

**EVALUACIÓN DE ALGORITMOS DE ENRUTAMIENTO EN REDES AD HOC Y
SU INFLUENCIA EN EL CONSUMO DE ENERGÍA USANDO QUALNET®**

TESIS DE MAESTRIA PRESENTADA POR:

ING. MARIO DUQUE HOYOS

ASESOR

NÉSTOR MISAEL PEÑA TRANSLAVIÑA. Ph.D



UNIVERSIDAD DE LOS ANDES
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
PROGRAMA MAGÍSTER
BOGOTÁ
AGOSTO DE 2005

Agradezco

Muy especialmente a mi tío Antonio y a su familia por su enorme colaboración en momentos difíciles de mi vida.

A mi esposa por su paciencia y apoyo incondicional.

Y a mi hija Mariana por ser la inspiración para este trabajo.

A mi asesor por sus consejos, su paciencia y el tiempo dedicado.

Para todos muchas gracias.

CONTENIDO

INTRODUCCIÓN.....	II
OBJETIVOS.....	V
1 ALGORITMOS DE ENRUTAMIENTO.....	1
1.1 Dynamic Source Routing Protocol DSR [13].....	1
1.2 Ad hoc On Demand Distance Vector AODV . [12].....	2
1.3 Fisheye State Routing Protocol FSR [14].....	4
2 MODELO DE CONSUMO DE ENERGÍA.....	5
2.1 Modelo de Qualnet® 3.8[21].....	5
2.2 Modelo implementado	6
2.3 Limitaciones del modelo.....	7
2.4 Validación del modelo.....	7
3 SIMULACIONES.....	10
3.1 Escenario de Simulación	10
3.2 Métricas Evaluadas.....	10
3.3 Consumo vs. Tráfico Ofrecido.....	11
3.4 Consumo vs. Radio	16
3.5 Consumo vs. Movilidad.....	22
4 CONCLUSIONES Y RECOMENDACIONES.....	27
5 REFERENCIAS BIBLIOGRÁFICAS	29
Anexo 1. Modelo de consumo de energía en modo PSP	32
Anexo 2. Archivo de configuración del escenario.....	39

LISTA DE FIGURAS

Figura 1. Escenario de Validación	8
Figura 2. Relación de Paquetes Entregados vs. Tráfico Ofrecido.....	14
Figura 3. Retardo Promedio vs. Tráfico Ofrecido	15
Figura 4. Encabezado de Enrutamiento Normalizado vs. Tráfico Ofrecido	16
Figura 5 Consumo de Energía PSP vs. Tráfico Ofrecido	17
Figura 6. Consumo de Energía Normalizado vs. Tráfico Ofrecido.....	18
Figura 7. Relación de Paquetes Entregados vs. Radio Alcance.....	19
Figura 8. Retardo Promedio vs. Radio Alcance	20
Figura 9 Encabezado de Enrutamiento vs. Radio Alcance.....	21
Figura 10 Consumo de Energía en PSP vs. Radio Alcance	21
Figura 11. Consumo de energía Normalizado vs. Radio Alcance.....	22
Figura 12. Relación de Paquetes Entregados vs. Movilidad	23
Figura 13. Retardo Promedio vs. Movilidad.....	24
Figura 14. Encabezado de Enrutamiento vs. Movilidad.....	24
Figura 15 Consumo de Energía en PSP vs. Movilidad.....	25
Figura 16. Consumo de Energía Normalizado vs. Movilidad.....	26

INTRODUCCIÓN

Una red inalámbrica Ad Hoc es una colección de nodos o terminales autónomos que se comunican con otros formando una red de radio de múltiples enlaces y se mantiene una conectividad de una manera descentralizada. Como los nodos se comunican por enlaces inalámbricos ellos deben manejar los efectos de las radio comunicaciones como ruido, atenuación y radio interferencia[1].

Las redes Ad Hoc son motivo de creciente estudio por sus aplicaciones militares y comerciales. El diseño de protocolos de enrutamiento es una de las áreas de mayor investigación dentro de las redes Ad Hoc y aunque hay consenso de la importancia del consumo de energía en el diseño de protocolos, se ha realizado poco trabajo en la caracterización del consumo de los algoritmos conocidos[2].

En general se ha caracterizado el desempeño energético con la eficiencia en el uso del ancho de banda, pero las métricas de consumo de energía y tráfico de control son substancialmente diferentes[2].

Varios algoritmos de enrutamiento se han propuesto para reducir el consumo de energía en redes Ad hoc, una de las principales estrategias es utilizar una métrica de potencia para guiar al algoritmo a encontrar rutas que presenten menor consumo como en [3] y [4]. En [5] además de utilizar como métrica la potencia de transmisión, el algoritmo de enrutamiento distribuye el tráfico a través de los nodos cuyas baterías tengan mayor

expectativa de vida para maximizar el tiempo de operación de la red. Pero solamente en [6], utilizaron para evaluar el desempeño energético una métrica de consumo de energía por paquete transmitido.

Otros algoritmos orientados a un uso eficiente de la energía en la red utilizan estrategias para reducir el tráfico de control y los encabezados como en [7], donde apoyados en la información geográfica de la red, reportan un 25% de reducción de los encabezados, pero no presenta métricas de consumo de energía.

Por lo descrito, se evidencia la importancia de realizar un análisis del desempeño de los algoritmos de enrutamiento e incluir una métrica que permita establecer el consumo de energía y su eficiencia.

Otros trabajos en la caracterización del consumo de energía en redes Ad Hoc son:

Ephremides[7] presenta un análisis conceptual de factores que pueden influir en el desempeño energético en redes inalámbricas, incluidas las redes Ad Hoc e identifica como punto crítico estrategias que se puedan incorporar en la capa de acceso al medio.

Goldsmisth[8] presenta un análisis de los múltiples elementos a considerar en el diseño de redes inalámbricas en las que existen restricciones en el consumo de energía. Además, plantea consideraciones sobre la colaboración de los algoritmos de enrutamiento en la coordinación del acceso al medio y la posibilidad de ahorrar hasta un 40% de la energía con la utilización del estado de suspensión (SLEEP).

Cunha[10] analiza de forma teórica algunas técnicas de ahorro de energía en redes Ad Hoc, por otro lado Carvalho[11], estudió el consumo del protocolo IEEE 802.11, para una red Ad Hoc de un solo salto y en estado de saturación, introdujo el concepto de eficiencia de consumo en [J/b], y establece una relación entre ésta y el tamaño de la red.

En Margi [11], se describe la implementación de un modelo de consumo para Glomosim, el modelo está desarrollado en la capa física y permite medir los ahorros de energía obtenidos por estrategias implementadas en cualquier capa superior. Además argumenta la necesidad de un modelo detallado del consumo de energía para aprovechar las ventajas de simulación versus el estudio experimental.

En el presente trabajo se analiza la influencia de diferentes algoritmos de enrutamiento en el consumo de energía en la red y la sensibilidad de estos algoritmos con diferentes escenarios de carga, radio alcance y movilidad. A continuación en este documento se hace una breve descripción de los algoritmos de enrutamiento estudiados, en la sección II, se presentan el modelo de consumo disponible en Qualnet® y la descripción del modelo implementado en este trabajo. En la sección III, se describen el escenario de simulación y los resultados obtenidos en la comparación del desempeño de los algoritmos variando la carga ofrecida a la red, el radio alcance y la movilidad de los nodos. Por último en la sección IV se exponen las principales conclusiones de este estudio.

OBJETIVOS

- Evaluar el desempeño de algoritmos de enrutamiento en Redes Ad Hoc.
- Analizar la influencia del algoritmo de enrutamiento en el consumo de energía.
- Evaluar el desempeño energético bajo diferentes parámetros.

1 ALGORITMOS DE ENRUTAMIENTO

En este estudio se analizaron los algoritmos de enrutamiento AODV (Ad hoc On Demand Distance Vector), DSR (Dynamic Source Routing Protocol) y FSR (Fisheye State Routing Protocol) por ser ampliamente estudiados [16]-[19] lo cual posibilita confrontar los resultados obtenidos con los reportados en la literatura técnica. A continuación se presenta una breve descripción de las características principales de cada uno de ellos.

1.1 Dynamic Source Routing Protocol DSR [13]

DSR es un protocolo de enrutamiento reactivo, solo determina rutas en demanda, se compone de dos mecanismos, descubrimiento y mantenimiento de ruta. No posee ninguna actividad periódica, ni descarga esta responsabilidad en otras capas, por tal motivo los mensajes de control pueden bajar considerablemente cuando los nodos son estacionarios unos a otros y las rutas de comunicación han sido descubiertas. En presencia de movilidad son actualizadas las rutas que están en uso, todos los otros cambios son ignorados.

DSR dispone de un mecanismo de recepción en modo promiscuo que utiliza intensamente para el aprendizaje de rutas, aunque este mecanismo no es indispensable para el protocolo, con él puede aprender y almacenar rutas sin aumentar el tráfico de control. La utilización del modo promiscuo, exige al dispositivo un trabajo adicional de procesamiento y un mayor consumo de energía, en redes donde la capa de acceso al medio permite

pasar a estado de reposo, cuando en el medio se están cursando comunicaciones, que no están dirigidas, al nodo, en el caso de la MAC 802.11, configurada para trabajar en redes Ad Hoc, el modo promiscuo es modo de trabajo por defecto, por lo que esto no representa un aumento en el consumo de energía debido al algoritmo de enrutamiento.

CONFIGURACIÓN DE PARAMETROS DSR	Valor
DSR-BUFFER-MAX-PACKET	50
DSR-BUFFER-MAX-BYTE	0

Tabla I Configuración de Parámetros de DSR

Una de sus principales características de DSR es el enrutamiento desde la fuente, en el cual cada paquete lleva la ruta completa de fuente a destino, este es el modo básico de operación, aunque puede ser deshabilitado. Este mecanismo ofrece una carga adicional por el incremento de los encabezados en los paquetes de datos, esta sobrecarga es justificable para mantener libres de ciclos las rutas utilizadas, y soportar otras optimizaciones como balance de carga y aprendizaje de rutas en modo promiscuo.

La parametrización de DSR utilizada en este trabajo se presenta en la Tabla I

1.2 *Ad hoc On Demand Distance Vector AODV. [12]*

La operación de este protocolo es muy similar a DSR, pero a diferencia de este utiliza números de secuencia para mantener libres de ciclos las rutas seleccionadas, evitando problemas de “conteo al infinito” asociados a los

clásicos protocolos de vector de distancia. El número de secuencia al destino lo asigna cada destino para cada ruta solicitada, un nodo con dos rutas a un destino seleccionará la que tenga el número de secuencia mayor.

Para solicitar una nueva ruta un nodo difunde de manera controlada un RREQ. La ruta al destino es determinada, cuando el paquete RREQ alcanza el nodo destino, o a un nodo intermedio con una “ruta fresca” al destino. Una ruta es considerada “fresca”, cuando el número de secuencia es al menos mayor que el número asociado al requerimiento.

