

**Networking over Next-Generation Satellite Systems**

by

*Thomas Ross Henderson*

B.S. (Stanford University) 1990

M.S. (Stanford University) 1991

A dissertation submitted in partial satisfaction of the  
requirements for the degree of  
Doctor of Philosophy

in

Engineering– Electrical Engineering  
and Computer Sciences

in the

GRADUATE DIVISION  
of the  
UNIVERSITY of CALIFORNIA at BERKELEY

Committee in charge:

Professor Randy H. Katz, Chair  
Professor Dorit S. Hochbaum  
Professor Steven R. McCanne

Fall 1999

The dissertation of *Thomas Ross Henderson* is approved:

---

Chair

Date

---

Date

---

Date

University of California at Berkeley

Fall 1999

# **Networking over Next-Generation Satellite Systems**

Copyright Fall 1999  
by  
*Thomas Ross Henderson*

## Abstract

### Networking over Next-Generation Satellite Systems

by

*Thomas Ross Henderson*

*Doctor of Philosophy in Engineering— Electrical Engineering  
and Computer Sciences*

*University of California at Berkeley*

*Professor Randy H. Katz, Chair*

Thanks to both the rapid deployment of the Internet and advances in satellite technology, the market for broadband satellite services is poised for substantial growth in the coming decade. Current communications satellite systems have generally been designed to provide either voice or data transaction (low data rate) services through small terminals, or trunking (high data rate, or broadband) services through large terminals. However, technological advances are enabling new systems that combine broadband data rates with small terminals, thereby providing more affordable “last-mile” network access to home and small business users worldwide. In particular, two types of broadband satellite systems are under development: high-power satellites deployed at traditional geostationary (GEO) orbits, and large constellations of satellites deployed at much lower (LEO) orbits.

In this thesis, we explore research problems that have arisen from this shift in satellite network architectures. When using GEO satellites to provide Internet access service, the performance of the Internet’s Transmission Control Protocol (TCP) is degraded by the high latency and high degree of bandwidth asymmetry present in such systems. We therefore undertook a comprehensive study of TCP performance in the context of broadband satellite systems used for network access. We first studied whether TCP’s congestion avoidance algorithm can be adjusted to provide better fairness when satellite connections are forced to share bottleneck links with other (shorter delay) connections. Our data suggests that adjustments of the policy used in that algorithm may yield substantial fairness benefits without compromising utilization. We next demonstrated how minor variations in TCP implementations can have drastic performance implications when used over satellite links (such as a reduction in file transfer throughput by over half), and from our observations constructed a satellite-optimized TCP implementation using standardized options. We explored the performance of TCP for short data transfers such as Web traffic, and found that two experimental options relating to how TCP starts a connection, when used together, could reduce the user-perceived latency by a factor of two to three. However, because not all of these options are likely to be deployed on a wide scale, and because even the best satellite-optimized TCP implementation is vulnerable to the fairness problems identified above, we explored the performance benefits of splitting a TCP connection at a protocol gateway within the satellite network, and found that such an approach can allow the performance of the satellite connection to approach that of a non-satellite connection. Carrying this work one step further, we construct a satellite-optimized transport protocol that can be used in such a split-connection environment, and demonstrate how it outperforms

TCP in a satellite environment characterized by large amounts of bandwidth asymmetry. In particular, our protocol, which we have dubbed the “Satellite Transport Protocol,” uses up to an order of magnitude less traffic on the bandwidth-constrained reverse channel than is needed by TCP.

In contrast to research on GEO systems, research on LEO systems is still in its infancy. While LEO systems are being designed specifically to avoid the high latencies found in GEO systems, their design is challenging from a packet routing perspective due to the highly time-varying nature of the LEO network topology. Moreover, even the most basic system properties of such constellations is not well documented in the literature. We constructed a detailed packet-level LEO network simulator and identified some fundamental delay performance results of commercially proposed constellations. We then explored whether or not geographic-based network addresses can be constructively used in the design of distributed or centralized packet routing systems. We found that the construction of a distributed packet routing algorithm based on geographic-based packet forwarding is fundamentally challenging due to subtle topological properties of LEO networks. However, we demonstrated that geographic-based addresses are useful in centralized routing systems, enabling significant reductions in routing traffic and on-board routing tables. Specifically, we constructed routing strategies based on geographic-based addresses that potentially reduce by an order of magnitude the amount of routing traffic exchanged between satellite nodes and a centralized routing center on the Earth’s surface, and the number of forwarding table entries required for non-local destinations.

