

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA



Trabajo de Fin de Máster

**Implementación y análisis de técnicas de
selección de acceso en redes
heterogéneas**

**(On the implementation and analysis of
access selection techniques in
heterogeneous networks)**

Para acceder al Título de

**Máster en Tecnologías de la Información y
Comunicaciones en Redes Móviles (TICRM)**

Autor: Luis Francisco Díez
Julio - 2013

Resumen

El incremento de las capacidades de los dispositivos móviles hace necesario asegurar un uso óptimo de los recursos de la red. Esta mejor gestión debe ir acompañada de la cooperación de todos los procedimientos involucrados sin importar la capa o segmento de red en que se encuentre. A fin de satisfacer esta nueva demanda el marco OConS provee servicios de conectividad adaptativos que satisfacen los requerimientos de aplicaciones, usuarios y las políticas de los operadores a la vez que trata con las condiciones heterogéneas y cambiantes del llamado Internet del futuro. Este trabajo presenta la implementación de los principales componentes de la arquitectura OConS con especial atención a técnicas de selección de acceso.

Sí la mejora de la gestión de los recursos va a ser clave en el futuro, es en la parte de acceso donde su importancia es mayúscula. Siguiendo esta premisa este trabajo presenta un estudio analítico que proporciona, mediante técnicas de optimización, el mejor rendimiento de una red de acceso y su comparativa con los obtenidos mediante algoritmos realistas.

Palabras clave — *Servicios de conectividad, implementación, selección de acceso, redes heterogéneas*

Abstract

The increased capabilities of mobile devices makes it necessary ensuring an optimum use of network resources. The improve in the resources management must be accompanied by cooperation of all procedures involved no matter the layer or network segment they are in. In order to satisfy this demand, OConS framework offers adaptive connectivity services to seamlessly address user and service requirements while complying with operator policies and dealing with the heterogeneous and changing network conditions of the future Internet. This work details the implementation of the most relevant architecture components, especially focused on access selection techniques.

If the network resources management is to be a key element in the future, it will be especially importante in regards to the access network. In this context, this work presents an analytical study which, by means of optimization techniques, aims at find the best performance in an access network and, then, compares the results with those obtained by different algorithms.

Key words — *Connectivity services, implementation, access selection, heterogeneous networks*

Índice general

	Página
Índice general	I
Índice de tablas	III
Índice de figuras	IV
1 Introducción y objetivos	1
2 Arquitectura OConS	4
2.1. Perspectiva general	6
2.2. Comunicación y señalización	9
2.3. Funcionalidad de Orquestación	15
2.3.1. Visión funcional de la orquestación	17
3 Implementación de OConS	19
3.1. Descripción funcional	20
3.2. Orquestación	24
4 Propuesta de movilidad	27
4.1. Solución de movilidad	28
4.1.1. Selección de acceso	29
4.1.2. Gestión dinámica de movilidad	31
4.1.3. Descripción del escenario	32
4.2. Estándar IEEE 802.21	33
4.3. Comparativa	38
5 Estudio analítico	40
5.1. Análisis teórico	40
5.1.1. Formulación del problema	42
5.1.2. Implementación	44
5.1.3. Funciones de utilidad	46
5.1.3.1. Conectividad	46
5.1.3.2. Traspaso	47
5.1.3.3. Parámetro de utilidad de precio	47
5.1.3.4. Parámetro RAT <i>Affinity</i>	48
5.1.4. Combinaciones de parámetros	49
5.1.5. Resultados	49
5.2. Comparación de algoritmos prácticos	57
5.2.1. Función de utilidad	58
5.2.2. Optimización	61
5.2.3. Escenario de red	63
5.2.4. Resultados	64
6 Conclusiones y líneas futuras	72

Lista de acrónimos	75
Referencias	77

Índice de tablas

Tabla	Página
5.1. Funciones de utilidad	49
5.2. Parámetros del análisis	51
5.3. Funciones de utilidad para optimización	63
5.4. Parámetros de rendimiento	65

Índice de figuras

Figura	Página
2.1. Niveles de mecanismos OConS con algunos ejemplos de mecanismos	9
2.2. Arquitectura básica del nodo	10
2.3. Cabecera común de mensajes Open Connectivity Service (OConS)	11
2.4. Ejemplo de procesos de comunicación local y remotas	13
2.5. Ejemplo de <i>bootstrapping</i> de orquestación	17
3.1. Estructura de la clase-plantilla genérica EE	22
3.2. Estructura de la clase-plantilla genérica IE	22
4.1. Escenario de movilidad	29
4.2. Herramienta de visualización	31
4.3. Modelo de capas del estándar 802.21	34
4.4. Servicios Media Independent Handover (MIH)	35
4.5. Mapeo de servicios de eventos; 4.5(a) representa el modelo local mientras que 4.5(b) representa el remoto	36
4.6. Mapeo de comandos MIH; 4.6(a) representa el modelo local mientras que 4.6(b) representa el remoto	37
5.1. Máquina de estados para un par usuario-servicio	44
5.2. Ejemplo de fotografías de servicios	45
5.3. Funciones de utilidad relativas al precio	48
5.4. Despliegue de red	50
5.5. Tasa media de finalización de servicio con éxito Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1	52
5.6. Número medio de trasposos por servicio Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1	53
5.7. Precio medio por servicio y TU Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1	54
5.8. RAT <i>affinity</i> medio por servicio Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1	55
5.9. Rendimiento de las diferentes funciones de utilidad. Línea punteadas para función ligada al precio; línea discontinua para función ligada al RAT <i>affinity</i> ; línea continua representa la combinación de las anteriores.	56
5.10. Precio de la estación base en función de la capacidad disponible	59
5.11. Despliegue de red	64
5.12. Número medio de trasposos Vs. el parámetro de descuento χ . Las líneas continuas representan a usuarios <i>business</i> y las punteadas a los <i>consumer</i>	66

5.13. Evolución del precio medio por servicio con el parámetro de descuento χ	67
5.14. Precio medio por servicio para los diferentes esquemas analizados. En el caso de las optimizaciones (columnas izquierda y central) los marcadores rellenos se corresponden con la función de utilidad que que considera traspasos, mientras que las vacías se obtienen sin considerar el parámetro γ . Las líneas continuas corresponden a los usuarios <i>business</i> mientras que las puntuadas a los <i>consumer</i>	68
5.15. Número medio de traspasos por servicio para los diferentes esquemas analizados. En el caso de las optimizaciones (columnas izquierda y central) los marcadores rellenos se corresponden con la función de utilidad que que considera traspasos, mientras que las vacías se obtienen sin considerar el parámetro γ . Las líneas continuas corresponden a los usuarios <i>business</i> mientras que las puntuadas a los <i>consumer</i>	69
5.16. Figuras de rendimiento para los diferentes esquemas analizados. Las líneas continuas representan la tarifa plana, las punteadas la política de tarificación dinámica con ψ 0.6 y las que tiene punto/raya se corresponden con ψ 0.8	71

Introducción y objetivos

La irrupción de terminales móviles de altas capacidades en los últimos años ha supuesto que se tengan que revisar los pronósticos que, en el campo de las comunicaciones móviles, se habían realizado. Esta proliferación de terminales ha venido acompañada de cambios en el modo de consumo por parte de los usuarios, los cuales toman cada vez un papel más activo en la creación de contenido y cuya demanda de calidad de los servicios produce un incremento notable del volumen de tráfico.

En este sentido se pueden encontrar estudios, como [1], que estiman que el tráfico debido a terminales móviles crecerá en torno a 15 veces para finales de 2017, a su vez se estima que en torno a un 85% de la población mundial tendrá la posibilidad de usar conexiones móviles de alta capacidad como WCDMA/HSPA.

Además de las capacidades de computación de los terminales actuales (*smartphones* y *tablets*), la gran mayoría están equipados con diferentes tecnologías de acceso, por lo que las llamadas *Heterogeneous Networks* (*Het-Nets*) son de gran interés tanto para la comunidad científica como para la industria.

Con lo anterior en mente (aún sin considerar el impacto que puedan llegar a tener las comunicaciones *Machine-to-Machine* y el *Internet-of-Things*), la gestión óptima de los recursos en las redes de acceso inalámbrico es, si cabe, más importante que en el pasado. A pesar de los considerables esfuerzos llevados a cabo durante la última década, especialmente tras la aparición del paradigma *Always Best Connected* [2], todavía existen nuevos retos y aspectos que deben ser estudiados.

Las redes heterogéneas, que definen escenarios caracterizados por un amplio abanico de tecnologías, formarán parte de lo que se ha llamado *Network of Future*. En estos escenarios el amplio abanico de servicios ofrecidos así como de perfiles de usuarios finales precisará una gestión más granular tanto de la Quality of Service (QoS) como de la tarificación que se haga por

servicio. También se cree que el grado de responsabilidad de los usuarios finales en cuanto a la decisión de conexión irá en aumento; hoy en día ya existe un primer paso en esta dirección, especialmente el terminales móviles que permiten configurar perfiles de conectividad. Siguiendo esta tendencia, un usuario sería capaz de seleccionar la alternativa de acceso que mejor satisficiera sus necesidades para un servicio determinado (hoy en día no existe este nivel de granularidad) y quizá teniendo en cuenta el contexto concreto (por ejemplo, podría no precisarse los mismos requerimientos para una video-llamada de trabajo que para una privada). A pesar del esfuerzo que durante varios años se ha llevado a cabo en esta línea por parte de la comunidad científica dedicada a comunicaciones inalámbricas, la mayoría de los estudios se centran en las condiciones de la red y parámetros físicos. Por otro lado, también se podría argumentar que otros parámetros (por ejemplo precio) puede jugar un papel crucial al determinar la satisfacción (Quality of Experience (QoE)) del usuario al hacer uso de un servicio.

Estas nuevas exigencias, tanto el lo referente a gestión de la red como preferencias de usuarios, deben estar soportadas por arquitecturas de red que sean suficientemente flexibles y abiertas. A fin de responder a estos nuevos requerimientos, en el marco del proyecto europeo Scalable Adaptive Internet Solutions (SAIL) (FP7-ICT-2009-5-257448) se ha definido la arquitectura OConS. Ésta tiene como objetivo definir un marco común que dé soporte, de manera homogénea, a los diferentes servicios de conectividad tanto actuales como a los que puedan aparecer en el futuro.

Este trabajo se enmarca dentro tanto de la arquitectura OConS como del proyecto Comunicaciones Cognitivas y Cooperativas sobre Entornos Heterogéneos (C3EM) [3] (TEC2009-14598-C02-01) y se centra en el estudio, desarrollo y análisis de técnicas de selección de acceso *cross-layer*. De esta manera se pretende dotar a los nodos de la capacidad de seleccionar el mejor de acceso de acuerdo a información de muy diversa naturaleza, potencialmente proveniente de cualquier punto de la capa de protocolos y segmento de red. El trabajo posee dos partes principales, la primera está orientada a la implementación de soluciones de selección de acceso siguiendo la especificación OConS, mientras que la segunda parte se centra en el estudio analítico de diversas técnicas. Una descripción más detallada de la estructura del documento se muestra a continuación:

- En el Capítulo 2 se hace una introducción de la arquitectura OConS justificando su desarrollo e indicando las carencias que pretende solventar. Dada la complejidad de la arquitectura, la explicación se ha limitado a los aspectos más relevantes para el resto del documento.
- En el Capítulo 3 se detalla la implementación realizada de los componentes principales de la arquitectura. Se trata de una implementación *software* que ha sido usada a lo largo del proyecto SAIL y que da soporte a desarrollos posteriores.
- El Capítulo 4 toma como punto de partida el desarrollo anterior. A partir de éste se ha implementado una solución de movilidad que,

siguiendo la filosofía OConS, presenta una serie de ventajas conceptuales. Esta solución se ha comparado con la norma Institute of Electrical and Electronics Engineers (IEEE) 802.21 [4] ya tratarse del estándar más extendido en este campo.

- A fin de analizar el rendimiento de las estrategias de selección acceso que se podría esperar en una red heterogénea, en el Capítulo 5 se han realizado una serie de estudios analíticos. En primer lugar, se ha modelado el problema de selección de acceso como un problema de optimización a fin de hallar el mejor comportamiento que cabe esperar de acuerdo con los parámetros tenidos en cuenta. Seguidamente se ha realizado una campaña de simulaciones de diferentes estrategias de selección, todas ellas siguiendo la arquitectura OConS, y se ha estudiado su rendimiento de acuerdo con una serie de figuras de mérito.
- Finalmente, en el Capítulo 6 se resumen las principales conclusiones que se han obtenido, a la vez que se indican las líneas de investigación futura que surgen a raíz del proyecto.

Arquitectura OConS

En este capítulo se va a presentar el marco OConS cuyo principal objetivo es el de aportar una solución global y armonizada a la integración de diferentes servicios de conectividad.

La motivación del desarrollo de OConS radica en la identificación de una serie de carencias que se dan en los mecanismos de conectividad de las redes actuales y que este marco pretende paliar. Estas carencias se pueden resumir en dos aspectos generales:

- Falta de coordinación entre mecanismos: típicamente, los diferentes mecanismos de conectividad trabajan de forma aislada por lo que no es posible aprovechar las sinergias que su coordinación pueda producir o advertir los conflictos que se puedan dar entre ellos. Esta falta de coordinación se da tanto de forma vertical (en diferentes capas de la pila Open System Interconnection (OSI)) como horizontal (en una misma capa).

Un ejemplo de esta estanqueidad se podría encontrar en los diferentes mecanismos o protocolos de gestión de movilidad IP, en los que una detección anticipada de la movilidad por parte de los protocolos de nivel de enlace mejora en gran medida las prestaciones del traspaso a nivel IP. Esta interacción entre capas se da en [5], conocido como *Fast Handover for Mobile IPv6*, en el que se definen primitivas entre las capas de enlace y red. Sin embargo, este tipo de soluciones concebidas para problemas concretos vuelven a ser estancas con respecto a otras.

- Carencia de flexibilidad y expresividad en el control de red: este problema está estrechamente ligado con el anterior y es consecuencia del funcionamiento aislado de los mecanismos. Las políticas y reglas que rigen el control de la red son, a menudo, específicas para cada mecanismo (por ejemplo políticas de *firewalls*, políticas para redes fijas o

móviles, políticas de *forwarding*). Estas políticas y reglas suelen afectar únicamente a un segmento de la red; por un lado este aislamiento simplifica las decisiones pero, a su vez, hace que se pierda expresividad, es decir, una decisión en un segmento de la red, aún cuando puede afectar a otro, no tiene sentido en el segundo ya que no se ha definido para este.

Esta falta de expresividad puede ser un lastre en la gestión de usuarios diferentes (con diferentes políticas y/o preferencias) por el hecho de que las políticas o reglas que gestionen su tráfico se van a limitar a un único segmento de la red, cuando sería necesario un mayor control en diferentes niveles (capas) y nodos a lo largo del camino de datos.

Como ejemplo de esta falta de expresividad se puede destacar la gestión de la QoS en redes celulares como Long Term Evolution (LTE), la cual no puede ser estrictamente considerada como extremo a extremo dado que únicamente se asegura un cierto nivel de QoS en el interior de la red LTE. Esto se debe a que al salir de la red LTE (más allá del *Serving Gateway*) los mecanismos y reglas de su interior no se puedan entender fuera de ella.

Para intentar paliar estas carencias, la arquitectura OConS dota a los servicios de conectividad de un marco capaz de gestionarlos de una manera homogénea, coordinada y consistente. Esto se lleva a cabo a través de la abstracción de los mecanismos haciendo uso de interfaces que permitan su coordinación (orquestación en terminología OConS) y de políticas con mayor nivel de expresividad que permitan la gestión extremo a extremo. Por otro lado, la generalidad de la arquitectura hace que la integración de nuevos servicios resulte sencilla y transparente para el resto.

Teniendo en mente la continua integración de los mecanismos de control en los entornos de red y el aumento de capacidades de los dispositivos, uno de los enfoques más relevantes hoy en día es el llamado Software-defined networking (SDN) [6], [7]. En este campo el marco *Open Flow* es el ejemplo más conocido de implementación SDN. Dado que la implementación de OConS también se enmarcaría dentro de las SDN merece la pena realizar una breve comparación entre ellos.

En el ámbito *Open Flow* uno o varios controladores lógicamente centralizados gestionan la conmutación del plano de datos mediante políticas a nivel de flujo que son aplicadas por los *switches Open Flow*. Los requerimientos de operador/usuario/aplicación se integran en el controlador (o en las funciones de control por “encima” de éste), sin embargo esta asignación se realiza de forma implícita, no explícita; no se podría decir explícitamente que el usuario *Alice* tiene preferencia frente al usuario *Bob*. La mayor parte de las decisiones se realizan en el controlador, por lo que sólo las reglas de reenvío se llevan a los nodos intermedios del camino de datos, los *switches*. En resumen, la funcionalidad de *Open Flow* puede verse como:

OpenFlow = IF (paquete/flujo, estado de red/contexto) THEN (aplicar ejecuciones de manera secuencial)

El concepto OConS es similar, pero va un paso más allá. Es similar en el sentido de que un mecanismo recibe información desde la red y los nodos finales toman una decisión en base a la información y aplican las políticas sobre el plano de datos a través de las entidades de ejecución. Sin embargo OConS extiende este concepto habilitando la orquestación de varios mecanismos. Se define un interfaz que permita una comunicación vertical entre los mecanismos y la inteligencia de orquestación e interfaces de comunicación horizontal entre los diferentes mecanismos. Además, OConS provee una manera de definir políticas con mayor expresividad que los simples flujos ya que tiene en cuenta perfiles de usuario, aplicación y requerimientos de operador. Además OConS posee capacidades que van más allá del *forwarding* de paquetes ya que integra mejoras a nivel de enlace, flujo y red e involucra a todos los nodos en este proceso y no sólo a los *switches* o *routers*. De este modo, la ejecución de las decisiones en función de las políticas resultan en una cadena de ejecución que la conforman partes de diferentes mecanismos. En resumen, la ejecución se podría definir como:

OConS = IF(paquete/flujo, estado de red/contexto, requerimientos de usuario/aplicación/operador) THEN (aplicar cadena de Decisiones/Funciones de Control + Cadena de ejecución)

Una vez presentada la idea que subyace bajo el desarrollo de la arquitectura OConS, a lo largo de este capítulo se va a explicar su diseño, de manera que facilite la comprensión de la plataforma que se ha desarrollado (Capítulos 3 y 4) y que sirve como prueba de concepto de los aspectos más relevantes de la arquitectura.

Dado que la descripción de la arquitectura completa resultaría muy extensa, este capítulo se va a centrar en los aspectos que resultan más relevantes. Así, en la Sección 2.1 se dará una visión general tanto de la arquitectura como de sus componentes principales. Posteriormente, en las Secciones 2.2 y 2.3 se detallarán los aspectos más importantes de la comunicación, señalización y orquestación.

2.1. Perspectiva general

En términos generales OConS es un marco de control que provee la capacidad de orquestación a un conjunto de servicios de conectividad presentes en uno o varios nodos interconectados. Se define un servicio como un conjunto de mecanismos de conectividad que responden a unos ciertos requerimientos. A fin de llevar a cabo el diseño y desarrollo de los mecanismos de manera sencilla, de facilitar la integración de mecanismos existentes y de dar la capacidad de reutilizar sus funcionalidades, los mecanismos OConS se modelan como la descomposición de entidades funcionales de monitorización (Information Management Entity (IE)), de toma de decisiones (Decision Making Entity (DE)) y de ejecución (Execution and Enforcement Entity (EE)). Esta abstracción en entidades funcionales responde a la necesidad de integrar mecanismos de muy diferente naturaleza (diferentes capas, tecnologías...) con pocos aspectos comunes.

La arquitectura funcional de OConS tiene como centro el llamado Service Orchestration Process (SOP), esta entidad es la inteligencia capaz de orquestar diferentes servicios, a su vez compuestos por mecanismos. La selección de los servicios para un determinado usuario OConS (ya sea una aplicación, operador o usuario final) responde a los requerimientos o preferencias en cuanto a conectividad que el usuario tenga. Estos requerimientos son comunicados a través de un interfaz llamada Orchestration Service Access Point (OSAP) que hace de punto de entrada del usuario a la arquitectura OConS. Mediante este interfaz el usuario también puede recibir información acerca del estado del nivel de conectividad requerido, permitiéndole conocer si se han alcanzado las exigencias solicitadas o se requiere algún tipo de negociación que reduzca sus expectativas. A fin de almacenar la información de los diferentes mecanismos, reglas y políticas, así como del estado de la red, el SOP hace uso de una base de datos, el Orchestration Register (OR).

Finalmente la comunicación entre los nodos de la arquitectura, o entre las entidades dentro de ellos, se realiza por medio de un interfaz llamado Inter/Intra-Node Communication (INC). Por medio de este interfaz se consigue crear una red virtual superpuesta que hace uso de las diferentes tecnologías de comunicación de los nodos para comunicar los componentes de la arquitectura.

Con todo lo anterior en cuenta, el único requerimiento básico que debe tener un nodo OConS es poseer un INC. Por otro lado si un nodo tuviera una entidad SOP y por lo tanto fuera capaz de orquestar servicios, esta entidad debería ser conocedora de la manera de acceder a un OR, el cual puede ser local o remoto.

Como se ha comentado anteriormente de forma general, para ser capaces de diseñar y combinar nuevos mecanismos (así como de integrar los actualmente desplegados), éstos deben ser modelados de acuerdo a un marco estándar que pueda extraer los elementos básicos de cada mecanismo y representar mecanismos en esencia diferentes de una forma uniforme. Tras analizar algunas nuevas propuestas y otros mecanismos existentes, se llegó a la conclusión de que su funcionalidad es relativamente similar. Por ello, se pudo descomponer cada mecanismo en entidades de menor tamaño y más fáciles de manejar, definir y configurar. Esto permite compartir, reutilizar y diseñar más fácilmente funciones de red.

Tomando esta idea como base, se han definido tres entidades funcionales genéricas:

- Information Management Entity (IE): esta entidad se encarga de obtener información y llevar a cabo labores de monitorización. La información puede ser procesada (por ejemplo filtrada o agregada) antes de ser suministrada a quien la haya requerido o se haya suscrito a ella.
- Decision Making Entity (DE): es el punto de toma de decisiones. Esta entidad usa información proveniente de IEs o de la salida de otros DEs para tomar las decisiones. Se contemplan los casos de decisiones

centralizadas en un nodo o distribuidas en varios nodos de la topología de red, en este último caso una determinada DE tomará como información de entrada decisiones de otras DEs.

- Execution and Enforcement Entity (EE): por último, esta entidad es la encargada de ejecutar las acciones que se hayan considerado oportunas y de informar a los DEs del estado de la ejecución si fuera necesario.

Desde un punto de vista práctico estas entidades son procesos (fragmentos de *software*) que pueden ser distribuidos, instanciados y ejecutados en nodos con capacidades OConS. Estas entidades serán provistas con identificadores que permita diferenciarlas y harán uso de interfaces lógicas que las dote de capacidades de comunicación.

A partir de las entidades descritas, se puede definir un mecanismo OConS como un conjunto ellas. De este modo, un mecanismo ha de estar compuesto por, al menos, un DE y debe involucrar uno y varios IEs y EEs. Dado que la definición de las entidades es independiente de cualquier capa o protocolo, todos los mecanismos pueden ser definidos a partir de ellas. Además, tomando esta metodología de diseño, los mecanismos OConS (ya sean nativos OConS o modificados para incorporar sus funcionalidades) pueden ser orquestados.

En la definición de la arquitectura, y a fin de simplificar el proceso de orquestación, se han establecido tres categorías de mecanismos en función del nivel en el que se encuentran:

- Mecanismos de flujo: son específicos para un flujo y sesión. Ya sean extremo a extremo o *edge-to-edge*, los mecanismos de este nivel serán de transporte y encaminamiento y son dependientes de la aplicación.
- Mecanismos de red: se trata de servicios específicos para los nodos y los componen mecanismos de transporte y encaminamiento. En este caso los mecanismos son independientes de las aplicaciones, implican uno o más nodos y se extienden a uno o más saltos.
- Mecanismos de enlace: son específicos de un enlace en concreto y no se extienden más de un salto. Son típicamente implementados en las capas física y de enlace.

La Figura 2.1 ilustra los niveles de mecanismos OConS con algunos ejemplos para cada nivel. Dado que los mecanismos deben ser coordinados para proveer un determinado servicio de conectividad, se precisa de una forma común de representar sus características en términos prestaciones, recursos que necesita, limitaciones, posibles sinergias, conflictos con otros mecanismos, etc. Todas las características propias de un mecanismo se especifican mediante una estructura de datos llamada *Mechanism Manifest* que contiene la información necesaria de un mecanismo en concreto. Dado que en los desarrollos que se van a presentar de la arquitectura OConS en capítulos posteriores no se hace uso del *Mechanism Manifest*, no se va a

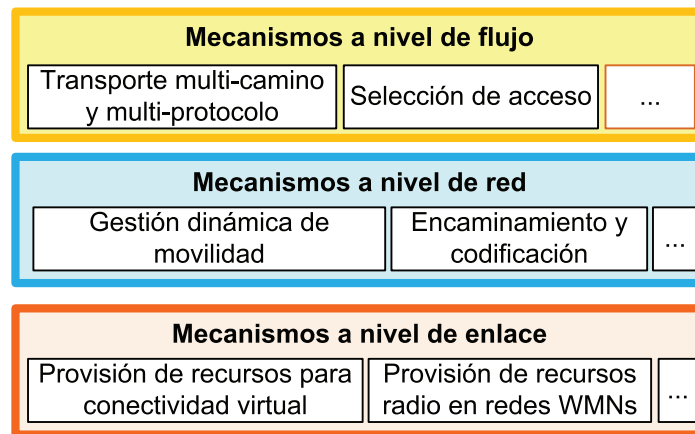


Fig. 2.1: Niveles de mecanismos OConS con algunos ejemplos de mecanismos

detallar su estructura. No obstante, un mayor detalle de este elemento, y de toda la arquitectura, se puede encontrar en [8].

