



Teoría de la Señal, Telemática y Comunicaciones:

Una aproximación al
conocimiento a través de los
Trabajos Fin de Grado y Máster

Curso 2014/2015

Luz García Martínez
Jorge Navarro Ortiz
Editores

Teoría de la Señal, Telemática y Comunicaciones:

***Una aproximación al conocimiento a través de los
Trabajos Fin de Grado y Máster***

Curso 2014 / 2015

Editores:

Luz García Martínez

Jorge Navarro Ortiz



Departamento de
Teoría de la Señal,
Telemática y
Comunicaciones

ISBN-13: 978-84-617-5449-6

Editores: Luz García Martínez y Jorge Navarro Ortiz (Dpto. Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada)

El contenido de los trabajos que componen este libro es propiedad de los autores de los mismos y está protegido por los derechos que se recogen en la Ley de Propiedad Intelectual. Los autores autorizan la edición de este libro y su distribución, sin que esto, en ningún caso, implique una cesión a favor de la Universidad de Granada de cualesquiera derechos de propiedad intelectual sobre los contenidos de los trabajos. Ni la Universidad de Granada, ni los editores, serán responsables de aquellos actos que vulneren los derechos de propiedad intelectual sobre estos trabajos.

© 2016, los autores

Portada: Pilar Andrés Maldonado

Maquetación: Luz García Martínez y Jorge Navarro Ortiz

Presentación

El Real Decreto (RD 1393/2007) que regula la ordenación de las enseñanzas oficiales, establece que tanto el Trabajo Fin de Grado (TFG) como el Trabajo Fin de Máster (TFM) deben de estar orientados a la evaluación de competencias asociadas al título. Para cubrir la oferta formativa, el Dpto. de Teoría de la Señal, Telemática y Comunicaciones, que imparte docencia en el grado de Ingeniería de Tecnologías de Telecomunicación y en el máster de Ingeniería de Telecomunicación, oferta cada año a los alumnos un considerable número de posibles trabajos en los campos del procesado de señal, redes, telemática o comunicaciones.

En esta línea, y con el objetivo de motivar al alumnado el Dpto. creó los premios a los TFG y TFM para aquellos proyectos que se realizan y se presentan en el seno del Dpto. Este libro, titulado *“Teoría de la Señal, Telemática y Comunicaciones: una aproximación al conocimiento a través de los Trabajos Fin de Grado y Fin de Máster”*, recoge los resúmenes de los mejores TFG y TFM presentados a la tercera convocatoria de los premios correspondiente al curso 2014-2015.

Para esta edición, se han seleccionado un total de 13 trabajos que abarcan el campo del procesado de señal, las comunicaciones móviles, las redes de ordenadores y la ciberseguridad; estos trabajos constituyen un buen ejemplo del desempeño realizado por los alumnos que cursan sus estudios de Ingeniería de Telecomunicación en la Universidad de Granada.

Para finalizar, sirvan también estas líneas para agradecer a los alumnos participantes y a los profesores la labor que realizan que nos permite seguir avanzando con el objetivo de mejorar la sociedad.

M. Carmen Benítez Ortúzar

Directora del Dpto. de Teoría de la Señal, Telemática y Comunicaciones

ETS Ingenierías Informática y de Telecomunicación

Universidad Granada

Índice de contribuciones

Áreas de Teoría de la Señal y Comunicaciones e Ingeniería Telemática

Comunicaciones móviles

Prototipo de estación base GSM haciendo uso de OpenBTS.....	3
<i>A. González Garrido (tutores J. Navarro Ortiz y J.C. Segura Luna)</i>	

Área de Teoría de la Señal y Comunicaciones

Procesado de señales bioquímicas

Clasificación de proteínas representadas mediante señales extraídas de índices bioquímicos	11
<i>A.O. Villegas Morcillo (tutores V. Sánchez Calle y A.M. Peinado Herreros)</i>	

Procesado de señales sísmicas

Application of the method of instantaneous power in automatic picking of synthetic explosions in the experiment TOMO-ETNA	17
<i>F.J. Cuberos Muñoz (tutoras M.C. Benítez Ortúzar y L. García Martínez)</i>	

Picking de terremotos sismo-volcánicos y su aplicación a la tomografía computerizada en el experimento TOMO-ETNA.....	23
<i>D. García Sánchez (tutoras M.C. Benítez Ortúzar y L. García Martínez)</i>	

Uso de métodos de polarización instantánea avanzados para la mejora de la fase de llegada de señales sísmicas	29
<i>I. Vico Triviño (tutoras M.C. Benítez Ortúzar y L. García Martínez)</i>	

Área de Ingeniería Telemática

Aplicaciones en red

Diseño de un juego con requisitos de tiempo real para redes móviles.....	37
<i>P. Peña Martínez (tutores J.J. Ramos Muñoz y J.M. López Soler)</i>	

Implementación de una aplicación cliente para Chromecast	43
<i>I. Herrera López (tutor J. Navarro Ortiz)</i>	

Redes definidas por software

Análisis de redes SDN utilizando Mininet e implementación de un Deep Packet Inspector	49
<i>M. Sánchez López (tutor J. Navarro Ortiz)</i>	

Diseño y evaluación de un servicio OpenFlow de provisión de Calidad de Experiencia sobre Mininet	55
<i>C.A. Prieto Sánchez (tutor J.J. Ramos Muñoz)</i>	

Redes móviles

Soporte para comunicaciones máquina a máquina en sistemas 5G.....	61
<i>P. Andrés Maldonado (tutor P. Ameigeiras Gutiérrez)</i>	

Seguridad en redes

Análisis de malware en Smart TV: ataques y defensas	67
<i>J. López Arredondo (tutor P. García Teodoro)</i>	
Diseño e implementación de retos de seguridad en redes de telecomunicación en el laboratorio 3.7 con fines docentes	73
<i>F. López Pérez (tutor R.A. Rodríguez Gómez)</i>	
MEDA-toolbox: interfaz gráfica en MATLAB para la detección de anomalías en red	79
<i>P. Sánchez Robles (tutor J. Camacho Páez)</i>	

Teoría de la Señal y Comunicaciones e Ingeniería Telemática

Prototipo de estación base GSM haciendo uso de OpenBTS

Autor: Alejandro González Garrido, e-mail: weborro@correo.ugr.es

Tutores: Jorge Navarro Ortiz; e-mail: jorgenavarro@ugr.es

José Carlos Segura Luna; e-mail: segura@ugr.es

Titulación: Ingeniería de Telecomunicación

Departamento de Teoría de la Señal, Telemática y Comunicaciones

Universidad de Granada

Resumen—Según los últimos cálculos de la Unión Internacional de Telecomunicaciones, la red de segunda generación representa más de dos tercios de los aproximadamente 7000 millones de abonados móviles celulares del mundo, es decir, unos 4600 millones de usuarios en el mundo. Esto quiere decir que, aunque actualmente está en fase de definición la quinta generación, la gran mayoría de los usuarios mundiales utilizan la red de segunda generación actualmente. A pesar de haber sido desplegada desde comienzos de los años 90, ésta segunda generación está muy presente en el mercado y sigue siendo una tecnología muy utilizada.

El presente proyecto pretende desarrollar una estación base de telefonía móvil de segunda generación, haciendo uso de un equipo de radio definida por software y la plataforma OpenBTS, permitiendo desplegar una red de telefonía de bajo coste con múltiples utilidades, desde docencia hasta comunicaciones en zonas remotas.

Palabras clave—Asterisk, GPRS, GSM, Modulación GMSK, OpenBTS, OsmocomBB, Radio definida por software, SIP, Voz sobre IP.

I. INTRODUCCIÓN

La sociedad de la información es aquella en la cual las tecnologías que facilitan la creación, distribución y manipulación de la información juegan un papel esencial en las actividades sociales, culturales y económicas. La telefonía móvil ha jugado un papel importante ya que a día de hoy es posible realizar casi todas esas tareas desde el terminal móvil, gracias al acceso a Internet que nos ofrecen.

El presente proyecto ofrece una solución de bajo coste para el despliegue de una red GSM+GPRS como plataforma docente para asignaturas de comunicaciones móviles, todo ello gracias a la reducción de costes de los equipos de radio definida por software y el aumento de la capacidad de cómputo de los procesadores actuales. Gracias al desarrollo de dicha red, el alumno podrá: conectar dos teléfonos móviles a la red; realizar llamadas entre ellos; enviar y recibir SMS; navegar por Internet; con la flexibilidad de poder controlar todos los parámetros de la red y analizar desde la transmisión de RF, hasta los mensajes intercambiados entre los diferentes elementos de la red.

II. RESUMEN DEL ESTÁNDAR GSM

Este apartado será un breve resumen de las características principales del estándar GSM, de forma que el lector que no

esté familiarizado con él pueda comprender los conceptos de los siguientes apartados.

Las redes móviles se basan en el uso de antenas de poca distancia de cobertura, pero usando una gran cantidad de ellas, de forma que el abonado móvil no necesite de gran potencia de transmisión ya que normalmente el equipo del usuario estará alimentado con baterías. Por lo tanto, la clave de una red móvil es conocer la ubicación del abonado en todo momento, para que cuando cambie de una zona de cobertura a otra, éste no pierda conectividad y sea transparente para el usuario.

La segunda generación viene a cubrir el hueco existente en los servicios de primera generación, ya que estos sistemas fueron desarrollados por cada país y eran totalmente incompatibles. Por lo tanto esta generación unifica las comunicaciones móviles de diferentes países, de forma que un usuario pueda ir a otro país y seguir utilizando su terminal móvil. Otra de las mejoras de esta generación es el uso de modulación digital frente a la analógica, mejorando la calidad del audio y la eficiencia espectral de las transmisiones. El estándar de segunda generación más utilizado es GSM.

Éste estándar estaba pensado para la transmisión de voz, por lo que está orientado a circuitos, sin embargo para la transmisión de datos, es más eficiente la conmutación de paquetes, por ello con posterioridad se desarrolló el estándar GPRS, que funciona sobre GSM. De forma que los cambios a realizar para poder acceder a una red de datos a la red GSM fueron mínimos.

A. Elementos de la red GSM actual

Comenzaremos nuestro repaso de una red GSM mostrando un diagrama de su arquitectura, los elementos que la forman y las interfaces entre los mismos:

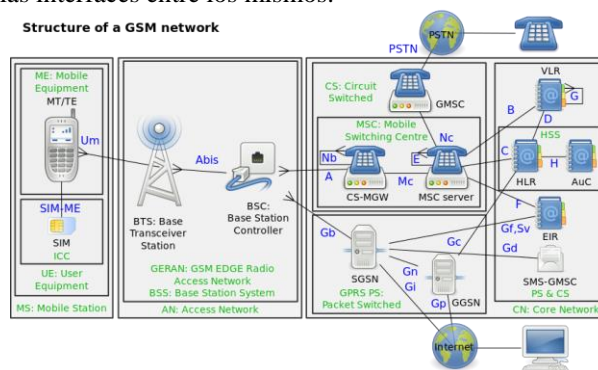


Diagrama 1. Arquitectura de una red GSM

En el diagrama 1 de la red GSM podemos observar 3 grupos de elementos, que cumplen diferentes funciones, como se verá a continuación:

- El equipo del abonado: compuesto por el terminal móvil y la tarjeta SIM (*Subscriber Identity Module*). De esta forma la tarjeta inteligente SIM es la que identifica al abonado en la red, y permite desasociar el usuario del teléfono móvil.
- La red de acceso: Compuesta por los elementos de comunicación radio, como son las BTS (*Base Transceiver Station*) y las BSC (*Base Station Controller*), en este caso las BSC son las encargadas de gestionar un grupo de zonas de cobertura. Dicha zona de cobertura la ofrece la BTS, que es la encargada de radiar la señal electromagnética y comunicarse directamente con los teléfonos móviles.
- La red de conmutación: Es la encargada de dirigir las llamadas dentro de la red, y hacia fuera de la misma. El elemento más importante es el MSC o *Mobile Switching Center*, el conmutador de llamadas. También dentro de este grupo hay diferentes elementos auxiliares para dar soporte a servicios de SMS, y bases de datos de abonados con permisos y ubicaciones para poder dirigirles las llamadas entrantes (VLR, HLR, EIR, AuC). En el diagrama aparecen también los elementos relativos a ofrecer servicio de GPRS en la red GSM y su conexión a Internet.

B. Interfaces de la red GSM

Una vez vistos los diferentes elementos de la red GSM, se puede apreciar que la conexión entre cada uno de ellos requiere de una interfaz específica, definida por un protocolo y unos mensajes intercambiados. En este proyecto nos hemos centrado en la interfaz radio, porque es la que se ha recreado siguiendo el estándar. El resto de interfaces se han simplificado como se verá en el siguiente apartado.

La interfaz radio es la encargada de la comunicación directa entre la red GSM y los equipos móviles de los abonados. Las bandas, modulación, codificación del canal, y entrelazado están descritos en los documentos del estándar [1], [2], [3] y [4].

El estándar GSM, en el documento [2], describe que es un sistema FDD/FDMA/TDMA, e incluye todas las combinaciones de bandas de frecuencias permitidas por el estándar. En nuestro caso nos centraremos en la versión europea:

- FDD: Diferentes bandas de frecuencia para los enlaces descendente y ascendente, lo que permite una comunicación full dúplex.
- FDMA: Cada banda de frecuencias, ascendente y descendente, está dividida en canales, cada uno de ellos con una frecuencia diferente.
- TDMA: Cada canal está a su vez dividido en slots temporales.

Para la parte FDD, el sistema GSM900 usa la banda de frecuencias de 890 hasta 915Mhz para el enlace ascendente y la banda 935 a 960Mhz para el enlace descendente. En el

sistema DCS1800, el enlace ascendente está entre 1710 y 1785Mhz y el descendente entre 1805 y 1880Mhz como se puede ver en el diagrama 2.

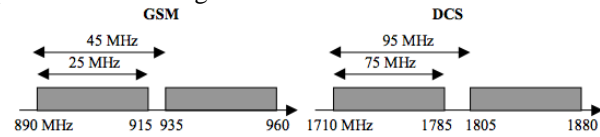


Diagrama 2. Espectro usado por GSM

Para FDMA, representado en el diagrama 3, los 25Mhz de ancho de banda del sistema GSM900 se subdividen en 124 canales de 200Khz de ancho de banda cada uno.

Esta división es similar para el sistema DCS1800, sólo que en este caso el ancho de banda de cada sentido es de 75Mhz por lo que se dispone de 372 canales. En total ambas bandas cuentan con 494 canales.

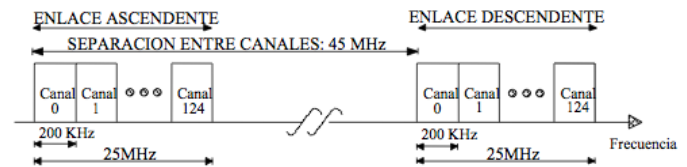


Diagrama 3. Canalización de la banda GSM900

Cada canal, tanto en GSM900 como en DCS1800 se identifica por su ARFCN (*Absolute Radio Frequency Channel Number*), para que sea más sencillo trabajar con ellos.

La parte de la señal GSM, ésta utiliza una modulación GMSK (Gaussian Minimum Shift Keyin) que es un caso particular de modulación MSK caracterizado por la alta eficiencia espectral del mismo. Para conseguir esto, a la señal binaria se la hace pasar por un filtro gaussiano de forma que se suavicen las transiciones entre 1 y 0 de entrada. Otro efecto de utilizar esta modulación es que produce una envolvente constante, de forma que es insensible a las no linealidades presentes en los amplificadores de potencia de las estaciones base. Por contrapartida, esta modulación presenta un problema de interferencia entre símbolos, debido a la reducción del espectro, pero es pequeña y se compensa con el uso de ecualizadores de canal. En el diagrama 4 se puede observar el espectro característico de una señal GSM:

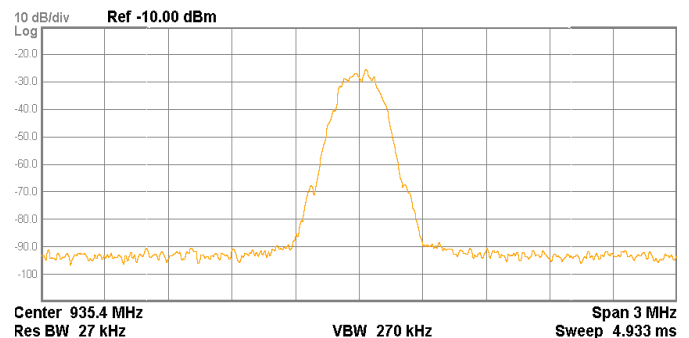


Diagrama 4. Espectro de una señal GSM

III. SISTEMA PROPUESTO

Ya se ha visto el funcionamiento del estándar GSM, como podrá apreciar el lector, al ver la arquitectura, ésta dispone de multitud de elementos e interfaces diferentes entre sí, ésta es la principal problemática del mismo, porque las soluciones comerciales suelen ser soluciones cerradas y con un coste

muy elevado, prohibitivo si se quiere tener una maqueta para prácticas de alumnos universitarios.

El presente proyecto viene a cubrir el hueco entre estudiar una red móvil de forma teórica, y poder hacer prácticas en una red GSM plenamente operativa. Gracias al mismo, se podrá desplegar una red móvil GSM en un aula, a bajo costo y analizar todos los elementos implicados. A continuación se va a detallar la solución desarrollada en el presente proyecto.

A. Arquitectura e interfaces de la red móvil

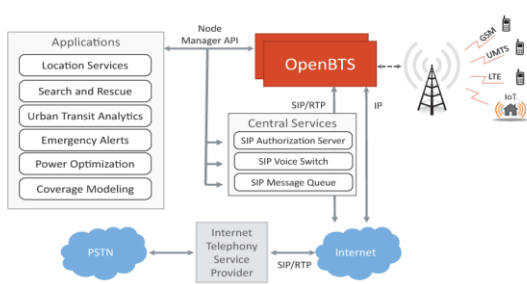


Diagrama 5. Arquitectura de la red GSM de OpenBTS

En el diagrama 5 aparece reflejada la arquitectura de una red móvil GSM haciendo uso de OpenBTS. Como podrá observar es mucho más sencilla que la red actual, ya que como elementos constitutivos sólo requiere de una interfaz radio, un servidor de aplicaciones con el paquete OpenBTS instalado y una conexión a Internet para poder dar servicio de llamadas externas a la red, y servicio de navegar por Internet a los abonados de la red.

Puesto que estamos hablando de una red privada, podemos editarla según nuestras necesidades, de forma que aparecen una serie de posibles aplicaciones al margen de la realización de llamadas, como se puede observar en la parte izquierda del diagrama 5 gracias a la interfaz de programación que nos ofrece OpenBTS.

Una vez vista la arquitectura de la red, describiremos los diferentes elementos que la componen en los siguientes apartados.

B. Hardware empleado



Diagrama 6. Equipo de radio definida por software

Comenzaremos con el hardware que se ha utilizado para desplegar la red, en nuestro caso se ha utilizado como elemento radiante un equipo de radio definida por software, éste equipo transforma las tramas UDP que le envía un servidor en una radiación electromagnética y viceversa, ya que es transmisor y receptor. El diagrama 6 es un ejemplo del dispositivo radio empleado.

Otro elemento hardware empleado ha sido un servidor tipo PC, con el sistema operativo Linux Ubuntu Server 12.04 32bits. Como elemento indispensable para la comunicación

con el equipo radio es que disponga de una interfaz Gigabit Ethernet o un conversor USB 3.0 a Gigabit Ethernet también funciona correctamente.

El último elemento hardware empleado han sido una serie de teléfonos móviles con diferentes características, pero todos con capacidad de conectarse a una red GSM900 y DCS1800 como elemento común. Cada teléfono móvil cuenta con su correspondiente tarjeta SIM, las tarjetas SIM empleadas han sido las que se han dado de baja de diferentes compañías telefónicas, esto no implica que no se puedan conectar a nuestra red, ya que tenemos la capacidad de discriminar que usuarios disponen de acceso y cuales no mediante el IMSI con el que se identifica el abonado.

C. Arquitectura del software OpenBTS

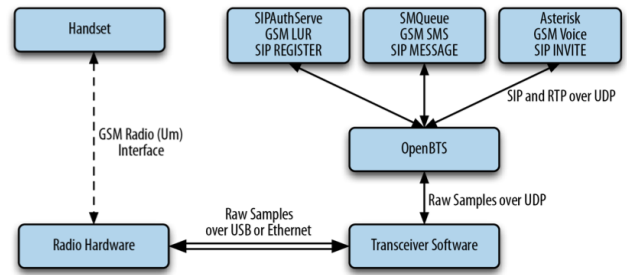


Diagrama 7. Elementos software de OpenBTS

En el diagrama 7 podemos observar los diferentes servicios e interfaces del paquete OpenBTS. Éste consiste en diferentes servicios que se ejecutan en el servidor, de forma que todos juntos ofrecen una pasarela entre la red GSM y una red VoIP. Como se puede ver en el diagrama 7, el paquete OpenBTS dispone de los siguientes servicios: “transceiver”, “OpenBTS”, “SIPAuthServe”, “SMQueue”. El paquete “Asterisk” no pertenece al grupo de servicios de OpenBTS, pero lo incluye por defecto al instalarlos. Cada uno de ellos cumple una función:

- **Transceiver:** Es el encargado de generar y recibir las tramas UDP procedentes del equipo radio. Se comunica con OpenBTS para generar la modulación, multiplexado y codificación del canal mediante la generación de las tramas en fase y cuadratura que se encapsulan en paquetes UDP. También se encarga de gestionar los canales físicos en las bandas de GSM.
- **OpenBTS:** Es el encargado de la conversión entre una llamada VoIP a una llamada GSM. También se encarga de gestionar los trasposos de llamada en el caso de redes con más de una celda de cobertura.
- **SIPAuthServe:** Es el servicio de seguridad de la red GSM-VoIP. Puesto que de cara al conmutador de llamadas, cada teléfono móvil es un usuario más de la red VoIP, este servicio se encarga de gestionar los accesos a la red mediante una tabla con el IMSI de cada abonado con permiso de acceso.
- **SMQueue:** Este es el servicio encargado de la recepción y retransmisión de SMS.

D. Conmutador de llamadas por software. Asterisk

Este paquete es un conmutador de llamadas virtual. Hace

uso de los protocolos RTP (*Real Time Protocol*) para la transmisión del audio de las llamadas y del protocolo SIP (*Session Initiation Protocol*) para los mensajes entre los diferentes elementos de la red. En nuestro caso, se incluye en el paquete de OpenBTS de forma que su configuración sea mucho más sencilla.

Asterisk está basada en dos planos, los usuarios y los planes de llamadas.

La gestión de usuarios se definen los usuarios que tienen permiso de acceso así como las necesidades de cada uno dentro de la centralita de llamadas. En nuestro caso, el servicio OpenBTS es un usuario más de la centralita, de forma que tiene que estar dicho usuario configurado en el archivo de usuarios. Otro usuario que se ha configurado, aunque no es obligatorio disponer de él, es el acceso a la red pública de telefonía mediante un servicio troncal de VoIP, dicho servicio contratado con la empresa Netelip, es un usuario más en nuestra central de conmutación y nos permite realizar llamadas con nuestros teléfonos móviles a teléfonos fuera de nuestra red.

La gestión de los planes de llamadas se organizan en lo que se llaman “contextos”. Cada usuario dispone de un contexto asignado, y es dicho contexto el que determina qué hacer cuando el usuario realiza una llamada, en otras palabras, los contextos indican de forma secuencial las acciones de la centralita cuando el usuario marca una extensión determinada.

E. Análisis del intercambio de mensajes gracias a OsmocomBB

El sistema OsmocomBB es un conjunto de software, firmware y hardware con la capacidad de crear un teléfono móvil virtual. Hace uso de un teléfono móvil modelo Motorola C123 al cual se le ha modificado el firmware para transmitir las tramas directamente al ordenador, de forma que sea el software instalado en el ordenador, el que realiza todo el procesado de la señal. De esta forma, al controlar todo el sistema mediante el software, se puede generar un archivo con el intercambio de mensajes entre una estación base GSM y un teléfono móvil para analizarlos a posteriori.

IV. RESULTADOS OBTENIDOS

En este apartado se van a mostrar los resultados obtenidos tras el desarrollo del prototipo de estación base. Comenzaremos con el análisis de los posibles canales libres en nuestra zona de pruebas, seguiremos con las diferentes medidas RF realizadas al prototipo, para finalizar con el análisis de los mensajes intercambiados entre los diferentes elementos de la red.

A. Canales libres en la zona de test.

Antes de comenzar a radiar en la banda GSM, se ha realizado un análisis de los canales que se encuentran libres en nuestra zona de test, en el edificio del CITIC (Centro de Investigación de Tecnologías de la Información y Comunicaciones). Para ello se ha buscado la información de la localización de las antenas, y sus características para los diferentes operadores de telecomunicaciones en la zona. Dicha información se encuentra en la web:

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

Tras analizar la información publicada en dicha web, concluimos que la zona de baja frecuencia en la banda GSM

es la que se encuentra más libre de señal. Decidimos utilizar el canal 2 para GSM900 y el canal 512 para DCS1800.

B. Medidas RF

Para las medidas de RF se ha hecho uso de un analizador vectorial modelo EXA del fabricante Agilent. Dispone de un ancho de banda de 3.6GHz, suficiente para cubrir las bandas GSM de nuestro interés.

Las medidas tomadas para ambas bandas son:

- Potencia del canal: Se mide la potencia de un slot temporal para un transmisor que emita el canal lógico de difusión o BCCH.
- Potencia en canales adyacentes: La finalidad de este test es la de asegurar que para un transmisor no interfiere ni introduce ruido por encima de un nivel máximo permitido.
- Ancho de banda ocupado: Devuelve en qué ancho de banda se concentra el 99% de la potencia de transmisión.
- Mascara de espectro de emisión: El objetivo de esta medida consiste en verificar si la potencia de cualquier emisión del transmisor no excede los niveles específicos.

Los resultados de las medidas se pueden encontrar en la memoria completa, con mucho más detalle, aquí solo comentar que las medidas son acordes al estándar GSM, según están definidos los test en [1].

C. Análisis de los mensajes intercambiados en diferentes escenarios.

El análisis de los mensajes intercambiados entre los diferentes elementos de la red se ha realizado con la herramienta Wireshark, ya que los archivos con las trazas de los mensajes añaden una cabecera a los mismos llamada GSMTAP, que es capaz de interpretar Wireshark.

A continuación mostraremos algunos escenarios en los que se han analizado los mensajes.

Comenzamos por analizar los mensajes que emite la estación base de forma periódica, en el diagrama 8 vemos un ejemplo de los mismos:

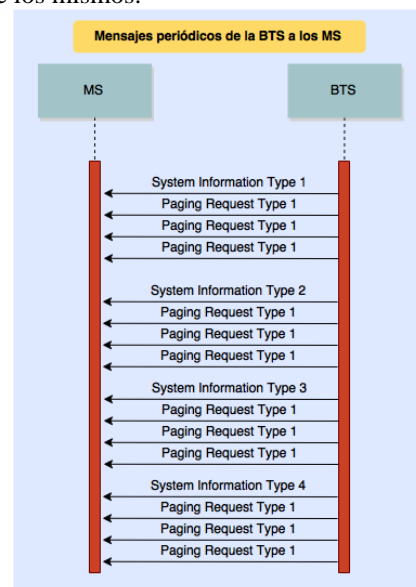


Diagrama 8 Mensajes periódicos de OpenBTS

Estos mensajes incluyen información acerca de la red, el número de secuencia del resto de mensajes para poder ordenarlos y parámetros necesarios para la correcta operación del teléfono móvil.

El siguiente escenario se presenta cuando un teléfono móvil se registra en la red, para ello envía un mensaje de actualización de la localización y espera hasta recibir la aceptación de dicha actualización. Los mensajes intercambiados se muestran en el diagrama 9:

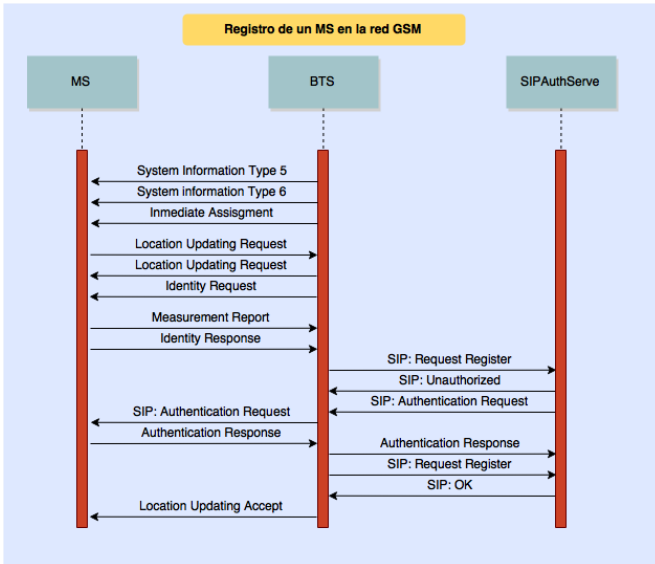


Diagrama 9. Mensajes intercambiados durante la conexión de un móvil a la red

En este caso entra en juego otro elemento de la red, el servicio de seguridad y autenticación, ya que se necesita saber si el abonado dispone de permiso para acceder a la red.

El siguiente escenario se produce cuando se envía un SMS por parte del abonado, y después éste recibe un SMS. Los mensajes en este caso corresponden al diagrama 10:

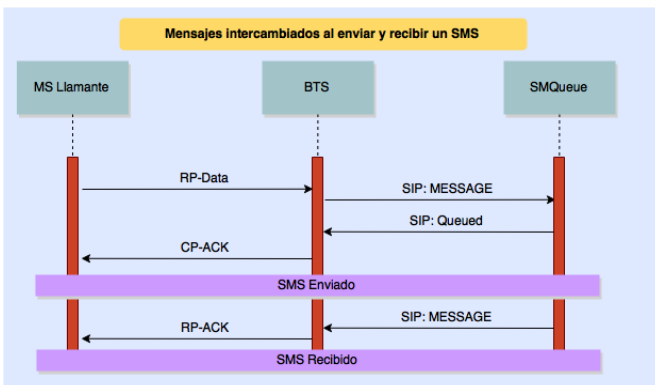


Diagrama 10. Mensajes intercambiados al enviar y recibir un SMS

En este caso entra en juego el servicio SMQueue, encargado de la gestión de los SMS. No se comunica directamente con el móvil, puesto que no puede acceder a la red GSM directamente, para ello hace uso de la pasarela OpenBTS.

V. APLICACIONES

En este apartado se va a hacer un breve resumen de las aplicaciones que actualmente se están desarrollando gracias

al empleo de este proyecto.

El detalle del presupuesto de una instalación de una sola celda de cobertura se puede ver en la memoria del proyecto, aquí nos quedaremos con el total del mismo, 3096.96 euros. Gracias a este bajo coste, la red puede ser desplegada en multitud de situaciones.

La principal aplicación está en el área de la docencia de redes inalámbricas y redes móviles, ya que los alumnos podrán realizar prácticas en un entorno real de comunicaciones móviles.

Otra aplicación del mismo es la telefonía en zonas donde no llega la cobertura de las grandes operadoras, ya que el despliegue del equipamiento de éstas tiene un coste muy alto para la poca densidad de población de estas zonas.

Y por último, siguiendo el ejemplo anterior, otra aplicación reside en el restablecimiento de las comunicaciones tras desastres naturales, ya que pueden verse afectadas las redes principales, y los equipos de emergencia necesitan un buen servicio de comunicaciones para ser lo más eficaces posibles. El diagrama 11 muestra un ejemplo de una instalación para casos de emergencia.



Diagrama 11. Ejemplo de instalación de una estación base en un entorno rural

VI. CONCLUSIONES

A modo de resumen se van a enumerar las lecciones aprendidas gracias al desarrollo del proyecto:

- Modulación y constelación GMSK. Espectro de la misma.
- Codificación del canal en GSM.
- Uso eficiente del espectro con la combinación FDD/FDMA/TDMA para que múltiples usuarios puedan comunicarse de forma simultánea.
- Estudio de los canales lógicos de GSM, tanto los de control por parte de la BTS, como los de señalización y tráfico del abonado.
- Estudio del intercambio de mensajes SIP en una red VoIP.
- Configuración de un plan de llamadas para una centralita VoIP Asterisk.
- Contratación de un servicio troncal SIP para la realización de llamadas fuera de la red privada.

Como siguientes pasos para continuar la línea del estudio de las redes celulares se han identificado las siguientes ramas:

- Desplegar más de una celda GSM y ver uno de los mecanismos más importantes en las redes celulares, como es el traspaso de una llamada que esté en curso. Para ello será necesario: el uso de varios equipos SDR; configurar la potencia en ambos equipos para que tengan un área de cobertura común pequeño; configurar los diferentes elementos de la red, OpenBTS y Asterisk, para poder realizar el traspaso de llamadas correctamente.

- La otra posibilidad es la evolución del sistema GSM+GPRS a la tercera generación de comunicaciones móviles, 3G o UMTS. Para ello el proyecto OpenBTS dispone de una versión del sistema OpenBTS para 3G que todavía está en fase de pruebas. Este paso nos permitiría estudiar en profundidad una red más actual a la GSM pero con funcionalidades similares.

REFERENCIAS

[1]TS 51.021, "Base Station System (BSS) equipment specification ," 3GPP

[2]TS 45.005, "Radio transmission and reception ," 3GPP

[3]TS 45.004, "Technical Specification Group GSM/EDGE Radio Access Network; Modulation ," 3GPP

[4]TS 45.001, "Physical layer on the radio path; General description ," 3GPP

Teoría de la Señal y Comunicaciones

Clasificación de proteínas representadas mediante señales extraídas de índices bioquímicos

Autor: Amelia Otilia Villegas Morcillo, e-mail: melyvm@correo.ugr.es

Tutores: Victoria Sánchez Calle; e-mail: victoria@ugr.es

Antonio Miguel Peinado Herreros; e-mail: amp@ugr.es

Titulación: Grado en Ingeniería de Tecnologías de Telecomunicación

**Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada**

Resumen—Durante las últimas décadas, la clasificación de proteínas se ha convertido en una tarea sustancial en el ámbito científico, debido a que una eficiente determinación de la función que realizan facilita el diseño de nuevos fármacos y vacunas para combatir distintas enfermedades humanas.

Las proteínas están compuestas por una secuencia variable de aminoácidos. La disposición de estas pequeñas moléculas es lo que determina la estructura y función de cada proteína en particular. Frente a las tradicionales técnicas de comparación de cadenas simbólicas, recientemente se han desarrollado representaciones numéricas a partir de distintas propiedades bioquímicas, permitiendo la aplicación de técnicas de procesado de señal.

Por tanto, el principal propósito de este trabajo es obtener una predicción de la función que realiza una proteína determinada, comparándola con otras ya conocidas.

Palabras clave—AMDF, célula, cepstrum, clasificación, discriminante covariante, distancia euclídea, enzima, índice de aminoácido, kernel, porcentaje de aciertos, procesado de señal, proteína, ProtLock, PseAAC, secuencia de aminoácidos, SVM, test de Jackknife.

I. INTRODUCCIÓN

LOS avances recientes han permitido conocer el conjunto de miles de genes que constituyen la vida. Así, la información genética de los seres vivos se transmite de manera codificada a través de moléculas de ADN, la cual es transcrita en otro tipo de moléculas, el ARN, que, a su vez, se traduce en las proteínas. Por tanto, las proteínas desempeñan un papel fundamental en la vida realizando diversas funciones biológicas importantes.

Las macromoléculas que forman las proteínas se caracterizan por una estructura tridimensional o conformación, así como por una secuencia específica de un número variable de aminoácidos, pequeñas moléculas con propiedades físico-químicas propias que las diferencian entre sí. La cadena lineal de aminoácidos es la que controla la estructura tridimensional de una proteína y, en consecuencia, la función que realiza.

La necesidad de encontrar una manera rápida y eficaz de predecir la función de nuevas proteínas es cada vez más notable, debido a que el número de proteínas desconocidas que entran en los bancos de datos crece de manera exponencial con los años. La clasificación de nuevas proteínas en los bancos de datos se realiza a partir de las

similitudes existentes con las secuencias de aminoácidos de proteínas que ya se conocen.

Tradicionalmente, esta clasificación se ha efectuado por medio de técnicas de comparación de cadenas, como Programación Dinámica, la cual se basa en determinar el coste de alineamiento de dos secuencias. En estos casos, el procesamiento se ha realizado a partir de la cadena simbólica de aminoácidos, donde cada uno se representa con una letra extraída de un alfabeto de 20 elementos (Tabla I).

TABLA I
NOMBRES Y ABBREVIACIONES DE LOS 20 AMINOÁCIDOS MÁS COMUNES [1]

Aminoácido	Código de tres letras	Código de una letra	Aminoácido	Código de tres letras	Código de una letra
Alanina	Ala	A	Leucina	Leu	L
Arginina	Arg	R	Lisina	Lys	K
Asparagina	Asn	N	Metionina	Met	M
Ácido Aspártico	Asp	D	Fenilalanina	Phe	F
Cisteína	Cys	C	Prolina	Pro	P
Glutamina	Gln	Q	Serina	Ser	S
Ácido Glutámico	Glu	E	Treonina	Thr	T
Glicina	Gly	G	Triptófano	Trp	W
Histidina	His	H	Tirosina	Tyr	Y

De manera reciente, se han comenzado a aplicar técnicas de procesado de señal, cuyas herramientas pueden ser destinadas al análisis y caracterización de las proteínas, mejorando la extracción de propiedades difícilmente detectables al trabajar con cadenas simbólicas. Para ello la proteína ha de ser representada como señal previamente, convirtiendo la cadena simbólica en otra numérica a partir de medidas bioquímicas de cada aminoácido (índices de aminoácido).

Los objetivos previstos para el trabajo se muestran en el esquema de la Fig. 1, en el cual se pueden ver los diferentes pasos a seguir para clasificar un conjunto de proteínas así como las interdependencias entre las distintas fases.



Fig. 1. Diagrama de flujo del trabajo realizado.

II. BASES DE DATOS DE PROTEÍNAS

En este trabajo se han considerado dos bases de datos de las muchas existentes: una con clases de enzimas, y otra basada en localización subcelular.

Además, se presenta el porcentaje de identidad promedio de cada una de las clases, el cual viene definido como el promedio entre el máximo número de componentes que encajan al comparar dos secuencias.

A. Predicción basada en clases de enzimas

Las enzimas se encuentran relacionadas a la familia o subfamilia a la que pertenecen debido a su función, y su especificidad [2]. Este conjunto está centrado en la familia de oxidoreductasas, contiene un total de 2513 proteínas clasificadas en 16 subfamilias (Tabla II).

TABLA II

CLASES EN LA BASE DE DATOS DE ENZIMAS Y SU IDENTIDAD PROMEDIO [2]

Clase	Nombre	Nº de secuencias	Identidad promedio (%)
1	Grupo CH-OH	297	13.16
2	Grupo aldehído/oxo	201	17.27
3	Grupo CH-CH	180	12.89
4	Grupo CH-NH ₂	122	13.94
5	Grupo CH-NH	105	13.47
6	NADH/NADPH	285	11.71
7	Otros compuestos nitrogenados	61	13.69
8	Grupo azufre	56	19.70
9	Grupo hemo	248	21.32
10	Difenoles y sustancias relacionadas	94	12.71
11	Peróxido	149	19.01
12	Donadores individuales	90	13.66
13	Donadores por parejas	255	15.71
14	Radicales superóxido	144	26.92
15	Grupos -CH ₂	82	18.55
16	Ferredoxina reducida	144	16.99

B. Predicción basada en localización subcelular

De acuerdo a la localización en una célula, las proteínas se clasifican generalmente en 14 categorías [3]. En este caso se han escogido dos subconjuntos de datos independientes de trabajo, uno de entrenamiento, con 3587 proteínas y otro de test, con 4362 (Tabla III).

TABLA III

CLASES EN LA BASE DE DATOS DE LOCALIZACIÓN SUBCELULAR Y SU IDENTIDAD PROMEDIO (SUBCONJUNTOS DE ENTRENAMIENTO Y TEST) [3]

Clase	Nombre	Nº sec. train	Identidad promedio train (%)	Nº sec. test	Identidad promedio test (%)
1	Pared celular	69	10.31	34	9.86
2	Centriolo	64	18.62	4	29.67
3	Cloroplasto	303	7.90	819	8.59
4	Citoplasma	991	7.24	171	6.76
5	Citoesqueleto	241	11.03	128	8.93
6	Retículo endoplasmático	288	11.28	136	9.04
7	Extracelular	381	7.60	1213	6.67
8	Aparato de Golgi	70	7.83	41	6.54
9	Lisosoma	123	8.66	57	6.72
10	Mitocondria	379	7.76	743	9.29
11	Núcleo	386	7.66	892	8.14
12	Peroxisoma	145	9.57	84	6.91
13	Membrana de plasma	65	13.07	24	9.67
14	Vacuola	82	13.19	16	8.32

III. GENERACIÓN DE LA SEÑAL

En este apartado se presentan los índices bioquímicos a considerar para cada aminoácido y la construcción de la señal completa a partir de la normalización previa de los mismos.

A. Representación con 3 índices

En [4] se emplean tres propiedades bioquímicas para interpretar cada aminoácido: hidrofobicidad, hidrofiliidad y masa. Estos índices son bastante utilizados ya que informan de la masa molecular de cada aminoácido y de la tendencia de las moléculas a formar enlaces de hidrógeno.

Los valores numéricos de estas propiedades se incluyen en una matriz \mathbf{R} y, en particular, cada una de sus filas y_i constituye un atributo:

$$y_i(j) = R(i, j) \quad i = 1, \dots, 3; \quad j = 1, \dots, 20 \quad (1)$$

B. Representación con 531 índices

En [5] la representación numérica de la cadena de aminoácidos se realiza empleando una base de datos, la AAIndex (*Amino Acid Index*). En concreto, la primera sección de la versión 9, AAIndex1, la cual posee 531 índices de aminoácido después de eliminar los índices incompletos. Por tanto, la matriz de índices \mathbf{R} contendrá 531 filas.

C. Construcción de la señal

En primer lugar se realiza una normalización en media y varianza de cada una de las señales contenidas en la matriz \mathbf{R} de la siguiente manera [4] [5]:

$$R(i, j) = \frac{R^0(i, j) - \sum_{n=1}^{20} \frac{R^0(i, n)}{20}}{\sqrt{\frac{\sum_{n=1}^{20} [R^0(i, n) - \sum_{m=1}^{20} \frac{R^0(i, m)}{20}]^2}{20}}} \quad (2)$$

Con $i = 1, \dots, M$, donde M es el número de señales (3 o 531). Por otro lado, si denominamos $a(n)$ a la cadena de aminoácidos para la proteína, cada una de las secuencias numéricas (señales) $s_i(n)$ se construye sustituyendo cada aminoácido por su valor correspondiente:

$$s_i(n) = y_i(a(n)) = R(i, a(n)) \quad n = 1, \dots, L \quad (3)$$

Donde L es el número total de aminoácidos.

IV. REPRESENTACIÓN EN LONGITUD FIJA

En esta parte se presentan las transformaciones que se aplicarán a las secuencias numéricas de proteína para convertirlas en cadenas de longitud fija.

A. Composición Pseudo-Aminoácida (PseAAC)

La composición aminoácida (*Amino Acid Composition*, AAC) es una representación basada en la frecuencia de aparición de cada uno de los 20 aminoácidos comunes en la secuencia [4].

Con objeto de incluir información acerca de las relaciones que puedan existir entre las posiciones que ocupan los aminoácidos en la cadena se ha desarrollado la composición pseudo-aminoácida (*Pseudo-Amino Acid Composition*, PseAAC) [4]. En esta representación se mantiene el vector de AAC y se añaden los factores de correlación:

$$\theta_k = \frac{1}{L-k} \sum_{n=1}^{L-k} \theta(n, n+k), \quad k = 1, \dots, \lambda \quad (4)$$

Donde k indica el número de posiciones que separan los aminoácidos a comparar, y λ es la distancia de correlación máxima. Por otro lado, la función de correlación entre los aminoácidos de las posiciones n y $n + k$ se obtiene como:

$$\Theta(n, n + k) = \frac{1}{M} \sum_{i=1}^M [s_i(n + k) - s_i(n)]^2 \quad (5)$$

Por tanto, una vez obtenidos los factores de correlación se puede formar el vector \mathbf{X} de $20 + \lambda$ dimensiones para representar una proteína, cuyas componentes toman los siguientes valores:

$$x_u = \begin{cases} \frac{f_u}{\sum_{i=1}^{20} f_{i+w} \sum_{k=1}^{\lambda} \theta_k}, & (1 \leq u \leq 20) \\ \frac{w \theta_{u-20}}{\sum_{i=1}^{20} f_{i+w} \sum_{k=1}^{\lambda} \theta_k}, & (21 \leq u \leq 20 + \lambda) \end{cases} \quad (6)$$

Donde f_u es la frecuencia de aparición del aminoácido u (de entre los 20 que ocupan la proteína \mathbf{X}) y w es el factor de ponderación elegido (en este caso $w = 0.05$).

B. Función AMDF

En [5] se propone una representación parecida a PseAAC, en la cual únicamente se introducen los efectos del orden en la secuencia concatenando las diferencias cuadráticas para cada uno de los índices de aminoácido:

$$\partial_k^i = \frac{1}{L-k} \sum_{n=1}^{L-k} D_i(n, n + k) \quad i = 1, \dots, M; k = 1, \dots, \lambda \quad (7)$$

$$D_i(n, n + k) = [s_i(n + k) - s_i(n)]^2 \quad (8)$$

En este caso el vector de características es M veces más largo. Las ecuaciones anteriores definen una autocorrelación de tipo aditiva entre los valores de la secuencia numérica s_i . En concreto, se trata de una versión cuadrática de la función AMDF (*Average Magnitude Difference Function*) [6], comúnmente empleada en la estimación del *pitch* o frecuencia fundamental de una señal de voz.

C. Cepstrum

El cepstrum [7] de una señal discreta $x(n)$ con espectro $X(\omega)$ se define como la transformada inversa de Fourier del espectro logarítmico:

$$\hat{x}(n) = \mathcal{F}^{-1}[\log X(\omega)] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log X(\omega) e^{j\omega n} d\omega \quad (9)$$

Donde n es un índice temporal denominado cuelfrecuencia. A esta transformación se la suele llamar cepstrum complejo a pesar de que la secuencia resultante es real. Por tanto, el cepstrum complejo poseerá también información acerca de la fase de la transformada de Fourier.

Por otro lado, si aplicamos el logaritmo sobre el módulo de la transformada, el resultado es el denominado cepstrum real:

$$c_x(n) = \mathcal{F}^{-1}[\log |X(\omega)|] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log |X(\omega)| e^{j\omega n} d\omega \quad (10)$$

Además, considerando formas rápidas para el cálculo de la transformada de Fourier, como la FFT (*Fast Fourier Transform*):

$$\hat{x}(n) = \text{IFFT}[\log \text{FFT}[x(n)]] \quad (11)$$

$$c_x(n) = \text{IFFT}[\log |\text{FFT}[x(n)]|] \quad (12)$$

Es posible medir la distancia euclídea entre los espectros logarítmicos de dos señales $x(n)$ e $y(n)$. En términos de cepstrum:

$$d(x, y) = \|\hat{\mathbf{x}} - \hat{\mathbf{y}}\|^2 = \sum_{n=-L}^L (\hat{x}(n) - \hat{y}(n))^2 = \sum_{n=-L}^{-1} (\hat{x}(n) - \hat{y}(n))^2 + (\hat{x}(0) - \hat{y}(0))^2 + \sum_{n=1}^L (\hat{x}(n) - \hat{y}(n))^2 \quad (13)$$

El término de cuelfrecuencia 0 se suele ignorar cuando se pretende que la distancia no dependa de la potencia de las señales. En el caso del cepstrum real, al poseer simetría par:

$$d(x, y) = \|\mathbf{c}_x - \mathbf{c}_y\|^2 = (c_x(0) - c_y(0))^2 + 2 \sum_{n=1}^L (c_x(n) - c_y(n))^2 \quad (14)$$

Normalmente se suele limitar la longitud del cepstrum aplicando una ventana de *liftering*, descartando así las cuelfrecuencias superiores y suavizando el espectro.

V. ALGORITMOS DE CLASIFICACIÓN

En este apartado se exponen los distintos algoritmos considerados en este estudio para la clasificación de proteínas.

A. Clasificación basada en mínima distancia euclídea

Este es uno de los algoritmos más básicos utilizados para la clasificación con AAC y PseAAC, empleado en sustitución de la distancia de Hamming [4]. La medida de similitud entre la proteína \mathbf{X} y cada uno de los centroides de las clases $\bar{\mathbf{X}}^\xi$ se define como:

$$D_E^2(\mathbf{X}, \bar{\mathbf{X}}^\xi) = \sum_{i=1}^{n_s} (x_i - \bar{x}_i^\xi)^2 = (\mathbf{X} - \bar{\mathbf{X}}^\xi)^T (\mathbf{X} - \bar{\mathbf{X}}^\xi) \quad (15)$$

Donde n_s es la longitud de la secuencia numérica y $\xi = 1, 2, 3, \dots, m$ es cada una de las clases del conjunto. La regla de predicción viene dada por:

$$D_E^2(\mathbf{X}, \bar{\mathbf{X}}^\mu) = \mathbf{Min}\{D_E^2(\mathbf{X}, \bar{\mathbf{X}}^1), D_E^2(\mathbf{X}, \bar{\mathbf{X}}^2), \dots, D_E^2(\mathbf{X}, \bar{\mathbf{X}}^m)\} \quad (16)$$

Donde μ define la clase en la cual se ha clasificado la proteína \mathbf{X} .

B. Algoritmo de clasificación ProtLock

Este algoritmo está basado en la distancia de Mahalanobis, la cual es una extensión de la distancia euclídea entre dos variables aleatorias teniendo en cuenta la correlación existente entre ambas [4]. La medida de similitud entre la proteína \mathbf{X} y cada uno de los centroides de las clases $\bar{\mathbf{X}}^\xi$ se define como:

$$D_P^2(\mathbf{X}, \bar{\mathbf{X}}^\xi) = (\mathbf{X} - \bar{\mathbf{X}}^\xi)^T \mathbf{C}_\xi^{-1} (\mathbf{X} - \bar{\mathbf{X}}^\xi) \quad (17)$$

Los elementos de la matriz de covarianzas para cada clase \mathbf{C}_ξ se obtienen como:

$$c_{i,j}^\xi = \frac{1}{n_\xi - 1} \sum_{k=1}^{n_\xi} [x_{k,i}^\xi - \bar{x}_i^\xi] [x_{k,j}^\xi - \bar{x}_j^\xi] \quad (i, j = 1, 2, \dots, n_s) \quad (18)$$

Donde n_ξ es el número de secuencias pertenecientes a la clase ξ . Luego, la regla de predicción viene dada por:

$$D_P^2(\mathbf{X}, \bar{\mathbf{X}}^\mu) = \mathbf{Min}\{D_P^2(\mathbf{X}, \bar{\mathbf{X}}^1), D_P^2(\mathbf{X}, \bar{\mathbf{X}}^2), \dots, D_P^2(\mathbf{X}, \bar{\mathbf{X}}^m)\} \quad (19)$$

C. Algoritmo del discriminante covariante

Este algoritmo se basa en la regla de decisión MAP (*Maximum a Posteriori*) aplicada sobre una distribución Normal multivariada [4] [8].

