

**ANÁLISIS DEL IMPACTO DEL NIVEL DE ENLACE DE DATOS EN EL DESEMPEÑO  
DEL PROTOCOLO TCP EN LAS REDES SATELITALES.**



**TRABAJO DE GRADO**

**Víctor Hugo Ruiz Guachetá.  
Juan Esteban Nava Villarreal.**

**Director: Mg. Francisco Javier Terán.**

**UNIVERSIDAD DEL CAUCA  
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES  
Departamento de Telecomunicaciones  
Línea de Investigación: Gestión Integrada de Redes, Servicios y Arquitecturas de  
Telecomunicaciones  
Popayán, Junio de 2010**

## TABLA DE CONTENIDO

<b>CAPÍTULO I</b>	<b>Pag</b>
1.0. INTRODUCCION.....	1
1.1. TCP EN LAS REDES SATELITALES.....	2
1.1.1. Características de las Redes Satelitales.....	2
1.1.2. Análisis del Rendimiento de TCP en las Redes Satelitales.....	3
1.2.1.1. El primer segmento de transmisión de TCP.....	4
<b>CAPÍTULO II</b>	
2.0. MECANISMOS TCP.....	6
2.1. Mecanismos de Control de Flujo y Opciones.....	6
2.1.1. TCP Reno.....	6
2.1.2. TCP New Reno.....	7
2.1.3. TCP Sack.....	8
2.1.4. TCP Hybla.....	10
2.1.5. TCP Westwood.....	11
2.1.6. TCP Westwood+.....	12
2.1.7. TCP HighSpeed.....	13
2.2. MODIFICACIÓN A LA ARQUITECTURA INTRODUCIENDO NODOS INTERMEDIOS EN LA RED E IMPLEMENTACIONES ENTRE LOS NODOS ESTÁNDAR O SOLUCIONES PROPIETARIAS.....	14
2.2.1. PEP.....	14
2.2.2. Tipos de PEPs.....	14
2.2.3. Implementaciones PEP en la Red VSAT.....	15
2.2.4. Soluciones PEP para Mejorar el Rendimiento del Protocolo TCP.....	15
2.3. MECANISMOS DEL NIVEL DE ENLACE DE DATOS.....	15
2.3.1. Proxy Spoofing.....	15
2.3.1.1. Proxy snoop.....	16
2.3.2. Técnicas de Acceso al Medio.....	18
2.3.2.1. ACCESO MÚLTIPLE POR DIVISIÓN DE FRECUENCIA.....	18
2.3.2.2. ACCESO MÚLTIPLE POR DIVISIÓN DE TIEMPO (TDMA).....	20
2.3.2.3. ACCESO MÚLTIPLE ALEATORIO (AMA).....	24
2.3.3. Estándar DVB-RCS.....	25
2.3.3.1. Políticas de asignación de recursos para el canal de retorno DVB-RCS.....	27
2.3.4. Corrección de Errores Hacia Adelante FEC.....	28

### CAPÍTULO III

3.0.	LA HERRAMIENTA DE SIMULACIÓN. ....	30
3.1.	ARQUITECTURA DE RED EMPLEADA. ....	31
3.1.1.	Parámetros de Entrada al Simulador NS-2. ....	34
3.1.2.	Métricas de Simulación para Evaluar el Desempeño de TCP y Rendimiento de la Red. ....	36
3.1.2.1.	Métricas TCP. ....	36
3.1.2.2.	Métricas del rendimiento de la red. ....	37
3.2.	OBTENCIÓN Y ANÁLISIS DE RESULTADOS EN EL DESEMPEÑO DE TCP WESTWOOD, HYBLA Y HIGH SPEED VARIANDO PARÁMETROS DE LA CAPA DE ENLACE DE DATOS. .38	
3.2.1.	Impacto al Variar el Número de Slots por Trama en TDMA. ....	38
3.2.1.1.	Análisis de la ventana de congestión, <i>ssthresh</i> , número de paquetes transmitidos y retransmitidos. ....	39
3.2.1.2.	Análisis de ACK, RTT y número de secuencia. ....	40
3.2.1.3.	Análisis del encolamiento y retardo de la red. ....	41
3.2.1.4.	Análisis del <i>throughput</i> y capacidad del enlace utilizado. ....	42
3.2.2.	Impacto al Variar el Tamaño del Slot en TDMA. ....	42
3.2.2.1.	Análisis de la ventana de congestión, <i>ssthresh</i> , número de paquetes transmitidos y retransmitidos. ....	43
3.2.2.2.	Análisis de ACK, RTT y número de secuencia. ....	44
3.2.2.3.	Análisis del encolamiento y retardo de la red. ....	45
3.2.2.4.	Análisis del <i>throughput</i> y capacidad del enlace utilizado. ....	45
3.2.3.	Impacto al Variar la Duración de la Trama en TDMA. ....	46
3.2.3.1.	Análisis de la ventana de congestión, el <i>ssthresh</i> y paquetes transmitidos. ....	47
3.2.3.2.	Análisis de ACK, RTT y número de secuencia. ....	48
3.2.3.3.	Análisis del encolamiento y retardo de la red. ....	50
3.2.3.4.	Análisis del <i>throughput</i> y capacidad del enlace utilizado. ....	50
3.2.4.	Impacto al Variar el Tiempo de Reprogramación TBTP en TDMA. ....	51
3.2.4.1.	Análisis de la ventana de congestión, el <i>ssthresh</i> y paquetes retransmitidos. ....	52
3.2.4.2.	Análisis de ACK, RTT y número de secuencia. ....	53
3.2.4.3.	Análisis del encolamiento y retardo de la red. ....	54
3.2.4.4.	Análisis del <i>throughput</i> y capacidad del enlace utilizado. ....	54
3.2.5.	Impacto al variar el Tipo de Asignación de Recursos en la NCC. ....	55
3.2.5.1.	Análisis de la ventana de congestión, <i>ssthresh</i> , paquetes transmitidos y retransmitidos. ....	56
3.2.5.2.	Análisis de ACK, y RTT. ....	57
3.2.5.3.	Análisis del encolamiento y retardo de la red. ....	57

3.2.5.4.	Análisis del desempeño de la red. ....	58
3.2.6.	Impacto al Utilizar los Diferentes Tipos de Asignación Bajo Demanda en TDMA.....	59
3.2.6.1.	Análisis para TCP hybla. ....	59
3.2.6.2.	Análisis para TCP HighSpeed.....	62
3.2.6.3.	Análisis de TCP Westwood.....	65
3.2.7.	Impacto del Protocolo Aloha en los Parámetros TCP y del Desempeño en la Red. ....	76
3.2.7.1.	Análisis de la ventana de congestión. ....	77
3.2.7.2.	Análisis de ACKs, número de secuencia y paquetes transmitidos. ....	77
3.2.7.3.	Análisis del RTT. ....	78
3.2.7.4.	Análisis del rendimiento y retardo de la red.....	79
3.2.8.	Impacto al Variar el Tiempo Aleatorio de Retransmisión. ....	84
3.2.8.1.	Análisis de la ventana de congestión. ....	84
3.2.8.2.	Análisis de ACKs, número de secuencia y paquetes transmitidos. ....	84
3.2.8.3.	Análisis del RTT. ....	85
3.2.8.4.	Análisis del rendimiento y retardo de la red.....	86
3.3.	MECANISMO DE LA CAPA DE ENLACE DE DATOS MODIFICANDO LA ARQUITECTURA DE LA RED.....	86
3.3.1.	Impacto del Proxy Snoop en el Desempeño de TCP.....	86
3.3.1.1.	Descripción de resultados para la prueba P con y sin la inclusión del proxy snoop....	88
3.3.1.2.	Descripción de resultados en la prueba Q con y sin la inclusión del proxy snoop. ....	91
3.3.1.3.	Descripción de resultados en la prueba R con y sin la inclusión del proxy snoop.....	94
3.3.2.	Impacto del Snoop sobre un Entorno con RTTs muy Altos.....	97
<b>CAPÍTULO IV</b>		
4.0.	CONCLUSIONES. ....	99
4.1.	RECOMENDACIONES. ....	101
4.2.	TRABAJOS FUTUROS. ....	101
4.3.	APORTES DEL PROYECTO. ....	101

## TABLA DE FIGURAS.

	Pag
Figura 1-1. Segmento tráfico TCP en bloques de ráfagas.....	¡Error! Marcador no definido.
Figura 2-1. Función de respuesta de <i>cwnd</i> según su tasa de pérdida. ....	¡Error! Marcador no definido.
Figura 2-2. Proceso del ACK en las secciones de la red TCP.....	¡Error! Marcador no definido.
Figura 2-3. Estructura física de TCP Spoofing.....	¡Error! Marcador no definido.
Figura 2-4.a Modulo snoop ack().....	¡Error! Marcador no definido.
Figura 2-4.b Modulo snoop data(). ....	¡Error! Marcador no definido.
Figura 2-5. Ocupación de un transpondedor de 36 MHz con FDMA; cada señal proviene de una población diferente y tiene su propia portadora asignada. ....	¡Error! Marcador no definido.
Figura 2-6. Rendimiento de FDMA. ....	¡Error! Marcador no definido.
Figura 2-7. Transmisión de estaciones terrenas de tráfico y de referencia utilizando TDMA.....	¡Error! Marcador no definido.
<b>Figura 2-8. TDMA con asignación fija y variable. ....</b>	<b>¡Error! Marcador no definido.</b>
Figura 2-9. Esquema de sincronismo TDMA. ....	¡Error! Marcador no definido.
Figura 2-10. Esquema temporal de frecuencia en MF-TDMA.....	¡Error! Marcador no definido.
Figura 2-11. Flujo de paquetes en un canal ALOHA. ....	¡Error! Marcador no definido.
Figura 2-12. Sistema DVB-RCS. ....	¡Error! Marcador no definido.
Figura 3-1. Primer escenario de simulación.....	¡Error! Marcador no definido.
Figura 3-2. Segundo escenario de simulación con Aloha Puro. ....	¡Error! Marcador no definido.
Figura 3-3. Estructura empleada para analizar el impacto del protocolo TCP al variar los parámetros de los mecanismos del nivel de enlace de datos. ....	¡Error! Marcador no definido.
Figura 3-4. Evolución de los paquetes transmitidos. ....	¡Error! Marcador no definido.
Figura 3-5. Comportamiento de CWND. ....	¡Error! Marcador no definido.
Figura 3-6. Comportamiento de ACKs.....	¡Error! Marcador no definido.
Figura 3-7. Comportamiento del número de secuencia.....	¡Error! Marcador no definido.
Figura 3-8. Comportamiento RTT.....	¡Error! Marcador no definido.
Figura 3-9. <i>Throughput</i> de la red. ....	¡Error! Marcador no definido.
Figura 3-10. Utilización del canal por todas las conexiones. ....	¡Error! Marcador no definido.
Figura 3-11. Comportamiento de la ventana de congestión. ....	¡Error! Marcador no definido.
Figura 3-12. Paquetes transmitidos. ....	¡Error! Marcador no definido.
Figura 3-13. Comportamiento de los ACKs.....	¡Error! Marcador no definido.
Figura 3-14. Comportamiento del RTT. ....	¡Error! Marcador no definido.
Figura 3-15. <i>Throughput</i> de la red.....	¡Error! Marcador no definido.
Figura 3-16. Utilización del canal satelital por todas las conexiones.....	¡Error! Marcador no definido.
Figura 3-17. Comportamiento de CWND en las pruebas E y F. ....	¡Error! Marcador no definido.
Figura 3-18. Comportamiento de CWND en la prueba G. ....	¡Error! Marcador no definido.
Figura 3-19. Comportamiento de los paquetes transmitidos. ....	¡Error! Marcador no definido.
Figura 3-20. Comportamiento del ACK en las pruebas E, F y G.....	¡Error! Marcador no definido.

Figura 3-21a. Pruebas E y F para la evolución del RTT.....¡Error! Marcador no definido.

Figura 3-21b. Evolución del RTT en la prueba G. ....¡Error! Marcador no definido.

Figura 3-22. *Throughput* de la red para las tres versiones de TCP analizadas.¡Error! Marcador no definido.

Figura 3-23. Pruebas E y F para la utilización de canal por todas las conexiones de la red.....¡Error! Marcador no definido.

Figura 3-24. Utilización de canal por todas las conexiones de la red en la prueba G.¡Error! Marcador no definido.

Figura 3-25. Comportamiento de CWND para diferentes TBTP. ....¡Error! Marcador no definido.

Figura 3-26. Comportamiento de los paquetes transmitidos. ....¡Error! Marcador no definido.

Figura 3-27. Comportamiento de los ACKs.....¡Error! Marcador no definido.

Figura 3-28. Comportamiento del RTT. ....¡Error! Marcador no definido.

Figura 3-29. Comportamiento del *throughput* para distintos valores de TBTP.¡Error! Marcador no definido.

Figura 3-30. Utilización del canal.....¡Error! Marcador no definido.

Figura 3-31. Comportamiento de la ventana de congestión. ....¡Error! Marcador no definido.

Figura 3-32. Comportamiento de los paquetes transmitidos. ....¡Error! Marcador no definido.

Figura 3-33. Comportamiento de los ACKs.....¡Error! Marcador no definido.

Figura 3-34. Comportamiento del RTT. ....¡Error! Marcador no definido.

Figura 3-35. Asignación y utilización de recursos para TCP Westwood. ....¡Error! Marcador no definido.

Figura 3-36. *Throughput* de la red observado en las pruebas B y K. ....¡Error! Marcador no definido.

Figura 3-37. Utilización del canal por todas las conexiones. ....¡Error! Marcador no definido.

Figura 3-38a. Slots solicitados y asignados con la política RVBDC.....¡Error! Marcador no definido.

Figura 3-38b. Ancho de banda *uplink* y utilización del canal empleado con RVBDC.¡Error! Marcador no definido.

Figura 3-39. Relación entre los paquetes encolados y los recursos solicitados por la RCST. ....¡Error! Marcador no definido.

Figura 3-40a. *Throughput* y utilización del canal con la política RVBDC.....¡Error! Marcador no definido.

Figura 3-40b. Retardo en la red con RVBDC.....¡Error! Marcador no definido.

Figura 3-41a. Slots solicitados y asignados con la política RBDC. ....¡Error! Marcador no definido.

Figura 3-41b. Ancho de banda *uplink* y utilización del canal empleado con RVBDC.¡Error! Marcador no definido.

Figura 3-42. Comportamiento de los parámetros de TCPW.....¡Error! Marcador no definido.

Figura 3-43. Comportamiento de los parámetros de TCPW.....¡Error! Marcador no definido.

Figura 3-44. Desempeño del *throughput* y utilización del canal.....¡Error! Marcador no definido.

Figura 3-45. Comportamiento del retardo de la red.....¡Error! Marcador no definido.

Figura 3-46. Asignación y utilización de recursos de la red.....¡Error! Marcador no definido.

Figura 3-47. Comportamiento de los parámetros TCPW-CRA. ....¡Error! Marcador no definido.

Figura 3-48. *Throughput* y utilización de la red. ....¡Error! Marcador no definido.

Figura 3-49. Comportamiento de los parámetros TCPW con VBDC.....¡Error! Marcador no definido.

Figura 3-50. Asignación y utilización de recursos de la red.....¡Error! Marcador no definido.

Figura 3-51. Rendimiento de la red. ....	¡Error! Marcador no definido.
Figura 3-52. Comportamiento de CWND en las tres versiones de TCP.....	¡Error! Marcador no definido.
Figura 3-53. Comportamiento de parámetros TCP Hybla, TCPW, TCP HS.¡	¡Error! Marcador no definido.
Figura 3-54. Paquetes transmitidos en TCP Hybla, TCPW, TCP HS.....	¡Error! Marcador no definido.
Figura 3-55. Comportamiento del RTT. ....	¡Error! Marcador no definido.
Figura 3-56. Evolución BW en el tiempo.....	¡Error! Marcador no definido.
Figura 3-57. Bits por segundos recibidos en los nodos receptores.....	¡Error! Marcador no definido.
Figura 3-58. Evolución del <i>throughput</i> de TCP. ....	¡Error! Marcador no definido.
Figura 3-59. Evolución del retardo.....	¡Error! Marcador no definido.
Figura 3-60. Comportamiento de CWND. ....	¡Error! Marcador no definido.
Figura 3-61. Comportamiento de parámetros TCP. ....	¡Error! Marcador no definido.
Figura 3-62. Comportamiento del RTT. ....	¡Error! Marcador no definido.
Figura 3-63. Comportamiento de CWND. ....	¡Error! Marcador no definido.
Figura 3-64. Evolución del número de secuencia y ACK con Snoop. ....	¡Error! Marcador no definido.
Figura 3-65. Evolución del número de secuencia y ACK sin Snoop. ....	¡Error! Marcador no definido.
Figura 3-66. Comportamiento del RTT con y sin snoop. ....	¡Error! Marcador no definido.
Figura 3-67. Utilización del canal con y sin snoop.....	¡Error! Marcador no definido.
Figura 3-68. Comportamiento de CWND. ....	¡Error! Marcador no definido.
Figura 3-69. Evolución del número de secuencia y ACK con Snoop. ....	¡Error! Marcador no definido.
Figura 3-70. Evolución del número de secuencia y ACK sin Snoop. ....	¡Error! Marcador no definido.
Figura 3-71. Comportamiento del RTT con y sin snoop. ....	¡Error! Marcador no definido.
Figura 3-72. Utilización del canal con y sin snoop.....	¡Error! Marcador no definido.
Figura 3-73. Comportamiento de CWND y <i>ssthresh</i> con snoop. ....	¡Error! Marcador no definido.
Figura 3-74. Comportamiento de CWND y <i>ssthresh</i> sin snoop. ....	¡Error! Marcador no definido.
Figura 3-75. Evolución del número de secuencia y ACK con Snoop. ....	¡Error! Marcador no definido.
Figura 3-76. Evolución del número de secuencia y ACK sin Snoop. ....	¡Error! Marcador no definido.
Figura 3-77. Comportamiento del RTT con y sin snoop. ....	¡Error! Marcador no definido.
Figura 3-78. Utilización del canal con y sin snoop.....	¡Error! Marcador no definido.

## LISTADO DE TABLAS

Pag

Tabla 3-1. Parámetros de entrada al simulador. ....	¡Error! Marcador no definido.
Tabla 3-2. Parámetros de las versiones de TCP.....	¡Error! Marcador no definido.
Tabla 3-3. Pruebas realizadas para evaluar el impacto al modificar el N° slots por trama en TCP....	¡Error! Marcador no definido.
Tabla 3-4. Pruebas realizadas para evaluar el impacto de modificar el tamaño del slot en TCP.....	¡Error! Marcador no definido.
Tabla 3-5. Impacto de TCP al variar la duración de la trama.....	¡Error! Marcador no definido.
Tabla 3-6. Impacto de TCP Westwood al variar el TBTP. ....	¡Error! Marcador no definido.
Tabla 3-7. Impacto de TCP Hybla y TCP HS al variar el TBTP. ....	¡Error! Marcador no definido.
Tabla 3-8. Impacto de TCP al variar el tipo de asignación de recursos. ....	¡Error! Marcador no definido.
Tabla 3-9. Impacto de TCP al variar los diferentes tipos de asignación bajo demanda.	¡Error! Marcador no definido.
Tabla 3-10. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión uno.	¡Error! Marcador no definido.
Tabla 3-11. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión dos.	¡Error! Marcador no definido.
Tabla 3-12. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión tres.	¡Error! Marcador no definido.
Tabla 3-13. <i>Throughput</i> promedio y utilización del canal en las pruebas L, M, N y O.	¡Error! Marcador no definido.
Tabla 3-14. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión uno.	¡Error! Marcador no definido.
Tabla 3-15. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión dos.	¡Error! Marcador no definido.
Tabla 3-16. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión tres.	¡Error! Marcador no definido.
Tabla 3-17. <i>Throughput</i> promedio y utilización del canal en las pruebas L, M, N y O.	¡Error! Marcador no definido.
Tabla 3-18. Comportamiento de <i>CWND</i> , <i>ssthresh</i> , PKTX y ACKs.....	¡Error! Marcador no definido.
Tabla 3-19. Comportamiento de algunos parámetros TCP. ....	¡Error! Marcador no definido.
Tabla 3-20. Comportamiento de <i>CWND</i> , <i>ssthresh</i> , PkRTX y ACKs.....	¡Error! Marcador no definido.
Tabla 3-21. Comportamiento de <i>CWND</i> , <i>ssthresh</i> , PkRTX y ACKs.....	¡Error! Marcador no definido.
Tabla 3-22. Parámetros de desempeño y de TCPW en las pruebas L, M, N y O para la conexión uno.	¡Error! Marcador no definido.



- Tabla 3-23. Parámetros de desempeño y de TCPW en las pruebas L, M, N y O para la conexión dos. **¡Error! Marcador no definido.**
- Tabla 3-24. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión tres. **¡Error! Marcador no definido.**
- Tabla 3-25. Rendimiento de la red.....**¡Error! Marcador no definido.**
- Tabla 3-26. Parámetros de simulación. ....**¡Error! Marcador no definido.**
- Tabla 3-27. Resultados para TCP Westwood. ....**¡Error! Marcador no definido.**
- Tabla 3-28. Resultados para TCP HS. ....**¡Error! Marcador no definido.**
- Tabla 3-29. Resultados para TCP Hybla. ....**¡Error! Marcador no definido.**
- Tabla 3-30. Resultados para TCPW. ....**¡Error! Marcador no definido.**
- Tabla 3-31. Resultados para TCP HS. ....**¡Error! Marcador no definido.**
- Tabla 3-32. Resultados para TCP Hybla. ....**¡Error! Marcador no definido.**
- Tabla 3-33. Probabilidad de colisión para *mean-backoff* de 1s. ....**¡Error! Marcador no definido.**
- Tabla 3-34. Impacto del protocolo TCP con y sin el uso del proxy snoop. ....**¡Error! Marcador no definido.**
- Tabla 3-35. Parámetros de evaluación en la prueba P con Snoop. ....**¡Error! Marcador no definido.**
- Tabla 3-36. Parámetros de evaluación en la prueba Q con Snoop.....**¡Error! Marcador no definido.**
- Tabla 3-37. Parámetros de evaluación en la prueba R con Snoop.....**¡Error! Marcador no definido.**
- Tabla 3-38. Parámetros de evaluación en en la prueba P sin Snoop. ....**¡Error! Marcador no definido.**
- Tabla 3-39. Parámetros de evaluación en la prueba Q sin Snoop.....**¡Error! Marcador no definido.**
- Tabla 3-40. Parámetros de evaluación en en la prueba R sin Snoop. ....**¡Error! Marcador no definido.**
- Tabla 3-41. Parámetros configurados sobre un entorno con RTTs altos con y sin snoop. **¡Error! Marcador no definido.**
- Tabla 3-42. Parámetros de evaluación en la prueba S con Snoop con la política RBDC. **¡Error! Marcador no definido.**
- Tabla 3-43. Parámetros de evaluación en la prueba T con Snoop con la política RBDC. **¡Error! Marcador no definido.**
- Tabla 3-44. Parámetros de evaluación en la prueba U con Snoop con la política RBDC. **¡Error! Marcador no definido.**
- Tabla 3-45. Parámetros de evaluación en la prueba S sin Snoop con la política RBDC. **¡Error! Marcador no definido.**
- Tabla 3-46. Parámetros de evaluación en la prueba T sin Snoop con la política RBDC. **¡Error! Marcador no definido.**
- Tabla 3-47. Parámetros de evaluación en la prueba U sin Snoop con la política RBDC. **¡Error! Marcador no definido.**

## ACRÓNIMOS

**ACK:** Reconocimiento, *Acknowledgement*.

**AMA:** Acceso Múltiple Aleatorio, *Random Multiple Access*.

**ARQ:** Solicitud de Repetición Automática, *Automatic Repeat Request*.

**ATM:** Modo de Transferencia Asíncrona, *Asynchronous Transfer Mode*.

**AVBDC:** Volumen Absoluto Basado en Capacidad Dinámica, *Absolute Volume Based Dynamic Capacity*.

**BER:** Tasa de Error de Bit, *Bit Error Rate*.

**CBR:** Tasa de *Bit* Constante, *Constant Bit Rate*.

**CDMA:** Acceso Múltiple por División de Código, *Code Division Multiple Access*.

**CRA:** Asignación de Tasa Continúa, *Continuous Rate Assignment*.

**CWND:** Ventana de Congestión, *Congestion Window*.

**DAMA:** Acceso Múltiple por División en Frecuencia con Asignación por Demanda, *Demand Assigned Multiple Access*.

**DTDMA:** Acceso al Medio por División de Tiempo Bajo Demanda, *Distributed Time Division Multiple Access*.

**DVB:** *Transmisión de Televisión Digital, Digital Video Broadcasting*.

**DVB-S:** *Transmisión de Televisión Digital por Satélite, Digital Video Broadcasting by Satellite*.

**ELN:** Notificación Explícita de Pérdida, *Explicit Loss Notification*.

**ERE:** *Estimación de Tasa Elegible, Eligible Rate Estimate*.

**FAK:** Reconocimiento de Envío, *Forward Acknowledgement*.

**FCA:** Asignación de Capacidad Disponible, *Free Capacity Assignment*.

**FDMA:** Acceso Múltiple por División de Frecuencia, *Frequency Division Multiple Access*.

**FEC:** Corrección de Errores hacia Adelante, *Forward Error Correction*.

**FIFO:** Primero en Entrar, primero en Salir, *First Input First Output*.

**FTP:** Protocolo de Transferencia de Archivo, *File Transfer Protocol*.

**GEO:** Órbita Terrestre Geoestacionaria, *Geostationary Earth Orbit*.

**GES:** Pasarelas de la Estación Terrena, *Gateway Earth Station*.

**HTTP:** Protocolo de Transferencia de Hipertexto, *Hypertext Transfer Protocol*.

**IP:** Protocolo de Internet, *Internet Protocol*.

**LAN:** Red de Área Local, *Local Area Network*.

**MAC:** Control de Acceso al Medio, *Media Access Control*.

**MPEG:** Grupo de Expertos en Imagen en Movimiento, *Moving Picture Experts Group*.

**MTU:** Unidad Máxima de Transferencia, *Maximum Transmission Unit*.

**NACK:** Reconocimiento Negativo, *Negative Acknowledgement*.

**NCC:** Centro Control de Red, *Network Control Center*.

**NS:** Simulador de Redes, *Network Simulator*.

**PCM:** Modulación por Impulsos Codificados, *Pulse Code Modulation*.

**PEP:** *Proxy que Mejora el Rendimiento, Performance Enhancing Proxy*.

**RB:** Ráfaga de Referencia, *Reference Burst*.

**RBDC:** Tasa Basada en la Capacidad Dinámica, *Rate Based Dynamic Capacity*.

**RCST:** Estación Terrena con Canal de Retorno, *Return Channel Satellite Terminal*.

**RTO:** Retransmisión del *timeout*, *Retransmission Timeout*.

**RTT:** Tiempo de Ida y Vuelta, *Round Trip Time*.

**SACK:** Reconocimientos Selectivos, *Selective Acknowledgement*.

**SCPT:** Única Portadora por Transpondedor, *Single Carrier per Transponder*.

**SMS:** Servicio de Mensaje Corto, *Short Message Service*.

**TB:** Ráfagas de tráfico *Traffic Burst*.

**TBTP:** Tiempo de Planificación de Ráfaga del Terminal, *Terminal Burst Time Plan*.

**TCM:** Modulación por Codificación de *Trellis*, *Trellis Coded Modulation*.

**TCP:** Protocolo de Control de Transmisión, *Transmission Control Protocol*.

**TDMA:** Acceso Múltiple por División de Tiempo, *Time Division Multiple Access*.

**TLS:** Seguridad de la Capa de Transporte, *Transport Layer Security*.

**UDP:** Protocolo de Datagrama de Usuario, *User datagram protocol*.

**UW:** Palabra Única, *Unique Word*.

**VBDC:** Asignación de Capacidad Basado en Volumen, *Volume Based Dynamic Capacity*.

**VSAT:** Terminal de Apertura muy Pequeño, *Very Small Aperture Terminal*.

**WAN:** Red de Area Extendida, *Wide Area Network*.

## CAPITULO I

### GENERALIDADES DEL PROTOCOLO TCP EN LAS REDES SATELITALES.

#### 1.0. INTRODUCCION.

Los avances tecnológicos de las últimas décadas han producido una verdadera revolución en los campos de la producción, la prestación de servicios, la educación y las relaciones interpersonales, así como también en la forma como se organizan y dirigen los procesos [1], un ejemplo de esto, se presenta con las comunicaciones, que permiten al hombre afrontar las necesidades diarias con mayor eficiencia y rapidez, un caso concreto de estos avances son las redes satelitales que tuvo sus orígenes en el año 1954, cuyo desarrollo ha sido vertiginoso y hoy en día es considerado un factor indispensable en el progreso tecnológico de los países [1].

Muchas de las tecnologías y protocolos que existen en la actualidad han tenido que modificarse para obtener un mejor desempeño y así adaptarse a las condiciones que otorgan las nuevas redes y servicios, un ejemplo específico de esto, se presenta en las redes satelitales, donde se ha optado por utilizar el protocolo TCP como protocolo de transporte. Como se conoce este protocolo es uno de los más importantes en la red de Internet y sus mecanismos de control de congestión fueron originalmente desarrollados para medios guiados [2], los cuales se caracterizan por presentar un menor ancho de banda, bajos tiempos de retardo de propagación, una tasa de error de bits pequeña; características contrarias en el canal satelital, especialmente en el caso de las redes satelitales GEO, donde se presentan grandes productos por retardo, incrementos en la estimación del RTT, debido a los largos tiempos de propagación que se encuentran alrededor de los 250 y 300 milisegundos, si a esto se le suma los tiempos de señalización, acceso al medio, gestión de búfer, pérdidas por congestión de la red y errores de transmisión, hacen que este tipo de redes presenten grandes productos por retardo por ancho de banda e incrementos en el cálculo del RTT. Por otro lado, para una red satelital GEO el crecimiento del tamaño de la ventana de congestión depende del número de reconocimientos ACKs que llegan al transmisor [3,4], aunque en algunos casos las pérdidas en el ambiente satelital son causadas por la congestión, la alta tasa de error de bits del canal y la alta latencia, son factores que afectan este sistema. Como TCP no distingue las causas que ocasionan pérdidas de paquete, asume que son debidas a congestión y reacciona activando sus algoritmos de control de congestión disminuyendo su tamaño de ventana, lo que implica una utilización deficiente del ancho de banda disponible reduciendo el rendimiento de la red.

Debido a los inconvenientes presentes en los enlaces satelitales se proponen algunas soluciones que ayudan a mejorar el comportamiento TCP y optimizar el desempeño de la red, dentro de estos mecanismos se proponen soluciones de nivel inferior, como los códigos FEC y ARQ, o el mecanismo *Path MTU Discovery* [5]. Otro tipo de soluciones se enfocan en hacer modificaciones a los mecanismos de control de flujo TCP (inicio lento, evasión de congestión, retransmisión rápida y recuperación rápida), la opción de escalado de ventana, los reconocimientos selectivos y los reconocimientos de envío [5]. Por último se encuentran soluciones que hacen uso de *proxies*, conocidos como PEPs, estos son empleados para compensar las características degradantes específicas a las que pertenece un enlace o una subred en su trayectoria, su principal objetivo es poder aislar la alta latencia y el gran producto por retardo por ancho de banda que se tiene en el segmento satelital del enlace terrestre y así poder implementar procesos o técnicas que optimicen su desempeño [5, 6].

Las soluciones nombradas, intentan mitigar el efecto de utilizar TCP dentro de este tipo de redes, algunas de sus soluciones están enfocadas a una mejora del protocolo, sin embargo otros factores externos a éste también deben ser considerados en conjunto para poder dar una solución integral que tome en cuenta estos factores, como los son los mecanismos de gestión y planificación de colas, los tipos de acceso al medio, los protocolos de aplicación, enlace de datos y de nivel físico implementados en la red satelital [3].

De acuerdo a lo anterior en este proyecto se estudió y analizó los diferentes mecanismos TCP empleados en las redes satelitales y particularmente se enfoca en el impacto que tiene el variar los mecanismos de la capa de enlace de datos en el desempeño del protocolo TCP en sus versiones TCP Westwood, TCP Hybla y TCP HighSpeed haciendo uso de la herramienta de simulación NS-2 en su versión 2.31.

Este trabajo de grado tiene una estructura definida en capítulos atendiendo a los temas tratados durante el desarrollo de la misma.

En el capítulo I se nombran algunas generalidades del protocolo TCP en un ambiente satelital.

En el capítulo II se estudia el comportamiento de las diferentes versiones de TCP, como también los distintos mecanismos de la capa de enlace de datos.

En el capítulo III se hace una descripción y justificación de los escenarios de simulación, se realiza una descripción de la herramienta de simulación a utilizar, además se analiza el impacto que genera implementar los distintos mecanismos de la capa de enlace de datos en las versiones TCP seleccionadas.

En el capítulo IV se presentan las conclusiones a las que se llegó con el desarrollo del trabajo teniendo en cuenta los objetivos planteados al inicio del proyecto, se muestra también las recomendaciones para quienes deseen continuar con trabajos e investigaciones que estén relacionadas con esta temática, por último se muestran los trabajos futuros y aportes del trabajo de grado.

## **1.1. TCP EN LAS REDES SATELITALES.**

El protocolo TCP fue estructurado para utilizarse en redes terrestres que utilizan medios guiados, donde el retardo de propagación es muy bajo y las distancias que tiene que recorrer son muy cortas, debido a que el protocolo es ampliamente utilizado como mecanismo de transporte de datos y además de ser el soporte de todas las aplicaciones de Internet, se le han hecho algunas modificaciones para que pueda adaptarse a las redes satelitales, las cuales se caracterizan por tener un ancho de banda limitado, largos retardos de propagación, y largas distancias para recorrer desde una estación terrena al satélite, entre otras que se comentarán más adelante, estas características hacen que TCP satelital no tenga un buen desempeño, sin embargo se han creado diferentes mecanismos que mitigan estos problemas.

### **1.1.1. Características de las Redes Satelitales.**

- **RTT:** Debido a los largos tiempos de propagación que presentan las redes satelitales GEO, al transmisor le toma mucho tiempo en determinar si ha recibido correctamente un paquete,

lo que afecta notablemente el comportamiento del RTT afectando los mecanismos de control de congestión y estimación del RTO de TCP [2, 6].

- **Gran producto por retardo por ancho de banda:** Este retardo define la cantidad de datos que un protocolo ha transmitido y que se encuentran en el canal de comunicaciones sin que todavía se haya tenido algún reconocimiento por parte del transmisor (paquetes en transición), este parámetro permite hacer un buen uso de la capacidad disponible de la red, pero a su vez limita el tamaño de la ventana de congestión de TCP [2, 6].
- **Errores de transmisión:** Los errores de transmisión en la red satelital son más frecuentes debido al uso de enrutadores, que afectan los mecanismos de control de congestión, y disminuye la tasa de transmisión de TCP [6, 7].
- **Uso Asimétrico:** En este tipo de redes se presenta que el enlace de subida es diferente al de bajada, el problema que resulta es que el tráfico de ráfagas degrada el desempeño de TCP y reduce la tasa de transmisión [2, 6].
- **RTTS variables:** En entornos satelitales GEO el retardo de propagación desde y hacia el satélite no varía con el tiempo, el RTT se ve afectado por la alta latencia presente en la red (tiempo de procesamiento del satélite, el retardo del encolamiento en la pasarela o el enrutador y el tiempo de señalización) [6].

### 1.1.2. Análisis del Rendimiento de TCP en las Redes Satelitales.

El análisis que se realizó tuvo en cuenta las fases de los algoritmos de inicio lento y evasión de congestión, considerando los parámetros de uso de conexión y el tiempo de transmisión en cada estado.

En la figura 1-1 se muestra el ancho de banda que es utilizado por un segmento en las diferentes fases de control de congestión TCP, el valor de  $2D$  es el tiempo requerido para que un paquete sea reconocido por el transmisor,  $p$  es el valor del RTT que se requiere para completar una transmisión y  $q$  es el número de segmentos transmitidos en la fase de evasión de congestión. A continuación se hará un análisis y cálculo del ancho de banda utilizado por diferentes conexiones TCP.

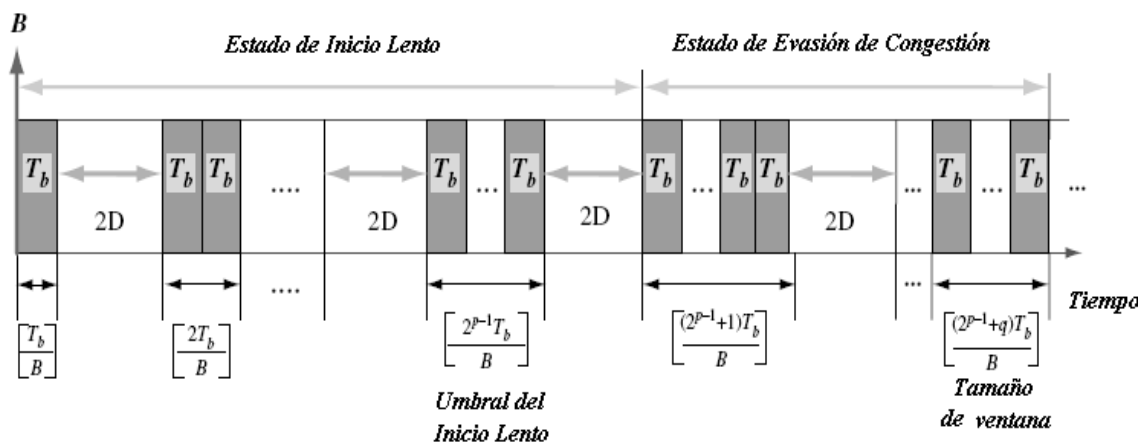


Figura 1-1. Segmento tráfico TCP en bloques de ráfagas [8].

### 1.2.1.1. El primer segmento de transmisión de TCP.

Para calcular el uso del ancho de banda en TCP es necesario hacerlo después de haber establecido la conexión y cuando TCP haya transmitido el primer segmento de datos; para tal efecto se utiliza la ecuación 1 para determinar el rendimiento de TCP en redes satelitales.

$$U = \frac{T}{2D + T} = \frac{T_b / B}{2D + T_b / B} = \frac{1}{2\left(\frac{DB}{T_b}\right) + 1} \quad (1)$$

Donde T es el tiempo para transmitir el primer segmento  $T_b$ ; D es el tiempo propagación; B indica el ancho de banda durante una sesión TCP y U es el rendimiento de TCP satelital. En la ecuación 1 se observa cómo puede afectar el producto por retardo por ancho de banda en el rendimiento de TCP. El tiempo para completar la transmisión de datos en TCP es de  $2D + T_b/B$  [8].

En la transmisión de TCP en el estado de inicio lento, luego de que TCP haya enviado el primer segmento satisfactoriamente, y si no hay ninguna pérdida de paquetes el algoritmo tendrá un crecimiento exponencial (dos paquetes de tráfico  $T_b$  se transmiten por cada ACK reconocido por el transmisor). Las ecuaciones 2 y 3 indican una secuencia de tamaños de paquetes transmitida  $F(n)$  y el uso de la conexión de TCP respectivamente.

$$F(n) = \left( \sum_{i=0}^{n-1} 2^i \right) T_b = (2^n - 1)T_b; \quad n \in N \quad (2)$$

$$U_{F(n)} = \frac{F(n) / B}{2nD + F(n) / B} = \frac{(2^n - 1)T_b / B}{2nD + (2^n - 1)T_b / B} = \frac{1}{1 + \left( \frac{2n}{(2^n - 1)T_b} \right) (DB)} \quad (3)$$

Donde n es el número total de RTT necesarios para completar la transmisión.

Las ecuaciones 4 y 5 muestran los segmentos de datos transmitidos y el uso del enlace, cuando TCP finaliza la etapa de inicio lento, respectivamente (sin ninguna pérdida de segmentos).

$$F = \left( \left( \sum_{i=0}^{n-1} 2^i \right) + \alpha 2^n \right) T_b = (2^n (1 + \alpha) - 1)T_b \quad \text{Donde } 0 \leq \alpha < 1 \quad (4)$$

$$U_F = \frac{1}{1 + \left( \frac{2(n+1)}{(2^n (1 + \alpha) - 1)T_b} \right) (DB)} \quad (5)$$

Durante una transmisión TCP, esta alcanza el estado de evasión de congestión cuando el tamaño de bloque de datos es igual al umbral del Inicio lento  $2^{p-1}T_b$ , lo que da pie a que comience la fase del algoritmo de evasión de congestión con un crecimiento lineal, con el que se procura aumentar el tamaño de ventana. Las ecuaciones 6 y 7 indican los segmentos de datos transmitidos y la utilización del enlace en éste estado [8].



$$F_l = \left( \sum_{i=0}^{p-1} 2^i \right) T_b + \left( \sum_{j=1}^m (2^{p-1} + j) \right) T_b + \beta 2^{p-1} T_b = \left( 2^p + \frac{m^2 + 3m}{2} - 1 \right) T_b \quad (6)$$

$$U_{F_l} = \frac{1}{1 + \frac{2(p+m)}{\left( 2^p + \frac{m^2 + 3m}{2} + \beta - 1 \right) T_b} (DB)} \quad (7)$$

Donde  $0 \leq \beta < 1$ , y  $2^m T_b \leq W$ , son parámetros del tamaño de ventana.

En el anexo A se describen aspectos generales de una red satelital GEO, del protocolo TCP, problemas y posibles soluciones que ayudan a mejorar el desempeño de TCP en un ambiente satelital.

## CAPÍTULO II

### MECANISMOS DE CONTROL DE FLUJO TCP Y MECANISMOS DEL NIVEL DE ENLACE DE DATOS.

#### 2.0. MECANISMOS TCP.

Los mecanismos TCP que se estudian a continuación intentan contrarrestar algunas de las características particulares de las redes satelitales que degradan el rendimiento en este tipo de redes, cada uno de ellos se centra en aspectos específicos y hacen variaciones a parámetros del mecanismo TCP y de la capa de enlace de datos. El objetivo del proyecto es utilizar todas estas alternativas que se presentan para poder determinar cómo puede influir las distintas combinaciones de los parámetros del nivel de enlace de datos en el rendimiento del protocolo TCP en una red satelital. A continuación según [9] se describen las principales propuestas de solución para mejorar el rendimiento en TCP:

Las propuestas o soluciones adoptadas pueden ser clasificadas como las siguientes:

- Enfoque de nivel inferior.
- Mecanismos de control de flujo modificados para implementar el uso de TCP.
- Modificación a la arquitectura introduciendo nodos intermedios en la red e implementaciones entre los nodos estándar o soluciones propietarias.

#### 2.1. Mecanismos de Control de Flujo y Opciones.

Los mecanismos de control de flujo que se describen en el anexo B son los mismos que se manejan en TCP estándar, salvo que se han hecho algunas modificaciones a sus parámetros para mejorar su rendimiento en una red satelital, si se evalúan todos estos algoritmos cada uno de ellos tendrá un mejor comportamiento en ciertas condiciones particulares del medio.

##### 2.1.1. TCP Reno.

TCP Reno se basa en cuatro mecanismos para el control de congestión de la ventana, denominados; inicio lento, evasión de congestión, retransmisión rápida y recuperación rápida, para ello utiliza dos variables: *cwnd* y el umbral de inicio lento (*ssthresh*) que son las encargadas de actualizar los paquetes en la red según sea la congestión en la red [10, 11].

TCP Reno se caracteriza por ser un mecanismo reactivo [12] ya que no permite manejar adecuadamente la congestión de la red, puesto que utiliza la misma pérdida de segmentos para predecir la disponibilidad del ancho de banda de la red. Es un mecanismo que no hace ninguna diferenciación entre pérdidas por congestión y pérdidas por transmisión.

El esquema de TCP Reno se comporta de manera similar a TCP Tahoe, sin embargo este último no utiliza el mecanismo de recuperación, lo que implica que regresa a la fase de inicio lento después de la pérdida de un segmento [11].

TCP Reno comienza el incremento de ventana de forma exponencial en la fase de Inicio lento siempre que el valor de *cwnd* sea menor que el *ssthresh*, se aumenta *cwnd* en 1 por cada paquete satisfactoriamente recibido, duplicándose *cwnd* por cada ACK reconocido, cuando el valor de *cwnd* es mayor que el *ssthresh* comienza la fase de evasión de congestión, el cual es un mecanismo más conservador en el incremento de *cwnd* ya que lo hace de una manera lineal, por cada ACK reconocido,  $cwnd = cwnd + 1/cwnd$ .

Los dos algoritmos mencionados anteriormente funcionan de esta manera siempre y cuando no haya ocurrido la pérdida de un segmento, si este es el caso, entonces puede ocurrir dos situaciones: que el número de ACKs repetidos supera un umbral, generalmente 3, lo que conlleva a que se retransmita el segmento perdido, *cwnd* se reduce a la mitad y el *ssthresh* se fija al nuevo *cwnd*, hay que tener en cuenta que si la red puede recibir 3 ACKs duplicados es una indicación de que la red no está tan congestionada y es una buena señal para disminuir el tamaño de ventana [11, 13]; la otra situación se presenta cuando el *timeout* expira, por lo tanto *cwnd* se fija en 1 y el *ssthresh* se establece a la mitad del valor anterior *cwnd*, además el algoritmo entra en la fase de inicio lento.

Se activa la recuperación rápida cuando el transmisor ha recibido los 3 ACKs duplicados, entonces se retransmite el paquete perdido y *cwnd* se incrementa en el mismo número de ACKs duplicados, conservando el principio de que si un ACK duplicado ha sido recibido por el receptor, implica que otro podría ser introducido, luego se espera el ACK del segmento retransmitido en el transmisor para reanudar con la fase de evasión de congestión, donde *cwnd* se ve reducido a la mitad y el *ssthresh* al nuevo *cwnd* [12, 14]. A continuación se indican cada uno de los casos mencionados en el algoritmo de TCP Reno.

Uno de los principales inconveniente de TCP Reno es que degrada notoriamente su rendimiento cuando ocurren múltiples pérdidas de segmentos en una sola ventana lo que obliga al transmisor a que expire el temporizador de retransmisión para poder continuar [15, 16].

```
If Se inicia la conexión y no se producen pérdidas de segmentos
  If (cwnd < ssthresh), se fija cwnd = cwnd + 1; // Fase de Inicio lento
  else Se fija cwnd = cwnd + 1 / cwnd. // Fase Evasión de Congestión.

else if
  if ( 3 ACKs duplicados ocurren)
    Se fija ssthresh = cwnd / 2, luego cwnd = ssthresh;
  else
    if (timeout expira)
      Se establece cwnd = ssthresh / 2; // Se regresa al inicio lento
    Se establece cwnd = 1.
```

### 2.1.2. TCP New Reno.

TCP New Reno es una versión mejorada de TCP Reno, modifica el algoritmo de recuperación rápida con el propósito de poder mejorar el rendimiento cuando se presenta múltiples pérdidas en una sola ventana, esta versión de TCP se encuentra documentada en el RFC 2582.

