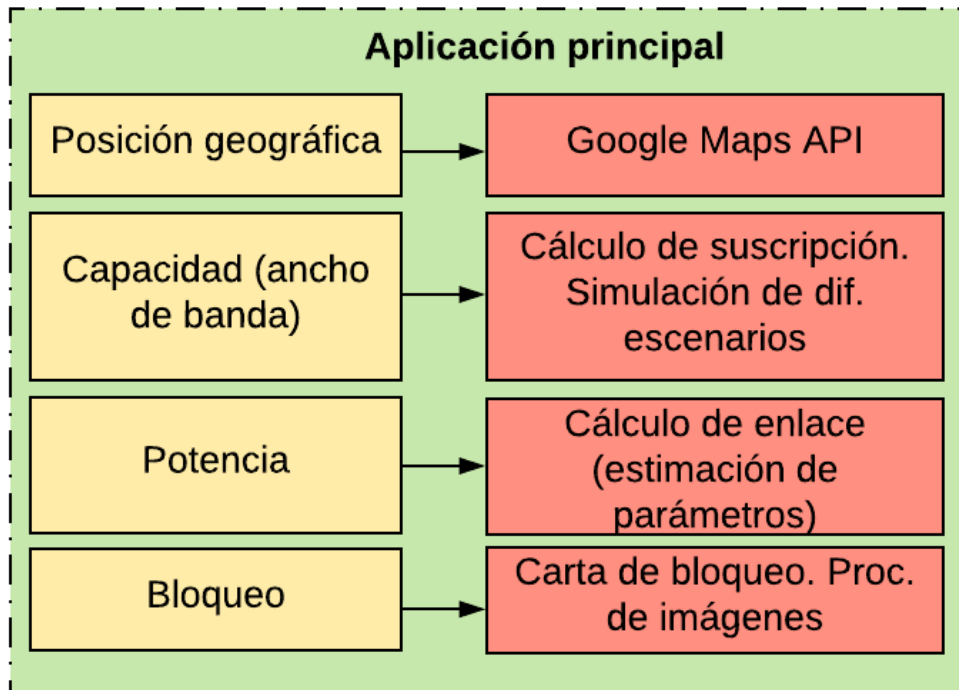


Figura 2.8: Esquema de la Aplicación principal

A continuación, analizaremos cada uno de esos cuatro bloques, fundamentando el porqué de su elección y analizando su funcionamiento e interacción con la base de datos.

1) *Posición geográfica*: dependiendo de la posición geográfica de la estación remota, ésta puede estar en la zona de cobertura de más de un satélite (o ninguno), esto es, puede tener más de una opción de conexión. En la figura 2.9 vemos un planisferio donde están representadas algunas de las tantas diferentes zonas de cobertura o *footprints* de distintos satélites, así como las remotas, distribuidas por todo el globo y representadas por pequeños puntos.

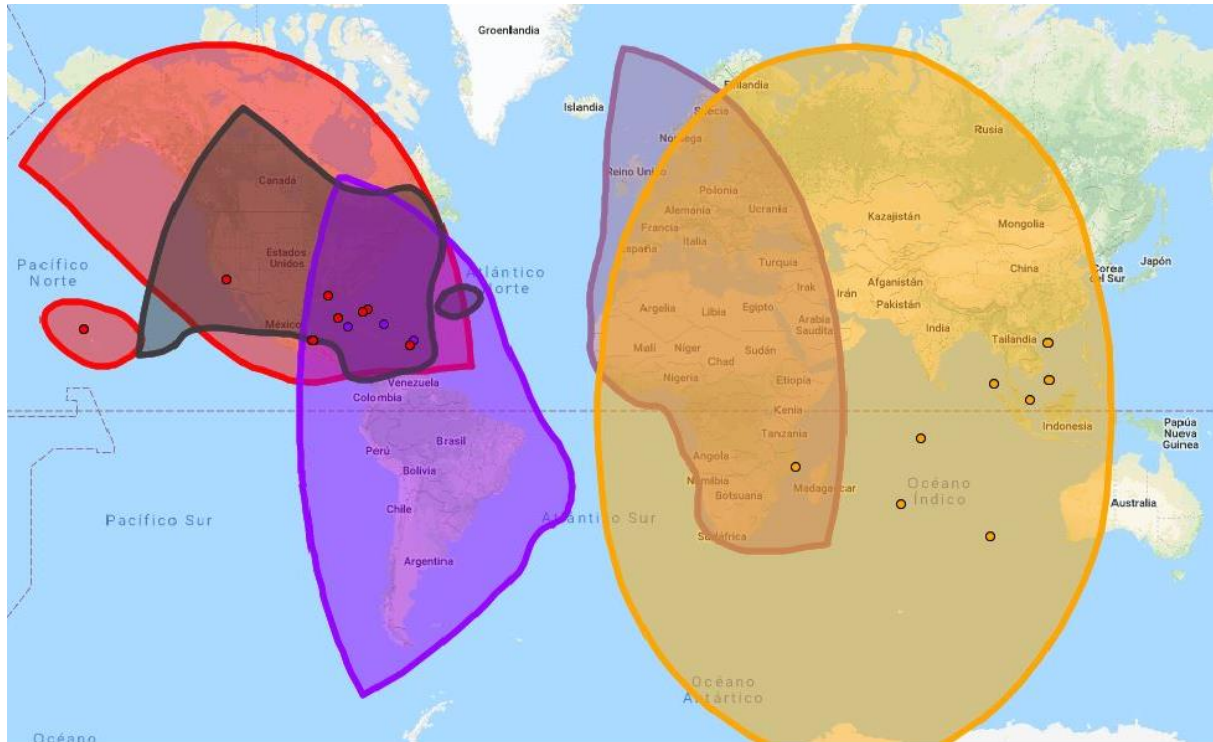


Figura 2.9: Coberturas o “footprints” de algunos satélites. Los puntos indican las estaciones remotas; cómo se observa, cada una puede estar cubierta por uno o más satélites

En este trabajo, podemos conocer la latitud y longitud de cualquier remota, así como también ver su posición geográfica en un planisferio gracias a una Interfaz de Programación de Aplicaciones o API (*Application Programming Interface*) de *Google Maps*. Para lograr esto, la aplicación debe consultar la base de datos cada vez que el operador lo requiera, por lo que la información siempre estará lo más actualizada posible. En el mapa también es posible ver la zona de cobertura de cada red y de esta manera determinar, por ejemplo, que redes podrían eventualmente brindar servicio a la remota.

Podemos ver en la figura 2.10 un diagrama esquemático del funcionamiento de este bloque para conocer como interactúa con los demás elementos y en la figura 2.11 su diagrama de flujo. Podemos notar que para poder hacer uso de la aplicación *Google Maps*, debemos configurarle una llave o *key* la cual - si bien posee un período de prueba gratuito – tiene un costo que será analizado más adelante.

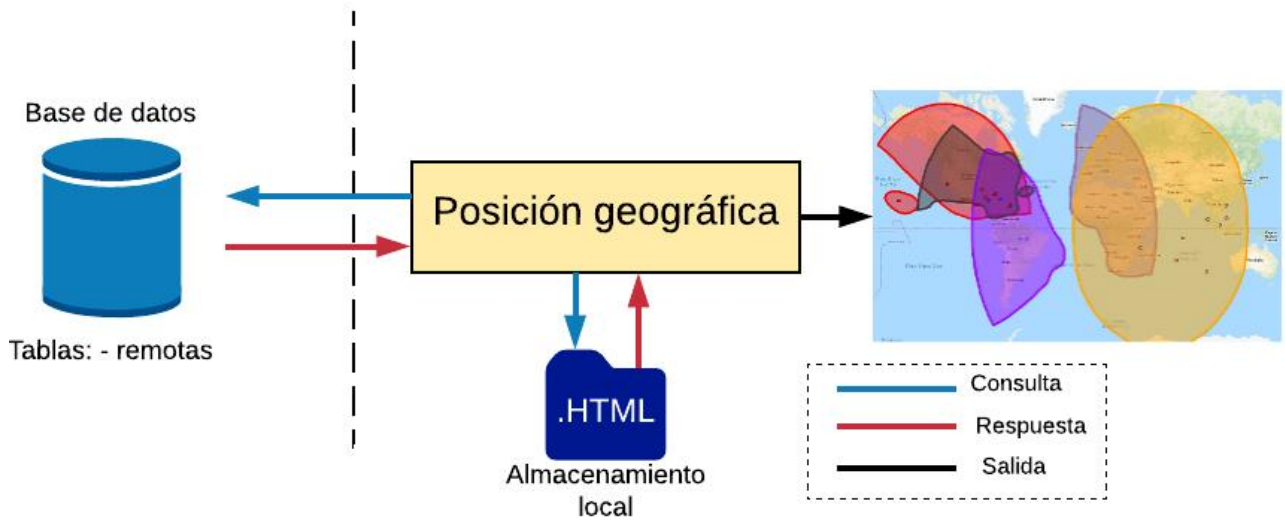


Figura 2.10: Esquema del bloque Posición geográfica

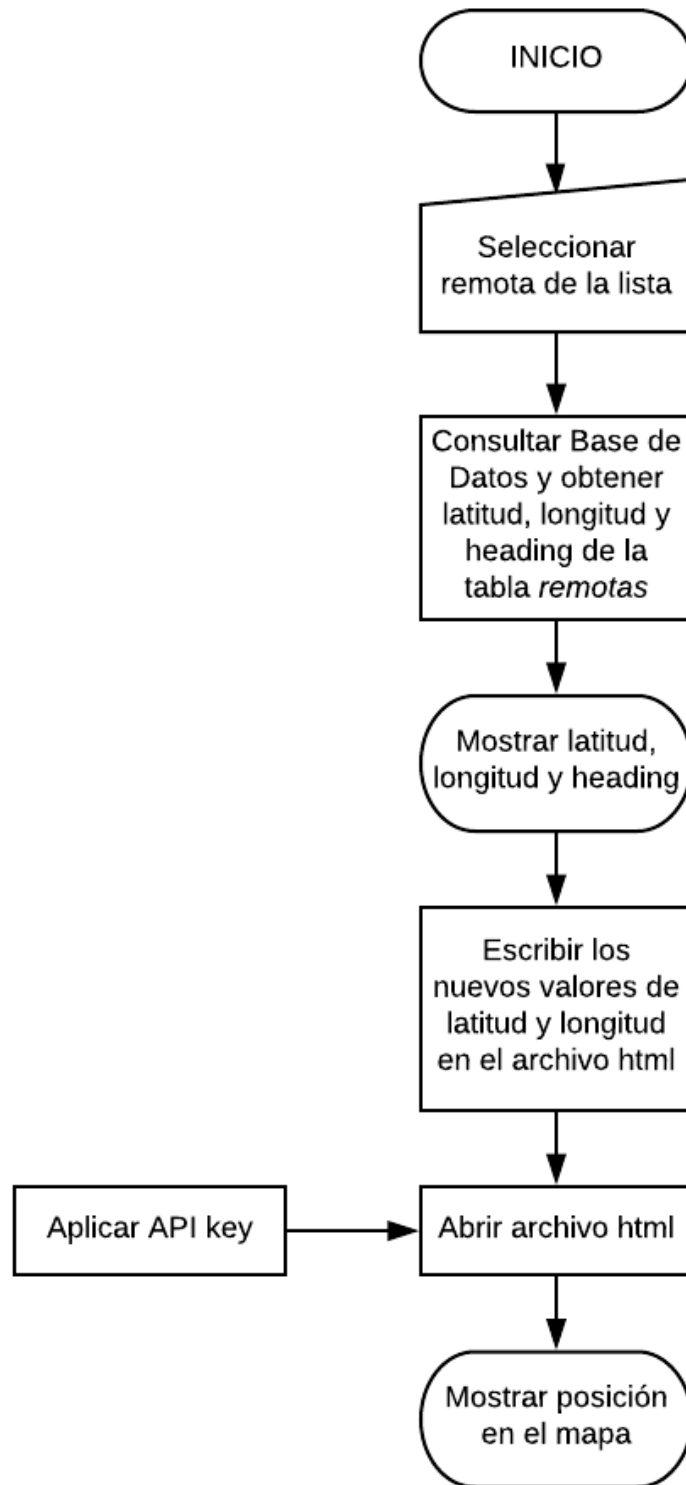


Figura 2.11: Diagrama de flujo del bloque Posición geográfica

Como se muestra en la figura 2.10, este bloque interactúa constantemente con la tabla *remotas* de la base de datos para consultar la latitud y longitud de determinada remota. Una vez obtenidos, debe modificar el archivo html que finalmente podrá ser visualizado en cualquier navegador web (Internet Explorer, Google Chrome, Mozilla Firefox, etc.) no sin antes aplicarle al mismo una clave o key a través de un script html. Este archivo html ya tiene cargadas las zonas de cobertura de cada red y debe ser

almacenado localmente en la estación de trabajo de cada operador. Se decidió realizarlo de esta manera para acelerar el proceso de visualización aprovechando además que las mencionadas zonas no son, generalmente, modificadas con frecuencia. La figura 2.12 muestra una captura de pantalla donde se puede ver una remota – representada por un ícono con forma de barco – ubicada en Caracas, Venezuela (latitud: 10.49 Norte, longitud: 66.9 Oeste) y las 5 zonas de cobertura utilizadas de ejemplo para este trabajo: *IS903*, *SES6*, *Anik G1*, *IS22 ME* y *Telstar 11N*. Estas zonas de cobertura existen en la actualidad y fueron reproducidas tratando de asemejarlas lo más posible a la realidad. Nótese también que el nombre de cada red es mostrado cada vez que hacemos clic sobre ella.

