

**MÁSTER UNIVERSITARIO EN
INGENIERÍA INDUSTRIAL**

TRABAJO FIN DE MÁSTER

***ANÁLISIS DE LOADNG COMO PROTOCOLO
DE RUTADO DINÁMICO PARA LAS
COMUNICACIONES PLC EN MT***

DOCUMENTO N°1: MEMORIA

Alumno *Rodríguez Fernández, Jorge*
Director *de la Vega Moreno, David*
Departamento *Ing. de Comunicaciones*
Curso académico *2017/2018*

Bilbao, a 6 de Febrero de 2018

Índice

1	Introducción.....	8
2	Objetivos.....	11
3	Beneficios.....	12
3.1	Beneficios técnicos	12
3.2	Beneficios económicos	12
3.3	Beneficios sociales.....	13
4	Estado del arte	14
4.1	La red eléctrica	14
4.1.1	Los centros de transformación.....	15
4.1.2	Contadores inteligentes.....	16
4.2	Características de las líneas eléctricas	17
4.2.1	Impedancia	17
4.2.2	Ruido y otras perturbaciones	17
4.2.3	Atenuación	18
4.3	Evolución hacia la red inteligente	18
4.4	LOADng	20

- 4.4.1 Introducción 20
- 4.4.2 Información almacenada por cada router LOADng 21
- 4.4.3 Formato de los paquetes 22
- 4.4.4 Formato de <message> para RREQ y RREP 23
- 4.4.5 Formato de <message> para RERR..... 24
- 4.5 Generación y tratamiento de los mensajes..... 24
 - 4.5.1 Route Request 24
 - 4.5.2 Procesamiento de un mensaje RREQ..... 25
 - 4.5.3 Route Replies..... 26
 - 4.5.4 Procesamiento del RREP 27
- 4.6 Mantenimiento de las rutas 27
- 4.7 Reparación de las rutas 28
 - 4.7.1 Procesamiento del RERR..... 28
- 5 Análisis de alternativas y selección de la solución 29
 - 5.1 Posibilidades..... 29
 - 5.1.1 Técnica de modelado..... 29
 - 5.1.2 Herramientas de simulación 30
 - 5.2 Criterios de selección..... 32

- 5.2.1 Criterios de selección de la técnica de modelado 32
- 5.2.2 Criterios de selección de la herramienta de simulación 33
- 5.3 Selección de la solución 33
 - 5.3.1 Técnica de modelado 33
 - 5.3.2 Herramienta de simulación 34
- 6 Análisis de riesgos 35
- 7 Descripción del software realizado 37
 - 7.1 Módulo router LOADng 37
 - 7.2 Módulo controlador de eventos 39
- 8 Análisis del protocolo LOADng en redes de media tensión 40
 - 8.1 Punto de partida 40
 - 8.2 Redes de referencia 40
 - 8.2.1 Topologías de 10 nodos 41
 - 8.2.2 Topologías de 17 nodos 42
 - 8.2.3 Topologías reales de 30 nodos 42
 - 8.3 Consideraciones de estudio 48
 - 8.4 Resultados 49
 - 8.4.1 Topologías de 10 nodos 50

8.4.2	Topologías de 17 nodos	51
8.4.3	Topologías reales de 30 nodos.....	52
8.5	Situación real 2 feeders	53
8.6	Conclusiones.....	57
9	Metodología seguida en el desarrollo del trabajo	59
9.1	Definición de los paquetes de trabajo y tareas	59
9.2	Desarrollo de los paquetes de trabajo y tareas.....	61
9.3	Diagrama Gantt	61
10	Entregables.....	68
10.1	Software	68
10.2	Manuales.....	68
11	Bibliografía	69

Índice de ilustraciones

Ilustración 1 Esquema red eléctrica.....	14
Ilustración 2 Centro de transformación.....	15
Ilustración 3 Contador inteligente.....	16
Ilustración 4 Evolución de red convencional a red inteligente.....	20
Ilustración 5 Diagrama de flujo procesamiento RREQ.....	26
Ilustración 6 Módulo Router LOADng.....	37
Ilustración 7 Interior Router LOADng.....	38
Ilustración 8 Modulo controlador de eventos.....	39
Ilustración 9 Topologías de 10 nodos.....	41
Ilustración 10 Topologías de 17 nodos.....	42
Ilustración 11 Distribución tranformadores urbanos.....	43
Ilustración 12 Distribución transformadores semiurbanos.....	44
Ilustración 13 Distribución transformadores rurales.....	45
Ilustración 14 Topología urbana.....	46
Ilustración 15 Topología semiurbana.....	47
Ilustración 16 Topología rural.....	47

Ilustración 17 Conexión física 2 feeders 54

Ilustración 18 Conexión lógica 2 feeders 55

Ilustración 19 Resultado 2 feeders 56

Ilustración 20 Diagrama Gantt..... 67

Índice de tablas

Tabla 1 Tabla de modelado	34
Tabla 2 Herramienta de simulación	34
Tabla 3 Resumen simulaciones.....	49
Tabla 4 Búsqueda inicial 10 nodos.....	50
Tabla 5 Búsqueda tras desconexión 10 nodos	50
Tabla 6 Búsqueda inicial 17 nodos.....	51
Tabla 7 Búsqueda tras la desconexión 17 nodos	51
Tabla 8 Búsqueda inicial 30 nodos.....	52
Tabla 9 Búsqueda tras la desconexión 30 nodos	53

1 Introducción

Las comunicaciones sobre la línea eléctrica o PLC (Power Line Communication) se refieren a cualquier tecnología que permita transferir datos a través de la red eléctrica. Su principio consiste en la superposición de una señal de alta frecuencia con bajos niveles de energía sobre la señal de 50 Hz de la red eléctrica.

Esta tecnología está ampliamente desarrollada en la red de baja tensión, especialmente para comunicaciones orientadas al Smart Metering mediante tecnologías como pueden ser PRIME (Powerline Intelligent Metering Evolution), estándar pensado como solución abierta que busca la interoperabilidad de los equipos entre distintos proveedores, G3-PLC, desarrollado por Sagem, ERDF y Maxim que busca también la interoperabilidad en la comunicación pero focalizado más directamente al Smart Grid, o Meters and More, protocolo que Endesa tiene implementado en sus medidores electrónicos.

La extensión de las tecnologías PLC a las redes de media tensión para mantener la comunicación entre dispositivos es una opción recientemente considerada, pues permitiría realizar acciones de control o monitorización utilizando la infraestructura ya existente entre transformadores.

La red de media tensión en comparación con la red de baja tensión es una red más mallada, con menor número de nodos y con mayor distancia entre ellos. Así la tecnología PLC en media tensión puede reducir la tasa de colisiones debido a estas características y, por tanto, aumentar la tasa neta

de ocupación del canal, siento todo esto siempre condicionado por los patrones de generación de información.

Aunque hay situaciones que la misma tecnología que se utiliza para PLC de baja tensión se puede utilizar en media tensión rebajando los requerimientos de la red, ZIV busca desarrollar tecnologías que solucionen los problemas que propone esta nueva implantación de PLC y así poder seguir ampliando el concepto de Smart Grids en la red eléctrica pudiendo beneficiarse de ésta.

Dado que la red de media tensión resulta ser una topología mallada con amplia cantidad de conexiones redundantes (al contrario que la red de baja tensión que tiene estructura de árbol), se puede disponer de rutas de comunicación alternativas para el caso de indisponibilidad de algunos tramos. Los dispositivos que estén conectados a la red deben disponer de tablas de enrutamiento que les permitan conocer además de la ruta preferente, rutas alternativas en caso de que se produzca alguna indisponibilidad en la ruta principal y así poder seguir manteniendo la comunicación. Es entonces necesario la utilización de protocolos de enrutamiento que aseguren las comunicaciones en todo momento.

Los protocolos de rutado LLN (Low Power and Lossy Networks) son protocolos pensados para comunicaciones Wireless, redes en las que se producen conexiones y desconexiones de forma frecuente y en las que se busca minimizar el gasto energético optimizando las rutas. Dichas características podrían ser útiles para las comunicaciones PLC en la red de

media tensión. Por esta razón, ZIV está estudiando la adaptación de estos protocolos a la red de media tensión.

Entre todos los protocolos LLN, el primero a estudiar resulta ser LOADng (LLN On-demand Ad hoc Distance-vector Routing Protocol - Next Generation), protocolo derivado de AODV, que funciona bajo demanda, de forma descentralizada, y con el que calcular y comparar rutas para seleccionar finalmente la más óptima.

2 Objetivos

El objetivo de este proyecto es el de crear una herramienta que sirva para evaluar la aplicabilidad del protocolo de enrutamiento LOADng en diferentes escenarios de la red de media tensión.

Para ello, la herramienta tiene que dar la posibilidad de crear la topología que se quiera examinar, estableciendo parámetros de la red como puede ser el número de nodos y características de los enlaces y además ser capaz de simular los eventos que se pueden producir en estas redes como conexión/desconexión de un enlace o aparición/desaparición de nuevos nodos.

También debe aportar datos estadísticos de interés que permitan al usuario recabar la información deseada de las simulaciones y así estudiar el rendimiento del protocolo con los parámetros simulados, en redes de diferente número de nodos y diferente topología.

Otro punto importante a destacar del proyecto es que la herramienta no debe requerir conocimientos de programación en OMNeT++ para poder realizar las simulaciones.

3 Beneficios

A continuación, se exponen los beneficios que se obtienen gracias a la realización de este proyecto.

3.1 Beneficios técnicos

Añadir nueva funcionalidad a la red eléctrica pudiendo utilizarla con otra finalidad además de la distribución de la electricidad, la de medio para realizar comunicaciones.

Monitorizar y realizar acciones de control con la red eléctrica como medio de intercambio de información de forma transparente, ya que la red se configura de forma automática sin necesidad de intervención, el protocolo buscará rutas para realizar comunicaciones de forma autónoma.

Con este proyecto se va a disponer de un modelado que permite realizar el análisis del rendimiento del protocolo LOADng y con el que se podrá modelar distintos escenarios. Además, este modelado podrá incluir nuevas funcionalidades o modificar las ya existentes y evaluar el impacto de esas nuevas características

3.2 Beneficios económicos

Destacar los bajos costes de despliegue de la instalación ya que la red ya existe y no necesita mayor modificación que la adhesión de routers a los transformadores para su comunicación. También señalar que los dispositivos consumen poca potencia y no tienen que estar conectados todos centros de transformación.

Por otra parte, obtener el rendimiento del protocolo antes de su implementación permite analizar en detalle el comportamiento del mismo con menos coste de lo que costaría su desarrollo en un entorno real debido al despliegue requerido. Los protocolos modelados posteriormente podrán beneficiarse del hecho de tener una base de comparación que permita medir su pérdida o ganancia con el protocolo LOADng y así obtener unos datos fiables para una mejora posterior si es necesario sin necesidad de incurrir en gastos adicionales en sus primeras etapas de desarrollo.

3.3 Beneficios sociales

Todos usuarios de la red eléctrica salen beneficiados, la red gana nuevas funcionalidades que permiten hacer control remoto del uso de la red y así monitorizar mejor su demanda y optimizar su funcionamiento.

4 Estado del arte

4.1 La red eléctrica

El sistema eléctrico está compuesto por tres subsistemas: la red de generación, cuya función es introducir la energía en el sistema, la red de transporte, que se encarga de transportar la energía hasta los núcleos donde se suministra, y por último la red de distribución, encargada de hacerla llegar a los puntos de consumo finales. Este esquema es posible gracias a los transformadores de tensión.

