Centro de Investigación Científica y de Educación Superior de Ensenada

Analisis de Variantes de TCP en Servicios IP Sobre Redes Satelitales

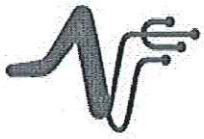
TESIS
MAESTRIA EN CIENCIAS

JOSE DAMIÁN CANTU DAVILA

ENSENADA BAJA CFA, MEXICO DICIEMBRE DE 2005

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y DE EDUCACIÓN SUPERIOR DE ENSENADA

PROGRAMA DE POSGRADO EN CIENCIAS
EN ELECTRÓNICA Y TELECOMUNICACIONES
CON ORIENTACIÓN EN TELECOMUNICACIONES



ANÁLISIS DE VARIANTES DE TCP EN SERVICIOS IP SOBRE REDES SATELITALES.

TESIS

que para cubrir parcialmente los requisitos necesarios para obtener el grado de
MAESTRO EN CIENCIAS

Presenta:
JOSÉ DAMIÁN CANTÚ DÁVILA

Ensenada, Baja California, México, Diciembre del 2005.

Posgrado

TESIS DEFENDIDA POR

José Damián Cantú Dávila

Y APROBADA POR EL SIGUIENTE COMITÉ

Dr. Roberto Conte Galván.
Director del Comité.

Dr. David Hilario Covarrubias Rosales.
Miembro del Comité.

Dr. José Antonio García Macías.
Miembro del Comité.

M.C. Jorge Enrique Preciado Velasco.
Miembro del Comité.

M.C. Raúl Tamayo Fernández.
Miembro del Comité.

Dr. Arturo Velázquez Ventura.
Coordinador del programa de posgrado en Electrónica y Telecomunicaciones.

Dr. Raúl Ramón Castro Escamilla.
Director de Estudios de Posgrado.

2 de Diciembre de 2005

RESUMEN de la Tesis presentada por **José Damián Cantú Dávila**, como requisito parcial para la obtención del grado de **MAESTRO EN CIENCIAS** en **ELECTRÓNICA Y TELECOMUNICACIONES CON ORIENTACIÓN TELECOMUNICACIONES**. Ensenada, Baja California, México. Diciembre de 2005.

ANÁLISIS DE VARIANTES DE TCP EN SERVICIOS IP SOBRE REDES SATELITALES

Resumen aprobado por:

Dr. Roberto Conte Galván
Director del Comité

El reciente crecimiento en uso y tamaño de las redes de comunicaciones basadas en el Protocolo de Internet (IP) se ha dado en gran medida al éxito que ha experimentado la Internet. El protocolo de transporte dominante para este tipo de redes es el Protocolo de Control de Transmisión (TCP), y a pesar de que fue un protocolo creado inicialmente para redes cableadas terrestres, su uso se ha extendido actualmente hacia ambientes inalámbricos, celulares y satelitales. El ambiente satelital impone una serie de retos al desempeño de TCP, entre los que se destacan el gran retardo de propagación y el deterioro del canal. TCP fue diseñado para redes cableadas, donde las pérdidas de paquetes son debidas a la congestión en la red misma. En el ambiente satelital las pérdidas se presentan a veces por congestión, en otras debido al gran retardo extremo a extremo (latencia) del enlace satelital, y la alta tasa de error de bit (BER) del canal puede también afectar la información. Algunas opciones propuestas para el uso de TCP en el espacio incluyen utilizar satélites de órbita baja, ya que ofrecen menores pérdidas y retardos de propagación, y se han propuesto variantes de TCP que mejoran su desempeño ante situaciones de congestión. De ahí surge la inquietud de este trabajo de tesis de evaluar el desempeño de algunas variantes del protocolo TCP considerando el efecto de sus algoritmos de control de congestión en escenarios satelitales simulados de órbita baja. El objetivo es establecer un marco de referencia que permita determinar si es posible ofrecer Calidad de Servicio similar a la de VoIP utilizando variantes de TCP en un ambiente satelital de órbita baja. En este trabajo se realiza el análisis y modelado de variantes de TCP tales como Tahoe, Reno, NewReno, SACK y Westwood, evaluando particularmente parámetros de QoS tales como latencia, jitter, BER, PLR y Caudal Eficaz. En base al análisis y modelos planteados en esta tesis, se realizaron simulaciones de transferencia de información con el simulador de red ns2 (network simulator 2) a través de un satélite de órbita baja utilizando variantes de TCP. A partir de los resultados obtenidos en este trabajo se establece la posibilidad de ofrecer QoS a un nivel similar al de VoIP, con algunas limitaciones, cuando se utilizan ciertas variantes de TCP sobre redes de información IP usando satélites de órbita baja.

Palabras clave: TCP, Algoritmo de control de congestión, variantes de TCP (Tahoe, Reno, NewReno, SACK, Westwood), satélite de órbita baja, QoS, VoIP.

ABSTRACT of the Thesis of **José Damián Cantú Dávila**, presented as a partial requirement to obtain the **MASTER in SCIENCE** degree in **ELECTRONICS AND TELECOMMUNICATIONS ORIENTED TO TELECOMMUNICATIONS**. Ensenada, Baja California, México. December 2005.

ANALYSIS OF TCP VARIANTS IN IP SERVICES OVER SATELLITE NETWORKS

Abstract approved by:

Dr. Roberto Conte Galván
Thesis Adviser

Recent growth in the use and size of communications networks based in the Internet Protocol (IP) has happened mainly due to the success experimented by the Internet. The dominant transport protocol for this type of networks is the Transmission Control Protocol (TCP), and although it was initially created for terrestrial cabled networks, its use has presently extended towards wireless, cellular and satellite environments. The satellite environment presents a series of challenges to TCP performance, underlining the large propagation delay and channel impairments. TCP was designed for cabled networks, where the loss of packets is due to congestion inside the network itself. In a satellite environment information losses sometimes happen due to congestion, but also due to the long end-to-end delay (latency) of the satellite link, and the high channel Bit Error Rate (BER) may also affect the data. Some proposals for the use of TCP in space include the use of low earth orbit (LEO) satellites, since they generate less losses and propagation delays, and variations of TCP have been proposed in order to improve its congestion performance. Consequently, there is interest to evaluate the performance of some TCP variants considering the effect of their congestion control algorithms in simulated LEO satellite scenarios. The objective is to establish a frame of reference for determining if it is possible to offer a Quality of Service (QoS) similar to that required for Voice over IP (VoIP) applications using TCP variants in a LEO satellite environment. In this work an analysis and modeling of TCP variants such as Tahoe, Reno, NewReno, SACK and Westwood is performed, evaluating particular QoS parameters such as latency, jitter, Bit Error Rate (BER), Packet Loss Rate (PLR) and Throughput. Based on the analysis and modeling presented on this work, information transfer simulations were performed through a LEO satellite using TCP variants with the ns2 (Network Simulator 2) network simulator. From the results obtained in this work, the possibility to offer a QoS similar to VoIP level can be established, with some limitations, when certain TCP variants are used over IP information networks using LEO satellites.

Keywords: TCP, Control congestion algorithm, TCP variants (Tahoe, Reno, NewReno, SACK, Westwood), Low Earth Orbit (LEO) satellites, QoS, VoIP.

DEDICATORIA

A mi Madre

María del Socorro

Y a mis Abuelos

José(†) y Socorro

Por su Fe en mí. Por sus enseñanzas. Por su Apoyo. Por su Amor.
GRACIAS.

A mis Tías

Isabel y Ana Laura

Por su apoyo diario y por hacerme sentir cerca de Tampico y de la Familia a pesar de la distancia.

Al resto de mi Familia

Familia Palomo Cantú (Rosa María, Víctor, Luisa y Ana)
Familia Dávila Cantú (Jorge y Cecilia)

De manera especial

a mi Tía Luz María y mis primos Alexandra y David

Por abrirme las puertas de su casa y recibirme en Navidad y Año Nuevo.

AGRADECIMIENTOS

Le doy Gracias a Dios por estar siempre conmigo y por ayudarme a lograr este objetivo que me propuse. El Señor es mi pastor nada me falta (Salmo 23).

Le agradezco a mi director de tesis y amigo *Dr. Roberto Conte Galván* quien no permitió que me dejara vencer por los obstáculos diarios y llevó a buen término este proyecto y también por las largas pláticas de música, política, religión, etc.

Al *Dr. David Hilario Covarrubias Rosales* por sus valiosas contribuciones a este trabajo de investigación.

A los miembros del comité de tesis: *Dr. José Antonio García Macías, M.C. Jorge Enrique Preciado Velasco*, *Dr. David Hilario Covarrubias Rosales y M.C. Raúl Tamayo Fernández*, por sus comentarios y observaciones realizados durante el desarrollo de este proyecto.

Al *M.C. Raúl Rivera Rodríguez* por el apoyo y amistad brindada.

A mis *compañeros y amigos de generación 2003-2005:* Ruth, Ely, Aby, Karen, Ararón, Ana, Darío, Ulises, Daniel, Néstor, Caro y Rocío. A mis amigos los *GUAMARAS (Adris, Gabis, Ivis, Rafis y Rogis)*, por el apoyo total, la amistad brindada, las carnes asadas y por no dejarnos vencer por los retos diarios. A todos un Abrazo y un Hasta Pronto.

A mis compañeros de casa Adán, Hugo y Arturo un reconocimiento sincero por la amistad y el apoyo brindado. Se volvieron mi Familia en Ensenada.

A mis amigos: Paúl, Luis, Cabanillas, Canek, Julio, César, Carlos, Marcial, Muraoka, Zeus, Juan Pedro. Don Víctor de la Cocina de la Abuela y Lupita por sus ricas comidas. Ah y al Bitoki: si homeeeeee.

A las secretarias de Electrónica Aurora, Rossy y Laura y a Lupita y Ceci de Biblioteca. Gracias por el apoyo y la sonrisa con la que siempre me recibieron.

Al *CICESE* por darme la oportunidad de estudiar esta maestría en particular al Departamento de Electrónica y Telecomunicaciones..

Al *Consejo Nacional de Ciencia y Tecnología (CONACYT)* por la beca otorgada y el apoyo para la realización de esta maestría..

CONTENIDO

Página

I. INTRODUCCION 1
I.1 Antecedentes 1
I.2 Planteamiento del problema 3
I.3 Objetivos de la Tesis 4
I.4 Infraestructura utilizada 4
I.5 Organización del documento de tesis 4
II. INTRODUCCION A TRANSMISION DE TCP POR SATELITE 6
II.1 Antecedentes 6
II.2 Internet por satélite (acceso remoto a Internet) 7
II.2.1 Descripción de TCP original (RFC 793) 7
II.2.2 Mejoras a TCP sobre enlaces de gran retardo (RFC 2488) 10
II.2.2.1 Control de congestión 10
II.2.2.1.1 Inicio Lento y Evasión de Congestión 10
II.2.2.1.2 Retransmisión Rápida y Recuperación Rápida 11
II.2.2.1.3 Ventanas grandes de TCP 13
II.2.2.1.4 Estrategias de reconocimiento 14
II.2.2.1.5 Reconocimientos selectivos 14
II.3 Consideraciones de TCP por satélite 14
II.3.1 Consideraciones de órbitas satelitales 15
II.3.2 TCP sobre satélites GEO 18
II.3.3 TCP sobre satélites LEO 19
II.3.4 Introducción a enrutamiento de TCP a bordo 20
II.3.5 Diseño de constelaciones satelitales 20
II.4 Parámetros de QoS para TCP por satélite 25
II.4.1 Latencia 26
II.4.2 Variación en el retardo (Jitter) 27
II.4.3 Tasa de Error de Bit (BER) 27
II.4.4 Tasa de Pérdida de Paquetes (PLR) 27
II.4.5 Caudal Eficaz 27
II.5 Conclusiones 27
III. ANALISIS DEL PROTOCOLO TCP 29
III.1 TCP puro (original) 29
III.2 Congestión en la transmisión de TCP 32
III.3 Mecanismo de Control de Flujo en TCP 36
III.3.1 Mecanismo de reconocimiento 38
III.3.2 Mecanismo de retransmisión 39
III.4 Soluciones propuestas para TCP por satélite 42
III.4.1 Soluciones de capas inferiores 43
III.4.2 Modificaciones de la IETF a TCP 44
III.4.3 Modificaciones a TCP no estandarizadas 44
III.4.3.1 Enfoque de conexión dividida 45
III.4.3.1.1 Soluciones de último salto 46

CONTENIDO (Continuación).

Página

III.4.4 Protocolos de transporte alternativos 46
III.4.4.1 Protocolo de Transporte – Especificaciones de Protocolo de Comunicaciones Espaciales 47
III.4.4.2 Protocolo de Transporte Satelital 48
III.4.4.3 TCP inalámbrico 48
III.4.4.4 Protocolo de Inspección y Espera 49
III.5 Variantes de TCP 50
III.5.1 TCP Tahoe 50
III.5.2 TCP Reno 51
III.5.3 TCP NewReno 52
III.5.4 TCP SACK 53
III.5.5 TCP Vegas 54
III.5.6 TCP Westwood 55
III.6 Conclusiones 56
IV. MODELADO DE VARIANTES DE TCP Y DE PARAMETROS DE QoS 57
IV.1 Modelado de TCP 57
IV.1.1 Modelo Periódico 59
IV.1.2 Modelo detallado de Pérdida de Paquete 62
IV.1.2.1 Expiraciones 66
IV.2 Modelos gráficos de variantes de TCP 69
IV.2.1 Variante TCP Tahoe 69
IV.2.2 Variante TCP Reno 70
IV.2.3 Variante TCP NewReno 71
IV.2.4 Variante TCP SACK 72
IV.2.5 Variante TCP Vegas 73
IV.2.6 Variante TCP Westwood 74
IV.3 Modelado de parámetros de Calidad de Servicio en Redes Satelitales de Órbita Baja 75
IV.3.1 Modelado de la latencia en constelaciones satelitales 75
IV.3.1.1 Retardo de transmisión 76
IV.3.1.2 Retardo de propagación 76
IV.3.1.2.1 Retardo de propagación para enlace ascendente y descendente 76
IV.3.1.2.2 Retardo de propagación de enlaces intersatelitales 77
IV.3.1.3 Retardo por procesamiento y conmutación 77
IV.3.1.4 Retardo del Protocolo de Control de Acceso al Medio 78
IV.3.2 Modelado de la variación en el retardo en constelaciones satelitales 79
IV.3.3 Modelado de la tasa de error de bit (BER) en constelaciones satelitales 79
IV.3.4 Modelado de la Tasa de Pérdida de Paquetes (PLR) 82
IV.3.5 Modelado del Caudal Eficaz 83
IV.4 Conclusiones 83
V. RESULTADOS DE SIMULACIÓN Y ANÁLISIS NUMÉRICO 84
V.1 Consideraciones de simulación 84

CONTENIDO (Continuación).

Página

V.2 Introducción al simulador de red Network Simulator 2 (ns2) 85
V.2.1 Parámetros de entrada en el simulador de red Network Simulator 2 (ns2) 86
V.3 Resultados de Simulación 88
V.3.1 Evolución de ventana de congestión y Producto Retardo-Ancho de Banda 88
V.3.1.1 Evaluación del desempeño de TCP mediante parámetros de Calidad de Servicio 98
V.3.1.1.1 Latencia 99
V.3.1.1.2 Variación en el retardo (Jitter) 103
V.3.1.1.3 Tasa de error de bit (BER) 107
V.3.1.1.4 Tasa de paquetes perdidos (PLR) y Caudal Eficaz 109
V.4 Conclusiones 112
VI. CONCLUSIONES 114
REFERENCIAS 117

Lista de Figuras

Página

Figura 1 Modelo conceptual de capas 8
Figura 2. Internet basado en satélite con la arquitectura de repetidor 17
Figura 3. Internet basado en satélite con arquitectura de OBP e ISL 17
Figura 4. Relación geométrica entre satélite y usuario 21
Figura 5. Consideraciones geométricas en un sistema satelital 21
Figura 6. Relaciones geométricas de los hexágonos inscritos en las huellas de cobertura .. 24
Figura 7. Formato del segmento TCP 30
Figura 8. Encapsulación de segmento TCP 33
Figura 9. Inicio Lento en TCP 34
Figura 10. Evasión de congestión 35
Figura 11. Ventana deslizante de TCP 37
Figura 12. Establecimiento de conexión usando saludo de tres vías 40
Figura 13. Terminación de una conexión TCP usando saludo de cuatro vías 41
Figura 14. Modelo periódico de las dinámicas de la ventana de TCP 60
Figura 15. Dinámica de la tasa de transmisión de TCP [Hassan y Jain, 2004] 60
Figura 16. Análisis detallado de los paquetes enviados durante 63
Figura 17. Intervalos que constituyen la dinámica de la ventana en un ciclo 67
Figura 18. Modelo gráfico del funcionamiento de TCP Tahoe 70
Figura 19. Modelo gráfico del funcionamiento de TCP Reno 71
Figura 20. Modelo gráfico del funcionamiento de TCP NewReno 72
Figura 21. Modelo gráfico del funcionamiento de TCP SACK 73
Figura 22. Modelo gráfico del funcionamiento de TCP Vegas 74
Figura 23. Modelo gráfico del funcionamiento de TCP Westwood 75
Figura 24. Evolución de ventana de congestión de TCP Reno con buffer de 20 paquetes .. 89
Figura 25. Evolución de ventana de congestión de TCP Reno [Lang, 2002] 90
Figura 26. Evolución de ventana de congestión de TCP Reno con buffer de 60 paquetes .. 91
Figura 27. Evolución de ventana de congestión de TCP Reno [Lang, 2002] 91
Figura 28. Evolución de ventana de congestión de TCP Reno con buffer de 40 paquetes .. 92
Figura 29. Evolución de ventana de congestión de TCP Reno [Lang, 2002] 93
Figura 30. Evolución de ventana de congestión de TCP Tahoe con buffer de 40 paquetes 94
Figura 31. Evolución de ventana de congestión de TCP Tahoe [Lang, 2002] 95
Figura 32. Evolución de ventana de congestión de TCP NewReno con buffer de 40 paquetes 95
Figura 33. Evolución de ventana de congestión de TCP NewReno [Lang, 2002] 96
Figura 34. Evolución de ventana de congestión de TCP SACK con buffer de 40 paquetes 97
Figura 35. Evolución de ventana de congestión de TCP SACK [Lang, 2002] 97
Figura 36. Evolución de ventana de congestión de TCP Westwood con buffer de 40 paquetes 98
Figura 37. Retardo del paquete entre estaciones terrenas utilizando TCP Tahoe 99
Figura 38. Retardo del paquete entre estaciones terrenas utilizando TCP Reno 100
Figura 39. Retardo del paquete entre estaciones terrenas utilizando TCP NewReno 101

Lista de Figuras (Continuación).

Página

Figura 40. Retardo del paquete entre estaciones terrenas utilizando TCP SACK 101
Figura 41. Retardo del paquete entre estaciones terrenas utilizando TCP Westwood..... 102
Figura 42. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP Tahoe..... 104
Figura 43. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP Reno 105
Figura 44. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP NewReno..... 105
Figura 45. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP SACK 106
Figura 46. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP Westwood..... 107
Figura 47. Variación de la Tasa de Error de Bit (BER) en el enlace entre Ensenada y México, D. F. con respecto a distintos ángulos de elevación 109

Lista de Tablas

Página

Tabla I. Banderas de TCP 31
Tabla II. Parámetros de entrada al simulador 88
Tabla III. Estadísticas obtenidas del retardo 103
Tabla IV. Estadísticas obtenidas de la variación en el retardo (Jitter) 108
Tabla V. Estadísticas de la Relación Señal Portadora a Ruido (C/N) y 110
Tabla VI. Estadísticas de Tasa de Paquetes Perdidos (PLR) y Caudal Eficaz obtenidas para las distintas variantes de TCP y diferentes tamaños de buffer 111

I. INTRODUCCION

I.1 Antecedentes

El gran éxito que ha tenido la Internet en los últimos años, ha llevado a la adopción del Protocolo de Internet (IP) para construir distintos tipos de redes de comunicación, entre las que se puede mencionar las redes privadas de grandes corporaciones de negocios, redes militares, caseras e inclusive celulares. El protocolo de transporte predominante para las redes de tipo IP es el protocolo TCP (Protocolo de Control de Transmisión). El protocolo TCP es un protocolo de transporte confiable que fue diseñado para redes cableadas, sin embargo su uso se ha ido extendiendo hacia enlaces inalámbricos, incluyendo redes celulares y satelitales.

El crecimiento de las redes IP ha sido tan grande en tamaño y uso que uno de los problemas que más se presenta es la congestión de información debido a las variaciones aleatorias de tráfico en segmentos de la Internet. Los efectos negativos que esto produce deben ser controlados para mantener niveles de Calidad de Servicio (QoS) adecuados de acuerdo a las aplicaciones de los usuarios que así lo demanden, como puede ser la transmisión de voz o video por Internet.

Las redes terrestres cableadas no han sido las únicas que han ayudado al crecimiento de Internet, ya que junto a ellas también han participado los enlaces inalámbricos, en particular los satelitales. En un principio los enlaces satelitales eran utilizados solo para comunicaciones militares, telefonía internacional y difusión de televisión, pero hoy en día también son utilizados para transportar tráfico TCP/IP y para ofrecer acceso a Internet.

Algunas de las razones por las cuales se destacan los sistemas satelitales son por su capacidad de cobertura global, su capacidad de difusión, su flexibilidad de ancho de banda en demanda y su habilidad para soportar movilidad de sus usuarios. Esto lo vuelve un candidato inmejorable para proveer servicios integrados de banda ancha a usuarios globalmente dispersos o a usuarios que cuentan con una infraestructura terrestre de comunicaciones pobre o inexistente. La evolución de Internet ha empujado el desarrollo de

infraestructura de red híbridas (red satelital/red terrestre), sin embargo el protocolo TCP al haber sido diseñado para redes terrestres cableadas ve impactado fuertemente su desempeño cuando es utilizado en ambientes inalámbricos, sobre todo del tipo satelital. Algunas de las características que lo afectan es el gran retardo de propagación, deterioro del canal y la asimetría del ancho de banda. Para tratar de resolver estos problemas se ha propuesto un gran número de mejoras, las cuales se pueden dividir en tres conjuntos:

a) Soluciones de capas inferiores del modelo OSI[1] para tratar de resolver los problemas de congestión pero sin alterar el TCP [Chaskar *et al*, 1999].
b) Modificaciones a TCP aceptadas y estandarizadas por la IETF (Internet Engineering Task Force, Fuerza de Trabajo de Ingeniería de Internet) [Jacobson *et al*, 1992] y [Mathis *et al*, 1996].
c) Soluciones experimentales propuestas que incluyen desde modificar TCP, hasta inclusive reemplazarlo con protocolos alternativos de transporte [Ghani y Dixit, 1999] y [Barakat *et al*, 2000].

De lo anteriormente expuesto se establece como interés de éste trabajo de tesis el evaluar el desempeño del protocolo TCP y sus variantes en redes satelitales de órbita baja contra los parámetros de Calidad de Servicio[2] (Quality of Service, QoS) usados como referencia para ofrecer servicio de VoIP (Voz sobre Protocolo de Internet) los cuales son retardo, variación en el retardo (jitter), tasa de error de bit (BET), tasa de paquetes perdidos (PLR) y caudal eficaz. El razonamiento detrás de comparar el desempeño de TCP con los parámetros de QoS en VoIP es el que éstos son los más estrictos en lo que a comunicación en Internet se refiere, dando una referencia para establecer qué tan factible sea el ofrecer QoS al utilizar alguna de las variantes del protocolo TCP en un ambiente satelital.

---

[1] El modelo OSI (Interconexión de Sistemas Abiertos) fue propuesto por la ISO (Organización Internacional de Normas) y es un estandar para comunicación de datos que permite que sistemas computacionales de diferentes fabricantes se comuniquen.

[2] La Recomendación E.800 de la ITU define la Calidad de Servicio como el efecto colectivo del desempeño de servicio el cual determina el grado de satisfacción de un usuario del servicio.

I.2 Planteamiento del problema

TCP provee un servicio de flujo de datos confiable (con una entrega en orden garantizada extremo a extremo) a las aplicaciones. Un transmisor de TCP acepta datos de una aplicación en tamaños variados y los empaqueta en segmentos de longitud variable para transmitirlos dentro de los datagramas IP, donde cada uno se identifica con un número de secuencia. El receptor de TCP responde a la recepción exitosa de datos regresando un reconocimiento al transmisor y entregando los datos a la aplicación receptora; el transmisor TCP puede utilizar estos reconocimientos para determinar si algún dato requiere retransmisión.

Las redes redes satelitales se han vuelto una alternativa para aquellos lugares con una infraestructura de comunicaciones pobre o nula, con capacidad para cubrir áreas geográficas amplias y una conectividad inmediata. Mucho del tráfico que fluye por sus canales tiene a TCP como protocolo de transporte, imponiendo una serie de retos a enfrentar para su buen desempeño, ya que TCP fue diseñado para redes terrestres cableadas donde las pérdidas de paquetes son consecuencia de la congestión en la red misma. Aunque algunas veces las pérdidas en el ambiente satelital son debidas a congestión, muchas otras son debidas a que la información ha sido afectada por la alta tasa de error de bits del canal o a la alta latencia extremo a extremo. Como TCP no distingue fácilmente la causa de pérdida de paquetes asume que son debidas a congestión y reacciona activando sus algoritmos de control de congestión (inicio lento, evasión de congestión).

Recientemente los sistemas satelitales de órbita baja (LEO, Low Earth Orbit) se han propuesto como una opción para los servicios o aplicaciones que requieran una alta Calidad de Servicio, debido a los tiempos cortos de respuesta y retardo que ofrece [Hassan y Jain, 2004]. De aquí surgen los objetivos de ésta tesis que son evaluar el desempeño del protocolo TCP y sus variantes en satélites de órbita baja y comparar los resultados con los parámetros de QoS de VoIP que son retardo, variación en el retardo, BER, PLR y caudal eficaz.

I.3 Objetivos de la Tesis

- El *objetivo general* de este trabajo es el análisis de mecanismos de control de congestión en redes IP mediante distintas variantes de TCP para transmisión de información en redes de satélites de órbita baja y evaluar su desempeño comparando con los parámetros de QoS.
- El *objetivo particular* de éste trabajo de tesis es el modelado, diseño y simulación de enlaces satelitales TCP/IP para comparar su nivel de desempeño e impacto en parámetros de QoS tales como retardo, variación en el retardo, tasa de errores de bit, tasa de paquetes perdidos y caudal eficaz de la red. Se modelarán los algoritmos de distintos mecanismos de control de congestión en TCP tal como Tahoe, Reno, NewReno, SACK y Westwood.
- El *objetivo específico* de éste trabajo de tesis es el desarrollo de programas de computadora para simular la transmisión de aplicaciones TCP/IP sobre enlaces de satélites de órbita baja comparando su nivel de desempeño e impacto en parámetros de QoS.

I.4 Infraestructura utilizada

- Computadora personal portátil AMD Athlon XP2000+ 1.66 GHz con 704 MB de RAM.
- Aplicaciones de computadora como: Microsoft Word, Microsoft PowerPoint, Microsoft Visio Professional, Mathworks Matlab, SAVI y Network Simulator 2 (ns2).
- Material de biblioteca (CICESE y UCSD).
- Artículos, libros y revistas del área.
- Acceso a Internet.

I.5 Organización del documento de tesis

El presente documento se encuentra organizado de la siguiente manera:

El capítulo II presenta un breve análisis acerca del acceso a Internet mediante redes satelitales así como también del protocolo de transporte TCP y las modificaciones propuestas para su uso en un ambiente satelital. También se ofrece una descripción de los parámetros de Calidad de Servicio con los que se va a medir el desempeño de TCP.

En el capítulo III se analiza la operación del protocolo TCP, los mecanismos internos que lo componen y las variantes desarrolladas de su algoritmo de control de congestión.

El capítulo IV presenta un modelo matemático de TCP, los modelos analíticos de sus variantes más representativas y también se desarrollan modelos matemáticos de parámetros de Calidad de Servicio para evaluar el desempeño de TCP en un ambiente satelital.

El capítulo V presenta el escenario de simulación, consideraciones, presentación de resultados y análisis numérico propuestos para validar este trabajo de tesis.

Por último el capítulo VI presenta las conclusiones y aportaciones de este trabajo, así como una serie de ideas y recomendaciones para trabajo futuro.

II. INTRODUCCION A TRANSMISION DE TCP POR SATELITE

Se presenta un breve análisis acerca del acceso a Internet mediante redes satelitales así como del protocolo de transporte más utilizado por éste tipo de tráfico, que es el TCP. Se tratan las características de éste protocolo en su diseño original y las modificaciones que se han propuesto para su mejor desempeño en un ambiente satelital, finalizando con la descripción de los parámetros de Calidad de Servicio (QoS, Quality of Service) con los cuales se comparará el desempeño de TCP.

II.1 Antecedentes

Debido a la inherente capacidad de difusión de los satélites, los cuales pueden conectar sitios remotos cuando no existe infraestructura terrestre ofreciendo al mismo tiempo canales de relativa alta velocidad, existe un interés creciente en proveer interconexión y servicios multimedia utilizando enlaces satelitales [Marchese, 2001]. El uso de la pila TCP/IP en ambientes satelitales es realmente importante ya que muchas aplicaciones están basadas en ésta familia de protocolos y cualquier conexión con la Internet del futuro implicará el uso de TCP/IP.

La red utilizada afecta fuertemente el comportamiento del protocolo de transporte. Aún en un ambiente satelital, los problemas son distintos si se usa un sistema satelital LEO, MEO o GEO. Aunque muchos factores específicos a las comunicaciones satelitales afectan a TCP produciendo un desempeño bajo, el tráfico de tipo TCP/IP cada vez se incrementa más en enlaces satelitales. En éste capítulo se abordará al protocolo TCP desde su nivel de diseño original, para luego pasar a las modificaciones que se han ido proponiendo para mejorar su desempeño en redes satelitales GEO y LEO, introduciendo los parámetros de QoS a utilizar para evaluar su desempeño.

II.2 Internet por satélite (acceso remoto a Internet)

En un sistema de Internet basado en satélite, los satélites interconectan segmentos de red heterogéneos, proporcionando acceso directo a Internet desde hogares y negocios. La Internet ha gozado de un explosivo crecimiento en los pasados años. Al mismo tiempo, la proliferación de nuevas aplicaciones y la expansión en el número de computadoras y usuarios conectados ha impuesto nuevos retos técnicos a su desarrollo. Se requieren nuevas tecnologías e infraestructura capaces de proveer servicios de alta velocidad y calidad para acomodar aplicaciones multimedia con diversos requerimientos de QoS. Más aún, para poder proveer acceso a Internet desde cualquier lugar, se requiere un soporte de movilidad apropiado. Un sistema de comunicación satelital se distingue por su cobertura global, capacidad de difusión inherente, flexibilidad de ancho de banda en demanda y la habilidad de soportar movilidad, por lo que se vuelve un candidato excelente para proveer servicios integrados de banda ancha de Internet a usuarios globalmente dispersos [Hu y Li, 2001].

II.2.1 Descripción de TCP original (RFC 793)

El estándar del Protocolo de Control de Transmisión (TCP, Transmission Control Protocol) está definido por el RFC (Request for Comments, Petición de Comentarios) 793 de la IETF (Internet Enginnering Task Force, Fuerza de Trabajo de Internet). La especificación original escrita en 1981 se basó en investigaciones y experimentaciones en la ARPANET [Postel, 1981]. TCP basa su diseño y funcionamiento en conceptos descritos en el trabajo de Cerf y Kahn publicado en 1974. El TCP se ubica dentro de una arquitectura de protocolo de capas, justo encima del Protocolo de Internet el cual provee la manera para que TCP envie y reciba segmentos de longitud variable de información adjunta en datagramas de Internet. El datagrama de Internet provee los medios para el direccionamiento desde y hacia la fuente y destino de los TCPs en diferentes redes.

El protocolo de Internet también se ocupa de cualquier fragmentación y reensamble de segmentos TCP requerida para lograr transporte y entrega a través de múltiples redes. El protocolo de Internet también lleva información de la prioridad, clasificación de seguridad y compartición de los segmentos TCP, con lo cual ésta información puede ser comunicada

extremo a extremo a través de múltiples redes. A continuación se presenta en la Figura 1 el modelo conceptual de capas de TCP/IP.

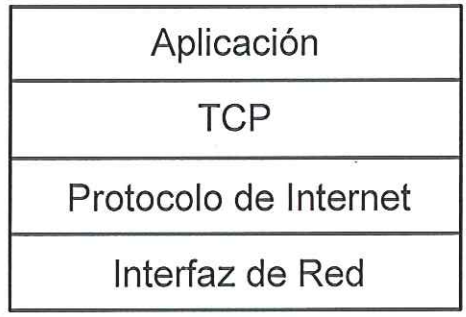


Figura 1. Modelo conceptual de capas

El propósito primario de TCP es el proveer servicio de conexión o circuito lógico confiable entre procesos pares. Para poder hacer esto encima de un sistema de comunicación de Internet menos confiable se requiere de ciertos mecanismos en las siguientes áreas: Transferencia de datos básica, Confiabilidad, Control de flujo, Multiplexaje, Conexiones y Prioridad y Seguridad [Postel, 1981].

Transferencia de datos básica. Se refiere a la capacidad de TCP de transferir un flujo continuo de octetos en cada dirección entre sus usuarios, al empaquetar un número de octetos en segmentos para su transmisión a través de Internet.

Confiabilidad. El TCP debe recuperar datos que puedan haberse dañado, perdido, duplicado o entregado fuera de orden por el sistema de comunicación de Internet. Esto se logra asignando un número de secuencia a cada octeto transmitido, y requiriendo un reconocimiento positivo (ACK) del receptor de TCP. Si el ACK no es recibido dentro de un intervalo de expiración, los datos se retransmiten. En el receptor, los números de secuencia se usan para ordenar correctamente segmentos que pudiesen haber sido recibidos fuera de orden y para eliminar duplicados. El daño es manejado agregando un bit de paridad a cada segmento transmitido, checando en el receptor, y descartando segmentos dañados.

Control de flujo. TCP provee medios al receptor para regular la cantidad de datos enviados por el transmisor. Esto se logra regresando una "ventana" con cada ACK indicando un intervalo aceptable de números de secuencia más allá del último segmento

recibido exitosamente. La ventana indica un número permitido de octetos que el transmisor puede enviar antes de obtener permiso adicional de poder enviar más.