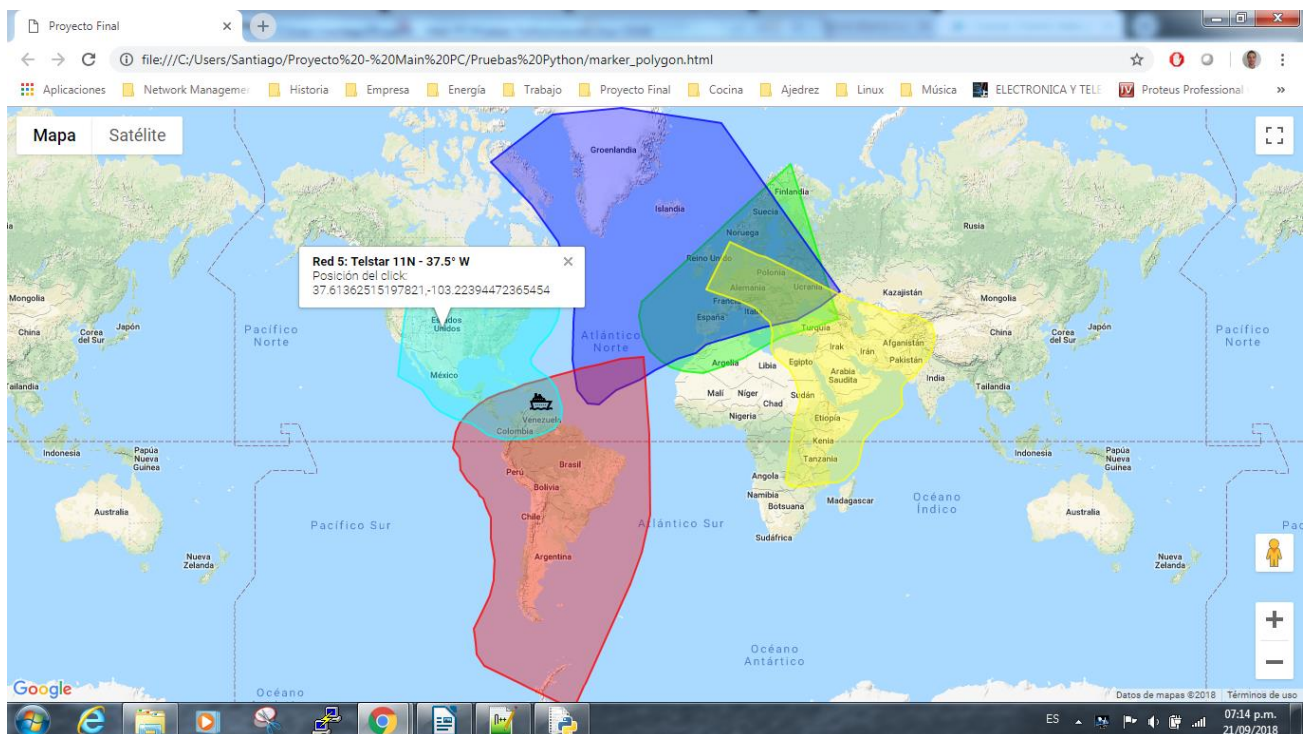


Figura 2.12: Captura de pantalla para el ejemplo de una remota ubicada en Caracas, Venezuela

Es importante mencionar que, dentro de una misma zona de cobertura, los niveles de potencia tanto de transmisión como de recepción varían (ver figura 2.2). En general, estos niveles suelen atenuarse a medida que nos acercamos al borde la huella por lo que necesitaremos contar con una mayor potencia en la estación remota, así como también dimensiones de antena mayores las cuales, muchas veces, no están presentes. Por otro lado, los niveles de suscripción que maneja cada red son diferentes y también deben ser analizados. Por esta razón, surge la necesidad de analizar otras variables además de la posición geográfica, como veremos a continuación.

II) Capacidad disponible: una vez confirmada la ubicación de la remota dentro de la zona de cobertura de determinado satélite, podemos analizar la capacidad disponible en la red actual o en la de destino. Cuando hablamos de capacidad, nos referimos al ancho de banda (medido generalmente en Mega bits por segundo [Mbps]) de la red. Como sabemos, cada remota tiene asociado un CIR y un MIR (también medido en Mbps) que

deberá ser tenido en cuenta a la hora de registrarlo a una determinada red para saber si la misma será capaz de garantizarle su ancho de banda. Si los niveles de suscripción de la red a donde deseamos registrar nuestra remota son muy elevados, quizás sea conveniente optar por otra opción.

Este bloque permite conocer tanto la suscripción actual de cada red, así como también simular diferentes escenarios a través de la relocalización de una remota de una red a otra. Muchas veces, debido a la alta suscripción de una red, es necesario relocalizar alguna de las remotas registradas allí hacia otra red, lo que suele conocerse como *balance de cargas*. De esta manera, podremos conocer con anticipación el estado de las diferentes redes antes de realizar cualquier movimiento y evitar así inconvenientes o cortes innecesarios en el servicio.

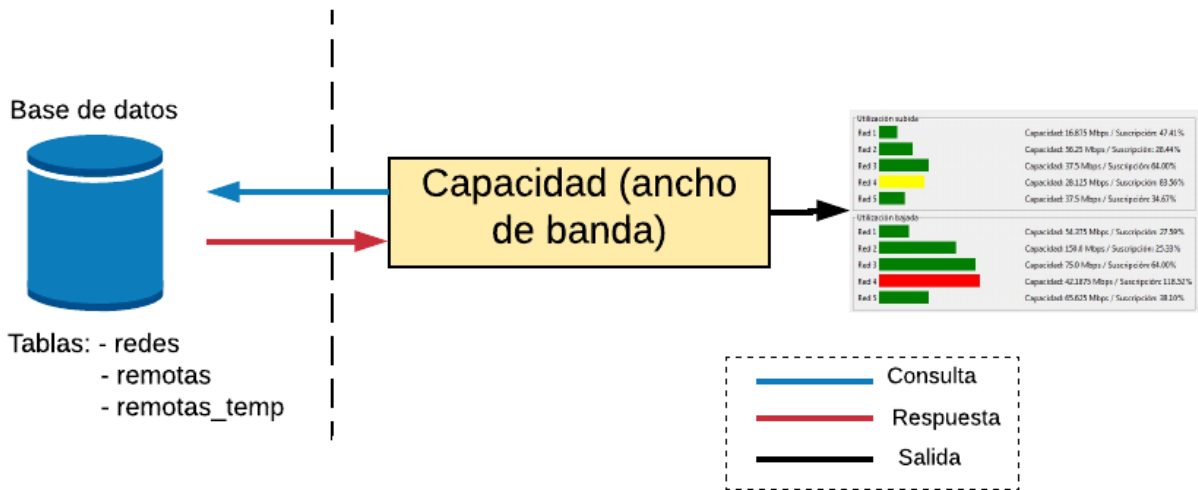


Figura 2.13: Esquema del bloque Capacidad

Del esquema de la figura 2.13, podemos ver algunas diferencias respecto al anterior esquema de la figura 2.10. Por el lado de la base de datos, esta vez además de la tabla *remotas* consultamos las tablas *redes* y *remotas_temp* que es una versión idéntica a *remotas* usada solo con fines de visualización. Por otro lado, aquí no hacemos uso alguno del almacenamiento interno.

Debido a que este bloque permite no solo la visualización de la suscripción sino también la simulación de diferentes escenarios, vamos a dividir en dos el diagrama de flujo como se muestra en la figura 2.14, representando uno de los diagramas solo la visualización y el otro el resultado de la simulación.

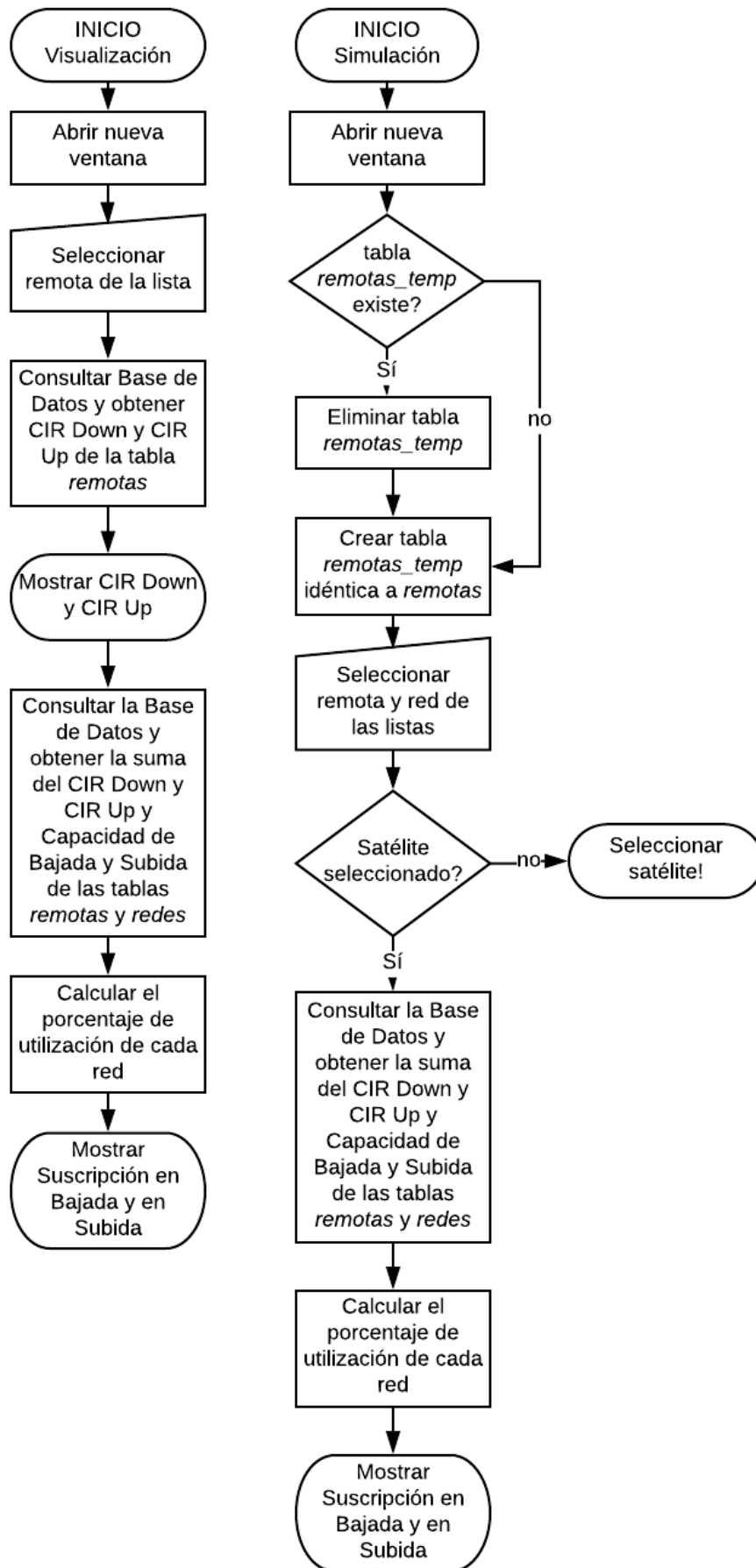


Figura 2.14: Diagrama de flujo del bloque Capacidad

Podemos ver que ambos diagramas de la figura 2.14 guardan cierta similitud en su estructura general. La principal diferencia es que consultan tablas diferentes y, por lo tanto, mostrarán diferentes resultados.

Hasta aquí hemos analizado dos condiciones o variables que deben ser tenidas en cuenta a la hora de considerar registrar una remota a determinada red: la posición geográfica y la capacidad de la red. Veremos a continuación dos condiciones más que muchas veces no son tenidas en cuenta, implicando ello cortes innecesarios en el servicio y, por ende, pérdidas de tiempo.

III) Potencia: los requerimientos de potencia necesaria para entrar o permanecer en determinada red varían de una red a otra y dependen de múltiples variables, ya sea relacionadas a elementos propios del enlace (potencia del BUC, tamaño de la antena, sensibilidad del transpondedor del satélite, ancho de banda de las señales portadoras, etc.) o a cuestiones geográficas, como la ubicación relativa de la remota dentro de la huella del satélite. Podemos calcular esos requerimientos basados en diferentes objetivos de rendimiento como ser la tasa de errores para determinadas condiciones de operación, la disponibilidad del enlace, etc., o bien estimarlos obteniendo también buenos resultados.

Veremos ahora un diagrama y una serie de fórmulas que nos ayudarán a comprender mejor cuales son las variables y los parámetros que entran en juego en todo enlace satelital.

La relación señal a ruido C/N en un sistema o enlace establece de alguna manera el rendimiento de la porción de radiofrecuencia del mismo y se define como la relación entre el nivel de potencia de la señal recibida y el nivel de potencia de ruido a la entrada del receptor. La figura 2.15 ilustra cómo algunos elementos del enlace afectan el C/N recibido, los cuales finalmente afectarán a la calidad del servicio.

