

# SR-DMM: A SDN-Based DMM Solution for Future Mobile Networks

J. Mendoza, J. Carmona, J. González, J. Calle, and D. Cortés

**Abstract**—The exponential growth in the number of mobile subscribers, as well as the data traffic generated by them is showing a major challenge in wireless Internet. Existing mobile networks are starting to be insufficient in meeting this demand, in part due to their inflexible and expensive equipment as well as complex and non-agile control plane. In this context, in which the heterogeneity on the access is even more extended, the mobility support becomes more crucial. Thus, the mobility management mechanisms need to evolve to fit with the cutting-edge technologies that are going to rule next generation mobile networks. A key enabler to 5G will be SDN (Software-Defined Networking), which brings a natural solution to decouple the network control plane from the data plane, allowing higher flexibility and programmability to the network. Based on SDN and following the recommendations of the DMM (Distributed Mobility Management) approach, this article presents the SR-DMM solution (SDN Redirection DMM), a mobility management mechanism that takes full advantages of the softwarization capabilities of the network. This proposal avoids the use of IP-IP tunnels and it adds the dynamic flow management capability provided by SDN. In order to analyse the performance of SR-DMM, an analytical model is developed to compare it with NB-DMM (Network-based DMM), one of the main DMM solutions. Additionally, SR-DMM performance is also evaluated with an experimental testbed. The results allow to measure the handover latency in real scenarios and numerical investigations, also shows that SR-DMM achieves better efficiency in terms of signalling and routing cost than NB-DMM solution.

**Index Terms**—SDN, OpenFlow, Distributed Mobility Management, SDN Redirection DMM.

## I. INTRODUCCIÓN

A lo largo de los últimos años, las tecnologías de comunicaciones móviles están continuamente evolucionando y provocando cambios en la sociedad, hasta el punto de convertirse en la primera forma de acceso de un gran número de personas a las TIC (Tecnologías de la Información y las Comunicaciones).

J. M. Mendoza, Grupo de Investigación de Ingeniería Telemática Aplicada y Comunicaciones Avanzadas (GITACA), jmendozah@unex.es.

J. Carmona, Departamento de Ingeniería de Sistemas Informáticos y Telemáticos, Universidad de Extremadura, Cáceres, España, jcarmon@unex.es.

J. L. González, Centro Extremeño de Investigación, Innovación Tecnológica y Supercomputación (CénitS), Cáceres, España, joseluis.gonzalez@cenits.es.

J. Calle, Centro Extremeño de Investigación, Innovación Tecnológica y Supercomputación (CénitS), Cáceres, España, jesus.calle@cenits.es.

D. Cortés, Centro Extremeño de Investigación, Innovación Tecnológica y Supercomputación (CénitS), Cáceres, España, david.cortes@cenits.es.

Actualmente, el consumo de los nuevos servicios y aplicaciones ofertados por los operadores están motivando una constante demanda de mayores tasas de datos por parte de los suscriptores móviles. Al mismo tiempo, la proliferación de dispositivos como *smartphones* y *tablets* generan un crecimiento masivo de datos móviles en la red. Tal es así, que en el año 2016 el tráfico móvil experimentó un crecimiento del 63 %, alcanzándose un consumo mensual de datos de 7 exabytes, de los cuales el tráfico de vídeo representa un 60 %. Además, las estimaciones indican que este incremento va a continuar y, se espera que en 2021 se consuman 49 exabytes de datos, siendo 11,6 billones los dispositivos conectados a Internet [1].

Estos datos reflejan la importancia de las redes móviles en los próximos años. Sin embargo, estas redes están basadas en dispositivos especializados integrados con sistemas operativos propietarios, lo que las hace dependientes de los proveedores. Este hecho imposibilita la fácil administración de los dispositivos y hace que las redes móviles sean difícilmente escalables [2].

En este escenario, resulta fundamental el proporcionar un servicio continuo al usuario móvil, mientras este cambia su punto de conexión con la red (proceso de *handover*). La gestión de la movilidad durante el desplazamiento del terminal móvil o MN (*Mobile Node*) cobra pues especial importancia, sobre todo en despliegues densos y heterogéneos donde coexisten distintas tecnologías como LTE, Wimax, UMTS y WLAN [3]. La sesión de movilidad debe estar soportada además por soluciones dinámicas, flexibles y escalables.

Como respuesta a esta necesidad de movilidad, la industria y la academia han estado desarrollando protocolos de gestión de la movilidad que permitan al MN comunicarse de forma ininterrumpida con el CN (*Correspondent Node*). Esta evolución ha llevado al desarrollo de protocolos centralizados, donde la movilidad estaba localizada en un nodo específico de la red que realizaba las labores de ancla, o distribuidos donde las funciones de movilidad están repartidas entre diferentes nodos de la red.

Este artículo presenta SR-DMM (*SDN Redirection Distributed Mobility Management*), un mecanismo de gestión de la movilidad que se basa en las capacidades de SDN (*Software Defined Networking*) para ofrecer una solución de movilidad transparente a los usuarios en redes de próxima generación. Además, se ha analizado el rendimiento del mecanismo propuesto, comparándolo con la solución NB-DMM (*Network-Based Distributed Mobility Management*) [4].

El resto del artículo está organizado de la siguiente forma. La sección II presenta el estado del arte y la evolución de los

protocolos de gestión de la movilidad. La sección III recoge la especificación de SR-DMM, detallando la señalización empleada para gestionar la movilidad de los MNs. La sección IV muestra la comparativa analítica entre NB-DMM y SR-DMM. Por último, en la sección V se describe la implementación de la solución propuesta. En este apartado también se comprueba el rendimiento de SR-DMM en un prototipo experimental, prestando especial atención a un parámetro crítico en estos protocolos como es la latencia de *handover*.