Broadband satellite networks are likely to become an important niche of the future Internet because of their unique ability to provide network access from almost any point on the globe, particularly those underserved by terrestrial infrastructure. However, because the design of Internet protocols is driven by the performance of the wired Internet, satellite network engineers must be vigilant in assisting in the design and deployment of satellite-friendly protocols and in considering how satellite networks interwork with the wired Internet. Broadband LEO networks are likely to be deployed later than their GEO counterparts, and the design of these networks is still in its infancy, particularly since such networks are significantly more ambitious technically than anything that has been previously attempted. It is our hope that the findings presented in this thesis may contribute to a better understanding of how to design protocols for these next-generation systems while stimulating further work in this area.

---

Professor Randy H. Katz  
Dissertation Committee Chair

# Contents

<b>List of Figures</b>	<b>vi</b>
<b>List of Tables</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Satellites and the Contemporary Internet . . . . .	3
1.1.1 A Brief Overview of Satellite Communications . . . . .	3
1.1.2 The Internet Protocol Architecture . . . . .	4
1.1.3 Satellites and the Internet– A Match Worth Making? . . . . .	6
1.2 Contributions and Thesis Overview . . . . .	7
1.2.1 Statement of Research Problems . . . . .	7
1.2.2 Contributions . . . . .	8
1.2.3 Thesis Overview . . . . .	9
<b>2 Background and Related Work</b>	<b>11</b>
2.1 Reliable Transport Protocols in a Satellite Environment . . . . .	11
2.1.1 Transmission Control Protocol (TCP) Overview . . . . .	11
2.1.2 TCP Performance over Satellite Links . . . . .	15
2.1.3 Other Related Protocols . . . . .	20
2.2 Packet Routing for LEO Networks . . . . .	21
2.2.1 Network Characteristics of LEO Constellations . . . . .	21
2.2.2 Routing in LEO Satellite Networks . . . . .	25
2.3 Summary of Related Work . . . . .	28
<b>3 Methodology</b>	<b>30</b>
3.1 Research Strategy . . . . .	30
3.2 Simulation Environment . . . . .	31
3.2.1 HTTP Traffic Generator . . . . .	32
3.2.2 Satellite Transport Protocol . . . . .	32
3.2.3 LEO Satellite Network Extensions . . . . .	33
3.3 Experimental Testbed . . . . .	34
3.3.1 Experimental Machines and Software . . . . .	34
3.3.2 Ricochet Packet Radio Network . . . . .	35
3.3.3 DirecPC Satellite System . . . . .	36

3.4	Summary	37
<b>4</b>	<b>TCP Performance over Satellite Links</b>	<b>38</b>
4.1	TCP Fairness in a Heterogeneous Environment	39
4.1.1	Introduction	39
4.1.2	Methodology	42
4.1.3	Benchmark Results	44
4.1.4	Performance of the Constant-Rate Policy	45
4.1.5	Selectively Modifying the Additive Increase Policy	49
4.1.6	Implementation Issues	51
4.1.7	Summary	52
4.2	End-to-End TCP Performance over Satellite Links	52
4.2.1	Introduction	52
4.2.2	Methodology	53
4.2.3	Performance for Large File Transfers	55
4.2.4	Performance for Web Transfers	60
4.3	Split TCP Connections	65
4.3.1	Split Connection Approaches	65
4.3.2	Split Connection Performance	68
4.4	Summary	68
<b>5</b>	<b>Satellite Transport Protocol</b>	<b>70</b>
5.1	Design Goals	70
5.1.1	Basic Design	71
5.1.2	Our Protocol Modifications	76
5.2	Simulation Results	78
5.2.1	Simulation Configuration	78
5.2.2	Bulk Transfer Performance of STP	80
5.2.3	STP Performance in a High BER Environment	81
5.3	Experimental Results	84
5.4	Summary	87
<b>6</b>	<b>Packet routing for LEO networks</b>	<b>89</b>
6.1	Why is LEO Packet Routing an Interesting Problem?	89
6.2	Simulation Model and Key Parameters	90
6.3	Basic Performance Results	95
6.3.1	Delay Profiles	95
6.3.2	Routing Cost Metrics	101
6.4	Geographic Addressing and Cellular Geometry	102
6.4.1	Geographic Addressing and Mobility	102
6.4.2	Cellular Geometry	103
6.5	Design and Evaluation of a Distributed Routing Protocol	105
6.5.1	Overview	105
6.5.2	Construction of a Distributed Routing Protocol	106
6.5.3	Performance	114