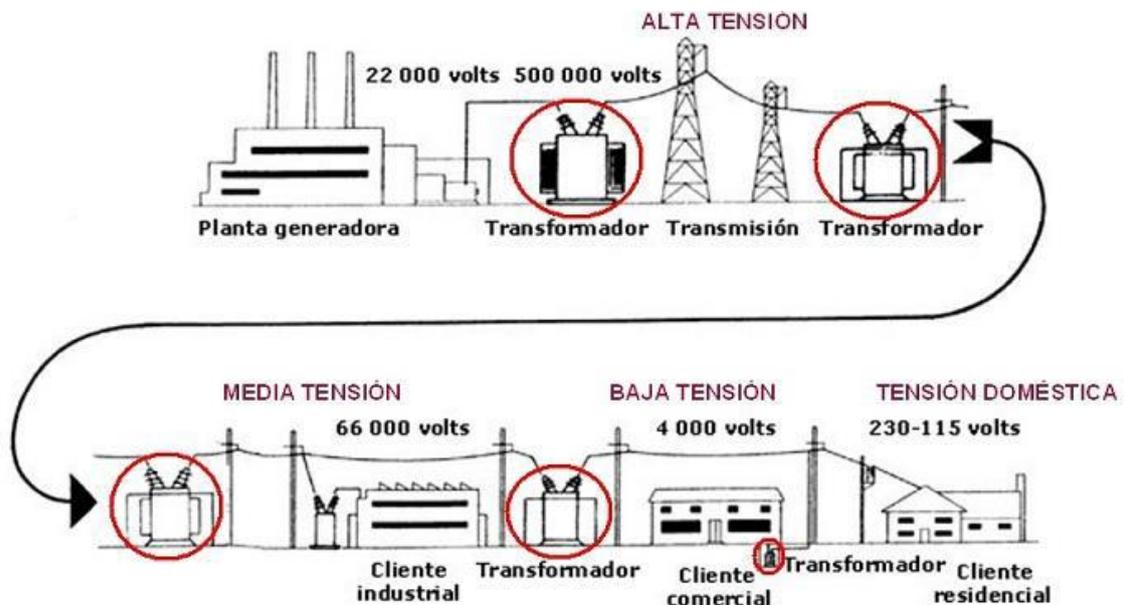


Ilustración 1 Esquema red eléctrica

Tanto la red de transporte como la red de distribución se componen de subestaciones y líneas eléctricas, y la diferencia fundamental está relacionada con los niveles de tensión, y por consiguiente, con los materiales y dimensiones que adquieren unos y otros.

La red eléctrica es, por tanto, el despliegue jerarquizado de medios utilizados para llevar la energía desde los centros donde se elabora, hasta los puntos donde se consume. Y esto se producirá de una forma eficiente si la red utiliza unos niveles de tensión apropiados, desde los más altos en los centros de producción hasta los más bajos en los puntos de suministro.

Nos centraremos en la red de distribución por ser el objeto de desarrollo de este proyecto. Y más concretamente centraremos nuestra atención en la parte de esta red que abarca desde los centros de transformación hasta los puntos de suministro.

4.1.1 Los centros de transformación

Los centros de transformación son emplazamientos donde se transforma la tensión de MT a BT por medio de transformadores eléctricos.



Ilustración 2 Centro de transformación

4.1.2 Contadores inteligentes

Un contador es un instrumento que contabiliza la energía consumida en un determinado punto de suministro a lo largo de un periodo de tiempo, y que puede encontrarse en los puntos de suministro.

A lo largo del tiempo, estos aparatos han sufrido una gran evolución pasando de instrumentos electromagnéticos a electrónicos. Y esto ha supuesto la inclusión de un mayor número de prestaciones y una flexibilidad muy superior.



Ilustración 3 Contador inteligente

Su implantación en el mercado ha sido muy escasa, debido principalmente a su alto coste. No obstante, el Real Decreto 1110/2007 obligaba a cambiar los contadores por otros que pudiesen ser telegestionados, y esto de alguna manera ha supuesto un auge en el uso de estos contadores electrónicos.

Por ello se les intenta sacar el máximo rendimiento, buscando funcionalidades más allá de la de medida.

4.2 Características de las líneas eléctricas

El cableado eléctrico no está originalmente diseñado para transportar datos, sino que se escogió por sus características para transportar señales a una cierta tensión y frecuencia, tal que 220V 50Hz.

Esta sección introduce algunas propiedades físicas y características propias que limitan a su vez sus capacidades.

4.2.1 Impedancia

La red eléctrica tiene una impedancia Z (normalmente entre 10 ohmios y 1kohmio), que es el valor absoluto de los componentes resistivos, inductivos y capacitivos de la red eléctrica. En función de si se conectan o desconectan dispositivos este valor cambiará haciendo difícil ajustar el medio de comunicaciones y por tanto la transmisión del canal. Aunque también sufrirá cambios, si variamos su modo operacional, la velocidad, el diseño etc.

4.2.2 Ruido y otras perturbaciones

El canal de transmisión adquiere cierto nivel de ruido procedente de los dispositivos conectados al cableado eléctrico o que se encuentran en sus proximidades. Podemos identificar diferentes tipos de ruido:

- Ruido impulsivo (Producido al apagar y encender dispositivos eléctricos).

- Ruido blanco de banda ancha (La densidad espectral de potencia es la misma en todas las frecuencias).
- Ruido periódico de múltiples frecuencias.
- Ruido armónico (Compuesto por múltiples frecuencias usadas por el equipo eléctrico conectado a la línea y son múltiplos de la frecuencia de la línea).

4.2.3 Atenuación

De la misma forma que una onda de radio es atenuada a medida que viaja por el espacio, las señales que envía el equipo PLC también son atenuadas a medida que viajan por el cableado eléctrico.

Las variaciones en la impedancia de la red eléctrica provocan efectos dañinos como el multitrayecto, dando lugar a picos de amplitud en la señal, los cuales se hacen más considerables a ciertas frecuencias.

Las emisiones electromagnéticas que puede producir un equipo eléctrico, deben cumplir con las normas establecidas, para reducir lo máximo posible todo este tipo de efectos no deseados.

4.3 Evolución hacia la red inteligente

Las redes inteligentes han de ofrecer soluciones en los ámbitos relacionados con la operación de la red eléctrica, siendo capaces de ser desplegados con rapidez y eficiencia de costes y que utilicen protocolos abiertos.

Éstas permiten proporcionar servicios de valor añadido para los clientes y las propias empresas eléctricas.

La medida inteligente es una funcionalidad adicional a la red inteligente, y por ello, estas redes han de ofrecer servicios innovadores para el negocio eléctrico. Ejemplos de estos servicios son pueden ser:

- Lectura remota de energía y potencia.
- Lectura remota de parámetros de calidad.
- Programación remota de tarifas.
- Conexión y desconexión a distancia
- Supervisión de MT y BT.
- Alarmas de las instalaciones.
- Detección de averías en la red.
- Automatización y control en tiempo real de la MT.
- Mejora en la atención a los clientes.

RED CONVENCIONAL



RED INTELIGENTE

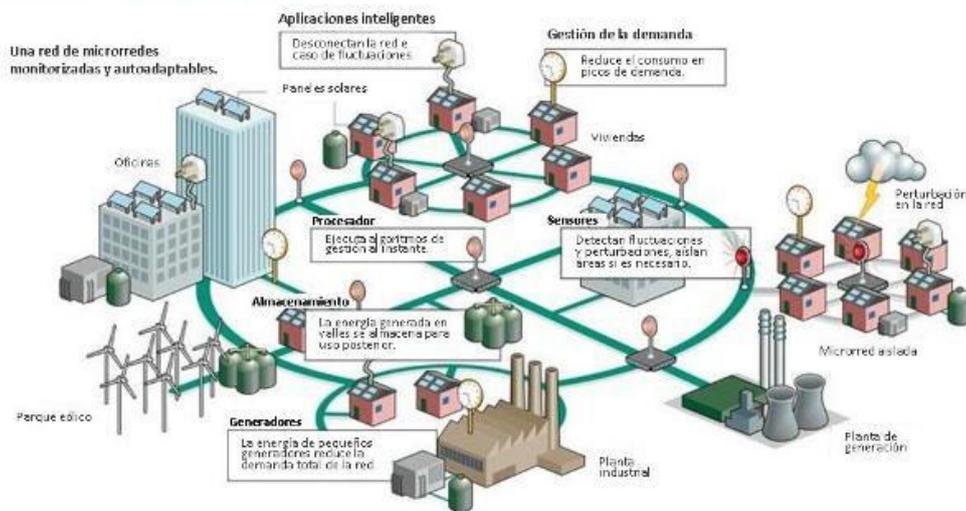


Ilustración 4 Evolución de red convencional a red inteligente

4.4 LOADng

4.4.1 Introducción

LOADng (LLN On-demand Ad hoc Distance-vector Routing Protocol Next Generation) es un protocolo de enrutamiento derivado de AODV (Ad hoc On-Demand Distance Vector Routing) y usado en redes de baja potencia con pérdidas (Low power Lossy Networks, LLNs). Se trata de un protocolo reactivo y sus operaciones principales son:

- Generar una petición de ruta (RREQs) por un router LOADng para encontrar la ruta a un destino.



- Reenviar el RREQs hasta que llegue al destino deseado.
- Una vez alcanzado el destino, generar una respuesta de ruta (RREPs) que salto a salto lleve ese RREPs hasta el router que generó el RREQs creando así la ruta para enviar y recibir información del origen al destino.
- Si se produce alguna desconexión de alguna ruta, este protocolo es capaz de generar un mensaje de error de ruta (RERR), que hará que se reinicie el proceso de creación de una nueva ruta que permita llegar al destino.

El objetivo de este protocolo es que cada router LOADng sea capaz de realizar de forma independiente:

- Descubrir un camino bidireccional a cualquier destino de la red adaptándose a posibles cambios en la red.
- Establecer la conexión solo cuando hay información que mandar a través de la ruta.
- Mantener la conexión únicamente cuando hay tráfico usando la ruta.
- Generar tráfico de control en función de los eventos que sucedan en la red, eso es cuando se necesite una ruta o cuando se detecte una ruta como rota. Este protocolo no necesita enviar señales de forma periódica.

4.4.2 Información almacenada por cada router LOADng

Con este protocolo no se crea una tabla global de enrutamiento con todos los caminos si no que es cada router LOADng el que independientemente

guarda información, de forma temporal, que permitirá enviar el mensaje en la dirección correcta dependiendo de su destino.

La información que guarda cada router para cada camino conocido es:

$$(R_dest_addr, R_next_addr, R_dist, R_time)$$

Donde:

- R_dest_addr es de la dirección del destino.
- R_next_addr es la dirección del próximo salto para llegar a R_dest_addr .
- R_dist es el coste del camino seleccionado para llegar a R_dest_addr .
- R_time es el tiempo que va a ser guardada esta dirección en el router.

Con esta información cuando un mensaje le llega a un router, este es capaz de direccionarlo al siguiente router y así salto a salto el mensaje llegará a su destino.

4.4.3 Formato de los paquetes

Todos los paquetes usados por este protocolo siguen la siguiente estructura:

$$\langle packet \rangle := \langle type \rangle \langle tlv_block \rangle \langle message \rangle$$

Donde:

- `<type>` son 4 bits que especifica que el tipo de `<message>` es (RREQ, RREP o RERR).
- `<tlv-block>` son un conjunto de TLV (valores de longitud tipo) que informan sobre la codificación del mensaje
- `<message>` donde se guarda la información que cada tipo de mensaje (RREQ, RREP o RERR) necesita para realizar su cometido.