La ruta se hace disponible, cuando se envía de regreso un RREP, al nodo origen, cada nodo intermedio que recibe el RREP, va almacenando la ruta al destino en su tabla de enrutamiento.

CONFIGURACIÓN DE PARAMETROS AODV	Valor
AODV-NET-DIAMETER	35
AODV-NODE-TRAVERSAL-TIME	40MS
AODV-ACTIVE-ROUTE-TIMEOUT	3S
AODV-MY-ROUTE-TIMEOUT	6S
AODV-HELLO-INTERVAL	1S
AODV-ALLOWED-HELLO-LOSS	2
AODV-RREQ-RETRIES	2
AODV-ROUTE-DELETION-CONSTANT	5
AODV-PROCESS-HELLO	NO
AODV-LOCAL-REPAIR	NO
AODV-SEARCH-BETTER-ROUTE	NO
AODV-BUFFER-MAX-PACKET	100
AODV-BUFFER-MAX-BYTE	0
AODV-OPEN-BI-DIRECTIONAL-CONNECTION	YES
AODV-TTL-START	1
AODV-TTL-INCREMENT	2
AODV-TTL-THRESHOLD	7

Tabla II Configuración de Parámetros de AODV

Cada nodo verifica el estado del enlace al siguiente salto en la ruta, cuando un nodo determina que un enlace ha fallado, envía un RERR.

La parametrización de AODV utilizada en este trabajo se presenta en la Tabla II

1.3 *Fisheye State Routing Protocol FSR [14]*

Este protocolo de enrutamiento es proactivo, con algunas optimizaciones para mantener en cierto nivel los paquetes de control utilizando una técnica conocida como “ojo de pescado”, en la que se captura con gran detalle los puntos cerca al foco y se reduce el detalle a medida que se aleja del punto focal.

Un nodo almacena el estado de los enlaces en la red, luego periódicamente difunde actualizaciones a sus vecinos, con una frecuencia que depende de la distancia al destino. Entre mayor es la distancia menor es la frecuencia de actualización. Con esta información cada nodo construye una topología de red y calcula rutas eficientes a cada destino.

FSR no envía mensajes de control cuando un enlace falla, esto es adecuado en ambientes donde el cambio de topología es alto, en cambio espera hasta el siguiente periodo de actualización para difundir la información respecto de los cambios, esto significa que FSR no utiliza ninguna actualización iniciada por evento, con esta estrategia busca limitar el uso de los mensajes de control y mantenerlos en bajos niveles aun en casos de alto cambio.

CONFIGURACIÓN DE PARAMETROS FSR	Valor
FISHEYE-SCOPE	2
FISHEYE-INTRA-UPDATE-INTERVAL	5S
FISHEYE-INTER-UPDATE-INTERVAL	15S
FISHEYE-NEIGHBOR-TIMEOUT-INTERVAL	15S

Tabla III Configuración de Parámetros de FSR

Adicionalmente a estas estrategias la información a propagar es seleccionada acorde a la regla que su propagación debería causar una actualización de la topología de sus vecinos, excluyendo toda información redundante para sus vecinos y logrando una reducción en los mensajes a difundir.

La parametrización de FSR utilizada en este trabajo se presenta en la Tabla III

2 MODELO DE CONSUMO DE ENERGÍA

Los modelos de consumo de energía son abstracciones que reflejan de manera simplificada el consumo de los dispositivos físicos según los diferentes estados de operación por los que atraviesan.

2.1 Modelo de Qualnet® 3.8[21]

El modelo de consumo de energía implementado en Qualnet® 3.8 es un modelo de dos estados, donde se contabiliza la energía consumida en transmisión e inactividad.

El consumo en transmisión se modela utilizando dos componentes, una constante que representa el consumo de energía de los elementos físicos

(hardware) y una componente variable correspondiente a la cantidad de energía empleada en amplificar la señal.

Adicionalmente el modelo de Qualnet® se basa en dos supuestos, el primero que el consumo del hardware en transmisión es igual al consumo en recepción y el segundo que el consumo en estado de inactividad es igual al consumo del dispositivo físico en recepción argumentando este supuesto en que las redes Ad Hoc los nodos están constantemente censando el medio.

2.2 *Modelo implementado*

Cada fabricante de tarjetas tiene un algoritmo propietario para mejorar el consumo de sus dispositivos según las variables de operación, estas mejoras pueden incluir transiciones entre estados dependiendo del tráfico en la red, o incluir paquetes de datos pequeños, junto con los paquetes de control CTS o ACK.[2]

Se distinguen tres modos de operación configurables en los dispositivos físicos. El primero de ellos es CAM (Constantly Awake Mode), donde la interfaz física siempre se encuentra en estado de transmisión o en estado de recepción, presenta el más alto consumo. Este modo de operación corresponde a las características del modelo de consumo implementado actualmente en Qualnet® 3.8.

El modo Fast PSP (Power Save Mode), permite transiciones rápidas entre transmisión, recepción e inactividad es el modo de operación sugerido para las redes Ad Hoc[22]

Y el modo de operación Max PSP Mode (Max Power Savings) donde la interfaz física además de los estados de transmisión, recepción e inactivo, puede pasar a un estado de suspensión, en el que consume muy poca energía, pero no puede recibir mensajes destinados a ella.

El modelo implementado es una extensión del provisto en Qualnet® 3.8 agregando el estado de recepción y corresponde al modo de operación Fast PSP.

Adicionalmente se hicieron las modificaciones necesarias para que Qualnet® recibiera los valores de consumo como parámetros de entrada de la red a simular y poderlos modificar según las especificaciones de los fabricantes. Esta opción no se encontraba disponible en el modelo de Qualnet® y los valores de consumo eran constantes independientemente del dispositivo físico a simular.

2.3 Limitaciones del modelo

En este estudio se analizó el comportamiento de los algoritmos desde el punto de vista de eficiencia energética y no se consideró la supervivencia de la red, por lo que los nodos en el modelo cuentan con una cantidad de energía no agotable durante el tiempo de simulación, lo que no es preciso para otro tipo de estudios.

2.4 Validación del modelo

Para la validación del modelo implementado, se siguió un procedimiento similar al reportado en[11], en el que se compara un ejercicio analítico con

los resultados obtenidos por simulación. En la Figura 1 se muestra el escenario de 5 nodos ubicados en los vértices de un cuadrado de 311m y se ajustó el radio alcance a 250m. El nodo 1 envía un paquete de datos al nodo 5 y el nodo 2 envía un paquete al nodo 4.

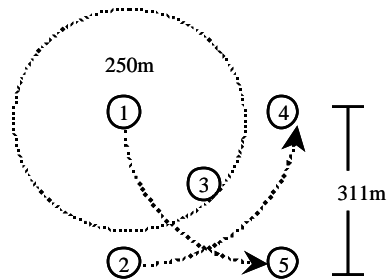


Figura 1. Escenario de Validación

El consumo de energía está dado por la cantidad de tiempo en cada estado el cual es función del tamaño del paquete y la tasa de datos empleada. En la Tabla IV se presentan los paquetes que transmite y recibe cada nodo y en la Tabla V se presentan los cálculos utilizando una tasa de 2 Mbps y un paquete de datos de 380 bytes junto con los resultados obtenidos con el simulador para 10 réplicas del experimento.

Nodo	TX	RX
1	RTS+DATA	$2*(RTS+DATA)+2*(CTS+ACK)$
2	RTS+DATA	$2*(RTS+DATA)+2*(CTS+ACK)$
3	$2*(RTS+DATA)+$ $2*(CTS+ACK)$	$2*(RTS+DATA)+$ $2*(CTS+ACK)$
4	CTS+ACK	$2*(RTS+DATA)+2*(CTS+ACK)$
5	CTS+ACK	$2*(RTS+DATA)+2*(CTS+ACK)$

Tabla IV. Paquetes Transmitidos y Recibidos.

Nodo	Analítico [ms]		Obtenidos [ms]	
	TX	RX	TX	RX
1	2.21	5.41	2.21	5.69
2	2.21	5.41	2.21	5.54
3	5.41	5.41	5.41	5.41
4	0.50	5.41	0.50	5.61
5	0.50	5.41	0.50	5.49

Tabla V. Resultados del Modelo vs. Teórico

La diferencia entre los valores reportados para el tiempo en recepción, se debe a que la cercanía de los nodos hace que de manera aleatoria un nodo cambie el estado de inactividad a recepción erróneamente. Esto se verificó al correr las simulaciones y comprobar que el efecto sucede solo para la recepción y de manera aleatoria entre los cuatro nodos en los vértices. Este comportamiento es normal en la capa física donde una señal débil puede hacer cambiar de estado la unidad aunque una transmisión entre ellas sea imposible.[10]

3 SIMULACIONES

3.1 *Escenario de Simulación*

El escenario de simulación es una red con 50 nodos, distribuidos aleatoriamente en un área de 1500 x 300, en la capa física se utilizó el modelo "Two Ray". La frecuencia de operación fue 2,4 GHz, y la capa de acceso al medio MAC 802.11b en DCF. La potencia de transmisión se ajustó para obtener un radio de alcance de 250 m, a la tasa de transmisión de 2 Mbps. Este escenario es similar al empleado en [17].

Los valores de consumo se tomaron de los obtenidos en [23], para hacer los resultados comparables con otros estudios, estos valores fueron 850 [mW] para el estado de inactividad, 970 [mW] para recepción y 1330 [mW] para transmisión.

El tiempo de simulación fue de 900s, los experimentos se realizaron con 10 semillas¹, los resultados entregados tienen un intervalo de confianza del 95%, según procedimiento descrito en [24].

3.2 *Métricas Evaluadas.*

Las métricas utilizadas se seleccionaron para reflejar de manera clara el desempeño de la red en cuanto a QoS como para reflejar el consumo de energía. Se introdujo la métrica de Consumo de Energía Normalizado para medir la eficiencia energética de la red. A continuación se describen las métricas utilizadas.

¹ Para algunos casos fue necesario realizar 5 réplicas adicionales

- **Relación de Paquetes Entregados:** Es la relación entre el número de paquetes de datos recibidos correctamente en la capa de aplicación, sobre el número de paquetes de datos enviados por los nodos fuente. Refleja la eficiencia de la red en cuando a la pérdida de paquetes.
- **Retardo Promedio:** Es el tiempo promedio empleado por los paquetes desde el nodo origen hasta el destino e incluye el tiempo en cola, transmisión y búsqueda de rutas.
- **Carga de Enrutamiento Normalizada:** Definida como la cantidad de bits de enrutamiento generados, normalizado sobre el número de bits de datos recibidos correctamente en la capa de aplicación, en [b/b]. Refleja la cantidad de bits de enrutamiento que se transmitieron por cada bit recibido correctamente en la capa de aplicación.
- **Energía Consumida:** Es la cantidad de energía consumida por todos los nodos de la red durante el tiempo de simulación, incluye la energía consumida en cada estado de operación de nodo (transmisión, recepción, e inactividad), en [KJ].
- **Energía Consumida Normalizada:** Similar a la anterior, pero normalizada por el número bits de datos recibidos correctamente por la capa de aplicación de los nodos destino, da un indicativo de la cantidad de energía consumida por bit transmitido correctamente en [mJ/b], y da un indicativo de la eficiencia energética de la red.