Como se ha mencionado anteriormente, cuando un usuario OConS solicita un cierto servicio de conectividad, éste se compone como la combinación de diferentes mecanismos. La combinación de los mecanismos que cumplan los requerimientos está a cargo de la funcionalidad de orquestación; ésta debe saber qué mecanismos son complementarios y en qué medida su combinación es beneficiosa, cuáles son ortogonales y si algunos de ellos pueden llegar a ser conflictivos.

2.2. Comunicación y señalización

Los nodos OConS se definen como elementos de red que albergan entidades funcionales; por lo tanto, se puede considerar un nodo OConS tanto a nodos de infraestructura como a terminales de usuario. Esta sección va a presentar la composición y los requerimientos de un nodo OConS así como la comunicación local o remota que tiene lugar entre las entidades residentes en ellos.

Antes de detallar la señalización es necesario indicar los diferentes componentes que intervienen en ésta. En la Figura 2.2 se pueden observar los diferentes componentes que constituyen el nodo y que han sido brevemente descritos con anterioridad. No todos los componentes deben estar presentes en todos los nodos, sino que pueden estar distribuidos entre varios de ellos. Los únicos que se consideran obligatorios son aquellos que permitan la comunicación; es decir, un INC.

El INC da soporte a la comunicación, tanto local como remota, entre todos los componentes de la arquitectura. Está al cargo de la recepción de mensajes de componentes locales y remotos y de dirigirlos a su destino, lo mismo se puede aplicar al sentido inverso. Además, el INC se encarga de la elección de la tecnología de transporte más adecuada para cada caso; en el caso de comunicaciones locales se puede usar Inter-process Commu-

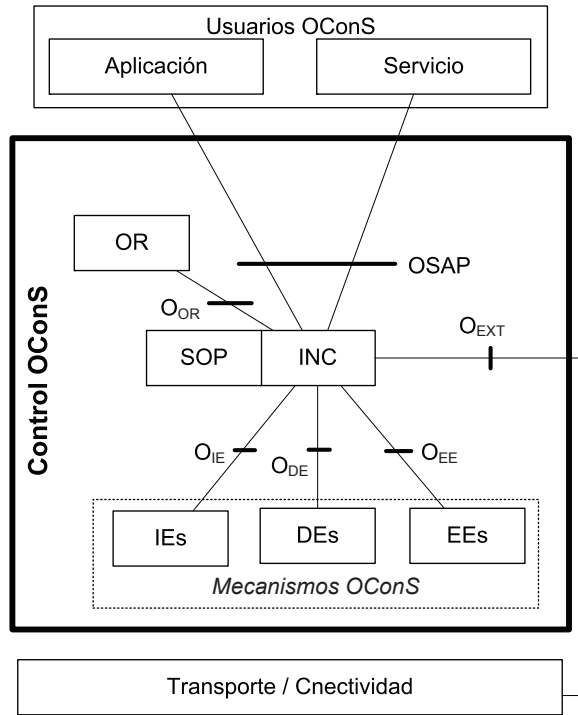


Fig. 2.2: Arquitectura básica del nodo

nication (IPC), mientras que cuando tienen lugar comunicaciones remotas seleccionará el interfaz que corresponda.

El uso del INC permite abstraer a las entidades de la tecnología de comunicación subyacente, dando lugar a un nivel de abstracción superior. Como se puede observar en la Figura 2.2, la comunicación se realiza a través de una serie de interfaces lógicas que comunican los diferentes componentes que pueden estar presentes:

- O_{IE} , O_{DE} , O_{EE} : se trata de los interfaces que permiten la comunicación con las entidades lógicas. Estos interfaces habilitan el anuncio, descubrimiento, registro y configuración de entidades, así como la comunicación que tenga lugar durante su operación.
- $OSAP$: se trata del interfaz externa de OConS mediante la cual el usuario comunica sus requerimientos a través de un perfil de demanda. Del mismo modo, el sistema OConS usa este interfaz para informar al usuario del estado del servicio de conectividad requerido (por ejemplo, indicando un código de error).
- O_{OR} : permite la comunicación con el OR. Por tanto, se usa para el registro de entidades o mecanismos, validación de mecanismos, etc.
- O_{EXT} : finalmente, este interfaz habilita la comunicación entre nodos remotos, concretamente entre dos INCs de diferentes nodos. En la práctica se trata del punto de unión del sistema OConS con las diferentes tecnologías que se usarán durante la comunicación.

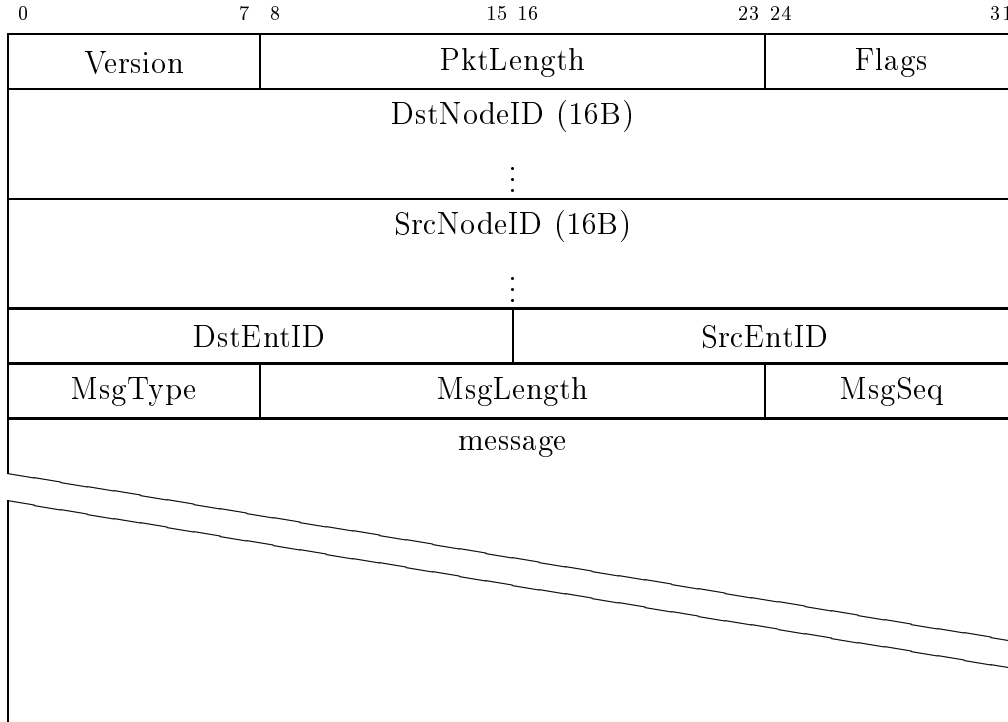


Fig. 2.3: Cabecera común de mensajes OConS. La longitud total del paquete **PktLength** es **MsgLength**+36 B.

Tras presentar, de forma genérica, los actores que intervienen en la comunicación y definir los interfaces virtuales que usan, se va a profundizar más en cómo la comunicación tiene lugar en la práctica. Todas las entidades se identifican unívocamente por un identificador (OConS ID) que se construye mediante un sistema jerárquico de dos niveles:

- Identificador de nodo (*node ID*): en primer lugar, mediante una dirección del 16 bytes se identifica de forma unívoca al nodo. Todas las entidades instanciadas en un mismo nodo comparten este identificador.
- Identificador de entidad (*entity ID*): a fin de diferenciar entidades locales y permitir al INC resolver el direccionamiento local, este campo de 2 bytes alude a la entidad dentro del nodo. Este último identificador se atribuye dinámicamente por el INC.

Mediante estos dos niveles de direccionamiento se pueden llevar a cabo dos tipos de comunicación: entre nodos (por medio de paquetes OConS) y entre entidades (mediante mensajes OConS). En el primer caso el direccionamiento únicamente tiene en cuenta los *node IDs* que se incluyen en las cabeceras de los paquetes. A su vez, cada paquete puede contener uno o más mensajes que contienen la dirección de la entidad dentro del nodo. En la Figura 2.3 se muestran los diferentes campos que constituyen las cabeceras, tanto de paquetes como de mensajes. Como se puede observar, el contenido de las cabeceras es mínimo y tienen como objetivo la representación de la

arquitectura de direccionamiento en las comunicaciones. Además, el uso de mensajes como componente del paquete permite llevar a cabo *piggybacking* (encapsular varios mensajes, dirigidos a diferentes entidades de un mismo nodo, dentro de un único paquete), lo que podría ahorrar transmisiones de señalización.

A pesar de que los paquetes tienen sentido a nivel de nodo, mientras que los mensajes lo hacen a nivel de entidad, las entidades deben ser capaces de distinguir unívocamente a sus entidades pares durante la comunicación. Es por ello que deben recibir tanto el identificador de nodo como de entidad, la forma en la que esto se lleve a cabo dependerá de la implementación en particular.

Durante el normal funcionamiento de un mecanismo, las entidades se envían mensajes entre ellas. Sin embargo, no son conocedoras de los mecanismos de comunicación que soporta su nodo ni de cuál de ellos (en caso de que haya varios) es el adecuado para comunicarse con una entidad en concreto; el único conocimiento que una entidad posee es el OConS ID (*node ID* + *entity ID*) de la entidad con la que se va a comunicar. La labor de seleccionar el método de comunicación adecuado está a cargo del INC.

A continuación, y con ayuda de la Figura 2.4, se van a describir los pasos más importantes que se llevan a cabo en la comunicación tanto local como remota. En el caso local la comunicación tiene lugar entre un DE que solicita a un EE la ejecución de una determinada acción. Los pasos que se siguen, denotados con letras, se describen a continuación:

- A** El DE envía un mensaje con la decisión que debe ser ejecutada hacia el INC a través del interfaz $\mathbf{O}_{DE}$ mediante algún sistema IPC.
- B** El INC comprueba, en la cabecera del paquete, que se trata de una entidad local (mismo *node ID*) y reenvía el paquete a la entidad apropiada; en este caso al EE que ejecuta la decisión adoptada.

Por otro lado, la comunicación remota involucra a un IE que envía información a un DE, por ejemplo como consecuencia de una suscripción previa. Los pasos que se siguen, indicados en la Figura 2.4 con números, se describen a continuación:

- 1** El IE envía un paquete a su INC local a través del interfaz $\mathbf{O}_{IE}$.
- 2** El INC selecciona el interfaz adecuado para comunicarse con el nodo destino. Realmente el procedimiento es el mismo que en el caso anterior, la única diferencia es que en este caso la selección da como resultado un interfaz de red (IPv6 en este caso) en lugar de un interfaz IPC.
- 3** El INC encapsula el paquete OConS dentro de la tecnología de comunicación oportuna y lo envía, a través del interfaz $\mathbf{O}_{EXT}$, hacia el interfaz de red. Si pensamos en comunicaciones mediante sockets, el interfaz $\mathbf{O}_{EXT}$ es precisamente el API socket.
- 4** Se transporta el paquete mediante la tecnología elegida.
- 5** En el nodo remoto el INC recibe y desencapsula el paquete (recepción mediante $\mathbf{O}_{EXT}$), de modo que se vuelve al ámbito OConS.
- 6** Finalmente, el INC remoto comprueba que la dirección del nodo destino se corresponde con la propia y selecciona el interfaz correspondiente,

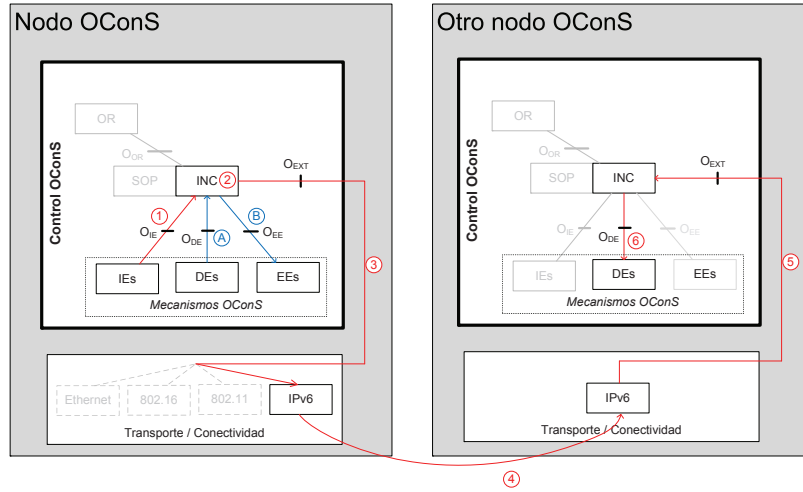


Fig. 2.4: Ejemplo de procesos de comunicación local y remotas

en este caso un IPC local. Finalmente se envía el mensaje a la entidad DE a través del O_{DE} .

En la última parte de esta sección se van a indicar los diferentes tipos de mensajes que se pueden generar en las diferentes comunicaciones. A pesar de que los componentes de orquestación se detallarán en la sección siguiente (Sección 2.3) los mensajes que en ella se intercambian se van a indicar aquí al entender que es más propio de señalización que de orquestación. Valga aclarar que la orquestación se divide en dos entidades principales: el SOP que implementa la inteligencia y el OR que almacena la información referida a entidades y mecanismos. La descripción de los tipos de mensajes facilitará la comprensión de las acciones que se llevan a cabo en la implementación que se describe en el Capítulo 3.

En primer lugar, cabe indicar que debido a la heterogeneidad de la información contenida en los mensajes ésta se encapsula siguiendo un formato Type Length Value (TLV) [9]. De forma general, atendiendo al sentido de la interacción, los mensajes se dividen en tres tipos principales:

- *Request*: no se solicitan previamente y provocan una respuesta por parte del receptor.
- *Response*: se envían como respuestas inmediatas ante la recepción de un *Request*.
- *Notification*: son la versión no solicitada de un *Response*. Típicamente tienen lugar como consecuencia de algún tipo de suscripción.

A partir de los tipos básicos de mensajes se han definido cinco clases, de diferente semántica (diferente significado dado por el contexto de la comunicación), atendiendo a su papel dentro de la señalización OConS: manejo de entidades, publicación/suscripción, manejo de mecanismos, comunicación entre entidades y gestión de información de usuario OConS. Dado que durante el resto del documento (especialmente en los Capítulos 3 y 4) únicamente se hace uso de algunos de estas clases, la descripción

se va a limitar el conjunto de mensajes referidos al manejo de entidades, comunicación entre entidades y publicación/suscripción.

Los mensajes que gestionan de entidades son usados por la funcionalidad de orquestación para registrar las diferentes entidades y así identificar los mecanismos disponibles. Los mensajes que se van a describir a continuación hacen uso de los interfaces **O_{IE}**, **O_{DE}**, **O_{EE}** y **O_{EXT}**, este último para comunicaciones remotas:

DISCOVERY_req : se envía por el SOP hacia las entidades para descubrir sus capacidades; puede ser enviado como respuesta a un anuncio por parte de éstas o basado en algunas políticas.

DISCOVERY_rsp : se trata de la respuesta al mensaje anterior.

CONFIGURE_req : lo envía el SOP para configurar parámetros de entidades. Esto está ligado a la monitorización de la red por parte del SOP para la gestión de los mecanismos.

CONFIGURE_rsp : lo envían las entidades como respuesta al anterior.

INFORMATION_notif : enviado por las entidades al SOP en base a alguna configuración (suscripción) o política.

REGISTER_req : lo envían las entidades hacia el OR para registrar su identificador y sus capacidades.

REGISTER_rsp : enviado desde el OR a las entidades para confirmar o rechazar el registro.

Los mensajes de publicación/suscripción se intercambian entre las entidades de orquestación para gestionar mecanismos y servicios. Este intercambio se realiza a través de los interfaces **O_{OR}** y **O_{EXT}**:

SUBSCRIBE_req : lo envía el SOP al OR para suscribirse a las modificaciones que puede presentar un mecanismo o servicio.

SUBSCRIBE_rsp : se trata de la respuesta por parte del OR al mensaje anterior.

PUBLISH_notif : lo envía el OR al SOP para informar sobre un mecanismo o servicio o los cambios que estos puedan haber sufrido.

VALIDATE_req : lo envía el SOP para validar un mecanismo o servicio una vez instanciado .

VALIDATE_rsp : reconocimiento al mensaje anterior.

Una vez que un mecanismo ha sido instanciado y configurado, su funcionamiento se basa en el intercambio de información entre entidades. Los mensajes entre entidades, que se describen a continuación, hacen uso de los interfaces **O_{DE}**, **O_{IE}**, **O_{EE}** y **O_{EXT}** como se pudo observar en la Figura 2.4:

EXECUTION_req/rsp : se intercambian entre un EE y un DE u otros EEs (en el caso de ejecución jerárquica) para llevar a cabo una acción. La respuesta contiene información acerca del estado de la ejecución.

INFORMATION_req/rsp/notif : se trata de mensajes enviados por un DE a un IE (o entre varios IEs en caso de estructuras jerárquicas de IEs) para obtener información. Un IE también puede enviar información mediante notificaciones si ha sido previamente configurado.

CONFIGURATION_req/rsp : se envían entre un DE y un IE o EE para configurarlos.

2.3. Funcionalidad de Orquestación

En esta sección se van a presentar los aspectos básicos de la orquestación OConS. Como se ha mencionado anteriormente, la capacidad de combinar de forma inteligente tanto las entidades para formar mecanismos, como los mecanismos para proporcionar un determinado servicio de conectividad es uno de los principales beneficios de esta arquitectura.

En primer lugar, y tras haber presentado los componentes básicos de la arquitectura (entidades) y la comunicación y señalización que tiene lugar entre ellos, se van a describir con detalle aquellos módulos que soportan la capacidad de orquestación de acuerdo a lo indicado en la Figura 2.2.

El SOP es el elemento central de la arquitectura y es responsable de descubrir y validar los mecanismos y crear los servicios. Es a éste a quien se dirigen los clientes OConS para demandar un servicio con unas ciertas características; como respuesta a esta demanda el SOP es capaz de instanciar y orquestar el servicio como una composición de varios mecanismos que son adecuadamente configurados y ejecutados en uno o varios nodos. Para ello, el SOP hace uso de una serie de reglas que mapean:

- La solicitud de conectividad por un usuario expresada como un perfil de demanda (*demand profile*) a unos requerimientos de conectividad.
- Los requerimientos de conectividad y estado de la red a mecanismos candidatos.
- Mecanismos candidatos a servicio compuesto.

En lo concerniente a este último punto, el SOP provee una serie de funcionalidades. En primer lugar es capaz de inicializar un servicio (*bootstrapping*) teniendo en cuenta los mecanismos descubiertos y aquellos establecidos por defecto. También tiene la capacidad de iniciar o re-configurar un determinado servicio atendiendo a cambios en la demanda de conectividad o en el estado de la red. Finalmente, debe ser capaz de monitorizar el estado de la red para poder actuar en consecuencia y presentar un comportamiento adaptativo.

Por otro lado, para conocer la disponibilidad de entidades, mecanismos, así como de servicio activos, el SOP recurre al OR. En este sentido

el OR puede ser visto como una base de datos que, dependiendo de la implementación que se lleve a cabo, puede ser local, remota, centralizada o distribuida. La definición de la arquitectura deja este punto abierto indicando que se trata del elemento que contiene los siguientes registros:

- Registro de entidades OConS: durante el proceso de *bootstrapping* en un nodo concreto todas las entidades se registran en el OR. Esto se puede llevar a cabo siguiendo un procedimiento de descubrimiento o proactivamente por las entidades. Para cada entidad, la información recogida es el OConS ID (*Node ID* + *Entity ID*), el tipo de entidad y sus capacidades. Esto es, para el caso de las IEs las capacidades son los parámetros que recoge, para las EEs la ejecución que puede llevar a cabo y para las DEs lo que es capaz de decidir y en función de qué lo hace (entradas de información).
- Registro de mecanismos OConS: durante la creación de los mecanismo cada uno de ellos se registra en el OR con un identificador único, sus capacidades y las dependencias en lo referente a entidades. Cabe destacar que en lo que se refiere a las capacidades de los mecanismos, éstas deben indicar las prestaciones que proporciona el mecanismo así como las dependencias con otros; esto es, sinergias constructivas o/y destructivas (conflictos).
- Registro de servicios OConS: finalmente cada servicio se registra en el OR con un identificador y los identificadores de aquellos mecanismos que lo integren.

La especificación de los procedimientos concretos de orquestación definidos en [8] son complejos y contienen aspectos que no han sido descritos en este proyecto. Por ello la descripción se va a centrar en los aspectos y casos de la orquestación más ligados a lo que será descrito en los Capítulos 3 y 4; en cualquier caso una descripción más detallada se puede encontrar en [8].

La orquestación se divide en tres fases principales: la configuración de los nodos, la inicialización de los mecanismos y la propia orquestación de los servicios.

Durante la primera fase se realiza el mapeo de perfiles de demanda de los usuarios (aplicaciones y/o servicios) en requerimientos de conectividad; es decir, se dota al sistema del conocimiento sobre cómo responder a los requerimientos que puedan llegar. Estos requerimientos se traducirán en mecanismos siguiendo diferentes metodologías: pueden ser definidos *off-line* (simulaciones, análisis matemático...), basados en el aprendizaje (*machine-learning*, inteligencia artificial...) o en la experiencia de planificadores de red y sugerencias de usuarios.

En la segunda fase tiene lugar el descubrimiento de las entidades y el registro de los mecanismos. Además, ciertas configuraciones puede precisar la instancia de algunos mecanismos por defecto que deben ser ejecutados siempre, sin atender a demandas concretas (por ejemplo, selección de acceso al encender interfaces radio).

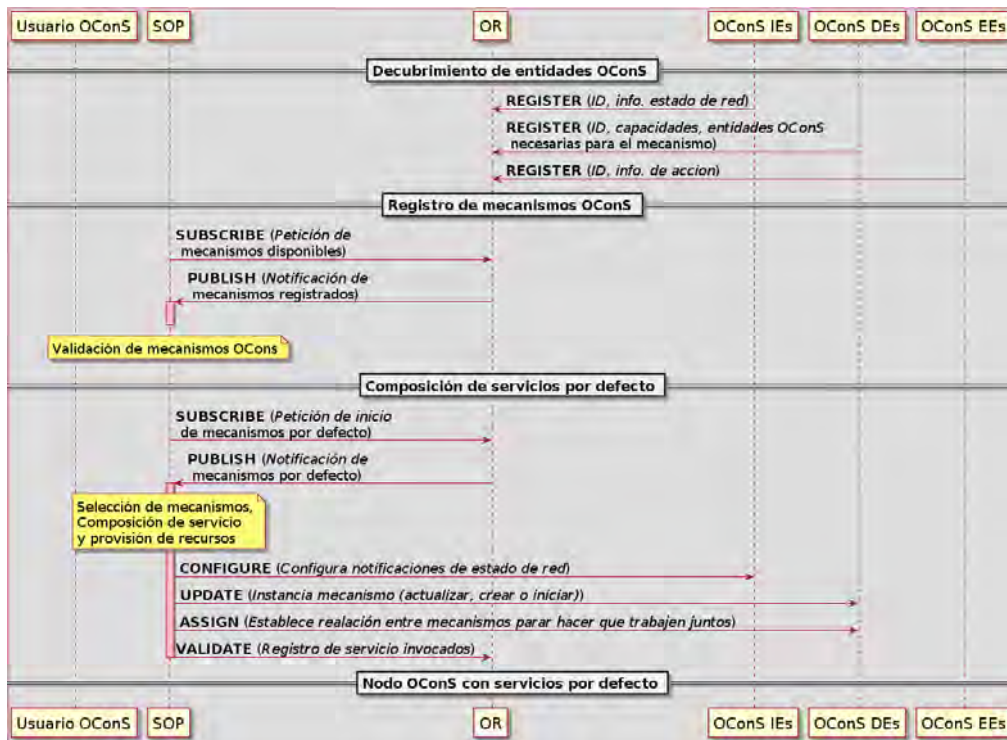


Fig. 2.5: Ejemplo de *bootstrapping* de orquestación

Finalmente, en la tercera fase se da la orquestación en si. Ante la demanda de conectividad de un usuario OConS, definida mediante el perfil de demanda, se selecciona el mecanismo o mecanismos que mejor satisfagan dicha demanda. Estos mecanismos se configuran de manera apropiada y se instancian dando lugar al servicio concreto. Además, se procede a la monitorización de la red a fin de reaccionar a cambios del contexto.

2.3.1. Visión funcional de la orquestación

A fin de validar el desarrollo que se ha realizado de la arquitectura OConS, esta sección va a presentar el funcionamiento práctico de la orquestación tal como se describe en [8] prestando especial atención en el *bootstrapping* o inicialización.

El *bootstrapping* tiene lugar en cada nodo OConS al encenderse y en él se llevan a cabo los procedimientos básicos que habiliten la posterior orquestación de mecanismos; por otro lado inicia los servicios básicos de conectividad. Se puede estructurar de acuerdo a tres etapas: descubrimiento de entidades, registro de mecanismos y composición de servicios por defecto; tal como se muestra en la Figura 2.5 (por simplificar, los reconocimientos mediante *replies* se han omitido y no se ha representado el INC que está presente en todas las comunicaciones). Para hacer la representación lo más genérica posible, no se ha indicado si los componentes están en el mismo o diferentes nodos (lo que solventaría el INC) y así se da una mejor idea del nivel de abstracción (con respecto a la tecnología de comunicaciones) que se da a nivel de entidades y demás componentes.

Como se puede ver, en la primera etapa las entidades se registran en el OR indicando su identificador y capacidades. En la segunda el SOP se suscribe para ser notificado de los mecanismos disponibles, notar que esta suscripción puede ser a una base de datos remota. Finalmente, los servicios que hayan sido configurados para iniciarse por defecto se ejecutan. Para ello el SOP interroga al OR sobre los mecanismos definidos como *default* y mediante su instancia compone los servicios necesarios. En esta etapa también se configurará la monitorización de la red para dichos servicios, esto permite modificar su configuración de manera dinámica ante cambios en el estado de la red o del contexto (políticas de usuario, por ejemplo).

Implementación de OConS

Una vez presentada la arquitectura OConS, a lo largo de este capítulo se va a describir, de forma detallada, la implementación que se ha llevado a cabo y que ha permitido el desarrollo de una propuesta de movilidad que será presentada en el Capítulo 4 y que sirvió como prueba de concepto de la arquitectura OConS dentro del proyecto SAIL.