La medida de similitud entre la proteína \mathbf{X} y cada uno de los centroides de las clases $\bar{\mathbf{X}}^\xi$ se define como:

$$F(\mathbf{X}, \bar{\mathbf{X}}^\xi) = D_P^2(\mathbf{X}, \bar{\mathbf{X}}^\xi) + \ln |\mathbf{C}_\xi| \quad (20)$$

Donde $D_p^2(\mathbf{X}, \bar{\mathbf{X}}^\xi)$ es la distancia cuadrática de Mahalanobis y $|\mathbf{C}_\xi|$ es el determinante de la matriz de covarianzas para la clase ξ . Luego, la regla de predicción viene dada por:

$$F(\mathbf{X}, \bar{\mathbf{X}}^\mu) = \text{Min}\{F(\mathbf{X}, \bar{\mathbf{X}}^1), F(\mathbf{X}, \bar{\mathbf{X}}^2), \dots, F(\mathbf{X}, \bar{\mathbf{X}}^m)\} \quad (21)$$

D. Máquinas de vectores de soporte (SVM)

Las máquinas de vectores de soporte (SVM, *Support Vector Machines*) son un tipo de algoritmo de aprendizaje supervisado [9]. El algoritmo emplea un conjunto de datos de entrenamiento para aprender una frontera de decisión lineal que separa dos clases. Dicha frontera es la que utiliza posteriormente para clasificar el conjunto de datos de test.

Cuando las fronteras de decisión no son lineales, SVM emplea funciones tipo *kernel* (Tabla IV) para mapear los datos a un espacio de orden superior donde se pueda encontrar una frontera óptima.

TABLA IV
KERNELS EMPLEADOS EN SVM

Producto escalar	Polinómico de orden d	Función base radial (RBF, Radial Basis Function) Gaussiana
$k(x, x') = \langle \mathbf{x}, \mathbf{x}' \rangle$	$k(x, x') = \langle \mathbf{x}, \mathbf{x}' \rangle^d$	$k(x, x') = \exp\left(-\frac{\ \mathbf{x} - \mathbf{x}'\ ^2}{2\sigma^2}\right)$

Por otra parte, la clasificación binaria propia de las SVM se puede extender a M clases mediante dos técnicas distintas. La primera se denomina "SVM uno contra el resto" y consiste en construir un clasificador para cada clase separándola de todas las restantes, por lo que se entrenan un total de M clasificadores binarios. La segunda técnica es denominada "SVM por pares" y consiste en construir clasificadores cada par de clases, generando un total de $(M - 1)M/2$ clasificadores binarios.

VI. RESULTADOS

El marco de experimentación realizado consta de las cuatro variables estudiadas: conjunto de proteínas, representación en longitud fija, clasificador y tipo de test.

En cuanto al tipo de test se han considerado dos distintos: test de Jackknife, también conocido como *leave-one-out*, que consiste en entrenar el clasificador con todas las secuencias del conjunto menos la que se va a clasificar, y experimento basado entrenamiento y test con dos conjuntos distintos.

Todas las representaciones, clasificadores y test han sido implementados con el software de Matlab© excepto las SVM, para las cuales se ha utilizado el paquete Gist 2.3 [10].

A continuación se muestra un estudio previo realizado sobre cada representación para determinar la longitud apropiada del vector resultante.

A. Estudio previo: AAC y PseAAC

En el caso de AAC, los vectores son de 20 elementos y para PseAAC se ha utilizado una distancia de correlación máxima $\lambda = 9$ [4], por lo que los vectores serán de 29 elementos tanto para 3 como para 531 índices. Sin embargo, solo $20 + \lambda - 1$ son independientes debido a la normalización realizada, por lo que se eliminará una componente cualquiera del vector.

B. Estudio previo: función AMDF

La distancia de correlación máxima viene determinada por la longitud de la secuencia más corta en los conjuntos de

proteínas considerados (que en este caso es 41). Por tanto, para pruebas con el test de Jackknife sobre la base de datos de enzimas, con $\lambda = [8, 20, 40]$ tanto para 3 como para 531 índices, los porcentajes de aciertos con el clasificador de distancia euclídea se muestran en la Tabla V.

TABLA V
PORCENTAJE DE ACIERTOS (%) PARA AMDF CON 3 Y 531 ÍNDICES, Y DISTINTAS DISTANCIAS DE CORRELACIÓN

Componentes AMDF	3x8	3x20	3x40	531x8	531x20	531x40
Porcentaje de aciertos (%)	28.85	31.44	31.48	45.76	46.12	45.68

Se observa que con 531 índices los resultados son mejores, mientras que el hecho de aumentar el valor de λ no incrementa los aciertos de manera considerable. Por tanto, las pruebas se realizarán con $\lambda = 8$ y 531 índices, proporcionando vectores de 4248 elementos.

C. Estudio previo: cepstrum

El orden de la FFT a aplicar viene dado por la longitud de la secuencia más larga en los conjuntos (que es de 4687), por lo que se ha utilizado la potencia de dos siguiente a este valor (8192) y se ha aplicado una ventana de *liftering* rectangular sobre el resultado. En este punto se han realizado pruebas con los dos subconjuntos de entrenamiento y test de la base de datos de localización subcelular, con el índice de hidrofobicidad y se han obtenido los porcentajes de aciertos para el clasificador de distancia euclídea en función de la longitud de la ventana para el cepstrum real y complejo, considerando o no el coeficiente de cuelfrencia 0 (Fig. 2).

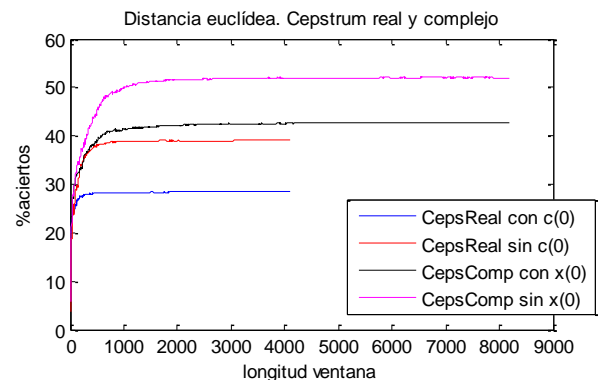


Fig. 2. Porcentaje de aciertos (%) en función del tamaño de ventana para cepstrum real y complejo considerando o no el coeficiente de cuelfrencia 0.

Se observa que el porcentaje de aciertos mejora conforme se aumenta el tamaño de la ventana hasta llegar un punto en el que se satura. En el caso del cepstrum real, dicha saturación se aprecia en torno a los 500-1000 coeficientes cepstrales y el porcentaje de aciertos es superior cuando se elimina el $c_x(0)$ (40% frente a 28%). En cuanto al cepstrum complejo, la saturación se produce entre los 1000 y 2000 coeficientes, y el porcentaje de aciertos también es superior cuando se elimina el $\hat{x}(0)$ (52% frente a 42%).

Por tanto, se obtienen mejores resultados con el cepstrum complejo sin considerar el coeficiente de cuelfrencia 0. Ahora se selecciona el número de índices y el de coeficientes cepstrales, para lo cual se ha realizado un estudio con 3 índices y 1500 coeficientes, 531 índices y $[8, 40, 1500]$ coeficientes (Tabla VI).

TABLA VI
PORCENTAJE DE ACIERTOS (%) PARA CEPSTRUM COMPLEJO CON 3 Y 531
ÍNDICES, Y DISTINTOS TAMAÑOS DE VENTANA DE LIFTERING

Componentes cepstrum complejo	3x1500	531x8	531x40	531x1500
Porcentaje de aciertos (%)	58.46	33.31	53.99	64.19

Se observa que con el clasificador de distancia euclídea y 1500 coeficientes se obtiene un 58% para 3 índices y un 64% para 531. Sin embargo, el tamaño de los vectores resultantes en este último caso es notable, por lo que las siguientes pruebas se realizarán con 3x1500 (vectores de 4500 componentes) y 531 índices con 8 coeficientes, proporcionando 4248 elementos como en AMDF.

D. Discusión de los resultados

Con estas consideraciones se ha redefinido el marco de pruebas a realizar, las cuales se dividen en los dos tipos de test considerados. El test de Jackknife se realizará para el conjunto de enzimas y el subconjunto de entrenamiento de localización subcelular. Para las representaciones AAC/PseAAC se utilizarán los clasificadores basados en distancia y probabilidad, mientras que para la función AMDF y el cepstrum complejo solo se podrá utilizar el clasificador de distancia euclídea, ya que los otros implican la inversión de una matriz de covarianzas singular en este caso. Para el otro experimento basado en entrenamiento y test se utilizarán los dos subconjuntos de localización subcelular y se añadirá el clasificador SVM a todas las representaciones, considerando los dos tipos de clasificación multiclase.

En primer lugar, en la Tabla VII se muestran los resultados de los clasificadores basados en distancia y probabilidad para los tres test realizados. En el caso de distancia euclídea, las cuatro primeras representaciones muestran un comportamiento parecido para los tres test, mientras que para el cepstrum complejo con 3x1500 el porcentaje aumenta hasta el 53-58%. Por otro lado, los algoritmos ProtLock y discriminante covariante aumentan los porcentajes de aciertos de las representaciones AAC y PseAAC con 3 y 531 índices, consiguiendo un máximo de 72% con el algoritmo del discriminante covariante y 531 índices.

Por otra parte, en la Tabla VIII se muestran los resultados obtenidos con SVM para los dos clasificadores multiclase, los tres kernels y las seis representaciones consideradas. Se observa que el porcentaje aciertos para el caso de SVM uno contra el resto obtiene mejores resultados que el caso de SVM por pares, siendo RBF el mejor kernel en prácticamente todas las representaciones. Los mejores resultados se obtienen con AAC y PseAAC, así como con la función AMDF, la cual alcanza el 87% de aciertos.

VII. CONCLUSIONES

En el presente trabajo se ha realizado un estudio sobre distintos métodos de clasificación de proteínas tratadas como señales numéricas. A pesar de que los porcentajes de identidad promedio para cada clase de las bases de datos escogidas eran bajos, se han conseguido predecir las clases originales de cada proteína con un porcentaje de aciertos bastante elevado en algunos casos.

TABLA VII
PORCENTAJE DE ACIERTOS (%) CLASIFICADORES DE DISTANCIA EUCLÍDEA, PROTLOCK Y DISCRIMINANTE COVARIANTE

Tipo de test	Clasificador	Representación					
		AAC	PseAAC 3 índices	PseAAC 531 índices	AMDF 531x8	Cepstrum complejo 3x1500	Cepstrum complejo 531x8
<i>Test de Jackknife (Base de datos de enzimas)</i>	Distancia euclídea	46.32	47.47	47.39	45.76	53.64	39.20
	Algoritmo de Protlock	51.01	53.88	53.44	---	---	---
	Discriminante covariante	63.59	69.88	72.03	---	---	---
<i>Test de Jackknife (Base de datos de localización subcelular)</i>	Distancia euclídea	34.10	35.24	36.02	33.65	53.19	34.04
	Algoritmo de Protlock	39.03	34.68	37.22	---	---	---
	Discriminante covariante	61.30	67.19	67.94	---	---	---
<i>Entrenamiento y test (Base de datos de localización subcelular)</i>	Distancia euclídea	29.76	31.93	31.73	30.35	58.46	33.31
	Algoritmo de Protlock	57.59	56.79	58.16	---	---	---
	Discriminante covariante	61.62	68.80	69.21	---	---	---

TABLA VIII
PORCENTAJE DE ACIERTOS (%) CLASIFICADORES SVM MULTICLASE

Tipo de test	Clasificador	Kernel	Representación					
			AAC	PseAAC 3 índices	PseAAC 531 índices	AMDF 531x8	Cepstrum complejo 3x1500	Cepstrum complejo 531x8
<i>Entrenamiento y test (Base de datos de localización subcelular)</i>	SVM uno contra el resto	Producto escalar	85.88	76.07	85.21	84.20	61.90	76.00
		Cuadrático	73.41	76.89	75.52	79.02	60.50	73.84
		Función base radial	85.10	85.99	85.51	87.21	61.21	72.44
	SVM por pares	Producto escalar	44.02	44.38	43.33	40.67	66.32	67.42
		Cuadrático	46.47	46.68	45.12	46.52	66.71	71.32
		Función base radial	75.26	76.25	75.79	81.02	66.76	70.13

En cuanto a las representaciones de señal y algoritmos de clasificación considerados se puede comentar lo siguiente:

- La composición aminoácida (AAC) ha demostrado una vez más ser una representación bastante robusta, pues produce buenos resultados a pesar de su sencillez. Al añadir información sobre la posición que ocupa cada aminoácido en la secuencia (PseAAC) la clasificación mejora, principalmente en los algoritmos basados en probabilidad.

- La función AMDF consigue mejores resultados con 531 índices que con 3 ya que el número de componentes a considerar es superior. Por otro lado, el hecho de aumentar la distancia de correlación hasta el máximo posible no mejora la clasificación. Para esta representación se justifica el empleo de clasificadores SVM debido a que trabajan bien con grandes conjuntos de datos.

- El cepstrum complejo logra mejores resultados que el cepstrum real, por lo que es asumible que fase de la transformada de Fourier contiene información relevante para este tipo de señales. No obstante, se necesitan alrededor de 1500 coeficientes cepstrales por cada índice de aminoácido para producir buenos resultados, generando un coste computacional demasiado alto.

Asimismo, la representación con 3 índices y 1500 coeficientes consigue superar a las demás representaciones con el clasificador de distancia euclídea, lo cual no ocurre para 531 índices y 8 coeficientes. Este hecho refleja la enorme distorsión producida en el espectro de la señal al disminuir la ventana de *liftering* a un tamaño tan reducido.

En comparación, cuando se emplean los 531 índices, la representación basada en cepstrum complejo mejora sus resultados al aumentar el número de coeficientes considerados, mientras que la función AMDF no consigue mejores resultados al incrementar la distancia de correlación.

En resumen, podemos decir que el cepstrum complejo es la representación que ha demostrado contener más cantidad de información, repartida en un número amplio de coeficientes cepstrales, por lo el manejo de la misma se hace intratable en clasificadores sofisticados como los basados en probabilidad.

AGRADECIMIENTOS

A mis tutores Victoria Sánchez y Antonio Peinado, por guiarme en la realización de este trabajo. A mis padres y a mi hermano, por apoyarme y confiar en mí. A mis amigos, por sacarme siempre una sonrisa. A mis compañeros de clase, por acompañarme en este camino. A todos, gracias.

REFERENCIAS

- [1] T. Plötz, *Advanced Stochastic Protein Sequence Analysis*, Bielefeld, 2005.
- [2] K. C. Chou and D. W. Elrod, "Prediction of enzyme family classes," *Journal of Proteome Research*, vol. 2, pp. 183-190, 2003.
- [3] K. C. Chou and Y. D. Cai, "Prediction and classification of protein subcellular location—sequence-order effect and pseudo amino acid composition," *Journal of Cellular Biochemistry*, vol. 90, pp. 1250-1260, 2003.
- [4] K. C. Chou, "Prediction of protein cellular attributes using pseudo-amino acid composition," *PROTEINS: Structure, Function, and Genetics*, vol. 43, pp. 246-255, 2001.
- [5] B. Liu, X. Wang, Q. Chen, Q. Dong and X. Lan, "Using amino acid physicochemical distance transformation for fast protein remote homology detection," *PLoS ONE*, vol. 7, no. e46633.

doi:10.1371/journal.pone.0046633, 2012.

- [6] M. J. Ross, H. L. Shaffer, A. Cohen, R. Freudberg and H. J. Manley, "Average magnitude difference function pitch extractor," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 22, no. 5, pp. 353-362, 1974.
- [7] A. M. Peinado, J. L. Carmona, N. Bochud, Á. M. Gómez y G. Rus, «Representación cepstral de ultrasonidos para evaluación,» de *TecniAcustica*, León, 2010.
- [8] R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification*, Wiley, 2000.
- [9] B. Schölkopf and A. J. Smola, *Learning with Kernels. Support Vector Machines, Regularization, Optimization, and Beyond*, Massachusetts: The MIT press, 2002.
- [10] W. S. Noble and P. Pavlidis, "Gist," UBC Centre for High-Throughput Biology, 2002. [Online]. Available: <http://www.chibi.ubc.ca/gist/>. [Accessed 06 2015].



Autora: Amelia O. Villegas Morcillo (Granada, 1993). Recibió su título de Grado en Ingeniería de Tecnologías de Telecomunicación (con especialidad en Sistemas de Telecomunicación) por la Universidad de Granada en 2015, año en el que comienza sus estudios de Máster en Ingeniería de Telecomunicación en la Universidad Politécnica de Madrid, con la intención de seguir ampliando sus conocimientos en la rama de bioingeniería.



Tutora: Victoria Sánchez Calle
Es profesora titular del Departamento de Teoría de la Señal, Telemática y Comunicaciones de la Universidad de Granada y trabaja como investigadora permanente en el grupo SIGMAT. Es autora de numerosas publicaciones relacionadas con el procesamiento de señal, audio, vídeo y voz. Entre sus principales líneas de investigación se encuentran el procesamiento de señal aplicado a la proteómica y la transmisión robusta de señales multimedia.



Tutor: Antonio M. Peinado Herreros
Es catedrático del Departamento de Teoría de la Señal, Telemática y Comunicaciones, así como coordinador del grupo de investigación SIGMAT. Es autor de numerosas publicaciones, incluyendo el libro *Speech Recognition Over Digital Channels*. Entre sus principales líneas de investigación se encuentran el reconocimiento robusto de voz, la codificación de audio y el procesamiento de señal ultrasónica.

Application of the method of the instantaneous power in automatic picking of synthetic explosions en the experiment TOMO-ETNA

Author: Francisco José Cuberos Muñoz, e-mail: francicm92@correo.ugr.es

Tutors: M. Carmen Benítez Ortúzar, e-mail: carmen@ugr.es

Luz García Martínez, e-mail: luzgm@ugr.es

Degree: Degree in Telecommunications Technologies Engineering

Department of Signal Theory, Telematic and Communications

University of Granada

Abstract—Nowadays, the study of volcanoes and earthquakes associated with their eruptions is of great importance. Due to this reason, the experiment TOMO-ETNA has been carried out within the framework of the European Project MED-SUV.

Based on the study and monitoring of one of the most active volcanoes in the world, Etna volcano, which is placed in Sicily (Italy). The main objective is to make a high resolution tomography of the volcano studying the propagation velocity of P seismic waves, in order to know his internal structure with higher accuracy.

This project carries out the study of the method of the specific instantaneous power, in order to apply it to seismic signals which have been recorded by the seismic stations. This method improve signal to noise ratio in seismic signals and allows to get P wave arrival times with higher accuracy. Resulting in a better tomography of the volcano.

Key words—Earthquakes, Etna, experiment, method of the specific instantaneous power, P seismic wave, seismic signals, signal to noise ratio, tomography, volcano.

I. INTRODUCTION

A BIG part of world's population is concentrated around very active volcanic and seismic areas. The humanity lives and develops in a planet in constant motion, where volcanoes play a crucial role in the life and the environment of human being. For causes of development and survival, humanity, since its inception, has established its urban nucleus very close to volcanic areas. Mainly, due to minerals that generate volcanic emanations and do very productive lands that surround them.

In this way, volcanoes simultaneously represent both wealth and risk effects for those living in the surrounding areas. However, the fact that the economy of the first civilizations was based mainly on agriculture, justified the risk of living with such threats. Although volcanic eruptions and earthquakes are a serie of events and natural phenomena that represent a potential risk for human security, including the environment, also have played a crucial role in the development of societies.

Volcanic and seismic phenomena, in all its varied manifestations, have awakened the interest of scholars and the curious of the matter through the ages of history. From antiquity to modern times, major disasters have been described: from earthquakes that have destroyed entire populations and caused

thousands of victims to volcanic eruptions which have buried villages and destroyed the entire ecosystem surrounding.

Due to these reasons, an experiment named TOMO-ETNA has been carried out within the framework of the European Project MED-SUV.

II. DEFINITION OF SEISMIC WAVES (P AND S)

The seismic waves [1] are waves which propagate outward from the place in the interior of the Earth where the earthquake occurred. There are two main types of waves, the internal waves (P and S) who are travelling through the interior of the Earth and surface waves, making it only by the Earth's surface.

In this project are of interest the internal seismic waves only, specifically P waves but also the main characteristic of the S waves are described.

A. P seismic waves

The primary waves, also called P waves [1] are so called due to the fact that they are the first to reach the Earth's surface. They are pressure or longitudinal waves, so vibration produced in the material has the same direction as the wave propagation. They produce a movement of compression and expansion in the material. They have higher propagation velocity than the rest of the waves and can traverse both solids and liquids materials.

B. S seismic waves

The secondary or S waves [1] owe their name to the fact that reach the Earth's surface after the P waves. Therefore, they have lower propagation velocity. They are cut or transverse waves, so they produce a temporary deformation of the material perpendicular to the direction of the wave propagation. They just travel through solid material.

III. ETNA VOLCANO

Etna volcano [2] is a stratovolcano located in the East coast of Sicily, close to Messina and Catania. It is the highest active volcano in Europe, whose elevation today is 3322 metres approximately. The base of the volcano has an amplitude of 150 kilometers and a surface covering 1190 square kilometers whose base circumference is around 140 kilometers.

These characteristics make Etna one of the three biggest active volcanoes in Italy, along with Stromboli and Vesuvius, which is the biggest. Due to its constant volcanic activity, Etna volcano is considered as one of the most active volcanoes that exist in today's world, so it maintains its activity constantly. As a result of the volcanic activity of Etna volcano so close to inhabited areas, has been recognized by the Organization of the United Nations as one of the "Volcanoes of the Decade".

IV. SEISMIC TOMOGRAPHY

A seismic tomography [3]-[4] is a powerful tool that is allowing a major advance in the knowledge of the internal structure of the Earth. It is based on the study of the propagation velocity of P and S seismic waves through the interior of the planet, using high sensitivity digital seismographs (earthquake recording instruments) distributed over the surface. Seismic data are recorded on computer support, stored and undergo a computational processing.

The objective is to locate anomalies in the velocity of seismic waves that, coming from different directions, they travel through an area at a given depth. These anomalies are due to changes of temperature, physical state and composition of the interior of the Earth. They are processed by the computer allowing to obtain a reconstruction in 2D or 3D of the internal structure of the Earth.

In the case of P waves, the areas of negative anomalies (lower velocity than the initial model) have been associated with areas with higher temperature and lower density molten material, among other characteristics. Positive anomalies (higher velocity than the initial model), on the other hand, have been associated with denser materials.

Depending on the type of experiment and the data obtained, it has performed seismic tomographies that can go from a few hundred meters (e.g., archaeological studies) to global tomographies of the Earth.

A. Seismic tomography in Etna volcano

The dynamics of eruption of Etna volcano [2] is the result of a complex interaction between the magma ascension and the tectonic regime of the area (deep cortex), along with local structures (cortex surface) controlled in part by the instability of the eastern flank of the volcano.

The conditions of magma ascension (configuration of the structure, tectonic forces) are still unknown. The main limitations in the understanding of the eruption dynamics of this volcano are due to poor knowledge of its medium depth crust.

In Etna volcano [2] seismic tomography studies carried out, based on natural seismicity and made using velocities of P and S seismic waves [1], have shown that intrusion by dykes (ducts that make their way through the rock) before the eruptions may be associated with anomalies in the ratio V_p/V_s in the volcanic surface ducts system.

Although Etna volcano [2] is one of the best studied volcanoes using seismic tomography, there are a number of shortcomings in the information that currently limits the reliable determination of restrictions in volcanic processes.

Therefore, a new seismic experiment has been considered necessary to deal with the knowledge of the internal structure of Etna volcano roots [2] and the crust that surrounds it. Ranging from the volcano base to the top layer.

The results to be obtained in this experiment will form an important source of information. It will facilitate the understanding of the relation between volcano and tectonic plates, which still have some deficiencies that the seismic experiment can clarify. As well as the rise of magma and the eruption dynamic of this volcano.

With this vision in mind, it was planned to carry out the active seismic experiment TOMO-ETNA [5] during the summer of 2014.

V. TOMO-ETNA EXPERIMENT DESCRIPTION

The experiment TOMO-ETNA [5], which is part of the European project "MEDiterranean SUPersite Volcanoes (MED-SUV)" was coordinated by the National Institute of Geophysics and Vulcanology (Italy) along with the University of Granada (Spain). The experiment includes financing agencies from Italy, Spain and Germany.

The global aim of this experiment is to cover the investigation of the Etna volcano [2] and surrounding areas by means of active and passive refraction/reflection seismic methods. The main objective is to get a new high resolution tomography [3]-[4] to improve image 3D of crust structures, which exist under the volcano.

During the experiment, were realised around 9500 air shots along a route of about 2560 kilometers, generating artificial seismic signals which were recorded by a number of seismic stations placed around the volcano. Three types of stations were used, short period stations or "cubes", portable broadband stations and permanent broadband stations.

The source of acoustic pulses consisted in a formation of two batteries of air shots, each having 8 guns. The guns were maintained to a depth of 15 meters below the sea level and the shots were carried out every 90 seconds, this shooting interval was used to prevent records contaminated by noise induced by previous shots. The configuration of air guns was timely chosen through previous simulations to obtain lower frequencies and allow a better penetration of signals in the subsoil.

Therefore, the objective of this project within the experiment is to process automatically the huge amount of data obtained by stations described previously to obtain the arrival times of P seismic waves (picking) [1]. Which are contained in the seismic signals generated by the shots. This will be possible using the picking algorithm AMPA (algorithm based on multiband adaptive processing) [8].

In addition, the method of specific instantaneous power [9]-[10] will be applied to the recorded signals in order to improve the SNR of these signals to reinforce the picking. In such a way that these times can be used as input data to obtain a [3]-[4] higher resolution tomography of the volcano.

VI. STATE OF ART IN AUTOMATIC PICKING TECHNIQUES

A. Automatic phase picking

Automatic phase selectors [6] are algorithms used to scan seismic traces looking for phase arrivals. When an arrival is detected, they are used to measure the relevant parameters required by a seismologist who studies the earthquake.

A "selector" in this way incorporates the function of a "detector", i.e., recognition of a seismic phase arrival in presence of background noise. Parameters measured will include the arrival time and some measure of the event size, as well as the address of the first movement and a number expressing the "weight" or confidence to be taken into account in a future treatment.

A selector should plan a good arrival phase with an accuracy of not less than a few hundredths of second. While in a detector is normally allowed at least one second margin, although usually more.

B. STA/LTA algorithm

Nowadays, the STA/LTA algorithm [7] (short time average/long time average) is the algorithm used in weak motion seismology for events detection. Continuously it calculates the average of the absolute amplitude values of a seismic signal in two consecutive time windows.

The short time average (STA) is sensitive to seismic events while the long term average (LTA) provides information about the temporal amplitude of seismic noise in the workplace. When this ratio exceeds a preset value, an event is declared and data begin to be recorded.

The success in capturing seismic events depends on a proper configuration of trigger parameters, in order to obtain higher sensitivity as possible for a given seismic signal type.

C. AMPA algorithm

The picking algorithm based on adaptive multiband processing (AMPA) [8] is oriented to the estimation of the beginning of P seismic waves. Whose objective is to determine the arrival time of signals strongly affected by noise.

The measurement precision of the arrival time of a seismic wave has a great importance in many seismic applications. The P wave is the first wave to reach a station, since it has higher propagation velocity and denotes the beginning of the earthquake.

The confidence interval of the estimated arrival time will depend on the SNR, as well as the spectral differences between seismic signals and background noise.

This algorithm proposes two important contributions for the detection of P wave arrival time.

- 1) A background noise reduction stage based on an adaptive multiband processing.
- 2) A filter stage that enhances the earthquake arrival, minimizing signals response that not belong to earthquakes as other types of natural signals or noise caused by human.

The signal is processed to minimize the noise effects and improve the response to a seismic event. The algorithm effectuates a multiband adaptive analysis including envelope

detection and noise reduction for each band, as well as a filter which improves the response to an earthquake arrival.

Therefore, can be obtained a signal characteristic function that can be used to perform an accurate picking.

VII. PICKING IN TOMO-ETNA EXPERIMENT

Seismic stations record a large number of seismic signals. Because of this, it was necessary to carry out a process to automate the application of the method of the specific instantaneous power to all signals of the data base, whose main objective is to improve the SNR of each signal. Although signals have three components (Z, E and N) only will be processed the Z component because it contains the most part of the information.

Picking algorithm AMPA [8] is applied to signals which have the instantaneous power applied in order to obtain more precise P wave arrival times. Once obtained these times, it applies a quality or trust parameter to reduce the information and get the best quality.

Finally, it gets a final file with this classified information that will be used as input for the program that will perform the tomography. Using these files obtained from the improved picking times, it will be possible to obtain a better tomography of the volcano.

Original signals and signals with method of the instantaneous power applied picking times were obtained, for a later comparison between them to check the improvement of the latest.

An important point to keep in mind will be to apply the algorithm AMPA [8] to the original signals and signals with the power applied. Depending if it is one type of signal or another the range of frequencies that should cover the algorithm AMPA [8] internal filters bank have to be modified.

In the case of the original signals this frequency range will be [4-12] Hz (fig. 1), because the most signal frequency content is found there. On the other hand, in the case of signals with the power applied the frequency range will be [8-20] Hz (fig. 2), because to obtain the power signal is necessary to multiply two signals whose frequency content is between [4-12] Hz.

Therefore, frequency content of the resulting signal will be double.

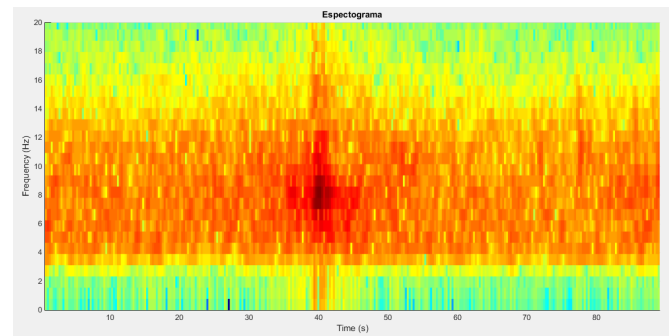


Fig. 1. Spectrogram of original Z component shot

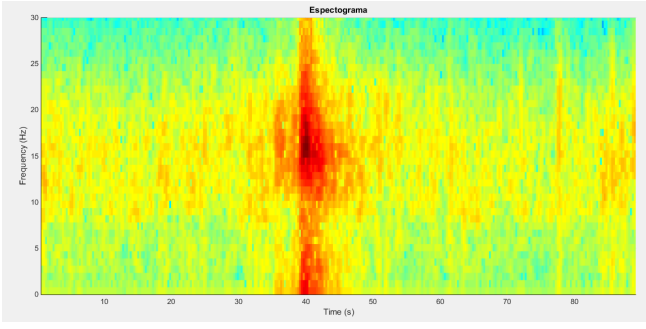


Fig. 2. Spectrogram of shot Fig. 1 with power applied

A. Specific instantaneous power

The instantaneous power [9]-[10] developed by a certain force is equal to force by velocity product in the point where it applies, and all multiplied by the cosine of the angle formed by force and velocity vectors:

$$P = |\vec{F}(t)| \cdot |\vec{v}(t)| \cdot \cos(\widehat{F(t), v(t)}) \quad (1)$$

This definition is applied to air shots which were carried out in the experiment. When a shot is performed, it impacts against the marine bottom with a certain force and velocity that have the same direction. Consequently, it produces a ground movement with a determined velocity and acceleration in a determined direction.

Velocity and force vectors have the same direction at the moment when they impact on the ground. The angle that velocity and force vectors formed is zero, so the last expression would be:

$$P = |\vec{F}(t)| \cdot |\vec{v}(t)| \quad (2)$$

It is well known that $|\vec{F}(t)| = m \cdot |\vec{a}(t)|$. Therefore:

$$P = m \cdot |\vec{a}(t)| \cdot |\vec{v}(t)| \quad (3)$$

We are treating with signals so we don't have any mass "m" variable. This unknown is cleared from the equation and it passes dividing to the power variable. Resulting in the obtaining of the specific instantaneous power "p(t)" [9]-[10], i.e., the translation power developed by the force at the moment when the wave hits a particle per unit of mass:

$$p(t) = |\vec{a}(t)| \cdot |\vec{v}(t)| = \left| \frac{d\vec{v}(t)}{dt} \right| \cdot |\vec{v}(t)| \quad (4)$$

If the shot hits the ground and there is low background noise, it will produce a good link between acceleration and velocity vectors and this will allow to stand out the signal over the noise.

On the other hand, if there is a higher background noise this will interfere in the link between acceleration and velocity vectors. Although SNR is improved, the resulting signal won't be so good and consequently it will be more difficult to differentiate what is noise in the signal and what not.

Signal derivative is performed using segmental interpolation, also named splines [11]. It samples the signal obtaining a number of discrete points that form intervals and each interval

divides itself in small subintervals adding an additional number of samples in each one.

In each interval is used a grade 3 polynomial, which allows to obtain more precise and softer curves for interpolation linking segments in each subinterval. The connection is carried out through certain continuity conditions that exist in intervals extremes. The result will be a better signal representation.

VIII. RESULTS

A series of results which demonstrate the validity of applying this method to the seismic signals recorded by stations are shown.

In each example that will be shown, it has performed a comparison between the original unfiltered signal, the filtered signal to try to reduce as far as possible the environmental noise, the signal with the power applied with the three components simultaneously and the signal with the power applied only with the Z component.

In Fig. 3 it shows the analysis of a good quality signal, in Fig. 4 it shows a medium quality signal and in Fig. 5 it shows a bad quality signal.

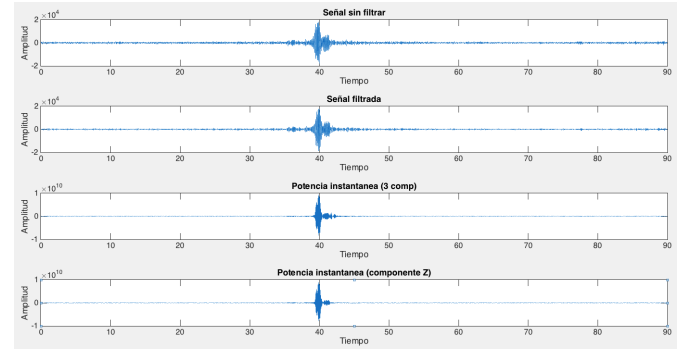


Fig. 3. Good quality shot

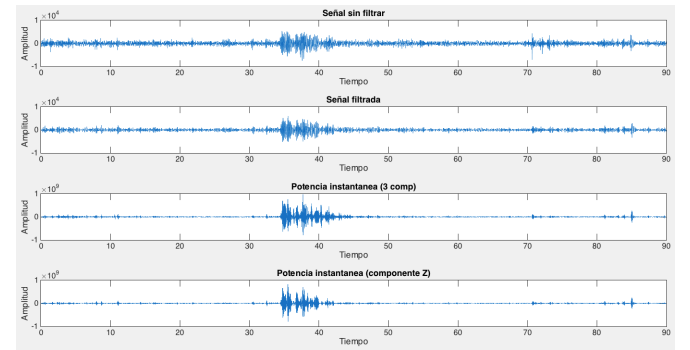


Fig. 4. Medium quality shot

In the previous examples is observed how is produced a large reduction of the noise in signals, i.e., it has improved the SNR of the original signal. In all cases this is appreciated, which leads to a general signal improvement although in the case of the good quality shot the result is better because there is less noise in the original signal.

The power with the three components simultaneously is the best signal to observe and analyze because it has the contribution of each component so it is enough only to

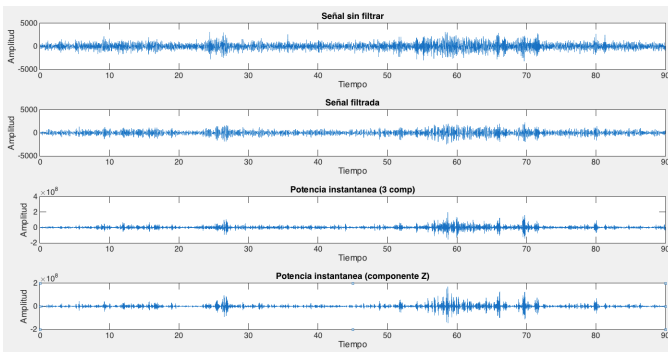


Fig. 5. Bad quality shot

represent this signal. However, it performs the comparison with the power only with the Z component to see if there is any difference between them.

In the majority of cases both signals are practically identical as it is to be expected because the Z component contains the most part of the information. This take place in the results showed previously.

However, something interesting happened. Some results were obtained and showed that there are exceptions in the previous theory. An example can be observed in Fig. 6.

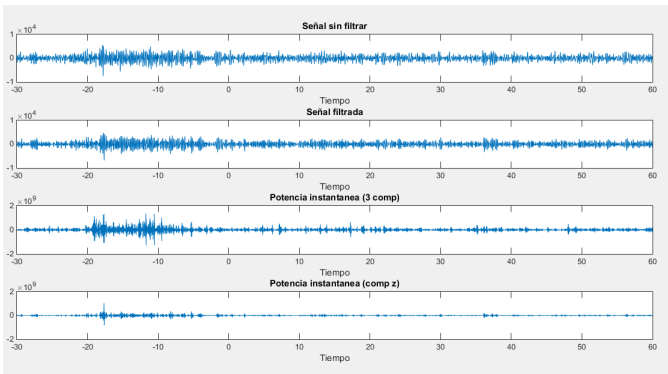


Fig. 6. Exception in the result

In the previous picture is observed how the power signal only with Z component shows an improvement in comparison with the power signal with three components simultaneously. This is because the last one has the contribution of three components so if components E and N are noisier, the noise in these signals can mask the power signal in Z component. Therefore, the result is a noisier signal.

In Fig. 7 and Fig. 8 are shown two graphics examples where can be seen clearly the improvement of performing the picking in a signal with the power applied in comparison with an original signal.

In Fig. 9 can be observed the comparison between two file extracts, belonging to permanent stations. In the first one there are original signals data, while in the second one there are signals with power applied data. It compares arrival times, the quality or confidence parameter and the SNR of each one. If the quality parameter has a high value in a certain signal, it means that the picking will be more reliable in that signal.

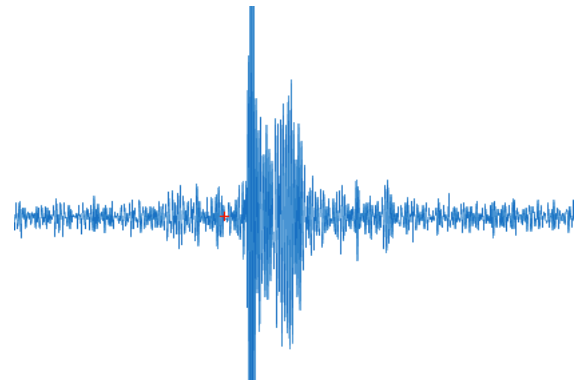


Fig. 7. Filtered signal picking

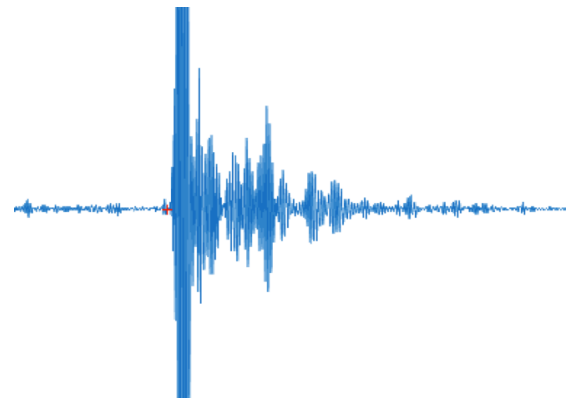


Fig. 8. Signal with power applied picking

traveltime	Quality(ZMAX)	Quality(SNR)	traveltime	Quality(ZMAX)	Quality(SNR)
5.95000	8.30650	5.24851	5.93000	8.86667	698.02796
3.66000	8.66997	38.39793	3.81000	9.87833	585.05953
6.16000	7.29940	1.96677	6.59000	8.58570	79.13694
6.40000	7.00613	2.82242	11.20000	8.87560	6.24428
6.00000	7.29676	2.88624	4.55000	7.47879	29.23972
6.16000	7.45628	9.46679	7.03000	8.52234	92.98791

Fig. 9. Picking times comparison

In this comparison can be clearly appreciated how in signals with power applied the SNR is improved and signals have a higher value of the quality parameter. Therefore, this mean that picking times are more reliable.

Finally, in Fig. 10 is shown a histogram where can be appreciated the number of signals in the Y axis, and the value of the quality parameter in the X axis.

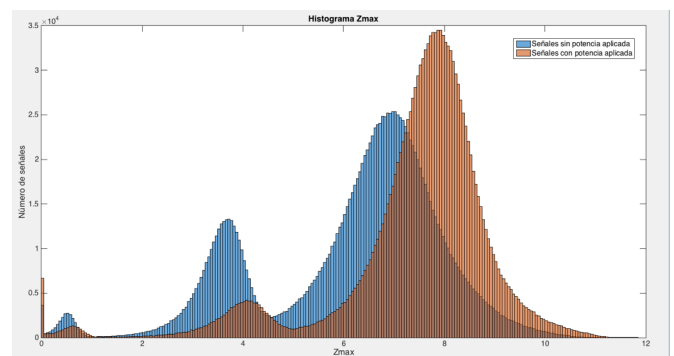


Fig. 10. Original signals and signals with power applied histogram

It can be observed how the orange graphic (signs with power applied) presents a higher number of signals with higher values of the confidence parameter, i.e., there is a large number of signals whose picking time can be considered reliable.

On the other hand, if we see the histogram of the original signals (blue graphic), we observe that there is a large number of signals with a low quality parameter value, which means that there are more signals whose picking time is unreliable.

In this way, can be appreciated the improvement produced by the application of the method of instantaneous power in seismic signals in the detection of P waves arrival time to perform the tomography. Therefore, the application of this method to recorded seismic signals improves the reliability of P wave arrival time. Which means that the resulting tomography will be more precise.

IX. CONCLUSIONS

In the present end of degree project it has performed the automatic processing of a large amount of data recorded by the network of seismic stations in the experiment TOMO-ETNA [5].

This processing has include the application of the method of the specific instantaneous power to seismic signals recorded by the seismic stations, in order to improve the their SNR. In addition, due to the magnitud of the data base it was necessary to implement an automatization process to apply this method to all signals stored in the data base.

After that, it applies the picking algorithm AMPA [8] to these improved signals. In this way, the improvement of the SNR of the signals also causes an improvement in the detection of P wave arrival time. From these times it obtains the compatible files with the tomography program. This programa will use these files as input data in order to perform a better and more precise tomography of the Etna volcano.

ACKNOWLEDGMENTS

I would like to thank Prof. Carmen Benítez and Prof. Luz García for giving me the oportunity to do this project under his guidance. In addition, also thank Prof. Gerardo Alguacil for the helpful discussions during the development of the project and the great research moments as well as the knowledge that he has transmitted to me which have been very helpful.

I also want to thank my friends, which I have met during these years of career, their constant support.

Finally, I would like to thank my parents the oportunity they gave me to study an university degree because without their help I would not have been able to realize it. I also want to mention my sister for their constant support and all my family because they have always believed in me.

Thanks to all.

REFERENCES

- [1] Instituto Nacional de Prevención Sísmica. Available from: <http://www.inpres.gov.ar/docentes/docentes.html>.
- [2] "Monte Etna: erupciones, ubicación, altura y características". Available from: <http://historiaybiografias.com/etna/>.

- [3] M. A. García, "Estudio de heterogeneidades laterales de volcanes activos: tomografía sísmica de alta resolución de la isla de Tenerife, anomalías de propagación de ondas sísmicas de la isla Decepción y otros efectos".
- [4] E. S. Lira, "Estudio de sismicidad, tomografía sísmica y modelo de física de rocas: potencial sistema geotermal asociado al complejo volcánico Tinguiririca". *Universidad de Chile, facultad de ciencias físicas y matemáticas, departamento de geofísica*, August 2011.
- [5] J. M. Ibañez, J. Prudencio, O. Cocina, L. Zuccarello and A. Díaz-Moreno, "Report on the TOMO-Etna experiment". *MEDiterranean SUPersite Volcanoes (MED-SUV)*.
- [6] R. Allen, "Automatic phase pickers: Their present use and future prospects". *Bulletin of the Seismological Society of America*, Vol. 72, No. 6 pp.S225-S242, December 1982.
- [7] A. Trnkoczy, "Understanding and parameter setting of STA/LTA trigger algorithm". September 1999.
- [8] I. Álvarez, L. García, S. Mota, G. Cortés, C. Benítez and Á. de la Torre, "An Automatic P-Phase Picking Algorithm Based on Adaptive Multiband Processing". *IEEE geoscience and remote sensing letters*, vol. 10, No. 6, November 2013.
- [9] B. Olea, G. Alguacil, F. Vidal and M. Feriche, "Parámetros de movimiento intenso y su relación con la intensidad macrosísmica en el área euro-mediterránea". *Instituto Andaluz de Geofísica, Universidad de Granada, España*, May 2011.
- [10] G. Alguacil and F. Vidal, "Medidas instrumentales de la intensidad del movimiento del suelo. Aplicación a terremotos europeos". *Instituto Andaluz de Geofísica y Departamento de Física Teórica y del Cosmos, Universidad de Granada, España*.
- [11] "Interpolación con funciones splines". Available from: http://www4.ujaen.es/angelcid/Archivos/An_Met_Num_INFORMATICA/Splines.pdf.



Francisco José Cuberos Muñoz was born in Granada on January 14th, 1992. He was graduated and received a Degree in Telecommunications Technologies Engineering with specialization in Telecommunications Systems from University of Granada in 2015.

Nowadays, he courses the oficial master of Telecommunications Engineering given in the Telecommunication and Computing Engineerings High School from the University of Granada.

Picking de terremotos sismo-volcánicos y su aplicación a la tomografía computerizada en el experimento TOMO-ETNA

Autor: Diego García Sánchez, e-mail: diegar@correo.ugr.es

Tutores: Carmen Benítez Ortúzar; e-mail: carmen@ugr.es

Luz García Martínez; e-mail: luzgm@ugr.es

Titulación: Grado en Ingeniería de Tecnologías de Telecomunicación

Departamento de Teoría de la Señal, Telemática y Comunicaciones

Universidad de Granada

Resumen— El objetivo principal de este proyecto es minimizar los efectos de los desastres naturales en el mundo, en este caso, los terremotos. Para ello, se ha llevado a cabo en el marco del Proyecto Europeo MED-SUV [1] un experimento denominado TOMO ETNA. Este experimento consistirá en realizar un gran número de explosiones controladas para realizar una tomografía del volcán Etna (Italia). Además de las explosiones controladas, la red de sismógrafos extendida capturará también los terremotos volcano-tectónicos ocurridos durante la realización del experimento y medirá los tiempos de llegada de las mismas a una red de estaciones sísmicas alrededor del volcán Etna. Dicho en otras palabras, este proyecto se propone estudiar determinadas estrategias para detectar de manera automática los tiempos de llegada de las ondas sísmicas generadas por los terremotos volcano-tectónicos y detectadas por la red de estaciones sísmicas instaladas.

Palabras clave—Terremoto, sismómetro, picking, AMPA, onda sísmica, tomografía, polarización, Etna, volcán.

I. INTRODUCCIÓN

En el mundo de la sismología volcánica, es de especial importancia conocer una serie de conceptos para comprender mejor los resultados obtenidos al final de este experimento.

A. Ondas sísmicas

Las ondas sísmicas son un tipo de onda elástica fuerte en la propagación de perturbaciones temporales del campo de tensiones, los cuales pueden generar pequeños movimientos en las placas tectónicas. Estas ondas tienen su origen en movimientos tectónicos naturales, aunque también puede ser provocada por acciones humanas; por ejemplo, explosiones. Existen dos tipos de ondas sísmicas: ondas internas y ondas superficiales.

Las ondas internas se caracterizan por describir caminos curvos, debido a la variada densidad y composición del interior de la Tierra, las cuales se clasifican en ondas primarias (P) y ondas secundarias (S).

Las ondas superficiales son ondas internas que logran

llegar a la superficie. Estas ondas se clasifican en:

- Oscilaciones libres.
- Ondas Love.
- Ondas Rayleigh.

B. Tipos y fuente de señales sismo-volcánicas

Existen muchos términos diferentes usados para la clasificación de los eventos volcánicos. Para ello, se realizará una clasificación en la que se dividirá las señales conocidas en dos tipos de señales: transitorias y continuas.

Las señales transitorias hacen referencia a señales cuya fuente actúa en un tiempo relativamente corto y posteriormente se generarán las ondas de los diversos eventos:

- Eventos sismo-volcánicos (profundos y superficiales).
- Eventos de baja frecuencia.
- Eventos híbridos, eventos multifase.

Las señales continuas en volcanes activos demuestra la profunda diferencia entre un terremoto tectónico y la sismología del volcán:

- Temblor volcánico (baja y alta viscosidad).
- Procesos superficiales.

II. ¿QUÉ ES LA TOMOGRAFÍA?

La tomografía sísmica es una potente herramienta, la cual permite al usuario representar, tanto en 2D como en 3D, la estructura interna de la Tierra en base a sus propiedades, a partir de los datos de las ondas sísmicas analizadas. Estas propiedades suelen ser la velocidad de las ondas P y S, usando el factor de calidad Q.

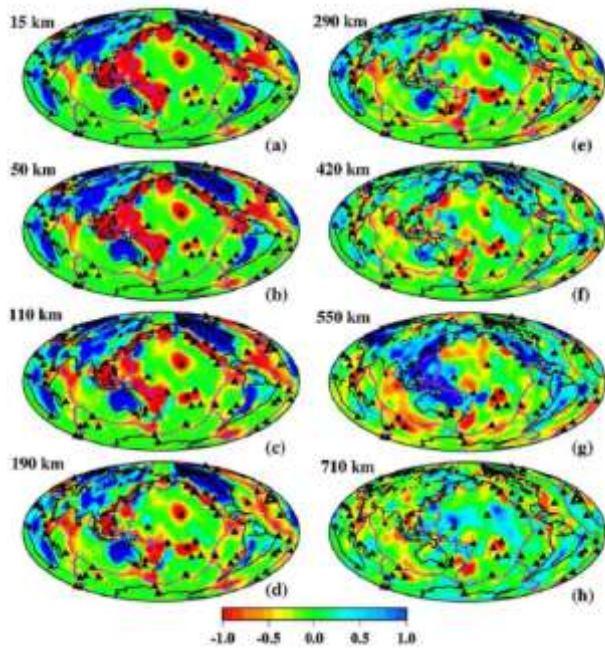


Fig. 1. Articulo [2]. Tomografía de ondas P del manto realizado por el investigador Zhao (2009).