Cuando ocurre una pérdida de un segmento el mecanismo de recuperación rápida de TCP Reno se activa, retransmitiendo el paquete, pero si la red detecta que otro segmento se ha

perdido el mecanismo tendrá nuevamente que incorporarse a la fase de recuperación rápida, lo que afecta notoriamente el desempeño de TCP. TCP New Reno soluciona este inconveniente evitando permanecer en la fase de recuperación rápida haciendo uso del concepto ACK parcial [11, 16], este reconocimiento se origina cuando el transmisor recibe el ACK del primer paquete retransmitido, donde se reconocen algunos paquetes transmitidos pero no todos los que se enviaron antes de entrar a la fase de retransmisión rápida. Se debe tener en cuenta que el mecanismo permanece en esta fase siempre y cuando se reciban ACKs parciales.

El mecanismo de recuperación rápida se inicia cuando recibe los 3 ACKs duplicados [17], donde se registra el número de secuencia más alto enviado, con el fin de saber si todos los segmentos enviados antes de la pérdida han sido detectados por el receptor o no. El valor del *ssthresh* se ajusta de acuerdo a la ecuación 10.

$$ssthresh = \max (FS / 2, 2 *SMSS) \quad (10)$$

Donde **FS**: Datos enviados pero no reconocidos.  
**SMSS**: tamaño de segmento.

Después se retransmite el segmento perdido y *cwnd* toma el valor dado en la ecuación 11, donde se tiene en cuenta el número de secuencia más alto enviado.

$$cwnd = ssthresh + 3*SMSS \quad (11)$$

En la medida que el transmisor reciba un ACK duplicado este podrá aumentar su tamaño de ventana en un segmento, cuando llega un ACK que reconoce nuevos datos, el transmisor observa si se trata de un ACK parcial o un ACK que confirma todos los segmentos de datos que estaban pendientes entre el primer segmento perdido y la recepción del tercer ACK duplicado, si este es el caso entonces se ajusta *cwnd* (ver ecuación 12) y finaliza la fase de recuperación rápida; dando a entender que en la red no se ocasionó otra pérdida, por el contrario si es un ACK parcial, significa que se han perdido más paquetes en la red, por lo tanto el transmisor disminuye la ventana de congestión en la misma cantidad de datos reconocidos, y se retransmite sólo un paquete perdido por RTT. El proceso termina cuando se haya reconocido todos los datos pendientes al inicio de la fase de recuperación rápida [17, 18].

$$cwnd = \min (ssthresh + SMSS) \quad (12)$$

Uno de los inconvenientes de TCP New Reno y TCP Reno es que incurren en múltiples retransmisiones rápidas innecesarias puesto que el transmisor no puede identificar un reconocimiento duplicado generado por un segmento en fuera de orden como producto de una retransmisión que ya ha sido recibido por el receptor [19], lo que degrada el rendimiento de TCP.

### 2.1.3. TCP Sack.

TCP SACK se desarrolló para contrarrestar los problemas presentados por TCP Tahoe<sup>1</sup> y TCP Reno, donde experimentan múltiples pérdidas en una misma ventana, ya que su

---

<sup>1</sup> TCP Tahoe es la versión estándar definida en el RFC 793.

implementación en el transmisor sólo les permite conocer la pérdida de un segmento por RTT y se consideraba una información limitada para su recuperación [11]. Con TCP SACK, el receptor puede informar al transmisor sobre todos los segmentos que han sido recibidos con éxito, y de esta manera el transmisor sólo transmite los paquetes perdidos. El mecanismo de TCP SACK se encuentra definido en el RFC 2018 y extendido en el RFC 2883.

TCP SACK utiliza la opción SACK para mantener la información sobre el bloque de paquetes recibidos, de tal forma que identifica tanto el primer como el último paquete en ese bloque, esta opción le permite al transmisor conocer sobre el estado de la red, en muchas circunstancias se puede presentar la pérdida de más de un paquete en la misma ventana de congestión, por lo tanto la opción SACK envía la información sobre varios bloques según sea el número de secuencia de los paquetes perdidos [11].

TCP SACK entra en la etapa de retransmisión y recuperación rápida de igual forma que New Reno, y finaliza cuando todos los datos que estaban pendientes hayan sido recibidos, antes de entrar en la fase de retransmisión rápida.

Durante la fase de recuperación rápida, se utiliza la variable *pipe* que equivale al número de segmentos pendientes que se encuentra en la red, *pipe* se incrementa en una unidad cuando el transmisor envía un nuevo paquete o retransmite uno viejo y se decrementa en uno cuando recibe un ACK duplicado con la opción de SACK. Al momento de transmitir un paquete, se considera los reconocimientos anteriores de la opción SACK, que se encuentran disponibles en el transmisor como una tabla de datos, para poder retransmitir el siguiente paquete que se deduce perdido por el receptor. Se puede presentar el caso en que un segmento ya retransmitido se pierda, por lo que el mecanismo detecta la pérdida por *timeout*, se retransmite el segmento perdido y comienza nuevamente con el inicio lento [15, 20]. A continuación se explica cómo está conformada la opción SACK.

Debido a las condiciones de la red se pierden segmentos o simplemente llegan en desorden, es necesario que el receptor mantenga bloques de datos no contiguos en su cola. La opción SACK utiliza dos enteros de 32 bits no señalados para representar un bloque no contiguo de datos continuos; el "Borde izquierdo de un bloque" es el primer número de secuencia de este bloque, "el Borde derecho del bloque" es el número de secuencia que sigue inmediatamente al último número de secuencia de este bloque. Hasta 4 bloques SACK pueden ser transportados por un paquete de reconocimiento; ya que SACK requiere para su uso 8 bytes, sin embargo depende si otras opciones de TCP lo utilizan como es el caso *timestamp* [21].

La opción *SACK-permitted* es una de las dos opciones que permiten el reconocimiento selectivo, que se envía mediante un segmento SYN, indicando que la opción SACK puede utilizarse cuando se establece la conexión, para que de esta forma habilite al receptor para generar opciones SACK propias. (Tener en cuenta que la elección por parte del receptor de la opción SACK, debe aplicar unas reglas, las cuales se explican detalladamente en [22].

Según [22], el envío de las opciones SACK deben ser incluidas en todos los ACKs que no notifiquen el número de secuencia más alto de la cola del receptor de datos, esta situación se presenta cuando se pierde o simplemente se han desordenado los datos, lo que conlleva a que el receptor mantenga datos no continuos en su cola.

Con respecto al manejo de TCP SACK, cuando el transmisor recibe un ACK parcial, tiene un manejo especial, para estos reconocimientos el transmisor decrementa la variable *pipe* en dos

paquetes lo que implica que la red bajo estas circunstancias no logra recuperarse más rápido que en la fase de inicio lento [15, 20]. Un tema que no es abordado por el RFC 2018 es la opción de SACK cuando un segmento duplicado es reconocido.

Por último es bueno recalcar que el buen desempeño de TCP SACK depende de la disponibilidad de ancho de banda que tenga la red, y que su incorporación no modifica el algoritmo de control de congestión de TCP Reno.

#### 2.1.4. TCP Hybla.

TCP Hybla es un mecanismo propenso para mejorar el desempeño de las redes satelitales, donde los altos RTTs no permiten un crecimiento adecuado del tamaño de ventana y se presentan múltiples pérdidas en una misma ventana de congestión [23, 24]. El objetivo principal de TCP Hybla es eliminar la dependencia del RTT en la actualización de la ventana de congestión. Si se observan los mecanismos de control de congestión, inicio lento y evasión de congestión, la tasa de crecimiento de la ventana se hace de una forma exponencial y lineal cada RTT dependiendo el valor del *ssthresh*, lo que significa que para un RTT más alto como el que se presenta en las redes GEO [24, 25], su tasa de crecimiento no será igual a uno o menor, ya que tardaría más tiempo en obtenerlo. Para contrarrestar el hecho que para un mayor RTT se ve degradado su rendimiento, las conexiones deben tener la misma tasa de transmisión instantánea,  $B(t)$  de una conexión relativamente rápida de referencia, para cumplir este objetivo  $W(t)$  debe ser independiente del RTT y segundo se debe compensar el efecto de la división por RTT. A continuación se hace referencia a la ecuación 13 que indica la tasa de transmisión, expresado en segmentos por segundo, donde  $W(t)$  es la evolución de la ventana de congestión en el tiempo, en unidades SMSs.

$$B(t) = W(t)/RTT \quad (13)$$

TCP Hybla incorpora la variable  $\rho = RTT/RTT_0$ , que es un RTT normalizado, donde  $RTT_0$  es el RTT de la conexión de referencia al cual se quiere igualar su desempeño. Con el fin de no disminuir la conexión que presente un  $RTT < RTT_0$ , TCP Hybla fija a  $\rho$  en un valor mínimo de 1, y de esta manera se logra aprovechar la estimación del ancho de banda. A continuación se indican las ecuaciones de los mecanismos de control de congestión para *cwnd*, en las fases de inicio lento y evasión de congestión, al ajustar los cambios anteriormente nombrados resulta la ecuación 14.

$$cwnd = \begin{cases} cwnd + 2^\rho - 1, & \text{Inicio Lento} \\ cwnd + \rho^2/cwnd, & \text{Evasión de Congestión} \end{cases} \quad (14)$$

El resultado del mecanismo de actualización de la ventana de congestión de TCP en conexiones con largo RTT, es un promedio más grande de *cwnd*, como lo muestra la ecuación 14, como consecuencia de ello se presentarán múltiples pérdidas en la ventana de congestión, en estas situaciones TCP Hybla sugiere el uso de la opción SACK, que le permitirá al transmisor saber cuáles paquetes han sido enviados con éxito, y recuperarse más rápido de una eventual pérdida de segmentos; ya que podrá transmitir más de un paquete por RTT, la opción *timestamp* es recomendado por TCP Hybla para hacer un seguimiento a los paquetes,

con el fin de ajustar el RTO debido a que se ve alterado con una mayor frecuencia al utilizar tamaños de ventanas más grandes [25].

TCP Hybla hace una mejora a la estimación del umbral del inicio lento, el valor del *ssthresh* a utilizarse no debe ser tan bajo para que no cambie demasiado pronto a la fase de inicio lento y poder aprovechar así el ancho de banda de la red, como tampoco tan alto para no verse afectado por múltiples pérdidas debido al aumento de los paquetes transmitidos en la fase de inicio lento; para solucionar estos inconvenientes TCP Hybla utiliza la estimación del *ssthresh* de extremo a extremo de Hoe [26], el cual se basa en obtener un valor del *ssthresh* a un valor cercano al producto por retardo de ancho de banda, para obtenerlo se utiliza la opción de *timestamp* que observa los retardos de propagación de los ACKs que llegan al transmisor para segmentos transmitidos consecutivamente, y así adquirir un valor apropiado para este [26,27]. Otra mejora que incorpora TCP Hybla está relacionado con el valor asignado de *cwnd* en la fase de evasión de congestión, el cual es muy alto al finalizar esta fase, para ello adopta la opción fija de recuperación, que se encarga de ajustar *cwnd* igual al valor mínimo del *ssthresh* y el actual número de paquetes que se encuentre en transición [27]. Otro inconveniente que genera aumentar el *cwnd*, es el aumento del tráfico en ráfagas, para contrarrestarlo, TCP Hybla utiliza las técnicas de espaciamiento de paquetes.

#### 2.1.5. TCP Westwood.

Es un mecanismo que realiza modificaciones únicamente del lado de la fuente de TCP New Reno, donde modifica el algoritmo de recuperación rápida con el propósito que no se vea reducido a la mitad el umbral del inicio cuando ha ocurrido la pérdida de un segmento; TCP Westwood (TCPW) se basa en la incorporación de un algoritmo dinámico que permite deducir el estado de la red a partir de los ACKs recibidos, de esta manera el umbral de inicio lento se ajusta a la tasa de datos del RTT estimado [11, 28].

TCPW hace uso del ancho de banda estimado para adquirir una mejor utilidad del ancho de banda y además lo alcanza sin perjudicar a otras conexiones, su implementación requiere hacer modificaciones sólo en el lado del transmisor.

En el transmisor se realiza un monitoreo continuo de los ACKs a partir del receptor para calcular la estimación de la tasa elegible ERE y con este valor ajustar el valor del *ssthresh* como el de *cwnd* después de la pérdida de un segmento [29, 30]. Debido a las condiciones cambiantes de la red, donde algunos paquetes pueden presentar un mayor retardo, es indispensable emplear filtros de precisión donde se tengan en cuenta tanto la estimación de los paquetes actuales como sus anteriores, para ello se utiliza el teorema de *Nyquist* que elimina las estimaciones que se encuentran muy alejadas de los intervalos de tiempo, considerando que la congestión ocurre cuando las componentes de baja frecuencia del ancho de banda superan la capacidad disponible del enlace de cuello de botella, luego se requiere un filtro pasa bajo que promedie estas muestras, además con ello ayuda a filtrar el ruido producido por el retardo de los paquetes [11, 31].

Al ocurrir la pérdida de un paquete ya sea por la recepción de 3 ACKs duplicados o la expiración del *timeout*, el transmisor establece *cwnd* y el *ssthresh* de la siguiente manera [11, 29]:

```

if ( 3 ACKs duplicados son recibidos)
    ssthresh=(ERE * RTTmin) /seg_size;
    if (cwnd>ssthresh)
        cwnd=ssthresh;
    endif
endif
endif

if (timeout ordinario expira)
    cwnd=1;
    ssthresh=(ERE * RTTmin)/seg_size;
    if (ssthresh<2)
        ssthresh=2;
    endif
endif
endif

```

Donde ERE, es igual a la tasa de datos que satisfactoriamente han sido entregados por el receptor,  $RTT_{min}$  es el mínimo valor RTT de todos los paquetes reconocidos (se utiliza  $RTT_{min}$  para que la fuente tenga como referencia el tiempo necesario para vaciar los segmentos después de la congestión), y  $seg\_size$  es el tamaño de un segmento TCP en bytes. A continuación se describe como es el procedimiento para el cálculo ERE.

Para el cálculo de la tasa de estimación del ancho de banda se requiere la información de los tiempo de llegada como también del incremento de los datos entregados a su destino, la muestra del ancho de banda se calcula como  $b_k = d_k / (t_k - t_{k-1})$ , donde  $t_{k-1}$  es el tiempo en que el anterior ACK fue recibido,  $d_k$  son los octetos que se han recibido por el receptor, si  $\Delta t_k = t_k - t_{k-1}$ , entonces  $b_k = d_k / \Delta t_k$ ; y de esta manera se realiza el cálculo cada vez que el transmisor recibe un ACK la ecuación 15 indica la manera de cómo se calcula las muestras del ancho de banda en TCPW:

$$b_k = \frac{2T - \Delta t_k}{2T + \Delta t_k} b_{k-1} + \frac{\Delta t_k}{2T + \Delta t_k} (b_k + b_{k-1}) \quad (15)$$

Donde T es la constante del filtro, según se aprecia la ecuación 15 se puede dar cuenta que el valor de la muestra  $b_{k-1}$  tiene un menor impacto si el valor  $\Delta t_k$  incrementa y por el contrario se tiene un menor impacto cuando la red no se encuentra congestionada [28, 30, 31].

### 2.1.6. TCP Westwood+.

TCP Westwood+ intenta mejorar la estimación del ancho de banda disponible en la red cuando se encuentra en presencia de ACKs comprimidos, estos se acumulan generando retardos, que generalmente ocurren cuando la ruta del ACK está congestionada. La forma de calcular el ancho de banda disponible en el momento de la congestión es por cada RTT usando todos los datos reconocidos en un RTT, y no como lo implementa TCPW donde calcula una muestra por ACK recibido, de esta manera TCPW+ evita los posibles errores de los ACKs comprimidos sin perjudicar innecesariamente a  $cwnd$  [32, 33].

Las consecuencias de los ACKs comprimidos es que sobreestiman el ancho de banda, donde las muestras medidas contienen componentes de alta frecuencia que son descartadas en un filtro pasa bajo (ver figuras 1 y 2 en [33]), lo que podría afectar la equidad de las conexiones. Para promediar las muestras del RTT se utiliza un filtro exponencial y para las muestras un filtro pasa bajo. En el momento que ocurre una pérdida TCP Westwood+ ajusta  $cwnd$  a un valor que coincida con el ancho de banda disponible en el momento de la congestión. A continuación se indica la lógica del algoritmo ante la pérdida de un segmento y tras la recepción de un ACK [33, 34].



- a) Al recibir un ACK, la ventana de congestión se comporta de igual forma que New Reno.
- b) Cuando 3 ACKs duplicados son recibidos, se toma en cuenta el valor del ancho de banda estimado en la red al momento de la pérdida.

$$ssthresh = \max [2, (BWE * RTTmin/seg\_size)];$$

$$cwnd = ssthresh;$$

- c) Cuando expira el *timeout*:

$$ssthresh = \max [2, (BWE * RTTmin/seg\_size)];$$

$$cwnd = 1;$$

Para el cálculo de la muestra del ancho de banda,  $b_k$  es igual a la relación entre la cantidad de datos reconocidos durante el último RTT, es decir  $D_k$  y RTT, por lo tanto,  $b_k = D_k / RTT$ , y con ello se filtran las componentes de alta frecuencia producidas por el ACK comprimido, que es equivalente a tener un filtro anti *aliasing*. Para este mecanismo el valor del filtro cumple con el teorema de *Nyquist*,  $RTT \leq T_f / 2$  [32].

### 2.1.7. TCP HighSpeed.

TCP HighSpeed es una de las versiones de TCP que fue modificada para trabajar en redes con un gran producto por retardo de ancho de banda características de las redes de alta velocidad, teniendo en cuenta la versión base como es TCP Reno, su objetivo principal es hacer un mejor uso del ancho de banda de la red sin alcanzar un gran número de pérdidas debido al incremento de su ventana de congestión, para ello modifican la función de respuesta de la ventana de congestión teniendo en cuenta el número de ACKs que llegan al transmisor y el número de paquetes perdidos en la red [35, 36].

TCP HS define tres parámetros para determinar cuál será la función de respuesta de la red, el primero de ellos es *High\_P* que indica la tasa de error de la conexión ajustada a un valor de  $10^{-7}$  que corresponde a un tamaño de ventana de 83000 segmentos, el parámetro *Low\_Window* indica el umbral inferior de la función de respuesta de la ventana de congestión, cuando *cwnd* presenta un rendimiento comparable a la versión estándar de TCP, *Low\_Window* se configura a una tasa de pérdida de  $10^{-3}$ , el último parámetro es *High\_Window* su valor por defecto se establece según su tasa de pérdida para este caso es *High\_P* como lo indica la figura 2-1.

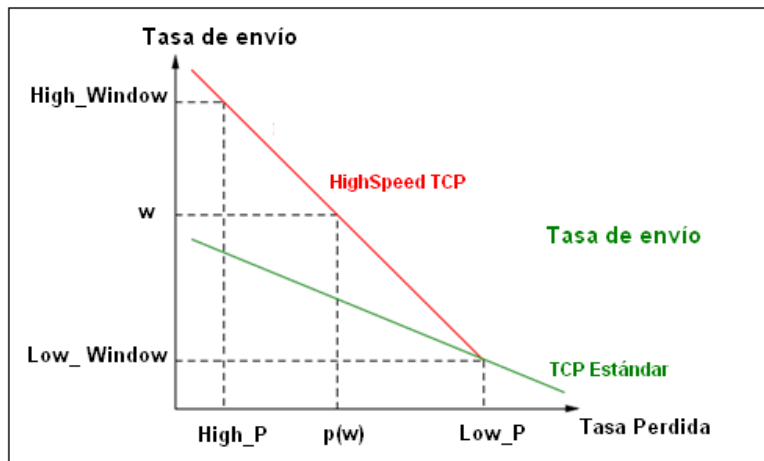


Figura 2-1. Función de respuesta de *cwnd* según su tasa de pérdida [37, 38].

La función de respuesta de la ventana de congestión se incrementa o se decrementa según sea la recepción de ACKs o la pérdida de paquetes respectivamente como se indica en la ecuación 16, si el transmisor observa que el número de ACKs se incrementa, este utiliza un incremento de ventana de congestión más agresiva que le permita utilizar mejor el ancho de banda de la red, igualmente sucede con las pérdidas si estas no son muchas la ventana disminuye más lentamente; en las ecuaciones 16 y 17 se indica el incremento o decremento de *cwnd* por TCP HS [37, 38].

$$\text{ACK: } cwnd = cwnd + a(cwnd)/cwnd \quad (16)$$

$$\text{Drop: } cwnd = cwnd - b(cwnd) \times cwnd \quad (17)$$

Los valores de *a* y *b* se ajustan a 1 y 0.5 respectivamente.

En el anexo B se describen otras versiones de TCP que modifican los mecanismos de control de congestión, que en ciertas condiciones de la red podrían mejorar el desempeño de TCP.

## **2.2. MODIFICACIÓN A LA ARQUITECTURA INTRODUCIENDO NODOS INTERMEDIOS EN LA RED E IMPLEMENTACIONES ENTRE LOS NODOS ESTÁNDAR O SOLUCIONES PROPIETARIAS.**

### **2.2.1. PEP.**

En esta sección se hablará de otros tipos de soluciones conocidos como PEPs que hacen uso de *proxies* que mejoran el rendimiento. El concepto de PEP se emplea para compensar las características degradantes específicas a las que pertenece un enlace o una subred en su trayectoria, su principal objetivo es poder aislar el segmento terreno del satelital para implementar procesos o técnicas que permitan mejorar su desempeño en el segmento satelital, en este tipo de soluciones se encuentran los protocolos de Internet y entre ellos el protocolo TCP. Específicamente en este trabajo la optimización de TCP en redes satelitales. Hay muchas soluciones PEPs que al implementarse siguen manteniendo el mismo *stack* del protocolo, modificando algunos aspectos en el transmisor y/o en el receptor para su funcionamiento, sin embargo existen otras técnicas PEP que modifican todo el *stack* del protocolo, estas soluciones están más allá del propósito del documento [23, 39].

Específicamente las soluciones PEP en entornos satelitales se encargan de aislar la alta latencia y el gran producto por retardo por ancho de banda que los enlaces por satélite presentan; en realidad esta solución hace un mayor énfasis en una mejora de la arquitectura, en lugar de una mejora del funcionamiento del protocolo. Los PEPs se encuentran documentados en el RFC 3135 [40].

### **2.2.2. Tipos de PEPs.**

Dentro de los tipos de PEPs existen muchas soluciones para mejorar diferentes características del enlace, las más utilizadas que obtienen un mejor desempeño en la red son los PEPs de la capa de enlace y los PEPs de la capa de transporte. En general los PEPs pueden desempeñarse en muchas capas del protocolo, pero generalmente funcionan en una o dos capas.

Los PEPs de la capa de transporte hacen uso de los *proxies* para mejorar el desempeño TCP, es decir, son mecanismos que interactúan con los nodos intermedios y la capa TCP, conocidos como TCP PEP [40]. El propósito de estos mecanismos es mejorar el algoritmo de inicio lento, para que su ventana crezca rápidamente y así aprovechar el ancho de banda de la red, compensando el gran producto por retardo por ancho de banda que se presenta en estas redes; para ello se generan reconocimientos locales los cuales son interceptados por las pasarelas. Al hacer uso de la información de TCP, no se está alterando la configuración del protocolo y más bien se busca que su funcionamiento se haga de una forma transparente a los usuarios [23, 40]. Las dos técnicas PEPs más utilizadas son TCP *Spoofing* y TCP *Splitting*.

Uno de los principales inconvenientes de estas técnicas es su uso conjunto con IPsec, e intentar no violar la semántica de extremo a extremo para mantener la confiabilidad y seguridad de TCP como se explicará más adelante, las principales características que presentan los PEPs se describen en el anexo B.

### **2.2.3. Implementaciones PEP en la Red VSAT.**

En [40], explican el comportamiento de PEP dentro de cada uno de los entornos de las redes LAN, WAN y satelitales, dentro de las redes satelitales hacen un mayor énfasis en las redes VSAT, las cuales se caracterizan por ser empleadas en redes GEO y utilizar una topología en estrella con un concentrador (*hub*), el cual se encarga de recibir y transmitir la señal a las demás VSATs. Se emplean en redes asimétricas y suelen usarse en redes privadas que se encuentran dispersas geográficamente de su estación central, las cuales necesitan intercambiar información.

Los PEPs empleados en las redes VSAT ayudan a mejorar los servicios de FTP y HTTP, además pueden optimizar el rendimiento de TCP, hacen uso de una implementación asimétrica distribuida donde utilizan una conexión *split* (dividida) con reconocimientos y retransmisiones locales. Los inconvenientes de usar PEP en las redes VSAT, es que impiden el uso de IPsec, por lo que requiere implementar niveles de seguridad en otras capas, por ejemplo TLS o podrían implementar algún mecanismo propietario.

### **2.2.4. Soluciones PEP para Mejorar el Rendimiento del Protocolo TCP.**

Dentro de los mecanismos PEPs más implementados se encuentran las soluciones TCP *Splitting*, *Web Caching* y TCP *Spoofing*, sin embargo la solución TCP *Spoofing* es el único que pertenece a la capa de enlace de datos y se explicará más adelante.

## **2.3. MECANISMOS DEL NIVEL DE ENLACE DE DATOS.**

### **2.3.1. Proxy Spoofing.**

Es una técnica PEP, que hace uso de un agente en el medio cerca de la conexión por satélite, que se encarga de suplantar al receptor, de esta manera la pasarela reconoce los paquetes de datos a nivel local a la máxima tasa de envío como si la comunicación fuera a nivel LAN y dar la ilusión que la red presenta un corto retardo. Como se indica en la figura 2-2, la pasarela tiene la

responsabilidad de almacenar todos estos segmentos en su búfer para transmitir los datos al enlace satelital, además de eliminar los ACKs que recibe por parte del receptor, también tiene la responsabilidad de retransmitir la pérdida de los paquetes perdidos entre la pasarela y el receptor real [41, 42].

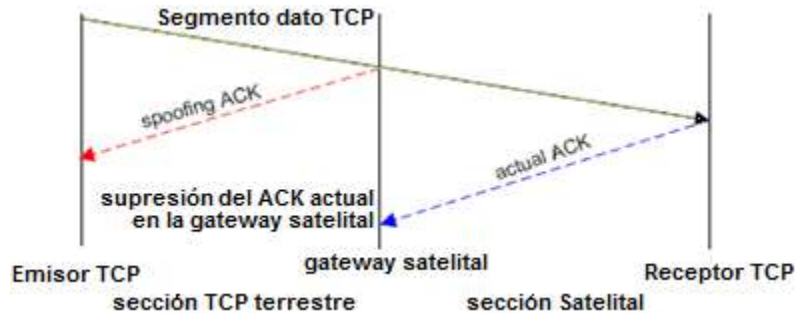


Figura 2-2. Proceso del ACK en las secciones de la red TCP [43].

En la figura 2-3 se indica la estructura empleada por una técnica de TCP *Spoofing* donde se observa que el segmento que está más cerca a la pasarela satelital es aislada de la alta latencia del enlace satelital; es importante tener en cuenta que este mecanismo hace un reconocimiento prematuro de los datos, además en el segmento satelital no se utiliza una versión de TCP optimizado y desde luego rompe la semántica de extremo a extremo de TCP.

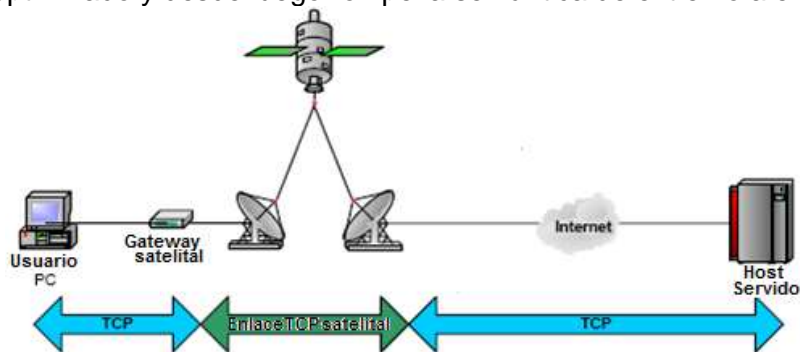


Figura 2-3. Estructura física de TCP Spoofing [41].

Los inconvenientes de TCP *Spoofing* tienen que ver que con el tamaño del búfer, ya que estos deben tener una gran capacidad de procesamiento y de almacenamiento para asumir el trabajo de reconocimiento de los datos del transmisor, en segundo lugar esta técnica no se puede emplear en redes asimétricas, ya que requieren utilizar la misma trayectoria para el flujo de datos TCP y los ACKs, un tercer problema ocurre cuando hay un cambio de ruta o simplemente el agente intermedio (por lo general una *pasarela* satelital) se bloquea, ocasionando una pérdida de datos, y por último al utilizar una técnica PEP este requiere leer y modificar el encabezado TCP, sin embargo si se utiliza IPsec este esquema no se podría aplicar [6, 44].

### 2.3.1.1. Proxy snoop.

El protocolo Snoop, es una solución de la capa de enlace que se clasifica como TCP inconsciente, que se caracteriza por emplear técnicas FEC y ARQ para detectar y retransmitir los paquetes perdidos. Es una solución híbrida que emplea los conceptos de la capa de enlace

y el esquema de conexión dividida de PEP; fue desarrollada para resolver los problemas de rendimiento TCP causado por los errores de bit en entornos inalámbricos [16, 45]. El protocolo snoop, utiliza un agente snoop en la estación base que consta de dos módulos, el primer módulo es snoop data() que es el encargado de monitorear todos los paquetes que son enrutados en un enlace inalámbrico donde se registra el último número de secuencia transmitido, también se responsabiliza de almacenar en cache los segmentos de datos de las unidades móviles que aún no han sido reconocidos por el transmisor, el segundo módulo se llama snoop ack(), su función es la de monitorear los ACKs que han sido enviados desde el receptor TCP al transmisor. Como se observa en la figura 2-4.a, la llegada de un reconocimiento puede desencadenar en distintas operaciones por el agente, si el ACK tiene un número de secuencia más alto que el número de secuencia reconocido anteriormente, el agente limpia el búfer, actualiza su RTT y envía los reconocimientos al transmisor, pero en caso contrario se estaría hablando de un reconocimiento duplicado (del anterior ACK), el agente averigua si tiene una copia en el cache local, si aquella está disponible se retransmite, y cualquier paquete le informa al transmisor para que no retransmita el paquete, además se realiza una estimación del número máximo de ACKs duplicados que existían cuando fueron transmitidos entre la pérdida de los paquetes y su retransmisión, estos reconocimientos de regreso a la red inalámbrica del host son descartados para evitar una interpretación incorrecta de congestión, en caso de que no esté disponible en cache debe ser retransmitido por el emisor [46, 47].

El agente snoop también utiliza las retransmisiones basadas en el *timeout*, snoop se basa en el RTT local estimado por la conexión inalámbrica de cada paquete donde la estación base incorpora un temporizador más fino que en el transmisor TCP para realizar las retransmisiones a nivel local.

Este protocolo es un ejemplo de PEP asimétrica, opera de una manera diferente para el flujo de datos TCP y para el flujo ACK, de hecho, este protocolo se diseñó específicamente para transferencias desde el *host* fijo al móvil, para el enlace entre el receptor y la estación base se pueden utilizar NACK con la opción SACK habilitada o ELN para informar a la estación base que la pérdida de paquetes no estaba relacionada con la congestión, si se emplea ELN se utilizan uno de los bits no reservados en el encabezado TCP [16, 48].



Figura 2-4.a Modulo snoop ack() [18].

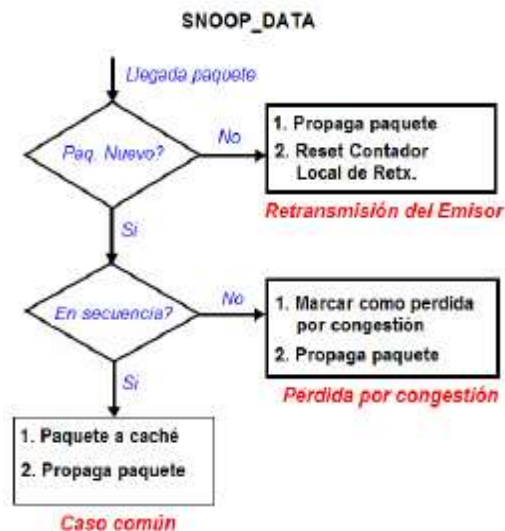


Figura 2-4.b Modulo snoop data() [18].

Su principal ventaja está en diferenciar con certeza entre las pérdidas de congestión y las pérdidas por corrupción, lo que conlleva a que la ventana de congestión no sea reducida por los mecanismos de control de congestión, a nivel de las aplicaciones, este esquema es realmente atractivo para el uso de la red porque mantiene la semántica de extremo a extremo de TCP. Sus desventajas se relacionan a que deben emplear grandes búferes para almacenar su información y su incompatibilidad con IPsec.

Esta solución es importante, a pesar de que no fue pensando para su uso en las redes satelitales, se han hecho estudios sobre su desempeño sobre las redes GEO [7, 41], debido a que las redes inalámbricas comparten algunas características con los enlaces satelitales, estos sistemas presentan pérdidas considerables por corrupción y anchos de bandas asimétricos, su principal diferencia está en el retardo de propagación como el uso de *handoff* en la red inalámbrica [49]. El protocolo snoop en redes satelitales utiliza el proxy snoop, el cual se ubica en las pasarelas de la estación terrena (GES) conectándose con la LAN. El PEP snoop es el encargado de recibir los reconocimientos locales y de retransmitir los segmentos perdidos a la red satelital, al utilizar los reconocimientos de TCP se activa la retransmisión de la capa de enlace a GES y se eliminan los ACKs duplicados que se propagan al transmisor. Su implementación requiere modificaciones en los nodos intermedios y búferes grandes para el almacenamiento en cache de la información.

### **2.3.2. Técnicas de Acceso al Medio.**

En los sistemas satelitales es muy importante el uso eficiente de los recursos limitados que presenta la red (potencia y ancho de banda), las cuales deben ser compartidos por un gran número de usuarios no coordinados que acceden al sistema de forma corta e impulsiva; para que no ocurra ningún tipo de conflicto entre las señales que llegan simultáneamente al satélite, se establece un orden mediante una técnica de acceso múltiple el cual se divide en tres tipos principales que son: por división de frecuencia, por división de tiempo y por división de código. El acceso múltiple es una variante de multiplexado y describe el procedimiento a usar para compartir los recursos de comunicaciones entre un gran número de usuarios.

El protocolo de acceso múltiple que se use afecta a todos los elementos del sistema, determina su capacidad, flexibilidad, y tiene una gran incidencia en los costos, por lo tanto deben considerarse los diversos protocolos de acceso múltiple disponibles en la fabricación de las estaciones terrenas, así como los métodos utilizados para el análisis comparativo de sus prestaciones.

#### **2.3.2.1. ACCESO MÚLTIPLE POR DIVISIÓN DE FRECUENCIA.**

El ancho de banda total de un satélite es de 500 MHz, el cual se divide en varios transpondedores, y que una forma usual de hacerlo es con ranuras de 36 MHz, significa que el amplificador de cada transpondedor puede darle cabida a una gran diversidad de información que ocupe en total un ancho de banda de 36 MHz. Sin embargo, cada estación terrena que transmite desde la Tierra no tiene necesariamente el suficiente tráfico para generar información que ocupe todo ese ancho de banda y que puede enviarse con una sola frecuencia portadora determinada; supóngase tres regiones A, B y C en donde la demanda de la región A es mayor que la región B y que la región C es una zona rural y todas desean transmitir al mismo tiempo, estas deben hacerlo con frecuencias portadores diferentes para que no se interfieran entre sí; si

la suma del ancho de banda de las tres estaciones es equivalente o cercana al ancho de banda del transpondedor tal y como lo indica la figura 2-5 separadas por bandas de guarda, a esta forma de uso del transpondedor por varias estaciones terrenas estén o no situadas en una misma región se conoce con el nombre de acceso múltiple por división de frecuencia (FDMA) de esta manera el espectro radioeléctrico del transpondedor fue dividido en ranuras de frecuencia para cada estación terrena [50, 51].

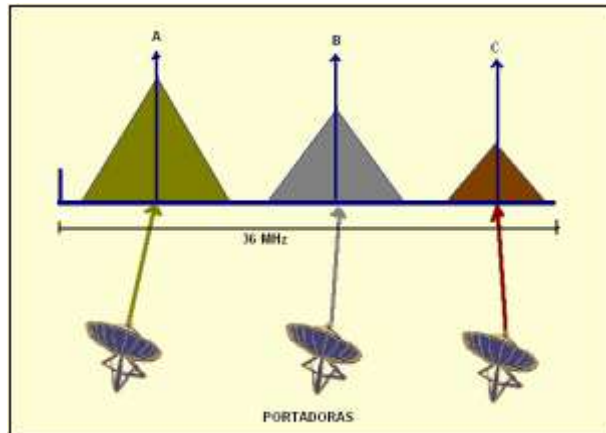


Figura 2-5. Ocupación de un transpondedor de 36 MHz con FDMA; cada señal proviene de una población diferente y tiene su propia portadora asignada [50].

Cuando el número de portadoras aumenta, el ancho de banda de cada una de estas disminuye lo que implica una reducción de la capacidad de las mismas; como la capacidad total es la suma de las capacidades de las portadoras, se pensaría que la capacidad debería mantenerse sensiblemente constante; para reducir la intermodulación, se debe aumentar el *backoff* lo cual hace que haya una reducción sensible de la capacidad, en la figura 2-6 se muestra dicha relación [52].

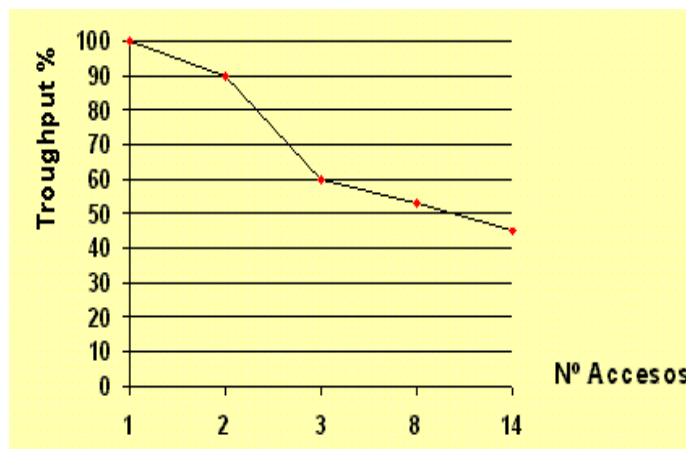


Figura 2-6. Rendimiento de FDMA [52].

### 2.3.2.1.1. ACCESO MÚLTIPLE CON DIVISIÓN DE FRECUENCIA CON ASIGNACIÓN FIJA.

Esta clase de acceso al medio en red satelital se utiliza principalmente en sistemas comerciales de alta capacidad; la configuración de este tipo de acceso al medio es rígida e invariable, es

decir cada estación terrena debe transmitir siempre con la misma frecuencia central o portadora, esto es viable siempre y cuando se garantice que durante la mayor parte del tiempo cada una de ellas ocupara el ancho de banda que se le fue asignado [51].

### **2.3.2.1.2. ACCESO MÚLTIPLE POR DIVISIÓN EN FRECUENCIA CON ASIGNACIÓN POR DEMANDA (DAMA).**

Cuando el tráfico generado por las diferentes estaciones terrenas que comparten un transpondedor no es el adecuado para tener una asignación fija y por el contrario es un tráfico esporádico e intermitente (tipo ráfaga), la capacidad de dicho transpondedor no es aprovechada de manera eficiente con la técnica nombrada anteriormente, para ello recurre a la técnica DAMA el cual consiste en que a cada estación terrena se le da acceso al canal sólo cuando esta requiera transmitir. La frecuencia de la portadora transmitida por cada estación terrena cambia con el tiempo, moviéndose de lugar en el espectro radioeléctrico del amplificador, por ello la estación debe estar debidamente equipada para poder hacerlo [52, 53].

La técnica DAMA aprovecha al máximo las ranuras de frecuencia y la potencia del satélite cuando el tráfico es tipo ráfaga, pues las ranuras de frecuencia son asignadas a las estaciones terrenas solamente durante el tiempo que necesitan transmitir. Si la demanda pico del sistema iguala a la capacidad del sistema y si el tráfico es del tipo ráfaga, el sistema está la mayor parte del tiempo sin ser aprovechado totalmente. Sin embargo, por el uso de búferes y el procedimiento DAMA, un sistema con capacidad reducida puede manejar un tráfico del tipo ráfaga, a costa de retardos debidos al búfer. De esta forma, se utiliza un canal con una capacidad igual al promedio de los requerimientos de los usuarios [51]. Cada vez que una estación terrena desee transmitir, esta debe solicitar a una estación central que coordina el banco de frecuencias, que se le asigne una de ellas para su portadora, este mismo banco de frecuencias se comunica con el punto destino para informar que va a transmitir y en qué frecuencia debe sintonizarse para que reciba la señal; solamente hasta que la estación transmisora y la receptora hayan recibido la asignación de sus frecuencias de operación, se puede iniciar el enlace [50, 51].

Hay otra variante de un sistema FDMA llamado acceso múltiple con asignación por demanda con canal único por portadora modulado en PCM (SPADE) el cual se describe en el anexo B.

### **2.3.2.2. ACCESO MÚLTIPLE POR DIVISIÓN DE TIEMPO (TDMA).**

TDMA es una técnica que inicialmente fue definida en el estándar IS-54 y actualmente especificado en la serie IS-13x con recomendaciones de la EIA/TIA. Esta técnica consiste en compartir la capacidad de una sola portadora por todas las estaciones de una red mediante turnos de tiempos (slot de tiempos), en forma tal que el tráfico de entrada y de salida de cada una pueda ser continuo.

TDMA es una técnica totalmente digital, por lo cual es aplicable a portadoras digitales y es ideal para evitar las limitaciones de FDMA originadas por los efectos de intermodulación; lo que se logra con SCPT, en esta circunstancia TDMA tiene una gran ventaja, el transpondedor puede trabajar a su potencia máxima en la región de saturación permitiendo un aumento en la PIRE descendente y la eliminación del ruido de intermodulación, lo que hace posible aumentar la capacidad de canales mediante los recursos de intercambio entre potencia y ancho de banda [50, 51].



Cada estación de una red TDMA transmite con la misma frecuencia pero en ráfagas separadas por pequeñas guardas de tiempo para evitar que se superpongan debido a factores o variaciones de las condiciones de propagación; para que cada ráfaga se transmita en el momento preciso debe ajustarse la temporización a la distancia de cada estación al satélite. Para ello se debe aplicar métodos de sincronización en donde una de las estaciones terrenas de la red puede ser la de referencia para los demás, actuando también otra como estación de referencia de respaldo, conteniendo cada una el equipo terminal de referencia que realiza estas y otras funciones de control; de esta manera se evitaría el traslape de las ráfagas en el tiempo [8, 51, 54].

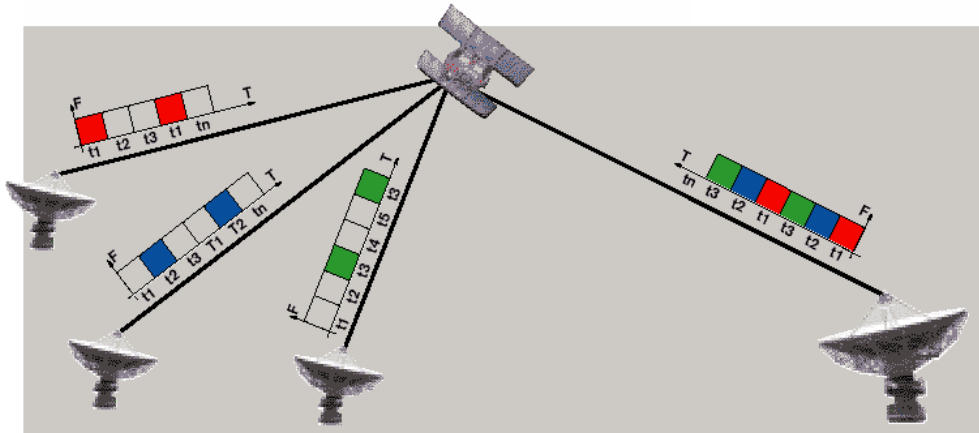


Figura 2-7. Transmisión de estaciones terrenas de tráfico y de referencia utilizando TDMA [51].

El tiempo  $T$  asignado a cada estación no es necesariamente igual en todos los casos, puesto que algunas estaciones conducen más tráfico que otras, por lo tanto, la ranura de tiempo que se les asigne debe ser más larga que la de las estaciones pequeñas. Estos tiempos asignados pueden ser fijos por estación, en cuyo caso se tiene acceso múltiple por división en el tiempo con asignación fija, o bien puede variar con el tiempo cuando algunas estaciones tengan exceso de tráfico (horas pico). En estas condiciones, es preciso reorganizar la distribución de los tiempos con una nueva estructura de trama de transmisión, dándole ranuras de tiempo más largas a las estaciones con exceso de tráfico y ranuras más cortas a las de poco tráfico; la nueva estructura de trama se repite secuencialmente hasta que haya la necesidad de hacer otro cambio [51, 52].

### 2.3.2.2.1. TDMA con asignación fija.

TDMA con asignación fija es una técnica sin contienda donde el acceso al medio es bastante simple, con este tipo de asignación cada estación terrena tiene una porción fija de su tráfico destinada a cada una de las demás estaciones con las que se quiere comunicar, luego los  $M$  slots que conforman la trama de la red son asignados a largo plazo a cada estación terrena. Esta técnica es ideal y bastante eficiente cuando el tráfico es predecible, continuo y no varía en el tiempo, la principal desventaja de esta técnica es que todos los recursos de la red se encuentran asignados de manera permanente y en el momento que una estación terrena ya no tenga datos que transmitir durante una trama, estos recursos se estarían desperdiciando ya que otra estación terrena no los podría utilizar [53, 54], para tratar de dar solución a este problema se procede a hacer una asignación bajo demanda que se explicara a continuación, en la figura 2-8 se muestra el esquema de asignación fija y por demanda.

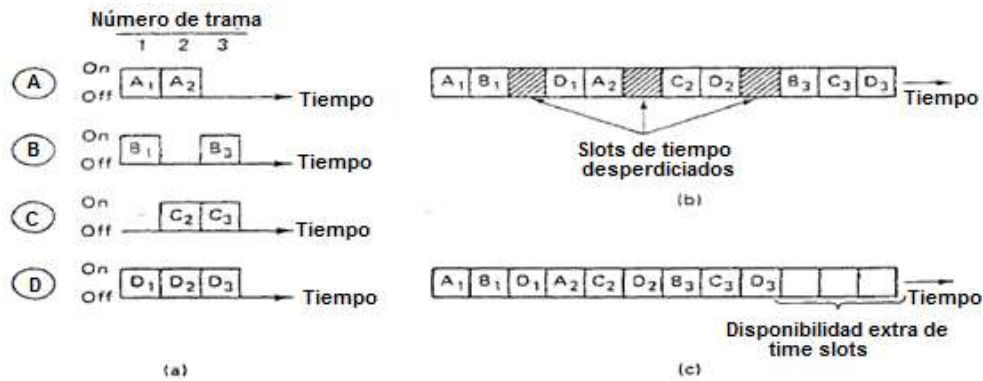


Figura 2-8. TDMA con asignación fija y variable [53].