4.4.4 Formato de `<message>` para RREQ y RREP

Tanto el formato del mensaje Route Request (RREQ) como el de Route Reply (RREP) siguen la siguiente estructura:

```
<message >:=<flags><addr-length><cost-type><weak-links><rreq-ID><route-cost><destination><originator>
```

Donde:

- `<flags>` 4 bits reservados para la interpretación del mensaje
- `<addr-length>` 4 bits que indican la distancia entre `<destination>` y `<originator>`
- `<cost-type>` 4 bits que determinan cómo es calculado `<route-cost>`
- `<weak-links>` 4 bits que almacena la cantidad de enlaces débiles que hay en la ruta
- `<rreq-ID>` 8 bits que identifican de forma única el mensaje
- `<route-cost>` 8 bits que almacenan el coste de la ruta desde el origen al destino final.

- `<destination>` identificador que especifica la dirección final del mensaje
- `<originator>` identificador que especifica la dirección original del router que solicita enviar información a un destino.

4.4.5 Formato de `<message>` para RERR

```
<message >:=<error-code><addr-  
length><source><destination>
```

Donde:

- `<error-code>` son 4 bits que informan sobre porqué se ha generado este mensaje de error.
- `<addr-length>` 4 bits que indican la distancia entre `<destination>` y `<originator>`
- `<source>` informa sobre la dirección de destino del mensaje que no ha podido llegar a su `<destination>`
- `<destination>` dirección a la cual el mensaje no ha podido llegar y se ha reportado el error.

4.5 Generación y tratamiento de los mensajes

4.5.1 Route Request

Un mensaje Route Request (RREQ) se genera por un router LOADng cuando este router tiene información que mandar a un destino y no hay una ruta creada. Este mensaje se manda a todos los routers LOADng vecinos.

Una vez mandado este mensaje, el router originador espera hasta obtener un mensaje Route Replies (RREP). Si no hay RREP de respuesta en el tiempo estimado, se vuelve a enviar el RREQ hasta un máximo de reintentos estimado por la topología de la red.

4.5.2 Procesamiento de un mensaje RREQ

Cuando un router LOADng recibe un mensaje RREQ lo primero que debe hacer es comprobar si es un mensaje válido para él, si no es así, es desechado sin realizar más operaciones con él. Una vez comprobado que se trata de un mensaje válido se realizan las siguientes operaciones:

1. Si el RREQ se recibió por un enlace débil se incrementa en uno `<weak-links>`
2. Se comprueba en la lista de enrutamiento si tiene la ruta al destino.
3. Si no hay ruta creada, se crea una nueva.
4. Sea nueva o no la ruta, se compara con la información recibida en el RREQ:
 - Si $R_dist \leq \text{<route-cost>}$ la ruta que se encuentra ya en router es mejor que la posible ruta que está realizando el RREQ así que no se reenvía.
 - Si la ruta que está realizando el RREQ es mejor que la que se encuentra en el router:
 - A. Se actualiza la tabla de enrutamiento del router:
 $R_next_addr = \text{Anterior salto por el que llegó el RREQ}$
 $R_dist = \text{<route-cost>}$
 $R_time = \text{tiempo actual} + \text{tiempo de espera}$

- B. Se procesan los TLV según su especificación.
- C. Si <destination> es el router donde se encuentra, se genera el RREP.
- D. Si no es el destino, se reenvía el RREQ a todos sus routers vecinos excepto a su predecesor y se actualiza <route-cost>

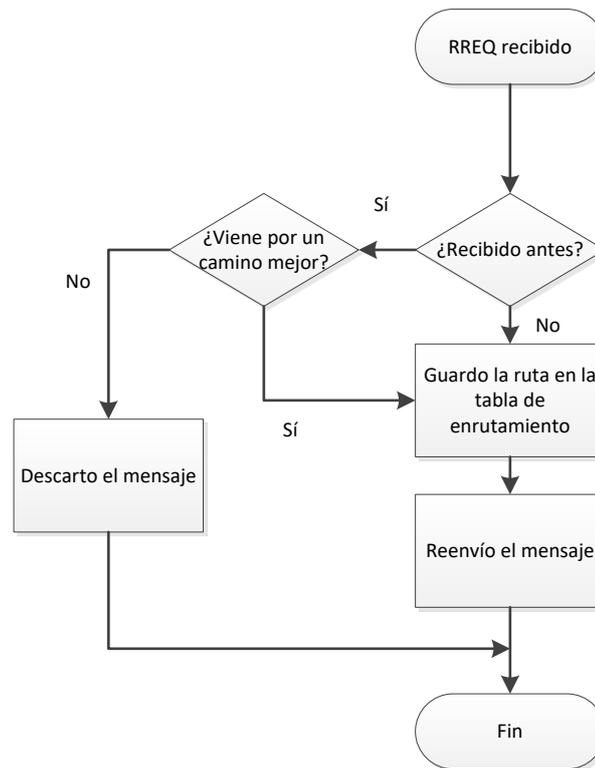


Ilustración 5 Diagrama de flujo procesamiento RREQ

4.5.3 Route Replies

Un mensaje Route Replies (RREP) lo genera un router LOADng en respuesta a un RREQ cuando su <destination> corresponde con el router en el que se encuentra. El RREP se manda salto a salto y de forma única hasta el router LOADng que creó el RREQ mediante las tablas almacenadas en cada router creadas por el avance de los RREQ. Una vez el RREP llega a su

<originator> se establece el camino bidireccional para el envío de información.

4.5.4 Procesamiento del RREP

Cuando un router LOADng recibe un RREP lo único que debe hacer es leer la información de la tabla de enrutamiento creada por los RREQ, crear una nueva tabla por cada router que pase (como se hizo con el RREQ) estableciendo la ruta de ida de la información que se mandará y reenviar el RREP al siguiente destino.

Cuando el RREP alcanza el router LOADng que corresponde con <originator> ya queda definida la ruta bidireccional desde <originator> a <destination>.

4.6 Mantenimiento de las rutas

El proceso de mantenimiento de las tablas de enrutamiento de los routers se realiza con los siguientes métodos ya antes comentados:

1. Intercambio de RREQ y RREP que crea y modifica las rutas.
2. Envío con éxito de información que corrobora que la ruta creada por RREQ y RREP es funcional.
3. Envío fallido de información que reiniciará el proceso de envío de RREQ y RREP para crear una nueva ruta.
4. Señales externas que indique que algún punto de la ruta ha sido desconectada necesitando una nueva ruta.

4.7 Reparación de las rutas

Si se detecta la rotura de una ruta, se intentará buscar una nueva ruta que permita llegar al destino. Para ello se crea un Route Error (RERR) que se manda a través del camino de vuelta hasta el <originator> del RREQ informándole de que no ha sido posible la entrega de la información.

4.7.1 Procesamiento del RERR

El Route Error es enviado de forma única por el camino de vuelta de la ruta establecida hasta el <originator> actualizando las tablas de enrutamiento. Cuando llega al router LOADng que inició la comunicación si iniciará un nuevo proceso de búsqueda de ruta al destino con los RREQ y RREP.

5 Análisis de alternativas y selección de la solución

Se presentan a continuación las diferentes alternativas susceptibles de análisis a la hora de desarrollar el proyecto, el proceso de selección que se ha seguido para la elección de la solución y la solución final alcanzada.

5.1 Posibilidades

- Técnica de modelado
- Herramienta de simulación

Mencionar en este caso que la herramienta de simulación es una alternativa secundaria que ha surgido después de elegir la opción de simulación como mejor alternativa en técnica de modelado.

5.1.1 Técnica de modelado

En general, se dispone de tres técnicas de evaluación de rendimiento en sistemas y redes; análisis matemático, simulación en ordenador y la realización de medidas en una implementación real. A continuación vamos a exponer brevemente cada una de ellas relacionándolas a este proyecto en particular:

- Análisis matemático

Este método consiste en construir un modelo lógico-matemático que describa mediante ecuaciones y relaciones el comportamiento básico del protocolo NEMO BS. El estudio, análisis y la verificación de rendimiento podrían hacerse por métodos analíticos o numéricos.

- Simulación

Esta técnica consiste en construir un programa de ordenador que describa el comportamiento del protocolo LOADng y proceder a experimentar con el mencionado programa para llegar a conclusiones en cuanto a su rendimiento.

- Implementación real

Esta alternativa consiste en implementar el protocolo LOADng en equipos reales y proceder a experimentar en entornos también reales para los distintos escenarios de rendimiento.

5.1.2 Herramientas de simulación

- ns-3:

NS-3 es un simulador de redes de eventos discretos distribuido bajo licencia GNU GPL versión 2. Orientado principalmente para investigación y entornos educativos, nació como sustituto de ns-2 el estándar de facto de investigación académica para protocolos de red y métodos de comunicación mejorando en realismo, facilidad de depuración y diseño de arquitectura entre otros. Para hacer los modelos más cercanos a la implementación real al que representan, ns-3 enfatiza en emulación. En ns-2, el sistema bilingüe de C++/Tcl hacía complejo la depuración pero en ns-3 es suficiente el conocimiento en C++ si bien los programas de simulación pueden ser scripts de python. Un único lenguaje como arquitectura es más robusta a largo plazo y tiene un estándar de codificación, revisión de código

e infraestructura de testeo más rigurosa que su predecesor haciendo un simulador más estable. Viene acompañado de Nam, una herramienta de animación basada en Tcl/TK para ver las trazas de simulación de la red, de ahí el nombre del proyecto.

- **OMNeT++ (Objective Modular Network Testbed ++):**

OMNeT++ es un entorno de simulación de eventos discretos (DES, Discret Event Simulation) de Licencia Pública Académica. Utiliza un diseño de arquitectura abierto, con componentes modulares donde componentes simples y pequeños programados en C++ se ensamblan en módulos más complejos usando el lenguaje construido internamente en OMNeT++, NED (NEtwork Description). Está diseñado para simular sistemas de eventos discretos pero su principal área de aplicación es la simulación de redes de comunicación gracias a su framework genérico y flexible y por consiguiente, su uso se está generalizando en comunidades científicas como plataforma de simulación de redes. Posee una interfaz gráfica (GUI, Graphic User Interface) que posibilita que tanto el diseño como la ejecución sean visualizadas.

- **Riverbed SteelCentral™ NetModeler Suite/OPNET Modeler Suite (Optimized Network Engineering Tool Suite):**

Es un simulador de eventos discretos que aunque comercial, se ofrece bajo términos favorables (gratuito) para uso académico. Dispone de herramientas productivas y eficaces a la hora de generar modelos, ejecución de simulaciones y análisis de datos además de una amplia gama de librerías para simular distintos sistemas de comunicación.

5.2 Criterios de selección

5.2.1 Criterios de selección de la técnica de modelado

- **Precisión:** Mediante este criterio se valorará la exactitud con la que permita medir cada técnica. La puntuación de 0 sería para un resultado poco preciso y la de 10 para una precisión que roza la perfección. Este criterio de evaluación tiene un peso del 30% sobre la decisión final.
- **Complejidad:** Se refiere a la dificultad que tiene cada técnica para el correcto análisis de rendimiento. La puntuación de 0 sería para una tarea demasiado compleja y la de 10 para una tarea de complejidad nula. Este criterio de evaluación tiene un peso del 15% sobre la decisión final.
- **Flexibilidad:** Con este criterio se pretende valorar la flexibilidad que permite cada técnica de modelado en cuanto a los escenarios de rendimiento a analizar. La puntuación de 0 sería para una variedad nula de escenarios y la de 10 para todos los escenarios imaginables. Este criterio de evaluación tiene un peso del 25% sobre la decisión final.
- **Coste:** Mediante este criterio se valorará el dinero que haya que desembolsar para obtener el software y el hardware necesarios para cada técnica. La puntuación de 0 sería lo más costoso y la de 10 lo menos. Este criterio de evaluación tiene un peso del 30% sobre la decisión final.