## II. ESTADO DEL ARTE

Durante los últimos años, la gestión de la movilidad basada en redes IP ha sido una de las líneas de investigación más activas en el campo de las comunicaciones. Gracias a la estandarización por parte del IETF (*Internet Engineering Task Force*) de los protocolos MIPv6 (*Mobile IPv6*) [5] y PMIPv6 (*Proxy Mobile IPv6*) [6], se introdujo la capacidad de dar continuidad a las sesiones abiertas de los MNs con los CNs. Este tipo de soluciones presentan ciertas limitaciones como encaminamiento subóptimo, problemas de escalabilidad derivados de la centralización del tráfico, y limitaciones en cuanto a fiabilidad, debido a que el agente de movilidad puede ser un punto único de fallo [7]. Por todo ello, el IETF creó un nuevo grupo de trabajo denominado DMM (*Distributed Mobility Management*) *Working Group* cuyo objetivo es el de plantear soluciones que resuelvan los problemas de las soluciones centralizadas, distribuyendo las funciones de movilidad entre diferentes nodos de la red. La principal diferencia de DMM es la de disgregar las funciones de anclaje y localización de los terminales móviles al borde de la red de acceso, más cerca del MN. Esto permite distribuir el plano de datos y las funciones de movilidad a lo largo de distintas redes. De hecho, dependiendo de las diferentes implementaciones de DMM, se pueden distinguir soluciones DMM basadas en el nodo, en las cuales el nodo móvil es parte activa de la gestión de la movilidad y participa en la gestión de los túneles IP-en-IP para mantener las sesiones activas. Esta implementación está basada en Mobile IP y se conoce como *Host-based DMM*. Otra de las implementaciones propuestas de DMM son aquellas basadas en la red, en las que el nodo móvil no influye en la gestión de la movilidad y es la red la encargada de proporcionar los mecanismos para mantener las sesiones abiertas mientras el nodo móvil se mueve. Esta última opción de DMM está basada en PMIPv6 y se denomina *Network-based DMM*, existiendo diferentes implementaciones como [4] y [8].

Por otra parte, la tecnología SDN ha aparecido como un nuevo paradigma de red en el cual el hardware de reenvío es separado de las decisiones de control, lo que simplifica la administración de la red y habilita la evolución de la misma [9]. El plano de control queda centralizado en una entidad conocida como controlador de la red, que gestiona el plano de datos de la misma a través de protocolos como OpenFlow [10], permitiendo que la red sea programable a través de software. La tecnología SDN está siendo considerada como la base del plano de control de las arquitecturas de las redes móviles de próxima generación [11], así como de distintas soluciones de gestión de

la movilidad [12, 13]. SDN permite la configuración, mantenimiento y administración de la infraestructura ante escenarios con multitud de dispositivos y tipos de servicios sin necesidad de modificar la pila de protocolos de los nodos que intervienen en la red. Además, permite una gestión dinámica y precisa de los flujos de paquetes, ofreciendo optimización de recursos ante las distintas necesidades de las aplicaciones, que pueden tener diferentes requerimientos en cuanto a características y calidad de servicio [14].

Adicionalmente a SDN, cabe destacar otros mecanismos que participarán en este proceso de softwarización en las redes de próxima generación, como NFV (*Network Function Virtualization*) o *Network Coding*, que introducen cambios relevantes en la forma de gestionar la red y permiten alcanzar los requisitos de baja latencia y alta fiabilidad requerida por los servicios que se ofrecerán en las futuras redes 5G [15].

## III. PROPUESTA SR-DMM

El objetivo de gestionar la movilidad de los MNs a través de la tecnología SDN es el de conseguir soluciones donde el plano de control está centralizado y separado del plano de datos, que está distribuido. La gestión de la movilidad se proporciona como un servicio desarrollado en forma de aplicación SDN, que se ejecuta en el controlador de red, y administra un plano de control formado por hardware genérico. De este modo, el propósito principal de nuestra propuesta SR-DMM es el de proporcionar flexibilidad y escalabilidad a las redes móviles, aprovechando las capacidades de la arquitectura SDN para ofrecer la gestión de la movilidad como un servicio y aportando precisión a nivel de flujo de datos.

El controlador centralizado tiene una visión global de la red, de la cual se beneficia la aplicación SR-DMM a través del canal OpenFlow. De esta forma, el servicio de movilidad conoce el estado global del plano de datos del dominio DMM para tomar decisiones. Por otra parte, a través de la interfaz de OpenFlow, el controlador permite dotar a los conmutadores del borde de la red de la capacidad de anclar los flujos de paquetes de los terminales móviles. La aplicación SR-DMM configura las tablas de flujo de los OFSwitches (OpenFlow Switches) con funcionalidad de ancla de movilidad, proporcionándoles además capacidades de descubrimiento de vecinos [16] y control de acceso. Ya que el controlador de red conoce la localización de los MNs a través de su visión global, la propuesta permite evitar la sobrecarga introducida por el *tunneling* entre anclas de movilidad. Por el contrario, SR-DMM realiza redirección de los flujos de paquetes.

La Figura 1 muestra la arquitectura general de nuestra propuesta, donde la red de acceso está formada por conmutadores OpenFlow, que son gestionados por el controlador de red. En este caso un conmutador situado en el borde de la red (OF-Switch1) está anclando un flujo de paquetes iniciado cuando el MN se encuentra situado en la red inalámbrica a la que sirve. El terminal móvil continúa su desplazamiento por el dominio DMM, adquiriendo nuevos prefijos de IPv6 las redes visitadas. De una forma parecida a la de la Figura 1, la solución NB-DMM define dos entidades principales llamadas MAAR (*Mobility Anchor and Access Router*) y CMD (*Central Mobility Database*). Donde el MAAR

actúa como primer salto IP, además de gestionar la movilidad y anclar los prefijos IPv6 que advierte; mientras que el CMD la entidad que almacena la tabla de vínculo de los MNs.