En el anexo B se explica otra variante de TDMA llamada acceso múltiple por división de tiempo *satellite-swiched* (SS-TDMA) el cual es utilizada en satélites que tiene procesamiento a bordo.

#### 2.3.2.2.2. TDMA BAJO DEMANDA (DTDMA).

El sistema TDMA por asignación bajo demanda, se basa en un acceso al medio donde los recursos no son disputados y cada estación terrena debe transmitir en un instante determinado, los slots son asignados dinámicamente según sea la demanda del servicio y la disponibilidad de la red. Para ello se utiliza una estructura de trama que permita el sincronismo entre las estaciones terrenas y la estación de control, evitando así colisiones de los slots cuando las estaciones inician o terminan su transmisión. La trama está compuesta por dos tipos de subtramas, las ráfagas de tráfico (TB) que es la carga útil del sistema que transporta la información de los usuarios y la ráfaga de referencia (RB), esta se encuentra al inicio de cada trama y contiene la información de control, recuperación de portadora y sincronismo de tramas, en la figura 2-9 se muestra esta técnica de acceso.

Para garantizar el sincronismo en DTDMA, se utiliza UW que es la encargada de determinar la posición inicial de la ráfaga dentro de la trama y la posición de los bits en la ráfaga para cada estación terrena, se utiliza también los tiempos de guarda los cuales son variables para facilitar la sincronización de las tramas

El hacer una asignación bajo demanda de los recursos de la red, se está garantizando que todos los *slots* estén siendo utilizados por las diferentes estaciones terrenas, aumentando de esta manera el rendimiento de la red, mientras que para una asignación fija se presenta el caso donde los recursos son desperdiciados, es decir una estación terrena ya no tiene datos que transmitir pero conserva los slots asignados, bajando de esta manera el rendimiento de la red [54, 55].

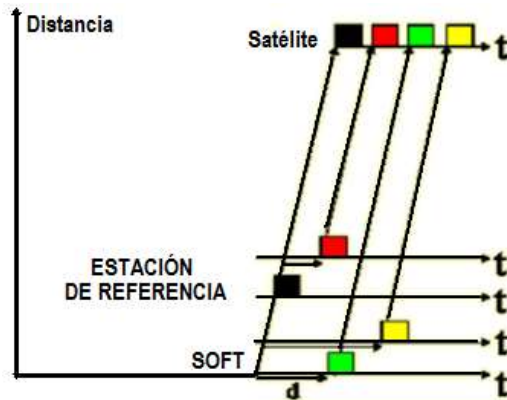


Figura 2-9. Esquema de sincronismo TDMA [55].

En [51] se describe el uso de esta técnica dentro de una red satelital, donde el objetivo principal del esquema es reducir el retardo extremo a extremo; aquí se utiliza para el enlace de subida DTDMA, en el cual cada intervalo de tiempo es asignado bajo demanda a cada una de las estaciones terrenas, con un mini intervalo de tiempo para que cada estación terrena haga su solicitud, de tal manera que el retardo de acceso sea menor que en un TDMA fijo, cuando el tráfico sea fijo o des-balanceado; para el enlace de bajada se utiliza TDMA asíncrono con el objetivo de reducir el retardo de transmisión; se utiliza un sistema de colas independientes como modelo para su análisis.

### 2.3.2.2.3. TDMA en multi-frecuencia.

MF-TDMA es una técnica de asignación que emplea múltiples portadoras y no solo una; cada portadora la divide en una serie de *slots* temporales que se asignan dinámicamente en función de la demanda. La asignación es realizada por la estación central, encargada de mantener el sincronismo general del sistema. Este sistema aumenta altamente la capacidad total al ser multiportadora. Es empleado como método de acceso en los terminales terrestres a los recursos del satélite como en el estándar DVB-RCS. En el enlace de retorno, el tiempo de transmisión hacia el satélite varía en función de la posición de transmisor, donde se puede producir diferencias temporales entre los *slots* vecinos que interrumpen y dañen la transmisión. Así esta diferencia entre los terminales que están en la huella del satélite, se compensa mediante el uso de *slots* más grandes, que las ráfagas transmitidas por los terminales, existiendo un tiempo de guarda lo suficientemente grande para prevenir con las ráfagas de los *slots* vecinos de la trama TDMA [54, 56]. En la figura 2-10 se indica el esquema MF-TDMA.

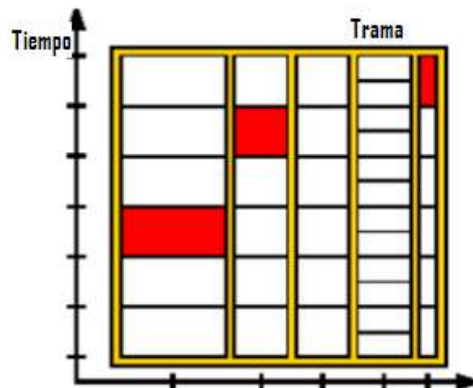


Figura 2-10. Esquema temporal de frecuencia en MF-TDMA [55].

#### **2.3.2.2.4. Acceso al medio FODA.**

Esta técnica es un esquema TDMA satelital adaptativo, el cual soporta tráfico de tiempo real, como tráfico que no es de tiempo real. FODA adopta una aproximación bastante conservadora en la asignación de ancho de banda, ya que se desea una mínima porción de pérdida de paquetes. La asignación del ancho de banda está controlada de forma centralizada por una estación que envía una ráfaga de referencia cada 20ms. Cada trama FODA se encuentra dividida en un flujo de subtramas, los cuales son asignados en un tiempo de establecimiento y un datagrama de subtrama; una cola FIFO para el flujo y dos colas FIFO para datagramas [7, 56].

#### **2.3.2.3. ACCESO MÚLTIPLE ALEATORIO (AMA).**

Esta técnica de acceso al medio, el acceso a un transpondedor por las estaciones de una red se realiza en forma de contienda por el uso de cada portadora, y puede no haber control centralizado de asignación permanente o temporal, o puede haber una forma simple de autocontrol de cada estación individualmente. Esta técnica no proporciona alta eficiencia en el uso de la capacidad de un transpondedor y es adecuada para redes donde este factor no es tan fundamental, es muy útil en redes VSAT interactivas en que el tráfico medio por estación sea muy bajo y en forma de paquetes.

En un sistema AMA puro cada estación terrena transmite ráfagas de bits de información en cualquier momento que ocasionalmente pueden interferirse con las de otros dificultando su recepción, por no contarse como en los sistemas CDMA con un código para extraer la información útil que se encuentra inmersa en la interferencia. En algunos casos cada estación puede recibir su propia señal con el tiempo de retraso de propagación (satélite) y compararla con la original para determinar si existe colisión con la señal de otra estación, en caso de haberla, la estación terrena retransmite la ráfaga de la misma forma hasta un cierto número de veces  $n_R$  con un tiempo de retardo aproximadamente de 0.27 segundos de la primera transmisión en el caso de los satélites GEO. En los sistemas AMA al aumentar el número de estaciones con tráfico, o el número de ráfagas por estación terrena, aumenta la probabilidad de que exista colisión en el primer intento de transmisión de un paquete de información y al repetir su transmisión se congestiona más el canal, hasta llegar a un punto donde se reduce el número total de paquetes transmitidos exitosamente por unidad de tiempo [50].

#### **2.3.2.3.1. Aloha puro.**

El funcionamiento de este protocolo es bastante sencillo, aquí cada estación terrena transmite sus datos en el momento que lo requiera, pero esto tiene una gran desventaja ya que la probabilidad de que dos paquetes colisionen es alta, sin embargo cada estación terrena vigila continuamente el canal de transmisión para verificar si sus datos han colisionado con los datos de otras estaciones; si un paquete colisiona, la estación terrena tiene que esperar un tiempo aleatorio para volver a transmitir el paquete colisionado ya que si no los paquetes retransmitidos estarían colisionando continuamente con los datos de otra estación terrena [53, 57]. En la figura 2-11 se muestra un ejemplo básico de este protocolo.

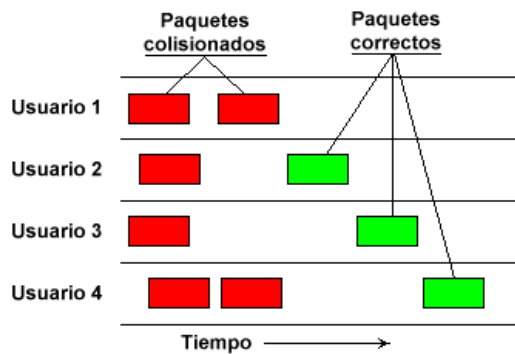


Figura 2-11. Flujo de paquetes en un canal ALOHA [58].

La relación entre el tráfico ofrecido  $G$ , y el rendimiento del protocolo viene dada por:

$$\varepsilon = Ge^{-2G} \quad (18)$$

El máximo rendimiento que se obtiene con ALOHA se da cuando  $G = 0.5$  y es de 0.18, o sea que cuando se alcanza la mejor utilización del canal ésta es de un 18%; no es un resultado muy bueno, pero la sencillez del protocolo hace que sea un protocolo muy atractivo y susceptible de ser mejorado [51, 53, 58].

En el anexo B se explica el funcionamiento de Aloha ranurado y con reserva.

### 2.3.3. Estándar DVB-RCS.

En las últimas décadas las redes satelitales basadas en el estándar DBV-S han tenido un gran auge, pero se han visto limitadas por las necesidades de los usuarios en especial con el gran auge de Internet, debido a que DBV-S utiliza un canal de retorno terrestre de baja calidad y no posee el control de este canal para hacer un control de calidad de servicio, razón por el cual se vio la necesidad de tener una comunicación bidireccional por satélite que fuera asequible al usuario común; surge entonces a principios del año 2000 el estándar de la ETSI 3001 790 para el canal de retorno DVB-RCS, el cual es un sistema abierto que permite tener aplicaciones completamente interactivas y bidireccionales de transmisión de paquetes mediante un sistema satelital [60, 61].

El estándar DVB-RCS fue creado con un objetivo muy claro que es el de cubrir varios tipos de mercados desde pequeños usuarios residenciales o comerciales hasta grandes empresas, donde se elimina la dependencia de las redes terrestres e incrementa las velocidades de transmisión de las estaciones terrenas gracias al sistema bidireccional que se tiene, permitiendo así la transmisión y recepción de datos haciendo uso del protocolo IP lo que da la posibilidad de transmitir voz sobre IP [60, 61].

En la figura 2-12 se muestra un sistema DVB-RCS, este estándar puede ser configurado para redes tipo estrella o tipo malla en donde está compuesto por un *hub*, múltiples RCSTs que pueden ser transmisoras y receptoras al tiempo y un satélite.

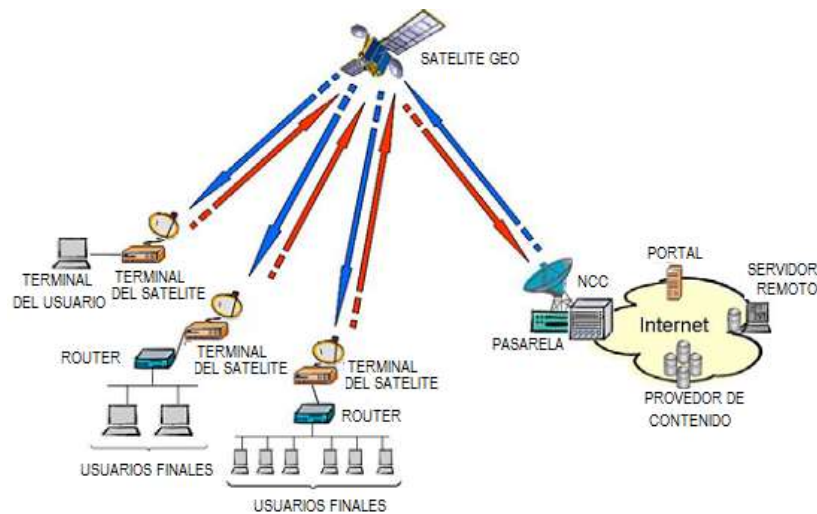


Figura 2-12. Sistema DVB-RCS [62].

Una red satelital basada en la norma DVB-RCS debe presentar dentro de su estructura los siguientes módulos:

- Una estación de control NCC.
- Una pasarela de tráfico.
- Un alimentador.
- Un satélite.

Cabe resaltar que las funcionalidades que se describieron para el estándar DVB-S en cuanto a la modulación, procesamiento de la señal y codificación de canal son aplicadas a esta norma también.

**El satélite:** Está conformado por varios transpondedores que pueden tener anchos de banda de 25, 32 y 76 MHz, el estándar permite al satélite operar dentro de las bandas Ku, C, Ka, para los cuales se tiene definido valores de frecuencia diferentes para el canal de subida y el canal de bajada.

**Estación de control NCC:** Esta se puede considerar como el núcleo de la red, ya que esta proporciona funcionalidades de control de acceso al medio, monitoreo y asignación de recursos a las diferentes estaciones terrenas, para ello la NCC emite señales de sincronización y control para el correcto funcionamiento de la red satelital [60].

**Pasarela de tráfico:** Esta es una estación terrena la cual le llegan todas las señales de retorno de las RCSTs, esta tiene la responsabilidad de hacer funciones de contabilidad, proveer servicios privados o propietarios interactivos que demanden los usuarios.

**El alimentador:** Esta es una estación terrena, la cual emite la señal para el enlace de subida basada en la norma DBV-S, que fue descrito anteriormente.

En una red satelital bajo la norma DVB-RCS, se definen dos canales de comunicación, uno para la transmisión basado en el estándar DVB-S y otro para el canal de retorno para el cual se define como tipo de acceso al medio MF-TDMA, aquí la información se transmite en ráfagas sobre un conjunto de portadoras [60].

Características del canal de retorno:

- Modulación: Es una combinación entre QPSK y el código GRAY, el cual permite una menor probabilidad de errores y una alta eficiencia espectral.
- Acceso al medio: MF-TDMA.
- Tipo de codificación: Se tiene 2 posibilidades que son los códigos *Red-Solomon* y *Viterbi* pertenecientes a los códigos FEC.
- Formato de trama: Utiliza tramas basadas en la norma MPEG-2 o celdas ATM.
- Transporte de datos: Hace uso de los datagramas IP, los cuales son segmentados y enviados a través de celdas ATM o paquetes MPEG-2.

El estándar DVB-RCS tiene definido cuatro tipos de ráfagas; una es la ráfaga de tráfico TRF que es la encargada de transportar la información entre las RCSTs y la pasarela haciendo uso de celdas ATM o paquetes MPEG-2, el segundo tipo de ráfaga es la de adquisición ACQ, la tercera ráfaga de sincronización SYNC y finalmente la ráfaga de señalización de canal común CSC, estas últimas tres clases de ráfagas son usadas para establecer el flujo de señalización de entrada entre la NCC y las estaciones RCSTs [60, 61].

### 2.3.3.1. Políticas de asignación de recursos para el canal de retorno DVB-RCS.

Tener un red satelital con la norma DVB-RSC lo hace muy eficiente en términos de asignación de recursos para el canal de retorno, ya que no existe un *slot* de tiempo que no quede asignado a una terminal, para esto DVB-RSC define dentro de su norma una políticas de asignación de recursos, con el fin de adaptar cualquiera de estas políticas a una arquitectura de red dependiendo de los servicios a ofrecer [63]. Las políticas de asignación de recursos para el canal de retorno son 5:

**Asignación de tasa continua CRA:** Es un tipo de asignación fija y estática de los recursos de la red, después de una negociación entre la estación NCC y el RCST; un número de *slots* se asignan permanentemente a la RCST dentro de una supertrama. Si la estación RCST implementa CRA esta capacidad está dedicada y reservada exclusivamente para esta RCST así la estación solicitadora no haga uso de estos recursos, hasta que la RCST envía un mensaje de liberación de recursos [63, 64, 65].

El comportamiento de la política CRA no es muy eficiente para redes que tienen acceso a Internet donde el tráfico sea variable, pero es ideal para aplicaciones sensibles al retardo con una tasa fija de datos, como son las videoconferencias.

**Tasa basada en la capacidad dinámica RBDC:** En este esquema de asignación por demanda, la RCST periódicamente está enviando un mensaje de solicitud de recursos a la NCC, teniendo en cuenta la tasa de tráfico de entrada a la RCST, cada solicitud es absoluta lo que indica que las solicitudes anteriores son anuladas y solo se tiene en cuenta la última que es recibida en la NCC, esta política tiene dos parámetros que son la tasa mínima RBDC ( $RBDC_{min}$ ) y la tasa máxima RBDC ( $RBDC_{max}$ ), estas tasas son establecidas al inicio de la conexión. La estación RCST solicita dinámicamente en cada intervalo de tiempo la capacidad absoluta que necesita a la NCC, si esta solicitud no supera el  $RBDC_{max}$ , la estación de control le seguirá asignando de manera dinámica recursos, de tal manera que una estación terrena está continuamente solicitando ancho de banda conforme lo vaya necesitando. Una de las

desventajas de esta política es la manera injusta como se hace la asignación de recursos, debido que en ciertos momentos una RCST tiende a monopolizar los recursos de la red, dejando a las demás con pocos recursos disponibles para transmitir sus datos [63, 64, 65].

**Asignación de capacidad basado en volumen VBDC:** La capacidad dinámica basada en volumen es solicitada por la RCST a la NCC a través de peticiones explícitas acumulativas, es decir cada petición se suma a todas las peticiones que han llegado previamente del mismo RCST, indicando el número de slot de tiempo que necesita para transmitir la información que se encuentra encolada. Esta política maneja dos parámetros que son el máximo y mínimo VBDC que puede ser asignado a una estación terrena. VBDC no es recomendable utilizarla para aplicaciones que son bastante sensibles al retardo, ya que esta introduce un retardo inicial mientras se está haciendo la asignación de recursos [63, 64, 65].

**Volumen absoluto basado en capacidad dinámica AVBDC.** Las peticiones de la RCST a la NCC son peticiones explícitas por volumen, pero al contrario que en VBDC, las peticiones no se acumulan, sino que cada vez que el RCST solicita un nuevo volumen, este sustituye cualquier petición anterior por la actual. AVBDC es utilizado para recuperar las pérdidas que se produjo en el modo VBDC. Al igual que VBDC la solicitud de recursos que hace la RCST a la NCC lleva información del número de *slots* que requiere para transmitir la información que se encuentra encolada [64, 65].

**Asignación de Capacidad Disponible FCA:** Es la capacidad que asigna la NCC y que está formada por la capacidad sobrante que no utilizaron las demás políticas anteriores; ésta asignación de recursos es automática, es decir no se requieren solicitudes por parte de la RCST a la NCC. Se puede mirar como una asignación extra en la capacidad para reducir los retardos en tráficos que presentan muchas fluctuaciones [63, 64, 65].

Cabe resaltar que en cada supertrama los *slots* son asignados por la NCC en forma ordenada de acuerdo a la categoría de solicitud, es decir CRA > RBDC > (A)VBDC > FCA, la asignación de los recursos a cada RCST se hace mediante un mensaje TBTP.

**TBTP:** Es un mensaje que es enviado por la estación de control a todas las estaciones RCSTs, esta lleva información de cuantos *slots* de tiempo y en que frecuencias puede transmitir cada estación terrena, como también lleva información de sincronización de trama.

Una vez la estación de control reciba una petición de recursos, se envía la correspondiente asignación a través del plan de asignación de ráfagas TBTP hacia las RCSTs, el TBTP se envía en cada supertrama y actualizado por cada solicitud de recursos que llega a la NCC. Por otro lado se tiene que si la NCC no dispone de los recursos suficientes para atender las solicitudes de las RCSTs, el TBTP no será actualizado con dicha petición por lo que el rechazo o aceptación de la petición se encuentran implícitos en la difusión del plan de ráfagas [60, 64].

#### **2.3.4. Corrección de Errores Hacia Adelante FEC.**

Los códigos FEC se basan en el uso de bits redundantes en la palabra de código transmitida para poder detectar y corregir errores dentro de un canal de comunicaciones, los cuales utilizan un mayor ancho de banda para el envío de datos [66].



El uso de una buena técnica de control de errores en un canal satelital para el protocolo TCP podría hacerlo operar de una manera más eficientemente, ya que al poder corregir los problemas de pérdidas por corrupción que se presentan en las redes satelitales se evitarían los inconvenientes que se producen cuando los mecanismos de congestión se utilizan en la red. Este tipo de códigos corrigen los errores de transmisión en el receptor donde hacen uso de decodificadores, que dependiendo del tipo de FEC que se utilice, depende su grado de complejidad, en cualquiera de ellos se adicionará un retardo en el procesamiento tanto en el codificador y decodificador del código [7, 8].

Las técnicas basadas en FEC que se usan en los sistemas de comunicación por satélite se podrían clasificar dentro de dos categorías según [8]:

- *Códigos convolucionales*, dentro de este se encuentra TCM. Los códigos *convolucionales* pueden ser divididos dentro de tres clases.
  - a) Códigos ortogonales de alta tasa asociados con un umbral de codificación.
  - b) Códigos de memoria mediana y corta, asociados a los códigos con decodificación de *Viterbi* usando una decisión flexible.
  - c) Códigos de larga memoria con decodificación secuencial.
- *Códigos de bloques binario y no binario, con descodificación algebraica*. En esta categoría, el código que presentan un mejor rendimiento es *Reed- Solomon*.

## CAPÍTULO III

### ANÁLISIS DEL IMPACTO DEL NIVEL DE ENLACE DE DATOS EN EL DESEMPEÑO DEL PROTOCOLO TCP EN LAS REDES SATELITALES GEO.

#### 3.0. LA HERRAMIENTA DE SIMULACIÓN.

Para el desarrollo del trabajo de grado se trabajó con la herramienta de simulación *Network Simulator 2*, más conocido como NS-2 en su versión 2.31, dado que es la más adecuada para el desarrollo del trabajo, además de las múltiples utilidades que presenta la herramienta. Una de las razones por la cual se decidió trabajar con NS-2 fue por los trabajos desarrollados en [2, 5, 10, 18, 62, 67] ya que estos realizaron estudios del protocolo TCP en una red satelital.

NS-2 cuenta con muchas versiones algunas de ellas presentan un mejor rendimiento bajo un tipo específico de sistema operativo y según sea su versión se incorporan ciertos módulos. Es muy importante saber escoger la versión de NS-2 que más se ajuste a las necesidades del proyecto, que soporten los módulos TCP, MAC y la capa de enlace de datos, por lo tanto la versión escogida fue NS-2.31, esta fue instalada bajo el sistema operativo Linux con la distribución Ubuntu 9.04.

NS-2 es una herramienta de simulación que permite trabajar con diferentes topologías de redes cableadas, inalámbricas y satelitales, cuenta con distintos protocolos en la capa de aplicación (FTP, CBR, HTTP), en la capa de transporte (TCP, UDP) y en la capa de enlace de datos (MAC del tipo CSMA/CA). Además permite trabajar en modo *unicast* o *multicast* y maneja varios algoritmos para la planificación de colas. NS-2 es un simulador de tiempo discreto que posee una licencia de distribución libre, tiene una muy buena acogida a nivel académico por su versatilidad y profundización en los temas de investigación en las diferentes áreas.

El simulador NS-2 consta de dos lenguajes de programación como lo son C++ y OTcl. C++ es el lenguaje de programación interno de NS-2 orientado a objetos, que se encarga de manejar grandes volúmenes de datos de una manera más rápida que OTcl a la hora de realizar la implementación de los protocolos de comunicaciones, también conocido como la jerarquía compilada; OTcl es la jerarquía que interpreta a C++ y se encarga de la configuración de los escenarios de simulación que actúa como una interfaz ante el usuario. OTcl es el lenguaje donde se definen y se configuran los elementos de la topología de red.

NS-2 en la actualidad se sigue desarrollando a través de CONSER enfocados en el desarrollo e investigación de la evaluación de los protocolos de red. En NS se está desarrollando la versión 3, la cual cuenta con muchas mejoras a la versión 2 con otros módulos y diferentes protocolos implementados.

Dentro de las ventajas que presenta el simulador NS-2 se encuentra que:

- Es un software muy versátil.
- Es un sistema multiplataforma que se ejecuta en diferentes sistemas operativos como Windows 9x/ME/2000/XP o Linux en la mayoría de sus distribuciones.
- A nivel académico es una herramienta ampliamente extendida, que permite la implementación y el desarrollo de muchos protocolos de comunicación.

- Es un software con licencia de distribución libre.

Las desventajas que presenta el simulador NS-2:

- La documentación es a menudo limitada y fuera de fecha con la versión actual del simulador.
- Presenta una serie de limitaciones en cuanto al módulo satelital, ya que aún se encuentra en desarrollo.

En el anexo F se explica detalladamente la instalación de la herramienta de simulación NS-2.31 en el entorno linux con la distribución de Ubuntu 9.04.

### 3.1. ARQUITECTURA DE RED EMPLEADA.

Antes de empezar a describir la arquitectura de red utilizada en el proceso de simulación, se nombran algunas consideraciones con el objetivo de hacer más sencilla las simulaciones, la obtención y el análisis de los resultados que permitan cumplir con los objetivos de este trabajo.

- Se consideró un enlace satelital fijo.
- La arquitectura de red está conformada por un satélite geoestacionario en modo repetidor o transparente (*bent pipe*).
- Para este trabajo de grado solamente se utilizó algunos componentes de la norma DVB-RCS en un ambiente satelital como son las políticas de asignación de recursos, el tipo de estación terrena RCST, la estación de control NCC y el satélite en modo repetidor.
- Las políticas de asignación de recursos en este trabajo de grado se implementaron en el canal de transmisión de datos para evaluar cuál es el desempeño que tiene TCP en la red satelital y no en el canal de retorno como lo especifica la norma DVB-RCS.
- Dentro de las técnicas de acceso al medio descritas en el capítulo dos y en el anexo B, solamente se optó por simular TDMA en conjunto con las políticas de asignación de recursos y el protocolo Aloha puro, la razón es porqué durante la investigación que se realizó sobre la herramienta de simulación no se encontraron los módulos correspondientes que permitieran simular otros protocolos de acceso al medio como lo son FDMA y CDMA en un ambiente satelital.
- Para la evaluación del proxy snoop no se realizó ninguna adaptación o modificación a sus clases dentro del simulador NS-2.
- En NS-2 la ventana ofrecida se fija por el transmisor y permanece constante durante toda la simulación, donde no tiene en cuenta la información del receptor [68].
- De todos los mecanismos de control de flujo descritos en el capítulo dos, se eligieron tres de ellos para evaluar su comportamiento y ver el impacto que se obtiene al variar parámetros de la capa de enlace de datos, estos son TCP Hybla, TCP Westwood y TCP HighSpeed, la

razón de su elección es porque estas versiones de TCP presentan un buen desempeño en una red satelital teniendo en cuenta los estudios hechos en [2, 5, 36, 37, 69].

La arquitectura de red que se propone para la obtención de los resultados se compone de dos escenarios de simulación que se muestran en las figuras 3-1 y 3-2.

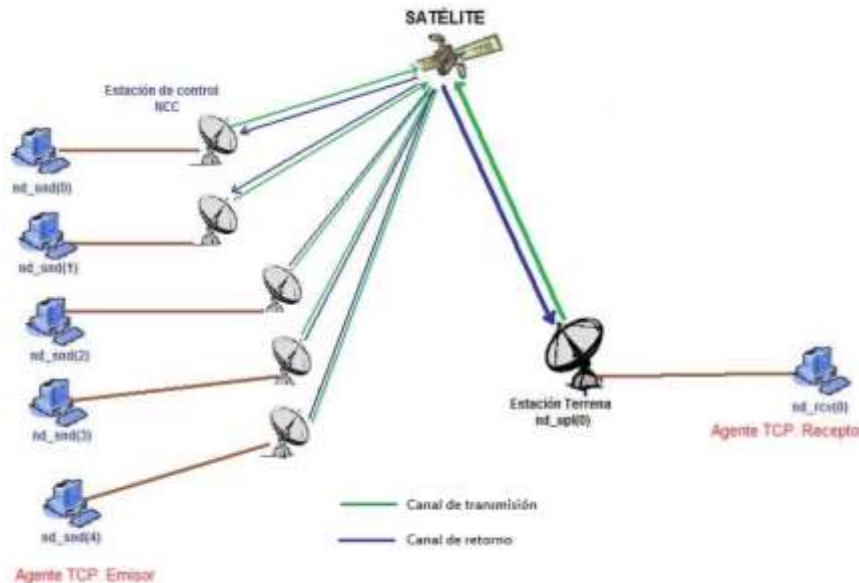


Figura 3-1. Primer escenario de simulación [63].

Como se observa en la figura 3-1 la arquitectura de red que se propone está basada en la plataforma DVB-S/RCS que se describió en el capítulo dos y se tuvo en cuenta la norma ETSI EN 301 790, como también los trabajos desarrollados en [2, 61, 63, 66]. La arquitectura de red está conformada por las siguientes partes:

- **Estación de control NCC:** Como se describió en el capítulo dos y según [60], la estación de control es la encargada de realizar funciones de monitoreo, señalización y asignación de recursos según el TBTP asignado por la NCC a las estaciones RCSTs, la arquitectura de red solamente tiene una única estación NCC, que hará las veces de estación receptora de tráfico.
- **Estaciones terrenas RCST:** Según [61] la norma DVB-RCS existen tres tipos de RCSTs que se utilizan en un canal de retorno, para la arquitectura de red propuesta se utilizan 5 terminales RCST tipo A, ya que esta clase es la que más se adapta a los requerimientos del trabajo. En [61] las RCSTs también se pueden ver como estaciones VSATs dadas sus características de precio, tamaño, facilidad de instalación y configuración del enlace lo que facilita la simulación.
- **Estación terrena receptora:** Para este escenario se tiene una sola estación terrena receptora, que a su vez está conectada a un usuario receptor, quien es el encargado de recibir todo el tráfico generado por las 5 estaciones terrenas transmisoras.

La figura 3-1 muestra la arquitectura de red del primer escenario de simulación, para el enlace de subida se utiliza las cuatro políticas de asignación de demanda de DVB-RCS sin emplear MF-TDMA, aunque estas hayan sido definidas para el canal de retorno. Si se empleará la

arquitectura de red como lo recomienda la norma DVB-RCS el impacto que generan estas en TCP no es muy notorio, ya que el volumen de datos de los ACKs es muy pequeño, donde la estaciones terrenas no harían uso de todos los recursos disponibles (generalmente un slot), tal y como se describió en el capítulo dos.

El objetivo de este escenario es evaluar el impacto que tiene el protocolo TCP en la arquitectura propuesta, mediante la variación de las políticas de asignación por demanda en conjunto con la técnica de acceso al medio TDMA, dentro de este escenario también se podrá observar cuál es el impacto que se tiene al variar parámetros como la duración de la trama, el número de *slots* por trama, el tamaño del *slot* y el TBTP en el rendimiento de la red, como en el comportamiento de los parámetros del protocolo TCP.

Este escenario también permite evaluar cuál es el impacto que se presenta en TCP al implementar el proxy snoop, teniendo en cuenta el desempeño de los mecanismos de control de flujo TCP y en el rendimiento de la red.

Las técnicas FDMA y CDMA no son considerados en este escenario dado que no fue posible encontrar un parche o modulo para la herramienta de simulación que permita simular estos tipos de acceso al medio en un ambiente satelital, si se deseara trabajar con estas técnicas se tendría que diseñar y desarrollar los respectivos módulos en NS-2, lo cual no hace parte de los objetivos de este trabajo.

En la figura 3-2 se observa el segundo escenario de simulación, la técnica de acceso al medio utilizada es el protocolo Aloha Puro, el objetivo de este segundo escenario es poder evaluar cuál es el impacto que tiene Aloha en el desempeño de los mecanismos de control de flujo TCP y en el rendimiento de la red, también se analiza cuál es la probabilidad de que dos paquetes colisionen para las diferentes versiones de TCP.

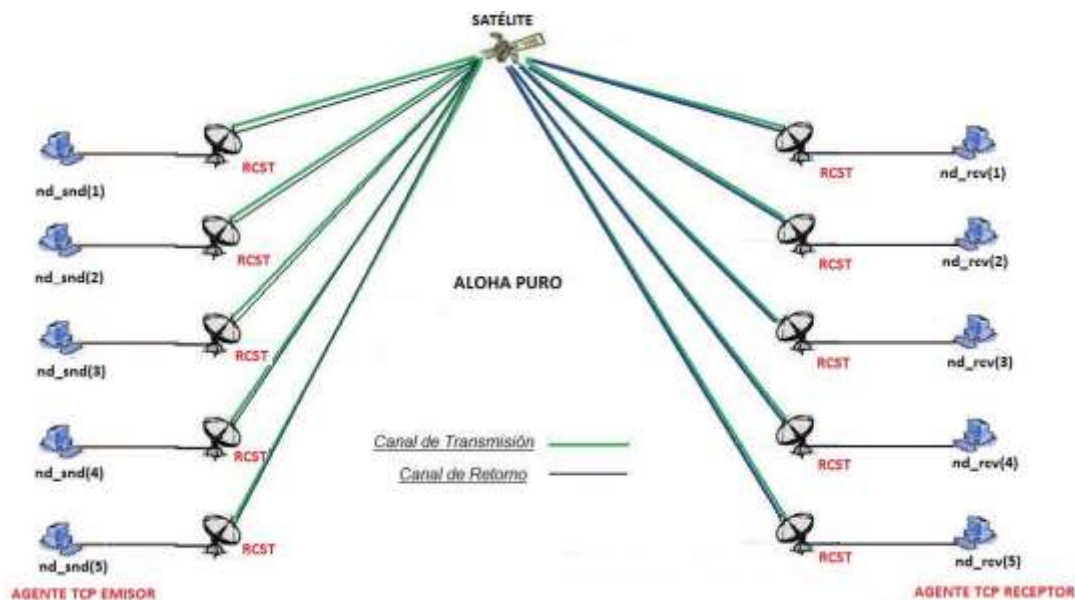


Figura 3-2. Segundo escenario de simulación con Aloha Puro.

En esta arquitectura de red no fue posible evaluar las técnicas de acceso al medio del escenario uno, ya que durante el proceso de simulación, la herramienta no puede determinar si el acceso al medio esta en el canal de retorno o en el canal de transmisión de datos,

presentando datos erróneos y ambigüedades en la asignación de los slots para transmitir los datos.

### 3.1.1. Parámetros de Entrada al Simulador NS-2.

**El satélite:** De acuerdo a los escenarios de simulación que se muestran en las figuras 3-1 y 3-2, estos cuentan con un satélite en modo repetidor, un tamaño de búfer igual al producto por retardo por ancho de banda según se especifica en el trabajo realizado por [37], este es un tamaño que ayuda a analizar el desempeño de TCP.

**Enlace Satelital:** Para el enlace satelital se decidió por un enlace dúplex de 4.0 Mbps, tanto para el enlace de subida como para el de bajada, esta es una velocidad de enlace comúnmente utilizada en redes de acceso satelital.

**Enlace Terrestre:** El enlace terrestre entre el nodo transmisor y la estación terrena es un enlace dúplex de 10Mbps con un mecanismo de gestión de colas tipo *Drop Tail*, con un tamaño de búfer igual al producto por retardo por ancho de banda según [37], para las estaciones terrenas, dado que si el tamaño del búfer es más grande que el producto por retardo por ancho de banda entonces se crea una cola de paquetes en el búfer, desperdiciando recursos de la red, si se tiene en cuenta el caso contrario entonces el tamaño de la ventana de congestión nunca será capaz de alcanzar el máximo tamaño para utilizar el ancho de banda disponible.

**Agentes TCP:** Los agentes seleccionados son TCP Hybla, TCP Westwood y TCP Highspeed, los parámetros comunes para estos agentes son un tamaño de segmento de 1460 bytes, un tamaño de ventana de congestión de 800 segmentos.

**Tiempos:** Para los efectos de simulación se consideró un RTT teórico de 600 milisegundos, que corresponde a un tiempo promedio para enlaces geoestacionarios, un tiempo de simulación de 1000 segundos y para el caso de Aloha de 300s, éste tiempo es adecuado para analizar los resultados arrojados por la simulación.

En la tabla 3-1 se muestra un resumen de los principales parámetros de entrada para los escenarios de simulación.

El tiempo de simulación seleccionado es de 1000 segundos, el cual permitió analizar con mayor claridad el comportamiento de los parámetros de TCP y de la red, el RTT teórico calculado es de 600 ms, que es el tiempo que tarda un paquete en ser confirmado por el transmisor, el mecanismo de gestión de colas predeterminado por el simulador es *Drop Tail*, a pesar de ser un mecanismo que causa la sincronización global de TCP, no fue modificado ya que no hace parte de los mecanismos del nivel de enlace de datos, los valores en los tamaños del búfer en la RCST, el *hub*, el satélite, y el tamaño de la ventana de congestión, se ajustaron al producto por retardo por ancho de banda de la red, el segmento TCP de 1460 bytes se empleó con el objetivo de evitar fragmentaciones en el datagrama IP. Finalmente, la probabilidad de error de  $2 \times 10^{-6}$  es el caso más crítico en el desempeño de TCP para un entorno satelital.

Tabla 3-1. Parámetros de entrada al simulador.

Tiempo de simulación	1000 (s)
Enlace satelital <i>Uplink</i>	Variable
Enlace satelital DownLink	Variable
Enlace terreno	10 Mbps
RTT teórico	600 (ms)
Tipo de encolamiento	DropTail
Agentes TCP implementados	TCP Hybla, TCPW, TCP HS y la opción SACK
Tamaño de la ventana de congestión	800 segmentos
Tamaño del segmento TCP	1460 bytes
Tamaño del búfer de la RCST	500 paquetes
Tamaño de búfer del Hub	500 paquetes
Tamaño del búfer del satélite	230 paquetes
Duración de la trama	Variable
Tamaño del slot	variable
Tamaño supertrama	Variable
Probabilidad de error	0.000002

La figura 3-3, muestra la estructura empleada para analizar el impacto de las diferentes versiones del protocolo TCP al variar los parámetros de los mecanismos del nivel de enlace de datos como TDMA, políticas de asignación de recursos en la NCC, políticas de solicitud de recursos bajo demanda en TDMA, proxy snoop y protocolo Aloha.

Para determinar que mecanismos presentaron el mejor desempeño en una red satelital se evaluaron los parámetros de rendimiento como *throughput*, utilización del canal, retardo, encolamiento, número de slots solicitados y asignados, en conjunto con los parámetros TCP.



Figura 3-3. Estructura empleada para analizar el impacto del protocolo TCP al variar los parámetros de los mecanismos del nivel de enlace de datos.

Considerando la estructura de la figura 3-3, se realizaron diferentes casos de estudio, para el escenario de simulación de la figura 3-1.

El primer caso de estudio se realizó con TDMA, donde se varió el ancho de banda para la duración de la trama, el número de slots por trama, el tamaño del slot, y el tiempo de planificación de recursos, seleccionando la configuración que presentó el mejor rendimiento de la red.

En el segundo caso de estudio se evaluaron las políticas de asignación de recursos en la NCC (FODA y Proporcional) teniendo en cuenta los parámetros que presentaron el mejor comportamiento de la prueba anterior.

En el tercer caso de estudio se analizaron las políticas de solicitud de recursos bajo demanda (RBDC, VBDC, AVBDC y CRA), considerando la política de asignación de recursos que obtuvo el mejor desempeño en la NCC.

Por último, se evaluó el impacto del *proxy snoop* con la política de solicitud de recursos que presentó el mejor comportamiento de los parámetros TCP y de la red; de esta forma se pudo concluir cuales son los mecanismos del nivel de enlace de datos que presentaron el mejor desempeño en las tres versiones de TCP evaluadas.

En el protocolo Aloha puro se varió el tiempo aleatorio de retransmisión de recursos para evaluar el comportamiento del protocolo TCP en las versiones seleccionadas, el escenario de simulación se presenta en la figura 3-2, y los parámetros de entrada al simulador se indican en la tabla 3-26.

### 3.1.2. Métricas de Simulación para Evaluar el Desempeño de TCP y Rendimiento de la Red.

Para analizar cuál es el impacto que se ocasiona en el protocolo TCP al variar los mecanismos del nivel de enlace de datos y el rendimiento de la red satelital, se tiene en cuenta un conjunto de indicadores que permiten medir cual es el comportamiento de los parámetros de TCP y el rendimiento de la red en un ambiente satelital, estos indicadores de medida se nombran a continuación [16, 36].

#### 3.1.2.1. Métricas TCP.

- **Ndatapack:** Número de paquetes que un flujo TCP transmite en un intervalo de tiempo, este se mide en el nodo transmisor.
- **Pktx:** Número de paquetes que se han tenido que retransmitir por causa de congestión de la red o *timeout*.
- **Cwnd:** Tamaño de la ventana de congestión en cualquier instante de tiempo, la unidad está dada en segmentos, y su medida se realiza en el nodo transmisor.
- **Ssthresh:** Umbral que indica si la ventana de congestión se encuentra en la fase de inicio lento o evasión de congestión.
- **Seq:** Numero de secuencia del segmento TCP, este se mide en el nodo transmisor.
- **ACKs:** Indica el número de paquetes que han llagado al receptor con éxito y se mide como el número de ACKs que arriban al transmisor en un intervalo de tiempo.



- **RTT:** Tiempo en que se demora el transmisor en conocer si un paquete ha llegado a su destino, la unidad está dada en milisegundos y se mide en el nodo transmisor.

### 3.1.2.2. Métricas del rendimiento de la red.

- **Throughput:** Indica la cantidad de bytes por segundo que arriban al nodo receptor, este se mide cada segundo en cada conexión.
- **Utilización del enlace:** Capacidad del enlace utilizado por un solo flujo TCP en un intervalo de tiempo, este parámetro se mide entre la estación terrena (HUB) y el nodo receptor.
- **Capacidad del enlace total:** Porcentaje de utilización total del canal por todos los flujos TCP en un intervalo de tiempo, este parámetro se mide entre la estación terrena (HUB) y el nodo receptor.
- **Encolamiento:** Este parámetro indica la cantidad de paquetes encolados en el búfer de cada estación terrena transmisora.
- **Retardo:** Este parámetro indica la latencia que tiene un paquete desde el momento que entra al búfer de la estación terrena transmisora hasta que llega a la estación terrena receptora, este tiempo está dado en segundos, para el protocolo Aloha el retardo indica el tiempo que demora un paquete en llegar al destino, este se mide entre el nodo transmisor y el nodo receptor.
- **Slots solicitados:** Cantidad de slots que la estación terrena solicita a la NCC, este número se mide en la estación RCST.
- **Slots asignados:** Número de slots que la NCC le asigna a cada estación terrena de acuerdo a las solicitudes hechas por las RCST y de acuerdo a la política de asignación de recursos que se tenga implementado.
- **Utilización del enlace de subida (*capused*):** Capacidad usada por cada flujo TCP en el enlace de subida de la red satelital.

En la tabla 3-2 se muestran todas las variables que se utilizaron en las pruebas de simulación para las diferentes versiones de TCP seleccionadas, las cuales permiten obtener los datos para evaluar el desempeño de TCP.

Tabla 3-2. Parámetros de las versiones de TCP.

Parámetros	Descripción
<b>Seqno</b>	Número de secuencia.
<b>RTT</b>	Tiempo de ida y vuelta
<b>Srtt</b>	RTT estimado.
<b>Rttvar</b>	Varianza del RTT.
<b>Backoff</b>	Variación del RTO
<b>Dupacks</b>	ACKs duplicados
<b>Wnd_</b>	Ventana ofrecida
<b>Cwnd</b>	Ventana de congestión
<b>Packetsize</b>	Tamaño del paquete
<b>Ack</b>	Número de ACKs.
<b>Ssthresh</b>	Umbral
<b>Maxseq</b>	Último número de secuencia recibido.
<b>Ndatapack</b>	Cantidad de paquetes transmitidos.
<b>Nrexmit</b>	Número de retransmisiones por <i>timeout</i> .

En el anexo F se describe en detalle un ejemplo de cómo implementar una red satelital en el simulador NS-2, y que sirve como base para la configuración de los programas en tcl.

### 3.2. OBTENCIÓN Y ANÁLISIS DE RESULTADOS EN EL DESEMPEÑO DE TCP WESTWOOD, HYBLA Y HIGHSPEED VARIANDO PARÁMETROS DE LA CAPA DE ENLACE DE DATOS.

En esta sección se muestran los resultados obtenidos del escenario uno con su respectivo análisis, mostrando cuál es impacto que tienen los mecanismos seleccionados de la capa de enlace de datos en el desempeño del protocolo TCP como el rendimiento de la red.

**Nomenclatura:** La nomenclatura utilizada a lo largo del documento para identificar el comportamiento de las diferentes versiones TCP son: el color azul corresponde a TCP Westwood, el rojo a TCP Hybla y el verde para TCP HighSpeed.

#### 3.2.1. Impacto al Variar el Número de Slots por Trama en TDMA.

Para determinar cuál es el impacto que tiene el variar el número de slots por trama en el desempeño de las versiones de TCP y el rendimiento de la red satelital GEO, se realizaron las pruebas A y B donde los parámetros modificados son los que se muestran en la tabla 3-3.

Tabla 3-3. Pruebas realizadas para evaluar el impacto al modificar el N° slots por trama en TCP.

Parámetros a modificar	Prueba		
	A	B	
Duración trama	0,05	0,05	FIJOS
Supertrama	20	20	
Tamaño Slot	1504	1504	
N° Slots	20	14	VARIABLES
Ancho de Banda	4.812.800	3.368.960	

Los datos que se observan en la tabla 3-3 fueron calculados mediante la ecuación 19 que es la utilizada por el parche TDMA-DAMA [70].

$$Slotframe = \frac{(BW\ UPLink) * Duracion\ Trama}{8 * Tamaño\ del\ Slot} \quad 19$$

Donde *Slotframe* indica el número máximo de *slots* por trama, en la ecuación 19 se observa que la duración de la trama, como el ancho de banda para el enlace de subida del canal satelital,

son directamente proporcionales al número máximo de *slots* por trama, por lo que se tiene que un incremento del ancho de banda o la duración de la trama inmediatamente se incrementan *Slotframe* en la misma proporción; por otro lado se tiene que el tamaño del *slot* es inversamente proporcional al número máximo de *slots* por trama, por ello para un tamaño de *slot* pequeño se va a tener un incremento en el número de *slots* por trama o viceversa.

### 3.2.1.1. Análisis de la ventana de congestión, *ssthresh*, número de paquetes transmitidos y retransmitidos.

En figura 3-4 se observa el comportamiento de la ventana de congestión y el número de paquetes transmitidos para las versiones de TCP Westwood, Hybla y HighSpeed, el impacto que genera el tener un mayor número de *slots* por trama (prueba A) y un BW más grande en *CWND* es un incremento más rápido durante la fase de inicio lento, además de una recuperación más rápida ante la presencia de ACKs duplicados durante la fase de evasión de congestión, dado que *CWND* crece de forma lineal con pendientes bien pronunciadas, lo cual se ve reflejado en un mejor comportamiento del *ssthresh* conservando umbrales superiores para un mayor número de *slots* por trama.