Multiplexaje. Para permitir que muchos procesos dentro de una computadora utilicen simultáneamente los mecanismos de comunicación de TCP, éste provee un conjunto de direcciones o puertos dentro de cada host[3]. Las direcciones de red y computadora de la capa de comunicación de Internet forman un socket[4]. Un par de sockets únicamente identifica cada conexión, con lo cual un socket puede ser utilizado simultáneamente en múltiples conexiones. La vinculación de puertos con procesos es manejada de manera independiente por cada computadora. Sin embargo, es útil unir procesos usados frecuentemente a sockets fijos, los cuales se dan a conocer al público. Estos servicios pueden entonces ser accesados a través de las direcciones conocidas.

Conexiones. Los mecanismos de confiablidad y control de flujo descritos anteriormente requieren que TCP inicie y mantenga cierta información de estado para cada flujo de datos. La combinación de ésta información, incluyendo sockets, números de secuencia y tamaños de ventana, es llamada *conexión*. Cada conexión es especificada únicamente por un par de sockets identificando sus dos lados. Cuando dos procesos desean comunicarse, sus TCPs deben primero establecer una conexión. Cuando se ha completado la comunicación, la conexión es terminada o cerrada para liberar los recursos y que otros los puedan usar. Ya que las conexiones deben ser establecidas entre computadoras poco fiables y sobre un sistema de comunicación de poca confianza como es Internet, se usa un mecanismo de saludo o "handshake" con números de secuencia basados en tiempo para evitar inicializaciones erróneas de comunicación.

Prioridad y Seguridad. Los usuarios de TCP pueden indicar la seguridad y prioridad de su comunicación dentro de su encabezado. Existen valores por omisión cuando éstas características no son necesarias.

---

[3] Una computadora. De manera particular una fuente o destino de mensajes desde el punto de vista de la red de comunicación.

[4] Una dirección que de manera específica incluye un identificador de puerto que es la unión de la Dirección de Internet con un puerto TCP.

II.2.2 Mejoras a TCP sobre enlaces de gran retardo (RFC 2488)

Esta sección puntualiza los mecanismos de TCP que pueden ser necesarios en redes satelitales o redes híbridas satelitales/terrestres para utilizar de mejor manera la capacidad disponible del enlace de datos.

II.2.2.1 Control de congestión

Para evitar generar una cantidad inapropiada de tráfico, durante una conexión TCP emplea cuatro mecanismos de control de congestión. Estos algoritmos son *inicio lento*, *evasión de congestión*, *retransmisión rápida* y *recuperación rápida*, y se utilizan para ajustar la cantidad de datos sin reconocer que pueden ser inyectados en la red y para retransmitir segmentos descartados por la red.

El transmisor de TCP usa dos variables para realizar éste control de congestión. La primera es la ventana de congestión (cwnd), y representa un límite superior en la cantidad de datos que el transmisor puede inyectar en la red antes de recibir un reconocimiento (ACK). El valor de cwnd está limitado a la ventana anunciada por el receptor, y aumenta o disminuye durante la transferencia de acuerdo a la cantidad de congestión que deduce existe en la red.

La segunda variable es el umbral de inicio lento (ssthresh) y determina cuál algoritmo se usará para incrementar el valor de cwnd. Si cwnd es menor que ssthresh se usa el algoritmo de inicio lento para incrementar el valor de cwnd; si cwnd es mayor o igual a ssthresh se usa el algoritmo de evasión de congestión. El valor inicial de ssthresh es el tamaño de la ventana anunciada del receptor.

II.2.2.1.1 Inicio Lento y Evasión de Congestión

Cuando una computadora comienza a enviar datos sobre una conexión TCP, no tiene conocimiento del estado actual de la red que está entre ella y el receptor, y para evitar transmitir una ráfaga inapropiadamente grande de datos, el transmisor usa el algoritmo de inicio lento al comienzo de la transferencia. El inicio lento comienza inicializando cwnd a un segmento y ssthresh a la ventana anunciada del receptor. Esto forza a TCP a transmitir un segmento y esperar por el correspondiente ACK. Por cada ACK que se recibe durante

inicio lento, el valor de cwnd se incrementa en un segmento, con lo que tiene un crecimiento exponencial y continuará de esa manera hasta que cwnd llegue o exceda ssthresh, o detecte una pérdida de información.

Cuando el valor de cwnd es igual o mayor a ssthresh se usa el algoritmo de evasión de congestión para incrementar a cwnd. Este algoritmo incrementa el tamaño de cwnd de manera más lenta en comparación a como lo hace inicio lento, ya que evasión de congestión se usa para sondear la red en busca de capacidad adicional. Durante evasión de congestión, cwnd se incrementa 1/cwnd por cada ACK que llega, de ahí que si se recibe un ACK por cada segmento de datos, cwnd se va a incrementar aproximadamente un segmento por Tiempo de Viaje Completo (RTT, Round Trip Time). Debido a su funcionamiento ambos algoritmos pueden hacer que se presente una utilización pobre del ancho de banda del canal disponible cuando se usan redes satelitales de gran retardo. Esto se puede comprobar si se utiliza un satelite GEO, donde al transmitir un segmento se tiene que esperar por su correspondiente ACK, y durante aproximadamente 500 ms no se realiza transmisión de datos, de ahí que el inicio lento tome mas tiempo en un GEO que en un canal terrestre [Allman, 1999].

II.2.2.1.2 Retransmisión Rápida y Recuperación Rápida

El mecanismo por omisión que utiliza TCP para detectar pérdida de segmentos es la expiración de tiempo, lo que quiere decir que si el transmisor no recibe un ACK por un paquete dado dentro del tiempo esperado, el segmento va a ser retransmitido. La expiración de la retransmisión (RTO, Retransmission Timeout) se basa en observaciones del RTT. Aparte de detectar una pérdida por medio de la expiración de RTO, TCP también utiliza el segmento perdido como un indicador de congestión en la red En respuesta a ésta congestión, el valor de ssthresh se fija a la mitad del cwnd y el valor de cwnd es reducido a un segmento. Esto hace que el algoritmo de inicio lento incremente cwnd hasta que el valor de cwnd alcance la mitad de su valor (que fue cuando se detectó la congestión). Después de la fase de inicio lento, el algoritmo de evasión de congestión es utilizado para sondear la red en búsqueda de capacidad adicional.

TCP siempre reconoce el segmento en orden más alto que ha llegado, de ahí que un ACK por el segmento X también reconozca todos los segmentos menores a X. Si llega un segmento fuera de orden, el ACK que se va a enviar será por el segmento más alto en orden en lugar del segmento que haya llegado.

El algoritmo de retransmisión rápida utiliza estos reconocimientos duplicados para detectar segmentos perdidos. Si tres reconocimientos duplicados llegan al transmisor de datos, TCP asume que un segmento se ha perdido y retransmite el segmento perdido sin esperar a que expire el RTO. Después de que un segmento sea reenviado utilizando retransmisión rápida, se usa el algoritmo de recuperación rápida para ajustar la ventana de congestión de la siguiente manera: en primera instancia el valor de ssthresh se fija a la mitad del valor de cwnd y en seguida el valor de cwnd se reduce a la mitad. Después de eso el valor de cwnd se incrementa artificialmente en un segmento por cada reconocimiento duplicado que haya llegado. Este incremento artificial puede ser realizado debido a que cada ACK representa un segmento que ha dejado la red. Cuando el cwnd lo permite, TCP es capaz de transmitir nuevos datos.

El funcionamiento antes descrito permite que TCP mantenga el flujo de los datos a través de la red a la mitad de la tasa de cuando se detectó la pérdida. Cuando llega un ACK por el paquete retransmitido, el valor de cwnd es reducido de regreso a ssthresh, que es la mitad del valor de cwnd cuando la congestión fue detectada.

Generalmente la retransmisión rápida puede reenviar solo un segmento por ventana de datos enviados, y cuando múltiples segmentos se pierden en una ventana de datos dada, uno de los segmentos será reenviado utilizando retransmisión rápida. El resto de los segmentos descartados usualmente esperan por el tiempo de expiración (RTO), lo que causa que TCP regrese a inicio lento. La manera en que TCP responde ante la detección de congestión varía dependiendo de la manera en que se haya percatado de ella. Si el temporizador de retransmisión causa que un paquete sea reenviado, TCP reduce ssthresh a la mitad del valor actual de cwnd y cwnd es reducido a un segmento (volviendo a inicio lento), pero si un segmento es reenviado por medio de retransmisión rápida, ambos ssthresh y cwnd son fijados a la mitad del valor actual de cwnd y se utiliza evación de congestión para enviar nuevos datos.

La diferencia entre las dos opciones anteriores es que cuando la retransmisión es debida a reconocimientos duplicados, TCP sabe que los paquetes aun están fluyendo a través de la red y de esa manera puede darse una idea de que la congestión aún no es tan grave. Con respecto al otro caso, cuando se reenvía un paquete debido a que expiró el temporizador de retransmisión, TCP no puede inferir nada acerca del estado de la red y por eso debe proceder de manera conservadora enviando nuevos datos por medio del algoritmo de inicio lento.

II.2.2.1.3 Ventanas grandes de TCP

El estándar máximo de tamaño de ventana de TCP (65,535 bytes) no es adecuado para permitir que una conexión sencilla de TCP utilize todo el ancho de banda disponible en los canales satelitales. El caudal de TCP está limitado por la ecuación (1) [Postel, 1981]:

$$CE = TdV / RTT \qquad (1)$$

Donde CE es el Caudal Eficaz, TdV es el Tamaño de Ventana y RTT es el Tiempo de Viaje Completo. De la ecuación (1), si se utiliza un tamaño de ventana máximo de 65,535 bytes y un canal satelital geosíncrono con un RTT de 560 ms, el máximo caudal eficaz está limitado por la ecuación (2):

$$CE = 65{,}535 \text{ bytes} / 560 \text{ ms} = 117{,}027 \text{ bytes} / \text{segundo} \qquad (2)$$

y comparando el resultado de la ecuación (2) con un canal satelital GEO de capacidad T1 de 1.536 Mbits/segundo (192,000 bytes /segundo) se determina que una conexión sencilla de TCP no logra utilizarlo de manera completa. Por esto TCP ha sido modificado para poder soportar grandes ventanas [Jacobson *et al.*, 1992], y aconseja usar las opciones de escalamiento de ventana en el ambiente satelital. Al utilizar estas adecuaciones se requiere que la pila de TCP sea sintonizada manualmente tanto en la aplicación del cliente como del servidor para poder hacer uso de esta extensión.

II.2.2.1.4 Estrategias de reconocimiento

Existen dos métodos estándar que pueden ser usados por los receptores de TCP para generar reconocimientos, uno lo propuso Postel en [Postel, 1981] y el otro lo propuso Braden en [Braden, 1989]. Postel expone un método para generar un ACK por cada paquete que llegue, mientras que Braden establece la alternativa de que las computadoras usen reconocimientos retardados. Utilizando la propuesta de Braden, un ACK se genera cada segundo segmento y la ventana de congestión se incrementa basándose en el número de ACKs entrantes y ACKs retardados, reduciendo el número de ACKs que envía el receptor. De ahí que el crecimiento de cwnd ocurre de manera mucho más lenta cuando se utilizan ACKs retardados, comparado con el caso cuando el receptor reconoce cada paquete entrante. Sin embargo, si no se desea usar este mecanismo de reconocimiento retardado simplemente se deshabilita esta opción.

II.2.2.1.5 Reconocimientos selectivos

Los reconocimientos selectivos (SACK) permiten a los receptores de TCP informar a los transmisores de manera exacta cuáles paquetes han llegado, y de acuerdo a eso posibilitan al TCP que se recupere más rápidamente de segmentos perdidos, evitando retransmisiones innecesarias [Allman 1999].

II.3 Consideraciones de TCP por satélite

El Protocolo de Control de Transmisión (TCP) es una parte integral de muchas aplicaciones populares de Internet como correo electrónico, transferencia de archivos y navegación en Internet. Históricamente el desempeño de TCP sobre canales satelitales ha sido subóptimo debido a cuestiones de algoritmo de protocolo y configuración [Hassan y Jain, 2004]. A continuación se dará revisión a los retos que se enfrenta TCP en ambientes particulares como son órbitas GEO y LEO.

II.3.1 Consideraciones de órbitas satelitales

Antes de pasar a las cuestiones de diseño y desempeño se dará un repaso de las características de las redes satelitales. Las principales ventajas de las comunicaciones satelitales son:

- *Cobertura ubicua*: Un sistema satelital sencillo puede llegar a cada usuario potencial en un continente entero sin importar su localidad, en particular áreas con baja densidad de suscriptores y/o áreas difíciles de cubrir o imposibles de alcanzar.
- *Flexibilidad de ancho de banda*: Un sistema satelital puede proveer distintas configuraciones de ancho de banda de acuerdo a las necesidades del usuario. *Conectividad*: Las redes satelitales proveen comunicaciones punto a multipunto y multipunto a multipunto facilitadas por la Internet y la capacidad de difusión.
- *Recuperación ante desastres*: Representan una alternativa a las redes de fibra óptica dañadas y son una opción para comunicaciones de emergencia.
- *Despliegue y servicio*: Pueden comenzar a dar servicio a un continente entero al estar ubicados en el espacio, y se cuenta con la ventaja de que nuevos usuarios pueden ser agregados fácil y rápidamente.
- *Confiabilidad y seguridad*: Los satélites se encuentran entre las tecnologías de comunicación más confiables y seguras.
- *Costo*: Es independiente de la distancia, ya que con la amplia área de cobertura que tiene, cuesta lo mismo recibir el servicio en los extremos que en el centro del área cubierta.

Ya vistas las ventajas de un sistema satelital, se pasa a dar revisión a como está compuesto. Un sistema satelital de comunicaciones se compone de un segmento espacial y de un segmento terrestre. El segmento terrestre de Estaciones de Interconexión, o Gateway (EG), Centro de Control de Red (CCR) y Centros de Control de Operación (CCO) [Hu y Li, 2001]. Los CCR y CCO llevan a cabo el manejo de recursos de red, operación del satélite y control de órbita, mientras que las EG actúan como interfaces de red entre varias redes externas y la red satelital, realizando conversiones de protocolo, direccionamiento y formato. El segmento espacial se compone de satélites de órbita geoestacionaria (SOG) y

satélites de órbita no geoestacionaria (SONG), los cuales se dividen en satélites de órbita media (MEO) y de órbita baja (LEO).

Los SOG se encuentran a 35,786 km sobre el ecuador, logrando que su período de rotación sea igual al de la Tierra cubriendo un tercio de su superficie. Dentro de las desventajas que presentan está su elevado costo de lanzamiento. Su gran altitud y su alta degradación de señal requiere grandes antenas y potencia de transmisión tanto en el satélite como en las estaciones terrenas, y su retardo de propagación va de los 250 a los 280 ms. Los satélites MEO tienen una altitud de 10,000 a 15,000 km y su retardo de propagación típico está entre los 110 y 130 ms de ida y vuelta, mientras que los LEO se localizan entre los 700 y 2,000 km sobre la superficie de la Tierra y su retardo de propagación de una vía está entre 5 y 30 ms [Hassan y Jain, 2004].

La carga útil del satélite es la responsable de las funciones de comunicación. Los satélites tradicionales, especialmente GEOs, sirven como repetidores entre dos puntos en la Tierra. Algunos permiten Procesamiento A Bordo (OBP) lo cual incluye demodulación y remodulación, decodificación y recodificación, conmutación de haz y transponder y enrutamiento para proveer una utilización del canal más eficiente. También pueden establecer enlaces intersatelitales de alta capacidad entre dos satélites con línea de vista. Las bandas de frecuencia que se utilizan para las comunicaciones satelitales son la C (4-8 GHz), Ku (10-18 GHz) y Ka (18-31 GHz). La mayoría de los satélites de difusión directa usan banda Ku tanto para difusión como para conexiones de Internet desde el servidor a los usuarios, con un enlace de retorno terrestre.

El Internet basado en satélite tiene varias opciones de arquitectura debido a los diversos diseños de los sistemas satelitales en tipos de órbita (GEO, MEO, LEO), tipo de carga útil (OBP y repetidor) y diseños de enlaces intersatelitales (ISL). Una red satelital puede servir como parte de una dorsal de Internet, como una red de acceso de alta velocidad o ambos. Se pueden destacar dos escenarios, uno con satélites repetidores solamente como el de la Figura 2, y otro que puede implementarse con OBP e ISLs para construir una red en el cielo como en la Figura 3.

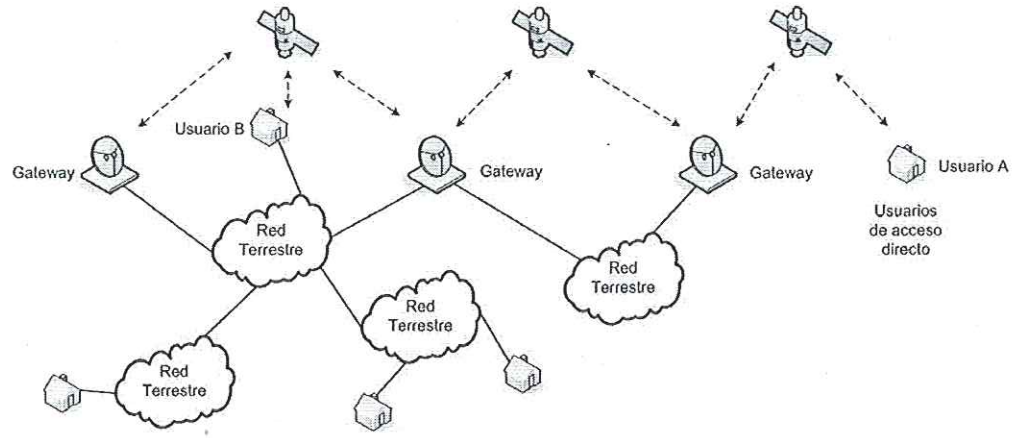


Figura 2. Internet basado en satélite con la arquitectura de repetidor

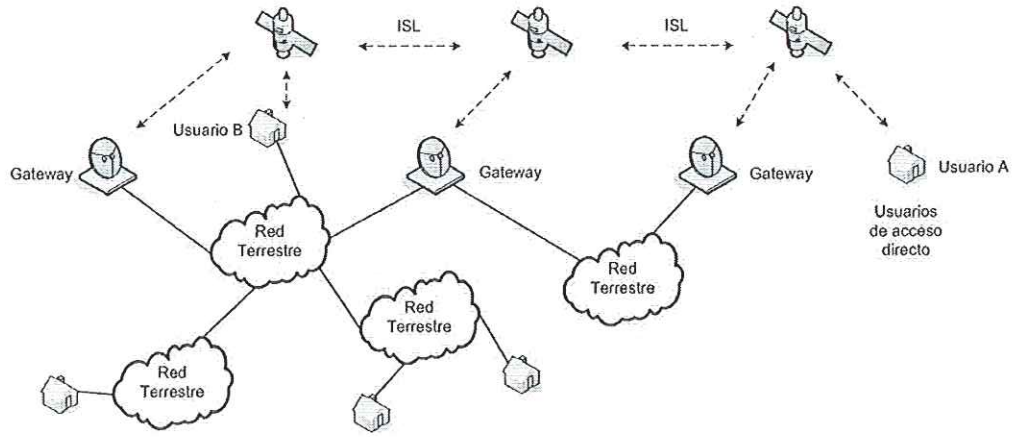


Figura 3. Internet basado en satélite con arquitectura de OBP e ISL

La arquitectura de red de la Figura 2 puede proveer acceso a Internet así como servicio de troncales de datos. La red satelital se conecta con la infraestructura de red terrestre por medio de estaciones Gateway en la Tierra. Puede ser el único método de acceso para algunos usuarios, como el usuario A, o un respaldo para una red de acceso terrestre existente, como el usuario B. Las desventajas que presenta son baja eficiencia espectral y altos retardos debidos a la falta de trayectorias de comunicación directas en el espacio. Por otro lado, el escenario propuesto en la Figura 3 permite construir una red en el cielo, donde la gran conectividad que presenta puede proveer mayor flexibilidad pero también puede traer cuestiones de enrutamiento complejas.

Dentro de los servicios que pueden prestar los sistemas satelitales se encuentran los siguientes:

a) Servicios Satelitales Fijos: Son los que se refieren a los servicios de comunicación entre estaciones terrenas.
b) Servicios Satelitales Móviles: Comprenden todas las comunicaciones entre una estación terrena móvil y uno o más satélites.
c) Servicios Satelitales de Difusión: Es el servicio de comunicación en el cual las señales del satélite son recibidas directamente por el público en general.

II.3.2 TCP sobre satélites GEO

Los satélites GEO tienen una órbita ecuatorial circular (este-oeste) a 36,000 km encima del ecuador, y rotan a la misma velocidad angular que la Tierra, tomando aproximadamente 24 horas en realizar una órbita circular [Hassan y Atiquzzaman, 2000]. Un aspecto importante de ésta velocidad angular es que desde una estación terrena el satélite parece estar fijo en el cielo. Debido a su gran altitud, cada satélite GEO puede cubrir un área amplia de Tierra y una cobertura global puede ser lograda con sólo tres satélites. Una de las ventajas de utilizar éste tipo de satélites para comunicación es la conexión ininterrumpida entre la estación terrena y el satélite por 24 horas al día, siete días a la semana. Otra ventaja es su amplia cobertura de la superficie de la Tierra ya que estaciones terrenas bajo la cobertura del mismo satélite GEO se pueden comunicar sin la necesidad de retransmitir hacia otros satélites en el cielo.

La principal desventaja de los GEO es el gran retardo de propagación debido a su altitud, el cual típicamente está alrededor de 270 ms, de ahí que el RTT para un reconocimiento de TCP sea de alrededor de 540 ms sobre un enlace GEO. Otra desventaja es el gran Producto Retardo Ancho de Banda (PRAB), que se define como la cantidad de datos que están "en vuelo" en el canal de transmisión (datos que han sido transmitidos pero todavía no son reconocidos). El retardo usado en éste caso es el RTT (que para satélites GEO es grande) y el ancho de banda es la capacidad del enlace cuello de botella en la trayectoria. El tener un PRAB alto afecta directamente el sistema de control de flujo de ventana y el tamaño de memoria temporal (buffer) de TCP. Con un tamaño de buffer de

65,536 bytes, lo cual directamente indica el máximo tamaño de ventana de transmisión, la utilización del enlace resulta en un desempeño pobre de TCP. Para compensar esto se puede aumentar el tamaño de ventana incrementando la cantidad de segmentos transmitidos dentro de un RTT sencillo, pero se incrementa la posibilidad de múltiples segmentos perdidos y de igual manera se afecta el desempeño de TCP. En las redes terrestres la mayoría de los segmentos perdidos es causada por congestión de red, de ahí el sobreflujo en el buffer. En un ambiente satelital esto cambia ya que el BER es alto causando pérdida de segmentos TCP, lo que trae en consecuencia que el TCP esté volviéndose continuamente a la fase de inicio lento.

II.3.3 TCP sobre satélites LEO

Las órbitas LEO suelen ser órbitas inclinadas con respecto al ecuador o polares (norte-sur) y se encuentran en el intervalo de altitud de 700 a 2000 km. A pesar de que por la altitud logran reducir el retardo de propagación, esto mismo ocasiona que se incremente la variación en el retardo o jitter, por las siguientes razones [Hassan y Atiquzzaman, 2000] :

- *Traspasos*: Debido a su baja altitud, para lograr una cobertura global se requieren múltiples satélites por órbita para que cuando un satélite se aleje de la línea de vista en el horizonte, otro satélite del mismo plano aparezca en el horizonte opuesto. Esta continuidad requiere un traspaso de la conexión del satélite que va desapareciendo hacia el que va ascendiendo, y éste movimiento es lo que puede incrementar la variación del retardo.
- *Reenrutamiento*: Los satélites LEO no sólo se mueven con respecto a las estaciones terrenas, sino que también cambian ubicación de manera relativa a otros satélites en otras órbitas. De ahí que cuando algunos satélites de múltiples órbitas toman parte en una conexión, puede ser necesario realizar reenrutamiento dinámico de los paquetes para mantener optimizada la ruta. El reenrutamiento resulta en profundos cambios a los retardos.
- *Reordenamiento de paquetes*: El reenrutamiento dinámico puede causar que los paquetes aparezcan fuera de orden en el destino. Estos paquetes fuera de orden

requieren ser almacenados en buffers en el extremo de la red hasta que el ordenamiento se complete, conduciendo a una variación en el retardo.

II.3.4 Introducción a enrutamiento de TCP a bordo

Se han diseñado algunas extensiones a TCP para resolver algunas de las limitaciones del TCP estándar sobre enlaces satelitales. Una vertiente que han tomado para tratar de aliviar los efectos del gran retardo extremo a extremo es dividir la conexión TCP en dos o más partes en las estaciones gateway que conectan las redes satelital y terrestre. Existen tres enfoques para dividir las conexiones TCP sobre enlaces satelitales [Hu y Li, 2001]:

- *Engaño a TCP (Spoofing)*: Las conexiones divididas en segmento terrestre y segmento inalámbrico son aisladas por las estaciones gateway, las cuales prematuramente envían reconocimientos de engaño al recibir paquetes. Las estaciones gateway en los puntos de división son también responsables de la retransmisión de cualquier dato perdido.
- *División de TCP (Tunneling)*: En lugar de engañar a TCP, la conexión se divide completamente. Un protocolo de transporte propietario puede ser usado en una red satelital sin interferir con el TCP estándar en la red terrestre. Es más flexible, aunque algún tipo de convertidor de protocolo debe ser implementado en los puntos de división.
- *Almacenamiento de Web (Web caching)*: La conexión TCP es dividida por un caché de Web en la red satelital. Los usuarios en la red satelital conectada a este caché de Web no necesitan establecer conexiones de TCP hacia los servidores en las afueras si los contenidos requeridos están disponibles en el caché. El almacenamiento de web reduce de manera efectiva el retardo de conexión y consumo de ancho de banda.

II.3.5 Diseño de constelaciones satelitales

En éste punto se describen los parámetros y fórmulas que se requieren para realizar el diseño de una constelación satelital. A diferencia de un sistema satelital geoestacionario donde tres satélites son suficientes para una cobertura global [Jamalipour, 1998], un sistema satelital LEO requiere de decenas de satélites para lograr una cobertura global similar. A

partir de las Figura 4. y Figura 5 se determinan los elementos geométricos que intervienen y se ofrece su descripción:

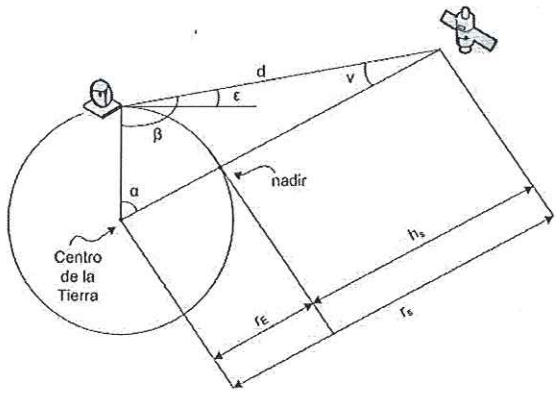


Figura 4. Relación geométrica entre satélite y usuario

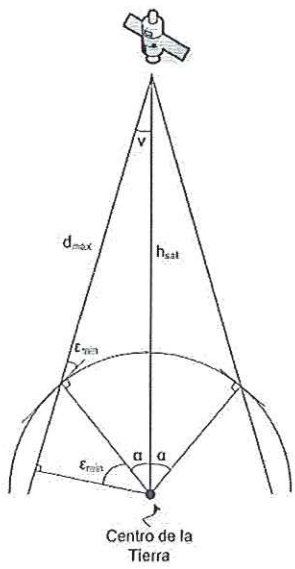


Figura 5. Consideraciones geométricas en un sistema satelital

A continuación se da una descripción de los parámetros que intervienen en el diseño de las constelaciones satelitales:

- Angulo de elevación ε_{min} sobre el cual un usuario puede ver el satélite por encima del horizonte.
- Angulo de nadir v el cual se toma desde el satélite hacia el centro de la Tierra y hacia la estación terrena.
- Angulo α de cobertura o central de la Tierra entre la estación terrena y el satélite.

- Radio de la Tierra r_E.
- Altura orbital h_s que representa la distancia desde la superficie de la Tierra al satélite.
- Radio satelital r_s que es la distancia del centro de la Tierra al satélite y se obtiene sumando el radio de la Tierra r_E y la altura satelital h_s.
- El ángulo formado del centro de la Tierra a la estación terrena y del satélite a la estación terrena es conocido como ángulo beta β y se obtiene sumando 90° al ángulo de elevación.
- Distancia d entre la estación terrena y el satélite.

Algunas precisiones que se deben hacer respecto de los parámetros mencionados anteriormentes son las siguientes:

- El ángulo de elevación de una estación terrena debe ser mayor o igual al ángulo de elevación mínimo para que pueda haber una correcta comunicación con el satélite. Si el ángulo de elevación real es menor al ángulo de elevación mínimo, la estación terrena no podrá comunicarse al satélite.
- La altura orbital se define a partir del tipo de constelación que se está diseñando (LEO, MEO, GEO), y se debe tener cuidado en su correcta selección ya que en función de ella está el período de rotación del satélite y por ende su tiempo de visibilidad.
- La distancia d no es un valor fijo, sino que va cambiando conforme el satélite se desplaza.

Con la Figura 4 se puede obtener la siguiente ecuación por medio de la ley de senos de un triángulo:

$$\frac{d}{sen\,\alpha} = \frac{r_S}{sen\,\beta} = \frac{r_E}{sen\,\nu} \tag{3}$$

Tomando los coeficientes del centro y de la derecha obtiene el ángulo de nadir:

$$\frac{r_S}{sen\,\beta} = \frac{r_E}{sen\,\nu} \tag{4}$$

Por lo tanto,

$$v = sen^{-1}\left[\frac{r_E}{r_S} sen\beta\right] \tag{5}$$

Se sabe que los ángulos internos de un triangulo suman 180° con lo cual se elabora la siguiente igualdad:

$$\alpha + \beta + v = 180^o \tag{6}$$

A partir de la expresión anterior se encuentra el valor del ángulo de cobertura central α con la ecuación número (7):

$$\alpha = 180^o - \beta - v \tag{7}$$

En caso de que se desee comprobar el valor obtenido, existe otra manera de calcular α en [Jamalipour, 1998]:

$$\alpha = \cos^{-1}\left(\frac{r_E}{r_S}\cos\varepsilon_{\min}\right) - \varepsilon_{\min} \tag{8}$$

Ahora se puede usar el valor de α para calcular el número necesario de satélites en una constelación. Para cubrir la superficie entera de la Tierra las huellas de los satélites deben traslaparse. Sin asumir ningún tipo específico de constelación y para encontrar el mínimo número de satélites para cobertura global, se considera la huella más grande posible de un satélite como el más grande hexágono inscrito en la huella, tal como se ve en la Figura 6:

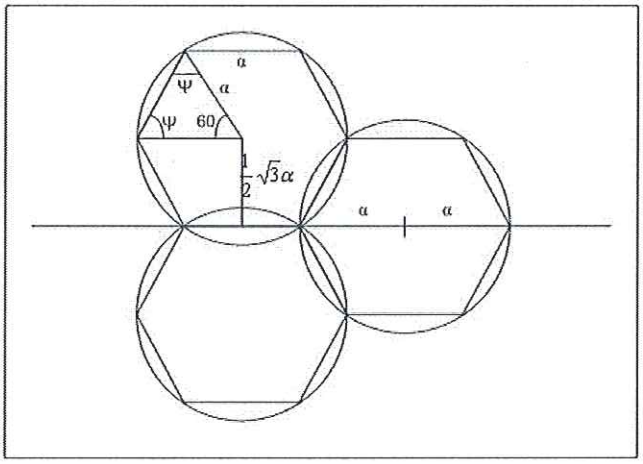


Figura 6. Relaciones geométricas de los hexágonos inscritos en las huellas de cobertura

Cada hexágono consiste de seis triángulos isósceles, cada uno con un ángulo de 60° y dos ángulos idénticos Ψ en la periferia de la huella. Considerando una forma esférica de la Tierra, la relación para el ángulo Ψ está dada por:

$$\tan\psi = \frac{\frac{1}{2}\sqrt{3}}{\frac{1}{2}\alpha \bullet \cos\alpha} = \frac{\sqrt{3}}{\cos\alpha} \tag{9}$$

Se calcula el exceso esférico de los triángulos σ con la ecuación (10):

$$\sigma = 2\psi - \frac{2\pi}{3} \tag{10}$$

Con la información obtenida ya es posible calcular el área del hexágono, que representa el área de cobertura efectiva, con la ecuación (11):

$$A_{hex} = 6r_e^2\,\sigma \tag{11}$$

El análisis presentado aquí puede ser suficiente para un sistema satelital GEO en el cual los satélites se encuentran en el plano del ecuador y perfectamente espaciados. En cambio para un sistema LEO, aparte del número mínimo de satélites, también se debe determinar el número mínimo de órbitas de esos satélites. Para eso, se considera la cobertura de un satélite en el ecuador. Entonces, en la condición de al menos dos satélites en cada órbita, cada órbita podría cubrir $3r_E\alpha$ del ecuador. De ahí, el mínimo número de órbitas para una cobertura global está dado por:

$$\Omega = \left\lceil \frac{2\pi}{3\alpha} \right\rceil \tag{12}$$

donde $\lceil x \rceil$ es el entero más pequeño igual o mayor a x.

Para calcular el número de satélites por plano orbital para una constelación LEO se usa la expresión (13):

$$Sat/Orb = \left\lceil \frac{2\pi}{\sqrt{3}\alpha} \right\rceil \tag{13}$$

Y por último para calcular el número total de satélites en la constelación se usa la expresión (14):

$$Número\,de\,Satélites = \left(\frac{Sat}{Orb}\right)(\Omega) \tag{14}$$

II.4 Parámetros de QoS para TCP por satélite

El rápido crecimiento y disponibilidad de la Internet es la fuerza que conduce la convergencia de las aplicaciones de datos, voz y video. Sin embargo, algunas aplicaciones demandan más ancho de banda y tienen requerimientos de tiempo de entrega exigentes, por lo cual se requiere el desarrollo de una infraestructura de red que vaya más alla del simple mejor esfuerzo con el que trabaja el Protocolo de Internet. Las comunicaciones satelitales juegan un papel importante en el acceso a Intenet a través de redes híbridas satelitales/terrestres. Para proveer una alta tasa de acceso a Internet y conectividad global, la provisión de Calidad de Servicio (QoS) en las futuras redes satelitales es crucial.

En éste apartado se definirá el concepto de Calidad de Servicio y los parámetros con los que se evaluará el desempeño del Protocolo de Control de Transmisión (TCP) pero antes se dará un breve repaso a dos mecanismos que se utilizan para mejorar la QoS de la Voz sobre IP que aunque no se emplean en la realización de este trabajo se considera pertinente presentarlos.