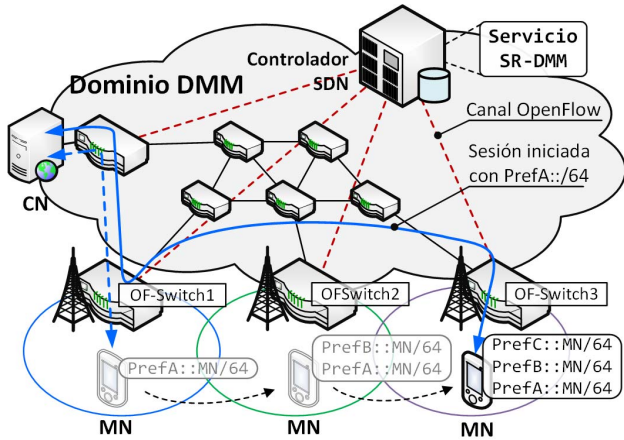


Fig. 1. Arquitectura general de la propuesta SR-DMM.

Por otra parte en NB-DMM, cuando se produce un evento de *handover*, el MAAR en cuestión tuneliza el tráfico IP y lo reenvía al nuevo agente de movilidad al cual se ha desplazado el MN. En este caso el CMD es la entidad análoga al controlador SDN, mientras que los MAARs son los dispositivos situados al borde de la red de acceso, al igual que los OFSwitches en SR-DMM.

A. Operación de Registro Inicial

Cuando un MN se conecta por primera vez a un OFSwitch del dominio DMM, envía un mensaje RS (*Router Solicitation*) a este dispositivo. Este OFSwitch, según su tabla de flujos, encapsula el RS en un mensaje OpenFlow *Packet\_In* y lo envía al controlador de red. El controlador de red autentifica y localiza al MN, almacenando un vínculo con su dirección MAC (*Media Access Control*), su identificador MN-ID y el ancla al que está actualmente asociado el terminal. Seguido de esto el controlador de red obtiene el prefijo de red asociado, formando un mensaje RA (*Router Advertisement*) con dicho prefijo, encapsulándolo en un mensaje OpenFlow *Packet\_Out* y lo envía al conmutador. En la recepción del RA, el MN autoconfigura su dirección IPv6, quedando anclado con este prefijo al OFSwitch.

B. Operación de Handover

Con respecto a la operación de *handover*, el servicio SR-DMM permite detectar, al recibir el controlador SDN un nuevo RS, la localización del MN. En esta ocasión al llegar el *Packet\_In* al controlador de red y verificar que el OFSwitch actual es distinto del anterior, obtiene tantos prefijos de red como OFSwitches previos haya visitado el MN. A través de estos prefijos de red el terminal móvil podrá autoconfigurar sus direcciones IPv6, una vez haya recibido el RA por parte de su nuevo agente de movilidad.

SR-DMM instala una regla de flujo en cada conmutador previo donde están anclados los flujos iniciados por el MN. La regla instalada permite redirigir los paquetes al agente de

movilidad actual del terminal móvil, estableciendo como dirección IPv6 destino la formada con el prefijo de red asociado al OFSwitch actual y el identificador MN-ID del MN en el dominio DMM. La regla de flujo instalada en el agente de movilidad actual permite restablecer la dirección IPv6 original de los paquetes pertenecientes a las sesiones previas, de modo que el MN los reciba y mantenga la continuidad de las sesiones de forma transparente. Además, dado que las reglas instaladas en los OFSwitches expiran en un periodo de tiempo determinado si no existe el *match* correspondiente y este evento es recibido por el controlador, las sesiones que expiran no tendrán soporte de movilidad. El flujo de los mensajes de control intercambiados en SR-DMM se muestra en la Figura 2.

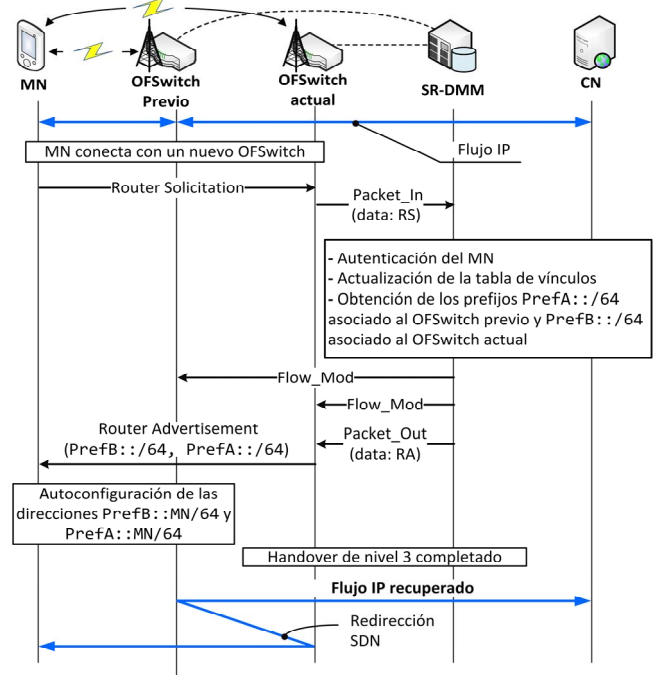


Fig. 2. Mensajes de control intercambiados en el handover de SR-DMM en la operación de *handover*.

Según los procesos descritos anteriormente, la solución permite de forma transparente al usuario, y sin implicar al MN en la señalización de control, la redirección del tráfico de datos desde cada switch OpenFlow previo hasta la nueva ancla según la localización del terminal móvil. Este proceso es análogo al funcionamiento de NB-DMM pero sin la necesidad de sobrecargar la red con cabeceras de control introducidas en los túneles IP-IP. Mientras que otras implementaciones como [12] establecen la ruta óptima aplicando reglas de flujo a todos los OFSwitches que conforman el camino entre CN y MN, con SR-DMM sólo es necesario instalar nuevas reglas en los conmutadores de borde visitados por el MN. Esto se debe a que los caminos hasta cada red destino son calculados previamente y de forma dinámica por el servicio de movilidad, alcanzándose la ruta óptima entre el OFSwitch previo y el actual, como en NB-DMM. Además, a diferencia de [13], en SR-DMM no es necesario utilizar ningún agente de movilidad especializado. Todos los dispositivos del dominio SR-DMM son genéricos, y su funcionalidad viene determinada por el controlador de red,

que elige si estos actúan, o no, como agentes de movilidad.