3.3 Consumo vs. Tráfico Ofrecido

En este grupo de simulaciones se analizó el comportamiento de los algoritmos de enrutamiento bajo diferentes niveles de carga ofrecida a la red. Adicionalmente se observó el efecto del tamaño del paquete en el desempeño y consumo de la red. Utilizar paquetes grandes se presume más eficiente que enviar varios paquetes pequeños.

El hecho de tener más tráfico para transmitir hace que los nodos permanezcan más tiempo en estado de transmisión que presenta un alto consumo respecto a los otros dos estados de recepción y reposo, lo que por obvias razones aumenta el consumo de energía.

Para el siguiente grupo de simulaciones se utilizó la red sin movilidad, por lo que las diferencias en el desempeño y el consumo de energía se pueden atribuir exclusivamente al protocolo de enrutamiento.

Se analizaron distintas cargas de tráfico, en todos los casos se utilizaron 20 sesiones de tráfico CBR que permanecen hasta el final de la simulación, las sesiones están distribuidas aleatoriamente en toda el área de simulación, y su inicio se realizó con una distribución aleatoria uniforme entre 0 y 10 segundos para disminuir la posibilidad de sincronizaciones no deseables entre diferentes sesiones de la red.

Se generó tráfico de dos formas, en la primera se varió el tamaño del paquete desde 32 bytes hasta 2020 bytes² manteniendo constante el intervalo en 4 paquetes por segundo. Los valores más altos de desempeño se lograron para un tamaño de paquete de 256 bytes, por lo

² Máximo valor permitido para no-fragmentación en la capa de red.

que se decidió tomar este valor para un segundo escenario y variar el intervalo de generación de paquetes como se muestra en la Tabla VI, en la primera columna se presenta la carga en Erlangs, en la segunda columna los valores para una tasa de 4 pq/s la variación en el tamaño del paquete y la tercera para un paquete de 256 bytes la variación en el número de paquetes por segundo.

Los resultados de desempeño y consumo de energía se muestran en las siguientes gráficas, para el mismo valor de carga los puntos obtenidos al variar el tamaño del paquete o la tasa de paquetes por segundo.

En la Figura 2 se muestra la relación de pérdida de paquetes, con baja carga los algoritmos reactivos entregan la totalidad de los paquetes, por debajo de ellos se encuentra FSR, que aun con bajas cargas tiene que realizar un trabajo considerable para calcular la topología de la red. Cerca de 0.1 Erlang la red comienza a saturarse y el desempeño de los tres algoritmos decae, pero menos el algoritmo proactivo FSR.

Bajo cargas moderadas se aprecia mejor desempeño de la red, transportando paquetes de mayor tamaño, por el ahorro en tráfico de control que esto representa, para bajas cargas la diferencia no es apreciable.

2 Mbps	4 Pq/s	256 Bytes
Erlangs	Bytes	# Pq/S
0.010	32	0.5
0.020	64	1
0.041	128	2
0.082	256	4
0.164	512	8
0.328	1024	16
0.646	2020	32

Tabla VI. Tráfico Ofrecido.

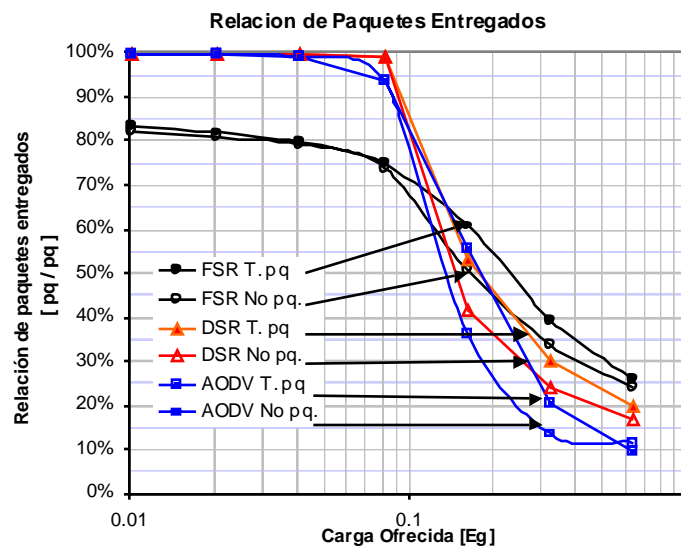


Figura 2. Relación de Paquetes Entregados vs. Tráfico Ofrecido

El comportamiento del retardo promedio es presentado en la Figura 3, en escala logarítmica para apreciar mejor el crecimiento del retardo después del punto de saturación, en especial para los algoritmos reactivos que es de dos órdenes de magnitud. Aunque FSR presenta mayor retardo para cargas bajas, su crecimiento es moderado en función de la carga debido a la independencia del tráfico de control y el tráfico de datos. Para bajas cargas FSR presenta menor retardo con paquetes pequeños, esta diferencia no es apreciable para los algoritmos reactivos.

En la Figura 4 se presenta el encabezado de enrutamiento normalizado, se puede apreciar que para los algoritmos reactivos hay un crecimiento mayor del encabezado comparado con el tráfico entregado a medida que la carga ofrecida aumenta. Se observa que para el algoritmo FSR, la proporción de tráfico transportado y el tráfico de control disminuyen, por que este solo depende del número de vecinos y de la frecuencia de actualización y no de la tasa de paquetes por segundo de las sesiones en la red.

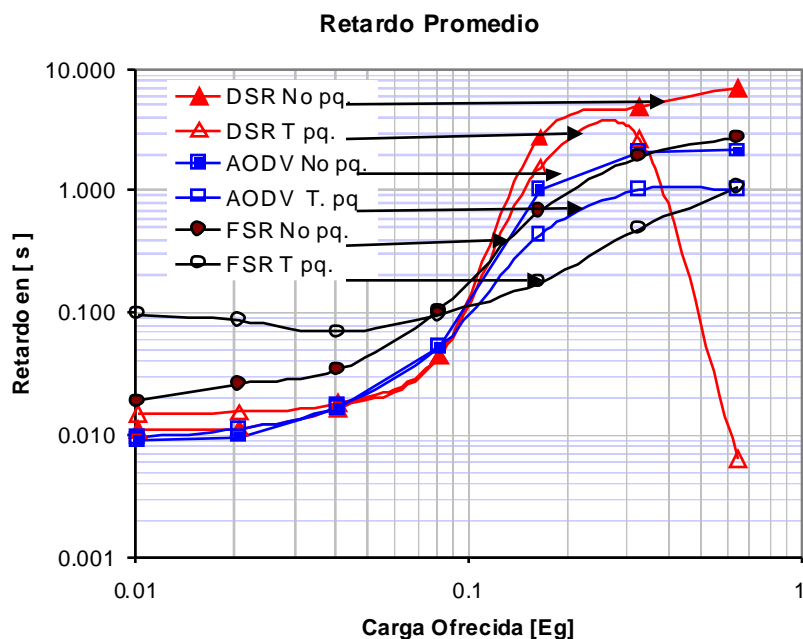


Figura 3. Retardo Promedio vs. Tráfico Ofrecido

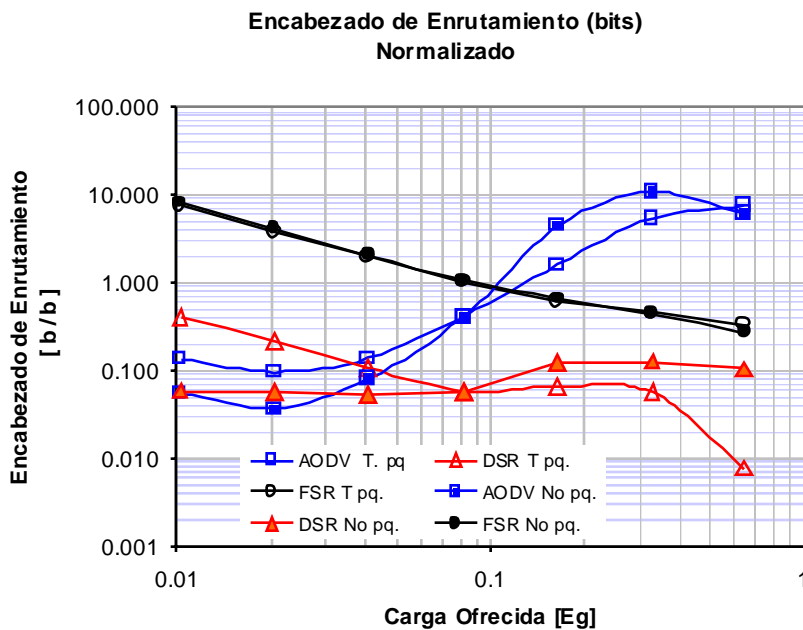


Figura 4. Encabezado de Enrutamiento Normalizado vs. Tráfico Ofrecido

En la Figura 5, el consumo de energía presenta un crecimiento en función del tráfico. Cerca al punto de saturación el consumo de los tres algoritmos es muy similar.

En la Figura 6, se presenta el consumo normalizado, se puede concluir que para bajas cargas, la eficiencia en el consumo de energía es baja, debido al alto consumo en estado de inactividad, a medida que la carga ofrecida aumenta el costo de transmitir un bit disminuye, por lo que es más eficiente energéticamente la red operando cerca al punto de saturación.

Luego del punto de saturación, los algoritmos reactivos generan gran cantidad de tráfico de control y la eficiencia para ellos disminuye nuevamente, mientras que para el algoritmo proactivo mejora.

3.4 Consumo vs. Radio

Al aumentar el radio transmisión se eleva la potencia de transmisión, con un consecuente aumento del consumo de energía en el dispositivo. Dado que la potencia se atenúa, con el cuadrado de la distancia e incluso con valores más altos, se considera factible transmitir de manera más económica en cuanto al consumo de energía a través de varios enlaces cortos que pocos de enlaces largos.

