

MANETs en INET

Práctica 5 – Diseño de Redes

Instrucciones

Elabore una **memoria** explicando los resultados de simulación obtenidos al seguir los pasos indicados. Incluya todas las figuras que se piden, así como cualquier figura adicional que considere importante para explicar los resultados de simulación. Las explicaciones deben hacerse a la luz de lo explicado en la teoría de la asignatura.

- El único formato aceptado es **PDF**.
- Utilice la plantilla **LaTeX Overleaf** oficial del Trabajo Fin de Grado disponible en la Wiki FIC.
- **Referencie todas las figuras** en el texto.
- El contenido de las figuras debe ser legible **sin necesidad de hacer zoom**.
- Cuide la **redacción** y la **ortografía**. Ambas serán tenidas en cuenta.
- Utilice el **formato vectorial SVG** para almacenar las figuras siempre que sea posible. Para incluirlas en la memoria, puede convertirlas a PDF con la utilidad *rsvg-convert*, incluida en el paquete *librsvg2-bin* de Ubuntu. El siguiente comando convierte a PDF todos los archivos SVG de una carpeta:

```
$ for i in *.svg; do rsvg-convert -f pdf -o ${i%.*}.pdf $i; done
```

Escenario

El objetivo de esta práctica es la simulación de una red *mobile ad hoc* (MANET) sobre INET para analizar las diferencias existentes entre un protocolo reactivo (AODV) y uno proactivo (DSDV).

Construya dos redes (*Aodv* y *Dsdv*), ambas con dos nodos estáticos (*static1* y *static2*) separados 800 metros **horizontalmente** y con un número variable de nodos móviles (*mobile[n]*) situados entre ellos. **Todos los nodos (estáticos y móviles)** serán de tipo *AodvRouter* en la red *Aodv* y *ManetRouter* en la red *Dsdv*. Los nodos de tipo *AodvRouter* solo implementan el protocolo AODV, pero los de tipo *ManetRouter* pueden implementar diferentes tipos de enrutamiento y por lo tanto es necesario configurarlos con `**.routing.typeName = "Dsdv"`.



Los nodos móviles en ambas redes seguirán un tipo de movilidad `MassMobility` con un `angleDelta` y un `changeInterval` aleatorios y **tendrán una posición inicial aleatoria** en el escenario (para que no salgan todos del mismo punto utilice `**mobile[*].mobility.initFromDisplayString = false`). Asigne una velocidad de 1m/s a los nodos móviles.

Para facilitar el seguimiento de la simulación asigne la IP 8.1.1.1/8 a *static1*, 8.2.2.2/8 a *static2*, y 8.3.3.n/8 a cada nodo *mobile[n]*. Para esto último es necesario asignar manualmente la IP 8.3.3.0 al nodo *mobile[0]* antes de asignar 8.3.3.x al resto de nodos *mobile[*]*.

Para que las simulaciones AODV y DSDV sean comparables los nodos deben moverse siguiendo las mismas rutas en los dos casos. Esto puede conseguirse utilizando un generador de números aleatorios diferente para la movilidad con:

```
num-rngs = 2
**.mobility.rng-0 = 1
```

El nodo *static1* tendrá una `UdpBasicApp` y el nodo *static2* una `UdpEchoApp`. Configure el nodo *static1* para que envíe un datagrama cada 200 milisegundos hacia *static2*, con `startTime = 10s`. Configure el *visualizer* para que muestre las IPs de los nodos (`displayInterfaceTables = true`, `interfaceFilter = "wlan"` y la ruta que siguen los paquetes UDP (`displayRoutes = true`, `packetFilter = "UdpBasicAppData"`).

Elija una semilla para el generador aleatorio de movilidad (`seed-1-mt = k`, siendo **k** de nuevo la salida de ejecutar `echo $RANDOM` en un terminal) de manera que exista **al menos una ruta posible** entre *static1* y *static2* cuando comience a transmitir la `UdpBasicApp` (es decir, el solape de las coberturas de los nodos móviles en el instante $t = 10s$ **debe permitir la comunicación** desde *static1* a *static2* a través de varios nodos).