#### IV. ANÁLISIS Y EVALUACIÓN DE RENDIMIENTO

En esta sección se describe el modelo analítico utilizado para la evaluación del rendimiento de SR-DMM y NB-DMM. A través de este modelo, se analiza el coste de señalización de ambos protocolos de gestión de la movilidad y el esfuerzo que debe realizar la red a la hora de transmitir un paquete desde el CN hasta el MN.

##### A. Análisis de las Propuestas

El dominio sobre el que se realiza el análisis está compuesto por  $N$  celdas conectadas a diferentes *routers* de acceso, que serán los primeros nodos con capacidad IP de la red. La topología de la red se describe en la Figura 3.

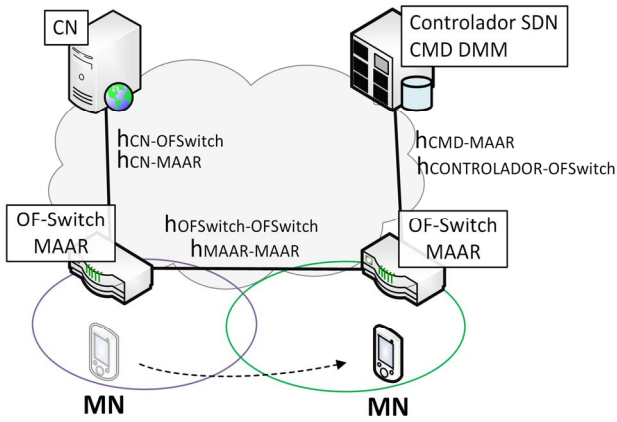


Fig. 3. Topología de la red usada en el análisis.

Las propuestas NB-DMM y SR-DMM son evaluadas a través del modelado analítico para calcular el coste de señalización y el coste de transporte de los paquetes. Además, la propuesta SR-DMM ha sido estudiada con un *testbed* real que será descrito en las siguientes secciones. NB-DMM ha sido elegida para la comparativa debido a la similitud de arquitectura con respecto a SR-DMM en tanto en cuanto existe un elemento centralizador en ambas que gestiona la movilidad y el resto de funciones son distribuidas en el borde de la red.

El modelo analítico ha sido desarrollado usando el *framework* descrito en [17]. El coste de transmisión de paquetes en una red IP es proporcional a la distancia en número de saltos entre el origen y los nodos destino.

Analizando la señalización de las propuestas estudiadas, se puede definir el coste de actualización de registro durante una sesión,  $C_u$ , como la carga de tráfico acumulada en el intercambio de mensajes de señalización. Para cada movimiento a una nueva subred, se envían un mensaje *Proxy Binding Update* (PBU)/*Proxy Binding ACK* (PBA) para la propuesta basada en DMM. Además, hay que tener en cuenta la actualización de la tabla de vínculos durante el tiempo de vida del prefijo, así como eliminar el prefijo una vez no exista ninguna sesión activa; es decir, los procesos de *binding refresh* y *deregistration*.

En el caso de SR-DMM, los mensajes de señalización

intercambiados son propios de SDN para actualizar la ruta entre agentes de movilidad a través de mensajes *FlowMod*. Además, los prefijos inactivos son eliminados al expirar los temporizadores de las entradas de la tabla de flujo de movilidad de los OFSwitches, no siendo necesario los mecanismos de refresco como el proceso de *binding refresh*.

Las siguientes expresiones desarrollan el coste de señalización del desplazamiento del MN. La fórmula se puede descomponer en tres factores, la señalización debido a un *handover*, el borrado de un prefijo inactivo y el mantenimiento de un prefijo activo en el CMD/Controlador. Las expresiones que resumen el comportamiento se describen a continuación; donde se define  $h_{(x,y)}$  como la distancia en número de saltos IP entre dos nodos de la red  $(x, y)$ ,  $\mu_c$  como el ratio de cruce de la celda en segundos,  $N_{pr}$  como el número de prefijos activos de los que dispone el MN (número de MAARs/OFSwitches por los que ha pasado, que mantienen alguna sesión abierta),  $S(\bullet)$  como el tamaño en bytes de los paquetes y  $R_{BCE}$  la tasa de refresco de la caché de vínculos.

$$C_u(NB - DMM) = \mu_c \cdot [(S_{PBU} + S_{PBA}) \cdot h_{CMD-MAAR} \cdot (N_{pr} + 1) + 2 \cdot (S_{PBU} + S_{PBA}) \cdot h_{CMD-MAAR} + R_{BCE} \cdot (S_{PBU} + S_{PBA}) \cdot h_{CMD-MAAR}] \quad (1)$$

$$C_u(SR - DMM) = \mu_c \cdot [(N_{pr} + 1) \cdot 2 \cdot S_{FlowMod} \cdot h_{CONTROLADOR-OFswitch} + (S_{PIN} + S_{RS} + S_{POUT} + S_{RA}) \cdot h_{CONTROLADOR-OFswitch}] \quad (2)$$

Asimismo  $N_{pr}$  puede ser calculado como en [17], siendo  $\delta$  el tiempo medio de vida en segundos de los prefijos activos mientras el MN permanece en una red visitada:

$$N_{pr} = \frac{\mu_c}{\delta} \quad (3)$$

Una vez la sesión ha sido notificada a los nodos de la red, el MN se comunica con el CN a través del nodo que sirve de ancla para la sesión. El coste de entrega de paquetes en bytes/segundo  $P_{cost}$  se define entonces como:

$$P_{cost}(NB - DMM) = N_{p/s} \cdot [(N_{pr} - 1) \cdot (S_{DATA} + S_{IP}) \cdot h_{MAAR-MAAR} + (S_{DATA} \cdot h_{CN-MAAR}) + (S_{DATA} \cdot h_{MN-MAAR})] \quad (4)$$

$$P_{cost}(SR - DMM) = N_{p/s} \cdot [(N_{pr} - 1) \cdot (S_{DATA} \cdot h_{OFswitch-OFswitch} + (S_{DATA} \cdot h_{OFswitch-CN}) + (S_{DATA} \cdot h_{MN-OFswitch}))] \quad (5)$$

Siendo  $N_{p/s}$  el número medio de paquetes por segundo enviados por cada flujo.