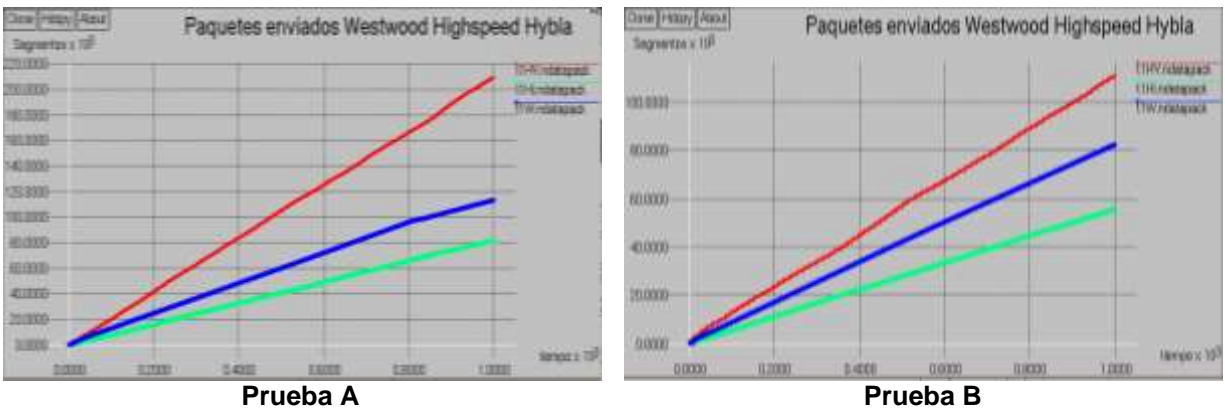
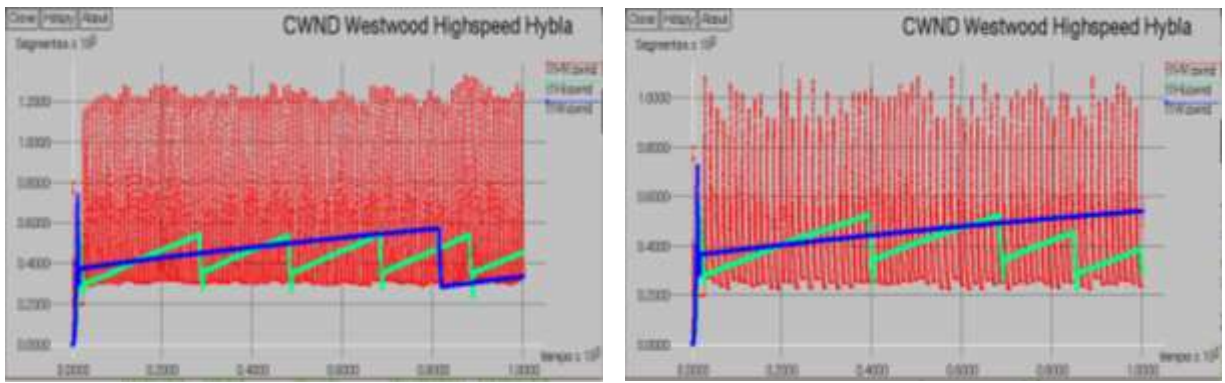


Figura 3-4. Evolución de los paquetes transmitidos.

Por otro lado se tiene que para un mayor número de *slots* por trama se presentan más paquetes transmitidos como lo indica la figura 3-5, esto como consecuencia del gran número de oscilaciones que presenta *CWND* que alcanza valores altos en sus diferentes fases de crecimiento, pero a cambio de esto se presentan más ACKs duplicados lo cual se traduce en un aumento de paquetes retransmitidos. En los anexos C, D y E se explica en detalle el comportamiento de estos parámetros para cada versión de TCP.



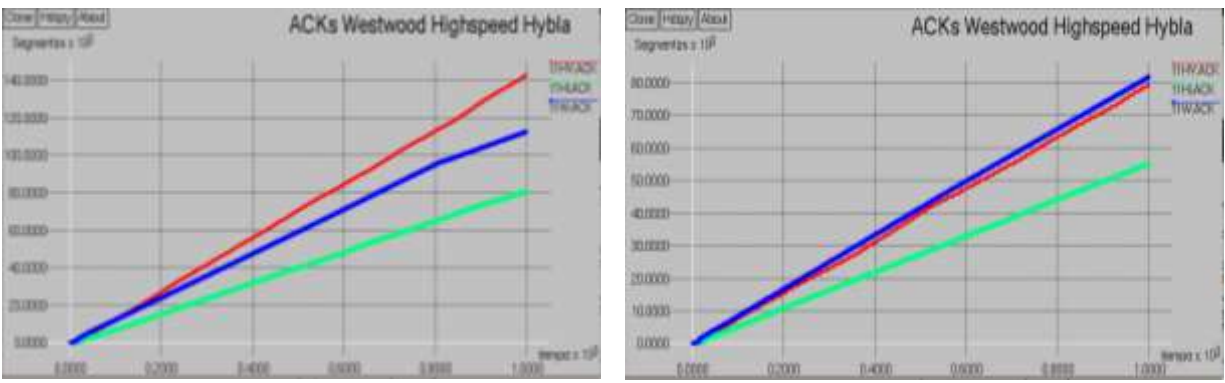
Prueba A Prueba B  
 Figura 3-5. Comportamiento de CWND.

Cabe resaltar que para la prueba que tiene un mayor número de slots por trama, implica tener un mayor ancho de banda lo que permite que *CWND* crezca más rápido, como consecuencia se transmiten muchos más paquetes.

**3.2.1.2. Análisis de ACK, RTT y número de secuencia.**

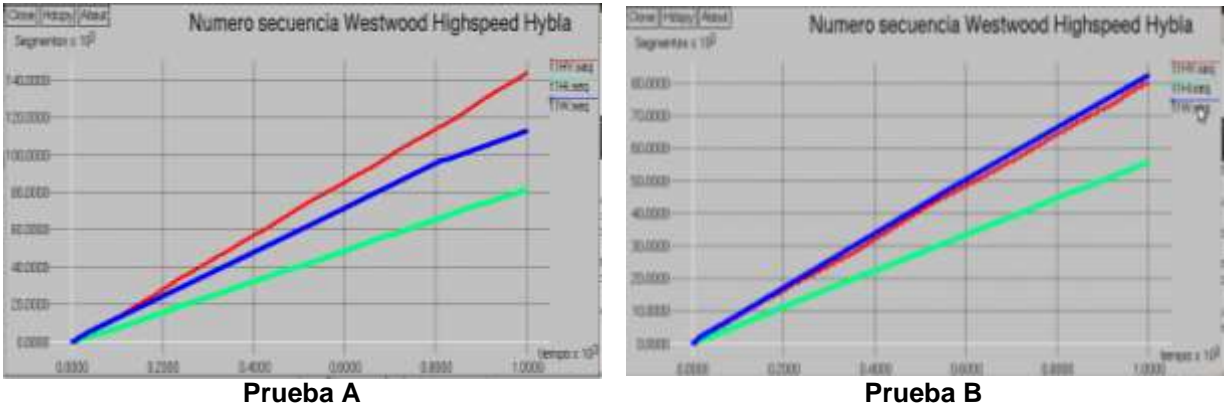
En la figura 3-6 se muestra el comportamiento de los ACKs y el número de secuencia para las tres versiones TCP a analizar, donde se observa que el impacto que genera el tener un mayor número de *slots* por trama en los ACKs, es un crecimiento más rápido, de manera lineal con pendientes más pronunciadas, lo que indica que en el transmisor están llegando más ACKs; por lo tanto *CWND* va a crecer más rápido y se va a tener un aumento de los paquetes transmitidos; se deduce también que para un mayor número de slots por trama con un ancho de banda mayor, los ACKs presentan menores intervalos de tiempo donde permanecen constantes, lo que indica la rapidez con que sale de la fase de recuperación rápida a la fase de evasión de congestión, caso contrario ocurre para un menor número de *slots* por trama.

En cuanto al impacto que se tiene con el número de secuencia es muy similar al de los ACKs ya que estos crecen de manera proporcional a como llegan los ACKs al transmisor como indican las figuras 3-6 y 3-7.

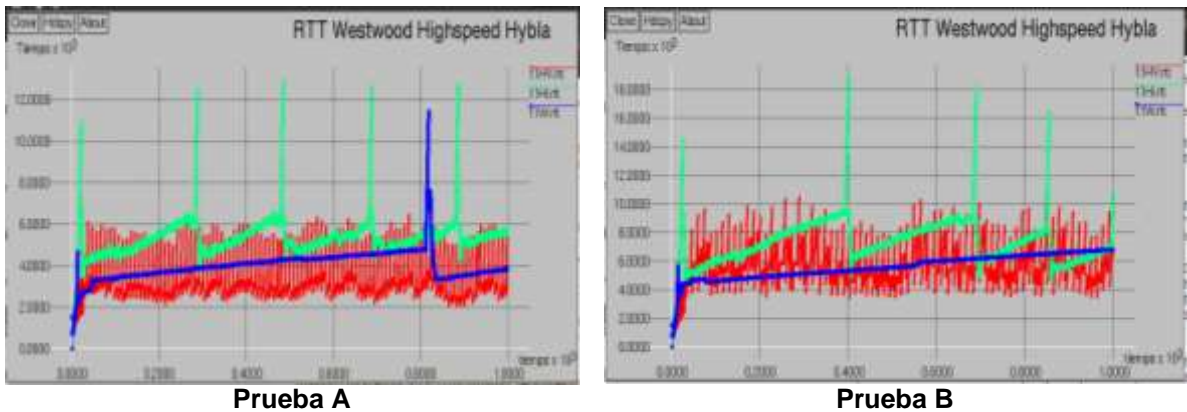


Prueba A Prueba B  
 Figura 3-6. Comportamiento de ACKs.

En cuanto al impacto que se tiene sobre el RTT, el tener un mayor número de *slots* por trama, presenta un mejor comportamiento de este, ya que se obtuvieron RTTs bajos para las tres versiones analizadas, en comparación con los RTTs para un menor número de *slots* por trama donde sus promedios son superiores, esto se observa en la figura 3-8. También se observa que al tener valores altos de RTT (prueba B), impacta notablemente en el comportamiento de la red, debido a que es un parámetro que indica el grado de congestión en la red satelital.



Prueba A Prueba B  
 Figura 3-7. Comportamiento del número de secuencia.



Prueba A Prueba B  
 Figura 3-8. Comportamiento RTT.

### 3.2.1.3. Análisis del encolamiento y retardo de la red.

Al observar las figuras<sup>2</sup> de los anexos C, D y E se visualiza el impacto al utilizar un mayor número de *slots* por tramas, donde para las tres versiones analizadas se presentó un menor tiempo de retardo en la red, lo que implica que los paquetes encolados se están transmitiendo de manera mucho más rápida, lo que influye positivamente en el comportamiento de los parámetros de TCP, dado que *CWND* va a crecer más rápido en intervalos de tiempo más cortos, mayor *ACKs* llegan al transmisor y se obtiene un buen comportamiento del RTT. Se observa también que para TCP Hybla y HighSpeed se presenta un sub-dimensionamiento del búfer, lo que implica que la gran mayoría de las pérdidas es debido al búfer, pero esto se ve compensado al tener más número de *slots* por trama, el cual permite una recuperación más

<sup>2</sup> Ver Sección 1.4 del anexo E, Sección 1.3 anexo D y Sección 1.4 del anexo C.

rápida en un menor tiempo ante la presencia de una pérdida, en comparación cuando se utiliza un menor número de *slots* por trama, una descripción más detallada se puede ver en los anexos C, D y E.

### 3.2.1.4. Análisis del *throughput* y capacidad del enlace utilizado.

En la figuras 3-9 y 3-10 se observa que para un mayor número de *slots* por trama tanto el *throughput* de la red como la utilización del canal por todas las conexiones presentan un mejor desempeño debido a que el volumen de datos transmitidos es mayor que en la prueba B, esto ratifica el impacto positivo que tiene sobre los mecanismos de control de flujo y desempeño de TCP en las tres versiones analizadas.

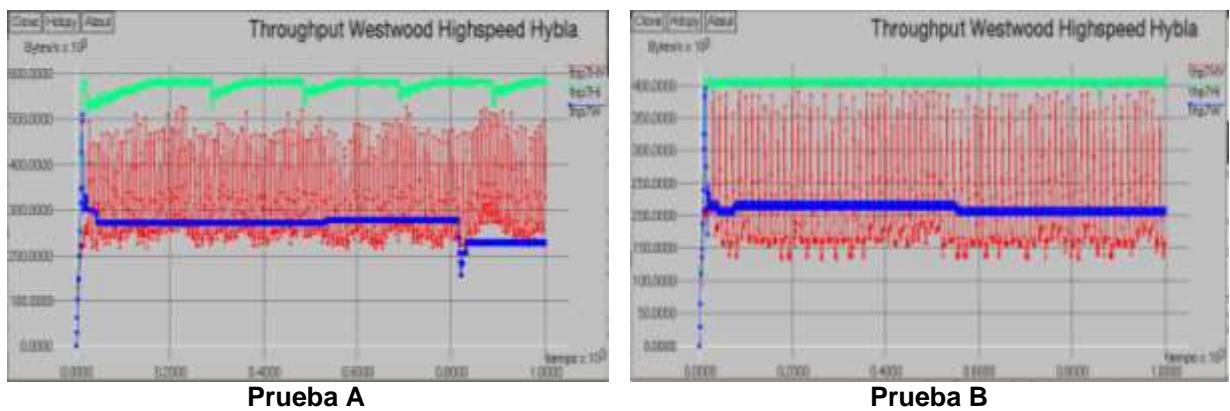


Figura 3-9. *Throughput* de la red.

A manera de conclusión se puede decir que el tener un mayor número de *slots* por trama (prueba A) se genera un impacto positivo del desempeño del protocolo TCP, con un mejor comportamiento para TCP HS, seguido por TCP Hybla y TCPW.

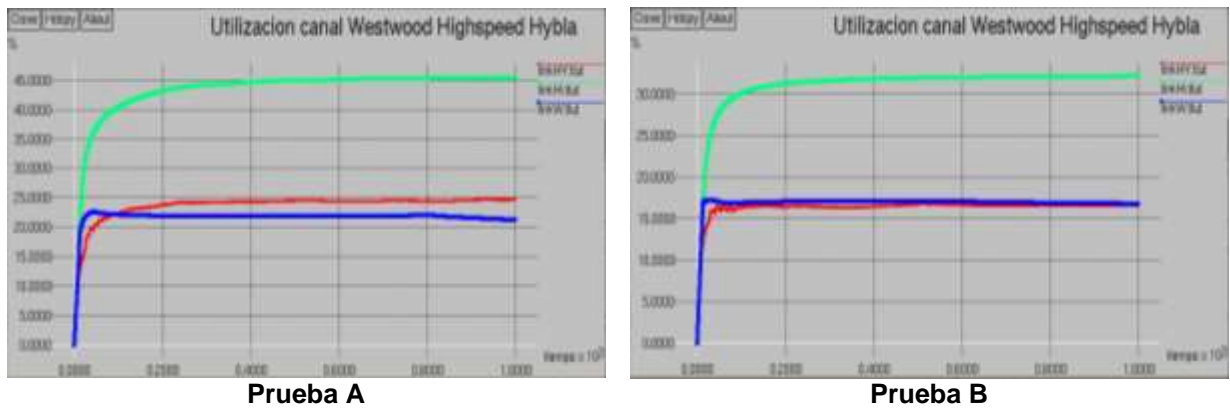


Figura 3-10. Utilización del canal por todas las conexiones.

### 3.2.2. Impacto al Variar el Tamaño del Slot en TDMA.

Este caso de estudio se evaluó el impacto que tiene utilizar tamaños de slots diferentes en el desempeño del protocolo TCP, para realizarlo se hicieron las pruebas C y D con los parámetros de la tabla 3-4.



Tabla 3-4. Pruebas realizadas para evaluar el impacto de modificar el tamaño del slot en TCP.

Parámetros a modificar	Prueba		
	C	D	
Duración trama	0,04	0,04	FIJOS
Supertrama	25	25	
Nº Slots	14	14	
Tamaño del Slot	184	1504	VARIABLES
Ancho de Banda	515.200	4.211.200	

### 3.2.2.1. Análisis de la ventana de congestión, *ssthresh*, número de paquetes transmitidos y retransmitidos.

La figura 3-11 muestra el comportamiento de *CWND* para tamaños de slots diferentes, el impacto producido al utilizar un tamaño de slot más grande, es un crecimiento más rápido de *CWND* tanto en la fase de inicio lento como en la fase de evasión de congestión caso contrario se presenta para un tamaño de slot pequeño.

Para las versiones TCP Hybla y TCP HS el crecimiento de *CWND* se ve limitada por el tamaño del búfer tanto en la etapa de inicio lento como en la fase de evasión de congestión al tener un tamaño de *slot* más grande produciendo un aumento en la pérdida de paquetes y la retransmisión de los mismos, esto se ve compensado con el rápido crecimiento de *CWND*.

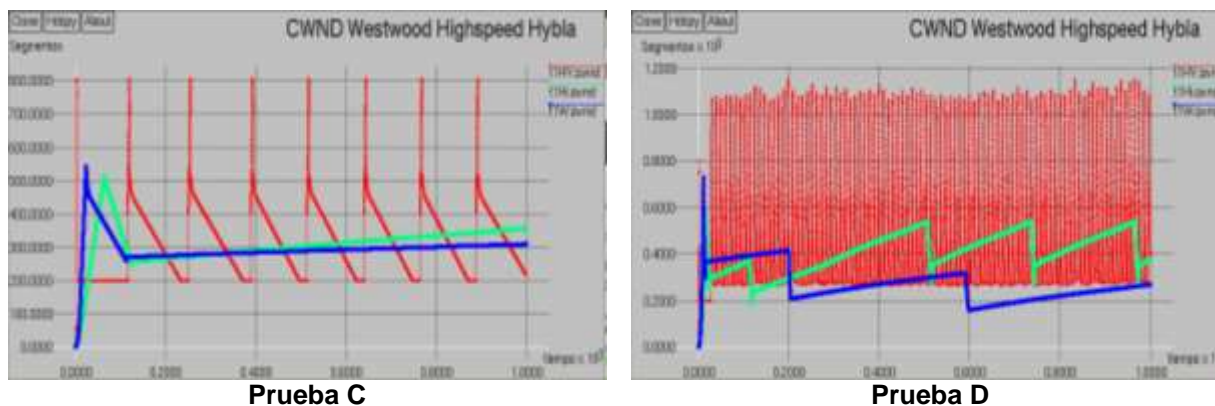


Figura 3-11. Comportamiento de la ventana de congestión.

Por otro lado se tiene que para un tamaño de *slot* pequeño el comportamiento de *CWND* es muy conservador, es decir durante la fase de inicio lento y evasión de congestión el crecimiento de *CWND* es lento y menos oscilante, lo que indica que se están transmitiendo menos paquetes (ver figura 3-12), la razón de este comportamiento es causado por el gran retardo de propagación que se presenta en la red, producto de un menor ancho de banda con que se dispone para su uso, el cual se ve reflejado en menos retransmisiones de paquetes y un bajo

rendimiento de la red, para una explicación más detallada del comportamiento de la ventana de congestión para cada versión remitirse a los anexos C, D y E.

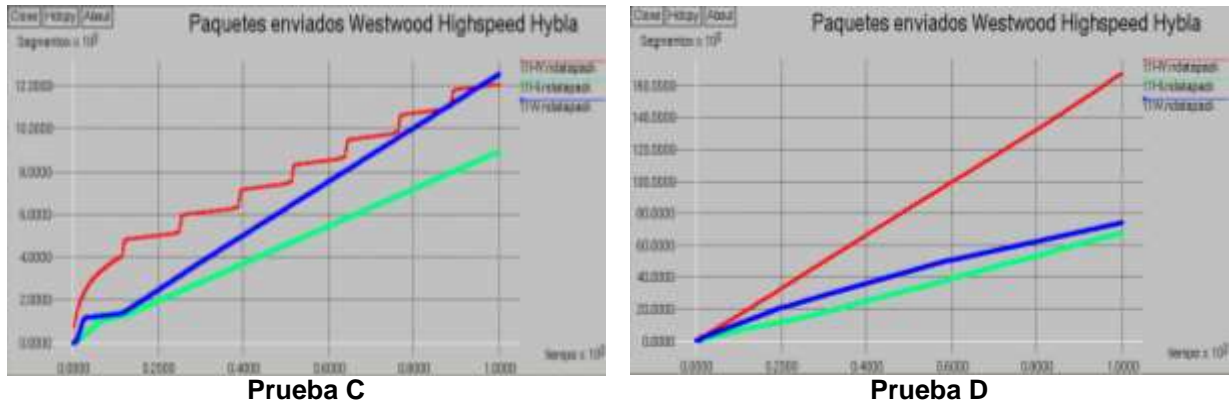


Figura 3-12. Paquetes transmitidos.

### 3.2.2.2. Análisis de ACK, RTT y número de secuencia.

En la figura 3-13 se observa el comportamiento de los ACKs que arriban al transmisor.

El impacto que genera al tener un tamaño de *slot* más pequeño, es un menor número de ACKs al receptor, el cual se ve reflejado en un crecimiento de *CWND* más conservado, que implica que la red se encuentre congestionada; la red siempre espera transmitir un mayor volumen de datos pero como los ACKs se retrasan debido al retardo de la red que se tiene y al incremento del RTT (ver figura 3-14); el rendimiento que alcanza la ventana de congestión no es el óptimo. Al utilizar un tamaño de *slot* más grande con un ancho de banda mayor los paquetes de datos transmitidos periódicamente se aumentan y hacen uso de una mayor capacidad de la red, lo que permite que los ACKs estén continuamente creciendo.

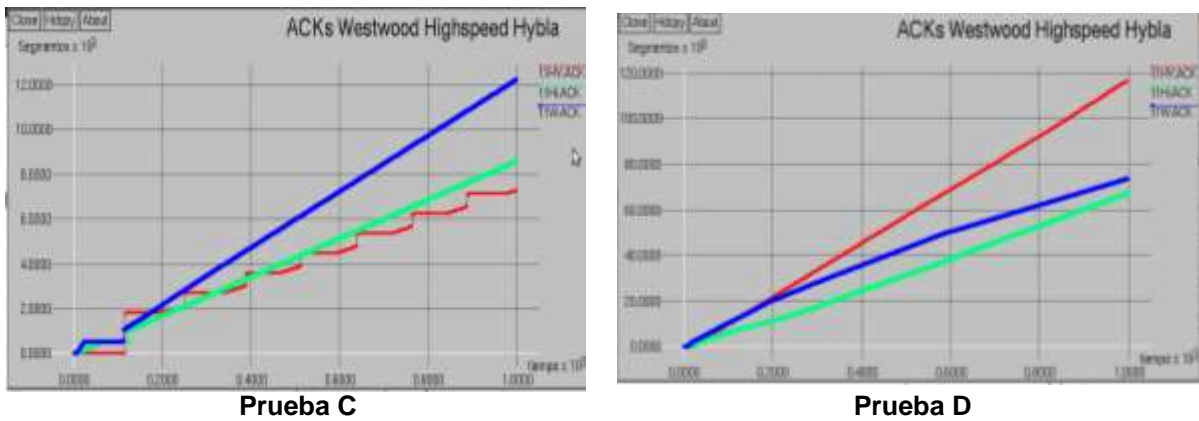


Figura 3-13. Comportamiento de los ACKs.



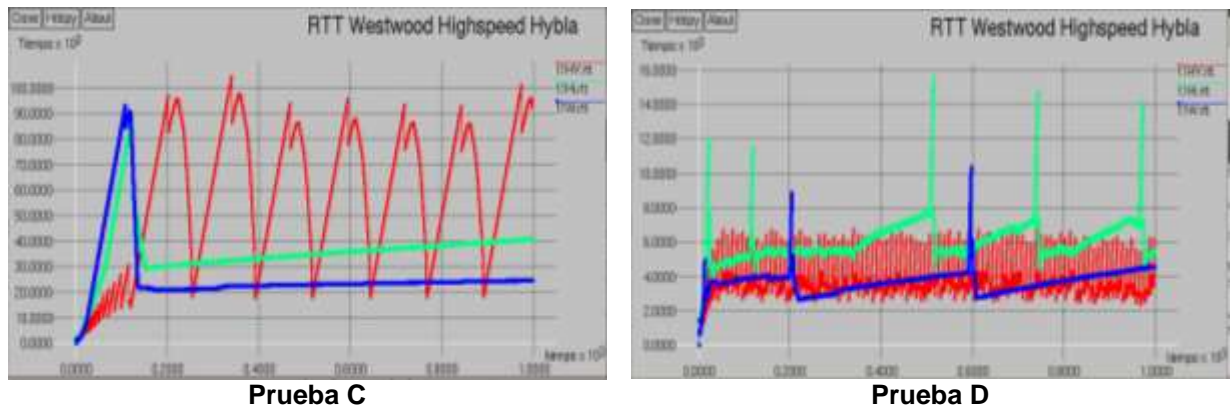


Figura 3-14. Comportamiento del RTT.

Por lo tanto un tamaño de *slot* pequeño implica retrasos en la llegada de los ACKs lo que se ve reflejado en un incremento bastante notorio del RTT dando a entender al transmisor que la red se encuentra congestionada y por ende el crecimiento de *CWND* es muy conservado y limitado.

Cabe aclarar que el comportamiento de *CWND* en la prueba C (ver figura 3-11) es más oscilante para un tamaño de *slot* más grande, lo que indica que la red se encuentra congestionada, pero esto no ocurre, ya que su RTT está muy por debajo en comparación con el RTT para un tamaño de *slot* más pequeño; la razón de estas oscilaciones de *CWND* es por las pérdidas ocasionadas por un sub-dimensionamiento del búfer, donde TCP los interpreta como congestión de la red, provocando una disminución de *CWND*, sin embargo la recuperación de la ventana en la prueba C es muy rápida, el cual no afecta el rendimiento de la red.

### 3.2.2.3. Análisis del encolamiento y retardo de la red.

El encolamiento de paquetes en la prueba C tiene un nivel de pendiente mayor, por ello se deduce que más paquetes están siendo transmitidos y por lo tanto un mayor número de slots han sido solicitados a la NCC; por estas razones la red no se encuentra tan congestionada, como se observa en la figuras<sup>3</sup> de los anexo C, D y E.

Un tamaño de *slot* pequeño implica un menor número de paquetes por transmitir, donde las estaciones solicitadoras tienen que esperar un tiempo adicional para poder transmitir los paquetes encolados, en ese tiempo de espera el retardo de la red se incrementa como se visualiza en las figuras<sup>2</sup> de los anexos C, D y E, esto conlleva a RTTs muy altos perjudicando el rendimiento de la red, ya que un menor volumen de datos va a ser transmitidos por las estaciones terrenas.

### 3.2.2.4. Análisis del *throughput* y capacidad del enlace utilizado.

En la figura 3-15 se aprecia el rendimiento de la red para todas las conexiones, donde se observa que para un tamaño de *slot* más grande se obtiene un mejor desempeño en comparación con un tamaño de *slot* más pequeño.

---

Ver Sección 1.4 del anexo E, Sección 1.3 anexo D y Sección 1.4 del anexo C.

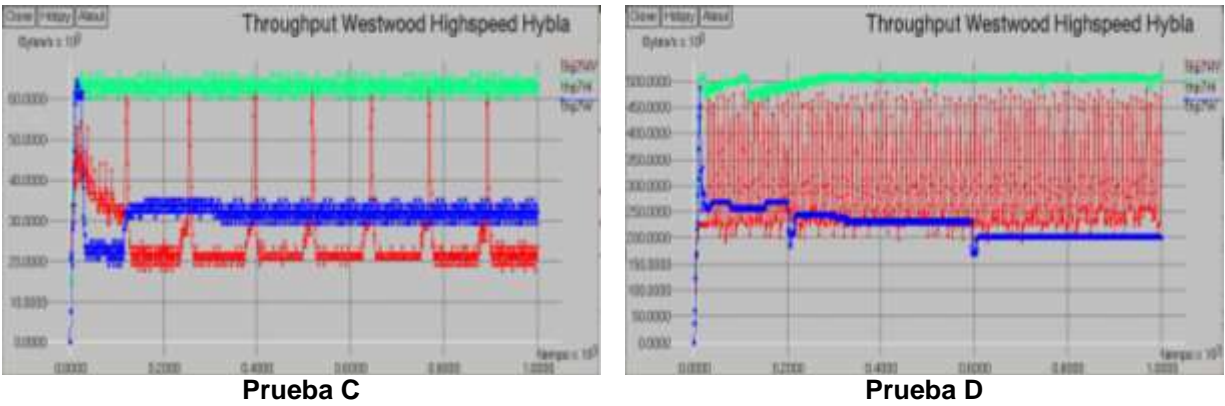


Figura 3-15. Throughput de la red.

En cuanto a la utilización del canal por todas las conexiones en la figura 3-16 se aprecia que para un tamaño de *slot* más grande se presenta una mejor utilización del canal satelital, indicando el impacto positivo que tiene este sobre el desempeño de TCP.

Como conclusión el tener un tamaño de *slot* más pequeño afecta el desempeño de TCP Hybla y TCP HS al presentar RTTs más altos y menor número de ACKs que llegan al transmisor producto del gran retardo de la red.

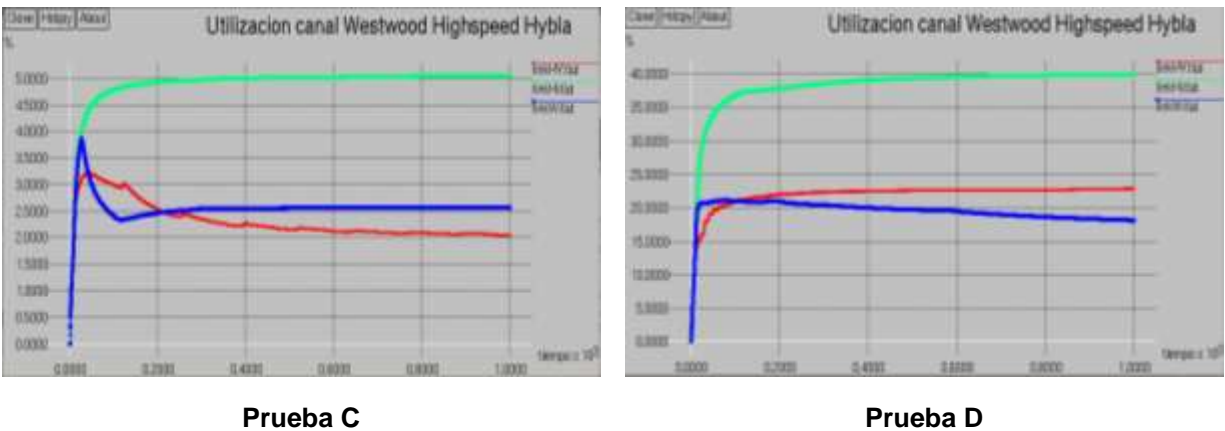


Figura 3-16. Utilización del canal satelital por todas las conexiones.

### 3.2.3. Impacto al Variar la Duración de la Trama en TDMA.

Para este caso de estudio se midió el impacto, al variar la duración de la trama en el desempeño del protocolo TCP, realizando algunas pruebas como se indica en la tabla 3-5.

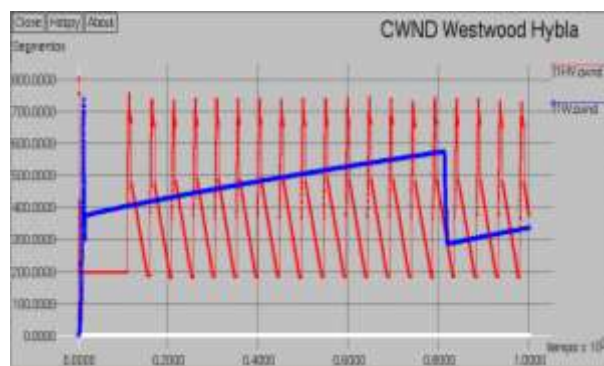
En TCP Hybla, se evaluó su impacto con las pruebas E y G, en TCPW con E y F y en TCP HS con F y G; respectivamente.

Tabla 3-5. Impacto de TCP al variar la duración de la trama.

	<i>Prueba</i>			
Parámetros a modificar	E	F	G	
Tamaño Slot	1504	1504	1504	<b>FIJOS</b>
Ancho de Banda	4.812.800	4.812.800	4.812.800	
Nº Slots	20	24	16	<b>VARIABLES</b>
Duración trama	<b>0,05</b>	<b>0,06</b>	<b>0,04</b>	
Supertrama	20	17	25	

### 3.2.3.1. Análisis de la ventana de congestión, el *ssthresh* y paquetes transmitidos.

En las figuras 3-17 y 3-18 se observa el comportamiento de *CWND*, donde el impacto que genera el tener una duración de trama más pequeña (conservando el mismo TBTP para todas las pruebas) es un crecimiento más rápido de *CWND* en la fase de inicio lento y en evasión de congestión, lo que se ve reflejado en la transmisión de más paquetes por parte del transmisor (ver figura 3-19).

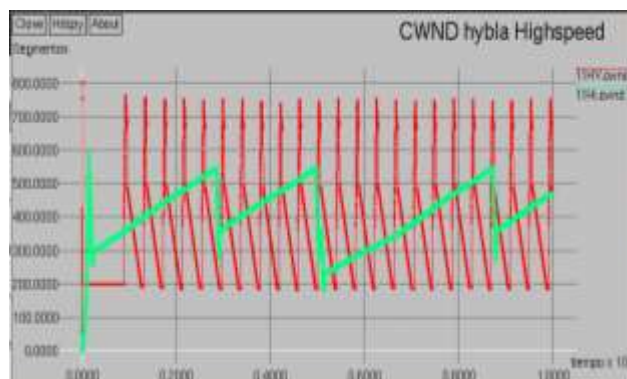


**Prueba E**



**Prueba F**

Figura 3-17. Comportamiento de *CWND* en las pruebas E y F.



**Prueba G**

Figura 3-18. Comportamiento de *CWND* en la prueba G.

Por otro lado el impacto que tiene una duración de trama más pequeña sobre *CWND* es un comportamiento más oscilante (o diente de sierra), lo que indica un aumento de ACKs duplicados o pérdida de paquetes, lo que genera más retransmisiones de paquetes, pero a cambio de esto *CWND* se recupera mucho más rápido ante una pérdida, lo que no ocurre para una duración de trama más grande, donde se presentan menos retransmisiones, pero ante una pérdida *CWND* tarda mucho más tiempo en recuperarse, impidiendo que se transmitan más paquetes.

Es de aclarar que la gran mayoría de los paquetes retransmitidos para una duración de trama más pequeña es producto de un desbordamiento del búfer de la estación terrena transmisora y no a causa de la congestión de la red.

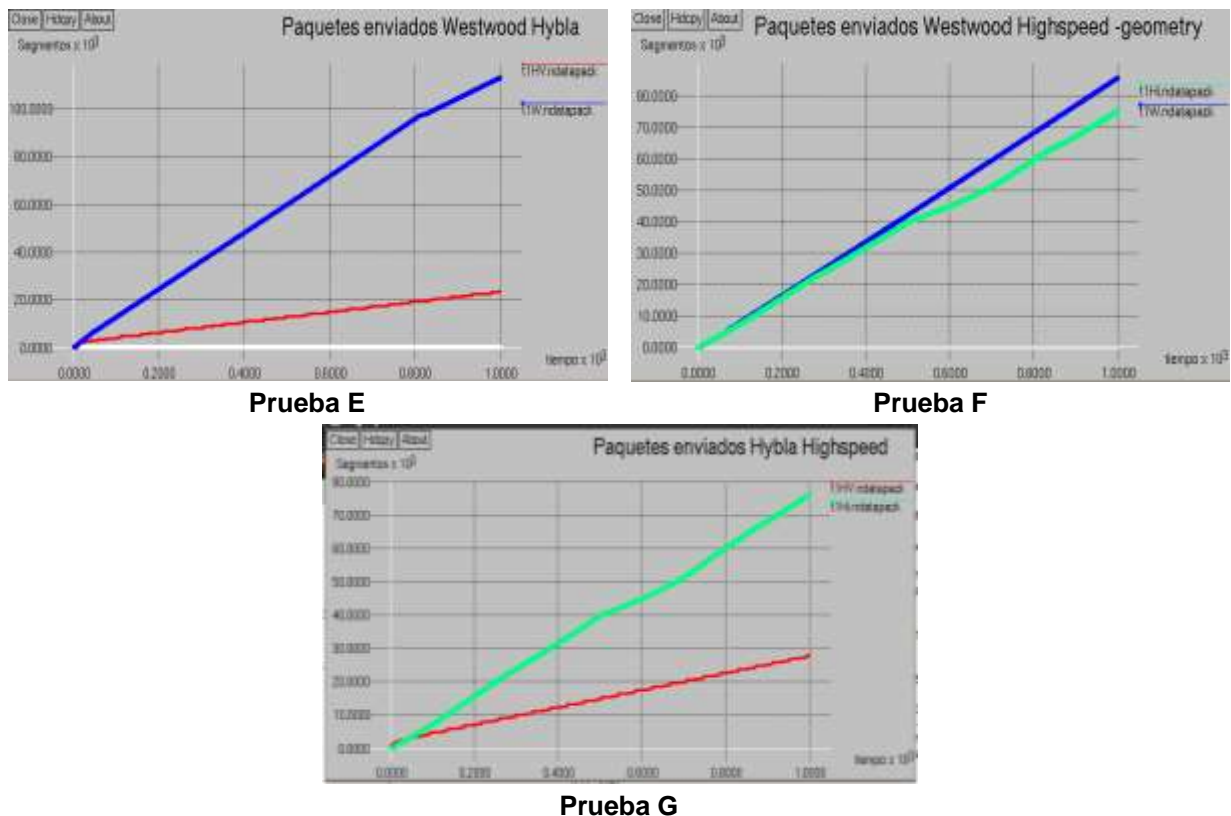
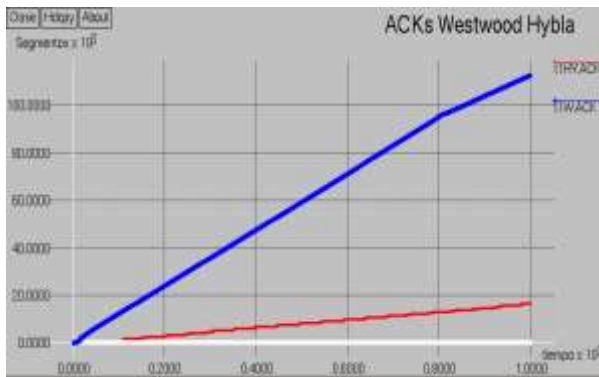


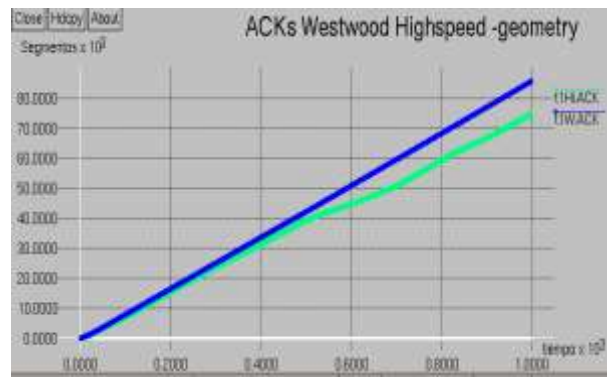
Figura 3-19. Comportamiento de los paquetes transmitidos.

### 3.2.3.2. Análisis de ACK, RTT y número de secuencia.

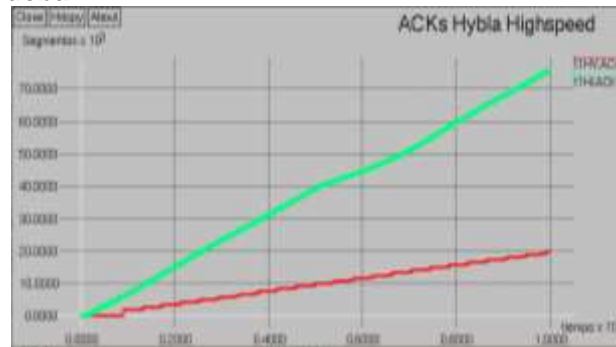
En las figuras 3-20 se aprecia el impacto que tiene una duración de trama pequeña sobre los ACKs que llegan al receptor, donde su crecimiento es casi lineal con pendientes bien pronunciadas durante las fases de inicio lento y evasión de congestión, esto es producto del buen comportamiento de *CWND*, en las figuras 3-21a y 3-21b se tiene que el comportamiento del RTT para una duración de trama pequeña no es el mejor ya que se presentan valores promedios más altos que para una duración de trama grande, esto para TCP Westwood, para las otras versiones el comportamiento del RTT es muy similar.



**Prueba E**

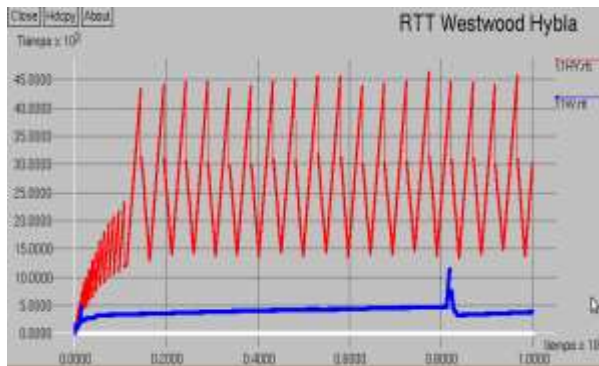


**Prueba F**

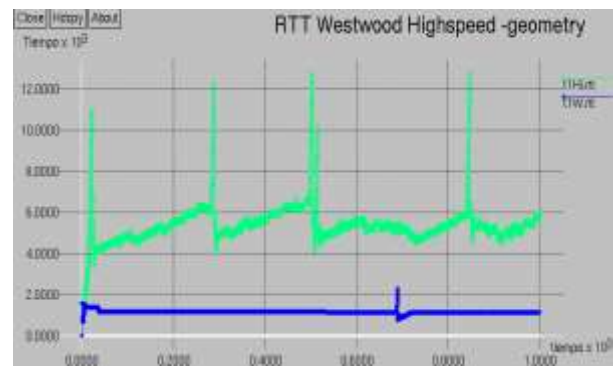


**Prueba G**

Figura 3-20. Comportamiento del ACK en las pruebas E, F y G.

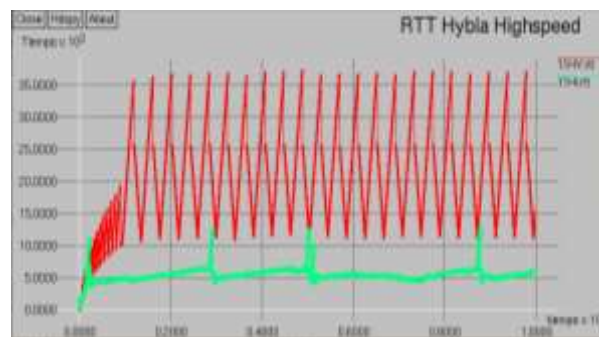


**Prueba E**



**Prueba F**

Figura 3-21a. Pruebas E y F para la evolución del RTT.



**Prueba G**

Figura 3-21b. Evolución del RTT en la prueba G.



### 3.2.3.3. Análisis del encolamiento y retardo de la red.

El comportamiento general que se tiene para todas las conexiones y las diferentes versiones de TCP es que para una duración de trama más pequeña y un TBTP alrededor de un segundo, se presenta un mayor encolamiento de paquetes, pero a su vez un menor retardo de encolamiento, lo que indica que el búfer de la terminal está creciendo y decreciendo con pendientes bien pronunciadas, lo cual impacta positivamente en el desempeño de TCP y a su vez en el rendimiento de la red; para TCP Westwood el comportamiento es diferente, ya que para una duración de trama más grande no se presentan paquetes encolados ni retardo de encolamiento debido que el número de paquetes que entran al búfer es igual a los que salen obteniendo un RTT mucho menor. Una descripción más detallada de cada versión TCP remitirse a los anexos C, D y E.

### 3.2.3.4. Análisis del *throughput* y capacidad del enlace utilizado.

En las figuras 3-22 se observa el *throughput* de la red para las diferentes pruebas donde se concluye que para una duración de trama más pequeña presenta un mejor rendimiento de la red, siempre y cuando se conserve un TBTP alrededor de un segundo.

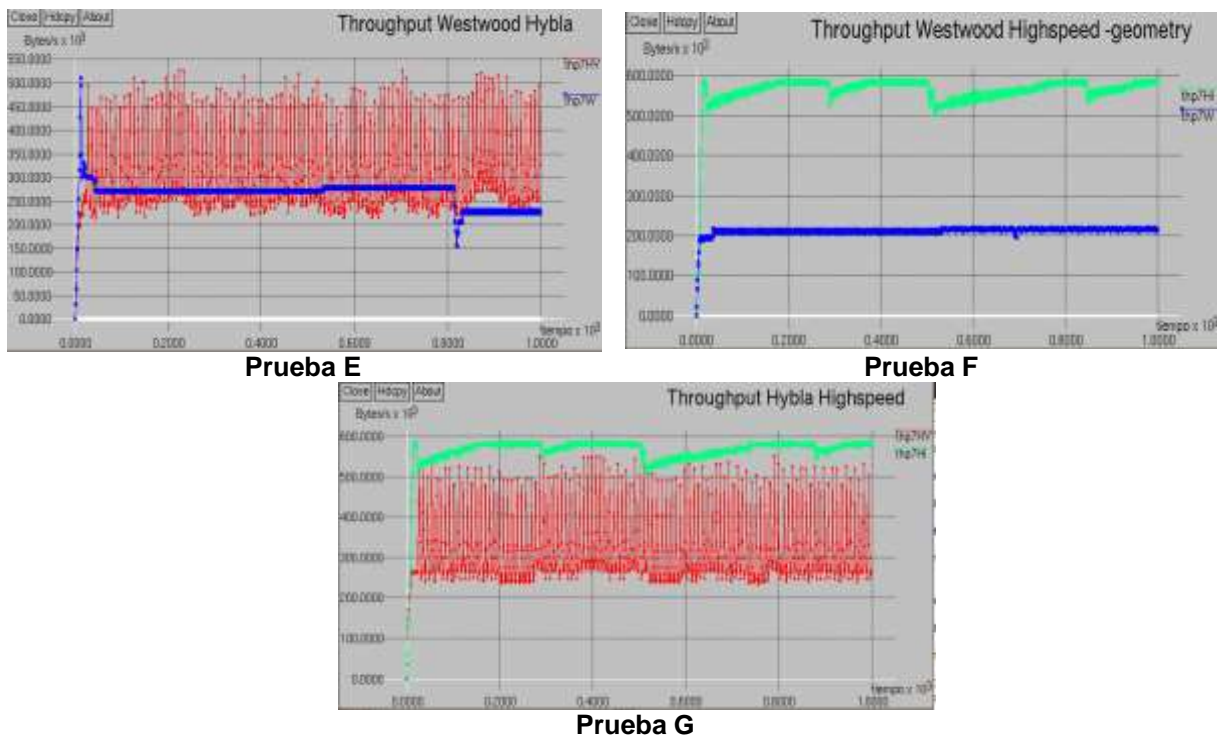


Figura 3-22. *Throughput* de la red para las tres versiones de TCP analizadas.