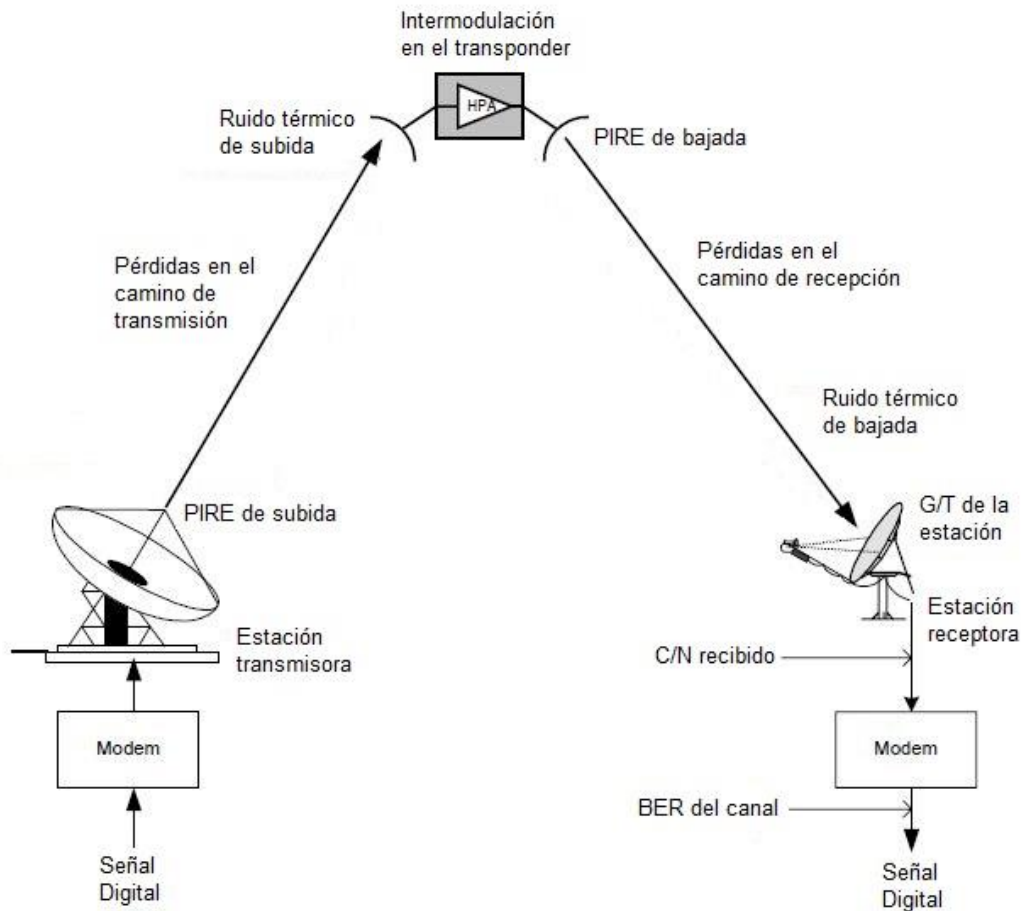


Figura 2.15: Elementos presentes en un enlace satelital típico. INTELSAT, (1999), Typical Satellite Link. Recuperado de Earth Station Technology

Podemos expresar el C/N recibido por una antena de la siguiente manera:

$$(2) \quad C/N = PIRE - L - G - 10\log kTB$$

donde:

- PIRE*: Potencia Isotrópica Radiada Equivalente de la antena transmisora
- L*: Pérdidas de transmisión
- G*: Ganancia de la antena receptora
- k*: Constante de Boltzman ($1,38 \times 10^{-23}$ W/°K/Hz)
- T*: Temperatura del sistema receptor en °K
- B*: Ancho de Banda en Hz

Los primeros tres términos de la ecuación (2) hacen referencia al nivel de potencia de la portadora recibida y el último es la potencia de ruido del sistema receptor.

Las pérdidas de transmisión en general están formadas por cuatro componentes, que son:

$$(3) \quad L = L_o + L_{atm} + L_{lluvia} + L_{track}$$

donde:

L_o : Pérdidas en el espacio libre

L_{atm} : Pérdidas en la atmósfera

L_{lluvia} : Atenuación debido a los efectos de la lluvia

L_{track} : Pérdidas debidas a errores en el apunte de la antena.

Por otro lado, la antena del satélite tiene una determinada configuración en su patrón de emisión que termina “dibujando” sobre la superficie terrestre formas con diferente PIRE y G/T. Tanto el PIRE como el G/T están referenciados al borde de la huella y en el análisis de toda estación remota se deben aplicar distintos factores de corrección referidos a ese borde. Esos factores de corrección – generalmente conocidos como factores β – se definen como la diferencia entre la ganancia en el borde de la huella y la ganancia en la ubicación de la estación remota.

Un cálculo aproximado de estos factores puede realizarse basándonos en gráficos como los de la figura 2.2 y los de la figura 2.16, donde cada línea o contorno representan, incrementos de 1 dB desde el borde. Si se desean valores más precisos, los mismos pueden ser solicitados al proveedor satelital del satélite en cuestión.

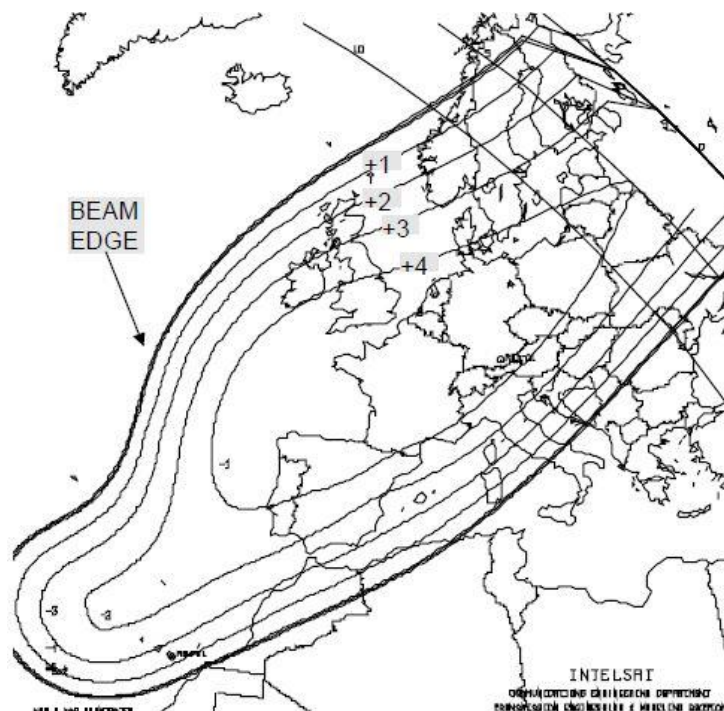


Figura 2.16: Ejemplo de huella satelital. INTELSAT, (1999), Example of Pattern Advantage. Recuperado de Earth Station Technology

En el presente trabajo hemos dividido el análisis de este punto en dos partes: una dedicada al *Downlink* y otra al *Uplink*, ambos para la banda Ku que va de los 12 a los 18 GHz. En la primera, estimamos si la remota será capaz de recibir correctamente la señal del satélite basado en su posición geográfica respecto al footprint del satélite y al tamaño de su antena. Para ello, hemos dividido la zona de cobertura en tres regiones o zonas (1, 2 y 3) en función del PIRE de la antena del satélite – ver un ejemplo para el satélite Telstar

11N en la figura 2.17. Entonces, conociendo la región dentro de la cual se encuentra la remota y sabiendo el tamaño mínimo de antena para ese PIRE (información tabulada que puede verse en la tabla 2.3), podemos finalmente determinar si la remota será capaz de recibir correctamente la señal.

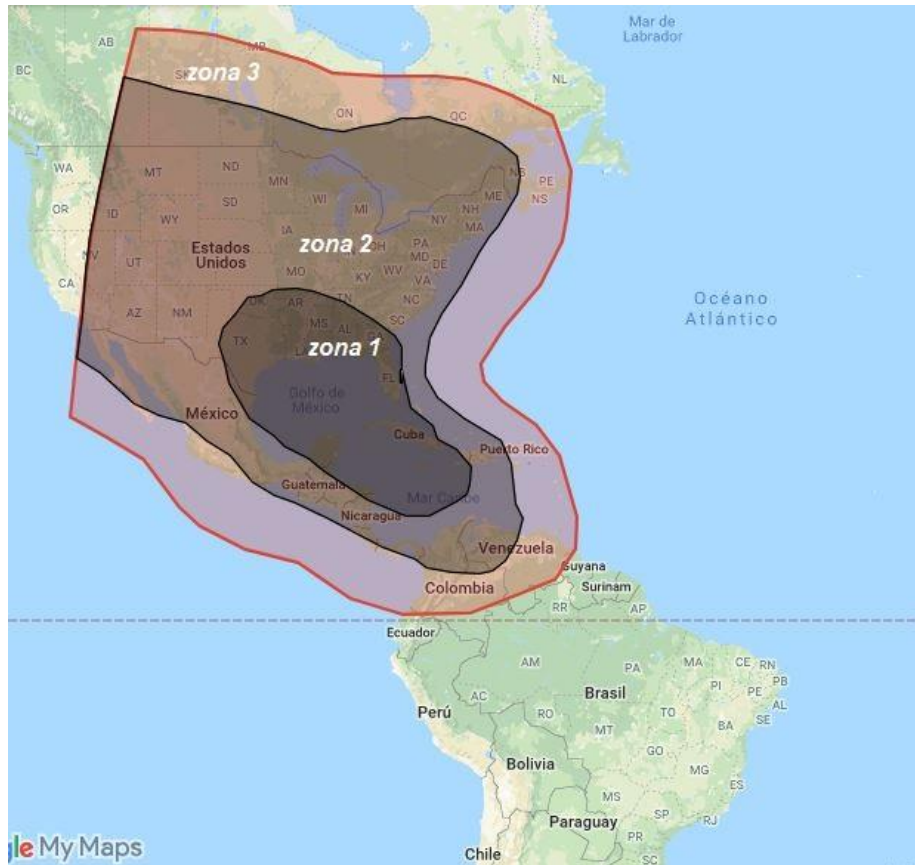


Figura 2.17: Zonas de cobertura en función del PIRE.

Dish Size Estimate (Ku-band)																							
EIRP (dBW)	<30	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	>50
Min (cm)	>535	475	425	380	335	300	270	240	215	190	170	150	135	120	105	95	85	75	65	60	55	50	50
Max (cm)		600	535	475	425	380	335	300	270	240	215	190	170	150	135	120	105	95	85	75	65	60	

Tabla 2.3: Tamaños mínimos y máximos del reflector en función del PIRE del satélite

Por otro lado, en la parte correspondiente al Uplink, hemos tomado como referencia 5 portadoras de diferentes tamaños que nos servirá para estimar no solo si la remota será capaz de entrar en una determinada red, sino también para poder determinar si es o no posible realizar un *upgrade* o mejora en el ancho de banda de la portadora de subida con el BUC instalado.

Los cálculos para poder determinar la potencia del BUC necesario son los siguientes:

Si desdoblamos la ecuación (2), obtenemos lo siguiente:

$$(3) \quad C/N = PIRE - L + G - 10\log k - 10\log T - 10\log B$$

La diferencia $G - 10\log T$ es lo que conocemos como Figura de mérito. Entonces, reemplazando en la ecuación (3) obtenemos:

$$(4) \quad C/N = PIRE - L + G/T - 10\log k - 10\log B$$

Debido a que el ancho de banda de la señal recibida es dependiente de la modulación utilizada y el C/N siempre está caracterizado por un determinado ancho de banda, es que suele recurrirse al cálculo del C/N_0 . El C/N_0 es un parámetro que no depende de ninguna propiedad de la señal y puede pensarse como el C/N que podría alcanzarse en un ancho de banda de 1Hz. Por ejemplo, si un sistema receptor (por ejemplo, un demodulador) está recibiendo una señal con un C/N de 11 dB de una portadora con una tasa de símbolos (Symbol Rate) de 1 Msps (Mega símbolo por segundo), cabría preguntarse cuál es el C/N que alcanzaría una señal de 2Msps manteniendo la misma potencia. La respuesta en este caso, sería un C/N de 8dB para la señal de 2Msps. Vemos que, aunque la potencia se ha mantenido igual, al duplicar el ancho de banda estamos también duplicando la potencia del ruido a la entrada del receptor; por esa razón el C/N disminuye a la mitad (-3dB). Es por ello que muchas veces, en lugar de hacer comparaciones con referencias arbitrarias corriendo el riesgo de errores y confusión, se prefiere el uso del C/N_0 en lugar de C/N.

El C/N_0 es igual al $C/N + 10\log(\text{Ancho de Banda})$ y se expresa en dBHz.

Entonces, siguiendo con el análisis:

$$(5) \quad C/N = C/N_0 - 10\log B$$

Reemplazando (5) en (4):

$$(6) \quad C/N_0 = PIRE - L + G/T - 10\log k$$

Despejando el PIRE, obtenemos:

$$(7) \quad PIRE_{op} = C/N_0 + L - G/T + 10\log k$$

donde:

$PIRE_{op}$: es el que finalmente deberá estar presente a la salida de la antena del satélite para lograr el C/N deseado.

C/N_0 : valor requerido a la entrada del receptor para una señal determinada.

L : pérdidas en el espacio libre para el trayecto satélite-estación remota

G/T : factor de mérito de la antena receptora

Debido a que el transpondedor del satélite no es un dispositivo lineal, debe ser operado por debajo de su punto de saturación para evitar distorsiones no lineales.

Por lo tanto, debe considerarse un *backoff* tanto a la entrada (IBO, Input Backoff) como a la salida del mismo (OBO, Output Backoff). Podemos entender al backoff como a algunos decibelios que deberemos considerar, pero no podremos utilizar. Lamentablemente, esto es una pérdida de potencia inevitable con el fin de evitar inconvenientes como los ya mencionados.

El backoff de entrada o IBO se define como la relación entre la Densidad de Flujo de Saturación o SFD (por sus siglas en inglés, *Saturation Flux Density*) y la Densidad de Flujo de operación (OFD). El SFD es la densidad de flujo de potencia total que, llegada al satélite desde la estación terrena, producirá el PIRE de saturación en el satélite.

Por otro lado, el backoff de salida u OBO se define como la relación entre el PIRE de saturación y el PIRE de operación. Aunque también podemos definirlo como:

$$(8) \quad OBO = IBO - X$$

X es la relación de compresión entre el IBO y el OBO. Este valor se encuentra ya tabulado y depende, entre otras cosas, de si estamos trabajando en un esquema de portadora simple o multiportadora en el transpondedor.