En primer lugar se definen los mensajes de señalización de los diferentes protocolos analizados; PBU y PBA de DMM, *Packet In* (PIN), *Packet Out* (POUT) y *FlowMod* de SDN, y finalmente, RS y RA de IPv6. Por último, Data e IP denotan el tamaño de los paquetes de datos y de la cabecera IP-en-IP respectivamente.

### B. Evaluación de Rendimiento

En esta subsección se comparan las propuestas descritas anteriormente de forma analítica. Los parámetros y valores por defecto usados en este análisis son similares a los usados en los artículos [17-18], con  $E[\mu_c]$  como el tiempo medio de estancia en la celda y  $E[\delta]$  como el tiempo medio de vida de los prefijos activos:

$E[\mu_c]=[50-1800]$  seg,  $h_{CMD-MAAR}=h_{Controlador-OFSwitch}=2$  saltos,  $h_{MAAR-MAAR}=h_{OFSwitch-OFSwitch}=2$  saltos,  $h_{CN-MAAR}=h_{CN-OFSwitch}=5$  saltos,  $h_{MN-MAAR}=h_{MN-OFSwitch}=1$  salto,  $E[\delta]=60$  seg.

La Figuras 4 y 5 muestran respectivamente la comparativa del coste de señalización y el coste de entrega de paquetes para los protocolos analizados. Como se puede observar, el coste de gestionar la movilidad para la propuesta SR-DMM es superior a la presentada por NB-DMM debido a que esta última está basada en protocolos IP. SR-DMM al ser una propuesta basada en SDN, emplea primitivas OpenFlow y por tanto el intercambio de mensajes se realiza a través de TCP. Además, SR-DMM se beneficia de las opciones que incorpora SDN de forma que no requiere que se mantenga el refresco de la caché de vínculos tal y como incorpora DMM. Debido a esto, cuando el tiempo de estancia en la celda se incrementa, la señalización introducida en la red se decremanta, incluso por debajo de la señalización de DMM, basada en mensajes IP.

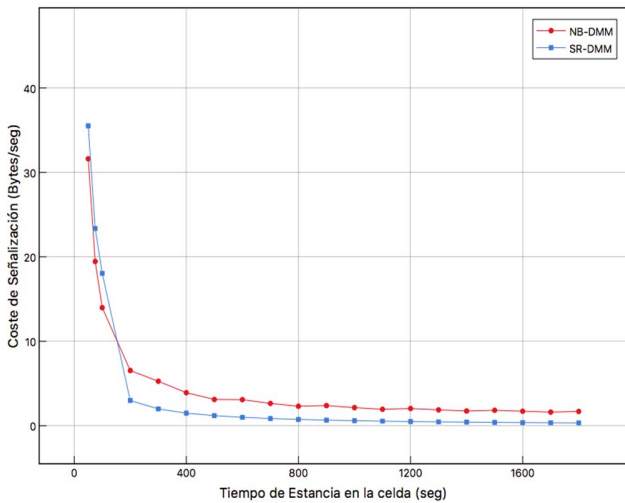


Fig. 4. Coste de señalización en función del tiempo de estancia en la celda.

### V. EVALUACIÓN EXPERIMENTAL

En esta sección se realiza un análisis experimental comparativo desplegando un escenario real con una implementación de SR-DMM. El escenario usado en la evaluación es presentado en la Figura 6. El plano de control es implementado usando el *framework* de SDN Ryu, empleando como interfaz de control el protocolo OpenFlow 1.3 [19]. Asimismo, el servicio SR-DMM es una aplicación Python que se ejecuta en el controlador de red. Por otra parte, el plano de datos que da servicio a los nodos móviles está formado por tres OFSwitches, que tienen asociado un prefijo de red de 64 bits calculado por SR-DMM, y cuyas tablas de flujo están inicialmente vacías. Los conmutadores OpenFlow disponen

además una interfaz inalámbrica funcionando en modo punto de acceso.

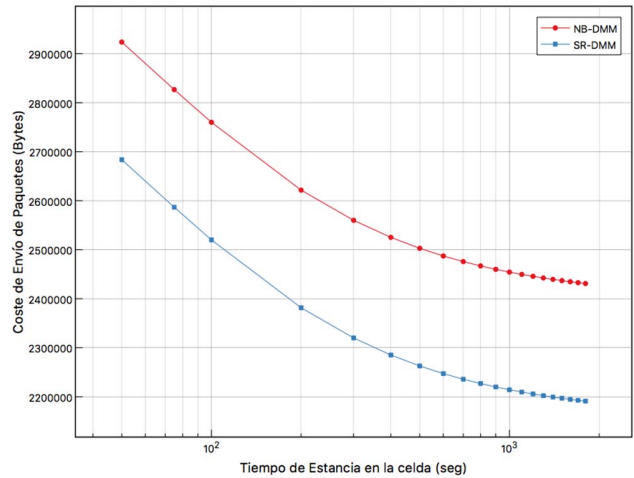


Fig. 5. Coste de entrega de paquetes en función del tiempo de estancia en la celda.