Se tiene también que para una duración de trama pequeña tiene una mejor utilización total del canal, esto debido al buen desempeño que tiene TCP, como se observa en la figuras 3-23 y 3-24.

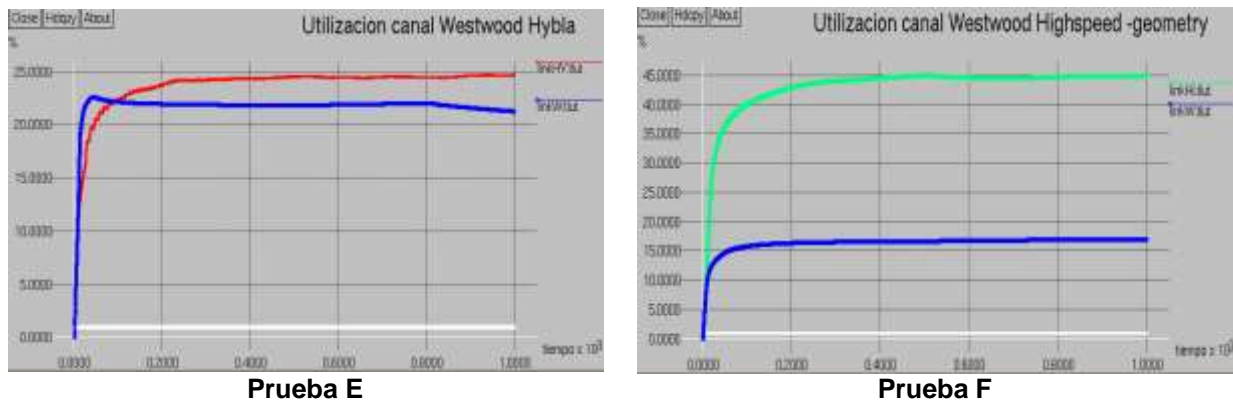


Figura 3-23. Pruebas E y F para la utilización de canal por todas las conexiones de la red.

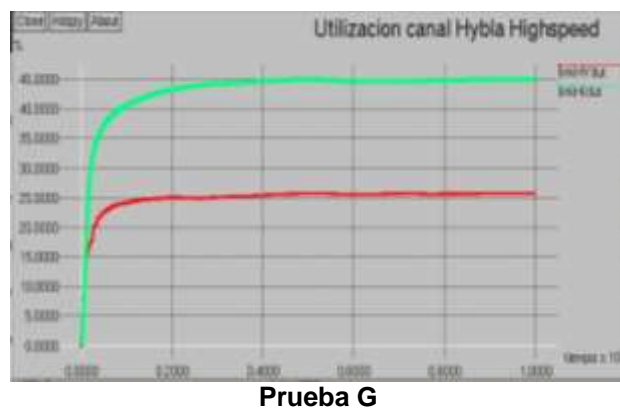


Figura 3-24. Utilización de canal por todas las conexiones de la red en la prueba G.

Como conclusión se afirma que el variar el tamaño de la trama impacta de manera más significativa a TCPW, que en las otras versiones.

### 3.2.4. Impacto al Variar el Tiempo de Reprogramación TBTP en TDMA.

En este caso de estudio se mide el impacto que presenta el variar el tiempo de planificación para la asignación de recursos en el desempeño del protocolo TCP, configurando los valores de las tablas 3-6 y 3-7 para las diferentes versiones TCP. El único dato que se modifica en este caso es el número de supertramas que permite fijar el valor del tiempo de reprogramación TBTP en la estación de control. En [70] se sugiere un TBTP alrededor de 1 segundo, para obtener los valores del TBTP en las pruebas se multiplica el número de tramas por supertramas por la duración de la trama.

Tabla 3-6. Impacto de TCP Westwood al variar el TBTP.

	Pruebas		
Parámetros a modificar	H	I	
Tamaño Slot	1504	1504	<b>FIJOS</b>
Ancho de Banda	4.812.800	4.812.800	
N° de Slots por trama	20	20	
Duración trama	0,05	0,05	
Supertrama	<b>10</b>	<b>30</b>	<b>VARIABLE</b>

Tabla 3-7. Impacto de TCP Hybla y TCP HS al variar el TBTP.

	Prueba	
Parámetros a modificar	J	
Tamaño Slot	534	<b>FIJOS</b>
Ancho de Banda	4.101.120	
N° de Slots por trama	24	
Duración trama	0,025	
Supertrama	<b>32</b>	<b>VARIABLE</b>

### 3.2.4.1. Análisis de la ventana de congestión, el *ssthresh* y paquetes retransmitidos.

En la figura 3-25 se observa el comportamiento de *CWND* para diferentes tiempos del TBTP, donde el impacto que se tiene en *CWND* es poco significativo, ya que para las tres versiones de TCP su comportamiento es similar.

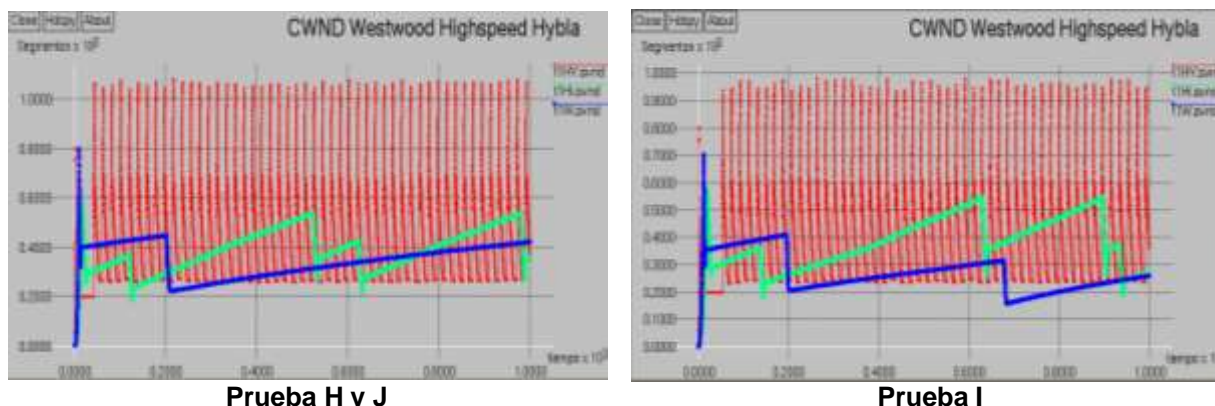
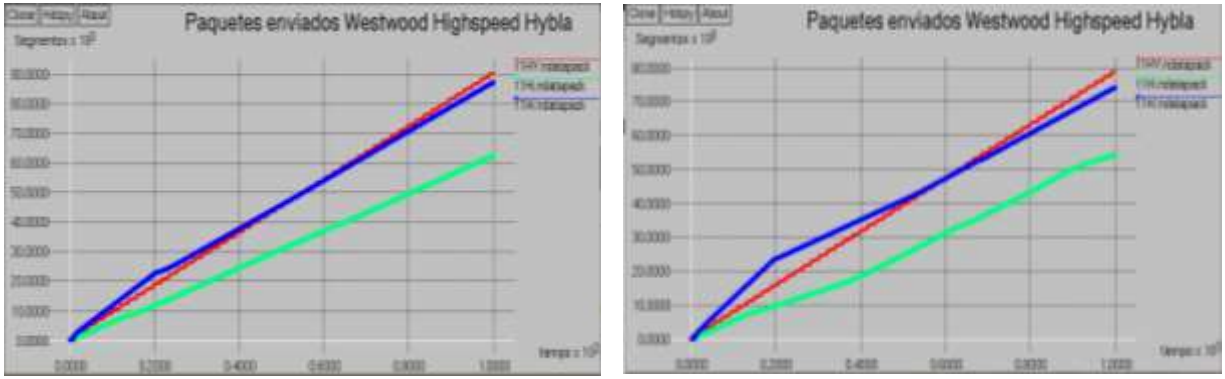


Figura 3-25. Comportamiento de *CWND* para diferentes TBTP.



Sin embargo se tiene que para un menor tiempo del TBTP el crecimiento de *CWND* en la etapa de inicio lento y evasión de congestión es más rápido, con pendientes más pronunciadas que para TBTPs mayores. En consecuencia se tiene que hay más retransmisiones de paquetes donde la gran mayoría de estos se debe a un desbordamiento del búfer tal y como se analizará más adelante.

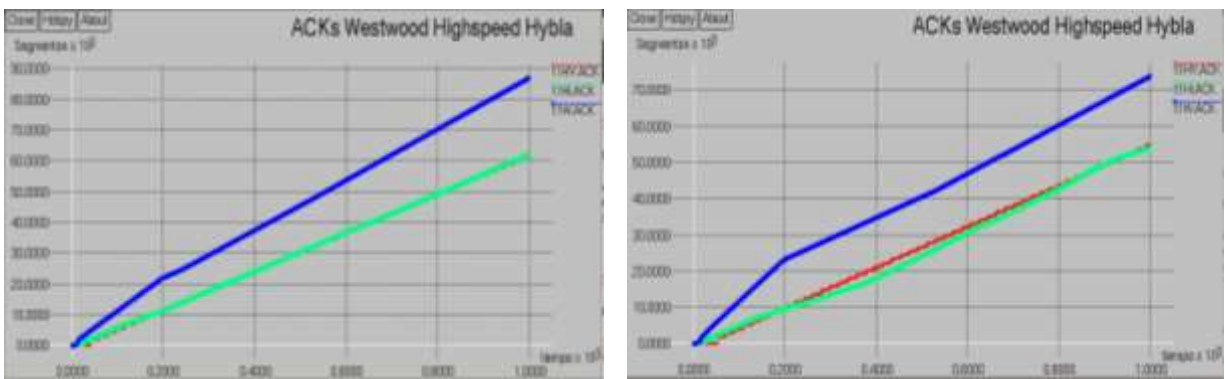
Por otro lado se tiene que al variar el TBTP también impacta sobre los paquetes transmitidos, a TBTP mayores se va a tener menos paquetes transmitidos el cual concuerda con el comportamiento de *CWND*, esto se observa en las figuras 3-25 y 3-26.



**Prueba H y J** **Prueba I**  
 Figura 3-26. Comportamiento de los paquetes transmitidos.

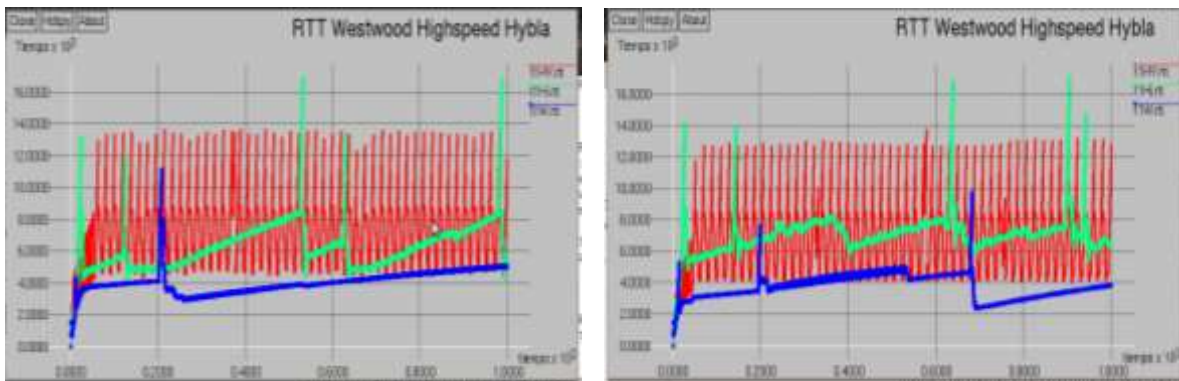
### 3.2.4.2. Análisis de ACK, RTT y número de secuencia.

Para las tres versiones de TCP analizadas se tiene el mismo comportamiento, dado que a un menor tiempo del TBTP se tiene mayor número de ACKs que arriban al transmisor tal como observa en la figura 3-27, lo anterior concuerda con el impacto que tiene la ventana de congestión en estas pruebas.



**Prueba H y J** **Prueba I**  
 Figura 3-27. Comportamiento de los ACKs.

Por otro lado se tiene que el RTT presenta un comportamiento casi igual para las diferentes pruebas de TBTP donde el impacto sobre este parámetro no es muy notorio, esto se observa en la figura 3-28.



Prueba H y J

Prueba I

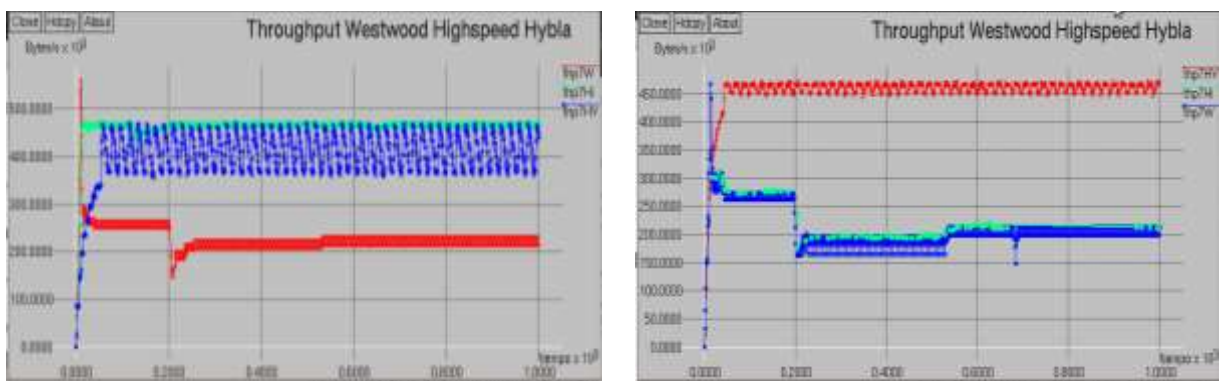
Figura 3-28. Comportamiento del RTT.

### 3.2.4.3. Análisis del encolamiento y retardo de la red.

En los anexos C, D y E en las figuras<sup>4</sup> se aprecia el comportamiento que tienen los paquetes encolados y el retardo de la red, donde se tiene que para valores de TBTP que se encuentran alrededor de un segundo se presenta un retardo menor, el cual se ve reflejado en un mejor comportamiento del búfer.

### 3.2.4.4. Análisis del *throughput* y capacidad del enlace utilizado.

En la figura 3-29 se observa el comportamiento de *throughput* para las diferentes pruebas de TBTP donde se tiene que éste no afecta de manera significativa el rendimiento de la red, debido a que las redes satelitales GEO presentan altos tiempos de propagación, el TBTP no requiere ser actualizado de manera muy frecuente [70].



Prueba H y J

Prueba I

Figura 3-29. Comportamiento del *throughput* para distintos valores de TBTP.

<sup>4</sup> Ver sección 4.3 anexo E y sección 4.4 para los anexos C y D

Por otro lado se tiene que la utilización del canal no se ve tan afectada por una variación del TBTP, ver figura 3-30, más bien se le atribuye al tipo de asignación de recursos que tenga cada estación terrena dentro de la red satelital.

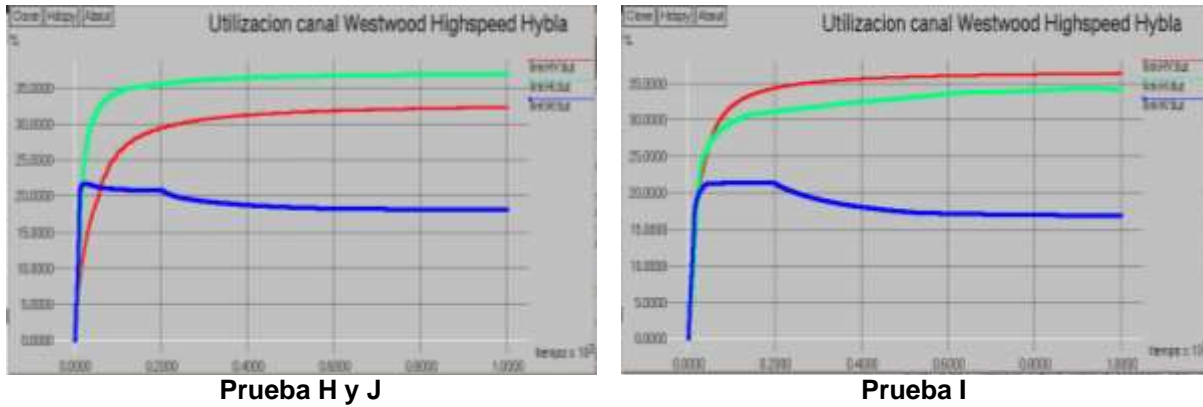


Figura 3-30. Utilización del canal.

### 3.2.5. Impacto al variar el Tipo de Asignación de Recursos en la NCC.

En este caso de estudio se utilizó la asignación de recursos en la NCC *Proportional*, considerando la prueba que presentó el mejor desempeño en la red de acuerdo al análisis anterior, y se comparó con la prueba B que emplea la asignación FODA como lo indica la tabla 3-8.

Tabla 3-8. Impacto de TCP al variar el tipo de asignación de recursos.

Mecanismos y parámetros a modificar	Prueba		
	B	K	
Duración trama	0,05	0,05	<b>FIJOS</b>
Supertrama	20	20	
N° de Slots por trama	20	20	
Tamaño Slot	1504	1504	
Ancho de Banda	4.812.800	4.812.800	
Política de asignación de recursos en la NCC	<b>FODA</b>	<i>Proportional</i>	<b>VARIABLE</b>

### 3.2.5.1. Análisis de la ventana de congestión, *ssthresh*, paquetes transmitidos y retransmitidos.

En las figura 3-31 se observa el impacto que tiene el asignar una política u otra sobre el comportamiento de *CWND* en las tres versiones de TCP, donde se aprecia que para una asignación de recursos tipo Proporcional el crecimiento de *CWND* es mucho más rápido tanto en la etapa de evasión de congestión, como en la fase de inicio lento, mientras que para una asignación tipo FODA *CWND* le cuesta recuperarse cuando se presenta una pérdida y por ende volver a recuperar el nivel que tenía anteriormente.

*CWND* tiene una recuperación más rápida con la presencia de ACKs duplicados en la política de asignación de recursos proporcional, porque hace una asignación de recursos más equitativa en las terminales durante todo el tiempo, lo que le permite a *CWND* crecer más rápido y no tener que esperar a que se le asigne recursos como lo hace la política FODA.

Por otro lado se tiene que con la política de asignación de recursos proporcional los paquetes transmitidos crecen de manera más rápida para todas las conexiones de la red gracias al buen comportamiento de *CWND*, también se observó una asignación de recursos mucho más equitativa en las diferentes estaciones terrenas, esto se observa en las figuras 3-32.

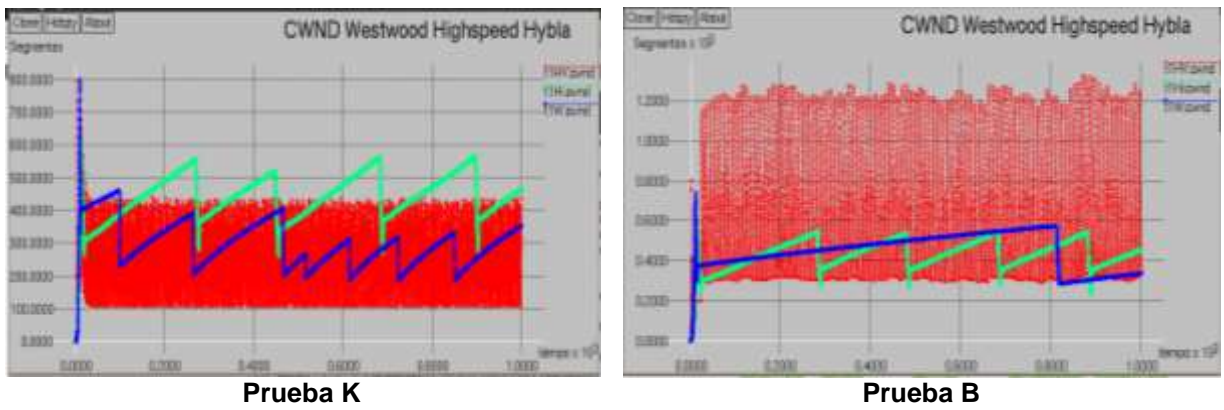


Figura 3-31. Comportamiento de la ventana de congestión.

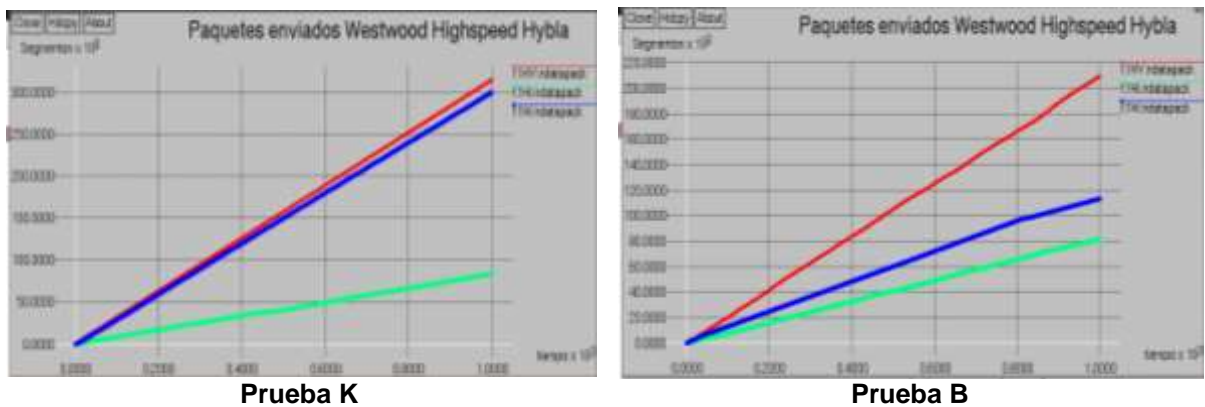


Figura 3-32. Comportamiento de los paquetes transmitidos.

### 3.2.5.2. Análisis de ACK, y RTT.

Debido al impacto positivo que tiene la asignación proporcional sobre *CWND*, esta se ve reflejada de la misma forma en el comportamiento de los ACKs, donde se tiene un crecimiento más rápido y lineal de ACKs que arriban al transmisor para todas las conexiones, caso que no ocurre para FODA debido a la manera poco equitativa en que se asignan los recursos, esto se observa en la figura 3-33.

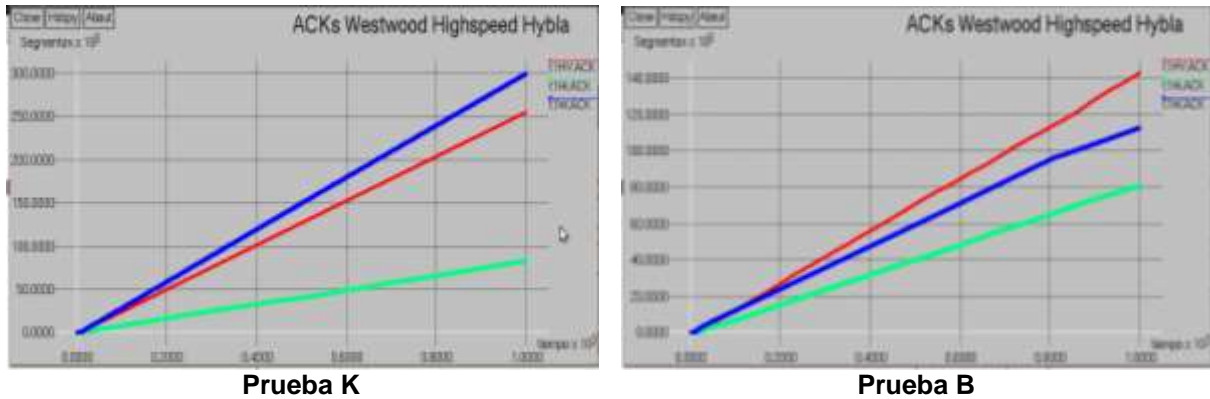


Figura 3-33. Comportamiento de los ACKs.

El impacto que se genera el tener una asignación proporcional sobre el RTT es un menor valor en comparación con la política FODA, esto se puede apreciar en la figura 3-34.

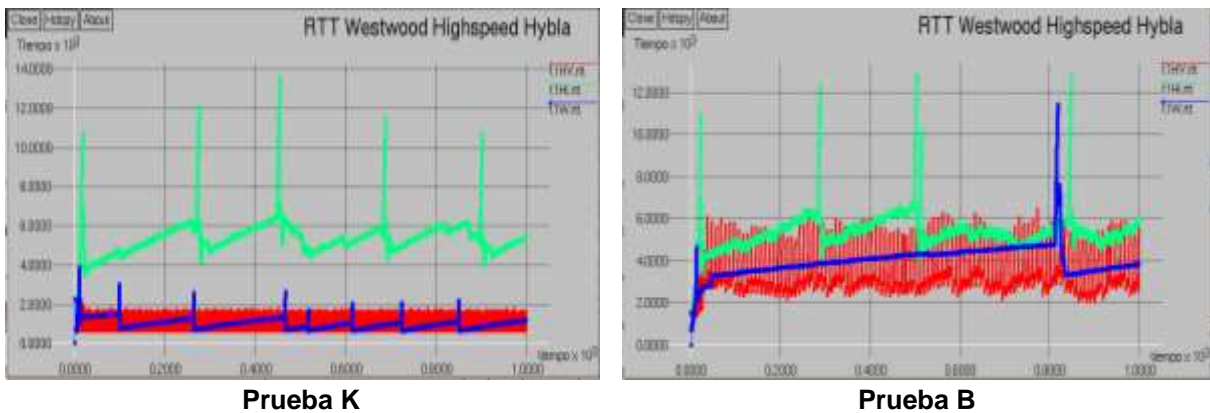


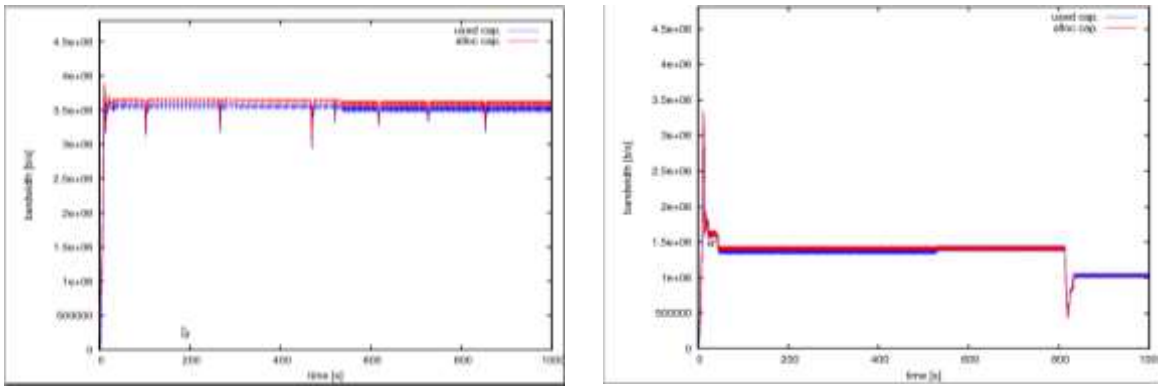
Figura 3-34. Comportamiento del RTT.

### 3.2.5.3. Análisis del encolamiento y retardo de la red.

Para visualizar cuál es el impacto que se genera en el búfer al utilizar una política de asignación de recursos, se analiza el archivo *curral.tr* generado por la simulación, este indica como ha sido la asignación de los slots para cada una de las estaciones teniendo en cuenta las solicitudes hechas a la NCC, se observa que al implementar el tipo de asignación Proporcional se implementa un criterio más equitativo para todas las conexiones, es decir el número total de slots son repartidos a todas las conexiones en cada duración de trama de manera uniforme, y de esta manera evita que cualquier estación monopolice la capacidad de asignación, caso contrario al que se observa al implementar la política FODA, además para algunos intervalos



esta asignación es cero, lo que implica un mayor encolamiento de los paquetes, y un aumento en el retardo de la red que se ve reflejado en el desempeño del protocolo TCP, en la figura 3-35 se muestra un ejemplo para la versión de TCP Westwood.



Utilización de recursos en la prueba K

Utilización de recursos en la prueba B

Figura 3-35. Asignación y utilización de recursos para TCP Westwood.

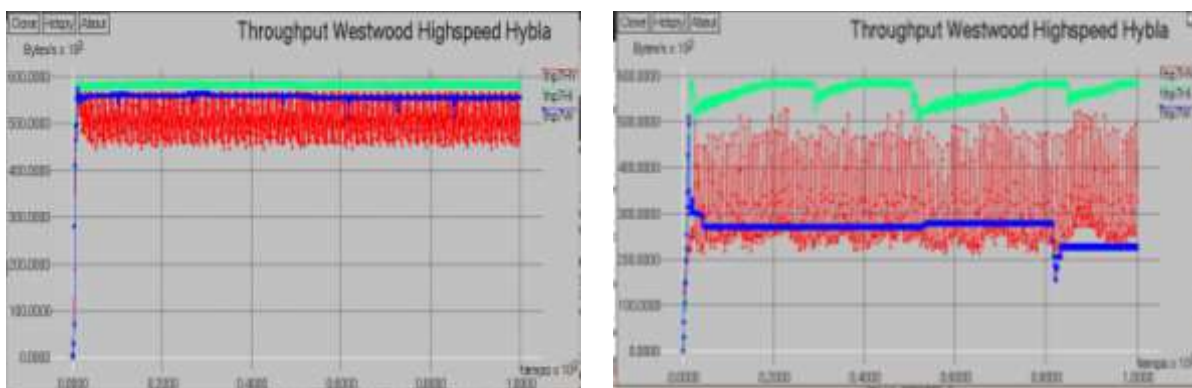
Por tanto para una política de asignación de recursos proporcional se tiene un menor retardo en la red, un mayor número de paquetes que entran y salen del búfer, el cual se ve reflejado en un buen desempeño de TCP caso contrario para la política FODA.

En los anexos C, D y E se puede observar una descripción más detallada de cómo influye de manera particular el tipo de asignación en las versiones de estudio TCP.

### 3.2.5.4. Análisis del desempeño de la red.

Los resultados indican que al utilizar una asignación tipo “Proporcional” se obtiene un mejor aprovechamiento de la capacidad del canal, además de aumentar el rendimiento de todas las conexiones y por ende el *throughput* de la red tal como se muestra en las figuras 3-36 y 3-37.

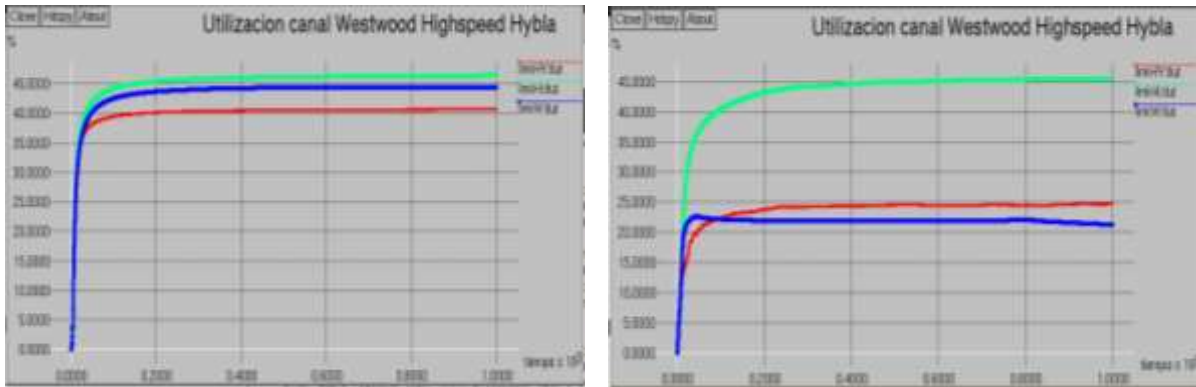
Se concluye que el desempeño de TCP en sus diferentes versiones se ve afectado de manera positiva o negativa de acuerdo al tipo de política de asignación de recursos que se escoja, los parámetros que se ven más afectados son la ventana de congestión y el RTT, por otro lado se ve también afectado el encolamiento de búfer y el retardo de la red los cuales son factores que contribuyen a un buen o mal desempeño del protocolo TCP.



Prueba K

Prueba B

Figura 3-36. *Throughput* de la red observado en las pruebas B y K.



**Prueba K** **Prueba B**  
 Figura 3-37. Utilización del canal por todas las conexiones.

### 3.2.6. Impacto al Utilizar los Diferentes Tipos de Asignación Bajo Demanda en TDMA.

En esta sección se analiza cuál es el impacto que tienen las políticas de asignación por demanda que se describieron en el capítulo dos sobre los parámetros y desempeño del TCP Westwood, Hybla y HighSpeed, como también se evalúa cual es el rendimiento de la red, la utilización del canal satelital y cómo influye el encolamiento en los parámetros TCP, para ello se realizan las pruebas de la tabla 3.9.

Tabla 3-9. Impacto de TCP al variar los diferentes tipos de asignación bajo demanda.

Mecanismos y parámetros a modificar	Pruebas				
	L	M	N	O	
Duración trama	0,05	0,05	0,05	0,05	<b>FIJOS</b>
Supertrama	25	25	25	25	
N° de Slots por trama	20	20	20	20	
Tamaño Slot	1504	1504	1504	1504	
Ancho de Banda	4.812.800	4.812.800	4.812.800	4.812.800	
Política de asignación de recursos en la NCC	<i>Proportional</i>	<i>Proportional</i>	<i>Proportional</i>	<i>Proportional</i>	
Políticas de solicitud de recursos bajo demanda	<b>CRA</b>	<b>RBDC</b>	<b>VBDC</b>	<b>RVBDC</b>	<b>VARIABLE</b>

#### 3.2.6.1. Análisis para TCP hybla.

Mediante la siguiente prueba se pretende determinar cuál política de asignación de demanda presenta un mejor comportamiento en los parámetros TCP Hybla y cual obtiene un mejor desempeño en la red, todas las conexiones fueron configuradas con la misma política de

asignación. Los datos almacenados en las tablas 3-10, 3-11 y 3-12 indican el comportamiento de las cuatro políticas teniendo en cuenta los parámetros TCP y su desempeño.

Tabla 3-10. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión uno.

Prueba	CWND	RTT	Pktx	Link	Encolamiento	Retardo	Ndatapack
O	540.10	6177.14	43791	8.74	346.83	4.41	119376
M	223.78	116042.32	17771	3.44	311.718	109	47141
L	545.87	7127.22	39407	7.81	357.16	5.06	106893
N	559.17	6055.02	47526	9.30	360.32	4.31	128060

Tabla 3-11. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión dos.

Política	Ventana	RTT	Pktx	Link	Encolamiento	Retardo	Ndatapack
O	524.11	6213.02	43283	8.74	345.30	4.40	118011
M	142.16	172863.73	5118	0.86	272.06	155	12503
L	514.46	6998.60	38902	7.81	348.06	4.91	105666
N	556.40	6019.92	47485	9.39	359.45	4.23	128545

Tabla 3-12. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión tres.

Política	Ventana	RTT	Pktx	Link	Encolamiento	Retardo	Ndatapack
O	521.64	6163.43	44038	8.91	342.96	4.34	119287
M	499.36	3936.72	66609	17.33	298.34	2.91	214558
L	513.54	6967.66	38537	7.81	346.93	4.91	105301
N	544.15	6057.11	46020	9.16	351.04	4.20	125308

En la tabla 3-13, se indica el *throughput* promedio y la utilización total del canal para todas las conexiones para las diferentes políticas.

Tabla 3-13. *Throughput* promedio y utilización del canal en las pruebas L, M, N y O.

Política	<i>Throughput</i>	Utilización del canal
O	561809.21	44.95
M	551589.77	44.13
L	488247.47	39.06
N	576926.69	46.29



**a) Análisis de la ventana de congestión para las cuatro políticas de asignación.**

La política VBDC, presenta el mejor comportamiento de ventana de congestión lo que indica que un mayor volumen de datos se ha transmitido por la fuente origen; presentando una menor latencia en la red. Para RBDC las ventanas de congestión dependen mucho de cómo se distribuyen los recursos en las distintas conexiones y de la ausencia o presencia del *backoff*. Cuando la utilización del canal es buena la ventana de congestión es muy oscilante y su nivel de *ssthresh* es alto, en caso contrario el desempeño se ve muy afectado por los elevados RTTs, y retardo de la red.

Las políticas RVBDC y CRA, presentan un tamaño de ventana muy similar a VBDC pero a un nivel inferior.

**b) Análisis del RTT para las cuatro políticas de asignación.**

El RTT de la política VBDC presentó el mejor desempeño tal y como se muestra en las tablas 3-10, 3-11 y 3-12, esto se ve reflejado en un incremento en los paquetes transmitidos, donde no son afectados por la congestión que experimenta la red. La política RVBDC, presenta un rendimiento muy similar a VBDC, mientras que para CRA se incrementa un poco más, en RBDC el desempeño depende del porcentaje de utilización del canal, aunque su rendimiento en la red es bueno.

**c) Análisis de los paquetes retransmitidos para las cuatro políticas de asignación.**

Los paquetes retransmitidos aumentan de la misma forma en que la ventana crece, por lo tanto la política CRA presenta el menor número de paquetes retransmitidos, luego RVBDC, VBDC y por último RBDC; la política RBDC presentó el peor rendimiento si se tiene en cuenta el corto tiempo que el canal está siendo utilizado por la conexión.

**d) Análisis de los recursos asignados por las cuatro políticas en las diferentes conexiones.**

La cantidad de recursos asignados para cada una de las conexiones es similar entre las políticas CRA, RVBDC y VBDC, la política RBDC fue poco equitativa en la distribución de los recursos, lo que conlleva a que una conexión monopolice la red.

**e) Análisis del retardo de de la red para las cuatro políticas de asignación.**

El retardo de propagación es un excelente parámetro para observar que nivel de congestión está presentando la red, si se analizan las tablas 3-10, 3-11 y 3-12; las políticas que presentaron un buen comportamiento fueron VBDC, RVBDC y CRA. Obteniendo el mejor resultado con la política VBDC, y el peor caso en RBDC.

**f) Análisis del encolamiento para las cuatro políticas de asignación.**

Un mayor encolamiento implica que un gran volumen de datos tendrán que transmitirse para disminuir la cola del búfer, entre menor sea esta, la red no se encuentra congestionada y los datos en la medida que llegan al búfer serán transmitidos, sin embargo como las estaciones tienen que esperar su momento para transmitir según sean los slots asignados, el encolamiento en el búfer por lo general siempre está presente, por esta razón entre mayor sean los slots

asignados, el búfer se desocupará más rápido. Si el comportamiento de búfer es oscilante y menos espaciado está indicando que un mayor volumen de datos han sido transmitidos, cuando el transmisor se da cuenta que ha ocurrido una pérdida, los paquetes a transmitir se disminuyen y solo se encuentran los que están en transición, estos paquetes son los que están saliendo de la cola, entre más rápido se desocupe podrán ser reconocidos por el transmisor un mayor número de datos y de esta manera se incrementará nuevamente su ventana de congestión; según lo observado en los resultados obtenidos VBDC presenta el mejor encolamiento, seguido RVBDC y CRA, el peor caso fue para RBDC.

**g) Análisis del número de paquetes transmitidos para las cuatro políticas de asignación.**

La variable *ndatapack*, indica de cierta manera cuál ha sido el comportamiento de la ventana de congestión. Las políticas que presentaron el mejor rendimiento son VBDC, RVBDC y CRA como lo muestran las tablas 3-10, 3-11 y 3-12, el peor caso fue para RBDC.

**h) Análisis del rendimiento y utilización total del canal para todas las conexiones según las cuatro políticas de asignación.**

En la tabla 3-13 se muestra cuál fue el comportamiento del *throughput* promedio obtenido por las cuatros políticas.

**3.2.6.2. Análisis para TCP HighSpeed.**

Mediante la siguiente prueba se quiere determinar cuál política de asignación bajo demanda presentó el mejor comportamiento con la versión TCP HS, todas las conexiones fueron configuradas con la misma política de asignación.

Tabla 3-14. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión uno.

	<i>CWND</i>	<i>RTT</i>	<i>Pktx</i>	<i>Link</i>	<i>Encolamiento</i>	<i>Retardo</i>	<i>Ndatapack</i>
<b>O</b>	388.42	5266.21	54	8.92	339.93	4.51	76275
<b>M</b>	390.81	7405.03	59	7.61	349.13	6.60	65266
<b>L</b>	413.90	6315.87	52	7.80	370.77	5.56	66760
<b>N</b>	385.82	5352.49	53	8.54	338.69	4.63	73207

Tabla 3-15. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión dos.

	<i>CWND</i>	<i>RTT</i>	<i>Pktx</i>	<i>Link</i>	<i>Encolamiento</i>	<i>Retardo</i>	<i>ndatapack</i>
<b>O</b>	420.66	5659.97	67	8.92	371.68	4.91	76324
<b>M</b>	186.67	130336.13	296	3.19	274.511	65.53	27495
<b>L</b>	432.06	6564.24	50	7.80	389.30	5.78	66874
<b>N</b>	439.21	5350.23	52	9.68	385.82	4.65	82843

Tabla 3-16. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión tres.

	<b>CWND</b>	<b>RTT</b>	<b>Pktx</b>	<b>Link</b>	<b>Encolamiento</b>	<b>Retardo</b>	<b>ndatapack</b>
<b>O</b>	415.31	5595.48	58	8.88	366.78	4.84	76739
<b>M</b>	432.70	6072.68	62	8.92	383.95	5.33	76336
<b>L</b>	405.04	6182.88	41	7.80	362.73	5.44	66739
<b>N</b>	414.75	5340.04	53	9.19	364.92	4.64	78664

Tabla 3-17. *Throughput* promedio y utilización del canal en las pruebas L, M, N y O.

	<b>Throughput</b>	<b>Utilización del canal</b>
<b>O</b>	557200.76	44.58
<b>M</b>	545965.55	43.69
<b>L</b>	487875.56	39.03
<b>N</b>	577116.32	46.18

**a) Análisis de la ventana de congestión para las cuatro políticas de asignación.**

La política de asignación que presentó el mejor rendimiento es CRA sin embargo esta se ve afectada por un mayor nivel de congestión. Las políticas VBDC, RVBDC presentaron un comportamiento un poco inferior a CRA, aunque alcanzaron un menor retardo. La política RBDC mejora el rendimiento notablemente al compararse con la versión de TCP Hybla, su ventana tiene un mejor desempeño cuando la utilización del canal es buena.

**b) Análisis del RTT para las cuatro políticas de asignación.**

EL RTT es uno de los factores más influyentes para que la ventana de congestión obtenga un buen desempeño, la política VBDC presentó el mejor comportamiento, por lo tanto la congestión en la red es menor. En segundo lugar se encuentra RVBDC, luego la política CRA y por último RBDC tal como lo indican las tablas 3-14, 3-15 y 3-16.

**c) Análisis de paquetes retransmitidos para las cuatro políticas de asignación.**

Un menor número de paquetes retransmitidos se presenta para la política CRA, donde también se observa que el número de paquetes transmitidos es menor; las políticas con un mayor volumen de datos a transmitir presentan más paquetes retransmitidos.

**d) Análisis de los recursos asignados por las cuatro políticas en las diferentes conexiones.**

Los resultados obtenidos demostraron que al utilizar la política CRA y RVBDC asignación de recurso es ecuánime, el peor caso se registró con RBDC donde los slots asignados no es equitativo y se tiende a monopolizar el canal por algunas conexiones.

#### **e) Análisis del encolamiento para las cuatro políticas de asignación.**

Para llegar a una conclusión, se debe tener en cuenta que algunas políticas tienen mayor número de paquetes encolados debido a que más slots son asignados, para visualizar esto se debe ver el número de paquetes de datos transmitidos, además dependiendo del retardo de encolamiento indicará que tan rápido es la transmisión, si es lento implica que muchos paquetes son encolados debido a la congestión pero si es rápida demuestra lo contrario; por lo tanto para determinar qué política presenta un mejor encolamiento se tiene que analizar el número de paquetes transmitidos y el retardo de la red, de acuerdo a lo anterior se tiene que VBDC presentó el mejor encolamiento debido a que más paquetes son transmitidos y presenta un menor retardo, en segundo lugar está RVBDC debido a su comportamiento similar con VBDC, en tercer lugar se encuentra CRA el cual obtiene un gran volumen de transmisión, sin embargo sus slots son muy pocos para transmitir esta información y por ello su retardo aumenta, por último se encuentra RBDC.

#### **f) Análisis del retardo de de la red para las cuatro políticas de asignación.**

El retardo es un parámetro que directamente se relaciona con el nivel de congestión que se presenta en la red, según las tablas 3-14, 3-15 y 3-16 indican que un menor retardo se presentó al utilizar la política VBDC, luego RVBDC y en último lugar fue para RBDC, cabe resaltar que al utilizar la versión de TCP HS, el comportamiento de esta política mejora en la mayoría de los parámetros de evaluación.

#### **g) Análisis del número de paquetes transmitidos para las cuatro políticas de asignación.**

La variable *ndatapack* se comporta muy similar en todas las políticas a excepción de RBDC, las tablas 3-14, 3-15 y 3-16, indican que TCP HS presentó un mejor desempeño.

#### **h) Análisis del rendimiento y utilización total del canal para todas las conexiones según las cuatro políticas de asignación.**

Teniendo en cuenta la tabla 3-17 el rendimiento y utilización del canal para las políticas VBDC y RVBDC presentaron el mejor comportamiento con respecto a RBDC, tanto su *throughput* como la utilización es muy similar.

Para determinar qué política de asignación presentó el mejor rendimiento depende de muchos parámetros y principalmente del tipo de tráfico a utilizar. La política de asignación CRA hace uso de una menor capacidad, sin embargo los recursos asignados en sus conexiones son equitativos y presentó un menor número de paquetes retransmitidos; la utilización del canal permanece en un mismo nivel, sin embargo su RTT es más alto que las políticas AVBDC y VBDC, además presentó el peor rendimiento de las 4 políticas.

El uso de la política RBDC, se caracterizó por la inequidad de los recursos compartidos, presencia de *timeouts* debido a retransmisiones y aumento de los paquetes retransmitidos. Las conexiones que obtienen un buen porcentaje de la utilización total del enlace se caracteriza por un RTT bajo y no tan variante. El *throughput* que obtiene esta política junto con la utilización del canal es apropiado, se comporta casi de igual manera que en las políticas RVBDC y VBDC.

Las simulaciones realizadas con la política RVBDC, hacen uso de la capacidad del enlace de subida de una forma casi uniforme y constante para todas sus conexiones, el RTT es bajo, su

retardo es aceptable, sin embargo en la medida que se encuentre un mayor número de paquetes encolados este aumenta, por lo tanto el comportamiento de su ventana de congestión es muy bueno.