Dicho lo anterior, sigamos con nuestro análisis:

$$(9) \quad OBO = PIRE_{sat} - PIRE_{op}$$

$PIRE_{sat}$ es un dato proporcionado por el proveedor satelital

$$(10) \quad IBO = OBO + X$$

Finalmente, el OFD o nivel de iluminación W (medido en dBW/m²) que lograremos será:

$$(11) \quad W = SFD - IBO$$

Ahora, una vez obtenido el valor de iluminación necesaria para lograr el C/N necesario en nuestra remota, vamos a concentrar nuestro análisis en el segmento correspondiente a la estación terrena que nos arrojará, finalmente, la potencia del BUC o HPA (*High Power Amplifier*) necesaria para lograrlo.

Podemos expresar el PIRE necesario en una antena terrestre para lograr un W determinado como:

$$(12) \quad PIRE_{dBW} = W + L_{ou} - G_{1m^2}$$

donde:

L_{ou} : pérdidas en el espacio libre para el trayecto estación remota-satélite
 G_{1m^2} : ganancia de una antena de 1 m² con una eficiencia del 100%

La potencia necesaria del HPA en dBW será (sin considerar pérdidas en las partes o conexiones de la antena):

$$(13) \quad P_{HPA} = PIRE - G$$

Por último, si queremos expresar esa potencia en Watts, no tenemos más que aplicar la fórmula:

$$(14) \quad P_{(W)} = 1W \cdot 10^{(P_{(dBW)}/10)}$$

El resultado de la ecuación (14) será entonces la potencia necesaria en nuestro BUC para transmitir una señal determinada.

Como hemos hecho en los puntos anteriores, veremos ahora en las figuras 2.18 y 2.19, un esquema y un diagrama de flujo de este bloque.

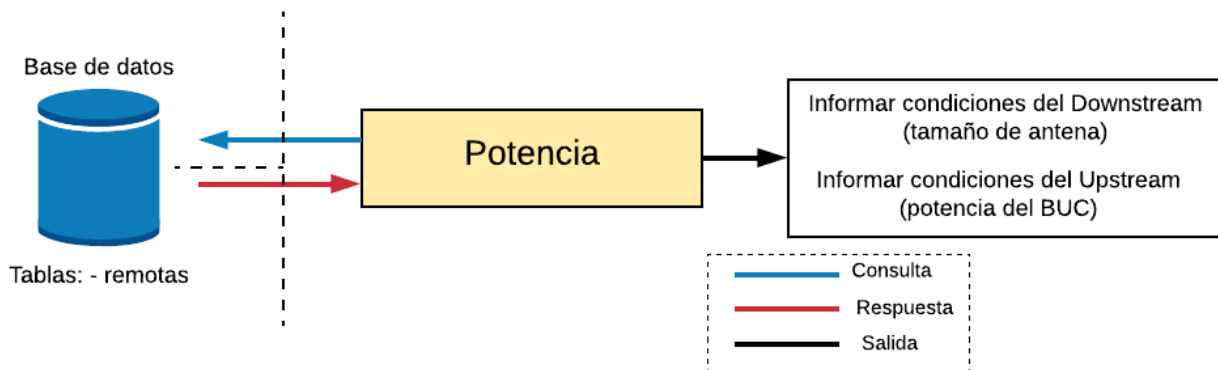


Figura 2.18: Esquema del bloque Potencia

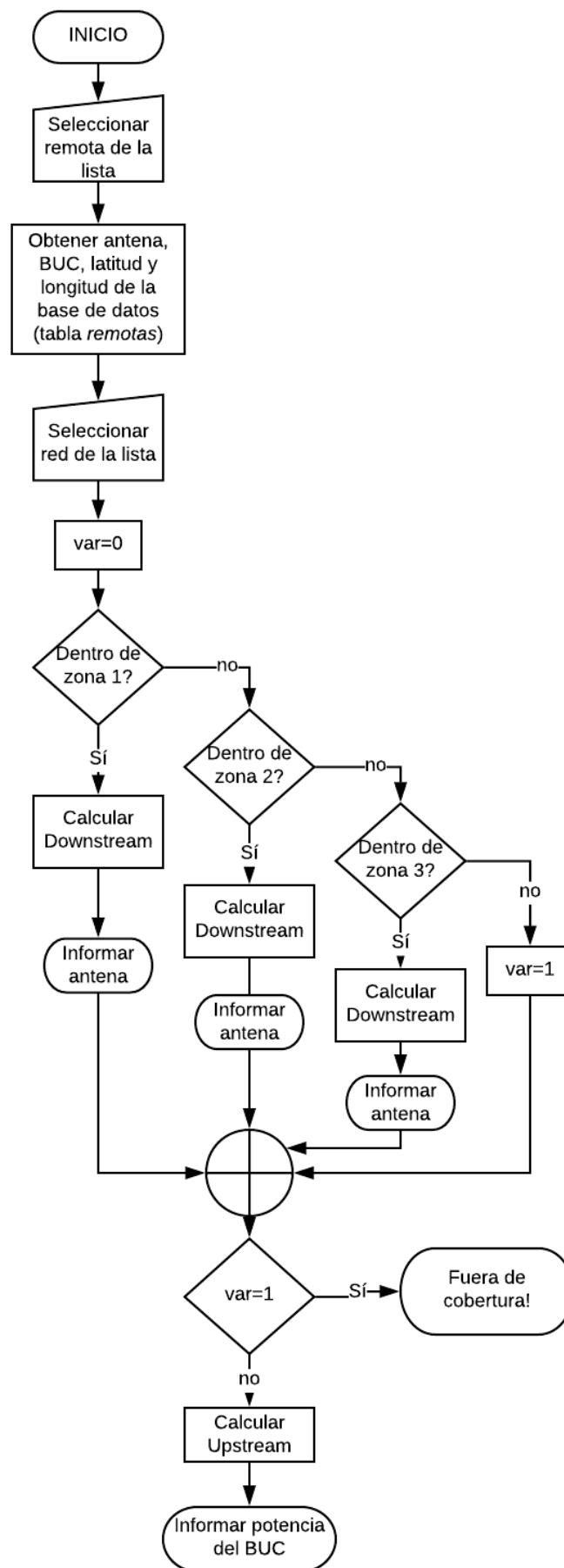


Figura 2.19: Diagrama de flujo del bloque Potencia

IV) *Bloqueo*: resulta paradójico que en la práctica este punto sea el menos tenido en cuenta siendo que su no cumplimiento invalidaría los tres puntos anteriores. Es decir, de nada sirve que la remota se encuentre dentro de los límites de la zona de cobertura de cierto satélite, la capacidad disponible de la red destino sea más que suficiente y además tenga potencia en exceso si la línea de visión entre la antena y el satélite no está completamente libre de obstrucciones (con una mínima obstrucción podría llegar a funcionar, pero el buen rendimiento no estaría garantizado).

Para determinar si la línea de visión entre la antena de la estación remota y el satélite se encuentra o no libre de bloqueo, en el presente trabajo analizamos la Carta de Bloqueo de cada remota y determinamos el estado de la línea de visión remota-satélite (Libre, Bloqueada o Cercana al bloqueo). La Carta de Bloqueo es una representación en el plano del espacio que rodea a la antena, indicando en sendos ejes x e y, los ángulos de azimut y elevación, respectivamente. Un ejemplo de Carta de Bloqueo aparece en la figura 2.20. La misma es confeccionada por el técnico que hace la instalación de la antena y es única y diferente para cada remota (embarcación, en este caso).

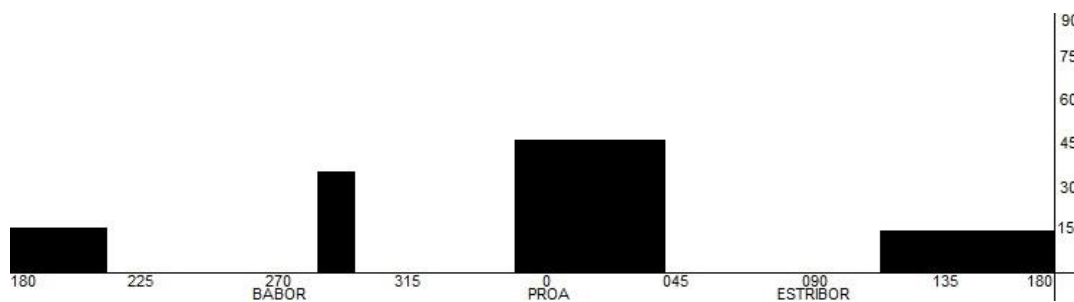


Figura 2.20: Ejemplo de Carta de Bloqueo

Una vez que conocemos la posición orbital del satélite a utilizar, así como la latitud, longitud y heading de nuestra remota, podemos calcular los ángulos de elevación y azimut relativo de nuestra antena. La intersección de esos dos valores nos dará como resultado un punto en nuestra Carta de Bloqueo. A través de un procesamiento de imágenes, asociamos el color del píxel del punto encontrado a uno de dos estados posibles: *negro = bloqueo*, *blanco = libre de bloqueo*.

Es importante mencionar que para que el sistema pueda procesar correctamente la imagen, la misma debe tener extensión JPG con una dimensión de 740x200 píxeles. Considerando que la Carta de Bloqueo fue confeccionada de forma manual y su exactitud depende del buen criterio u observación del técnico instalador, si el punto encontrado se encuentra cercano (aproximadamente $\pm 10^\circ$ del punto central) a un color negro, tendremos un tercer estado llamado *Cercano a un bloqueo*.

En el esquema de la figura 2.21, vemos que este bloque interactúa tanto con la base de datos (sólo tabla *remotas*) como con el almacenamiento local donde se deben guardar las cartas de bloqueo de las diferentes remotas para finalmente arrojarlos uno de tres resultados posibles indicados con los colores verde, amarillo o rojo.

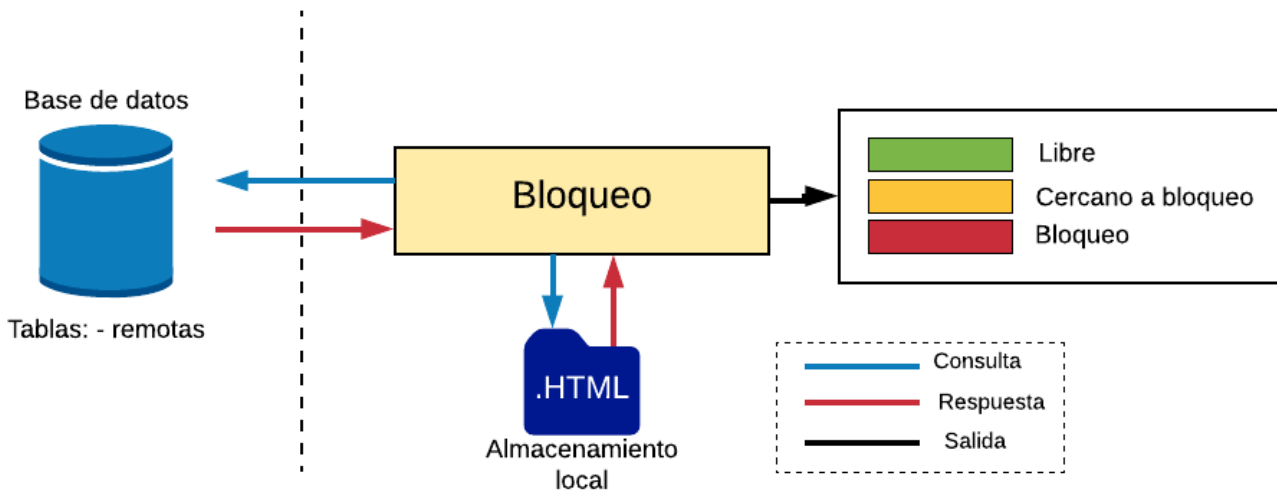


Figura 2.21: Esquema del bloque Bloqueo

Similar al bloque *Capacidad*, éste permite mostrar la Carta de Bloqueo y además conocer el estado de la línea de visión entre la antena terrestre y el satélite. Por esa razón, hemos dividido nuevamente el diagrama de flujo en dos, como se muestra en la figura 2.22.