Sin embargo, para hablar de la tomografía sísmica de un volcán es necesario saber qué es un volcán. Un volcán es una estructura geológica en el cual, a través de una caldera o cráter, efectúa la emisión de magma volcánico (roca fundida), acompañado de gases.

El ascenso de magma ocurre en diversos episodios de actividad violenta (erupciones volcánicas) que pueden variar en intensidad, duración y frecuencia. En función de esto se puede clasificar un volcán como peligroso o no, ya que no todos los volcanes son iguales debido a sus estructuras heterogéneas y complejas, tanto por sus características morfológicas como geológicas.

Cuando una onda sísmica o un disparo atraviesa un volcán, hasta llegar a un sismógrafo, el tiempo de llegada a éste puede variar, debido a que, dentro de un volcán podemos encontrar materiales sólidos (materiales vítreos), líquidos (magma, agua) y gaseosos. Todo ello provocará un cambio en la estructura interna del volcán, lo que provocará la producción de distintos tipos de erupciones.

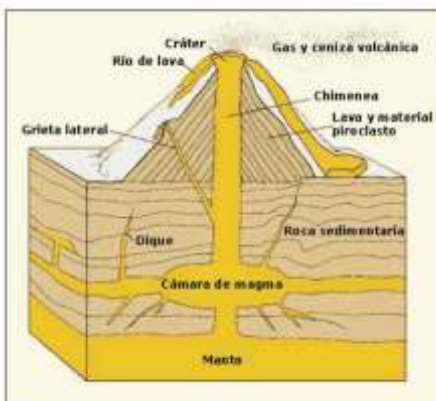


Fig. 2. Estructura de un volcán

III. DESCRIPCIÓN DEL ENTORNO DE EXPERIMENTACIÓN

Nuestro experimento se realizará en los alrededores del Etna, un volcán activo situado en la costa este de Sicilia (Italia). Tiene alrededor de 3.322 metros de altura (valor variable debido a las constantes erupciones) y es considerado el volcán activo de mayor altura de la placa Euroasiática, la cual cubre una superficie de 1.190 km², con una circunferencia basal de 140 kilómetros.



Fig. 3. Volcán Etna visto desde Catania.

Este volcán es uno de los volcanes más activos del mundo, estando casi siempre en constante erupción. Sin embargo, no está considerado como volcán peligroso, viviendo en sus alrededores miles de personas.

A. Localización de los terremotos

Un terremoto es un fenómeno que ejecuta una sacudida brusca y pasajera de la corteza terrestre, debido a la liberación de energía que ha sido acumulada en forma de ondas sísmicas. Los terremotos más comunes son los producidos por la ruptura de fallas geológicas (Fig. 4).



Fig. 4. Esquema de un terremoto y sus partes principales.

Como pueden verse en la imagen anterior, la ruptura de una falla da lugar a un terremoto, en la cual hay que destacar los siguientes puntos:

-Hipocentro o foco: es la zona en el interior de la Tierra donde se inicia la ruptura de la falla geográfica. En dicho punto es donde se inicia la propagación de las ondas sísmicas.

- Epicentro: se trata de un punto de la superficie terrestre, el cual está situado directamente encima del hipocentro. Con ambos puntos se puede determinar la profundidad exacta del origen del seísmo (hipocentro).

B. Estudio de los sismómetros y el catálogo

En el estudio de los seísmos es de gran importancia conocer de antemano el instrumento con el cual se realizará la recogida de estas señales: el sismómetro. Un sismógrafo o sismómetro es un instrumento cuya función es medir terremotos o pequeños temblores, los cuales han sido provocados por el movimiento de las placas litosféricas.

En este experimento se emplearán diversos tipos de sismómetros con el fin de registrar un determinado tipo de onda. Entre ellos se deberán de destacar:

- Cubos: El primer tipo de estaciones que vamos a describir son las denominadas *estaciones portátiles de bajo coste*. Estas estaciones se utilizan para representar los datos a lo largo de una cierta medida de interés. Estos datos pueden estar representados en 2D o 3D, incluso en más dimensiones, donde cada dimensión representa algún atributo de la base de datos.

- Estaciones fijas: Las estaciones fijas son pertenecientes al *Istituto Nazionale di Geofisica e Meteorologia* (INGV) [3]. Se caracterizan simplemente por usar unos sensores Nanometrics Trillium de 40 segundos.

- Estaciones de banda ancha: Estas estaciones, al igual que las estaciones fijas, llevan sismómetros Nanometrics Trillium Compact. Estas estaciones se caracterizan por registrar señales con gran ancho de banda, de ahí su nombre.

Con los sismómetros colocados en los alrededores del Etna, se podrá realizar un estudio detallado de varios aspectos, entre los cuales hay que destacar: magnitud del seísmo, latitud y longitud, tiempo de llegada... Según este tiempo de llegada, si la onda sísmica atraviesa el volcán hasta llegar a un determinado sismómetro, puede ser de mayor o menor magnitud, en función del estado del interior del volcán. De esta forma, se podrá deducir si el volcán trae consigo lava o no, una bolsa de gas,...

Toda esta información, la cual es muy importante para poder realizar el estudio de los seísmos, es recogida en un catálogo proveniente del INGV, el cual va actualizándose día a día. Dicho catálogo está disponible a todo el mundo en la página web citada en [3].

IV. DETECCIÓN DEL TIEMPO DE LLEGADA: MÉTODOS DE PICKING

Unos de los objetivos primordiales de este proyecto, junto con la colaboración de otros compañeros que abarcarán distintas partes de éste, es la detección de los terremotos

registrados en las distintas estaciones, clasificarlos según una estructura dada y hallar el momento exacto del momento de llegada del terremoto (picking), siendo la onda P la primera en llegar a una estación sismológica, con el fin de realizar una tomografía de la zona. Una medición exacta del tiempo de llegada de la onda sísmica es de gran importancia en muchas aplicaciones sismológicas, como puede ser la tomografía de la velocidad.

A. STA/LTA

El método STA/LTA [4] fue uno de los primeros algoritmos de detección automática, cuyo objetivo es registrar los eventos sísmicos de la forma más precisa posible.

Este algoritmo de detección mejorará significativamente el registro de los terremotos débiles en comparación con otros algoritmos de detección automática. Al mismo tiempo, disminuirá el número de registros falsos provocados por el ruido sísmico natural y artificial. Este algoritmo es más beneficioso en sitios sísmicamente más tranquilos, donde el ruido natural sísmico (por ejemplo, el ruido marino) es el ruido sísmico dominante.

El algoritmo STA/LTA tiene como objetivo realizar una serie de cambios en la amplitud del ruido sísmico, con el fin de ajustar el momento exacto de la onda sísmica. Estos cálculos serán realizados repetidamente en tiempo real, obteniéndose un considerable número de periodos.

El algoritmo STA/LTA filtra las señales sísmicas en dos ventanas de tiempo móviles: STA, que significa “ventana promedio de corto plazo”, mide la amplitud instantánea de la señal sísmica y visual para terremotos; mientras tanto LTA, que significa “ventana promedio de largo plazo”, cuida el promedio actual de la amplitud del ruido sísmico.

Primero, calculamos la amplitud absoluta de cada muestra de datos de una señal de entrada. Después, el promedio de las amplitudes absolutas de ambas ventanas. En un paso más, un ratio de ambos valores (relación STA/LTA) se calcula. Esta proporción se comparará continuamente con un valor umbral seleccionado de STA/LTA. Si la proporción es superior a este valor umbral, se declara un disparo del canal.

Además de los datos adquiridos durante el periodo de detección, las redes y registros sísmicos deberán añadir una cierta cantidad de datos sísmicos en el archivo del evento antes del registro.

B. STA/LTA+TAK

En el campo de la sismología, algunos modelos autorregresivos han sido ampliados, como pueden ser el método propuesto por Takanami y Kitagawa. Este método se encarga de proporcionar resultados más precisos que

STA/LTA. Sin embargo, si a priori la estimación de la hora de llegada no es precisa, la búsqueda del límite para los modelos autorregresivos debe cubrir un extenso intervalo de tiempo, dando lugar a una limitación computacional muy compleja. Por otro lado, el intervalo de confianza de la hora estimada de llegada depende no sólo de la relación señal ruido, sino también de las diferencias espectrales de la señal sísmica y ruido de fondo.

C. AMPA

El método AMPA [5] (algoritmo de selección multibanda adaptativa) está enfocado a determinar el tiempo de llegada de aquellas señales que están afectadas por el ruido, el cual es más exacto que el método STA/LTA y menos costoso computacionalmente que el método STA/LTA+TAK. El algoritmo AMPA propone dos importantes contribuciones a la detección del instante de llegada de la fase P: 1) una etapa de reducción de ruido de fondo basado en un proceso multibanda adaptativo, y 2) una etapa de filtrado que realiza la llegada del terremoto, minimizando la respuesta a éste, tales como otros tipos de señales naturales o humanas inducida por ruidos.

El AMPA (solo y en combinación con el algoritmo Takanami) ha sido comparado con el STA/LTA convencional (también solo y en combinación con el algoritmo Takanami) usando los tiempos de picking obtenidos. Los resultados confirman la utilidad de poder elegir el algoritmo para la identificación precisa de la llegada de la fase P. Por otra parte, cuando los algoritmos AMPA y Takanami se combinan, los resultados de cada algoritmo mejoran significativamente. La combinación de ambos algoritmos también reduce la complejidad computacional del algoritmo Takanami.

Una de las principales limitaciones de elegir entre estos algoritmos es el ruido que se puede apreciar en los sismogramas. Este ruido presenta un comportamiento muy variado desde las perspectivas de tiempo y frecuencia. Sin embargo, el problema se complica porque las señales sísmicas tienen una frecuencia de tiempo y una distribución variable.

Usando el algoritmo AMPA, la señal es procesada para minimizar los efectos de contaminación acústica y mejorar la respuesta a un evento sísmico. El algoritmo realizará un análisis de adaptación multibanda incluyendo una detección de envolvente y una reducción de ruido para cada banda y, para finalizar, un filtrado que mejorará la respuesta a la llegada de un terremoto.

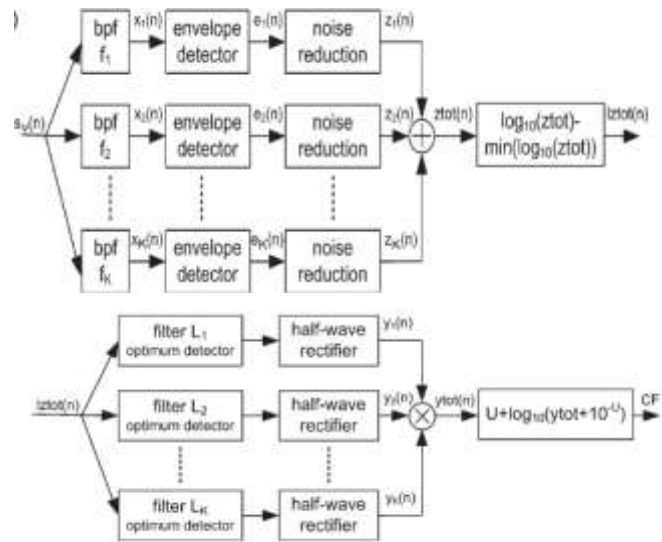


Fig. 5. Diagrama de bloques completo del proceso de adaptación multibanda

V. IMPLEMENTACIÓN PRÁCTICA DE LA PROPUESTA

Una vez conocida la teoría necesaria para la realización del siguiente experimento, se procederá a seguir cada uno de los pasos del diagrama de bloques expuesto en la figura 6:

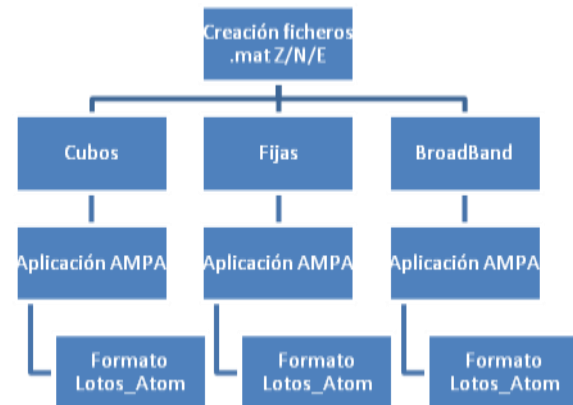


Fig. 6. Diagrama de bloques de la programación a realizar.

A. Procesado de los registros sísmicos

Tal y como se ha comentado anteriormente, hay disponible un catálogo en el cual viene recogida toda la información disponible para la localización de los seísmos registrados. Junto con este catálogo, también se tendrá una serie de ficheros de las diferentes estaciones, almacenados en diversos días. Cada uno de esos ficheros es un registro de 24 horas (es decir, cada archivo corresponde a un día en concreto), en el que viene almacenado la grabación/registro de los terremotos de dicho día. Sin embargo, esto no quiere decir que todos los archivos tenga un terremoto registrado, ya que puede ser que dicho día contenga uno, dos, varios terremotos o ninguno.

En el momento de grabación de los terremotos, estos han sido registrados en tres tipos de estaciones: cubos, fijas (INGV) y banda ancha (BroadBand). Por lo tanto, el paso importante es el siguiente: almacenar la información más

relevante de cada día y clasificar según sea la estación en la que ha sido registrada. A través de un código programado, se crearán unos archivos que corresponderán a un día en concreto, en un sismógrafo concreto y en un tipo de estación concreta.

Finalmente se obtendrá los ficheros *.mat* con la información más importante para la compilación en los siguientes códigos.

AltShots	[-7.7900,-12.6400,-16.9400,-9.6200]
AltStation	5
DTiempoShotStation	[-0.0050,-0.0050,-0.0050,-0.0050]
LatShotsGrados	[38.2300;38.0490;36.9560;38.0660]
LatStationGrados	37.3946
LongShotsGrados	[15.1560;15.1160;14.9950;15.1460]
LongStationGrados	15.0586
NumShotsDia	4
NumShotsRegistrados	4
Xshots	<9001x4 double>
fs	100
indHay	[1,2,3,4]
indNoHay	[]
tiempo	<1x9001 double>

Fig. 7. Datos de un archivo *.mat*

Por lo tanto, está será la información definitiva que tendrá los archivos creados.

B. Implementación del picking con AMPA

En este momento tendremos tres carpetas con los ficheros anteriores creados con sus tres componentes (z,n,e). Digamos que, según el diagrama de bloques Fig. 6, el proceso de programación va por el segundo nivel. El siguiente paso será pasar al tercer nivel: aplicación del AMPA.

La función de este método es muy simple: consiste en encontrar el instante exacto de llegada de la onda sísmica generada por el terremoto a la estación.

En este proceso se empleará la función *AMPA* creada y nos ofrecerá, entre otros muchos valores, el tiempo de llegada de la onda sísmica. Este proceso se realizará para los tres tipos de estaciones y en sus tres componentes.

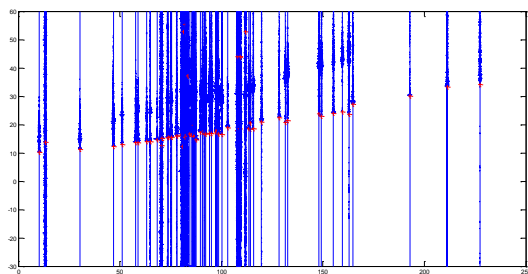


Fig. 8. Momento de llegada de un terremoto en 175 estaciones, ordenadas de menos a mayor distancia al hipocentro del terremoto.

Se puede apreciar en la Fig. 8 como el momento exacto de llegada de la onda sísmica (cruces rojas) es muy exacto y lineal. Esto es debido a que este terremoto presenta una

magnitud elevada y alcanza un rango mayor. Para un terremoto con magnitud menor, la señal registrada será más paupérrima y el picking menos exacto y lineal.

C. Resultado final

Únicamente quedaría el último y definitivo nivel: la creación de un fichero de texto llamado formato *Lotos_Atom*. ¿Qué contiene ese archivo?

Ahora mismo, tras la aplicación del AMPA, se tiene un archivo el cual contiene todos los terremotos de todas las estaciones. Sin embargo, tal y como se pudo comprobar, habrá terremotos con excelente calidad y otros pésimos.

El objetivo primordial será aplicar una serie de condiciones tales que tengamos los resultados definitivos para la creación de la tomografía. El resultado final será un archivo de texto con todos los terremotos en todas las estaciones con la calidad más óptima, aplicando para ello unas condiciones de calidad necesarias. Con estos datos se procederá a la realización de la tomografía.

D. Implementación del picking con AMPA con técnicas de polarización

La idea principal de la aplicación de un filtrado de polarización consiste en conseguir un tiempo de picking más exacto utilizando el conocimiento de la polarización de la onda sísmica. Cuando la señal buscada y el ruido tienen características similares de energía, amplitud y frecuencia este método puede ser muy útil. Para ello, se necesitará usar las tres componentes de cada tipo de estación de forma simultánea.

Para nuestro experimento hemos diseñado un filtro de polarización basado en la idea clásica de Vidale (1986, [6]). La filosofía de la técnica consiste en definir ventanas temporales a lo largo de todo el registro sísmico, en las que se analiza la matriz de covarianzas de las 3 componentes de señal registradas. Usando dicha matriz de covarianzas y mediante el análisis de componentes principales, se detecta la dirección de máxima energía de la señal. Cada una de las 3 componentes originales registradas se proyecta sobre dicha dirección de máxima energía. La suma de todas las proyecciones es la señal polarizada a la que le aplicaremos el algoritmo de picking.

E. Comparación de resultados

A continuación se procederá a realizar una comparación visual de los tiempos de picking y se observará como han sido mejorados sus valores. Para ello, se comparará resultados sin polarizar con los polarizados. Además del tiempo de picking, también podrá apreciarse un aumento de la calidad de la señal, obteniéndose una SNR considerablemente alta.

ESTACIONES	USALLEY/AGNE	USALLEY/AGNE	USALLEY/AGNE	USALLEY/AGNE	USALLEY/AGNE
4.48120	11.25100	11.25100	11.25100	11.25100	11.25100
14.48120	11.25100	11.25100	11.25100	11.25100	11.25100
24.48120	11.25100	11.25100	11.25100	11.25100	11.25100
34.48120	11.25100	11.25100	11.25100	11.25100	11.25100
44.48120	11.25100	11.25100	11.25100	11.25100	11.25100
54.48120	11.25100	11.25100	11.25100	11.25100	11.25100
64.48120	11.25100	11.25100	11.25100	11.25100	11.25100
74.48120	11.25100	11.25100	11.25100	11.25100	11.25100
84.48120	11.25100	11.25100	11.25100	11.25100	11.25100
94.48120	11.25100	11.25100	11.25100	11.25100	11.25100
104.48120	11.25100	11.25100	11.25100	11.25100	11.25100
114.48120	11.25100	11.25100	11.25100	11.25100	11.25100
124.48120	11.25100	11.25100	11.25100	11.25100	11.25100
134.48120	11.25100	11.25100	11.25100	11.25100	11.25100
144.48120	11.25100	11.25100	11.25100	11.25100	11.25100
154.48120	11.25100	11.25100	11.25100	11.25100	11.25100
164.48120	11.25100	11.25100	11.25100	11.25100	11.25100
174.48120	11.25100	11.25100	11.25100	11.25100	11.25100
184.48120	11.25100	11.25100	11.25100	11.25100	11.25100
194.48120	11.25100	11.25100	11.25100	11.25100	11.25100
204.48120	11.25100	11.25100	11.25100	11.25100	11.25100
214.48120	11.25100	11.25100	11.25100	11.25100	11.25100
224.48120	11.25100	11.25100	11.25100	11.25100	11.25100
234.48120	11.25100	11.25100	11.25100	11.25100	11.25100
244.48120	11.25100	11.25100	11.25100	11.25100	11.25100
254.48120	11.25100	11.25100	11.25100	11.25100	11.25100
264.48120	11.25100	11.25100	11.25100	11.25100	11.25100
274.48120	11.25100	11.25100	11.25100	11.25100	11.25100
284.48120	11.25100	11.25100	11.25100	11.25100	11.25100
294.48120	11.25100	11.25100	11.25100	11.25100	11.25100
304.48120	11.25100	11.25100	11.25100	11.25100	11.25100
314.48120	11.25100	11.25100	11.25100	11.25100	11.25100
324.48120	11.25100	11.25100	11.25100	11.25100	11.25100
334.48120	11.25100	11.25100	11.25100	11.25100	11.25100
344.48120	11.25100	11.25100	11.25100	11.25100	11.25100
354.48120	11.25100	11.25100	11.25100	11.25100	11.25100
364.48120	11.25100	11.25100	11.25100	11.25100	11.25100
374.48120	11.25100	11.25100	11.25100	11.25100	11.25100
384.48120	11.25100	11.25100	11.25100	11.25100	11.25100
394.48120	11.25100	11.25100	11.25100	11.25100	11.25100
404.48120	11.25100	11.25100	11.25100	11.25100	11.25100
414.48120	11.25100	11.25100	11.25100	11.25100	11.25100
424.48120	11.25100	11.25100	11.25100	11.25100	11.25100
434.48120	11.25100	11.25100	11.25100	11.25100	11.25100
444.48120	11.25100	11.25100	11.25100	11.25100	11.25100
454.48120	11.25100	11.25100	11.25100	11.25100	11.25100
464.48120	11.25100	11.25100	11.25100	11.25100	11.25100
474.48120	11.25100	11.25100	11.25100	11.25100	11.25100
484.48120	11.25100	11.25100	11.25100	11.25100	11.25100
494.48120	11.25100	11.25100	11.25100	11.25100	11.25100
504.48120	11.25100	11.25100	11.25100	11.25100	11.25100
514.48120	11.25100	11.25100	11.25100	11.25100	11.25100
524.48120	11.25100	11.25100	11.25100	11.25100	11.25100
534.48120	11.25100	11.25100	11.25100	11.25100	11.25100
544.48120	11.25100	11.25100	11.25100	11.25100	11.25100
554.48120	11.25100	11.25100	11.25100	11.25100	11.25100
564.48120	11.25100	11.25100	11.25100	11.25100	11.25100
574.48120	11.25100	11.25100	11.25100	11.25100	11.25100
584.48120	11.25100	11.25100	11.25100	11.25100	11.25100
594.48120	11.25100	11.25100	11.25100	11.25100	11.25100
604.48120	11.25100	11.25100	11.25100	11.25100	11.25100
614.48120	11.25100	11.25100	11.25100	11.25100	11.25100
624.48120	11.25100	11.25100	11.25100	11.25100	11.25100
634.48120	11.25100	11.25100	11.25100	11.25100	11.25100
644.48120	11.25100	11.25100	11.25100	11.25100	11.25100
654.48120	11.25100	11.25100	11.25100	11.25100	11.25100
664.48120	11.25100	11.25100	11.25100	11.25100	11.25100
674.48120	11.25100	11.25100	11.25100	11.25100	11.25100
684.48120	11.25100	11.25100	11.25100	11.25100	11.25100
694.48120	11.25100	11.25100	11.25100	11.25100	11.25100
704.48120	11.25100	11.25100	11.25100	11.25100	11.25100
714.48120	11.25100	11.25100	11.25100	11.25100	11.25100
724.48120	11.25100	11.25100	11.25100	11.25100	11.25100
734.48120	11.25100	11.25100	11.25100	11.25100	11.25100
744.48120	11.25100	11.25100	11.25100	11.25100	11.25100
754.48120	11.25100	11.25100	11.25100	11.25100	11.25100
764.48120	11.25100	11.25100	11.25100	11.25100	11.25100
774.48120	11.25100	11.25100	11.25100	11.25100	11.25100
784.48120	11.25100	11.25100	11.25100	11.25100	11.25100
794.48120	11.25100	11.25100	11.25100	11.25100	11.25100
804.48120	11.25100	11.25100	11.25100	11.25100	11.25100
814.48120	11.25100	11.25100	11.25100	11.25100	11.25100
824.48120	11.25100	11.25100	11.25100	11.25100	11.25100
834.48120	11.25100	11.25100	11.25100	11.25100	11.25100
844.48120	11.25100	11.25100	11.25100	11.25100	11.25100
854.48120	11.25100	11.25100	11.25100	11.25100	11.25100
864.48120	11.25100	11.25100	11.25100	11.25100	11.25100
874.48120	11.25100	11.25100	11.25100	11.25100	11.25100
884.48120	11.25100	11.25100	11.25100	11.25100	11.25100
894.48120	11.25100	11.25100	11.25100	11.25100	11.25100
904.48120	11.25100	11.25100	11.25100	11.25100	11.25100
914.48120	11.25100	11.25100	11.25100	11.25100	11.25100
924.48120	11.25100	11.25100	11.25100	11.25100	11.25100
934.48120	11.25100	11.25100	11.25100	11.25100	11.25100
944.48120	11.25100	11.25100	11.25100	11.25100	11.25100
954.48120	11.25100	11.25100	11.25100	11.25100	11.25100
964.48120	11.25100	11.25100	11.25100	11.25100	11.25100
974.48120	11.25100	11.25100	11.25100	11.25100	11.25100
984.48120	11.25100	11.25100	11.25100	11.25100	11.25100
994.48120	11.25100	11.25100	11.25100	11.25100	11.25100
1004.48120	11.25100	11.25100	11.25100	11.25100	11.25100

Fig. 9. Valores INGV sin polarización y con polarización.

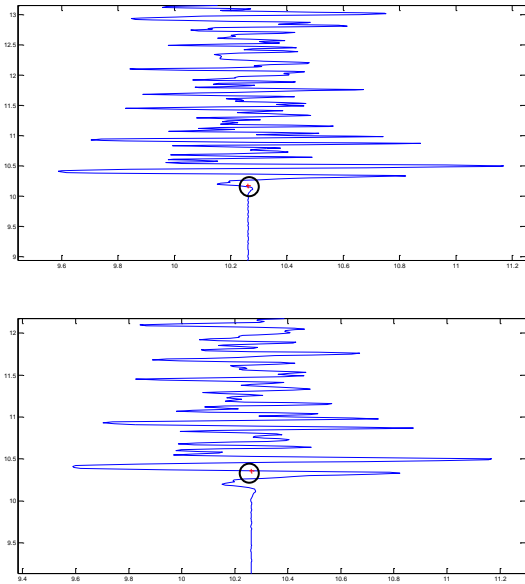


Fig. 10. Arriba: terremoto con picking no mejorado. Abajo: terremoto con picking mejorado.

La Fig. 9 muestra una comparativa de los resultados obtenidos sin polarizar y polarizados. Se puede apreciar como varían los valores de *traveltime* (picking) de manera positiva, ajustándose al tiempo exacto de llegada. De esta forma, la calidad de los resultados obtenidos también mejorará significativamente en la tomografía.

En la Fig. 10 podemos ver con claridad el momento exacto del inicio del terremoto en ambas gráficas, verificando con total seguridad la mejoría del método de AMPA con la polarización en la segunda imagen. Se puede apreciar en la segunda imagen como el momento de picking (cruz roja) está situado en el momento de mayor amplitud, correspondiente a la llegada de la onda P, mientras que en la primera imagen ese tiempo de picking está situado ligeramente antes de la llegada de la onda P.

VI. CONCLUSIONES Y VÍAS FUTURAS

El objetivo de este proyecto era obtener unos valores de picking más óptimos para poder realizar la tomografía. Para ello se ha debido de cumplir los siguientes objetivos:

- Procesado automático de los datos brutos registrados por la red de estaciones sísmicas.
- Creación de los ficheros formato *.dat* con toda la información necesaria.

- Aplicación de la polarización.
- Creación de los ficheros compatibles con el formato de entrada del programa de tomografía (Lotos_Atom).

Todo el mundo sabe que un trabajo nunca está perfecto. Siempre existirá un margen de mejora y así poder seguir dando un paso adelante en la investigación. Estas mejoras no tiene porque ser introducir pasos nuevos y ajenos a lo realizado. Pueden ser una versión sobre los códigos ya programados: una ejecución más rápida, un nuevo proceso de análisis,... Algunas de estas mejoras pueden ser las siguientes:

- Mayor número de estaciones.
- Mejorar la función AMPA.
- Usar ordenadores más potentes.

REFERENCIAS

- [1] MED-SUV, "MEDiterranean SUPersite Volcanoes". Available from: <http://med-suv.eu/>
- [2] M. A. García, "Estudio de heterogeneidades laterales de volcanes activos: tomografía sísmica de alta resolución de la isla de Tenerife, anomalías de propagación de ondas sísmicas de la isla de la decepción y otros efectos", 2010. Available from: <http://hera.ugr.es/tesisugr/18972809.pdf>
- [3] INGV, "Istituto Nazionale di Geofisica e Vulcanologia". Catálogo de los terremotos en el este de Sicilia - Sur de Calabria (1999-2015). Available from: <http://www.ct.ingv.it/ufs/analisti/catalogolist.php>
- [4] A. Trnkoczy, "Understanding and parameter setting of STA/LTA trigger algorithm", September 1990. Available from: http://gfzpublic.gfz-potsdam.de/pubman/item/escidoc:4097:3/component/escidoc:4098/IS_8.1_rev1.pdf
- [5] I. Álvarez, L. García, S. Mota, G. Cortés, C. Benítez, Á. De la Torre, "An Automatic P-Phase Picking Algorithm Based on Adaptive Multiband Processing", IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, VOL. 10, NO. 6, November 2013.
- [6] J. E. Vidale, "Complex Polarization Analysis of Particle Motion", Bulletin of the Seismological Society of America, Vol. 76, No. 5, pp. 1393-1405, October 1986.



Diego García Sánchez, 4 de marzo de 1991, Jaén. Graduado en Ingeniería de Tecnologías de Telecomunicación en la Universidad de Granada con mención en sistemas de telecomunicación y alumno del máster oficial de Ingeniería de Telecomunicación de la Universidad de Granada.

Uso de métodos de polarización instantánea avanzados para la mejora de la fase de llegada de señales sísmicas

Autor: Ismael Vico Triviño, e-mail: isvitri@correo.ugr.es

Tutoras: María del Carmen Benítez Ortúzar, e-mail: carmen@ugr.es

Luz García Martínez, e-mail: luzgm@ugr.es

Titulación: Grado en Ingeniería en Tecnologías de Telecomunicación
Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Resumen—Una de las mayores dificultades en el análisis de señales sísmicas es encontrar el evento de interés dentro de una señal ruidosa. Esto se convierte en una tarea tediosa y a veces impracticable cuando intentamos procesar el gran volumen de información generado en una estación sísmica que registra continuamente la actividad terrestre.

En esta disertación se propone un algoritmo de polarización instantánea capaz de maximizar la relación señal ruido de dichas señales sísmicas. El procedimiento comienza con el cálculo del semieje mayor de la trayectoria elíptica de las partículas del suelo para cada componente de la señal, que se encuentra dispuesta en un espacio tridimensional. Teniendo en cuenta que el ruido mantiene una amplitud constante en todas las direcciones de polarización, la proyección de dicha señal sísmica sobre los semiejes mayores de tal elipse resulta en un considerable incremento de la señal de interés frente al ruido. Asimismo, la señal también se mejora mediante un filtro de polarización muestra a muestra usando la relación entre las tres componentes de las medidas correspondientes a las tres direcciones espaciales para estimar el grado de polarización.

Este procedimiento permite utilizar algoritmos básicos de detección, ampliamente conocidos en el análisis de señales sísmicas, aumentando su robustez y fiabilidad gracias al análisis de polarización. También agiliza considerablemente el procesado de la información, así como reduce el volumen de datos almacenado, ya que selecciona únicamente el terremoto dentro de todo el registro de muestras y elimina aquellas que carecen de interés. En definitiva, se optimiza la eficiencia en el procesado de las señales sísmicas.

Palabras clave—polarización, instantánea, señal sísmica, terremoto, mejora detección

I. INTRODUCCIÓN

La necesidad de la mejora en la robustez en el instante de llegada de una onda sísmica a un sensor determinado, radica en el experimento de realización de una tomografía inversa en el volcán Etna, en Italia. Este proyecto recibe el nombre de Tomo-Etna.

Existen una serie de sensores ubicados alrededor del volcán, realizando medidas registradas en las tres componentes cartesianas del espacio (ver figura 1). Las ondas sísmicas que se registran son disparos controlados realizados desde barcos alrededor de la isla. De esta forma, podemos realizar un análisis de los tiempos de llegada y el retardo que presenta la onda tras atravesar el volcán, para caracterizar el estado del interior del mismo. Dado el ruido que aparecerá en los

registros de los sensores que puede llegar a dañar el instante de llegada, es muy conveniente mejorar la detección.

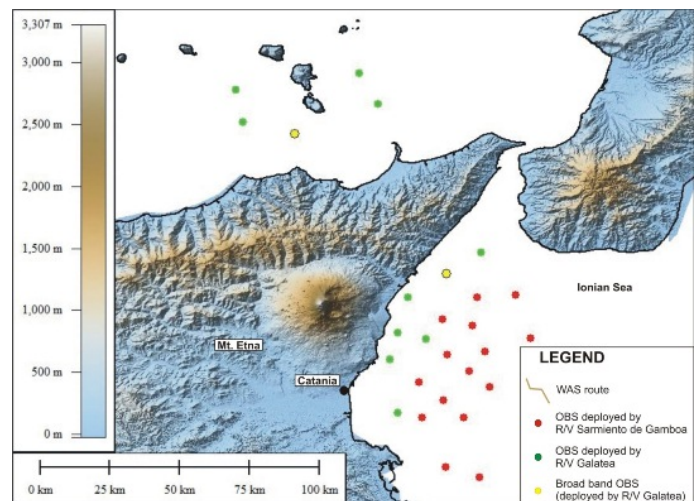


Figura 1. Distribución de sensores alrededor del volcán Etna. Imagen obtenida de [1].

Para realizar el análisis de la polarización de las ondas sísmicas que tenemos registradas usamos la herramienta Matlab, implementando el desarrollo matemático necesario para optimizar la orientación de los ejes de los sensores y adecuarla a la dirección de máxima energía de la onda.

II. MARCO TEÓRICO

A. Ondas sísmicas

Una onda sísmica es una perturbación del medio material que se genera en un punto fuente (hipocentro), y que se extiende omnidireccionalmente. La vibración que se propaga por el interior de la corteza terrestre se llama onda de cuerpo, mientras que las que lo hacen a través de la superficie terrestre desde el epicentro son las llamadas ondas superficiales.

Existen dos tipos de onda, según la forma en la que actúan perturbando el medio: ondas P y ondas S. Las P son las más rápidas y las primeras que se registran en un sismograma. Son ondas de presión, y actúan sobre la dirección de propagación. Por otro lado, las ondas S, torsionan el medio de forma

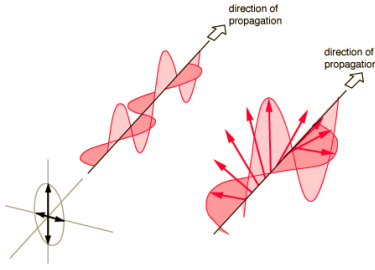


Figura 2. Polarización elíptica. Obtenida de [2]

esférica desde el hipocentro. Éstas últimas también suelen llamarse ondas secundarias o ondas de corte.

B. Polarización de una onda

Las ondas sísmicas están polarizadas en una elipse en el espacio tridimensional. Se dice que una onda está polarizada cuando su evolución espacial describe un lugar geométrico determinado, a saber, lineal, circular o elíptico (ver Figura 2). La polarización elíptica se produce cuando las distintas componentes espaciales de la onda tienen distintas amplitudes o distintas fases. Las ecuaciones (1) describen una polarización elíptica bidimensional.

$$\begin{aligned} \mathbf{E}_x(z, t) &= E_{x0} \text{sen}(\omega t - kz \pm \xi_x) \mathbf{i} \\ \mathbf{E}_y(z, t) &= E_{y0} \text{sen}(\omega t - kz \pm \xi_y) \mathbf{j} \end{aligned} \quad (1)$$

Cuando trabajamos con sensores que están orientados sobre una determinada dirección de polarización, si ésta y la de la onda que se pretende registrar no están perfectamente alineadas se producirá una pérdida de energía recibida. En el caso de la medida de una onda polarizada elípticamente, la dirección de máxima recepción de energía se encuentra sobre el semieje mayor de la elipse (ver Figura 2).

III. METODOLOGÍA E IMPLEMENTACIÓN

El método utilizado está basado en el enfoque de Gallart *et al.* en [3] ampliado con el de Morozov *et al.* en [4].

Pretendemos realizar la mejora de la señal usando filtros de polarización en el dominio del tiempo usando sistemas de datos con tres ejes. En determinados puntos el filtro puede implementar coherencia en la polarización de forma espacial (con otras componentes), de manera que puede mejorar la eliminación del ruido.

En definitiva, el problema es la eliminación de reflexiones de la señal, ya que son similares entre sí y más difíciles que detectar que un ruido con distinta frecuencia o potencia. Para ello, haremos uso de medidas direccionales coherentes, y con ello podremos atenuar el ruido por completo, incluyendo estas señales nocivas.

A. Medidas y estructura de los datos

Este procedimiento de polarización instantánea utiliza una estructura de señales con tres componentes espaciales, a través de las cuales comprueba el nivel de amplitud de una señal en una única componente, y si se encuentra también en las demás. Si es así, la señal estará poco polarizada y el algoritmo la minimizará.

B. Análisis matemático de la onda sísmica

Primero observaremos la señal analítica y su utilidad para determinar los vectores de polarización instantáneos para señales sísmicas tridimensionales. Los vectores de polarización llevan la orientación instantánea de los semiejes mayor y menor de una elipse que representa el movimiento de la señal en un espacio tridimensional. Podremos usar esos ejes para observar el grado de polarización y para poder así eliminar las señales menos polarizadas.

La señal analítica de $\mathbf{u}(t)$ es de tipo complejo, y se define sólo con una señal real y su transformada de Hilbert.

$$\mathbf{u}^c(t) = \mathbf{u}(t) + iH[\mathbf{u}(t)] \quad (2)$$

La señal analítica permite descomponer una señal como una envolvente real (la amplitud $C(t)$) y una fase instantánea real ($\phi(t)$), dependientes del tiempo (ver ecuación 3).

$$\mathbf{u}^c(t) = C(t)e^{i\phi(t)} \quad (3)$$

La ventaja de la señal analítica es que factoriza la señal como una función de baja frecuencia (envolvente, $C(t)$) y otra de alta frecuencia (fase, $\phi(t)$). Ya sabemos que la transformada de Hilbert introduce un desfase de $-\pi/2$, de hecho su forma temporal es $\frac{1}{i\pi t}$, para obtener la parte compleja ortogonal de la señal analítica. La señal analítica permite determinar atributos directamente de la señal en un análisis muestra por muestra. Así que gracias a eso podemos determinar la polarización instantánea de la señal tridimensional.

Este método puede mejorar señales con movimientos elípticos y lineales sin necesidad de tener una polarización predefinida. Las señales con polarización lineal se proyectarán sobre su dirección principal y las elípticas sobre dos direcciones (una sobre la dirección del eje mayor), otra por ortogonalidad sobre el eje menor y otra sobre otra dirección ortogonal a ambas.

1) *Medida del grado de polarización:* Las señales polarizadas son aquellas cuyas características no cambian su polarización en toda la señal (en el tiempo). Para realizar una ponderación de lo polarizada que está una señal en cada muestra, se utiliza el que llamaremos *grado de polarización* o $c(t)$ y que queda definido en la ecuación (4).

$$c(t) = \left[\frac{1}{Long + 1} \sum_{\tau=t-Long/2}^{t+Long/2} \left| \frac{\mathbf{m}(t)}{|\mathbf{m}(t)|} \frac{\mathbf{a}(\tau)}{|\mathbf{a}(\tau)|} \right|^{\mu_1} \right]^{\mu_2} \quad (4)$$

$Long$ es una ventana corta de $Long + 1$ muestras y $\mathbf{m}(t)$ es el vector de medias de polarización de la ventana de datos. $\mathbf{a}(t)$ es el semieje mayor de la elipse. Las $Long + 1$ proyecciones de los vectores unitarios de polarización se suman al vector unitario medio. Debido a la incertidumbre de fase, los valores absolutos se usan para evitar interferencias destructivas en los cambios de fase. La transición entre grados de polarización altos y bajos se controla con los superíndices μ_1 y μ_2 . Esto hace que se atenúe más el ruido cuando las diferencias entre las señales más y menos polarizadas son pequeñas, es decir, va convergiendo. μ_1 y μ_2 actúan sobre las proyecciones individuales y su suma, respectivamente.

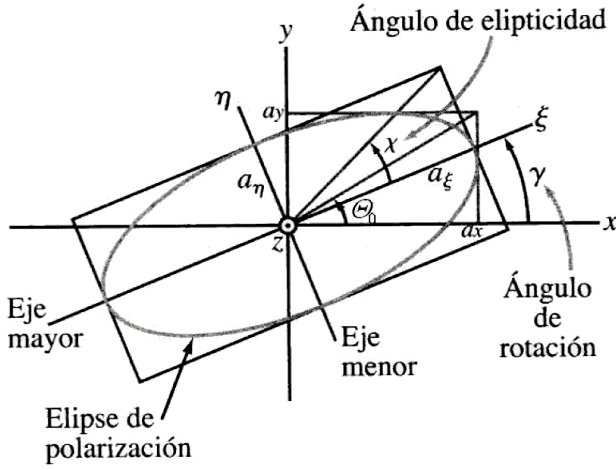


Figura 3. Elipse. Obtenida de [2]

En nuestro caso, usaremos un valor μ único para ambos exponentes. $c(t)$ es una función escalar que se mueve entre 0 y 1 indicando cómo de polarizada homogéneamente está la señal a lo largo de una ventana $Long$. $c(t)$ es igual para las tres direcciones del espacio en las que queda registrado el seísmo. Si la dirección del semieje mayor varía demasiado, las proyecciones en el vector de medias $\mathbf{m}(t)$ se hacen más pequeñas y el factor de polarización disminuirá. El vector $\mathbf{m}(t)$ es un vector de medias que caracteriza la polarización en la ventana de datos. Puede adaptarse para diferentes aplicaciones y estadísticas.

$$\mathbf{m}(t) = \frac{1}{Long + 1} \sum_{\tau=t-Long/2}^{t+Long/2} \frac{a(\tau)}{|a(\tau)|} \quad (5)$$

Usaremos la segunda aproximación para calcular la media, ya que es independiente de la amplitud al ser una relación entre amplitudes y no una media directa.

Definiremos también un vector auxiliar para el caso en el que el semieje mayor y el semieje menor sean prácticamente iguales. Si esto sucede, se pueden confundir los ejes y realizarse mal la polarización. Para ello, definimos el *vector de planaridad* (ver ecuación 6) como el producto vectorial de los dos semiejes, es decir, un nuevo vector ortogonal a ambos, con lo cual se solventa el problema de la confusión de los semiejes.

$$\mathbf{p}(t) = \mathbf{a}(t) \times \mathbf{b}(t) \quad (6)$$

2) *Deducción matemática de la maximización de la fase:* Necesitaremos entonces maximizar la ecuación 7 para poder calcular la fase que optimiza la polarización. Para ello, separaremos los términos de la ecuación y los desarrollamos para encontrar una forma más compacta de la fase (ecuación 8).

$$\Phi_r(e^{-i\psi} \mathbf{u}^c(t)) = \sum_i Re[e^{-i\psi} \mathbf{u}_i^c(t)]^2 + \epsilon \left\{ \sum_i Re[e^{-i\psi} \mathbf{u}_i^c(t)] \right\}^2 \quad (7)$$

$$\sum_i (Re[\mathbf{u}_i^c(t)e^{-i\psi}])^2 \quad (8)$$

La exponencial $e^{-i\psi}$ se descompondrá en senos y cosenos gracias a la identidad de Euler (9)

$$e^{-i\psi} = \cos(\psi) + i\sin(\psi) \quad (9)$$

Simplificamos la expresión 8

$$\sum_i Re \left[\left(\frac{\mathbf{u}_i^c + \mathbf{u}_i^{c*}}{2} + i \frac{\mathbf{u}_i^c - \mathbf{u}_i^{c*}}{2} \right) (\cos(\psi) - i\sin(\psi)) \right]^2 \quad (10)$$

si desarrollamos, finalmente obtenemos la expresión simplificada

$$\sum_i (Re[\mathbf{u}_i^c e^{-i\psi}])^2 = \frac{1}{4} \sum_i (\mathbf{u}_i^c e^{i\psi} + \mathbf{u}_i^{c*} e^{-i\psi})^2 \quad (11)$$

Una vez deducido el primer sumando, desarrollaremos el segundo:

$$\epsilon \left(\sum_i (Re[\mathbf{u}_i^c e^{-i\psi}]) \right)^2 \quad (12)$$

Si desarrollamos el cuadrado y agrupamos sumatorias:

$$\frac{\epsilon}{4} \left(\sum_i \mathbf{u}_i^{c2} e^{-2i\psi} + \sum_i \mathbf{u}_i^{c*2} e^{2i\psi} + \sum_i \mathbf{u}_i^c \mathbf{u}_i^{c*} \right) \quad (13)$$

Definimos unos nuevos parámetros auxiliares para no recargar la notación:

$$A = \frac{1}{2} \sum_i \mathbf{u}_i^{c2} \quad (14)$$

$$B = \frac{1}{2} \left(\sum_i \mathbf{u}_i^c \right)^2 \quad (15)$$

Para calcular la maximización de la fase de la polarización elíptica, compactamos la ecuación 7, además de sustituir $A + \epsilon B$ por Z para simplificar.

Derivamos

$$\begin{aligned} 0 = -2iZ e^{-2i\psi_0} &= 2iZ^* e^{-2i\psi_0} = \\ &= -2iZ (\cos(2\psi_0) - i\sin(2\psi_0)) \\ &+ 2iZ^* (\cos(2\psi_0) - i\sin(2\psi_0)) = \\ &= -2iZ \cos(2\psi_0) + 2Z \sin(2\psi_0) \\ &+ 2iZ^* \cos(2\psi_0) - 2Z \sin(2\psi_0) \end{aligned} \quad (16)$$

donde ϵ es un factor de regularización de B.

Si desarrollamos, tenemos finalmente:

$$\psi_0 = \frac{1}{2} \arg(\mathbf{A} + \epsilon \mathbf{B}) + k\pi \quad (17)$$

con $k \in \mathbb{Z}$

Y esta es la fase que utilizaremos para calcular la polarización. El factor $k\pi$ se debe a la periodicidad de la arcotangente, cuyo periodo es π , con lo cual, se introduce incertidumbre de un número entero de veces π .

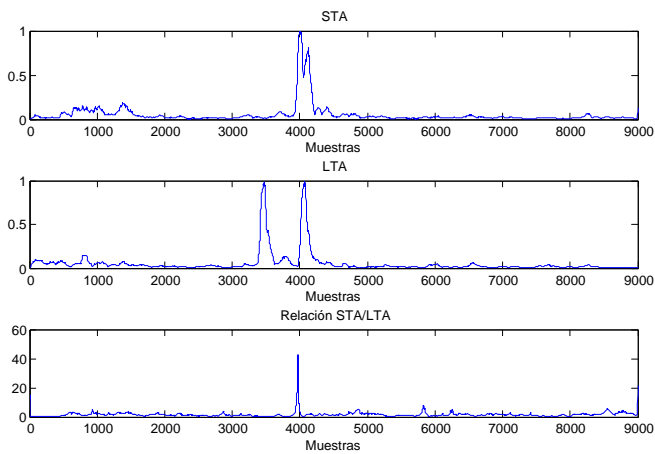


Figura 4. Promediado de muestras en ventana corta (STA, primera figura), larga (LTA, segunda figura) y cociente STA/LTA (tercera figura).

C. Algoritmos de detección

1) *Algoritmo STA/LTA*: Para poder comprobar la fiabilidad del método que se plantea, es necesario utilizar un algoritmo de detección del instante temporal en que se registra la primera llegada de la onda en el sismograma. En el tratamiento de señales sísmicas, es ampliamente utilizado el STA/LTA (Short Time Average / Long Time Average). Este procedimiento realiza un cociente entre una ventana corta que promedia el sismograma, y una ventana larga. Se establecerá un umbral mínimo para ese valor, y cuando se supere se localizará el instante que deseamos encontrar. En ese punto se observará un cambio brusco en la ventana STA, que detecta los cambios rápidos como el frente de subida de la señal sísmica, y un nivel de ruido en la LTA razonablemente bajo, de manera que el cociente STA/LTA crezca. Esto puede observarse perfectamente en la figura 4. Es imperativo adecuar los tamaños de las ventanas a los eventos que queremos localizar. Por ejemplo, una ventana STA demasiado larga no encontrará cambios muy bruscos, y una muy corta no eliminaría los ruidos impulsivos que puedan perjudicar la detección. Para más información sobre el método revisar [5].

2) *Algoritmo AMPA*: Este algoritmo, planteado por Álvarez *et. al.*, en [6], está pensado para terremotos reales, y se basa en bancos de filtros adaptados a la forma de onda de una onda sísmica (subida abrupta y decaimiento suave). Es mucho más potente que el STA/LTA para este tipo de señales, y más complejo. Utilizaremos este método únicamente para comprobar la mejora de las detecciones y contrastar con el STA/LTA.

Para aplicar correctamente el AMPA es necesario que adaptemos el filtrado en frecuencia que realiza a la banda que corresponda a cada señal que analizamos. Trabajaremos con tres tipos: los terremotos artificiales, generados en Matlab, y utilizados para poder obtener una medida numérica de la mejora de la detección; los disparos reales medidos en los alrededores del volcán Etna, que se utilizan para realizar la tomografía inversa del mismo; y terremotos reales, con la misma estructura que los anteriores.

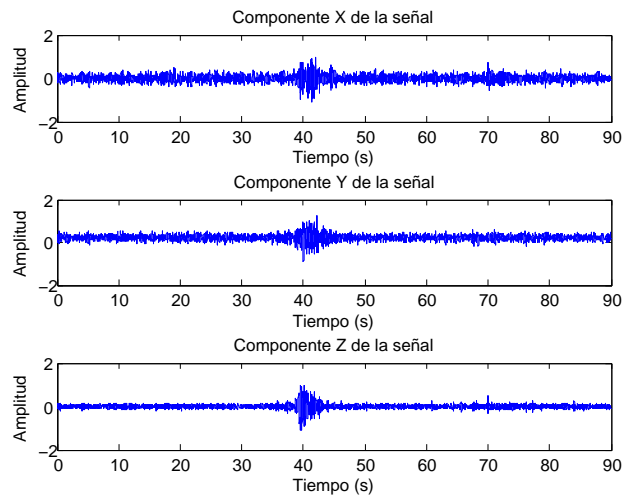


Figura 5. Componentes espaciales E, N y Z de un disparo.

IV. RESULTADOS

A. Forma de los disparos, filtrado y proyección

La forma de los disparos está conformada únicamente por ondas de superficie, al contrario que los terremotos. Estos últimos tienen su origen dentro de la tierra, y llegan ondas de cuerpo al medidor desde el hipocentro, además de las ondas de superficie del epicentro. Es por eso, que los disparos están menos distorsionados, en general, que los terremotos. Las medidas de los disparos que tenemos registradas de los sensores alrededor del volcán Etna se dividen en 3 componentes espaciales (ver figura 5). Las componentes son la Este-Oeste (E), Norte-Sur (N) y vertical (Z), que hacemos corresponder con X, Y y Z respectivamente. Necesitaremos trabajar coherentemente con las tres componentes espaciales para calcular el grado de polarización de cada una de las muestras y aplicar el filtrado.