La política VBDC, en la mayoría de las simulaciones presentó un buen encolamiento donde se observa un mayor número de paquetes transmitidos, sus ventanas de transmisión crecen más rápido y el retardo de la red es pequeño. Las solicitudes realizadas a la NCC por las estaciones son similares. El número de paquetes retransmitidos es casi igual que en la política CRA a pesar de realizar un mayor volumen de transmisión de datos

### 3.2.6.3. Análisis de TCP Westwood.

#### a) Política RVBDC.

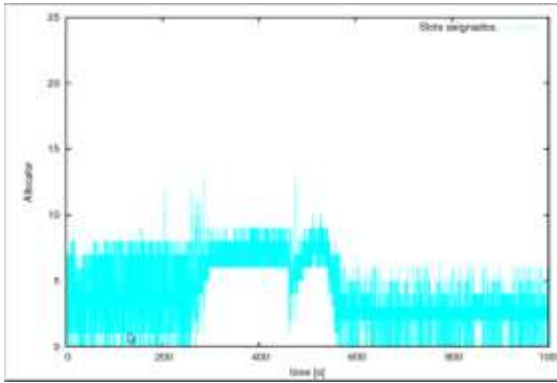
Como se describió en el capítulo dos esta política de asignación por demanda realiza peticiones absolutas las cuales cambian en el tiempo en función del búfer para poder vaciar la cola de la estación terrena. Para poder evaluar cuál es el impacto de esta política se configuraron las estaciones terrenas con RVBDC, y para la estación de control con la política de asignación proporcional.

Tabla 3-18. Comportamiento de *CWND*, *ssthresh*, PKTX y ACKs.

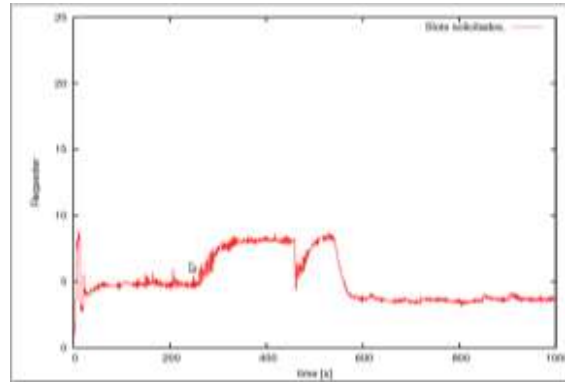
Tiempo (s)	<i>CWND</i> (Segmentos)	<i>Ssthresh</i>	PKTX	ACKs
(0 - 12,8)	0 - 602	800	0	559
(12,9 - 20)	602 - 267	301	52	601
(20,9 - 456)	300 - 426	301	1	42000
(461 - 1000)	214 - 344	213	0	83592

En la tabla 3-18 se observa el comportamiento de algunos parámetros de TCPW, donde se nota que *CWND* crece de manera lineal en casi todos los intervalos de tiempo con una pendiente suave, únicamente se presentan 53 retransmisiones de paquetes durante todo el tiempo de simulación.

Al observar la figura 3-38a y el archivo *curral.tr* se aprecia que las solicitudes y la asignación de recursos (slots de tiempo) se hace de manera casi equitativa entre las diferentes estaciones terrenas (ver figura 3-38b utilización del canal para las diferentes conexiones), se puede ver también que hay un número pequeño de slots que no se usan, lo que implica que se están haciendo una buena utilización de los recursos disponibles por la red, se observa también que el encolamiento (véase figura 3-39 encolamiento RVBDC) está muy relacionado con el número de slots solicitados que hacen las estaciones terrenas para vaciar sus búferes, a un mayor encolamiento se tendrá una mayor petición de slots (ver intervalo de tiempo 300 - 550).

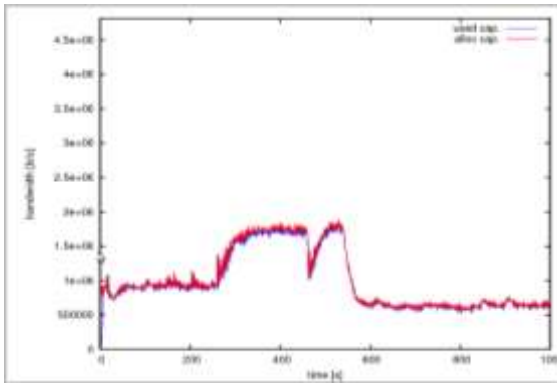


Slots asignados a una conexión

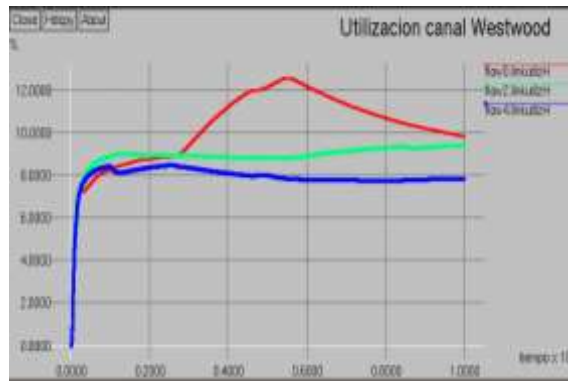


Slots solicitados a una conexión

Figura 3-38a. Slots solicitados y asignados con la política RVBDC.



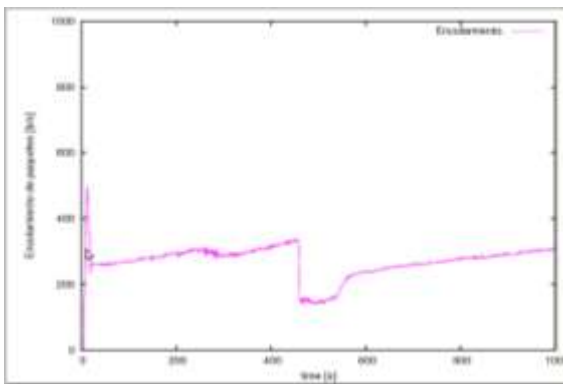
Ancho de banda utilizado y asignado



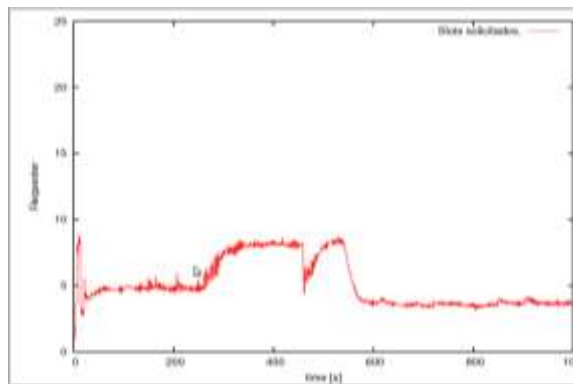
Utilización del canal por varias conexiones  
flujo0 flujo2 flujo4

Figura 3-38b. Ancho de banda *uplink* y utilización del canal empleado con RVBDC.

Los datos descritos anteriormente están relacionados con el comportamiento de *CWND*, *ACKs*, *PKTX* y el *RTT*, por lo que el impacto que tiene esta política sobre los parámetros TCP es un comportamiento casi uniforme en todas las conexiones.



Paquetes encolados



Slots solicitados

Figura 3-39. Relación entre los paquetes encolados y los recursos solicitados por la RCST.

En las figuras 3-40a y 3-40b se observa que la utilización del enlace con esta política es del 44% al 45% en promedio, con un *throughput* promedio de 560000 Bytes/s y un retardo de 4.7seg.

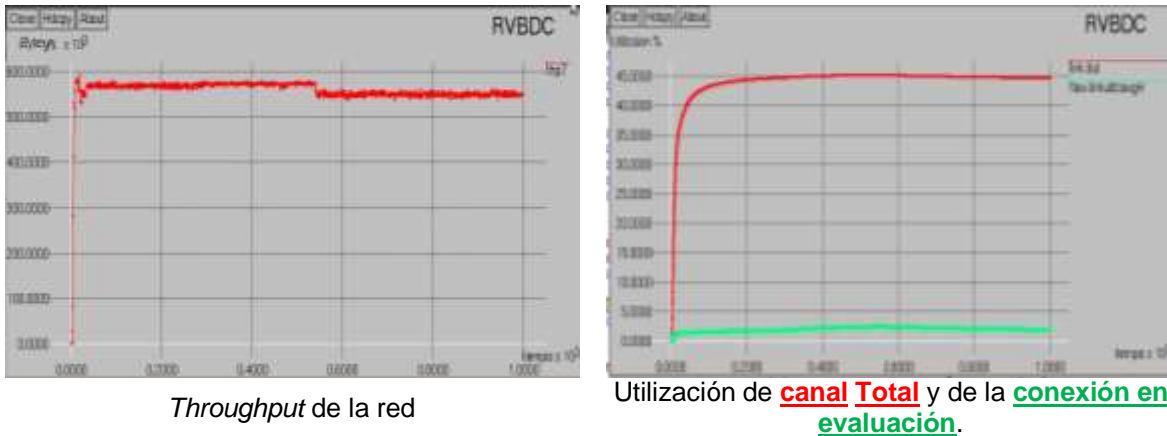


Figura 3-40a. *Throughput* y utilización del canal con la política RVBDC.

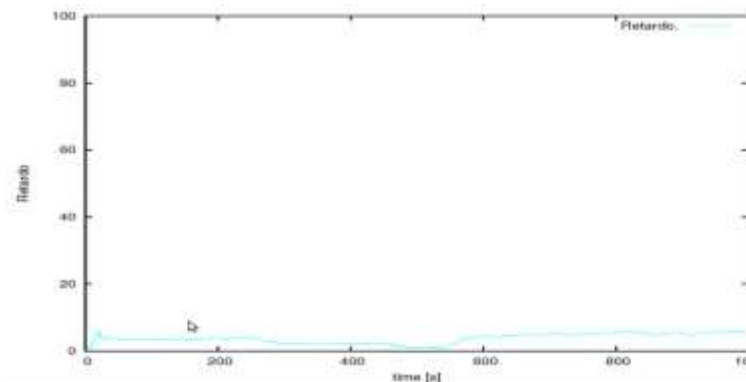


Figura 3-40b. Retardo en la red con RVBDC.

**b) Política RBDC.**

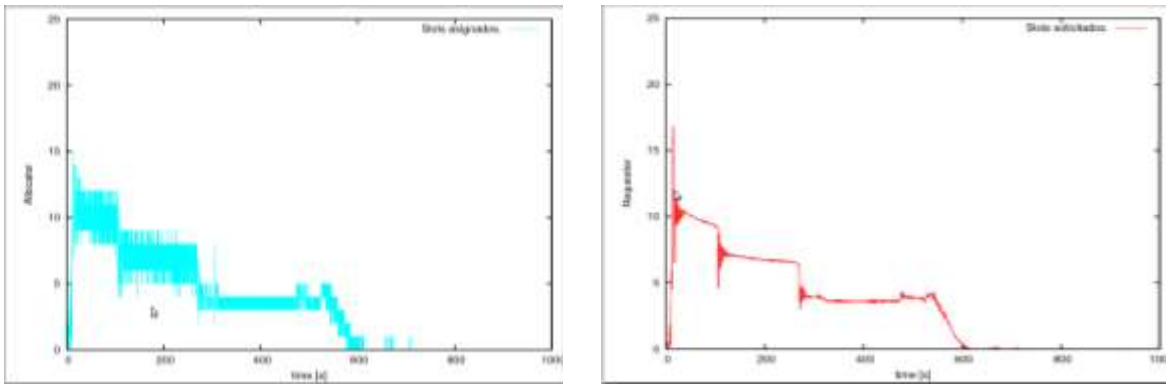
En la tabla 3-19 se muestran el comportamiento de *CWND*, el *ssthresh* y los ACKs para esta política.

Tabla 3-19. Comportamiento de algunos parámetros TCP.

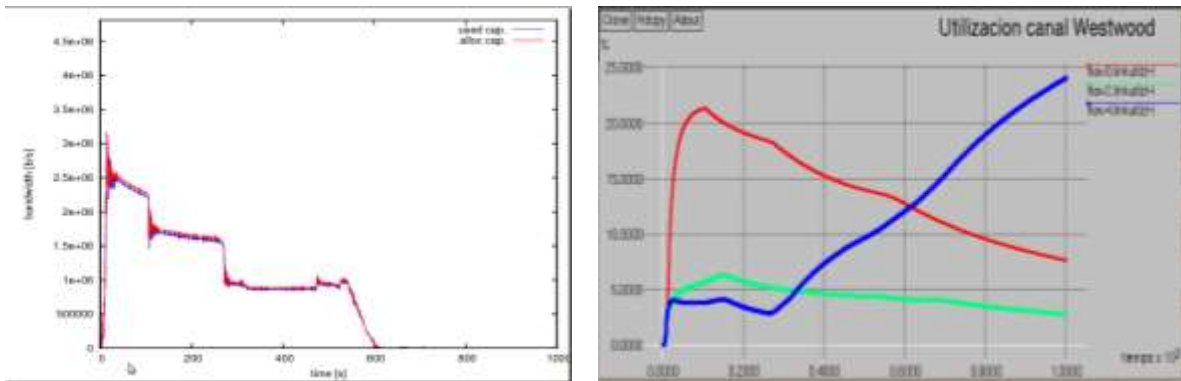
Tiempo (s)	<i>CWND</i> (paquetes)	<i>Ssthresh</i>	PKTX	ACKs
(0 – 15)	0 – 720	800	101	718
(15 – 17,9)	720 - 278	360	101	1534
(20 - 104)	360 - 406	360	103	18813
(105 - 270)	203 - 294	203	103	41915
(270 - 655)	147 - 260	147	103	65272
(655 - 1000)	1 – 10	103 -2	126	65281

En la figura 3-41a y el archivo (*curral.tr*) se observa que durante el intervalo de tiempo (1 a 107seg) las asignaciones de los slots oscilan entre 8 y 15, correspondiente a la RCST(0) lo que indica que el resto de slots están distribuidos entre las demás estaciones terrenas, en el intervalo (107 a 400) se hace una distribución de los slots un poco más equitativa pero la

RCST(0) sigue teniendo un mayor número de slots asignados, en el intervalo (650 a 1000) las solicitudes y asignaciones de la RCST(0) son cero a excepción de dos ocasiones, lo que muestra que los recursos están siendo monopolizados por otra estación terrena.



Slots asignados una conexión  
Slots solicitados una conexión  
Figura 3-41a. Slots solicitados y asignados con la política RBDC.



Ancho de banda utilizado y asignado  
Utilización del canal por varias conexiones  
flujo0 flujo2 flujo4

Figura 3-41b. Ancho de banda *uplink* y utilización del canal empleado con RVBDC.

Según los datos de la tabla 3-19 y de acuerdo a la manera como se solicitan y se asignan los recursos, se puede decir que no hay una asignación equitativa (ver figura 3-41b, utilización de canal), por lo que esto afecta notablemente el desempeño de TCPW, ya que durante los intervalos de tiempo donde se tiene una buena asignación de recursos el crecimiento de *CWND* es linealmente suave, tiene un RTT promedio de 1.5seg a 2seg, una buena recepción de ACKs en el transmisor, mientras que para  $t > 600s$  *CWND* empieza a decrecer hasta llegar a un valor de uno, el RTT se incrementa de manera escalonada y rápida, hay presencia de *backoff* y retransmisiones del *timeout*, los ACKs permanecen constantes, por lo que no hay transmisión de paquetes (ver figuras 3-42 y 3-43 ).





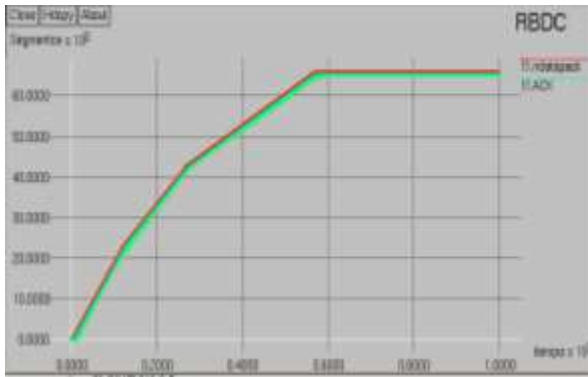
**CWND** **ssthresh**



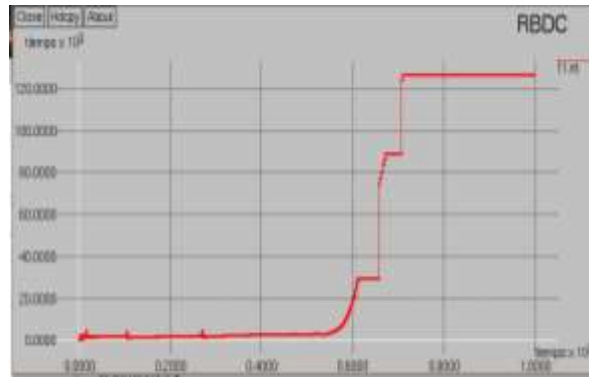
**Paquetes retransmitidos**

Figura 3-42. Comportamiento de los parámetros de TCPW.

Con lo mencionado, se concluye que esta política de petición de recursos afecta notablemente el desempeño de TCPW, por la manera como se asignan los recursos.



Paquetes **transmitidos** y **ACKs**



**RTT**

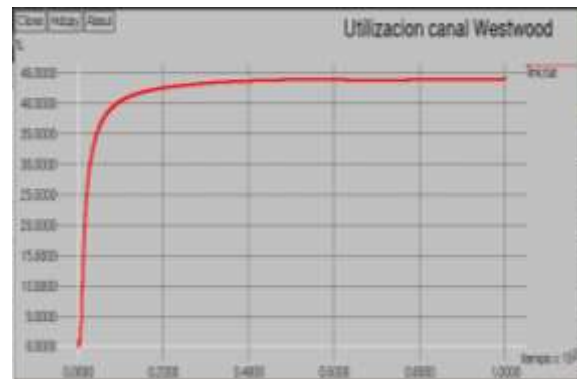


**Backoff** **Número de retransmisiones por timeout**

Figura 3-43. Comportamiento de los parámetros de TCPW.



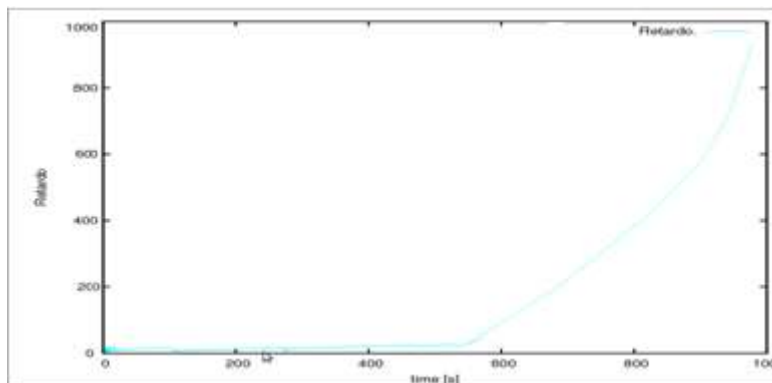
**Throughput** de la red



**Utilización de canal**

Figura 3-44. Desempeño del *throughput* y utilización del canal.

Si se observa la figura 3-45, el tiempo de encolamiento para esta política es bastante grande después de un periodo de tiempo, lo que se ve reflejado en el desempeño de TCPW. La utilización del canal y el *throughput* obtienen un buen rendimiento, pero esto es debido a la forma indiscriminada que realiza la asignación de recursos, lo que implica que solo unas pocas estaciones terrenas utilizan todo el canal de la red.



**Retardo de la red**

Figura 3-45. Comportamiento del retardo de la red.

### c) Política CRA.

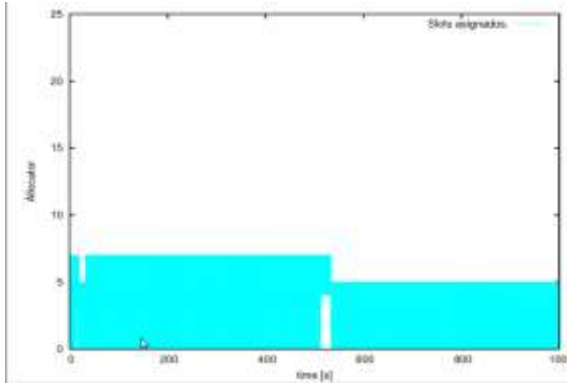
Tabla 3-20. Comportamiento de *CWND*, *ssthresh*, *PkRTX* y *ACKs*.

Tiempo (s)	<i>CWND</i> (paquetes)	<i>Ssthresh</i>	<i>PkRTX</i>	<i>ACKs</i>
(0 – 13,7)	2 - 576	800	0	573
(13,8 – 22,5)	576 - 261	288	36	651
(23 - 707)	262 - 409	288	36	44056
(707 - 1000)	205 - 262	205	37	57686

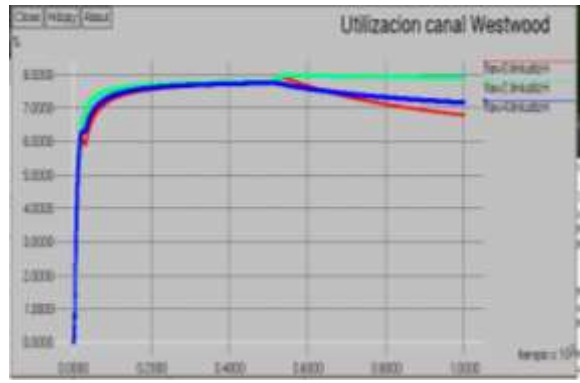
En las figuras 3-46 y el archivo *curral.tr* se observa que la asignación de los slots permanece constante y ecuánime para todas las conexiones, al igual que las solicitudes hechas por las diferentes estaciones terrenas.

En esta política de asignación de recursos los parámetros de TCPW se comportan de una manera uniforme en todas las conexiones, ya que se tiene un *CWND* que está creciendo

constantemente en intervalos de tiempo más grandes, se presentan pocas retransmisiones de paquetes, un buen comportamiento del *ssthresh*, el parámetro que más se afecto es el RTT con un promedio de 5.9s tal y como se muestra en las figuras 3-47.



Slots asignados



Utilización de canal por varias conexiones

**flujo0** **flujo2** **flujo4**

Figura 3-46. Asignación y utilización de recursos de la red.



**Ventana de congestión** y **ssthresh**



**Paquetes Retransmitidos**



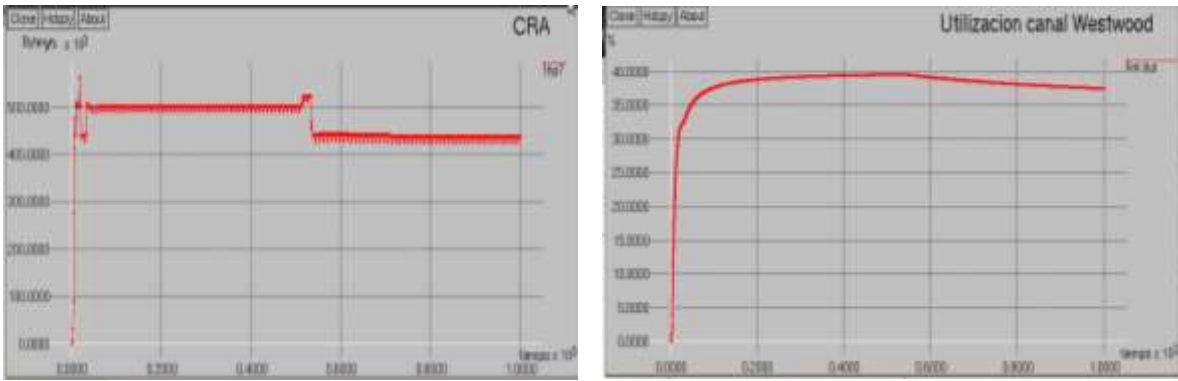
**ACKs** y **paquetes transmitidos**



**RTT**

Figura 3-47. Comportamiento de los parámetros TCPW-CRA.

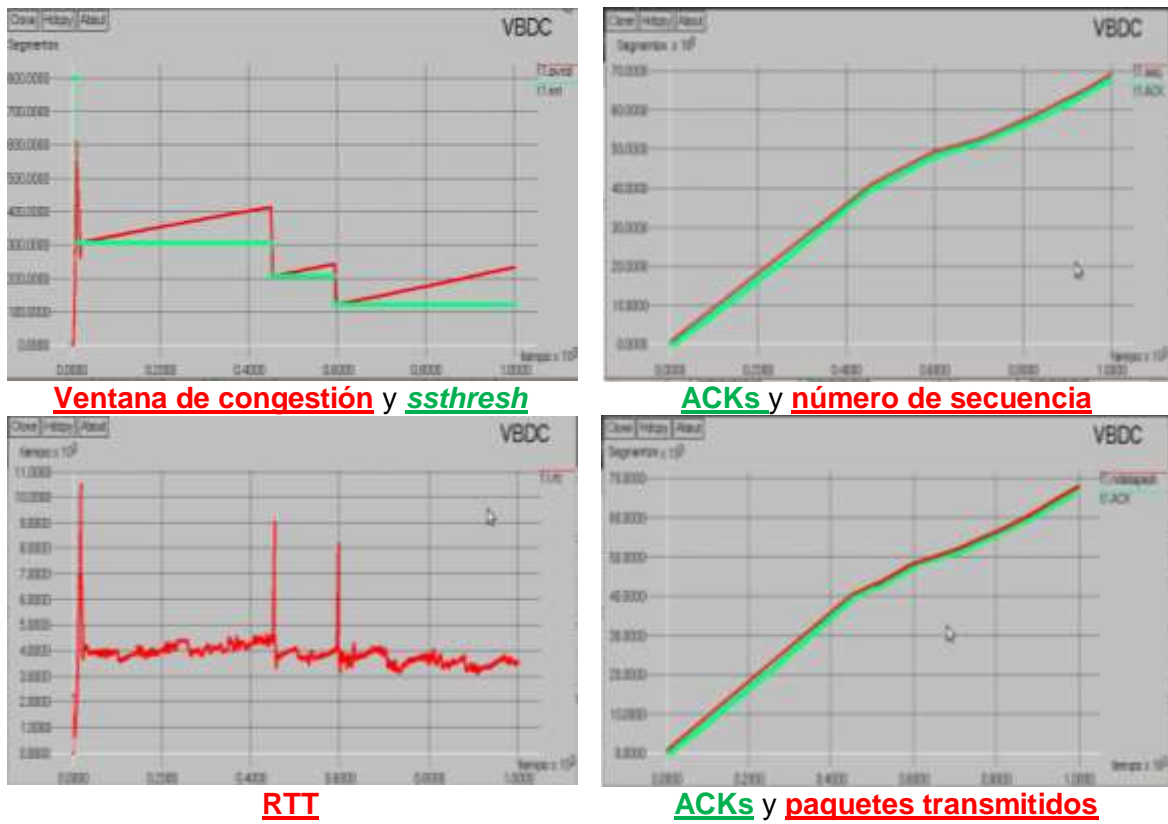
Con esta política de asignación de recursos se nota que hay un desperdicio de los recursos, ya que en ocasiones la NCC no asigna todos los slots disponibles a las estaciones terrenas, solo asigna los slots solicitados lo que influye en el *throughput* de la red y la utilización del canal (ver figura 3-48).



Throughput Utilización del canal  
 Figura 3-48. Throughput y utilización de la red.

**d) Política VBDC.**

Como se observa en la figura 3-49 y en la tabla 3-21 se tiene un buen comportamiento de los parámetros de TCPW, *CWND* crece de manera lineal, hay pocas retransmisiones de paquetes lo que se ve reflejado en más ACKs recibidos por el transmisor, un mayor número de paquetes transmitidos, y por ende un buen comportamiento del RTT.



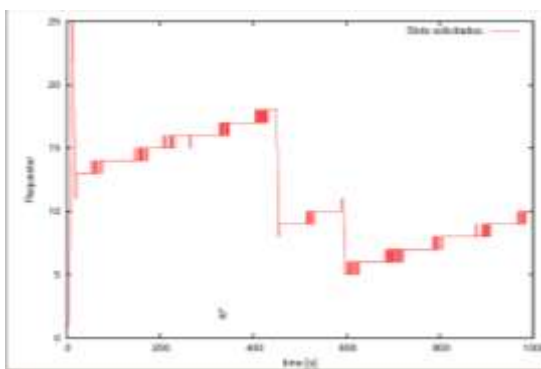
**RTT** **ACKs y número de secuencia**  
**ACKs y paquetes transmitidos**  
 Figura 3-49. Comportamiento de los parámetros TCPW con VBDC.

En las figuras 3-50 se puede apreciar que las estaciones terrenas al inicio solicitan bastantes recursos y en ocasiones superan los disponibles por la red, mientras que los slots asignados por la NCC se mantienen de manera equitativa entre todas las estaciones intentando satisfacer

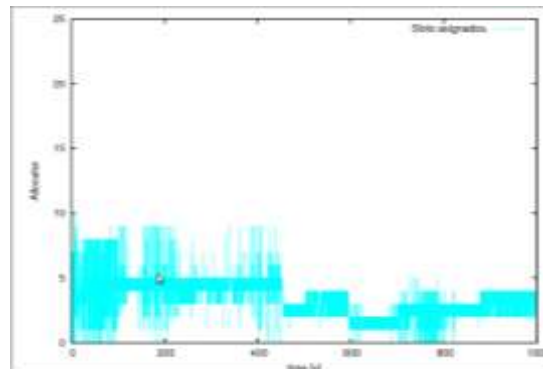
las solicitudes hechas por las estaciones terrenas en base al encolamiento que se tiene en el búfer (ver figura 3-50 encolamiento y utilización de canal), que se ve reflejado en un menor retardo de la red afectando de una manera menos significativa el desempeño de TCP.

Tabla 3-21. Comportamiento de *CWND*, *ssthresh*, *PkRTX* y *ACKs*.

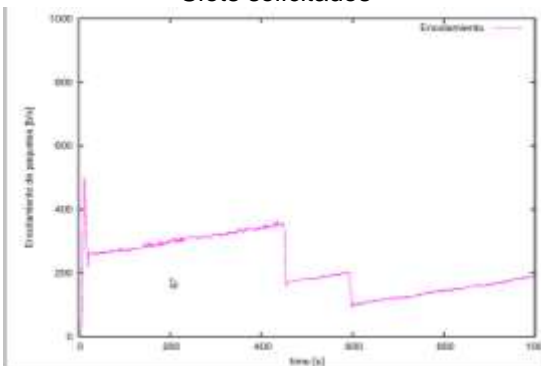
Tiempo (s)	<i>CWND</i> (paquetes)	<i>Ssthresh</i>	<i>PkRTX</i>	<i>ACKs</i>
(0 – 12,36)	2 - 615	800	0	607
(12,36 - 20)	615 - 263	307	62	722
(20 – 449,8)	263 - 413	307	63	39238
(454 593)	206 - 242	206	64	47750
(595 - 1000)	121 - 233	121	64	67797



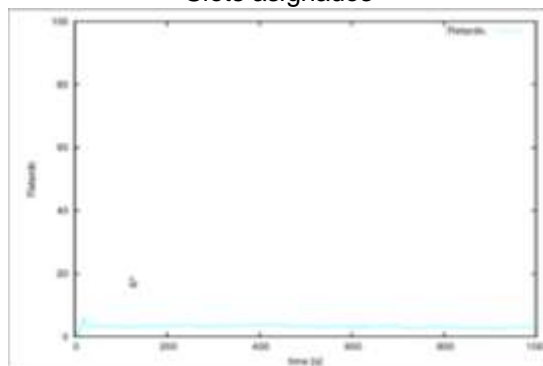
Slots solicitados



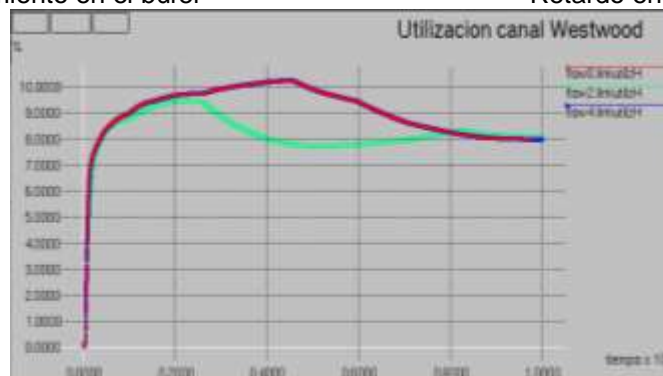
Slots asignados



Encolamiento en el búfer



Retardo en la red



Utilización del canal por varias conexiones, [flujo0](#) [flujo2](#) [flujo4](#)

Figura 3-50. Asignación y utilización de recursos de la red.

En cuanto a la capacidad del canal utilizado se observa que se tiene un porcentaje del 46% y un *throughput* promedio de 570000 Bytes/s, indicando un buen desempeño de la red (ver figuras 3-51) esto gracias a la manera de como se asignan los recursos de la red y se ve reflejado en el buen comportamiento de los parámetros de TCPW. Se concluye entonces que esta política de solicitud de recursos impacta de manera muy positiva en el desempeño de TCPW y en el rendimiento de la red.

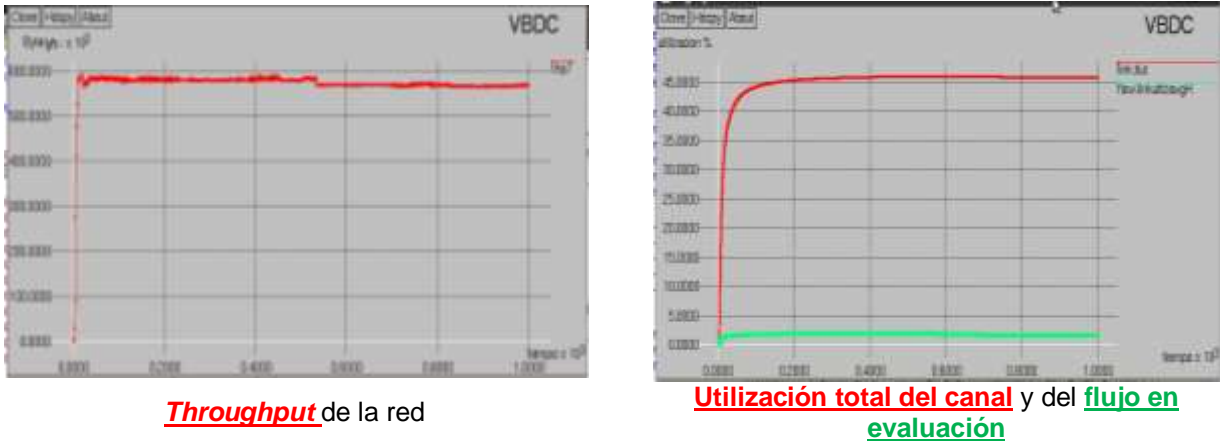


Figura 3-51. Rendimiento de la red.

**e) Comparación de las políticas de asignación de recursos.**

En las tablas 3-22, 3-23, 3-24 y 3-25 se observa el comportamiento de la red satelital y el desempeño que tiene TCPW para tres conexiones diferentes, el cual permitirá concluir cual de las cuatro políticas presenta el mejor impacto en el desempeño del protocolo TCPW y en el rendimiento de la red.

Tabla 3-22. Parámetros de desempeño y de TCPW en las pruebas L, M, N y O para la conexión uno.

	CWND	RTT	pkTx	Link	dequeue	Retardo	ndatapack	ACKs	nrexmit
<b>O</b>	320	6	37	8%	285	4,8	58221	57691	0
<b>M</b>	310	4,2	64	9%	280	3,5	68312	67804	0
<b>L</b>	320	4,8	56	11%	300	4,5	84560	83592	0
<b>N</b>	120	50	126	18%	200	128	65904	65281	5

Tabla 3-23. Parámetros de desempeño y de TCPW en las pruebas L, M, N y O para la conexión dos.

	CWND	RTT	PKTX	Link	dequeue	Retardo	ndatapack	ACKs	nrexmit
<b>O</b>	360	5,0	51	8%	365	4,8	68101	67279	0
<b>M</b>	350	4,3	49	8%	280	3,8	69250	68594	0
<b>L</b>	400	5,0	53	9%	320	4,4	80951	80040	0
<b>N</b>	110	13,8	59	5%	170	12	24548	23970	3

Tabla 3-24. Parámetros de desempeño y de TCP en las pruebas L, M, N y O para la conexión tres.

	CWND	RTT	PKTX	Link	dequeue	Retardo	ndatapack	ACKs	nrexmit
<b>O</b>	240	4,8	52	8%	200	4,8	61421	60863	0
<b>M</b>	310	4,2	49	9%	310	3,7	80687	79339	0
<b>L</b>	180	3,1	52	8%	150	2,9	67451	66528	0
<b>N</b>	220	2,5	65	22%	100	1,5	204980	204564	1

De la tabla 3-25 se observa que la política con mejor rendimiento y utilización de canal que se presentó en la red satelital fue VBDC, debido a la manera equitativa como se hace la asignación de recursos, en segundo lugar esta RVBDC ya que ésta también intenta hacer una asignación equitativa de los recursos de la red, en tercer y cuarto lugar están RBDC y CRA respectivamente. La política RBDC presenta una utilización de canal muy buena sin embargo es afectada por los altos tiempos de retardo de propagación, con respecto a la política CRA para ciertos intervalos de tiempo hay desperdicio de los recursos de la red.

Tabla 3-25. Rendimiento de la red.

Políticas	Throughput		Utilización Canal (%)
	Max	Prom	
CRA	525	475	37 – 39
VBDC	587	570	45 – 46
RVBDC	585	560	45
RBDC	560	550	43 - 44

Como se observa en los datos de las tablas 3-22, 3-23, 3-24 y 3-25 se llegan a las siguientes afirmaciones de cómo impacta las diferentes políticas de asignación de recursos DAMA sobre el desempeño de TCPW, estas son:

- Las políticas VBDC y RVBDC tienen un impacto positivo sobre el RTT de la red, a pesar de que estas introducen un retardo de acceso a la red el cual puede aumentar el RTT estimado por el transmisor, este se ve compensado con el rápido crecimiento de los ACKs que llegan al transmisor, el cual permite calcular el RTT de manera más continua, lo que se ve reflejado en más paquetes transmitidos, un buen uso del canal y un buen desempeño de la red satelital.
- El impacto que tienen las políticas CRA y RBDC sobre el RTT es mayor en comparación con VBDC, ya que estas presentan más ACKs duplicados o expiraciones del *timeout* para el caso de RBDC, lo que afecta en el número de ACKs que llegan al receptor, influyendo en el cálculo del RTT en TCPW, también se presenta más paquetes retransmitidos afectando el rendimiento de la red
- El comportamiento de *CWND* presenta un mejor desempeño cuando emplea las políticas VBDC y RVBDC, gracias a la manera casi equitativa de asignar recursos a las estaciones terrenas, esto se ve reflejado en un crecimiento de *CWND* lineal con pendientes levemente pronunciadas durante las etapas de evasión de congestión para

todas las estaciones terrenas, lo que conlleva a una mejor utilización de canal y a un mejor *throughput* de la red.

- Con la política RBDC el impacto que causa sobre *CWND* es bien notorio y negativo dado que durante algunos intervalos de tiempo *CWND* crece linealmente con pendientes bien pronunciadas, pero en otros intervalos *CWND* decrece de manera abrupta, lo que conlleva a que se active el *timeout*, un incremento en la retransmisión de paquetes y en menos ACKs que llegan al receptor, el comportamiento anterior es consecuencia a la manera inequitativa de cómo se asignan los recursos entre las estaciones terrenas.
- El impacto que tiene VBDC y RVBDC sobre el comportamiento de los ACKs que llegan al transmisor es positivo, ya que se tiene un crecimiento lineal con pendientes bastante pronunciadas como lo muestran los resultados obtenidos; en cuanto a las políticas CRA y RBDC el impacto que tienen estas sobre los ACKs es regular debido a que crecen de manera rápida pero se presentan más intervalos donde los ACKs permanecen constantes, lo que determina más congestión en la red.

En conclusión se tiene que la VBDC tiene un impacto muy positivo sobre el desempeño de TCPW, en segundo lugar RVBDC, seguido por CRA y por último se encuentra RBDC.

### 3.2.7. Impacto del Protocolo Aloha en los Parámetros TCP y del Desempeño en la Red.

En este caso de estudio se analizó el impacto al utilizar el protocolo Aloha en el desempeño de las versiones de TCPW, TCP Hybla y TCP HS como también sus parámetros de rendimiento en la red.

En la tabla 3-26 se muestra los parámetros de entrada al simulador para el análisis de este escenario.

Tabla 3-26. Parámetros de simulación.

Variable	Valor	
Tiempo de simulación	300 seg	
Tamaño paquete	TCP HighSpeed	1400
	TCP Westwood	1400
	TCP Hybla	1440
N° máximo de retransmisiones	3	
Tiempo aleatorio max RTx	500ms	
Búfer Satélite	230	
Tipo de encolamiento	Drop Tail	
Tráfico	FTP	

Antes de analizar el comportamiento de los parámetros de TCP, se aclara que dentro de una red satelital quien detecta la colisión es el satélite, y este es el que emite una señal *broadcast* hacia las estaciones terrenas. La herramienta de simulación está configurada de tal manera que cuando se detecta una colisión en la red, el encargado de retransmitir el paquete es la estación terrena, más no el nodo transmisor [71, 72].



### 3.2.7.1. Análisis de la ventana de congestión.

En la figura 3-52 se muestra el comportamiento de *CWND* para las tres versiones TCP, se observa que el crecimiento y comportamiento de *CWND* en TCPW y TCPHS son muy similares, la razón de este comportamiento es porque ambas versiones son una derivación de TCP New Reno [11, 29], mientras que la ventana de congestión en TCP Hybla crece de manera más rápida.

El impacto que se genera al utilizar el protocolo Aloha en una red satelital es un crecimiento de *CWND* muy suave para TCPW y TCP HS, mientras que en TCP Hybla *CWND* crece un poco más rápido, pero no alcanza los valores obtenidos para el caso de TDMA-DAMA y es más visible para la etapa de inicio lento; esto se debe a la gran cantidad de colisiones que presenta el protocolo Aloha donde TCP los interpreta como congestión de la red impidiendo el crecimiento de *CWND* de manera más agresiva.

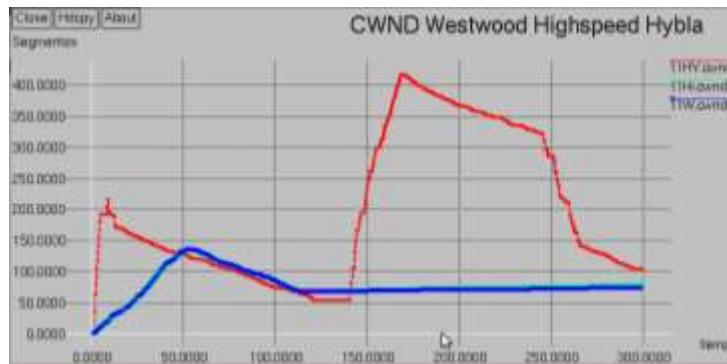
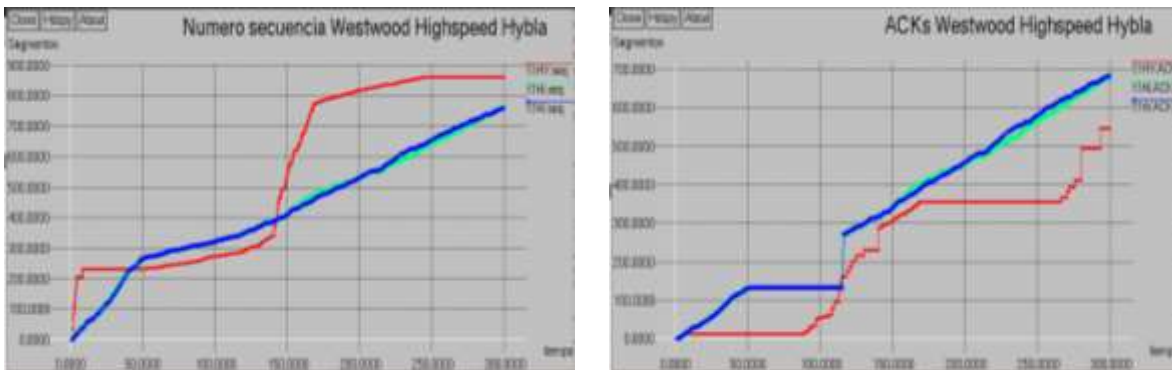


Figura 3-52. Comportamiento de *CWND* en las tres versiones de TCP.

### 3.2.7.2. Análisis de ACKs, número de secuencia y paquetes transmitidos.

En la figura 3-53 se muestra el comportamiento de los ACKs para las tres versiones de estudio, donde se observa que el crecimiento de los ACKs es mayor en TCPW y TCP HighSpeed que en TCP Hybla, mientras que el número de secuencia alcanza valores mayores en TCP Hybla en comparación con las demás versiones, la razón de este comportamiento es debido a la rápida recuperación y el crecimiento agresivo que tiene TCP Hybla en su *CWND* en comparación con TCPW y TCP HS; por otro lado se tiene que el número de paquetes transmitidos es mayor en TCP Hybla que en las demás versiones.

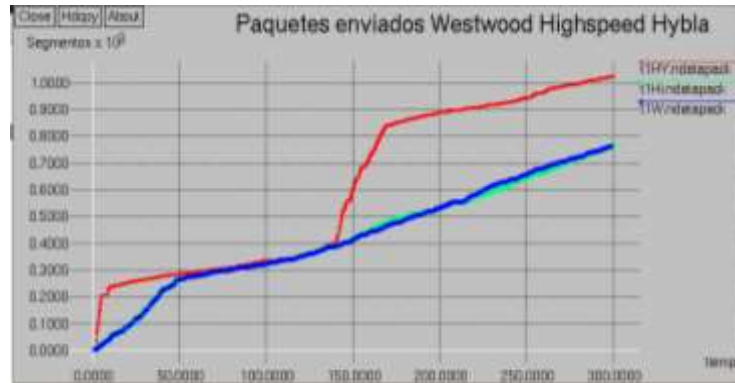


Comportamiento del número de secuencia

Comportamiento de los ACKs

Figura 3-53. Comportamiento de parámetros TCP Hybla, TCPW, TCP HS.

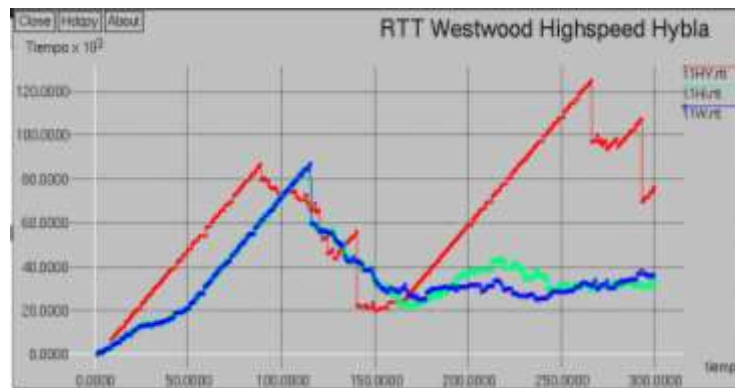
El impacto que tiene el protocolo Aloha sobre cada una de las versiones TCP analizadas es un crecimiento bastante pobre de los ACKs que arriban al transmisor, lo que se ve reflejado en un bajo desempeño de *CWND*, un crecimiento muy suave de los paquetes transmitidos y número de secuencia, esto debido a los grandes retardos que presenta el protocolo Aloha producto al gran número de colisiones que se presentan en la red impidiendo un buen desempeño de estos parámetros TCP.