Para mejorar la QoS de VoIP se han desarrollado dos mecanismos conocidos como Modelo de Servicios Integrados (Intserv) y Marco de Servicios Diferenciados (Diffserv) [Conte, 2000].

Intserv debe manejar recursos como el ancho de banda y buffer para cada aplicación de tiempo real, y lo que hace es que en cada enrutador reserva recursos utilizando el protocolo de reservación de recursos (RSVP) para suministrar una QoS específica para una cadena de paquetes o flujos.

La arquitectura Diffserv puede ofrecer a cada usuario un rango de servicios diferenciados en base al desempeño. El tráfico que llegue a la red es clasificado y condicionado en los límites de la red y el nivel de desempeño que requieran los usuarios se realiza en base a un análisis de paquete por paquete.

Calidad de Servicio es la habilidad de un elemento de red (una aplicación, computador o ruteador) de tener algún nivel de seguridad de que sus requerimientos de tráfico y servicio pueden ser satisfechos [Kota *et al.*, 2004].

De acuerdo a la Recomendación E.800 de la ITU-T, se define a la QoS como el efecto colectivo del desempeño de servicio el cual determina el grado de satisfacción de un usuario del servicio. El término QoS es utilizado de muchas maneras, y varía desde la percepción del usuario que tiene de un servicio, hasta un conjunto de parámetros necesarios para lograr un nivel de calidad de servicio. A continuación se definen los parámetros de QoS que se utilizarán en éste trabajo de tesis.

II.4.1 Latencia

Es el tiempo que le toma a un paquete ser transportado del transmisor al receptor. Para el protocolo TCP, los niveles altos de retardo se transforman en cantidades de datos que están en tránsito en la red, lo cual afecta los contadores y temporizadores del protocolo. TCP es un protocolo que se toma el tiempo a sí mismo, y la tasa de transmisión se ajusta dinámicamente de acuerdo al flujo de información que viene desde el receptor por medio de los reconocimientos. Grandes retardos pueden estar disparando retransmisiones continuas que no permitirán al TCP salir de la fase de inicio lento. El máximo nivel de retardo con el que se va a trabajar va a ser de 400 ms [Goyal *et al.*, 1999] que es el máximo permitido para Voz sobre IP.

II.4.2 Variación en el retardo (Jitter)

Es la variación que existe en el retardo de transporte extremo a extremo de los paquetes. Si se tienen altos niveles de jitter se provoca que el protocolo TCP haga estimaciones muy conservadoras del Tiempo de Viaje Completo (RTT), lo que causa que el protocolo opere de manera ineficiente cuando se vale de las expiraciones de temporizador para reestablecer el flujo de datos. El nivel máximo de jitter con el que se estará trabajando es de 20 ms [González, 2000].

II.4.3 Tasa de Error de Bit (BER)

Se define como el número de errores de bit que ocurren para un número de bits transmitidos. La máxima tasa permitida en BER es de 10^{-5} [Nguyen, 2001].

II.4.4 Tasa de Pérdida de Paquetes (PLR)

Es la relación que existe entre el número total de paquetes recibidos con respecto al número total de paquetes enviados. Las razones por las que se puede presentar una alta tasa de paquetes perdidos puede ser porque los sistemas de conmutación estén saturados de tráfico y se estén perdiendo paquetes, o se estén entregando fuera de orden, o porque los puntos de interconexión de las redes no tengan una configuración adecuada. El nivel límite a manejar es de 3% a 5 %. [Beltrán, 1999].

II.4.5 Caudal Eficaz

Es el porcentaje del tráfico ofrecido que fue recibido de manera exitosa en el extremo receptor. Este porcentaje puede servir para determinar la cantidad de información que se está transmitiendo por unidad de tiempo.

II.5 Conclusiones

En este capítulo se analizaron las características que presentan los servicios de acceso a Internet por satélite, se hizo un breve análisis del protocolo TCP original y se vieron las modificaciones que se han propuesto para su uso en redes satelitales. Se dio un repaso a los retos a los que se enfrenta TCP cuando es usado en satélites GEO y LEO y se

vieron los elementos que intervienen en el diseño de constelaciones satelitales. Por último se analizaron los parámetros de QoS con los cuales se va a estar evaluando el desempeño de TCP.

III. ANALISIS DEL PROTOCOLO TCP

En el capítulo anterior se abordaron los conceptos relacionados con los fundamentos del protocolo TCP a nivel de su RFC original, así como de las mejoras propuestas para su utilización en ambientes satelitales. Se expusieron los retos a los que se enfrenta cuando se usa en redes satelitales LEO y GEO, se habló del diseño de constelaciones satelitales y se analizaron los conceptos de Calidad de Servicio (QoS) y las métricas que se habrán de usar para evaluar su desempeño. En este capítulo se analizará el protocolo TCP y los diversos mecanismos de funcionamiento interno que lo componen. Se evaluarán varias soluciones propuestas para el uso del TCP por satélite y las variantes que se han propuesto para su control de congestión.

III.1 TCP puro (original)

El Protocolo TCP es un protocolo muy complejo, ya que para su funcionamiento requiere que sus mecanismos internos interactúen con elementos y ambiente de comunicación que lo rodean constantemente. Algunas cuestiones relacionadas con su funcionamiento interno fueron introducidas en el punto II.2.1, y las restantes serán analizadas más adelante. Dentro de los servicios que ofrece TCP a las aplicaciones y que fueron explicados anteriormente se encuentran los siguientes: Servicio Orientado a Conexión, Servicio de Entrega en Flujo, Confiable y de Semántica Extremo a Extremo [Hassan y Jain, 2004 y Postel, 1981].

Formato de Encabezado de TCP. Cada segmento de TCP está dividido en dos partes: una de longitud estándar de 20 bytes que es el encabezado y otra que es la carga útil que contiene los datos de la aplicación y es de longitud variable. La información que maneja este encabezado se va a analizar con ayuda de la Figura 7:

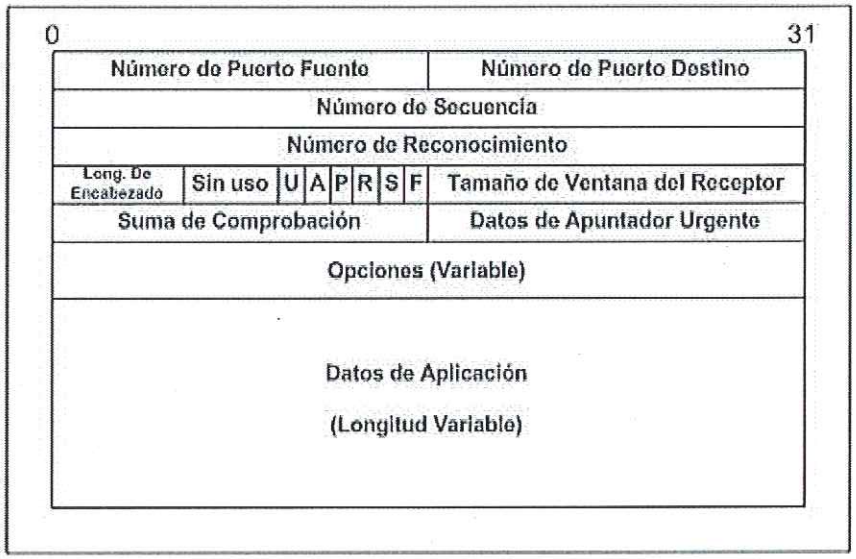


Figura 7. Formato del segmento TCP

Elementos que forman el Segmento TCP [Hassan y Atiquzzaman, 2004 y Hassan y Jain, 2004].

Número de Puerto Fuente. Tiene una longitud de 16 bits y se utiliza para identificar de manera única a cada aplicación TCP en el computador fuente. La identificación del puerto permite el multiplexaje y demultiplexaje de múltiples conexiones TCP sobre el mismo proceso del protocolo.

Número de Puerto Destino. Tiene una longitud de 16 bits y se usa para identificar a una aplicación TCP en el computador destino. Cuando un segmento TCP es recibido en el host destino, este número de puerto es usado para entregar los segmentos de datos a la aplicación correcta.

Número de Secuencia. Tiene una longitud de 32 bits y este campo contiene el número de secuencia del primer byte de datos que es llevado en el segmento TCP. Por ejemplo, si el segmento anterior comenzó con un número de secuencia 3001 y contenía 1570 bytes de datos, entonces el número de secuencia del siguiente segmento TCP es el 4571.

Número de Reconocimiento. Tiene una longitud de 32 bits y el destino usa este campo para reconocer los datos recibidos correctamente.

Longitud de Encabezado. Tiene una longitud de 4 bits y se utiliza para indicar la longitud de un encabezado TCP en palabras múltiplo de 32-bit. En la mayoría de los casos la longitud de encabezado de un segmento TCP es 20 bytes, pero esto varía dependiendo si se usa el campo de opciones. Ya que el encabezado puede ser de longitud variable, el campo de longitud también puede ayudar a identificar dónde comienza la carga útil.

Reservado. Tiene una longitud de 6 bits y están reservados para uso futuro o experimental.

Banderas. Son diferentes tipos de mensaje de protocolo que un segmento de TCP puede llevar y cada bit en el campo de bandera se utiliza para identificar cada tipo. La Tabla I muestra los distintos tipos de bandera y su propósito. Cabe mencionar que pueden ser utilizados al mismo tiempo.

Tabla I. Banderas de TCP

Bandera	**Descripción**
ACK (A)	Campo de reconocimiento válido
FIN (F)	Segmento final del transmisor
PSH (P)	Operación de empuje invocada. El proceso receptor requiere notificación
RST (R)	Reinicio de la conexión
SYN (S)	Comienzo de una nueva conexión
URG (U)	Campo de apuntador urgente válido

Tamaño de Ventana de Recepción. Tiene una longitud de 16 bits y por medio de este campo el receptor anuncia al transmisor su ventana o espacio de buffer disponible. Esta ventana de recepción es usada por el transmisor para propósitos de control de flujo.

Suma de Comprobación. Tiene una longitud de 16 bits y se calcula sumando el encabezado TCP, la carga útil de TCP y el pseudoencabezado que se compone de las

direcciones IP fuente y destino así como del campo de longitud del encabezao IP. El campo de Suma de Comprobación protege el encabezado y la carga útil del segmento TCP.

Apuntador de Urgente. Tiene una longitud de 16 bits y se utiliza cuando un segmento TCP transporta datos que necesitan un trato prioritario. Para que los datos reciban este trato se debe fijar la bandera URG y los datos con esta característica son procesados antes de cualesquiera otros que estén esperando en el buffer.

Opciones. Tiene una longitud variable y se especifican utilizando múltiplos de bytes de la siguiente manera: habrá dos bytes extra antes de cada opción. El primer byte indica el tipo de opción seguido por un segundo byte que indica la longitud de la opción en bytes, incluyendo estos dos bytes. Algunos ejemplos de opciones son: El Máximo Tamaño de Segmento (MSS) con una longitud de 16 bits y la usa el TCP transmisor cuando se está estableciendo la conexión para negociar el MSS que se utilizará en la conexión. Cabe mencionar que los 16 bits que se usan en éste campo limitan el MSS a 64 KB. Timestamp, con una longitud de 8 bytes, se usa para hacer cálculos más precisos del Tiempo de Viaje Completo (RTT) de la siguiente manera: TCP llena el campo con el valor real del tiempo, y el receptor manda de vuelta el valor total del tiempo recibido en el segundo campo en un segmento ACK, lo que facilita el cálculo más preciso del RTT.

Encapsulación en IP. Una vez que un segmento TCP está listo para ser transmitido, se pasa a la capa IP donde el segmento completo (encabezado y carga útil) es encapsulado en la carga útil del datagrama IP, como se ve en la Figura 8.

III.2 Congestión en la transmisión de TCP

En este punto se describirá cómo reaccionan los mecanismos internos de TCP cuando detectan congestión. La estrategia básica que sigue es enviar paquetes a la red sin realizar una reservación y luego reaccionar a los eventos que ocurran [Peterson y Davie, 1996].

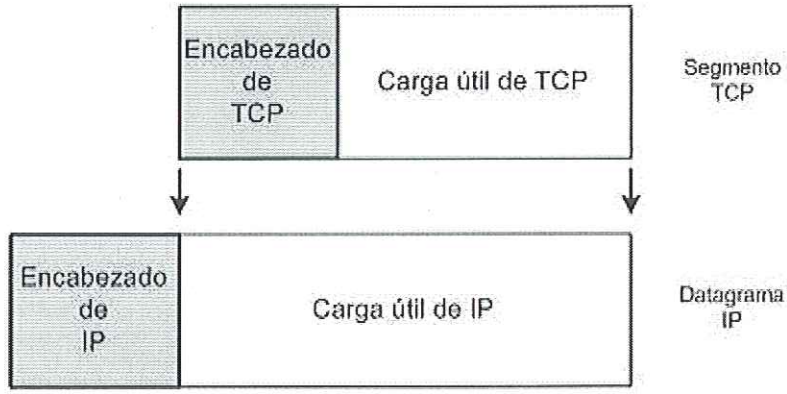


Figura 8. Encapsulación de segmento TCP

El control de congestión de TCP fue introducido a finales de la década de 1980 por Van Jacobson. Un poco antes de esto, la Internet comenzaba a sufrir de episodios de gran congestión, debido a que las computadoras conectadas a la red enviaban paquetes tan rápido como la ventana anunciada se los permitía. Esto ocasionaba congestión en algunos enrutadores, que los paquetes fueran descartados y los temporizadores de las conexiones expiraran disparando nuevas retransmisiones, lo cual traía en consecuencia una mayor congestión. A grandes razgos, lo que hace el control de congestión de TCP es determinar qué tanta capacidad hay disponible en la red, para de esa manera determinar qué tantos paquetes puede enviar de manera segura.

Ya que la fuente ha enviado los segmentos, espera en respuesta la llegada de paquetes ACK como una señal de que los paquetes han sido recibidos en su destino y al mismo tiempo determinar que es seguro enviar nuevos paquetes hacia la red sin elevar el nivel de congestión. Se dice que TCP es auto regulado (self clocking) debido al hecho de que usa los ACKs como medio para elevar o disminuir la transmisión. A continuación se describen los algoritmos que usa TCP para control de congestión.

El principio detrás del control de congestión es el ajustar la ventana de transmisión del transmisor de tal manera que se pueda prevenir el desbordamiento de los buffers tanto del receptor como de los enrutadores intermedios. Para lograr eso, TCP utiliza una variable de control de ventana llamada *VentanadeCongestión*, y la idea que utiliza para manejarla es determinar cuál es el espacio de buffer disponible en el enrutador más congestionado en la conexión TCP de la trayectoria extremo a extremo. Entonces podría fijar la

VentanadeCongestión como el mínimo de la *VentanaAnunciada* y *VentanadeCongestión*, lo que podría prevenir el desbordamiento del buffer tanto en el receptor como en la red. El reto que se presenta para TCP es cómo determinar cuál es el espacio de buffer disponible en los enrutadores de la red, y para sobreponerse a esto asume congestión en la red siempre que expire el temporizador de retransmisión y reacciona a la congestión de red ajustando la *VentanadeCongestión* mediante tres algoritmos: inicio lento, evasión de congestión y disminución multiplicativa, lo que se explica a continuación.

Inicio Lento. El mecanismo de inicio lento comienza a transmitir con un tamaño de ventana pequeño, para luego incrementarlo lentamente conforme vayan llegando los reconocimientos o ACKs. Este crecimiento lo hace para sondear la red y tratar de determinar cuál es el espacio de buffer disponible en la red. El funcionamiento del inicio lento de TCP se explicará con ayuda de la Figura 9.

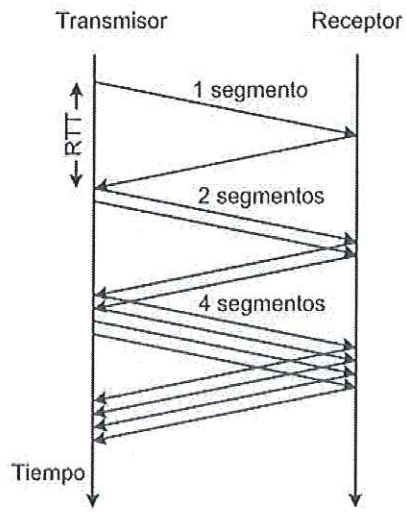


Figura 9. Inicio Lento en TCP

Inicialmente la *VentanadeCongestión* se fija a un segmento y se incrementa un segmento cada vez que uno es reconocido, lo cual se ilustra en la Figura 9. Al principio el transmisor de TCP envía un segmento; después de recibir el ACK por éste, envía dos segmentos más, por lo que la *VentanadeCongestión* se incrementa a dos. Cuando estos dos segmentos son reconocidos, envía cuatro nuevos segmentos (*VentanadeCongestión* es incrementada a cuatro) y así sucesivamente.

Evasión de Congestión. Como se vió en el ejemplo de la Figura 9, después de cada RTT el tamaño de ventana se duplica, lo que se traduce en la transmisión del doble de segmentos. Este crecimiento exponencial que experimenta la *VentanadeCongestión* se puede ver en la Figura 10.

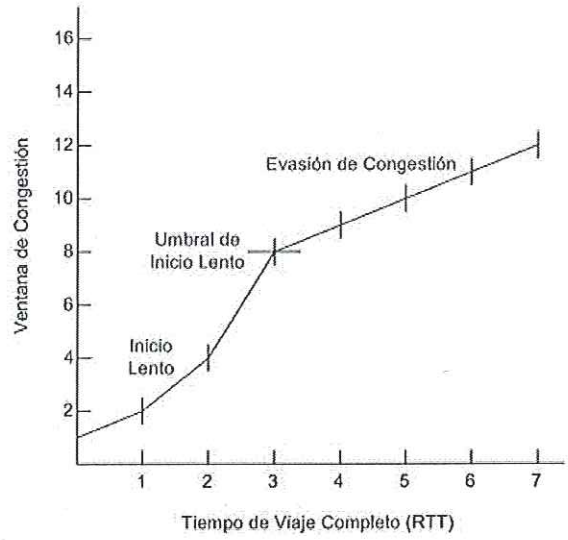


Figura 10. Evasión de congestión

Si no se marca un punto para que se detenga este crecimiento, puede conducir a congestión, y para prevenir que esto suceda TCP utiliza un algoritmo llamado evasión de congestión, el cual cambia el crecimiento de la *VentanadeCongestión* de exponencial a lineal una vez que ha llegado a un umbral. Este umbral es ajustado dinámicamente por medio de una variable llamada ssthresh. El incremento lineal durante evasión de congestión se logra incrementando la *VentanadeCongestión* en 1/VentanadeCongestión cada vez que se recibe un reconocimiento o ACK. En la Figura 10 se puede ver como con el algoritmo de inicio lento se tiene un crecimiento exponencial hasta el umbral de 8 para éste ejemplo. Después de este punto, la *VentanadeCongestión* entra en la fase de evasión de congestión y crece linealmente.

Disminución multiplicativa. Las transiciones de las fases de inicio lento a evasión de congestión son controladas por la variable ssthresh, y a su vez ésta variable es controlada por el algoritmo de Disminución multiplicativa. Con este algoritmo TCP se encarga de

disminuir el valor de ssthresh a la mitad del valor actual de la *VentanadeCongestión* cada vez que se presente una expiración de temporizador hasta un mínimo de dos segmentos. Cuando lo anterior sucede (expiración de temporizador) el tamaño de la *VentanadeCongestión* se reduce a un segmento para forzar un inicio lento. Si se presentan expiraciones de temporizador consecutivas, lo que significa congestión severa de la red, la Disminución multiplicativa reduce la tasa de transmisión exponencialmente, ocurriendo lo inverso al inicio lento. El conjunto del Incremento Aditivo de la *VentanadeCongestión* durante la fase de evasión de congestión y la Disminución Multiplicativa del umbral de inicio lento (ssthresh) son conocidas como el algoritmo Incremento Aditivo, Disminución Multiplicativa (AIMD, Additive Increase, Multiplicative Decrease) [Hassan y Jain, 2004].

III.3 Mecanismo de Control de Flujo en TCP

El control de flujo es el mecanismo que ayuda a prevenir que un transmisor rápido inunde o desborde a un receptor lento. Cada receptor TCP destina un espacio de buffer para una conexión,. los datos recibidos de manera correcta y en orden forman una cola en este buffer para que la aplicación a la que van encaminados los lea y al mismo tiempo lo vaya desocupando. El inconveniente que presenta es que si se presenta un transmisor rápido y el receptor no puede sostener una tasa tan rápida el buffer se puede desbordar. Para evitar que esto suceda, TCP utiliza el mecanismo de control de flujo para manejar la tasa de transmisión del transmisor. Para llevar a cabo el control de flujo TCP utiliza el mecanismo de ventana deslizante.

Ventana deslizante. Para este mecanismo TCP controla el número de bytes que tiene en tránsito (que han sido transmitidos pero aún no han sido reconocidos) por medio del tamaño de ventana. Cuando se tiene una ventana completa de datos en su viaje al receptor, TCP no transmite mas segmentos y espera a que lleguen los reconocimientos de regreso. Cuando estos llegan TCP comienza a transmitir otra vez nuevos datos, pero sin exceder el número de bytes que fueron reconocidos. Para explicarlo de manera más sencilla se utiliza la Figura 11. Para este ejemplo el tamaño de ventana es de seis bytes, lo que quiere decir que TCP no tendrá más de seis bytes en tránsito en cualquier momento.

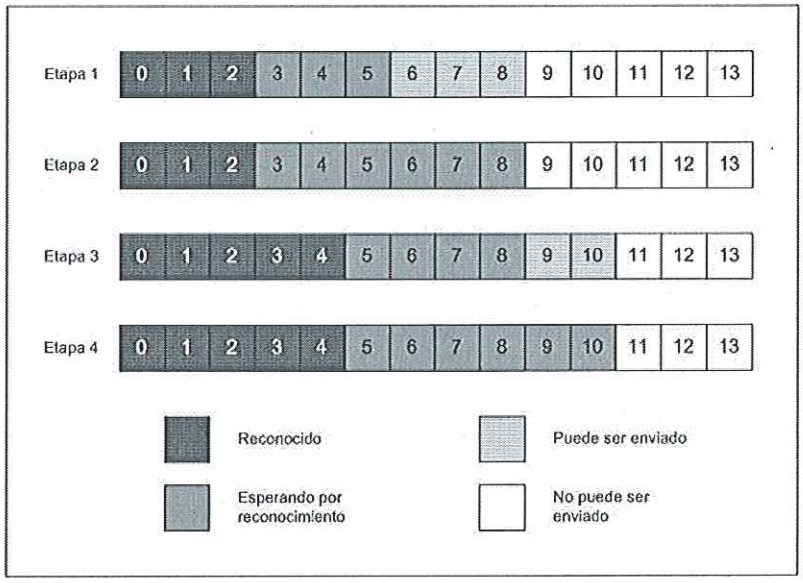


Figura 11. Ventana deslizante de TCP

Con respecto de la Figura 11 se tienen las siguientes etapas en la transmisión:

- Etapa 1. Los bytes 0, 1 y 2 ya han sido transmitidos y reconocidos por el receptor. Los bytes 3, 4 y 5 ya fueron enviados y el transmisor está esperando el reconocimiento o ACK. Como el tamaño de ventana es de 6 bytes, se pueden transmitir los bytes 6, 7 y 8, y los bytes mayores a 9 no pueden ser transmitidos por la limitación en el tamaño de ventana.
- Etapa 2. Ya enviados los bytes 6, 7 y 8, el transmisor está esperando reconocimiento para todos los segmentos de la ventana actual. Como todos los segmentos están en tránsito no pueden ser enviados más datos en esta etapa.
- Etapa 3. El transmisor recibe los reconocimientos para los bytes 3 y 4, por lo cual la ventana deslizante se desliza dos espacios a la derecha y es posible enviar los bytes 9 y 10
- Etapa 4. El transmisor de TCP envía los bytes 9 y 10 y espera los reconocimientos de los bytes 5 a 10.

Explicado de una manera más sencilla, el lado derecho de la ventana se desliza cuando se envía un byte y el lado izquierdo cuando se recibe un reconocimiento. Aunque en el ejemplo anterior el tamaño de ventana permaneció fijo con un valor de 6 bytes, en la

práctica es ajustado de manera dinámica de acuerdo al espacio de buffer disponible en el TCP receptor

III.3.1 Mecanismo de reconocimiento

Para asegurar que los segmentos de datos son entregados correctamente al otro extremo de la conexión, TCP utiliza los reconocimientos o ACKs. A continuación se verán las características del mecanismo de retransmisión.

Reconocimiento Acumulativo. Cada reconocimiento (ACK) es una confirmación de que los bytes hasta el número del ACK han sido recibidos correctamente, lo que podría presentar una ventaja en caso de que se perdieran muchos reconocimientos, ya que el de más alto número compensa por aquellos que no llegaron.

Segmento de solo-ACK y de llevar a cuestas. Un paquete de reconocimiento o ACK es indicado por medio de un campo de reconocimiento en el encabezado de TCP. Para reconocer bytes recibidos correctamente un receptor TCP tiene dos opciones: una es crear segmentos de "solo-reconocimiento" o "solo ACK", donde el segmento solo lleva el encabezado que contiene el número de reconocimiento y no lleva datos. La otra opción es que el receptor envía el reconocimiento en un segmento de datos, que se llama "llevar a cuestas" y resulta benéfico para reducir el tráfico de reconocimientos en la dirección de regreso.

ACK o reconocimiento retardado. Un receptor de TCP tiene dos opciones para generar los reconocimientos: una es generarlos tan pronto vaya recibiendo los segmentos de datos y la otra es retrasarlos un cierto tiempo. La ventaja del receptor al retardar el ACK es que puede ser capaz de reconocer dos segmentos al mismo tiempo y con ello reducir el tráfico de ACKs, pero si lo retarda demasiado puede ocasionar una expiración de temporizador, lo que hace que lo tenga que retransmitir. Se recomienda que un receptor no retarde los ACKs mas de 500 ms.

ACK duplicado. Este tipo de reconocimiento se genera cuando un segmento se pierde en la red, y el receptor continúa recibiendo segmentos de datos con número mayor al segmento que se perdió. Lo que hace el receptor de TCP es almacenarlos en buffer y vuelve a generar un ACK por los últimos bytes recibidos en secuencia correcta. En el TCP original

el transmisor simplemente ignoraba el ACK duplicado, aunque recientemente se han propuesto variantes en lo que respecta a cómo responde el TCP cuando se presentan ACKs duplicados.

III.3.2 Mecanismo de retransmisión

La retransmisión representa uno de los soportes básicos del servicio de transferencia confiable de TCP; si se pierde en la trayectoria de la fuente al destino, debe ser retransmitido. Para detectar la pérdida de un segmento, TCP mantiene un temporizador de retransmisión por cada segmento enviado. La duración del temporizador está fija para un período llamado expiración de retransmisión (RTO). Si el reconocimiento o ACK llega dentro de ese período, el temporizador se reinicializa, si ocurre lo contrario, el temporizador expira y el segmento se retransmite. Es importante fijar un valor adecuado para el RTO ya que puede beneficiar o afectar el desempeño del protocolo, por lo siguiente. El período de expiración debe ser mayor que el retardo de viaje completo (RTT), ya que en la trayectoria que cursa el segmento se va a encontrar con diversos tipos de retardos como retardo de transmisión, de propagación en el enlace, de procesamiento de encabezado, de generación de reconocimiento o ACK. También los encontrará en la trayectoria de regreso, por lo que se tienen dos retos por eludir. Por un lado si se fija el RTO a un valor mas grande del necesario, hay un mayor retardo para las aplicaciones si las pérdidas son frecuentes. Por otro lado si se fija el RTO a un valor menor del requerido ocasiona retransmisiones prematuras, lo que se traduce en desperdicio de los recursos de comunicación disponibles (ancho de banda, tiempo de procesamiento, etc.)

Para resolver esto, TCP hace un estimado del RTT para cada una de sus conexiones, lo representa con la variable *RTTEstimado* y lo calcula a partir de una muestra del RTT de la conexión. La muestra del RTT la representa con la variable *RTTMuestreado*, y se define como el tiempo desde que se transmitió un segmento TCP hasta que se recibe el ACK por ese segmento. Debido a que *RTTMuestrado* varía en su valor cada vez que se toman mediciones, lo que se puede deber entre otras cosas a los retardos de encolamiento variables en los ruteadores intermedios, se usa un promedio para calcular el *RTTEstimado* de la siguiente manera:

$$RTTEstimado = (1 - \alpha) * RTTEstimado + \alpha * RTTMuestreado \quad (15)$$

Un valor típico que se utiliza para α es el de 0.125, el cual tiene el impacto de darle un peso bajo al valor medido de *RTTMuestreado* en el período previo y el de un alto peso a los datos históricos representados por *RTTEstimado*[Hassan y Jain, 2004]. Una vez que se conoce el valor de RTT, el RTO se estima con la siguiente expresión:

$$RTO = RTTEstimado + 4 * desviación \quad (16)$$

El factor desviación ayuda a reducir las fluctuaciones entre *RTTMuestreado* y *RTTEstimado*.

Establecimiento y Terminación de Conexión. El servicio orientado a conexión que provee TCP lo realiza a través de dos mecanismos que son el establecimiento y la terminación de conexión. Antes de comenzar una transmisión TCP debe establecer una conexión y cuando esta finaliza termina la conexión.

A continuación con ayuda de la Figura 12 se explica el procedimiento de establecimiento de conexión:

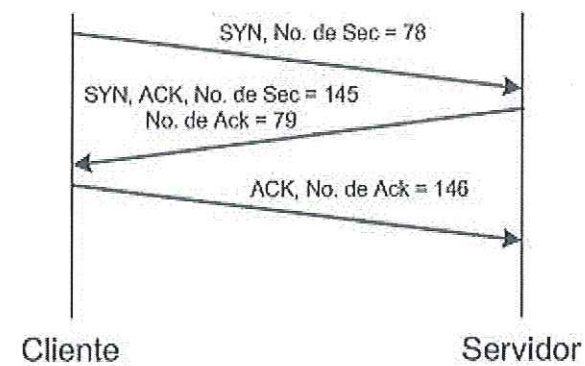


Figura 12. Establecimiento de conexión usando saludo de tres vías

Al proceso de TCP de establecer una conexión se le conoce como Saludo de Tres Vías. La Figura 12 muestra los tres pasos de cómo sucede esto utilizando como ejemplo la comunicación entre un cliente y un servidor.

1. El cliente envía un segmento SYN (habilita el bit SYN en el encabezado) al servidor con un número de secuencia inicial (que en este ejemplo es el 78) que es el que va a usar para esta conexión.
2. El servidor contesta enviando un segmento que tiene los bits de SYN y ACK habilitados en la bandera (SYN + ACK, Número de ACK = 79, Número de secuencia = 145). El número de ACK (Número de ACK = 78) indica que el servidor ha recibido correctamente los bytes hasta el 78 y está a la espera de recibir del número de secuencia 79 en delante. Con el número de secuencia (No. de Sec = 145) el servidor le avisa al cliente que va a usar el número 145 como número de secuencia. Tanto el cliente como el servidor pueden utilizar distintos números de secuencia iniciales.
3. Por último el cliente reconoce el número de secuencia del servidor (No. de Ack = 146) con un segmento ACK. Con esto termina el establecimiento de conexión de TCP de manera correcta y ahora el cliente y el servidor pueden intercambiar información entre ellos.

Terminación de Conexión. El proceso de terminar una conexión es conocido como saludo de cuatro vías y se explica con ayuda de la Figura 13.

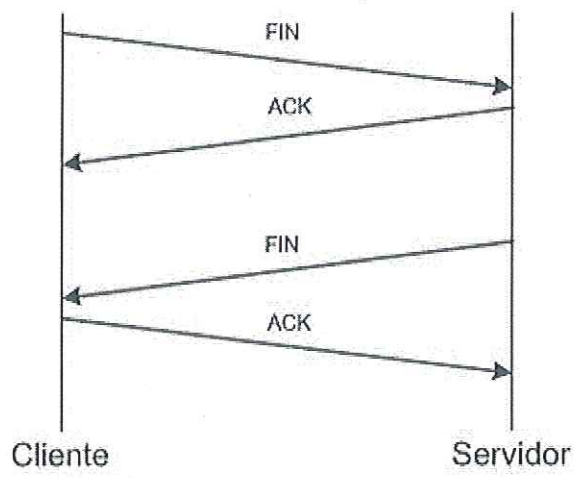


Figura 13. Terminación de una conexión TCP usando saludo de cuatro vías

1. El cliente envía un segmento FIN (habilita el bit en el encabezado) hacia el servidor para darle a conocer que desea terminar la conexión.
2. El servidor envía un ACK para confirmar que recibió un segmento FIN, y es en este momento que la comunicación en la dirección de cliente a servidor se termina. Se puede dar el caso que en la dirección del servidor al cliente todavía haya comunicación por si la transferencia de información aún no termina.
3. Cuando el servidor está listo, al igual que hizo el cliente en el paso 1, envía un segmento FIN hacia el otro extremo. Los pasos 2 y 3 no se pueden combinar, ya que, aunque el servidor reciba un segmento FIN del cliente para que finalice la comunicación, no necesariamente lo hace.
4. El cliente reconoce que recibió el segmento FIN con un segmento ACK, y de común acuerdo los dos finalizan la conexión.

Hay que mencionar que el saludo (como se le conoce a este mecanismo de conexión) introduce retardos, ya que los segmentos SYN y FIN tienen que viajar de extremo a extremo, de ahí que se considere que es una de las fuentes que mayor retardo introduce en las comunicaciones de distancias grandes, como por ejemplo las comunicaciones satelitales.

III.4 Soluciones propuestas para TCP por satélite

Se ha realizado mucha investigación para mejorar el desempeño de TCP sobre satélite. Estas soluciones se pueden agrupar en tres categorías. Las soluciones de capa inferior tratan de resolver algunos de los problemas que se presentan sin alterar el TCP mismo, haciendo que la capa inferior realice funciones que apoyen a TCP. En segundo lugar están las soluciones estandarizadas de la IETF que son modificaciones aceptadas al TCP. En tercer lugar se encuentra una considerable cantidad de soluciones experimentales propuestas por diferentes estudios e investigaciones. Estas posibles soluciones arrojan resultados desde alterar TCP en una manera que todavía no es seguro estandarizar, hasta incluso reemplazar TCP con protocolos de transporte alternativos.