Incluya el contenido completo de los ficheros .ned y .ini en la memoria y responda a las siguientes cuestiones:

1. Simule el escenario con AODV con **entre 15 y 20** nodos móviles y una potencia de 1.5mW. ¿En qué momento se realiza la primera transmisión (de cualquier tipo de paquete)? Muestre el **contenido** de ese paquete explicando los campos más importantes.
2. ¿Llega **el primer** RREQ enviado por *static1* hasta *static2* a través de alguna de las rutas existentes? ¿Por qué? ¿En qué instante recibe *static2* su primer RREQ?
3. Elija un nodo de la zona intermedia (i.e., fuera del alcance directo de *static1* y de *static2*) por el que vayan a pasar los paquetes UDP tras el establecimiento de la ruta. Muestre la **tabla de enrutamiento** justo antes y justo después de recibir el primer RREQ. Explique las entradas que se crean. ¿Para qué ruta son útiles esas entradas?
4. Muestre la **tabla de enrutamiento** de ese mismo nodo justo antes y justo después de recibir el RREP de respuesta. Explique las entradas creadas por el RREP. ¿Para qué ruta son útiles estas entradas?
5. ¿Qué ruta sigue el RREP hasta llegar a *static1*? ¿Qué IP destino utilizan los RREP? ¿En qué se diferencia del RREQ en cuanto a propagación? Muestre una **captura del escenario** en la que se vea la ruta seguida por `UdpBasicAppData-0` mostrada por el *visualizer*.

6. ¿En qué instante recibe el paquete *UdpBasicAppData-0* el nodo *static2*? ¿Cuánto tiempo transcurre desde el *startTime*? ¿A qué se debe el retardo?
7. Añada las siguientes líneas a la configuración para simular la caída de un nodo en *t=13s*:

```
**.hasStatus = true
*.scenarioManager.script = xml("<scenario>\n
<at t='13'><shutdown module='mobile[k]'/></at></scenario>")
```

Donde *k* es el número de nodo de la ruta activa en ese instante inmediatamente anterior a *static2*. ¿Cómo se da cuenta de la caída el nodo anterior? ¿Cuánto tarda en notificar al resto de nodos? Muestre una captura del **log** del nodo que genera el RERR y el **contenido del paquete** explicando los campos importantes.
8. ¿Qué IP tiene como destino el RERR? ¿Qué TTL tiene? ¿Es reenviado por todos los nodos que lo reciben? ¿Cómo se propaga por la red?
9. ¿Qué hace un nodo al recibir un RERR? Muestre capturas de la **tabla de enrutamiento** antes y después de recibirlo y explique en qué cambia. ¿Qué hace *static1* en los instantes posteriores a recibir el RERR?
10. Simule ahora el escenario con DSDV con el mismo número de nodos móviles y la misma potencia que en el apartado 1. ¿En qué instante se realiza ahora la primera transmisión (de cualquier tipo de paquete)? ¿Qué diferencia hay con AODV? Muestre el **contenido del paquete** explicando los campos que considere relevantes.
11. Muestre algún ejemplo de paquete *Hello* con valor de *hopdistance* **mayor que 1** y explique el significado de los campos *srcAddress* y *nextAddress* **utilizando para explicarlos una captura de la tabla de enrutamiento del nodo que lo genera**.
12. Explique la actualización que se produce en la **tabla de enrutamiento** del nodo que lo recibe relacionándola con el contenido del paquete mostrado en la cuestión anterior.
13. ¿En qué instante llega ahora el paquete *UdpBasicAppData-0* a *static2*? ¿Llega antes o después que en AODV? ¿Por qué?
14. Vuelva a simular la caída del último nodo de la ruta hacia *static2* en *t=13s* (es posible que el nodo sea diferente al de la cuestión 7). ¿Se notifica de alguna manera al resto de nodos? ¿Cómo se repara la ruta en este caso?
15. Simule AODV durante 500s con 15 nodos móviles a una velocidad de 3m/s y con una potencia de 2mW para todos los nodos (sin la caída del nodo de los apartados 7 y 14). ¿Qué porcentaje de paquetes llega desde *static1* a *static2*? ¿Qué porcentaje llega desde *static2* a *static1*? Explique los valores.
16. Haga lo mismo con DSDV. Explique las diferencias con respecto a AODV en porcentajes de ida y de vuelta. ¿Qué diferencia hay en cantidad de tráfico con AODV? (Para medir el tráfico utilice los valores *fwd*, *up* mostrados en el módulo *ipv4* de un nodo móvil al azar haciendo doble click sobre él en el.)
17. Repita la simulación de DSDV modificando el parámetro *helloInterval* a 10s y a 2s (el valor por defecto es 5s). ¿Cómo afecta cada valor al porcentaje de paquetes perdidos? ¿Por qué? ¿Cómo afecta a la cantidad de tráfico?