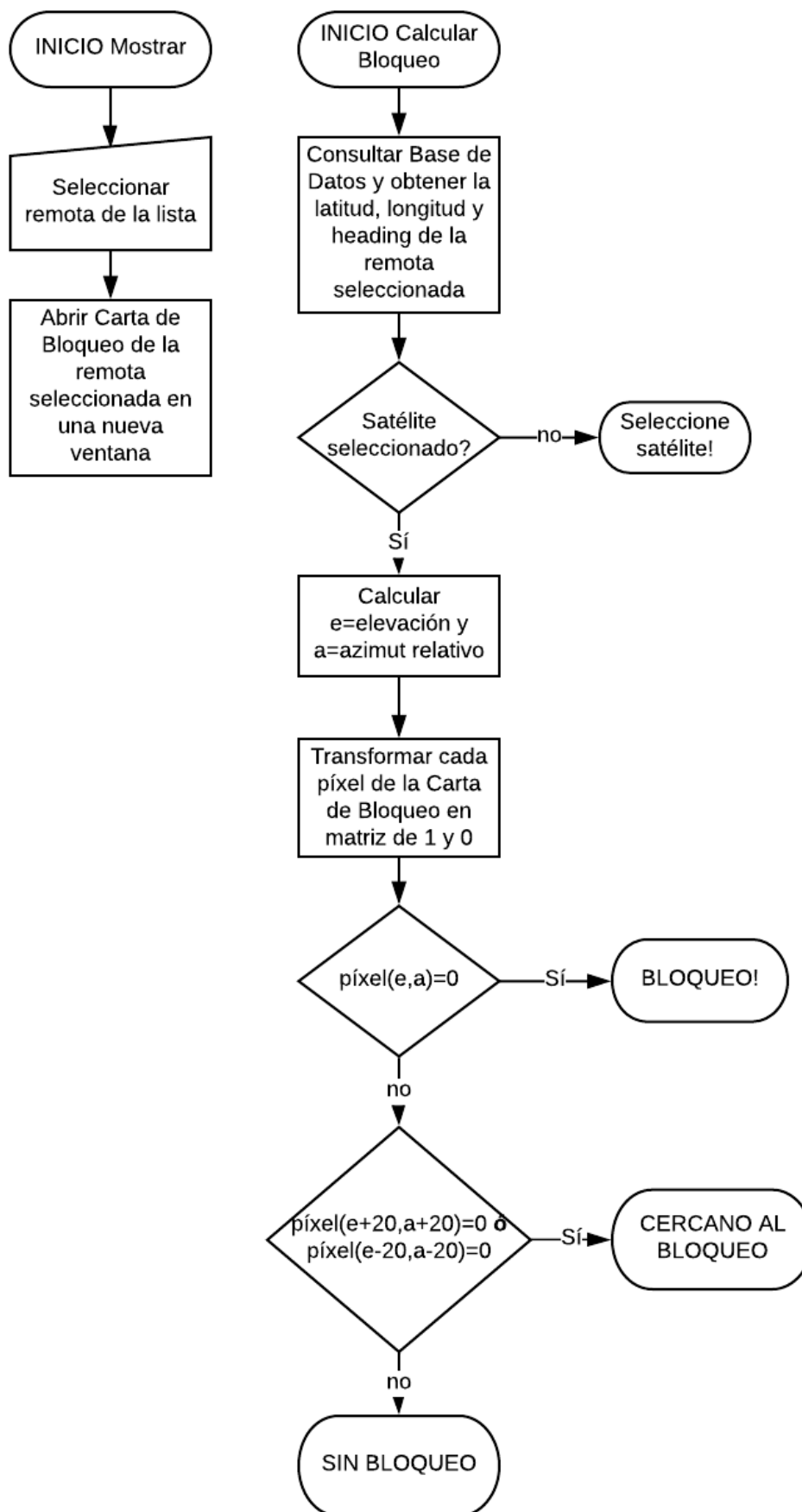


Figura 2.22: Diagrama de flujo del bloque Bloqueo

2.2.3 Interfaz con el usuario

Hemos visto hasta aquí cómo está conformada la Base de Datos y cada uno de los cuatro bloques que forman parte de la aplicación principal. Veremos en esta sección cuáles son los elementos y herramientas con los que cuenta el usuario a la hora de interactuar con la aplicación. Toda la visualización fue desarrollada gracias al paquete *Tkinter*, que es el paquete estándar de Python para el manejo de gráficos.

1) *Ventana principal*: cuando la aplicación es iniciada, nos encontramos con la ventana de la figura 2.23.

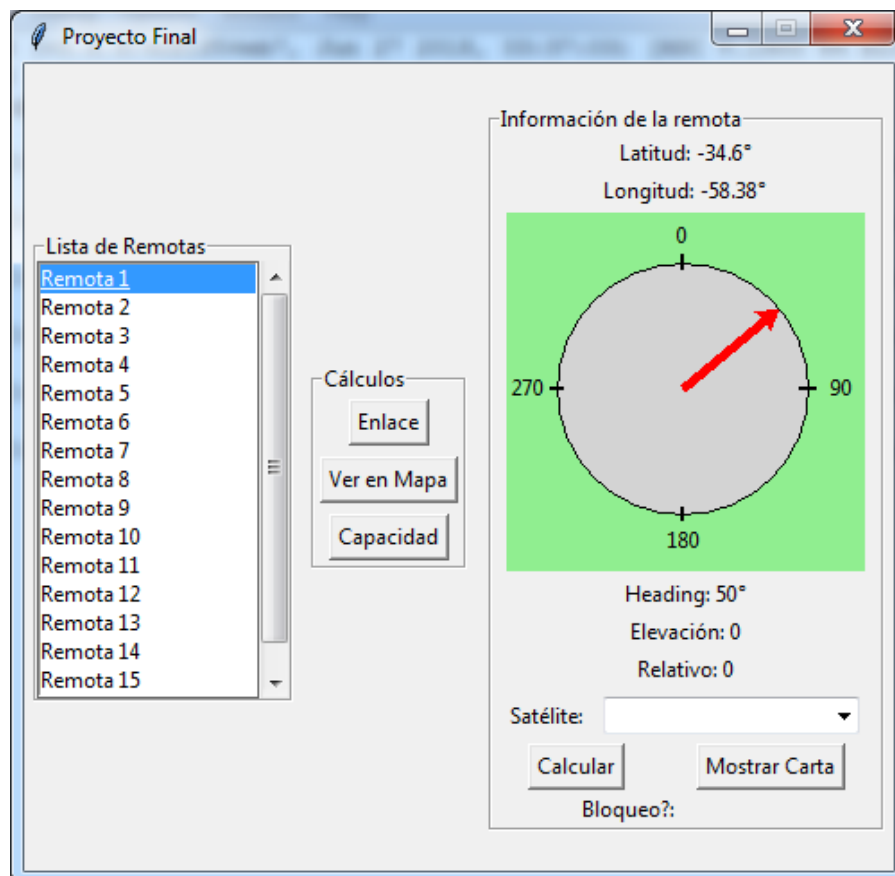


Figura 2.23: Captura de pantalla de la ventana principal

Podemos acceder a algunas de las prestaciones desde esta misma ventana o a través de diferentes botones, como veremos a continuación.

La *Lista de Remotas* se llena con el número y nombre de las remotas actualmente activas en el sistema luego de consultar la base de datos. Para los fines de este trabajo, se les ha dado un nombre que sigue una numeración consecutiva y ascendente, pero el mismo podría ser cualquiera.

Cada vez que una remota es seleccionada, veremos dentro de la caja *Información de la Remota* cierta información referida a ella: su *Latitud*, *Longitud* y *Heading*. Además, el valor de este último se ve representado por una flecha encerrada en una circunferencia con el fin de representar esta información de manera más gráfica e intuitiva.

A través del botón **Mostrar Carta** accederemos a una nueva ventana donde se nos mostrará la Carta de Bloqueo de la Remota seleccionada, la misma de la que vimos un ejemplo en la figura 2.20.

Siguiendo en la caja *Información de la Remota*, luego de elegir un satélite de la lista desplegable y de presionar el botón **Calcular** obtendremos además los ángulos de Elevación y Azimut Relativo, así como el estado del Bloqueo: Libre, Cercano o Bloqueado con los colores Verde, Amarillo y Rojo, respectivamente. Por ejemplo, en la figura 2.24 vemos el caso de la Remota 10 con los valores de latitud, longitud y heading mostrados, apuntando al satélite IS903 en situación de Bloqueo.

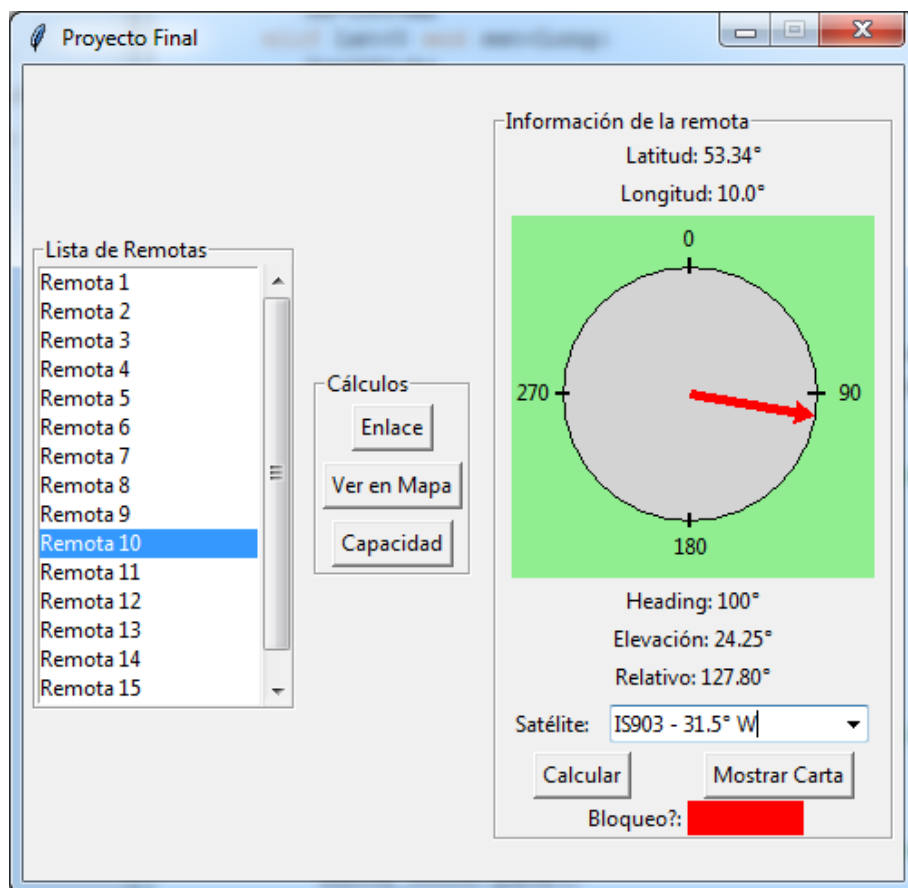


Figura 2.24: Información mostrada para el ejemplo de la Remota 10

Podemos verificar lo anterior abriendo la Carta de Bloqueo de la remota seleccionada y, usando los ángulos de Elevación y Relativo obtenidos (24.25° y 127.8°, respectivamente), comprobar que efectivamente el punto se encuentra bloqueado. La figura 2.25 muestra en color rojo la ubicación aproximada del punto de intersección de sendos ángulos.

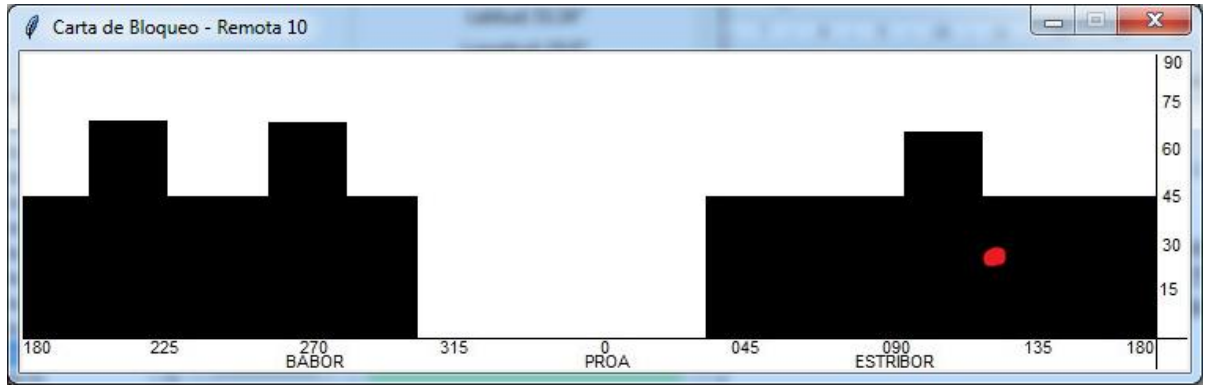


Figura 2.25: Ubicación del punto de intersección de los ángulos de elevación y azimut relativo para el ejemplo de la Remota 10

Una de las ventajas de poder visualizar la Carta de Bloqueo es que nos permite prever que tan lejos estamos de salir de una zona de bloqueo o, por el contrario, de entrar en una.

Hasta aquí hemos visto las herramientas u opciones a las que tenemos acceso desde la ventana principal. A continuación, y en los próximos puntos, veremos las herramientas a las que podemos acceder a través de los tres botones centrales:

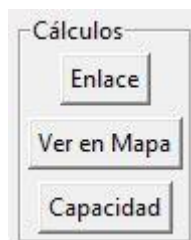


Figura 2.26: Conjunto de botones para acceder a nuevas ventanas.

II) *Ventana Cálculo de enlace*: al presionar en el Botón Enlace de la Ventana Principal, se abrirá una nueva ventana, como la mostrada en la figura 2.27.

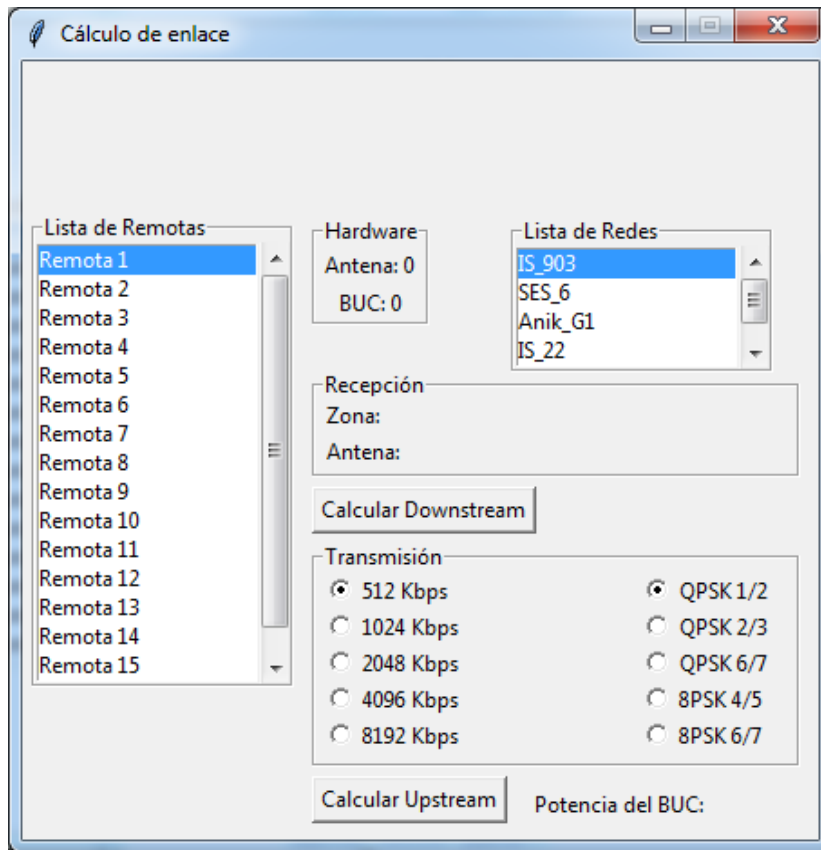


Figura 2.27: Ventana para el cálculo de enlace

Como se podrá deducir, esta ventana está destinada a analizar todo lo concerniente al cálculo de enlace o análisis de potencia, como vimos anteriormente. Tenemos nuevamente una lista de remotas y, además, una con las redes disponibles. Similar al punto anterior, cada vez que una remota es seleccionada, la base de datos será consultada y obtendremos información referida a ella. En este caso esa información estará relacionada con el hardware presente en la remota (tamaño de antena y BUC), como puede verse en la caja *Hardware*.