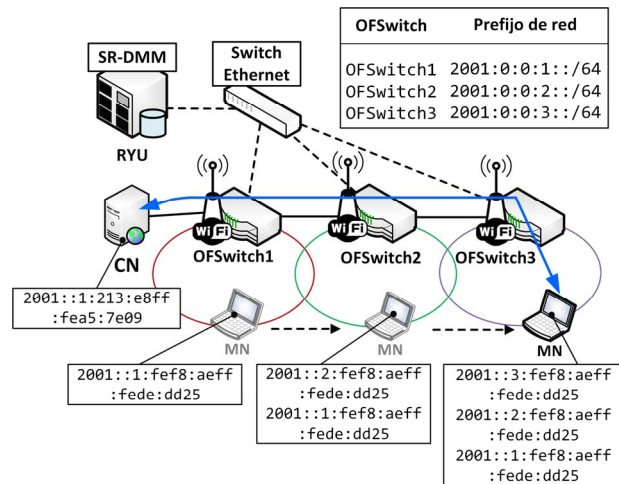


Fig. 6. Escenario utilizado en la evaluación experimental de SR-DMM.

En el escenario experimental cada OFSwitch propaga una WLAN (IEEE 802.11g) distinta. Se utilizan además dos hosts extremos de la comunicación, CN a modo de nodo correspondiente y MN como nodo móvil. Al ser SR-DMM una solución de movilidad basada en la red, ninguno de los dispositivos utilizados requiere modificación en su pila de protocolos.

#### A. Análisis de la Latencia de Handover

La latencia de *handover* mide el tiempo transcurrido entre la recepción del último paquete en la asociación del MN con el agente de movilidad anterior, y la recepción del primer paquete del mismo flujo recibido cuando el MN se asocia con la siguiente ancla de movilidad. Esto permite conocer el tiempo en el que no se están recibiendo paquetes, y es una medida directamente proporcional a la cantidad de paquetes perdidos durante el *handover*.

En el escenario, el terminal MN se asocia con el OFSwitch1, dispositivo al que también está conectado el CN. En ese instante, el CN inicia una sesión de paquetes usando UDP con



el MN. Posteriormente, el MN realiza un *handover*, quedando anclado al OFSwitch2 a través de su AP y provocando un primer evento de *handover*. Este proceso se repetirá con el tercer OFSwitch. Gracias al servicio de gestión de la movilidad implantado en el controlador de red, los conmutadores OpenFlow pueden anclar los flujos de paquetes y redirigirlos hasta el OFSwitch que sirve la conexión en cada momento al MN, de modo que la sesión iniciada con el CN permanece activa.

Este experimento se ha repetido 300 veces, con lo que se han realizado unos 600 *handovers*. La Figura 7 muestra los resultados de la latencia de *handover* usando una función de distribución acumulada, CDF (*Cumulative Distribution Function*). Los resultados obtenidos a partir de esta CDF han sido recogidos en la Tabla I.

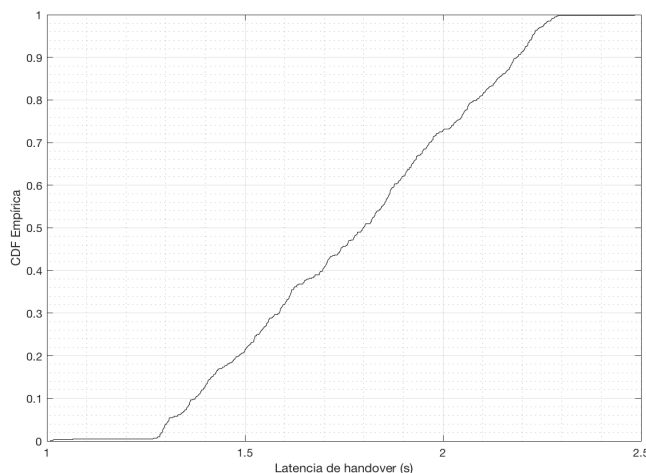


Fig. 7. Latencia de handover experimental de SR-DMM.

TABLA I  
RESULTADOS EMPÍRICOS DE LATENCIA DE HANDOVER  
OBTENIDOS A PARTIR DE LA CDF

	Mínimo	Máximo	Media	Desviación Estándar	Percentil 95
<b>Tiempo (s)</b>	1,009	2,485	1,780	0,296	2,226

A la vista de los resultados de la Tabla I, se puede decir que el tiempo medio empleado en el *handover* del MN en el estudio realizado es de 1,78 segundos. Por lo tanto, durante ese tiempo el terminal móvil dejará de recibir paquetes, hasta que se restaure la sesión a través del protocolo de gestión de movilidad, en este caso SR-DMM.

### B. Análisis de la Secuencia TCP en la Comunicación

La segunda prueba analiza el comportamiento de TCP en la comunicación usando SR-DMM. Para ello se ha realizado una experimentación donde el CN actúa como servidor de *streaming* empleando TCP como protocolo de transporte, mientras que el MN recibe la sesión de *streaming*. Debido a la naturaleza de TCP, cualquier paquete perdido durante la fase de *handover* será recuperado extremo a extremo. La Figura 8 muestra la representación de los números de secuencia TCP para la transferencia realizada entre el CN y el MN.

Tal y como muestra la Figura 8, el número de secuencia de los paquetes recibidos por MN se va incrementando hasta los

20 segundos de transferencia, donde se produce el primer *handover*.