III.4.1 Soluciones de capas inferiores

Existen varias soluciones de capas inferiores que se dirigen principalmente hacia dos problemas de un enlace inalámbrico o satelital: BER alto y asimetría del canal. Las soluciones propuestas mas comunes para el problema de un BER alto son la Petición de Repetición Automática (ARQ), lo cual son retransmisiones de la capa de enlace, y Corrección de Errores (FEC), que es el envío de paquetes de código redundante para corregir los errores en el receptor. Tales soluciones de capa inferior son la manera estándar para enfrentar estos retos en las arquitecturas interredes, y las modificaciones propuestas solo conciernen a la capa de enlace de datos. Uno de los esquemas posiblemente más utilizados es el protocolo Snoop, el cual es un esquema de capa de enlace que de hecho entiende las funcionalidades de TCP, suprimiendo los reconocimientos duplicados para prevenir retransmisiones TCP mientras la capa de enlace intenta recuperar los paquetes perdidos.

Otra propuesta para tratar de lidiar con la dificultad de distinguir entre pérdidas por congestión y corrupción a través de las capas inferiores es la notificación de congestión explícita (ECN) o notificación de corrupción explícita [Lang, 2002]. Para la notificación de congestión explícita se requiere que los ruteadores intermedios de la red estén midiendo la congestión (niveles de encolamiento) y que indiquen el estado congestionado al transmisor o receptor, mientras que para la notificación de corrupción explícita solo requiere ser generada por los ruteadores adyacentes al enlace inalámbrico. Ambos mecanismos requieren cambios a los enrutadores de red, así como al protocolo de transporte TCP para darle la habilidad de manejar este tipo de mensajes entrantes.

Existen otras alternativas de capa inferior para TCP que trabajan mejor sobre canales asimétricos, como el control de congestión de reconocimientos, el filtrado de ACKs y las rutinas de calendarización de paquetes. En el control de congestión de reconocimientos los enrutadores intermedios generan mensajes de notificación de congestión explícita para los reconocimientos y el receptor reduce la tasa a la cual envía reconocimientos. En el filtrado de ACKs los enrutadores intermedios remueven reconocimientos viejos de sus colas si llegan nuevos, y las rutinas de calendarización de paquetes que se han propuesto están relacionadas con tratar de transmitir reconocimientos

antes de los paquetes de datos. Los enrutadores y TCP necesitan ser modificados de la misma manera que sucede cuando se usa FEC y ARQ para que trabajen estas soluciones.

III.4.2 Modificaciones de la IETF a TCP

La IETF ha estandarizado varios cambios que están enfocados a mejorar el desempeño de TCP en ambientes inalámbricos. Uno de los cambios mas importantes está relacionado con la limitación que impone la ventana de recepción y que impacta el caudal eficaz en enlaces con un alto Producto Retardo Ancho de Banda (PRAB). Esta modificación se llama escalamiento de ventana, y permite que la ventana de recepción anunciada sea fijada a un valor hasta de 2^{30} bytes, lo suficientemente alto para la mayoría de las redes actuales.

Otra propuesta ya estandarizada son los reconocimientos selectivos (SACKs) que permiten la recuperación más eficiente de información perdida por tiempo de viaje completo. Los reconocimientos selectivos permiten al receptor avisar al transmisor qué paquetes no han llegado, agilizando la recuperación ante pérdidas que anteriormente estaban limitadas a un paquete perdido por tiempo de viaje completo (RTT). Otra solución experimental para acelerar la fase de Inicio Lento que es analizada por la IETF, es la posibilidad de que el valor inicial de la ventana de congestión sea mayor a uno, y lo que se discute es opción de elevar el tamaño de la ventana de congestión hasta a cuatro segmentos.

III.4.3 Modificaciones a TCP no estandarizadas

Una modificación que enfrenta el problema de los envíos poco frecuentes o de pérdida de los reconocimientos, común en las redes asimétricas, es el Conteo de Bytes, y fue creada para tratar de incrementar el caudal eficaz de TCP cuando se usan reconocimientos retrasados, esto es, que se reconoce cada segundo segmento de datos. La idea que maneja el Conteo de Bytes es el incrementar la ventana de congestión de acuerdo a la cantidad real de datos que han sido reconocidos, y no de acuerdo al número de reconocimientos que han sido recibidos, como en TCP. Esta manera de trabajar contrarresta el efecto lento del crecimiento de ventana cuando los reconocimientos son enviados de manera infrecuente o se pierden debido a que se desbordó el buffer. La desventaja es que

incrementa el efecto de ráfagas en la transmisión de datos, ya que le da la posibilidad al transmisor de transmitir más de un paquete por reconocimiento recibido.

III.4.3.1 Enfoque de conexión dividida

La propuesta de dividir el protocolo de transporte en dos, con una parte que funcione bien en el segmento terrestre y la otra parte para el segmento inalámbrico, se ha venido discutiendo cada vez más, ya que se dice que es difícil diseñar un protocolo de transporte que trabaje bien en todo tipo de redes y ambientes tecnológicos. Otro argumento utilizado es que cuando se sugieren modificaciones que incluyen realizar cambios a las computadoras o enrutadores de Internet, la idea de la conexión dividida toma fuerza, ya que el realizar cambios en enrutadores y computadoras se antoja difícil sino es que imposible. La idea que propone la conexión dividida es dividir la conexión de TCP en el gateway enrutador y que en el segmento inalámbrico se use un protocolo de transporte especialmente diseñado [Hassan y Jain, 2004]. La ventaja de este enfoque es el uso de protocolos especializados en el segmento inalámbrico, con lo cual se solucionaría la cuestión de actualizar o modificar todas las computadoras de Internet. Se podrían usar mecanismos no estandarizados en el segmento inalámbrico, aparte de que esta división de conexión podría reducir el tiempo de viaje completo. Para las computadoras móviles con capacidad limitada de procesamiento, la conexión dividida es útil porque el gateway enrutador puede manejar mucho del sobreencabezado de comunicación del computador móvil.

Una desventaja que presenta es que no cumple con la semántica extremo a extremo, que es la base de la confiabilidad de TCP (sección II.2.1) que establece que los ACKs solo serán generados por el TCP receptor, y solamente después de que los datos han sido recibidos correctamente. Al tener la conexión dividida y no cumplir con este servicio de TCP el establecimiento y finalización de conexión se vuelve delicado. Otra desventaja es que la división de conexión genera un considerable sobreencabezado en los enrutadores responsables de las conexiones divididas, por lo que requieren funcionalidad adicional y mayor buffer. No es posible dividir la conexión si se usa IPSec para encriptar los paquetes de datos e incluso si se usa el enfoque de división de conexión deben de existir gateways

enrutadores tanto en la trayectoria de transmisión como de recepción de la conexión [Barakat *et al.*, 2000].

III.4.3.1.1 Soluciones de último salto

Se supone que el enlace inalámbrico es el último salto hacia el receptor que tiene una conexión dividida entre el usuario inalámbrico y el gateway enrutador aunque esta propuesta tiene que ser usada en conjunto con el enfoque de conexión dividida. La ventaja que ofrece es que sólo un subconjunto de las funcionalidades de la capa de transporte son requeridas, ya que como no hay enrutadores intermedios y el acceso al enlace inalámbrico es regulado por el protocolo MAC, no hay necesidad de los mecanismos de control de congestión. Debido a esta limitante en su funcionalidad, las soluciones de último salto de capa de transporte no pueden ser utilizadas en el Internet en general. Algunos ejemplos de estas propuestas son el TCP-Móvil, el Protocolo de Transporte de Extremo-Móvil y el M-TCP. Ninguna de ellas incluye mecanismos de control de congestión explícita, se enfocan en estrategias de retransmisión eficientes, en ahorrar ancho de banda del enlace inalámbrico a través de compresión de encabezados y en ahorrarle energía de procesamiento al dispositivo móvil al reubicar las funciones complejas de la comunicación en la estación base.

III.4.4 Protocolos de transporte alternativos

Existen una gran cantidad de protocolos de transporte alternativos que han sido desarrollados para los ambientes satelitales o inalámbricos. Estos protocolos proveen las funcionalidades de TCP como establecimiento y finalización de conexión, confiabilidad y control de congestión, de ahí que podrían ser reemplazos de TCP aun en Internet y se evitaría el enfoque de conexión dividida. Sin embargo el modificar o actualizar todas las computadoras de Internet es una tarea difícil, por lo que los protocolos de transporte alternativos podrían ser usados solamente en el segmento inalámbrico. Se mencionan cuatro protocolos principales.

III.4.4.1 Protocolo de Transporte – Especificaciones de Protocolo de Comunicaciones Espaciales

El Protocolo de Transporte – Especificaciones de Protocolo de Comunicaciones Espaciales (SCPS-TP) es parte de todo un nuevo protocolo para comunicación en ambientes inalámbricos; satelitales y espaciales, y ha sido desarrollado por el Departamento de Defensa de los Estados Unidos y la ITU [Durst *et al.*, 1997]. Para su creación tomaron mucho cuidado en incluir mecanismos que puedan enfrentar todos los retos que presentan este tipo de enlaces, como es el alto BER. El control de congestión Vegas se usa porque no utiliza pérdida de paquetes como su indicador primario de congestión. El SCPS-TP (por sus siglas en inglés) incluye un parámetro configurable para indicar la fuente primaria de pérdida de paquetes (congestión o corrupción). Dependiendo del tipo de notificación que se reciba, ya sea congestión o corrupción, es el valor que tomará.

El SCPS-TP también difiere de TCP en la manera en que indica las pérdidas de paquetes y cómo maneja las retransmisiones. Utiliza reconocimientos selectivos negativos (SNACKs) de longitud variable para señalarle al transmisor los agujeros que existen en el espacio de buffer receptor. El transmisor trata la recepción de un SNACK como una petición inmediata de retransmisión y retransmite todos los paquetes especificados en el SNACK. No tiene implementado el mecanismo de Retransmisión Rápida, debido a que con reconocimientos con alto retardo es poco probable que reciba el número umbral de reconocimientos duplicados necesarios para entrar a Retransmisión Rápida.

SCPS-TP propone el uso de reconocimientos con alto retardo, posiblemente tan raro como un reconocimiento por tiempo de viaje completo. Se implementa como una solución para enlaces asimétricos para comenzar la reducción en el crecimiento de la ventana de congestión el conteo de bytes. El escalamiento de la ventana de recepción, como se especifica para TCP, se utiliza en SCPS-TP para permitir el crecimiento necesario de ventana para ambientes de alto retardo. Los valores mínimos y máximos para la expiración de la retransmisión también son alterados para reflejar los retardos más altos posibles [Durst *et al.*, 1997].

III.4.4.2 Protocolo de Transporte Satelital

El Protocolo de Transporte Satelital (STP, por sus siglas en inglés) [Henderson y Katz, 1997] es un protocolo de transporte diseñado especialmente para ser usado en enlaces satelitales y redes satelitales LEO. Tiene muchos conceptos que son bastante similares a SCPS-TP, por ejemplo utiliza reconocimientos con alto retardo para enfrentar la asimetría del canal. Sin embargo, distinto a SCPS-TP, STP utiliza el control de congestión Reno e incluye la entrega en ritmo para dispersar la transmisión de datos en ráfagas. La estrategia de control de congestión en STP no incluye mecanismos para distinguir entre pérdidas por congestión y pérdidas por corrupción, por lo que todas las pérdidas son tomadas como un indicador de congestión de la misma manera que sucede en TCP [Hassan y Jain, 2004].

Para la señalización de paquete perdido STP propone utilizar SNACKs al igual que SCPS-TP. Existen dos tipos de STP SNACKs: uno donde la información que el SNACK envía de manera poco frecuente cubre todo el espacio de buffer de recepción, y otro donde los SNACKs son enviados para cubrir un espacio de buffer que se acaba de presentar. Ambas formas de SNACKs son peticiones para retransmisión inmediata porque la Retransmisión Rápida fallaría con reconocimientos con alto retardo. Las pérdidas detectadas por el receptor se hacen por medio de SNACKs.

III.4.4.3 TCP inalámbrico

El TCP Inalámbrico [Sinha *et al.*, 1999 y Hassan y Jain, 2004] es un protocolo de transporte diseñado para el uso en redes de área amplia inalámbricas (W-WANS). Su mecanismo de control de congestión no está basado completamente en TCP. No incluye ni Inicio Lento ni Evasión de Congestión y la transmisión de datos se controla por tasa en lugar de utilizar una ventana de congestión. WTCP determina su tasa de transmisión inicial por medio del enfoque de paquete en pareja, y de ahí actualiza su tasa usando un mecanismo basado en retardos interpaquete. El transmisor envía el valor del retardo entre cada par de paquetes y el receptor compara este retardo interpaquete del transmisor con el retardo interpaquete que observa en la recepción de paquetes. Cuando estos valores son aproximadamente iguales, la tasa de transmisión se incrementa, pero si el retardo interpaquete es mucho mayor que el retardo con el cual los paquetes fueron transmitidos, la

tasa se disminuye. Esta comparación y cálculo de tasa de transmisión se hace en el receptor y se envía al transmisor por medio de los reconocimientos.

Para proporcionar confiabilidad WTCP utiliza un mecanismo basado en reconocimientos selectivos (SACK), sin expiraciones como lo hace STP. Para trabajar en canales asimétricos los reconocimientos y reconocimientos selectivos son enviados de manera poco frecuente y varios a la vez para evitar inundar el canal. WTCP también incluye un método para distinguir entre pérdida por congestión y pérdida por corrupción, la cual también se basa en retardos interpaquete. Cuando se detecta una pérdida de paquete, el receptor calcula el retardo interpaquete de los paquetes perdidos. Si el resultado es cercano al retardo interpaquete que normalmente se observa en el receptor, se asume que las pérdidas son debidas a corrupción, de lo contrario se asume que son debidas a congestión y la tasa de transmisión se disminuye a la mitad.

III.4.4.4 Protocolo de Inspección y Espera

Otro protocolo de transporte que ha sido específicamente desarrollado para canales inalámbricos es el Protocolo de Inspección y Espera (Wave and Wait Protocol). En primera instancia fue desarrollado para redes de bajo ancho de banda y dispositivos móviles pequeños con energía de batería limitada [Tsaoussidis *et al.*, 2000]. Al igual que WTCP, el mecanismo de control de congestión de WWP trata de estimar la congestión actual de la red del lado del receptor, y para lograr esto el receptor necesita conocer el patrón de transmisión del lado transmisor. Los transmisores WWP transmiten grupos de paquetes llamados de inspección. El protocolo define diferentes niveles de inspección. Para un nivel de inspección alto el número de paquetes transmitido por inspección es mayor que para las inspecciones previas. El receptor siempre decide sobre el nivel de inspección basandose en el caudal eficaz que se ha observado durante la última inspección y transmite esta información de regreso al transmisor en los reconocimientos. Para retransmisión WWP utiliza SNACKs que especifican los espacios en el buffer de recepción como sucede en SCPS-TP. Los SNACKs solo se transmiten cuando una inspección completa ha sido recibida o después de una expiración, si los paquetes de la inspección se pierden o están muy retardados. No existe mecanismo para distinguir entre pérdidas por congestión y

pérdidas por corrupción, pero cuando el número de paquetes perdidos por inspección es muy alto, sin importar la causa, el nivel de inspección se fija a cero y comienza un ciclo de sondeo. El transmisor y receptor intercambian un conjunto de sondeos y los receptores utilizan estos sondeos para determinar el tiempo real de viaje completo. Cuando el ciclo de sondeo se completa exitosamente, el receptor decide el nuevo nivel de inspección basándose en el tiempo de viaje completo real observado.

III.5 Variantes de TCP

En los puntos III.1 hasta III.3 se introdujeron los conceptos básicos de operación del protocolo TCP original, y aunque al inicio fue capaz de dar soporte al crecimiento continuo y sostenido de Internet, comenzó a presentar algunas fallas en su mecanismo de control de congestión. De manera histórica, los primeros momentos de gran congestión en la red se comenzaron a presentar a mediados de la década de los 1980s. Esto condujo al diseño de nuevos algoritmos de control de congestión que ahora forman una parte esencial de TCP. En el trabajo [Van Jacobson, 1988] se presentó el desarrollo del algoritmo de control de congestión básico de TCP conocido como TCP Tahoe, y desde entonces un número importante de variaciones a la fase de inicio lento, de evasión de congestión o de respuesta ante pérdidas, se han venido presentando. A continuación se hablará más en detalle sobre cómo funcionan las variantes de TCP más representativas.

III.5.1 TCP Tahoe

Como se mencionaba anteriormente, a mediados de los 1980s el TCP original fue aumentado con una estrategia de control de congestión, implementación conocida como Tahoe. El TCP Tahoe utiliza dos variables para realizar el control de congestión, cwnd es Congestion Window o Ventana de Congestión y Ssthresh es Slow Start Threshold o umbral de inicio lento. Cwnd regula la cantidad de datos que generalmente le es permitida transmitir a una conexión, mientras que ssthresh determina en cuál fase del procedimiento de control de congestión se encuentra la conexión. El mecanismo de congestión de Tahoe consiste de dos fases: Inicio Lento y Evasión de Congestión. El inicio lento se ejecuta al comienzo de la conexión y después de cada pérdida de paquete, mientras que la fase de

evasión de congestión se presenta cuando cwnd alcanza el valor de ssthresh. Durante inicio lento la ventana de congestión (cwnd) es igual a uno, y se incrementa en uno por cada reconocimiento (ACK) recibido, con lo cual se duplica la ventana de congestión cada Tiempo de Viaje Completo (RTT) y presenta un crecimiento exponencial. En evasión de congestión la ventana de congestión (cwnd) solamente se incrementa 1/cwnd por cada reconocimiento (ACK) recibido, lo que conduce a un incremento lineal de uno por cada Tiempo de Viaje Completo (RTT).

Se propuso un mecanismo de retransmisión llamado Retransmisión Rápida [Hassan y Jain, 2004] como adición a Tahoe. En Retransmisión Rápida se toma un número umbral de reconocimientos duplicados (dupacks) como indicación de pérdida y se retransmite el paquete inmediatamente.En el caso de que no lleguen suficientes reconocimientos duplicados el transmisor solo espera a que expire el temporizador de retransmisión para comenzar a transmitir los paquetes perdidos. El número umbral de reconocimientos duplicados es tres, y fue escogido así para prevenir las retransmisiones falsas causadas por la entrega en desorden. Después de una pérdida de paquete, la ventana de congestión se fija a uno y se regresa a inicio lento, mientras que el umbral de inicio lento se fija a la mitad del tamaño de la ventana de congestión cuando ocurrió la pérdida, lo cual se considera como una tasa de transmisión segura.

III.5.2 TCP Reno

TCP Reno retiene todas las mejoras de TCP Tahoe, pero incorpora un nuevo algoritmo llamado algoritmo de Recuperación Rápida, el cual está basado en el hecho de que un reconocimiento duplicado (dupack) indica que un segmento ya ha dejado la red. Por lo tanto, cuando el transmisor recibe tres dupacks retransmite el segmento perdido, actualiza el ssthresh, y reduce cwnd como lo hace en retransmisión rápida a la mitad. La Recuperación Rápida lleva la cuenta del número de dupacks recibidos y trata de estimar la cantidad de datos que aún están pendientes en la red, de ahí que infla cwnd en un segmento por cada dupack recibido, con lo que el flujo de tráfico se mantiene, y también de la misma forma mantiene vivo el mecanismo de autocronómetro de TCP. El transmisor sale de Recuperación Rápida cuando recibe un reconocimiento (ACK) por el segmento cuya

pérdida resultó en los ACKs duplicados, con lo que TCP entonces desinfla la ventana regresándola a ssthresh y entra en la fase de evasión de congestión. Si se pierden múltiples segmentos dentro de la misma ventana de datos, TCP Reno espera por una expiración de retransmisión, retransmite el segmento y se va al modo de inicio lento. Esto pasa cuando Reno entra a Recuperación Rápida por cada segmento,, reduciendo su ventana de congestión, y aborta Recuperación Rápida cuando recibe un ACK parcial que reconoce algunos pero no todos los segmentos pendientes. Después de las múltiples reducciones, la ventana de congestión se vuelve tan pequeña que no hay suficientes dupacks para que ocurra la Recuperación Rápida y lo último que queda es una expiración de temporizador.

III.5.3 TCP NewReno

TCP NewReno contiene cambios tanto para el control de congestión como para el mecanismo de retransmisión de TCP Reno. TCP NewReno extiende el mecanismo de Recuperación Rápida de Reno. Cuando se detecta una pérdida de paquetes mediante el empleo de Retransmisión Rápida, se recuerda el número de secuencia más alto transmitido hasta entonces y se entra a Recuperación Rápida. La Recuperación Rápida solamente finaliza cuando se recibe el reconocimiento por el número de secuencia más alto enviado antes de la pérdida. Durante una Recuperación Rápida NewReno las pérdidas adicionales son detectadas por la recepción de reconocimientos parciales. Un reconocimiento parcial es un reconocimiento para datos nuevos pero con un número de secuencia menor que el más alto paquete de datos transmitido antes de la pérdida. Empleando éste mecanismo TCP NewReno es capaz de recuperar múltiples paquetes perdidos por ventana de congestión sin una expiración. Un paquete perdido es retransmitido cada tiempo de viaje completo. Este comportamiento modificado mejora el caudal eficaz comparado con TCP Reno, debido a que Reno entra a varias Recuperaciones Rápidas consecutivas si se pierden múltiples paquetes por ventana, cortando la ventana de congestión varias veces o aún peor, tienen que esperar una expiración para recuperar los paquetes perdidos.

III.5.4 TCP SACK

Una manera de tratar los múltiples segmentos perdidos es informar al transmisor cuáles segmentos han llegado al receptor. El Reconocimiento Selectivo (SACK) es una versión de TCP que adopta este enfoque [Mathis *et al.*, 1996]. El receptor usa cada bloque TCP SACK para indicar al transmisor que un bloque contiguo de datos ha sido recibido fuera de orden en el receptor. Cuando los bloques SACK son recibidos por el transmisor, son usados para mantener una imagen de la cola del receptor, indicando cuáles segmentos se perdieron y cuáles han llegado al receptor. Utilizando ésta información, el transmisor retransmite sólo aquellos segmentos que se han perdido, sin esperar a una expiración de retransmisión, y sólo cuando ningún segmento requiere ser retransmitido se envían nuevos datos.

La implementación SACK puede todavía utilizar los mismos algoritmos de control de congestión que Reno o NewReno. Recurre al mecanismo de expiración de retransmisión para entregar un segmento perdido al receptor si los reconocimientos aún no se reciben a tiempo. La principal diferencia entre SACK y Reno es el comportamiento en el evento de múltiples pérdidas de segmento. En SACK, igual que en Reno, cuando el transmisor recibe tres reconocimientos duplicados o dupacks entra a Recuperación Rápida, el transmisor retransmite el segmento y disminuye a la mitad la ventana de congestión. Para indicar el número de segmentos pendientes que aún están en tránsito, SACK mantiene una variable llamada "pipe" (tubería). Durante la Recuperación Rápida, el transmisor envía datos (nuevos o retransmitidos) sólo cuando el valor que tiene "pipe" de segmentos en tránsito es menor que el valor de la ventana de congestión. El valor de "pipe" se incrementa en uno cuando el transmisor envía un segmento (nuevo o retransmitido) y disminuye en uno cuando el transmisor recibe un reconocimiento duplicado con SACK mostrando que han recibido nuevos datos. El transmisor disminuye el valor de "pipe" en dos cada vez que recibe reconocimientos parciales. Como sucede en NewReno, la Recuperación Rápida termina cuando un reconocimiento que llega reconoce todos los datos, incluyendo aquellos que estaban pendientes cuando comenzó el procedimiento de Recuperación Rápida.

III.5.5 TCP Vegas

La variante Vegas de TCP fue presentada en 1994 antes que NewReno y SACK fueran desarrollados [Brakmo *et al.*, 1994]. Vegas es fundamentalmente diferente de otras variantes de TCP en que no espera por pérdidas para disparar reducciones de la ventana de congestión. En Vegas el caudal eficaz esperado de una conexión se estima para que sea el número de segmentos en el "pipe", que es el número de bytes viajando del transmisor al receptor. Para incrementar el caudal eficaz, se debe incrementar la ventana de congestión. Si existe congestión en la red, el caudal eficaz real es menor que el caudal eficaz esperado,lo que usa Vegas para decidir si debiera incrementar o disminuir la ventana. Esta variante de TCP lleva la cuenta del tiempo en que se envía cada segmento. Cuando llega un reconocimiento, estima el tiempo de viaje completo (RTT) como la diferencia entre el tiempo real y el tiempo guardado de cuándo fue enviado el segmento. Vegas define una variable llamada *BaseRTT* para cada conexión para que sea el mínimo RTT visto hasta entonces. Después calcula el caudal eficaz esperado como:

$$CaudalEsperado = TamañodelaVentana / BaseRTT \tag{17}$$

donde *Tamaño de la Ventana* es el tamaño de la ventana de congestión actual. Este entonces calcula el caudal eficaz real (cada tiempo de viaje completo) midiendo el RTT para un segmento en particular en la ventana y los bytes transmitidos entre ellos. La diferencia calculada entre los caudales eficaces esperado y real es mantenida en la variable *Diff*. Si *Diff* es menor que un parámetro α, se incrementa linealmente la ventana de congestión en el siguiente RTT. Si *Diff* excede el parámetro β, la ventana de congestión se disminuye linealmente en el siguiente RTT. Los factores α y β (usualmente fijados a 2 y 4) representan muy pocos o demasiados datos en la red, respectivamente. Lo anterior representa la manera como funciona el esquema de evasión de congestión en Vegas [Hassan y Jain, 2004].

El inicio lento y la evasión de congestión requieren pérdidas para darse cuenta del comienzo de congestión en la red. Vegas hace uso de un algoritmo de inicio lento modificado donde trata de encontrar el tamaño de ventana correcto sin incurrir en una

pérdida. Esto lo logra incrementando exponencialmente su ventana cada segundo RTT y utilizando el primer RTT para calcular *Diff*, cuando no existe cambio en la ventana de congestión. Para realizar el cambio de inicio lento a evasión de congestión, Vegas hace una comparacion del valor que tiene guardado la variable γ, y lo realiza si el caudal eficaz es menor que el caudal eficaz esperado. Esta adición de detección de congestión del inicio lento da un mejor estimado del ancho de banda disponible para la conexión. El TCP Vegas también modifica la política de retransmisión, y define que un segmento será retransmitido después de un reconocimiento duplicado (sin esperar por los tres reconocimientos duplicados) si el tiempo de viaje completo estimado es mayor que el valor de expiración. Esto ayuda en aquellos casos donde el transmisor nunca recibirá tres dupacks porque una cantidad de segmentos dentro de ésta ventana están perdidos, o el tamaño de ventana es muy pequeño. La misma estrategia se aplica para un ACK no duplicado después de una retransmisión.

III.5.6 TCP Westwood

La innovación clave de TCP Westwood es el uso de la estimación del ancho de banda para conducir de manera directa la ventana de congestión y el umbral de inicio lento [Mascolo *et al.*, 2001 y Hassan y Jain, 2004]. El TCP está monitoreando continuamente los reconocimientos que le llegan del receptor y calcula el Estimado de Tasa Elegible (ERE, por sus siglas en inglés), el cual se basa en la tasa de reconocimientos y en su carga útil. En lo referente a la indicación de una pérdida de paquete (tres dupacks o una expiración), el transmisor fija la ventana de congestión y el umbral de inicio lento basándose en el ERE. TCP Westwood fue mejorado y se desarrolló TCP Westwood para Tasa Mayor (TCPW BK, por sus siglas en inglés), el cual diferencía entre pérdida de congestión y pérdida de transmisión utilizando una combinación de mediciones de RTT y el hueco entre las tasas esperadas y logradas. Cuando se determina que las pérdidas son debidas a errores, TCP BK utiliza tres nuevos mecanismos: Repetición en Masa, Expiración de Retransmisión Fija, y Ajuste de Ventana Inteligente. En la Repetición en Masa, cuando se recibe un ACK parcial, el transmisor de TCP retransmite todos los paquetes pendientes en la ventana de congestión real en lugar de enviar solo un paquete perdido, como sucede en NewReno. Esto podría

ahorrar multiples RTTs necesarios para recuperar todos los paquetes perdidos en la ventana. En la Expiración de Retransmisión Fija, si hay expiraciones consecutivas y no hay entrega exitosa de paquete, el siguiente intervalo de expiración se mantiene igual al anterior en lugar de duplicarlo como sucede en NewReno. Este cambio mejora el desempeño cuando el BER es alto. En el Ajuste de Ventana Inteligente, la ventana de congestión no se reduce al umbral de inicio lento después de una pérdida. Aparte de las mejoras de desempeño, otra ventaja de TCP Westwood y TCP Westwood para Tasa Mayor es que las modificaciones se requieren sólo en el transmisor de TCP.

III.6 Conclusiones

En este capítulo se analizaron los mecanismos que componen al protocolo TCP como fue diseñado originalmente. También se vieron algunas propuestas que se han hecho a mecanismos de control de congestión para su uso en ambientes satelitales, tanto soluciones experimentales como respaldadas por la IETF, y por último se analizaron algunas de las variantes conocidas de TCP. En el siguiente capítulo se establecerán los modelos de variantes de TCP que se usaron para este trabajo de tesis, y cómo se implementaron en el simulador de redes network simulator 2 (ns2).

IV. MODELADO DE VARIANTES DE TCP Y DE PARAMETROS DE QoS

En el capítulo anterior se analizó el protocolo TCP y los mecanismos de funcionamiento interno que lo componen. También se presentaron las distintas soluciones propuestas para obtener un desempeño adecuado cuando este protocolo sea utilizado en satélites, así como las variantes de mecanismos de control de congestión de TCP mas representativas. En este capítulo se presenta un análisis de los modelos de algunas variantes de TCP, así como de los principales parámetros de Calidad de Servicio con los cuales se evaluará el desempeño de cada una de estas variantes al ser utilizadas en un ambiente satelital. Para lograr esto se deducirán expresiones y fórmulas matemáticas que permitan modelar y evaluar los parámetros de Calidad de Servicio en satélites.

IV.1 Modelado de TCP

Para comenzar, es importante considerar las razones por las que puede ser de interés el modelado del protocolo TCP. Por un lado, para aquellas personas con una orientación hacia las matemáticas esto presenta la oportunidad de utilizar modelos matemáticos para capturar las dinámicas de TCP, mientras que para alguien involurado en el desarrollo del protocolo o escribiendo código que implemente TCP esto no les presenta el mayor interés. Al tener un conocimiento profundo de los mecanismos internos de TCP, no requieren de técnicas matemáticas para explicar como trabaja el protocolo. Un aspecto que respalda la importancia de utilizar el modelado es que TCP es el protocolo de transporte que opera en la mayoría de las aplicaciones y sistemas que integran la Internet, uno de los sistemas mas grandes y complejos hechos por el hombre, surgiendo la oportunidad de utilizar modelos matemáticos para analizarlo en su magnitud y dimensión. Otra razón es que se presentan muchos aspectos desconocidos del ambiente en que trabaja TCP, los que incluyen el tamaño de los archivos que se transfieren por las conexiones de TCP o el número de conexiones en existencia. Existen además en Internet otros protocolos de transporte que pueden forzar a TCP a responder de maneras que no se pueden anticipar. La red misma

puede presentar cambios en sus tablas de enrutamiento o fallas en sus enlaces, lo que resulta en que los datos sean transportados a través de trayectorias alternativas. Esta incertidumbre puede ser representada mediante procesos estocásticos para tratar de obtener la respuesta de TCP ante esto. Un punto importante es que TCP no fue construido utilizando optimización y mientras que su diseño y desempeño fue adecuado para ciertas condiciones, cuando estas cambian, se vuelve necesario hacerle algunas "adecuaciones" o "parches" que le permitan seguir trabajando. Dado que la popularidad de la que goza le impide ser desechado para adoptar un nuevo protocolo de transporte, a TCP se le conoce como un "protocolo de parches" y un ejemplo de eso son los algoritmos de Retransmisión Rápida y Recuperación Rápida. Por lo expuesto si alguien está interesado en diseñar protocolos de transporte que logren un desempeño óptimo, los modelos matemáticos son requeridos para cuantificar métricas que definan el desempeño del sistema y que desarrollen estrategias de control que resulten en el óptimo logrado. Al modelar el protocolo actual, se puede determinar qué tan cercano está de lograr el óptimo y se pueden identificar cuestiones que requieran ser tratadas en protocolos de control futuros.

A continuación se mostrará un modelo utilizado para explicar la dinámica de TCP. Cabe mencionas que muchos de los modelos se enfocan a la variante Reno y a sus modificaciones mas notables Newreno y SACK, debido a que el interés en modelar TCP ha coincidido con el crecimiento que ha experimentado Internet. Al ser el protocolo de transporte dominante, Newreno y SACK tienen características particulares que han contribuido a su expansión y popularidad. Existen dos procesos clave que un modelo de TCP necesita incluir: a) Las dinámicas de la ventana que define el número de paquetes que una fuente de TCP puede transmitir en la red, y b) El proceso de pérdida de paquetes que indica cargas de tráfico actuales o congestión dentro de la red. Ambos procesos son observados desde el punto de referencia de la fuente TCP.

El símbolo típico para el tamaño de ventana actual es W(t) y las dinámicas esenciales de éste tamaño de ventana son su incremento lineal y su disminución multiplicativa. Las implementaciones de TCP normalmente incrementan la ventana en un paquete cada viaje completo, que es la fase de incremento lineal, y reducen el tamaño de ventana a la mitad cuando hay pérdida. Algunos modelos han sido desarrollados usando la

tasa de transmisión de paquete X(t) y con esto facilitan el análisis que sigue al desarrollo del modelo. La suposición que surge es que el tamaño de ventana está relacionado con la tasa de transmisión por medio del tiempo de viaje completo (RTT):

$$X(t) = \frac{W(t)}{RTT} \tag{18}$$

Sin importar si un modelo usa la variable de estado de tamaño de ventana o tasa de transmisión, todos los modelos incorporan las dinámicas del incremento lineal y decremento multiplicativo de TCP.

El otro componente principal de un modelo TCP es el proceso de pérdida de paquete, el cual hace que la fuente TCP reduzca su tamaño de ventana Este proceso agrega información concerniente a las condiciones de red respecto de todos los nodos a lo largo de la trayectoria de la conexión TCP. La conexión TCP que está siendo considerada está compitiendo por recursos de red a lo largo de la trayectoria con otras conexiones TCP que tienen rutas que se intersectan con su trayectoria, a la vez que también compite por ancho de banda con otro tráfico en general. Estas variaciones en la carga de tráfico introducen incertidumbre sobre la información del arribo de paquetes perdidos en fuente de TCP. Esto es modelado típicamente como un proceso estocástico, tomando en cuenta la probabilidad p de perder un paquete en particular en la red o los intervalos entre instancias cuando los paquetes perdidos son detectados. El punto clave es que los modelos incorporan la información de los paquetes perdidos, con la fuente de TCP respodiendo con la disminución de ventana.