Trabajaremos en Matlab con conjuntos de disparos para polarizar y detectar antes y después de la polarización. En general, se realizarán las operaciones sobre la componente Z de la medición, ya que es la que más información aporta (ver figura 5).

La detección la realizaremos sobre las proyecciones en el semieje mayor de la polarización elíptica, ya que se presentan, por lo general, más claras que sobre las señales filtradas mediante coeficiente de grado de polarización, aunque la forma de onda cambia. Esto puede suponer un inconveniente para algunas aplicaciones, pero para detección mediante STA/LTA funciona mucho mejor. Para calcular la proyección, simplemente realizamos el producto escalar de la señal original con el semieje mayor de la polarización elíptica que se ha calculado desfasando el sistema coordenado de la onda para adaptarlo a los ejes con los que nosotros estamos trabajando. Al realizar la detección (ver figura 7) sobre los disparos del 320 al 330, comprobamos que el algoritmo STA/LTA mejora notablemente su precisión.

Sin embargo, las señales filtradas mediante grado de polarización tienen una forma de onda más fiel a la señal original y son más sintonizables mediante los parámetros μ y longitud de la ventana. Para realizar un filtrado correcto definimos una

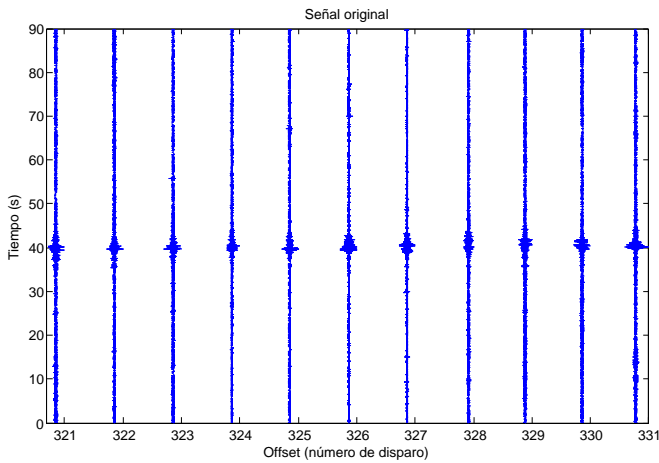


Figura 6. Conjunto de disparos de componente Z

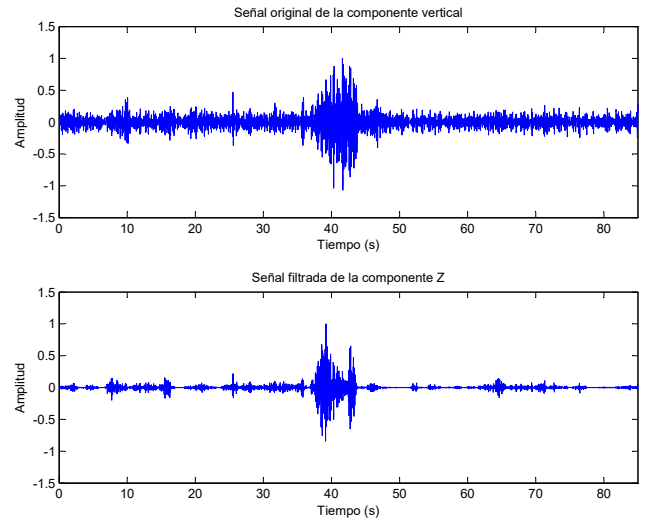


Figura 8. Señal filtrada mediante grado de polarización.

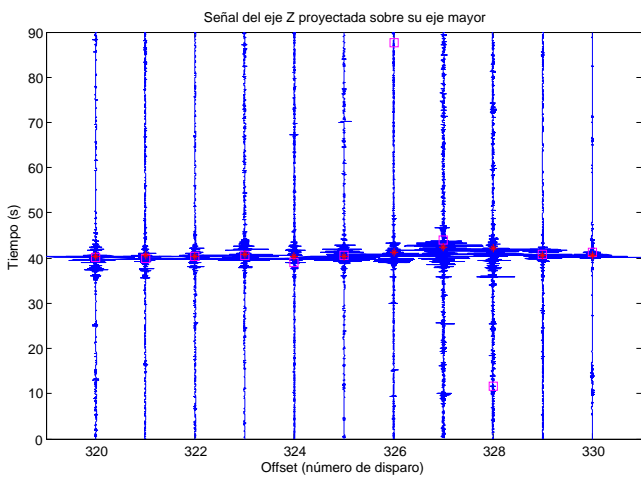


Figura 7. Detección de los disparos. Tiempo detectado después de polarización con asterisco rojo, y antes con cuadrado rosa.

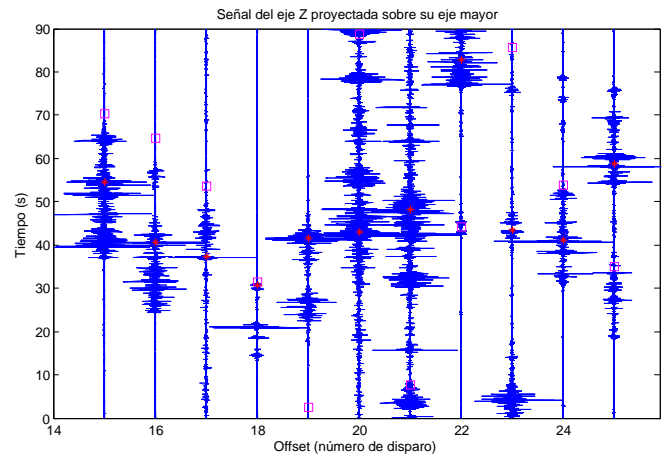


Figura 9. Terremotos reales proyectados sobre la dirección del semieje mayor.

longitud de ventana de 20 muestras y un μ (exponente del grado de polarización) de 2. Al tener una ventana pequeña, la señal es menos suave.

La filtración mediante grado de polarización realiza la señal que queremos detectar (ver figura 8). Para cada tipo de disparo, según si tiene más o menos nivel de ruido, es conveniente jugar con los parámetros de polarización, de forma que el filtrado sea más o menos agresivo según las necesidades y se suavicen los picos.

Para poder comparar el conjunto de señales originales y proyectados con los filtrados mediante el grado de polarización $c(t)$, se muestra el filtrado de una señal en la figura 8. Al eliminarse ruidos coherentes a las tres componentes espaciales se recupera la forma de onda de un disparo real (subida abrupta y decaimiento suave).

Este método es aplicable también a terremotos reales. Las medidas de las que disponemos siguen la misma estructura que la de los disparos, con lo cual es posible importarlas en Matlab y realizar todo en análisis de polarización. En la figura 9 se observa un conjunto de terremotos detectados antes y después de realizar la proyección sobre el semieje mayor

de la polarización elíptica. En este caso, como no son ondas sísmicas controladas, no se puede realizar una comprobación objetiva de la mejora de la detección tras aplicar la proyección.

B. Terremotos artificiales. Medidas cuantitativas.

Para ofrecer una medida cuantitativa objetiva de la mejora de la detección, hemos generado ondas sísmicas artificiales en Matlab (subida abrupta y decaimiento suave), con un tiempo de inicio conocido (ver figura 10). Dichas medidas quedan registradas como desviaciones temporales respecto del tiempo de inicio real en la Tabla I. La proyección mejora enormemente la precisión de la detección mediante STA/LTA y AMPA, siendo más notable el cambio en la primera, ya que STA/LTA es menos potente que AMPA. La señal filtrada mediante grado de polarización mejora la detección, aunque en menor medida que las proyecciones. Además, el AMPA funciona peor al realizar el filtrado que al proyectar la onda, ya que, si los parámetros del grado de polarización están configurados para evitar cambios bruscos, pueden suavizar la subida abrupta de la llegada de la onda y cambiar la forma de onda. Por lo tanto, concluimos que AMPA no es adecuado si

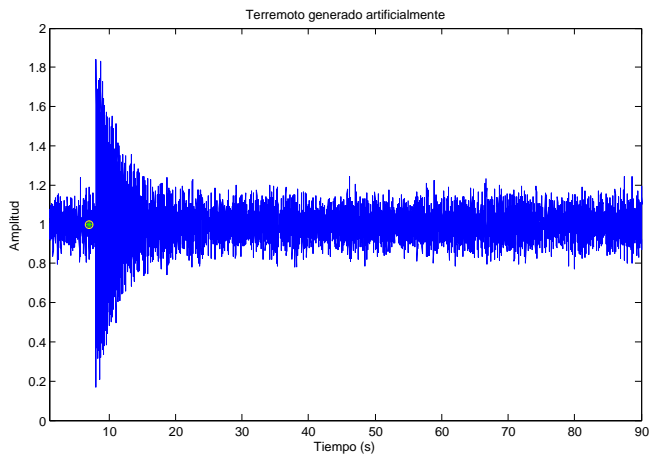


Figura 10. Terremoto artificial generado en Matlab con fondo de ruido de varianza 0.5.

Señal	Desv. temporal STA/LTA	Desv. temporal AMPA
Original	4.16 s	0.9 s
Proyectada	0.43 s	0.38 s
Filtrada	1.4 s	2.53 s

Tabla I

DESVIACIONES TEMPORALES DE LOS INSTANTES DETECTADOS PARA UN FONDO DE RUIDO DE VARIANZA 0.5

se realizan cambios significativos en la forma de onda, y que STA/LTA aplicado tras el algoritmo de análisis de polarización es equiparable en potencia a AMPA.

V. CONCLUSIONES

A lo largo de esta disertación hemos realizado un estudio sobre detección de señales sísmicas artificiales generadas mediante Matlab, disparos superficiales y terremotos reales. Para realizar la detección se ha utilizado el algoritmo STA/LTA ampliamente utilizado en la detección de sismos.

En la mayoría de los casos prácticos las medidas tomadas mediante un sismógrafo introducen réplicas por efecto multipath, ruidos cercanos a los sensores, etcétera. Un algoritmo como el STA/LTA presenta limitaciones en cuanto al ruido añadido a la señal. Sin embargo, si realizamos la polarización de la señal y la proyectamos sobre su eje principal, los distintos ruidos no polarizados se reducen, ampliando mucho el rango de SNR que tolera el STA/LTA.

Para terremotos generados artificialmente en Matlab, como el nivel de ruido es bajo, la limpieza mediante polarización permite observar muy claramente el sismo original, y nos permite concluir que el algoritmo STA/LTA funciona correctamente, comparándolo para ello con el AMPA, y que mejora su funcionamiento considerablemente con respecto a la detección de la señal no polarizada.

El objeto central del análisis de polarización de este Trabajo de Fin de Grado es la limpieza de disparos y terremotos reales para su mejor detección. A lo largo del análisis mediante Matlab, concluimos que la proyección sobre el eje principal de la elipse es mucho más fácil de detectar para el STA/LTA que la señal filtrada a través del grado de polarización. Esto se debe a que el algoritmo de detección se basa en la diferencia de amplitud del sismo con el fondo de ruido, y la proyección realza considerablemente el temblor.

En resumen, el algoritmo de polarización, y la detección sobre la señal proyectada aumentan considerablemente la precisión en la localización de un sismo dentro de una medida temporal registrada con un sismógrafo, sin necesidad de aplicar otros algoritmos de detección más complejos y potentes como el AMPA.

Como hemos analizado durante el apartado de resultados, según el tipo de ruido o el nivel del mismo, es conveniente sintonizar los parámetros de polarización para eliminarlo. En un futuro, sería interesante depurar aún más el algoritmo de polarización mediante la investigación de una forma de adaptar los parámetros de polarización a cada registro de la manera más eficiente para cada uno de ellos y de forma automática.

REFERENCIAS

- [1] J. M. Ibáñez, "TOMO-ETNA MED-SUV.ISES a seismic experiment at Mt. Etna volcano. An integrated marine and onshore geophysical survey. - MED-SUV- MEDITerranean SUPersite Volcanoes." [Online]. Available: <http://med-suv.eu/spip.php?article91>
- [2] Z. Sears and F. Young, "Física Universitaria." Pearson, 2009, vol. 2, pp. 1134–1139.
- [3] J. Gallart and M. Schimmel, "The use of instantaneous polarization attributes for seismic signal detection and image enhancement," *Geophysical Journal International*, vol. 155, no. 2, pp. 653–668, 2003.
- [4] I. B. Morozov and S. B. Smithson, "Instantaneous polarization attributes and directional filtering," *Geophysics*, vol. 61, no. 3, pp. 872–881, 1996.
- [5] A. Trnkoczy, "Understanding and parameter setting of STA/LTA trigger algorithm," in *New Manual of Seismological Observatory Practice 2 (NMSOP-2)*, bormann, p. (ed.) ed., 2012.
- [6] I. Alvarez, L. Garcia, S. Mota, G. Cortes, C. Benitez, and A. de la Torre, "An Automatic P-Phase Picking Algorithm Based on Adaptive Multiband Processing," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 6, pp. 1488–1492, Nov. 2013.

Ingeniería Telemática

Diseño de un juego con requisitos de tiempo real para redes móviles

Autor: Pablo Peña Martínez; e-mail: pablop@correo.ugr.es

Tutores: Juan José Ramos Muñoz; e-mail: jjramos@ugr.es

Juan Manuel López Soler; e-mail: juanma@ugr.es

Titulación: Ingeniería de Telecomunicación

Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Resumen—La industria de los videojuegos cerró el año 2015 con una facturación de 90.000 millones de dólares. Gran parte de los videojuegos desarrollados actualmente tienen un enfoque multijugador para permitir la interacción entre jugadores. Puesto que en las redes *best effort*, como Internet, los juegos con requisitos en tiempo real no tienen garantizada la tasa de transferencia que requieren y, especialmente en conexiones con baja tasa de bits, como el caso de algunas conexiones en redes móviles, es necesario disponer de protocolos que permitan adaptarse rápidamente a la tasa de bits disponible de forma dinámica para aplicaciones como juegos en red. En el presente proyecto se ha llevado a cabo el diseño, la implementación y el análisis de un videojuego multijugador en línea con soporte para hasta 4 jugadores, conectados de forma simultánea según el modelo cliente-servidor, así como un protocolo que adapte su tasa de transferencia según la tasa de bits disponible. Para su desarrollo se ha hecho uso del motor gráfico Unity 3D y del lenguaje de programación C#. El protocolo y el esquema de adaptación propuesto han sido evaluados mediante experimentación. Como resultado, se ha conseguido reducir el retardo debido a la limitación de ancho de banda, de 20 segundos a 800 ms en el peor caso estudiado.

Palabras clave—C#, multijugador, protocolo adaptativo, Unity 3D, videojuego.

I. INTRODUCCIÓN

La industria de los videojuegos se ha convertido en la primera industria de ocio audiovisual e interactivo de España con un consumo de 996 millones de euros y un volumen de usuarios de 13 millones, un 36% del total de la población. Hoy día los videojuegos pueden ejecutarse en multitud de dispositivos: consolas de sobremesa, consolas portátiles, ordenadores, *smartphones* y *tablets*.

Algunos de los juegos más vendidos cuentan con soporte multijugador en red [1]: FIFA 16 (consolas de sobremesa y PC), Mario & Sonic En Los Juegos Olímpicos Rio 2016 (consolas de sobremesa y consolas portátiles), Grand Theft Auto V (consolas de sobremesa y PC), Tom Clancy's The Division (consolas de sobremesa y PC), Overwatch (consolas de sobremesa y PC) o Uncharted 4 El Desenlace del Ladrón (consola de sobremesa Playstation 4).

La opción multijugador supone una parte fundamental en el desarrollo de un videojuego ya que se espera que los jugadores puedan interactuar entre sí sin que haya problemas que frustren la experiencia de juego. Sin embargo, todos estos juegos se ejecutan sobre Internet. Internet se basa en el protocolo IP (RFC 791), que ofrece un servicio *best-effort*. Esto significa que no puede garantizar que no existan

pérdidas de paquetes o retardo acotado. Finalmente, dado que los jugadores pueden usar plataformas de juegos muy diferentes, se dan distintos tipos de acceso con distintos anchos de banda.

II. REVISIÓN DEL ESTADO DEL ARTE

A. Motores de juego y librerías especializadas

Actualmente, los juegos profesionales suelen desarrollarse sobre plataformas de desarrollo que facilitan la interacción entre artistas, programadores, músicos, etc. Estas plataformas aceleran el proceso de elaboración del juego, y suelen integrar el desarrollo en un entorno, y el motor que gobernará el juego en tiempo de ejecución. Entre los motores de juego destacan Unity 3D, Unreal Engine y CryEngine y como librería especializada LibGDX, un *framework* de desarrollo que da al programador acceso a sistemas de archivos, dispositivos de entrada y audio y acceso a las interfaces OpenGL.

B. Juegos similares en el mercado

Wipeout consiste en una serie de circuitos donde hasta 8 jugadores en línea pueden competir entre sí por llegar a la primera posición. Mario Kart también permite la competición de hasta 8 jugadores. ModNation Racers permite competir de manera offline con hasta 4 jugadores o de manera online con hasta 12 jugadores.

C. Técnicas de ahorro de ancho de banda

Los videojuegos online tienen que cumplir una premisa fundamental: todos los jugadores deben obtener una experiencia de juego lo más cercana posible a si se jugase de manera local. Si esta premisa no se cumpliera por saturación del ancho de banda disponible, se producirían situaciones que no son deseables en el transcurso de la partida como. Para mitigar estos problemas, en la literatura relacionada se han propuesto diversas técnicas que ayudan a optimizar la experiencia de juego.

Tunelado, compresión y multiplexado: Esta técnica consiste en un ahorro del ancho de banda mediante un agente local que comprime las cabeceras y utiliza un túnel para enviar varios paquetes dentro de uno multiplexado. Este algoritmo se usa cuando los usuarios comparten la misma red [2]. La idea surgió de la compresión del tráfico VoIP (*Voice over IP*) - RFC 6405- mediante TCRTTP (*Tunneling Multiplexed Compressed RTP*) -RFC 4170- que permite la compresión y multiplexación de flujos RTP (*Real-time Transport Protocol*) -RFC 1889-. Para ello se comprimen las cabeceras mediante ECRTP (*Enhanced Compressed RTP*) -

RFC 3545-, a continuación se multiplexan varios paquetes en uno mediante PPPMux (Point-to-Point Protocol Multiplexing) –RFC 3153- para finalmente ser enviado extremo a extremo mediante un túnel L2TP (Layer 2 Tunneling Protocol) –RFC 2661-.

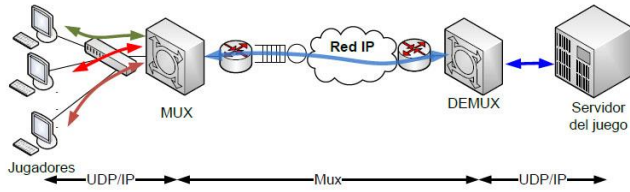


Fig. 1. Tunelado, compresión y multiplexación

Kahawai: Esta tecnología ha sido implementada por investigadores de la Universidad de Duke y de Microsoft Research [3]. Está orientada a reducir el ancho de banda consumido en juegos de *streaming* hasta en una sexta parte usando, para ello, a los clientes en el proceso de renderizado de los contenidos en lugar de dejar todo el proceso al servidor. Es decir, los recursos se comparten entre el servidor remoto y la GPU del cliente. Ello permite que Kahawai (*stream* en hawaiano) pueda usarse en dispositivos que tengan una conexión a Internet más limitada, proporcionando mayor calidad al haber liberado parte del ancho de banda. En la figura 2, el servidor renderiza las versiones altas (“*Hi Detail*”) y bajas (“*Lo Detail*”) de cada *frame* y calcula la diferencia visual en un *delta frame* (“*Delta*”) para codificar un video, usando H264, de todos los *delta frames*. El cliente decodifica el video y cada *frame* (“*Patch*”) lo añade a la versión baja (“*Lo Detail*”) generada por la GPU del cliente obteniendo así la versión alta (“*Hi Detail*”). Por último, el cliente informa al servidor de que ha sido capaz de generar esta versión.

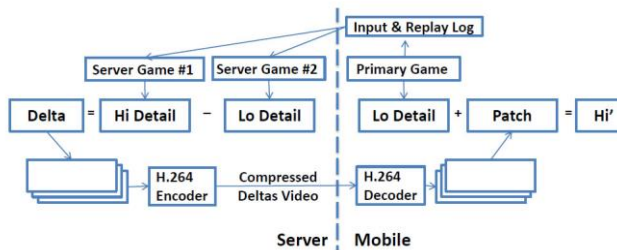


Fig. 2. Técnica Kahawai

III. TEMPORIZACIÓN Y COSTES

Las fases en las que se divide el proyecto y su coste (Tabla I) se describen a continuación: **establecer los objetivos del proyecto, análisis de juegos de similares características, estimar los costes asociados a la ejecución del proyecto, diseñar el videojuego, implementar el videojuego, realizar las pruebas y analizar los resultados, elaborar la documentación del proyecto.**

IV. DISEÑO

A continuación se establece el diseño que debe tener el videojuego y el protocolo adaptativo.

A. Descripción general del juego

El videojuego diseñado es un simulador de naves que compiten entre sí por llegar a la meta y obtener el máximo número de puntos. La puntuación aumentará cada vez que el

disparo realizado por una nave alcance a otra y también aumentará en función del orden de llegada de cada nave a la línea de meta. En el transcurso de la partida irán apareciendo cajas que permitirán al jugador tener algún tipo de bonificación: las cajas azules aumentarán la velocidad de disparo durante cuatro segundos y las cajas naranjas aumentarán la velocidad de movimiento durante dos segundos. El videojuego permite que hasta 4 jugadores jueguen de forma simultánea y compitan entre sí por un circuito, en un juego de carreras de naves. Para ello se hace uso de un servidor que coordinará las funciones básicas del juego y un cliente por cada jugador que interpretará los mensajes que envíe el servidor.

TABLA I
ESTIMACIÓN DE COSTES

Recurso	Cantidad	Coste unitario(€)
Recursos humanos		
	horas	€/hora
Juan José Ramos Muñoz	60	30
Pablo Peña Martínez	318	30
Recursos materiales		
Ordenador con Windows 10	1	600
Ordenadores para pruebas	3	500
Router	1	35
Cables Ethernet	4	2
Herramientas ofimáticas	1	70
Entorno Unity 3D	1	0
Network Emulator Client	1	0
Analizador Wireshark	1	0

COSTE TOTAL: 13553

B. Características y requisitos del juego

Las características del juego son las siguientes:

- Todos los eventos deben quedar validados por el servidor: cuando un jugador cambie su posición, además de moverse de forma local, éste se lo comunicará al servidor quien variará la posición de ese cliente en su propio escenario y le responderá con la ubicación en la que debe estar para que, en caso de ser diferente a la que tiene el cliente, sea corregida. De igual forma, cuando un usuario realice un disparo, éste se lo comunicará al servidor quien se encargará de comunicar al resto de usuarios que esa nave ha disparado. Por último, los disparos que acierten a otras naves sólo puntuarán si en el servidor ha ocurrido esa colisión.
- Será necesario un equipo que haga las funciones de servidor con el que se comunicaran los equipos de los clientes. Se hará uso de un servidor centralizado para que los clientes estén actualizados de todo lo que ocurre en el juego.
- Dado que servidor y cliente son entidades diferentes, cada uno tendrá su propia ventana de inicio: el servidor dispondrá de la opción de iniciarse quedando a la espera

de que lleguen los jugadores, mientras que el cliente tendrá la opción de ver instrucciones básicas del juego, introducir un nombre de usuario e iniciar la partida.

- Las bonificaciones que van apareciendo por el circuito serán acumulables entre sí siempre que sean del mismo tipo, en caso contrario se perderá la bonificación actual y se activará la que se acaba de coger.
- Puesto que al finalizar el circuito las naves que han llegado a la línea de meta no se pueden mover, hay que evitar que los disparos de otras naves que les alcancen sumen puntuación.
- Cuando todos los jugadores hayan llegado a la línea de meta se dará opción a los jugadores de salir de la partida.

C. Esquema de ejecución del videojuego

En la figura 3 se muestra la ejecución del juego, tanto si es se trata de un cliente como del servidor.

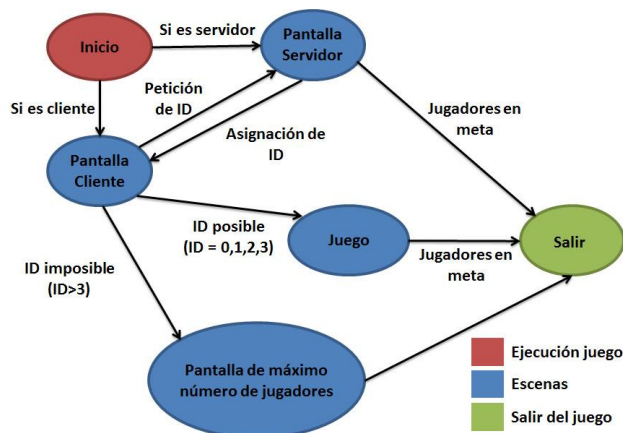


Fig. 3. Esquema de ejecución del juego

Inicio: En el entorno de desarrollo de Unity 3D habrá una variable booleana para establecer qué versión exportada del videojuego actuará como servidor y cuál actuará como cliente. En esta pantalla de inicio se leerá esa variable para mostrar una pantalla u otra.

Pantalla servidor: En esta pantalla está la opción de lanzar el servidor quien se encargará, en primera instancia, de asignar el identificador de usuario a cada cliente nuevo.

Pantalla cliente: En la pantalla destinada al cliente habrá un menú para poder leer las instrucciones básicas del juego, indicar su nombre de usuario e iniciar el juego.

Juego: A esta pantalla irán accediendo los jugadores con un identificador válido para poder jugar. Puesto que el número máximo de jugadores es cuatro, los identificadores permitidos para poder jugar serán el 0, 1, 2 y 3. Cuando se alcance el número máximo de jugadores el juego comenzará.

Pantalla de máximo número de jugadores: Esta pantalla se mostrará cuando, habiéndose alcanzado el número máximo de jugadores para comenzar la partida, entre un nuevo jugador con la intención de unirse a ella.

Salir: La opción de salir del juego se mostrará cuando todos los jugadores hayan llegado a la línea de meta (tanto en el servidor como en los clientes) o cuando un jugador quiera unirse a la partida estando ya el cupo completo.

TABLA II
TIPOS Y SUBTIPOS DE MENSAJES

Tipo	Subtipo
solicitarID	noSubtipo
asignarID	idPosible, idImposible
posicionJugadores	noSubtipo, masPosicion, menosPosicion
rotacionJugadores	noSubtipo
saltar	noSubtipo
mover	noSubtipo
disparar	noSubtipo
puntuacion	noSubtipo, puntuacionFinal, puntuacionPerdida
boostVelocidad	noSubtipo
boostDisparo	noSubtipo
circuito	circuitoIniciado, circuitoAcabado

D. Comunicación entre servidor y cliente

En el diseño del videojuego, el servidor controlará todos los eventos que ocurran en el desarrollo del juego (movimiento de las naves, disparos, incremento de la puntuación,...) e informará de ello a los clientes.

La función del servidor a implementar es la de un **servidor autorizado**, es decir, todos los mensajes generados por los clientes deben ser enviados al servidor y será éste quien se encargue de validarlos y de enviar la información correspondiente al resto de clientes. Los clientes, por su parte, enviarán al servidor los eventos que afecten al movimiento y a la generación de disparos, además de interpretar los mensajes recibidos por parte éste.

La comunicación entre servidor y cliente se harán mediante UDP (*User Datagram Protocol*). Para tal fin se diseñará un protocolo capaz de transmitir cualquier tipo de mensaje y con las opciones que requiera cada uno de ellos.

Para el desarrollo del proyecto se ha tenido en cuenta que los mensajes más importantes para obtener una buena experiencia de juego (mensaje de posición y mensaje de rotación) sean periódicos, evitando así que la pérdida de un paquete afecte a la totalidad de la partida.

E. Formato de los mensajes

La estructura del mensaje a enviar se detalla en la figura 4. Se ha considerado que el mensaje periódico fundamental para que el juego se desarrolle con normalidad es el mensaje de posición de los jugadores.

Durante la ejecución del videojuego, tanto servidor como cliente intercambiarán mensajes de distintos tipos (Tabla II).

Por parte del servidor: asignación de identificador, posición de los jugadores, rotación de los jugadores, generación de disparo, envío de la puntuación, bonificación de velocidad y bonificación de disparo.

Por parte del cliente: petición de identificador, saltar, mover, disparar, reenvío de la puntuación, inicio/fin del circuito y cambio en la periodicidad de los mensajes.

F. Descripción del algoritmo adaptativo

El esquema de adaptación (figura 5) que se ha diseñado en este proyecto consiste en modificar el periodo de envío tp_i de la información del servidor al cliente i -ésimo. De esta manera, cuando el cliente i -ésimo detecte que varía el ancho

de banda disponible, también variará la tasa de envío a ese cliente. Para ello se tendrán en cuenta dos factores:

- El primero es identificar si el periodo de envío es mayor que la tasa de bits disponible o no. Para ello se hará un estudio del tiempo transcurrido entre los mensajes recibidos y los esperados. En caso de que esta diferencia de tiempo sea positiva se solicitarán menos mensajes, en caso contrario se solicitarán más.
- El segundo es limitar esa diferencia de tiempo a un valor umbral para el que, al sobrepasarse, se envíe la solicitud de cambiar la periodicidad de los mensajes.

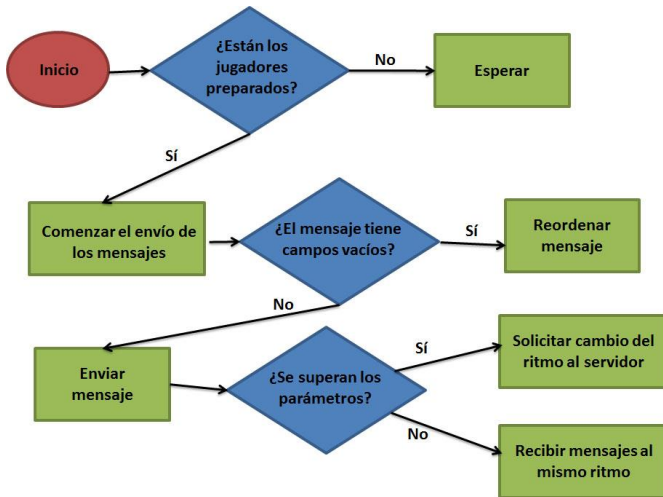


Fig. 5. Esquema de funcionamiento del protocolo

G. Estimación de cambio en la tasa de bits disponible

En cada *frame*, cada cliente calcula la secuencia del mensaje que espera recibir, s_e , siguiendo la siguiente fórmula:

$$s_e = \frac{t_a - t_0}{tp_i} + s_a \quad (1)$$

- t_a es el instante del *frame* actual.
- t_0 es el instante en el que comienza a ejecutarse el juego.
- tp_i es el periodo de envío del paquete del servidor al cliente i .
- s_a es la secuencia anterior al cambio de tp_i (inicialmente tiene valor 0).

Posteriormente se calcula el retardo entre la secuencia recibida y la esperada. Este retardo puede ser positivo o negativo, en el primer caso se detecta un aumento de la tasa de bits disponible y en el segundo, una disminución de la tasa de bits disponible. Si este retardo supera un determinado valor umbral, γ , tanto por exceso como por defecto, para un determinado número de secuencias α en un periodo de tiempo β , el cliente i solicitará al servidor una variación en el periodo de envío del paquete tp_i .

V. IMPLEMENTACIÓN

A. Implementación de la interfaz

Unity 3D facilita la creación de las distintas escenas entre las que se irá navegando:

Menú: En esta escena se indica si la aplicación ejecutada es el servidor o el cliente.

Menú Servidor: Escena destinada al servidor y cuya única función es iniciar el servidor.

Menú Cliente: Escena que se ejecuta en cada uno de los clientes donde se puede consultar las instrucciones del juego, elegir un nombre de usuario y ejecutar el juego.

Instrucciones: Escena con los controles y el funcionamiento del juego.

Escena: Se trata del circuito donde se desarrolla la competición.

Salir sin ID: Escena que se muestra en caso de que un nuevo usuario quiera unirse a la partida habiéndose alcanzado el número máximo de jugadores.

B. Implementación del protocolo adaptativo

Para poder enviar el tipo y el subtipo dentro del mismo byte, se hace uso de 4 bits para identificar al tipo y otros 4 bits para el subtipo (declarados como constantes en hexadecimal). El proceso se detalla a continuación:

- **Emisor del mensaje:** Se realiza la operación “OR” tipo|subtipo.
- **Receptor del mensaje:** Se realiza la operación “AND” recibido&F0 para obtener el tipo y recibido&0F para el subtipo.

VI. ESTUDIO DEL TRÁFICO DE RED Y EVALUACIÓN DEL PROTOCOLO ADAPTATIVO

Para evaluar el diseño del juego y del protocolo adaptativo se hará uso del programa *Network Emulator Client* [4] para limitar la red y del capturador de paquetes *Wireshark* [5] para analizar el tráfico generado.

A. Análisis para un jugador

A continuación se muestran los resultados obtenidos cuando existe un único jugador. Las gráficas representan el número de secuencia frente al tiempo, tanto para los mensajes recibidos (rojo) como los esperados (verde).

- Sin limitación de ancho de banda:

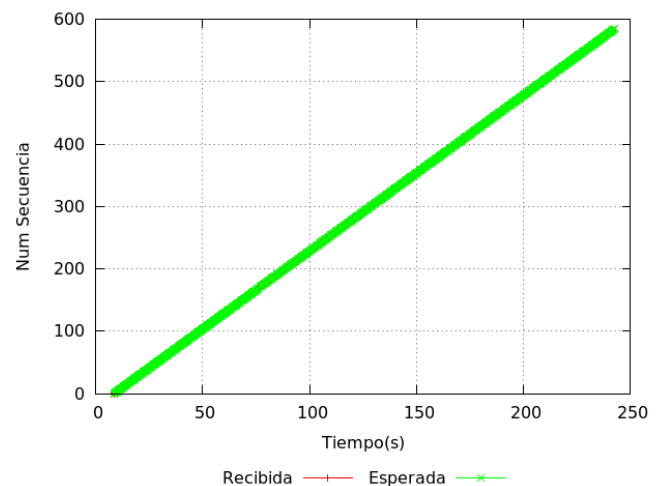


Fig. 6. Sin limitación

- Con limitación a 2.2kbps:

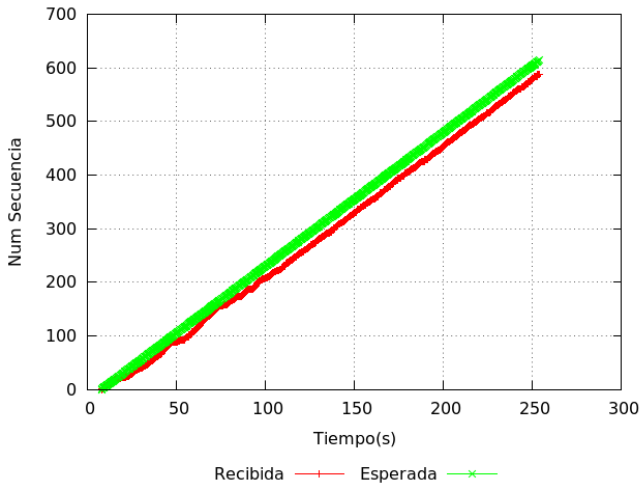


Fig. 7. Con 2.2kbps de limitación

- Con limitación a 1.6kbps:

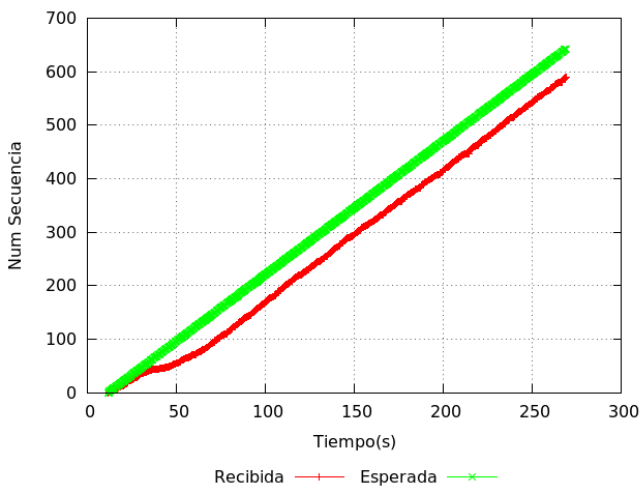


Fig. 8. Con 1.6kbps de limitación

A medida que se limita el ancho de banda el retardo se hace cada vez más notorio.

B. Análisis para un jugador con protocolo adaptativo

Se analizan los resultados obtenidos con el protocolo adaptativo cuando hay un jugador activo.

- Sin limitación de ancho de banda:

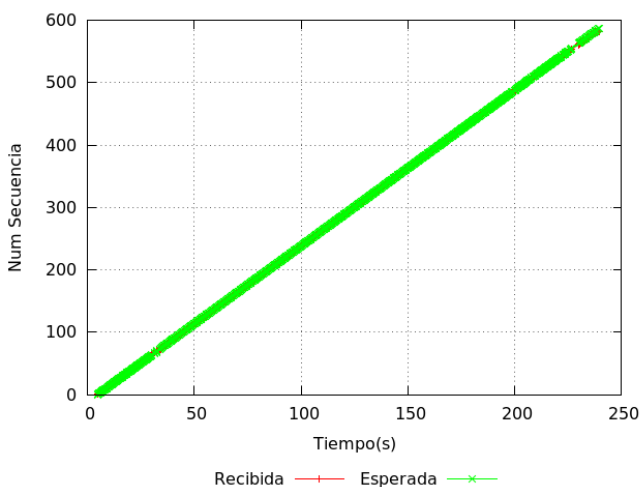


Fig. 9. Sin limitación con protocolo adaptativo

- Con limitación a 2.2kbps:

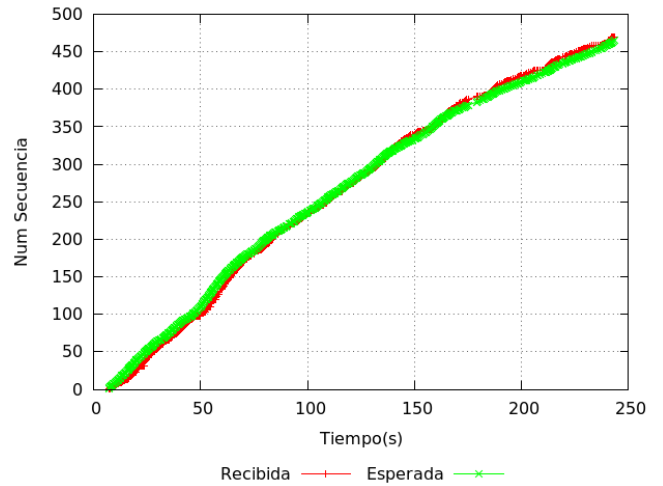


Fig. 10. Con 2.2kbps de limitación con protocolo adaptativo

- Con limitación a 1.6kbps:

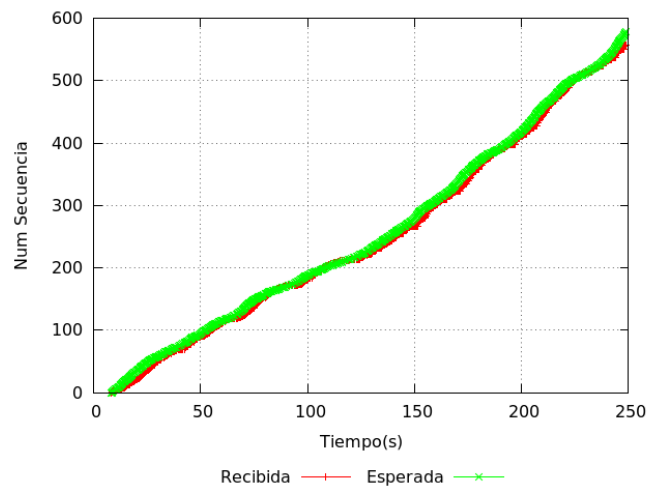


Fig. 11. Con 1.6kbps de limitación con protocolo adaptativo

Se puede apreciar que, a medida que se ejecuta el juego, el protocolo va adaptando la frecuencia de los mensajes para que el retardo obtenido sea el mínimo.

C. Evaluación del retardo medio

En la figura 12 se representa el retardo medio asociado a los mensajes de posición para el caso de un jugador tanto sin técnica (rojo) como con ella (verde).

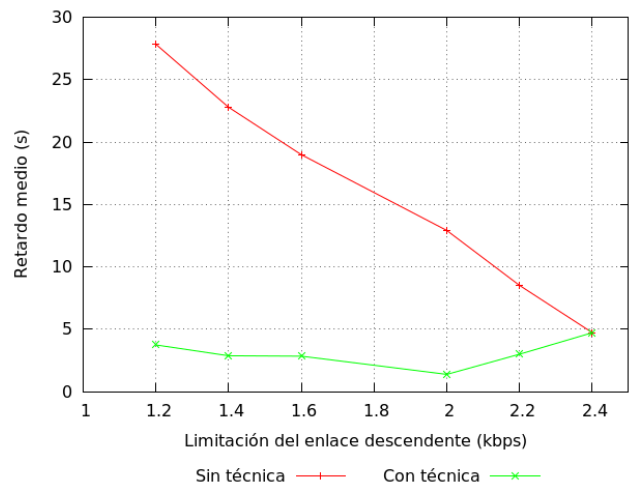


Fig. 12. Retardo medio para un jugador

Como puede observarse, al aplicar el algoritmo adaptativo, siempre se reduce el retardo, funcionando mejor cuando hay más limitación de ancho de banda. Concretamente, se reduce de 27.83 a 3.73 (7.76 veces menos) en el caso de la limitación de 1.2kbps.

D. Evaluación de las bonificaciones obtenidas

En la figura 13 se evalúa el resultado que experimentaría un usuario de forma numérica. Cada nave está programada para ir recogiendo todas cajas de bonificación.

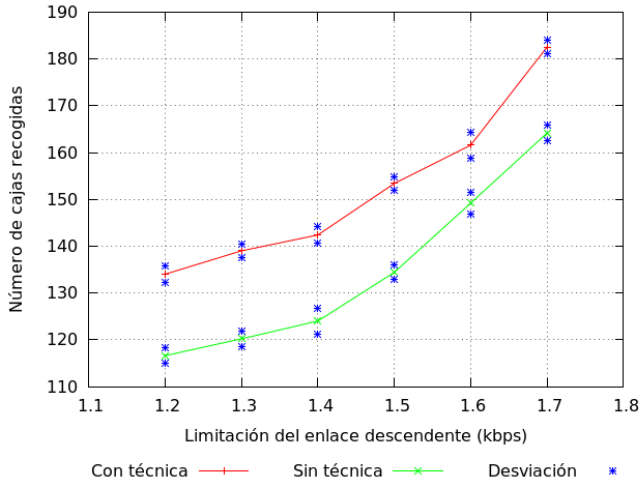


Fig. 13. Bonificaciones obtenidas para un jugador

VII. CONCLUSIONES

A. Conclusiones generales

En el proyecto se ha desarrollado un juego multijugador en línea con hasta 4 jugadores jugando de forma simultánea y un

protocolo adaptativo que mitiga los efectos del retardo asociado a la limitación del ancho de banda, mejorando la calidad de experiencia de los usuarios.

B. Trabajo futuro y mejoras

Pese a que el juego cumple las funciones establecidas, se enumeran a continuación una serie de mejoras que podrían añadirse:

- Añadir más circuitos permitiendo a los jugadores realizar una clasificación global tras haberlos recorrido todos y no siempre dar vueltas sobre el mismo circuito.
- Añadir objetos o enemigos (ajenos a los jugadores) que pretendan colisionar con nosotros para hacer que perdamos puntuación o ralentizarnos.
- Adaptar los controles para poder ejecutarse en *smartphones* y *tablets* (pantalla táctil, uso del giroscopio,...).

El protocolo adaptativo puede mejorarse en los siguientes puntos:

- Evaluar el retardo en los mensajes de información secundaria y decidir también para ellos cuándo es necesario aumentar o disminuir la frecuencia de los mismos, ya que en esta versión los mensajes de posición son los más importantes.
- Calcular la distancia entre las naves y, si es superior a un valor, no enviar mensajes referentes a ese jugador para ahorrar ancho de banda.
- Diseñar un mecanismo de control que detecte la pérdida de paquetes y permita el reenvío de los que se han perdido.

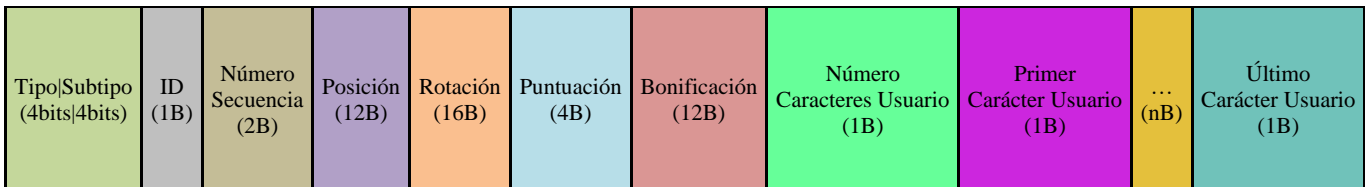


Fig. 4. Estructura general del mensaje

Mobile Gaming Using GPU Offload: <http://research.microsoft.com/en-us/people/cuervo/mobi093f-cuervoA.pdf>

AGRADECIMIENTOS

Agradezco a mi familia el apoyo y ánimo recibidos durante todos estos años que ha durado esta etapa de mi vida. Gracias también a los profesores Juan José Ramos Muñoz y Juan Manuel López Soler por la dedicación y aporte de conocimientos para que la realización de este proyecto haya sido posible.

REFERENCIAS

- [1] Asociación Española de Videojuegos. *Ranking General*, abril 2016: <http://www.aevi.org.es/la-industria-del-videojuego/los-videojuegos-mas-vendidos/>
- [2] José Mª Saldaña, Julián Fernández Navajas, José Ruiz Mas, José I. Aznar, Eduardo Viruete Navarro, Luis Casadesus. *Ahorro de Ancho de Banda en Juegos Online Mediante el uso de Técnicas de Tunelado, Compresión y Multiplexación*: http://diac.unizar.es/~jsaldana/personal/juegos_jitel_2011_in_proc.pdf
- [3] Eduardo Cuervo, Alec Wolman, Landon P. Cox, Kiron Lebeck, Ali Razeen, Stefan Saroiu, Madanlal Musuvathi. *Kahawai: High-Quality*

- [4] Network Emulator Client: <https://blog.mrpol.nl/2010/01/14/network-emulator-toolkit/>
- [5] Web de Wireshark: <https://www.wireshark.org/>



Pablo Peña Martínez nacido el 22 de Octubre de 1988 en Granada. Ingeniero de Telecomunicación por la Universidad de Granada desde Junio de 2016. Disfrutó de una beca ERASMUS en l'École Polytechnique d'Ingénieurs de l'Université de Nice-Sophia Antipolis (Francia).

Implementación de una aplicación cliente para Chromecast

Autor: Irene Herrera López, e-mail: ireneherlo@gmail.com

Tutor: Jorge Navarro Ortiz, e-mail: jorgenavarro@ugr.es

Titulación: Grado en Ingeniería de Tecnologías de Telecomunicación
Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Resumen—El proceso evolutivo del ser humano y sus etapas han estado marcadas por avances significativos que han tenido lugar en diferentes ámbitos de la sociedad. Durante los últimos años, la vida del ser humano ha estado ligada a la evolución tecnológica, aprovechando sus ventajas y permaneciendo en una continua búsqueda de nuevas funcionalidades innovadoras que incorporar a su vida diaria. Así, se han ido insertando dispositivos tecnológicos, cada vez más inteligentes, en el ámbito profesional. En este Trabajo Fin de Grado se pretende aprovechar esta tendencia para hacer del proceso de proyección de material docente una tarea que proporcione al ponente de una mayor libertad frente a las limitaciones que suponen los sistemas cableados actuales. Para ello, se aunarán las ventajas que presentan los dispositivos de transmisión y reproducción de contenido multimedia, más concretamente, las ventajas de Chromecast, junto a la utilidad de los dispositivos móviles inteligentes o *smartphones*. La conjunción de ambos elementos supondrá una pieza clave para la obtención de una solución *software* que permita al docente o estudiante de la Universidad de Granada proyectar material, con fines educativos, desde su *smartphone* a una pantalla secundaria o panel de proyección de manera inalámbrica, simple y sencilla.

Palabras clave—Android, Chromecast, docencia, Google Cast.

I. INTRODUCCIÓN

ESTE proyecto puede considerarse como un estudio de investigación en el ámbito de la tecnología educativa, por lo que en un primer lugar se procederá con una introducción sobre el uso innovador de las TICs (Tecnologías de la Información y Comunicación) en la docencia universitaria. La enseñanza ha sido, y sigue siendo, un proceso interactivo, por lo que consideraremos la tecnología como una herramienta personal más que como un medio para educar. La tecnología no pretende reemplazar el valor humano, sino servir como una extensión de éste [1]:

A continuación se van a describir algunas de las ventajas más significativas que supone el uso de la tecnología con fines docentes [2].

- Aumentar la atención, la motivación y la participación del alumnado.
- Facilitar la enseñanza, el aprendizaje, la comprensión de los temas y la consecución de objetivos marcados.
- Favorecer la renovación metodológica del docente o ponente.
- Aumentar la satisfacción, la motivación y la autoestima del docente.

II. MOTIVACIÓN

La sociedad se ha acostumbrado a tener acceso permanente y ubicuo a la comunicación e información. Ello ha supuesto un giro en el concepto de percepción de la realidad y en los hábitos de la comunidad y, por supuesto, también en las aulas y la docencia. Y no es sólo la sofisticación tecnológica lo que nos interesa, sino también la revolución creativa que todo ello supone.

Desde las reuniones en las ágoras de la Antigua Grecia hasta los proyectores multimedia inalámbricos, pasando por pizarras convencionales y transparencias de acetato, han sido múltiples vicisitudes las que se han superado. Pero, ¿cuál es el estado actual de los sistemas de proyección de material de apoyo a la docencia? La mayoría se rigen por dos configuraciones, una primera en la que el aula dispone de un proyector y un ordenador conectado a éste, a disposición de estudiantes y profesorado, para que puedan introducir en él sus dispositivos de almacenamiento y memoria externa. El segundo escenario más común depende de la disponibilidad de un ordenador portátil personal, y sus correspondientes adaptadores, por parte del ponente que va a realizar su clase magistral.

A diferencia de otras TICs, los *smartphones* ya están en las manos de alumnos y profesores, ya que son casi universalmente accesibles. Este hecho representa un menor coste que el equipamiento de las aulas con ordenadores y proyectores inalámbricos. Aunque no está ampliamente extendido dentro de los sistemas de proyección de material docente, prodría suponer un aspecto a tener en cuenta gracias, entre otras ventajas, a la movilidad y flexibilidad que estos aportan. Y es por esto por lo que se ha pretendido encontrar una solución que aunara la comodidad de las tecnologías inalámbricas y móviles junto al equilibrio calidad-precio.