5.2.2 Criterios de selección de la herramienta de simulación

- **Sencillez y usabilidad:** Por un lado, este criterio medirá una herramienta modular con la que sea fácil familiarizarse y por otro lado, también se valorará una buena documentación para reducir la curva de aprendizaje. La puntuación de 0 sería para una complejidad extrema en cuanto al manejo de la herramienta se refiere y la de 10 para una herramienta casi instintiva. Este criterio de evaluación tiene un peso del 40% sobre la decisión final
- **Variedad de módulos:** Se valorará la funcionalidad disponible en cuanto a protocolos, sistemas y escenarios de la herramienta de simulación. La puntuación de 0 sería para una variedad nula de módulos y la de 10 para una librería de módulos completa. Este criterio de evaluación tiene un peso del 30% sobre la decisión final.
- **Rapidez y ligereza:** Este criterio evaluará favorablemente la utilización de pocos recursos de la máquina y la rapidez de las simulaciones. La puntuación de 0 sería para una herramienta lenta y pesada mientras que la de 10 sería para una herramienta de alto rendimiento. Este criterio de evaluación tiene un peso del 20% sobre la decisión final.

5.3 Selección de la solución

5.3.1 Técnica de modelado

	Precisión (30%)	Complejidad (15%)	Flexibilidad (25%)	Coste (30%)	Total
Análisis Matemático	5	6	8	8	6,8
Simulación	8	6	7	7	7,15
Implementación real	9	5	5	5	6,2

Tabla 1 Tabla de modelado

Así que será la **simulación** el método de análisis elegido

5.3.2 Herramienta de simulación

	Usabilidad (40%)	Módulos (30%)	Rapidez (25%)	Total
NS-3	4	9	7	6,25
OMNet++	8	7	8	7,7
SteelCentral NetModeler	6	9	6	6,9

Tabla 2 Herramienta de simulación

La herramienta seleccionada entonces para la simulación resulta ser **OMNet++**.

6 Análisis de riesgos

Existen ciertos riesgos que pueden surgir tanto durante el desarrollo del proyecto, como en su posterior uso. Éstos deben ser valorados y tenidos en cuenta antes de comenzar a desarrollarse. Por ello, se han tenido en cuenta los riesgos que se detallan a continuación.

- **Error en las estimaciones temporales realizadas:**

Durante el proyecto, se ha intentado respetar el tiempo, pero en muchas ocasiones no se ha podido seguir al pie de la letra por errores que han ido surgiendo, y que al final han dado lugar a retrasos importantes.

- **No disponibilidad temporal del ordenador:**

En un principio hubo una serie de problemas con la versión de OMNet++. No obstante, no se ha perdido mucho tiempo en solucionarlo, con lo cual no ha supuesto un obstáculo para el desarrollo del proyecto. A parte, le proyecto finalmente se ha realizado en mi ordenador portátil propio, así que cualquier problema de este tipo lo he podido solucionar sin la necesidad de estar en el aula ZIV

- **Pérdida de información almacenada en el ordenador por posibles fallos del mismo, o en otros dispositivos:**

No se ha producido ninguna pérdida de información de ninguna de las versiones estables de esta herramienta. Para evitarlo se han realizado

copias de seguridad periódicas de forma semanal para evitar cualquier problema.

- **Mala especificación funcional de la herramienta a realizar:**

La funcionalidad de la herramienta estaba clara desde el principio. La empresa ZIV tenía claro lo que demandaban.

7 Descripción del software realizado

La red de media tensión está formada por un conjunto de centros de transformación conectados mediante cable de la red eléctrica. Cada centro de transformación contaría con un router con el protocolo LOADng implementado para manejar las comunicaciones.

Puesto que OMNeT++ utiliza un sistema modular de elementos para la realización de simulaciones, cada centro de transformación se modeliza como un router configurable por el usuario y estos routers están conectados mediante cable eléctrico el cual también puede ser configurado por el usuario.

Dado que en la red de media tensión pueden producirse modificaciones en la red, también se ha modelizado un elemento que se encarga de manejar estos posibles eventos, como son, conexión/desconexión de algún elemento o modificación de los parámetros ya sea de un router o de alguna conexión. Con esto se busca simular el comportamiento del protocolo de enrutamiento ante cambios en la topología de la red y ver su efectividad en situaciones cambiantes.

7.1 Módulo router LOADng



Ilustración 6 Módulo Router LOADng

Este módulo simula el funcionamiento de un router con el protocolo LOADng. Como OMNeT++ funciona modularmente dentro de este módulo se encuentran otros submódulos, cada uno con una funcionalidad, pero que conectados entre sí, funcionan en conjunto como un router LOADng.

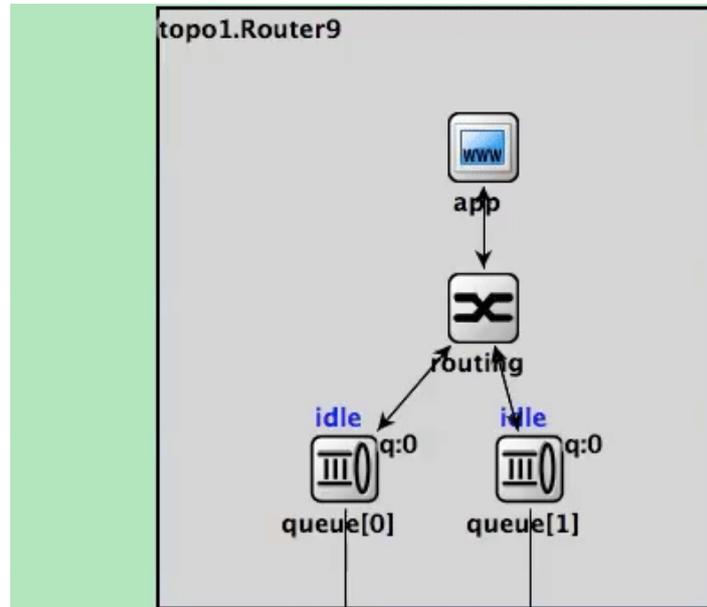


Ilustración 7 Interior Router LOADng

Este módulo está compuesto por un submódulo llamado *app* que se encarga de generar los paquetes que posteriormente el protocolo LOADng decidirá por donde enviar si ya tiene ruta existente o comenzará el proceso de búsqueda de ruta. Los mensajes aquí se crean según los parámetros configurados de destino e instante de creación.

EL submódulo *routing* representa la capa donde se encuentra el protocolo LOADng, los mensajes pueden llegar aquí desde el submódulo antes comentado o desde otro router LOADng. Este módulo se encarga de tratar el mensaje según el tipo que sea y su destino siguiendo la lógica del protocolo. Sobre este módulo irán apareciendo un listado de las rutas

encontradas y que se actualiza de forma dinámica según lo que vaya sucediendo en la simulación.

Los submódulos llamados *queue* son los que conectan con los otros routers y representan una cola de entrada y salida por cada conexión exterior que tenga el router. Son colas FIFO y se crean automáticamente según las conexiones que tenga el router.

7.2 Módulo controlador de eventos



Ilustración 8 Modulo controlador de eventos

El módulo *scenarioManger* se encarga de manejar los posibles eventos que sucederán con la configuración de la simulación como son: aparición/desaparición de un router, aparición/desaparición de una conexión, modificaciones en las características de los router o de las conexiones etc...

Este módulo no está formado por otros submódulos en su interior, únicamente utiliza un script que leerá los eventos de un documento .xml y los ejecutará en los momentos pertinentes.

Todas las topologías deben contar con este módulo si se desean ejecutar eventos.

8 Análisis del protocolo LOADng en redes de media tensión

Para la realización del análisis del protocolo se han seleccionado una serie de topologías que han permitido en primer lugar la verificación y debuggeo de la herramienta creada y la comparación tanto entre distintas configuraciones de los routers como el estudio de la adaptación de las redes ante distintos eventos.

8.1 Punto de partida

Una vez desarrollada la herramienta que permite realizar las simulaciones del protocolo en la red de media se tensión, se ha procedido a comprobar su correcto funcionamiento con topologías simples que permitan verificar el software desarrollado.

Se han utilizado varias topologías comenzando por dos nodos simples hasta varios nodos con redundancia de conexiones para comprobar que realmente el protocolo encontraba el camino con el criterio seleccionado.

Una vez comprobado el funcionamiento del protocolo con los diferentes criterios, se le han añadido eventos que le obligasen a recalcular las rutas y así poder ver cómo y cuánto tiempo tarda en reaccionar la red a ellos.

8.2 Redes de referencia

Tras comprobar el funcionamiento del software se han seleccionado tres grandes grupos de topologías con mismo número de nodos entre sí, de tal

forma que se puedan comparar y observar con ello el efecto que tienen distintos niveles de redundancia en las conexiones.

8.2.1 Topologías de 10 nodos

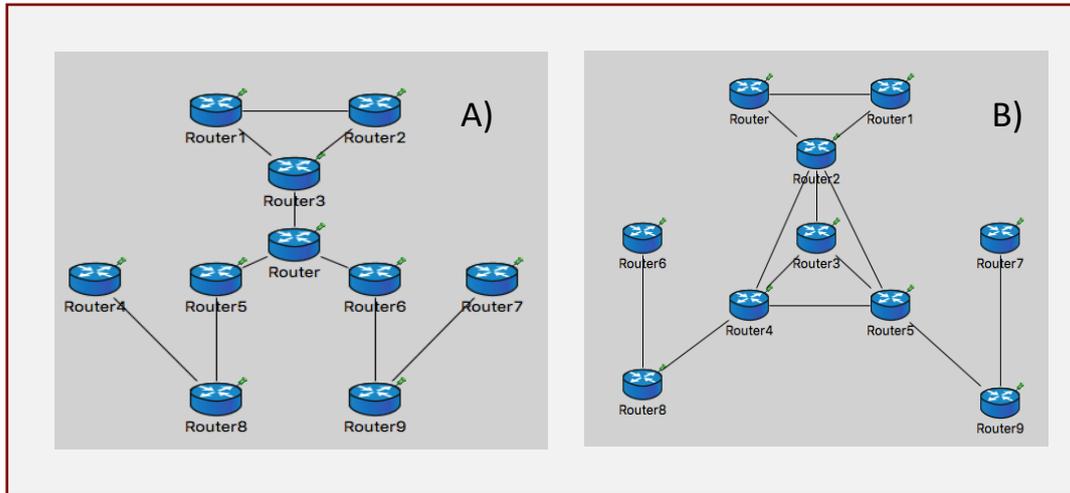


Ilustración 9 Topologías de 10 nodos

Se trata de unas topologías formadas por 10 nodos. Para el caso de la topología A), representa una conexión con centro conectado en estrella, y para la topología B) representa una conexión con centro conectado en estrella y triángulo similar a la A) pero con mayor redundancia de enlaces entre los routers.