IV.1.1 Modelo Periódico

El modelo mas sencillo para TCP considera un patrón periódico en la dinámica de la ventana de congestión y el proceso de pérdida de paquete en estado estable. No se asume una versión específica de TCP y la ventana evoluciona de acuerdo a dinámicas genéricas comunes a las distintas variantes de TCP en las cuales se presenta una reducción multiplicativa de ventana con cada evento periódico de pérdida. Debido a que se analiza el estado estable, se logra siempre un tamaño máximo de ventana W al momento en que se pierde un paquete, lo que resulta en la reducción de ventana a W/2. La operación del

sistema en estado estable significa que las pérdidas de paquetes ocurren con probabilidad constante p, por lo que se transmiten $1/p$ paquetes en promedio sobre la red por cada paquete perdido (incluyendo una reducción simultánea en el tamaño de ventana).

Hay que mencionar que ya que se consideró TCP en estado estable, por lo que los estados transitorios como inicio lento no aparecen en la representación de las dinámicas de la ventana TCP. Estado estable significa que se considera que las dinámicas de la ventana han alcanzado un patrón regular que continua por todo el tiempo, por lo que , cuando se considera el desempeño logrado en el largo plazo por el control de flujo TCP, se pueden ignorar los transitorios que ocurren durante el período de inicio. En la Figura 14 se puede apreciar la periodicidad del diente de sierra que se obtiene a partir del tamaño de ventana.

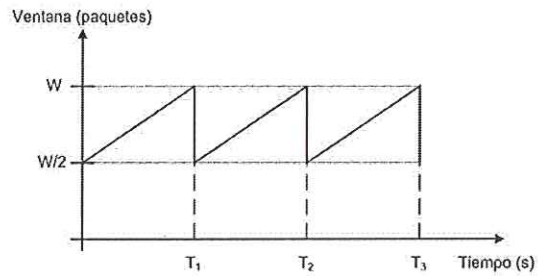


Figura 14. Modelo periódico de las dinámicas de la ventana de TCP en estado estable [Hassan y Jain, 2004].

Si ocurren pérdidas de paquetes aleatoriamente, lo cual es más realista para TCP, una gráfica de la evolución de la ventana no tendría una forma tan perfecta como la anterior pero en cambio tendría posiblemente la de la Figura 15.

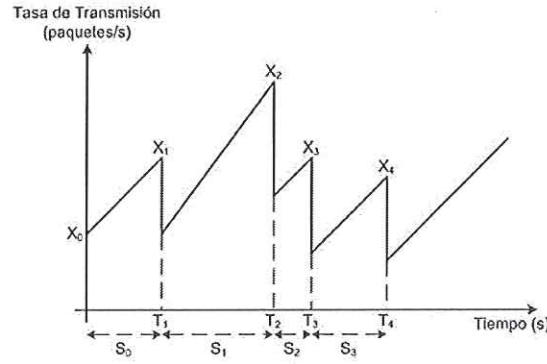


Figura 15. Dinámica de la tasa de transmisión de TCP [Hassan y Jain, 2004].

A pesar de que el modelo periódico es menos probable, permite derivar una propiedad fundamental con respecto al desempeño de TCP, la cual se conoce como la *Ley Raíz Cuadrada Inversa p*. Debido a la forma periódica de la ventana, el desempeño de una conexión TCP se puede derivar determinando el número de paquetes transmitidos en un período y la longitud del período. Como se mencionó antes, se tiene la probabilidad de pérdida de paquete constante p, de ahí el número de paquetes está dado por $1/p$, ya que se tiene que transferir este número de paquetes antes de que se pierda un paquete.

Una manera de encontrar el número total de paquetes transferidos en un período es encontrar el área debajo de la curva del tamaño de ventana, lo cual se obtiene de:

$$Número\ de\ paquetes = \frac{1}{2}\frac{T}{RTT}\left(\frac{W}{2}+W\right) \tag{19}$$

donde T es el período entre la detección de pérdida de paquetes.

El tiempo debe ser escalado por el tiempo de viaje completo, ya que la fuente de TCP no envía paquetes continuamente, y el tiempo debe avanzar por un período de viaje completo antes de que un nuevo conjunto total de paquetes, contenido por el tamaño de ventana, contribuya al número global de paquetes transmitidos. Ya que se sabe que TCP incrementa su ventana a la tasa de un paquete por tiempo de viaje completo durante su fase de incremento lineal, el tiempo tomado para incrementar su ventana de $W/2$ a W es $T = RTT * W/2$. De ahí, se tiene que igualando el número total de paquetes transmitidos durante este período:

$$\frac{W}{4}\left(\frac{W}{2}+W\right)=\frac{1}{p} \tag{20}$$

Despejando W se obtiene:

$$W=\sqrt{\frac{8}{3p}} \tag{21}$$

Ahora se puede encontrar la tasa de envío promedio $\bar{X}(p)$ de la fuente TCP, la cual es el número de paquetes transmitidos durante cada período:

$$\bar{X}(p)=\frac{1/p}{RTT\cdot W/2}=\frac{1}{RTT}\sqrt{\frac{3}{2p}} \tag{22}$$

Este resultado es conocido como la *Ley de Raíz Cuadrada Inversa p*, y es una relación importante y bien conocida que gobierna el desempeño de TCP. Muestra que la tasa de transmisión de una fuente TCP está relacionada inversamente al tiempo de viaje completo y la raíz cuadrada de la probabilidad de pérdida de paquete promedio.

IV.1.2 Modelo detallado de Pérdida de Paquete

En esta sección se considera un modelo más detallado de TCP que incorpora elementos expuestos anteriormente, pero que agrega varios aspectos que vuelven esto más realista, como los reconocimientos duplicados y las expiraciones. Se considera un modelo estocástico de TCP, donde la información concerniente a las pérdidas de paquetes está dada solamente por reconocimientos triple duplicados. El enfoque es aquel de la sección anterior, donde se calcula el número de paquetes *Yi* transferidos en el período de longitud *Ai* entre pérdidas de paquetes. Como el interés está enfocado hacia la tasa de transmisión de estado estable de la fuente, el objetivo es encontrar:

$$\bar{X} = \frac{E[Y]}{E[A]} \tag{23}$$

Si αi es el número del paquete perdido en el período *i*-ésimo, se tiene $Y_i = \alpha_i + W_i - 1$ como el número total de paquetes enviados en este período, lo que se puede ver en la Figura 16.

Se asume que las pérdidas de paquete ocurren independientemente con probabilidad *p*, así que la probabilidad de que *k-1* paquetes sean enviados antes de que se pierda un paquete está dada por:

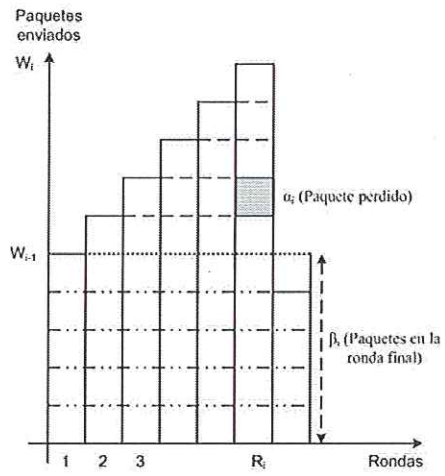


Figura 16. Análisis detallado de los paquetes enviados durante cada ronda de incremento de ventana [Hassan y Jain, 2004].

$$P[\alpha = k] = (1-p)^{k-1} p \tag{24}$$

Así el número de paquetes que se pierde es:

$$E[\alpha] = \sum_{k=1}^{\infty} k \cdot (1-p)^{k-1} p \tag{25}$$

$$E[\alpha] = \frac{1}{p}$$

Se puede usar este resultado para simplificar la expresión para el número esperado de paquetes transmitidos por la fuente:

$$E[Y] = E[\alpha] + E[W] - 1$$

$$E[Y] = \frac{1-p}{p} + E[W] \tag{26}$$

Esta sería la primera expresión para $E[Y]$, y sigue encontrar otra expresión para el número esperado de paquetes, lo cual puede hacerse determinando el número de paquetes que son enviados, ya que la ventana misma es incrementada por los reconocimientos que regresan a la fuente, y para esto se necesita introducir el concepto de rounds o rondas. Ya que cada ronda marca una unidad de incremento al tamaño de ventana, el tamaño final de ventana *Wi* está dado por:

$$W_i = \frac{W_{i-1}}{2} + R_i \tag{27}$$

donde R_i es el número total de rondas en el i-ésimo período. Calculando el valor esperado a ambos lados se tiene:

$$E[W] = 2E[R] \tag{28}$$

Si se consideran los paquetes contados por cada ronda se puede encontrar una segunda expresión para el número total de paquetes transmitidos:

$$Yi = \sum_{k=0}^{R_i - 1} \left(\frac{W_{i-1}}{2} + k \right) + \beta_i$$

$$= \frac{R_i W_{i-1}}{2} + \frac{R_i}{2}(R_{i-1}) + \beta_i \tag{29}$$

donde β_i es el número de paquetes enviados en la última ronda.

Si se asume que el número de rondas en un período es independiente del tamaño de ventana, se puede tomar el valor esperado de cada cantidad de manera individual de la ecuación (29), y se obtiene la siguiente:

$$E[Y] = \frac{E[R]}{2}\left(\frac{E[W]}{2} + E[W] - 1 \right) + E[\beta] \tag{30}$$

Usando la suposición de que el número de paquetes enviados en la última ronda está uniformemente distribuido entre uno y W_{i-1}, se tiene que $E[\beta] = E[W]/2$. Al comparar las dos expresiones para el número esperado de paquetes transmitidos se puede resolver para el tamaño de ventana esperado al final de cada período por medio de la siguiente ecuación:

$$\frac{1-p}{p} + E[W] = \frac{E[W]}{4}\left(\frac{E[W]}{2} + E[W] - 1 \right) + \frac{E[W]}{2} \tag{31}$$

de (31) se obtiene la ecuación cuadrática:

$$3(E[W])^2 - 6E[W] + \frac{8(p-1)}{p} = 0 \tag{32}$$

ya que $E[W] \geq 0$ se toma la raíz positiva:

$$E[W] = 1 + \sqrt{\frac{8}{3p} - \frac{5}{3}} \tag{33}$$

Al acercarse a cero la probabilidad de paquete perdido, se puede aproximar el tamaño de ventana esperado como $E[W] \approx \sqrt{\frac{8}{3p}}$. Utilizando las ecuaciones (28) y (33) se puede encontrar el número esperado de rondas por período $E[R]$ obteniendo la siguiente expresión:

$$E[R] = \frac{1}{2} + \sqrt{\frac{2}{3p} - \frac{5}{12}} \tag{34}$$

Para calcular la tasa de transmisión de la fuente TCP se requiere la duración esperada del período $E[A]$.

Se observa que el tiempo entre el envío del primer paquete en la ronda j y el reconocimiento recibido por ese primer paquete es un tiempo de viaje completo rtt_{ij}, de ahí que

$$Ai = \sum_{j=1}^{Xi+1} rtt_{ij} \tag{35}$$

donde el tiempo de viaje completo extra es el tiempo tomado para detectar la pérdida de paquete.

Al asumir que los tiempos de viaje completo son independientes del tamaño de ventana, y tomando la igualdad (35) se obtiene la siguiente expresión:

$$E[A] = (E[R]+1)E[rtt]$$

$$E[A] = \left(\frac{3}{2} + \sqrt{\frac{2}{3p} - \frac{5}{12}}\right) RTT \tag{36}$$

donde RTT representa el tiempo de viaje completo promedio. Para obtener el número de paquetes por segundo que es $\overline{X}(p)$ se sustituyen las ecuaciones (26), (33) y (36) en (23) obteniéndose lo siguiente:

$$\overline{X}(p) = \frac{E[Y]}{E[A]}$$

$$\bar{X}(p) = \frac{\frac{1-p}{p} + 1 + \sqrt{\frac{8}{3p} - \frac{5}{3}}}{RTT\left(\frac{3}{2} + \sqrt{\frac{2}{3p} - \frac{5}{12}}\right)} \tag{37}$$

De acuerdo a [Hassan y Jain, 2004] la ecuación (37) se puede expresar de la siguiente forma:

$$\bar{X}(p) \approx \frac{1}{RTT}\sqrt{\frac{3}{2p}} \tag{38}$$

la cual es la *Ley de la Raíz Cuadrada Inversa p* que se encontró en la sección del Modelo Periódico, pero la (37) es una expresión más precisa para la tasa de transmisión para probabilidades de pérdida moderadas. En la siguiente sección se toma en cuenta la expiración de temporizador.

IV.1.2.1 Expiraciones

Aún no se incluye el efecto de las temporizaciones, lo cual puede ocurrir frecuentemente. Si se recibe un número fijo de reconocimientos duplicados, se interpreta como que se perdió un paquete en particular. Varias implementaciones de TCP reducen a la mitad el tamaño de ventana después de recibir tres reconocimientos duplicados, mientras que si no reciben reconocimientos duplicados o recibe menos de tres la fuente determina que el paquete se perdió cuando expira el temporizador de retransmisión. El tamaño de ventana se reduce a uno y el umbral de inicio lento se fija a la mitad del tamaño de ventana antes de que ocurriese la pérdida de paquete, de ahí el paquete perdido se retransmite y el temporizador de retransmisión se restaura con un período doble al anterior de expiración. Por lo que las expiraciones de temporizador se adaptan al modelo de pérdida de paquete para demostrar el efecto que tienen sobre la reducción de la tasa de transmisión de la fuente TCP.

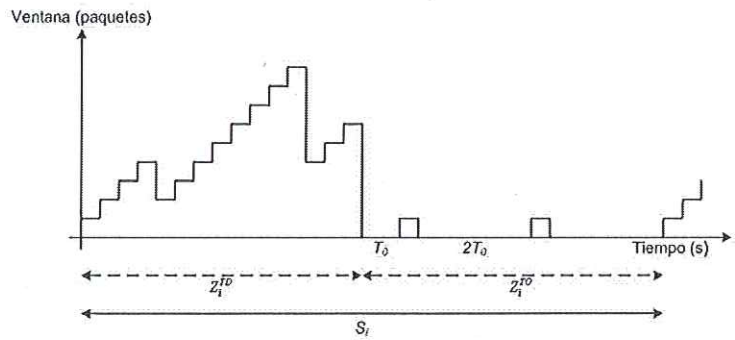


Figura 17. Intervalos que constituyen la dinámica de la ventana en un ciclo incluyendo período de expiración [Padhye *et al.*, 2000].

La Figura 17 muestra las caidas y cortes de los intervalos que forman el periodo S_i que son importantes para el modelo, de manera que se puede tener $Si = Z_i^{TD} + Z_i^{TO}$. El intervalo Z_i^{TD} incluye los periodos donde las pérdidas son recuperadas por reconocimientos duplicados, y Z_i^{TO} se refiere al periodo cuando la fuente solo recibe señales de expiración del temporizador de retransmisión. Al definir M_i como el número total de paquetes enviados durante un intervalo S_i, la tasa esperada de la fuente TCP está dada por:

$$\overline{X} = \frac{E[M]}{E[S]}$$

Resolviendo de manera similar a la sección anterior para los períodos de reconocimiento triple duplicado, y representando Y_{ij} como el número de paquetes enviados, A_{ij} la duración de los períodos donde las pérdidas son detectadas por reconocimientos duplicados y Y_i^{TO} como el número de paquetes enviados durante las expiraciones, se obtienen las siguientes expresiones:

$$Mi = \sum_{j=1}^{ni} Y_{ij} + Y_i^{TO} \tag{39}$$

$$Si = \sum_{j=1}^{ni} A_{ij} + Z_i^{TO} \tag{40}$$

en las que n_i representa el número de triple duplicados. Tomando que n_i es independiente del número de paquetes enviados en cada intervalo se obtienen las siguientes expresiones:

$$E[M] = E[n]E[Y] + E[Y^{TO}]$$
$$E[S] = E[n]E[A] + E[Z^{TO}] \quad (41)$$

Por cada S_i solo ocurre un período de expiración, y por esto, la probabilidad Q de que una pérdida de paquete resulte en un período de expiración está dada por $Q = 1/E[n]$ y se puede reescribir la tasa de envío como:

$$\bar{X} = \frac{E[Y] + Q \cdot E[R]}{E[A] + Q \cdot E[Z^{TO}]} \quad (42)$$

De acuerdo a lo expuesto por [Padhye *et al.*, 2000], se necesita determinar las cantidades de Q, $E[Y^{TO}]$ y $E[Z^{TO}]$, las cuales son:

$$Q(w) \approx \min(1, \frac{3}{w})$$
$$E[Y^{TO}] = \frac{1}{1-p} \quad (43)$$
$$E[Z^{TO}] = T_0 \frac{f(p)}{1-p}$$

El valor esperado del tamaño de ventana está dado por la expresión (33) y $f(p) = 1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6$. Al sustituir la (43) en la (42) la expresión completa para la tasa de transmisión de una fuente TCP es la siguiente:

$$\bar{X}(p) = \frac{\frac{1-p}{p} + E[W] + Q(E[W])\frac{1}{1-p}}{RTT(E[X]+1) + Q(E[W])T_0\frac{f(p)}{1-p}} \quad (44)$$

El desarrollo completo puede ser encontrado en [Padhye *et al.*, 2000]. Combinando los resultados de las secciones anteriores se puede obtener la caracterización completa de la tasa de transmisión de una fuente TCP como sigue:

Si el receptor limita el tamaño de ventana, la tasa de transmisión de la fuente TCP se expresa de la siguiente manera:

$$\bar{X}(p) = \frac{\frac{1-p}{p} + W_m + Q(E[W_m])\frac{1}{1-p}}{RTT(\frac{W_m}{8} + \frac{1-p}{pW_m} + 1) + Q(E[W_m])T_0\frac{f(p)}{1-p}} \quad (45)$$

donde W_m es el límite que el receptor impone en el tamaño de ventana; mientras que si no hay límite en el tamaño de ventana del receptor se tiene la siguiente expresión:

$$\bar{X}(p) = \frac{\frac{1-p}{p} + E[W] + Q(E[W])\frac{1}{1-p}}{RTT(\frac{E[W]}{2}+1) + Q(E[W])T_0\frac{f(p)}{1-p}} \tag{46}$$

Estas expresiones han sido usadas en el diseño de lo que llaman "algoritmos amigables de TCP" los cuales son protocolos de transporte diseñados para tráfico en tiempo real, como video y voz, y proveer de alguna manera tasas de transmisión con variaciones suaves. Estos algoritmos responden a tasas de pérdida de paquetes de largo plazo en una manera similar a TCP, de manera que los flujos en tiempo real no se roben el ancho de banda de otras conexiones TCP. Las principales referencias utilizadas para ésta sección son [Padhye *et al.*, 2000 y Hassan y Jain, 2004]. Sin embargo existen otros trabajos que complementan lo que se presentó como son [Mathis *et al.*, 1997], [Altman *et al.*, 2000], [Barakat, 2001 y Khalifa y Trajkovic, 2004].

IV.2 Modelos gráficos de variantes de TCP

A continuación se presentan los modelos gráficos que describen el funcionamiento de seis variantes de TCP que se analizaron en este trabajo de tesis.

IV.2.1 Variante TCP Tahoe

Esta variante es la más antigua y su implementación funciona de acuerdo a lo establecido en [Allman *et al.*, 1999]. Básicamente consiste de 2 fases, inicio lento y evasión de congestión. Inicio lento se encarga de incrementar la ventana de congestión y de realizar su respectiva comparación con el umbral de inicio lento. Si el tamaño de la ventana es igual al del umbral, se pasa a evasión de congestión, si no, permanece en esa fase.

En evasión de congestión se permanece hasta que termina la transmisión o se presenta una pérdida de paquete. Si hay pérdida y recibe tres reconocimientos duplicados, TCP Tahoe pasa a retransmisión rápida, envía el paquete perdido, reduce el umbral de inicio lento a la mitad de la ventana de congestión, inicializa la ventana a uno y regresa a

inicio lento. Si no recibe los tres reconocimientos duplicados, espera a que expire el temporizador de retransmisión, establece el umbral de inicio lento y la ventana de congestión como se dijo y se cambia a inicio lento. La Figura 18 muestra el modelo gráfico de cómo funciona TCP Tahoe.

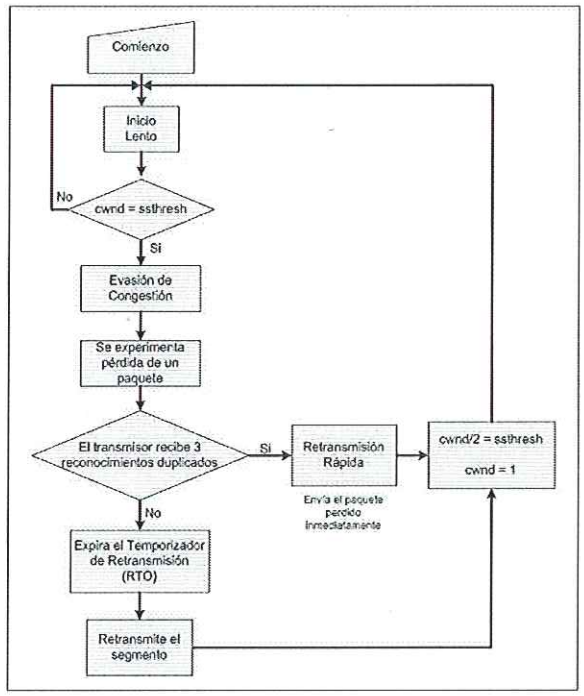


Figura 18. Modelo gráfico del funcionamiento de TCP Tahoe

IV.2.2 Variante TCP Reno

La implementación de TCP Reno funciona de acuerdo a lo establecido en [Lang, 2002 y Hassan y Jain, 2004]. A las dos etapas establecidas en Tahoe se agrega una tercera, que es la de Retransmisión Rápida/Recuperación Rápida (RR/RR). Inicio lento funciona de manera igual a Tahoe. Si recibe tres reconocimientos duplicados en evasión pasa a RR/RR; si no, se va directo a expiración de temporizador.

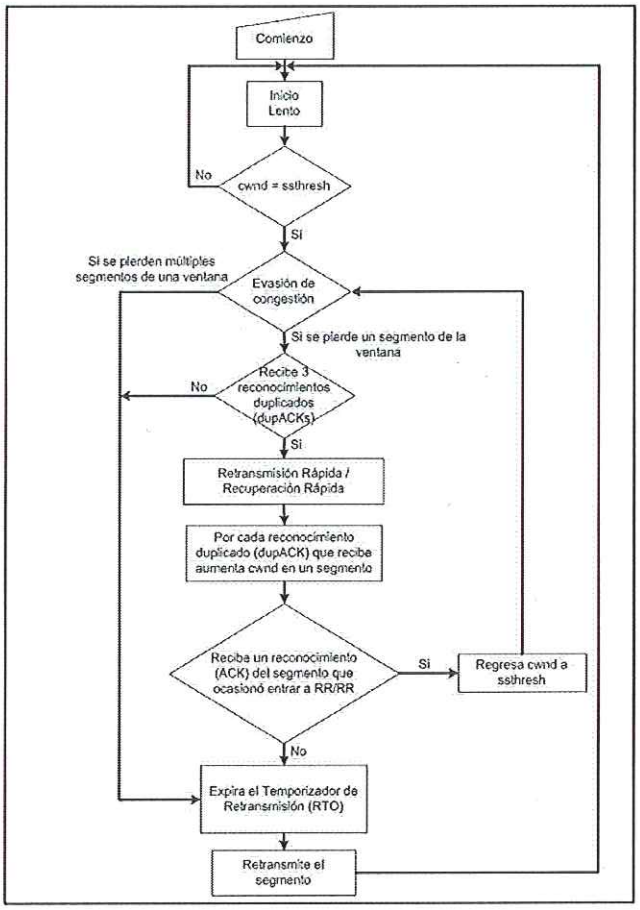


Figura 19. Modelo gráfico del funcionamiento de TCP Reno

Por cada reconocimiento duplicado que recibe en RR/RR aumenta la ventana de congestión en uno. De recibir un reconocimiento del segmento que ocasionó entrar a RR/RR, Reno regresa la ventana de congestión al umbral de inicio lento y vuelve a empezar la evasión de congestión. Si no recibe los reconocimientos, espera la expiración, retransmite el segmento perdido y regresa a inicio lento. La Figura 19 muestra el modelo gráfico de cómo funciona TCP Reno.

IV.2.3 Variante TCP NewReno

TCP NewReno está implementado de acuerdo a lo descrito en [Floyd y Henderson, 1999]. El algoritmo de inicio lento y evasión funcionan de manera similar a Reno. Cuando pasa de evasión a RR/RR para evitar detener la transmisión por pérdidas por cada

reconocimiento recibido automáticamente transmite el segmento que sigue al reconocimiento.

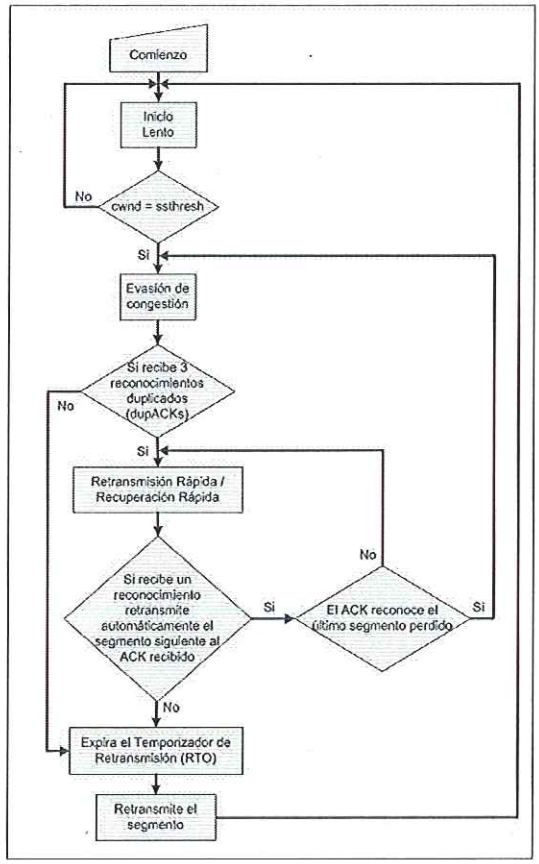


Figura 20. Modelo gráfico del funcionamiento de TCP NewReno

Si el reconocimiento hace referencia al último segmento perdido regresa a evasión de congestión. Si no recibe los tres reconocimientos duplicados, o expira el temporizador de retransmisión, retransmite el segmento y regresa a inicio lento. La Figura 20 presenta el modelo gráfico del funcionamiento de TCP NewReno.

IV.2.4 Variante TCP SACK

La implementación de TCP SACK corresponde a [Mathis *et al.*, 1996]. La característica sobresaliente de SACK está en el hecho de que el receptor usa un bloque TCP SACK para indicarle al transmisor que un bloque de datos se recibió fuera de orden en el receptor.

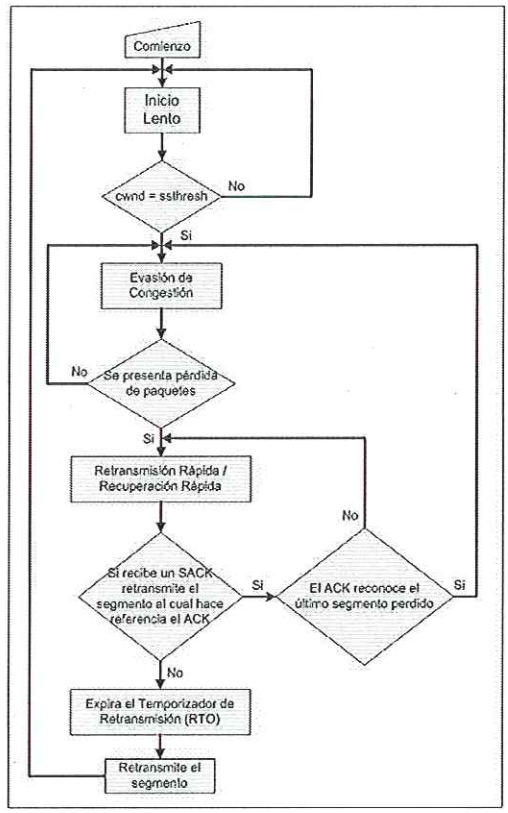


Figura 21. Modelo gráfico del funcionamiento de TCP SACK

Estos bloques SACK los usa el transmisor para mantener una imagen de la cola del receptor, lo que le indica cuáles segmentos se recibieron de manera exitosa y cuáles se perdieron. Como se mencionó en la sección III.5.4 donde se ofreció una explicación más detallada de SACK, esta variante puede utilizar los mismos algoritmos de control de congestión que Reno o NewReno. De esta manera maneja el inicio lento al comienzo de la transmisión, para después cambiar a la siguiente etapa que es evasión de congestión y si se presentan pérdidas, reacciona con una Retransmisión Rápida/Recuperación Rápida. Durante esta fase transmite los segmentos de acuerdo a los reconocimientos selectivos recibidos. La Figura 21 muestra el modelo gráfico del funcionamiento de TCP SACK.

IV.2.5 Variante TCP Vegas

La implementación de TCP Vegas corresponde a lo establecido en [Brakmo *et al.*, 1994]. Maneja un inicio lento donde continuamente se calcula *Nb* que es el número de paquetes

aún pendientes en la red, lo cual hace con cada segundo *RTT*. Con el valor que obtiene realiza una comparación con los valores umbral que tiene y dependiendo estos resultados permanece en inicio lento o cambia a evasión. En evasión de congestión continua haciendo mediciones, pero ahora de caudal eficaz. De los resultados que obtiene, realiza comparaciones con valores umbral y determina la estrategia que tiene que usar sobre la ventana de congestión. La Figura 22 presenta el modelo gráfico del funcionamiento de TCP Vegas.

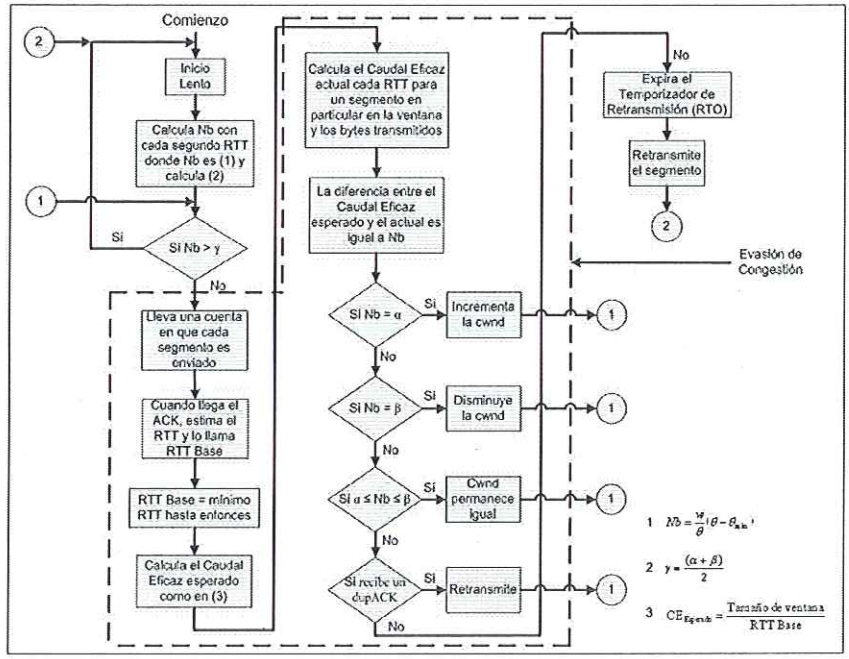


Figura 22. Modelo gráfico del funcionamiento de TCP Vegas

IV.2.6 Variante TCP Westwood

La implementación de TCP Westwood funciona de acuerdo a lo establecido en [Zanella *et al.*, 2001]. En el inicio lento calcula el estimado de ancho de banda y el umbral de inicio lento. En la etapa de evasión al recibir reconocimientos duplicados retransmite los segmentos calcula el nuevo umbral de inicio lento y ventana de congestión, con las fórmulas quese presentan dentro del modelo gráfico del funcionamiento de TCP Westwood en la Figura 23.

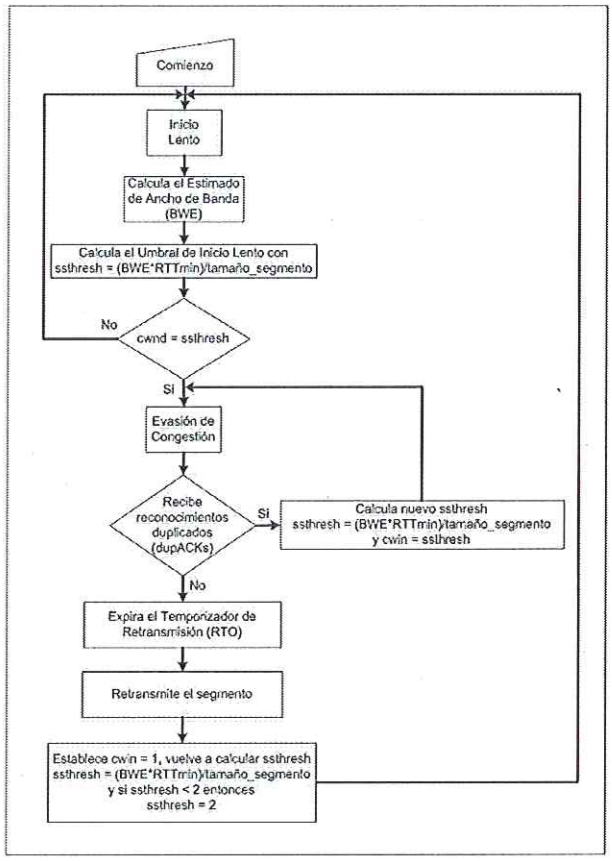


Figura 23. Modelo gráfico del funcionamiento de TCP Westwood

IV.3 Modelado de parámetros de Calidad de Servicio en Redes Satelitales de Órbita Baja

En ésta sección se presenta un análisis de los principales parámetros de Calidad de Servicio con los que se pretende evaluar el desempeño de las variantes de TCP en un ambiente satelital. Para realizar esto se presentarán expresiones y fórmulas matemáticas que permitan modelar y evaluar el retardo, variación en el retardo, tasa de error de bit, tasa de pérdida de paquetes y caudal eficaz.