El marco OConS ha sido concebido para dar soporte a servicios de conectividad de naturaleza heterogénea, así como para permitir el desarrollo de servicios nuevos de una manera uniforme. De este modo se intenta evitar soluciones monolíticas cuyos componentes no puedan ser reutilizados y cuya funcionalidad no se pueda combinar con otras.

En este sentido, dado el elevado número de mecanismos que la arquitectura debería ser capaz de integrar y orquestar, se hace necesario el desarrollo de prototipos o demostradores que sirvan como prueba de concepto que permitan salvar la distancia entre la descripción de la arquitectura y su implementación en plataformas realistas. Con este objetivo en mente, a fin de facilitar futuras implementaciones se han desarrollado una serie de herramientas (módulos de la arquitectura) genéricas y reutilizables que permiten la creación y extensión de mecanismos; estos módulos pueden ser utilizados tanto por mecanismos novedosos como por otros ya asentados a los que se les quiera hacer formar parte de la arquitectura.

Este capítulo se divide en dos partes, en la primera (3.1) se van a presentar los componentes reutilizables desde la señalización básica hasta la orquestación, pasando por el desarrollo de entidades genéricas de información (IE) y ejecución (EE). Por otro lado, en la segunda parte 3.2 se describirán las capacidades de orquestación que surgen de la interacción de los componentes anteriores.

3.1. Descripción funcional

El aspecto básico ha llevar a cabo para asegurar la apropiada operación de OConS es la comunicación entre entidades, ya sea de manera local o remota. Entendiendo la arquitectura como un conjunto distribuido de componentes para los cuales no está totalmente definida la conectividad (por ejemplo, pueden tener diferentes sistemas de comunicaciones) debido a su naturaleza *cross-layer*, resulta clave ofrecer un sistema de comunicaciones entre entidades que “desacople” la señalización OConS de la tecnología de comunicaciones subyacente. Como se ha descrito en el Capítulo 2 este papel se lleva a cabo por el INC, que se puede entender como un *hub* encargado de encapsular, cuando sea necesario, y reenviar los mensajes a su destino.

A fin de abordar estos requerimientos comunicación presentes en cada nodo de la arquitectura, se ha desarrollado una librería genérica que implementa la funcionalidad INC así como los diferentes interfaces de las entidades. Siguiendo el ejemplo de otras implementaciones como Open Dot Twenty ONE (ODTONE) [10], se ha optado por realizar la librería en lenguaje de programación *C++*. Las funcionalidades más importantes que provee esta librería se describen a continuación comenzando por aspectos de más alto nivel para describir, posteriormente, los de nivel de abstracción más bajo:

- INC: esta funcionalidad se ha implementado como un “demonio” que se encuentra permanentemente a la escucha en los diferentes interfaces de comunicaciones del nodo en el que se encuentre, así como en un interfaz local; de este modo es capaz de distinguir señalización proveniente tanto de entidades locales como remotas. Aunque puede ser fácilmente modificado de acuerdo al tipo de nodo en el que se encuentre, debido a su modularidad, en lo referente a la escucha local el INC hace uso de sistemas IPC, concretamente *Unix Domain Sockets*. Además de escuchar los diferentes interfaces, este módulo realiza el reenvío de los mensajes a las entidades locales o remotas. Para llevar a cabo estos reenvíos, el INC almacena una tabla identificadores globales de entidad (esto es, el identificador de nodo junto al de entidad local) y lo mapea en el interfaz correspondiente. Durante la resolución de direcciones, en primer lugar se mira el *node ID* y en caso de ser remoto se hace uso del interfaz de red correspondiente sin inspeccionar el *entity ID*. Si resultara que el reenvío se tuviera que hacer a una entidad local se usaría el *entity ID* para enviarlo por el *socket* local correspondiente.

Además, debido a los dos niveles de direccionamiento, y por lo tanto de comunicación (entre entidades y entre nodos), este módulo es también capaz de encapsular varios mensajes hacia un mismo nodo posibilitando el uso de “piggy-backing”. En terminología OConS varios mensajes (con diferentes cabeceras de entidad) se envían en un mismo paquete (bajo una misma cabecera de nodo).

- Manipulación de mensajes: ya sea el INC o las diferentes entidades que componen los mecanismo, se los debe dotar de la capacidad de manejar los mensajes y paquetes. La librería también ofrece una serie de funcionalidades (“helpers”) que permiten, de una manera sencilla, la manipulación, lectura y creación tanto de las cabeceras como de la información de los mensajes. En particular, la librería ofrece una manera simplificada para la encapsulación/des-encapsulación de datos en/desde un mensaje y serialización/des-serIALIZACIÓN de mensajes en/desde paquetes. Además de lo anterior, se simplifica la creación y lectura de cabeceras consiguiendo, de este modo, simplificar tanto la lectura como la creación de la señalización OConS.
- Manipulación de TLVs: finalmente, y como soporte a la señalización, la librería ofrece la formación de datos. En este sentido, se soporta la creación y lectura de datos TLV de una manera genérica a fin de facilitar la abstracción de datos y evitando la labor tediosa y propensa a errores de la manipulación de nivel de bit. Aunque conceptualmente, este aspecto no resulta muy importante, desde el punto de vista de desarrollo el uso de técnicas de programación genérica que permitan a un cliente de la librería crear datos TLV sin preocuparse del tamaño de los datos y la gestión de memoria, resulta clave en términos de productividad en creación de código y especialmente en su depuración.

Con los tres componentes principales de la librería genérica, se dota a los nodos OConS de las capacidades necesarias (comunicación) para poder abordar el desarrollo de servicios de conectividad de una forma uniforme y simplificada.

El siguiente paso debería ser la implementación del servicio *per se*, es decir, el desarrollo de las diferentes entidades que constituyen los mecanismos. En este punto, vuelve a estar presente la idea de generalidad siempre ligada a las arquitecturas abiertas. Si se piensa en las IEs de diferentes mecanismos, se puede llegar a la conclusión de que, en términos de diseño, son esencialmente los mismo: un módulo software con un interfaz definido (el conjunto de mensajes que usa no varía) capaz de obtener información y enviarla. El mismo razonamiento se puede aplicar a las EEs (cambiando el obtener datos por realizar una acción), no así a las entidades de decisión (DEs), cuya posible complejidad y diversidad en los tipos de datos que pueden manejar sus algoritmos, hacen que un diseño genérico sea muy complejo y seguramente carente de características requeridas en futuros desarrollos.

Con lo anterior en mente, se ha realizado una implementación genérica de las IEs y EEs haciendo uso del patrón de diseño conductual *Template Method* [11], que se ajusta a nuestros requerimientos. Dado que esta implementación hará uso de la librería genérica, y que el lenguaje *C++* posee de forma nativa el concepto de plantilla o *template*, el desarrollo se ha llevado a cabo también en este lenguaje.

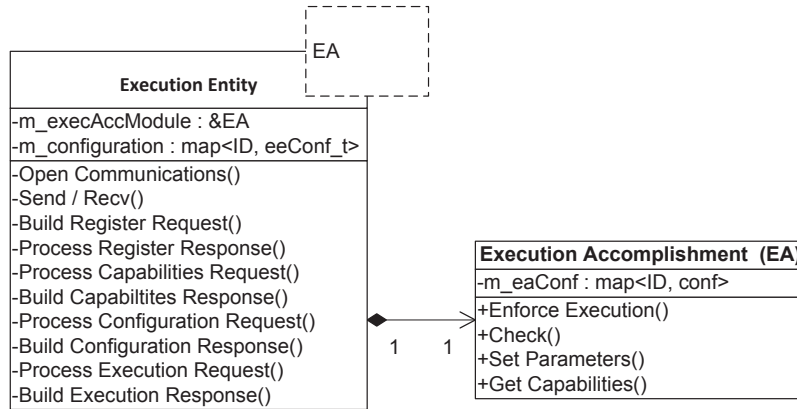


Fig. 3.1: Estructura de la clase-plantilla genérica EE

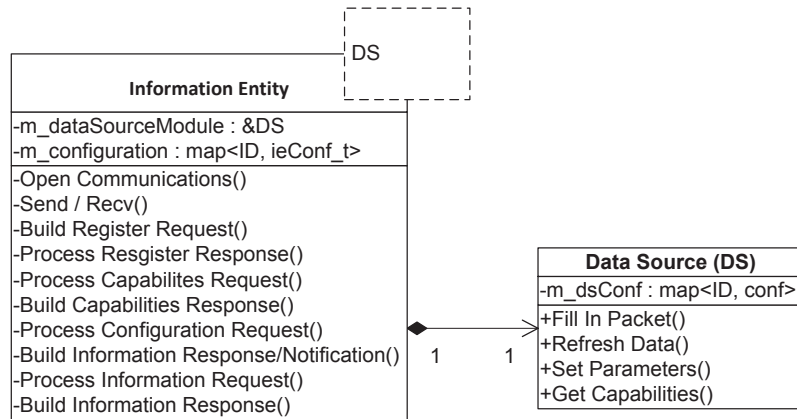


Fig. 3.2: Estructura de la clase-plantilla genérica IE

En términos software, las entidades IEs y EEs se definen como clases-plantilla (*templated classes*) como se muestran en los diagramas 3.2 y 3.1 correspondientemente. En ambos casos, las clases implementan un estructura básica que da soporte a las diferentes necesidades de comunicación, para lo cual se hace uso de la librería genérica. Además, de acuerdo a la implementación particular de diferentes entidades que conformen los servicios, estas clases-plantilla se especializan por medio de un parámetro (de hecho, otra clase: Data Source (DS) o Execution Accomplishment (EA) según el tipo de entidad) implementado para llevar a cabo acciones particulares.

Desde el punto de vista de implementación, las clases-plantilla abstraen de los aspectos relacionados con OConS, permitiendo el desarrollo de entidades únicamente definiendo las acciones particulares que se llevarían a cabo. Otro aspecto quizá más importante de este diseño genérico, es la posibilidad de facilitar la migración de servicios no-OConS a servicios OConS. Las Figuras 3.2 y 3.1 muestran un conjunto de métodos relacionados con la comunicación entre entidades (concretamente con la creación y procesamiento de los mensajes) que hacen uso de la librería descrita anteriormente. A su vez, las clases que especializan las entidades (clases interiores) presentan un interfaz común que permiten la coordinación de la creación de dichos mensajes y de acuerdo a la configuración o configuraciones que, en

un momento determinado, presente la entidad. Con todo esto, si se tuviera un desarrollo capaz de adquirir un determinado tipo de información (por ejemplo, suscripciones al *kernel* para conocer parámetros de bajo nivel de la comunicación), se podría definir una clase interior que mapeara las llamadas OConS a procedimiento concretos, llamadas al *kernel* en este caso, sin modificar el desarrollo anterior. Lo mismo se podrían aplicar a fragmentos de mecanismos ya existentes, por lo que estas clases interiores hacen la labor de *wrappers* que permitan a procedimientos ya implementados formar parte de OConS.

Debido a gran abanico de entidades que pueden formar un determinado mecanismo OConS, se hace necesario definir un conjunto de parámetros de configuración que se consideren genéricos. Además, la posibilidad de reutilizar las entidades por varios mecanismos (una determinada información puede ser usada por varios mecanismos) es siempre deseable. Por estas razones, la configuración de estas entidades genéricas cubre ambos aspectos: dar soporte a multi-configuración y a la configuración genérica.

En esta implementación en particular, las entidades genéricas admiten configuraciones múltiples que pueden provenir de varios DEs, así como del SOP para tareas de monitorización. Estas configuraciones se almacenan, como se muestra en las Figuras 3.2 y 3.1 (concretamente en la variable *m_configuration*), permitiendo el reuso de las entidades por varios mecanismos. Esta característica es especialmente relevante para las IEs, cuya información puede ser pertinente en varios mecanismos, y para las EEs, especialmente si varios mecanismos hacen uso de los mismos protocolos de red (por ejemplo, en tareas de gestión de recursos).

En concreto, la configuración en las entidades se lleva a cabo en dos etapas: al recibir un mensaje *ConfigurationRequest*, la clase genérica toma los parámetros de configuración básica y reenvía (mediante el método *SetParameters*) el mensaje a la clase interior; llegado a este punto, esta última toma los parámetros de configuración específicos para el mecanismo. La configuración particular de ambos tipos de entidades se realiza mediante el envío de unos TLV de tipo secuencia (véase [9] para la descripción de tipos TLV) que se almacenan en estructuras de datos específicas (*ieConf_t* y *eeConf_t*), a continuación se describe cada una de las configuraciones:

■ *ieConf_t*

Parámetro *notPeriod*: establece el periodo de notificación después del cual el IE debe informar al solicitante de información con el valor actual. Este parámetro lo almacena la clase genérica y en el momento en que el temporizador expira se llama al método *FillInMessage* para obtener la información.

Parámetro *threshold*: como su nombre indica, este parámetro establece un umbral definido como la variación relativa de la métrica en cuestión respecto a la medida anterior. Como la métrica puede diferir en gran medida de unas entidades a otras (especialmente en el tipo de dato que la contiene), este parámetro es usado por la clase interior.

Parámetro *checkPeriod*: por su parte, este parámetro establece el intervalo temporal que se usa para tomar una nueva muestra de la métrica y decidir si el umbral se ha excedido. Una vez que expira este período, se realiza una llamada al método *RefreshData* que indica si el umbral ha sido sobrepasado o no para un solicitante en particular. En caso de que el umbral se haya excedido se haría una llamada al método *FillInPacket* con el fin de obtener la información actual e informar al DE o SOP que hubiera realizado la configuración.

■ *eeConf_t*

Parámetro *numAttempts*: este parámetro fija el número máximo de intentos que el EE debe realizar (en caso de ejecuciones fallidas) antes de enviar el mensaje de tipo *ExecutionResponse* como respuesta a un *ExecutionRequest*. Este parámetro es usado por la clase exterior junto a los métodos *EnforceExecution* y *Check* que le permiten ser consciente de si la ejecución se ha llevado a cabo de forma correcta.

Además de la configuración genérica que se ha detallado, en ambos casos el TLV de tipo secuencia contiene configuración específica destinada a la clase interior que especifica la información concreta a la que se suscribe (en el caso del IE) o para indicar respuestas a medida a cerca del estado de la ejecución, para el caso del EE. Por ejemplo, si un IE contiene el perfil de usuario, un determinado DE puede estar únicamente interesado en ciertos parámetros de dicho usuario, por lo que indicará en su suscripción que únicamente se le debe informar con algunos parámetros del conjunto.

3.2. Orquestación

El principal objetivo de la arquitectura OConS es el de proveer de un marco en el que se pueda incluir cualquier tipo de mecanismo de conectividad. Esto requiere, como se ha expuesto, un gran nivel de abstracción en la definición de los componentes lo conforman; a fin de tratar esta generalidad y versatilidad se han definido tres componentes básicos (IE, EE y DE), así como los interfaces que permiten la comunicación entre ellos.

Además, se debe dar soporte a la configuración dinámica de estos componentes básicos de forma que la configuración de los servicios de conectividad sea capaz de adaptarse a variaciones en el estado de la red y a nuevos mecanismos que mejores, modifiquen o, en general, se incluyan en el servicio. Por ello, las instancias de los componentes básicos deben ser correctamente coordinadas; esta coordinación está a cargo de funciones de orquestación.

Basadas en la librería previamente descrita en 3.1, se han implementado algunas de las funcionalidades básicas de orquestación capaces de establecer una operatividad básica de los mecanismos de conectividad; el

caso de uso en el que se han incluido estas funciones será descrito en profundidad en el Capítulo 5. Las funcionalidades implementadas se adecúan a la descripción de la arquitectura de referencia presentada en el Capítulo 2 y se describen a continuación:

- Registro de entidades y sus capacidades en el OR local: mediante el envío de mensajes de tipo **REGISTER_req** las entidades son capaces de registrar sus capacidades y adquirir los identificadores necesarios (tanto a nivel de nodo como el propio de entidad local). Al OR se le ha dotado de la capacidad de hacer un seguimiento de todas las entidades conocidas permitiendo la incorporación dinámica de nuevas entidades o la modificación de sus capacidades. Cabe destacar que el OR local también contiene las definiciones de mecanismos (como agregación de capacidades).
- Verificación de los mecanismos: esto tiene lugar una vez que las entidades se han registrado y el SOP comprueba a través del OR si los mecanismos que se requieren se encuentran disponibles. Esta capacidad de verificación incluye la actualización del mecanismo solicitado en el caso de que en algún momento las capacidades que lo componen se van modificadas. Este último caso puede dar lugar a mejoras en el comportamiento de los mecanismos debido a la aparición de nuevas entidades (por ejemplo, fuentes de información) cuyas capacidades enriquezcan el mecanismo de conectividad en cuestión.
- Registro de entidades remotas: dado que la comunicación de entidades se sitúa en un nivel de abstracción por encima de la topología de la red, éstas deben ser tratadas de la misma manera tanto si son locales como remotas. Por ello, se ha dotado de la capacidad de descubrimiento y registro de entidades remotas en un OR local de manera que los diferentes OR de los nodos sean conscientes de las capacidades de las que pueden hacer uso. Este registro remoto da soporte a la mejora o enriquecimiento de los mecanismos descrito anteriormente.
- Configuración de entidades por el SOP: una vez que se comprueba que un mecanismo puede ser iniciado (el SOP confirma positivamente, a través del OR, que el mecanismo está disponible), el SOP configura el o los DEs mediante el envío de un **CONFIGURE_req**, de modo que habilita la comunicación del DE con el resto de entidades que forman parte del mecanismo. De igual manera, si el mecanismo se ve modificado (por ejemplo, aparecen nuevas entidades) sucesivos mensajes del mismo tipo actualizarán las comunicaciones entre las entidades.
- Configuración de nivel bajo: a nivel de mecanismo, se produce un almacenamiento secundario de información de entidades en los DEs, lo que permite mantener la información a cerca de los IDs de las entidades involucradas en el mecanismo así como la información necesaria a cerca de la configuración particular que es preciso enviar

a las entidades según el funcionamiento propio del DE (tal como se ha descrito en los niveles de configuración de las entidades genéricas previamente comentados). Ante la recepción del mensaje **CONFIGURE_req** desde el SOP, el DE almacena la información de las entidades y les envía la configuración pertinente.

Como conclusión, a partir de la librería de comunicaciones y teniendo en cuenta el diseño de las entidades genéricas, se han desarrollado una serie de componentes de orquestación que permite la correcta configuración y ejecución de los mecanismos que conforman un determinado servicio de conectividad. Estas funcionalidades se han desarrollado como módulos software que, si bien tienen un comportamiento diferenciado (abstraídos por medio de clases con interfaces bien definidos), y dado que proporcionan funcionalidad local, se han integrado en una solución que incluye tanto la gestión de la comunicación (INC) como la orquestación de los mecanismos de los mecanismos.

Propuesta de movilidad

A partir de la descripción del marco de referencia OConS y de los componentes desarrollados, se ha llevado a cabo la implementación de una solución de movilidad que posee un doble objetivo. Por un lado, proporcionar una comunicación sin costuras que permita a los usuarios mantener servicios activos a medida que varían su posición; además, se busca presentar una prueba de concepto que ilustre el funcionamiento real de la arquitectura en un entorno controlado.

Antes de proseguir, se van a realizar una serie de definiciones de términos de movilidad y de su clasificación. A pesar de que en muchos casos se utilicen indistintamente, cabe aclarar la diferencia entre *handoff* y *handover* o traspaso. El primero se refiere al cambio de medio físico mientras que el segundo se sitúa un nivel por encima e implica tanto el cambio de medio físico como el traspaso de la sesión o sesiones que estén en curso, esto, además, implica una necesaria provisión de recursos.

En cuanto al *handoff*, se clasifica atendiendo a propiedades del medio físico. Se habla de *hard handoff* cuando un terminal abandona un medio para conectarse a otro, esto implica una pérdida de paquetes durante el tiempo de re-conexión. Como alternativa al *hard handoff* aparece el termino *semi-soft handoff*, que se refiere a la capacidad de mantener dos conexiones activas, la antigua y la nueva, durante un cierto tiempo, lo que mejora el rendimiento en cuanto a pérdida de paquetes.

En lo que a los *handovers* se refiere, éstos admiten varios tipos de clasificación, sin embargo únicamente se van a comentar las clasificaciones de las que se hará uso: la que atiende a propiedades del rendimiento del traspaso y la que atiende a las tecnología involucradas.

Atendiendo al rendimiento se caracteriza la pérdida de paquetes y el retardo, de este modo los trasposos se dividen en:

- *Fast Handover*: traspaso de duración mínima, mínimo retardo.

- *Smooth handover*: traspaso de pérdida mínima de paquetes.
- *Seamless handover*: traspaso sin pérdida de calidad según la percepción del usuario/aplicación.

Por otro lado, cuando lo que se estudia son las tecnologías involucradas, los *handovers* se dividen en:

- *Handover* horizontal: en los casos en que la tecnología de acceso antes y después del *handoff* es la misma.
- *Handover* vertical: corresponde a los casos en que la tecnología antes y después del *handoff* son diferentes.

En este capítulo se va a describir la solución propuesta y se va a comparar con el estándar IEEE 802.21 [4], más conocida como *Media Independent Handover*. Esta comparación está motivada por el interés que el estándar 802.21 suscita y que lo lleva a ser considerado como referencia en el desarrollo de soluciones de movilidad genéricas (tanto vertical como horizontal).

Así, en la Sección 4.1 se va a describir el servicio de conectividad OConS que da lugar a la solución propuesta; en la Sección 4.2 se presentará de manera general el estándar 802.21; finalmente en la Sección 4.3 se hará una comparación entre ellas.

4.1. Solución de movilidad

La solución propuesta está compuesta por dos mecanismos diferentes cuya interacción ofrece un servicio de conectividad mejorado. La implementación de esta plataforma ha sido publicada en [12] y mostrada en la misma conferencia, *4th International Conference on Mobile Networks and Management* (MONAMI).

En primer lugar, cabe indicar que se trata de una solución de movilidad centrada en el usuario (*user-centric*) y que, por lo tanto, es éste quien toma la decisión de realizar traspasos. En terminología OConS, el servicio de movilidad se compone de dos mecanismos: selección de acceso y gestión dinámica de movilidad.

El mecanismo de selección de acceso es capaz de obtener información de diferentes elementos de la red y de diferentes niveles; con esta información el mecanismo es capaz de tomar una decisión aplicando una función de múltiples parámetros. A pesar de que la información utilizada para tomar las decisiones pertenece a varias capas, las acciones que se lleven a cabo afectan únicamente a la capa de enlace. Esto significa que mientras los IEs se encuentran en capas diferentes el EE ejecuta acciones de capa de enlace.

Por otro lado, la gestión dinámica de movilidad ofrece una asignación dinámica de puntos de referencia en la red de acceso (direcciones IP activas). La idea que subyace tras este mecanismo es la de mejorar el enrutamiento de los flujos de tráfico IP en la parte de acceso de la red; esto se lleva a cabo mediante la creación dinámica de túneles IPv6. De esta

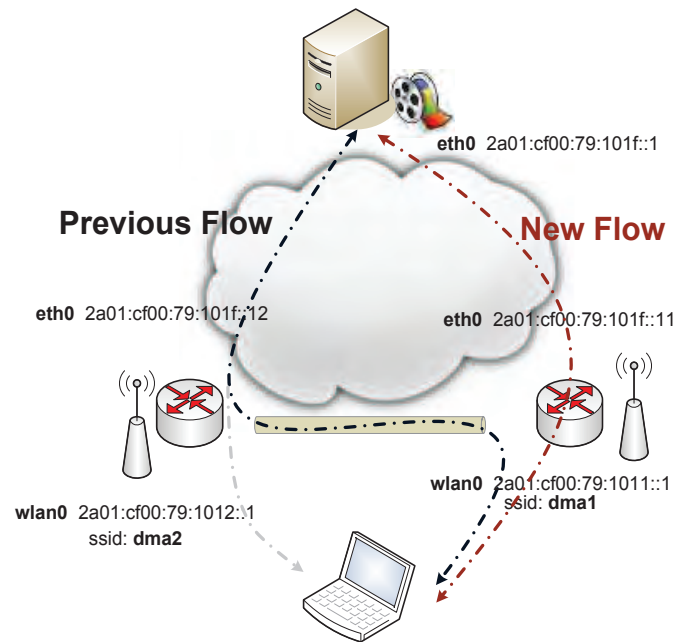


Fig. 4.1: Escenario de movilidad

manera, cuando una sesión (con un flujo de datos asociado) ha sido iniciada en el elemento de acceso al que está conectado el encaminamiento se realiza de manera directa; por otro lado, si la sesión ha sido iniciada en otro nodo de acceso, el flujo se reenvía a través de un túnel al nodo en que se inició, mientras que los nuevos flujos usan como referencia el nodo actual. En la Figura 4.1 se muestra el escenario de referencia de este servicio.

A continuación se va a describir con más detalle cada uno de los mecanismos y la integración de ellos. Cabe aclarar que el mecanismo de gestión de movilidad fue propuesto y desarrollado por otro integrante del proyecto SAIL. Esta propuesta ha sido presentada como Request for Comments (RFC) al Internet Engineering Task Force (IETF) y presentado públicamente en la reunión del IETF que tuvo lugar el 29 de Marzo de 2012 en Paris.

4.1.1. Selección de acceso

Uno de los objetivos de este desarrollo ha sido el de explotar los beneficios que la implementación genérica, descrita en el Capítulo 3, presenta en lo que se refiere a despliegue de mecanismos.

Desde la perspectiva OConS el mecanismo está formado por un DE que obtiene información de varios IEs y hace uso de un EE para llevar a cabo la conexión a un nuevo punto de acceso. En lo referente a las tareas de orquestación, se han implementado tanto el OR como el SOP usando las diferentes funcionalidades descritas en el Capítulo 3. Por otro lado, la comunicación entre las diferentes entidades (o entre ellas y el INC) hace uso de la librería genérica que se ha descrito. Finalmente, las entidades IEs y EEs que forman parte del mecanismo se han desarrollado a partir

de las entidades plantilla, a las que se ha dotado de un comportamiento específico.

En este caso, todos los nodos que intervienen en el mecanismo poseen las tareas de orquestación, lo que implica que implementan el INC, SOP y OR. Esto es así porque se ha asumido que no existe una conectividad previa, por lo que los nodos deben poseer cierta autonomía (por ejemplo, deben ser capaces de registrar entidades).