8.2.2 Topologías de 17 nodos

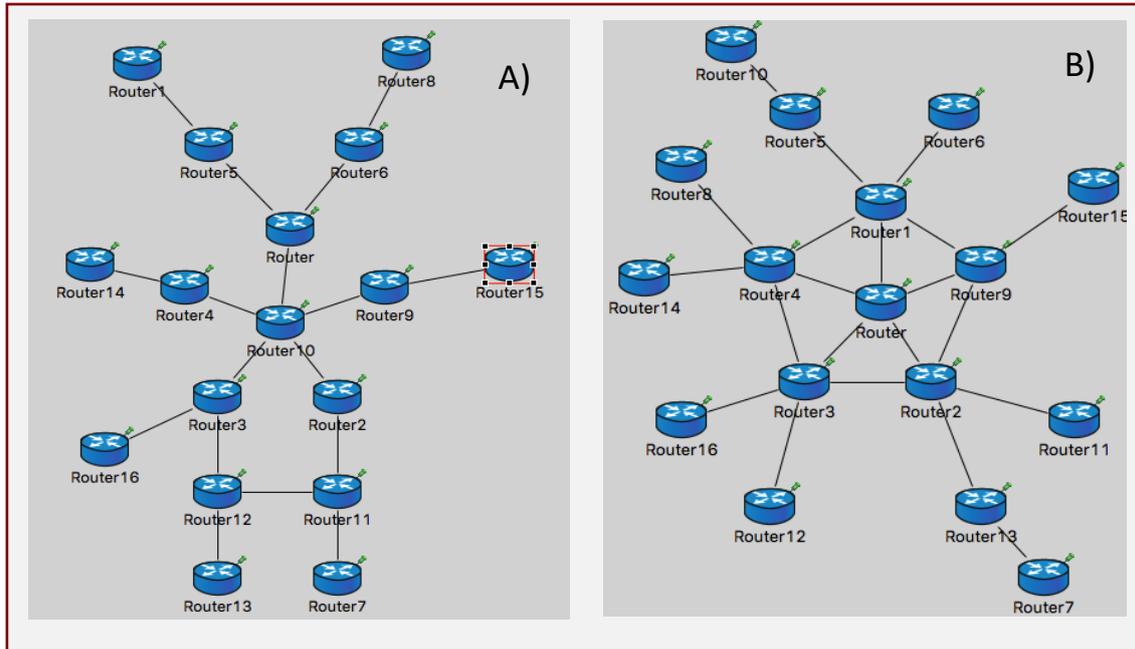


Ilustración 10 Topologías de 17 nodos

Topologías formadas por 17 nodos, al igual que para el caso 10 nodos, la topología A) se ha realizado con un centro conectado en estrella y para la topología B) el centro se encuentra conectado en estrella y triángulo con mayor redundancia de enlaces.

8.2.3 Topologías reales de 30 nodos

Para la realización de estas topologías se ha utilizado un documento publicado por el Joint Research Center llamado “*DISTRIBUTION SYSTEM OPERATORS OBSERVATORY*” en que utiliza métodos estadísticos para representar la distribución de los transformadores en distintas áreas. Utiliza parámetros como la densidad poblacional y demanda para así poder aproximar la localización de los transformadores en localizaciones urbanas,

semiurbanas y rurales. Obteniendo los siguientes planos y datos de su composición (cada color representa un feeder distinto):

- Topología Urbana



Ilustración 11 Distribución transformadores urbanos

Feeders parameters

	LV network [km]	MV network [km]	MV/LV Substations
MV Feeder #1	8.00	4.97	15
MV Feeder #2	11.49	5.86	25
MV Feeder #3	11.81	6.21	26
MV Feeder #4	11.01	5.34	31
MV Feeder #5	11.57	6.98	29

Formada por 126 subestaciones agrupadas en 5 feeders con semejante tamaño, 5 conexiones redundantes entre los feeders y cercanía entre las subestaciones.

- Topología Semiurbana

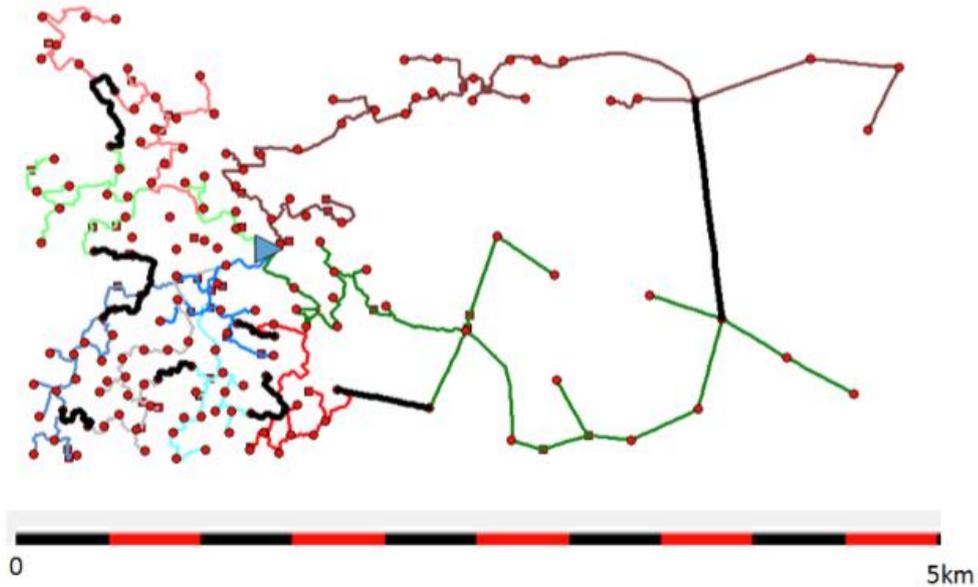


Ilustración 12 Distribución transformadores semiurbanos

	LV network [km]	MV network [km]	MV/LV Substations
MV Feeder #1	11.58	4.35	16
MV Feeder #2	12.39	4.94	16
MV Feeder #3	10.17	3.43	16
MV Feeder #4	11.81	4.49	16
MV Feeder #5	9.33	3.31	15
MV Feeder #6	9.51	2.50	10
MV Feeder #7	9.59	3.95	12
MV Feeder #8	19.94	8.59	30
MV Feeder #9	14.19	8.18	21
MV Feeder #10	6.45	1.97	9

Formada por 161 subestaciones agrupadas en 10 feeders de distintos tamaños en función de si es la parte rural de la red o la urbana. Se encuentran conexiones cercanas en la zona rural y más distanciadas en la zona rural. Contiene también 8 conexiones redundantes entre feeders.

- Topología Rural

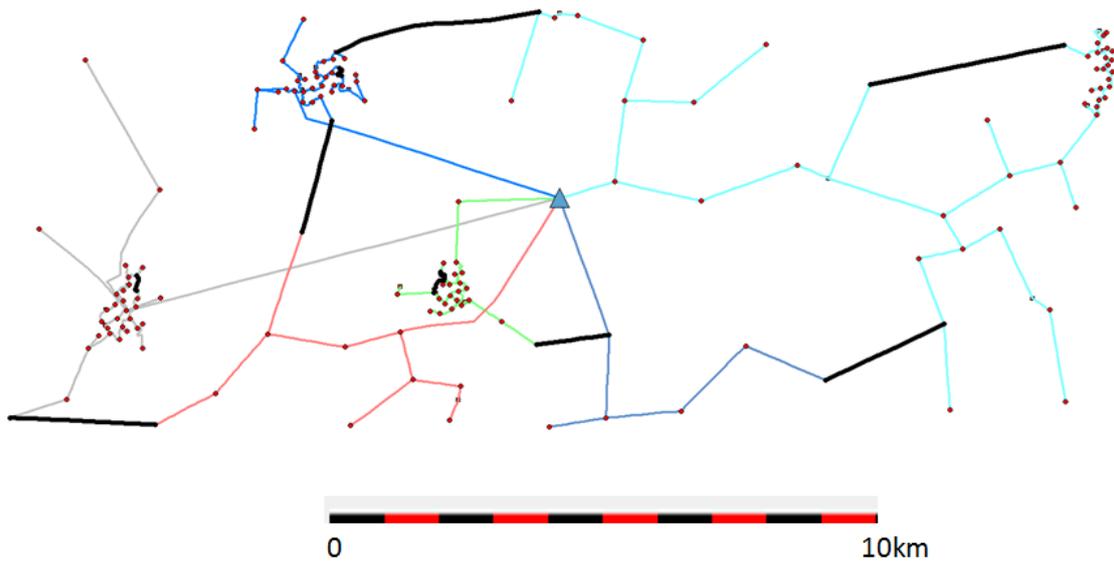


Ilustración 13 Distribución transformadores rurales

	LV Network [km]	MV Network [km]	MV/LV Substations
MV Feeder #1	11.54	9.95	6
MV Feeder #2	21.61	15.21	10
MV Feeder #3	43.94	24.93	33
MV Feeder #4	26.26	10.00	25
MV Feeder #5	67.51	38.09	47
MV Feeder #6	40.73	14.95	31

Formada por 152 subestaciones en 6 feeders con distintos tamaños dependiendo de su cercanía a poblaciones. En este caso consta de 6 conexiones redundantes entre feeders.

Para asemejar las redes entre sí y poder hacer simulaciones que resultasen útiles para comparar se han limitado a 30 el número de subestaciones para los 3 casos, pero eso sí, manteniendo la proporcionalidad entre los tamaños de los feeders y las conexiones redundantes entre ellos.

Obtenemos entonces las siguientes topologías que serán las simuladas con el protocolo:

- Topología Urbana

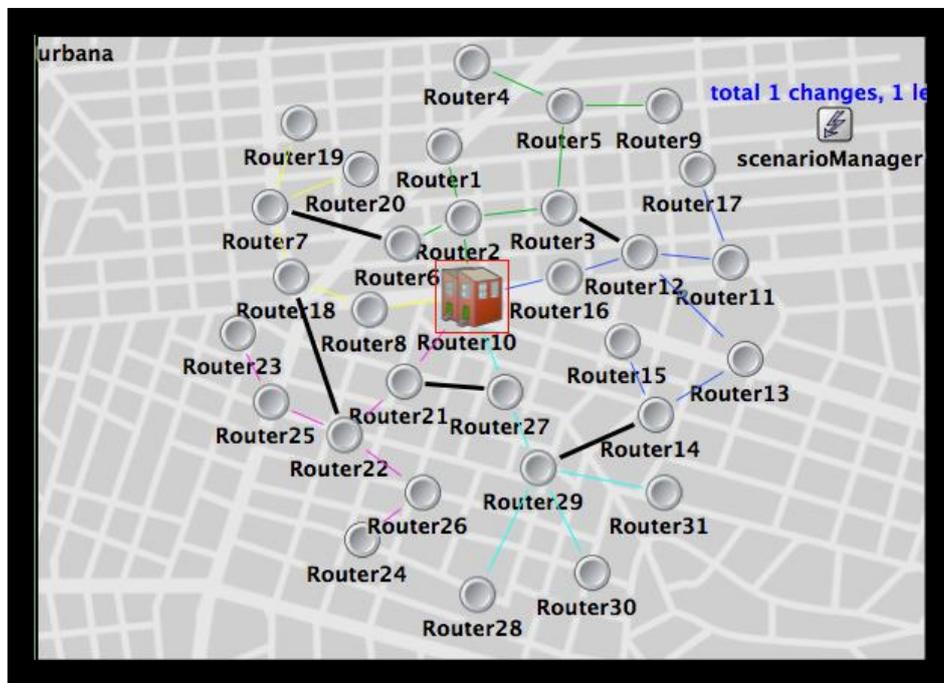


Ilustración 14 Topología urbana

- Topología Semiurbana

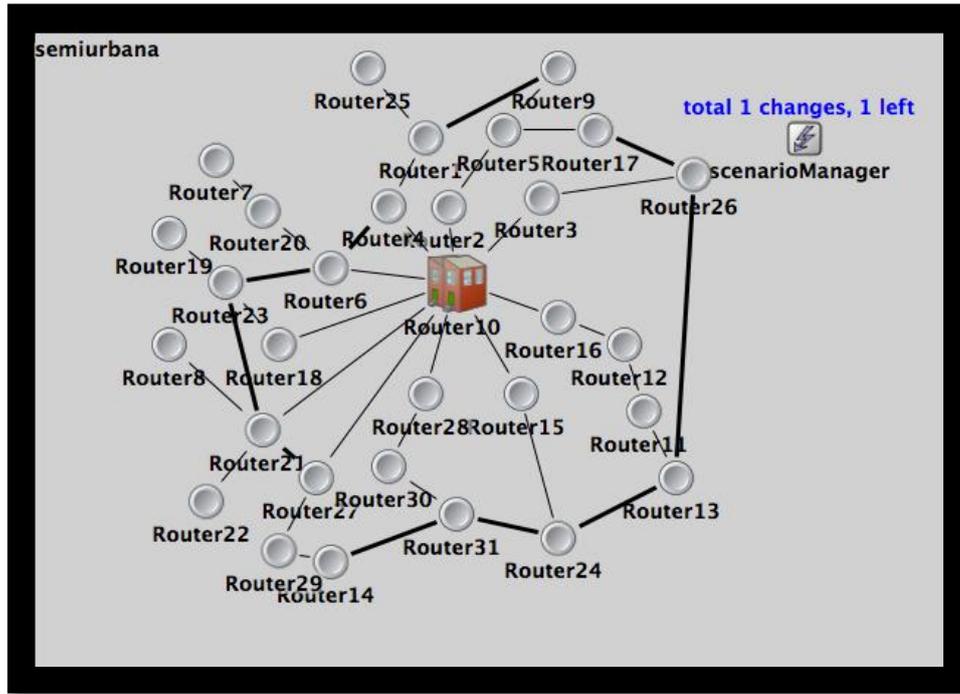


Ilustración 15 Topología semiurbana

- Topología Rural

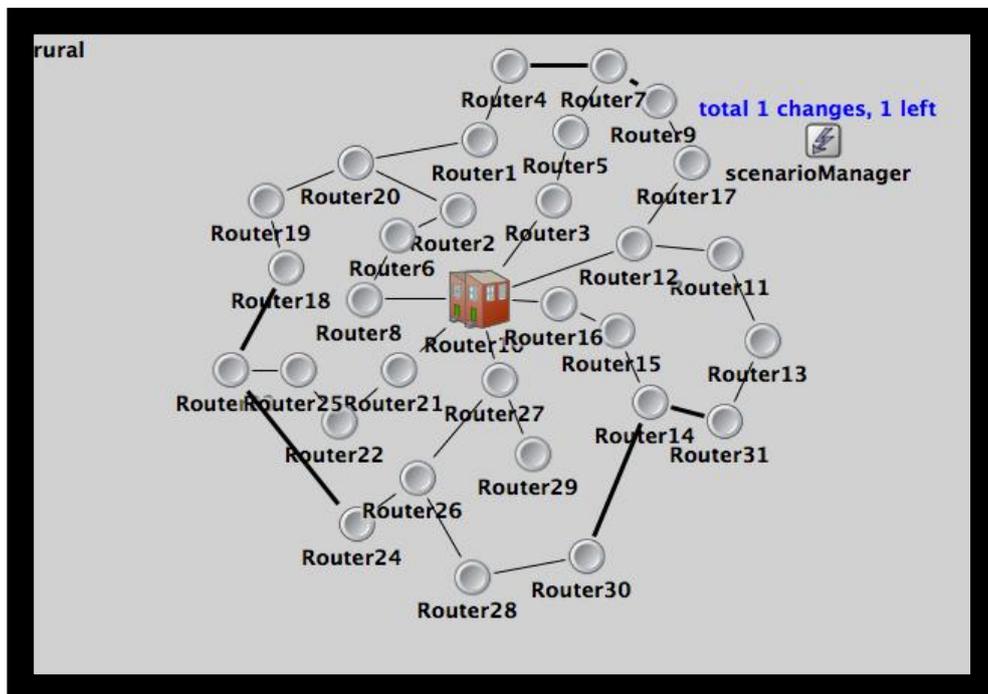


Ilustración 16 Topología rural



8.3 Consideraciones de estudio

Para que todas las simulaciones resulten útiles y puedan ser usadas de forma comparativa se han definido una serie de consideraciones con las que realizar el estudio del protocolo LOADng en todas ellas:

- 1) En cada simulación el router que origina el mensaje que se desea enviar se encuentra a 3 saltos de distancia del concentrador central y su destino se sitúa también a 3 saltos de distancia del concentrador. De esta manera se intentará observar cómo influye el método de conexión alrededor del centro de la topología.
- 2) Inicialmente todos los enlaces tienen un mismo delay predefinido de 1ms excepto uno, hay un enlace con un retardo de 10ms en la ruta principal de tal forma que si el criterio de ruta óptima es el menor número de saltos, no influirá en la ruta. Pero si el criterio es el de menor delay, el protocolo buscará la ruta evitando ese enlace con más retardo.
- 3) Inicialmente no hay ningún enlace marcado como débil excepto uno, de tal forma que si el criterio de ruta óptima es el menor número de saltos, no influirá en la ruta. Pero si el criterio es el de evitar enlaces débiles, el protocolo buscará la ruta evitando ese enlace marcado como débil.

Una vez se ha producido la convergencia de la red, es decir, se ha encontrado la ruta, se producirá un evento que forzará la búsqueda de una nueva ruta óptima. En cada topología y con cada criterio de selección de ruta óptima se producirá un corte de un enlace adyacente al nodo que

busca la ruta para así asegurarnos de que se va a buscar una nueva ruta que evite dicho corte.

Se realizan entonces 35 simulaciones, que serían las 7 topologías con los 3 distintos tipos de configuración de los enlaces y con los 3 criterios de selección de ruta óptima. En primera instancia se produce la convergencia de la red y tras esto, se produce una desconexión de un enlace de la ruta principal y se espera a que converja de nuevo la red, obteniendo así el tiempo de reacción del protocolo a un evento que obliga a buscar una nueva ruta.

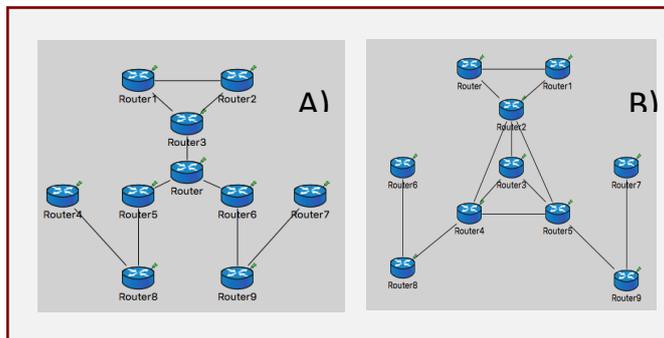
Topologías		Enlaces	Criterios
10 nodos	10 a	<ul style="list-style-type: none"> • Iguales • Delay • Débiles 	<ul style="list-style-type: none"> • Menor N° de saltos • Menor delay • Evitar enlaces débiles
	10 b		
17 nodos	17 a		
	17 b		
30 nodos	Urbana		
	Semirural		
	Rural		

Tabla 3 Resumen simulaciones

8.4 Resultados

Los resultados aquí agrupados muestran en milisegundos el tiempo que tarda la red en converger según los 3 criterios de búsqueda de ruta óptima y según las 3 configuraciones de enlaces.

8.4.1 Topologías de 10 nodos



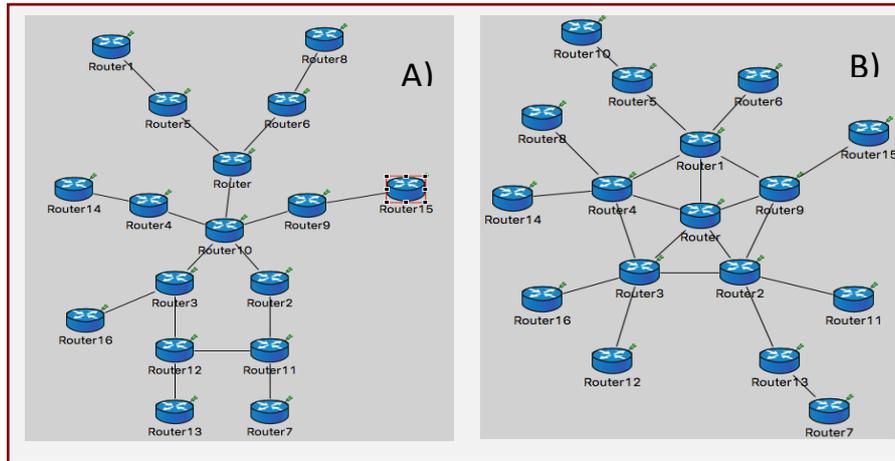
Búsqueda inicial					
		Enlaces			
		Criterio	Iguales	Delay	Débiles
Topología A	Nº de saltos		0,01	0,02	0,01
	Menor delay			0,012	
	Enlaces débiles				0,012
Topología B	Nº de saltos		0,008	0,026	0,008
	Menor delay			0,01	
	Enlaces débiles				0,01

Tabla 4 Búsqueda inicial 10 nodos

Búsqueda tras desconexión					
		Enlaces			
		Criterio	Iguales	Delay	Débiles
Topología A	Nº de saltos		0,012	0,012	0,012
	Menor delay			0,012	
	Enlaces débiles				0,012
Topología B	Nº de saltos		0,01	0,028	0,01
	Menor delay			0,012	
	Enlaces débiles				0,012

Tabla 5 Búsqueda tras desconexión 10 nodos

8.4.2 Topologías de 17 nodos



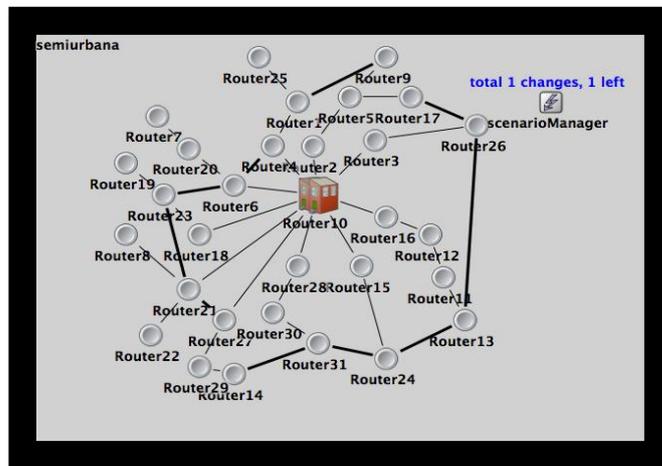
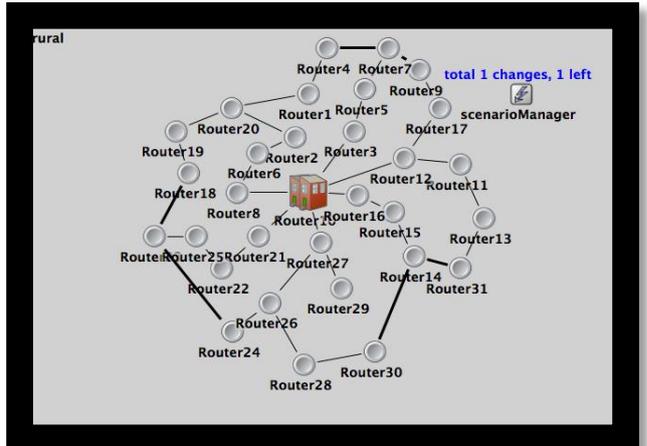
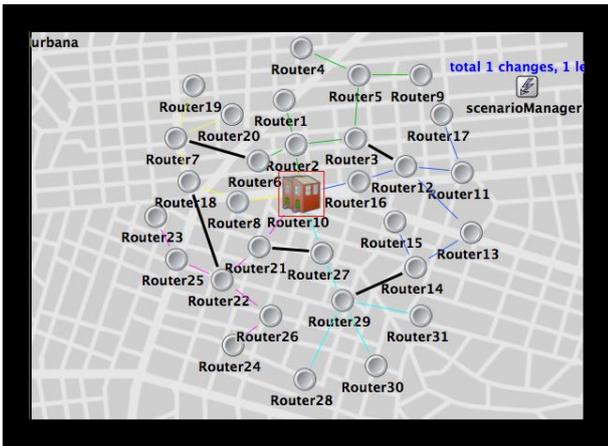
Búsqueda inicial					
		Enlaces			
		Criterio	Iguales	Delay	Débiles
Topología A	Nº de saltos		0,012	0,029	0,012
	Menor delay			0,014	
	Enlaces débiles				0,014
Topología B	Nº de saltos		0,012	0,029	0,012
	Menor delay			0,014	
	Enlaces débiles				0,014