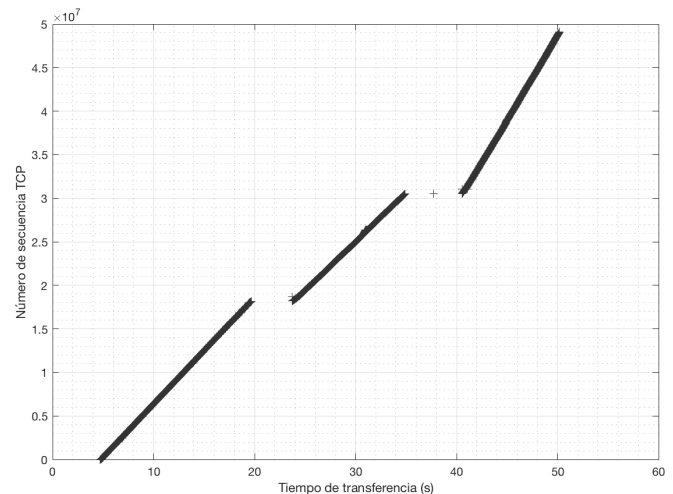


Fig. 8. Secuencia TCP experimental obtenida con SR-DMM.

El flujo de datos se recupera tras, aproximadamente, 4 segundos, y el número de secuencia vuelve a aumentar hasta que se produce el segundo *handover*, en torno a los 35 segundos de la comunicación. El flujo vuelve a ser recibido por el nodo móvil aproximadamente a los 41 segundos de sesión, tras lo cual se finaliza la transferencia de paquetes. De forma que se puede observar, cómo, usando TCP, la comunicación tarda entre 4-6 segundos en recuperar los paquetes perdidos y continuar la reproducción.

### C. Análisis de los Niveles de Latencia de Handover

Este último análisis busca segregar el tiempo total de desconexión en las diferentes capas involucradas en la comunicación. El análisis ha sido realizado tanto para tráfico UDP como para TCP. Definimos, por lo tanto, los tiempos de *handover* en los diferentes niveles como:

- L2 (*handover* de nivel 2). Es el tiempo transcurrido desde que el MN pierde su enlace con el punto de acceso inalámbrico hasta establecerlo con el nuevo, enviando un RS a su nueva ancla de movilidad.
- LSDN (tiempo de procesamiento en el controlador SDN). Este es el tiempo consumido desde que el MN envía un RS al OFSwitch en la nueva red visitada, hasta que éste le comunica al MN los parámetros IP de esa red con el envío de un RA.
- L3 (tiempo completo de recuperación del flujo en el nivel IP). Este es el tiempo transcurrido desde que el MN configura su nueva dirección IPv6, hasta la recepción del primer paquete perteneciente al flujo previamente iniciado.

La Figura 9 muestra la descomposición de la latencia de *handover* experimental obtenida con SR-DMM. La mayor parte del tiempo es consumido en la desconexión de la red inalámbrica anterior y la asociación con la nueva red (L2), donde la tecnología empleada en este nivel es IEEE 802.11g. El tiempo consumido por procesamiento en el controlador de red (LSDN) es alrededor del 0,3 - 0,7% del tiempo total de *handover*, siendo este nivel el encargado del procesado de los mensajes de control. Con respecto a la recuperación completa del flujo de datos (L3), pueden apreciarse claras diferencias

entre los flujos UDP y TCP, donde este último emplea un 40% del tiempo de *handover* frente al 6% de UDP.

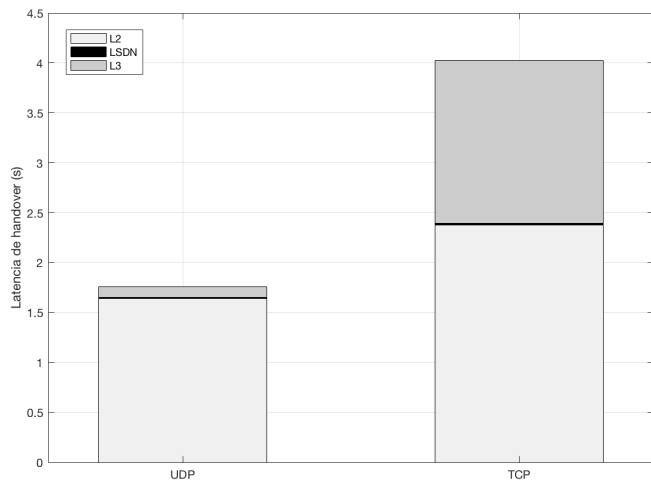


Fig. 9. Descomposición de la latencia de *handover* experimental obtenida con SR-DMM.

La Tabla II recoge el valor de latencia de *handover* para cada tipo de tráfico, según los niveles descritos anteriormente.

TABLA II  
LATENCIA DE HANDOVER PARA TRÁFICO UDP Y TCP

	L2 (s)	LSDN (s)	L3 (s)
UDP	1,638	0,013	0,109
TCP	2,376	0,015	1,632

## VI. CONCLUSIONES

Este artículo presenta una solución de gestión de la movilidad basada en la arquitectura SDN, llamada SR-DMM. La solución se ha planteado con el objetivo principal de aportar flexibilidad y escalabilidad al proceso de movilidad en redes de próxima generación. Se han analizado también las similitudes de esta propuesta con la solución DMM basada en la red, comparando ambas soluciones a través de un modelo analítico.

La propuesta basada en SDN presenta una mayor señalización de control que la propuesta basada en DMM cuando el nodo móvil se mueve muy rápidamente entre diferentes OFSwitches. Por el contrario, aporta la simplicidad del despliegue del protocolo de gestión de la movilidad sin tener que modificar los nodos de la red, aportando también sencillez en el plano de datos, donde no es necesario incorporar ningún mecanismo de encapsulación adicional.

Por otra parte, y con respecto a la evaluación experimental de la solución, se puede concluir que SR-DMM presenta mejoras de escalabilidad sustanciales, ya que gracias a la información que el controlador tiene de la infraestructura, el protocolo de gestión de la movilidad se reduce. Esto queda constatado en el tiempo requerido por SR-DMM para la gestión de la movilidad, indicado como LSDN en la sección anterior, y que supone menos del 1% del tiempo de *handover* total de la comunicación.