III. REQUISITOS

La idea principal de este estudio es desarrollar una aplicación cliente para el dispositivo Chromecast [3], que permita a los usuarios enviar contenido multimedia (con fines docentes) a una pantalla secundaria. Se pretende que ese proceso sea sencillo y transparente para el docente o ponente. Buscando la comodidad de éste, se procurarán reducir las conexiones cableadas para así dotar al usuario de una mayor movilidad. Además, se definirá un sistema de control de sesiones y un gestor de archivos que facilite al usuario la búsqueda y proyección del material deseado. Junto a los

anteriormente definidos requisitos funcionales, pasaremos a describir los requisitos no funcionales. El primero de ellos, y el más restrictivo, es el uso del dispositivo Chromecast. Este pequeño, pero a la vez potente, dispositivo fue presentado en el anteproyecto por el tutor como una condición o cláusula indispensable para la obtención de la solución final. Como motivación adyacente, el coste económico y las diferentes posibilidades que presentan las APIs (*Application Programming Interface*) de Google suponen una gran ventaja frente a otros dispositivos similares presentes en el mercado como pueden ser Apple TV [4], productos de la familia Roku [5] y la más reciente incorporación a este mercado, el MatchStick [6] de Firefox. Otro requisito no funcional será el uso del sistema operativo Android para desarrollar la aplicación móvil, ya que al estar desarrollado por Google, al igual que Chromecast, existe una amplia variedad de recursos y librerías disponibles para estas herramientas.

IV. CHROMECAST

En este apartado vamos a presentar al principal protagonista de este trabajo.

Chromecast es un dispositivo con formato de *stick* o pincho que presenta un puerto HDMI (*High-Definition Multimedia Interface*) a través del cual se conecta a la pantalla secundaria en la que queramos realizar la proyección. Su principal propósito es la transmisión y reproducción de contenido multimedia. Diseñado y distribuido por Google, entre sus características más importantes podemos destacar que soporta el estándar IEEE 802.11b/g/n aunque sólo en la banda de 2,4 GHz, presenta un sólo núcleo o *core* y es necesario conectarlo a una fuente de corriente haciendo uso del puerto USB (*Universal Serial Bus*) que presenta.

Aunque se ha realizado una investigación más amplia sobre el funcionamiento del dispositivo, a manera de resumen, procederemos a listar los protocolos más significativos que toman parte en las comunicaciones durante una sesión con Chromecast.

- El dispositivo Chromecast opera con el protocolo DIAL (*Discovery And Launch*) [7], que habilita al cliente a descubrir al servidor DIAL (Chromecast) en su red local y, así, obtener acceso a los servicios DIAL.
- El protocolo RAMP (*Remote Application Media Protocol*) es propietario de Google y se usa para enviar mensajes de control de contenido multimedia al dispositivo. Estos comandos se envían a través de un *websocket* en formato JSON, siendo un *websocket* una tecnología que proporciona un canal de comunicación bidireccional y full-dúplex sobre un único *socket* TCP.
- Para enviar contenido al dispositivo existen dos alternativas principales. La primera de ellas es enviar la pantalla que estemos visualizando desde nuestro navegador web Google Chrome. Para ello, necesitaremos la extensión *Google Cast* y el Chromecast cargará la página web actual haciendo uso de WebRTC (*Web Real Time Communication*). La otra alternativa, y la que emplearemos en este estudio, es enviar el contenido multimedia desde una aplicación cliente. Al realizar el envío, el dispositivo cargará una página web ligera usando los protocolos, HTML5 (*HyperText Markup Language, versión 5*), JavaScript y CSS (*Cascading Style Sheets*), entre otros.

V. DISEÑO E IMPLEMENTACIÓN

En esta sección centraremos nuestra atención en el diseño e implementación de la aplicación para Android que se ha desarrollado como solución a este proyecto. Esta solución *software* se puede dividir en diferentes bloques, que iremos abordando de manera individual a continuación.

A. Autenticación y autorización

Antes de proceder con la implementación de este bloque, vamos a definir los conceptos de autenticación y autorización. La autenticación no determina qué tareas puede llevar a cabo el usuario o qué datos puede visualizar, ya que sólo identifica y verifica al usuario. Por otro lado, la autorización consiste en dar acceso a una serie de recursos al usuario, previamente autenticado. En nuestro caso, podemos tomar ambos conceptos como un solo bloque, debido a que el usuario que se autentique correctamente tendrá acceso a la aplicación en su totalidad. Para la implementación del presente bloque se ha recurrido a las siguientes herramientas:

- PHP (*Hypertext Preprocessor*) [8]. Lenguaje de programación utilizado para definir los archivos de configuración y las tareas que se van a llevar a cabo en el sistema de autenticación (acceso, alta y baja de usuarios, comprobación de credenciales...)
- HTML (*HyperText Markup Language*). Utilizaremos este lenguaje para crear una simple página web que recoja el formulario para el alta de nuevos usuarios. En este formulario, el usuario podrá definir la combinación única de nombre de usuario y contraseña que lo identificará posteriormente en el sistema.
- Base de datos, MySQL. Gestor de la base de datos en la que almacenaremos toda la información referente a los usuarios. Será consultada para discernir si la autenticación es correcta o no.

En la figura 1 podemos observar el diagrama de estados que seguirá la actividad de Android correspondiente a este bloque. Esta será la primera actividad que se ejecutará al lanzar nuestra aplicación y se encargará de recoger los datos introducidos por el usuario (nombre de usuario y contraseña). También contiene el método que realiza la validación del usuario con los datos dados. Dichos datos serán enviados al sistema de autenticación, siendo la respuesta a la validación un objeto con formato JSON (*JavaScript Object Notation*). Si la autenticación tiene éxito, se invocará la siguiente actividad.

B. Sistema de archivos

El material docente que se pretende proyectar será almacenado en la memoria externa del dispositivo (en la tarjeta de memoria SD *Secure Digital*). Para tener acceso a dicho directorio, haremos uso de la clase *Environment* [9] de Android, que proporciona acceso a distintas variables del entorno y, más concretamente, usaremos el método `getExternalStorageDirectory` de dicha clase. Este método devuelve el directorio raíz del almacenamiento externo del dispositivo móvil (e.g. `/storage/emulated/0/`). A *grosso modo*, primero comprobaremos que esa ruta devuelta exista. En caso afirmativo, crearemos una lista en la que iremos almacenando los distintos directorios a los que se

van accediendo, con el fin de conseguir la ruta completa del archivo final que seleccionemos. Se ha habilitado la opción de navegación hacia atrás, con lo que se eliminará el último directorio añadido de la lista mencionada anteriormente. Además de la ruta, debemos obtener el nombre y la extensión del archivo deseado. La extensión será un parámetro muy importante, clave para discernir el siguiente paso de nuestra aplicación. El dispositivo Chromecast presenta una serie de limitaciones en cuanto a formatos de archivos que soporta se refiere. Es por ello por lo que se ha decidido limitar los formatos soportados en nuestra aplicación a las extensiones de imágenes PNG (*Portable Network Graphics*), GIF (*Graphics Interchange Format*) y JPG/JPEG (*Joint Photographic Experts Group*). Cualquier otro tipo de imágenes lanzará un mensaje de error. También se aceptarán los archivos de tipo PDF (*Portable Document Format*) que, aunque no es un formato soportado directamente por Chromecast, es el principal objetivo de esta solución *software*. En la figura 2 podemos ver de manera más gráfica el proceso descrito anteriormente.

C. Conversión de PDF

Para solventar las limitaciones mencionadas en el apartado anterior de la SDK (*Software Development Kit*) *Google Cast* [10] que rige el comportamiento del dispositivo Chromecast, se ha optado por convertir el archivo de tipo PDF que se desea enviar al Chromecast en imágenes, es decir, obtener una imagen por cada una de las páginas que conformen el archivo PDF.

En la figura 3 se detalla el proceso para realizar dicha conversión. En ella, se muestran dos procesos paralelos para el tratamiento de las imágenes y ello se debe a que, además de visualizar el documento en una pantalla secundaria, queremos que también se muestre en la pantalla de nuestro *smartphone*. Así, utilizaremos un objeto de tipo *WebView* [11] en Android, que se encarga de mostrar páginas web ligeras. A continuación vamos a resumir el proceso llevado a cabo para lograr esta tarea. Primeramente obtendremos un objeto de tipo *PDFFile* a partir de un *buffer* de bytes que conforman el documento PDF seleccionado. Una vez hecho esto, debemos extraer la

primera página del documento. Después, obtendremos una escala a partir del ancho del elemento *WebView* y posteriormente convertiremos la página en una imagen de tipo *bitmap* en función de esa escala obtenida. Esa imagen obtenida debemos guardarla como un *array* o cadena de bytes para codificar dicha cadena en base 64. Así obtenemos una cadena de caracteres que conformarán un objeto de tipo HTML, al que iremos añadiendo consecutivamente el resto de imágenes tratadas siguiendo este mismo proceso. Para proyectar las imágenes con Chromecast podríamos reutilizar el resultado de la codificación en base 64 para enviar cada una de las imágenes, codificadas como una cadena de caracteres, con el método *sendMessage* disponible en la SDK *Google Cast*, para posteriormente decodificarlas usando una aplicación receptora de tipo *Custom Receiver* [12]. Sin embargo, no es recomendable usar dicho método para enviar grandes cantidades de datos, ya que ese canal está indicado para ser usado como canal de control y no para enviar datos. Por esta razón, se ha optado por una alternativa más robusta que consiste en empotrar o implementar un servidor web en nuestra aplicación local, es decir, en el teléfono móvil que realizará las funciones de emisor. Así, "serviremos" las imágenes obtenidas de cada una de las páginas del archivo PDF (convertidas previamente a un formato de imagen compatible) al Chromecast haciendo uso del *Styled Media Receiver* [13] que hemos adaptado para que muestre el logotipo y nombre de esta aplicación durante los tiempos de espera. Como servidor web hemos elegido la herramienta *NanoHTTPD* [14], de código abierto y fácil de configurar.

D. Conexión y envío de imágenes

La API (*Application Programming Interface*) *Media Router* [15] habilita las conexiones entre los dispositivos Android y

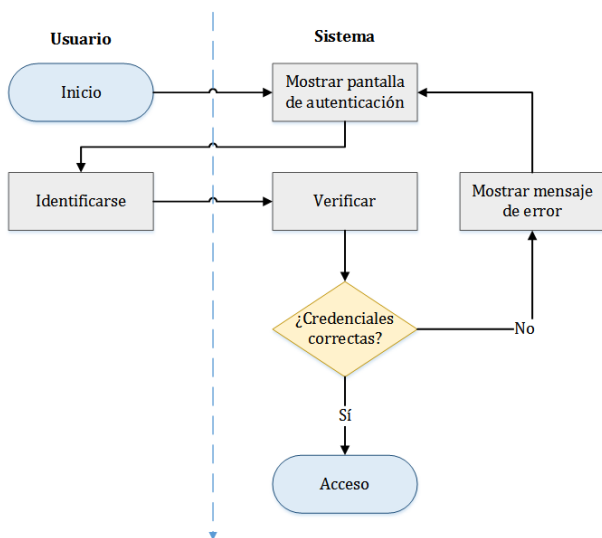


Fig. 1. Diagrama de estados del sistema de autenticación.

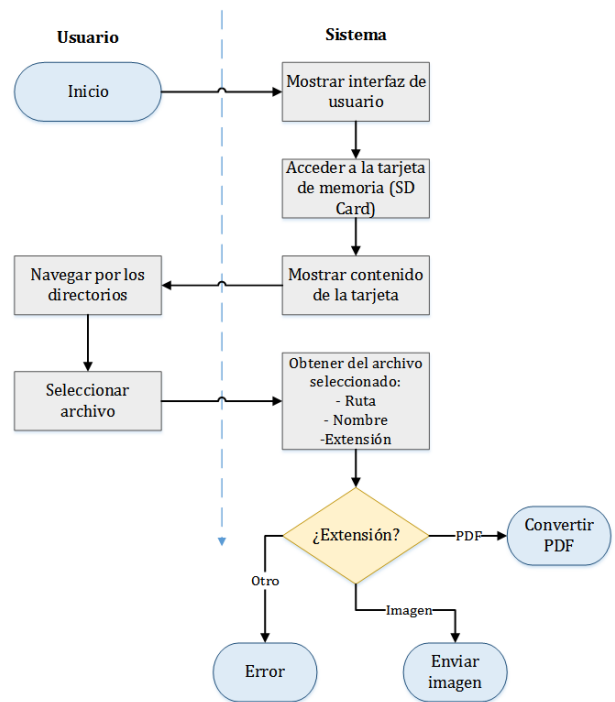


Fig. 2. Diagrama de estados del sistema de archivos.

Chromecast para poder reproducir contenido multimedia. En la figura 4 se visualiza el esquema de funcionamiento de la mencionada API. También debemos añadir a nuestro proyecto la librería de los servicios de *Google Play* para el correcto funcionamiento de la SDK de *Google Cast*.

En la figura 5 se muestran los botones que se han habilitado en la cabecera de la aplicación para facilitar el proceso de reproducción de contenido multimedia.

- El primer botón que se muestra es el denominado botón *Cast*. Informa del estado de la conexión con el dispositivo Chromecast en función de su color.
- El segundo botón sirve para lanzar el proceso de envío de material. Haremos uso de la clase *MediaInfo* de la SDK *Google Cast*, que agrega información sobre el elemento multimedia que pretendemos enviar. Crearemos una instancia de esta clase mediante un constructor, que será usado por la clase *RemoteMediaPlayer* para cargar el contenido multimedia en la aplicación receptora.
- Por último, se pueden apreciar dos botones con forma de flechas cuya función es avanzar y retroceder las páginas que conforman el documento PDF. Cada vez que pulsemos alguno de estos botones, obtendremos la dirección IP del dispositivo móvil (aquella que

apunta a la imagen que queremos visualizar con Chromecast y que ha sido servida por medio del servidor web embebido, junto al puerto en el que éste está disponible). Además, añadiremos al final de esta dirección la hora actual en milisegundos con el método `System.currentTimeMillis()`. Si no incluyésemos esta marca temporal, el receptor siempre estaría almacenando en la caché la misma imagen, ya que la URL (*Uniform Resource Locator*) que estamos pasando es siempre la misma para todas las imágenes. Por ello, hay que forzar al receptor a recargar la imagen añadiendo dicha marca temporal (o *timestamp*) o cualquier otro marcador a la URL, para que el navegador la reconozca como un nuevo recurso. Cabe destacar que en el caso de que se desee enviar una sola imagen permanecerán desactivados estos dos últimos botones ya que no existen páginas que avanzar o retroceder.

E. Otros aspectos de la implementación

A continuación vamos a mencionar otros aspectos referentes a la implementación. Se trata de funcionalidades secundarias que tienen como principal objetivo facilitar el uso de la aplicación por parte del usuario.

- Se ha diseñado un menú desplegable, accesible en el lateral de la aplicación, en el que se muestran las diferentes opciones de navegación. Para ello se ha elegido *Navigation Drawer* [16] de Android.
- Perfil de usuario. Es uno de los apartados disponibles en el menú, en el que se muestra información referente a la red a la que estamos conectados. También se han habilitados dos botones, uno de ellos para cerrar la sesión y el otro permite dar de baja a nuestro usuario. El proceso de dar de baja la cuenta de un usuario significa la eliminación de sus credenciales de nuestra base de datos.
- Sugerencias. Otro apartado disponible en el menú. En éste se muestra información sobre la aplicación y un botón que permite elegir un cliente de correo electrónico para enviar un email con sugerencias o reportes. Por defecto, se rellenará el campo *To* con el correo de la autora de este estudio.

Por último, y para finalizar esta sección, en la figura 6 se muestran algunos ejemplos de la interfaz de usuario de la aplicación implementada.

VI. RESULTADOS Y PRUEBAS

La metodología que se ha seguido para completar la primera versión totalmente funcional de esta aplicación ha sido un modelo de desarrollo en espiral, en el que se ha alcanzado la solución a partir de distintas iteraciones. Cada una de ellas ha ido aportando funcionalidades al sistema diseñado y se han podido aislar los problemas mediante la realización de

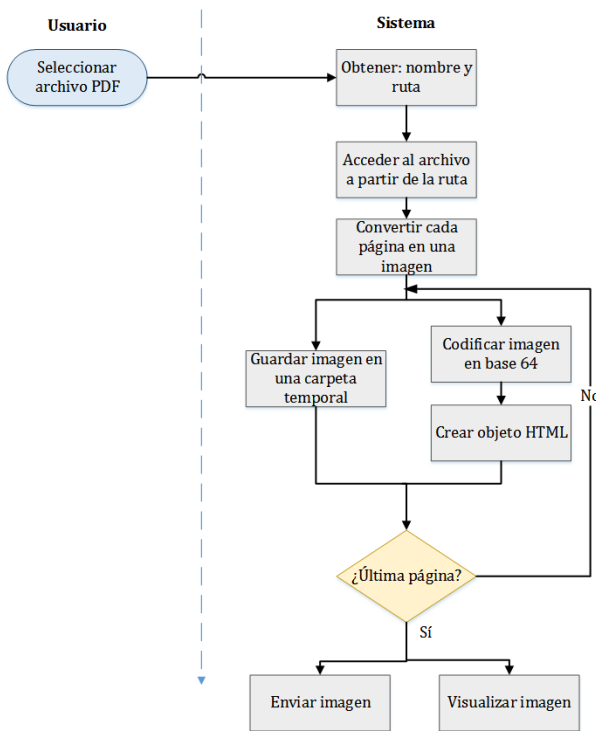


Fig. 3. Diagrama de estados del sistema de conversión de archivos PDF.

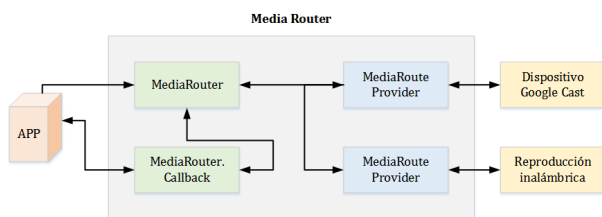


Fig. 4. Diagrama del uso de clases de la API *Media Router*.



Fig. 5. Botones disponibles en la cabecera de la aplicación.

pruebas unitarias. Todo ello ha facilitado la validación de la aplicación completa.

La fase de análisis y control de calidad podría definirse como una de las etapas críticas en el ciclo de vida del desarrollo de un producto *software*.

- Pruebas de rendimiento. En este bloque se han realizado una serie de pruebas en las que se han analizado estadísticas de red, información del sistema y memoria ocupada. Se ha comprobado, entre otros aspectos, que la cantidad de datos que se consumen durante una sesión de proyección es ligeramente superior al tamaño total del archivo transmitido. El excedente se debe a los mensajes de control que se intercambian en la comunicación. En cuanto al porcentaje de uso de la CPU (*Central Processing Unit*), ésta no supone una gran carga para el dispositivo *smartphone*. También se ha analizado la memoria dinámica, donde el término *heap* hace referencia a un espacio de memoria en tiempo de ejecución que se usa para almacenar las instancias de clases, objetos y arrays [17]. Esto resulta relevante para controlar el uso de la pila de un proceso. Durante el proceso de proyección, el tamaño de la pila crece desordenadamente. Si el usuario realiza una pausa entre pases de diapositivas, podemos percibir como el tamaño de la pila se estabiliza.

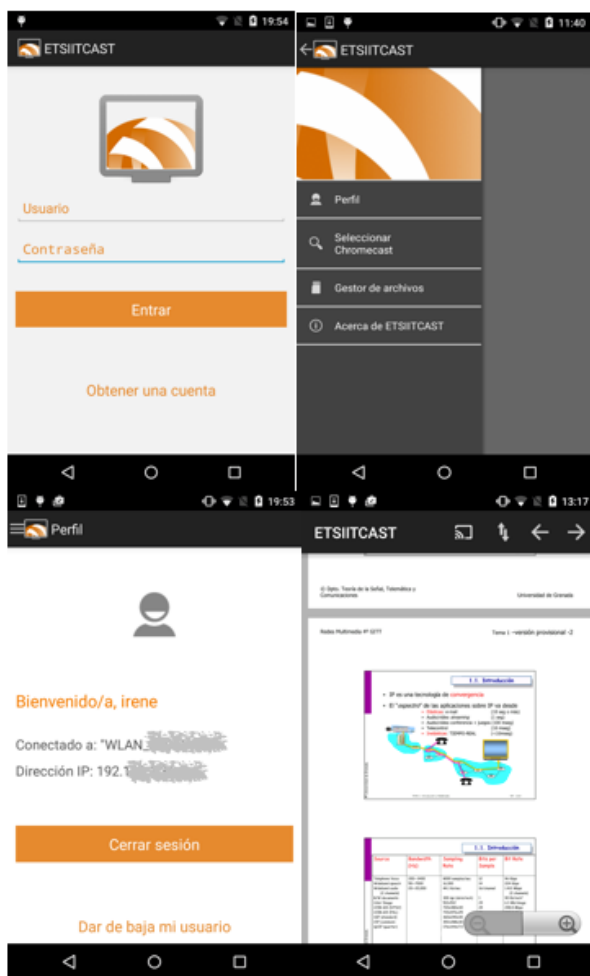


Fig. 6. Capturas de pantalla de la interfaz de usuario de diferentes apartados de la aplicación.

- Pruebas de compatibilidad. Se ha comprobado el correcto funcionamiento de la aplicación bajo diferentes configuraciones de *hardware* y diferentes versiones del sistema operativo Android. Se ha establecido como versión mínima soportada de Android la versión 4.0 (*Ice Cream Sandwich*), que corresponde con el nivel 15 de la API. Al establecer este mínimo, se asegura alcanzar hasta al 94% de los dispositivos Android registrados. Para esta versión mínima, se ha detectado que no se visualiza el icono de la aplicación dispuesto en la cabecera de la misma. Exceptuando ese aspecto, el comportamiento y los tiempos de espera son similares en los distintos casos de uso y se consideran aceptables.
- Pruebas de mantenimiento y uso. Esta solución *software* se ha diseñado siguiendo un esquema modular, de tal manera que pueda ajustarse a posibles cambios en los requerimientos de la misma en un futuro. Por otro lado, esta herramienta puede ser usada fácilmente por los usuarios para los que ha sido diseñada, con el apoyo de la documentación disponible.

VII. CONCLUSIONES Y VÍAS FUTURAS

En resumen, se ha incorporado la versatilidad del dispositivo Chromecast en el ámbito docente, rompiendo las limitaciones que las conexiones cableadas imponían en el ponente o usuario final de esta aplicación. Por otro lado, debemos recordar que la incorporación del Chromecast a la infraestructura del aula supondría una reducción en el presupuesto que se destinaría a los ordenadores, cuya función sería equivalente.

Aunque se han alcanzado los objetivos que se propusieron en un principio, presentaremos una serie de mejoras o funcionalidades adicionales que pueden ser objeto de líneas futuras de trabajo:

- Soporte para otros sistemas operativos móviles como *iOS* o *Windows Phone*.
- Integración con el sistema de autenticación de la Universidad de Granada. Este aspecto es actualmente incompatible con la tecnología del dispositivo Chromecast.
- Sistema de recordatorio de contraseña.
- Compatibilidad con otros formatos como *Powerpoint*, *Word* de *Microsoft Office* o sus equivalentes para otros paquetes ofimáticos.
- Conexión con la nube. Se plantea la posibilidad de habilitar la conectividad con plataformas de almacenamiento en la nube como *Google Drive* o *Dropbox*.

En este apartado no sólo se quiere hacer referencia a la solución *software* alcanzada y su efectividad, sino que se pretenden recuperar los aspectos referentes a la tecnología y su implicación tanto en la sociedad en general, como en la educación en particular, que se introdujeron al principio de este escrito. Durante el desarrollo de este estudio ha surgido la reflexión sobre los límites de los avances de la tecnología, ¿cuáles serán las tendencias educativas del futuro? Es imposible predecir las tecnologías que existirán dentro de veinte o treinta años, pero sí se pueden observar las tendencias, sin olvidar que la clave es pensar cómo queremos que sea la educación de ese futuro y, llegado ese momento, emplear la tecnología existente

A. Referencias

AGRADECIMIENTOS

Han sido muchas las personas que han compartido conmigo esta etapa y a las que me gustaría expresar mi gratitud por las palabras de cariño, apoyo y confianza que siempre me han dedicado. Y no sólo durante el periodo que ha abarcado el desarrollo de este Trabajo Fin de Grado, sino más importante aún, durante todo el proceso formativo que ha supuesto este Grado. No ha sido un camino fácil. Gracias a todos por los buenos momentos.

REFERENCIAS

- [1] Andrew T. Kemp, John Preston, C. Steven Page, Rebecca Harper, Benita Dillard, "Technology and Teaching: A Conversation Among Faculty Regarding The Pros And Cons of Technology", en *The Qualitative Report*, 2014, volumen 19, artículo 6, págs. 1-23.
- [2] María Domingo, Pere Marquès, "Aulas 2.0 y uso de las TICs en la práctica docente", en *Revista Científica de Educomunicación, Comunicar*, 2011, volumen 19, número 37.
- [3] Chromecast. Disponible en: https://www.google.com/intl/es/es/chromecast/?utm_source=chromecast.com
- [4] Apple TV. Disponible en: <http://www.apple.com/es/tv/?cid=wwa-es-kwg-appletv>
- [5] Roku. Disponible en: <https://www.roku.com>
- [6] MatchStick. Disponible en: <http://www.matchstick.tv>
- [7] DIAL, *Discovery And Launch Protocol*, versión 1.7.2. Disponible en: <https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGlhbC%201tdWx0a%20XNjcmVlbi5vcmd8ZGlhbHxneDo4MDQ4YjgzYjQ3MTAwYjU>
- [8] PHP, *Hypertext Preprocessor*. Manual de usuario. Disponible en: <https://secure.php.net/manual/es/index.php>
- [9] Environment - Android Developers. Disponible en: <http://developer.android.com/intl/es/reference/android/os/Environment.html>
- [10] Google Cast. Disponible en: <https://www.google.com/cast/>
- [11] WebView - Android Developers. Disponible en: <http://developer.android.com/intl/es/reference/android/webkit/WebView.html>
- [12] Aplicación Custom Receiver. Disponible en: https://developers.google.com/cast/docs/custom_receiver
- [13] Aplicación Styled Media Receiver. Disponible en: https://developers.google.com/cast/docs/styled_receiver
- [14] NanoHTTTPD. Disponible en: <http://nanohttpd.org>
- [15] Media Router - Android Developers. Disponible en: <https://developer.android.com/intl/es/guide/topics/media/mediarouter.html>
- [16] Navigation Drawer - Android Developers. Disponible en: <http://developer.android.com/intl/es/training/implementing-navigation/nav-drawer.html>
- [17] José Vázquez, "Definición de Java Heap y Java Heap Dump". Disponible en: <http://www.josevazquez.net/definicion-de-java-heap-y-java-heap-dump/>
- [18] Fundación Telefónica, "20 Claves Educativas para el 2020. ¿Cómo debería ser la educación del siglo XXI?", *Encuentro Internacional de Educación*.



Irene Herrera López nacida en Granada el 26 de Junio de 1991, graduada en Ingeniería de Tecnologías de Telecomunicación, con mención en Telemática, por la Universidad de Granada en 2015. Actual alumna del Máster Oficial de Ingeniería de Telecomunicación en el curso 2015/2016 impartido por la Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación (ETSIT) de la Universidad de Granada.

Análisis de redes SDN utilizando *Mininet* e implementación de un *Deep Packet Inspector*

Tutor: Jorge Navarro Ortiz; e-mail: jorgenavarro@ugr.es

Titulación: Grado en Ingeniería de Tecnologías de telecomunicación
Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Autor: Manuel Sánchez López, e-mail: manysl@correo.ugr.es

Resumen- Las limitaciones de las redes actuales junto al creciente desarrollo de nuevos servicios y aplicaciones son algunas de las causas de la búsqueda de alternativas a la vigente arquitectura de red. *Software Defined Networking (SDN)* es un tipo de arquitectura de red dinámica, gestionable, económica y adaptable, siendo ideal para soportar las aplicaciones que se desarrollan hoy en día. Esta arquitectura separa los planos de control y de datos, permitiendo ofrecer una visión global de la red y facilitando a los programadores la gestión a partir de un controlador centralizado. Dicho controlador dotará al sistema de abstracción con el objetivo de facilitar la creación de nuevas aplicaciones y servicios de red. Tomando en consideración todo lo anterior, este Trabajo de Fin de Grado pretende mostrar una idea del funcionamiento de estas redes en un entorno real y su potencial para mejorar la calidad de servicio actual. En concreto, en este trabajo se ha diseñado e implementado un *Deep Packet Inspector* en el controlador, de forma que los paquetes son marcados basándonos en el tipo de tráfico, para más tarde ser utilizadas por otras funcionalidades de la red (e.g. gestión de colas) con el objetivo de priorizar ciertos tipos de tráfico y, en última instancia, ofrecer calidad de servicio. Este DPI, además de permitir la detección de tipos de tráfico comunes como pueden ser el tráfico *web* o VoIP, se centra en la detección del tráfico de vídeo de YouTube. Por otro lado, su diseño modular facilita al desarrollador añadir nuevos tipos de tráfico al mismo. Para conseguir detectar este último tipo de tráfico, debido a que YouTube encripta los datos mediante HTTPS, se ha desarrollado un método basado en la detección de los paquetes DNS referentes a YouTube para determinar estos flujos.

Palabras clave- SDN (*Software defined Network*), *Mininet*, *OpenFlow*, *ODL (OpenDayLight)*, *DPI (Deep Packet Inspector)*, *Calidad de Servicio (QoS - Quality of Service)*, *Java*, *Controller*, *ToS (Type of Services)*, *YouTube*.

I. INTRODUCCIÓN

El *streaming* de vídeo se está convirtiendo en uno de los servicios más demandados en la red, siendo el único que aumentará significativamente en los próximos años según Cisco [1]. Dentro de este tipo de tráfico, cabe destacar el referente a YouTube, ya que se encuentra entre las tres páginas *web* más visitadas en Internet. Sus estadísticas oficiales [2] reflejan un incremento de visualización de contenido de un 50% cada año.

Teniendo en cuenta estas perspectivas, el principal reto es ofrecer a los usuarios un servicio mejorando la calidad del

mismo. Cisco propone la implementación de las siguientes características [3]:

- Clasificación y marcado del tráfico de manera que la red pueda diferenciarlo.
- Marcado de tasas de tráfico por encima de los umbrales específicos como prioridad menor.
- Descartar paquetes cuando las tasas alcancen umbrales específicos.
- Tanto la planificación de paquetes como los paquetes de mayor prioridad transmiten desde las colas de salida antes que los paquetes de menor prioridad.
- Gestión de colas de salida.

Por otro lado, las arquitecturas de red implementadas en la actualidad reflejan ciertas limitaciones a la hora de adaptarse a los nuevos servicios de red. Esto provoca la necesidad de implementar una nueva arquitectura capaz de paliar los problemas derivados de las nuevas necesidades de los usuarios.

Las redes definidas por *software (SDN)* tratan de dar respuesta a las limitaciones de la arquitectura de red actual, ofreciendo una visión global de la misma a partir de un controlador capaz de gestionar un grupo de dispositivos enrutadores. Desacoplando el plano de control del plano de datos se consigue centralizar la gestión y administración de la red en estos controladores.

Además, las redes SDN están diseñadas para proporcionar los siguientes beneficios adicionales:

- Reducción del *Capex (Capital Expenditures)* mediante la posibilidad de reutilizar *hardware* existente y reducir *Opex (Operating Expense)* ya que SDN permite un control algorítmico de los elementos de red.
- Proporcionar agilidad y flexibilidad, permitiendo a organizaciones desplegar aplicaciones, servicios e infraestructuras rápidamente.
- Permite innovar creando nuevos tipos de aplicaciones y modelos de negocio.

II. ANÁLISIS DE OBJETIVOS

El objetivo principal de este proyecto es la implementación de un *Deep packet Inspector* en un entorno de red SDN. Su desarrollo se lleva a cabo en el controlador de red por el que pasarán previamente los paquetes antes de añadir los flujos.

Dentro de nuestro DPI detectaremos diferentes tipos de flujo como son el de tipo *web*, VoIP y tráfico de vídeo

procedente de YouTube. Este último es la piedra angular de nuestro detector de paquetes y al que mayor tiempo se le ha dedicado para su desarrollo.

Una vez detectados estos flujos se añadirá cierta calidad de servicio mediante la modificación del campo *DiffServ* del protocolo IP. Para ello modificaremos el *ToS (Type of Service)* de los paquetes a un valor que lo diferencie del resto de tráfico de la red.

Como objetivos adicionales cabe destacar dos fundamentalmente. El primero trata de reproducir el DPI tanto en la arquitectura *AD-SAL (API-Driven Service Abstraction Layer)*, que es la que más desarrollada está dentro de *OpenDayLight*, como *MD-SAL (Model-Driven SAL)*, la cual se postula como la dominante en los próximos años. El segundo objetivo adicional corresponde a la consecución de un *producto* aislado e independiente de *Deep Packet Inspector* capaz de ser introducido en una red actual sin ningún tipo de incompatibilidad. De esta forma lo que se promueve es la inclusión de SDN y QoS en las redes que imperan a día de hoy sin necesidad de grandes cambios en la misma.

III. ESTUDIO DEL ESTADO DEL ARTE

A. *DPIs open source.*

Los DPIs de código abierto más destacados actualmente son los siguientes:

- ***nDPI.***

Este inspector de paquetes [4] forma parte de la herramienta para monitorizar paquetes *ntop*. Publicado bajo licencia LGPL, está disponible tanto para plataformas UNIX como para Windows. La lista de protocolos soportados por *nDPI* es muy extensa, entre los que destacan FTP, POP, SMTP, IMAP, DNS, IPP, HTTP, AVI, MPEG, RTSP, RTP, SSH, SSL o YouTube. La detección de éste último es importante ya que es uno de los tráficos que nosotros detectamos en el DPI que conforma nuestro proyecto.

- ***Bro-IDS.***

Bro Network Security Monitor [5] es una herramienta de análisis bastante potente. Es adaptable, lo cual permite:

- proporcionar políticas de monitorización específicas dependiendo del entorno.
- realizar un análisis profundo que facilite el reconocimiento de un gran número de protocolos.
- habilitar el análisis de la semántica de alto nivel en la capa de aplicación.

Además, es flexible, no restringiendo a detecciones particulares, sino que proporciona interfaces abiertas. Esto posibilita el intercambio de información con aplicaciones en tiempo real.

B. *DPIs comerciales.*

A continuación vamos a mostrar algunos de los inspectores profundos de paquetes más importantes del mercado y las compañías que venden estos productos según la página de *Policy Control* [6].

- ***Policy Enforcement Manager.***

Este DPI [7] proporciona la información necesaria para entender el comportamiento del tráfico de los clientes y permite gestionarlo eficazmente, ofreciendo una amplia gama de aplicación de políticas.

Como otro de sus puntos fuertes expone la posibilidad de nuevos modelos de negocio y servicios y ofrece la posibilidad de crear servicios adaptados basados en el uso de aplicaciones

de abonado, la clasificación del tráfico y búsqueda de patrones, y lanzamiento de nuevos servicios de manera más rápida.

Por otro lado permite la monitorización de tráfico en tiempo real, posibilitando la modificación del ancho de banda dinámicamente e implementando ciertas políticas para reducir la congestión de la red.

F5 también apuesta por las redes SDN en su DPI, ofreciendo un servicio de arquitectura flexible y dinámica que responde eficientemente a las condiciones del tráfico y a los requerimientos de las aplicaciones.

- ***7750 Service Router – Mobile Gateway.***

Este *router* posee capacidades de DPI [8] además de ser una pasarela para la red móvil. Por lo tanto, este DPI actúa en las capas 4-7 de las redes móviles procesando el tráfico IP.

Implementando este DPI, Alcatel pretende monitorizar, informar y optimizar los usos de los recursos de la red, proteger estos recursos, habilitar nuevos servicios y crear una política de control de la red.

- ***DPI Application Platform (7000).***

Este DPI [9] creado por *Bivio* ofrece un alto rendimiento y es totalmente programable. Combina un único *hardware* de procesado de paquetes con una plataforma *software* que incluye estándares basados en Linux. Está diseñado específicamente para proporcionar una velocidad de procesado de paquetes sobre los 10Gbps, fusionando componentes de procesamiento de red con las CPU de procesamiento de aplicaciones para ofrecer un rendimiento y flexibilidad muy altos.

C. *Comparativa con DPIs del mercado.*

En cuanto a los DPIs *open source*, el más competitivo es *nDPI*. Este DPI ofrece la detección de una cantidad de protocolos y aplicaciones, entre las que se incluye el tráfico *YouTube* tal y como permite nuestro DPI. No obstante, su integración en las redes SDN no existe, por lo que en este aspecto el DPI implementado en este proyecto se encuentra a la vanguardia en este tipo de redes.

En lo referente a la comparativa entre DPIs comerciales, la principal ventaja radica en que nuestro DPI, al ser de licencia GPL, puede ser modificado y mejorado por el usuario que desee desarrollarlo. Además, es gratuito y permite la posibilidad de mejorar bastantes aspectos en cuanto a las redes SDN a partir de las herramientas utilizadas. En cuanto al precio por el que podemos conseguir estos DPIs, como el ofrecido por *Bivio* o *Alcatel*, vienen con *hardware* incorporado, por lo que sus precios se incrementan notablemente y, para obtenerlos, se debe contactar con los representantes previamente. De igual manera ocurre con el que oferta *F5*, aunque en este se puede adquirir de forma virtual.

Por otro lado, no existen muchos DPIs orientados a este tipo de redes, por lo que es una buena vía para comenzar a desarrollar este tipo de herramientas en las que presumiblemente serán el futuro de las comunicaciones en red.

IV. DISEÑO E IMPLEMENTACIÓN

En lo referente a nuestro DPI, hemos focalizado nuestros esfuerzos en la detección de tráfico *YouTube*, por lo que será este tipo de tráfico el que mayor importancia tenga. Sin embargo, también se han implementado las detecciones de otros tipos de tráfico más sencillos.

También se mostrará la flexibilidad de nuestro DPI para poder ser implementado como un producto independiente en

una red actual, acoplándose a otros entornos de red sin verse afectadas sus características.

A. Implementación de un DPI en controlador.

Comenzamos pues a describir la implementación de un DPI en *OpenDayLight* ofreciendo una emulación de red con *Mininet*. Vamos a centrarnos en los métodos que afectan de forma directa o indirecta al desarrollo del Deep Packet Inspector en el controlador.

• Detección de tráfico web y VoIP.

Para la detección de estos dos tipos de tráfico nos hemos centrado en detectar los puertos que utilizan ambos servicios a parte del protocolo sobre el que estos servicios se soportan. Por lo tanto, su detección es más bien sencilla a priori.

Para el tráfico web hemos implementado una función que detecta si el puerto utilizado es el 80 (http) o el 443 (https). En el caso de web lo que buscaremos son paquetes de tipo TCP mientras que en VoIP son los paquetes UDP los que nos interesan.

```
/**
 * Función que comprueba si es tráfico VoIP
 * @param udpPacket paquete
 * @return devolvemos una booleana que informará
 * si el paquete es VoIP o no
 */
private boolean isVoIPTraffic(UDP udpPacket){
    boolean VoIPTraffic = false;

    if( udpPacket.getDestinationPort()==4569
        || udpPacket.getDestinationPort()==9082
        /*Prueba Skype --> UDP*/ ){
        VoIPTraffic=true;
    }
    return VoIPTraffic;
}
/**
 * Función que comprueba si el tráfico es de tipo
 * WEB
 * @param tcpPacket Paquete tcp recibido
 * @return devolvemos una booleana que informará
 * si el paquete es http o no
 */
private boolean isWebTraffic(TCP tcpPacket){

    boolean webTraffic = false;
    int dstPort = tcpPacket.getDestinationPort();
    int srcPort = tcpPacket.getSourcePort();
    if(dstPort==80 || srcPort==80 || dstPort==443 ||
    srcPort==443){
        webTraffic=true;
    }
    return webTraffic;
}
}
```

Con estas dos funciones seremos capaces de identificar estos dos tipos de tráfico. Esta sería la parte correspondiente al DPI. No obstante, el controlador deberá implementar dichas funciones y añadir el flujo correspondiente.

• Detección de tráfico YouTube.

La detección de tráfico YouTube podemos dividirla en dos partes. En primer lugar, la detección de los paquetes DNS referentes a este tipo de tráfico es fundamental para conseguir la dirección del servidor que nos proporcionará el vídeo. Por lo tanto, en primera instancia hay que realizar un análisis (*parsing*) de los mensajes DNS. En segundo lugar habrá que

esperar la llegada de los paquetes con la dirección previamente almacenada para añadir el flujo correspondiente.

La primera condición para comenzar el proceso de detección de tráfico YouTube será la llegada de un paquete DNS. Si dicho paquete DNS contiene el *string* “XXX.googlevideo.com” podremos identificarlo como un paquete enviado por los servidores DNS con información relativa al tráfico de YouTube. El método implementado para este cometido se muestra a continuación:

```
/**
 * Función que comprueba si el tráfico pertenece
 * a un flujo de streaming de vídeo enviado por
 * YouTube.
 * @return Nos devuelve una booleana indicando si
 * el paquete es de tráfico YouTube o no.
 */
private boolean isYoutubeTraffic(byte[]
arrayPacketData){
    boolean youtubeTraffic = false;
    String datosUDP = new String(arrayPacketData,
                                UTF_8);
    if(datosUDP.contains("googlevideo")){
        // ..añadiremos el ToS más adelante en el
        flujo en caso de "true"
        log.info("Es YouTube");
        youtubeTraffic=true;
    }
    return youtubeTraffic;
}
```

Una vez detectemos dicho *string* pasaremos a analizar el paquete DNS. El análisis hemos de implementarlo ya que, a día de hoy, no existe un módulo de *parsing* para DNS implementado en *OpenDayLight*. Por lo tanto, a partir de los paquetes UDP que vayan dirigidos al puerto 53 se buscará la respuesta a la petición DNS realizada por el equipo que desea acceder al contenido del vídeo de YouTube.

En primer lugar habrá que comprobar que el paquete es de tipo YouTube y que el puerto origen es el 53, ya que lo que buscamos son las respuestas DNS, en las que se incluye la dirección IP del servidor de vídeo. Una vez comprobados estos dos requisitos añadimos la dirección IP del servidor DNS de YouTube a una lista de direcciones. El motivo de añadir el servidor DNS se debe a que necesitamos no añadir flujos con esta dirección, ya que en ese caso no podríamos optar a detectar nuevos tráficos de vídeo, ya que el tráfico dejaría de pasar por el controlador. Tras registrar esta dirección, pasamos a realizar el análisis del paquete mediante la función “*parseoDNS*”. El *parsing* se realizará a partir del *payload* del paquete UDP en los paquetes referentes a la respuesta DNS.

Primero se muestran 4 bytes que son la cabecera del paquete DNS, los cuales no contienen información interesante para nosotros. Seguidamente encontramos 16 bits indican el número de peticiones. Esto es importante para después poder determinar dónde empiezan las respuestas. Otros 16 bits que indican el número de respuestas. Después tenemos 16 bits, que representan el número de autoridades. A continuación, 16 bits, indican el número de registros de recursos adicionales. A partir de aquí empiezan las peticiones (preguntas que se han hecho en mensajes anteriores y que se responden con este paquete). Tienen primero la cadena de la pregunta, 16 bits que indican el tipo de registros de recursos (*Address*, *Canonical Name*, etcétera) y 16 bits que indican la clase.

Para determinar la longitud de las peticiones (cuya longitud es variable), hemos optado por la detección del carácter “\0” (un byte a 00) que indica el fin de la cadena. Tras

la detección de todas las peticiones pasamos a las respuestas, donde estará la dirección IP del servidor de vídeo que buscamos.

Las respuestas tienen un campo de 16 bits que hace referencia al recurso sobre el que se responde, después 16 bits de tipo (buscaremos una respuesta de tipo *Address*=1, ya que en esta respuesta se incluirá la dirección IP del servidor), 16 bits de clase, 32 bits de TTL (*Time To Live*), 16 bits de longitud de la respuesta (4 bytes si la respuesta es de tipo *Address*, ya que hay que poner la dirección IP; pero cualquier valor si es una respuesta de tipo *Canonical Name*, ya que vendría representado por una cadena de caracteres indicando la *url*) y después la respuesta (con el número de bytes indicado por el campo anterior).

Una vez claros estos conceptos los hemos plasmado en el código para implementar un análisis que facilitara la dirección IP del servidor.

```
/**
 * Función encargada de parsear los paquetes DNS.
 * @param udpRawPayload: paquete UDP en el que se
 encuentra incluido el paquete DNS
 * @return srcIPservidorVideoYoutube: IP del
 servidor.
 */
private InetAddress parseoDNS(byte[]
                                udpRawPayload){
    String udpRawPayDataISO = new
        String(udpRawPayload, ISO_8859_1);

// Inicialización de las variables a utilizar
...

if(numAnswers!=0){
    for(int j=0;j<numQueries;j++){
        urlencontrada=false;
        for(int i=longitudPeticiones;
            i<udpRawPayload.length-2;i++){
            if(udpRawPayload[i+1]==0 &&
                !urlencontrada){
                log.info("Primer 0 encontrado");
                if(udpRawPayload[i+2]==0){
                    log.info("Segundo 0 encontrado");
                    longitudPeticiones=i+8;
                    log.info("número de petición "+j);
                    j++;
                    urlVideoYoutube=udpRawPayDataISO.
                        substring(13,i+1);
                    longitudURL=urlVideoYoutube.length();
                    urlencontrada=true;
                }
            }
        }
    }
    longitudRespuestas=longitudPeticiones;
    for(int j=0; j<numAnswers;j++){
        Type = new byte[numAnswers];
        Type[j] =
            udpRawPayload[longitudRespuestas+1];
        log.info("Type: "+Type[j]);
        Class = new byte[numAnswers];
        Class[j] =
            udpRawPayload[longitudRespuestas+3];
        log.info("Class: "+Class[j]);
        if (Type[j]==5){
            longitudRespuestas =
```

```
longitudRespuestas+14+(longitudURL-17);
//El primary Name tiene 17 bytes menos
que la url
    }
    if (Type[j]==1){
        byte[] ipAddr=new byte[4];
        ipAddr[0] =
            udpRawPayload[longitudRespuestas+10];
        ipAddr[1]=
            udpRawPayload[longitudRespuestas+11];
        ipAddr[2]=
            udpRawPayload[longitudRespuestas+12];
        ipAddr[3] =
            udpRawPayload[longitudRespuestas+13];
        try {
            srcIPservidorVideoYoutube =
                srcIPservidorVideoYoutube.
                    getByAddress(ipAddr);
        } catch (UnknownHostException e) {
            e.printStackTrace();
        }
    }
}
log.info("IP del servidor de Vídeo YouTube:
    "+srcIPservidorVideoYoutube);
return srcIPservidorVideoYoutube;
}
```

En este punto ya tendríamos la dirección IP del servidor de YouTube. El siguiente paso sería almacenarla en una lista de direcciones para que, una vez llegue un paquete de esta dirección, se detecte y se proceda a añadir un nuevo flujo cuyo ToS lo diferencie del resto para asignarle si es preciso una calidad de servicio mayor.

Podemos ver cómo se incluye una variable booleana para indicar si es tráfico YouTube o no. Esta variable booleana la utilizaremos a la hora de añadir el flujo. En el siguiente fragmento de código se llevaría a cabo la implementación de acciones al flujo.

```
if (l3Pkt instanceof IPv4) {
    IPv4 ipv4Pkt = (IPv4) l3Pkt;
    Object l4Datagram = ipv4Pkt.getPayload();
    srcIPAddr = intToInetAddress(
        ipv4Pkt.getSourceAddress());
    dstIPAddr = intToInetAddress(
        ipv4Pkt.getDestinationAddress());
    if(srcIPservidorVideoYoutubeList.contains(
        srcIPAddr) && srcIPAddr!=null){
        servidorVideoYoutube=true;
        log.info("Lista de direcciones de YouTube:
            "+srcIPservidorVideoYoutubeList);
    } else{
        log.info("srcIP: "+srcIPAddr+" ipDNS:
            "+srcIPservidorVideoYoutube);
    }
}
```

Se puede apreciar cómo se incluye una variable booleana para indicar si es tráfico YouTube o no. Esta variable booleana la utilizaremos a la hora de añadir el flujo. En el siguiente fragmento de código se llevaría a cabo la implementación de acciones al flujo.

```
if (dst_connector != null) {
    List<Action> actions = new
        ArrayList<Action>();
    if(servidorVideoYoutube){
```

```

servidorVideoYoutube=false;
añadeFlujo=true;
log.info("Nuevo ToS para tráfico YouTube");
actions.add(new SetNwTos(8));
} else if(direccionesServidoresDNSYoutube.
contains(srcIPAddr)){
añadeFlujo=false;
log.info("IP de servidor DNS YouTube; no
añadir flujo");
...

```

• Ejemplo práctico.

Pasamos ahora a realizar un ejemplo de detección de tráfico YouTube en una red conformada por un *switch* del que penden tres *hosts*.

En primer lugar arrancamos el controlador y la topología en dos terminales diferentes. La topología constará de un único *switch* al que se conectarán tres *hosts*.

Una vez la red está en marcha vamos a desplegar una consola para la *host 1*. En dicho terminal vamos a configurar el servidor DNS que recibirá nuestras peticiones. Esto variará según la red a la que estemos conectados. En nuestro proyecto se ha trabajado en dos redes: la red doméstica y la red de la UGR. Para la red doméstica indicamos el servidor DNS 8.8.8.8 correspondiente a Google. Si estamos en la red de la UGR podemos indicar el 150.214.35.10. Estos cambios se realizan en el fichero `/etc/resolv.conf`.

Ejecutamos el navegador de Firefox en la terminal del *host 1* y buscamos un vídeo de YouTube para reproducirlo:



Fig. 1. Reproducción de vídeo de YouTube en *host 1*.

Ahora vamos a pasar a ver si se han instalado los flujos del servidor de vídeo de YouTube tal y como deseamos. Primero ejecutamos el comando que muestra las tablas de flujo del *switch*:

```

mininet> sh ovs-ofctl -O OpenFlow13 dump-flows s1

```

En la siguiente imagen se muestra el flujo correspondiente con el ToS cambiado al valor que deseábamos:

```

cookie=0x0, duration=44.784s, table=0, n_packets=9986, n_bytes=14136627, send_t
low_ren_priority=0, tcp, in_port=4, dl_src=00:00:00:00:00:04, dl_dst=00:00:00:00:00:00:01, nw_src=130.206.193.12, nw_dst=10.0.0.1, tp_dst=40434 actions=set_field:8->nw_to_s_shifted,output:1

```

Fig. 2. Flujo añadido de tráfico de vídeo YouTube.

Esta dirección a la que se le asigna un nuevo ToS y que, por lo tanto, se presenta como la del servidor de vídeo YouTube se ha contrastado utilizando Wireshark. Lo que se va a realizar mediante el cambio del ToS será modificar el campo DSCP (Differentiated Services Code Point) dentro del campo DiffServ de la cabecera IP al valor indicado. Además, como

podemos comprobar el número de paquetes es muy elevado, claro reflejo de que es un streaming de vídeo.