A fin de ilustrar la capacidad de abstracción de la arquitectura OConS en lo que se refiere a la distribución de entidades, éstas se han distribuido entre el terminal de usuario y la red. De esta manera el mecanismo está compuesto por un DE y un EE situados en el nodo de usuario y un conjunto de IEs que proporcionan la información de contexto para tomar la decisión. Los diferentes IEs se describen a continuación:

- *IE_{LQ} (Link Quality)*: esta entidad escanea los radio-canales (concretamente canales *WiFi*) a fin de informar a sus suscriptores acerca de la calidad del enlace, Signal to Noise Ratio (SNR) o Received Radio Strength Indicator (RSSI), de los puntos de acceso disponibles. La entidad se sitúa en el terminal de usuario.
- *IE_{UP} (User Profile)*: dado que no todos los usuarios son iguales ni tienen las mismas necesidades, resulta necesario tener en cuenta el tipo de usuario a la hora de seleccionar el acceso que mejor se adapte a sus características. Para ello esta entidad informa acerca de los parámetros y preferencias del suscriptor del terminal; por ejemplo, redes permitidas, prohibidas o favoritas. Como en el caso anterior, esta entidad se sitúa en el terminal de usuario.
- *IE_{TL}*: este módulo de información monitoriza la carga de tráfico en los routers de acceso. Esta entidad está situada en la parte de red y es accesible por el nodo móvil cuando se conecta a cualquiera de las redes de acceso.

A partir de la información proporcionada por las diferentes entidades de información, el DE es capaz de tomar diferentes decisiones de acuerdo al tipo de usuario. El algoritmo básico se ha implementado como una función lineal de los diferentes parámetros (calidad del enlace y carga de la red) a los que se les modula con diferentes pesos en función del tipo de usuario del que se trate. De esta manera el algoritmo es capaz de dar preferencia a la QoS percibida por el usuario, asignando un mayor peso a la calidad del enlace, o a la gestión de la red a través del balanceo de carga, asignando mayor peso al parámetro de carga.

Además de la integración de los diferentes componentes presentados anteriormente, se ha desarrollado una herramienta de visualización que permite monitorizar los diferentes procedimientos que se están llevando a cabo de forma similar a un analizador de protocolos, una muestra se puede ver en la Figura 4.2. Esta herramienta muestra los mensajes que se intercambian entre las diferentes entidades de la arquitectura, ya sean locales o remotas, y se encuentra asociada al INC.

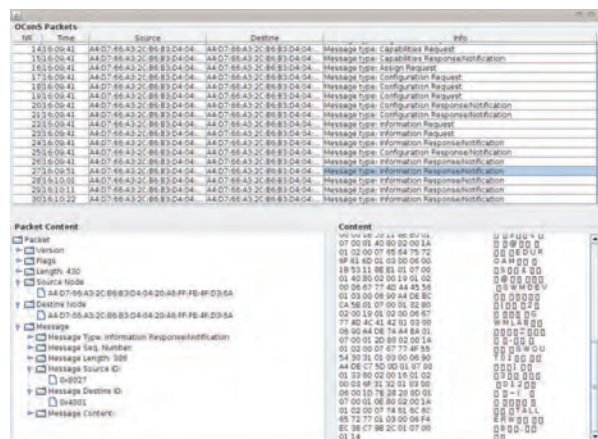


Fig. 4.2: Herramienta de visualización

4.1.2. Gestión dinámica de movilidad

El objetivo de este mecanismo ha sido el de crear de manera dinámica punto de referencia en los routers de acceso, Access Router (AR), de modo que se pueda realizar encaminamiento IP directo del tráfico de los flujos o sesiones iniciados en el AR actual. Por otro lado, con el objetivo de lograr traspasos que se puedan considerar *seamless*, aquellas sesiones que que hayan sido iniciadas en otros routers son reenviadas a ellos por medio de túneles. Este reenvío hace que no sea necesario volver a establecer la sesión con el servidor.

El demostrador que se integró y mostró públicamente en [12] muestra un ejemplo mediante sesiones de video de acuerdo a la Figura 4.1. Así, si un usuario que se ha conectado inicialmente a la red de nombre *dma2* (y por lo tanto adquiere una dirección con prefijo $2a01:c:f00:79:1012::64$) inicia una sesión de video, estando en esta red, con el servidor (indicado con una línea gris en la figura), el flujo de datos de la sesión se encamina de forma directa al servidor. En algún momento, mientras el servicio permanece activo, se realiza un *handover* a la red *dma1*, lo que, en circunstancias normales provocaría que se debiera re-iniciar la sesión entre la nueva dirección adquirida en la nueva red y el servidor. Sin embargo, la gestión dinámica de los puntos de referencia (*anchors*) permite que la red de acceso se encargue de abstraer al servidor de los cambios que se han producido. La solución hace uso de la capacidad *multi-home* (soporte de varias direcciones simultáneamente) inherente al protocolo IPv6, y que permite al nodo mantener la dirección antigua y obtener una nueva (en este caso con prefijo $2a01:c:f00:79:1011::64$) en la misma interfaz de red. Además, cuando la red detecta que un nodo con una sesión activa ha realizado un traspaso se establece un túnel entre el router de la red *dma2* y el de la red *dma1*, tal como se ilustra en la Figura 4.1. Finalmente, el router de la red *dma1* envía el tráfico recibido por el túnel a la dirección "antigua" del nodo móvil, haciendo el traspaso totalmente transparente a la sesión que se encontraba activa antes del *handover*. Por otro lado, las nuevas sesiones (ilustradas con una línea marrón en la figura) se encaminan

del manera directa por el AR actual.

4.1.3. Descripción del escenario

A pesar de que una de los principales objetivos de OConS es la interoperabilidad y fácil configuración, este no es el caso en la mayoría de las tecnologías subyacentes. Debido a esto, las opciones a la hora de implementar una plataforma experimental se ven limitadas a aquellos componentes de *hardware* cuyo *firmware* (*drivers*) ofrezca la posibilidad de acceder y manipular parámetros de bajo nivel. Además, es siempre deseable el uso de tecnologías que estén implementadas en la mayoría de los equipos. Considerando las limitaciones que puede tener, la plataforma ha tomado como tecnología de acceso el estándar del IEEE 802.11, mediante la cual la presencia de diferentes redes se emula desplegando puntos de acceso configurados para trabajar en canales ortogonales.

Este planteamiento, con indudables limitaciones, también posee ciertas ventajas. Se puede encontrar *firmware* para Wireless Fidelity (WiFi) que proporciona acceso a información e interfaces de control, ejemplo de ello es el proyecto *MadWiFi*¹. Además, otros proyectos como *hostap*² van incluso un paso mas allá y permiten a ordenadores convencionales comportarse como puntos de acceso que pueden ser fácilmente gestionados, lo que facilita el despliegue o modificaciones en el lado de la red.

Respecto al nivel de red, el sistema completo mostrado en la Figura 4.1 está basado en comunicaciones IPv6 y su funcionalidad ha sido implementada en el lenguaje de programación C/C++. El servicio se ha distribuido entre los diferentes nodos de la red (de acuerdo a la disposición de las entidades). En concreto el nodo móvil posee un sistema operativo *Ubuntu/Linux* (versión estable 10.04) mientras que los routers de acceso se han desplegado como máquinas virtuales que usan el sistema *Fedora/Linux* con versión de kernel 3.1.8 que ha sido modificada por *Orange Labs.* para dar soporte al mecanismo de gestión dinámica de movilidad.

Al nodo móvil se le ha provisto con una tarjeta de red *Atheros (ORINOCO 11 a/b/g ComboCard)* para dotarlo de conectividad. Esta tarjeta se puede manipular de manera sencilla mediante llamadas input/output control (ioctl) (usadas por el *firmware* de *MadWifi*) y es, por tanto, usada por las entidades *IE_{LQ}* y *EE* para escanear el medio radio y realizar la conexión respectivamente. La parte radio de los elementos de acceso usa adaptadores USB (*TL Link TL-WN821N*), los cuales, con ayuda del *firmware hostap* pueden ser configurados como punto de acceso.

¹<http://madwifi.org>

²<http://hostap.epitest.fi/hostap>

4.2. Estándar IEEE 802.21

La proliferación de dispositivos de altas prestaciones y dotados de varias tecnologías de acceso a precios asequibles es un hecho. La presencia de múltiples interfaces permite un mayor nivel de libertad en la selección de acceso, lo que, a su vez, complica la gestión de la movilidad de estos terminales. A fin de dar una solución de movilidad capaz de integrar las diferentes tecnologías, el IEEE presentó el estándar **802.21** que define una arquitectura que da soporte al traspaso de servicios abstrayendo el procedimiento de la tecnologías subyacentes; esta arquitectura recibe el nombre de MIH.

Esta arquitectura tiene como principal objetivo el proporcionar al usuario una conectividad permanente a través de la tecnología que presente las mejores condiciones en cada momento. Además, esta gestión de acceso se realiza de manera transparente al usuario (no precisa de su intervención directa), lo que garantiza la continuidad de las sesiones activas antes de realizar el *handoff* o cambio de tecnología.

Muchas de las funcionalidades necesarias para dar soporte a los traspasos de manera que se asegure la continuidad de las sesiones dependen de las tecnologías involucradas; es caso más básico es la detección de movimiento que de lugar a la aparición de nuevas alternativas. El estándar **802.21** [13] proporciona una abstracción de los mecanismos complejos de cada tecnología a través de un marco de señalización común; esto permite a las capas superiores (principalmente red y sesión) interactuar con las capas física y de enlace sin necesidad de conocer la tecnología involucrada.

De esta manera, el IEEE aseguró la compatibilidad con las principales tecnologías desplegadas que se indican a continuación:

- IEEE 802.3 (*Ethernet*)
- IEEE 802.11 (*WiFi*)
- IEEE 802.16 (*WiMAX*)
- 3GPP (*GSM/GPRS, UMTS/HSPA, LTE...*)
- 3GPP2 (*CDMA2000, EV-DO...*)

Las tareas del MIH comprende la inicialización del traspaso, el descubrimiento de redes adyacentes disponibles y la selección de acceso óptimo. Una vez definida la nueva conexión, la ejecución del traspaso queda fuera del ámbito del estándar y depende la tecnología seleccionada el realizar la conexión y de las capas superiores el garantizar que la sesión continúa (movilidad a nivel IP, por ejemplo).

La independencia de las capas superiores e inferiores se logra mediante una entidad independiente y abstracta, conocida como Media Independent Handover Function (MIHF). Esta entidad define un interfaz genérico entre los niveles superiores, llamados en el estándar **802.21** Media Independent Handover User (MIHU), y los inferiores, dependientes de la tecnología.

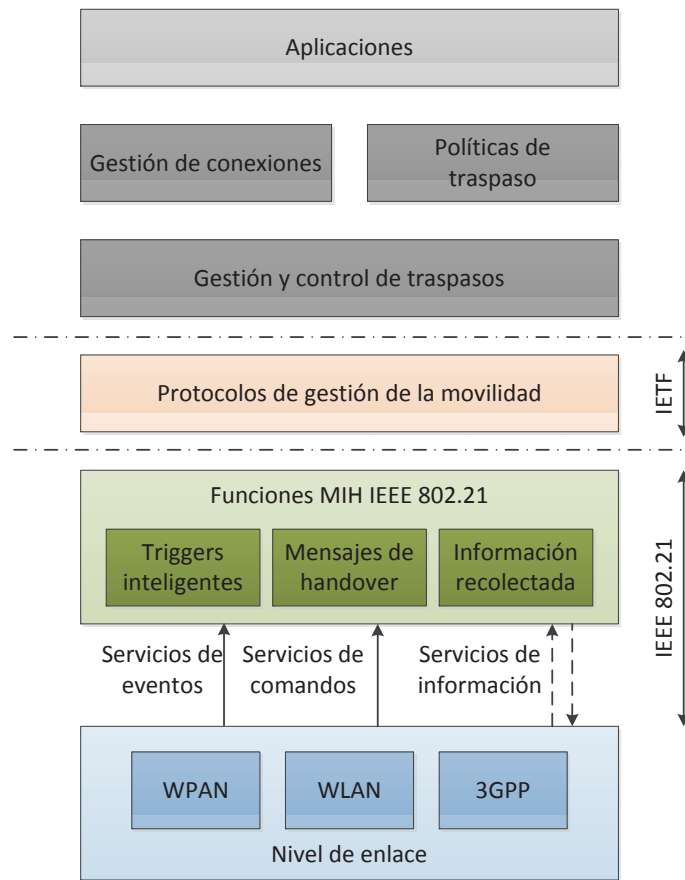


Fig. 4.3: Modelo de capas del estándar 802.21

Durante el proceso de traspaso, la información de los niveles inferiores y su gestión se presenta a los niveles superiores mediante una serie de puntos de acceso al servicio o Service Access Point (SAP). Además, se definen una serie de interfaces remotos que permiten la comunicación entre entidades separadas físicamente, esto habilita procedimientos de traspaso en los que la decisión no es autónoma (por ejemplo, decisiones tomadas por la red en entornos 3rd Generation Partnership Project (3GPP)) o adquirir información remota para la toma de decisiones.

De este modo, el MIHF actúa como una unidad intermedia entre las aplicaciones y las soluciones de gestión de movilidad y por encima de los niveles de enlace dependientes de la tecnología. En la Figura 4.3 se ilustra la visión general de la arquitectura MIH dentro del modelo de capas definido.

Como caso más concreto, en la Figura 4.4 se muestran los niveles de la arquitectura MIH para un dispositivo con interfaces IEEE 802.16 (Worldwide Interoperability for Microwave Access (WiMAX)) y 802.11 (WiFi). Se puede observar cómo los nodos que soportan el estándar 802.21 poseen una estructura común en torno a la entidad central MIHF. Esta entidad actúa como capa intermedia entre los niveles superiores (MIHUs) y los inferiores, permitiendo, de este modo, la coordinación del intercambio de información y comandos entre los diferentes elementos involucrados en la toma de decisiones y ejecución de los traspasos. El intercambio de información entre el

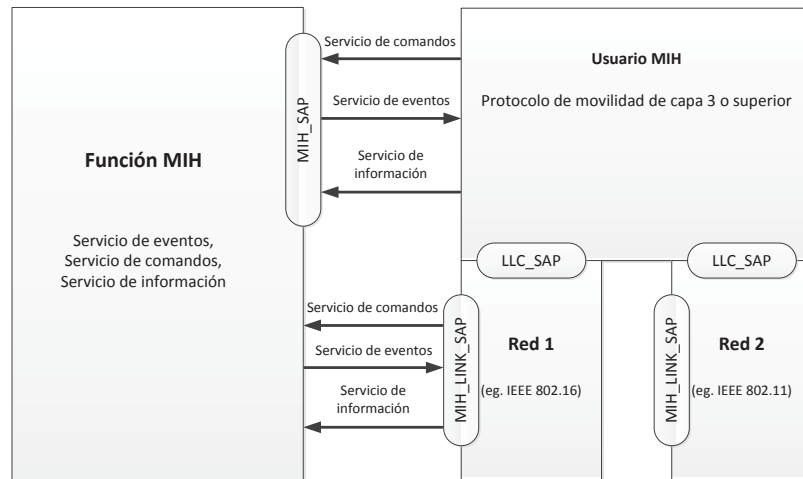


Fig. 4.4: Servicios MIH

MIHF y el resto de entidades se realiza a través de una serie de primitivas definidas en el estándar:

- MIH SAP: se trata del interfaz entre el MIHF y el nivel de enlace.
- MIH LINK SAP: es el interfaz entre el MIHF y los MIHUs ubicados en las capas superiores. Las primitivas empleadas en este punto son independientes de la tecnología subyacente, siendo ésta la responsable de interpretar y mapear los mensajes antes de hacerlos llegar a las capas inferiores.
- MIH NET SAP: este interfaz permite el intercambio de información entre MIHF situados en nodos diferentes, de manera remota.

Asimismo la norma define una serie de servicios para establecer la comunicación con el entidad central MIHF. Mediante estos servicios es posible acceder a la información relativa al establecimiento de un traspaso, enviar comandos a los niveles adyacentes o reaccionar de manera asíncrona a un sistema de notificaciones y eventos producidos en cualquier punto de la arquitectura MIH. A partir de esta descripción se realiza una clasificación de los servicios en tres categorías:

1. Servicios de eventos; Media Independent Event Service (MIES).

Se trata de un servicio basado en suscripciones mediante el cual cada nodo se registra al grupo de notificaciones del que desee obtener información. Cuando tiene lugar un evento (por ejemplo, la calidad de la señal traspasa un umbral establecido), la entidad correspondiente informa a los elementos registrados, los cuales han sido almacenados en una tabla de registro. Si una entidad recibe un mensaje sin haberse suscrito, éste será ignorado.

Los eventos pueden se pueden originar en los niveles inferiores (generalmente capa de enlace) o, incluso, dentro del propio MIHF, ya sea

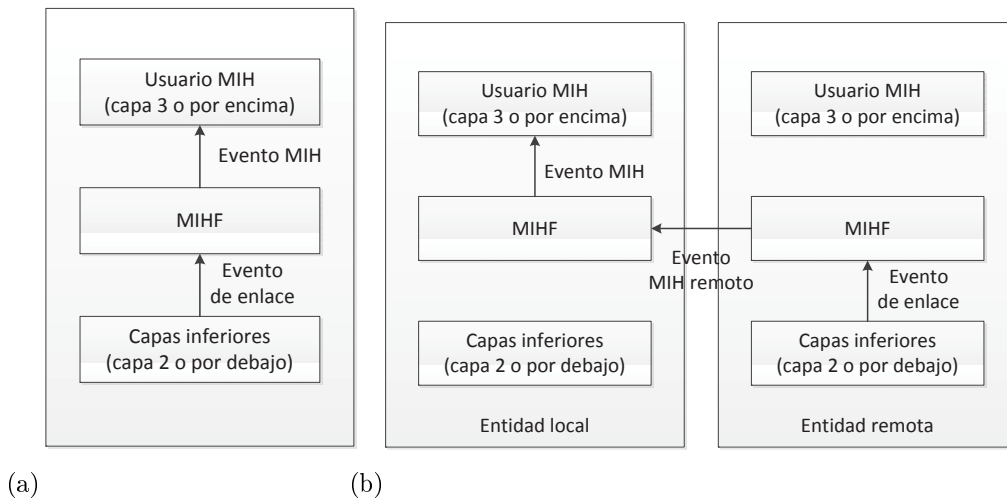


Fig. 4.5: Mapeo de servicios de eventos; 4.5(a) representa el modelo local mientras que 4.5(b) representa el remoto

de manera local o remota. En la Figura 4.5 se muestran los diferentes casos que siempre llevan un sentido ascendente.

En el estándar los eventos se clasifican en los siguientes grupos:

- Eventos de cambio de estado de capas física y enlace: se encargan de notificar cambios en los niveles inferiores, por ejemplo al insertar una nueva tarjeta de red o deshabilitar un interfaz de red via *software*.
- Eventos de los parámetros de enlace: se inician como consecuencia de un cambio en alguno de los parámetros de los niveles inferiores. La notificación de estos eventos puede realizarse de manera síncrona estableciendo un intervalo o asíncrona fijando un umbral.
- Eventos de sincronización de enlace: envían información relativa a los niveles inferiores y que son relevantes para las entidades MIHUs. Por ejemplo, cuando un interfaz de red (su nivel de enlace) decide prescindir de los gestión de movilidad que proporciona 802.21.
- Eventos de transmisión de enlace: notifican el estado de las conexiones. Un ejemplo ilustrativo se correspondería a la pérdida de conectividad a nivel de enlace tras realizar un traspaso.

2. Servicio de comandos; Media Independent Command Service (MICS).

Mediante este servicio se reciben los comandos enviados desde los protocolos de las capas superiores para de obtener información o controlar aspectos del terminal a fin de llevar a cabo el traspaso. Los comandos pueden ser tanto locales (dentro de un mismo nodo) como remotos (entre entidades situadas en nodos diferentes), tal como se ilustra en la Figura 4.6. Por medio de los comandos remotos se puede indicar a un terminal que debe realizar un traspaso.

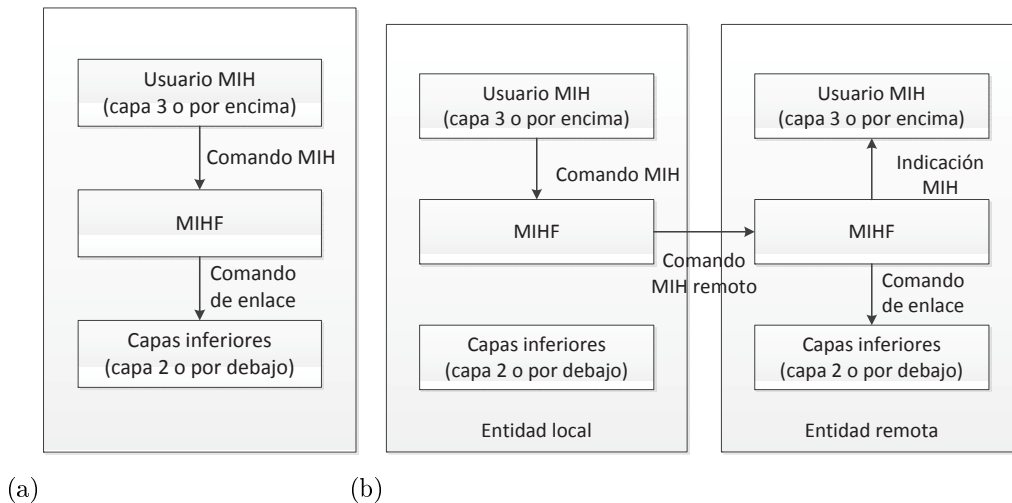


Fig. 4.6: Mapeo de comandos MIH; 4.6(a) representa el modelo local mientras que 4.6(b) representa el remoto

De forma similar a las notificaciones, los comandos también se agrupan en categorías:

- Comandos MIH: se envían desde las capas altas al MIHF. En caso de provenir de una entidad remota se entrega al MIHF local que procede a reenviarlo a la entidad pertinente a través de uno de los puntos de acceso al servicio definidos.

Cabe destacar que la funcionalidad de este servicio finaliza en el momento en el que da comienzo un traspaso; llegado a este punto los protocolos de gestión de movilidad serán los encargados de que el servicio finalice con éxito.

- Comandos de enlace: se originan siempre de manera local en el MIHF y tienen como propósito el configurar y controlar el comportamiento de los niveles inferiores. Este grupo de comandos son dependientes de la tecnología subyacente, lo que requiere la implementación de primitivas específicas para cada una de ellas.

3. Servicio de información; Media Independent Information Service (MIIS).

Este servicio ofrece un marco a través del cual el MIHF obtiene los fragmentos de información necesarios para inicializar los traspasos. Para lograr esto el servicio notifica acerca de parámetros muy diversos como zona de cobertura, mapas de conectividad u otros parámetros de nivel de enlace; de igual modo proporciona información de los niveles superiores (algoritmos de seguridad, servicios disponibles...). Estos elementos resultan cruciales a la hora de seleccionar el acceso óptimo ya que permiten elegir la conexión óptima en función de, por ejemplo, nivel de señal o mejor gestión de calidad de servicio.

El principio básico de este servicio es la cooperación entre los diferentes nodos que componen un escenario a través del envío periódico de

mensajes que completen el conocimiento de cada nodo en lo relativo a los elementos que lo rodean.

Los elementos de información se dividen en los siguientes grupos:

- Información general: indican el tipo de red, identificación del operador o proveedor del servicio...
- Información específica de la red de acceso: se trata de información relativa a cada tecnología de acceso y operador que la gestiona. Comprende aspectos diversos como los relacionados con la seguridad, provisión de QoS, opciones de *roaming*, tarificación...
- Información específica de los elementos de acceso: proporciona información sobre cada punto de conexión para cada tecnología de acceso y operador. Hace referencia a direcciones de capa física (dirección MAC), localización geográfica, capacidad del enlace...
- Información de capas superiores por punto de acceso: esta información está ligada a los servicios disponibles en el punto de acceso y la red. Por ejemplo, indica el número de subredes que se soportan, los métodos de configuración de direcciones IP o el listado de servicios soportados.
- Otra información: se trata de elementos no clasificados en el estándar. De este modo se deja la puerta abierta para dar soporte a futuros requerimientos.

Cabe destacar que los terminales de usuario han de ser capaces de determinar si la red a la que se conecta es o no compatible con el estándar IEEE 802.21. Esto implica el uso de métodos de descubrimiento o de obtención de información de los servicios. Esta información puede conseguirse incluso antes de la autenticación (si es necesaria) en un nuevo punto de acceso; de este modo se puede realizar la comprobación de los mecanismos de seguridad, soporte de QoS y otros parámetros de red antes de llevar a cabo el traspaso.

4.3. Comparativa

A pesar de que las dos soluciones de movilidad son difícilmente comparables, dado que una se trata de un estándar y la otra de la realización concreta de una propuesta de arquitectura, existen algunos aspectos, desde un punto de vista general, que merece la pena destacar.

La principal motivación del estándar 802.21 es el de permitir la interacción entre las capas superiores e inferiores mediante un interfaz común y, de esta manera, habilitar los traspasos tanto horizontales como verticales. Por otro lado, el modelo OConS no hace distinción entre capas, sino entre capacidades de las entidades. En este sentido, se trata de una visión más abierta.

En cuanto a la información, las piezas necesarias para la toma de decisiones de movilidad no se limitan a ciertas capas, sino que la única

premisa que deben cumplir es implementar el interfaz y haberse registrado. Aunque el estándar 802.21 también permite recoger información de niveles superiores e inferiores, existe una sutil diferencia entre una arquitectura por capas y otro en la que ese concepto prácticamente desaparece.

Si nos limitamos a la selección de acceso, la principal diferencia radica en la ejecución. El 802.21 se restringe a la adquisición de información y toma de decisiones y deja a las tecnologías concretas la ejecución. Sin embargo, OConS en general y la solución implementada en particular, integra la propia tecnología subyacente en la arquitectura (en este caso mediante la EE), lo que integra la decisión y la ejecución; como se ha explicado el EE está configurado por el DE e informa a éste del estado de la ejecución.