IV.3.1 Modelado de la latencia en constelaciones satelitales

En el punto II.4.1 se presentó la definición de latencia, y en lo que respecta a una red satelital de órbita baja se puede expresar como la suma de los retardos que experimenta a lo largo de su trayectoria, los cuales son el retardo de transmisión, retardo de propagación,

retardo de los enlaces intersatelitales, retardo por procesamiento y conmutación y retardo del protocolo de control de acceso al medio. La siguiente expresión permite realizar el cálculo de la latencia total:

$$Latencia = L_t + L_{propUL} + L_{propDL} + L_{ISLs} + L_{MAC} \tag{47}$$

donde L_t es el retardo de transmisión, L_{propUL} es el retardo de propagación de enlace ascendente, L_{propDL} es el retardo de propagación de enlace descendente, L_{ISLs} es el retardo de los enlaces intersatelitales y L_{MAC} es el retardo del protocolo de control de acceso al medio

IV.3.1.1 Retardo de transmisión

El retardo de transmisión L_t representa el tiempo que se toma en transmitir un paquete de información para una tasa de datos definida. Se puede calcular con la siguiente expresión:

$$L_t = \frac{tamaño\ del\ paquete}{tasa\ de\ datos} \tag{48}$$

IV.3.1.2 Retardo de propagación

El retardo de propagación se compone de la suma del retardo de propagación de la estación terrena al primer satélite de la constelación (que también es conocido como enlace ascendente) o L_{propUL}, el retardo de propagación del satélite final de la constelación a la estación terrena (que también se conoce como enlace descendente) o L_{propDL} y el retardo de propagación por enlaces intersatelitales o L_{ISLs}.

IV.3.1.2.1 Retardo de propagación para enlace ascendente y descendente

El retardo de propagación para enlace ascendente y descendente se pueden calcular de manera similar. Primero se calcula la distancia máxima del satélite a las estaciones terrenas, con la ecuación 3 de la sección II.3.5:

$$d_{\max} = r_E \frac{sen\alpha}{sen\nu} \tag{49}$$

Ya que se definió la distancia, con la siguiente ecuación (50) se calcula el retardo de propagación:

$$L_{propUL} = L_{propDL} = \frac{d}{c} \tag{50}$$

donde d es la distancia de la estación terrena al satélite (ascendente) y viceversa (para el enlace descendente) y c es la constante de la velocidad de la luz.

IV.3.1.2.2 Retardo de propagación de enlaces intersatelitales

Con respecto al retardo de propagación de enlaces intersatelitales, este se refiere a la suma de retardos que generan los enlaces de los satélites que se comunican directamente entre si. La ecuación (51) permite calcular el retardo de propagación de los enlaces intersatelitales:

$$L_{ISLs} = \frac{\sum ISLs}{c} \tag{51}$$

donde $\sum ISLs$ es la sumatoria de distancias entre satélites a lo largo de la trayectoria en km y c es la constante de la velocidad de la luz. La longitud del enlace intersatelital se obtiene con la ecuación (52):

$$D_{ISL} = \sqrt{2(1-\cos\theta)r_s^{\,2}} \tag{52}$$

donde θ es el ángulo de separación entre satélites adyacentes y r_s es el radio satelital.

IV.3.1.3 Retardo por procesamiento y conmutación

Este trabajo no considera retardos por procesamiento y conmutación a bordo del satélite y retardo por el control de acceso al medio ya que el simulador de red ns2 no los soporta, sin embargo el procesamiento a bordo en los sistemas satelitales GEO y LEO tiene cada vez mayor auge por lo que es relevante presentar este análisis. Muchos de los sistemas que se planean y proponen para utilizar procesamiento a bordo utilizan un esquema de conmutación similar o de tipo ATM por lo que para realizar el estudio se utiliza como referencia el desarrollo presentado por [Stallings, 1998 y Kota, 2004], donde se considera que los satélites son elementos activos de red y pueden decidir hacia donde enviar los paquetes que reciben (entre sus opciones tienen a la estación terrena u otro satélite a través de un ISL) y este tipo de arquitectura satelital es llamada S-ATM. Para un conmutador S-ATM [Stallings, 1998] emplea un modelo de colas M/D/1 y como las celdas S-ATM tienen

un tamaño fijo existe un tiempo determinístico de servicio con un solo servidor y un buffer de tamaño infinito. Se calcula la capacidad de conmutación de celdas con la ecuación (53):

$$R_{celdas} = \frac{R_{conmutación}}{(53*8)} \tag{53}$$

donde R_{celdas} está dado en celdas/segundo, $R_{conmutación}$ es la capacidad de conmutación en bits/segundo, el 53 se refiere al tamaño en bytes de las celdas ATM satelitales y el 8 es el factor de conversión a bits. [Stallings, 1998] establece la obtención de la tasa de drenado de buffer del conmutador S-ATM de la siguiente manera:

$$\frac{1}{T_s} = \frac{1}{\frac{1}{R_{celdas}} + 1x10^{-6}} \tag{54}$$

La ecuación anterior está dada en celdas/segundo. Para drenar una celda de S-ATM solo basta obtener el inverso de la tasa de drenado de buffer.

El retardo por encolamiento es el retardo que sufre la celda S-ATM es espera de ser atendida por el conmutador y puede ser calculado con la ecuación (55):

$$Retardo\ por\ encolamiento \leq \frac{Tamaño\ del\ buffer}{Tasa\ de\ drenado\ del\ buffer} \tag{55}$$

El tiempo promedio de espera T_q que experimenta cada celda para ser atendida por el conmutador se compone de la suma del tiempo de drenado T_s y el tiempo promedio de espera por cada arribo T_w.

$$T_q = T_w + T_s \tag{56}$$

A este retardo se le conoce como *retardo por procesamiento*.

IV.3.1.4 Retardo del Protocolo de Control de Acceso al Medio

Los protocolos de control de acceso al medio son utilizados para coordinar la transmisión de paquetes, retransmitir los paquetes dañados y resolver las colisiones entre estaciones cuando contienden por el acceso al recurso o por la oportunidad para transmitir. En un esquema de acceso múltiple por división de tiempo (TDMA), el retardo del protocolo MAC representa el tiempo entre el arribo de un paquete y el final de la duración del tiempo de trama en la que llega. Se calcula de la siguiente manera:

$$L_{MAC} = \frac{1}{2}\frac{N_{ET}b}{c} \tag{57}$$

donde N_{ET} representa el número de estaciones terrenas que comparten el canal TDMA, b el tamaño de paquete a enviar y c es la tasa de envío.

IV.3.2 Modelado de la variación en el retardo en constelaciones satelitales

La definición de variación en el retardo o jitter se presentó en la sección II.4.2. El jitter hace que el protocolo TCP haga estimaciones muy conservadoras del RTT, operando de manera ineficiente cuando utiliza las expiraciones de temporizador para reestablecer el flujo de datos. Esto se hace más evidente con los traspasos entre satélites cuando se calcula el retardo por propagación de enlaces intersatelitales con la ecuación (52), ya que las estimaciones de longitud al estar variando afectan al temporizador interno de TCP. Al tener un traspaso entre satélites y movimiento continuo de la constelación, se incrementa el almacenamiento en buffer de los paquetes y, si no se desborda, en el mejor de los casos incrementa el tiempo promedio de espera y el retardo por encolamiento de las ecuaciones (56) y (57).

IV.3.3 Modelado de la tasa de error de bit (BER) en constelaciones satelitales

Cuando una señal se propaga entre un satélite y una estación terrena, el efecto básico es la atenuación de señal debido a las pérdidas por espacio libre. La atenuación se incrementa si aumenta la distancia entre el satélite y la estación terrena [Nam, 2003]. La potencia recibida C en watts está dada por:

$$C = \frac{P_T G_T G_R}{(4\pi d/\lambda)^2} = \frac{P_T G_T G_R}{L_0} \tag{58}$$

donde λ es la longitud de onda, P_t es la potencia transmitida, G_t es la ganancia de la antena transmisora, G_r es la ganancia de la antena receptora, d es la distancia entre el satélite y la terminal terrena y L_0 es llamada la pérdida de espacio libre y representa la relación de la potencia transmitida a la recibida en un enlace entre el satélite y el usuario [Maral y Bousquet, 2003]. La ganancia de recepción G_r de una antena se calcula con la ecuación (59):

$$G_R = \eta\left(\frac{\pi D}{\lambda}\right)^2 \tag{59}$$

donde η es la eficiencia de la antena, D el diámetro de la antena y λ la longitud de onda. A partir de la ecuación (59) se puede obtener la Potencia Isotrópica Radiada Efectiva (*PIRE*):

en forma lineal $$PIRE = P_T G_T \tag{60}$$

en dB $$PIRE\big|_{dBW} = 10\log P_T + G_T\big|_{dBi} \tag{61}$$

El *PIRE* se define como una potencia de transmisión equivalente a la potencia irradiada que pretende alcanzar la misma densidad de flujo de potencia de una antena directiva. Para analizar de manera conjunta todos los elementos que intervienen en el cálculo de la potencia de la portadora se sustituye la ecuación (61) en la (58) y se obtienen las siguientes expresiones:

$$\begin{aligned} C\big|_{dBW} &= PIRE\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \\ C\big|_{dBW} &= 10\log P_T\big|_{dBi} + G_T\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \end{aligned} \tag{62}$$

La ecuación (62) representa la potencia de la portadora en la estación terrena transmisora. Cabe aclarar que el término P_T y G_T corresponden a la estación terrena mientras que el G_R corresponde al satélite.

Para el análisis descendente los papeles emisor y receptor se cambian entre el satélite y la estación terrena, y ahora el emisor es el satélite y el receptor la estación terrena. El cálculo de la potencia recibida en el enlace de bajada es similar al de subida, donde ahora el *PIRE* es generado por el satélite, la antena receptora se encuentra en la estación terrena y la frecuencia que se utiliza es la frecuencia de bajada. La expresión de la potencia de la portadora recibida en la estación terrena es la siguiente:

$$\begin{aligned} C\big|_{dBW} &= PIRE\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \\ C\big|_{dBW} &= 10\log P_T\big|_{dBi} + G_T\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \end{aligned} \tag{63}$$

Un parámetro que se debe tomar en cuenta al momento de realizar el análisis del sistema es el ruido, que en mayor o menor medida acompaña siempre a la señal de información. El ruido básicamente se deriva del ruido térmico irreducible debido al

$$G_R = \eta\left(\frac{\pi D}{\lambda}\right)^2 \tag{59}$$

donde η es la eficiencia de la antena, D el diámetro de la antena y λ la longitud de onda. A partir de la ecuación (59) se puede obtener la Potencia Isotrópica Radiada Efectiva (*PIRE*):

en forma lineal $$PIRE = P_T G_T \tag{60}$$

en dB $$PIRE\big|_{dBW} = 10\log P_T + G_T\big|_{dBi} \tag{61}$$

El *PIRE* se define como una potencia de transmisión equivalente a la potencia irradiada que pretende alcanzar la misma densidad de flujo de potencia de una antena directiva. Para analizar de manera conjunta todos los elementos que intervienen en el cálculo de la potencia de la portadora se sustituye la ecuación (61) en la (58) y se obtienen las siguientes expresiones:

$$\begin{aligned} C\big|_{dBW} &= PIRE\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \\ C\big|_{dBW} &= 10\log P_T\big|_{dBi} + G_T\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \end{aligned} \tag{62}$$

La ecuación (62) representa la potencia de la portadora en la estación terrena transmisora. Cabe aclarar que el término P_T y G_T corresponden a la estación terrena mientras que el G_R corresponde al satélite.

Para el análisis descendente los papeles emisor y receptor se cambian entre el satélite y la estación terrena, y ahora el emisor es el satélite y el receptor la estación terrena. El cálculo de la potencia recibida en el enlace de bajada es similar al de subida, donde ahora el *PIRE* es generado por el satélite, la antena receptora se encuentra en la estación terrena y la frecuencia que se utiliza es la frecuencia de bajada. La expresión de la potencia de la portadora recibida en la estación terrena es la siguiente:

$$\begin{aligned} C\big|_{dBW} &= PIRE\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \\ C\big|_{dBW} &= 10\log P_T\big|_{dBi} + G_T\big|_{dBW} + G_R\big|_{dBi} - L_0\big|d_B \end{aligned} \tag{63}$$

Un parámetro que se debe tomar en cuenta al momento de realizar el análisis del sistema es el ruido, que en mayor o menor medida acompaña siempre a la señal de información. El ruido básicamente se deriva del ruido térmico irreducible debido al

movimiento aleatorio de los electrones en los conductores [Elbert, 2001]. La potencia de ruido en la carga para el ancho de banda que interesa está dado en watts por:

$$N = kT_{eq}B \tag{64}$$

donde k representa la constante de Boltzmann que tiene un valor de $1.38\text{x}10^{-23}$ J/K, T_{eq} es la temperatura equivalente de ruido en Kelvin y B es el ancho de banda de la información de interés en Hertz. El ruido blanco se refiere a que la potencia por cada Hertz o unidad de ancho de banda es constante, de ahí que el ruido térmico tenga una distribución uniforme en el espectro de frecuencia [Cabanillas, 2004]. El ruido blanco permite calcular la densidad de ruido N_0 en Joules o W/Hz de la siguiente manera:

$$N_0 = \frac{N}{B} = kT \tag{65}$$

Otro factor a tomar en cuenta es el parámetro G/T que es una figura de mérito de una estación terrena o satélite, y su propósito es aplicarlo directamente al presupuesto de enlace para suministrar una medida de la relación portadora a ruido [Elbert, 2001]. El análisis completo de G/T considera el ruido total recibido debido a la antena, guía de onda, amplificación de bajo ruido y etapas subsecuentes. De ahí que su valor varía en cada satélite dependiendo de las características de la antena, ganancia y patrón de radiación, número de transpondedores y otros elementos del subsistema de comunicaciones.

$$\left(\frac{C}{N_0}\right)_{UL} = \underbrace{PIRE\big|_{dBW}}_{\text{De la Estación Terrena}} - \underbrace{L_{\text{Propagación}}\big|_{dB}}_{\text{Pérdidas de espacio libre absorción atmosférica, lluvia, etc.}} + \underbrace{\left.\frac{G_{Sat}}{T}\right|_{Sat\ dB/k}}_{\text{Fig. de Mérito en el satélite}} - \underbrace{10\log K\big|_{dB\cdot Hz}}_{\text{Constante de Boltzmann}} \tag{66}$$

$$\left(\frac{C}{N_0}\right)_{DL} = \underbrace{PIRE\big|_{dBW}}_{\text{Según el contorno que pase por la ET receptora}} - \underbrace{L_{\text{Propagación}}\big|_{dB}}_{\text{Pérdidas de espacio libre absorción atmosférica, lluvia, etc.}} + \underbrace{\left.\frac{G_R}{T}\right|_{ET\ dB/k}}_{\text{Fig. de Mérito en ET}} - \underbrace{10\log K\big|_{dB\cdot Hz}}_{\text{Constante de Boltzmann}} \tag{67}$$

C es la potencia de la portadora en dBW y N_0 es la densidad espectral de ruido en dBW/Hz. En lo que respecta a la tasa de error de bit (BER) esperado este es una función del cociente de la Energía de Bit (E_b) y la Densidad de ruido (N_0), y al mismo tiempo se puede expresar con ayuda de la relación señal portadora-ruido de la siguiente manera:

en forma lineal $$\left(\frac{E_b}{N_0}\right)_T = \frac{(C/N_0)_T}{R_c} \quad (68)$$

en dB $$\left(\frac{E_b}{N_0}\right)_{dB} = (C/N_0)_{T\,dB} - (R_c)_{dB} \quad (69)$$

donde R_c es la tasa de bit de transmisión sobre el canal satelital.

Para determinar el *BER* se puede considerar un canal satelital lineal sin interferencia, donde la probabilidad de error sea la obtenida teóricamente y no exista degradación debido al filtrado o a no linealidades [Cabanillas, 2004]. El *BER* total del enlace satelital para satélites regeneradores se puede expresar como la probabilidad de tener un error en el enlace de subida y no presentar errores en el de bajada, o viceversa, lo que se presenta de la siguiente manera:

$$BER_T = (\text{Prob. de 1 error})_{subida}(\text{Prob. de No errores})_{bajada} + (\text{Prob. de No errores})_{subida}(\text{Prob. de 1 error})_{bajada} \quad (70)$$

Con respecto a un satélite regenerador al ser capaz de corregir errores en el enlace de subida se puede concluir que el BER total se puede resumir a los errores introducidos en el enlace de bajada, lo que se expresa de la siguiente manera:

$$BER_T = BER_{bajada} \quad (71)$$

IV.3.4 Modelado de la Tasa de Pérdida de Paquetes (PLR)

En el punto II.4.4 se estableció que algunas de las razones por las que se puede presentar pérdida de paquetes es porque los sistemas de conmutación están saturados de tráfico y los paquetes se desbordan o la línea de comunicación está dañada. Una manera de calcular el PLR es obtener la relación del número de paquetes que se envían con respecto al número de paquetes recibidos como se presenta en la ecuación (72):

$$Tasa\ de\ pérdida\ de\ paquetes = \frac{Número\ de\ paquetes\ perdidos}{Número\ de\ paquetes\ enviados} \quad (72)$$

IV.3.5 Modelado del Caudal Eficaz

En la sección II.4.5 se presentó la definición de caudal eficaz. Se puede expresar de la siguiente manera:

$$Caudal\ Eficaz = \frac{Total\ de\ paquetes\ recibidos}{Total\ de\ paquetes\ enviados} \tag{73}$$

IV.4 Conclusiones

En este capítulo se presentaron un método para modelar el protocolo TCP y seis modelos gráficos de variantes de TCP que explican sus mecanismos particulares, asi como también se modelaron los parámetros de calidad de servicio (retardo, variación en el retardo, tasa de error de bit, tasa de pérdida de paquetes y caudal eficaz) con los cuales se va a evaluar el desempeño de las variantes de TCP en el ambiente satelital.

En el siguiente capítulo se va a explicar el proceso de simulación de las transmisiones de información en una red satelital de órbita baja utilizando a TCP como protocolo de transporte.

V. RESULTADOS DE SIMULACIÓN Y ANÁLISIS NUMÉRICO

En el capítulo anterior se presentaron un método para modelar el comportamiento de TCP y modelos gráficos de sus variantes, así como también se presentó un análisis y modelado de los parámetros de Calidad de Servicio con los que se evalua el desempeño de TCP en un ambiente satelital de órbita baja. En éste capítulo se presentan los resultados de simulación y se realiza un análisis numérico de las estadísticas obtenidas. Se establece cuáles son las consideraciones de simulación y se presenta una breve introducción del Simulador de Red Network Simulator 2 (ns-2).

V.1 Consideraciones de simulación

Antes de de comenzar con la explicación del proceso de simulación, se requiere establecer las consideraciones que se hicieron en el presente trabajo. A continuación se presentan las consideraciones de simulación:

- Traspaso y movilidad del usuario no son considerados en este trabajo. El enfoque está dirigido al desempeño para el caso de usuarios en una red de acceso satelital.
- El impacto del saludo de tres vías durante el establecimiento de la conexión tampoco se considera. Cuando se realizan transferencias cortas en enlaces de alto retardo, el tiempo que se utiliza para abrir la conexión es una cuestión a considerar como lo establecen [Henderson y Katz, 1999], pero con su reducción solo se ganaría un tiempo de viaje completo. Para este trabajo se usa un tamaño de archivo grande y la presencia o ausencia del saludo de tres vías no hace mucha diferencia.
- Con el fin de simplificar el análisis y proceso de simulación se consideró un enlace satelital fijo, lo que permite reducir la complejidad. Esto es suficiente para el análisis que se está realizando, ya que el objetivo es evaluar el impacto que tiene el ambiente satelital en las variantes de TCP.

- La arquitectura de red implementada consiste de dos estaciones terrenas, transmisora y receptora, conectadas por medio de un satélite repetidor. Las ubicaciones de las estaciones son las siguientes:

Estación Terrena transmisora:

Ensenada, México (Latitud 31.5° Norte y Longitud 116.3° Oeste)

Estación Terrena receptora:

Ciudad de México, México (Latitud 19.43° Norte y Longitud 99.1° Oeste)

- Para el cálculo de la tasa de error de bit se considera para el satélite propuesto una altura orbital y un ángulo mínimo de elevación como parámetros más significativos. Para las estaciones terrenas transmisora y receptora se toman en cuenta la potencia de transmisor y receptor, diámetro de la antena transmisora y receptora, elevación sobre el nivel del mar y región pluvial. La banda de frecuencia para el enlace de subida y bajada es la banda Ku con 14.25 GHz y 11.95 GHz respectivamente.

V.2 Introducción al simulador de red Network Simulator 2 (ns2)

Ns2 es un simulador de red orientado a objetos y conducido por eventos, que permite la simulación de una variedad de redes de área local y amplia. Implementa diferentes protocolos de transporte, como TCP y UDP; fuentes de tráfico de distintos tipos, por ejemplo, FTP, Web, CBR (Tasa de Bit Constante) y exponencial on/off; mecanismos de manejo de cola, como RED y DropTail y protocolos de enrutamiento. Ns2 está escrito en C++ y Otcl. La razón por la que ns2 utiliza dos lenguajes es porque distintas tareas tienen diferentes requerimientos. Por ejemplo, la simulación de protocolos requiere una manipulación eficiente de los bytes y encabezados de paquete, haciendo que la velocidad de simulación sea muy importante. Por otro lado, en estudios de red donde el objetivo sea variar parámetros y examinar rápidamente un número de escenarios, el tiempo para cambiar el modelo y volverlo a correr es más importante.

En ns2, C++ es utilizado para una implementación de protocolo detallada, y en general, para aquellos casos donde cada paquete de un flujo tenga que ser procesado. Por ejemplo, si se quiere implementar una nueva disciplina de encolamiento, C++ es el lenguaje que se utilizaría. En cambio Otcl, está dirigido a la configuración y manipulación del

programa. Otcl corre un tanto lento, pero su ventaja se presenta en el hecho de que se le pueden realizar cambios de manera rápida, lo que vuelve el aspecto de la programación y de la construcción de escenarios en algo más sencillo. En el simulador ns2 los objetos compilados de C++ se pueden hacer disponibles a través del interpretador de Otcl, de ésta manera los objetos de C++ que ya están construidos y listos pueden ser controlados desde el nivel Otcl. Por último, se debe mencionar que este simulador no se ha enfocado recientemente en proveer simulaciones detalladas de las capas de enlace y físico, e inicialmente no fue diseñado para soportar movilidad de terminal o topologías dinámicas [Henderson y Katz, 1999].

V.2.1 Parámetros de entrada en el simulador de red Network Simulator 2 (ns2)

A continuación se presentan y justifican el escenario implementado y los parámetros mas comunes en las simulaciones realizadas. Como se comentó en el tercer punto de la sección V.1, se consideró un enlace satelital fijo entre una estación terrena transmisora y una receptora similar a las implementaciones realizadas en [Mascolo *et al.*, 2001 y Lang, 2002].

La idea original propuesta iba en la dirección de analizar el desempeño de las variantes de TCP en una red compuesta por varios satélites de órbita baja, muy similar a los proyectos Iridium y Teledesic. Como se comentó en la sección V.2, múltiples aspectos del simulador ns2 aún no están desarrollados y algunos como el segmento satelital están en sus inicios. Esto no significa que no se pueda hacer, pero la implementación rebasa en tiempo y recursos computacionales al objetivo planteado en este trabajo de investigación, razón por la que se optó por seguir el planteamiento donde se considera un satélite fijo entre dos estaciones terrenas.

Dentro de la investigación realizada en la literatura disponible como artículos, libros y la Internet, la mayoría de los trabajos encontrados proponen en sus planteamientos una serie de consideraciones que les permitan realizar el trabajo reduciendo el tiempo y complejidad al momento de la simulación. El caso de este proyecto se comenzó a trabajar con TCP en una red satelital tipo Iridium, pero dentro de la programación realizada en el simulador ns2 no se pudo obtener un comportamiento estable entre los agentes TCP y la red

satelital. Por esta razón se toman como referencia los trabajos de [Mascolo et al, 2001 y Lang, 2002] para reorientar este trabajo de tesis ya que un aspecto a resaltar de este trabajo es la evaluación del desempeño de las variantes de TCP en un ambiente satelital con los parámetros de Calidad de Servicio. En este caso, los parámetros de QoS para Voz sobre IP son los mas exigentes en lo que a transmisión de información se refiere y hasta donde se revisó y leyó no se encontró otro trabajo con un planteamiento similar.

Para el enlace satelital se utilizó un enlace duplex de 1.5 Mbps, la cual es una velocidad de enlace comúnmente utilizada para redes de acceso satelital como en [Ghani y Dixit, 1999 y Allman *et al.*, 1999]. El Máximo Tamaño de Segmento (MSS) de TCP fue fijado a 536 bytes el cual es el valor predefinido del MSS si el descubrimiento de trayectoria de la Máxima Unidad de Transferencia falla como se especifica en [Fall y Varadhan, 2003]. Para el enlace satelital se estableció un Tiempo de Viaje Completo (RTT) de 50 ms, el cual representa la suma del retardo de ida y el de vuelta de un Satélite de Órbita Baja (LEO) y una altura orbital de 1,000 km.

En el ns2 para un enlace se debe especificar la disciplina de encolamiento que se utiliza, y en este caso se eligió encolamiento Drop Tail ya que es el más comúnmente usado para aplicaciones de datos que usan TCP. El tamaño de cola se obtuvo del Producto Retardo-Ancho de Banda, que en este caso es de 40 paquetes, aunque se realizaron experimentos con otros tamaños pero solo con el fin de definir porqué el de 40es el adecuado. La **Tabla II** reune los parámetros de entrada al simulador.

De las seis variantes de TCP presentadas en la sección III.5 y IV.2 sólo se emplearon cinco dejando fuera a TCP Vegas debido a que los resultados de simulación para esta variante no presentaban un comportamiento similar a lo presentado por [Lang, 2002]. Se realizaron diversas modificaciones para tratar de resolver el problema y se trató de contactar a los autores de la variante sin ninguna respuesta.

Tabla II. Parámetros de entrada al simulador

Parámetros de Entrada al Simulador de Red ns2	
Ancho de banda del enlace	1.5 Mbps
Retardo del enlace	25 ms
Tiempo de Viaje Completo (RTT)	50 ms
Disciplina de encolamiento del enlace	Drop Tail
Agentes de TCP Implementados	Tahoe, Reno, NewReno, SACK, Westwood
Tamaño de ventana TCP	200 segmentos
Tamaño de segmento TCP	536 bytes
Tamaño de archivo de transferencia	10 MB

V.3 Resultados de Simulación

V.3.1 Evolución de ventana de congestión y Producto Retardo-Ancho de Banda

En ésta sección se presentan los resultados obtenidos sobre la evolución de la ventana de congestión de cada una de las variantes de TCP. Se abordó el aspecto del Producto Retardo-Ancho de Banda de la conexión, ya que este se encuentra ligado al desempeño que pueda tener TCP, y se muestra que para obtener un caudal eficaz óptimo el espacio de buffer mínimo en la trayectoria de conexión requiere ser igual al Producto Retardo-Ancho de Banda. El Producto Retardo-Ancho de Banda en paquetes, se puede expresar de la siguiente manera:

$$PRAB(paquetes) = \frac{AB(bits/seg) * RTT(seg)}{8 * MSS(bytes)} \tag{74}$$

donde PRAB es el Producto Retardo-Ancho de Banda, AB es el Ancho de Banda, RTT es el Tiempo de Viaje Completo y MSS es el Mínimo Tamaño de Segmento.

El TCP solo puede lograr su caudal óptimo si el mínimo tamaño de buffer en la trayectoria es igual al Producto Retardo-Ancho de Banda. Si el mínimo tamaño de buffer es más grande, se crea una cola de paquetes en el buffer desperdiciando recursos de red e introduciendo latencia. Si es más pequeño, la ventana de congestión nunca va a ser capaz de alcanzar el tamaño necesario para utilizar el ancho de banda disponible. Algunos otros trabajos que también abordan este tema son [Barakat *et al.*, 2000 y Laskhman y Madhow,

1997]. Utilizando la ecuación (74) se obtiene un Producto Retardo-Ancho de Banda de aproximadamente 40 paquetes, y los resultados obtenidos se explican con ayuda de las gráficas obtenidas para la variante Reno de TCP.

La Figura 24 muestra la ventana de congestión y la evolución del tamaño de cola de la transferencia de datos para un tamaño de buffer de 20 paquetes, el cual es la mitad del Producto Retardo-Ancho de Banda y en la Figura 25 se presenta una de las gráficas con las cuales se validó el simulador y los resultados obtenidos a partir de lo presentado por [Lang, 2002].

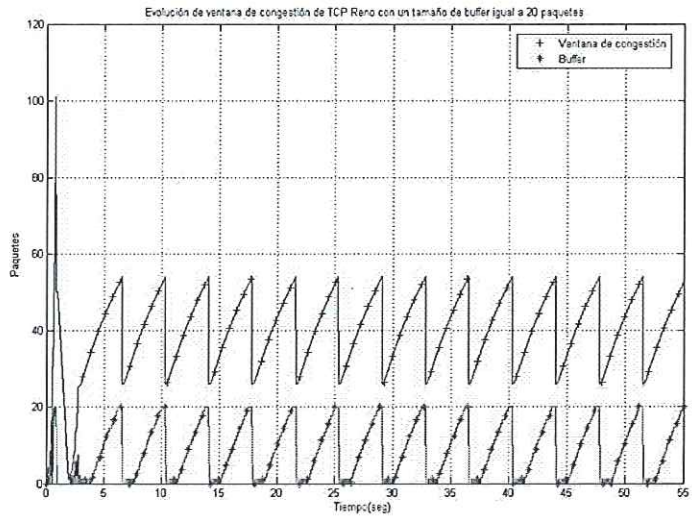


Figura 24. Evolución de ventana de congestión de TCP Reno con buffer de 20 paquetes

En el comienzo de la transferencia la conexión está en la fase exponencial de Inicio Lento. Al final de esta fase la ventana de congestión está cercana a $2*(PRAB+tamaño_buffer)$ paquetes. El buffer se desborda en este punto y cerca de la mitad de la ventana de datos se pierde. Estas pérdidas severas causan que TCP Reno experimente una expiración y que corte su umbral de inicio lento (ssthresh) dos veces. Después de la expiración se re-entra a Inicio Lento con un umbral de $\frac{(PRAB+tamaño_buffer)}{2}$, el cual es de 30 paquetes, y una ventana de congestión de uno.

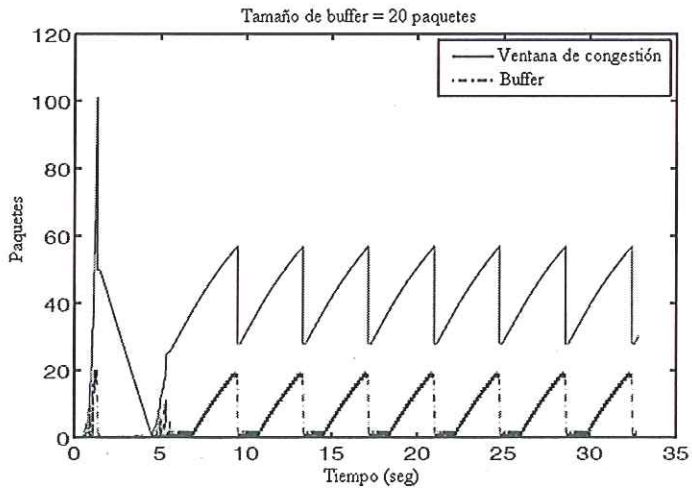


Figura 25. Evolución de ventana de congestión de TCP Reno [Lang, 2002]

Este nuevo umbral de Inicio Lento es 10 paquetes mas pequeño que el Producto Retardo-Ancho de Banda de 40. El efecto de fijar el umbral de inicio lento a un valor mas pequeño que el Producto Retardo-Ancho de Banda puede ser visto durante la fase estacionaria: después de cada pérdida, TCP espera por la mitad de la maxima ventana de congestión (aproximadamente 30 paquetes) de reconocimientos que regresen antes de que nuevos datos sean transmitidos.

Ya que solo hay 20 paquetes en el buffer, el buffer se vacía antes de que TCP comience a enviar nuevos datos, lo cual resulta en una considerable cantidad de tiempo durante el cual el enlace está vacío. Mientras más grande sea la diferencia entre el tamaño de buffer y el Producto Retardo-Ancho de Banda, más grande va a ser este tiempo disponible.

A continuación se considera el caso cuando el tamaño de buffer es más grande que el Producto Retardo-Ancho de Banda como muestra la Figura 26 y la Figura 27 es la gráfica con la cual se validó. La evolución general de la conexión es la misma que la del tamaño de buffer de 20 paquetes. Durante Inicio Lento la ventana de congestión crece casi hasta $2*(PRAB+tamaño_buffer)$, la mitad de los paquetes del último conjunto se perdieron, ssthresh es cortado dos veces, y se inicia una nueva fase de inicio lento, con un umbral de $\frac{(PRAB+tamaño_buffer)}{2}$ (50 paquetes para esta simulación).

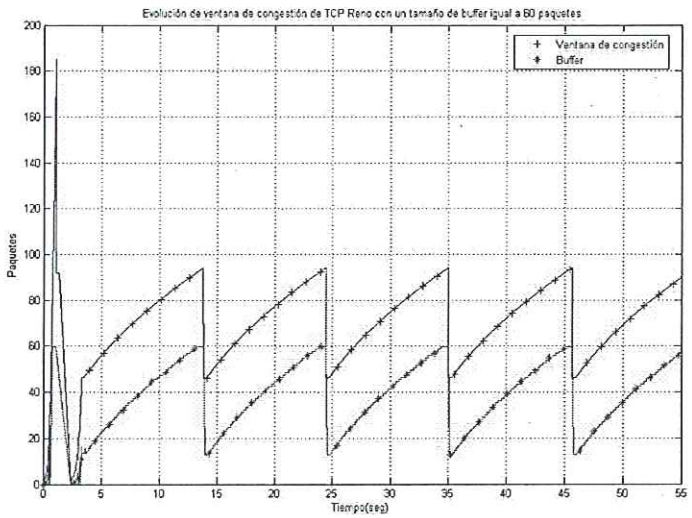


Figura 26. Evolución de ventana de congestión de TCP Reno con buffer de 60 paquetes

Cuando ocurre la siguiente pérdida TCP Reno otra vez espera por la mitad de una ventana de congestión de reconocimientos de regreso, la cual es de 50.