Luego, tenemos dos zonas bien diferenciadas dentro de esta ventana: una destinada a mostrar información sobre la *Recepción* – acompañada al botón

- y otra a la *Transmisión*, acompañada del botón .

Dentro de la caja *Recepción* veremos dentro de cuál de las zonas se encuentra nuestra remota (1, 2 o 3) o si se encuentra fuera de cobertura. La zona 1 aparecerá como “Excelente”, la zona 2 como “Buena” y la zona 3 como “En la periferia”. Al mismo tiempo, podremos saber cuál es el tamaño mínimo de antena necesario para recibir correctamente la señal proveniente del satélite. Por ejemplo, en la figura 2.28 vemos el ejemplo de una remota que se encuentra en la zona 1 (“Excelente”) de la pisada del satélite IS 903 y su antena de 0.98 metros es suficiente ya que esta zona requiere un tamaño mínimo de 0.5 m para recibir correctamente la señal.

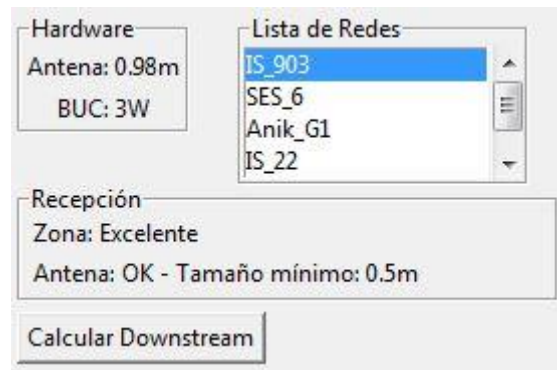


Figura 2.28: Remota dentro de Zona 1

Por otro lado, en la figura 2.29 vemos un ejemplo de una remota que se encuentra en zona 2 (“Buena”) y su antena es muy pequeña ya que es de 0.75 m cuando esta zona requiere un mínimo de 0.85 m.

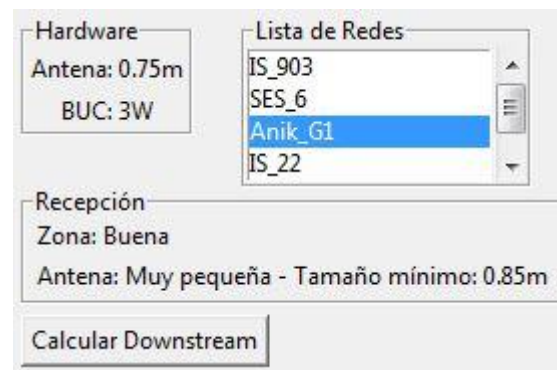


Figura 2.29: Remota dentro de Zona 2

Finalmente, en la figura 2.30 tenemos un ejemplo para una remota que se encuentra fuera de la zona de cobertura para el satélite seleccionado.

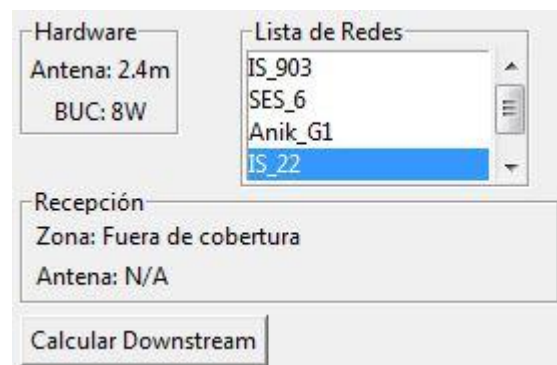


Figura 2.30: Remota fuera de la zona de cobertura

Dentro de la caja *Transmisión* vemos dos columnas que muestran, por un lado, la opción de elegir entre 5 tasas de información o *data rates* diferentes y, por el otro, 5 MODCOD (acrónimo de MODulation and CODing) distintos. El índice de modulación nos dice cuántos bits por símbolo puedo transmitir; así, una modulación QPSK tiene un índice de modulación igual a 2 y permite transmitir 4 bits (2^2) por símbolo. En el mismo sentido, una modulación 8PSK tiene un índice de modulación igual a 3 y permite transmitir 8 bits (2^3) por símbolo. La codificación – o también conocida como FEC (por sus siglas en inglés *Forward Error Correction*) – consiste en agregar bits redundantes a la información útil con

el fin de detectar y corregir errores causados por el canal de transmisión. Tanto el índice de modulación como el esquema de codificación utilizado en una señal nos da una idea de la eficiencia en materia de ancho de banda utilizado. Cuanto mayor sea el índice de modulación y FEC, menor será el ancho de banda requerido, con el agravante de que necesitaremos transmitir una mayor potencia de manera tal de lograr un mayor C/N que permita decodificar correctamente el mayor número de bits. En este trabajo, se han elegido 5 de los data rates y MODCOD más utilizados en la práctica y los límites usados en los cálculos están basados en portadoras TDMA utilizadas en sistemas iDirect [1].

Una vez seguros de que nuestra antena será capaz de recibir eficientemente la señal proveniente del satélite, podemos proceder a elegir de entre los data rate y MODCOD que mejor se asemejen al caso real para luego, al presionar el botón

Calcular Upstream descubrir si con el BUC disponible, seremos capaces de transmitir ese tipo de señal.

Por ejemplo, en la figura 2.31 vemos el caso de una remota dentro de zona 1 donde su tamaño de antena (1.2 m) es adecuada para recibir correctamente la señal del satélite, pero la potencia de su BUC (3W) es demasiado pequeña para transmitir la señal elegida (4096 kbps – QPSK 6/7) ya que la misma necesitaría 6,13 W.

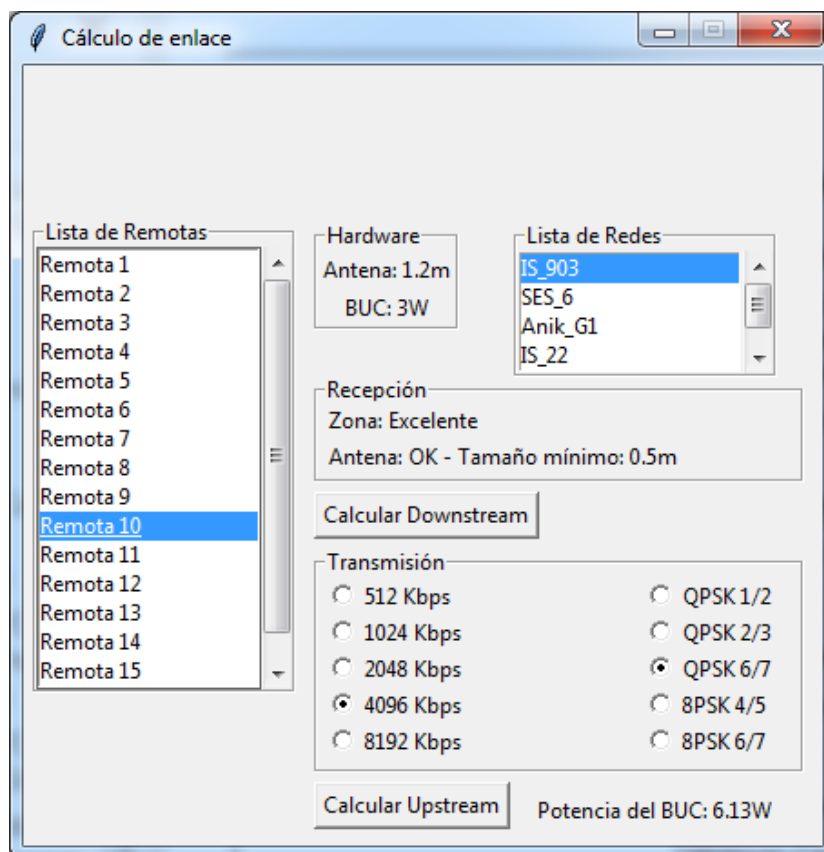


Figura 2.31: Remota con excelente recepción, pero potencia de BUC insuficiente

Aquí tenemos dos opciones posibles: o cambiar el BUC por uno de mayor potencia (opción generalmente no disponible en el corto plazo) o bien modificar las características de la portadora ya sea modificando su data rate ó MODCOD (o ambos) implicando esto último una merma en la eficiencia del sistema, ya que seremos capaces de transmitir una menor cantidad de bits por Hz al tiempo que aumentamos el ancho de banda de transmisión.

En la figura 2.32 vemos que esto se ha resuelto llevando el data rate a un valor de 2048 kbps y el MODCOD a QPSK 2/3, necesitando ahora un BUC de 1.65 W.

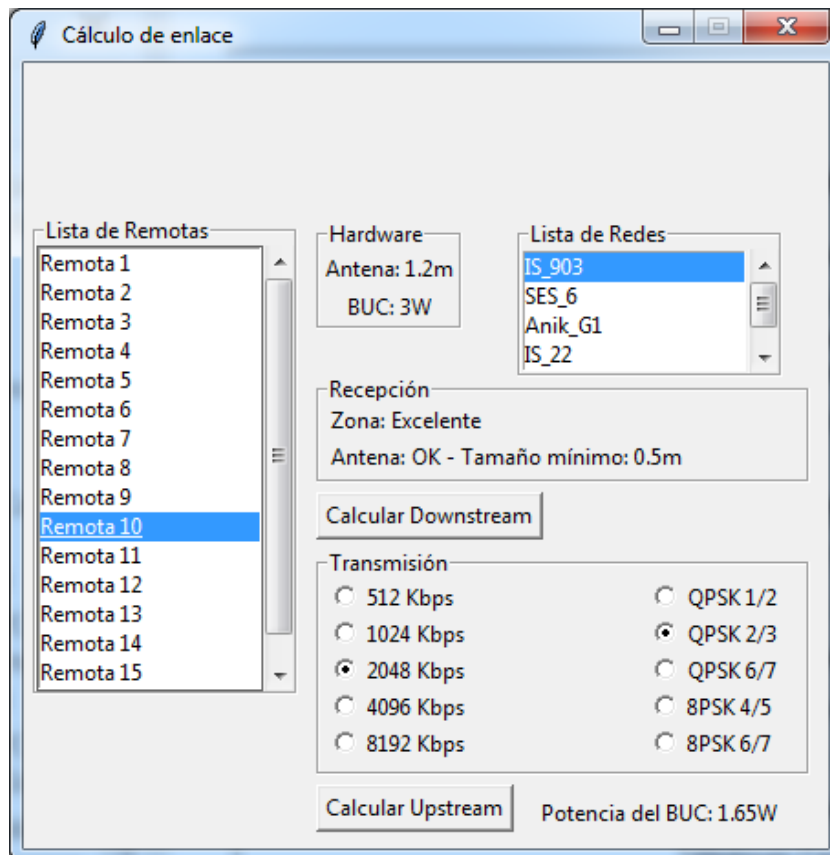


Figura 2.32: Remota con excelente recepción y potencia de BUC adecuada

Lo visto en esta última parte también nos sirve para estimar si una mejora en el ancho de banda (aumento del valor del data rate) podrá ser lograda con el BUC instalado.

III) *Ventana Ver en Mapa*: no nos detendremos demasiado en este punto ya que el aspecto y funcionalidad de esta ventana ya fue visto en el punto I) *Posición geográfica* de la sección 2.2.2. Podemos agregar que no se trata de una ventana como las anteriores, sino que es una ventana (o pestaña) de un navegador web. Luego de seleccionar una remota de la Ventana Principal y presionar el botón Ver en Mapa, se nos dirigirá a un navegador web donde veremos la aplicación Google Maps y nuestra remota indicada a través de un ícono como se muestra en la figura 2.12. Si bien es cierto que tanto el valor numérico de la latitud como de la longitud de la remota podemos verlo en la Ventana Principal, consideramos que el impacto de una imagen a la hora de brindar información referido a la ubicación geográfica es mucho más eficaz.

IV) *Ventana Capacidad*: al presionar sobre el botón Capacidad, se abrirá una ventana como la de la figura 2.33.