Pero sin duda la mayor diferencia radica en el ámbito de cada una de las soluciones. El estándar 802.21 se limita a la selección de acceso mientras que la arquitectura propuesta va un paso más allá al realizar funciones de encaminamiento. No parecería justo compararlas en este sentido ya que tienen objetivos diferentes, pero sí cabe indicar la facilidad de OConS a la cooperación entre capas y segmentos de red. Si se quisiera implementar una solución de movilidad que hiciera uso del estándar 802.21 y que diera soporte a encaminamiento ambas soluciones serían totalmente estancas, ya que el MIH no se ha diseñado para eso. Esto evidencia las carencias que OConS pretender solventar, falta de coordinación entre mecanismos y de expresividad en las reglas de cada uno (802.21 no se entendería con una solución de nivel de red porque hablan lenguajes diferentes).

Por otro lado, dentro del marco OConS la ejecución de una determinada decisión, como puede ser llevar a cabo un traspaso, puede ser seguida de procedimientos en diferentes niveles y segmentos de red para así proporcionar una determinada QoE al usuario final. En la solución que se ha mostrado existe una interrelación entre las decisiones de movilidad y su gestión en las capas superiores (creación de túneles), lo que podría ser mejorado si, por ejemplo, antes de ejecutar el traspaso se informara a la red para que fuera estableciendo el túnel y, de este modo, evitar todas las posibles pérdidas de paquetes. Esto requeriría que el DE indicase que se estableciera el túnel a la vez que indica que se realice el traspaso, algo perfectamente factible; incluso se podría implementar algún tipo de decisión distribuida.

Finalmente, indicar que otra de las ventajas de la arquitectura OConS radica en que lo abstracto de sus componentes la hacen fácilmente extensible. El hecho de añadir una entidad de información que monitorice otro parámetro no supondría variaciones en la arquitectura, sino una actualización del algoritmo de decisión.

Estudio analítico

A partir de la implementación de la solución de movilidad presentada anteriormente, se ha realizado un estudio analítico de diferentes soluciones sobre redes complejas, tanto en número de nodos como en heterogeneidad.

Este estudio, aunque de manera conceptual, respeta la arquitectura OConS, sin embargo resulta necesario dotarlo de un mayor nivel de abstracción con el fin de poder acometer campañas de simulación en redes con un número elevado de nodos.

El estudio analítico consta de dos partes: en primer lugar se ha definido la selección de acceso en el conjunto de la red como un problema de programación lineal y se ha desarrollado una metodología para hallar su solución y automatizarlo como se explicará en la Sección 5.1; en segundo lugar se ha realizado la comparativa de los resultados de la optimización con los obtenidos de la simulación de diferentes estrategias de selección de acceso, esta parte se detallará en la Sección 5.2.

5.1. Análisis teórico

Como se comentó en el capítulo introductorio, el aumento del tráfico que está surgiendo a raíz de las nuevas tecnologías y dispositivos debe ir acompañado de una mejor gestión de los recursos; esta optimización de los recursos es, si cabe, más importante en la parte de acceso radio en la que los recursos son muy limitados. El hecho de que los dispositivos estén dotados de diversas tecnologías es uno de los aspectos claves en lo referente a la gestión de los recursos radio y puede proporcionar mejoras tanto a la QoE percibida por el usuario como a los operadores de las redes de diversas maneras; por ejemplo, ayudando a la gestión de la red o incluso desde el punto de vista económico.

Con el propósito de mejorar el rendimiento desde la óptica de la selección de acceso se puede encontrar un elevado número de trabajos que

analizan diferentes técnicas, algoritmos y protocolos relacionados con gestión de recursos radio mediante el estudio de las nuevas posibilidades que brindan los elementos avanzados que han ido apareciendo con el tiempo. Un aspecto común de la mayoría de los trabajos que buscan una mejor operación de las redes heterogéneas, es que comparan el rendimiento logrado con otras alternativas o con las técnicas actuales, sin embargo no se evalúa lo cerca o lejos que estos resultados están del comportamiento óptimo [14], [15], [16]. De este modo, estos trabajos basan sus conclusiones en la comparación entre diferentes alternativas, proporcionando de esta manera un análisis cualitativo de las alternativas. Sin embargo son muy pocos los que tienen como objetivo hallar el comportamiento óptimo que se puede llegar a alcanzar para, de este modo, ser capaz de cuantificar la diferencia de rendimiento de las diferentes técnicas y algoritmos con respecto al mejor comportamiento posible. El principal objetivo de este trabajo es proponer un metodología para obtener la solución óptima, considerando diferentes figuras de mérito sobre redes altamente heterogéneas, en las cuales los usuarios finales cursan diferentes tipos de servicios.

En este sentido, esta sección presenta una metodología que trata estimar el mejor rendimiento que se podría esperar en una red heterogénea de acceso inalámbrico. A fin de proporcionar una respuesta razonable se plantea un problema de optimización que se modela como un problema de programación lineal. Para ello se define una función de utilidad genérica que es capaz de integrar diferentes figuras de mérito que tienen en cuenta tanto a los operadores de red (como puede ser la carga) como a los usuarios finales (como por ejemplo el precio).

Uno de los aspectos más relevantes de este trabajo es que en el planteamiento del problema se tiene en cuenta tanto la disposición de un usuario a tener una conexión activa (no todos los usuarios quieren estar siempre conectados) así como el resultado de las resoluciones anteriores del problema. Este último aspecto responde al consenso existente en que, desde el punto de vista de QoS, es preferible rechazar una nueva llamada a detener una que se encuentre en curso, por lo que el estado anterior de una conexión ha de tenerse en cuenta.

Como se ha mencionado anteriormente, el lema *Always Best Connected* fue propuesto originalmente por Gustafsson y Johnson en 2003 en [2]. Esta línea de investigación arraigó en la última década dando lugar a numerosas propuestas que buscan la gestión óptima de los recursos en redes de acceso inalámbrico proporcionando la mejor calidad de servicio (*QoS*) a los usuarios finales. Algunos de los trabajos realizados en esa época se mencionan en [17] y en sus referencias.

Con el paso del tiempo, se han propuesto nuevas posibilidades sustentadas en las nuevas técnicas y tecnologías que han aparecido. Un ejemplo claro es la capacidad de virtualizar recursos [18] como medio para ofrecer calidad de experiencia (*QoE*) a medida a los usuarios finales. Otra técnica que tiene un potencial impacto en el rendimiento de las redes de acceso inalámbrico es el *multi-path* (la posibilidad de dividir los flujos de tráfico por varios caminos con el fin de mejorar el rendimiento) que permiten los

dispositivos *multi-homed* [16]. Por otro lado, también se ha evidenciado que existen nuevos requerimientos a tener en cuenta, como es el caso del consumo energético [19]. La consideración de estos aspectos y el análisis de su influencia en el rendimiento y comportamiento de las *HetNets* es aún un aspecto que no se ha abordado.

Este trabajo es una evolución del marco que originariamente se presentó en [20]. Al trabajo previo se le ha añadido la capacidad de modelar de manera apropiada y diferenciada servicios de diferente naturaleza (como contrapartida al lema *Always Best Connected*), lo que ha tenido una clara influencia en cómo el problema ha de ser planteado y que ha llevado a la integración de un módulo que mantenga el histórico de las conexiones previas.

Como se ha mencionado, se propone el uso de programación lineal, concretamente programación lineal binaria, para obtener la solución óptima. Cabe destacar que no existen muchos trabajos que busquen este tipo de soluciones, aunque recientemente se han empleado otras técnicas matemáticas como pueden ser la teoría de juegos [21], [22] o la toma de decisiones multi-atributo [23]¹.

5.1.1. Formulación del problema

El problema de la selección de acceso trata de establecer la asociación óptima entre los flujos activos (servicios) de los usuarios entre las alternativas de acceso disponibles. El problema se ha definido sobre un escenario genérico que consta de N redes de tecnologías diferentes, por lo tanto dotadas de diferentes características en términos de cobertura y capacidad. También se asume que existen U usuarios finales, capaces de iniciar S servicios diferentes de forma simultánea; además se ha considerado que los usuarios poseen terminales capaces de establecer conexiones con todas las tecnologías presentes en el escenario.

En el planteamiento del problema se ha utilizado una formulación entera binaria que da lugar a $U \times N \times S$ variables básicas (x_{ijk}) que se definen como sigue:

$$x_{ijk} = \begin{cases} 1 & \text{si el usuario } i \text{ usa la red } j \text{ para el servicio } k \\ 0 & \text{en caso contrario} \end{cases} \quad (5.1)$$

Además, se incorpora una función de utilidad genérica (u_{ijk}), a fin de cuantificar el beneficio de una conexión en particular, que puede modularse de acuerdo a diferentes figuras de mérito. Es importante remarcar que esta utilidad, a pesar de que pueda depender de condiciones particulares (por ejemplo conectividad física), se mantiene constante en cada instancia del problema, garantizando de esta manera la linealidad del modelo (lo que quiere decir que u_{ijk} no depende de x_{ijk} en cada instancia). Asimismo, se definen tres restricciones al problema: 5.2b asegura que las variables básicas

¹En este caso, normalmente no se obtiene la solución óptima

se resuelven como variables binarias; 5.2c fuerza a que cada flujo esté asociado a un único acceso; mientras que 5.2d limita el número de conexiones que se pueden realizar sobre un acceso determinado (con capacidad C_j) de acuerdo a las capacidades requeridas por cada uno de los servicio(c_k). De esta manera, se puede plantear el problema como se muestra a continuación.

$$\text{Max.} \quad \sum_{i,j,k} u_{ijk} \cdot x_{ijk} \quad (5.2a)$$

$$\text{s.t.} \quad \forall i, j, k \quad x_{ijk} \in \{0, 1\} \quad (5.2b)$$

$$\forall i, k \quad \sum_j x_{ijk} \leq 1 \quad (5.2c)$$

$$\forall j \quad \sum_{i,k} x_{ijk} c_k \leq C_j \quad (5.2d)$$

Merece la pena aclarar que no todas las variables indicadas entran a formar parte del problema, ya que, por ejemplo, aquellas que carezcan de conectividad física entre el usuario i y la estación base j se descartarían ($x_{ijk} = 0 \quad \forall k$). De igual manera, cuando un determinado servicio (digamos k) se encuentra inactivo para un usuario i también se descartará añadiendo restricciones adicionales: $x_{ijk} = 0 \quad \forall j$.

La formulación presentada es una evolución de la propuesta en [20] a la cual se ha añadido la dimensión de servicio. A fin de reflejar una situación realista, cuando un servicio es rechazado o interrumpido (*dropped*) no se considera en las instancias sucesivas del problema hasta que es iniciado de nuevo. Con lo anterior en cuenta, la inclusión o no de un servicio en el problema debe tener en cuenta el resultado, para ese servicio, de la instancia anterior del problema según una máquina de estados. Se pueden distinguir cuatro situaciones diferentes que se explican a continuación:

- **Libre.** El usuario i no tiene un servicio k activo y, por lo tanto, no requiere una conexión para ese servicio.
- **Activo.** El usuario i mantiene una conexión activa para un servicio k que ha sido aceptado por alguno de los accesos disponibles.
- **Rechazado.** El usuario i inició un flujo asociado a un servicio k pero éste no fue aceptado por la red.
- **Interrumpido.** El usuario i mantenía un servicio activo k que fue inicialmente aceptado pero se ha interrumpido antes de finalizar debido a la movilidad del usuario u otros eventos.

Con los estados mencionados y considerando que la optimización, para una ejecución, consiste en la resolución de sucesivas “fotografías” del estado de la red, se puede representar la evolución de un par usuario-servicio concreto de acuerdo a la Figura 5.1. Las flechas representan la transición entre estados, que dependen de dos aspectos: la intención del usuario para ese servicio y el resultado de la instancia anterior del problema. Estos

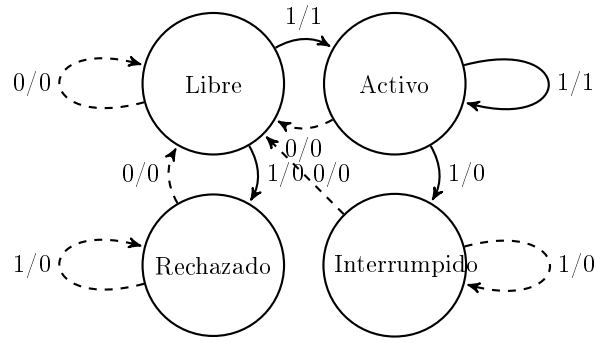


Fig. 5.1: Máquina de estados para un par usuario-servicio

valores se representan con la pareja de números sobre las flechas (*intención/resultado*). Resulta sencillo ver que, dependiendo del estado previo y de la intención actual, existen situaciones para las cuales la variable correspondiente no entrará a formar parte del problema; estas situaciones se representan con líneas discontinuas. Independientemente del estado en el que se encuentre, cuando un usuario i no tiene intención de tener un servicio k activo, la correspondiente variable básica no se incluye en el problema de optimización ($x_{ijk} = 0 \quad \forall j$) y, por lo tanto, pasa al estado *Libre*. Por otro lado, en el momento en que un servicio sea rechazado o interrumpido la variable asociada al par usuario-servicio se descarta para la instancia del problema sin tener en cuenta la intención del usuario. Se asume que las conexiones que se han perdido no se pueden recuperar, por lo que permanecen en el mismo estado (*Rechazado* o *Interrumpido*) hasta que vuelven al estado *Libre*.

5.1.2. Implementación

La solución de una instancia del problema se refiere a la conectividad óptima de una situación particular, que viene dada por la posición de los usuarios finales, su intención de mantener servicios activos, el resultado de la instancia anterior y el estado de la red (capacidad restante de los elementos de acceso). Estas situaciones se pueden entender como “fotografías” del escenario en el que los usuarios se mueven y generan flujos asociados a servicios durante un periodo de tiempo determinado. Estos patrones (tráfico y movimiento) se suministran por medio de trazas que se procesan por la herramienta que se ha desarrollado la cual posee dos módulos bien definidos. El primero resuelve el problema de optimización obtenido a partir las trazas y usando la librería GLPK [24]; se genera un proceso para cada instancia del problema y, por lo tanto, no existe una relación entre el problema actual y la solución previa. Como se ha mencionado, para plantear el problema de manera apropiada resulta imprescindible dotar a la herramienta de la capacidad de tener en cuenta los resultado previos del problema (servicios rechazados o interrumpidos, carga de los elementos de acceso...); a fin de proveer de esta capacidad se ha añadido a la herramienta otro módulo, el monitor, que no se consideró en el diseño anterior [20]. Este

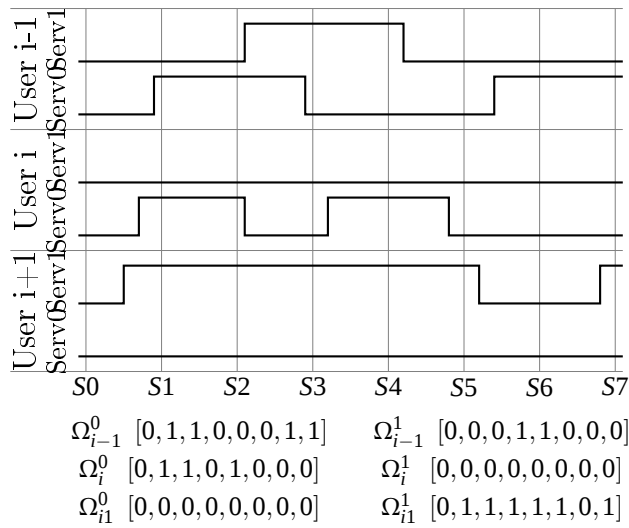


Fig. 5.2: Ejemplo de fotografías de servicios

segundo módulo analiza el resultado del problema y lo registra para que sea usado por la siguiente instancia; además, esta a cargo de mantener y actualizar las estadísticas del escenario.

Como se ha mencionado, a la herramienta se le suministran trazas consecutivas que deben reflejar: la posición de los usuarios finales de acuerdo a un patrón de movilidad y la intención de todos los usuarios de tener activos cada uno de sus servicios. La Figura 5.2 muestra un ejemplo ilustrativo de cómo los servicios correspondientes a tres usuarios se mapean en fotografías consecutivas. Cabe mencionar que el tiempo que un servicio se mantiene tanto en estado *Libre* como *Activo* debe ser siempre mayor que

Procedimiento 1 Optimización por fotografía

Configuración global

Despliegue del escenario

 parámetros de configuración: desde fichero

 usuarios finales: desde fichero

 elementos de acceso: desde fichero

Procesado inicial

Variables de optimización

 Establecimiento de conectividad

 Intención de servicio y estado previo

Funciones de utilidad

 Elemento de acceso previo

 Carga actual

 Precio actual

Optimización

Resolución del problema: GLPK

 Resultado del problema: actualizar estado

 Actualizar estadísticas

el tiempo que dura una fotografía con el fin de indicar de forma correcta la presencia de nuevos flujos (como es el caso del servicio 1 para el usuario i en la Figura 5.2). Para cada par usuario/servicio (i/j) se define el vector Ω_i^j [$\omega_0, \omega_1, \dots, \omega_N$], en el que el parámetro ω_k representa el estado del servicio en la $k^{ésima}$ fotografía. Como se puede observar, el formato de las trazas procesadas por la herramienta es muy genérico, por lo que puede ser fácilmente extendido para otros estudios con objeto de compararlos.

El procedimiento que se ha seguido para cada una de las fotografías del escenario se muestra en el Procedimiento 1 y se divide en tres fases. (1) Se establece el escenario de acuerdo a parámetros de configuración generales (posición y características de los elementos de acceso) y posición y servicios de los usuarios finales, todo ello se obtiene a partir de ficheros externos. (2) Una vez que el escenario ha sido desplegado, se verifica la conectividad física entre los usuarios y las diferentes estaciones base, así como el estado de los servicios. También se construye la función de utilidad considerando parámetros, dependiendo de la configuración concreta, como el elemento de acceso previo (si el par usuario/servicio mantenía un flujo activo), la carga actual de los elementos de acceso o el precio que se establecerá para un determinado servicio. (3) Finalmente se resuelve el problema por medio de la librería GLPK y el resultado se procesa permitiendo actualizar el estado de cada par usuario/servicio y de las estadísticas.

5.1.3. Funciones de utilidad

Tras presentar la herramienta queda por explicar el diseño de los parámetros que conforman la función de utilidad y que modulan el beneficio de cada conexión. Aunque existe un gran abanico de posibilidades, en el ámbito de este trabajo nos centraremos en dos aspectos concretos: precio y Radio Access Technology (RAT) *affinity* (éste último se ha definido como la preferencia de asociar un servicio determinado a una determinada tecnología de acceso) y la combinación de ambos parámetros. Además, también se estudiará la interacción de estos parámetros con el coste de realizar trasposos, entendiendo que es preferible evitar cambiar de elementos de acceso cuando el servicio se encuentra activo. Se podrá observar cómo las diferentes combinaciones de estos parámetros dan lugar a diferentes rendimientos, poniendo de relieve la importancia de una adecuada configuración.

Valga aclarar que en este trabajo se asume que tanto los servicios como los elementos de acceso usan una capacidad genérica y discreta denotada por TU (*Traffic Unit*) que representará recursos de la tecnología utilizada (*slots* en TDMA, códigos en CDMA, sub-portadoras en OFDMA, etc).

5.1.3.1. Conectividad

Como parámetro común para las diferentes estrategias, se dará una cierta utilidad a la conectividad *per se*, priorizando, de este modo, los servicios en curso sobre los nuevos al asumirse que siempre es preferible (desde

el punto de vista de la calidad de la experiencia) rechazar un nuevo servicio a interrumpir uno ya iniciado. Este parámetro (α_{ijk}) se define como:

$$\forall j \quad \alpha_{ijk} = \begin{cases} 1 \cdot \frac{c_k}{\max_{\forall k} c_k} & \text{si el par usuario/servicio } i/k \\ & \text{estaba conectado} \\ \lambda \cdot \frac{c_k}{\max_{\forall k} c_k} & \text{de otro modo} \end{cases} \quad (5.3)$$

en donde c_k representa la capacidad requerida por el servicio k y λ (parámetro de diseño) se selecciona a fin de asegurar que a un servicio en curso se le da siempre mayor prioridad que a nuevas llamadas sin importar la capacidad requerida por estas: $\lambda < \frac{\min_{\forall k} c_k}{\max_{\forall k} c_k}$.

Como se puede observar, esta función considera la capacidad requerida por cada servicio; de otro modo, a aquellos que necesitaran mayor capacidad se les daría (de manera indirecta) una prioridad más alta².

5.1.3.2. Traspaso

También se considera la influencia que el realizar un traspaso tendría sobre el rendimiento. En este sentido, se modela la preferencia por mantener el acceso actual con el objetivo de incluir el coste del cambio. A este fin se define el parámetro β_{ijk} como:

$$\beta_{ijk} = \begin{cases} 1 & \text{si el par usuario/servicio } i/k \text{ estaba} \\ & \text{conectado a la BS } j \\ 1 - \mu & \text{de otro modo} \end{cases} \quad (5.4)$$

en donde μ es un parámetro de diseño que se selecciona dependiendo de la configuración, como se explicará más adelante.

5.1.3.3. Parámetro de utilidad de precio

En este caso el objetivo es representar la inclinación de los usuarios finales por las alternativas más económicas. Para ello la utilidad de este parámetro debe crecer a medida que el elemento de acceso ofrece precios más bajos, por lo que se busca una función decreciente. A fin de permitir la comparación relativa entre los precios ofrecidos se propone el uso de una función logarítmica (see Fig. 5.3(a)) que se define como:

$$\forall k \quad \gamma_{ijk} = \begin{cases} -\log(p_j) & p_j \in [p_i^{\min}, p_i^{\max}] \\ 1 & \text{de otro modo} \end{cases} \quad (5.5)$$

donde $p_i^{\max}$ es el precio máximo que un usuario i está dispuesto a pagar y $p_i^{\min}$ se corresponde con el menor precio por debajo del cual no se incrementaría la utilidad. También se asume que aquellas estaciones base que ofrezcan un precio por encima del máximo admisible serían descartadas por el usuario. Se han utilizado unidades de precio relativas al máximo

²Una conexión de un servicio que requiriera 2 TUs tendría la misma utilidad que otra de 1 TU mientras que consumiría el doble de recursos. Por ello la optimización siempre se decantaría por dos conexiones de 1 TU en detrimento de una de 2 TUs.

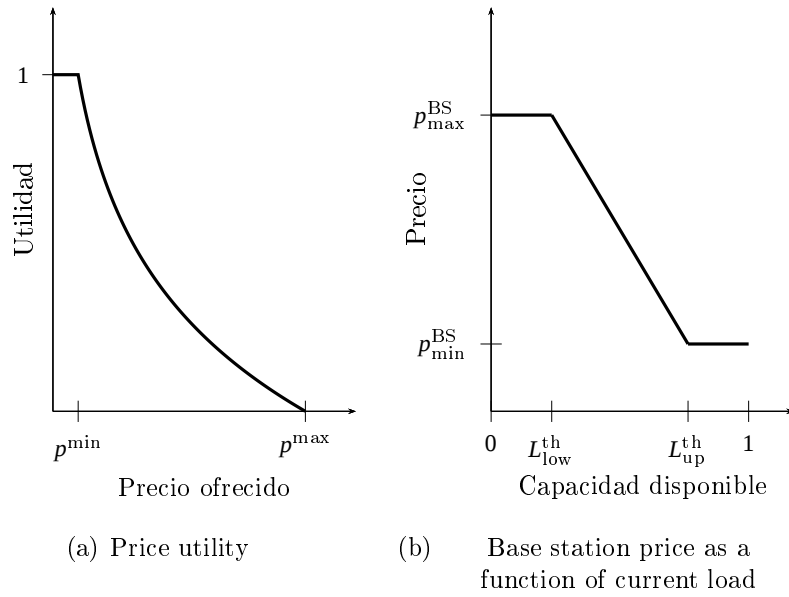


Fig. 5.3: Funciones de utilidad relativas al precio

precio ofrecido por los elementos de acceso que también se corresponderá con p_i^{max} ; estas unidades también son relativas a 1 UT y fotografía.

Por otro lado, se ha considerado que los elementos de acceso usan el precio ofrecido a los usuario como medio para incentivar o retraer a éstos de realizar una conexión; esto se refleja en la Figura 5.3(b), que representa la tarifa ofrecida por una estación base en función de su carga relativa (de la proporción de carga libre). Como se puede observar, cuando la estación base es encuentra altamente cargada (carga libre por debajo de L_{low}^{th}), el precio ofrecido alcanza el máximo permitido. Por otro lado, si la estación base se encuentra poco cargada el precio cae hasta el valor mínimo configurado. En este caso se ha optado por una función que decae linealmente entre los valores máximo y mínimo.

5.1.3.4. Parámetro RAT *Affinity*

Le propósito de este parámetro es el de favorecer que los servicios se asocien a las tecnologías más apropiadas a sus características. Por ejemplo, se podría considerar la tecnología WLAN como la más apropiada para servicios de datos sin restricciones de retardo al ofrecer un mayor ancho de banda. El parámetro correspondiente a esta utilidad (δ_{ijk}) se define como:

$$\forall i \quad \delta_{ijk} = \begin{cases} 1 & \text{si el servicio } k \text{ tiene afinidad por} \\ & \text{la tecnología de la BS } j \\ \nu & \text{de otro modo} \end{cases} \quad (5.6)$$

en donde ν es un parámetro de diseño que modula el peso que se le da a este aspecto es particular.

Tabla 5.1: Funciones de utilidad

	Price		RAT affinity		Combination	
	HO	No HO	HO	No HO	HO	No HO
Connectivity (<i>A</i>)	✓	✓	✓	✓	✓	✓
Handover (<i>B</i>)	✓		✓		✓	
Price (<i>C</i>)	✓	✓			✓	✓
RAT Affinity (<i>D</i>)			✓	✓	✓	✓

5.1.4. Combinaciones de parámetros

Mediante la combinación lineal de los parámetros descritos se puede definir la función de utilidad en forma general tal como se muestra a continuación.

$$u_{ijk} = A \cdot \alpha_{ijk} B \cdot \beta_{ijk} C \cdot \gamma_{ijk} D \cdot \delta_{ijk} \quad (5.7)$$

Existen dos maneras de favorecer cada uno de los parámetros: modificando el factor correspondiente (*A, B, C, D*) o usando una versión binaria de ellos (activado o no) y modulando su peso relativo mediante los parámetros de diseño (λ, μ, ν); en este trabajo se ha optado por la segunda alternativa. En concreto, se ha fijado $\lambda = \nu = 0.8$, mientras que el valor del parámetro μ se discute a continuación.