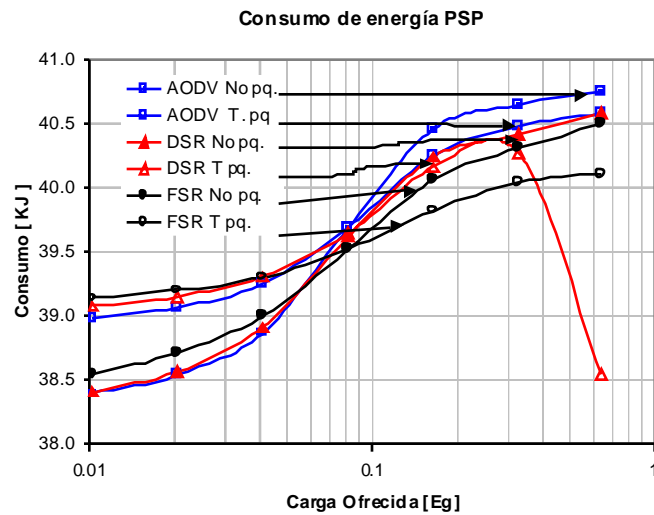


Figura 5 Consumo de Energía PSP vs. Tráfico Ofrecido

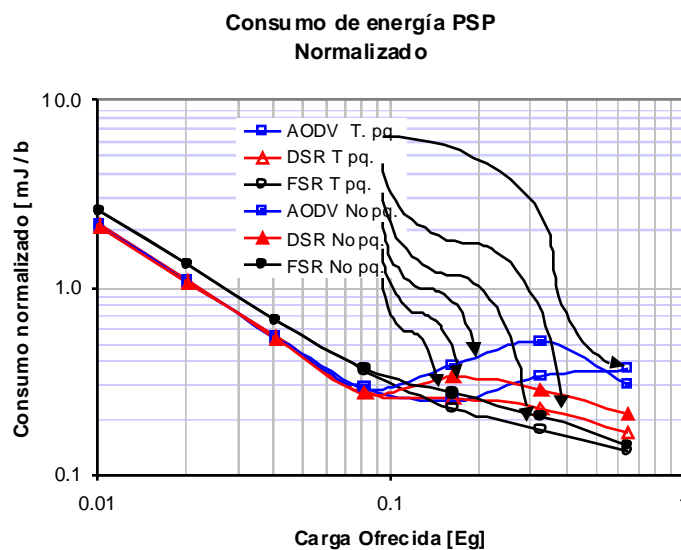


Figura 6. Consumo de Energía Normalizado vs. Tráfico Ofrecido

Manteniendo constantes todos los otros parámetros de simulación, con tráfico del tipo *constant-bit-rate* (CBR), 20 sesiones, con tiempos de inicio distribuidos uniformemente entre 0 y 10 segundos. Se utilizó una tasa de 4 paquetes por segundo y el tamaño del paquete de 256 bytes y para un escenario sin movilidad, se varió la potencia de transmisión para 2 Mbps, se obtuvieron los siguientes radios de transmisión como se muestra en la Tabla VII.

Potencia de Transmisión a 2 Mbps	Radio [m]
10dBm	254
12dBm	285
15dBm	339
18dBm	402
20dBm	452

Tabla VII. Radio Alcance

Los resultados de desempeño y consumo de energía se presentan en las siguiente gráficas.

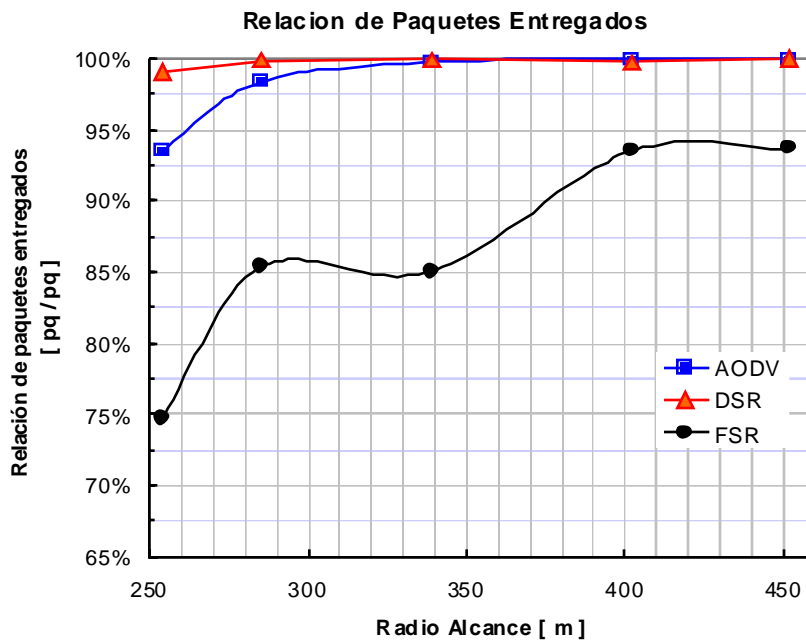


Figura 7. Relación de Paquetes Entregados vs. Radio Alcance

En la Figura 7 se presenta la relación de paquetes entregados y en la y Figura 8 el retardo promedio, se observa mejor desempeño aumentando el rango de transmisión, al necesitar menos saltos, la probabilidad de enlaces rotos y el retardo disminuyen.

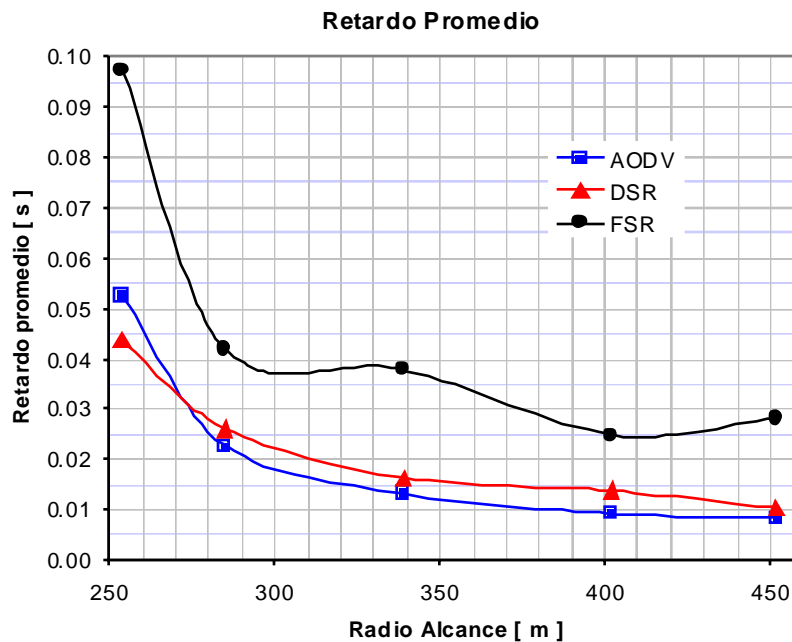


Figura 8. Retardo Promedio vs. Radio Alcance

En la Figura 9 Encabezado de enrutamiento normalizado, se observa un aumento en el tráfico de control para FSR, por el aumento en el número de vecinos a los que debe enviar actualizaciones, para DSR el tráfico de control permanece estable, mientras que para AODV se observa una disminución de éste.

En la Figura 10, se presenta el consumo en el modo de operación PSP, se observa un aumento en el consumo al aumentar el radio alcance como de esperarse. En la Figura 11 se aprecia el consumo normalizado en donde la eficiencia energética de los algoritmos de enrutamiento AODV y DSR, no es sensible a las variaciones del radio alcance, para el caso de FSR, se observa una pequeña mejoría al aumentar el radio. Esto muestra que el aumento en consumo se ve compensado en un mejor desempeño del algoritmo de manera más notoria para FSR.

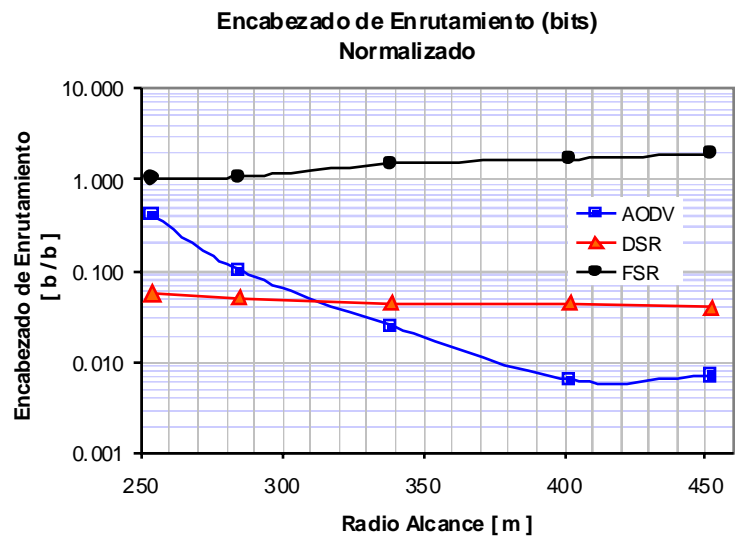


Figura 9 Encabezado de Enrutamiento vs. Radio Alcance

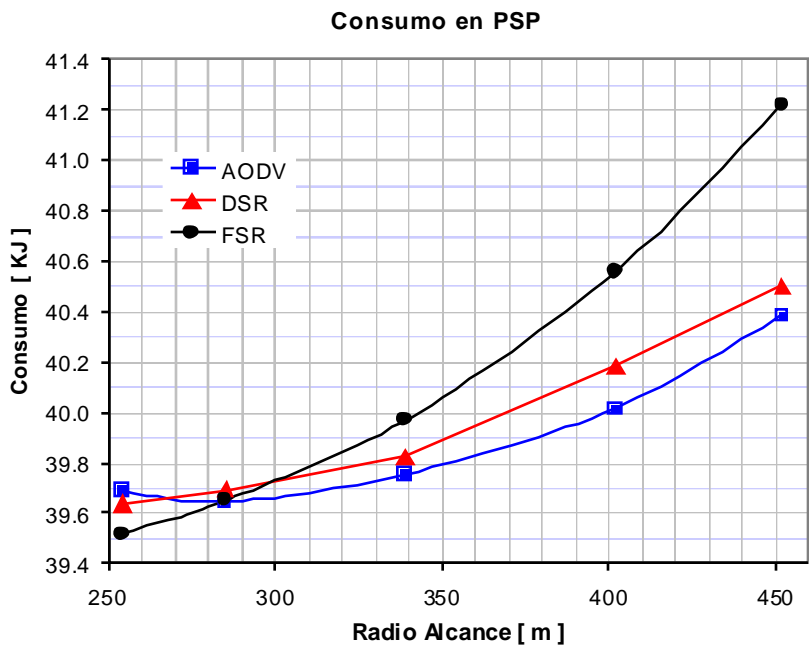


Figura 10 Consumo de Energía en PSP vs. Radio Alcance

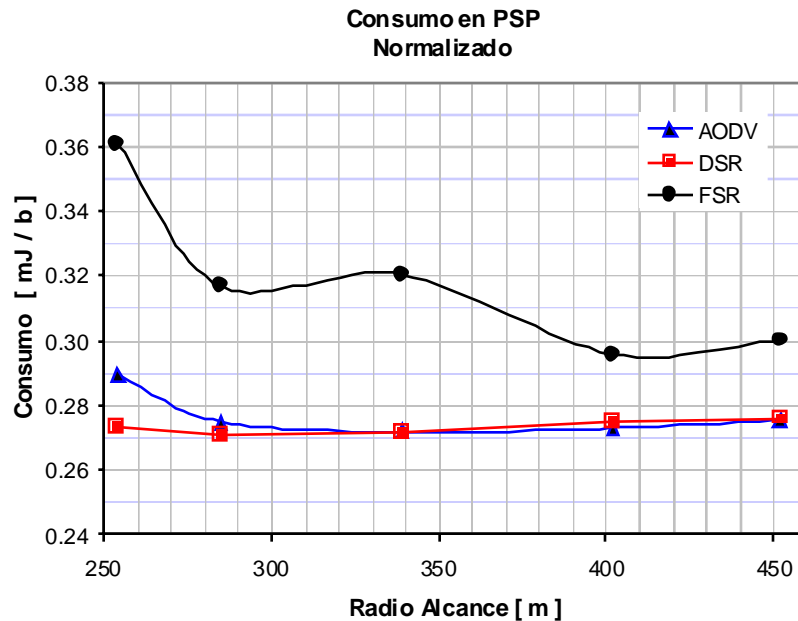


Figura 11. Consumo de energía Normalizado vs. Radio Alcance

3.5 Consumo vs. Movilidad

Para analizar el consumo en función de la movilidad se empleó el modelo de movilidad “Random Waypoint”, con las modificaciones sugeridas en [25], con una velocidad mínima de 1m/s y tiempos de pausa cero segundos.