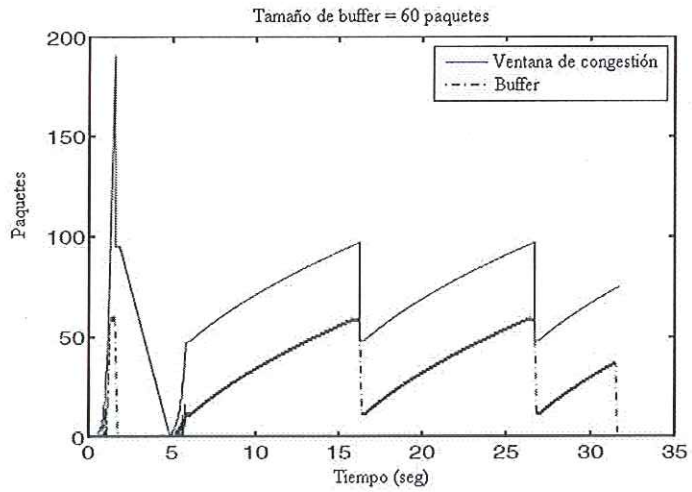


Figura 27. Evolución de ventana de congestión de TCP Reno [Lang, 2002]

Luego entonces solo 50 de los 60 paquetes encolados cuando el buffer está sobrecargado son transmitidos durante este tiempo disponible de TCP. Esto crea una cola constante en el buffer de 10 paquetes durante todo el tiempo de la conexión, el cual es solo drenado al final de la transmisión de datos.

En la Figura 28 finalmente se presenta el caso ideal donde el tamaño de buffer iguala el Producto Retardo-Ancho de Banda y la Figura 29 es la gráfica con la cual se validó.

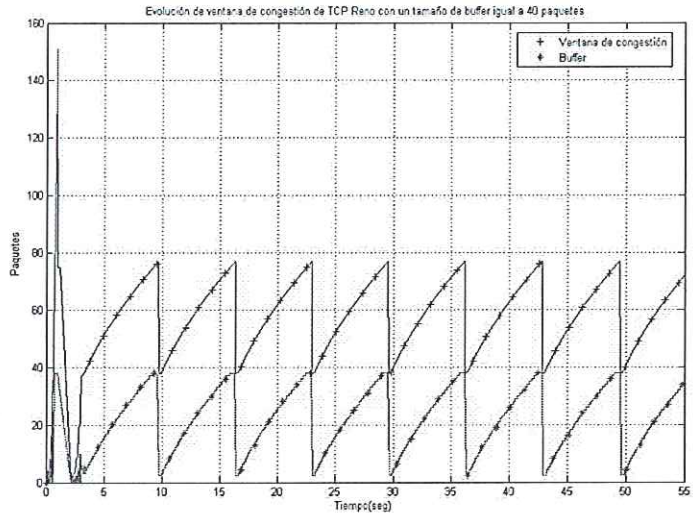


Figura 28. Evolución de ventana de congestión de TCP Reno con buffer de 40 paquetes

El ssthresh es otra vez fijado a $\frac{(PRAB + tamaño_buffer)}{2}$ después de las pérdidas al final del inicio lento. Debido a que el tamaño de buffer es igual al Producto Retardo-Ancho de Banda, el ssthresh es también igual al Producto Retardo-Ancho de Banda. Después de cada pérdida adicional, cuando la ventana de congestión alcanza $PRAB + tamaño_buffer + 1$, TCP espera exactamente 40 reconocimientos de regreso. Esto permite la transmisión de todos los paquetes en el buffer antes de que TCP recomience a enviar nuevos datos. Un tamaño de buffer igual al Producto Retardo-Ancho de Banda evita que el enlace esté vacío, así como la creación de una cola constante, lo cual ocurriría si el buffer es menor o mayor que el Producto Retardo-Ancho de Banda, respectivamente.

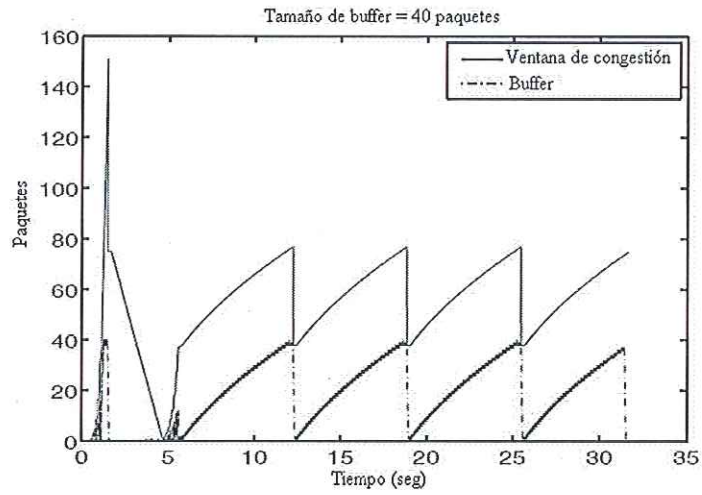


Figura 29. Evolución de ventana de congestión de TCP Reno [Lang, 2002]

Basándose en los experimentos de simulación, las siguientes observaciones generales pueden ser hechas en lo concerniente al tamaño de buffer, ventana de congestión y Producto Retardo-Ancho de Banda. Durante inicio lento la ventana de congestión necesita ser capaz de alcanzar $2*(PRAB+tamaño_buffer)$ para lograr un caudal eficaz óptimo. La ventana anunciada del receptor, la cual es siempre un límite superior a la ventana de congestión, tiene que ser fijada a un valor suficientemente grande para este crecimiento. Durante Evasión de congestión se tiene:

$$\max_cwnd = PRAB + tamaño_buffer$$

$$ssthresh = \frac{(PRAB + tamaño_buffer)}{2} \tag{75}$$

esto es igual a

$$ssthresh = PRAB + \frac{(tamaño_buffer - PRAB)}{2} \tag{76}$$

Si el tamaño de buffer es mayor que el Producto Retardo-Ancho de Banda el término $\frac{(tamaño_buffer - PRAB)}{2}$ es mayor que cero e igual a la cantidad de paquetes que siempre permanecen o quedan en el buffer y causa congestión constante. Por otro lado si $\frac{(tamaño_buffer - PRAB)}{2}$ es negativo, ssthresh siempre estará por debajo del Producto

Retardo-Ancho de Banda y la conexión nunca va a ser capaz de lograr caudal eficaz óptimo. En este caso $\frac{(tamaño_buffer - PRAB)}{2}$ indica el número de paquetes que debiese haber en el buffer para evitar tiempo de enlace disponible. Las figuras 30, 32, 34 y 36 presentan los resultados para las variantes Tahoe, NewReno, SACK y Westwood, seguidas por la gráfica de [Lang, 2002] con la cual se validó el resultado obtenido.

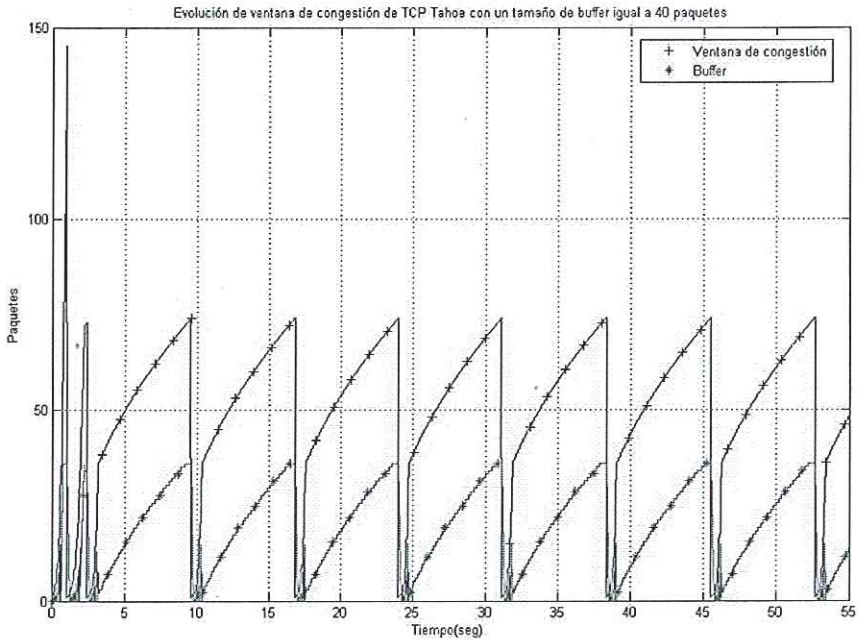


Figura 30. Evolución de ventana de congestión de TCP Tahoe con buffer de 40 paquetes

Se puede notar en la Figura 30 que cada vez que se sobrepasa el umbral de inicio lento y ocurre una pérdida, la transmisión se detiene, la ventana de congestión se reinicializa a un segmento y vuelve a comenzar su crecimiento.

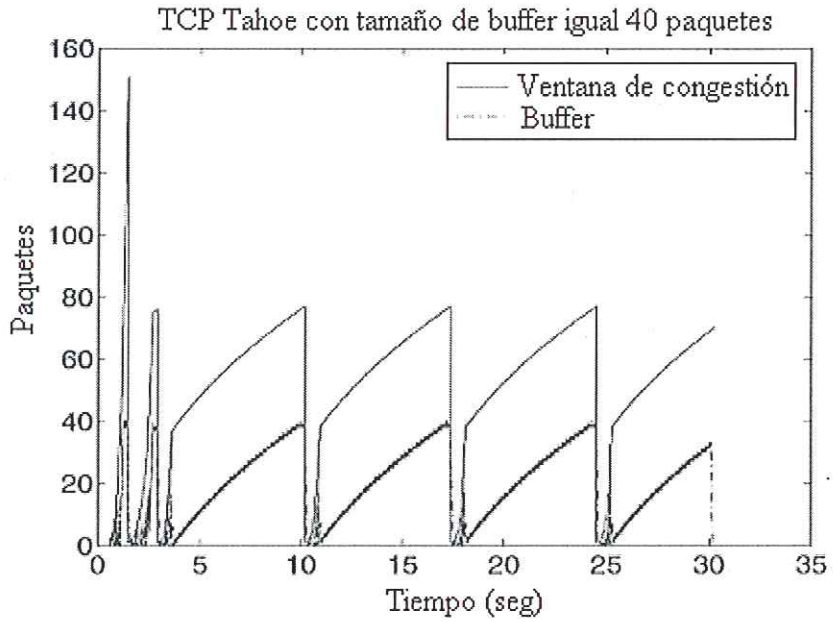


Figura 31. Evolución de ventana de congestión de TCP Tahoe [Lang, 2002]

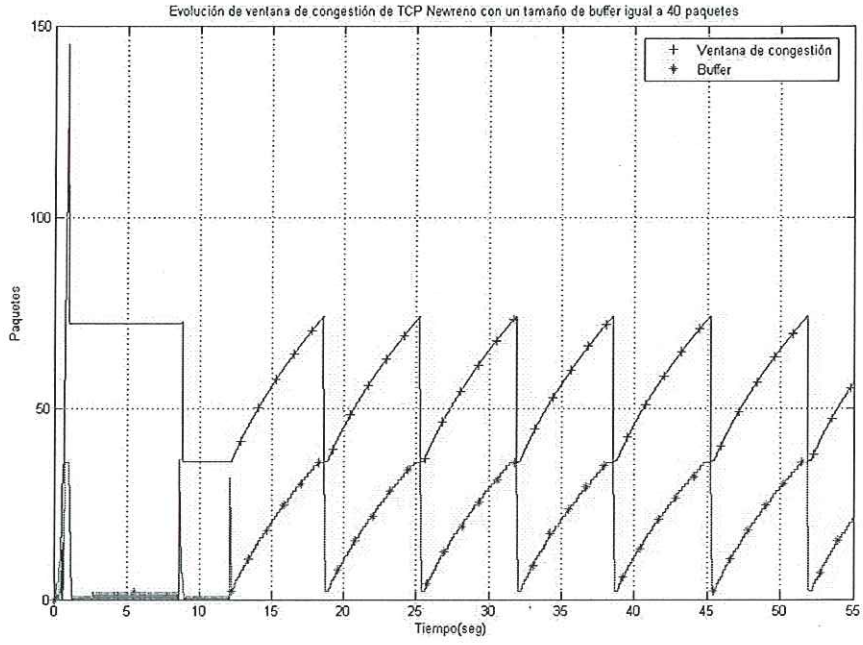


Figura 32. Evolución de ventana de congestión de TCP NewReno con buffer de 40 paquetes

En la Figura 32 se observa que después del crecimiento exponencial que registra la ventana, se presenta un escalón debido a que cuando NewReno experimenta pérdidas

continúa transmitiendo paquetes cada vez que recibe reconocimientos para no cortar la transmisión de manera inmediata. Estos reconocimientos son llamados parciales, y lo que hace NewReno cuando recibe un reconocimiento parcial es enviar el paquete siguiente al reconocimiento recibido para así poder tener una transmisión fluida. Debido a que está recuperando los paquetes poco a poco el buffer se encuentra vacío, y cuando logra recuperar el paquete perdido final vuelve a mandar una ráfaga de paquetes que desborda al buffer, y en la gráfica se ve la caída a un segundo escalón. Cuando recupera el último paquete perdido comienza la evasión de congestión con un umbral establecido de 40. Cuando se sobrepasa este límite, los reconocimientos que vienen de regreso no permiten que el canal esté vacío, logrando una ocupación del canal completa aún y cuando se presentaron pérdidas, pero solamente en la etapa de evasión que es donde se conoce el límite.

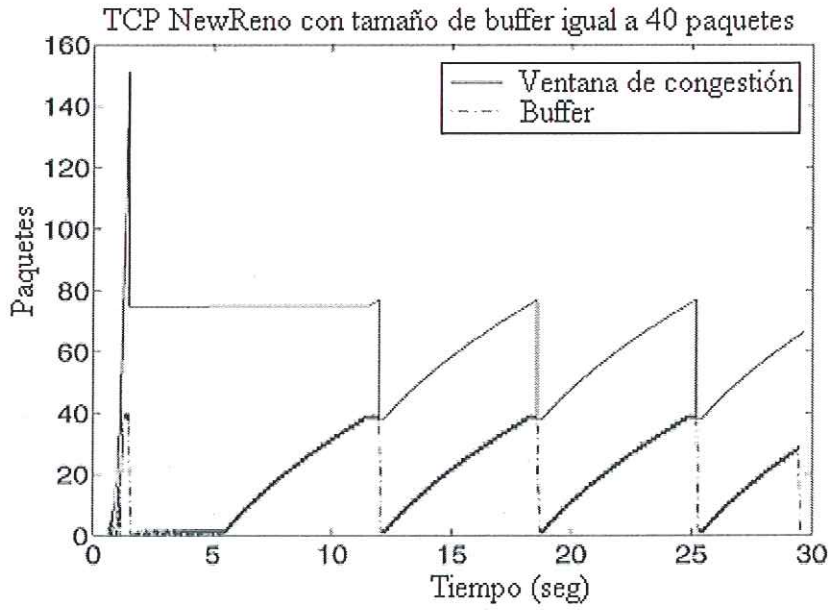


Figura 33. Evolución de ventana de congestión de TCP NewReno [Lang, 2002]

En la Figura 34 se encuentra la gráfica de la evolución de la ventana de congestión con TCP SACK. Después de que hay pérdidas se ve un pequeño escalón que se refiere a los paquetes que se recuperan, pero este no es tan grande como el de NewReno ya que en lugar de solicitar la retransmisión de todos los paquetes, simplemente se pide la retransmisión de los paquetes perdidos.

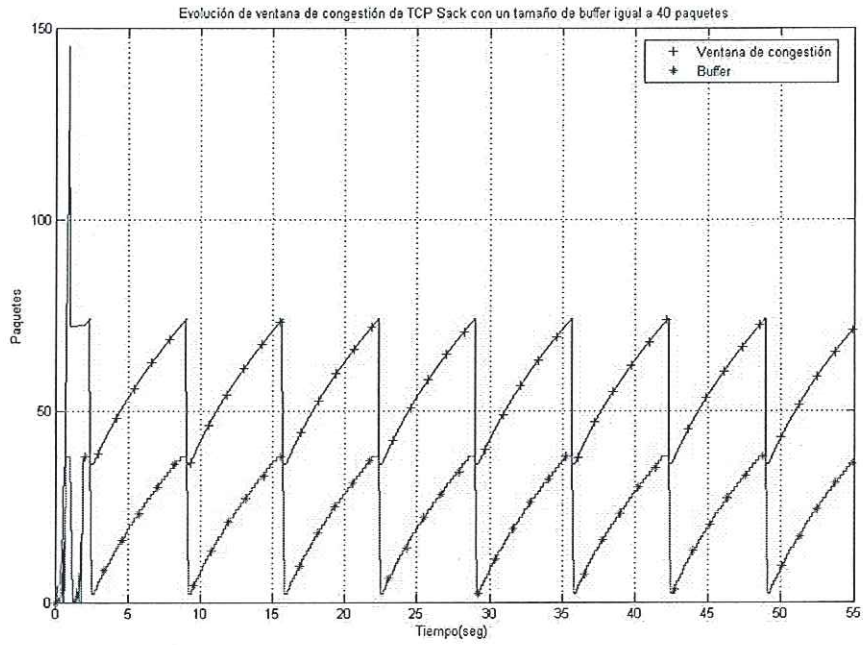


Figura 34. Evolución de ventana de congestión de TCP SACK con buffer de 40 paquetes

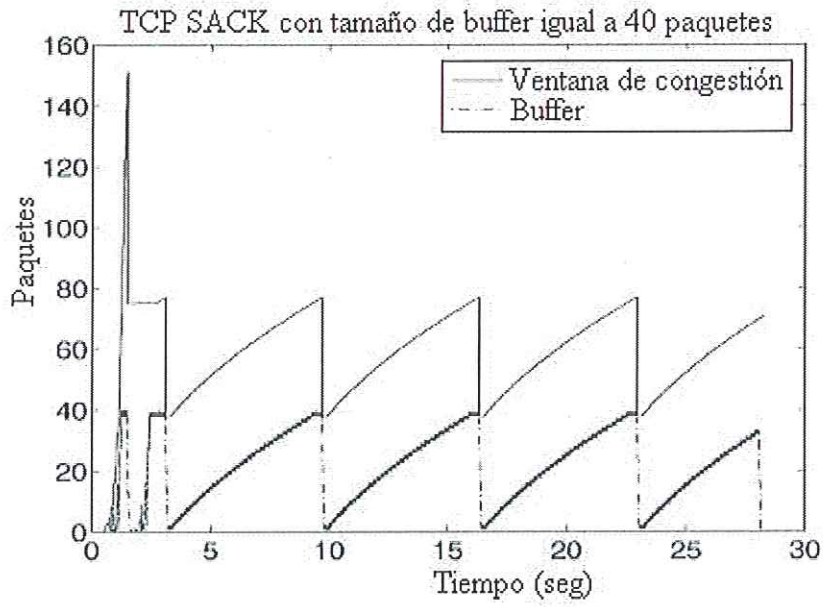


Figura 35. Evolución de ventana de congestión de TCP SACK [Lang, 2002]

En la Figura 36 se observa que TCP Westwood realiza un inicio muy agresivo, ya que de las variantes presentadas es la que logra un mayor crecimiento llegando casi a los

160 paquetes, lo que le sirve para determinar el umbral y comienza su paso hacia evasión de congestión más rápido que las demás variantes tanto en tiempo como en paquetes.

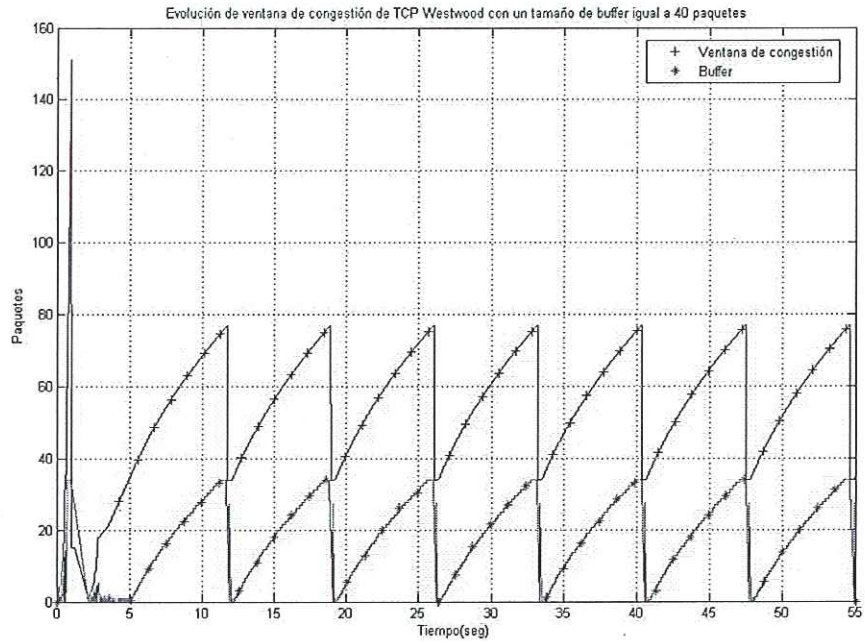


Figura 36. Evolución de ventana de congestión de TCP Westwood con buffer de 40 paquetes

V.3.1.1 Evaluación del desempeño de TCP mediante parámetros de Calidad de Servicio

A continuación se presentan los resultados obtenidos con la programación que se desarrolló en el simulador ns2 para evaluar el desempeño de las variantes de TCP (Tahoe, Reno, NewReno, SACK y Westwood) en un ambiente satelital con los parámetros de Calidad de Servicio como la latencia, variación en el retardo, tasa de paquetes perdidos y caudal eficaz. Para el cálculo de la tasa de error de bit se realizó el cálculo de presupuesto de enlace.

V.3.1.1.1 Latencia

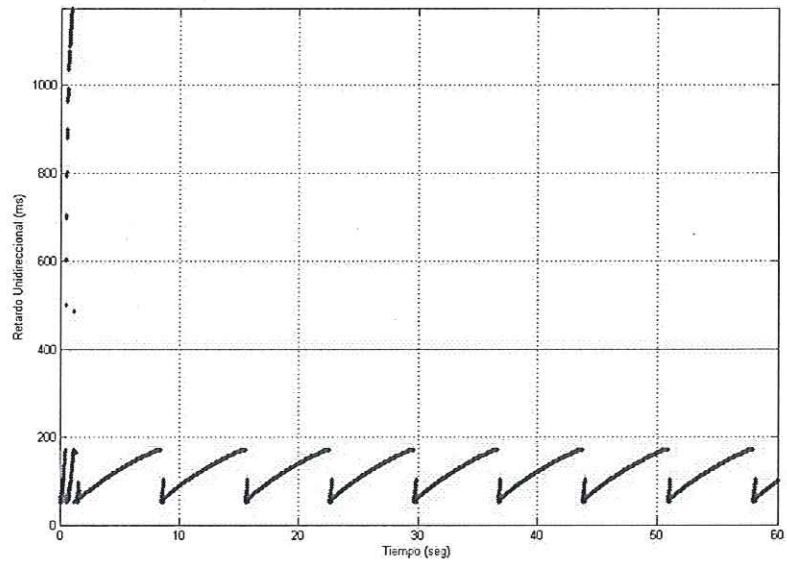


Figura 37. Retardo del paquete entre estaciones terrenas utilizando TCP Tahoe

La Figura 37 muestra la gráfica del retardo del paquete utilizando TCP Tahoe. Se observan tres fases que corresponden de manera directa al comportamiento presentado en la Figura 30 cuando se analizó el tamaño de buffer adecuado para la conexión. En la primera parte de la gráfica es donde se presenta el mayor retardo que experimenta uno de los paquetes transmitidos que es de aproximadamente 1175 ms y coincide con la etapa de inicio lento. Posterior a esto se presenta un segundo crecimiento que es donde recupera algunos de los paquetes perdidos a través del algoritmo de Retransmisión Rápida. A partir de este punto el transmisor ya determinó la capacidad del canal y disminuye por segunda ocasión el umbral de inicio lento, y esto se puede observar en la Figura 30 donde alrededor de los 36 paquetes comienza la etapa de evasión de congestión, mientras que en la Figura 37 se observa que a partir de los dos segundos comienza una forma periódica de la gráfica que representa la transmisión de los paquetes en la fase de evasión de congestión donde el crecimiento no es exponencial sino que a partir del umbral es de un paquete en uno y cuando se rebasa la capacidad del buffer y se pierde un paquete de deja de transmitir de ahí que en la gráfica no se observe una continuidad de la curva.

La Figura 38 corresponde a la gráfica del retardo para la variante TCP Reno. La primera línea vertical de retardos que se observa corresponde al inicio lento de TCP Reno, el cual es inclusive mayor al máximo registrado de TCP Tahoe. Esto se debe al hecho de que cuando se presentan las primeras pérdidas como está recibiendo reconocimientos logra enviar unos paquetes más. TCP Reno tiene un mecanismo adicional llamado Recuperación Rápida. Este mecanismo sólo se ejecuta cuando los paquetes perdidos los detecta a través de Retransmisión Rápida, y en la gráfica se observa que después de registrar un retardo máximo ya no realiza otro inicio lento con un umbral reducido como Tahoe si no por el contrario pasa a la etapa periódica de evasión de congestión manteniéndose por debajo del límite de 400 ms.

La Figura 39 es la variante TCP NewReno y es la primera que presenta un retardo por debajo de los 400 ms. Esto se debe al hecho de que a pesar de que se registran pérdidas ésta variante no corta la transmisión y por el contrario cada vez que recibe un reconocimiento envía el segmento siguiente, con lo que se mantiene el flujo de segmentos.

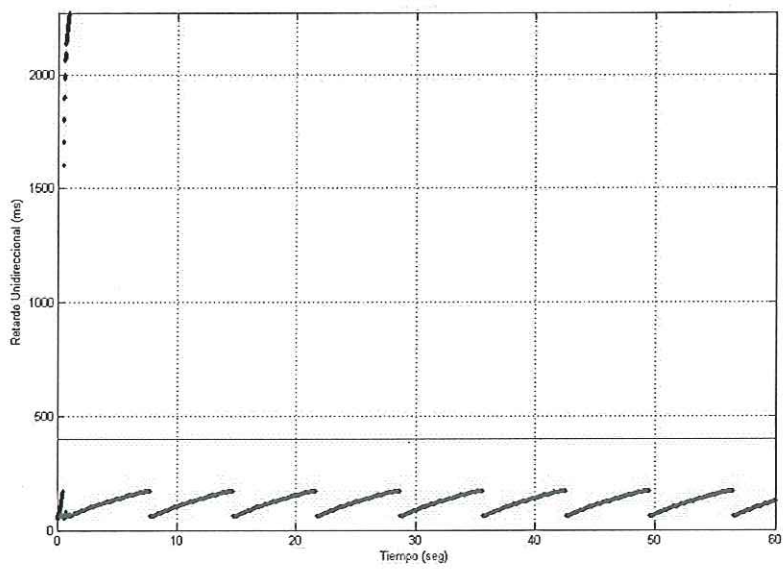


Figura 38. Retardo del paquete entre estaciones terrenas utilizando TCP Reno

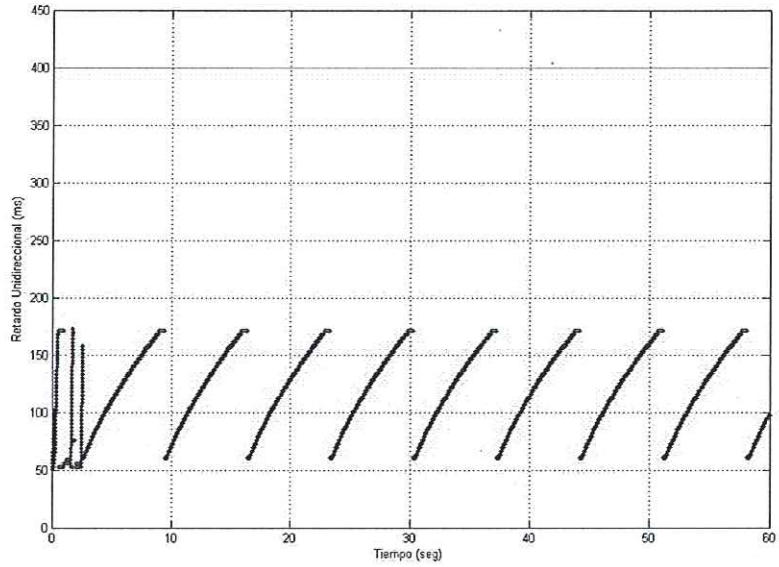


Figura 39. Retardo del paquete entre estaciones terrenas utilizando TCP NewReno

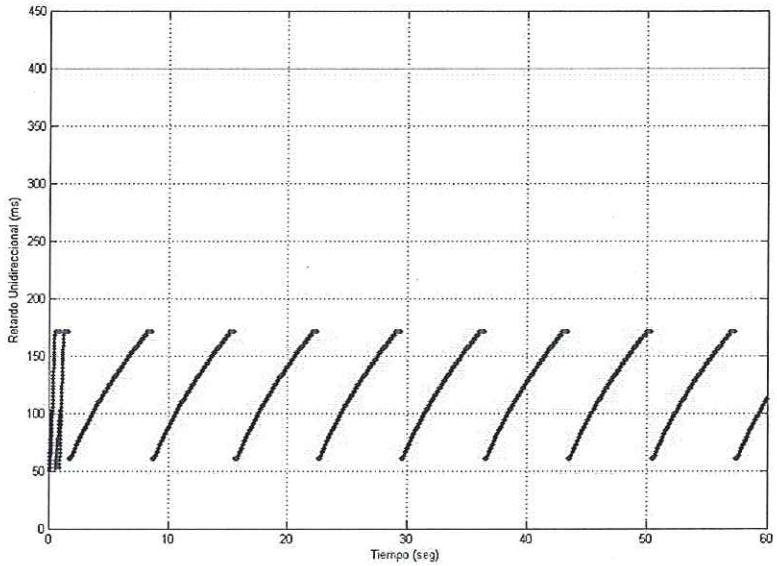


Figura 40. Retardo del paquete entre estaciones terrenas utilizando TCP SACK

La Figura 40 muestra el retardo de los paquetes para la variante TCP SACK que también presenta un retardo máximo por debajo de los 400 ms. Esto se debe a que cuando se presentan pérdidas esta variante solo solicita la retransmisión de los paquetes que se perdieron y se evita la retransmisión de todos.

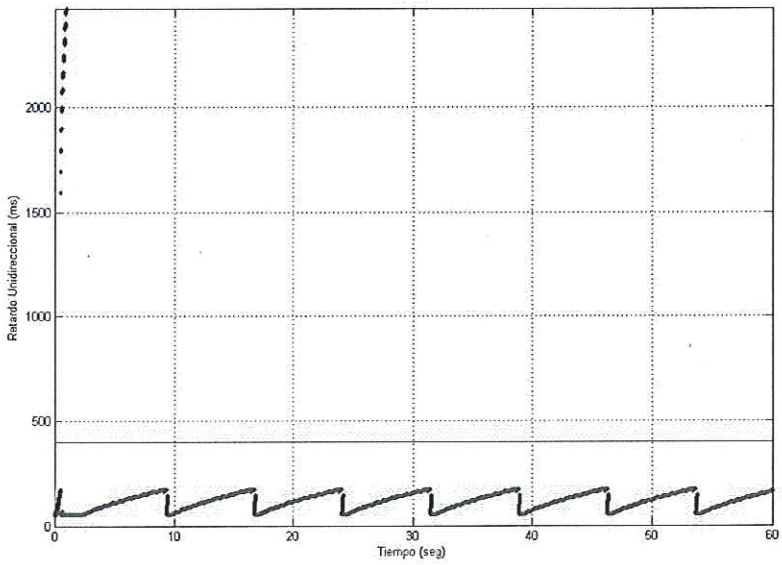


Figura 41. Retardo del paquete entre estaciones terrenas utilizando TCP Westwood

La Figura 41 muestra la gráfica del retardo de los paquetes para la variante TCP Westwood. De las cinco variantes de TCP es la que representa el mayor retardo, esto debido a que en un principio transmite de manera agresiva para determinar el nivel de congestión del canal y de esa manera manejar su comportamiento en las etapas posteriores con respecto al sensado que hace de la congestión.

En la Tabla III se encuentran las estadísticas obtenidas del proceso de simulación en lo que respecta al retardo de los paquetes entre estaciones terrenas para las distintas variantes de TCP, donde se observa que solo dos variantes logran estar por debajo de los 400 ms que fueron NewReno y SACK. Con respecto a las tres variantes restantes (Tahoe, Reno y Westwood) lo que las perjudica es la fase de inicio lento ya que presentan valores muy elevados de retardo y en los tres casos al pasar a la fase de evasión de congestión las tres variantes registran un comportamiento por debajo del límite que se está manejando de 400 ms. Al observar los valores de desviación estandar que se obtuvieron siendo los mayores los de Reno y Westwood se cree que es poco probable que influyan en los valores de retardo haciendo que se llegue o rebase el límite establecido de 400 ms.

Tabla III. Estadísticas obtenidas del retardo

Variantes de TCP	Retardo Límite Máximo Establecido de 400ms
Tahoe	Máximo: 1175.4 ms
	Mínimo: 50.213 ms
	Promedio: 121.82 ms
	Desviación Estandar: 66.089
Reno	Máximo:2272.6 ms
	Mínimo: 50.213 ms
	Promedio: 121.82 ms
	Desviación Estandar: 128.18
NewReno	Máximo: 172.88 ms
	Mínimo: 50.213 ms
	Promedio: 121.82 ms
	Desviación Estandar: 35.22
SACK	Máximo: 170.97 ms
	Mínimo: 50.213 ms
	Promedio: 124.89 ms
	Desviación Estandar: 34.27
Westwood	Máximo: 2474.9 ms
	Mínimo: 50.213 ms
	Promedio: 121.82 ms
	Desviación Estandar: 132.27

V.3.1.1.2 Variación en el retardo (Jitter)

De las gráficas obtenidas la etapa de evasión de congestión es evidente en cada una de las variantes y a continuación se explica para cada una.

En el caso de la variación en el retardo de los paquetes cuando se utiliza TCP Tahoe este se presenta cuando el buffer se desborda, y la transmisión vuelve a comenzar pasado un período de expiración o después de que se reciben reconocimientos duplicados, de ahí que se tengan estas diferencias en el arribo de los paquetes. La Figura 42 muestra la gráfica del jitter utilizando TCP Tahoe. Dentro de los valores obtenidos la mayoría están por

debajo de los 20 ms. La primera línea vertical corresponde al inicio lento, mientras que los siguientes puntos ordenados verticalmente corresponden a la fase periódica de evasión de congestión. Se observa que la mayoría de los paquetes transmitidos son recibidos de manera inmediata por el receptor pero los puntos que están ordenados de manera vertical representan aquellos paquetes que pierden una continuidad en su transmisión y encolan un pequeño tiempo en el buffer.