Si se desactiva el parámetro asociado al RAT *affinity* ($D = 0$), es posible comparar las utilidades de los accesos de dos alternativas para un servicio en curso (notar que el parámetro α_{ijk} es igual en ambos casos): el acceso que está siendo utilizado (1) y otro diferente (2) que ofrece un precio (p_2) menor que el del primero (p_1). En concreto se asume que p_2 es $100 \cdot \xi$ % más bajo que p_1 ($p_2 = (1 - \xi) p_1$, con $\xi < 1$). Por lo que se deduce:

$$\begin{aligned} u^{(1)} = u^{(2)} &\rightarrow \mu - \log(p_1) = -\log(p_1(1 - \xi)) \\ &\rightarrow \mu = -\log(1 - \xi) \end{aligned} \quad (5.8)$$

Si se establece el 20% como la variación del precio para la que un usuario estaría dispuesto a cambiar de estación base, el valor del parámetro μ es de ≈ 0.1 .

Teniendo en cuenta todo lo expuesto en esta sección, existen seis funciones de utilidad que se utilizan en los análisis que se han llevado a cabo y que se indican en la Tabla 5.1.

5.1.5. Resultados

Esta sección muestra algunos de los resultados que se han obtenido aplicando el método descrito sobre un escenario en particular con un doble objetivo: estudiar su viabilidad y discutir el impacto de una selección apropiada de la función de utilidad.

El escenario que se considera tiene un área cuadrada de $200 \times 200 \text{ m}^2$ en la que se han desplegado dos tipos de estaciones base diferentes (ver Figura 5.4). El primer tipo corresponde a una tecnología celular tradicional con un radio de cobertura de 150 m , que cubre el escenario por completo, y una capacidad de 16 TUs; por su parte, el segundo tipo emula puntos de acceso *WiFi* con una cobertura de 50 m y una capacidad de 8 TUs. Sobre este escenario se ha desplegado un número de usuarios que se ha incrementado de 20 a 200; éstos se mueven de acuerdo al modelo *Random Waypoint* cuyos parámetros se describen en la Tabla 5.2. Cada uno de los usuarios genera flujos de dos tipos de servicios, de acuerdo al modelo *ON-OFF*, y éstos mantienen una afinidad hacia una de las tecnologías. Todas las estaciones base hacen uso de la política de tarificación que se ha descrito anteriormente (Figura 5.3(b)), de modo que aumentan el precio a medida que se encuentran más cargadas. Se han realizado 10 ejecuciones independientes del escenario (cada una de ellas conlleva 360 problemas de optimización) y se han obtenido los valores medios de los resultados. Se han considerado las seis funciones de utilidad descritas en la Tabla 5.1 y, para ambos tipos de servicio (0 y 1), se han analizado las siguientes figuras de mérito:

- **Tasa de éxito (Success Rate - SR):** probabilidad de que un servicio finalice con éxito, es decir, que no sea rechazado ni interrumpido.
- **Trasposos (Handovers - HO):** número medio de trasposos que se llevan a cabo durante la duración de un servicio.
- **Precio por servicio (Price per Service - PS):** precio medio pagado por tiempo y unidad de tráfico.

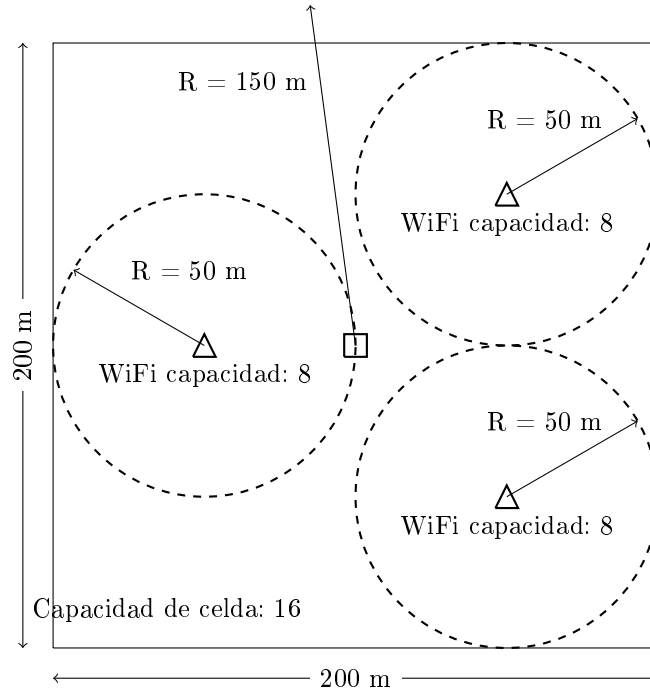


Fig. 5.4: Despliegue de red

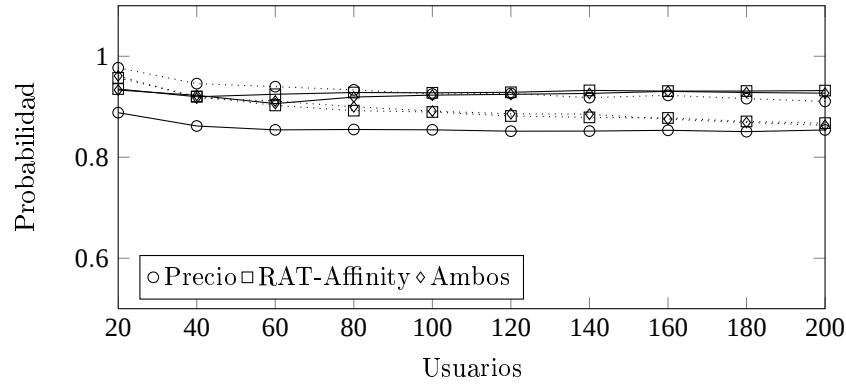
Tabla 5.2: Parámetros del análisis

Estaciones base	
<i>Celular</i>	
Cobertura (m)	150
Capacidad (TU)	16
Política de tarificación	$P \in [0.1, 1.0]$, Umbrales $\{0.2, 0.8\}$
<i>WiFi</i>	
Cobertura (m)	50
Capacidad (TU)	8
Política de tarificación	$P \in [0.1, 1.0]$, Umbrales $\{0.2, 0.8\}$
Modelo de movilidad	
<i>Random Waypoint</i>	
Velocidad (m/s)	$U(1, 3)$
Periodo de movimiento (s)	$U(800, 1000)$
Periodo de reposo (s)	$U(80, 100)$
Modelo de servicio	
<i>Servicio 0: Voz</i>	
Modelo	On-Off
Tiempo entre llegadas (s)	120
Tiempo del servicio (s)	180
Capacidad (TU)	1
RAT <i>Affinity</i>	Cellular
<i>Servicio 1: Transferencia de datos</i>	
Modelo	On-Off
Tiempo entre llegadas (s)	60
Tiempo del servicio (s)	120
Capacidad (TU)	2
RAT <i>Affinity</i>	WiFi
Parámetros generales	
Tiempo de simulción (s)	3600
# de fotografías	360
# of ejecuciones	10

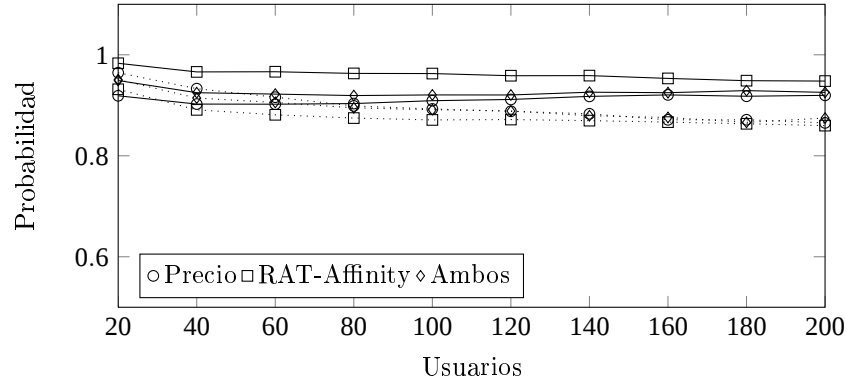
- **RAT *affinity* (RA):** este parámetro indica la proporción de tiempo que un servicio ha usado la tecnología hacia la que tiene afinidad.

En primer lugar, la Figura 5.5 muestra la probabilidad de finalizar un servicio con éxito. Como se puede observar, la definición de la función de utilidad (especialmente del parámetro α) hace que se obtengan resultados similares para ambos servicios; de no haberse considerado la capacidad en la definición del citado parámetro la tasa de éxito del servicio 1 hubiera sido mucho menor. Además, estos resultados muestran que no existe un impacto notable en ninguna de las funciones de utilidad por el uso del parámetro asociado a los traspasos.

Sin embargo, el impacto del parámetro asociado a los traspasos queda



(a) Sin parámetro de traspasos

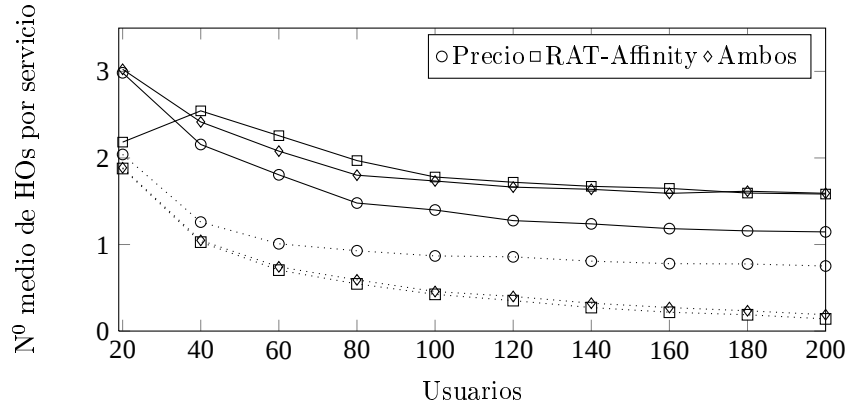


(b) Con parámetro de traspasos

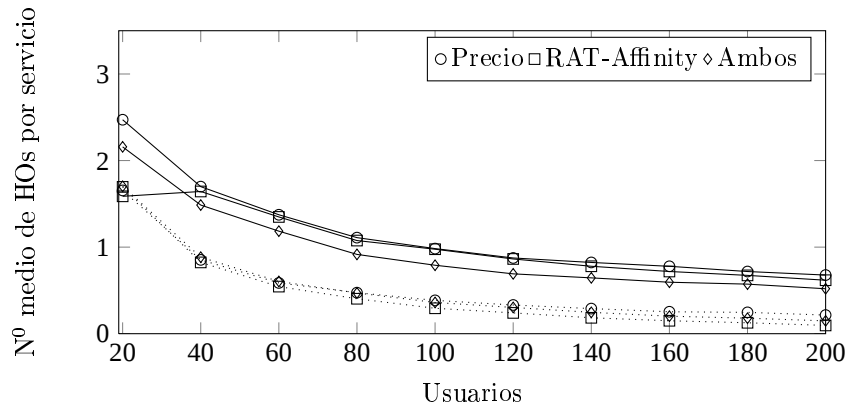
Fig. 5.5: Tasa media de finalización de servicio con éxito Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1

patente en la Figura 5.6. Se puede observar una notable reducción del número de traspasos (particularmente para el servicio 0) cuando este parámetro (β) se considera en la función de utilidad. El impacto es menos relevante para el servicio 1, especialmente con un número elevado de usuarios, ya que la red se encuentra fuertemente cargada (esto es más importante para la estación base celular) y no existen muchas alternativas de conexión. Por otro lado, cuando no se considera este parámetro ($B = 0$) también se observa que la función que considera el RAT *affinity* reduce el número de traspasos para el servicio 1, no siendo así para el servicio 0. Esto es parcialmente debido al efecto *ping-pong* que puede afectar a los flujos del servicio 0, ya que un doble cambio de acceso no tendrá ningún efecto en la utilidad global.

En lo que se refiere al precio, la Figura 5.7 arroja un resultado interesante. Para el servicio 0 se observa el comportamiento esperado, la función de utilidad que tiene en cuenta el precio abarata el servicio comparado con la que tiene en cuenta el RAT *affinity*. Sin embargo, para el servicio 1 se observa que es precisamente la que tiene en cuenta el RAT *affinity* la que conlleva los precios más bajos, incluso por debajo de los ofrecidos por la basada en precio (recordar que la optimización es global, por lo que abarata



(a) Sin parámetro de traspasos

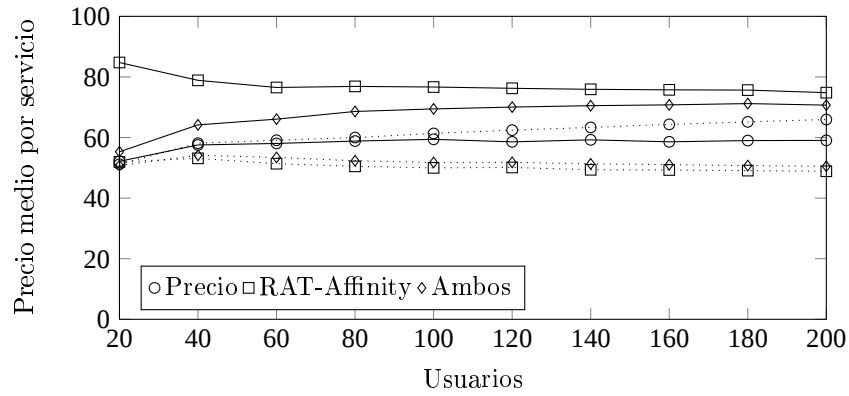


(b) Con parámetro de traspasos

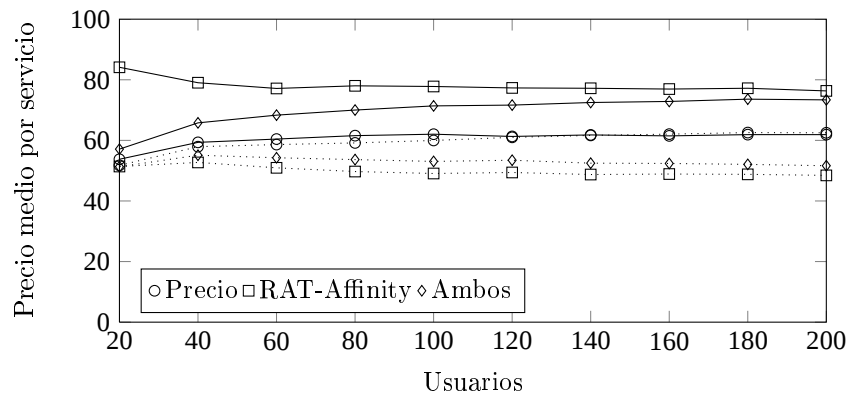
Fig. 5.6: Número medio de traspasos por servicio Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1

el conjunto de los servicios en el escenario y no cada uno de ellos). La razón es que los punto de acceso *WiFi* (debido a su cobertura) se encuentran menos cargados que la estación celular y, por lo tanto, el precio que ofrecen es menor. Como los servicios de tipo 0 se mantienen en las estación base celular (sin importar la tarifa que ofrece) de acuerdo a su RAT *affinity*, los precios ofrecidos por los puntos de acceso *WiFi* se mantienen bajos, dando lugar a precio por servicio de tipo 1 bastante bajos. Por otro lado, los resultados también reflejan que la función basada en precio no distingue por tipo de servicio, por lo que ambos tienden a pagar lo mismo. Por último, cabe destacar que no se observa una clara dependencia del parámetro de traspasos en lo relativo al precio.

La Figura 5.8 muestra la proporción de tiempo que un servicio se encuentra asociado a una tecnología por la que tiene afinidad. Se puede afirmar que el uso de la función de utilidad apropiada tiene un gran peso en este aspecto, ya que los resultados obtenidos para la utilidad basada en RAT *affinity* son mucho mejores, en este sentido, que aquellos que presenta la estrategia basada en precio. También se observa una sutil influencia del



(a) Sin parámetro de traspasos



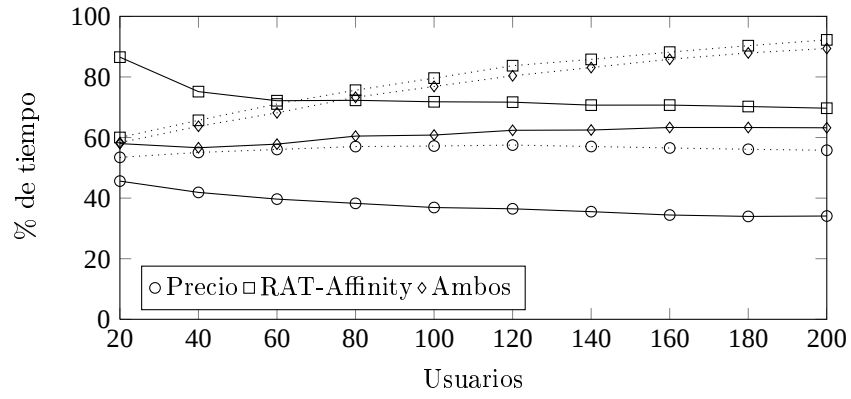
(b) Con parámetro de traspasos

Fig. 5.7: Precio medio por servicio y TU Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1

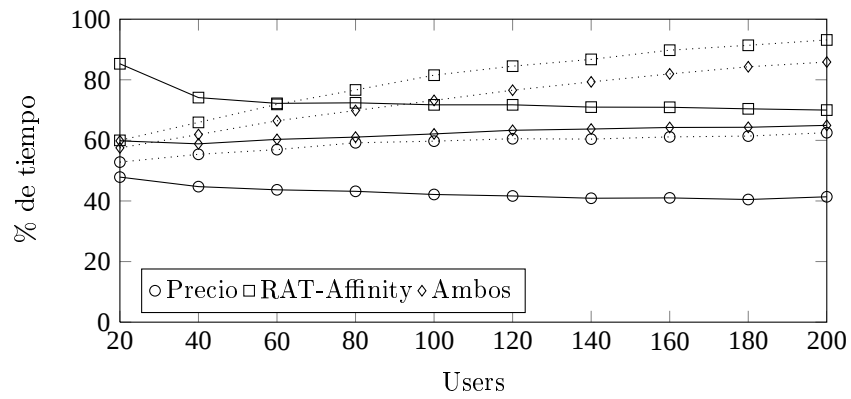
parámetro de traspasos. Por último se puede ver que el servicio 1, para todas las estrategias estudiadas, presenta un mejor comportamiento en lo que al RAT *affinity* se refiere.

A fin de obtener una visión más global del comportamiento de las diferentes estrategias de acuerdo a las funciones de utilidad definidas, la Figura 5.9 hace uso de diagramas *web* (tela de araña) en los que se representa las cuatro figuras de mérito para ambos tipos de servicio. Los bordes de los ejes representan el mejor resultado que se puede obtener, mientras que el centro establece el peor caso (los límites del número de traspasos se han establecido empíricamente por observación). Los resultados se han obtenido para el escenario con 200 usuarios. Como se indicó anteriormente, se puede ver una mejora en los casos en que se hace uso del parámetro relacionado con los traspasos ya que éste no afecta de forma relevante al resto y mejora de manera notable el número medio de traspasos por servicio. En el caso de la estrategia ligada al RAT *affinity* se puede ver que el uso del parámetro de los traspasos incluso proporciona una leve mejoría de la probabilidad de completar un servicio con éxito para el servicio 0.

En general, todos los resultados indican que la estrategia que tiene en cuenta tanto el precio como el RAT *affinity* tiene un rendimiento similar



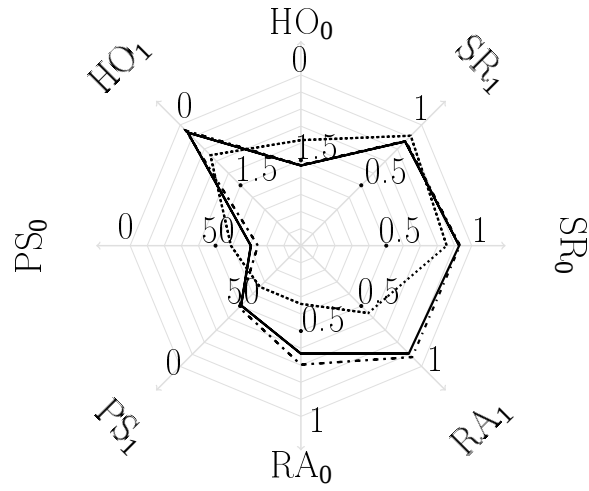
(a) Sin parámetro de traspasos



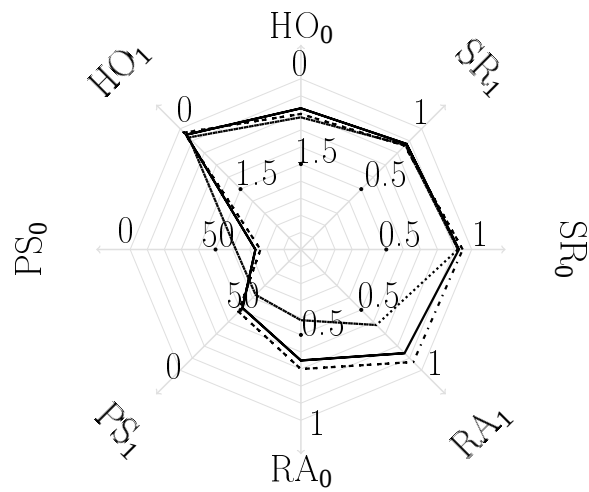
(b) Sin parámetro de traspasos

Fig. 5.8: RAT *affinity* medio por servicio Vs. número de usuarios. Las líneas continuas corresponden al servicio 0 y las discontinuas al servicio 1

al obtenido mediante la que únicamente tiene en cuenta el RAT *affinity* sin producir un perjuicio notable en el precio; por su parte la función basada sólo en precio tiene un rendimiento pobre en lo que a RAT *affinity* se refiere.



(a) Sin parámetro de trasposos ($B=0$)



(b) Con parámetro de trasposos ($B=1$)

Fig. 5.9: Rendimiento de las diferentes funciones de utilidad. Línea punteada para función ligada al precio; línea discontinua para función ligada al RAT *affinity*; línea continua representa la combinación de las anteriores.

5.2. Comparación de algoritmos prácticos

Tras haber estudiado el comportamiento óptimo del conjunto de la red que se puede esperar de acuerdo a los parámetros que se escojan, a lo largo de este capítulo se va a contrastar el rendimiento que pueden tener algoritmos realistas con respecto al óptimo global. De acuerdo a la definición de la función de utilidad de la sección anterior, ésta acepta diferentes parámetros en función del comportamiento que se lograr.

Desde el punto de vista del operador, el objetivo es establecer la estrategia más adecuada que les permita obtener grandes beneficios y una utilización óptima de sus recursos. Para esto, se hace necesario entender cómo la QoS ofrecida afecta a la demanda por parte de los usuarios. Por otro lado, desde la perspectiva de usuario, parece claro que los usuarios siempre se encontrarán ligeramente inclinados hacia el acceso más barato pero siempre buscando un equilibrio entre el precio y la QoS. Es por esto que se debe buscar una estrategia de operador que tenga en cuenta las preferencias propias y las impuestas por los usuarios. Para ello, se ha usado la estrategia de tarificación dinámica de la sección anterior que considera la carga instantánea de tráfico en las estaciones base. De este modo, el esquema proporciona balanceo de carga entre los diferentes nodos de acceso aliviando la carga en aquellos más saturados; por otro lado también se realiza una diferenciación entre tipos de usuarios diferenciándolos en el precio que estén dispuestos a pagar entre usuario *business* y *consumer*.

El estudio del rendimiento de los mecanismos de selección se ha llevado a cabo mediante una herramienta de simulación llamada *multi-Constraint Access Selection in heterogeneous Environments* (**mCASE**) cuya arquitectura se detalla en [25]. En pocas palabras, mCASE se trata de un simulador dirigido por eventos que permite la creación rápida de escenarios de red en los que se pueden definir diferentes participantes como elementos de acceso, usuarios finales, servicios, etc. La principal ventaja de esta herramienta es su escalabilidad, lo que la lleva a requerir tiempos de simulación reducidos en comparación con otras opciones; esto permite llevar a cabo análisis de redes complejas mediante numerosas simulaciones independientes en un tiempo razonablemente corto. Por otro lado, la agilidad que presenta mCASE tiene como contrapartida los requerimientos en cuanto a abstracción de los diferentes componentes de la red. Cabe destacar el modo en que mCASE modela los recursos de red, tanto los que gestionan los elementos de acceso como los que requieren los servicios; se hace a través de la unidad abstracta (*Traffic Unit*) de manera similar a la optimización.

En la literatura se pueden encontrar varios trabajos que han analizado la correspondencia entre la QoS percibida y la tarifa correspondiente al servicio en busca de un equilibrio entre ambos parámetros. Los autores de [26] han propuesto una estrategia de precios que lleva a que el operador aumente sus beneficios mediante el ajuste de niveles de QoS que se ofrecen a los usuarios. En este caso los autores no usan ninguna selección de acceso sino que únicamente se centran en la repercusión que tiene el precio sobre la demanda. En [27] se propone una función de aceptación que cuantifica el

nivel de satisfacción de los usuarios en función del precio y la QoS. En este último caso, ambos parámetros dependen del número de recursos asignados y los autores establecen una estrecha relación con los parámetros de la tecnología subyacente, mientras que el precio se modela con una función lineal. Estos autores definen dos tecnologías de acceso: la primera trata de asegurar los recursos asignados durante toda la conexión, mientras que la segunda no garantiza que no se produzcan variaciones en la QoS durante el uso del servicio.

Como se indicó al comienzo de la sección anterior, la mayoría de los trabajos que se encuentran en la literatura comparan el rendimiento de las estrategias propuestas con las estrategias convencionales o con otras nuevas; sin embargo no se analiza lo cerca que se encuentra la estrategia de un comportamiento óptimo. En este sentido, tras haber presentado el marco de optimización en la Sección 5.1, ahora el objetivo es cuantificar lo buenas que son varias estrategias de acceso mediante su comparación con un rendimiento óptimo.