Comportamiento de los paquetes transmitidos  
 Figura 3-54. Paquetes transmitidos en TCP Hybla, TCPW, TCP HS.

### 3.2.7.3. Análisis del RTT.

En la figura 3-55 se observa el impacto que tiene el protocolo Aloha sobre el RTT para cada una de las versiones de TCP, donde se aprecia un crecimiento rápido durante los primeros 100 segundos para las tres versiones de TCP, alcanzando valores muy altos, el cual influye de manera muy notoria en el desempeño de TCP, durante el resto de la simulación el RTT para el caso de TCPW y TCP HS disminuye un poco y tiende a mantenerse en ese valor, pero se siguen presentando valores altos, mientras que para TCP Hybla el RTT se decremента por un periodo corto y después vuelve a incrementarse de manera muy pronunciada; este comportamiento tan regular del RTT, en gran medida es atribuido a la cantidad de colisiones que presenta el protocolo Aloha en el canal satelital.



Comportamiento del RTT  
 Figura 3-55. Comportamiento del RTT.

### 3.2.7.4. Análisis del rendimiento y retardo de la red.

Para realizar el análisis del rendimiento de la red y el retardo para las diferentes versiones de TCP se tomó como referencia el análisis y las ecuaciones descritas en [73]; con los datos arrojados por la simulación, primero se procedió a calcular el ancho de banda utilizado en intervalos de 50 segundos por las 5 estaciones terrenas transmisoras.

$$BW = \frac{(\text{paquetes enviados} + \text{paquetes colisionados}) * \text{tamaño segmento} * 8}{\text{tiempo de simulación}} \quad (20)$$

$$CH = \frac{\text{bit recibidos}}{BW \text{ total}} * 100 \text{ [\%]} \quad (21)$$

Los resultados obtenidos se muestran en las tablas 3-27, 3-28 y 3-29 para las versiones de TCP evaluadas.

Tabla 3-27. Resultados para TCP Westwood.

Tiempo	Paquetes enviados	N° de colisiones	BW (b/s)	Bytes recibidos	Utilización del canal (CH) %
50	956	107	238112	845550	6,6
100	1701	245	217952	1485000	5,8
150	2508	362	214293	2465200	6,42
200	3286	486	211232	3259800	6,36
250	3817	524	194477	3761500	5,877
300	4345	733	189578	4370500	5,69
<b>Promedio</b>			<b>210940</b>		<b>6,12</b>

Tabla 3-28. Resultados para TCP HS.

Tiempo	Paquetes enviados	N° de colisiones	BW (b/s)	Bytes recibidos	Utilización del canal (CH) %
50	956	107	238112	845550	6,6
100	1702	245	218064	1485000	5,8
150	2510	360	214293	2465200	6,42
200	3281	489	211120	3216300	6,28
250	3882	608	201152	3761500	5,87
300	4370	733	190512	4367600	5,68
<b>Promedio</b>			<b>212208</b>		<b>6.10</b>

Tabla 3-29. Resultados para TCP Hybla.

Tiempo	Paquetes enviados	N° de colisiones	BW (b/s)	Bytes recibidos	Utilización del canal (CH) %
50	4574	131	1084032	307140	2,39
100	5411	251	652262	421870	1,64
150	6651	374	539520	1204120	3,13
200	7639	501	468864	1336730	2,61
250	8384	620	414902	1385900	2,16
300	9076	744	377088	2120470	2,76
<b>Promedio</b>			<b>589444</b>		<b>2.44</b>

En las tablas 3-27, 3-28 se aprecia que TCPW y TCP HS tiene resultados muy parecidos, en comparación con los resultados de TCP Hybla, las tablas también muestran la cantidad total de bit/s recibidos exitosamente en el receptor en los diferentes intervalos de tiempo, donde TCP Hybla presenta un promedio del 2.44% de la utilización de la red, para el caso TCPW y TCP HS presentan una mayor cantidad de bit/s recibidos exitosamente, el cual se ve reflejado en una mejor utilización de canal cuyos valores se encuentran en promedio de 6.12% y 6.10% respectivamente.

La figura 3-56 muestra la evolución del ancho de banda utilizado por cada una de las versiones TCP, donde se aprecia que para TCP Hybla utiliza un mayor ancho de banda durante los primeros segundos esto gracias al crecimiento agresivo de *CWND* durante la fase de inicio lento, después empieza a decrecer rápidamente, mientras que para las versiones TCPW y TCP HS alcanza un valor de 200000b/s y procura mantenerse en este valor, como se aprecia en la figura 3-56, el ancho de banda utilizado por las diferentes versiones de TCP no es tan bueno, esto debido al pobre desempeño que tiene el protocolo Aloha en una red de comunicaciones.

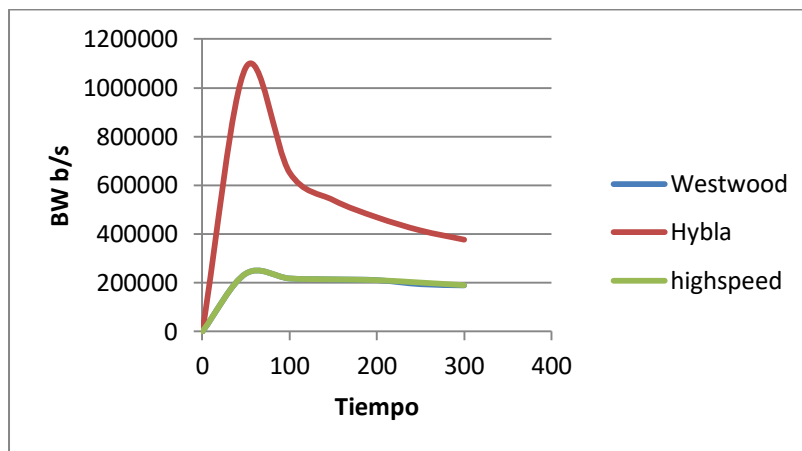
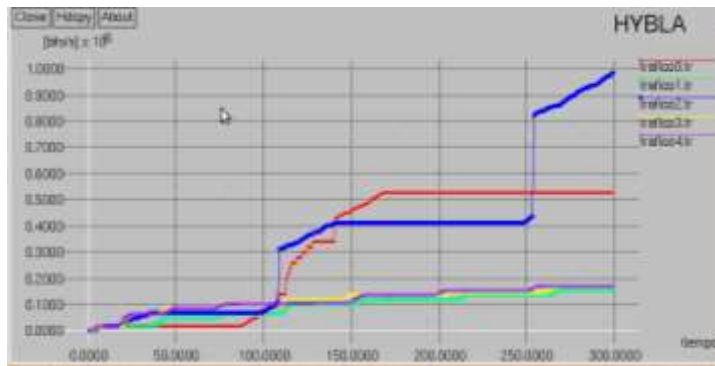


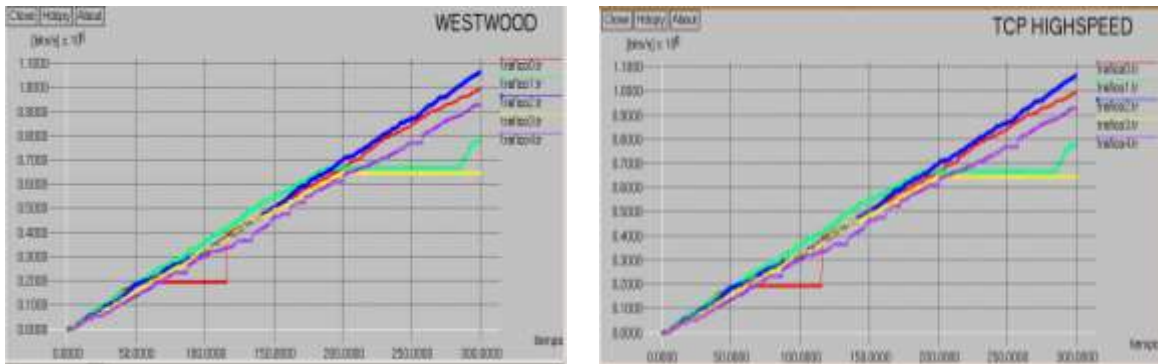
Figura 3-56. Evolución BW en el tiempo.

Las tablas 3-27, 3-28 y 3-29 también indican que a pesar de que TCP Hybla utiliza un ancho de banda más grande, la utilización del canal es bastante pobre si se compara con las otras versiones, este comportamiento es debido a que los bits recibidos en TCP Hybla son pocos en comparación con los que se han transmitido, lo que está indicando que los paquetes perdidos no todos son productos de las colisiones en la red, sino que también son generados por otros factores como congestión de la red, aumento de retardo de propagación, desbordamiento de los búferes etc. Comportamiento que no ocurre en TCPW y TCP HS, donde la gran mayoría de las pérdidas de datos es atribuida a las colisiones presentes en la red; esto se puede corroborar al comparar los datos de las tablas 3-27, 3-28 y 3-29.

En las figuras 3-56 y 3-57 se observa que TCP Hybla utiliza un mayor ancho de banda en comparación con TCPW y TCP HS, sin embargo presenta un tráfico menor para todas las conexiones y en consecuencia una mala utilización del canal, mientras que para las otras versiones se tiene un mejor comportamiento.



TCP Hybla [traf0](#) [traf1](#) [traf2](#) [traf3](#) [traf4](#)  
 Figura 3-57. Bits por segundos recibidos en los nodos receptores.



TCP Westwood [traf0](#) [traf1](#) [traf2](#) [traf3](#) [traf4](#)      TCP Highspeed [traf0](#) [traf1](#) [traf2](#) [traf3](#) [traf4](#)  
 Figura 3-58. Evolución del *throughput* de TCP.

Dentro de los resultados obtenidos de la simulación, también se puede encontrar cuál es la probabilidad de que un paquete colisione en la red, para calcular esto se considera los paquetes recibidos, los paquetes colisionados y la duración del paquete a transmitir; para esto se tiene en cuenta las ecuaciones 22 y 23 utilizadas en [73]:

$$P_t = e^{-2\alpha\tau} \quad (22)$$

Donde  $\alpha = \frac{\text{paquetes recibidos} + \text{paquetes colisionados}}{\text{tiempo de simulación}}$ ,  $P_t$  indica la probabilidad de que no haya una colisión entre dos paquetes; también se calculó el número promedio de retransmisiones de un paquete, que vienen dado por la ecuación:

$$N = e^{2\alpha\tau} \quad (23)$$

Las tablas 3-30, 3-31 y 3-32 muestran los resultados obtenidos después de ser aplicadas las ecuaciones 22 y 23.

Tabla 3-30. Resultados para TCPW.

Tiempo	Paquetes Recibidos	$\alpha$	Pt (%)	N
50	604	14,22	85,59	1,17
100	1060	13,05	86,70	1,15
150	1761	14,15	85,66	1,17
200	2328	14,07	85,73	1,17
250	2687	12,84	86,89	1,15
300	3122	12,85	86,88	1,15

Tabla 3-31. Resultados para TCP HS.

Tiempo	Paquetes Recibidos	$\alpha$	Pt (%)	N
50	604	14,22	85,59	1,17
100	1061	13,06	86,69	1,15
150	1760	14,13	85,68	1,17
200	2297	13,93	85,86	1,17
250	2687	13,18	86,65	1,15
300	3120	12,84	86,89	1,15

Tabla 3-32. Resultados para TCP Hybla.

Tiempo	Paquetes Recibidos	$\alpha$	Pt (%)	N
50	213	6,88	92,55	1,08
100	293	5,44	94,06	1,06
150	836	8,06	91,33	1,09
200	928	7,14	92,27	1,08
250	963	6,33	93,12	1,07
300	1472	7,39	92,02	1,08

Antes de analizar los datos obtenidos de las tablas 3-30, 3-31 y 3-32 se manipulo la ecuación 22 para poder interpretar bien los resultados obtenidos.

La ecuación 22 indica la probabilidad de que no haya una colisión entre dos paquetes, cuya función de probabilidad está dada por una distribución de *Poisson*, donde la transmisión de paquetes son eventos independientes, la variable aleatoria está dada por el número de paquetes exitosos que llegan al receptor, y la variable  $\alpha$  indica la ocurrencia promedio por unidad de tiempo, luego la ecuación 22 se puede escribir de la siguiente manera:

$$\alpha = \frac{\text{paquetes recibidos}}{\text{tiempo de simulación}} + \frac{\# \text{ colisiones}}{\text{tiempo simulacion}} = \beta + \theta \quad (24)$$

Luego

$$P_t = e^{-2\tau\beta} e^{-2\tau\theta} = P_s P_c \quad (25)$$

Donde  $P_s$  indica la probabilidad de que un paquete llegue al destino sin colisión y  $P_c$  indica la probabilidad de que un paquete no colisione, luego  $P_t$  está dada por la probabilidad conjunta entre  $P_s$  y  $P_c$ :

$$P_t = P(P_s \wedge P_c) = P_s P_c \quad (26)$$

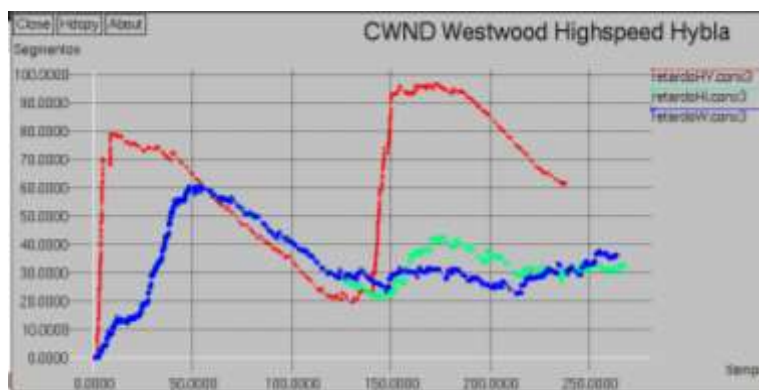
En la ecuación 26 se muestra que la transmisión de paquetes son eventos independientes lo cual es una de las condiciones para poder usar una distribución de *Poisson*, también se debe tener en cuenta las siguientes tres consideraciones:

- Se debe trabajar en el mismo intervalo de tiempo.
- El mismo espacio muestral.
- El mismo protocolo de comunicaciones.

Teniendo en cuenta lo descrito anteriormente y el análisis de la ecuación 22 se verifica que los datos de las tablas 3-30, 3-31 y 3-32 son correctos; por lo tanto no se pudo llegar a concluir que teniendo en cuenta el número de colisiones ocurridas en un intervalo de tiempo el rendimiento de la red sea mejor para una versión de TCP que para otra, dado que para cada prueba realizada se está utilizando un protocolo de comunicación diferente, donde se utiliza diferentes algoritmos para implementar el protocolo de transporte como lo son las versiones TCPW, TCP Hybla y TCP HS.

A manera de conclusión y de acuerdo a los datos de las tablas 3-30, 3-31 y 3-32 se puede decir que el tener una baja probabilidad de colisiones en una red permite tener un mejor desempeño del protocolo TCP, también se tiene que el promedio de retransmisiones de un paquete esta alrededor de uno para las tres versiones TCP analizadas, se deduce también que en los intervalos donde *CWND* crece de manera muy rápida, la probabilidad de que dos paquetes colisionen es mayor.

La figura 3-59 muestra el retardo de la red, donde TCP Hybla tiene un retardo mucho mayor en comparación con TCPW y TCP HS a excepción del intervalo (55 - 142) donde el valor del retardo de TCP Hybla es inferior a las otras versiones TCP, esto a consecuencia que durante ese intervalo la ventana de congestión de TCP Hybla decrece mucho más rápido que TCPW y TCP HS por lo tanto se están transmitiendo menos paquetes, lo que influye notablemente en el retardo, y por ende en el RTT.



Retardo de extremo a extremo  
Figura 3-59. Evolución del retardo.

En las tablas 3-30, 3-31 y 3-32 se observa que TCP Hybla transmite mucho más paquetes en comparación con TCPW y TCP HS, pero en el receptor se tiene muy pocos paquetes recibidos, lo que implica una gran pérdida de paquetes en la red, esto debido al gran retardo que presenta TCP Hybla.



### 3.2.8. Impacto al Variar el Tiempo Aleatorio de Retransmisión.

Para analizar cómo influye *mean-backoff* (tiempo aleatorio de retransmisión que tienen que esperar las estaciones terrenas después de una colisión) en el desempeño del protocolo TCP, se toman dos valores, de 0,5s y 1.0s; el análisis descrito anteriormente para Aloha corresponde a un *mean-backoff* de 0,5s, ahora se analiza el comportamiento para un valor de 1.0 segundo con los mismos parámetros que se muestra en la tabla 3-26.

#### 3.2.8.1. Análisis de la ventana de congestión.

En la figura 3-60 se observa el comportamiento de la ventana de congestión para las tres versiones, comparando la figura 3-52 con la figura 3-60 se observa que el impacto que genera el utilizar un *mean-backoff* más grande (1 segundo) en cada una de las versiones de TCP, es un crecimiento de *CWND* mucho más agresivo durante la etapa de inicio lento, pero cuando *CWND* entra en la etapa de evasión de congestión TCP Hybla se mantiene en un valor constante durante largos periodos de tiempo, este comportamiento está indicando que no se están transmitiendo paquetes o recibiendo ACKs en el transmisor, mientras que para TCPW y TCP HS *CWND* sigue creciendo hasta llegar a un máximo, después empieza a decrecer por un periodo largo el cual está indicando que le cuesta recuperarse por congestión de la red.

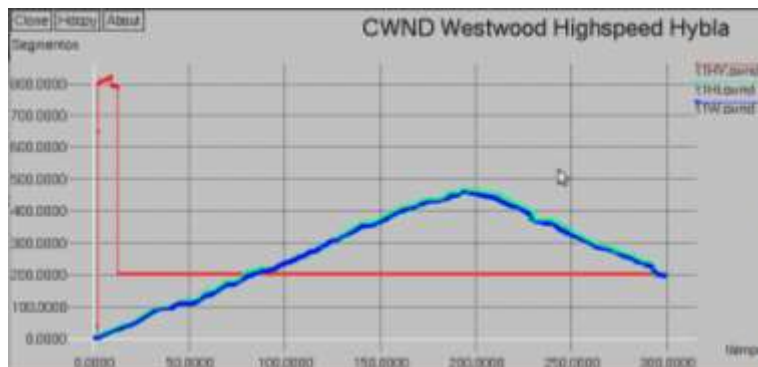
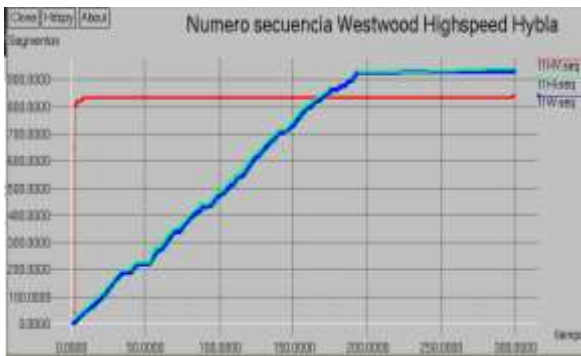


Figura 3-60. Comportamiento de CWND.

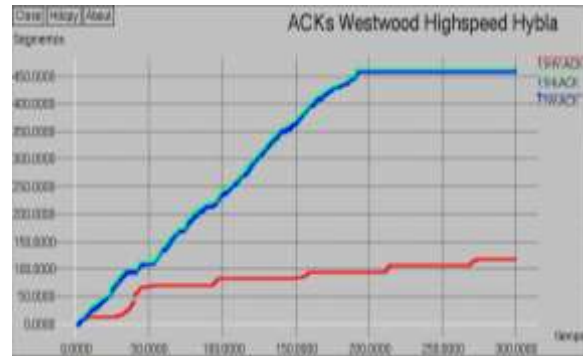
#### 3.2.8.2. Análisis de ACKs, número de secuencia y paquetes transmitidos.

Al comparar las figura 3-61 con las figuras 3-53 y 3-54 se observa el impacto que produce la variación de *mean-backoff* en el comportamiento de los ACKs, estos se ven reducidos de una manera considerable para el caso de TCP Hybla, en comparación con los datos transmitidos y el número de secuencia los cuales presentan crecimientos superiores para el caso de un *mean-backoff* de 1s. Este comportamiento indica que los paquetes se están perdiendo debido a la congestión de la red o por el alto índice de colisiones que se presenta en el medio de propagación, o por el desbordamiento del búfer para el caso de TCP Hybla, el cual hace que los ACKs no crezcan con la misma proporción que los paquetes transmitidos.





Comportamiento del número de secuencia.



Comportamiento de los ACKs.



Comportamiento de los paquetes transmitidos.

Figura 3-61. Comportamiento de parámetros TCP.

### 3.2.8.3. Análisis del RTT.

Comparando las figuras 3-62 y 3-55 se observa el impacto que se tiene al utilizar un *mean-backoff* más grande, el cual se ve reflejado en un incremento muy superior del RTT para las versiones de TCP HS y TCPW, mientras que para TCP Hybla se tiene un comportamiento en diente de sierra pero con un incremento mayor del RTT, por tanto un *mean-backoff* de un segundo implica un aumento significativo del RTT, el cual influye de manera negativa en el rendimiento de la red.

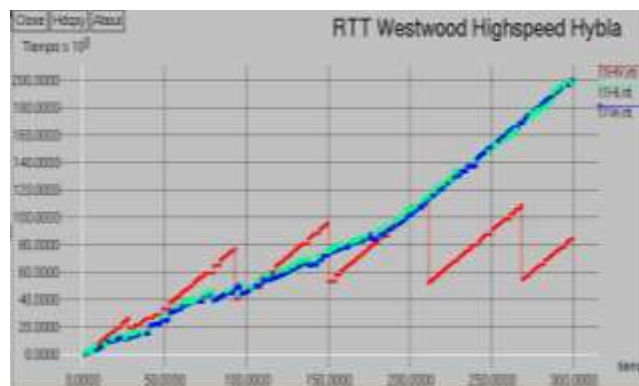


Figura 3-62. Comportamiento del RTT.

### 3.2.8.4. Análisis del rendimiento y retardo de la red.

Tabla 3-33. Probabilidad de colisión para *mean-backoff* de 1s.

TCP	Paquetes Enviados	Colisiones	Bytes recibidos	Paquetes recibidos	BW (b/seg)	Ch (%)	P <sub>t</sub> (%)	N
Wetswood	3595	573	3823850	2732	155605	4,98	88,64	1,12
HighSpeed	3604	573	3823850	2732	155605	4,98	88,64	1,12
Hybla	7978	515	2022130	1405	326131	2.63	93,05	1,07

En la tabla 3-33 se muestran los datos obtenidos para un *mean-backoff* de 1s, al comparar los resultados con las tablas 3-30, 3-31 y 3-32 para los valores finales de la simulación (t=300s) se infiere que la probabilidad de que dos paquetes colisionen se reduce, esto es causado por la reducción de paquetes enviados por el transmisor, lo que se refleja en menos paquetes recibidos, también se verifica con el comportamiento de los parámetros de TCP que se analizaron anteriormente, por lo cual se concluye que para un protocolo Aloha, la probabilidad de colisión disminuye si se tiene un *mean-backoff* grande, pero a cambio de esto se tiene un pobre desempeño del protocolo TCP, que se ve reflejado en el número de bits recibidos por el receptor.

En resumen se puede decir que el protocolo Aloha afecta notablemente el desempeño de los parámetros de TCP en sus diferentes versiones, afectando de manera significativa en la transmisión de paquetes, *CWND*, el RTT y el retardo de transmisión, ya que en este protocolo de acceso al medio no tiene una sincronización y señalización que reduzca la probabilidad de colisiones y por ende la pérdida de paquetes, para el cual TCP lo interpreta como congestión de la red o pérdidas por colisión, que se ve reflejado en la disminución del desempeño del protocolo TCP; por otro lado se tiene que entre más grande se configure el tiempo de *mean-backoff* la probabilidad de colisiones entre paquetes disminuye, pero esto se ve reflejado en un mal comportamiento del protocolo TCP y un mal rendimiento de la red.

## 3.3. MECANISMO DE LA CAPA DE ENLACE DE DATOS MODIFICANDO LA ARQUITECTURA DE LA RED.

### 3.3.1. Impacto del Proxy Snoop en el Desempeño de TCP.

En este caso de estudio se midió el impacto que ocasiona el protocolo snoop en el desempeño del protocolo TCP, se realizaron las pruebas que se indican en la tabla 3-34 donde se evaluó su impacto con y sin proxy.

Tabla 3-34. Impacto del protocolo TCP con y sin el uso del proxy snoop.

Parámetros y mecanismos a modificar	Prueba			
	P	Q	R	
Políticas de solicitud de recursos bajo demanda	<b>VBDC</b>	<b>VBDC</b>	<b>VBDC</b>	<b>FIJOS</b>
Tamaño Slot	1504	1504	1504	
Política de asignación de recursos en la NCC	<i>Proportional</i>	<i>Proportional</i>	<i>Proportional</i>	
Duración trama	0,05	0,05	0,06	<b>VARIABLES</b>
Nº Slots por trama	14	20	24	
Ancho de Banda	3.368.960	4.812.800	4.812.800	
Supertrama	20	20	17	

En la primera prueba se evalúa el desempeño de la red y el comportamiento de los parámetros TCP Hybla, TCPW y TCP HS con y sin el uso del proxy snoop. Los parámetros de rendimiento que se tienen en cuenta son el RTT promedio, el *throughput* total de la red, el retardo de la red y la utilización del canal para esa conexión, y respecto a los parámetros TCP se evalúan los paquetes transmitidos y retransmitidos, el número de secuencia y el ACK. Las tablas 3-35 a 3-37 indican los resultados para las pruebas P, Q y R al utilizar el proxy Snoop.

Tabla 3-35. Parámetros de evaluación en la prueba P con Snoop.

	ndatapack	PkRTX	RTT	Sec	ACK	Throughput	Retardo	Utilización
TCP Hybla	95845	11253	7120,97	84610	83783	214401,37	3,88	9,92
TCP HS	56130	40	7629,51	56090	55722	402737,77	6,84	6,55
TCPW	64346	38	5480,92	64308	64071	400143,22	4,84	7,53

Tabla 3-36. Parámetros de evaluación en la prueba Q con Snoop.

	ndatapack	PkRTX	RTT	Sec	ACK	Throughput	Retardo	Utilización
TCPHybla	182420	40997	3006,98	141423	140572	308901,79	1,71	16,95
TCP HS	81471	68	5333,01	81403	80947	568981,54	4,66	9,52
TCPW	82239	62	3884,41	82177	81841	527132,11	3,26	9,62

Tabla 3-37. Parámetros de evaluación en la prueba R con Snoop.

	ndatapack	PkRTX	RTT	Sec	ACK	Throughput	Retardo	Utilización
TCPHybla	213989	52642	2610,09	161347	160398	323415,1	1,54	19,23
TCP HS	77508	59	5317,92	77449	76997	562834,00	4,65	9,05
TCPW	104654	51	3951,85	104603	104047	554078,68	3,34	12,24

Las tablas 3-38 a 3-40 indican el comportamiento de los parámetros de evaluación sin snoop.

Tabla 3-38. Parámetros de evaluación en en la prueba P sin Snoop.

	ndatapack	PkRTX	RTT	SEC	ACK	Throughput	Retardo	Utilización
TCP Hybla	110928	30750	5650,07	80178	79584	207983,15	4,04	9,36
TCP HS	55598	38	7276,54	55560	55069	402618,5	6,57	6,50
TCPW	63737	34	5676,45	63703	63252	401499,83	5,08	7,44

Tabla 3-39. Parámetros de evaluación en la prueba Q sin Snoop.

	ndatapack	PkRTX	RTT	SEC	ACK	Throughput	Retardo	Utilización
TCP Hybla	209676	65893	3206,68	143783	142740	308392,97	2,07	16.85
TCPHS	81428	56	5303,57	81372	80915	568818,17	4,64	9.51
TCPW	18300	38	17729,18	18262	17920	126692,15	17,04	2.10

Tabla 3-40. Parámetros de evaluación en en la prueba R sin Snoop.

	ndatapack	PkRTX	RTT	SEC	ACK	Throughput	Retardo	Utilización
TCP Hybla	219360	71310	3135,67	148050	146798	300499,07	2,02	17.33
TCP HS	75160	54	5317,85	75106	74633	560959,55	4,67	8.78
TCPW	75250	51	4027,92	75199	74865	521598,13	3,43	8.80

### 3.3.1.1. Descripción de resultados para la prueba P con y sin la inclusión del proxy snoop.

En las figuras 3-63 se indica el comportamiento de la ventana de congestión para las versiones de TCP Hybla, TCPW y TCP HS con y sin el uso del proxy para la prueba P.

La ventana de congestión para TCP HS indica que con la presencia del proxy snoop, se ve menos afectado por la congestión que existe en la red y debido a ello *CWND* se decrementa en su último intervalo como lo indica la figura 3-63, donde se infiere que el transmisor no ha recibido el mismo número de ACKs para mantener su crecimiento. El comportamiento de *CWND* en TCPW con el uso del proxy snoop se ve afectado por la congestión evidenciando un mayor RTT en el último intervalo, producto de la congestión de la red; el uso del proxy no evita que *CWND* entre a la fase de recuperación y retransmisión, sin embargo el nivel de pendiente de crecimiento de *CWND* con snoop es levemente mayor, y esa diferencia hace que se activen los mecanismos de recuperación rápida más pronto. TCPW solo se ve afectado por las pérdidas de sub-dimensionamiento del búfer durante la fase de inicio lento ya que el transmisor

implementa una política más agresiva teniendo en cuenta el ancho de banda disponible en la red.

### 3.3.1.1.1. Resultados obtenidos de la ventana de congestión.

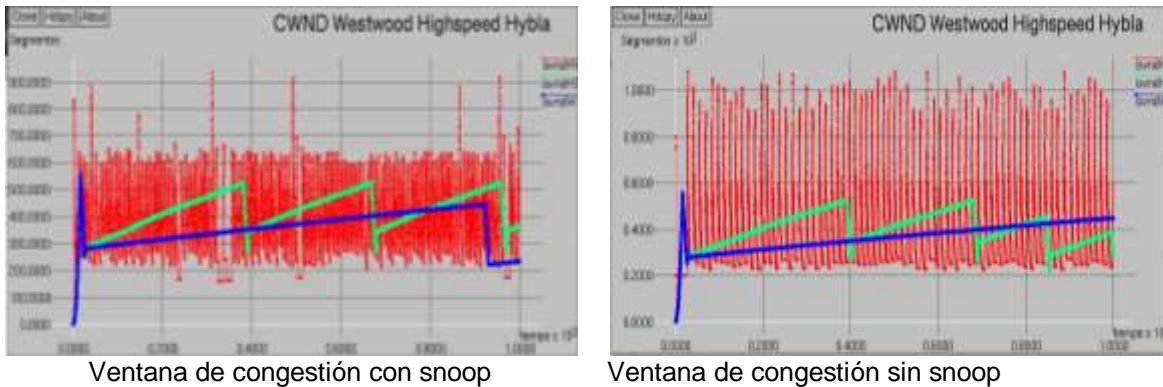


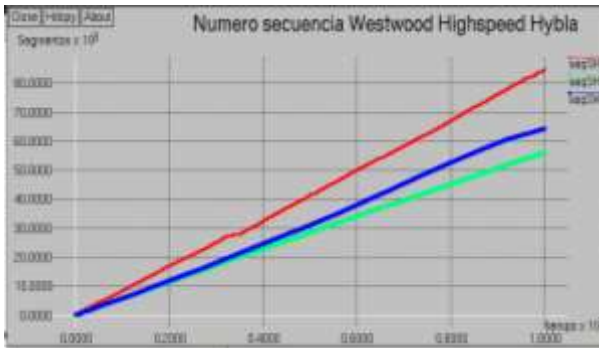
Figura 3-63. Comportamiento de CWND.

Uno de los puntos favorables para todas las versiones de TCP en estudio con el uso del proxy snoop está en la manera como *CWND* se comporta al comienzo de la simulación durante la fase de inicio lento (ver figura 3-63), la ventana de congestión crece mucho más rápido que sin snoop. Con respecto al comportamiento en TCP Hybla indica que sin snoop su nivel de *ssthresh* no es tan cambiante que con su uso, además el encolamiento de los paquetes es mayor con el uso de snoop, lo que implica un mayor retardo en la red viéndose afectado en más ocasiones por la congestión. *CWND* con snoop es más oscilante indicando que se recupera más rápido de la congestión presente en la red.

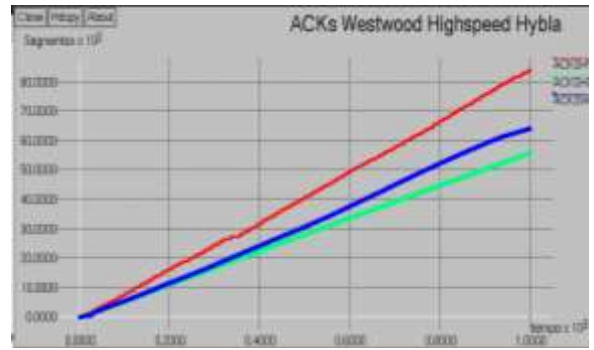
### 3.3.1.1.2. Datos obtenidos de ACK, y número de secuencia.

El valor del parámetro *sec* se incrementa según sea el número de ACKs que llegan al transmisor, su información indica cual es el número de paquetes que son recibidos correctamente por el transmisor, es decir, es la resta de los paquetes enviados menos el número de paquetes retransmitidos. En la prueba P el uso del snoop mejora el número de secuencia alcanzado en la red al compararse con el uso sin proxy para todas las versiones de TCP, sin embargo los valores obtenidos son similares cuando no se considera snoop. El mejor rendimiento se presentó con TCP Hybla aunque el número de paquetes transmitidos sea menor sin el uso del proxy, el número de paquetes retransmitidos se disminuye notoriamente y eso hace la diferencia.

Para las versiones de TCPW y TCP HS el número de paquetes retransmitidos son muy similares con y sin el proxy snoop. La evolución del número de ACKs indican que al utilizar el proxy snoop la red presenta un mejor comportamiento, sin embargo su mejoría no es tan notoria.

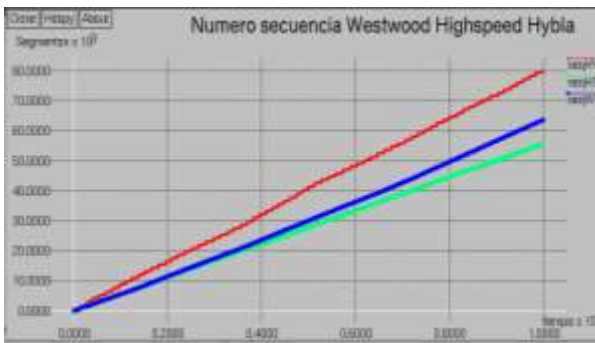


Evolución del número de secuencia

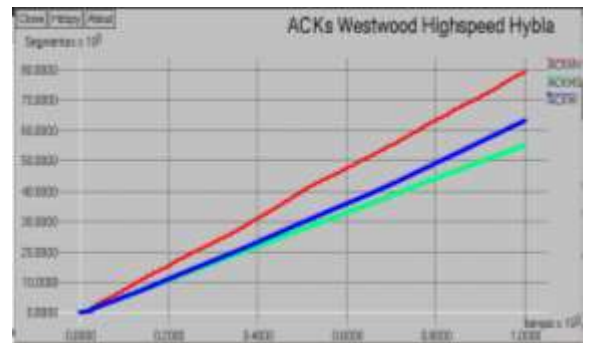


Evolución del ACK

Figura 3-64. Evolución del número de secuencia y ACK con Snoop.



Evolución del número de secuencia

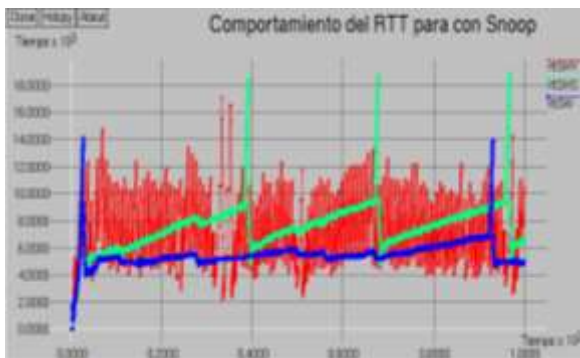


Evolución del ACK

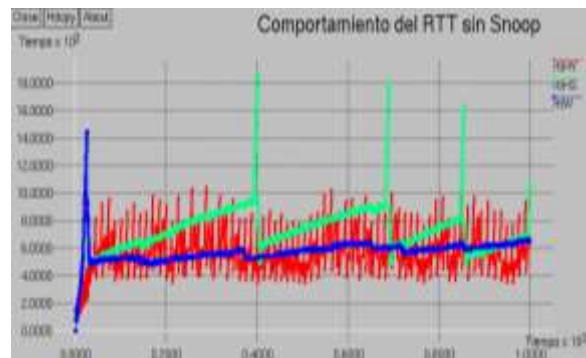
Figura 3-65. Evolución del número de secuencia y ACK sin Snoop.

### 3.3.1.1.3. Datos obtenidos de RTT y utilización del canal.

En las figuras 3-66 se indica el comportamiento del RTT con y sin snoop, y en las figuras 3-67 la utilización del canal para cada una de las versiones de TCP.



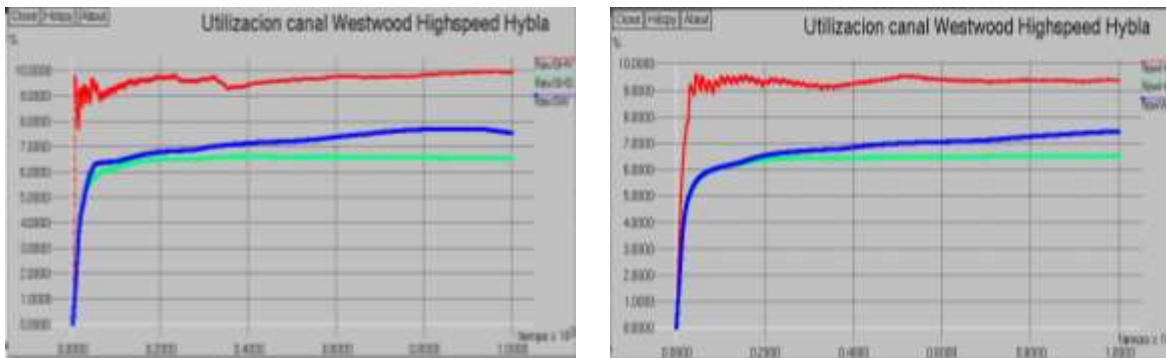
RTT con Snoop



RTT sin Snoop

Figura 3-66. Comportamiento del RTT con y sin snoop.





Utilización del canal con snoop.

Utilización del canal sin snoop

Figura 3-67. Utilización del canal con y sin snoop.

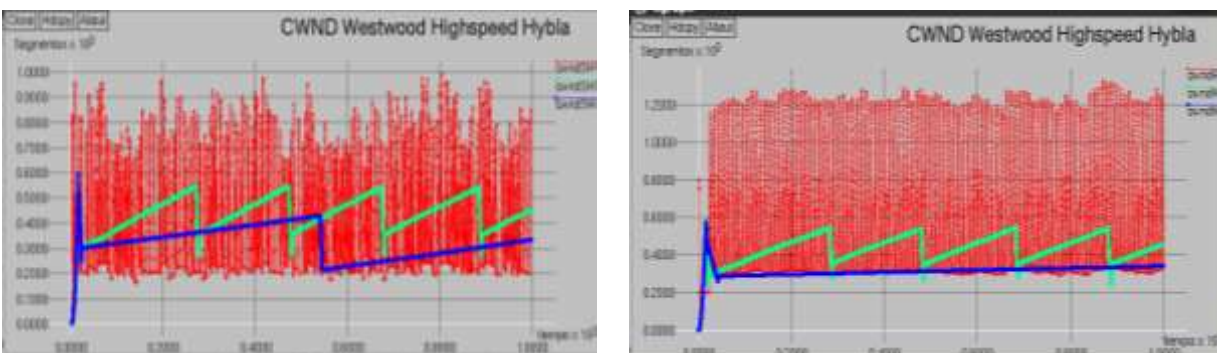
El comportamiento del RTT con el uso del snoop indica que para las versiones TCP Hybla y TCP HS el RTT se incrementa, sin embargo el desempeño de su ventana no se ve afectado por el nivel de congestión que se presenta en la red, este comportamiento está relacionado a la manera como el proxy snoop opera en la red, en TCP Hybla su comportamiento es más variante.

La utilización del canal sin el proxy es menor que con el proxy, aunque el *throughput* total de todas las conexiones no se mejora con el proxy como se observa en las tablas 3-35 y 3-38 para las versiones de TCP HS y TCPW donde su desempeño es muy similar. Las tablas 3-35 y 3-38 indican el comportamiento del retardo en la red con y sin el proxy snoop donde se obtiene un mejor resultado para TCP Hybla y TCP Westwood, que en TCP HighSpeed

### 3.3.1.2. Descripción de resultados en la prueba Q con y sin la inclusión del proxy snoop.

#### 3.3.1.2.1. Resultados obtenidos de la ventana de congestión.

En la figura 3-68 se observa el comportamiento de la ventana de congestión para las versiones de TCP Hybla, TCP HS y TCPW al hacer uso del proxy snoop.



Ventana de congestión con snoop

Ventana de congestión sin snoop

Figura 3-68. Comportamiento de CWND.

La ventana de congestión en TCPW al usar el proxy tiene un comportamiento más agresivo en la transmisión de segmentos en la red, con un nivel de pendiente mayor indicando que la red no se encuentra tan congestionada y donde el número de ACKs indica que el transmisor

constantemente incrementa los paquetes transmitidos. Además, el transmisor se ve en la necesidad de activar los mecanismos de retransmisión y recuperación rápida ya que con el uso del proxy la ventana se incrementa más rápido sin que el proxy pueda evitar activar esta retransmisión; los paquetes transmitidos en TCPW sin snoop son afectados por los altos tiempos de propagación.

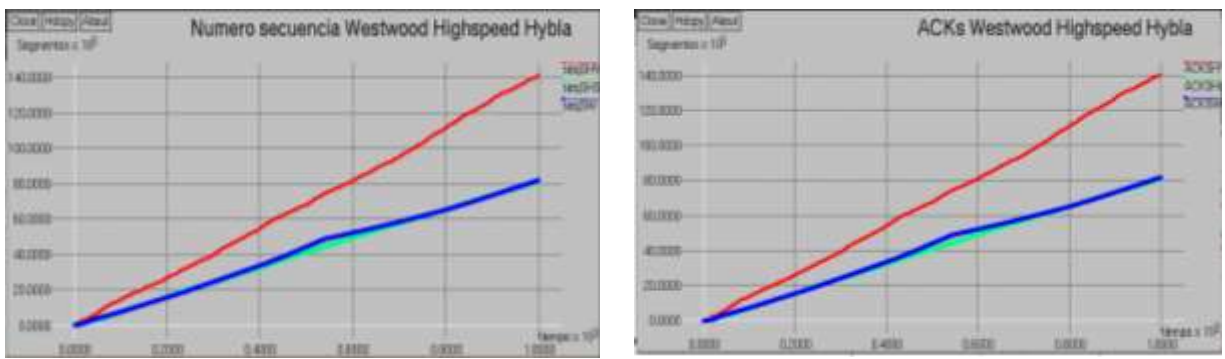
El comportamiento de la ventana de congestión para el caso de TCP Hybla muestra que presenta oscilaciones más continuas con el uso del snoop, con un nivel de *ssthresh* más bajo demostrando que un menor número de paquetes se transmiten ya que el valor de *CWND* y *ssthresh* en la figura 3-68 se mantiene sobre niveles más altos.

TCP Hybla con el uso del snoop presenta una mejor apertura y recuperación de la ventana de congestión en la fase de inicio lento y le permite hacer una mejor utilización del canal gracias a que el proxy elimina los ACKs duplicados que llegan del receptor, evitando que el transmisor active los mecanismos de recuperación. Al observar la ventana de congestión con TCP Hybla con snoop es de notar que debería presentar un mayor número de retransmisiones por el transmisor ya que se presentan muchas más pérdidas producidas por la limitación del tamaño del búfer en la RCST, sin embargo el uso del proxy elimina gran parte de los ACKs duplicados que activarían innecesariamente la ventana de congestión, donde el encargado de retransmitir los paquetes es el proxy snoop.

El comportamiento de *CWND* para TCP HS con y sin snoop es muy similar para esta prueba, como se indican en las figuras 3-68.

### 3.3.1.2.2. Datos obtenidos de ACK, y número de secuencia.

Las figuras 3-69 evalúan el incremento del número de secuencia con y sin el uso del proxy en la arquitectura de red, se observa claramente que TCPW con snoop presenta el mejor rendimiento al utilizar snoop ya que se ve menos afectado por los altos RTTs y el transmisor puede implementar una política más agresiva, las tablas 3-37 y 3-39 indican que con el uso del proxy snoop un mayor número de paquetes se transmiten, y el número de retransmisiones se incrementan debido a esa apertura de ventana y no porque el proxy no evite activar los mecanismos de recuperación rápida de ventana.



Evolución del número de secuencia

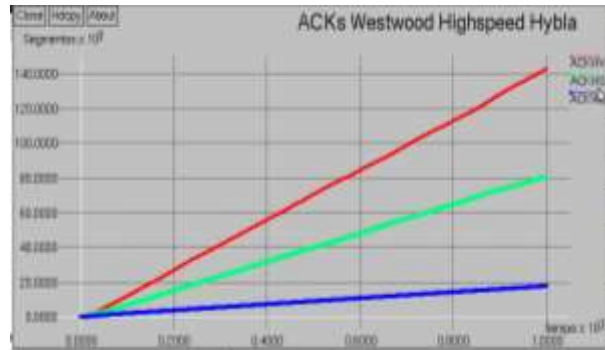
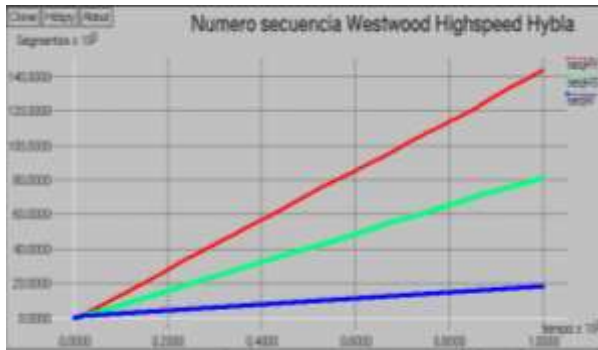
Evolución del ACK

Figura 3-69. Evolución del número de secuencia y ACK con Snoop.