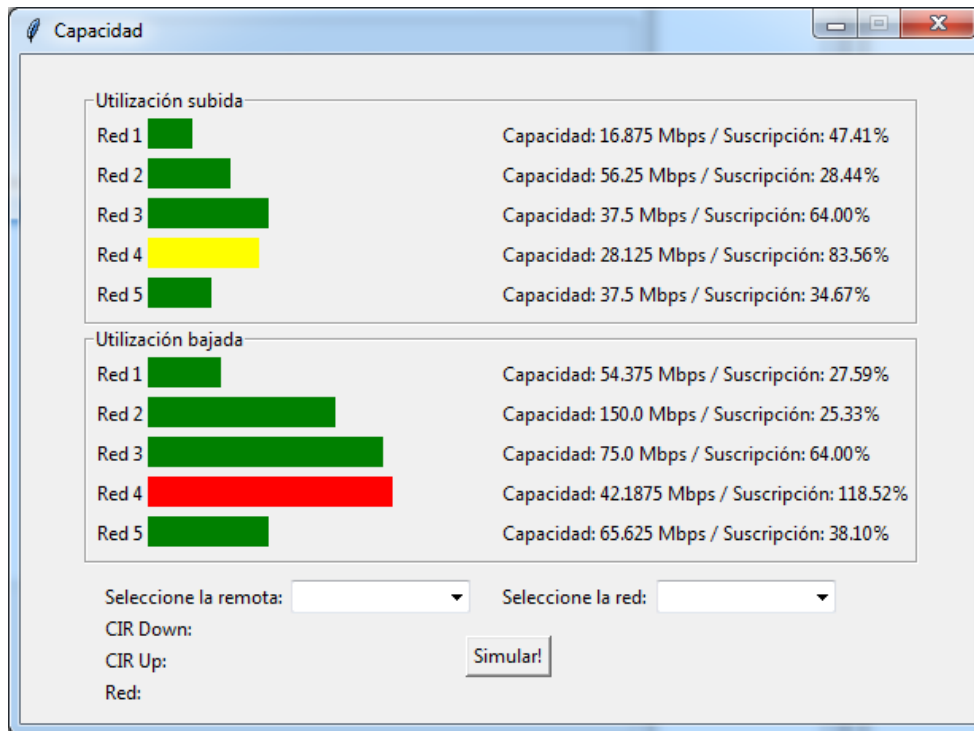


Figura 2.33: Ventana Capacidad

Esta ventana posee básicamente dos funciones, como ya hemos mencionado. La primera es mostrar información relevante a las redes como su capacidad en Mbps y el porcentaje de suscripción y la otra es la de proveer una herramienta que permita simular diferentes escenarios a través del movimiento o traspaso de remotas de una red a otra.

Antes de analizar la función de los elementos de una ventana, veamos de qué manera debe ser interpretada la información mostrada. En principio, vemos dos cajas dedicada una a la utilización en la subida y otra a la utilización en bajada, siendo el análisis el mismo para cada una. Vemos que en este caso tenemos 5 redes y al lado de cada una, una barra de largo variable, el cual está dado por la suma del CIR de las remotas registradas en esa red. Así, por ejemplo, la suma del CIR de las remotas registradas a la Red 2, es mayor que el de las registradas en la Red 1. Además, tenemos también información acerca de la Capacidad y Suscripción de cada red, tanto para la subida como para la bajada. Es importante mencionar que, en todos los casos analizados aquí, la Capacidad en Mbps mostrada en esta ventana es calculada considerando el ancho de banda en MHz contratado o disponible en el satélite referido a un MODCOD nominal o promedio igual a 8PSK $\frac{3}{4}$, por lo que el valor de la Capacidad, así como el de la Suscripción, deben ser tomados como referencia y no como un valor exacto.

Toda la información utilizada en los cálculos y la mostrada en esta ventana, es extraída directamente de la base de datos y se actualiza cada vez que abrimos esta ventana.

Veamos ahora cómo funciona el esquema de simulación. Cada vez que una remota es seleccionada desde la lista desplegable, obtendremos información acerca de su CIR y veremos en qué red se encuentra funcionando. Veamos un ejemplo en la figura 2.34 para la Remota 10.

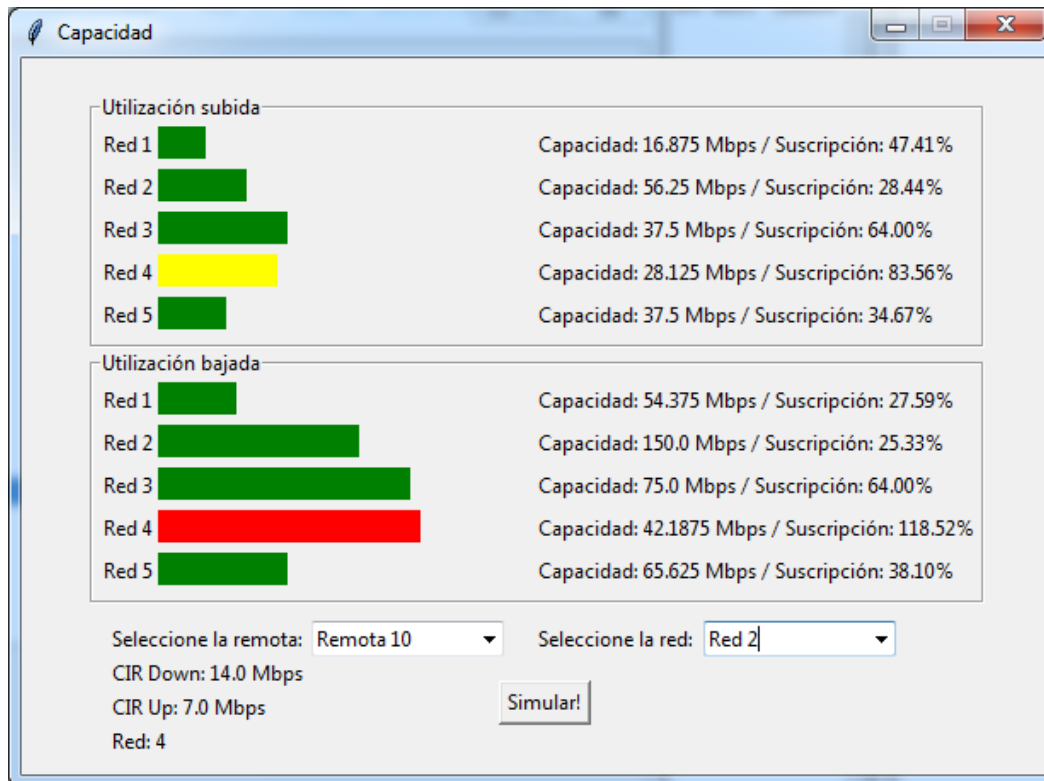


Figura 2.34: Ejemplo de simulación para la Remota 10

Podemos ver que esta remota posee un CIR de 14.0/7.0 Mbps y se encuentra actualmente trabajando en la Red 4 la cual podemos ver que está altamente utilizada, sobre todo en bajada. Aquí, podríamos proceder a hacer un traspaso desde esta red hacia otra con menores niveles de suscripción. Por ejemplo, si luego de comprobar que la remota se encuentra bajo cobertura de la Red 2 (a través del mapa), no posee bloqueo a ese satélite (analizando su Carta de Bloqueo) y tanto su antena como su BUC cumplen los requerimientos tanto de tamaño como de potencia (cálculo de Enlace), podemos seleccionar esa red desde la lista desplegable donde están las redes y luego de presionar el botón veremos el nuevo estado de las redes. Como se puede ver en la figura 2.35, el nuevo valor de suscripción de la red 4 y 2 ha sido actualizado, siendo menor para la primera y mayor para la última.

Es importante mencionar que estos movimientos de remotas de una red a otra solo sirven a modo de simulación y no serán cambiadas de red a menos que el operador lo realice a través de otros medios, ajenos a este trabajo.

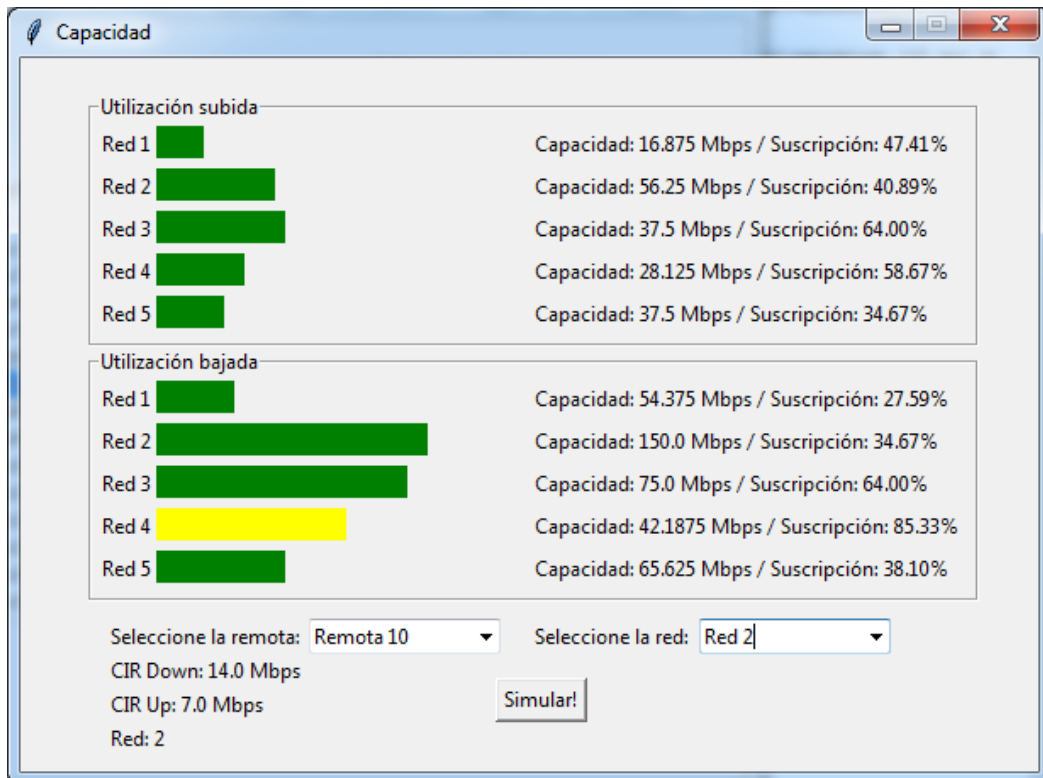


Figura 2.35: Resultado de la simulación de la Remota 10

2.3 Consideraciones acerca de la implementación

En la figura 2.36 vemos un esquema general de todo el trabajo donde tenemos, por un lado, la base de datos con sus dos tablas y por el otro la aplicación principal y su conjunto de herramientas.

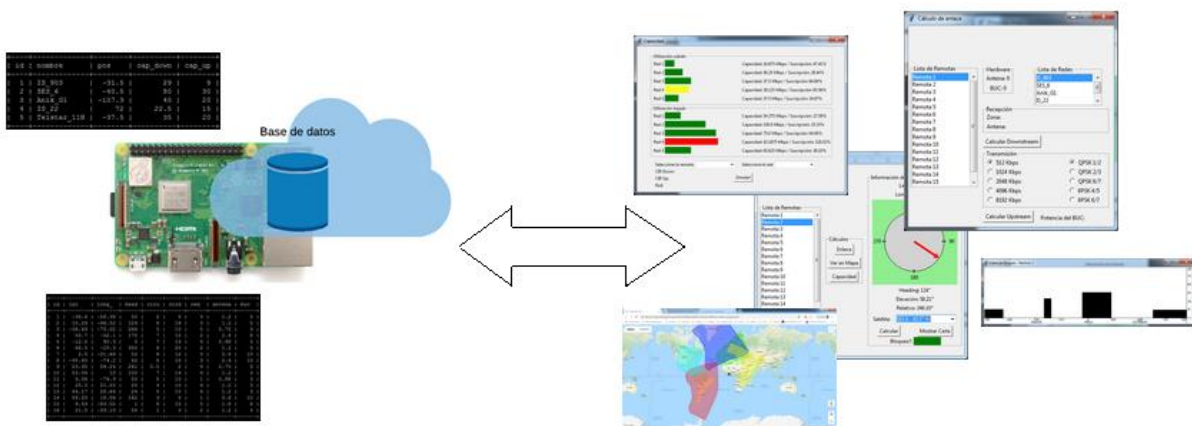


Figura 2.36: Esquema general del desarrollo completo

Veremos a continuación una serie de consideraciones que en mayor o menor medida deberán ser tenidas en cuenta a la hora de implementar o poner en producción el presente desarrollo.

2.3.1 Consideraciones sobre la base de datos

Si bien todo el entorno de la base de datos para este trabajo fue simulado y las condiciones pueden variar a la hora de implementarlo en un escenario real donde la base de datos es otra, mencionaremos algunos puntos que deben ser tenidas en cuenta para poder acceder y trabajar con la de este trabajo.

Como mencionamos, la base de datos está montada físicamente en una SBC Raspberry Pi (ver figura 2.37) y en el enrutador donde está conectada, el puerto 3306 debe ser abierto, asignado y reenviado hacia la dirección IP de LAN que hayamos configurado en la Raspberry Pi (en nuestro caso: 192.168.1.73). A fin de evitar o minimizar el ataque de agentes externos, es recomendable cambiar el número de puerto 3306 por otro que se encuentre libre.