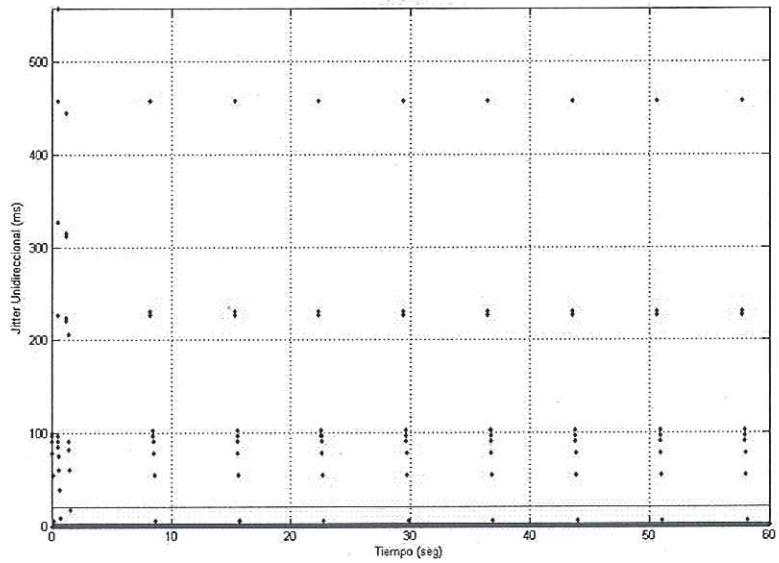


Figura 42. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP Tahoe

La Figura 43 presenta la gráfica del jitter utilizando TCP Reno. Los valores de jitter mayores se presentan en el inicio lento teniendo un máximo de 2758 ms y al igual que en Tahoe, se observan unas parejas de puntos ordenados de manera vertical que representan el momento en que ocurre pérdida en evasión de congestión y una vez que regresan los reconocimientos esta se reanuda.

La Figura 44 presenta la gráfica del jitter utilizando TCP NewReno. La diferencia con respecto a TCP Reno y TCP Tahoe se nota en el inicio lento, mientras que la evasión de congestión se comporta de manera muy similar a TCP Reno. Durante inicio lento cuando se presentan las pérdidas de los paquetes como continua recibiendo algunos reconocimientos TCP NewReno asume que algunos paquetes si llegaron a su destino por lo que en lugar de dejar de transmitir, cada vez que recibe un reconocimiento transmite el

paquete siguiente al reconocimiento del paquete que recibe, de ahí que se obtengan estos valores de jitter tan elevados. De manera igual a TCP Reno sucede en NewReno con los puntos obtenidos de la evasión de congestión.

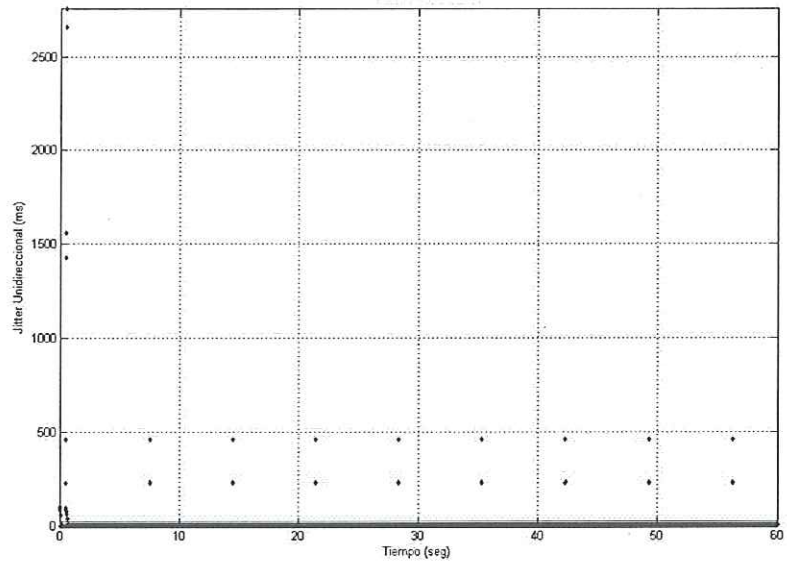


Figura 43. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP Reno

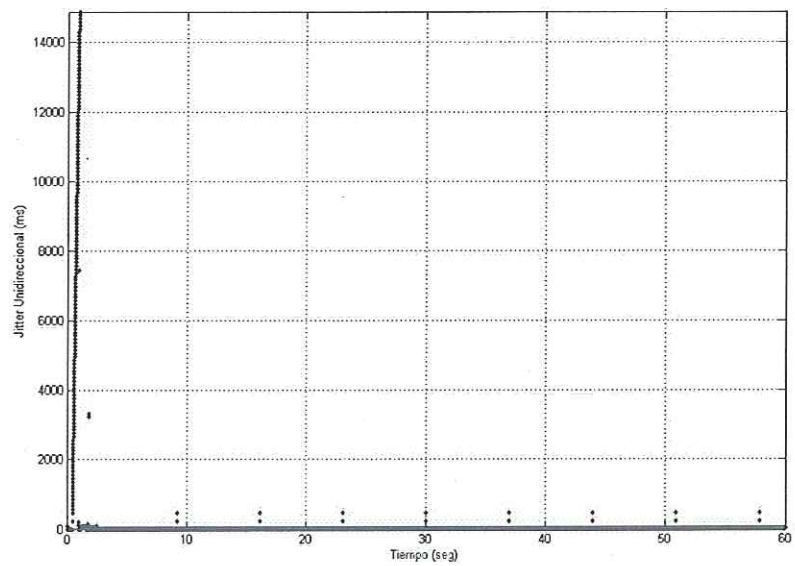


Figura 44. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP NewReno

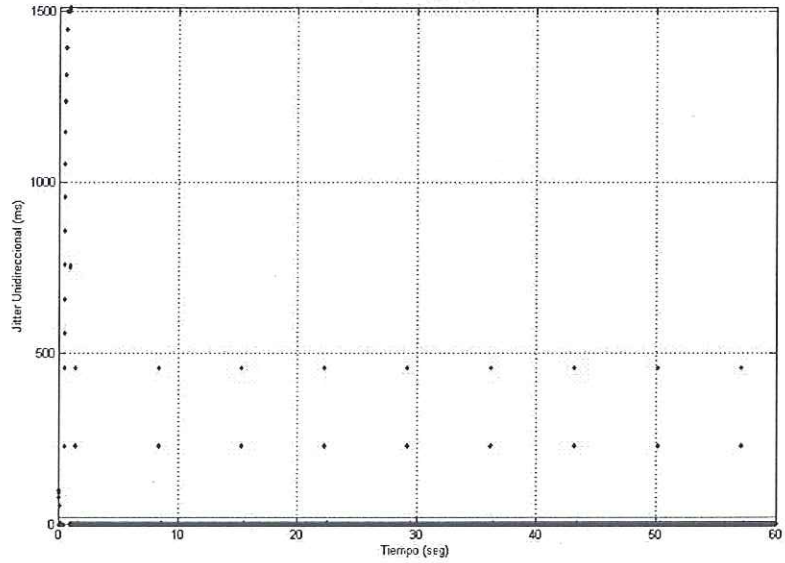


Figura 45. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP SACK

La Figura 45 muestra la gráfica del jitter para TCP SACK. A diferencia de TCP Reno y TCP NewReno, TCP SACK presenta un valor menor de jitter máximo el cual está alrededor de 1512 ms el cual como quiera está muy por encima del límite de 20ms. De manera comparativa el reconocimiento selectivo ayuda a mejorar el valor de jitter durante la fase de inicio lento.

La Figura 46 muestra la gráfica del jitter para TCP Westwood. Esta variante realiza una transmisión agresiva durante el inicio lento, y aunque no tiene demasiados paquetes con variación en el retardo con esos pocos llega a tener valores por encima de 1000 ms Al pasar a la etapa de evasión de congestión se observa que a diferencia de TCP Reno, NewReno y SACK, los valores que obtiene de jitter no son solamente parejas de puntos sino que son tercias, y esto se puede deber a que en su estrategia agresiva de transmisión de paquetes, antes de cruzar el umbral de inicio lento el encolamiento que se genera en el buffer es el que da el resultado de jitter para estos paquetes.

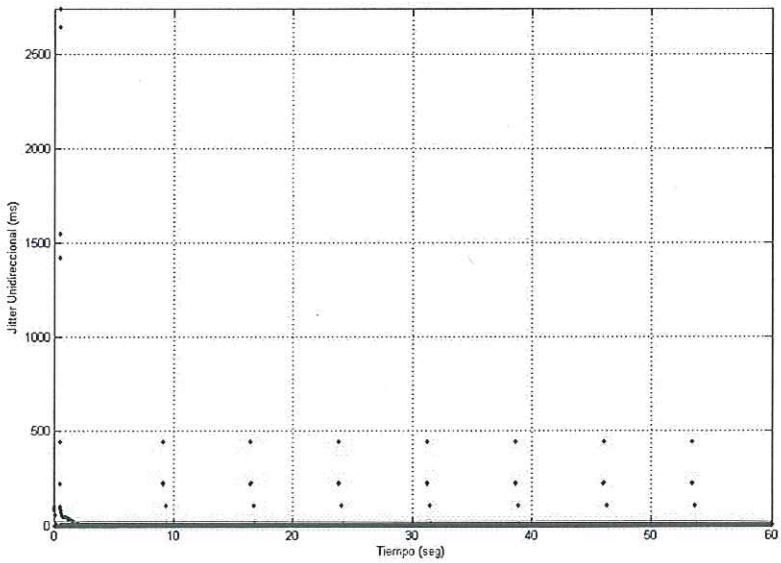


Figura 46. Variación en el retardo del paquete entre estaciones terrenas utilizando TCP Westwood

La Tabla IV muestra las estadísticas obtenidas para variación en el retardo y se observa que en cuanto a los resultados obtenidos y comparando con el límite de 20 ms que se está manejando para evaluar el desempeño de las cinco variantes todas tienen un comportamiento deficiente.

V.3.1.1.3 Tasa de error de bit (BER)

Para el cálculo de la Tasa de Error de Bit (BER) se consideró un solo satélite repetidor y los parámetros usados para obtener la Figura 47 y la Tabla V son los siguientes: Potencia del transmisor de la estación terrena transmisora de 80 W, diámetro de la antena transmisora de 8 m, potencia del transmisor de la estación terrena receptora de 3 W y diámetro de la antena receptora de 0.5 m.

La Figura 47 muestra la variación que tiene la Tasa de Error de Bit (BER) para el enlace propuesto a distintos ángulos de elevación. El cálculo del BER depende de la distancia que existe entre el satélite y la estación terrena. El valor mínimo de BER se presenta cuando el satélite se encuentra a 90° de la estación terrena que viene a ser la menor distancia que se va a registrar entre el satélite y la estación terrena. Cuando el satélite se encuentra a 40° o 140° de elevación en el horizonte los valores de BER van a estar en el

límite establecido de $1x10^{-5}$ como se observa en la gráfica y cuando el satélite se encuentre por debajo de los 40° o por encima de los 140° no se asegura una confiabilidad en la información transmitida.

Tabla IV. Estadísticas obtenidas de la variación en el retardo (Jitter)

Variantes de TCP	Variación en el Retardo (Jitter) Límite Máximo Establecido de 20ms
Tahoe	Máximo: 557.94 ms
	Mínimo: 0 ms
	Promedio: $3.5527x10^{-12}$ ms
	Desviación Estandar: 15.91
Reno	Máximo:2758.4 ms
	Mínimo: 0 ms
	Promedio: $1.7764x10^{-12}$ ms
	Desviación Estandar: 34.367
NewReno	Máximo: 14888 ms
	Mínimo: 0 ms
	Promedio: $7.1054x10^{-12}$ ms
	Desviación Estandar: 771.32
SACK	Máximo: 1512.3 ms
	Mínimo: 0 ms
	Promedio: $1.7764x10^{-12}$ ms
	Desviación Estandar: 123.86
Westwood	Máximo: 2746.1 ms
	Mínimo: 0 ms
	Promedio: $3.5527x10^{-12}$ ms
	Desviación Estandar: 34.024

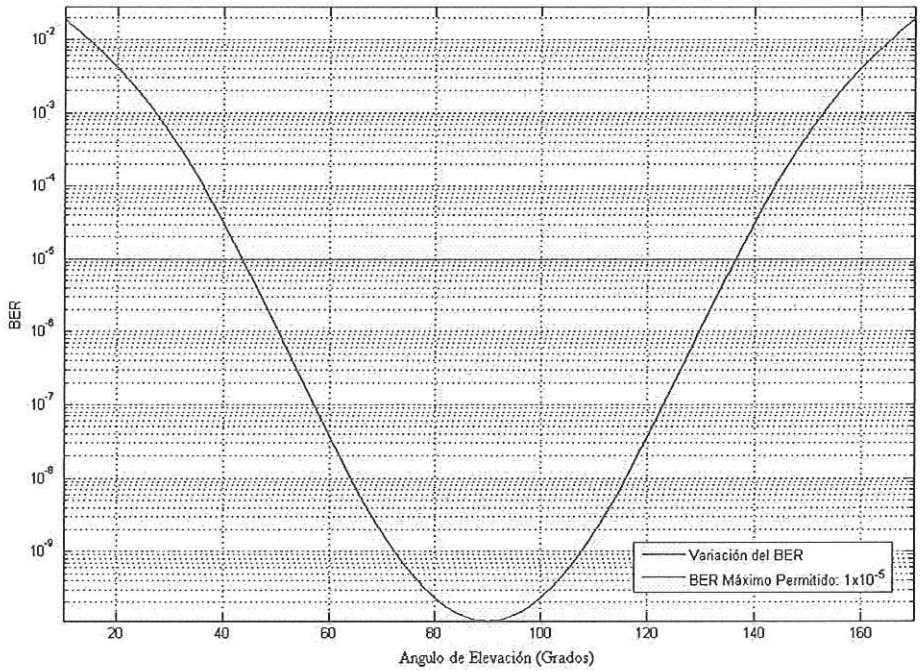


Figura 47. Variación de la Tasa de Error de Bit (BER) en el enlace entre Ensenada y México, D. F. con respecto a distintos ángulos de elevación

La Tabla V reúne las estadísticas obtenidas sobre la relación señal portadora a ruido (C/N) y Tasa de Error de Bit (BER) obtenidas del escenario de simulación en este trabajo de tesis. La tabla muestra que de acuerdo con el escenario planteado y el análisis realizado se puede obtener un BER mínimo de 1.1191×10^{-10} lo cual es un resultado más que satisfactorio, pero como se mencionó antes los valores aceptables del BER para este caso tienen la limitante de los ángulos de elevación entre los cuales se puede asegurar una confiabilidad en la información transmitida.

V.3.1.1.4 Tasa de paquetes perdidos (PLR) y Caudal Eficaz

En la Tabla VI se presentan los valores de Tasa de Paquetes Perdidos y Caudal Eficaz obtenidos para las distintas variantes de TCP. En cada variante se presentan los resultados con tres tamaños de buffer como se exploraron en la sección V.3.1. Se observó la importancia de determinar cual es el Producto Retardo-Ancho de Banda en relación a los paquetes que se requieren para ocupar el canal y en base a eso establecer que el tamaño de

buffer va a estar relacionado con el Producto Retardo-Ancho de Banda en paquetes. Al realizar esto se observa que el caudal eficaz óptimo de TCP se logra cuando el tamaño de buffer es igual al Producto Retardo-Ancho de Banda.

Tabla V. Estadísticas de la Relación Señal Portadora a Ruido (C/N) y Tasa de Error de Bit (BER) obtenidas

C/N Total	C/N Mínimo	7.6961 dB
	C/N Máximo	24.128 dB
	C/N Promedio	21.044 dB
	Desviación Estandar	4.6475
Tasa de Error de Bit (BER) Máximo de $1x10^{-5}$	BER Mínimo	$1.1191x10^{-10}$
	BER Máximo	$19.108x10^{-3}$
	BER Promedio	$1.1434x10^{-6}$
	Desviación Estandar	0.0039042

En los experimentos de simulación que se realizaron se modificaron los tamaños de buffer para tener tamaños menores y mayores al Producto Retardo-Ancho de Banda y los resultados obtenidos se presentan en la Tabla VI donde se observa que cuando el tamaño de buffer es menor al Producto Retardo-Ancho de Banda se tiene un caudal eficaz un tanto menor al óptimo, mientras que si se establece un tamaño de buffer mayor al Producto Retardo-Ancho de Banda el caudal eficaz tiende a disminuir ligeramente. Esto se puede deber a que al tener un tamaño de buffer mayor el impacto que llegue a presentar el inicio lento y las pérdidas iniciales no va a ser tan drástico o severo ya que la mayoría del tiempo de conexión se encuentra en la fase estacionaria, mientras que en el caso de que el tamaño de buffer sea menor al Producto Retardo-Ancho de Banda se tiene un caudal eficaz menor al óptimo y se puede deber al hecho de que el inicio lento va a estar impactando y cortando la transmisión y el cambio a la fase de evasión de congestión se retarda, aparte de que en

inicio lento no se logra una ocupación total del canal y menos si se define un tamaño de buffer pequeño en el que se estén desbordando paquetes de manera continua.

Tabla VI. Estadísticas de Tasa de Paquetes Perdidos (PLR) y Caudal Eficaz obtenidas para las distintas variantes de TCP y diferentes tamaños de buffer

Variantes de TCP	Buffer (Paquetes)	PLR (%)	Caudal Eficaz (%)
Tahoe	20	.745029 %	99.255 %
	40	.557432 %	99.4426 %
	60	.65391 %	99.3461 %
Reno	20	. 734309 %	99.2657 %
	40	.552072 %	99.4479 %
	60	.64319 %	99.3568 %
NewReno	20	.578871 %	99.4211 %
	40	.412714 %	99.5873 %
	60	.487753 %	99.5122 %
SACK	20	.418074 %	99.5819 %
	40	.262636 %	99.7374 %
	60	.326955 %	99.673 %
Westwood	20	.986225 %	99.0138 %
	40	.316235 %	99.6838 %
	60	.407354 %	99.5926 %

En lo que respecta a las estadísticas del PLR también se obtuvieron del simulador y son coherentes con los resultados de caudal eficaz presentados. Las cinco variantes presentan valores de PLR por debajo del límite establecido de 3% a 5%.

V.4 Conclusiones

En este capítulo se presentaron los resultados obtenidos del proceso de simulación de un enlace entre dos estaciones terrenas mediante un satélite repetidor de órbita baja. El funcionamiento del simulador se validó con el trabajo de tesis doctoral de investigación de [Lang, 2002].

Se mostró la importancia de definir un tamaño de buffer igual al Producto Retardo-Ancho de Banda, y se presentaron los casos con sus respectivas consecuencias que tiene en el sistema al tener un buffer menor o mayor al Producto Retardo-Ancho de Banda. Se evaluó el desempeño de las cinco variantes de TCP propuestas (Tahoe, Reno, NewReno, SACK y Westwood) con ayuda de los parámetros de Calidad de Servicio para Voz sobre IP.

Con respecto a los resultados de retardo, sólo dos variantes lograron estar por debajo del límite fijado de 400 ms, las cuales fueron SACK y NewReno.Respecto a las otras tres variantes (Tahoe, Reno y Westwood) si se ignorase el comportamiento de retardo excesivo que experimentan durante el inicio lento se puede decir que su comportamiento es aceptable, ya que durante la evasión de congestión no rebasan los 400 ms, de ahí que se diga que estas tres variantes tienen un comportamiento regular.

En la variación del arribo de los paquetes o jitter ninguna variante logró un desempeño aceptable, y todas estuvieron muy por encima del límite fijado de 20 ms, lo cual se debe al inicio lento y los paquetes perdidos durante esta fase y que requieren de ser recuperados y después en la fase de evasión de congestión se unos momentos donde el jitter rebasa de manera muy evidente los 20 ms, lo cual se debe a la pérdida de paquetes y a la pequeña falta de continuidad que se presenta en la transmisión cuando se está llenando el buffer.

En cuando al análisis de la tasa de error de bit se obtuvo que mientras el satélite tenga un ángulo de elevación entre 40° y 140° grados se puede tener un BER menor al límite, con lo que se puede tener una transmisión casi libre de errores. Los resultados obtenidos de PLR para las cinco variantes se encuentran por debajo del 3-5 % establecido y para el caudal eficaz se estableció que para tener un caudal óptimo se requiere que el

tamaño de buffer sea igual al Producto Retardo-Ancho de Banda, ya que si se tienen tamaños de buffer menores o mayores se obtienen valores de caudal menores al óptimo.

En el siguiente capítulo se presentarán las conclusiones de este trabajo, así como sus principales aportaciones y el trabajo futuro que se propone.

VI. CONCLUSIONES

A continuación se presentan las principales conclusiones y aportaciones de este trabajo de investigación.

De las simulaciones realizadas en el simulador de red Network Simulator 2 (ns2) se obtuvieron los siguientes resultados:

En el análisis del retardo, sólo dos variantes (SACK y NewReno) de las cinco analizadas estuvieron por debajo de los 400 ms, debido a la forma particular en que opera su algoritmo de congestión. Las otras variantes cortan su transmisión en cuanto se presenta la pérdida de paquetes, como es el caso de TCP Tahoe, entran en expiración como TCP Reno o disminuyen su tasa de transmisión como TCP Westwood. Del resultado obtenido en lo que respecta al análisis del retardo cuando se utilice TCP por satélite las dos variantes que se deben considerar son SACK y NewReno.

En el análisis de la variación en el arribo de los paquetes, o jitter, las cinco variantes presentaron resultados muy por encima del límite de 20 ms. Aunque SACK y NewReno fueron las únicas variantes que estuvieron dentro del límite tolerable de retardo eso no implica que sus valores de jitter sean aceptables. En el análisis de las cinco variantes que se realizó, todas presentaron un comportamiento deficiente.

En el análisis de la tasa de error de bit (BER) se realizó el cálculo con el presupuesto de enlace, obteniendo valores satisfactorios con ángulos de elevación entre 40° y 140° aproximadamente. Dentro de ese intervalo se pueden llegar a tener valores de BER tan bajos como 1.11×10^{-10}. Cuando la elevación de la antena se encuentre fuera del intervalo entre 40° y 140° no se asegura confiabilidad entre la información transmitida y recibida por el satélite.

En el análisis de la tasa de paquetes perdidos (PLR), las cinco variantes logran estar por debajo del límite que establece la norma de 3% a 5%, lo que indica un desempeño satisfactorio.

Con respecto al caudal eficaz (CE), se analizó el impacto que tiene el Producto Retardo-Ancho de Banda en el caudal eficaz para las cinco variantes de TCP: Tahoe, Reno,

NewReno, SACK y Westwood. Se recomienda que el tamaño de buffer mínimo sea igual o similar al Producto Retardo-Ancho de Banda, ya que tamaños menores o mayores no permiten que se alcance un caudal óptimo.

De acuerdo con los resultados obtenidos, se puede concluir de la evaluación realizada a las variantes de TCP con los parámetros de QoS para VoIP, que su desempeño no es aceptable. A pesar de que SACK y NewReno logran tener retardos de paquete extremo a extremo bastante menores a la norma establecida, este análisis fue realizado considerando el enlace satelital para cada variante al momento de realizar la simulación. Existe la posibilidad de que se incrementen los valores de retardo si se agrega tráfico de fondo o movilidad, lo que sucedería también con bastante probabilidad con los valores de jitter y PLR.

Con respecto a las aportaciones de este trabajo en cuanto a TCP, se realizó un modelado matemático que considera las etapas más representativas, las cuales incluyen dinámicas de la ventana (incremento lineal y disminución multiplicativa), así como el modelo de pérdida de paquetes y de expiraciones. Esto permite una caracterización de la tasa de transmisión de una fuente TCP con dos variantes, donde el receptor limita o no el tamaño de ventana, limitando su tasa de transmisión.

En cuanto a los algoritmos de control de congestión de las variantes de TCP, se elaboraron modelos gráficos que mediante diagramas de flujo expresan de manera clara su funcionamiento. Se muestran cada una de las etapas de las que están compuestas, tal como Inicio Lento, Evasión de Congestión, Retransmisión Rápida, Recuperación Rápida y la estrategia seguida cuando se presenta la pérdida de paquetes. Estos mismos modelos se implementaron en el simulador de red Network Simulator 2 (ns2) para evaluar su desempeño.

También se realizó un modelado matemático de los parámetros de QoS (retardo, jitter, BER, PLR y CE) para un ambiente satelital, fijando valores límite correspondientes a la aplicación de VoIP.

Dentro de la investigación realizada, no se encontraron publicaciones con la evaluación del desempeño de TCP mediante parámetros de QoS, por lo que esto se considera una de las principales aportaciones de este trabajo.

De acuerdo al objetivo planteado al inicio del trabajo, se estudiaron y se modelaron los mecanismos de control de congestión en redes IP de cinco variantes de TCP para transmisión de información, se estudiaron y modelaron los parámetros de QoS para VoIP por satélite y se simularon transferencias de información con variantes de TCP en el simulador de red Network Simulator 2 (ns2) para evaluar su desempeño.

Dentro de las recomendaciones de trabajo futuro, se propone lo siguiente:

- Evaluar las variantes de TCP SACK y NewReno con algunas de las mejoras propuestas en el RFC 2488 y el RFC 1323. Esto implicaría manejar el escalamiento de ventana, con tráfico de fondo y movilidad, y comparar los resultados obtenidos con lo presentado en este trabajo, y determinar si mejora el desempeño en un ambien satelital. Se recomienda además evaluar TCP Vegas en un ambiente satelital de órbita baja con las mismas condiciones.
- Otra posible línea de investigación sería analizar y evaluar el Protocolo de Transporte Satelital (STP), que es un protocolo de transporte específicamente diseñado para ambientes satelitales y representa una alternativa al uso de TCP como se menciona en [Henderson y Katz, 1997].
- Por último otra opción al bajo desempeño de TCP en ambientes satelitales sería el análisis y evaluación de un Proxy para Mejorar el Desempeño (PEP, Performance Enhancing Proxy), ya que propone aislar el enlace satelital del resto de la red y ofrece distintas opciones como el TCP Dividido y el Engaño a TCP (TCP Spoofing) mencionados en la sección II.3.4 y en [Hu y Li, 2001]

En base a lo presentado en este documento de tesis se puede concluir que TCP SACK es la variante que presenta el mejor desempeño en general, evaluando su funcionamiento con parámetros de QoS para VoIP en un ambiente satelital de órbita baja, seguida de TCP Westwood. SACK y NewReno presentan el mejor comportamiento para aplicaciones con restricciones en el retardo. Para aplicaciones donde no afecte el impacto del jitter TCP SACK o Tahoe representan una buena alternativa sobre satélites de órbita baja. TCP SACK y Westwood presentan los menores porcentajes de PLR y de manera coherente los mayores porcentajes de Caudal Eficaz. Se enfatiza que los resultados presentados pueden variar si se agrega tráfico de fondo en la red o movilidad del satélite.

REFERENCIAS

- Allman, M., Glover, D. y Sanchez, L. 1999. Enhancing TCP Over Satellite Channels using Standard Mechanisms. RFC 2488. IETF. http://www.faqs.org/rfcs/rfc2488.html. Octubre 2004.

- Allman, M., Paxson, V. y Stevens, W. 1999. TCP Congestion Control. RFC 2581. IETF. http://www.faqs.org/rfcs/rfc2581.html. Octubre 2004.

- Allman, M., Glover, D. y Sanchez, L. 1999. TCP Congestion Control. RFC 2488. IETF. http://www.faqs.org/rfcs/rfc2488.html. Octubre 2004.

- Altman, E., Avrachenkov, K. y Barakat, C. 2000. A Stochastic Model of TCP/IP with Stationary Random Losses. Proceedings of ACM SIGCOMM. Estocolmo, Suecia. Agosto 2000.

- Barakat, C. 2001. TCP/IP Modeling and Validation. IEEE Network. 15(3):38-47 p.

- Barakat, C., Altman, E. y Dabbous, W. 2000. On TCP Performance in a heterogeneous network: A Survey. IEEE Communications Magazine. 38(1):40-46 p.

- Beltrán, R. 1999. Impactos al implementar el servicio de voz a través de una red IP. Tesis de Maestría en Ciencias, CICESE, Ensenada, B.C. 118 p.

- Braden, R. 1989. Requirements for Internet Hosts – Communications Layers. RFC 1123. IETF. http://www.faqs.org/rfcs/rfc1123.html. Octubre 2004.

- Cabanillas Noris, J. C. 2004. Análisis de Satélites con Procesamiento a Bordo y su Impacto en QoS para Redes de Banda Ancha por Satélite. Tesis de Maestría en Ciencias, CICESE, Ensenada, B.C. 177 p.

- Chaskar, N.M., Lakshman, T.V. y Madhow, U. 1999. TCP over Wireless with Link Level Error Control: Analysis and Design Methodology. IEEE/ACM Transactions on Networking. 7(5): 605-615 p.

- Conte, R., 2000. Satellite Rural Telephone Network Design: A Methodology for Performance Optimization. Tesis de Doctorado en Filosofía en Ingeniería Eléctrica y Computación en el Instituto Politécnico de Virginia, Blacksburg, Virginia. 197 p.

- Durst, R.C., Miller, G.J. y Travis, E.J. 1997. TCP Extensions for Space Communications. Wireless Networks. 3(5): 389-403 p.

- Elbert, B. R. 2001. The Satellite Communication Ground Segment and Earth Station Handbook. Artech House. Boston. 370 p.

- Fall, K. y Varadhan, K. 2004. The ns Manual. The VINT Project. Una Colaboración entre investigadores de UC Berkeley, LBL, USC/ISI y XEROX PARC. 391 p. http://www.isi.edu/nsnam/ns/ns-documentation.html. Noviembre 2004.

- Floyd, S. y Henderson, T. 1999. The NewReno modification to TCP´s Fast Recovery Algorithm. RFC 2582. IETF. http://www.faqs.org/rfcs/rfc2582.html. Octubre 2004.

- Ghani, N. y Dixit, S. 1999. TCP/IP Enhancements for Satellite Networks. IEEE Communications Magazine. 37(7): 64-72 p.

- González, R. I. 2000. Análisis del Sistema de Señalización No. 7 para el Transporte de Voz Mediante el Protocolo de IP. Tesis de Maestría en Ciencias, CICESE, Ensenada, B.C. 141 p.

- Goyal, R., Jain, R., Goyal, M., Fahmy, S., Vandalote, B. y Kota, S. 1999. Traffic Management for TCP/IP over Satellite ATM Networks. IEEE Communications Magazine. 37(3): 56-61 p.

- Hassan, M. y Atiquzzaman, M. 2000. Performance of TCP/IP over ATM Networks. Artech House. Boston. 236 p.

- Hassan, M. y Jain, R. 2004. High Performance TCP/IP Networking. Concepts, Issues, and Solutions. Prentice Hall. Upper Sadle River, New Jersey. 383 p.

- Henderson, T. R. y Katz, R. H. 1999. TCP Performance over Satellite Channels. Universidad de California en Berkeley. Computer Science Technical Report 99-1083.

- Henderson, T. R. y Katz, R. H. 1997.Satellite Transport Protocol: An SSCOP-based Transport Protocol for Datagram Satellite Networks. Proceedings of 2nd International Workshop on Satellite-based Information Systems (WOSBIS´97).Budapest, Hungría. Octubre 1, 1997.

- Hu, Y. y Li, V. O. K. 2001. Satellite-Based Internet: A Tutorial. IEEE Communications Magazine. 39(3): 154-162 p.

- Jacobson, V. y Braden, R. 1988. TCP Extensions for Long-Delay Paths. RFC 1072. IETF. http://www.faqs.org/rfcs/rfc1072.html. Octubre 2004.

- Jacobson, V., Braden, R. y Borman, D. 1992. TCP Extensions for High Performance. RFC 1323. IETF. http://www.faqs.org/rfcs/rfc1323.html. Octubre 2004.

- Jammalipour, A. 1998. Low Earth Orbital Satellites for Personal Communication Network. Artech House. Boston. 273 p.

- Khalifa, I. y Trajkovic, L. 2004. An Overview and Comparison of Analytical TCP Models. IEEE International Symposium of Circuits and Systems.Vancouver, Canada. Vol. 5. Mayo 2004. 469-472 p.

- Kota, S., Pahlavan, K y Leppanen, P. 2004 Broadband Satellite Communications for Internet Access. Kluwer Academic Publishers. Norwell, Massachussetts. 421 p.

- Lang, T. 2002. Evaluation of Different TCP Versions in Non-Wireline Environments. Tesis de Doctorado en Filosofía, University of South Australia, Australia. 199 p.

- Laskhman, T. V. y Madhow, V. 1997. The Performance of TCP/IP for Networks with High Bandwith-Delay Products and Random Loss. IEEE/ACM Transactions on Networking. 5(3): 336-50 p.

- Maral, G. y Bousquet, M. 2003. Satellite Communication Systems. John Wiley & Sons, Cuarta Edición. Inglaterra. 757 p.

- Marchese, M. 2001. TCP Modifications over Satellite Channels: Study and Performance Evaluation. International Journal of Satellite Communications. 19(1): 93-110 p.

- Mascolo, S., Casetti, C., Gerla, M., Sanadidi, M. Y. y Wang, R. 2001. TCP Westwood: Bandwitdh Estimation for Enhanced Transport over Wireless Links. Proceedings of ACM Mobicom Conference. Roma, Italia. Julio 16-21, 2001. 287-297 p.

- Mathis, M., Mahdavi, J., Floyd, S. y Romanow, A. 1996. TCP Selective Acknowledgment Options. RFC 2018. IETF. http://www.faqs.org/rfcs/rfc2018.html. Octubre 2004.

- Mathis, M., Semke, J. y Mahdavi, J. 1997. The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm. ACM Computer Communications Review. 27(3): 67-82 p.

- Nam, N. H. 2003. Routing and Quality of Service in Broadband LEO Satellite Networks. Kluwer Academic Publishers.Norwell, Massachussetts. 187 p.

- Nguyen, T., Yengenoglu, F. y Sciuto, A. 2001. Voice over IP Service and Performance in Satellite Networks. IEEE Communications Magazine. 39(3):164-171 p.

- Padhye, J., Firoiu, V., Towsley, D. F. y Kurose, J. F. 2000. Modeling TCP Reno Performance: A Simple Model and its Empirical Validation. IEEE/ACM Transactions on Networking. 8(2):133-145 p.

- Peterson, L. L. y Davie, B. S. 1996. Computer Networks: A Systems Approach. Morgan Kauffman. San Francisco. 552 p.

- Postel, J. 1981. Transmission Control Protocol, STD 7. RFC 793. IETF. http://www.faqs.org/rfcs/rfc793.html. Octubre 2004.

- Stallings, W. 1998. High-Speed Networks – TCP/IP and ATM Design Principles. Prentice Hall. Primera Edición. New Jersey. 576 p.

- Zanella, A., Procissi, G., Gerla, M. y Sanadidi, M. Y. 2001. TCP Westwood: Analytic Model and Performance Evaluation. IEEE Globecom. Noviembre 2001. San Antonio, Texas. 1703-1707 p.