Por último, se puede concluir que SR-DMM es una solución de gestión de la movilidad que aporta flexibilidad a las redes tradicionales y aprovecha las capacidades de programabilidad

disponibles en las redes de próxima generación. Además de esto, añade una mejora sustancial en el coste de entrega de paquetes con respecto a las soluciones DMM que emplean mecanismos de *tunneling* para redirigir los flujos de datos hasta la nueva ubicación del MN, hecho que sobrecarga la red añadiendo señalización adicional en las cabeceras de control de los paquetes.

## AGRADECIMIENTOS

Este trabajo ha sido financiado, en parte, por a la Junta de Extremadura y el Fondo Europeo de Desarrollo Regional a través del proyecto de investigación IB18003.

## REFERENCIAS

- [1] Cisco Systems, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2016-2021," March 2017.
- [2] Yang, M., Li, Y., Jin, D., Zeng, L., Wu, X., & Vasilakos, A. V. "Software-Defined and Virtualized Future Mobile and Wireless Networks: A Survey," *Mobile Networks and Applications*, 20(1), 4–18, February 2015.
- [3] Ali-Ahmad, H., Cicconetti, C., de la Oliva, A., Mancuso, V., Sama, M. R., Seite, P., & Shanmugalingam, S. "An SDN-Based Network Architecture for Extremely Dense Wireless Networks," In *2013 IEEE SDN for Future Networks and Services (SDN4FNS)* (pp. 1–7), November 2013.
- [4] C. J. Bernardos, A. de la Oliva, and F. Giust, "A PMIPv6-based solution for Distributed Mobility Management," *Internet Engineering Task Force, Internet-Draft draft-bernardos-dmm-pmip-09*, September 2017.
- [5] D. B. Johnson, J. Arkko, and C. E. Perkins, "Mobility Support in IPv6," *RFC 6275*, July 2011.
- [6] K. Chowdhury, K. Leung, B. Patil, V. Devarapalli, and S. Gundavelli, "Proxy Mobile IPv6," *RFC 5213*, August 2008.
- [7] A. Chan, D. Liu, P. Seite, H. Yokota, and J. Korhonen, "Requirements for Distributed Mobility Management," *RFC 7333*, August 2014.
- [8] J. Calle Cancho, J. Carmona Murillo, D. Cortes Polo, J. L. Gonzalez Sanchez and F. J. Rodriguez Perez, "TE-DMM: A Proposal to Improve the Control Plane in PMIPv6-based DMM Networks," *IEEE Latin America Transactions*, vol. 13, no. 9, pp. 3149-3155, September 2015.
- [9] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617–1634, February 2014.
- [10] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling Innovation in Campus Networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, p. 69, March 2008.
- [11] 5GPPP Architecture Working Group, "View on 5G Architecture," December 2017, White Paper.
- [12] T.-T. Nguyen, C. Bonnet, and J. Harri, "SDN-based distributed mobility management for 5G networks," In *2016 IEEE Wireless Communications and Networking Conference*, pp. 1–7, April 2016.
- [13] F. Giust, L. Cominardi, and C. Bernardos, "Distributed mobility management for future 5G networks: overview and analysis of existing approaches," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 142–149, January 2015.
- [14] W. Chin, Z. Fan, and R. Haines, "Emerging technologies and research challenges for 5G wireless networks," *IEEE Wireless Communications*, vol. 21, no. 2, pp. 106–112, April 2014.
- [15] J. A. Cabrera, R. Schmoll, G. T. Nguyen, S. Pandi and F. H. P. Fitzek, "Softwarization and Network Coding in the Mobile Edge Cloud for the Tactile Internet," in *Proceedings of the IEEE*, vol. 107, no. 2, pp. 350-363, February 2019].
- [16] W. A. Simpson, D. T. Narten, E. Nordmark, and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)," *RFC 4861*, September 2007.
- [17] C. Makaya & S. Pierre, "An Analytical Framework for Performance Evaluation of IPv6-Based mobility Management Protocols," *IEEE Transactions on Wireless Communications*, 7(3), 972–983, March 2008.
- [18] Lee, J. H., Bonnin, J. M., You, I., & Chung, T. M., "Comparative handover performance analysis of IPv6 mobility management protocols". *IEEE Transactions on Industrial Electronics*, 60(3), 1077-1088.
- [19] Open Networking Foundation, "OpenFlow Switch Specification version 1.3 (Wire Protocol 0x04)," June 2012.



**José Manuel Mendoza Rubio** es Graduado en Ingeniería Telemática (2017) por la Universidad de Extremadura. Sus principales áreas de interés son las redes definidas por software y la gestión de la movilidad en redes de próxima generación.



**Javier Carmona Murillo** es Ingeniero Técnico en Informática de Sistemas (2003), Ingeniero en Informática (2005) y Doctor Ingeniero en Informática por la Universidad de Extremadura. Actualmente es profesor contratado doctor del área de Ingeniería Telemática del Departamento de Ingeniería de Sistemas Informáticos y Telemáticos de la UEx. Su investigación se centra en la provisión de QoS en redes móviles y en movilidad IPv6.



**José Luis González Sánchez** es Ingeniero en Informática y Doctor Ingeniero en Informática (2001) por la Universidad Politécnica de Cataluña, España. Actualmente es el Director General de la Fundación COMPUTAEX y el centro CénitS, donde también es el investigador principal de este centro.



**Jesús Calle Cancho** es Ingeniero en Informática (2013) e Ingeniero en Informática de Sistemas (2011) por la Universidad de Extremadura. Actualmente se encuentra desarrollando su labor como ingeniero en la Fundación COMPUTAEX y su centro CénitS. Su principal área de interés es la gestión de la movilidad IP en redes de próxima generación.



**David Cortés Polo** es Ingeniero en Informática por la Universidad de Extremadura (2006) y Doctor Ingeniero en Informática (2015). Sus áreas de interés son las redes móviles IPv6, MPLS-TE y la provisión de QoS. Actualmente es administrador de red de la Fundación COMPUTAEX y su centro CénitS.