Por otro lado, también merece la pena destacar otros trabajos que tratan de hallar un rendimiento óptimo en función de una serie de figuras de mérito. Los autores de [28] proponen mecanismos de selección en el que usan una función de utilidad que prioriza las conexiones más adecuadas para cierta aplicación y que produzcan un menor consumo energético. Estos autores emplean una técnica basada en programación lineal entera que maximiza el número de conexiones (siguiendo la función de utilidad) a las redes preferidas; de este modo se asegura una óptima satisfacción de los usuarios (asumiendo como tal la adecuación de la red a la aplicación/servicio) y un mejor rendimiento de la red en términos energéticos. En este caso no se considera el precio como un parámetro a tener en cuenta en la optimización y, además, tiene la limitación de que el proceso sólo tiene en cuenta los eventos de traspaso forzados, por lo que no se prevé que puedan darse situaciones en las que un traspaso fuera preferible aún cuando no fuera estrictamente necesario. Otros trabajos similares, por ejemplo [29] y [30], también usan técnicas de optimización, pero ninguno de ellos tiene en cuenta el precio en sus esquemas de decisión, sino que se centran en el balanceo de carga por medio de gestión de recursos radio. Por último, los autores de [31] usan como parámetro la QoS a nivel de aplicación y el precio del acceso a la red en la selección, sin embargo su análisis hace uso de una optimización estocástica (modelando la red como un sistema *restless bandit*) en lugar de una formulación de programación lineal.

5.2.1. Función de utilidad

En esta sección se van a presentar la función de utilidad definida en la herramienta mCASE para la selección de acceso de los usuarios. Aunque su definición es muy similar a la dada en la herramienta de optimización su significado es diferente. En este caso la decisión de los usuarios se toma de forma totalmente independiente por cada usuario haciendo uso de información local. Por ello, a pesar de que la definición (especialmente en la

tarificación) es parecida o similar al caso anterior, en este apartado se ha querido definir la utilidad usada por mCASE de forma separada.

Uno de los principales objetivos de este trabajo es el de analizar diferentes estrategias de tarificación que permitan a los operadores obtener un mayor beneficio reduciendo el número de llamadas rechazadas y estableciendo el precio más apropiado para cada servicio. A fin de conseguir esto se ha asumido que las estaciones base (o las funciones de *accounting* presentes en la red) son capaces de modular el precio ofrecido de acuerdo a la carga instantánea de modo que se induzca (bajando el precio) a los usuarios a establecer una conexión de acuerdo a la capacidad remanente. Además, se ha asumido que los usuarios finales pueden usar las diferentes estaciones base (asumiendo que existe conectividad física) y que no tienen ninguna preferencia hacia ninguna de ellas.

El precio se ha establecido como una función a trozos definida en la Figura 5.3(b). Como se puede observar las estaciones base usan dos umbrales: cuando la capacidad disponible se encuentra por debajo de C^{lower} establece el precio máximo ($p_{\max}$); por otro lado, cuando está por encima de C^{upper} se establece el mínimo precio aceptable $p_{\min}$ ($p_{\min} = \phi \cdot p_{\max}$, con $\phi < 1$); en otro caso se usa una función lineal decreciente. Esto implica que si la carga relativa de la estación base j es θ_j (esto es, su capacidad disponible es $(1 - \theta_j)$), se fijaría un precio como se indica en la Figura 5.3(b).

Por otro lado, a fin de hacer un planteamiento lo más general posible, se asume que la selección de acceso consiste en que los usuarios mantienen una lista ordenada de las estaciones base que detectan (aquellas con las que tiene conectividad física) de acuerdo a una función definida como la composición lineal de varios parámetros a los que se les asigna diferentes pesos de acuerdo a la configuración. Este planteamiento es bastante genérico y flexible, lo que permite implementar diferentes estrategias de acceso de acuerdo a los parámetros tenidos en cuenta como se puede ver en la Ecuación 5.9. En esta ecuación Φ_j es la función multi-criterio del elemento de acceso j , ω_k es el peso asignado al parámetro k y f_{kj} es el valor relativo a cada uno de los N parámetros para la estación base j . En [25] ya se pre-

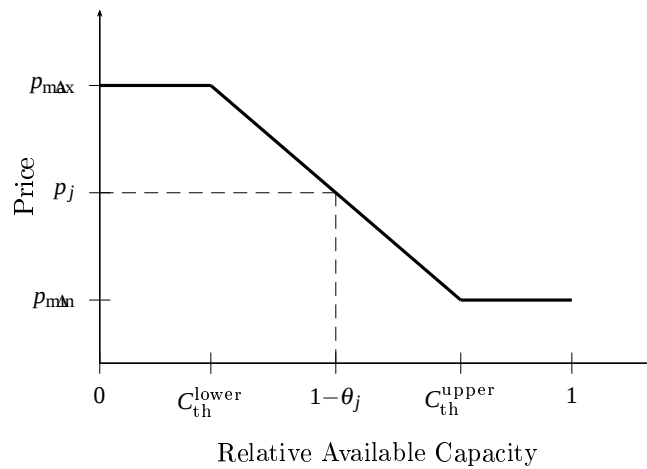


Fig. 5.10: Precio de la estación base en función de la capacidad disponible

sentó un planteamiento similar en el que se introdujo un elevado número de figuras de mérito a tener en cuenta en la utilidad. Todos estos parámetros fueron modelados con valores normalizados entre 0 y 1 y los pesos se definieron de manera que $\sum_{k=1}^N \omega_k = 1$; de esta manera el valor final de la función de utilidad siempre se encuentra entre 0 y 1. A diferencia de lo presentado en [25], donde se estudiaron en torno a 10 criterios diferentes, en este análisis únicamente se ha tenido en cuenta algunos de ellos que se describen a continuación:

$$\Phi_j = \sum_{k=1}^N \omega_k f_{kj} \quad (5.9)$$

- Precio: el objetivo de este criterio es en de analizar el uso del precio como medio para controlar la disponibilidad de los usuarios a establecer una conexión. Desde la perspectiva del operador este parámetro evitaría saturar las estaciones base por medio de la función de precios de la Ecuación 5.10. Una forma sencilla para modelar este parámetro sería por medio de una función triangular; sin embargo, esto tiene el inconveniente de que iguales variaciones en el precio llevan a iguales variaciones en la utilidad de este parámetro sin tener en cuenta si esa variación ha sido grande o pequeña en términos relativos. Para llevar a cabo una comparativa más justa entre los diferentes elementos de acceso se ha usado una función logarítmica A_j (si el precio excede el máximo admisible por el usuario el elemento de acceso será descartado):

$$A_j = \begin{cases} 1 & p_j < p_{\min}^{\text{user}} \\ -\log(p_j) & p_{\min}^{\text{user}} \leq p_j \leq p_{\max}^{\text{user}} \end{cases} \quad (5.10)$$

- Traspasos: desde la perspectiva del usuario podría ser atractivo elegir siempre el mejor elemento de acceso en términos de precio (el más barato). Sin embargo, es conocido el perjuicio que los traspasos provocan tanto en los usuarios (deterioro de la QoS, gasto energético...) como en la gestión de la red (incremento de la señalización, duplicidad de reserva de recursos...). De este modo, dado que los usuarios son conscientes de los elementos de acceso con los que tienen conectividad, se ha definido el parámetro B_j como:

$$B_j = \begin{cases} \sigma & \text{si el usuario final estaba conectado a } BS_j \\ 0 & \text{en otro caso} \end{cases} \quad (5.11)$$

Llegado este punto, se necesita estudiar el valor de σ de modo que ninguno de los parámetros explicados tenga un mayor impacto que el otro. Para conseguir esto, se ha igualado el peso de ambos parámetros a partir de la Ecuación 5.9 resultando como se muestra a continuación:

$$\Phi_j = \omega \cdot A_j \omega \cdot B_j \quad (5.12)$$

Si se considera que un usuario final está actualmente conectado a una estación base BS_a pagando un precio P_a , se ha de hallar el precio que debería

ofrecer otra estación base BS_b de modo que el usuario optase por la segunda e iniciase el traspaso. Asumiendo que el precio ofrecido por la segunda estación base es de $100 \cdot \chi\%$ más bajo que el ofrecido por BS_a , entonces $P_b = (1 - \chi \cdot P_a)$. Bajo esta premisa se puede evaluar la expresión 5.12 para la estación base BS_a resultando:

$$\Phi_a = \omega \cdot (-\log(P_a)) \quad \omega \cdot \sigma \quad (5.13)$$

De la misma manera, para que se requiera el traspaso la utilidad en BS_b será:

$$\Phi_b = \omega \cdot (-\log(P_b)) \quad (5.14)$$

Igualando las dos ecuaciones se obtiene el valor de σ que provoca el traspaso. Este valor se da cuando $\sigma = -\log(1 - \chi)$. De la misma forma que en el caso de la optimización, si se establece que el usuario decidirá cambiar de elemento de acceso cuando existan alternativas que ofrezcan precios un 20% más barato ($\chi = 0.20$), el valor de σ será aproximadamente de 0.1.

Cada usuario ha utilizado este procedimiento para ordenar las estaciones base. Además, se han definido dos tipos de usuarios (*business* y *consumer*) de acuerdo al límite que están dispuestos a pagar por el servicio.

Volviendo la vista a los capítulos 3 y 4, parece claro que se podría realizar un mapeo directo entre la función de utilidad aquí presentada y su correspondiente algoritmo de decisión y la arquitectura OConS. Así, los diferentes parámetros de la función de utilidad estarían representado por entidades de información; el ordenamiento de las alternativas disponibles y la elección de la mejor formaría parte de la implementación de la DE; y la conexión en sí correría a cargo de una EE.

5.2.2. Optimización

Como se ha comentado previamente, el principal objetivo de este análisis es el de comparar diferentes estrategias de acceso con el óptimo obtenido mediante la técnica presentada en la Sección 5.1. De esta manera, y a fin de llevar a cabo una comparación lo más justa posible, el procedimiento que se ha realizado es el de tomar “fotografías” periódicas del escenario que mCASE está ejecutando y usar estas fotografías, apropiadamente formateadas, como las trazas de entrada de la herramienta de optimización.

A pesar de todo, resulta muy complicado, sino imposible, modelar el problema de optimización para que se comporte exactamente igual que las estrategias de selección; en primer lugar la optimización se realiza sobre el conjunto de la red mientras que la selección se lleva a cabo de manera local con información limitada. Por esto se han estudiado dos formulaciones diferentes del problema de optimización:

- Perspectiva del operador: en este caso se asume que se optimiza de acuerdo a lo que preferiría el operador. Así, es razonable pensar que

el principal objetivo es el de admitir tantas conexiones como sea posible priorizando los usuarios *business* sobre los *consumer*. Además, como gestor de la red, también se intentará minimizar el número de traspasos lo que, a su vez, evita sobrecarga de señalización.

- Perspectiva del usuario: si la optimización se hiciera desde el punto de vista de los usuarios se podría asumir que el principal objetivo es pagar el precio más barato posible. Por otro lado, también los usuarios se inclinarían a evitar, en la medida de lo posible, los traspasos.

En los dos casos descritos, el problema de optimización tiene $U \times N$ variables a optimizar donde U y N son el número de usuarios finales y estaciones base respectivamente. Estas variables x_{ij} se definen como sigue:

$$x_{ij} = \begin{cases} 1 & \text{si el usuario } i \text{ usa el acceso } j \\ 0 & \text{en otro caso} \end{cases} \quad (5.15)$$

Tomando como caso genérico la formulación del problema de la sección anterior y eliminando la dimensión de servicio el problema queda como se muestra en 5.16, en este caso se ha asumido que todos los servicios consumen 1 TU.

$$\text{Max.} \quad \sum_{i,j} u_{ij} \cdot x_{ij} \quad (5.16)$$

$$\text{s.t.} \quad \forall i, j \quad x_{ij} \in \{0, 1\} \quad (5.17a)$$

$$\forall i \quad \sum_j x_{ij} \leq 1 \quad (5.17b)$$

$$\forall j \quad \sum_i x_{ij} \leq C_j \quad (5.17c)$$

Respecto a la función u_{ij} , ésta ha de ser definida de acuerdo a la optimización que se va a realizar. En este caso se van a usar los siguientes parámetros:

- Conectividad (α_i): se usa para priorizar aquellas llamadas que se encuentran activas con respecto a las nuevas; como se mencionó en el análisis de la optimización, se asume que es preferible rechazar llamadas entrantes a interrumpir las que están siendo cursadas.

$$\alpha_i = \begin{cases} 1 & \text{si el usuario } i \text{ posee servicio en curso} \\ 1 - \varepsilon & \text{en otro caso} \end{cases} \quad (5.18)$$

- Tipo de usuario (β_i): este parámetro se usa para priorizar a los usuarios *business* respecto a los *consumer*.

$$\beta_i = \begin{cases} 1 & \text{si el usuario } i \text{ es business} \\ 1 - \zeta & \text{en otro caso} \end{cases} \quad (5.19)$$

- Precio (δ_j): se ha usado este parámetro para modelar la disposición de los usuarios a elegir la estación base que presente el menor precio; aclarar que si una estación base ofrece un precio por encima del máximo admisible para el usuario, esta estación no entra a formar parte del problema para ese usuario ($u_{ij} = 0$). Este parámetro se define como:

$$\delta_j = \begin{cases} 1 & p_j \leq p_{\min}^{\text{user}} \\ -\log(p_j) & p_j \in [p_{\min}^{\text{user}}, p_{\max}^{\text{user}}] \\ 0 & \text{otherwise} \end{cases} \quad (5.20)$$

- Traspasos (γ_{ij}): este parámetro modela el coste de realizar un traspaso.

$$\gamma_{ij} = \begin{cases} 1 & \text{si el usuario } i \text{ Estaba conectado a BS } j \\ 1 - \eta & \text{en otro caso} \end{cases} \quad (5.21)$$

Teniendo en cuenta todo lo expuesto, en la Tabla 5.3 se han resumido las diferentes configuraciones de la función de utilidad que se han usado para el problema de optimización. Como se ha mencionado se han estudiado dos planteamientos diferentes en función de si se busca la perspectiva del operador (α_i, β_i and γ_{ij}) o del usuario (α_i, δ_i and γ_{ij}).

5.2.3. Escenario de red

El análisis se ha realizado sobre un escenario con 4 estaciones base de una única tecnología en el que cada elemento de acceso posee una capacidad de 16 TUs y un radio de cobertura de 600m. Esta red se a desplegado sobre una superficie cuadrada de 100m sin una planificación de red concreta pero respetando una distancia mínima entre ellas. El despliegue en concreto se puede ver en la Figura 5.11.

Los usuarios finales se han desplegado de manera aleatoria sobre la superficie cuadrada y se les ha dotado de un patrón de movilidad que sigue el modelo *Random Waypoint* [32] con una velocidad peatonal en el intervalo [1,3] m/s. En cuanto a los patrones de tráfico, los usuarios inician servicios de acuerdo al model *ON-OFF* en el que el tiempo entre llegadas y el tiempo de servicio se han modelado como variables aleatorias exponenciales negativas con medias de 120 y 60 segundos respectivamente.

Finalmente, indicar que se hecho uso de una unidad monetaria genérica (MU) como la cantidad de dinero que un usuario paga por TU y por segundo. Además para el usuario de tipo *business* se ha fijado el precio

Tabla 5.3: Funciones de utilidad para optimización

	Perspectiva de operador	Perspectiva de usuario
	$\epsilon = 0.4, \zeta = 0.6, \eta = 0.2$	
Considerando <i>HO</i>	$u_{ij} = \alpha_i + \beta_i + \gamma_{ij}$	$u_{ij} = \alpha_i + \gamma_{ij} + \delta_j$
No considerando <i>HO</i>	$u_{ij} = \alpha_i + \beta_i$	$u_{ij} = \alpha_i + \delta_j$

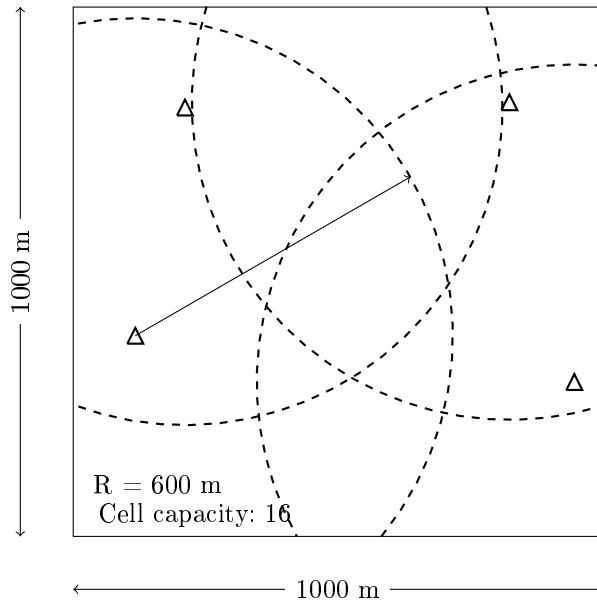


Fig. 5.11: Despliegue de red

máximo que está dispuesto a pagar en 1 MU por TU, mientras que para el usuario de tipo *consumer* el máximo varía entre simulaciones según el parámetro ψ que se ha definido como: $\psi = \frac{(p_{\max})_{\text{consumer}}}{(p_{\max})_{\text{business}}}$. Según la relación de precios máximos que los usuarios están dispuestos a pagar se han estudiado dos situaciones diferentes con valores de ψ igual a 0.6 y 0.8. Además, y únicamente para comparar, también se ha estudiado el caso en que las estaciones base ofrecen un precio aceptable para ambos tipos de usuarios ($0.5 \cdot (p_{\max})_{\text{business}}$).

5.2.4. Resultados

En cuanto al análisis llevado a cabo con mCASE, se han realizado 100 simulaciones independientes por cada caso de 3600 segundos de duración a fin de asegurar la validez estadística de los resultados. Para el problema de optimización, se han tomado “fotografías” del escenario cada 10 segundos de 10 simulaciones diferentes (cada simulación ha supuesto 360 problemas de optimización independientes) y los resultados mostrados son la media de las 3600 soluciones del problema.

Durante los diferentes experimentos se ha variado tanto el número de usuarios como el parámetro ψ , por otro lado la proporción de usuarios de tipo *business* se ha mantenido constante en un 30 %

Tabla 5.4: Parámetros de rendimiento

Traspasos (HO)	Mejor 3	Peor 0	Este parámetro indica el número medio de traspasos que se han realizado para cada uno de los servicios. Se analizará este parámetro tanto para los usuarios <i>business</i> como <i>consumer</i> (HO_{bus} y HO_{cons} respectivamente). El rango de valores se ha seleccionado de acuerdo a las observaciones realizadas durante el análisis en las que el valor máximo nunca ha superado 3 traspasos.
Tasa de éxito (SR)	0	1	Este parámetro mide la probabilidad de que un servicio se finalice con éxito; se define como la relación entre aquellos servicios completados respecto al global de los servicios. Por lo tanto, los servicios no finalizados con éxito son tanto aquellos que han sido rechazados como los interrumpidos. En este caso también se realiza la diferenciación entre tipos de usuarios dando como resultado las medidas SR_{bus} y SR_{cons} para los usuarios <i>business</i> y <i>consumer</i> respectivamente.
Precio por servicio (PS)	100	0	Se define como el precio medio que un usuario paga por servicio y unidad de tiempo y está limitado por la tarifa máxima que las estaciones base ofrecen. Nuevamente los resultados se presentan para ambos tipos de usuarios (PS_{bus} and PS_{cons}).
Factor de carga (LF)	0	10	Uno de los principales objetivos de la política de precios es el balanceo de carga. Este parámetro considera la diferencia media de la carga de cada estación base (L_j) con la carga media (m), que es la que tendrían las diferentes estaciones base si estuvieran equitativamente cargadas. Para medirlo se ha usado la siguiente expresión: $LF = \frac{1}{N} \sum_{j=0}^N \ L_j - m\ $ A la vista de los resultados obtenidos, se puede garantizar que nunca se excede el valor de 10 recursos por encima de la media teórica. Ya que este parámetro únicamente afecta a la gestión de la red, no se diferencia entre usuarios.
Beneficio de la red (NR)	60	0	Finalmente se ha estudiado el beneficio económico global de la red obtenido por cada estación base. Se ha definido como la ganancia media que se obtiene por TU y tiempo. De acuerdo con las observaciones empíricas se puede establecer un límite superior de 60. Al igual que en el caso anterior, al ser un parámetro global de red, no se hace diferencia entre tipos de usuarios.

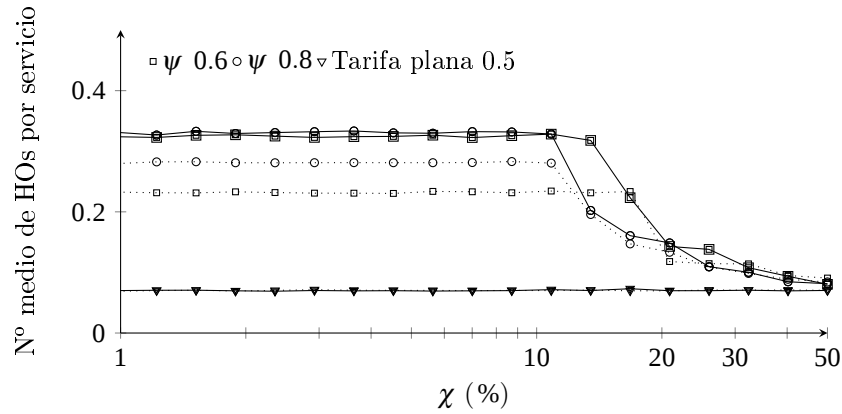


Fig. 5.12: Número medio de traspasos Vs. el parámetro de descuento χ . Las líneas continuas representan a usuarios *business* y las punteadas a los *consumer*

Con el objetivo de evaluar el rendimiento global de la red se ha hecho uso de las figuras de mérito definidas en la Tabla 5.4. En esta tabla también se indican los rangos de valores para cada una de las figuras (peor y mejor caso); para algunas de ellas estos rangos se basan en la propia definición del parámetro mientras que para otras provienen de la observación de los resultados.

Antes de presentar los resultados detallados de las figuras de mérito definidas se ha realizado un análisis que permite entender la influencia del parámetro de descuento (χ) sobre el comportamiento de la función de utilidad definida para la estrategia de selección, únicamente para el análisis mediante el simulador mCASE. Este parámetro se define como el descuento que un usuario necesitaría en una alternativa de acceso para decidir cambiar y, por lo tanto, realizar un traspaso; mientras que σ es su parámetro dual en la utilidad de los traspasos. Se ha incrementado el valor de ψ de 1% a 50% (σ se modifica de acuerdo a éste) y se han ejecutado 100 simulaciones independientes por cada valor. En la Figura 5.12 se ha representado la variación del número de traspasos de acuerdo a χ para los diferentes usuarios y para diferentes relaciones de tarifas permitidas por ellos (ψ). Se puede observar que el número de traspasos se mantiene estable, siendo ligeramente mayor para los usuarios de tipo *business*, hasta valores de χ del 10%; a partir de este punto el valor se reduce bruscamente hasta alcanzar el mínimo fijado por la tarifa plana (en este caso el único motivo para llevar a cabo un traspaso es el de pérdida de conectividad).

Por otro lado, en la Figura 5.13 se observa que el precio medio por servicio no depende de este parámetro y se corrobora que los usuarios *business* pagan más que los *consumer*, siendo esta diferencia mayor al aumentar la relación ψ .

A partir de estos resultados se ha fijado el factor de descuento χ en un 20% en la función de utilidad usada por mCASE ya que este valor asegura una reducción sustancial del número de traspasos en comparación con valores menores.

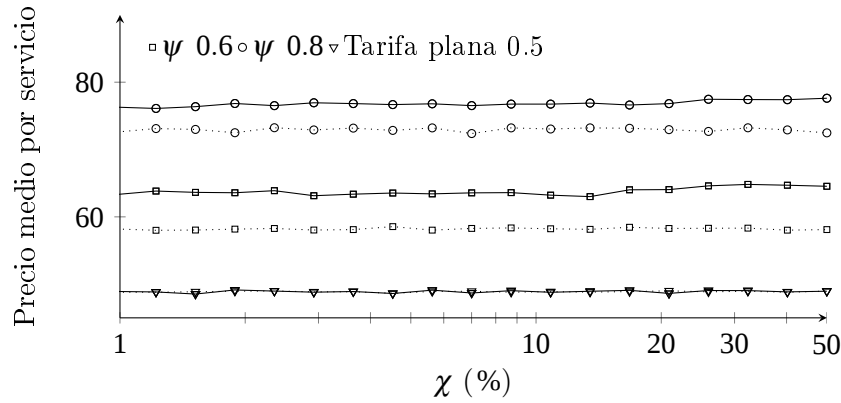
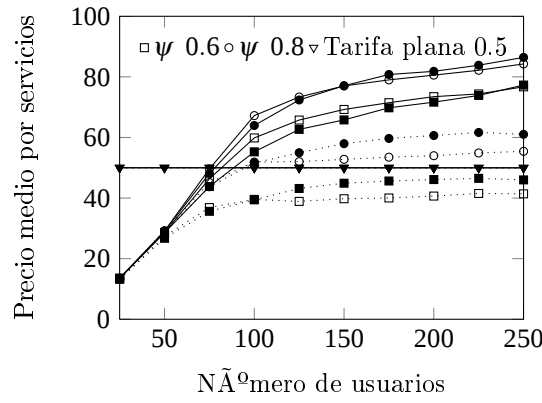


Fig. 5.13: Evolución del precio medio por servicio con el parámetro de descuento χ

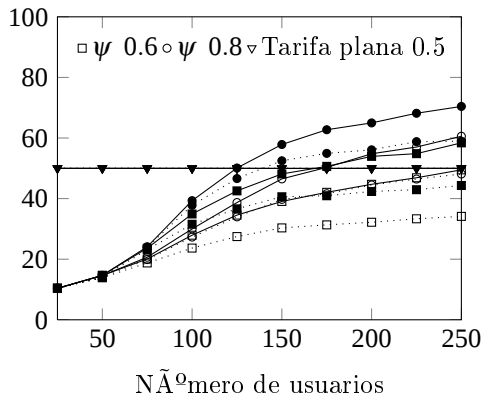
Como un primer conjunto de resultados se ha variado el número de usuarios desde 25 a 250 (can paso 25) manteniendo el porcentaje de usuarios *business* en el 30%. Un primer resultado a destacar es que la estrategia de selección propuesta para el mCASE no obtiene resultados, en términos de precio, tan buenos como la optimización; la comparación debería hacerse entre las Figuras 5.14(c) y 5.14(b) ya que esta última optimiza desde el punto de vista de los usuarios. Como se puede ver, el precio que necesitan pagar los usuarios en el caso del simulador es notablemente más alto que en el caso de la optimización basada en el usuario; tanto es así, que se acerca al de la optimización basada en el operador (ver Figura 5.14(a)). Esto es debido a que la optimización busca un mejor rendimiento global (teniendo en cuenta todas las posibles alternativas de acceso para todos los usuarios a la vez) mientras que el simulador únicamente usa información local de cada usuario.