La velocidad máxima se varió con los valores de 5, 10 y 20 m/s, lo que representa escenarios de media, alta y muy alta movilidad y se compararon contra escenarios sin movilidad.

El tráfico utilizado fue del tipo *constant-bit-rate* (CBR), 20 sesiones con tiempos de inicio distribuidos uniformemente entre 0 y 10 segundos. La tasa de envío de paquetes es de 4 paquetes por segundo y el tamaño del paquete es de 256 bytes.

La potencia de transmisión se ajustó para un alcance de 250m y se

mantuvieron constantes los otros parámetros de simulación.

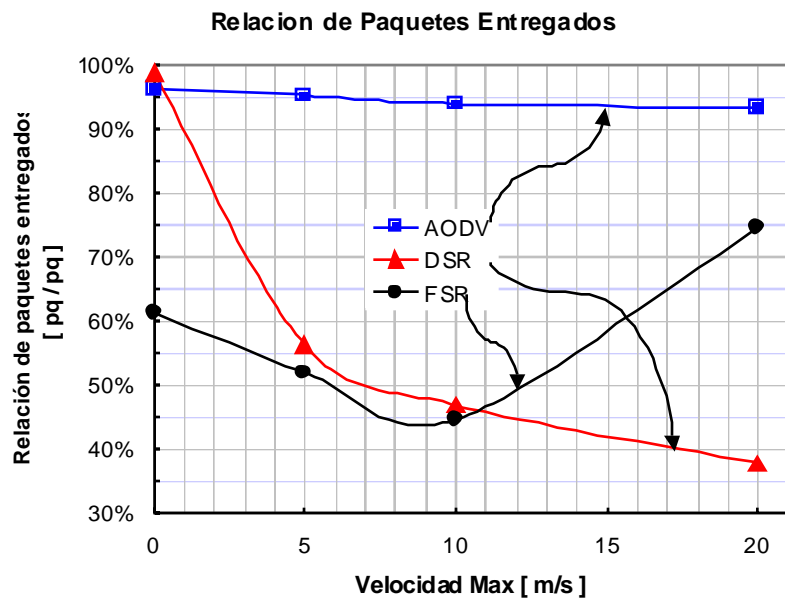


Figura 12. Relación de Paquetes Entregados vs. Movilidad

En la Figura 12 y Figura 13 se presentan, la relación de paquetes entregados y el retardo promedio, para los algoritmos AODV, DSR y FSR en función de la movilidad. Se puede apreciar un buen desempeño del algoritmo AODV y un bajo desempeño de DSR en todos los escenarios.

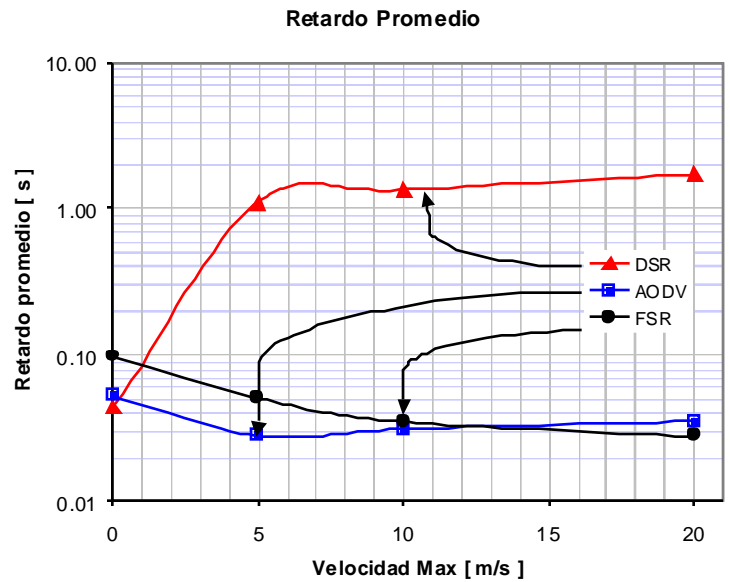


Figura 13. Retardo Promedio vs. Movilidad

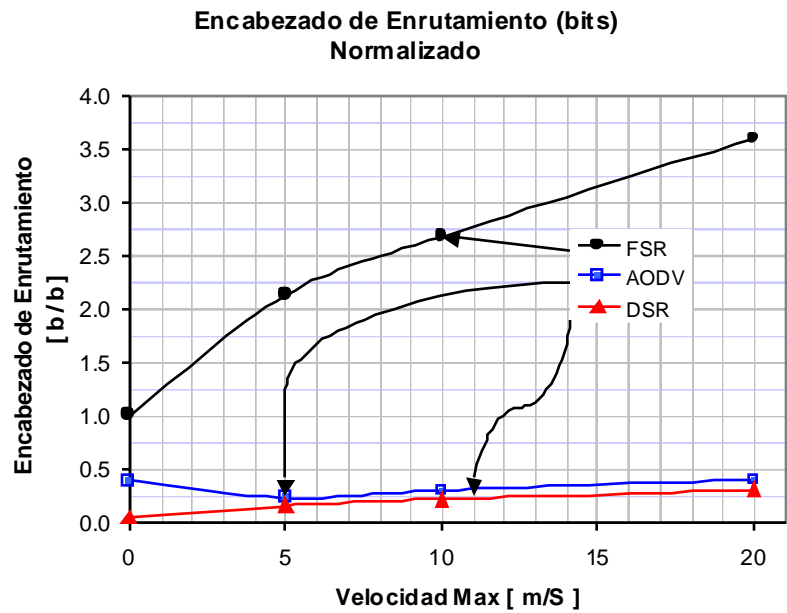


Figura 14. Encabezado de Enrutamiento vs. Movilidad

En la Figura 14 se presenta el encabezado de enrutamiento normalizado, se puede observar un crecimiento para FSR, esto por que a medida que la velocidad aumenta los cambios en la topología aumentan y FSR tiene

mayor información para propagar, a diferencia de los protocolos reactivos que solamente actualizan las rutas en uso.

En la Figura 15 se muestra el Consumo en el modo de operación PSP, se aprecia estable para los algoritmos reactivos y una disminución en el consumo para el algoritmo FSR. Comparando estos resultados con los de la Figura 16 se observa el consumo normalizado para AODV permanece constante en todos los rangos de velocidad, pero para los algoritmos DSR y FSR, la eficiencia disminuye a pesar de su aparente estabilidad y/o reducción en el consumo.

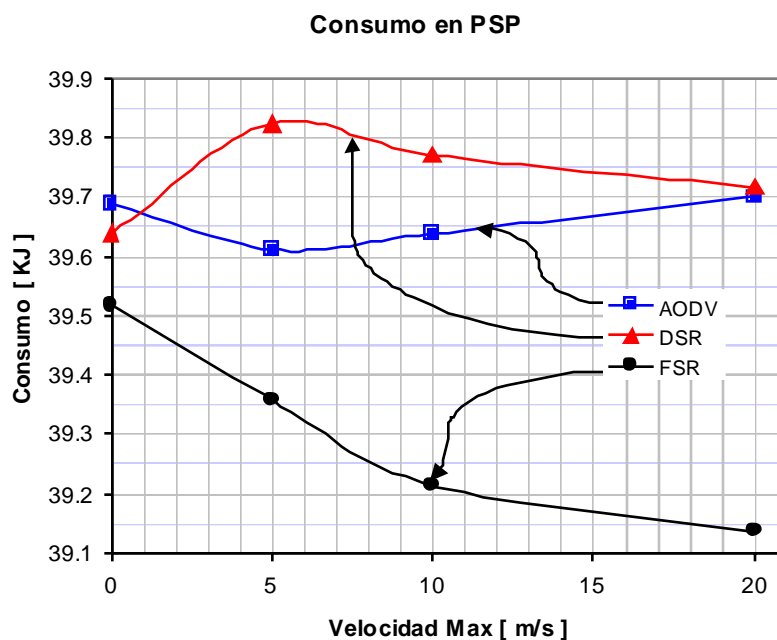


Figura 15 Consumo de Energía en PSP vs. Movilidad

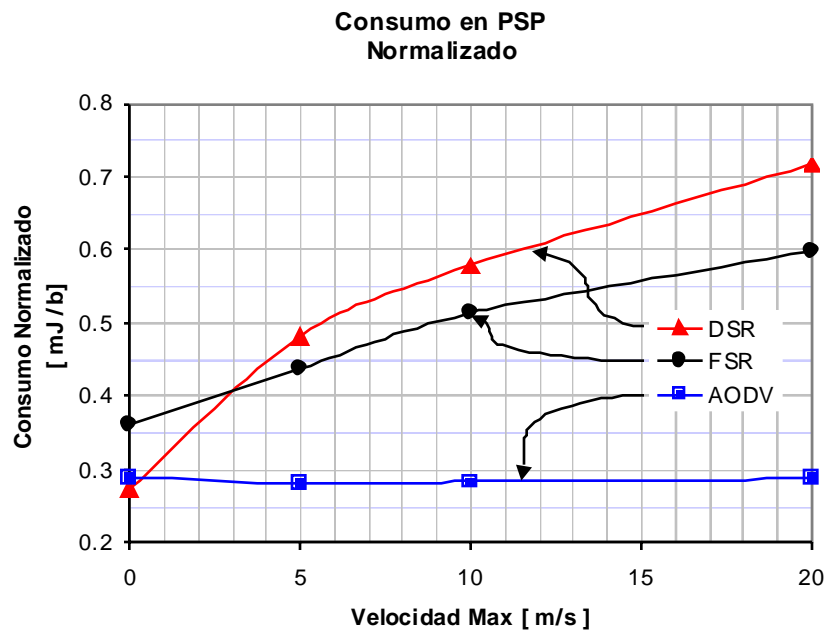


Figura 16. Consumo de Energía Normalizado vs. Movilidad

4 CONCLUSIONES Y RECOMENDACIONES

Al finalizar este análisis se puede afirmar que el comportamiento de los protocolos de enrutamiento en cuanto a consumo de energía se debe estudiar de manera relacionada con el desempeño de la red.