B. Implementación de controlador con DPI independiente y evaluación de resultados.

El objetivo de este punto es aislar nuestro producto de tal forma que sea capaz de operar correctamente al incluirlo dentro de una red actual. El objetivo por tanto será poder incluir nuestro controlador en la red y que éste inspeccione todo el tráfico que pasa por ella y sea capaz de modificar el ToS de los paquetes YouTube.

Para realizar este proceso hemos optado por la utilización de máquinas virtuales, las cuales facilitan el proceso de conexión entre equipos sin la necesidad de disponer de ellos físicamente. La herramienta utilizada para trabajar con las máquinas virtuales ha sido *VirtualBox* [10]. A continuación se muestra cómo quedaría la topología que pasaremos a explicar detalladamente en los siguientes párrafos.

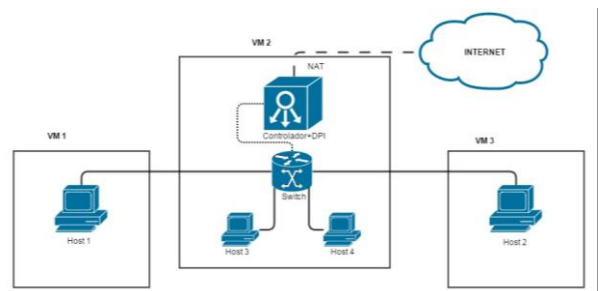


Fig. 3. Escenario de implementación de controlador con DPI independiente.

En una máquina virtual instalaremos el controlador con el DPI incorporado. Esta máquina virtual tendrá acceso a Internet. Independientemente, se crearán dos máquinas virtuales que contengan sendos *hosts* para comprobar el correcto funcionamiento. Adicionalmente se han añadido dos *hosts* virtuales en la máquina virtual donde reside el controlador para aumentar la carga del sistema.

Una vez instalados *OpenDayLight* y *Mininet* en la máquina virtual del controlador abrimos dos terminales y ejecutamos:

```

$ ./run.sh
$ sudo python topo--sw-2ports.py

```

Una vez configurado el sistema ejecutamos en cada uno de los cuatro *hosts* (dos dentro de la propia topología de *Mininet* más los dos incluidos en sendas máquinas virtuales) abrimos un navegador y reproducimos un vídeo de YouTube como se muestra en la siguiente figura.

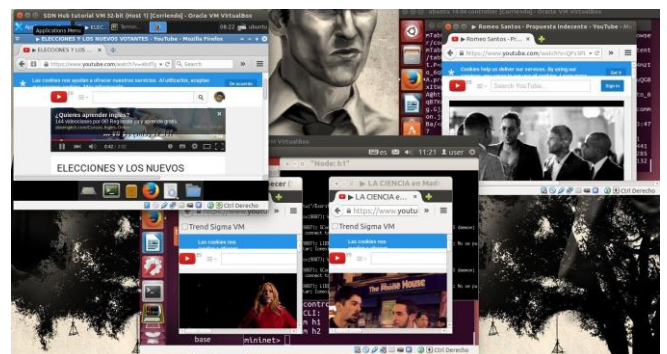


Fig. 4. Reproducción de vídeo desde cuatro *hosts* diferentes.

Como vemos, los vídeos se están reproduciendo al mismo tiempo por lo que debería haber añadido cuatro flujos al menos referentes a dichos flujos. En efecto, al comprobar los flujos instalados, se corrobora que se han instalado estos cuatro flujos con el ToS cambiado para las direcciones IP de los servidores de vídeo. Además se incluyen otros flujos que no son propiamente del vídeo en sí pero que tienen que ver con el servidor de vídeo. En las siguientes figuras mostramos cada uno de los cuatro flujos instalados que presumiblemente transportan el vídeo por la red:

```

cookie=0x0, duration=1934.096s, table=0, n_packets=14788, n_bytes=22368104, priority=0, tcp, in_port=3, dl_src=d6:ce:c5:e8:48:74, dl_dst=00:00:00:00:00:00:01, nw_src=173.194.15.179, nw_dst=192.168.1.1, tp_dst=47961 actions=mod_nw tos:32, output:5
cookie=0x0, duration=40.29s, table=0, n_packets=12300, n_bytes=18479798, priority=0, tcp, in_port=3, dl_src=d6:ce:c5:e8:48:74, dl_dst=00:00:00:00:00:02, nw_src=173.194.9.233, nw_dst=192.168.1.2, tp_dst=59726 actions=mod_nw tos:32, output:4
cookie=0x0, duration=276.625s, table=0, n_packets=13877, n_bytes=20848624, priority=0, tcp, in_port=3, dl_src=d6:ce:c5:e8:48:74, dl_dst=08:00:27:5c:93:8c, nw_src=173.194.9.145, nw_dst=192.168.1.4, tp_dst=46868 actions=mod_nw tos:32, output:1
cookie=0x0, duration=279.344s, table=0, n_packets=2118, n_bytes=3169482, priority=0, tcp, in_port=3, dl_src=d6:ce:c5:e8:48:74, dl_dst=08:00:27:8c:d9:3d, nw_src=173.194.9.134, nw_dst=192.168.1.5, tp_dst=49070 actions=mod_nw tos:32, output:2
    
```

Fig. 5. Flujos de vídeos de YouTube correspondientes a cada host.

El sistema, pese a estar ejecutado en un entorno bastante limitado ya que se dispone únicamente de un portátil con tres máquinas virtuales corriendo en él, ha respondido ante cuatro equipos diferentes solicitando vídeos de YouTube. Además de los flujos referentes a YouTube se han añadido multitud de flujos de otro tipo al mismo tiempo. Con esta prueba se trata de acercar el concepto de integrar nuestro DPI en una red real de forma flexible. Para realizar pruebas de carga habría que tener acceso a una red donde haya carga real como, por ejemplo, la red de la UGR. No obstante estas pruebas están más allá del ámbito de este proyecto.

V. CONCLUSIONES

En el presente Trabajo de Fin de Grado se ha diseñado un Deep Packet Inspector haciendo uso de *Mininet* y gracias a otra serie de herramientas que se han ido acoplando en diferentes partes del proyecto. De esta forma se ha conseguido detectar, entre otros, el tráfico generado por YouTube dentro de una red emulada SDN. Una vez realizada esta detección hemos podido modificar dicho tráfico de tal forma que se añadiese una acción a la tabla de flujo capaz de modificar el ToS de los paquetes y propiciar así la posibilidad de ofrecer una Calidad de Servicio (QoS) en la red. Además, se ha experimentado con la implementación para comprobar el correcto funcionamiento del sistema desarrollado.

Las principales contribuciones que nuestro proyecto ofrece son:

- Ofrecer una gran recopilación de información en lo referente a este tipo de redes emergentes como son las SDN en cuanto a diferentes aspectos como son la gestión de red, monitorización o calidad de servicio.
- Se ha experimentado con *Mininet* y *OpenDayLight* de manera que se aporta una gran fuente de información para comenzar a trabajar con estas herramientas.
- El DPI desarrollado permite la detección precisa de los paquetes del servidor de vídeo YouTube. Para ello se ha realizado previamente un análisis exhaustivo de paquetes

DNS necesario para obtener la dirección IP del servidor, a parte de un procedimiento posterior para detectar dicha IP tras la llegada de paquetes. Lo que nos permite este DPI por tanto será dar los medios (a través del campo ToS) para que la red proporcione calidad de servicio a ciertas aplicaciones o servicios, como en este caso el servicio de vídeo de YouTube.

- En cuanto al análisis de paquetes DNS, una aportación importante es el desarrollo de un analizador de este tipo de paquetes, el cual no estaba implementado hasta el momento en *OpenDayLight*.
- Otra aportación importante es la posibilidad de implementar a partir del DPI la detección de una gran multitud de tráfico referente a otras aplicaciones o servicios, ya que es un DPI modular y flexible en su desarrollo.

REFERENCIAS

- [1] Cisco Corporation, «The Zettabyte Era—Trends and Analysis,» [En línea]. Available: http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/VNI_Hyperconnectivity_WP.html.
- [2] YouTube, «Estadísticas oficiales de YouTube,» [En línea]. Available: <http://www.ciscopress.com/articles/article.asp?p=170743>.
- [3] CISCO, «CCNP Self-Study: Understanding and Implementing Quality of Service in Cisco Multilayer Switched Networks,» [En línea]. Available: <http://www.ciscopress.com/articles/article.asp?p=170743>.
- [4] ntop, «nDPI,» [En línea]. Available: <http://www.ntop.org/products/deep-packet-inspection/ndpi/>.
- [5] BRO, «The Bro Network Security Monitor,» [En línea]. Available: <https://www.bro.org/>.
- [6] The fast mode, «Deep Packet Inspection Vendors & Products,» [En línea]. Available: <http://www.thefastmode.com/deep-packet-inspection-vendors>.
- [7] F5, «Policy Enforcement Manager,» [En línea]. Available: <https://f5.com/products/service-provider-products/policy-enforcement-manager>.
- [8] Alcatel-lucent, «7750 Service Router - Mobile Gateway,» [En línea]. Available: <https://www.alcatel-lucent.com/products/7750-service-router-mobile-gateway>.
- [9] Bivio, «Bivio products,» [En línea]. Available: <http://www.bivio.net/products/b7000/>.
- [10] ORACLE, «VirtualBox,» [En línea]. Available: <https://www.virtualbox.org/>.
- [11] SNORT, «SNORT,» [En línea]. Available: <https://www.snort.org/>.

Diseño y evaluación de un servicio OpenFlow de provisión de Calidad de Experiencia sobre Mininet

Tutor: Juan José Ramos Muñoz; e-mail: jjramos@ugr.es
Titulación: Grado en Ingeniería de tecnologías de Telecomunicación
Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Autor: Cristian Alfonso Prieto Sánchez, e-mail: cristian.pr.sa@gmail.com

Resumen—Las redes definidas por software suponen una revolución tecnológica que se prevé domine el mercado de las redes de datos en el futuro, debido a la innegable necesidad de evolución de las redes actuales. Este nuevo paradigma de red presenta enormes ventajas teóricas frente a las redes tradicionales, y será objeto de este trabajo el estudio y explotación de estas nuevas capacidades.

Las enormes ventajas que presenta este nuevo paradigma, han atraído la atención de grandes compañías de la industria, que no han tardado en diseñar y lanzar al mercado sus propias soluciones. La expectación generada en torno a las redes definidas por software no es más que otro indicador de su potencial.

En este trabajo se realiza un amplio estudio que pretende desgranar los aspectos más técnicos de esta nueva arquitectura de red, confirmar todos los beneficios que proporcionan, y por último desarrollar servicios sobre la arquitectura Software Defined Networking para la provisión de calidad de experiencia. La solución final propuesta pretende aprovechar las características principales de SDN, centradas sobre tráfico multimedia. Para ello es necesario el diseño de un protocolo capaz de realizar diferenciación de tráfico, y además, reaccionar rápidamente ante cambios de red.

Palabras clave—Controlador, Dijkstra, Java, Mininet, OpenDayLight, OpenFlow, QoE, QoS, Redes, Routing, SDN.

I. INTRODUCCIÓN

EL crecimiento del tráfico multimedia experimentado en los últimos años, unido al crecimiento exponencial esperado (como las previsiones realizadas por Cisco en [1]), ahondan en la necesidad de diseñar una arquitectura de red capaz de soportar y cumplir los requisitos para soportar tal cantidad de tráfico.

Por otro lado, es recurrente la búsqueda de técnicas o arquitecturas de red capaces de reducir costes y aumentar capacidades sobre las que ofrecer nuevos servicios a los usuarios, por parte de operadores de red.

Estas dos necesidades básicas en la industria han impulsado la adopción de redes definidas por software (Software Defined Networking) como solución con mayor aceptación, dadas las ventajas en las se hará hincapié a continuación.

A. Motivación

Para justificar este trabajo atendemos a los dos conceptos ya presentados. Por un lado la necesidad de responder ante las demandas de tráfico de manera flexible y rápida, mejorando la gestión y automatización de redes. Por otro, el tráfico

multimedia precisa unos fuertes requisitos para su correcta transmisión. Esto lleva a las grandes compañías del sector a buscar nuevas fórmulas y modelos que permitan satisfacer las demandas para satisfacer los conceptos que se presentan a continuación.

- **QoS.** Definimos QoS como el rendimiento promedio de una red telemática. Cuantitativamente mide la calidad de los servicios en base a varios aspectos del servicio de red, tales como tasa de errores, ancho de banda, rendimiento, etc.
- **QoE.** Cuando se trata el término QoE se hace referencia a la calidad subjetiva que es percibida por un usuario final al disfrutar de un servicio ofrecido sobre una red telemática. Es una medida que se ve afectada por todos los elementos de la red extremo a extremo, que repercute en la percepción final por parte del usuario.

B. Objetivos

Los objetivos que a continuación se describen, pretenden sentar las bases sobre las cuales apoyar el resto del trabajo:

- Revisar el estado de la técnica sobre redes SDN, protocolo OpenFlow y principales controladores.
- Revisar el estado de la técnica sobre emulación de redes SDN, Mininet.
- Diseñar una solución capaz de proveer QoE en redes SDN.
- Implementar y evaluar dicha solución.

II. REVISIÓN DEL ESTADO DEL ARTE

A fin de sentar las bases sobre las que se desarrolle la solución que se propondrá, es necesario realizar un estudio previo sobre los principales conceptos de la arquitectura SDN, así como estudiar las principales tecnologías para el desarrollo de la solución.

A. Arquitectura Software Defined Networking

Las redes definidas por software se apoyan en la estructura que se muestra en la imagen 1, donde se aprecia con claridad la separación entre planos de control y datos. Esta separación supone un cambio respecto a las tecnologías que hasta ahora han dominado el sector. Se pretende que la toma de decisiones (plano de control) esté separada de las funciones de reenvío de paquetes (plano de datos).

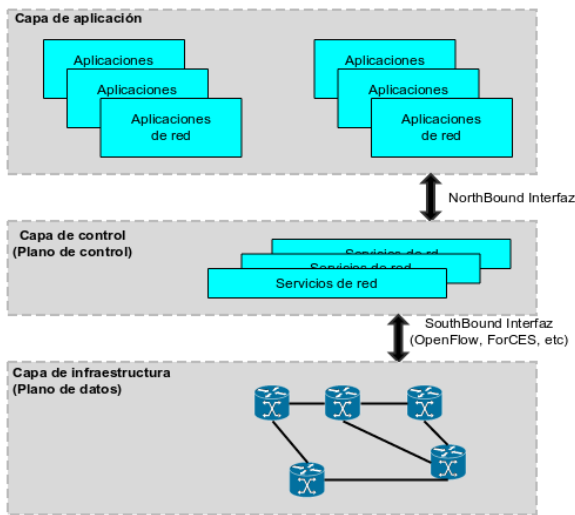


Fig. 1. Arquitectura para la separación de planos de control y datos.

Gracias a esta interpretación de red se consigue que todo el proceso de toma de decisiones se produzca en un solo plano, permitiendo la centralización de red. Ésta se verá reflejada en un elemento central, controlador, que será el encargado de la toma de decisiones en la red. De este modo se consigue:

- **Beneficios técnicos.**
 - Mejora en la toma de decisiones.
 - Mayor agilidad en la transmisión de órdenes a los elementos de red.
 - Mayor agilidad en la convergencia de red, al mantener una visión global en un único elemento.
 - Reducción de la complejidad de los elementos de red.
- **Beneficios económicos.**
 - Reducción de CAPEX (CAPital EXpenditure). Gasto asociado a la inversión inicial. La reducción de la complejidad se traduce en equipos con menor coste.
 - Reducción de OPEX (OPerational EXpenditure). Gasto asociado a los costes de operación. Mantener una visión global y centralizada de red, facilita el trabajo de los administradores de red.

Esta arquitectura está apoyada en los elementos que se definen a continuación.

- **Controlador SDN.** Elemento central que se encarga de aplicar la lógica definida (aplicaciones *NorthBound*) con los elementos de red (*southBound*). En la figura 2 se encuentra reflejada la comunicación entre controlador y el resto de elementos de la arquitectura, mediante el uso de APIs. Uno de los aspectos más interesantes de la arquitectura SDN, es que pretende aprovechar protocolos de red tradicionales (OSPF (Open Shortest Path First), BGP (Border Gateway Protocol), etc), aumentando de este modo la compatibilidad con los equipos de transporte.
- **APIs (Application Programming Interface).** Son las interfaces encargadas de permitir al controlador la comunicación con las aplicaciones SDN y con los elementos de red.

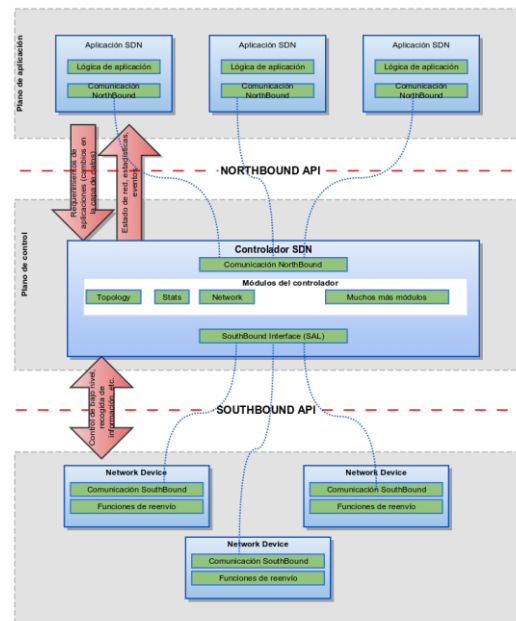


Fig. 2. Visión esquemática de la estructura SDN.

NorthBound API. Interfaz usada para facilitar la comunicación de aplicaciones con el controlador SDN. Esta API es crítica debido a la variedad de aplicaciones que pueden comunicarse con el controlador SDN, y que pueden provocar incompatibilidades siempre que no se respecto el modelo definido en cada caso. Para evitar problemas de este tipo, la Open Networking Foundation (<https://www.opennetworking.org/index.php>) crea un grupo de trabajo centrado en la definición de esta API, y el desarrollo de prototipos que ilustren las posibilidades de uso de ésta.

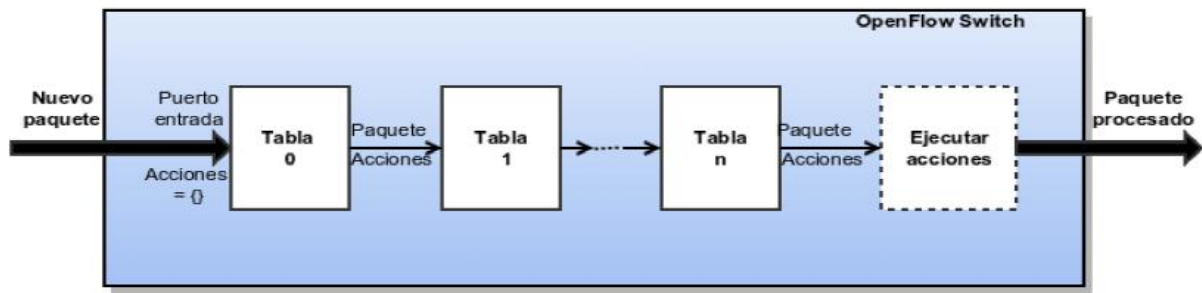
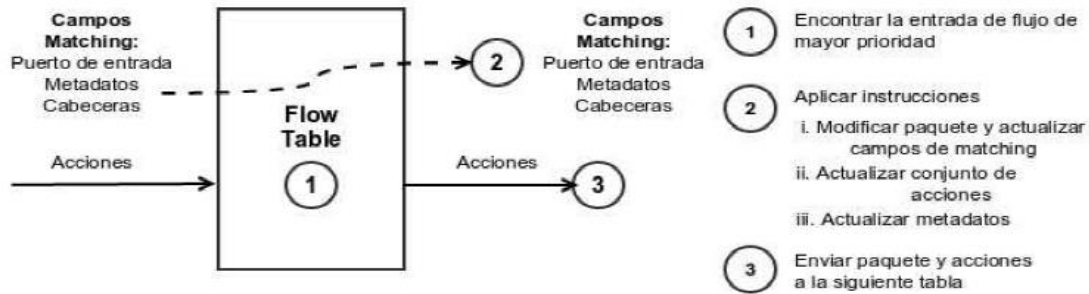
SouthBound API. Es la interfaz encargada de posibilitar la comunicación del controlador con los elementos de red. Permite hacer cambios dinámicos en los elementos para adecuarse a las necesidades de cada elemento. Al adaptarse a los elementos subyacentes, se obtiene independencia de fabricantes, favoreciendo la visión abierta de la arquitectura. Existen diferentes implementaciones de esta API, donde la más extendida (a día de hoy) es OpenFlow.

B. OpenFlow

OpenFlow [2] es la primera interfaz de comunicaciones entre las capas de control y transporte en una arquitectura SDN. Permite el acceso directo y la manipulación de los elementos del plano de control. Las tecnologías SDN basadas en OpenFlow están capacitadas para abordar el gran ancho de banda y la naturaleza dinámica de las aplicaciones actuales.

OpenFlow surge como solución para separar distintos tipos de tráfico dentro de switches y routers pensados solo para el transporte de datos. Uno de los objetivos es poder utilizar las tablas de flujo que ya implementan los switches para conseguir el objetivo de separación de tráfico.

En el estándar OpenFlow se definen dos elementos principales.

(a) Paso de paquetes por los múltiples *match* de las tablas de flujo

(b) Procesamiento de paquete en cada tabla

Fig. 3. Procesamiento de paquetes a través del proceso *pipeline*.

- **Switch.** Encargado del procesamiento de paquetes de acuerdo a las reglas instaladas previamente por el controlador. Estas reglas son instaladas en tablas de flujos del switch.
- **Controlador.** Elemento central de una red SDN y OpenFlow, capaz de evaluar el estado de red y añadir o eliminar entradas de flujo en los switches OpenFlow, de acuerdo a las aplicaciones instaladas en el controlador. El controlador puede ser una simple aplicación instalada en un PC, que instale flujos de forma sistemática, o por otro lado, puede ser un elemento dedicado a reaccionar de forma dinámica al estado de red.

El funcionamiento de OpenFlow está basado en la definición de flujos e instalación en los switches. Una vez se definen los flujos, cada paquete que llega a un switch, es procesado por el proceso pipeline, resumido en la figura 3.

Este esquema indica el camino a seguir por cada paquete que llega al switch. Normalmente los paquetes encontrarán coincidencias a través del procesamiento y se les aplicarán las acciones asociadas. Sin embargo puede darse el caso de no encontrar *match* (emparejamiento) alguno. En este segundo caso, los switches tienen varias acciones disponibles, la más común suele ser reenviar el paquete al controlador, el cual analizará el paquete e instalará los flujos pertinentes.

C. OpenDayLight.

Entre los controladores disponibles para SDN (y que hagan uso de OpenFlow), OpenDayLight (<https://www.opendaylight.org/>) es el más destacado. Es un proyecto de tipo abierto bajo la Linux Foundation, y que además, cuenta con el apoyo de muchas de las empresas dominantes en la industria.

OpenDayLight cuenta con unas características que en el

posterior desarrollo nos serán muy útiles. Entre ellas destacar que el desarrollo de aplicaciones se realiza en JAVA y, que cuenta con una estructura modular como la de la figura 4.

Esta estructura está organizada en tres capas.

- **Aplicaciones de red.** En la capa superior se encuentran las aplicaciones diseñadas para encargarse del control y monitorización de red.
- **Controlador.** Es la capa central donde se manifiestan las abstracciones de SDN. El controlador OpenDayLight cuenta con una serie de módulos implementados que permiten a las aplicaciones de la capa superior obtener datos e información sobre el estado de red. Además cuenta con una serie de APIs que permiten el desarrollo de las aplicaciones superiores (API REST, JAVA API, DOM API). Además se implementan protocolos para la comunicación con elementos inferiores.
- **Elementos de red físicos y virtuales.** La capa inferior está constituida por aquellos elementos de red que son programables mediante los protocolos implementados por el controlador. Gracias a la capa de abstracción del controlador se consigue que los elementos sean compatibles con este.

Apoyándonos en OpenFlow y el controlador OpenDayLight, diseñaremos una solución capaz de cumplir los objetivos propuestos.

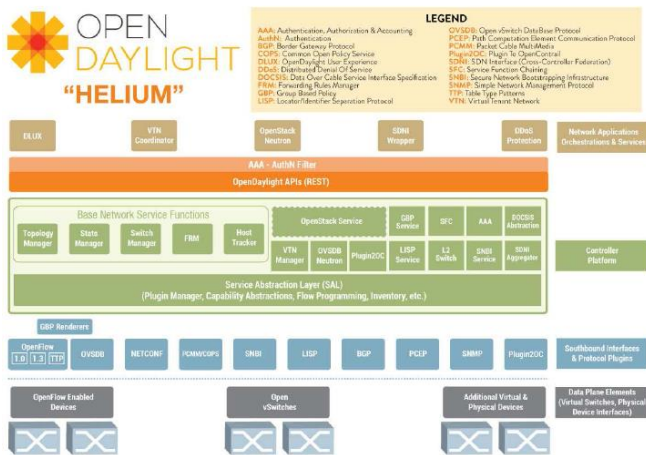


Fig. 4. Vista técnica del controlador OpenDayLight. Versión Helium.

III. DISEÑO DE LA SOLUCIÓN

A. Introducción.

La solución diseñada en este proyecto se basa en aprovechar la potencia de un controlador SDN centralizado, para poder satisfacer los requisitos de calidad de varios flujos multimedia.

Con este motivo, ha sido diseñado un protocolo de encaminamiento de estado de enlace para generar rutas óptimas por flujo multimedia. Este protocolo toma como costes distintas métricas que dependen del tipo de flujo para el que se crea la ruta. La solución se divide en 4 elementos diferenciados.

B. Elementos.

- **Clasificación de flujos de paquetes.** A fin de escoger rutas óptimas para cada flujo, es necesario identificar y analizar a qué tipo de flujo corresponden los paquetes. En la solución se proponen cuatro tipos de tráfico para analizar. Tráfico ICMP (Internet Control Message Protocol), tráfico TCP, tráfico RTP (Real Time-Transport Protocol) audio y tráfico RTP Vídeo.
- **Algoritmo de encaminamiento.** Una vez reconocido el tipo de flujo, se requiere la obtención de la mejor ruta en la red según el destino del flujo. Los parámetros de QoS de los enlaces que componen dicha ruta, deben maximizar la calidad que perciba el usuario final. Para aplicar el algoritmo de encaminamiento, previamente es necesario estimar un coste asociado para cada enlace. Finalmente se aplicará el algoritmo de encaminamiento Dijkstra [3] y [4].
- **Definición de matriz de costes por enlace.** La matriz de costes permite realizar una rápida asociación entre cada enlace y el coste asociado a este (para cada tipo de flujo). Para construir esta matriz de costes se definen varias funciones de calidad, y según el tipo de tráfico, se asignan unos pesos a cada función de calidad. Se definen cuatro funciones de coste: Latencia y jitter, pérdida de paquetes y carga en los enlaces.
- **Estimación de parámetros de calidad.** En función del tipo de flujo se ha de realizar una estimación sobre el peso de cada una de las funciones definidas para las matrices de costes. Como por ejemplo, la asignación realizada para RTP vídeo, donde la latencia tiene un peso menor a las pérdidas del enlace de transmisión.

- **Recuperación de enlaces.** Como método adicional a la provisión de calidad de experiencia, se ha previsto la inclusión de métodos de recuperación de enlaces. Este método (más propia de implementación) es el encargado de detección de cambios en la topología y eliminación de flujos obsoletos, que provocarían pérdidas de paquetes. Este método pretende ser capaz de detectar una caída de enlace y encaminar el tráfico por otra ruta del modo más rápido posible.

Con estos elementos se ha realizado la implementación de la solución, la cual no tiene cabida en este resumen. A continuación se presentan los resultados y conclusiones obtenidas.

IV. EVALUACIÓN DE LA SOLUCIÓN

Para llevar a cabo una correcta evaluación sobre la solución propuesta, se ha decidido realizar una evaluación gradual, donde cada fase del diseño e implementación se evalúa de forma individualizada y posteriormente se evalúa el sistema completo. Se definen las siguientes evaluaciones.

- **Evaluación sobre recolección de parámetros de red.** Esta primera fase está dedicada a comprobar la validez de los datos recogidos sobre la red, estadísticas de enlaces y latencias de estos.
- **Evaluación del cálculo de costes por enlace.** A lo largo de esta fase se ajustarán los parámetros (pesos) asociados a las funciones de coste en función de flujo. Una vez ajustados los pesos, se evaluará la idoneidad de estos.
- **Evaluación sobre el algoritmo de encaminamiento.** Una vez diseñadas las funciones de coste y ajustados los parámetros, se evaluará la obtención de caminos, a fin de cerciorar que gracias a las evaluaciones anteriores, se escoge el camino óptimo en cada caso.
- **Evaluación sobre la solución.** Por último se proponen diferentes situaciones para comprobar la validez de la solución propuesta en diferentes ámbitos. En estas situaciones, a partir de datos como la latencia de paquetes o los paquetes perdidos se realizará una estimación de la escala MOS gracias a los modelos estándares. MOS significa Medium Opinion Score, y hace referencia a la percepción que tendría un cliente usando el servicio, dando una valoración subjetiva del funcionamiento de éste.

A fin de centrar este resumen en los datos de mayor relevancia, se presentan los resultados sobre la solución final.

V. RESULTADOS

En esta sección se incluyen los resultados obtenidos para cada tipo de evaluación, sin embargo, a continuación solo se muestran aquellos sobre la solución final. Gracias al emulador de redes Mininet (<http://mininet.org/>) ha sido posible emular un entorno de red donde comprobar la validez de la solución propuesta. Para dicha evaluación se proponen los siguientes escenarios.

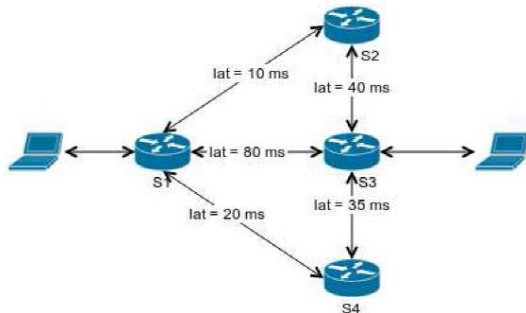


Fig. 5. Primer escenario para evaluación de la solución.

Para obtener la calidad de experiencia obtenida por un usuario, a partir de resultados objetivos (como latencia o pérdida de paquetes) se ha decidido hacer uso de la recomendación G. 107 de la ITU [5] y la recomendación G. 1070 de la ITU [6]. De este modo, a partir de datos objetivos, se obtendrán valores de la escala MOS.

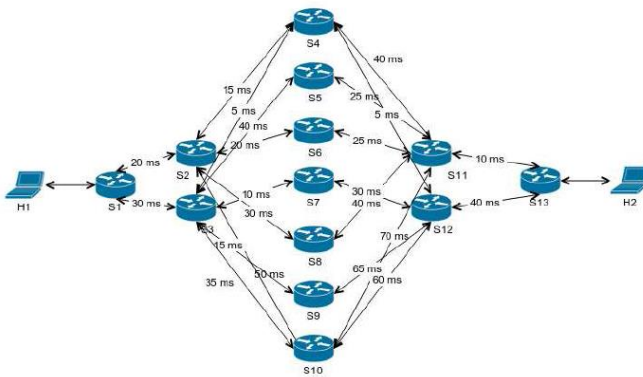


Fig. 6. Segundo escenario para la evaluación de la solución.

MOS	Calidad	Percepción de Problemas
5	Excelente	Imperceptibles
4	Buena	Algunos pero sin provocar descontento
3	Regular	Descontento aceptable
2	Mediocre	Descontento
1	Mala	Imposible usar

Tabla 1. Escala de valores MOS.

A continuación se presentan los valores medios obtenidos para las evaluaciones de audio y vídeo.

En el caso de audio se ha transmitido audio con el códec G. 711 durante 90 segundos en cada escenario y se han provocado 2 caídas de enlace durante este tiempo. Además se debe destacar que la red se encontraba en fase inicial, sin ningún tipo de memoria.

RTP audio	MOS Medio	MOS Ideal
Escenario 1	3.8	4.10
Escenario 2	3.65	4.07

Tabla 2. Valores MOS Medios para transmisión de audio.

En vídeo se ha realizado una prueba sobre el primer escenario enviando 3 tipos de vídeo diferentes, con las siguientes características.

	Vídeo 1	Vídeo 2	Vídeo 3
Duración	52 s	5 min 58 s	2 min 28 s
Resolución	400 x 226	1280 x 720	1920 x 1080
Códec	H.264	H.264	H.264
Fps	30	24	25
Tasa de bits	248 + 96 kbps	1529 + 191 kbps	2013 + 125 kbps

Tabla 3. Tabla de características de vídeo.

Los resultados obtenidos son los siguientes:

RTP vídeo	Caídas de enlace	MOS Medio	MOS Ideal
Vídeo 1	1	3.45	5
Vídeo 2	4	3.30	5
Vídeo 3	2	3.37	5

Tabla 4. Valores MOS Medios para transmisión de vídeo.

Además, a partir de los datos relacionados con la pérdida de paquetes se puede obtener el tiempo medio sin servicio en cada caso:

RTP audio	Tiempo sin servicio medio
Escenario 1	0.74 s
Escenario 2	1.17 s

Tabla 5. Tiempo medio sin servicio para transmisión de audio.

RTP vídeo	Tiempo sin servicio medio
Vídeo 1	0.48 s
Vídeo 2	0.72 s
Vídeo 3	0.58 s

Tabla 6. Tiempo medio sin servicio para transmisión de vídeo.

A partir de estos resultados, y de la realización de todo el trabajo en su conjunto, se extraen las conclusiones que se presentan en el siguiente apartado.

VI. CONCLUSIONES Y LÍNEAS FUTURAS

A. Conclusiones.

- SDN es una arquitectura de red que promete grandes ventajas sobre redes tradicionales.
- Se ha desarrollado una aplicación para el controlador OpenDayLight que explota las capacidades de las redes SDN.
- Se ha diseñado una solución capaz de realizar diferenciación de tráfico, cálculo de camino óptimo y recuperación de enlaces.
- Se ha comprobado el correcto funcionamiento de la solución diseñada a tenor de los resultados.
- Se ha optimizado la elección de caminos en función del tipo de tráfico.
- Se ha implementado recuperación de enlaces tras caída.

- Se obtienen resultados MOS aceptables de calidad de vídeo y voz.

B. Líneas futuras.

- Integrar la solución en las nuevas versiones del controlador.
- Mejorar la eficiencia y tiempos sin servicio obtenidos por la solución propuesta.
- Realizar un estudio sobre la QoE percibida por los usuarios, obviando el uso de modelos.
- Implementación de métodos de protección frente a errores de ráfaga.
- Separación de la solución en diferentes aplicaciones para aumentar la eficiencia.

AGRADECIMIENTOS

Dedicado a cada persona que me ha apoyado a sacar adelante este trabajo, y que gracias a todos ha tenido tanto éxito. Que nunca han dejado que me rinda. Como no a Manu, el apoyo de todos estos meses y en especial a ella que siempre está y siempre saca lo mejor de mí.

A mi tutor, Don Juan José Ramos Muñoz quiero agradecer en especial su gran apoyo para que este trabajo tenga la calidad que merecía. No cambies nunca Juanjo. Muchas gracias!

REFERENCIAS

- [1] C. V. Networking. (2014) "Cisco visual Networking index: Forecast and methodology", 2013-2018. Disponible: <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/whitepaperc11-481360.html>
- [2] O. N. Foundation (2015). Openflow definition by onf. Disponible: <https://www.opennetworking.org/sdn-resources/openflow>
- [3] Algoritmo dijkstra
- [4] SourceForge. (2009) Aplicación algoritmo Dijkstra para Java. Disponible: <http://jung.sourceforge.net/doc/api/edu/uci/ics/jung/algorithms/shortest-path/DijkstraShortestPath.html>
- [5] REC, I. T. U. T. G. 107. The E-model, a computational model for use in transmission planning," March, 2005.
- [6] ITU, "Opinion model for video-telephony applications," July, 2012. [Online]. Available: <http://www.igut.int/rec/T-REG-G.1070-201207-1/en>



Cristian Alfonso Prieto Sánchez (25 de Junio de 1992, Olivares (Granada)) es graduado en Ingeniería de Tecnologías de Telecomunicación con mención Telemática por la Universidad e Granada en 2015.

Soporte para comunicaciones máquina a máquina en sistemas 5G

Autor: Pilar Andrés Maldonado, e-mail: pam91@correo.ugr.es

Tutor: Pablo Ameigeiras Gutiérrez, e-mail: pamegeiras@ugr.es

Titulación: Ingeniería de Telecomunicación

Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Resumen—El incremento de las comunicaciones máquina a máquina (M2M) sobre las redes móviles impone nuevos requisitos y la aparición de nuevos desafíos en las redes, como son el crecimiento exponencial del número de M2M UEs (*User Equipment*) conectados, o características de tráfico únicas. Muchos M2M UEs realizarán pequeñas y poco frecuentes transmisiones de datos, lo que supone un desafío para las redes móviles actuales no diseñadas para este tipo de tráfico, donde la carga de señalización puede incrementarse significativamente y causar congestión en la red. El objetivo del proyecto es la propuesta y evaluación de nuevos procedimientos de control mejor adaptados a las comunicaciones M2M diseñados para una arquitectura 5G basada en *Software Defined Networking*, con respecto al esquema actual de las redes *Long Term Evolution* (LTE), centrándose para su diseño en la reducción de la señalización.

Palabras clave—M2M, LTE, Small Data Transmissions, SDT, SDN.

I. INTRODUCCIÓN

MÁQUINA a máquina es un término que puede ser usado para describir cualquier tecnología que permita a los dispositivos conectados a la red compartir información y realizar acciones sin interacción humana. Aunque hay muchos escenarios M2M, la mayoría de las comunicaciones M2M involucra una gran cantidad de dispositivos que desean comunicarse con sus aplicaciones en un corto periodo tiempo. Las características de tráfico M2M son muy diferentes de las comunicaciones entre personas (H2H) tenidas en cuenta hasta ahora para el diseño de las redes móviles. Normalmente, las comunicaciones M2M están relacionadas con muchos dispositivos de bajo consumo que solicitarán pequeñas transmisiones de datos [1], [2]. Estas nuevas características de tráfico dan lugar a limitaciones en las redes móviles para conseguir una implementación eficiente de las comunicaciones M2M.

En los primeros despliegues de servicios M2M se ha usado GPRS (*General Packet Radio Service*), ya que proporciona una gran cobertura, y el costo de los módems GPRS es lo suficientemente bajo para permitir modelos de negocio M2M viables [3]. Sin embargo, GPRS tiene algunas desventajas para M2M, que limitan la capacidad de GPRS para soportar las aplicaciones M2M, lo que ha hecho que el sector se fije en la siguiente generación de red móvil, LTE, para conseguir mayor capacidad para las comunicaciones M2M.

Actualmente, los M2M UEs se comunican con la red de la misma forma que lo hacen los H2H UEs, pero a diferencia de ellos, muchas aplicaciones M2M solo necesitan intercambiar una pequeña cantidad de datos, como por ejemplo, las aplicaciones relacionadas con telemetría. A pesar del pequeño

tamaño de estas transmisiones, activan los mismos procedimientos de control para reserva de recursos, reduciendo el soporte eficiente de los servicios M2M en la red. Un ejemplo de ello es la red convencional de LTE del 3GPP, donde se requiere la realización del procedimiento de acceso aleatorio (RA) y la configuración de los portadores radio antes de la transmisión de los datos. Si el M2M UE sólo quiere enviar un pequeño paquete IP, la carga de señalización generada es excesiva comparada con los datos a transmitir. Esta cantidad de señalización y el gran número de M2M UEs que iniciarían casi simultáneamente la solicitud podría ser una importante causa de congestión en el plano de control y un generador de cuellos de botella en los limitados recursos radio disponibles [4]. Esta situación junto con el incremento actual del tráfico de señalización H2H en las redes LTE, que crece más rápido que el tráfico de datos, generará una tormenta de señalización que podría paralizar la red.

Para superar los desafíos previstos, el grupo 3GPP ha propuesto mejoras dentro de LTE que permitan un soporte eficiente de las comunicaciones M2M, LTE para *Machine Type Communications* (LTE-M), donde se quieren proporcionar servicios M2M a bajo coste y bajo consumo energético con alta disponibilidad.

Este resumen se centrará en los aspectos y resultados más importantes del proyecto, enviados al *ACM Symposium on Applied Computing* (SAC 2016), que son: el estudio de los esquemas actuales de pequeñas transmisiones M2M, la propuesta de un nuevo procedimiento para una arquitectura de sistema 5G basada en *Software Defined Networking* (SDN), y la evaluación de las diferentes alternativas estudiadas. Tras las simulaciones, se muestra que el esquema convencional de LTE no es eficiente para los M2M UEs que soliciten pequeñas transmisiones, y que nuevas soluciones como LTE-M o el procedimiento propuesto pueden reducir significativamente la carga de señalización generada.

II. CONOCIMIENTOS PREVIOS

A. LTE

LTE fue diseñado para aplicaciones de banda ancha, nació para cubrir la necesidad de mayores velocidades de envío, menor latencia y una red más sencilla con la que operar comparada con sus antecesores. Algunos de los requisitos de LTE son [5]: velocidades de pico de 100 Mbps de bajada y 50 Mbps de subida, latencia en el plano de usuario de menos de 5ms y en el plano de control de menos de 100ms desde el

estado *idle* a *active*. LTE ha sido diseñado para soportar solo servicios de conmutación de paquetes, donde cada transmisión de datos se basa en un portador para proporcionar QoS.

Cuando un UE, registrado en la red, está inactivo porque no está usando ningún servicio, la red libera algunos de los recursos asignados al UE como los recursos radio e información relacionada, realizando el procedimiento de control *S1 release*, mostrado en la Fig. 1. Este procedimiento es utilizado por el eNB (*E-UTRAN Node B*) para liberar recursos, cambiando el UE de estado *connected* a *idle*. Cuando un UE en estado *idle* quiere enviar un paquete de datos, tiene que realizar primero el procedimiento de control *service request* para volver a activar y reasignar los recursos, en la Fig. 2 se puede ver la secuencia de mensajes del procedimiento. Por lo tanto, cada transmisión de datos desde el estado *idle* por parte del UE implica la reactivación de los portadores de datos liberados anteriormente.

El despliegue de las comunicaciones M2M sobre LTE impone ciertos desafíos en LTE, como la congestión causada por el gran número de M2M UEs conectados y sus características de tráfico. Esta congestión puede ocurrir en diferentes nodos de la red, por ejemplo en el eNB, debido al alto número de M2M UEs intentando conectar a la red casi simultáneamente, o en la parte troncal, cuando esta gran cantidad de M2M UEs se conectan a la red desde diferentes celdas, pero comparten el mismo *Mobility Management Entity* (MME).

Otro de los desafíos clave que impone M2M es la gestión eficiente de los limitados recursos radio. En LTE, un UE tiene que realizar el procedimiento de acceso aleatorio (RA) en diferentes situaciones, como por ejemplo para recibir o enviar nuevos paquetes de datos cuando el UE no está sincronizado

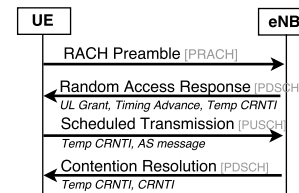


Fig. 3. Procedimiento de acceso aleatorio en LTE.

con el eNB, o no tiene recursos configurados [5]. Hay dos variantes del procedimiento RA en LTE, acceso basado en contención o no basado en contención. La variante de interés en este proyecto es el basado en contención, donde la red no tiene un preámbulo reservado para el UE, como es normal para el establecimiento de una conexión *Radio Resource Control* (RRC). El procedimiento RA basado en contención está compuesto por 4 pasos, mostrados en la Fig. 3. El procedimiento comienza con el UE enviando un preámbulo al eNB (Msg1), si el eNB recibe el preámbulo, responde con un mensaje "Random Access Response" (RAR, Msg2) donde se indican los recursos radio en los que el UE puede enviar el Msg3. Tras la recepción del Msg3, el eNB contesta al UE con el Msg4, este mensaje contiene la resolución de la contención del UE. Los mensajes 3 y 4 normalmente son usados como parte del procedimiento de establecimiento de la conexión RRC, como mensajes "RRC Connection Request" y "RRC Connection Setup", donde el UE solicita la conexión RRC y la red establece los portadores radios basándose en la causa del establecimiento, respectivamente.

B. LTE-M

Para resolver las ineficiencias de LTE con MTC (*Machine Type Communications*), el grupo 3GPP ha introducido varias soluciones para optimizar la implementación de MTC en LTE, LTE-M. La *Release 10* añadió funcionalidades para la congestión de la señalización y el control de sobrecarga. La *Release 11* se centró en el direccionamiento IP, identificadores y activación de dispositivos MTC, y la actual *Release 12* se ha centrado en la transmisión de pequeños paquetes, mejoras en la activación de dispositivos, monitorización, optimización del consumo energético del UE y funcionalidades basadas en grupos. Algunos nuevos requisitos de servicio para MTC implican una evolución en la arquitectura de red para incluir nuevas entidades funcionales con nuevos servicios, como por ejemplo, el *Services Capability Server* (SCS) o la *MTC-InterWorking Function* (MTC-IWF) [6].

C. Arquitectura 5G basada en SDN

Todos estos cambios propuestos por el 3GPP podrían ser insuficientes para una implementación eficiente de las comunicaciones M2M. Por ejemplo, la capacidad fija de la red troncal de LTE podría saturarse con una tormenta de señalización NAS (*Non Access Stratum*). En [7] se observó que la máxima capacidad de gestión de un MME era alrededor de 80.000 solicitudes en 5 min. Si millones de M2M UEs conectados reinician su operación, la capacidad del MME será un cuello de botella y los enlaces entre el *Home Subscriber Server*

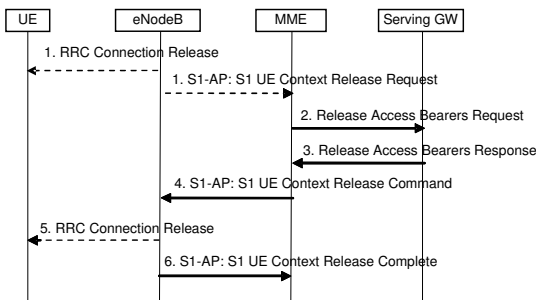


Fig. 1. Secuencia del procedimiento *S1 Release* [9].

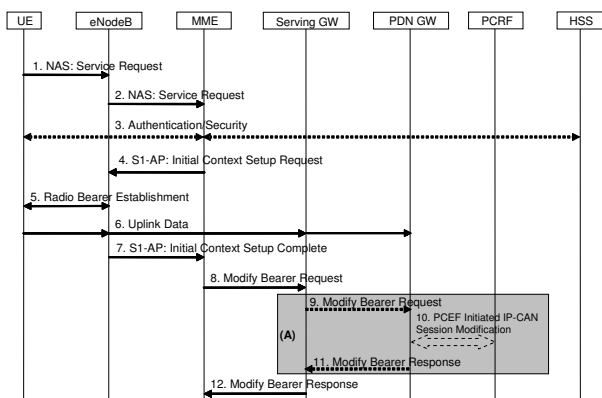


Fig. 2. Secuencia del procedimiento *Service request* [9].

(HSS) y el MME estarán también congestionados debido a la gran cantidad de mensajes de autenticación.

Para el proyecto, se ha supuesto una red 5G para superar las posibles limitaciones de LTE como la mencionada anteriormente. Se ha usado la nueva arquitectura de red explicada en [8], donde se propone una arquitectura de alto nivel para un sistema 5G basado en SDN, computación en la nube y *Network Functions Virtualisation* (NFV), con tres niveles de jerarquía de controladores SDN. NFV desacopla las funciones de la red del hardware dedicado, pudiendo ejecutarse estas funciones en software. SDN permite una nueva arquitectura de red donde el plano de control y el de datos están separados, la inteligencia de la red y el estado están lógicamente centralizados, y la infraestructura de red se abstrae de las aplicaciones finales, consiguiendo con ello la creación de nuevas redes inteligentes programables. NFV ejecuta un MME virtualizado, donde los cambios de la red solicitados son realizados por el controlador SDN.

Con esta nueva arquitectura, en este proyecto se han rediseñado los procedimientos de control principales de LTE [9], y se ha propuesto uno nuevo para reducir la señalización y mejorar la eficiencia de la red en el envío de pequeñas transmisiones M2M. Para ello, se propone la eliminación del uso de portadores en 5G para la gestión de las diferentes QoS, eliminando así la distinción de los diferentes flujos según su QoS y el usuario que los generó, y la unificación de los estratos de seguridad NAS y AS (*Access Stratum*), para eliminar redundancia. Los procedimientos rediseñados no incluidos en este resumen, pero que se pueden ver en la memoria del proyecto son: *Attach, SI Release, UE Triggered Service Request, Tracking Area Update y Detach*.

III. PROCEDIMIENTOS DE SEÑALIZACIÓN REDUCIDOS PARA M2M

La inclusión de las comunicaciones M2M, y en especial para este proyecto las pequeñas transmisiones de datos, pueden dar lugar a una sobrecarga de señalización si no se gestionan eficientemente [10], [11]. En esta sección se explicarán los dos principales trabajos en los que se ha basado este proyecto para el diseño del nuevo procedimiento de control propuesto para M2M.

A. Procedimiento SDT del 3GPP

Tras el reconocimiento por parte del 3GPP de la característica de pequeñas transmisiones en las comunicaciones M2M (*Small Data Transmissions, SDT*) [1]. El 3GPP ha propuesto varias soluciones para optimizar el envío o recepción de estas pequeñas transmisiones [12]. Dentro de las soluciones propuestas, este proyecto se basa en una de ellas donde se propone un nuevo procedimiento para optimizar la transferencia de un paquete IP y su respuesta, al que llamaremos procedimiento SDT. Este nuevo procedimiento SDT optimiza la secuencia de mensajes cuando el M2M UE está en estado *idle*, usando para ello el contexto de seguridad NAS preestablecido para enviar los datos como señalización NAS [12]. Cuando el M2M UE quiere enviar un paquete IP, solicita el establecimiento de una conexión RRC con la causa de establecimiento fijada a "*low priority small data*" o "*small data*". Estos valores en la causa del establecimiento

RRC permiten al eNB detectar que es un procedimiento de señalización corto. Tras completar el procedimiento RA y el establecimiento de la conexión RRC, el M2M UE envía un mensaje NAS con el paquete de datos IP al MME, en la Fig. 4 se puede ver el procedimiento completo.

Para la transferencia del pequeño paquete de datos IP, el procedimiento SDT hace uso del plano de control para evitar el establecimiento de la seguridad RRC y los portadores de datos. Esta solución incluye el paquete de datos en el mensaje inicial NAS y la transmisión recae en la capa RRC, sin ninguna confirmación de capas superiores a la RRC. Para la evaluación de este procedimiento, se ha asumido que el MME libera inmediatamente la conexión RRC después de completar la transmisión del paquete IP (denominada como "alt. A" en la Fig. 4).