Por otro lado, analizando el número medio de traspasos pr servicio (Figura 5.15(c)), se puede observar que la estrategia propuesta se comporta muy bien en comparación con las optimizaciones ya que se observan unos resultado que, incluso, mejoran los de la optimización. Nuevamente, estas discrepancias no quieren decir que la optimización no sea tal, sino que, teniendo en cuenta los diferentes parámetros a optimizar, el observado es el mejor rendimiento esperado. La mejora que la estrategia de selección supone en este sentido respecto a la optimización indica que un comportamiento basado en información local (de alguna manera más egoísta) mejora este parámetro en perjuicio de otros.

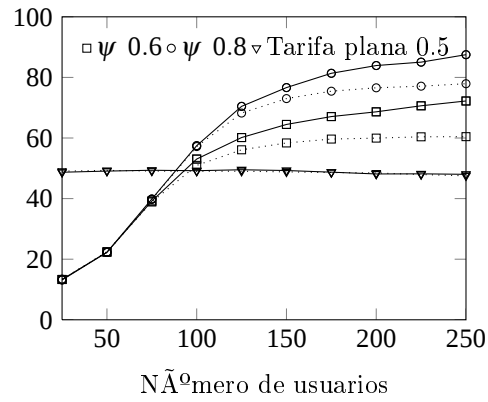
Si se comparan los resultados correspondientes a las diferentes configuraciones del problema de optimización, se observa que el efecto de incluir el parámetro de reducción de traspasos (γ) en la función objetivo tiene un impacto evidente. El número de traspasos se reduce de manera drástica para las diferentes configuraciones de relaciones de precios (ψ). Además, para la optimización basada en el usuario, el hecho de considerar este parámetro también lleva a un ligero aumento en el precio por servicio (Figura 5.14(b)) respecto a la optimización basada en el operador; esto se debe a que el pará-



(a) Precio por servicio - optimización basada en el operador



(b) Precio por servicio - optimización basada en el usuario



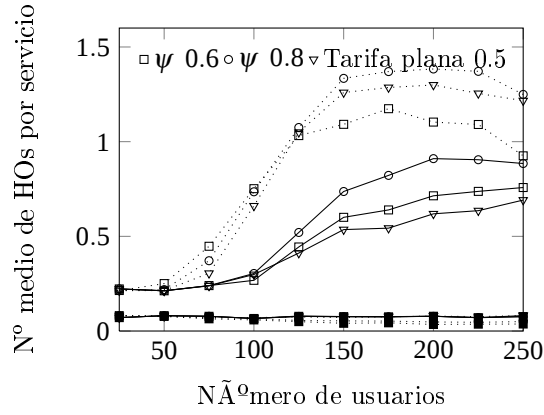
(c) Precio por servicio - mCASE

Fig. 5.14: Precio medio por servicio para los diferentes esquemas analizados. En el caso de las optimizaciones (columnas izquierda y central) los marcadores rellenos se corresponden con la función de utilidad que considera traspasos, mientras que las vacías se obtienen sin considerar el parámetro γ . Las líneas continuas corresponden a los usuarios *business* mientras que las puntuadas a los *consumer*

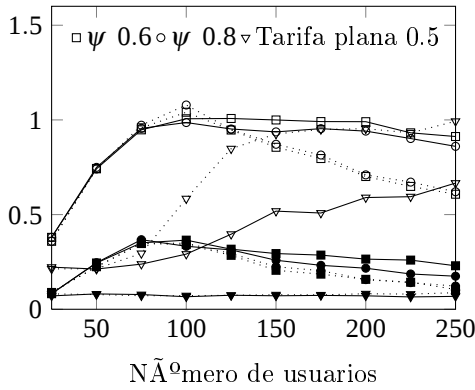
metro de precio (δ_j) no se ha considerado en la función objetivo basada en el operador mientras que sí lo fue en la basada en el usuario, ver Tabla 5.3.

Otro aspecto que se puede observar en las tres tipos de resultados es el relativo al cambio de tendencia del número medio de traspasos por servicio que realizan los usuarios de tipo *consumer* (ver Figuras 5.15(a), 5.15(b) y 5.15(c)). Cuando el número de usuarios es bajo el número de traspasos se incrementa con el aumento de los usuarios, pero llegado a un cierto valor esta tendencia varía y empieza a decrecer. La razón radica en que la carga en las estaciones base va en aumento, lo que provoca que, de acuerdo con la estrategia de tarificación que se ha definido, el hecho de que el precio aumente provoca que los usuarios *consumer* (dispuestos a pagar menos) eviten esos puntos de acceso.

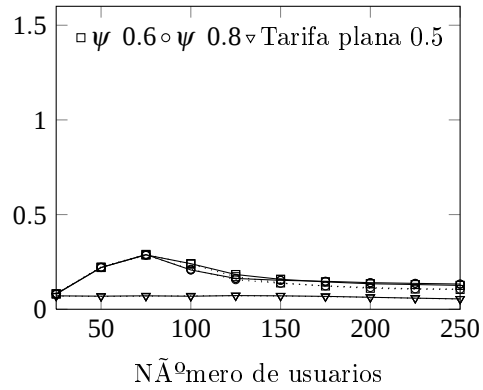
Finalmente, la Figura 5.16 provee una representación gráfica del rendimiento general que se ha observado para los diferentes procedimientos.



(a) Traspasos - optimización basada en el operador



(b) Traspasos - Optimización basada en el usuario



(c) Traspasos - mCASE

Fig. 5.15: Número medio de trasposos por servicio para los diferentes esquemas analizados. En el caso de las optimizaciones (columnas izquierda y central) los marcadores rellenos se corresponden con la función de utilidad que considera trasposos, mientras que las vacías se obtienen sin considerar el parámetro γ . Las líneas continuas corresponden a los usuarios *business* mientras que las punteadas a los *consumer*

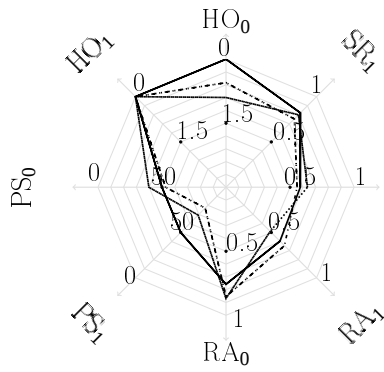
Los diagramas *web* usan como límites los indicados en la Tabla 5.4 fijando el mejor caso en los límites exteriores y los peores en el centro. Recordar que para el caso del análisis por medio del mCASE el factor de descuento (χ) es de 20%.

Una primera conclusión que se puede extraer es que la optimización basada en el operador ofrece peores resultados (especialmente en términos de balanceo de carga y precio por servicio) que el resto; esto es seguramente debido al hecho de que esta configuración prioriza claramente a los usuarios de tipo *business*, por lo que el rendimiento de los diferentes parámetros se ve comprometido (ver Figura 5.16(d)). También se observa que el rendimiento de mCASE es razonablemente bueno (Figura 5.16(c)) si se compara con la optimización basada en el usuario (Figura 5.16(e)).

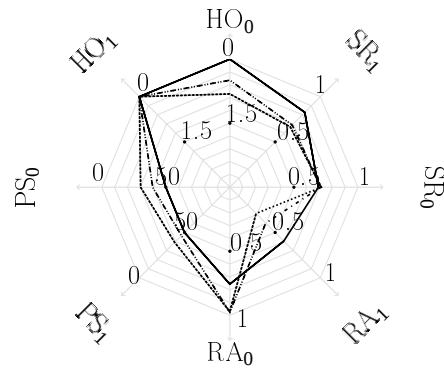
Como se vio antes, vale la pena destacar la importancia de una apropiada selección de la función objetivo en el caso de la optimización. Para

las dos configuraciones que se han estudiado se observa que añadir un coste por cambiar de estación base mejora en gran medida el rendimiento en este aspecto sin afectar negativamente al resto de parámetros.

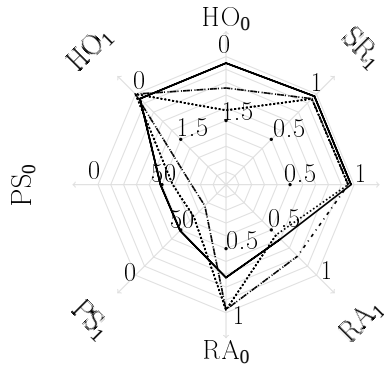
Otro aspecto interesante que se puede observar para los 3 casos es que la política de tarificación propuesta mejora en cierto grado el balanceo de carga en contraposición a la tarifa plana; en todos los casos la figura de mérito relativa al factor de carga se comporta peor usando tarifa plana. Esto prueba que el objetivo de esta estrategia de precios se ha logrado. También se puede ver, de acuerdo a la Figura 5.16(d), que la optimización basada en el operador tiene un comportamiento ligeramente peor en este sentido. Esto se debe a que el balanceo de carga se consigue mediante el precio, por lo que si, como ocurre en esta optimización (ver Tabla 5.3), no se considera el precio a la hora de seleccionar el elemento de acceso no hay ningún mecanismo que balancee la carga entre las estaciones base.



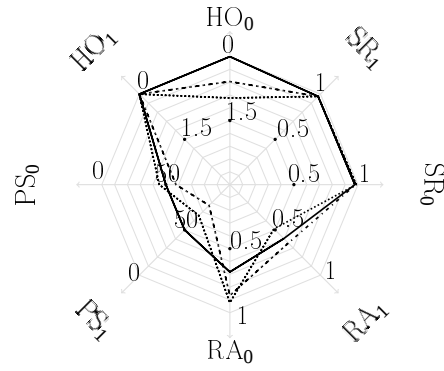
(a) Optimización basada en el operador sin parámetro de traspasos



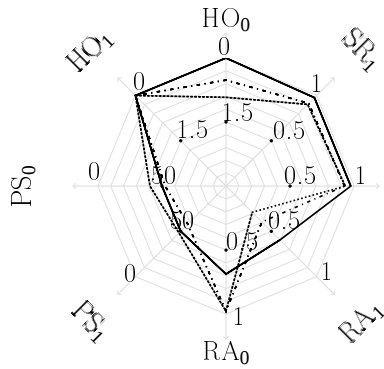
(b) Optimización basada en el usuario sin parámetro de traspasos



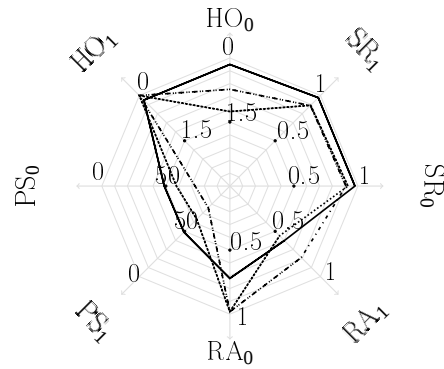
(c) mCase (χ 0.2)



(d) Optimización basada en el operador con parámetro de traspasos



(e) Optimización basada en el usuario con parámetro de traspasos



(f) mCase (χ 0.1)

Fig. 5.16: Figuras de rendimiento para los diferentes esquemas analizados. Las líneas continuas representan la tarifa plana, las punteadas la política de tarificación dinámica con ψ 0.6 y las que tiene punto/raya se corresponden con ψ 0.8

Conclusiones y líneas futuras

El aumento de las capacidades de los dispositivos móviles conlleva un incremento cada vez mayor de los recursos que consumen, por lo que la correcta gestión de las capacidades resulta clave, especialmente en la parte de acceso.

A fin de cubrir las necesidades de optimización de recursos de las redes, la arquitectura OConS proporciona un marco que integra, de manera abierta y homogénea, los diferentes servicios de conectividad. Esta integración proporcionará un mejor uso de los recursos mediante la interacción de las diferentes capas de la arquitectura de protocolos así como de los segmentos de red.

En este trabajo se ha llevado a cabo la implementación de los elementos principales que constituyen la arquitectura OConS. Estos elementos, siguiendo la filosofía de la propia arquitectura, tienen un diseño abierto lo que permite que se puedan reutilizar en soluciones futuras. A su vez, esta implementación ha permitido el desarrollo de una solución de acceso que posee una serie de ventajas frente a otras existentes; en concreto se ha comparado con el estándar IEEE 802.21. Dada la dificultad de realizar una comparación cuantitativa de definiciones de las soluciones, la comparación se ha centrado en las capacidades funcionales. En este sentido, resulta evidente que el hecho de que OConS permita la interacción entre capas y segmentos de red (tanto en la obtención de información, toma de decisiones como ejecución de acciones) puede proporcionar un valor añadido frente a otras soluciones.

Una vez comprobada la viabilidad de implementación de soluciones de movilidad en el marco OConS, se ha realizado un análisis de rendimiento de la red a fin de determinar los parámetros que habría que tener en cuenta en la decisión de la selección de acceso. Concretamente se ha hecho uso de técnicas de programación lineal con el objetivo de estudiar el rendimiento en entornos de acceso altamente heterogéneos. El marco que se

ha desarrollado, que considera la evolución temporal de los servicios, es lo suficientemente genérico como para extender su uso a diversos escenarios y casos de uso. Se ha estudiado la viabilidad del marco por medio de un escenario y mediante funciones de utilidad basadas en precio, RAT *affinity* y coste de realizar traspasos. Los resultados muestran que una elección apropiada de los parámetros configuración lleva a rendimientos diversos, por lo que esta elección debe ser estudiada cuidadosamente al fijar los parámetros que se van a tener en cuenta. En concreto los resultados revelan que la integración del coste de realizar traspasos en la función de utilidad tiene considerables beneficios ya que lleva a una mejora notable, en términos de número medio de traspasos por servicio, sin provocar perjuicio en el resto de figuras de mérito estudiadas; de hecho en algunos casos las mejora.

Finalmente, se han comparado los resultados que se obtienen de la resolución del problema de optimización anterior con lo que cabría esperar en una red cuyas decisiones se basaran en información local. Los algoritmos estudiados en este punto se han centrado principalmente en la mejora de la QoS mediante métodos de tarificación novedosos. Aunque la mayoría de las políticas de tarificación actuales se basan en algún tipo de tarifa plana, se está empezando a pensar en alternativas más apropiadas de cara al futuro. Por un lado, los operadores de red necesitan evolucionar sus políticas, especialmente debido al aumento de la demanda y diversidad de recursos y a la competitividad. Este trabajo ha presentado y estudiado la posibilidad de usar un esquema de tarificación dinámica que permitiría a los operadores modular el precio de acuerdo a la carga de red en cada momento. Esta estrategia, además de permitir realizar balanceo de carga de manera indirecta (ya que el precio aumenta al hacerlo la carga de las estaciones base), también se puede usar para proveer calidad de servicio diferenciada de acuerdo al perfil de usuario. En concreto se ha observado cómo los usuarios *business*, dispuestos a pagar más por acceder a la red, se benefician de esta estrategia de tarificación ya que perciben una mayor probabilidad de completar sus servicios satisfactoriamente.

Para este estudio se ha definido una estrategia de selección de acceso que hace uso de una función de utilidad genérica; concretamente esta función incluye la intención de los usuarios de conectarse a la alternativa más barata minimizando, a la vez, el número de traspasos. El análisis llevado a cabo con un simulador por eventos se ha complementado con los resultados obtenidos del problema de optimización anterior. Se ha observado una gran influencia de la función objetivo en el rendimiento global de la red ya que éste varía de manera muy notable en función de los parámetros seleccionados. En general, se puede decir que el rendimiento del esquema propuesto es aceptable comparado con los resultados de la optimización.

A partir del trabajo realizado las líneas futuras de investigación son variadas. Respecto al desarrollo de la arquitectura OConS, el *software* sirve como base para el desarrollo de nuevos servicios de conectividad. Sin embargo, dado que el proyecto SAIL ha concluido, el uso del desarrollo tal cual se ha definido (tipos de mensajes e interfaces) podría no adecuarse a los requerimientos, a pesar de ello la filosofía de diseño seguida puede ser

fácilmente exportable a otros casos de uso.

El campo que abre más líneas futuras de investigación es, sin duda, el que se refiere al análisis. En este sentido, el principal objetivo es el de explotar el potencial del marco desarrollado con el fin de evaluar (mediante comparación con el óptimo) el rendimiento de diferentes algoritmos distribuidos y procedimientos en entornos de selección en *HetNets*, esto incluirá el uso de recursos virtualizados y técnicas de gestión de flujos. Como se ha podido observar, el diseño de la herramienta es bastante genérico y no se limita a optimización; se puede usar para emular diferentes escenarios que se puedan representar en el formato de trazas usado para plantear diferentes problemas matemáticos. En este sentido, también se analizará el impacto que pueda tener diferentes modelos de tráfico, por ejemplo tráfico elástico que daría lugar a problemas de optimización no lineales.

La primera extensión del marco se llevará a cabo en los proyectos Low EMF Exposure Future Networks (LEXNNET) [33] (FP7-ICT-2011-8) y COConnectivity as a Service: Access for the Internet of the Future (COSAIF) (TEC2012-38574-C02-02) en diferentes aspectos (selección de acceso, gestión dinámica de potencia, etc).

Lista de acrónimos

SAIL	Scalable and Adaptative Internet Solutions
EE	Execution and Enforcement Entity
DE	Decision Making Entity
IE	Information Management Entity
INC	Inter/Intra-Node Communication
ipc	IPCInter Process Communication
OConS	Open Connectivity Service
OR	Orchestration Register
OSAP	Orchestration Service Access Point
QoE	Quality of Experience
QoS	Quality of Service
SAIL	Scalable Adaptive Internet Solutions
SOP	Service Orchestration Process
TLV	Type Length Value
OSI	Open System Interconnection
SDN	Software-defined networking
IPC	Inter-process Communication
IEEE	Institute of Electrical and Electronics Engineers
MIH	Media Independent Handover
MIHU	Media Independent Handover User
MIHF	Media Independent Handover Function
SAP	Service Access Point
3GPP	3rd Generation Partnership Project
WiMAX	Worldwide Interoperability for Microwave Access
WiFi	Wireless Fidelity

MIES	Media Independent Event Service
MICS	Media Independent Command Service
MIIS	Media Independent Information Service
LTE	Long Term Evolution
RFC	Request for Comments
IETF	Internet Engineering Task Force
SNR	Signal to Noise Ratio
RSSI	Received Radio Strength Indicator
ioctl	input/output control
AR	Access Router
DS	Data Source
EA	Execution Accomplishment
LEXNET	Low EMF Exposure Future Networks
C3EM	Comunicaciones Cognitivas y Cooperativas sobre Entornos Heterogéneos
COSAIF	COnnectivity as a Service: Access for the Internet of the Future
ODTONE	Open Dot Twenty ONE
RAT	Radio Access Technology

Referencias

- [1] Ericsson Consumer Lab. Traffic and market report - on the pulse of the networked society, 2012.
- [2] E. Gustafsson y A. Jonsson. Always best connected. *Wireless Communications, IEEE*, 10(1):49 – 55, feb. 2003. ISSN 1536-1284.
- [3] J. Baliosian et al. Gestión de políticas y precios en entornos de acceso heterogéneos. Informe técnico, Proyecto C3SEM (Comunicaciones Cognitivas, Cooperativas y Gestión Autónoma de Servicios).
- [4] Ieee 802.21: Media independent handover services.
- [5] R. Koodli. Fast Handovers for Mobile IPv6. RFC 4068 (Experimental), July 2005. Obsoleted by RFC 5268.
- [6] O. A. WG y D. Team. ONF Architecture Document - v0.0.1, December 2012. <http://www.opennetworking.org/>.
- [7] Architecture y F. D. Team. ONF Framework Document - v0.4.2, December 2012. <http://www.opennetworking.org/>.
- [8] SAIL. Architecture and Mechanisms for Connectivity Services. Deliverable FP7-ICT-2009-5-257448-SAIL/D.C.2, SAIL project, 2013. Available online from <http://www.sail-project.eu>.
- [9] *ISO/IEC 8824-1:1995/Cor.1:1996(E)*. Information technology - Abstract syntax notation one (ASN.1) : International standard ISO/IEC 8824. 1995.
- [10] *ODTONE - Open Dot Twenty ONE*, <http://hng.av.it.pt/projects/odtone>.
- [11] E. Gamma, R. Helm, R. Johnson y J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1994.
- [12] L. Diez, O. Mehani, L. Suciu y R. Agüero. Design and implementation of the open connectivity services framework. En *MONAMI 2012, 4th International Conference on Mobile Networks and Management*. 2012.
- [13] Ieee standard for local and metropolitan area networks - part21: Media independent handover services.
- [14] L. Badia y M. Zorzi. Dynamic utility and price based radio resource management for rate adaptive traffic. *Wirel. Netw.*, 14(6):803–814, diciembre 2008. ISSN 1022-0038.

- [15] O. E. Falowo, S. Zeadally y H. A. Chan. Dynamic pricing for load-balancing in user-centric joint call admission control of next-generation wireless networks. *International Journal of Communication Systems*, 23(3):335–368, 2010. ISSN 1099-1131.
- [16] U. Toseef, Y. Zaki, A. Timm-Giel y C. Görg. Uplink QoS Aware Multi-homing in Integrated 3GPP and non-3GPP Future Networks . En *Proceedings of the 4thd ICST International Conference on Mobile Networks & Management, MONAMI'12*. September 2012.
- [17] K. Pentikousis, R. Agüero, J. Gebert, J. A. Galache, O. Blume y P. Pääkkönen. The Ambient Networks heterogeneous access selection architecture. En *Proceedings of the 1st Ambient Networks Workshop on Mobility, M2NM*. Octubre 2007.
- [18] Y. Zaki, L. Zhao, C. Goerg y A. Timm-Giel. LTE mobile network virtualization. *Mob. Netw. Appl.*, 16(4):424–432, agosto 2011. ISSN 1383-469X.
- [19] L. Correia, D. Zeller, O. Blume, D. Ferling, Y. Jading, I. Goanddor, G. Auer y L. Van Der Perre. Challenges and enabling technologies for energy aware mobile radio networks. *Communications Magazine, IEEE*, 48(11):66 –72, november 2010. ISSN 0163-6804.
- [20] J. Choque, R. Aguero y L. Munoz. Optimum selection of access networks within heterogeneous wireless environments based on linear programming techniques. Aug 2011.
- [21] D. Niyato y E. Hossain. A game theoretic analysis of service competition and pricing in heterogeneous wireless access networks. *Wireless Communications, IEEE Transactions on*, 7(12):5150 –5155, december 2008. ISSN 1536-1276.
- [22] D. Niyato, P. Wang, E. Hossain, W. Saad y Z. Han. Game theoretic modeling of cooperation among service providers in mobile cloud computing environments. En *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*, páginas 3128 –3133. april 2012. ISSN 1525-3511.
- [23] C. Gu, Y. Zhang, W. Ma, N. Liu y Y. Man. Universal modeling and optimization for multi-radio access selection. En *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, páginas 1 –4. sept. 2009.
- [24] The GNU linear programming kit (GLPK). www.gnu.org/software/glpk, 2008.
- [25] J. Choque, R. Agüero y L. Muñoz. Simulation framework for the evaluation of access selection algorithms over heterogeneous wireless networks. En *3rd International ICST Conference on Mobile Networks and Management (MONAMI'2011)*. September 2011.

- [26] S. Mandal, D. Saha y A. Mahanti. A technique to support dynamic pricing strategy for differentiated cellular mobile services. En *Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE*, tomo 6, páginas 5 pp. –3392. dec. 2005. doi:10.1109/GLOCOM.2005.1578402.
- [27] L. Badia y M. Zorzi. Dynamic utility and price based radio resource management for rate adaptive traffic. *Wirel. Netw.*, 14(6):803–814, diciembre 2008. ISSN 1022-0038. doi:10.1007/s11276-007-0015-z.
- [28] L. Pirmez, J. C. C. Jr., F. C. Delicato, F. Protti, L. F. Carmo, P. F. Pires y M. Pirmez. Sutil - network selection based on utility function and integer linear programming. *Computer Networks*, 54(13):2117 – 2136, 2010. ISSN 1389-1286. doi:10.1016/j.comnet.2010.03.007.
- [29] M. Lucas-Estañ, J. Gozalvez y J. Sanchez-Soriano. Integer linear programming optimization of joint rrm policies for heterogeneous wireless systems. *Computer Networks*, 56(1):112–126, 2012. ISSN 1389-1286. doi:10.1016/j.comnet.2011.08.012.
- [30] H. Jia, Z. Zhang, P. Cheng, H.-H. Chen y S. Li. Study on network selection for next-generation heterogeneous wireless networks. En *Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on*, páginas 1–5. sept. 2006. doi: 10.1109/PIMRC.2006.253971.
- [31] P. Si, H. Ji y F. R. Yu. Optimal network selection in heterogeneous wireless multimedia networks. *Wireless Networks*, 16(5):1277–1288, julio 2010. ISSN 1022-0038. doi:10.1007/s11276-009-0202-1.
- [32] C. Bettstetter, G. Resta y P. Santi. The node distribution of the random waypoint mobility model for wireless ad hoc networks. *Mobile Computing, IEEE Transactions on*, 2(3):257 – 269, sep 2003. ISSN 1536-1233. doi:10.1109/TMC.2003.1233531.
- [33] Low emf exposure networks. <http://www.lexnet-project.eu/>.