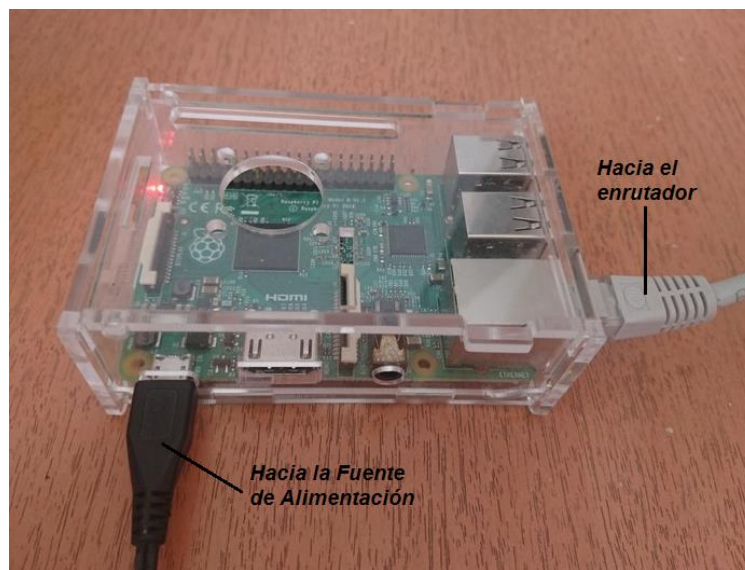


Figura 2.37: Fotografía de la Raspberry Pi utilizada en este trabajo

Si queremos acceder a la base de datos desde el exterior, el usuario que creamos en el entorno SQL no debe contener ninguna dirección IP en particular (tampoco *'localhost'*) en la sección de *host*, sino que debemos garantizar el acceso a todas las direcciones. Entonces, deberá ser del tipo *'usuario'@'%'*, donde *"%"* es el equivalente a decir "todas las direcciones". Los privilegios de acceso a las diferentes tablas dado a los usuarios dependen exclusivamente del administrador. Solo diremos que, para los fines de este trabajo, hemos creado un único usuario con acceso remoto ilimitado a todas las tablas de la base de datos.

Una vez que el usuario ha sido creado y los privilegios fueron dados, deberemos configurar MySQL para que acepte conexiones externas a la Raspberry Pi. Para ello, deberemos modificar el archivo de configuración generalmente localizado en la carpeta */etc/mysql*. Debemos comentar la línea *"bind-address"* agregando el símbolo de numeral al comienzo de la misma, de la siguiente manera:

```
#bind-address = 127.0.0.1
```

2.3.2 Consideraciones sobre la aplicación principal

Debido a que la aplicación principal de carácter portable, su instalación no presenta mayores inconvenientes debiendo estar la misma (archivo *.exe* para el caso de Windows o *.py* para el caso de Linux) en la misma carpeta que el conjunto de Cartas de Bloqueo en

formato JPG con el tamaño ya mencionado junto con el archivo de extensión HTML que contiene pre cargadas las zonas de cobertura de las redes. En la figura 2.38 vemos como estaría conformada la carpeta que contiene el conjunto de archivos para el caso de la plataforma Windows. Para una mejor interpretación, encerramos en un recuadro las carpetas y archivos mencionados.

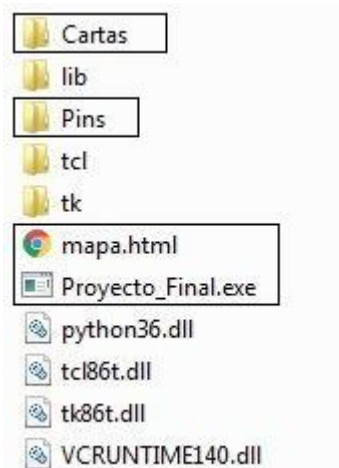


Figura 2.38: Contenido de la carpeta de la aplicación para la plataforma Windows

La carpeta *Cartas* contiene, como mencionamos, las Cartas de Bloqueo y para que la aplicación sea capaz de reconocerlas y asociarlas a una remota en particular, el nombre de las mismas debe estar formado solamente por un número que sea igual al identificador con el que la remota está registrada en la base de datos.

La carpeta *Pins* no es estrictamente necesaria y en este caso es utilizada para almacenar diferentes tipos de marcadores para representar a las remotas en Google maps. Si no está presente, el marcador utilizado será el usado por defecto (ver figura 2.39)



Figura 2.39: Marcador de Google Maps. Recuperado de <https://productforums.google.com/forum/#!topic/maps-es/>

Las demás carpetas y archivos son solo el producto de la acción del programa de conversión de la aplicación a un formato ejecutable para Windows y son transparentes para el usuario final.

Capítulo 3: Resultados

Como se ha venido mencionando a lo largo de las secciones anteriores, la principal ventaja del presente desarrollo consiste en facilitar el análisis integral de redes satelitales, evitando así tener que recurrir a múltiples y descentralizadas herramientas que no hacen más que ralentizar los procesos implicados en la determinación de la mejor red.

A través de una aplicación portable desarrollada en lenguaje Python, con un tamaño de aproximadamente 40 Mbytes y con probada compatibilidad con los sistemas operativos Windows (Windows 7) y Linux (Ubuntu 18.04.1 LTS), logramos establecer una conexión con una base de datos remota y brindar una interfaz gráfica a través de la cual interactuamos con los bloques ya vistos.

En la práctica y en la cotidianeidad del trabajo con redes satelitales, hemos podido comprobar luego de nuestra experiencia laboral en el sector, que de los cuatro bloques analizados en este trabajo (posición geográfica, capacidad, cálculo de enlace y carta de bloqueo), solo el del cálculo de enlace no se realiza sino en casos excepcionales y, los ejemplos analizados durante el desarrollo del presente trabajo, arrojaron resultados que coinciden con bastante precisión con lo hecho utilizando otras herramientas similares ya existentes (existen infinidad de herramientas para cálculos de enlace, algunas facilitadas por el mismo proveedor satelital, como por ejemplo el programa LST5 de Intelsat, y otras de terceras partes muchas de ellas basadas en planillas de cálculo).

Del resto de los bloques (posición geográfica, capacidad y carta de bloqueo), podemos decir que en el sector profesional donde nos tocó desempeñarnos, las mismas se encuentran descentralizadas y muchas veces se debe recurrir a cálculos indirectos o herramientas un tanto rudimentarias para llegar a un resultado.

Debido a que cada uno de los bloques están en contacto casi permanente con la base de datos, podemos notar cierto retardo o latencia a la hora de interactuar con la interfaz, aunque son valores esperables y normales dentro del marco de toda comunicación con un servidor remoto.

En cuanto a la interfaz con el usuario, consideramos que su estructura de ventanas nos brinda un fácil manejo del conjunto de las herramientas y tanto la comunicación con el almacenamiento local, como con la base de datos no presenta mayores inconvenientes, salvo los de latencia mencionados anteriormente.

Capítulo 4: Análisis de Costos

Debido a la falta de componentes o materiales en el presente proyecto, el análisis de costos quedará reducido entonces a la estimación del costo de las horas-hombre dedicadas al desarrollo, así como al del lenguaje de programación y/o cualquier paquete o extensión utilizada en el mismo. Dado que el servidor de la base de datos que finalmente será usada forma parte de una infraestructura ya existente, no lo consideraremos en este análisis. De todas maneras, incluiremos el costo de la plataforma utilizada para la realización de este trabajo, pero se debe tener en cuenta que el mismo no formará parte del costo final.

El lenguaje Python utilizado en casi la totalidad de este proyecto (salvo una pequeña parte desarrollada en HTML) posee una licencia de código abierto (Licencia Pública General de GNU) y forma parte de la comunidad de *software libre*, lo que implica, entre otras cosas, que su adquisición y uso no tiene costo alguno. En lo que respecta al desarrollo de la aplicación principal, hay un costo que debe ser tenido en cuenta y es el de la API (Interfaz de Programación de Aplicaciones) de Google Maps vista en el punto I de la sección 2.2.2 y que utilizamos en este trabajo para poder visualizar tanto las huellas de los satélites como la posición de las remotas. Si bien al momento de la realización del presente trabajo, estamos dentro de un plazo de prueba gratuito de 12 meses, hay muchos planes con precios basados en la utilización y también podemos obtener precios adaptados para empresas luego de contactar al equipo de ventas de Google. Como referencia, podemos utilizar el precio proporcionado por Google con el que nos garantiza que, para la mayoría de los usuarios, 200 pesos argentinos (ARS) mensuales son suficientes para cubrir sus necesidades.

Con respecto al precio de las horas-hombre, según un artículo periodístico de marzo de 2018 [2], el sueldo promedio en Argentina para un programador Junior es de 22.000 ARS.

En la tabla 4.1 presentamos la información en forma más clara y completa. Hemos dividido la misma en dos partes, una dedicada a analizar los costos incurridos para la realización del presente trabajo y la otra analizando los costos a los que eventualmente nos veremos atados en el caso de que el desarrollo entre en producción por el período de un año. Si bien, durante la realización del presente trabajo no nos adecuamos a un régimen estricto de determinadas horas diarias, hemos considerado un tiempo promedio de trabajo de cinco meses.

<i>*Precios indicados en pesos argentinos (ARS)</i>			
<i>**Cantidad expresada en meses</i>			
	Proyecto		
	Precio unitario*	Cantidad	
Infraestructura de Base de Datos (Raspberry Pi)	1500,00	1	1500,00
Software (API's)	0,00	1	0,00
Horas-hombre	22.000,00	5**	110000,00
		TOTAL * =	111500,00
	En Producción		
	Precio unitario*	Cantidad	
Infraestructura de Base de Datos (Raspberry Pi)	0,00	1	0,00
Software (API's)	200,00	12**	2400,00
Horas-hombre	0,00	0	0,00
		TOTAL * =	2400,00

Tabla 4.1: Análisis de Costos

Como vemos en la tabla anterior, una vez que el proyecto entra en producción, solo tendrá el costo fijo de la manutención mensual de la API de Google. En cuanto al costo del Proyecto (111,500 ARS), consideramos que el mismo es extremadamente bajo y, teniendo en cuenta la dinámica y los números que se manejan en el negocio satelital, el mismo podrá ser amortizado en pocos meses. Debido a su carácter global, este desarrollo puede ser implementado – y por lo tanto vendido – en cualquier parte del mundo donde se requiera un control centralizado de parámetros satelitales.

Capítulo 5: **Discusión y Conclusión.**

Teniendo en cuenta de que la idea de este trabajo nació debido a la necesidad de brindar un servicio que integre las herramientas más utilizadas por los técnicos u operadores encargados de administrar las redes satelitales, consideramos que los resultados obtenidos podrán ser evaluados plenamente una vez que el mismo entre en producción. De todas maneras, creemos que la misma viene a ocupar un lugar vacante en lo que hace a este tipo de desarrollos ya que, luego de una experiencia de casi 5 años en el sector, no hemos visto ni utilizado una herramienta de iguales características.

En la actualidad, para llegar al mismo resultado, los operadores deben recurrir a diferentes programas en ejecución simultánea, lo que muchas veces implica tener abiertas muchas ventanas de esos programas y tener que utilizar hasta 3 pantallas o monitores de manera de tener frente a ellos la información de forma clara.

En lo que respecta al análisis de Capacidad visto en el punto II de la sección 2.2.2, si bien existen infinidad de programas o aplicaciones que miden el ancho de banda disponible y la suscripción, no encontramos uno que permita también la simulación.

La posibilidad de prever o estimar si la remota será capaz de trabajar en una red determinada en términos de potencia, es otra de las innovaciones del presente desarrollo. Comúnmente, este punto no es tenido en cuenta y solo se evalúa una vez que las demás posibilidades fueron agotadas. Si bien el presente trabajo basa parte de los cálculos en estimaciones, como mencionamos en el capítulo 3, los resultados obtenidos coinciden con bastante precisión y puede servir como un punto de partida a la hora de estar buscando la solución a un problema.

Consideramos que uno de los puntos más fuertes de este trabajo es la posibilidad de analizar las Cartas de Bloqueos de las remotas y poder anticipar si la línea de visión a un satélite en particular se encuentra bloqueada o no. Hoy en día, lo mismo se realiza indirectamente a través de planillas de cálculo como Excel.

Como puntos a mejorar, si bien la gestión de diferentes usuarios es posible, la misma debe ser realizada por el administrador; creemos que el agregado de una etapa de autenticación a cargo del usuario final sería conveniente a la hora de acceder a la base de datos. Además, se podría ampliar el rango de frecuencias en el Cálculo de Enlace para que abarque también a la banda C, muy usada en comunicaciones satelitales. En lo que hace a la visualización de las remotas y las huellas satelitales en el mapa, no estaría mal incorporar la opción de ver todas las remotas simultáneamente, así como activar o desactivar las zonas de cobertura para ver solo aquellas que deseamos.

Capítulo 6: Literatura Citada.

[1] VT iDirect Inc., (2015), *iDX 3.3 – Technical Reference Guide*, Estados Unidos

[2] Archivo La Nación, (22 de marzo de 2018), ¿Cuánto cobra un programador en la Argentina en 2018?, *La Nación*, versión online.

- Roddy, D., (2001), *Satellite Communications*, Estados Unidos, McGraw-Hill.
- INTELSAT, (1999), *Earth Station Technology*, Washington, Estados Unidos, INTELSAT'S Assistance and Development Program.
- VT iDirect, (2014), *Link Budget Analysis Guide*, Estados Unidos
- Van Rossum, G., (2009), *El tutorial de Python*
- Teran Jibaja, C.D., (1998), *Gestión en redes satelitales* (Tesis de Grado), Facultad de Ingeniería Eléctrica, Quito.
- VT iDirect Inc., (2015), *iDX 3.3 – Link Budget Analysis Guide*, Estados Unidos
- Oracle Corporation, (2011), *MySQL 5.0 Reference Manual*, Estados Unidos.
- Salnikov, A., (2017), TkInter. Lugar de publicación: python.org.
<https://wiki.python.org/moin/TkInter>
- Lawhead, J., (2011), Lugar de publicación: GeospatialPython.com,
<http://geospatialpython.com/2011/01/point-in-polygon.html>
- Akshay, P., (2016), *Extracting pixel values of an image in Python*.
<https://www.hackerearth.com/practice/notes/extracting-pixel-values-of-an-image-in-python/>
- Google Inc., *Google Maps Platform*, <https://cloud.google.com/maps-platform/>
- Wright, S., (2014), *Install MySQL server on Raspberry Pi*, Lugar de publicación: Ste Wright, <https://www.stewright.me/2014/06/tutorial-install-mysql-server-on-raspberry-pi>