Tabla 6 Búsqueda inicial 17 nodos

Búsqueda tras la desconexión					
		Enlaces			
		Criterio	Iguales	Delay	Débiles
Topología A	Nº de saltos		0,013	0,029	0,013
	Menor delay			0,015	
	Enlaces débiles				0,015
Topología B	Nº de saltos		0,013	0,029	0,013
	Menor delay			0,015	
	Enlaces débiles				0,015

Tabla 7 Búsqueda tras la desconexión 17 nodos

8.4.3 Topologías reales de 30 nodos



Búsqueda inicial				
		Enlaces		
Criterio		Iguals	Delay	Débiles
Urbana	Nº de saltos	0,012	0,029	0,012
	Menor delay		0,016	
	Enlaces débiles			0,012
SemiUrbana	Nº de saltos	0,012	0,029	0,012
	Menor delay		0,016	
	Enlaces débiles			0,012
Rural	Nº de saltos	0,012	0,029	0,012
	Menor delay		0,018	
	Enlaces débiles			0,018

Tabla 8 Búsqueda inicial 30 nodos

Búsqueda tras desconexión					
		Enlaces			
		Criterio	Iguales	Delay	Débiles
Urbana	Nº de saltos		0,013	0,029	0,013
	Menor delay			*	
	Enlaces débiles				0,013
SemiUrbana	Nº de saltos		0,013	0,029	0,013
	Menor delay			*	
	Enlaces débiles				0,013
Rural	Nº de saltos		0,018	0,018	0,012
	Menor delay			0,018	
	Enlaces débiles				0,018

Tabla 9 Búsqueda tras la desconexión 30 nodos

8.5 Situación real 2 feeders

Como caso particular, se ha realizado la simulación de una situación real en la que un transformador está conectado a dos feeders de la siguiente manera:

Conexión física

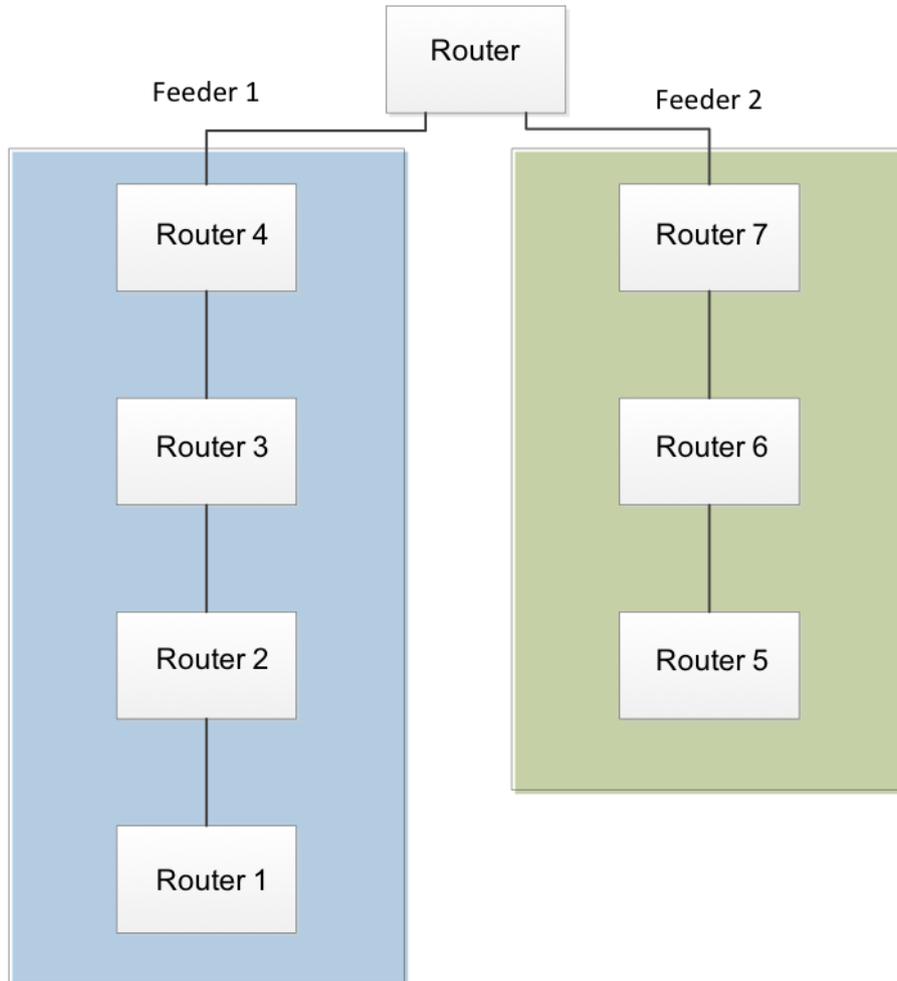


Ilustración 17 Conexión física 2 feeders

Al encontrarse los transformadores acoplados al mismo cable, la comunicación se podría realizar entre transformadores a dos saltos de distancia, obteniendo así el siguiente esquema de comunicaciones:

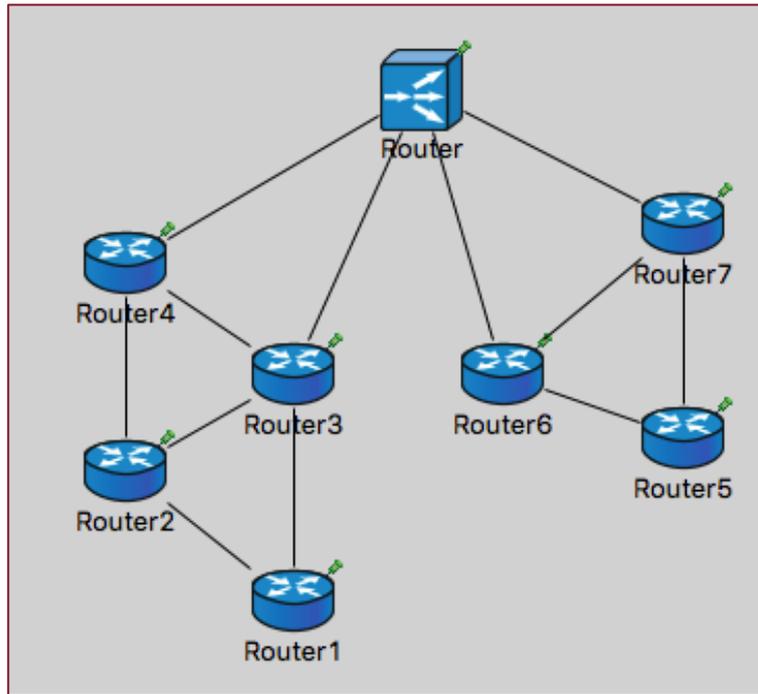


Ilustración 18 Conexión lógica 2 feeders

Este escenario se ha simulado de tal forma que todos los routers que se encuentran en los feeders se intenten comunicar con el router superior siguiendo una distribución exponencial de un segundo en la generación de paquetes y cada dos segundos se produce la conexión/desconexión del router superior con el feeder 1, es decir, la unión Router4<->Router y Router3<->Router.

Se ha obtenido el siguiente resultado:

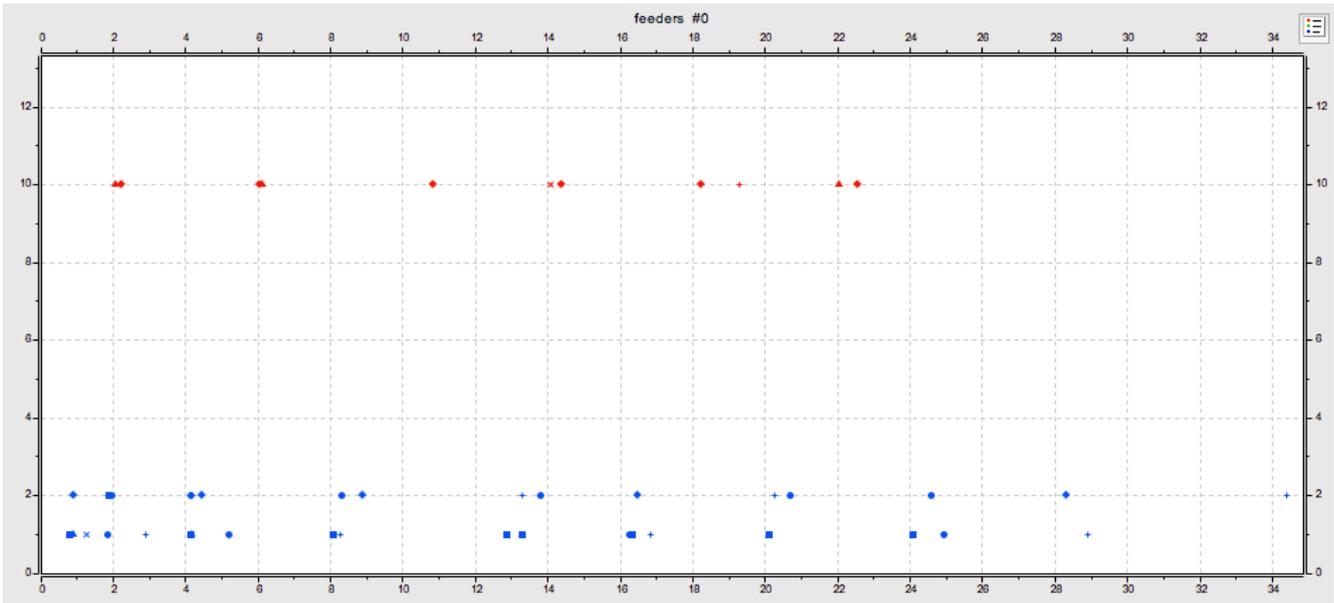


Ilustración 19 Resultado 2 feeders

Los puntos azules representan el instante en el que se produce una entrada en tabla de algún router y los puntos rojos representan el borrado de alguna entrada debido a un mensaje de error generado por el protocolo. En el eje horizontal se representa el tiempo y en el vertical el número de saltos con el que se produce la entrada en tabla.