Se observó poca influencia del algoritmo de enrutamiento en el consumo de energía. En general el consumo de energía para los tres algoritmos tuvo el mismo comportamiento en los diferentes escenarios excepto para los escenarios con movilidad, que es el caso más realista. La diferencia en magnitudes en el consumo de energía con el modelo de consumo implementado fue inferior al 3% en todos los casos.

Como lo presentado en otros estudios[7],[10], el consumo de energía en redes Ad Hoc está dominado por los consumos registrados en estado inactivo, por la gran cantidad de tiempo que las estaciones permanecen en éste. El consumo en estado de transmisión es poco relevante por el corto tiempo que los nodos duran en éste. Esto se evidencia por la baja eficiencia energética de la red, cuando se opera a cargas por debajo del punto de saturación.

El modelo de consumo de energía implementado refleja de manera más cercana el comportamiento de la interfaz física operando en 802.11b DCF.

Una de las principales observaciones presentadas en este análisis es la poca influencia del radio alcance en la eficiencia energética de la red, el

aumento en el consumo, se ve compensado con un aumento en el desempeño de la red.

Se evidenció que el tamaño del paquete tiene influencia en la eficiencia energética de la red para cargas por encima del punto de saturación, para cargas bajas el tamaño del paquete es indiferente.

El algoritmo DSR presentó bajo desempeño en escenarios con movilidad, este comportamiento no se asemeja al reportado en otros estudios [16]-[18], también su eficiencia energética se degradó con el aumento en la velocidad de los nodos aunque según lo presentado en [13] está diseñado para escenarios de alta y muy alta movilidad.

Este trabajo da las bases para la elaboración de una comparación del desempeño de algoritmos de enrutamiento diseñados para mejorar el consumo de energía con algoritmos como los presentados en este análisis, e incluir una métrica de supervivencia de la red.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1] National Institute of Standards and Technology NIST
<http://www3.nist.gov/wctg/manet/manet.html>
- [2] L.M. Feeney, "An energy-consumption model for performance analysis of routing protocols for mobile ad hoc networks" *Mobile Networks and Applications Vol. 3, No. 6, June 2001*.
- [3] S. Singh, C. S. Woo Mghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks" *ACM MOBICOM 98 Dallas Texas USA*
- [4] M. Subbarao "Dynamic Power-Conscious Routing for MANETs: An Initial Approach" *Proc. IEEE VTC Fall 1999, Amsterdam, The Netherlands, Sept. 1999*.
- [5] V. Marbukh, M. Subbarao "Framework For Maximum Survivability Routing For A Manet" *Proc. MILCOM 2000, Los Angeles, October 22-25, 2000, pp. 281-28*
- [6] P. Bergamo, A. Giovanardi, A. Travasoni, D. Maniezzo, G. Mazzini, M. Zorzi "Distributed power control for energy efficient routing in ad hoc networks" *Wireless Networks, January 2004 Volume 10 Issue 1 Pg 29 – 42*
- [7] X. Du "A Simulation Study of An Energy Efficient Routing Protocol for Mobile Ad Hoc Networks" *Proceedings of the 37th annual symposium on Simulation April 2004 IEEE*
- [8] Ephremides, "Energy Concerns in Wireless Networks" *IEEE Wireless Communications Agosto. 2002 P 48 – 59*
- [9] J Goldsmith, S. B. Wicker, "Design Challenges For Energy Constrained Ad Hoc Wireless Networks" *IEEE Wireless Communications Agosto. 2002 P 8 – 27*
- [10] D. O. Cunha, L. H. Costa, M. K. Duarte, "Analyzing the Energy Consumption of IEEE 802.11 Ad Hoc Networks", to appear in *Proc of Sixth IFIP IEEE International Conference on Mobile and Wireless Communication Networks MWCN'2004, 2004*
- [11] M. M Carvalho, C. B. Margi, K. Obraczka, and J. J. Garcia-Luna-Aceves. "Modeling energy consumption in single-hop IEEE 802.11 ad hoc networks." *In Thirteenth International Conference on Computer Communications and Networks (ICCCN'04), Chicago, USA, October 2004*

- [12]B. Margi, K. Obraczka "Instrumenting Network Simulators for Evaluating Energy Consumption in Power-Aware Ad-Hoc Network Protocols" *The IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS'04) 10 04 - 10, 2004 Volendam, The Netherlands P 337, 346*
- [13]E. Perkins and E. M. Royer, "Ad hoc on-demand distance vector routing," in *Proc. IEEE WMCSA'99, New Orleans, LA, Feb. 1999.*
- [14]B. Johnson, D. A. Maltz, Yih-Chun Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," *IETF Internet Draft, draft-ietf-manet-dsr-09.txt*
- [15]M. Gerla, X. Hong, G. Pei. "Fisheye State Routing Protocol (FSR)" *IETF Internet Draft draft-ietf-manet-fsr-03.txt June, 2002*
- [16]C. E. Perkins, E. M. Royer, S. Das, M. Marina "Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks". *INFOCOMM 2002 Conference*
- [17]J. Prokkola, L. Leppänen, T. Bräysy. "On the Effect Of Traffic Models To The Performance Of Ad Hoc Network" *Milcom 2003 IEEE Military Communications Conference Vol 22 No 1 Oct 2003 Pg 422 – 427.*
- [18]J. Broch, D. Maltz, D. B. Johnson, Yih-Chun Hu, J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols" *Proc. of IEEE/ACM Mobicom '98, Oct. 1998.*
- [19]P. Sholander, A. Yankopolus, P. Coccoli, S. S. Tabrizi, "Experimental Comparison of Hybrid and Proactive MANET Routing Protocols," *Disponibile en www.scires.com/pdfs/mr-paper1.pdf*
- [20]O. Jörg, "Performance Comparison Of MANET Routing Protocols In Different Network Sizes," *Disponibile en: www.stanford.edu/~amaaron/ee360/EE360_FINAL_PAPER.pdf*
- [21]Qualnet Forums, *Disponibile en : www.scalable-networks.com/training_and_support/support/forums/*
- [22]Cisco Aironet Wireless LAN Adapters Software Configuration Guide, *Disponibile en www.cisco.com*

- [23]L.M.Feeney, M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment" *Technical report T99:11, SICS*.
- [24]J. Banks. Discrete-event system simulation (*Prentice Hall international series in industrial and systems engineering*).
- [25]Yoon, Jungkeun Liu, Mingyan, Noble, Brian. "Random Waypoint Considered Harmful" *IEEE INFOCOM 2003*

ANEXO 1. MODELO DE CONSUMO DE ENERGÍA EN MODO PSP

Para implementar el nuevo modelo de consumo de energía en Qualnet® 3.8, se hicieron una modificaciones básicas al código de la capa física del protocolo 802.11.

Modificaciones de phy_802_11.h

Al archivo phy_802_11.h se le hicieron dos modificaciones pequeñas, en la primera se comentaron las definiciones del modelo original de Qualnet para luego incluirlas como variables

```
// #define BATTERY_SLEEP_POWER      (50.0 / SECOND)
// #define BATTERY_IDLE_POWER       (850.0 / SECOND)
// #define BATTERY_RX_POWER         (970.0 / SECOND)
// #define BATTERY_TX_POWER_OFFSET  (970.0 / SECOND)
// #define BATTERY_TX_POWER_COEFFICIENT (11.4 / SECOND)
```

Luego en la definición de las estructuras se agregaron las variables para el nuevo modelo.

```
/*
 * Structure for phy statistics variables
 */
typedef struct phy_802_11_stats_str {
    int totalTxSignals;
    int totalRxSignalsToMac;
    int totalSignalsLocked;
    int totalSignalsWithErrors;
    double energyConsumed;
    double energyConsumedPSP;
    clocktype turnOnTime;
    clocktype Rx OnTime;
    clocktype TxOnTime;
    int control;
} Phy802_11Stats;

/*
```



```

* Structure for Phy.
*/
typedef struct struct_phy_802_11_str {
    PhyData* thisPhy;
    int    txDataRateTypeForBC;
    int    txDataRateType;
    float  txPower_dBm;
    float  txDefaultPower_dBm[PHY802_11_NUM_DATA_RATES];
    int    rxDataRateType;
    double rxSensitivity_mW[PHY802_11_NUM_DATA_RATES];
    int    numDataRates;
    int    dataRate[PHY802_11_NUM_DATA_RATES];
    double numDataBitsPerSymbol[PHY802_11_NUM_DATA_RATES];
    int    lowestDataRateType;
    int    highestDataRateType;
    double directionalAntennaGain_dB;
    Message* rxMsg;
    double rxMsgPower_mW;
    clocktype rxTimeEvaluated;
    BOOL    rxMsgError;
    clocktype rxEndTime;
    Orientation rxDOA;
    int    channelBandwidth;
    clocktype rxTxTurnaroundTime;
    double noisePower_mW;
    double interferencePower_mW;
    PhyStatusType mode;
    PhyStatusType previousMode;
    Phy802_11Stats stats;
double IdlePower;
double RxPower;
double TxPowerOffset;
double TxPowerCoeff;
} PhyData802_11;

```

Modificaciones de phy_802_11.cpp

En el archivo phy_802_11.cpp se hicieron las siguientes modificaciones:

En la función Phy802_11Init(), se incluyeron las instrucciones para leer los parámetros desde el archivo de configuración y luego se inicializan así:

```

void Phy802_11Init(
    Node *node,

```

```

const int phyIndex,
const NodeInput *nodeInput)
{
    BOOL    wasFound;
    .
    .
    .
IO_ReadDouble(
    node->nodeId,
    node->phyData[phyIndex]->networkAddress,
    nodeInput,
    "BATTERY_IDLE_POWER",
    &wasFound,
    &(phy802_11->IdlePower));

    if (!wasFound )
    {
        ERROR_ReportError(
            "BATTERY_IDLE_POWER is missing\n");
    }

IO_ReadDouble(
    node->nodeId,
    node->phyData[phyIndex]->networkAddress,
    nodeInput,
    "BATTERY_RX_POWER",
    &wasFound,
    &(phy802_11->RxPower));

    if (!wasFound)
    {
        ERROR_ReportError(
            "BATTERY_RX_POWER is missing\n");
    }
IO_ReadDouble(
    node->nodeId,
    node->phyData[phyIndex]->networkAddress,
    nodeInput,
    "BATTERY_TX_POWER_OFFSET",
    &wasFound,
    &(phy802_11->TxPowerOffset));

    if (!wasFound )
    {
        ERROR_ReportError(
            "BATTERY_TX_POWER_OFFSET is missing\n");
    }
IO_ReadDouble(
    node->nodeId,
    node->phyData[phyIndex]->networkAddress,
    nodeInput,