El comportamiento en TCP Hybla respecto al crecimiento de ACKs al utilizar el snoop es inferior que sin snoop, en TCP HS el desempeño de la red para este parámetro es muy similar, sin



embargo el número de paquetes retransmitidos con snoop se aumenta, la mayoría de estos ocurren en la fase de inicio lento donde el uso del proxy snoop es más agresivo. En las figuras 3-69 y 3-70 se observa la evolución del ACK con y sin la presencia del snoop.



Evolución del número de secuencia

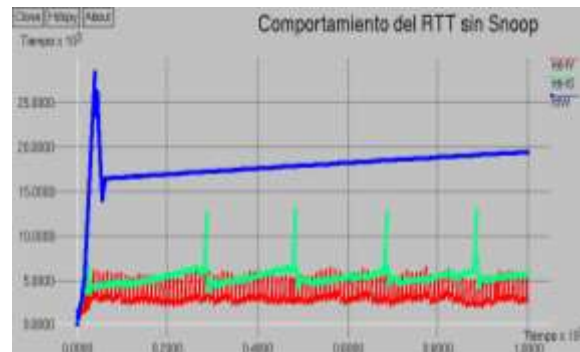
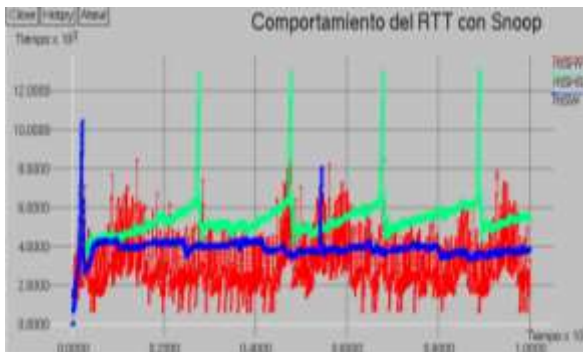
Evolución del ACK.

Figura 3-70. Evolución del número de secuencia y ACK sin Snoop.

### 3.3.1.2.3. Datos obtenidos de RTT y utilización del canal.

Los resultados arrojados en la simulación para el parámetro RTT indica que en TCP Hybla es menor con el uso del snoop, igualmente su retardo de encolamiento es menor, estas son las razones que justifican que con snoop se obtengan los mismos resultados en la utilización del canal que sin snoop.

En TCP HS el comportamiento del RTT y su utilización del canal son muy similares con y sin el uso del proxy. En las figuras 3-71 se indican el comportamiento del RTT con y sin snoop.



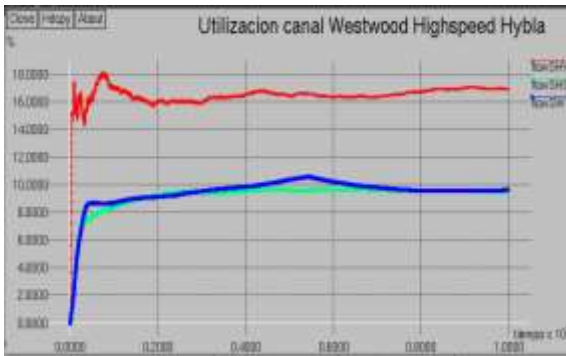
RTT con Snoop

RTT sin Snoop

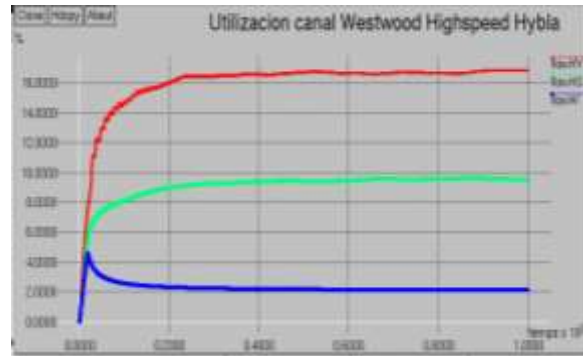
Figura 3-71. Comportamiento del RTT con y sin snoop.

Para las versiones TCP Hybla y TCP HS el uso del proxy snoop mejora la utilización del canal en sus respectivas fases de inicio lento, además se nota que para el caso de TCP HS su comportamiento no es afectado por el uso del proxy ya que conserva el mismo patrón observado que sin snoop. En las figuras 3-72 se indica la utilización del canal para las tres versiones de TCP.

En TCPW los resultados indican que el RTT sin snoop alcanza valores muy altos, donde su ventana presenta un nivel de pendiente muy bajo, de igual forma la utilización del canal es superior para el caso de snoop como se indica en las figuras 3-72.



Utilización del canal con snoop.



Utilización del canal sin snoop

Figura 3-72. Utilización del canal con y sin snoop.

### 3.3.1.3. Descripción de resultados en la prueba R con y sin la inclusión del proxy snoop.

#### 3.3.1.3.1. Resultados obtenidos de la ventana de congestión.

La figura 3-73 indica el comportamiento de la ventana de congestión para las versiones TCPW, TCP HS y TCP Hybla con el uso del snoop.

En la figura 3-73 TCP Hybla en los primeros segundos explora mejor la red gracias a la implementación del snoop, el número de paquetes retransmitidos con snoop es menor como lo indica la tabla 3-37, sin embargo el número de paquetes transmitidos es menor pero su valor es más cercano al de sin snoop. La ventana sin snoop indica que tarda más tiempo recuperarse del evento de una pérdida no obstante alcanza valores de ventana más altos, que demuestran un mayor volumen de paquetes encolados en el búfer de la RCST mientras que con el uso del snoop el búfer es menor y un mayor número de slots son asignados por la NCC.

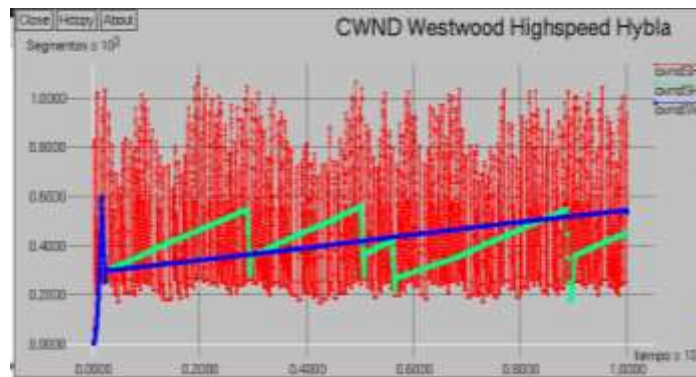


Figura 3-73. Comportamiento de CWND y ssthresh con snoop.

EL comportamiento de la ventana de congestión para TCP HS sin snoop en la figura 3-74 indica que alrededor del tiempo 500 se activan los mecanismos de recuperación y retransmisión rápida, sin embargo con el uso del proxy snoop ocurre un tiempo después; al principio el proxy elimina los ACKs duplicados como la congestión sigue avanzado el proxy no es capaz de retransmitir el segmento lo que obliga al transmisor hacerlo. Si se observa la ventana de TCPW con snoop es un ejemplo claro de cómo el snoop evita que el transmisor disminuya la ventana

de congestión cuando se presenta una pérdida. La tasa de paquetes enviados por el nodo fuente en los dos casos es igual, la diferencia ocurre cuando el snoop se encarga de la retransmisión del segmento.

Al observar las figuras 3-74 se indica el impacto de utilizar el proxy snoop dentro de la arquitectura de red.

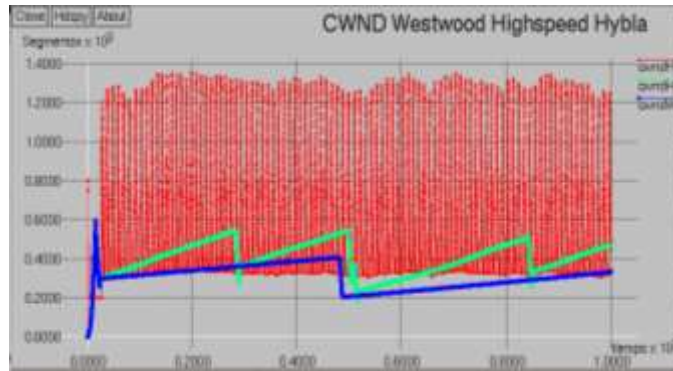
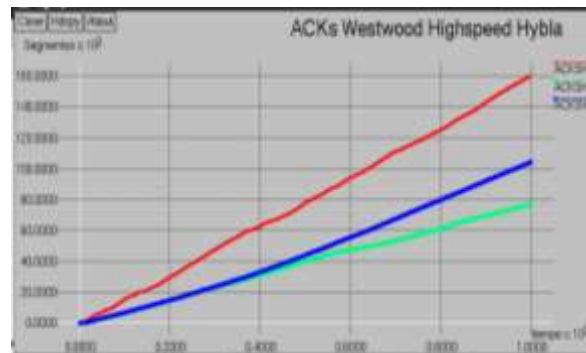
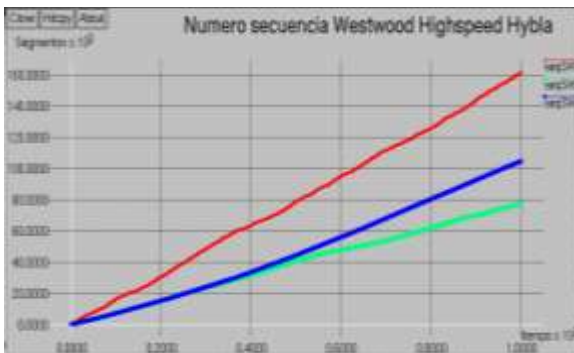


Figura 3-74. Comportamiento de CWND y ssthresh sin snoop.

### 3.3.1.3.2. Datos obtenidos de ACK, y número de secuencia.

Para todas las versiones de TCP la prueba R obtuvo el mejor desempeño con el uso del proxy snoop teniendo en cuenta el número de secuencia y la evolución del ACK, lo que conlleva a que la red presente un buen rendimiento, en las figuras 3-75 y 3-76 se muestran el comportamiento de estos parámetros con y sin el uso del proxy snoop.

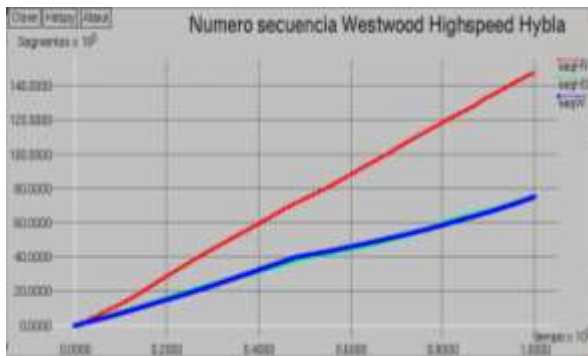
En esta prueba se observó que TCPW presentó el mejor desempeño con el uso del snoop al tener en cuenta el número de secuencia que indica una mejor relación entre paquetes transmitidos y retransmitidos. El desempeño en TCP Hybla sin el uso del proxy demuestra que su rendimiento no es tan bueno que con snoop, el número de paquetes a retransmitir en TCP HS indica que se presentan más pérdidas con el uso del proxy snoop y los paquetes de datos a transmitir se aumentan.



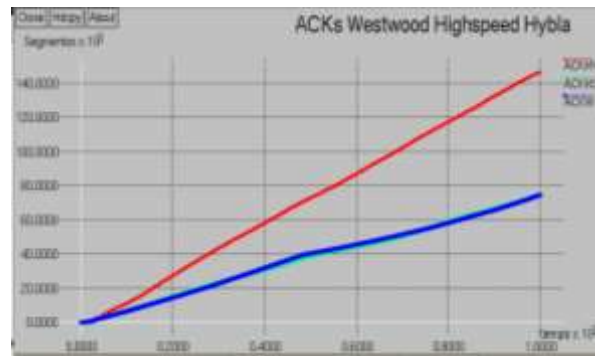
Evolución del número de secuencia

Evolución del ACK

Figura 3-75. Evolución del número de secuencia y ACK con Snoop.



Evolución del número de secuencia



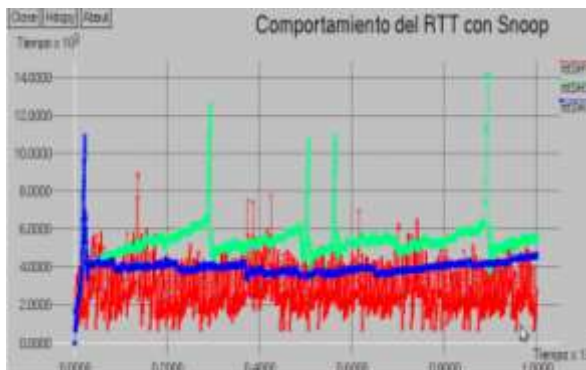
Evolución del ACK

Figura 3-76. Evolución del número de secuencia y ACK sin Snoop.

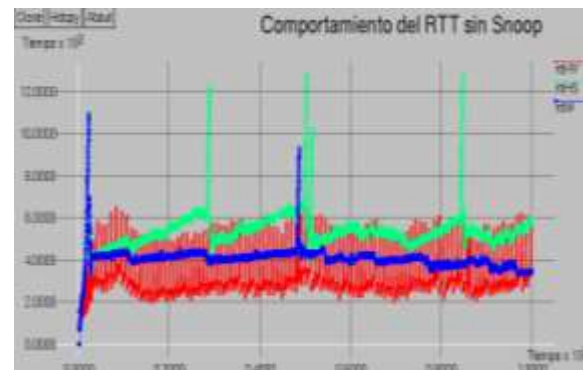
### 3.3.1.3.3. Datos obtenidos de RTT y utilización del canal.

Por último se indican las gráficas 3-77 y 3-78, las cuales indican el comportamiento del RTT y la utilización del canal con y sin la presencia del proxy snoop.

El retardo presente en la red así como el RTT es peor sin el uso del proxy snoop para todas las versiones de TCP, indicando que el transmisor se tarda más tiempo en incrementar su número de secuencia; los slots solicitados disminuyen al igual que la utilización del canal, tanto para la conexión en evaluación como para toda la red.

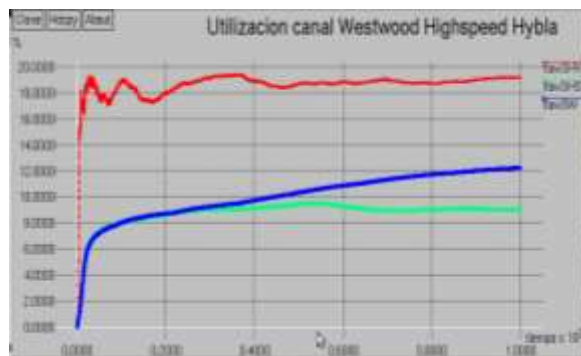


RTT con Snoop

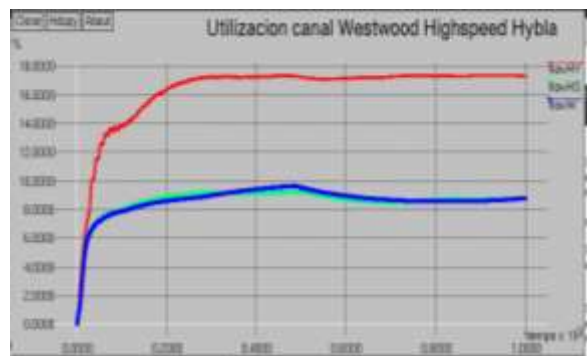


RTT sin Snoop

Figura 3-77. Comportamiento del RTT con y sin snoop.



Utilización del canal con snoop.



Utilización del canal sin snoop

Figura 3-78. Utilización del canal con y sin snoop.

### 3.3.2. Impacto del Snoop sobre un Entorno con RTTs muy Altos.

En la siguiente prueba se configura una política diferente para cada conexión como se indica en la tabla 3-41, además se utilizó el mecanismo de asignación de recurso Proporcional en la NCC. Si se hace uso de la política RBDC se conoce por pruebas anteriores que se presenta una monopolización de los recursos lo que conlleva un crecimiento inadecuado de CWND; debido a una alta presencia de retransmisiones y RTOs muy altos.

Tabla 3-41. Parámetros configurados sobre un entorno con RTTs altos con y sin snoop.

Parámetros y mecanismos a modificar	Pruebas			
	S	T	U	
Políticas de solicitud de recursos bajo demanda	Conexión		Política	FIJOS
	1		AVBDC	
	2		RBDC	
	3		CRA	
	4		VBDC	
	5		RBDC	
Tamaño Slot	1504	1504	1504	VARIABLE
Política de asignación de recursos en la NCC	Proportional	Proportional	Proportional	
Duración trama	0,05	0,05	0,06	
Nº Slots	14	20	24	
Ancho de Banda	3.368.960	4.812.800	4.812.800	
Supertrama	20	20	17	

En las tablas 3-42 a 3-47 se indica el comportamiento de la red con y sin la inclusión del proxy snoop, al implementar las diferentes políticas de asignación de demanda. La variable  $k$  indica el número de retransmisiones ocasionadas por la activación del *timeout*.

Tabla 3-42. Parámetros de evaluación en la prueba S con Snoop con la política RBDC.

	ndatapack	PkRTX	RTT	Sec	ACK	k	Throughput	Retardo	Utilización
TCP Hybla	859	14	192,22	845	7	11	3183,08	222,10	0,049
TCP HS	1203	33	402,11	1170	541	10	391186,11	450	0,0886
TCPW	1226	36	427,11	1190	544	14	385431,17	436.5	0,0928

Tabla 3-43. Parámetros de evaluación en la prueba T con Snoop con la política RBDC.

	ndatapack	PkRTX	RTT	Sec	ACK	k	Throughput	Retardo	Utilización
TCP Hybla	832	18	124,77	814	3	15	548666,34	125,84	0.025
TCP HS	1410	43	416,91	1367	573	8	416917,26	420,35	0.1230
TCPW	1442	53	439,30	1389	580	13	558372,75	450,8	0.1278



Tabla 3-44. Parámetros de evaluación en la prueba U con Snoop con la política RBDC.

	ndatapack	PkRTX	RTT	Sec	ACK	k	Throughput	Retardo	Utilización
TCP Hybla	835	13	29,58	822	3	11	555979,59	124,22	0.0251
TCP HS	1453	57	419,28	1396	582	12	564196,89	435,5	0.1291
TCPW	1805	345	361,76	1460	1418	6	561706,71	355,76	0.1759

Tabla 3-45. Parámetros de evaluación en la prueba S sin Snoop con la política RBDC.

	ndatapack	PkRTX	RTT	Sec	ACK	k	Throughput	Retardo	Utilización
TCP Hybla	37013	13361	36,22	23652	23188	3	377878,4	31,94	2.77
TCP HS	1212	39	31,57	1173	545	15	391320,08	32,5	0.0886
TCPW	1234	37	405,73	1197	549	11	387750,92	412,3	0.093

Tabla 3-46. Parámetros de evaluación en la prueba T sin Snoop con la política RBDC.

	ndatapack	PkRTX	RTT	Sec	ACK	k	Throughput	Retardo	Utilización
TCP Hybla	3842	2086	386,13	1756	18	6	548757,08	376,5	0.1832
TCP HS	1419	52	412,49	1367	581	10	560731,07	387,9	0.1222
TCPW	1441	52	435,74	1389	577	14	557731,43	423,6	0.1278

Tabla 3-47. Parámetros de evaluación en la prueba U sin Snoop con la política RBDC.

	ndatapack	PkRTX	RTT	Sec	ACK	k	Throughput	Retardo	Utilización
TCP Hybla	9866	4150	238,23	5716	5579	1	553461,08	245,54	0.6681
TCP HS	1478	67	431,16	1411	594	12	564220,01	437,8	0.1306
TCPW	1639	207	428,49	1432	598	7	562010,6	435,35	0.1492

Los resultados de las tablas fueron realizados para la conexión 5 con la política RBDC.

El comportamiento de TCP Hybla con snoop en presencia de RTTs muy altos es muy inferior al uso sin proxy, lo que indica que el temporizador en el transmisor se activa con más frecuencia dado que el proxy snoop elimina los ACKs duplicados que para el caso sin snoop, el cual le permitiría al transmisor disminuir su ventana de congestión para activar los mecanismos de recuperación y retransmisión rápida. TCP HS no se ve tan afectado en su desempeño al presentar un alto RTT en la red con el uso de snoop, el volumen de paquetes transmitidos es menor pero su valor es muy cercano al caso sin proxy, el número de retransmisiones como el número de *timeouts* disminuye lo que permite compensar el menor número de paquetes transmitidos.

## CAPITULO IV

### CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS

#### 4.0. CONCLUSIONES.

- Uno de los parámetros que se varió dentro del nivel de enlace de datos es el número de slots que conforman una trama TDMA, donde el impacto que genera el variar este parámetro en los mecanismos de control de flujo TCP es un crecimiento más rápido de la ventana de congestión durante las fases de inicio lento y evasión de congestión, una recuperación más rápida ante la presencia de ACKs duplicados, un crecimiento lineal con pendientes bien pronunciadas respecto al número de secuencia y los ACKs que arriban al transmisor, conllevando a un buen comportamiento del RTT y tiempos de encolamiento en el búfer pequeños, permitiendo de esta manera la transmisión de mucho más datos, incremento del *throughput* y una mejor utilización del canal de transmisión.
- Con un tamaño de slot pequeño el comportamiento de *CWND* es muy conservador, es decir durante la fase de inicio lento y evasión de congestión el crecimiento de *CWND* es lento y menos oscilante, lo que indica que se están transmitiendo menos paquetes, la razón de este comportamiento es causado por el retardo de propagación que se presenta en la red, debido a un menor ancho de banda con que se dispone para su uso, el cual se refleja en menos retransmisiones de paquetes y un bajo rendimiento de la red.
- Un tamaño de slot pequeño implica retrasos en la llegada de los ACKs lo que se ve reflejado en un incremento bastante notorio del RTT dando a entender al transmisor que la red se encuentra congestionada y por ende el crecimiento de *CWND* es muy conservado y limitado.
- Un tamaño de slot pequeño implica un menor número de paquetes por transmitir, donde las estaciones solicitadoras tienen que esperar un tiempo adicional para la asignación de recursos y transmisión de los datos, por tanto el transmisor tardará más tiempo en incrementar su ventana y en consecuencia el búfer seguirá aumentando.
- El tener una duración de trama más pequeña implica un crecimiento más rápido de *CWND* en la fase de inicio lento y en evasión de congestión, lo que se ve reflejado en la transmisión de más paquetes, una recuperación de *cwnd* mucho más rápido ante una pérdida, caso que no ocurre para una duración de trama más grande.
- El variar TBTP por encima y por debajo del valor sugerido por los autores del parche TDMA-DAMA (un segundo) no presenta un impacto muy significativo en el comportamiento de los parámetros de TCP, debido a que en una red satelital por los largos tiempos de propagación, el TBTP debe ser actualizado en función de un RTT ideal.
- *CWND* tiene una recuperación más rápida ante la pérdida de un paquete al utilizar la política proporcional ya que ésta hace una asignación de recursos mucho más equitativa entre todas las terminales durante todo el tiempo, lo que le permite a *CWND* crecer más rápido y no tener que esperar a que se le asigne recursos como lo hace la política FODA, lo que se ve reflejado en un mejor comportamiento de los demás parámetros de TCP.

- Cuando se utilizó el tipo de asignación de recursos FODA en la NCC, en muchas ocasiones se observó que los slots no fueron asignados en las estaciones terrenas por la NCC, presentándose un desperdicio de los recursos de la red, el cual influye en el comportamiento de TCP y el desempeño de la red, caso contrario se presenta en proporcional donde se hace un buen uso de la utilización del canal satelital.
- Al utilizar la política CRA con un tipo de tráfico FTP, el comportamiento de los parámetros TCP en sus diferentes versiones, se vio afectado por el número de slots solicitados a la NCC, presentando un incremento del RTT, un menor volumen de paquetes transmitidos y un mayor encolamiento del búfer, sin embargo estas condiciones están sujetas al número de slots fijos que se hayan configurado previamente en cada estación terrena.
- Una de las características que se observó en el comportamiento de RBDC es la manera poco equitativa de cómo se asignan los recursos a las diferentes estaciones terrenas, donde algunas conexiones tienden a monopolizar los recursos, luego el impacto que presenta RBDC en los mecanismos de control de flujo TCP es un bajo desempeño para aquellas conexiones que se ven afectadas por una disminución en los datos que ingresan al búfer.
- Con la política RBDC se presentaron RTTs muy altos, retransmisiones por *timeout* y una mayor congestión en la red, factores que influyen en el número de paquetes transmitidos, el número de ACKs que llegan al transmisor; resultando un mal comportamiento de TCP.
- Los mecanismos de control de flujo con la política VBDC presentaron el mejor comportamiento y desempeño de la red, debido a que las solicitudes realizadas por las RCSTs tienen en cuenta el encolamiento presente en el búfer de manera continua pretendiendo disminuir el encolamiento del búfer.
- Con la política RVBDC se observó un buen comportamiento de los parámetros de TCP, presentó la mejor utilización del canal durante los primeros segundos, en adelante se comporta de manera muy similar a como lo hace VBDC.
- El protocolo Aloha afecta notablemente el comportamiento de los parámetros de TCP en sus diferentes versiones, afectando de manera significativa en la transmisión de paquetes, *cwnd*, el RTT y el retardo de la red, ya que este protocolo no cuenta con mecanismos de sincronización y señalización que reduzcan la probabilidad de colisiones y por ende la pérdida de paquetes, resultando un menor número de ACKs que arriban al transmisor y una disminución del tamaño de *cwnd* el cual se ve reflejado en un bajo desempeño del protocolo TCP; por otro lado se tiene que entre mayor sea el tiempo aleatorio de retransmisión la probabilidad de colisiones entre paquetes disminuye, pero esto se ve reflejado en un mal comportamiento del protocolo TCP al presentarse altos RTTs obteniendo un mal rendimiento de la red.
- Al utilizar el proxy snoop se comprobó que su funcionamiento evita que el transmisor TCP entre continuamente en los mecanismos de recuperación y retransmisión rápida, sin embargo su funcionamiento en la red satelital con tráfico FTP no siempre mejora el desempeño de la red, los casos donde se observó un mejor rendimiento fue cuando el retardo entre la terminal RCST y el HUB es bajo.
- Se comprobó que en entornos donde la red presenta retardos de propagación muy altos el uso del proxy snoop se degrada su desempeño, la razón se debe a que el snoop elimina



inoportunamente los ACKs duplicados que le permitirían al transmisor (sin el caso del snoop) utilizar los mecanismos de retransmisión y recuperación rápida para disminuir su ventana de congestión, y no verse obligado a utilizar el mecanismo del *timeout* para su retransmisión.

#### **4.1. RECOMENDACIONES.**

- A pesar de que se encuentra muchos artículos relacionados con entornos satelitales, la información que se requiere para su programación en OTCL o su modificación en C++ es muy limitada, por ello se recomienda hacer una exploración previa para determinar el nivel de complejidad del proyecto.
- Para trabajos futuros en esta misma línea de investigación se recomienda contactar al grupo de investigación CNR de la universidad de Roma Italia.
- Se recomienda la exploración y uso de la herramienta de simulación NS-2 al interior de la Facultad de Ingeniería Electrónica y Telecomunicaciones ya que es una herramienta que permite profundizar y analizar diferentes temas relacionados al desempeño de redes satelitales, inalámbricas y terrestres.

#### **4.2. TRABAJOS FUTUROS.**

- Evaluar cuál es el impacto que presenta el protocolo TCP al implementar el mecanismo de acceso al medio MF-TDMA en la herramienta de simulación NS-2.
- Comparar el desempeño del protocolo TCP en una red satelital al utilizar los proxies que modifican la arquitectura de red que no fueron implementados en este trabajo, como lo son SatPET, TCPSpliting, y Web Caching.
- Evaluar cuál es el impacto que generan los diferentes mecanismos de gestión y planificación de colas en el protocolo TCP en una red satelital.
- Realizar un análisis similar al desarrollado por este trabajo pero en una red satelital LEO.
- Adaptar el módulo Snoop a las características de una red satelital que permita mejorar su desempeño.

#### **4.3. APORTES DEL PROYECTO.**

- Se realizó un estudio de las diferentes versiones TCP para una red satelital.
- Uno de los aportes que deja el proyecto es la implementación y configuración de una red satelital con las diferentes versiones de TCP y los diferentes procedimientos que permiten evaluar el desempeño de los parámetros TCP y de rendimiento en la red mediante la herramienta NS-2.
- Un estudio detallado de cómo impactan algunos mecanismos del nivel de enlace de datos en el desempeño del protocolo TCP en una red satelital.

## BIBLIOGRAFÍA

- [1] A. Romero, "Globalización y pobreza", Ediciones Unariño, Universidad de Nariño, Colombia, 2002, ch. Universidad y globalización, pp 72-81.
- [2] A. Salamanca, "*Study and Performance Evaluation of TCP versions over GEO and LEO Satellite Links between Performance Enhancement Proxies*", facultad de informática, Universidad Politécnica de Madrid, España, abril 2008.
- [3] M. Allman, D. Glover, y L. Sanchez, "*Enhancing TCP over Satellite Channels using Standard Mechanism*", IETF RFC 2488, 1999, [online].
- [4] M. Handley, "EL Protocolo TCP ".Universidad Politécnica de Cataluña, Disponible FTP:[http://www.tesisenxarxa.net/TESIS\\_UPC/AVAILABLE/TDX-1222106-164746//04AMCA04de15.pdf](http://www.tesisenxarxa.net/TESIS_UPC/AVAILABLE/TDX-1222106-164746//04AMCA04de15.pdf).
- [5] M.Luglio, C.Roseti, M.Gerla, "*TCP Performance over Satellite in case of Multiple Sessions per Links using Efficient Flow Control and Real OS*", M.S. tesis, Dep. Ing. Elect, Univ. de Roma Italia, 2004.
- [6] J. Border, M. Kojo, J. Griner, G. Montenegro, y Z. Shelby, "*Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations*", RFC 3135, Jun. 2001. [online]. Disponible: <http://www.faqs.org/rfcs/rfc3135.html>.
- [7] M. Graca, "Internet vía satélite", M.S. tesis, Dep. Ing. Elect y Telecc, Univ del Cauca, Colombia, 2000.
- [8] D. Brown, *Handbook on Satellite Communications (HSC)*, Ed. 3 Edición Wiley, 2005.
- [9] L. Wu, F. Peng, y V. Leung, "*Dynamic Congestion Control for Satellite Networks Employing TCP Performance Enhancement Proxies, IEEE paper 8272948, ISBN 0-7803-8523-3, Enero 2005*
- [10] L. Wu, F. Peng, y V. Leung, "*Dynamic Congestion Control for Satellite Networks Employing TCP Performance Enhancement Proxies 1*", *IEEE paper 8272948, ISBN 0-7803-8523-3, Enero 2005*
- [11] M. Todorović, *Comparative Study of the End-To-End Compliant TCP Protocols for Wireless Networks*, M.S. Tesis, Dep. Ing. Univ. de Texas, EEUU, Dic. 2005.
- [12] L. Brakmo y L. Peterson (1995, Oct), "*TCP Vegas: End-To-End Congestion Avoidance on a Global Internet*", *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*. Vol 13, N° 8, 1995, Disponible: <http://www.cs.toronto.edu/syslab/courses/csc2209/06au/papers/vegas.pdf>
- [13] X. Zhou y J. Baras (2008, Dic), "*TCP over Satellite Hybrid Networks: A Survey*", Universidad de Maryland, [En línea]. Disponible: [http://www.isr.umd.edu/~baras/publications/reports/2002/ZhouB\\_TR\\_2002-27.pdf](http://www.isr.umd.edu/~baras/publications/reports/2002/ZhouB_TR_2002-27.pdf).

- [14] M. Carter, "A Review of Transport Protocols as Candidates For Use in a Tactical Environment", M.S. Tesis, Dep defence, Australian Government, Australia, Dic. 2005.
- [15] M. Barreto, M. Belino, y M. San Martín (2003, May), "TCP SACK", Instituto de Ingeniería Electrónica Universidad de la República. [En línea]. Disponible: [http://iie.fing.edu.uy/ense/asign/perfredes/trabajos/trabajos\\_2004/TCPsack/TCPsack.pdf](http://iie.fing.edu.uy/ense/asign/perfredes/trabajos/trabajos_2004/TCPsack/TCPsack.pdf).
- [16] S. West y N. Vaidya, (1997. April). "TCP Enhancements for Heterogeneous Networks", Universidad de Texas A&M. EEUU. [En línea]. Disponible: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.34.9967&rep=rep1&type=pdf>
- [17] S. Floyd and T. Henderson, "The New Reno modification to TCP's fast recovery algorithm", RFC 2582, Abril 1999. [En línea]. Disponible:<http://www.faqs.org/rfcs/rfc2582.html>.
- [18] S. Fernández y A. Feliu, "Diseño de un Entorno para el Estudio de los Parámetros de Funcionamiento del Protocolo TCP", M.S tesis, Univ. Politécnica Cataluña, Jul. 2006.
- [19] D. Guzmán, y C. Lara. "Métodos de control de congestión". WebMastes Mag. [En línea]. Disponible: <http://profesores.elo.utfsm.cl>.
- [20] K. Fall y S. Floyd, "Simulation-based Comparisons of Tahoe, Reno, and SACK TCP", Lawrence Berkeley National Laboratory, 1997.
- [21] R. Kettimuthu y W. Allcock, "Improved Selective Acknowledgment Scheme for TCP", Argonne, IL 60439 EEUU, 2003.
- [22] M. Mathis, J. Mahdavi, S. Floyd, y A. Romanow, "TCP Selective Acknowledge Options", RFC 2018, Oct. 1996, [En línea], Disponible: <http://www.rfc-es.org/rfc/rfc2018-es.txt>.
- [23] C. Caini, R. Firrincieli, M. Marchese, T. de Cola, M. Luglio, C. Roseti, N. Celandroni, y F. Potortí (2006, Julio), "Transport layer protocols and architectures for satellite networks", INTERNATIONAL JOURNAL OF SATELLITE COMMUNICATIONS AND NETWORKING. [En línea]. Disponible: <http://www.tlcsat.uniroma2.it/wp-content/uploads/2007/04/tcpsat-ijsc06.pdf>.
- [24] C. Caini y R. Firrincieli (2004, Mayo). "TCP Hybla: a TCP enhancement for heterogeneous networks", INTERNATIONAL JOURNAL OF SATELLITE COMMUNICATIONS AND NETWORKING. [En línea], Disponible: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.100.2600&rep=rep1&type=pdf>.
- [25] E. Paaby, "Evaluation of TCP Retransmission Delays", M.S. tesis. Dep. Informatics. Univ de Oslo, May. 2006.
- [26] J. Hoe, "Improving the Start-Up Behavior of A Congestion Control Scheme for TCP", Instituto Tecnológico Massachusetts, 1997.
- [27] C. Caini, N. Candio, R. Firrincieli, y G. Giambenet. "TCP Hybla Performance in GEO Satellite Networks: Simulations and Testbed". IEEE paper INSPEC Accession Number: 9287208. Print ISBN: 1-4244-0118-6. Pag. 41-45. Septiembre 2006.

- [28] J. Chen, F. Paganini, R. Wang, M. Sanadidi, y M. Gerla (2003). "*Fluid-flow Analysis of TCP Westwood with RED*", *IEEE GLOBECOM 2003*. [En línea]. Disponible: <http://www.ee.ucla.edu/~paganini/PDF/Papers/westwood.pdf>.
- [29] R. Wang, G. Pau, K. Yamada, M. Sanadidi, y M. Gerla (2004). "*TCP Startup Performance in Large Bandwidth Delay Networks*", *IEEE INFOCOM 2004*. ISBN 0-7803-8356-7/04. [En línea]. Disponible: <http://www.cs.ucla.edu/~gpau/Papers-C/C018-2004-Infocom.PDF>.
- [30] A. Zanella, G. Procissi, M. Gerla, y M. Sanadidi (2001). "*TCP Westwood: Analytic Model and Performance Evaluation*", *Proceedings of IEEE Globecom*. [En línea]. Disponible: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.18.556>.
- [31] M. Gerla, B. Ng, M. Sanadidi, M. Valla, y R. Wang, "*TCP Westwood with Adaptive Bandwidth Estimation to Improve Efficiency/Friendliness Tradeoffs*", *Dep. Computer Science. Univ de California*. 2002.
- [32] S. Mascolo, L. Grieco, R. Ferorelli, P. Camarda, y G. Piscitelli (2003, Abril). "*Performance Evaluation of Westwood+ TCP Congestion Control*", *Politécnico di Bari, Italia*. [En línea]. Disponible:<http://c3lab.poliba.it/images/7/76/Perf-04.pdf>.
- [33] L. Grieco y S. Mascolo, "*Mathematical Analysis of Westwood+ TCP Congestion Control*", *Italia*, 2004.
- [34] E. Altman, C. Barakat, S. Mascolo, N. Möller, y J. Sun, "*Analysis of TCP Westwood+ in High Speed Networks*". *Francia*, 2005.
- [35] E. Sogard. "*Evaluation of TCP retransmission delays*". *M.S. Tesis. Dep. Informatics. Univ. de Oslo, Oslo* 2006.
- [36] E de Souza, "*A simulation-based study of HighSpeed TCP and its deployment*", *M.S. tesis. Universidad de California, EEUU, Mayo* 2003.
- [37] S. Kerkar, "*Performance Analysis of TCP/IP Over High Bandwidth Delay Product Networks*", *M.S. tesis. Dep. Of computer Science & Engineering. Univ. del Sur de Florida, EEUU* 2006.
- [38] E de Souza, D Agarwal, "*A HighSpeed TCP Study: Characteristics and Deployment Issues*", *State University of Campinas, Campinas, Brazil* 2003.
- [39] C. Caini, R. Firrincieli, y D. Lacamera (2005, junio). "*PEPsal: A TCP Performance Enhancing Proxy for Satellite Links*", *Universidad de Bologna, Italia*, 2005. [En línea]. Disponible:<http://www.danielinux.net/projects/pepsal.pdf>.
- [40] J. Border, M. Kojo, J. Griner, G. Montenegro, y Z. Shelby, "*Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations*", *RFC 3135, Jun. 2001*, [En línea]. Disponible: <http://www.faqs.org/rfcs/rfc3135.html> .
- [41] T. Anderson, "*TCP/IP over Satellite: Optimization vs. Acceleration, End to End Communications*", *Inc. - White Paper, Abril*. 2005.

- [42] J. Ishac y M. Allman (2001, Septiembre). "On The Performance of TCP Spoofing In Satellite Networks", NASA Center for Aerospace Information. EEUU. 2001. [En línea]. Disponible: <http://gltrs.grc.nasa.gov/reports/2001/TM-2001-211151.pdf>.
- [43] M. Omueti, "TCP-ADaLR: TCP with Adaptive Delay And Loss Response for Broadband Geo Satellite Networks", M.S. tesis, Univ. de Simon Fraser, 2007.
- [44] I. Akyildiz, G. Morabito, y S. Palazzo (Junio, 2001), "TCP-Peach: A New Congestion Control Scheme for Satellite IP Networks". *IEEE/ACM TRANSACTIONS ON NETWORKING*, Vol. 9, N°. 3. [En línea]. Disponible: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.9.5266&rep=rep1&type=pdf>.
- [45] V. Bharadwaj, J. Baras, y N. Butts (1999). "An Architecture for Internet Service via Broadband Satellite Networks", *Center for Satellite and Hybrid Communication Networks* Universidad de Maryland, EEUU. [En línea]. Disponible: [http://www.lib.umd.edu/drum/bitstream/1903/6015/1/TR\\_99-22.pdf](http://www.lib.umd.edu/drum/bitstream/1903/6015/1/TR_99-22.pdf).
- [46] J. Sing y B. Soh, "On the use of Snoop with Geostationary Satellite Links" *IEEE paper 8642386, Print ISBN: 0-7695-2316-*. Australia, 2005.
- [47] F. Teja, L. Vidal, y L. Alves (2004, Marzo), "TCP sobre enlaces wireless Problemas y algunas posibles soluciones existentes", Universidad de la República, España. [En línea]. Disponible: <http://www.unlu.edu.ar/~tyr/tyr/TYR-2005/TCPW.pdf>.
- [48] A. Khurri, "TCP: Error Handling with Explicit Notification Schemes", Universidad de Helsinki, Octubre 2004.
- [49] X. Zhou y J. Baras (2002, Enero), "TCP over Satellite Hybrid Networks: A Survey", *Center for Satellite and Hybrid Communication Networks and Universidad de Maryland*, EEUU. [En línea]. Disponible: [http://www.isr.umd.edu/~baras/publications/reports/2002/ZhouB\\_TR\\_2002-27.pdf](http://www.isr.umd.edu/~baras/publications/reports/2002/ZhouB_TR_2002-27.pdf).
- [50] C. Caini, R. Firrincieli, y D. Lacamera (2006), "PEPsal: A TCP Performance Enhancing Proxy for Satellite Links", Universidad de Bologna, Italia. [En línea]. Disponible: <http://www.danielinux.net/projects/pepsal.pdf>.
- [51] Rodolfo, Neri (2003, Mayo). *Comunicaciones por satélite* (2rd ed.). Editorial Thomson, Disponible: Biblioteca universidad del Cauca.
- [52] M. Calvo y R. Ramón (2008, junio). "Comunicaciones por Satélite, Acceso Multiple", Universidad Politecnica de Madrid. España. [En línea]. Disponible: [http://www.gr.ssr.upm.es/docencia/grado/csat/material/CSA08-2-FormatoTLE\\_2p.pdf](http://www.gr.ssr.upm.es/docencia/grado/csat/material/CSA08-2-FormatoTLE_2p.pdf).
- [53] L. Boggio y C. Lana, *Multiplexación y acceso múltiple* (1999). Webmaster [En línea]. Disponible: <http://www.eie.fceia.unr.edu.ar/ftp/Comunicaciones/MUX.PDF>.
- [54] D. Roddy (2005), *Satellite Communication* (3rd ed.), Edit McGraw-Hill.
- [55] A. Andueza (2006, Julio), *Tecnologías de acceso*, Universidad de Navarra, España. [En línea]. Disponible: <http://www.unavarra.es/organiza/etsiit/cas/estudiantes/>.
- [56] I. Blánquez y C. Lozano (2002), "Tecnología y servicios de internet por Satélite" Universidad de Chile, Chile. [En línea]. Disponible: <http://www.informatica-hoy.com.ar/internet/Internet-por-satelite.pdf>

[57] J. Alcober, "Aportación al Estudio de Protocolos de Acceso Múltiple: El protocolo Aloha Estabilizado de Ventana", M.S. tesis, Dep. matemática aplicada y telemática. Univ Politécnica de Cataluña, Mar. 1997.

[58] "El SS-Aloha en Constelaciones de Satélites LEO". WebMaster. [En línea]. Disponible:[http://www.upv.es/satelite/trabajos/Grupo13\\_99.00/tema5.html](http://www.upv.es/satelite/trabajos/Grupo13_99.00/tema5.html).

[59] M. Alonso Cuevas, M.Casas Lago (2003). "Televisión Digital Vía Satélite", Escuela Técnica Superior de Ingenieros de Telecomunicación de Vigo, España. [En línea]. Disponible:<http://www.com.uvigo.es/asignaturas/scvs/trabajos/curso0203/dvb-s/DVB-S.pdf>

[60] H. Barbara, J.Chafra, "Simulación de una red VSAT Full-Duplex para el acceso a internet usando la plataforma DVB-S y DVB-RCS", M.S. tesis, Escuela Politécnica Nacional, Quito Ecuador, Marzo 2006.

[61] R. Hernandez (2006), "Capitulo 2 transmision de señales de TV digital por satélite DVB-S", Escuela Politécnica Nacional, Ecuador. [En línea]. Disponible: <http://bieec.epn.edu.ec:8180/dspace/bitstream/123456789/1419/4/T11385%20CAP%202.pdf>

[62] C. Roseti (2004, Marzo). "*TCP behavior in a DVB-RCS environment*". 24th AIAA International Communications Satellite Systems Conference (ICSSC) and 4th 11 - 14 June 2006, San Diego, California. [En línea]. Disponible: [http://pdf.aiaa.org/preview/CDReadyMICSSC06\\_1269/PV2006\\_5404.pdf](http://pdf.aiaa.org/preview/CDReadyMICSSC06_1269/PV2006_5404.pdf).

[63] A.Gotta, F. Potorti, R.Secchi, "*Simulating Dynamic Bandwidth Allocation on Satellite Links*", Genova, Italia, 2005.

[64] P.Chinin, G. Giambene, C Roseti, M. Luglio (2005, Octubre). "Cross-layer Management of Radio Resources in an Interactive DVB-RCS-based Satellite Network" International symposium on computer and information sciences N°20, Istanbul, TURQUIE. [En línea]. Disponible: <http://cat.inist.fr/?aModele=afficheN&cpsidt=17415668>

[65] P.Chinin, G. Giambene, C Roseti, M. Luglio (2006, Marzo). "*Dynamic resource allocation based on a TCP-MAC Cross-Layer approach for Interactive Satellite Networks*". INTERNATIONAL JOURNAL OF SATELLITE COMMUNICATIONS AND NETWORKING. [En línea]. Disponible: <http://www.tlcsat.uniroma2.it/wp-content/uploads/2007/04/sat851-24-5-367-385.pdf>.

[66] S. Haykin (2006, Mayo), *Sistemas de comunicación (4ta ed.)* Editorial Limusa Willey.

[67] P.Chinin, G. Giambene, C Roseti, M. Luglio (2006, Septiembre), "*Dynamic Resource Allocation based on a TCP-MAC Cross-Layer Approach for Interactive Satellite Networks*", International Journal of Satellite Communications and Networking. [En línea]. Disponible: <http://www3.interscience.wiley.com/journal/113328332/abstract?CRETRY=1&SRETRY=0>

[68] A. Rodatos, "*A proxy evaluation within wired and wireless simulations*", M.S. tesis, Univ. de Cataluña, Sep. 2005.

[69] C. Caini y R. Firriencili (2006), "*End-to-end TCP enhancements performance on satellite links*". Proceedings of the 11th IEEE Symposium on Computers and Communications. [En línea]. Disponible: <http://portal.acm.org/citation.cfm?id=1157991>.

[70] R. Secchi (2006), "TDMA-DAMA" Italia. [En línea]. Disponible: <http://ala.isti.cnr.it/wnlab/tdmadama>.

[71] H. Peng. (2005). "*Aloha protocol*". MLDesign Technologies. EEUU. [En línea]. Available:[http://www.mldesigner.com/fileadmin/assets/MLDesigner/Examples/NetworkingApplications/ALOHA\\_Protocol/ALOHA.pdf](http://www.mldesigner.com/fileadmin/assets/MLDesigner/Examples/NetworkingApplications/ALOHA_Protocol/ALOHA.pdf).

[72] K. Fall (2007), "*The Ns Manual*". Universidad de Berkeley. EEUU. [En línea]. Disponible: <http://www.isi.edu/nsnam/ns/doc/>.

[73] C. Carreón, V. Gatica (2008, Noviembre, 26), "*Evaluacion del Desempeño del Protocolo Aloha*", Instituto Politécnico Nacional Escuela superior de Ingeniería Mecánica y Eléctrica. [En línea]. Disponible: <http://itzamna.bnct.ipn.mx:8080/dspace/handle/123456789/1714>.