Al iniciarse la simulación se van encontrando rutas según se producen las generaciones de paquetes en los routers. A los dos segundos de iniciarse la simulación se produce la desconexión del feeder 1, lo que provoca el borrado de las rutas que comunican el feeder 1 con el router superior, y esto sucede casi al instante de producirse la desconexión (depende de cuando se percaten los routers del feeder 1, y esto depende de la frecuencia de generación de los paquetes en los routers). En el segundo 4, se produce de nuevo la conexión del feeder 1 y vuelve a encontrar rutas para los elementos que se habían desconectado. Esto sucede cíclicamente con



particularidades como sucede con la desconexión en el segundo 8, donde por no tener demanda en una ruta, el protocolo solo observa una desconexión y casi un segundo más tarde de que se produzca.

8.6 Conclusiones

Del estudio de los datos aquí obtenidos durante las simulaciones se pueden sacar unas conclusiones generales:

- A mayor redundancia, mismo tiempo de convergencia de la red (o incluso menor): Al haber más caminos posibles para la obtención de una ruta cabría esperar que el protocolo tardase más tiempo en encontrar el camino óptimo, pero resulta no ser así. Por el funcionamiento del protocolo, una vez se encuentre la ruta más corta, el resto de rutas que se puedan encontrar serán desechadas si resultan ser peores, de esta forma, que haya más posibles caminos no ralentiza la convergencia de la red.
- Distintos retardos en las conexiones pueden crear entradas en tablas innecesarias e incluso falsas: Debido al carácter descentralizado del protocolo se ha observado en algunas simulaciones que, según el criterio de selección de ruta, los paquetes de búsqueda de ruta pueden llegar al destino final antes por una ruta no óptima. Cuando los paquetes lleguen por una ruta mejor, las rutas antes creadas quedarán obsoletas y habrá routers con direcciones que nunca serán utilizadas ya que hay otro camino mejor que será el utilizado. Los routers con las rutas no optimizadas nunca se enterarán de que esa

ruta no es óptima y serán entradas en tablas que no deberán ser utilizadas mientras que exista otra ruta mejor.

Una posible solución de esta problemática sería añadir un tiempo de espera en el router destino de tal forma que cuando le llegue un paquete que busque ese destino espere cierto tiempo para ver si le llega otro paquete por otra ruta mejor. Tras pasar el tiempo de espera, únicamente se creará la ruta por el camino óptimo evitando así la creación de las rutas no óptimas. Esto evitaría las entradas en tablas innecesarias o falsas y además contribuiría a evitar la saturación de la red con mensajes innecesarios a cambio de aumentar en cierto grado el tiempo de convergencia de la red puesto que como termino final, añadir este tiempo de espera aumentaría el tiempo que se tardaría en encontrar la ruta si el tiempo de espera resulta ser mucho mayor al tiempo que se tardaría en encontrar la ruta óptima.

9 Metodología seguida en el desarrollo del trabajo

9.1 Definición de los paquetes de trabajo y tareas

El proyecto se ha dividido en 5 fases

FASE 1: Adquisición de documentación y estudio del mismo

- **PT 1: Estudio del funcionamiento del protocolo de enrutamiento LOADng**
- **PT 2: Estudio de las particularidades en la red de media tensión**

FASE 2: Familiarización con el entorno de desarrollo

- **PT 3: Estudio del lenguaje de programación C++**
- **PT 4: Conocimiento del entorno de simulación OMNeT++**

FASE 3: Diseño de la aplicación

- **PT 5: Definición de los límites y objetivos del proyecto**

FASE 4: Desarrollo de la aplicación:

- **PT 6: Plan de trabajo para la realización de la herramienta**
- **PT 7: Definición de los parámetros para la realización de las simulaciones**
- **PT 8: Definición de la interfaz gráfica**
- **PT 9: Procesado de los datos obtenidos**

Tarea 9.1: Selección de topologías de simulación

Tarea 9.2: Verificación del funcionamiento de la herramienta

FASE 5: Pruebas de la herramienta y elaboración de un informe.

9.2 Desarrollo de los paquetes de trabajo y tareas

FASE 1: Adquisición de documentación y estudio del mismo

Paquete de trabajo 1 (PT 1):

Título: Estudio del funcionamiento del protocolo de enrutamiento LOADng		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
0	5	100 h

Para poder realizar este proyecto lo primero a realizar ha sido un estudio bibliográfico con el que aprender el funcionamiento de LOADng para su posterior implementación.

Paquete de trabajo 2 (PT 2):

Título: Estudio de las particularidades en la red de media tensión		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
4	10	120 h

Ya que este protocolo está pensado para redes inalámbricas de baja potencia y con pérdidas frecuentes de conexiones, ha sido necesario observar las diferencias entre la red para la que ha sido creado el protocolo y la red de media tensión.

FASE 2: Familiarización con el entorno de desarrollo

Paquete de trabajo 3 (PT 3):

Título: Estudio del lenguaje de programación C++		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
10	12	40 h

Recordar los conocimientos que ya poseía sobre este lenguaje de programación y conocer sus particularidades para esta aplicación.

Paquete de trabajo 4 (PT 4):

Título: Conocimiento del entorno de simulación OMNeT++		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
11	13	40 h

Familiarizarse con el programa OMNeT++ comenzando con los propios tutoriales que hay en su página web hasta ser capaz de realizar simulaciones propias con lo aprendido.

FASE 3: Diseño de la aplicación

Paquete de trabajo 5 (PT 5):

Título: Definición de los límites y objetivos del proyecto		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
13	15	40 h

Una vez conocida la herramienta de desarrollo y sus posibilidades, dejar claro cuál va a ser el alcance de la aplicación que se va a desarrollar

FASE 4: Desarrollo de la aplicación:

Paquete de trabajo 6 (PT 6):

Título: Plan de trabajo para la realización de la herramienta		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
15	16	5 h

Una vez estudiado el protocolo LOADng y conocido el entorno de desarrollo OMNeT++ se ha procedido al proceso de programación del protocolo en OMNeT++ para poder realizar las simulaciones de LOADng en la red de media tensión.

Paquete de trabajo 7 (PT 7):

Título: Definición de los parámetros para la realización de las simulaciones		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
15	16	5 h

Concretar los parámetros que se van a poder modificar para la realización de las simulaciones

Paquete de trabajo 8 (PT 8):

Título: Definición de la interfaz gráfica		
Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
15	25	200 h

Se elabora una interfaz global a la cual se le ha ido acoplado poco a poco cada una de las funcionalidades definidas anteriormente.

Paquete de trabajo 9 (PT 9):

Título: Procesado de los datos obtenidos
Tareas: Tarea 9.1: Selección de topologías de simulación Tarea 9.2: Verificación del funcionamiento de la herramienta

Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
25	38	260 h

Una vez realizada la implementación del protocolo en OMNeT++ se han creado una serie de topologías partiendo de situaciones simples e ideales y complicándolas hasta poder realizar simulaciones en topologías que se asemejen más a la realidad. Esto ha permitido el proceso de depuración de la aplicación desarrollada asegurando así su correcto funcionamiento final.

FASE 5: Pruebas de la herramienta y elaboración de un informe.

En esta última fase se han realizado pruebas de todo tipo, de forma que se puedan corregir las erratas o problemas que pudiera ir teniendo la herramienta, y se redactará un informe, para la posterior entrega en ZIV.

Fecha de inicio (semanas)	Fecha fin (semanas)	Carga de trabajo (horas)
38	40	40 h

9.3 Diagrama Gantt

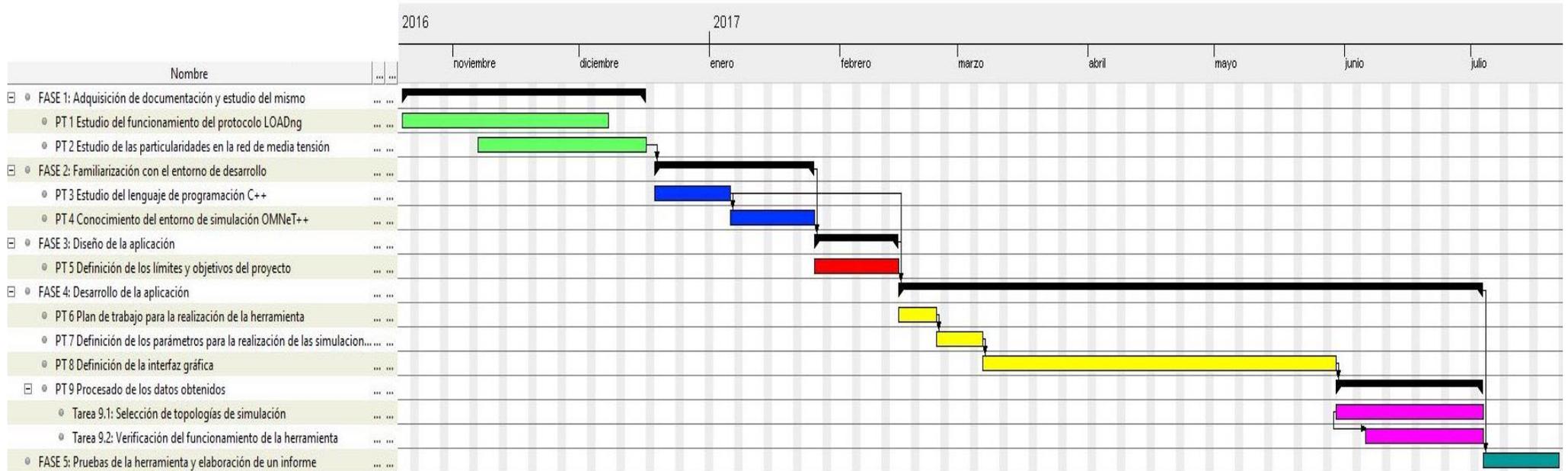


Ilustración 20 Diagrama Gantt

10 Entregables

10.1 Software

Se adjunta archivo .zip donde se encuentra la herramienta desarrollada para OMNeT++ 5.0 con el proyecto que se necesita importar. En él se incluyen también las topologías seleccionadas para las simulaciones realizadas que pueden valer como punto de partida para realizar nuevas simulaciones.

10.2 Manuales

Se adjuntan aquí dos manuales para ayudar al uso y entendimiento de la herramienta creada. Por un lado el *Manual de usuario* con explicación del uso de la herramienta orientado al usuario final junto con una guía de instalación. Por otro lado el *Manual del programador* donde se desglosa el funcionamiento de la herramienta para su posible modificación o mejora futura.

11 Bibliografía

- Bjarne Stroustrup, el lenguaje de programación c++, Addison Wesley, Madrid, ISBN 84-7829-019-2, 1998.
- Simulador de eventos discretos OMNeT++
<https://www.omnetpp.org/>
- Framework INET para OMNeT++ <https://inet.omnetpp.org/>
- Asier Llano, David Gil, José Antonio Moreno, Txetxu Arzuaga y Elsa Herbrich. (2016). Broadband OFDM PLC solution for Medium Voltage lines based on IEEE 802.15.4 and LOADng. *TENTH WORKSHOP ON POWER LINE COMMUNICATIONS*.
- Giuseppe PRETTICO, Flavia GANGALE, Anna MENGOLINI, Alexandre LUCAS, Gianluca FULLI. (2016) Distribution System Operators Observatory. JRC TECHNICAL REPORTS.
- Thomas Heide Clausen, Axel Colin de Verdière. The LLN On-demand Ad hoc Distance-vector Routing Protocol - Next Generation (LOADng). [Research Report] RR-7692, INRIA. 2011.
- Mališa Vučkinić, Bernard Tourancheau, and Andrzej Duda. (2014) Performance Comparison of the RPL and LOADng Routing Protocols in a Home Automation Scenario. University of Grenoble.

- Simulador de redes Opnet <http://www.opnet.com>