```

```

        "BATTERY_TX_POWER_COEFFICIENT",
        &wasFound,
        &(phy802_11->TxPowerCoeff));
    if (!wasFound )
    {
        ERROR_ReportError(
            "BATTERY_TX_POWER_COEFFICIENT is missing\n");
    }
.
.
.

//
// Initialize phy statistics variables
//
phy802_11->stats.totalRxSignalsToMac = 0;
phy802_11->stats.totalSignalsLocked = 0;
phy802_11->stats.totalSignalsWithErrors = 0;
phy802_11->stats.totalTxSignals = 0;
phy802_11->stats.energyConsumed = 0.0;
phy802_11->stats.turnOnTime = getSimTime(node);

// Incluida para el nuevo modelo
phy802_11->stats.energyConsumedPSP = 0.0;
phy802_11->stats.Rx OnTime = getSimTime(node);
phy802_11->stats.Tx OnTime = getSimTime(node);
phy802_11->stats.control = 0;

.
.
.
}

```

En la función Phy802_11Finalize() se calculan las potencias consumidas en el modo de operación CAM y PSP, según el tiempo de simulación y los tiempos acumulados en cada estado al momento de ser invocada por Qualnet al finalizar el tiempo de simulación. Esta función además imprime los resultados.

```

void Phy802_11Finalize(Node *node, const int phyIndex) {
.
.
.
    printf(buf, "Signals received but with errors = %d",
        phy802_11->stats.totalSignalsWithErrors);
}

```

```

IO_PrintStat(node, "Physical", "802.11", ANY_DEST, phyIndex, buf);

phy802_11->stats.energyConsumed
+= ( phy802_11->RxPower / SECOND )
  * (getSimTime(node) - phy802_11->stats.turnOnTime);
sprintf(buf, "Energy consumption at CAM (in mWs) = %.3f",
        phy802_11->stats.energyConsumed );
IO_PrintStat(node, "Physical", "802.11", ANY_DEST, phyIndex, buf);

// CALCULO DE POTENCIA PARA PSP
phy802_11->stats.energyConsumedPSP
+= ( phy802_11->IdlePower / SECOND )
  * (getSimTime(node) - phy802_11->stats.turnOnTime);

sprintf(buf, "Energy consumption at fast PSP (in mWs) = %.3f",
        phy802_11->stats.energyConsumedPSP );
IO_PrintStat(node, "Physical", "802.11", ANY_DEST, phyIndex, buf);

// La variable clockStrStatistic, es definida para hacer la conversión de tipo desde Clocktype
//... definido para Qualnet a un Char y poder enviar estos datos utilizando las función sprintf().
// esta conversión se logra con la función de Qualnet TIME_PrintClockInSecond().
char clockStrStatistic[24];
TIME_PrintClockInSecond(phy802_11->stats.Tx OnTime, clockStrStatistic);
sprintf(buf, "Tiempo en Transmisión (S) = %s",clockStrStatistic);
IO_PrintStat(node, "Physical", "802.11", ANY_DEST, phyIndex, buf);

TIME_PrintClockInSecond(phy802_11->stats.Rx OnTime, clockStrStatistic);
sprintf(buf, "Tiempo en Recepcion (S) = %s",clockStrStatistic);
IO_PrintStat(node, "Physical", "802.11", ANY_DEST, phyIndex, buf);

TIME_PrintClockInSecond(
getSimTime(node) - (phy802_11->stats.Tx OnTime + phy802_11->stats.Rx OnTime),
clockStrStatistic);
sprintf(buf, "Tiempo en Reposo (S) = %s",clockStrStatistic);
IO_PrintStat(node, "Physical", "802.11", ANY_DEST, phyIndex, buf);

}

```

En la función `Phy802_11SignalEndFromChannel` se recibe el mensaje desde el medio físico se calcula el tiempo en estado de recepción y se efectúan los cálculos del consumo en para este estado. Así:

```

void Phy802_11SignalEndFromChannel(
.
.
.

```

```

// Calculando el tiempo de recepción del mensaje para los calculos de consumo de energía
// en fast PSP
    PropTxInfo *propTxInfo
        = (PropTxInfo *)MESSAGE_ReturnInfo(propRxInfo->txMsg);
    clocktype rxDuration = propTxInfo->duration;
.
.
.
// Se agrega el calculo de las estadísticas de recepción
// phy802_11->stats.control++;
    phy802_11->stats.energyConsumedPSP
        += (double) (( phy802_11->RxPower / SECOND ) - ( phy802_11->IdlePower /
            SECOND )) * rxDuration;
    phy802_11->stats.RxOnTime += rxDuration;
}

```

De manera similar para el caso de transmisión, en la función StartTransmittingSignal se calcula el tiempo en estado de transmisión y se efectúan los cálculos del consumo en para este estado. Así:

```

StartTransmittingSignal
{
.
.
.
    phy802_11->mode = PHY_TRANSMITTING;

    duration = Phy802_11GetFrameDuration(
        thisPhy, phy802_11->txDataRateType, packetsize);
.
.
.
    /* Keep track of phy statistics and battery computations */
    phy802_11->stats.totalTxSignals++;
    phy802_11->stats.energyConsumed
        += duration * (( phy802_11->TxPowerCoeff / SECOND )
            * NON_DB(phy802_11->txPower_dBm)
            + ( phy802_11->TxPowerOffset / SECOND )
            - ( phy802_11->RxPower / SECOND ));
    phy802_11->stats.TxOnTime += duration;

// Para el calculo de las estadísticas en Fast PSP
    phy802_11->stats.energyConsumedPSP
        += duration * (( phy802_11->TxPowerCoeff / SECOND )
            * NON_DB(phy802_11->txPower_dBm)
            + ( phy802_11->TxPowerOffset / SECOND )
            - ( phy802_11->IdlePower / SECOND ));
}

```

```
}
```

Modificaciones de choices.xml

El archivo choices.xml es el que define los parámetros disponibles en la interfaz gráfica de Qualnet y permite que estos parámetros sean escritos en el archivo de configuración. En el archivo choices.xml se insertó el siguiente fragmento de código, el cual permite que las variables de potencia se puedan ajustar desde la interfaz gráfica.

Observación: En el archivo “choices.xml” no se deben utilizar tildes ni otros caracteres especiales por que no son reconocidos por Qualnet al cargar la interfaz gráfica.

```
<variable name="802.11 Potencia en reposo" key="BATTERY_IDLE_POWER" type="Fixed" default="850.0" help="Potencia en reposo 802.11 en mW"/>
```

```
<variable name="802.11 Potencia en recepcion" key="BATTERY_RX_POWER" type="Fixed" default="970.0" help="Potencia en recepcion 802.11 en mW" />
```

```
<variable name="802.11 Potencia en transmision offset" key="BATTERY_TX_POWER_OFFSET" type="Fixed" default="970.0" help="Potencia nominal en transmision (consumo de la circuiteria) 802.11 en mW" />
```

```
<variable name="802.11 Coef. de Transmision" key="BATTERY_TX_POWER_COEFFICIENT" type="Fixed" default="36.0" help="Coeficiente de transmision Adimensional, excedente del consumo de la circuiteria utilizado solo para transmision" />
```

ANEXO 2. ARCHIVO DE CONFIGURACIÓN DEL ESCENARIO

A continuación incluimos una copia del archivo de configuración “escenario.config” con el que se realizaron las simulaciones

```
# ***** QualNet Configuration File *****
# ***** General *****
VERSION 3.8
EXPERIMENT-NAME Sin-RTS-S4-L4
SIMULATION-TIME 900S
# The random number seed is used to initialize part of the seed of
various randomly generated numbers in the simulation. Use different
seeds to see the consistency of the results of the simulation.
SEED 2

# ***** Parallel Settings *****

# Method for assigning nodes to parallel partitions
PARTITION-SCHEME AUTO

# ***** Terrain *****

# The size of the physical terrain in which the nodes are being
simulated.

COORDINATE-SYSTEM CARTESIAN
# The size of the terrain in meters.
TERRAIN-DIMENSIONS ( 1500, 300 )
# Terrain altitude in meters.
DUMMY-ALTITUDES ( 1500, 1500 )
# If this is set to YES, the simulation terminates when it attempts
to use an elevation not included in the terrain data files. If it is
NO, the execution simply assumes that such elevations are 0.0.
TERRAIN-DATA-BOUNDARY-CHECK YES

# ***** Node Positioning *****

# ***** Nodes *****

# The number of nodes being simulated.
DUMMY-NUMBER-OF-NODES 50
# The node placement strategy.
NODE-PLACEMENT FILE
NODE-POSITION-FILE C:\Mis documentos\A
Tesis\Escenarios\Prokkola\Prokkola.nodes
```

```
# ***** Mobility *****

MOBILITY GAUSS-MARKOV
# Paso de tiempo en segundos
MOBILITY-GM-D-TIME 1.0
# meters per second
MOBILITY-GM-AV-SPEED 10
# meters per second
MOBILITY-GM-DS-SPEED 2
# radianes
MOBILITY-GM-AV-DIR 2.49
# radianes
MOBILITY-GM-DS-DIR 1.0
# Adimencional entre 0 y 1
MOBILITY-GM-COEF 0.5
MOBILITY-POSITION-GRANULARITY 1.0
# If yes, nodes get their altitude coordinate from the terrain file,
if one is specified.
MOBILITY-GROUND-NODE NO

# ***** Wireless Settings *****

# ***** Channel *****

PROPAGATION-CHANNEL-FREQUENCY 2400000000
PROPAGATION-MODEL STATISTICAL
# Signals with powers below PROPAGATION-LIMIT (in dBm) (before the
antenna gain at the receiver) are not delivered.
PROPAGATION-LIMIT -111.0
PROPAGATION-PATHLOSS-MODEL TWO-RAY
PROPAGATION-SHADOWING-MODEL CONSTANT
# in dB
PROPAGATION-SHADOWING-MEAN 4.0
PROPAGATION-FADING-MODEL NONE

# ***** Radio/Physical Layer *****

PHY-MODEL PHY802.11b
PHY802.11-AUTO-RATE-FALLBACK NO
# bandwidth in bps. supported data rates: 1Mbps, 2Mbps, 5.5Mbps,
11Mbps
PHY802.11-DATA-RATE 2000000
PHY802.11b-TX-POWER--1MBPS 9.8
PHY802.11b-TX-POWER--2MBPS 9.8
PHY802.11b-TX-POWER--6MBPS 15.0
PHY802.11b-TX-POWER--11MBPS 15.0
PHY802.11b-RX-SENSITIVITY--1MBPS -93.0
PHY802.11b-RX-SENSITIVITY--2MBPS -89.0
PHY802.11b-RX-SENSITIVITY--6MBPS -87.0
PHY802.11b-RX-SENSITIVITY--11MBPS -83.0
```



```

PHY802.11-ESTIMATED-DIRECTIONAL-ANTENNA-GAIN 15.0
PHY-RX-MODEL PHY802.11b
# Potencia en reposo 802.11 en mW
BATTERY_IDLE_POWER 850.0
# Potencia en recepcion 802.11 en mW
BATTERY_RX_POWER 970.0
# Potencia nominal en transmision (consumo de la circuiteria) 802.11
en mW
BATTERY_TX_POWER_OFFSET 970.0
# Coeficiente de transmision Adimensional, excedente del consumo de
la circuiteria utilizado solo para transmision
BATTERY_TX_POWER_COEFFICIENT 36.0
# Channels the radio is capable of listening to.
PHY-LISTENABLE-CHANNEL-MASK 1
# Channels the radio is currently listening to. Can be changed
during run time.
PHY-LISTENING-CHANNEL-MASK 1
# Temperature of the environment in K
PHY-TEMPERATURE 290.0
PHY-NOISE-FACTOR 10.0
ANTENNA-MODEL OMNIDIRECTIONAL
# antenna gain in dB
ANTENNA-GAIN 0.0
# antenna height in meters
ANTENNA-HEIGHT 1.5
# efficiency of the antenna
ANTENNA-EFFICIENCY 0.8
# antenna mismatch loss in dB
ANTENNA-MISMATCH-LOSS 0.3
# antenna cable loss in dB
ANTENNA-CABLE-LOSS 0.0
# antenna connection loss in dB
ANTENNA-CONNECTION-LOSS 0.2

# ***** MAC Protocol *****

MAC-PROTOCOL MAC802.11
MAC-802.11-DIRECTIONAL-ANTENNA-MODE NO
MAC-802.11-SHORT-PACKET-TRANSMIT-LIMIT 2
MAC-802.11-LONG-PACKET-TRANSMIT-LIMIT 2
MAC-802.11-RTS-THRESHOLD 2000
MAC-802.11-PCF-STATISTICS NO
# specifies an additional delay for messages sent by the MAC layer to
the phy layer. Some MAC protocols use a multiple of this value.
MAC-PROPAGATION-DELAY 1US
# must be set to YES if nodes want to overhear packets destined to
the neighboring node.
PROMISCUOUS-MODE YES

# ***** Adaptation Protocols *****
# ***** Adaptation Layer *****