B. Hybrid Random Access and Data Transmission Protocol

Para reducir la señalización excesiva en las comunicaciones M2M, en [13] se propone un nuevo diseño de un protocolo para el acceso aleatorio y la transmisión de datos. La solución permite enviar paquetes de datos a un M2M UE en estado *idle* después de la transmisión del preámbulo en el procedimiento RA, por lo tanto, no es necesario el establecimiento de una conexión RRC. La solución combina una asignación dinámica de canales de datos M2M en los recursos radio por la estación base, y un esquema de clases de restricción para reducir la sobrecarga en la red de acceso radio. El procedimiento tiene 6 pasos, resumidos en:

- 1) Planificación PRACH: La estación base estima el número de M2M UEs activos para decidir la posible configuración.
- 2) Restricción de acceso: Un M2M UE activo participa en el mecanismo de clases de restricción para iniciar el procedimiento RA.
- 3) Transmisión del preámbulo: Como en el procedimiento RA normal de LTE, el M2M UE transmite un preámbulo a la estación base.
- 4) Planificación de canales de datos: En este paso la estación base detecta todos los preámbulos transmitidos, planifica los recursos para los M2M UEs e informa de los resultados de la planificación a los M2M UEs.

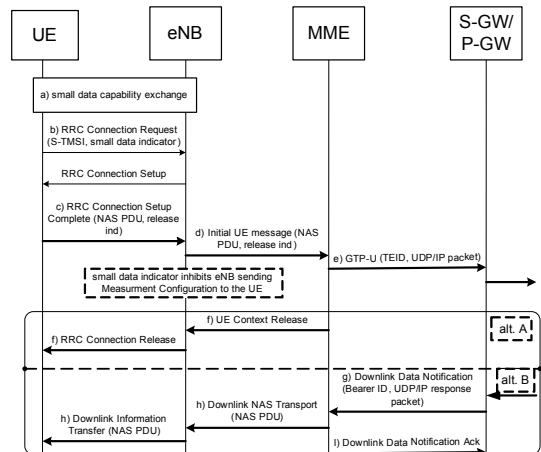


Fig. 4. Procedimiento SDT [12].

- 5) Transmisión de los datos: Tras la recepción de la respuesta de la estación base, cada M2M UE envía su paquete de datos sobre los recursos radio asignados por la estación base.
- 6) Confirmación: Tras descryptar todos los paquetes de datos recibidos, la estación base envía un mensaje de confirmación a los M2M UEs.

C. Procedimiento MO IP packet

El nuevo procedimiento de control propuesto en este proyecto, llamado *Mobile Originated IP packet (MO IP packet)*, está orientado a la transferencia de un pequeño paquete IP a la red cuando el M2M UE está en estado *idle*. El tipo de tráfico M2M que se beneficiaría de este procedimiento sería el que realiza transmisiones poco frecuentes y sin requisitos de QoS. La idea se origina como una combinación de las dos soluciones comentadas anteriormente para pequeñas transmisiones M2M. En el diseño del procedimiento, se supone el mismo diseño RAN que en LTE, y la nueva arquitectura 5G basada en SDN explicada. El procedimiento tiene 6 pasos, donde los 4 primeros pasos del procedimiento se basan en la mejora de acceso propuesta en la solución *Hybrid Random Access and Data Transmission Protocol* [13], donde los recursos M2M y H2H se gestionan de forma separada, y el eNB calcula la cantidad de recursos M2M a asignar y el parámetro de restricción de acceso. El procedimiento MO IP packet no necesita el establecimiento de una conexión RRC, tras el comienzo del procedimiento RA y la asignación de recursos del eNB en el segundo mensaje, el M2M UE puede enviar su pequeño paquete IP en el tercer mensaje del procedimiento RA al MME virtualizado, evitando así el establecimiento de los portadores radio. Cuando el MME virtualizado recibe el paquete de datos, autentica al M2M UE descryptando el paquete con el contexto de seguridad almacenado de ese M2M UE durante el procedimiento de *attach* [9]. Tras la correcta autenticación, el MME virtualizado envía el paquete de datos a la pasarela, como se muestra en la Fig. 5.

Sin el establecimiento de los portadores, se reducen los mensajes de señalización necesarios en la red para la conexión y reconfiguración RRC, la solicitud del eNB del contexto del UE, el establecimiento de la seguridad AS y la liberación de recursos cuando el usuario pase a estado *idle*. Como en el procedimiento SDT, se supone que no hay transmisión de

confirmaciones por encima de la capa RRC, la transferencia de datos recae en la capa RRC del plano de control, y que el contexto de seguridad NAS se ha mantenido almacenado en el M2M UE y en el MME virtualizado para la autenticación.

IV. EVALUACIÓN

En esta sección se compara la carga de señalización generada en tres esquemas diferentes: LTE convencional, LTE-M haciendo uso del procedimiento SDT, y 5G basado en la arquitectura SDN haciendo uso del procedimiento MO IP packet propuesto. El indicador que se usa para la evaluación de estos tres esquemas es la media del número de paquetes de señalización generados por segundo.

A. Modelo de simulación

La implementación del simulador se ha realizado en el simulador de eventos NS-3. Para comparar la carga de señalización generada en cada esquema, se ha simulado un sistema con H2H y M2M UEs. Los modelos de tráfico H2H incluidos son web, VoIP (*Voice-over-IP*) [14] y NRTV (*Near Real-Time Video*) [15], [16]. Para el tráfico M2M, se ha simulado el descrito en [17], donde se hace uso de cadenas de Markov coordinadas moduladas como procesos de Poisson para modelar transmisiones coordinadas o no coordinadas de pequeños informes. En la simulación se ha usado una versión simplificada de este modelo sin correlación espacial. Se ha supuesto el escenario M2M de gestión de flotas [18], donde se definen 3 nuevos estados para los M2M UEs (*state0*, *state1* y *alarm*). Las matrices de transición propuestas en [18] tienen un problema de convergencia debido al gran número de M2M UEs que cambian su estado a *alarm* y después a *state0*. Tras ese cambio masivo, se necesita un gran periodo de tiempo para converger a un estado estacionario en el sistema. Para resolverlo, se ha modificado la probabilidad de cambio al estado *alarm* en la matriz de transición del tráfico coordinado, el resto de los valores del modelo permanecen igual que los originales definidos en [18]. Además, se han programado alarmas periódicas en la simulación para modelar el tráfico sincronizado M2M, el cual genera picos de señalización. La Tabla I resume los parámetros más importantes del modelo de tráfico M2M. Se han considerado tres escenarios con 800 H2H UEs, y diferente número de M2M UEs: Escenario 1) 800 M2M UEs, Escenario 2) 2400 M2M UEs, Escenario 3) 4800 M2M UEs.

En la simulación, cada UE tiene un estado en la red (*connected*/*idle*/*detached*). Cuando un UE quiere enviar un paquete de datos, el simulador verificará el estado del UE con la red, si está en estado *connected*, el UE podrá enviar el paquete de datos, pero si el UE está en estado *idle* o *detached*, necesitará solicitar procedimientos de control antes de la transmisión del paquete para obtener suficientes recursos, como se muestra en la Fig. 6. Para un M2M UE en estado *idle*, los procedimientos de control realizados para el envío del pequeño paquete de datos son diferentes en cada esquema, para LTE, el M2M UE realizará un *service request*, en LTE-M, un procedimiento SDT, y para el esquema 5G basado en SDN, el procedimiento MO IP packet propuesto. En la Tabla II se resumen los paquetes intercambiados en la red para realizar un procedimiento de control en los tres esquemas supuestos.

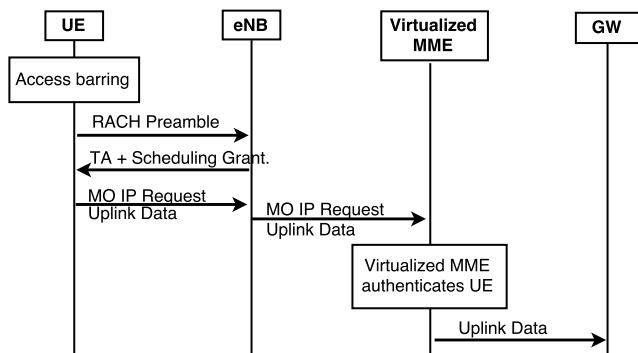


Fig. 5. Procedimiento MO IP packet.

Para poder evaluarlos, el simulador guarda las estadísticas sobre cada procedimiento de control solicitado, y calcula los mensajes de señalización necesarios para realizarlo. La duración de la simulación está fijada a 30000s, pero los primeros 5000s de la simulación se eliminan en el análisis de los resultados para evitar el comportamiento transitorio del sistema mientras los H2H y M2M UEs se conectan a la red. La resolución está configurada a 1s. La Tabla III resume los parámetros generales de la simulación.

B. Resultados

Para comparar los tres esquemas supuestos se ha analizado la media de los paquetes de señalización generados tanto por H2H UEs como por M2M UEs en cada escenario y esquema supuesto. El tráfico de señalización dominante es el generado por el tráfico M2M. Cuando se incrementa el número de M2M UEs, las pequeñas transmisiones que realizarán se volverán más comunes en el sistema, y la reducción de señalización conseguida será mayor.

Tabla I
PRINCIPALES PARÁMETROS DEL MODELO DE TRÁFICO M2M

Parámetro	Valor
Tamaño paquete	100 B
State0 Bitrate	0.15 B/s
State1 Bitrate	6.5 B/s
Alarm Bitrate	24 B/s
Periodo de alarmas	1800 s
Matrices de transición	
Tráfico no coordinado	0.9999325 0.000147 0.39
	0.0000675 0.999853 0
	0 0 0.61
Tráfico coordinado	0.93 0 0
	0 0.93 0
	0.07 0.07 1

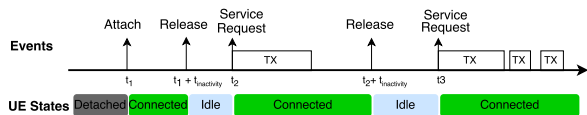


Fig. 6. Ejemplo funcionamiento simulador.

Tabla II
RESUMEN PAQUETES INTERCAMBIADOS EN CADA PROCEDIMIENTO DE CONTROL EN AL PARTE RADIO (R) Y TRONCAL (C) DE LA RED.

	LTE		LTE-M		5G SDN	
	R	C	R	C	R	C
S1 Release	1	5	1	5	1	3
Service Request	9	5	9	5	7	5
SDT			6	3		
MO IP packet					3	2

Tabla III
PARÁMETROS DE LA SIMULACIÓN

Parámetro	Valor
Bitrate LTE	10 Mbps
Bitrate LTE-M	10 Mbps
Bitrate 5G SDN	300 Mbps
Tiempo de conexión	Exponencial (media 1200s)
Tiempo de reinicio de sesión H2H	Exponencial (media 1200s)
Temporizador de inactividad	10 s

En la Fig. 7 se muestra la media de los paquetes de señalización por segundo generados en la interfaz radio por el tráfico H2H y M2M en los tres escenarios y esquemas evaluados. El esquema de 5G basado en SDN es el que consigue mejores resultados, ya que hace uso del procedimiento MO IP packet que sólo necesita 3 paquetes de señalización en la interfaz radio para transferir el paquete IP del M2M UE, en vez de los 9 paquetes necesarios en el service request de LTE, o los 6 paquetes del procedimiento SDT en LTE-M. Para el esquema de LTE-M, incluso se obtienen resultados algo peores que en el caso del esquema LTE donde se utilizan procedimientos no orientados a pequeñas transmisiones M2M. Esto es debido al modelo de tráfico M2M supuesto, donde aproximadamente la mitad de las transmisiones realizadas por los M2M UEs se realizan en un tiempo menor al periodo de inactividad fijado en la red. Para LTE, estas transmisiones no necesitarán la solicitud de ningún procedimiento de control para enviar el paquete de datos porque el M2M UE sigue conectado a la red, por lo que tiene asignados sus recursos y sus portadores activos, pero para LTE-M y 5G basado en SDN, donde se hace uso de los procedimientos SDT y MO IP packet respectivamente, cada pequeña transmisión necesita la realización del procedimiento para obtener los recursos necesarios para el envío de los datos.

En la parte troncal de la red, los resultados muestran una mayor reducción de la señalización, Fig. 8. Esta mejora es debida a la optimización en la asignación de recursos conseguida con los procedimientos SDT y MO IP packet, donde sólo unas pocas entidades de la red toman parte del procedimiento para la transmisión y no es necesario el establecimiento de portadores de datos, reduciendo así el número de paquetes de señalización intercambiados. Para resumir, en la Tabla IV se muestra la reducción total conseguida en la parte radio y troncal de la red con el tráfico H2H y M2M en LTE-M y 5G basado en SDN en comparación con el esquema LTE. En los eventos de alarma, la reducción total alcanzada es un poco mayor que la conseguida en momentos de tráfico normal debido al gran número de M2M UEs en estado idle que cambian su estado a alarm y quieren enviar un paquete a la red.

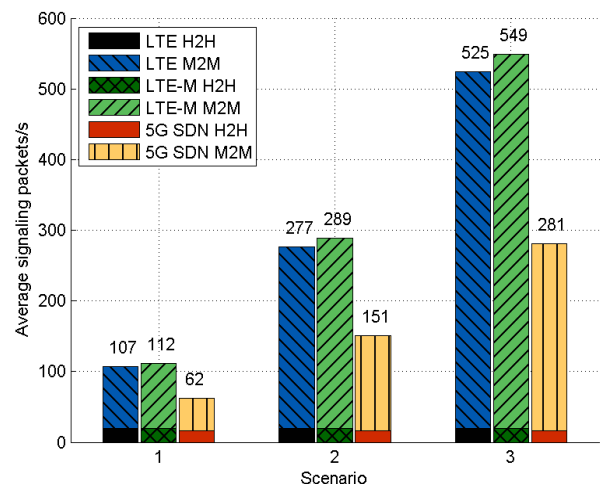


Fig. 7. Comparación de la media de los paquetes de señalización por segundo generados en la parte radio de la red.

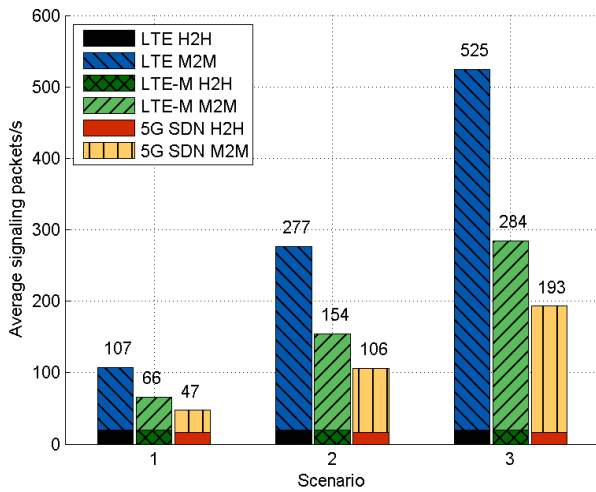


Fig. 8. Comparación de la media de los paquetes de señalización por segundo generados en la parte troncal de la red.

Tabla IV
REDUCCIÓN CONSEGUIDA DE LA MEDIA DE LOS PAQUETES/S DE SEÑALIZACIÓN GENERADOS POR EL TRÁFICO TOTAL H2H Y M2M EN COMPARACIÓN A LTE.

	LTE-M		5G basado en SDN	
	Radio	Core	Radio	Core
Escenario 1	- 4.67 %	38.31 %	42.05 %	56.07 %
Escenario 2	- 4.33 %	44.40 %	45.48 %	61.73 %
Escenario 3	- 4.57 %	45.90 %	46.47 %	63.23 %

V. CONCLUSIONES

En este resumen se han presentado los principales resultados de este proyecto donde se ha evaluado el impacto de señalización de las comunicaciones M2M en las redes móviles. Dado que el tráfico M2M puede afectar al correcto funcionamiento de la red, especialmente la congestión en el plano de control, se han revisado algunas propuestas orientadas a pequeñas transmisiones M2M y se han comparado con el esquema actual de LTE. Para modelar el tráfico M2M, se ha asumido una aplicación M2M de gestión de flotas que usa la transmisión de pequeños informes en sus comunicaciones.

En este proyecto se han rediseñado los principales procedimientos de LTE, y se ha propuesto un nuevo procedimiento de control, MO IP *packet*, que simplifica la transmisión de un paquete de datos usando el procedimiento RA para obtener los recursos radio suficientes para enviar el paquete en el tercer mensaje del procedimiento RA, evitando así la necesidad de una conexión RRC.

Los resultados obtenidos incluyen una evaluación de tres esquemas de transmisión para pequeñas transferencias de datos M2M: LTE con los procedimientos *service request* y *SI release*, LTE-M con el procedimiento SDT, y la arquitectura 5G basada en SDN con el procedimiento MO IP *packet* propuesto. La evaluación muestra que el esquema actual de LTE no es eficiente para soportar despliegues M2M donde los M2M UEs utilicen pequeñas transmisiones poco frecuentes, ya que el uso de los procedimientos *service request* y *SI release* cuando el M2M UE está en estado *idle* genera una cantidad de señalización excesiva. Nuevos mecanismos optimizados para las comunicaciones M2M consiguen mayor eficiencia en la red, pudiendo reducir significativamente la

carga de señalización generada por las comunicaciones M2M.

REFERENCIAS

- [1] Service requirements for Machine-Type Communications (MTC); Stage 1, 3GPP TS 22.368, 2014.
- [2] System improvements for Machine-Type Communications (MTC); Release 11, 3GPP TR 23.888, 2012.
- [3] J. Poncela, J.M. Moreno, M. Aamir, "Analysis of M2M capabilities in 4G", *4th Int. Conf. on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE)*, Aalborg, 2014, pp. 1-5.
- [4] N. Zhang, G. Kang, J. Wang, Y. Guo, F. Labeau, "Resource Allocation in a New Random Access for M2M Communications", *IEEE Commun. Lett.*, vol. 19, no. 5, pp. 843-846, 2015.
- [5] C. Cox, *An introduction to LTE: LTE, LTE-Advance, SAE, VoLTE, and 4G Mobile Communications*, 2^a ed, Wiley, 2014.
- [6] Architecture enhancements to facilitate communications with packet data networks and applications; Release 12, 3GPP TS 23.682, 2014.
- [7] Z. Zhou, Z. Zhao, R. Li, Y. Zhou, T. Chen, Z. Niu, H. Zhang, "Toward 5G: When Explosive Bursts Meet Soft Cloud", *IEEE Networks*, vol. 28, no. 6, pp. 12-17, 2014.
- [8] P. Ameigeiras, J.J. Ramos-Munoz, L. Schumacher, J. Prados-Garzon, J. Navarro-Ortiz, J.M. Lopez-Soler, "Link-level access cloud architecture design based on SDN for 5G networks", *IEEE Networks*, vol. 29, no. 2, pp. 24-31, 2015.
- [9] LTE; General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access; Release 12, 3GPP TS 23.401, 2014.
- [10] S. Andreev, A. Larmo, M. Gerasimenko, V. Petrov, O. Galinina, T. Tirronen, J. Torsner, Y. Koucheryavy, "Efficient small data access for machine-type communications in LTE", *IEEE Int. Conf. on Communications (ICC)*, Budapest, 2013, pp. 3569-3574.
- [11] A.G. Gotsis, A.S. Lioumpas, A. Alexiou, "M2M Scheduling over LTE: Challenges and New Perspectives", *IEEE Veh. Technol. Mag.*, vol. 7, no. 3, pp. 34-39, 2012.
- [12] Study on Machine-Type Communications (MTC) and other mobile data applications communications enhancements; Release 12, 3GPP TR 23.887, 2013.
- [13] D.T. Wiriaatmadja, K. W. Choi, "Hybrid Random Access and Data Transmission Protocol for Machine-to-Machine Communications in Cellular Networks", *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 33-46, 2015.
- [14] Next Generation Mobile Networks Radio Access Performance Evaluation Methodology, NGMN Alliance, May 2015, <https://www.ngmn.org/publications/all-downloads/article/ngmn-radio-access-performance-evaluation-methodology.html>
- [15] P. Ameigeiras, J.J. Ramos-Munoz, J. Navarro-Ortiz, J.M. Lopez-Soler, "Analysis and modeling of YouTube traffic", *Transactions on Emerging Telecommunications Technologies*, John Wiley & Sons, 2012, vol. 23, no. 4, pp. 360-377.
- [16] J.J. Ramos-Munoz, J. Prados-Garzon, P. Ameigeiras, J. Navarro-Ortiz, J.M. Lopez-Soler, "Characteristics of mobile Youtube traffic", *IEEE Wireless Commun.*, vol. 21, no. 1, pp. 18-25, 2014.
- [17] M. Laner, P. Svoboda, N. Nikaen, M Rupp, "Traffic Models for Machine Type Communications", *Proc. of the Tenth Int. Symp. on Wireless Commun. Systems (ISWCS)*, 2013, pp. 1-5.
- [18] C. Anton-Haro, M. Dohler, "Machine-to-Machine (M2M) Communications: Architecture, Performance and Applications", Woodhead Publishing, 2015.

Análisis de *malware* en *Smart TV*: ataques y defensas

Autor: Jose López Arredondo; e-mail: joselopar@gmail.com

Tutor: Pedro García Teodoro; e-mail: pgteodor@ugr.es

Titulación: Grado en Ingeniería de Tecnologías de Telecomunicación

Departamento de Teoría de la Señal, Telemática y Comunicaciones

Universidad de Granada

Resumen— Recientemente se ha reportado la ejecución de un ataque de denegación de servicio desde un frigorífico, evidenciándose así que la introducción de este tipo de dispositivos domésticos inteligentes constituye una nueva amenaza de seguridad para nuestra vida cotidiana, donde cada vez resulta más frecuente la compartición de información y realización de transacciones personales y económico-comerciales por la red. Tomando como base el creciente empleo de televisores inteligentes en los hogares, este trabajo aborda el estudio de *malware* para este tipo de dispositivos y analiza el posible desarrollo de mecanismos de defensa.

Para la realización se hace uso del SDK y del emulador provistos por Samsung, uno de los mayores fabricantes del mundo en lo que a televisores inteligentes se refiere, desarrollando una aplicación capaz de evidenciar los ataques de seguridad que uno de estos dispositivos puede sufrir.

Palabras clave—*Malware*, privacidad, *Smart TV*, seguridad

I. INTRODUCCIÓN

El desarrollo de dispositivos inteligentes es una realidad cada vez más creciente, algo demostrable a través de productos como los *smartphones*, las *tablets* o incluso los relojes inteligentes; lo último en lo que a moda se refiere. No obstante, uno de los aspectos que más ha avanzado en este terreno ha sido el equipamiento del hogar, lugar donde la tecnología tiene cada vez más protagonismo permitiéndonos seguir conectados al mundo digital sin perder las funciones tradicionales de cada uno de sus elementos.

Sin duda alguna, el máximo ejemplo de este desarrollo del hogar ha sido el *Smart TV* o televisor inteligente, un producto que además de ofrecer la funcionalidad tradicional, permite conectarse a Internet y disfrutar de contenidos digitales a la carta, además de hacer uso del dispositivo como si de un ordenador o *smartphone* se tratara. Poco a poco, los televisores inteligentes están tomando más protagonismo en los hogares españoles y, en general, en los hogares de todo el mundo. Pero esto, obviamente, también entraña peligros y es que los *Smart TV* se han convertido en una nueva puerta para todo tipo de *malware* capaz de, por ejemplo, espiar las costumbres de los usuarios y, claro está, obtener datos sensibles.

II. ESTUDIO DEL ESTADO DEL ARTE

Los televisores inteligentes, también conocidos como *Smart TV*, permiten disfrutar de un televisor convencional a la vez

que añaden ciertas funciones inteligentes como pueden ser la conexión a Internet, aplicaciones desarrolladas especialmente para el sistema operativo del propio electrodoméstico y la llamada televisión a la carta.

A. Dispositivos de televisión conectada

Desde hace años, los fabricantes han dedicado sus esfuerzos en intentar aumentar la funcionalidad de estos televisores, dando lugar a una gran base de dispositivos de televisión conectada como pueden ser *Chromecast* [1], un dispositivo USB creado por Google que permite realizar *streaming* de contenidos multimedia desde un *smartphone*, *tablet* u ordenador hasta un televisor, los *sticks* con Android, que ayudan a disfrutar del sistema operativo móvil por excelencia en los televisores, o los propios *Smart TV*. En la Tabla I puede verse un cuadro resumen con las diferentes soluciones estudiadas, las más importantes en la actualidad.

Como se puede intuir, el futuro de los dispositivos de televisión conectada es bastante alentador y se espera, según varios estudios de consultoras como *Strategy Analytics* [2], que se alcancen los 2.000 millones de unidades en el año

TABLA I
COMPARACIÓN DE LOS DISPOSITIVOS *SMART TV* ACTUALES

Dispositivo	Ventajas	Inconvenientes
<i>Televisores inteligentes</i>	<ul style="list-style-type: none">Integración absolutaVariedad de solucionesPosibilidad de conexión de más dispositivos	<ul style="list-style-type: none">PrecioDifícilmente actualizables
<i>Android TV</i>	<ul style="list-style-type: none">Integración absoluta (en <i>Smart TV</i>)Movilidad (<ul style="list-style-type: none">Precio (en <i>Smart TV</i>)Adopción lenta
<i>Android sticks</i>	<ul style="list-style-type: none">PrecioMovilidadSolución sencilla	<ul style="list-style-type: none">Sistema operativo no adaptado a televisores
<i>Apple TV</i>	<ul style="list-style-type: none">Posibilidad de conexión de más dispositivos	<ul style="list-style-type: none">Necesidad de un dispositivo iOS
<i>Chromecast</i>	<ul style="list-style-type: none">PrecioMovilidadCompatible con la mayoría de S.O. actuales	<ul style="list-style-type: none">Sin posibilidad de conexión con otros dispositivosAdopción lenta

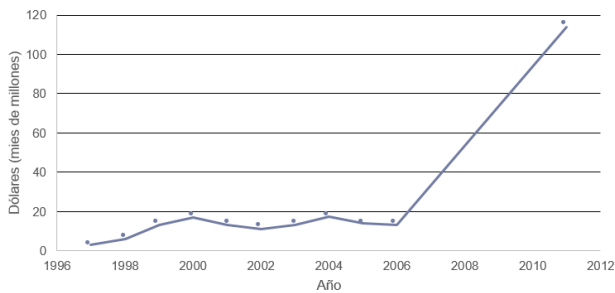


Fig 1. Impacto económico directo mundial del *malware* entre 1997 y 2011

2018, cuadruplicando los 500 millones actuales (estudio que no incluye productos como videoconsolas o reproductores Blu-ray capaces de hacer inteligente un televisor).

B. *Malware* y Smart TV

Por otra parte, prácticamente todos los dispositivos conectados a Internet conllevan el riesgo de ser afectados por software malicioso, algo en lo que también se debe tener en cuenta a los *Smart TV*. Concretamente, el *malware* está presente desde 1982 y, poco a poco, ha ido aumentando su presencia, existiendo en estos momentos más de 10 de tipos entre los que se encuentran virus, troyanos, *botnets*, *spam*... Todo ello ha conllevado un impacto socioeconómico muy grave en todo el mundo, algo que puede comprobarse en el año 2000, momento en que el virus I love you causó más de 2.000 millones de dólares en pérdidas a través del envío masivo de *spam* haciendo uso de ingeniería social, convenciendo al usuario de que tenía una carta de un ser querido y propagando a su vez el *malware* a sus contactos. Todo ello, a lo largo de los años, ha mermado la confianza de los usuarios a la hora de realizar compras y transacciones por Internet. En la Figura 1 puede observarse cómo ha evolucionado el impacto económico del *malware* en todo el mundo desde 1997 a 2011 [3].

En este aspecto, los *Smart TV* también han sido afectados ya que, por ejemplo, a través del acceso *root* pueden ejecutarse comandos en el televisor o grabar imágenes sin el consentimiento del usuario, problemas que comenzaron en el año 2012 [4] con el descubrimiento de la primera vulnerabilidad en un televisor inteligente de la marca Samsung. Poco a poco, estos dispositivos han sido atacados a través de distintos vectores, incluyendo las actualizaciones lanzadas por los fabricantes e incluso ciertas modificaciones del *firmware* no oficiales. Ahora bien, ¿es tan sencillo infectar un *Smart TV*? Esto es lo que se desea responder en este proyecto tal y como se muestra en el siguiente apartado.

III. ANÁLISIS DE OBJETIVOS

Nuestro estudio se basará en dos líneas de investigación y desarrollo con el fin de conocer si es tan sencillo infectar con *malware* un televisor inteligente. Por una parte, se convencerá a los usuarios mediante ingeniería social para que instalen en sus dispositivos una aplicación con fines maliciosos mientras que, por otro lado, se desarrollará un sistema de defensa capaz de evitar problemas a los usuarios. Así pues, los objetivos finales principales y secundarios del proyecto son:

- Se debe estudiar el *malware* que puede afectar a estos dispositivos para, posteriormente, desarrollar una aplicación que ponga de manifiesto los riesgos de seguridad de un *Smart TV*.
- Se debe estudiar la posibilidad de desarrollar un sistema de seguridad *anti-malware* para televisores inteligentes.
- La aplicación desarrollada debe ofrecer una interfaz de usuario sencilla, así como de gran usabilidad. Además, debe llamar la atención del usuario para que, en caso de distribución, consiga ser un éxito.
- Debe tratarse de una aplicación auto-actualizable de forma que se puedan añadir funciones cada cierto tiempo conforme el desarrollo avance.
- La aplicación debe comunicarse con un servidor fuera de la red local, es decir, abandonar el dominio *localhost*.
- Las aplicaciones deben ser instaladas en un televisor inteligente para demostrar el funcionamiento de la aplicación en un entorno real.

IV. DISEÑO Y DESARROLLO SDK

El primer paso ha sido investigar qué televisores inteligentes eran los más utilizados de forma que nuestro público objetivo fuera el mayor posible. Así pues, se descubrió que los televisores Samsung conseguían prácticamente un cuarto de cuota de mercado al año [5] y, además, su entorno de desarrollo de software es uno de los más completos como se comprobará posteriormente, por lo que las posibilidades eran mayores.

Por otra parte, teniendo en cuenta que más del 50% de los usuarios utilizan su *Smart TV* para ver vídeos bajo demanda según varias encuestas [6] y, con el fin de obtener el máximo rendimiento de la aplicación llamando la atención de los usuarios, finalmente se ideó una solución basada en el *streaming* de vídeo, concretamente en el ofrecimiento de películas de forma gratuita o en su defecto, sus *trailers*. A partir de este momento, se comenzó a crear la aplicación bautizada como *PelisSeriesGratis*.

Antes de comenzar con el diseño y el desarrollo propio del proyecto, hubo que aprender cómo funciona el *Software Development Kit (SDK)* de Samsung, las distintas API que ofrece y el funcionamiento general de estas. Todo ello se encuentra disponible en la página web de Samsung [7], lugar donde también se aloja una guía de desarrollo con toda la información –se encuentra dividida en dos partes: Topic, con diversos consejos sobre cómo programar y cómo funciona la guía, y API, donde se encuentran todos los métodos y funciones disponibles que se pueden utilizar– y un emulador que también se instalará en el ordenador para probar el funcionamiento correcto de la aplicación antes de llevarla a un televisor real.

Tras algunos desarrollos y análisis, la idea de un mecanismo de defensa se descartó debido a las exigencias temporales de desarrollo, por lo que se decidió crear una aplicación que pusiera de manifiesto las vulnerabilidades de un televisor inteligente. El fin malicioso principal sería, en principio, obtener la cuenta de Google del usuario haciendo uso de ingeniería social, lo que daría acceso a cuentas de correo electrónico, dispositivos móviles y mucho más.

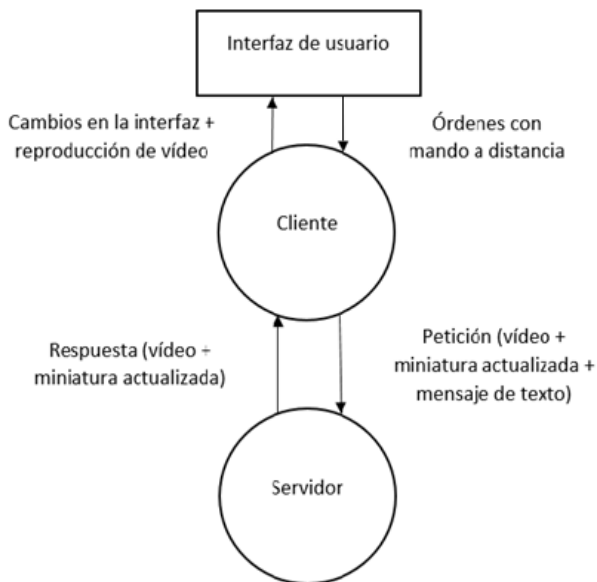


Fig 2. Diagrama cliente servidor con sus respectivos mensajes junto a la interfaz de usuario

Esto demostraría que la ingeniería social funciona, que se convence al usuario y que este es, sin duda, uno de los mayores riesgos a la hora de mantener los datos privados seguros.

No obstante, obtener la cuenta de Google de un usuario acabaría siendo un objetivo ilegal que no se desea realizar puesto que el único interés es conocer el peligro que entraña la base de usuarios de *Smart TV* en sus hogares. Por tanto, sencillamente se comprobará si existen usuarios que instalan la aplicación –en el caso de que fuera realmente peligrosa, una persona malintencionada podría hacerse con las credenciales del usuario y realizar cualquier acción como las que se nombraron anteriormente–. En definitiva, un *malware* benigno que a través de ingeniería social convence al usuario de que instale la *app* para posteriormente recibir un mensaje en un servidor creado a tal efecto indicando que un nuevo usuario o víctima ha instalado y utilizado la aplicación.

El diagrama de la Figura 2 muestra el funcionamiento básico del desarrollo completo, que contaría con un cliente y un servidor. El primero muestra una interfaz con la que el propio usuario mandaría órdenes a la aplicación a través del mando a distancia, respondiendo a su vez con cambios en la interfaz y con la reproducción de vídeos. Esto se logra gracias al envío a un servidor de una petición HTTP GET con el vídeo elegido, la miniatura correspondiente del vídeo y el mensaje de texto que se ha indicado en el párrafo anterior, mientras que el servidor responde con el vídeo y, claro está, la miniatura actualizada.

V. IMPLEMENTACIÓN

A. Implementación de la interfaz

PelisSeriesGratis consta de tres secciones principales: Películas, Series y *Login*, siendo esta última la pantalla donde el usuario podría introducir sus credenciales para obtener las supuestas mejoras que conlleva la aplicación –momento en el que se hace uso de la ingeniería social–. Cada una de las secciones anteriores cuenta con dos vídeos en formato MP4 alojados en un servidor, que es el mismo que recibe el

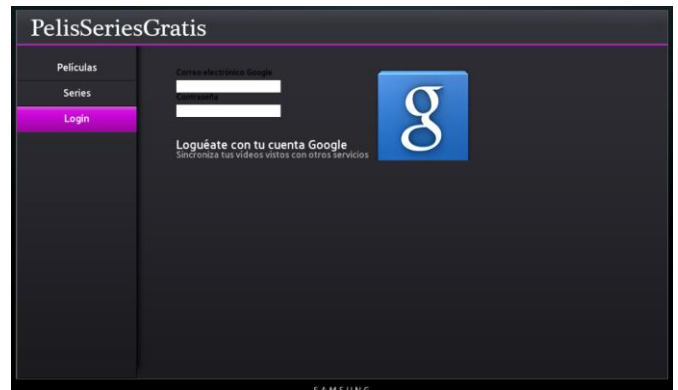


Fig 3. Interfaz de la sección *Login*



Fig 4. Escenas Categoría y Contenido con sus respectivos focos de color rosa

mensaje de instalación por parte del usuario. De forma interna, la *app* tiene varias escenas, aunque la más importante es “escena”, que integra las secciones “categoría”, “contenido” y “login”. A cada una de ellas se le asigna un foco para cambiar el estilo (por ejemplo, cuando una de las categorías está seleccionada, se marca de color rosa, mientras que si se trata de uno de los contenidos, aparecerá un recuadro rosa alrededor de la miniatura) y también un ancla, de forma que cuando un usuario se encuentre en cada una de las escenas, las teclas capaces de desencadenar un evento sean diferentes. Con el fin de que se entienda mejor, la Figura 3 muestra una captura de la sección *Login*, mientras que la Figura 4 muestra una captura de la escena principal editada con comentarios para conocer qué son exactamente los focos y las distintas subsecciones como “categoría” –que contiene Películas, Series o *Login*– y “contenido” –las distintas películas o series ofrecidas–.

La aplicación permite reproducir varios vídeos a modo de películas y series, marcando como “visto” cada uno de ellos mediante una imagen atenuada descargada del servidor (miniatura) a través de una petición HTTP GET. Esto se consigue gracias a dos variables llamadas *index*, que es la “categoría” (películas o series, ya que *login* no ofrece ningún vídeo) e *index_contenido*, un índice que indica el vídeo elegido de la respectiva categoría. Ambas variables definen el vídeo exacto que se ha reproducido ya que el ID (identificador) coincide con el nombre del propio vídeo en el servidor. Para simplificarlo, se ha añadido la Tabla 2 en la que puede observarse la correspondencia entre los ID y los distintos vídeos.

TABLA II
IDENTIFICADORES (ID) ASOCIADOS CON LAS CATEGORÍAS, EL CONTENIDO Y SUS ÍNDICES

ID	Categoría	Index	Contenido	Index_contenido
0_0	Películas	0	Primer vídeo	0
0_1	Películas	0	Segundo vídeo	1
1_0	Series	1	Primer vídeo	0
1_1	Series	1	Segundo vídeo	1

Además, para aprovechar la petición GET al máximo, se envían dos variables de texto (que podrían ser el usuario y contraseña del usuario) junto con el mensaje de que el usuario ha reproducido un vídeo, camuflando así los datos.

B. Implementación interna

La aplicación cliente, tal y como indica el desarrollo de Samsung *Smart TV* [8], aúna varios archivos CSS, Javascript y HTML. El CSS funciona como marcador de estilo, imágenes, colores. Por su parte, el HTML muestra la estructura de la aplicación, es decir, se crean los ID que hemos indicado anteriormente, los distintos vídeos que se van a utilizar y las anclas, referencias a los elementos definidos para que sea mucho más sencillo referenciarlos posteriormente en el código. Por último, se encuentra el lenguaje JavaScript como elemento de unión donde se utilizan librerías tan típicas como *jQuery* y se define el propio funcionamiento de la aplicación.

En total existen cuatro archivos Javascript: *Data*, para enviar el mensaje HTTP al servidor; *File*, que almacena los métodos destinados a manejar los archivos de texto y directorios; *GetPath*, para obtener y devolver el valor absoluto de la ruta donde se encuentra los archivos; y *Main*, el archivo principal que define el funcionamiento de la aplicación, sobre todo en lo que respecta a la interacción del usuario y a la reproducción de los vídeos. En este archivo se detallan métodos para recoger la tecla pulsada del mando y realizar la tarea asignada a cada uno de ellos dependiendo de dónde se encuentre el usuario a cada momento, es decir, comprobando donde se encuentra el foco de la aplicación y los índices *index* e *index_contenido* para, por ejemplo, reproducir el vídeo exacto que el usuario desea o pasar de una a otra categoría cuando pulsamos la flecha hacia abajo del mando, marcándose el menú con los recuadros de color rosa como se vio en la Figura 4.

Con el fin de comprender cómo funcionan los distintos métodos, la Figura 5 muestra un diagrama de flujo con las distintas opciones disponibles para el usuario cuando se encuentra reproduciendo un vídeo. Así, en este caso, puede pulsar tres teclas: *Return*, que detiene la reproducción y devuelve el foco al contenido, es decir, se vuelve a la pantalla anterior teniendo en cuenta qué vídeo se ha reproducido; *Stop*, para detener la reproducción; y *Play*, para reproducir el vídeo. Cualquiera de las demás teclas que se pulsen no tendrán consecuencia alguna en la aplicación, asegurando así su buen funcionamiento.

Por su parte, el servidor Apache [9] se encarga de recoger los mensajes enviados por el cliente y servir los vídeos reproducidos. Principalmente se compone de cuatro

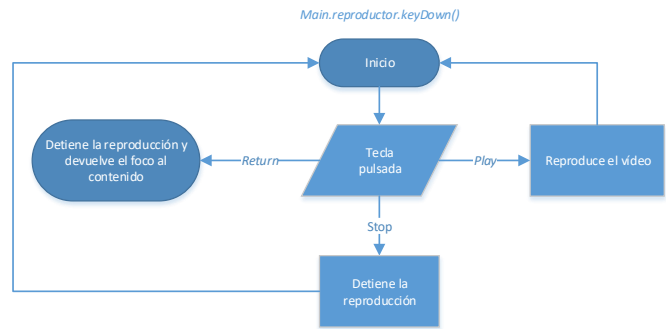


Fig 5. Diagrama resumen del funcionamiento del método *Main.reproductor.keyDown()*

elementos: el archivo que define su comportamiento llamado *proxy-img.php*, una carpeta “imágenes” en la que se encuentran los distintos vídeos y sus respectivas miniaturas, un archivo de texto llamado “*archivoSalida.txt*” que guarda en un registro la hora y la IP desde la que se ha contactado con el servidor y el *access.log* de Apache con el que se pueden comprobar las operaciones y las peticiones realizadas por los usuarios.

VI. EVALUACIÓN Y PRUEBAS

Tras desarrollar la aplicación al completo, se puso a disposición de los usuarios a través de Internet con una guía de instalación tanto en modo *root* como por USB, por si no estaba disponible el servidor en ese momento, en la página web <http://appsamsungsmarttv.blogspot.com.es/>. En lo que a nuestras pruebas respecta, el funcionamiento de la aplicación es el correcto, sirviendo los vídeos de una forma sencilla y rápida y obteniendo los mensajes en el servidor. Primero se realizaron pruebas en un emulador a través de Virtualbox [10] y luego en un televisor real tal y como se comprueba en la Figura 6.

Por otra parte, los objetivos del trabajo se han cumplido en su totalidad ya que se ha investigado el *Software Development Kit* de la plataforma de *Smart TV* Samsung, se han conocido a fondo las API y el emulador que ofrece la compañía, se ha estudiado cómo afecta el *malware* a un televisor inteligente con una aplicación con varias características interesantes y, además, se ha demostrado su funcionamiento en un dispositivo real.

El servidor, por su parte, también funciona correctamente, actualizándose sin problema con cada una de las visitas, de las reproducciones realizadas por los usuarios y de las descargas y ejecuciones de la aplicación como se muestra en la Figura 7. Por otra parte, tal y como indicamos en el apartado anterior, aparece un registro similar pero que guarda la hora y la IP de los clientes conectados.



Figura 6. Demostración de *PelisSeriesGratis* en un *Smart TV* real


```

37.29.143.4 - - [28/May/2015:01:28:06 +0200] "GET /proxy-img.php?file=0_0_reproducida.png&secreto=Da
37.29.143.4 - - [28/May/2015:01:28:06 +0200] "GET /videos/00.mp4 HTTP/1.1" 206 28044635 "-" "Lavf52.
37.29.143.4 - - [28/May/2015:01:28:12 +0200] "GET /videos/01.mp4 HTTP/1.1" 206 85800004 "-" "Lavf52.1
37.29.143.4 - - [28/May/2015:01:28:12 +0200] "GET /proxy-img.php?file=0_1_reproducida.png&secreto=Da
37.29.143.4 - - [28/May/2015:01:28:12 +0200] "GET /videos/01.mp4 HTTP/1.1" 206 97575 "-" "Lavf52.104
37.29.143.4 - - [28/May/2015:01:28:13 +0200] "GET /videos/01.mp4 HTTP/1.1" 206 8579956 "-" "Lavf52.1

```

Fig 7. Registro del servidor



Fig 8. Estadísticas extraídas de la página web

VII. CONCLUSIONES

Como se comentó anteriormente, se ha estudiado cómo funciona el SDK y su complejidad en lo que a términos temporales se refiere para el desarrollo de aplicaciones. Además, se ha desarrollado una aplicación con una interfaz sencilla y auto-actualizable que pone de manifiesto el riesgo para la seguridad gracias a la ingeniería social ya que, como se indicó, cualquier usuario malintencionado podría recoger los datos introducidos por los usuarios. Asimismo, el código de los distintos archivos ha sido suficientemente comentado para posibles futuros proyectos.

Por último, tras la investigación realizada durante estos meses, se ha llegado a la conclusión de que los electrodomésticos conectados a Internet y, en concreto, los televisores inteligentes, pueden resultar un grave problema para la privacidad y seguridad de los datos de los usuarios si estos instalan aplicaciones no verificadas y alojadas en tiendas y servidores diferentes a los oficiales. En definitiva, se demuestra que, al igual que ocurre con los *smartphones*, las *tablets* e incluso los ordenadores, un dispositivo capaz de conectarse a la red de redes es una amenaza potencial si este no se utiliza correctamente y no se tienen en cuenta las medidas de seguridad oportunas.

Todas estas conclusiones han sido propuestas, en parte, gracias a los resultados obtenidos en las pruebas reales. El *feedback* ha sido bastante positivo puesto que la web se ha visitado en varias ocasiones y ha habido usuarios que han utilizado la aplicación de forma continuada para ver películas y series como se muestra en la Figura 8, incluyendo aquellos que han colaborado "prestando" sus televisores para realizar las pruebas y los diversos vídeos que se grabaron, indicando que les gustaría que la aplicación siguiera desarrollándose para ser más completa y poder ver películas gratis en su televisor sin depender de servidores multimedia en su hogar u otros servicios en línea.

VIII. LÍNEAS FUTURAS

Aun habiendo cumplido la gran mayoría de los objetivos establecidos al inicio, aún quedan abiertas las siguientes líneas de trabajo futuras:

- Monitorización del usuario y del televisor: Algunas de las clases y métodos más interesantes que se pueden encontrar en el SDK no pueden utilizarse, o más bien

comprobarse, en el emulador, ya que se necesita un televisor compatible. Por ello, si es posible trabajar con uno de estos dispositivos, sería muy interesante obtener las listas de canales, cierta información del televisor como modelo, *firmware* que utiliza (para, por ejemplo, introducir un *firmware* con *malware* como actualización), identificador, etc., e incluso tomar el control de la cámara para grabar imágenes y enviarlas al servidor

- Reproducción en *streaming* de plataformas conocidas: La clase *Player* es capaz de reproducir vídeos en formato MP4 de forma remota mediante su URL, por lo que se ha utilizado un servidor para alojar algunos vídeos de demostración. No obstante, también es posible trabajar con alguna de las API desarrolladas por la comunidad, diseñadas para reproducir de forma más sencilla vídeos de YouTube y otras plataformas, lo que facilitaría la reproducción de otras películas y series.

- Ampliación de catálogo: A modo de prueba, la aplicación incluye cuatro vídeos en total, aunque lo más importante es que se amplíe el catálogo y ofrecer las mejores películas y series para aumentar el interés del usuario.

- Mecanismos de defensa: Una de las líneas de investigación más prometedoras sería la relacionada con la monitorización de los distintos elementos internos del televisor inteligente. Por ejemplo, podría crearse un sistema de detección de intrusos capaz de alertar al usuario si se detecta actividad sospechosa -por ejemplo, un aumento de temperatura del procesador sin que el propio usuario esté realizando ninguna actividad a priori exigente para el *Smart TV*, o el uso excesivo de la conectividad Ethernet o WiFi sin que ninguna aplicación esté siendo utilizada o actualizada-. Además, podría comprobarse si la cámara integrada o el micrófono están encendidos, lo que podría considerarse una amenaza para el usuario.

AGRADECIMIENTOS

Agradezco a toda mi familia los grandes esfuerzos de todo tipo que han realizado para que yo, finalmente, llegara a estudiar lo que quería, a mis padres y a mis hermanos.

Un enorme y tremendamente cariñoso gracias a mi abuelo Alfonso que lamentablemente no podrá ver, al menos desde aquí, cómo su nieto ha conseguido graduarse. ¡Va por ti, abuelo! Y gracias a vosotros, que desde Úbeda, Jaén, Málaga y otros lugares de España me habéis apoyado: gente del Interrail, Laura, Sofía y todas aquellas personas importantes en mi vida.

REFERENCIAS

- [1] Google, «Chromecast,» Disponible: <https://www.google.es/chrome/devices/chromecast/>
- [2] Strategy Analytics, «Global Connected TV Device Ownership Passes 1 Billion Units,», 2014. Disponible: https://www.strategyanalytics.com/strategy-analytics/news/strategy-analyticspress-releases/strategy-analytics-press-release/2014/12/09/global-connected-tvdevice-ownership-passes-1-billion-units-says-strategy-analytics#VWpcHc_tmko
- [3] Economics Computer, «2007 Malware Report: The Economic Impact of Viruses, Spyware, and Other Malicious Code,» 2007. Disponible: <http://www.computereconomics.com/page.cfm?name=malware%20report>.
- [4] RedesZone, «Samsung: sus Smart TV tienen un problema de seguridad,» 2012.

- Disponible: <http://www.redeszone.net/2012/12/12/samsung-sus-smart-tv-tienenun-problema-de-seguridad/>
- [5] Strategy Analytics, «Samsung Reasserts Smart TV Dominance,» 2014. Disponible: https://www.strategyanalytics.com/strategy-analytics/news/strategy-analyticspress-releases/strategy-analytics-press-release/2014/05/19/samsung-reassertssmart-tv-dominance-says-strategy-analytics#.VWpcQ8_tmko.
- [6] Tomorrow Focus Media, «Smart TV Effects 2014-1,» 2014. Disponible: http://www.tomorrow-focus-media.de/uploads/tx_mjstudien/TFM_SmartTVEffects_2014-1.pdf
- [7] Samsung, «Samsung D Forum,» Disponible: <http://www.samsungdforum.com/>
- [8] Samsung, «Development Guide,» Disponible: <http://www.samsungdforum.com/Guide/>
- [9] Apache Friends, «XAMPP Installers and Downloads,» Disponible: <https://www.apachefriends.org/es/index.html>
- [10] Oracle, «VirtualBox,» Disponible: <https://www.virtualbox.org/>



Jose López Arredondo (22 de Agosto de 1991, Úbeda, Jaén) es Graduado en Ingeniería de Tecnologías de Telecomunicación con mención en Telemática por la Universidad de Granada en 2015.

Adicionalmente, ejerce como periodista en varios medios digitales como el diario económico Cinco Días, en su sección de

tecnología Smartlife.



Pedro García Teodoro es Catedrático de Universidad del área de Ingeniería Telemática de la Universidad de Granada. Tanto su labor docente como su labor investigadora se desarrollan en el campo de las redes de comunicación e Internet, en el cual ha realizado numerosas contribuciones en forma de libros, artículos en revista y ponencias en congresos, tesis y proyectos y contratos.