```

```

ADAPTATION-LAYER-STATISTICS NO
ATM-STATIC-ROUTE NO
# ***** ARP Specific *****
# ***** ARP Enabled *****

ARP-ENABLED NO

# ***** ARP Specs *****

ARP-TIMEOUT-INTERVAL 20M

# ***** Network Protocols *****
# ***** Network Protocol *****
NETWORK-PROTOCOL IP
IP-ENABLE-LOOPBACK YES
IP-LOOPBACK-ADDRESS 127.0.0.1
IP-FRAGMENTATION-UNIT 2048
IP-QUEUE-NUM-PRIORITIES 1
IP-QUEUE-PRIORITY-INPUT-QUEUE-SIZE 7000
DUMMY-PRIORITY-QUEUE-SIZE YES
IP-QUEUE-PRIORITY-QUEUE-SIZE 50000
DUMMY-PRIORITY-WISE-IP-QUEUE-TYPE NO
IP-QUEUE-TYPE FIFO
# ECN as presented in RFC 2481. Requires one of the IP-QUEUE-TYPE
# (RED, RIO, or WRED). Furthermore, the source and destination nodes
# must be ECN enabled.
ECN NO
IP-QUEUE-SCHEDULER STRICT-PRIORITY
ROUTER-BACKPLANE-THROUGHPUT 0

# ***** Routing Protocol *****

DUMMY-ROUTING DYNAMIC
ROUTING-PROTOCOL AODV
# The maximum possible number of hops between two nodes in the
# network
AODV-NET-DIAMETER 35
# Conservative estimate of the average one hop traversal time for
# packets and should include queuing, transmission, propagation and
# other delays
AODV-NODE-TRAVERSAL-TIME 40MS
# Timeout time for an active route; each time a data packet is
# received, the lifetime of that route is updated to this value. A
# default value of 10 seconds is suggested for error detection through
# MAC layer message (like what 802.11 does)
AODV-ACTIVE-ROUTE-TIMEOUT 3S
# The destination of a RREQ replies with AODV-MY-ROUTE-TIMEOUT as the
# lifetime of the route.
AODV-MY-ROUTE-TIMEOUT 6S

```

```
# Lifetime of a hello message is determined by AODV-
ALLOWED_HELLO_LOSS * AODV-HELLO_INTERVAL
AODV-HELLO-INTERVAL 1S
# Lifetime of a hello message is determined by AODV-
ALLOWED_HELLO_LOSS * AODV-HELLO_INTERVAL
AODV-ALLOWED-HELLO-LOSS 2
# Specifies the number of times AODV will repeat expanded ring search
for a destination if no Route Reply is received within specified
amount of time.
AODV-RREQ-RETRIES 2
# A constant use for calculating the time after which an active route
should be deleted. After timeout of an active route, the route is
finally deleted from the routing table after a time period of (K *
max (AODV-ACTIVE_ROUTE_TIMEOUT, AODV-ALLOWED_HELLO_LOSS * AODV-
HELLO_INTERVAL)), Here K is AODV-ROUTE-DELETION-CONSTANT.
AODV-ROUTE-DELETION-CONSTANT 5
# If the value is set to YES, a node will send a hello message if
there is no broadcast within the last hello interval. Simulation time
will increase depending on the frequency of the hello updates.
AODV-PROCESS-HELLO NO
# If this value is set to YES, the node will try to locally repair a
broken route, if possible.
AODV-LOCAL-REPAIR NO
# If the source node of a route gets a route error message, it will
initiate a new Route Request for the destination if the value is set
to YES.
AODV-SEARCH-BETTER-ROUTE NO
# Maximum number of packets the message buffer of AODV can hold. If
the buffer fills up, incoming packets for the buffer will be dropped.
AODV-BUFFER-MAX-PACKET 100
# If nothing is specified, buffer overflow will be checked by number
of packets in the buffer. If some value is specified here, incoming
packets will be dropped if the incoming packet size + current size of
the buffer exceeds this value.
AODV-BUFFER-MAX-BYTE 0
AODV-OPEN-BI-DIRECTIONAL-CONNECTION YES
AODV-TTL-START 1
AODV-TTL-INCREMENT 2
AODV-TTL-THRESHOLD 7
HSRP-PROTOCOL NO
# Static routes have priority over those discovered by routing
protocols
STATIC-ROUTE NO
# Default routes have less priority than static routes and those
discovered by routing protocols
DEFAULT-ROUTE YES
DEFAULT-ROUTE-FILE C:\Mis documentos\A
Tesis\Escenarios\Prokkola\Prokkola.routes-default

# ***** MPLS configuration *****
MPLS-PROTOCOL NO
```

```
# ***** Transport Layer *****
TCP LITE
TCP-USE-RFC1323 NO
TCP-DELAY-ACKS YES
TCP-DELAY-SHORT-PACKETS-ACKS NO
TCP-USE-NAGLE-ALGORITHM YES
TCP-USE-KEEPALIVE-PROBES YES
TCP-USE-PUSH YES
TCP-MSS 512
TCP-SEND-BUFFER 16384
TCP-RECEIVE-BUFFER 16384

# ***** ATM Layer2 *****
# ***** ATM Layer2 *****
ATM-RED-MIN-THRESHOLD 5
ATM-RED-MAX-THRESHOLD 15
ATM-RED-MAX-PROBABILITY 0.02
ATM-RED-SMALL-PACKET-TRANSMISSION-TIME 10MS
ATM-QUEUE-SIZE 15000
ATM-SCHEDULER-STATISTICS NO
ATM-LAYER2-STATISTICS NO
ATM-QUEUE-STATISTICS NO

# ***** Traffic and Status *****
# ***** Application Layer *****
# Used to setup applications such as FTP and Telnet. Will be added
to any applications configured manually.
APP-CONFIG-FILE C:\Mis documentos\A
Tesis\Escenarios\Prokkola\Prokkola.app

# ***** Extras *****
# ***** Tracing *****
# Generates trace data compatible with Tracer viewing tool.
PACKET-TRACE NO
ACCESS-LIST-TRACE NO

# ***** Statistics *****
# ***** Statistics *****
# All the statistics are compiled together into a file called
"ExperimentName.stat"(where experiment name is specified on the
General settings) at the end of the simulation.

APPLICATION-STATISTICS YES
TCP-STATISTICS YES
UDP-STATISTICS YES
ROUTING-STATISTICS YES
ICMP-STATISTICS NO
IGMP-STATISTICS NO
EXTERIOR-GATEWAY-PROTOCOL-STATISTICS YES
NETWORK-LAYER-STATISTICS YES
QUEUE-STATISTICS YES
```

```

SCHEDULER-STATISTICS YES
MAC-LAYER-STATISTICS YES
PHY-LAYER-STATISTICS YES
MOBILITY-STATISTICS NO
MPLS-STATISTICS NO
MPLS-LDP-STATISTICS NO
RSVP-STATISTICS NO
SRM-STATISTICS NO
DIFFSERV-EDGE-ROUTER-STATISTICS NO
QOSPF-STATISTICS NO
# Network Statistics should be on
ACCESS-LIST-STATISTICS NO
POLICY-ROUTING-STATISTICS NO
ROUTE-REDISTRIBUTION-STATISTICS NO
SIGNALLING-STATISTICS NO
MOBILE-IP-STATISTICS NO

# ***** Node Specific *****
# ***** Device properties *****
USE-NODE-ICON YES
NODE-ICON C:\Mis documentos\A Tesis\Escenarios\Prokkola\DEFAULT.GIF

# ***** Router Specs *****
# ***** Router Configuration Specs *****
# ***** Node Orientation *****
AZIMUTH 0
ELEVATION 0
# ***** Parallel Properties *****
# Parallel partition to which to assign node.
PARTITION 0

#-----Default Subnet -----
SUBNET N8-192.0.0.0 { 1 thru 50 } Default
IP-FORWARDING NO
[ 1 thru 50 ] IP-FORWARDING YES
COMPONENT 0 {1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
46 47 48 49 50} 50 750.0 150.0 0.0 1500.0
COMPONENT 0 {300.0 3000.0

```