Diseño e implementación de retos de seguridad en redes de telecomunicación en el laboratorio

3.7 con fines docentes

Autor: Francisco López Pérez, e-mail: franclo@correo.ugr.es

Tutor: Rafael Alejandro Rodríguez Gómez; e-mail: rodgom@ugr.es

Titulación: Grado en Ingeniería de Tecnologías de la Telecomunicación
Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Los laboratorios de red dentro del ámbito docente constituyen hoy en día una de las herramientas de mayor utilidad para el trabajo práctico de los alumnos en diversas materias. No obstante, se requiere una evolución permanente dentro de los mismos asociada a una mayor flexibilidad en factores como el número de equipos de trabajo, la configuración en red, el equipamiento software o el sistema operativo. La idea es buscar una solución que permita a su vez a los alumnos desplegar, desarrollar y simular retos relacionados con la seguridad en red. En base a las limitaciones de sus antecedentes, el Proyecto de Innovación Docente 014-54 sobre el que trabaja el Departamento de Teoría de la Señal y Telemática y Comunicaciones de la Universidad de Granada trabaja para alcanzar ese modelo de laboratorio completo del que pueda hacer uso docente y alumno. El uso conjunto de virtualización y *live-USB* personalizados constituye una alternativa de trabajo con la que el docente puede personalizar sus propias máquinas virtuales según sus necesidades, así como la capacidad de acceder a máquinas internas o externas de la red física mediante configuraciones adecuadas. Dentro de este contexto, la herramienta Systemback permite crear imágenes copiando el estado vigente de sistemas operativos Ubuntu y que éstas puedan ser exportadas como *live-USB*.

Palabras clave: laboratorio de red, *live-USB*, innovación docente, seguridad, virtualización.

I. INTRODUCCIÓN

Hoy en día, los laboratorios de redes dedicados a la enseñanza son sin duda uno de los recursos docentes de mayor importancia para el desarrollo práctico de asignaturas relacionadas con la informática y las telecomunicaciones. Cada día, decenas de equipos son arrancados por los alumnos en estas instalaciones con el objetivo de simular de manera controlada escenarios de red sobre los que afianzar conceptos teóricos de estas asignaturas. Debido a la importancia que confiere, no sólo a los procedimientos de enseñanza de los docentes sino al aprendizaje de alumnado, es de vital importancia contribuir al asentamiento de estos entornos de trabajo. Un problema muy presente dentro de estos laboratorios es la limitación física de los mismos, ya que cuenta con un número fijo de equipos que puede resultar pequeño para el desarrollo de ciertas temáticas. Esto hace necesario la búsqueda de alternativas y tecnologías que permitan explotar el equipamiento físico de los mismos en busca de satisfacer ejemplos prácticos que requieran el uso de más de un equipo por alumno.

Este hecho también se considera dentro del proyecto de innovación docente (PID 014-54) en el que la Escuela Superior de Ingeniería Informática y Telecomunicaciones ya pretende afrontar esa problemática mediante la implementación de un laboratorio virtual de seguridad en redes.

Consecuencia de ello, se ha continuado la línea de trabajo planteada en esta investigación mediante la propuesta de un modelo de laboratorio docente basado en la evolución de un laboratorio de red convencional, cuyo objetivo a corto medio plazo es que forme parte de las clases prácticas. Para ello, se ha hecho necesaria la incorporación de herramientas de inicio llamadas *live-USB* o USB de arranque. Estos dispositivos, previamente configurados, permiten arrancar un sistema operativo plenamente funcional para trabajar a nivel de red a través de unas máquinas virtuales. El empleo de máquinas virtuales busca desarrollar distintas vías de trabajo preconfiguradas en una misma máquina física. La motivación de esta idea radica en la búsqueda de un entorno funcional y versátil en el que el alumno disponga del material necesario para trabajar en las prácticas de laboratorio sin preocuparse por el número de equipos o sistemas operativos requeridos para la experimentación.

La idea principal es que a larga sea posible plasmar esta experimentación en entornos de pruebas con ataques mediante las máquinas virtuales albergadas en los *live-USB*, de modo que únicamente sea necesario arrancar los dispositivos USBs de inicio en las máquinas físicas para poder acceder a un escenario de red de interés. Para ello, se pretenden alcanzar los siguientes objetivos:

- *Herramienta de virtualización en live-USB:* propuesta de una herramienta de trabajo basada en virtualización con *live-USB* como solución de modo que pueda ser personalizada por el usuario, incluyendo tanto las máquinas virtuales deseadas como la configuración de red adecuada. En este punto es importante familiarizarse con el empleo de herramientas que generan imágenes de sistemas operativos en vivo.
- *Funcionamiento de la herramienta en equipos físicos:* el objetivo es acoplar esta herramienta en equipos físicos con objeto de poder simular esquemas de red basados en virtualización pura que permitan la comunicación entre máquinas virtuales. Posteriormente, este trabajo es extrapolado al

laboratorio de red docente configurado en el aula 3.7 de la Escuela Técnica Superior de Ingeniería Informática y Telecomunicaciones.

- *Casos prácticos*: una vez adecuados los *hosts* y configurados los sistemas operativos *guests*, se desarrollan casos prácticos sobre los que puede trabajar el alumnado, ya sea a nivel distribuido entre equipos de la red o a nivel local entre las máquinas virtuales de una misma máquina física.

II. ESTADO DEL ARTE

El modelo de trabajo propuesto se presenta como alternativa a otras propuestas anteriores, que si bien son capaces de mejorar la calidad y funcionalidad de los laboratorios de redes en distintos aspectos, no se afianzan como una solución definitiva a adoptar de manera genérica.

A. Laboratorios remotos.

El acceso remoto a los laboratorios de red físicos ha sido una utilidad bastante incorporada en muchas redes físicas del mundo para permitir el trabajo simultáneo en el mismo equipamiento físico.

Este esquema de trabajo se basa en el concepto VPN, que permite el acceso a recursos alejados físicamente de nuestro equipo. Mediante el empleo de comunicaciones punto a punto de manera segura y privada a través de una infraestructura de red pública como es Internet, se logra que la información intercambiada no sea vista a través de la red compartida. Esto se consigue mediante el encapsulamiento de los paquetes a través de la inclusión de cabeceras que generan lo que se conoce como túnel. Esta tecnología es bastante sencilla de acoplar y no supone ningún consumo excesivo de los recursos del laboratorio. A su vez, la implantación de esta tecnología a nivel docente ha permitido, entre otras ventajas, solucionar el problema de sobrecarga en las aulas físicas de trabajo, posibilitando a su vez que todos los alumnos puedan interactuar de manera directa con la red sin encontrarse físicamente trabajando en la misma. Sin embargo, la implementación de esta técnica requiere una planificación temporal de acceso al servidor VPN, para evitar conexiones simultáneas de demasiados usuarios que puedan entorpecer entre sí sus desarrollos prácticos.

B. Sistema Rembo

Esta herramienta se fundamenta también en acceso remoto a servicios pero con un enfoque distinto. Esta arquitectura de red permite concentrar los recursos de sistemas operativos en una máquina centralizada, a la que se accede a través de la red.

El sistema Rembo se invoca durante el proceso inicial de arranque de los equipos antes de cargar cualquier sistema operativo del disco duro. Este servidor con una configuración de red adecuada por medio de DHCP y con suficiente almacenamiento para albergar distintas imágenes de sistemas operativos, debe permitir que los equipos arranquen de manera remota la imagen de cualquier sistema operativo que contenga a través de una arquitectura PXE (*Preboot eXecution Environment*), siendo capaz de usar la red gracias a

un entorno programado proporcionado por un chip situado en la tarjeta de red.

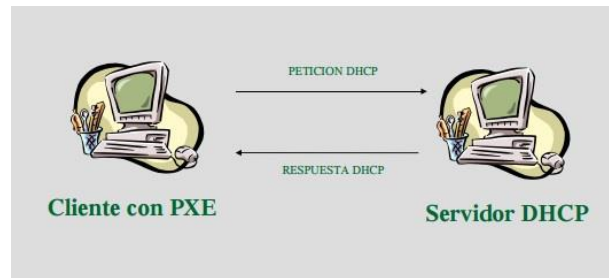


Figura 1. Arquitectura PXE para un servidor Rembo

La principal ventaja que aporta esta arquitectura es que permite concentrar los recursos de sistemas operativos en una máquina centralizada, permitiendo optimizar el tamaño de memoria disponible en los discos duros de las máquinas residentes en la red. Unido a ello, es posible establecer configuraciones previas en los sistemas operativos cargados en el servidor y conservar dichas imágenes intactas tras su uso por cualquier cliente.

Entre las limitaciones que presentan dichos laboratorios, cabe resaltar la lentitud que tienen las máquinas para cambiar su aspecto puesto que no pueden reconfigurarse como cualquier equipo que tiene instalados varios sistemas operativos en distintas particiones.

C. Redes VLAN

La introducción de redes virtuales de área local en los laboratorios físicos es otra propuesta generalizada en las clases prácticas. Por lo general, las redes VLAN permiten comunicar entre sí conjuntos de equipos que no tienen ninguna relación física entre sí, pero que requieren abstraer su comunicación del resto de los equipos.

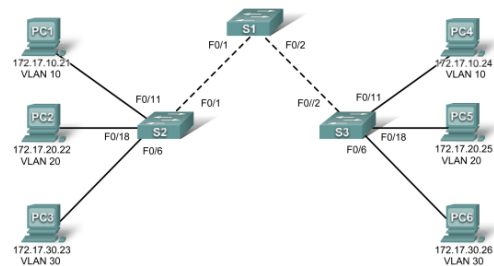


Figura 2. Esquema de red de varias VLAN

Esta tecnología suponen una nueva alternativa para evitar tiempos innecesarios en la reconfiguración de la red, puesto que permiten la compartición de los mismos medios físicos entre distintas redes lógicas, cada una con su extensión y dominio de difusión. Además, la adición de estas vías de comunicación suponen no sólo el afianzamiento de la conexión física de los equipos en red, sino que se pueden añadir tantas VLAN como se quieran sin minar los recursos en red.

No obstante, existe el problema de que el alumno no puede trabajar con este concepto fuera del laboratorio, ya que este concepto es introducido por los *switches* físicos del laboratorio y requiere de simuladores de red bastante avanzados como *OPNET*. Además, la flexibilidad de las configuraciones de red es menor y la optimización del tiempo

empleado para ello han llevado a la búsqueda de la abstracción de lo propiamente físico.

D. Virtualización

El empleo de técnicas de virtualización en la actualidad está bastante popularizado, puesto que permiten emular con gran realismo un sistema operativo como si se estuviera instalado en el disco duro de una máquina física, con unas prestaciones bastantes similares.

El empleo de máquinas virtuales permite establecer diferentes configuraciones de un mismo sistema operativo sin ocupar demasiado espacio, o incluso almacenar distintos sistemas operativos. El hecho de poder trabajar con varios sistemas operativos dentro una misma máquina física proporciona una gran versatilidad a la hora de realizar prácticas de red, tanto en modo local como en modo distribuido combinando la red física con la virtual. La gestión de varios sistemas operativos conectados en la red se podrá realizar de forma simultánea, pudiéndose modificar de forma paralela e incluso añadir o eliminar máquinas virtuales de manera rápida, lo que se traduce en un gran escalabilidad. Además, el empleo de elementos virtuales nos permite optimizar recursos y reducir gastos mediante la sustitución de ordenadores y equipos de interconexión físicos por máquinas virtuales dentro de las subredes que se puede interconectar dentro de un mismo equipo físico.

El principal inconveniente que presenta esta solución es que se reducen las prestaciones de los sistemas operativos virtualizados debido a la compartición de recursos como RAM, procesador o memoria entre todas las máquinas virtuales residentes en el equipo físico y el propio sistema operativo *host*.

E. Laboratorio virtual basado en *live-USB* con máquinas virtuales

La base de esta idea de laboratorio va ligada a extender el concepto de virtualización a un nivel mayor de abstracción, de modo que los equipos del laboratorio no requieran la instalación previa de ninguna herramienta o sistema operativo que vaya a ser utilizado.

Aquí es donde entra en juego la idea de generar *live-USB* personalizados. Estos dispositivos permiten arrancar un sistema operativo albergado en su interior desde cualquier máquina física, sin necesidad de instalarlo en el disco duro de la máquina que lo ejecuta. El hecho de que los *live-USB* sean compatibles con cualquier equipo de una red física hace que sea factible su integración en cualquier arquitectura de laboratorio. La idea consiste en crear *live-USB* específicos para cada práctica en concreto, de modo que los alumnos tenga todo lo necesario para trabajar dentro de unas máquinas virtuales previamente configuradas. Este propósito se consigue combinando estos dos conceptos (*live-USB* y virtualización), de modo que al arrancarse el dispositivo USB, se incluya dentro de su sistema *host* un software de virtualización con dichas máquinas virtuales disponibles para ejecutarse. De este modo, se puede establecer cualquier configuración de red entre las máquinas virtuales de los *live-USB*, con una escalabilidad y flexibilidad mayor de las que ofrece el equipamiento físico de red y con las herramientas y características que se requieran incorporadas para el trabajo práctico de los alumnos en las sesiones de laboratorio.

A diferencia de lo que pueda parecer, el procedimiento para generar *live-USB* personalizados con máquinas virtuales es bastante sencillo e intuitivo, factor que también a tener en cuenta en comparación con otras soluciones que requieren no sólo complejas configuraciones, sino conocimiento especializado de redes de telecomunicaciones. Unido a las ventajas que ya ofrecía el empleo de virtualización por sí solo, la inclusión de un *live-USB* a esta filosofía ofrece las siguientes ventajas:

- Aporta portabilidad al material de trabajo, permitiendo al alumno no sólo emplear el dispositivo dentro de cualquier equipo del laboratorio de redes, sino en cualquier máquina. Esto permite que el alumno pueda familiarizarse con el entorno de trabajo de manera local desde su propio portátil o desde los equipos que pone a disposición la universidad.
- Posibilidad de incluir la configuración software necesaria para una práctica en un formato no sólo reducido y fácil de utilizar, sino que permite ser clonado en otro dispositivo para configuraciones iguales o similares. De este modo, el proceso de generación anterior sólo se repite si se requiere un despliegue de máquinas virtuales completamente distinto.
- Adicionalmente, la configuración por defecto establecida en el sistema *host* permanece invariable cada vez que se arranca el dispositivo USB, independientemente de lo que contenga el disco duro de la máquina que lo inicia. Así, los cambios que se realicen en la configuración del sistema operativo *host* o en las máquinas virtuales desaparecerán al cerrar sesión.
- Los tiempos de arranque y reconfiguración del software necesarios quedan sensiblemente reducidos, ya que no se requiere el reinicio de la máquina física ni la instalación de nuevos elementos; todo está predeterminado en las máquinas virtuales del *live-USB*.
- Permite la reutilización de los recursos del laboratorio físico para otras aplicaciones dentro del ámbito docente. La utilización de los recursos y las máquinas es principalmente virtual, de manera que los cambios que se realicen afectan muy poco a la composición física de la red.

III. DESCRIPCIÓN DEL LABORATORIO DE SEGURIDAD

El despliegue y configuración del modelo de laboratorio docente escogido tiene como principal objetivo mostrar la evolución de un laboratorio físico en base a la tecnología adoptada, con objeto de solucionar sus carencias.

El diseño genérico que se plantea para este laboratorio va enfocado a continuar la experimentación ya iniciada en el Proyecto Final de Carrera de Anabel Reyes Maldonado dentro de su trabajo sobre herramientas de detección de intrusos para entorno docente. El objetivo de su proyecto consistía en la instauración de un laboratorio virtual de seguridad basado en máquinas virtuales sobre el que realizar

experimentación basada en esta temática. La base de su trabajo de experimentación partía de la inclusión del software de virtualización por emulación Oracle VM VirtualBox dentro de una máquina física, con objeto de proporcionar escalabilidad hacia un número de máquinas superior al disponible con los equipos físicos.

Sin embargo, en el planteamiento de Anabel, se deja entrever la posibilidad de evolucionar en el futuro esta propuesta de red con máquinas virtuales hacia un enfoque más flexible; una solución intermedia entre laboratorio físico y real y entornos de virtualización.

Los *live-USB* personalizados permiten completar la idea de laboratorio de seguridad combinando los conceptos de *live-USB* y virtualización. El objetivo es hacer posible el cambio de configuración y aspecto de los PCs del laboratorio de manera rápida y sencilla en función del trabajo a realizar. Gracias a esto, se puede desarrollar un laboratorio docente completo compuesto por una parte física convencional según la topología de red del laboratorio y una parte virtual que contribuye a extender el dimensionado de la red a un número de máquinas superior mediante el uso de redes internas adheridas a las anteriores.

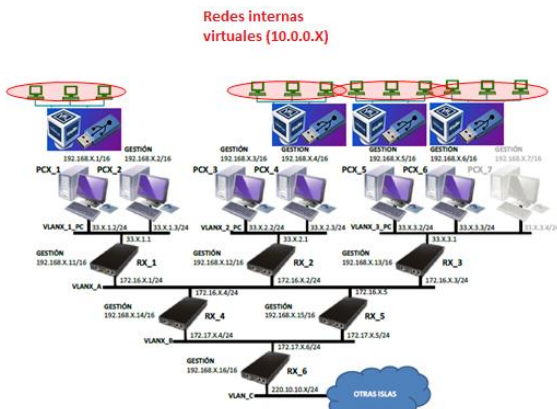


Figura 3. Ejemplo de red con *live-USB* de virtualización en una isla del laboratorio 3.7 de la ETSIT

La arquitectura de red física del laboratorio permanece inalterada de modo que los *hosts* siguen manteniendo los mismos dominios de colisión y difusión. El cambio se produce con la incorporación de redes internas a través de las interfaces de red que proporciona el software de virtualización. Esto hace que el *host* actúe como intermediario entre la red local de máquinas virtuales de cada equipo y la red distribuida del laboratorio.

Para que las máquinas virtuales sean accesibles desde el resto de la red y puedan comunicarse unas con otras, es necesario especificar un redireccionamiento de puertos entre la máquina *host* y sus *guests* correspondientes. De este modo, se podrán reenviar conexiones entrantes desde la máquina anfitrión a puertos internos de las máquinas virtuales. Esta configuración de red se conoce como NAT, y hace que la máquina física actúe a modo de *router*, traduciendo las conexiones que tienen su destino en equipos de la red interna virtual.

El mecanismo de trabajo es bastante sencillo una vez configurado todo. Una vez que el docente tiene generados los *live-USB*, cada alumno puede hacer uso de él para seguir el transcurso de la explicación práctica. La idea es que al arrancar el dispositivo en un equipo, se cargue el sistema

operativo del *host* que incluya distintas máquinas virtuales con la configuración necesaria para desenvolverse en el desarrollo de la práctica. Una vez acabada la sesión y apagados los equipos, todos los cambios realizados a nivel virtual no serán permanentes y desaparecen una vez que se vuelve a iniciar el sistema.

IV. EMPLEO DEL *LIVE-USB*

La inclusión de este complemento al modelo de laboratorio virtual contribuye a permutar más fácilmente el aspecto de los equipos del laboratorio y de manera totalmente planificada. Este resultado, unido al hecho de que ninguno de los cambios que se realicen en la apariencia de los equipos será permanente debido a la propia naturaleza del *live-USB*, proporciona un modelo de trabajo bastante completo.

Para poder generar *live-USB* existen una gran cantidad de herramientas preparadas para este propósito. No obstante, no todas son las más adecuadas para este cometido. Systemback es un programa presente en las últimas distribuciones de Ubuntu que hace posible la tarea de crear copias de seguridad del sistema y de los archivos de configuración de los usuarios. Incluye características adicionales como la copia del sistema, la instalación del sistema y la creación del sistema en vivo. Sin embargo, lo más interesante para el propósito propuesto es la creación de un *live-USB* instalable tomando como base el sistema *host*.

En la página oficial de Systemback, se proporcionan los comandos necesarios para su instalación, como adición de la descripción PPA en los repositorios del sistema.

```
sudo add-apt-repository ppa:nemh/systemback
sudo apt-get update
sudo apt-get install systemback
```

La interfaz del programa es bastante intuitiva, en la que las acciones a realizar se especifican en cada uno de los botones visibles.

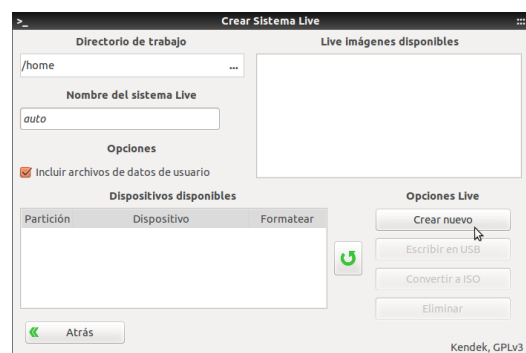


Figura 4. Interfaz gráfica de Systemback

Por tanto, para generar una imagen de sistema operativo basta con seleccionar *crear Sistema Live*. Una vez dentro, hay que declarar el directorio de trabajo del sistema que es replicado en la imagen y si deseamos incluir los archivos de datos en la copia que se genera. Pulsando en *crear nuevo*, se inicia la generación de la imagen. Este proceso, dependiendo del peso del sistema, tardará alrededor de una hora y hora y media para completarse.

Tras su finalización, resulta un archivo de formato .sblive que bien podrá convertirse en formato .iso si su tamaño es menor de 4 GB, o bien se puede cargar directamente en el dispositivo USB para conformar el dispositivo de arranque, seleccionando para ello *escribir en USB*. Este proceso no conlleva más de 20 minutos.

Como resultado se crea un *live-USB* que, una vez iniciado en cualquier equipo, carga un menú de inicio de Systemback donde se puede seleccionar el arranque del sistema Ubuntu como Live o instalarlo en el disco duro de la máquina donde se arranca. En definitiva, se comporta como una distribución oficial.

V. ENTORNO EXPERIMENTAL

El objetivo principal de esta experimentación es demostrar que se puede extrapolar cualquier esquema de red sencillo presente en una arquitectura de red física a una arquitectura mixta que combine conexiones físicas y virtuales.

Para ejemplificar su funcionalidad, se diferencian dos partes en la experimentación. La primera va enfocada al rendimiento de la herramienta *live-USB* en base al tiempo empleado para su generación. La segunda práctica de ámbito más práctico, se vale de Hadoop como herramienta de trabajo. A partir de su instalación y configuración, el enfoque de la experimentación va ligado al planteamiento de ejercicios sencillos con los que se puede comprobar el funcionamiento y adaptabilidad de un clúster a la nueva configuración de equipos.

A. Tiempo de trabajo de Systemback

Esta parte de la experimentación atiende a las prestaciones obtenidas por la herramienta Systemback.

Este software exclusivo de Ubuntu logra generar copias in vivo del sistema operativo en imágenes que pueden ser montadas en dispositivos USB para que trabajen como *live-USB*. Partiendo de su funcionamiento, puede ser interesante para el docente tener una estimación aproximada del tiempo que necesita para confeccionar distintos *live-USB* para una clase práctica.

Para clarificar esta idea, han sido configurados con cierto criterio distintos conjuntos de máquinas virtuales que contienen Lubuntu con la herramienta Hadoop y otros sistemas operativos de interés para el desarrollo práctico en las clases de docencia. El objetivo es plasmar el tiempo que tarda la herramienta Systemback en generar las imágenes .sblive en función de las máquinas *guest* de VirtualBox, así como el tiempo que la herramienta emplea para la generación del *live-USB* a partir de dicha imagen.

Tabla 1. Tabla de tiempos con las distintas máquinas *guest*

Guests incluidos	Tamaño(GB)	T.imagen(min)	T.Live-USB(min)
4 Lubuntu 2 XP y 1 Cloudera	21,16	95	24
6 Lubuntu	25,38	81	19
2 Lubuntu 2 Ubuntu	17,3	89	25
2 Cloudera VM	15,62	65	23
3 Lubuntu y 1 Cloudera VM	19,99	69	17
4 Lubuntu con Hadoop	16,92	51	15

B. Experimentación con Hadoop

El principal objetivo de estas pruebas es no sólo proporcionar información de interés sobre la generación de los *live-USB* y el trabajo en un clúster, sino también demostrar la funcionalidad que de esta herramienta se presume.

Para ello, se vale de una arquitectura de trabajo concreta basada en la exclusividad de máquinas Lubuntu con Hadoop. El primer objetivo es comprobar las prestaciones a la hora de iniciar un clúster Hadoop con distinto número de *datanode*. Para ello, se configuran varias máquinas virtuales de un mismo *live-USB*, de modo que haya un *namenode* y varios *datanodes*. Con las diferentes arquitecturas, se mide el tiempo transcurrido desde que se introduce la orden *start-all.sh* por terminal. El rango de experimentación va desde un *datanode* hasta cuatro en el mismo clúster.

Tabla 2. Tiempos de arranque del clúster en función del número de *datanodes*.

Nº Datanodes	p1(s)	p2(s)	p3(s)	Media(s)
4	92,58	98,81	96,47	95,95
3	83,29	79,25	77,92	80,15
2	65,3	67,12	69,87	67,43
1	60,25	63,8	67,89	63,98

En segundo lugar y como fin a este análisis de rendimiento de un *live-USB*, se comprueba el funcionamiento del clúster generado mediante peticiones de cliente al *namenode*.

Hadoop ofrece en su configuración por defecto ejemplos de tareas que permiten analizar el comportamiento del clúster lanzado. Para este ejemplo práctico, se utiliza la tarea que permite calcular el número π .

En la declaración para realizar la petición de esa tarea, se especifica tanto el número de *maps* como de *reduce* que se pide realizar a los *datanodes*. Manejar estos valores en función del número de nodos en el clúster puede hacer variar el tiempo de ejecución hacia valores óptimos; en nuestro caso, se le han dado valores 10 y 2 respectivamente.

Tabla 3. Tiempos de ejecución de la tarea π

Nº Datanodes	p1(s)	p2(s)	p3(s)	Media(s)
4	25.56	24.12	25.72	25.13
3	24.89	24.01	23.67	24.19
2	23.85	21.38	21.78	22.34
1	22.74	20.68	21.07	21.50

C. Análisis de los resultados

En primer lugar, se ha comprobado la gran inversión temporal que requiere generar un *live-USB*. El tamaño del sistema operativo en conjunto a los *guests* que contiene, unido al tipo de sistema operativo escogido en cada máquina virtual, influye de gran manera en el tiempo de ejecución de Systemback, tanto para la generación de la imagen copia del sistema in vivo como para su montaje en el dispositivo USB. En este punto, se aconseja que el docente ofrezca una organización de máquinas virtuales para cada sesión única,

de modo que el tiempo de trabajo consista en mayor medida en volcar la misma imagen tantas veces como sea necesaria. No obstante, es una tarea previa a la sesión práctica y además puede ser realizada de manera sistemática y en paralelo con otras obligaciones. En comparativa con la planificación actual que puede conllevar incluso un día completo de configuración de equipos, la comparativa de tiempo invertido es positiva.

Posteriormente, se ha realizado una experimentación más específica ligada a estimar los tiempos de arranque de clúster Hadoop sobre distinto número de *datanodes*. Se aprecia una relación entre ese número de nodos y el tiempo invertido en su funcionamiento, de modo que se puede preveer el tiempo empleado para un número cualquier de nodos. Sin embargo, la limitación de los *live-USB* en términos sobre todo de memoria RAM, limita la ejecución en paralelo a cuatro nodos. Siguiendo también el modelo de trabajo propuesto por Hadoop, se han realizado tareas sencillas para calcular el valor del número *pi* atendiendo a los parámetros introducidos para el proceso *mapreduce* y también considerando los *datanodes* presentes en el clúster. Gracias a estas simulaciones, se ha demostrado que un mayor número de procesos de reducción por cada mapeo realizado contribuye a proporcionar una mayor utilidad de los *datanodes* del clúster así como obtener unos resultados más precisos.

VI. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha logrado implementar con éxito una solución a la problemática de flexibilidad presente en los laboratorios de red docente.

Para lograr este objetivo, se ha buscado continuar la idea de trabajo docente introducida en proyectos anteriores, donde se proponía la necesidad de encontrar una solución que hiciera posible realizar pruebas de seguridad enfocadas al ámbito docente. Consecuencia de ello, ha resultado la generación de la herramienta *live-USB* que engloba las ventajas de la virtualización y la tecnología *Live* en una solución completa, personalizada y flexible. Este hecho ha sido evidenciado en su buen funcionamiento en equipos físicos, donde se han realizado diferentes pruebas y casos prácticos con los que se ha apreciado, en base a la experimentación realizada, una gran similitud al despliegue de red física que plantea.

Los resultados obtenidos dan buena idea de que es una solución a tener en cuenta para el trabajo de los docentes en el desarrollo y planificación de las clases prácticas en los laboratorios de red; no sólo con temática de seguridad, sino de cualquier otro tipo que involucre el empleo de redes.

La investigación desarrollada durante este trabajo no termina aquí. En paralelo a esta investigación, el proyecto de innovación docente PID 014-54 dentro del grupo de investigación NESG busca mejorar esta herramienta de manera que pueda ser utilizada en futuras clases prácticas de la Escuela Técnica Superior de Ingeniería Informática. Para ello, los planes previstos para el futuro son los siguientes.

- Afianzamiento de esta solución en entornos distribuidos que permita la comunicación fluida entre máquinas virtuales de distintos equipos físicos de un laboratorio.
- Adicción de herramientas de medición complementarias en los sistemas operativos de los

live-USB enfocadas a la temática de seguridad con las que puedan trabajar los alumnos.

- Búsqueda de alternativas más veloces para generar los *live-USB*.
- Exclusivizar los sistemas operativos *guests* empleados a máquinas *Lubuntu*.
- Planteamiento de temáticas trabajo originales destinadas a comprobar la funcionalidad específica de los *live-USB* como soluciones.

VII. PLANIFICACIÓN Y COSTES

En la siguiente figura, se muestra mediante un diagrama de Gantt la evolución temporal seguida en cada una de las fases principales del trabajo.

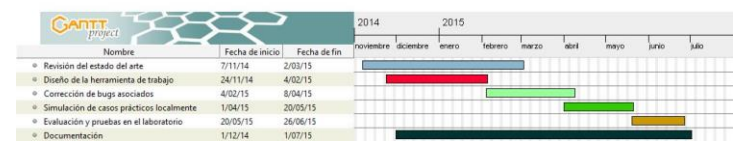


Figura 5. Planificación temporal del trabajo

El desembolso total que engloba recursos humanos, hardware y software viene desglosado en la tabla adjunta.

Tabla 4. Costes totales de recursos

Recursos	Coste (€)
Humanos	24240
Hardware	10952
Software	250
Total	35424

VIII. REFERENCIAS BIBLIOGRAFICAS

Diseño e Implantación de un Laboratorio para la Docencia de Redes Telemáticas para la ETSIIT. http://dtstc.ugr.es/~gmacia/papers/Jitel07_LabDocencia.pdf.

Ejemplo de la filosofía *live-USB* con virtualización. <https://www.dtic.ua.es/grupoM/recursos/articulos/JDARE-06-G.pdf>.

Gómez Muñoz, A. "Clustering en Big Data. Aplicación en Seguridad en Red." 2014.

Laboratorio de redes virtuales en la Universidad de Málaga. <http://upcommons.upc.edu/revistes/bitstream/2099/12001/1/a40.pdf>.

PFC de Eugenio Villar Fernández: Virtualización de servidores de telefonía IP en GNU/Linux (Universidad de Almería). http://www.adminso.es/images/d/dc/PFC_eugenio.pdf.

Reyes Maldonado, A. "Herramientas Big Data de detección de intrusiones para entorno docente y de investigación en Seguridad en Redes de Computadores". 2014.

Systemback o cómo crear puntos de restauración en Linux. <http://blog.desdelinux.net/systemback-o-como-crear-puntos-de-restauracion-en-linux/>.

Universidad de Caracas - Virtualización en el ámbito educativo. http://tecnologiaedu.us.es/cuestionario/bibliovir/La_virtualizacion_univ.pdf. (Visitado el 06/26/2015).

MEDA-Toolbox: interfaz gráfica en Matlab para la detección de anomalías en red

Autor: Pablo Sánchez Robles, e-mail: pablosanchez@correo.ugr.es

Tutor: José Camacho Páez, e-mail: josecamacho@ugr.es

Titulación: Ingeniería Informática

Departamento de Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada

Resumen—El objetivo del documento es el desarrollo de una interfaz gráfica para análisis de BIG DATA haciendo uso de la Toolbox MEDA, herramienta de software libre desarrollada en la Universidad de Granada, y orientar dicha interfaz a la detección de anomalías en redes de computadores con un amplio número de conexiones y usuarios finales. El objetivo principal es simplificar el manejo de MEDA-Toolbox, mediante la interfaz gráfica y conseguir que sea utilizada por un mayor número de usuarios. Parte fundamental del trabajo de fin de grado ha sido la revisión de las distintas herramientas y visualizaciones que incorpora MEDA-Toolbox. Dichas herramientas y visualizaciones se basan en el análisis multivariante de datos, destacando el Análisis por Componentes Principales para la reducción de la dimensionalidad del problema, y el Clustering para la reducción del volumen de datos. Finalmente, el trabajo estudia la aplicabilidad de la herramienta, de propósito general, en problemas de seguridad en red.

Palabras clave—Análisis de componentes principales (PCA), Análisis exploratorio de datos (AED), Big Data, datos de seguridad en red, detección de anomalías en red, GUIDE, MEDA-Toolbox.

I. INTRODUCCIÓN Y OBJETIVOS

El documento que aquí se resume viene motivado por las crecientes amenazas a la seguridad de los procesos informáticos y la necesidad que ello genera de una detección a tiempo real de los mismos para evitarlos. En este texto se pretende crear una interfaz gráfica que facilite el uso de la Toolbox MEDA desarrollada por la UGR, mas concretamente facilitar el uso de las funciones relacionadas con el análisis de Big Data. Otro objetivo a cumplir es el aprendizaje de las herramientas teóricas estadísticas que utiliza MEDA Toolbox para su funcionamiento en el análisis de grandes conjuntos de datos. A lo largo del documento se exponen las fases de análisis, diseño y desarrollo del software, también se trata un tema sobre la extracción de datos en red mediante la captura de paquetes o de flujos de tráfico. Por último se prueba el funcionamiento del software mediante un ejemplo práctico con datos reales de una red corporativa anonimizada.

II. CONCEPTOS TEÓRICOS

[1] [2][3] [4] En los primeros temas se tratan las técnicas estadísticas del análisis exploratorio de datos o EDA, que utiliza las técnicas de proyección de Análisis de componentes principales o PCA y Mínimos Cuadrados Parciales o PLS. Dichas técnicas de proyección se utilizan para la disminución del tamaño del problema. En el caso de PCA se dispone de una matriz de datos X de N observaciones y M variables

y mediante la generación de variables latentes se reduce el tamaño del problema de las M iniciales al número de variables latentes seleccionadas. En el caso de PCA las variables latentes se denominan componentes principales y se calculan como combinaciones lineales de las variables originales siendo la primera componente principal la recta que maximiza la varianza, como se puede apreciar en la Fig. 1. Por otra parte PLS funciona de forma similar, con la diferencia de que en este caso se tiene información sobre las relaciones entre variables en una segunda matriz, Y , que se utiliza para realizar el análisis. En estos primeros temas también se tratan los mecanismos de visualización que tiene la Toolbox MEDA para entender las correlaciones que existen entre variables (Fig. 4), las relaciones entre observaciones (Fig. 2), y las relaciones dobles entre variables y observaciones (Fig. 3).

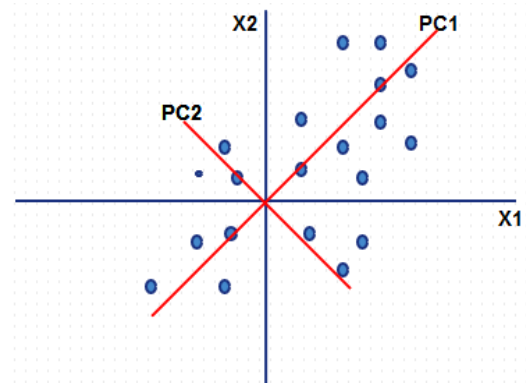


Fig. 1: Ejemplo de componentes principales

En estos primeros capítulos también se explican las técnicas de tratamiento de grandes conjuntos de datos de MEDA-Toolbox.

III. EXTRACCIÓN DE DATOS DE SEGURIDAD EN RED

[5] [6][7] En el documento se presenta un capítulo hablando sobre la extracción de datos en red necesarios para llevar a cabo el análisis de la seguridad en la misma. Se habla sobre los datos estáticos y las series temporales, así como los flujos de datos y el análisis de paquetes de datos. Se comenta también la información de relevancia que se encuentra en los ficheros de registro de las aplicaciones, sistemas operativos, firewalls, antivirus etc. Y de cómo sacar partido de ella para detectar fallas en la seguridad. También se habla sobre los sistemas de detección y prevención de intrusiones y sobre los test de penetración para analizar y encontrar fallos en la red.

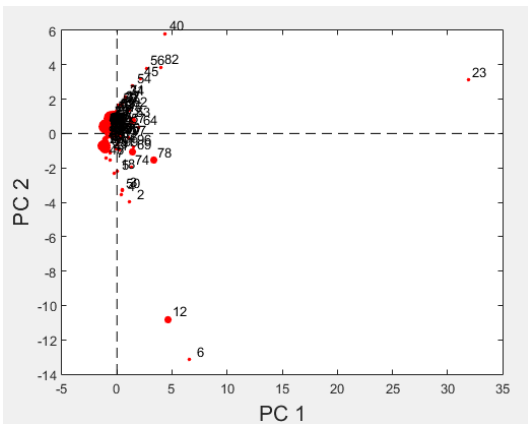


Fig. 2: Ejemplo de ScorePlot

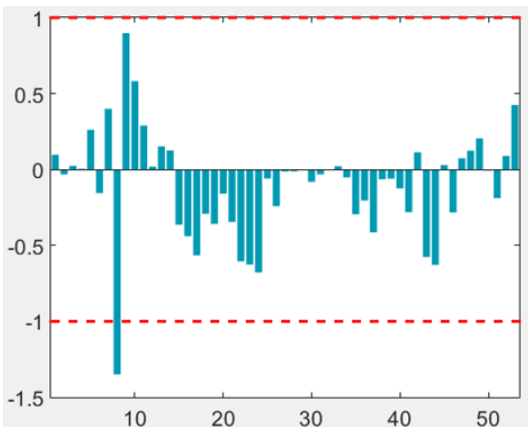


Fig. 3: Ejemplo de oMEDA

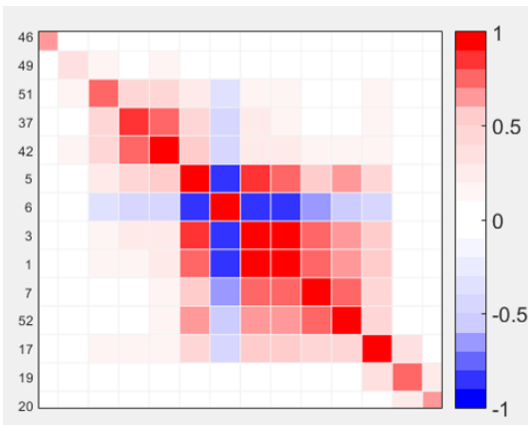


Fig. 4: Ejemplo de MEDA

IV. GUIDE EN MATLAB

En el documento se desarrolla un capítulo a modo de tutorial comentando los distintos elementos de los que dispone Matlab para el desarrollo de interfaces gráficas de usuario que se utilizarán para el desarrollo de la interfaz final. En este capítulo se explican las opciones iniciales de creación de las GUIs así como cada uno de los elementos que las integran para ayudar al lector a la comprensión del entorno. También se explica el manejo de los elementos gráficos desde el código y se muestra un ejemplo sencillo de aplicación para iniciarse en el desarrollo de GUIs.

V. ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE LA INTERFAZ

A. Análisis y especificación

En este capítulo se especifican los actores que interactuarán con la interfaz, los requisitos tanto funcionales como no funcionales y de información. También se desarrollan los casos de uso que van a componer el software y los diagramas que muestran su funcionamiento.

B. Diseño

Aquí se explica la parte de MEDA-Toolbox que se va a utilizar en la interfaz, los ficheros de los que va a estar compuesta, la arquitectura interna de los paquetes y se diseña el prototipo de la interfaz. El prototipo diseñado es el que se muestra en la Fig. 5.

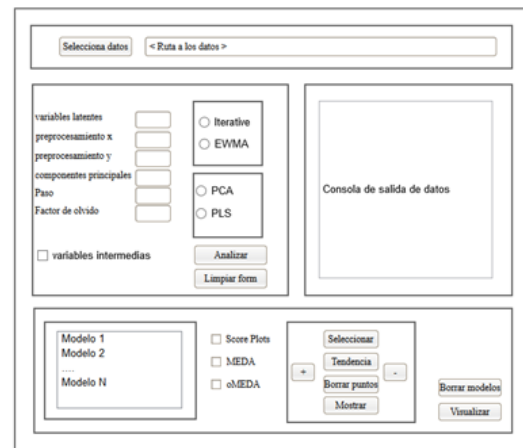


Fig. 5: Prototipo interfaz

C. implementación

Este capítulo trata el desarrollo de la interfaz comentando todas las funciones implementadas y todos los apartados de la GUI final con cada uno de los elementos que la componen. La versión final de la interfaz gráfica de usuario desarrollada se muestra en la Fig 6.

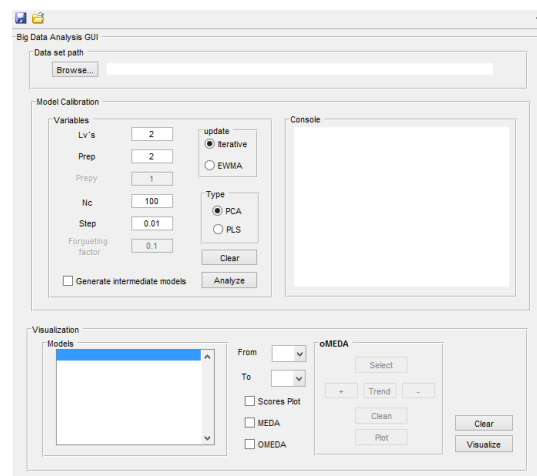


Fig. 6: Interfaz gráfica de usuario final.

VI. CASO DE ESTUDIO

Para demostrar el funcionamiento de lo desarrollado se realiza un caso de estudio orientado a la detección de ataques en red. Los datos que se utilizan han sido previamente anonimizados. En el caso de estudio se prueba el funcionamiento de la aplicación detectando el instante temporal en el que se produce el ataque, los puertos de origen y destino del ataque y los tamaños de paquetes que genera el mismo. No se muestran las IPs que participan en el ataque debido a que los datos fueron anonimizados, pero en el caso de estar en un contexto diferente se podría obtener toda la información sin problema. En la Fig. 7 se puede ver como se marcan los outliers (Observaciones que se salen de los valores "comunes") que van a ser estudiados en profundidad para entender porqué lo son.

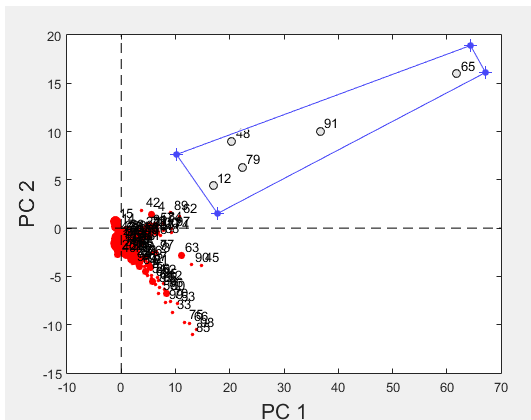


Fig. 7: Observaciones de interés.

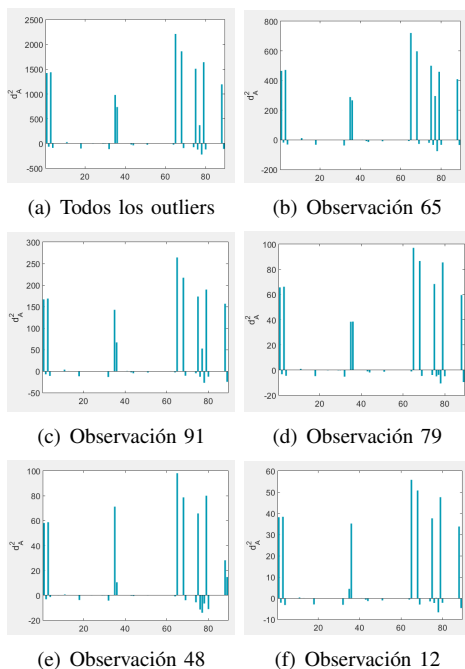


Fig. 8: Omeda de los outliers

La Fig. 8 muestra los valores de las variables que identifican cada una de las observaciones seleccionadas. El comportamiento de cada observaciones viene determinado por las

variables que la componen, por lo que en los gráficos oMEDA se pueden apreciar a simple vista que variables identifican cada observación.

En este caso se puede ver que las variables con mayor valor son comunes a todas las observaciones marcadas por lo que se entiende que son dichas variables las que producen la anomalía.

Para averiguar el instante temporal en el que se producen las anomalías se analizan uno a uno los ficheros de datos en lugar de todos en conjunto como se ha hecho hasta esta punto.

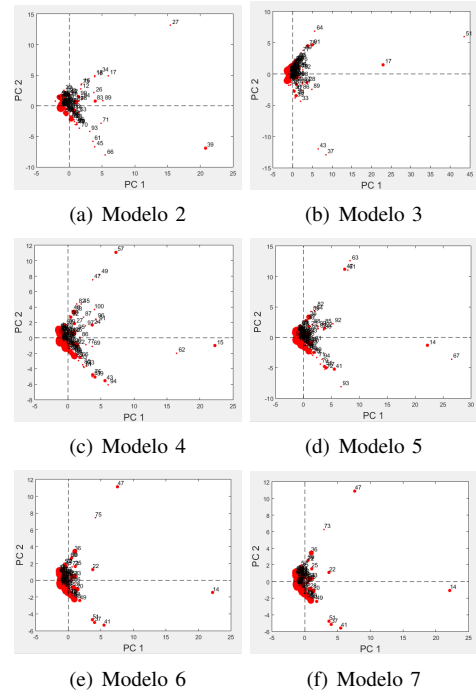


Fig. 9: Score Plots entre modelos 2 y 7.

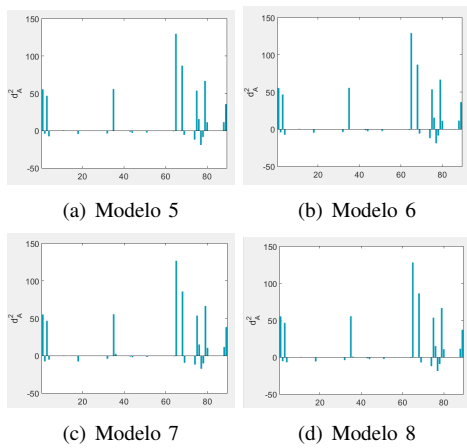
En la Fig. 9 se puede ver la evolución de los ScorePlot desde el fichero 2 hasta el 7. Se puede apreciar como a partir del modelo 5 aparece una observación con la etiqueta 14 que se mantiene hasta los últimos modelos. Ahora solo resta comprobar que dicha observación tiene las mismas variables notables que se observaron en el caso del modelo completo para determinar el instante del ataque.

Se puede observar en la Fig. 10 como las variables que predominan en dicha observación son prácticamente las mismas que se podían observar en el oMEDA de la Fig 8. Las variaciones que se hayan podido observar son debidas al tratamiento de los datos utilizado en cada uno de los métodos.

VII. CONCLUSIONES Y TRABAJO FUTURO

A. Conclusiones

El objetivo principal del proyecto era facilitar el manejo de MEDA-Toolbox para su utilización con Big Data. La interfaz gráfica desarrollada facilita el manejo y pone al alcance de cualquiera el uso de las herramientas de MEDA. Por otra parte a lo largo de todo el trabajo me he dado cuenta de la potencia que tiene la Toolbox tanto para la reducción del tamaño del problema como para la visualización mediante las gráficas, y como a simple vista con unos datos correctamente tratados se



Pablo Sánchez Robles Nacido en 1987 en Granada(España) obtuvo el grado en ingeniería informática con mención en Tecnologías de la Información en la universidad de Granada en septiembre de 2015. Sus principales intereses son el estudio de la seguridad en red y el desarrollo de nuevas tecnologías.

Fig. 10: oMEDA entre modelos 5 y 8 sobre la observación 14.

pueden ver las anomalías y patrones que se encuentran en los mismos

B. Trabajo futuro

Como trabajo futuro en lo que a MEDA-Toolbox se refiere se podrían incluir técnicas de inteligencia artificial como algoritmos bioinspirados para complementar la potencia de las técnicas EDA. Para ampliar el uso de la Toolbox se podría crear un servidor donde centralizar el cálculo y tratamiento de los datos y desarrollar unas aplicaciones que se conecten a dicho servidor para recibir las salidas gráficas. Esto se podría hacer mediante aplicaciones de escritorio, aplicaciones móviles o web.

AGRADECIMIENTOS

A mis padres, mi hermana y mi cuñado por aguantarme día a día como lo han hecho y por estar siempre ahí para todo lo que me ha hecho falta.

A mis amigos por ser la vía de escape en los momentos difíciles y no fallarme en ningún momento.

A mis compañeros de la ETSIT por darme todo cuando se les necesita y por hacer del trabajo en equipo un sistema de vida.

A mis abuelos por inculcarme el valor por el estudio desde que tengo uso de razón. Especialmente a todos los que ya no están porque la marca que han dejado jamás se perderá.

REFERENCIAS

- [1] José Camacho Páez, "Chemometrics and intelligent laboratory systems", 2011.
- [2] José Camacho Páez, "Latent Structures-based models", 2012.
- [3] Elena Jiménez Mañas, "PFC. Herramientas para la detección de ataques en tráfico de red", 2012.
- [4] José Camacho Páez, "Tackling the Big Data 4 Vs for Anomaly Detection". 2014.
- [5] Raffael Marty, "Applied Security Visualization". 2008.
- [6] Pablo González Pérez, Germán Sánchez Garcés, José Miguel Soriano de la Cámara, "Pentesting con Kali". 2013.
- [7] Pablo González Pérez, Chema Alonso, "Metasploit para pentesters", 2012.

Índice de autores

P. Andrés Maldonado	61	P. Peña Martínez.....	37
F.J. Cuberos Muñoz	17	C.A. Prieto Sánchez	55
D. García Sánchez	23	M. Sánchez López.....	49
A. González Garrido	3	P. Sánchez Robles	79
I. Herrera López.....	43	I. Vico Triviño.....	29
J. López Arredondo	67	A.O. Villegas Morcillo	11
F. López Pérez	73		

Índice de tutores

P. Ameigeiras Gutiérrez.....	61	J. Navarro Ortiz	3, 43, 49
M.C. Benítez Ortúzar	17, 23, 29	A.M. Peinado Herreros	11
J. Camacho Páez	79	J.J. Ramos Muñoz.....	37, 55
L. García Martínez	17, 23, 29	R.A. Rodríguez Gómez	73
P. García Teodoro	67	V. Sánchez Calle	11
J.M. López Soler.....	37,	J.C. Segura Luna.....	3



ISBN-13: 978-84-617-5449-6