6.5.4	Summary . . . . .	115
6.6	Centralized Routing Performance . . . . .	115
6.6.1	Cellular Structure and Addressing . . . . .	116
6.6.2	Reducing Routing Table Size Updates . . . . .	118
6.6.3	Address Aggregation . . . . .	121
6.7	Summary . . . . .	125
<b>7</b>	<b>Conclusions and Future Work</b>	<b>127</b>
7.1	Summary . . . . .	127
7.1.1	Transport Protocols for Broadband GEO Systems . . . . .	127
7.1.2	Unicast Packet Routing for LEO Constellations . . . . .	128
7.2	Software Availability . . . . .	129
7.3	Future Directions . . . . .	130
<b>A</b>	<b>Congestion Avoidance and Selective Retransmission Policies for TCP</b>	<b>132</b>
	<b>Bibliography</b>	<b>134</b>

# List of Figures

1.1	Projections of Annual Satellite Broadband Revenue (Source: Merrill Lynch and Co.).	2
1.2	Current commercial communications satellites in geostationary orbit (Source: Hughes Space and Communications Company; reproduced with permission).	4
1.3	The Internet protocol architecture.	6
2.1	Typical packet sequences for TCP and T/TCP.	12
2.2	Basic operation of TCP slow start and congestion avoidance (from [9]).	14
2.3	Example of a polar-orbiting satellite constellation. The figure (and subsequent SaVi-labelled figures) was generated using the SaVi software developed by the Geometry Center at the University of Minnesota.	21
2.4	Illustration of how orbital planes intersect near the poles. A minimal amount of orbital eccentricity guarantees that one orbit passes over the other and no collisions occur.	22
2.5	Satellite-fixed (nadir-pointing) vs. Earth-fixed footprints.	23
3.1	Three-phase research methodology— analysis, simulation, and implementation.	31
3.2	Extensions to the ns simulator (new elements listed in bold type).	33
3.3	Architecture of the Richochet packet radio network (Source: Metricom, Inc.; used with permission).	35
3.4	Architecture of the DirecPC satellite system (from [100]; used with permission).	36
4.1	Example of a broadband satellite network in which a satellite-based host communicates with a server in the Internet.	39
4.2	A demonstration of the unfairness of the current TCP congestion avoidance algorithm. The connections, from top to bottom, have RTTs of 10, 100, 200, 300, and 600 ms respectively.	40
4.3	Simulated sequence number evolution for connections with different RTTs under both the constant rate and increase-by-one policies.	41
4.4	Three simulation topologies.	43
4.5	Benchmark performance results using Topology 2: Utilization vs. queue size.	44
4.6	Benchmark performance results using Topology 2: Fairness vs. queue size.	45
4.7	Utilization vs. window constant of TCP SACK with fine-grained RTT estimates over topologies with bottleneck RED queues.	47
4.8	Fairness vs. window constant of TCP SACK with fine-grained RTT estimates over topologies with bottleneck RED queues.	48

4.9	The sensitivity of the proper selection of the constant $c$ when the effects of a standard TCP connection are added to the performance shown in Figure 4.8. . . . .	49
4.10	Improvement in fairness vs. window constant due to the IBK policy. . . . .	50
4.11	Configuration for network experiments. . . . .	53
4.12	Configuration for simulation experiments. . . . .	54
4.13	Throughput performance of TCP SACK NewReno, TCP SACK Reno, TCP NewReno, and TCP Reno over an experimental path with a TCP/IP bandwidth of 1.3 Mb/s and no transmission errors. Data points represent the sample means from 20 independent transfers of 10 MB each. In this and subsequent figures representing a large number of experimental results, error bars represent 95% confidence intervals. . .	55
4.14	Typical performance, using a standard BSD TCP (Reno) implementation, of a large file transfer over a GEO satellite channel. . . . .	57
4.15	Correct NewReno behavior, using a modified BSD TCP implementation, of a large file transfer over a GEO satellite channel. . . . .	58
4.16	Incorrect NewReno behavior, using a modified BSD TCP implementation, of a large file transfer over a GEO satellite channel. This behavior is due to a burst of packets transmitted at the end of recovery. . . . .	59
4.17	Correct SACK behavior, using a modified BSD TCP implementation (including NewReno loss recovery), of a large file transfer over a GEO satellite channel. . . .	60
4.18	Closeup of rapid SACK recovery of multiple losses. . . . .	61
4.19	Agreement between simulation and experimental results for TCP SACK and TCP NewReno. . . . .	62
4.20	The effect of a single competing short-delay connection on the satellite connection's throughput. The competing connection was a persistent file transfer using TCP SACK NewReno with a nominal 20 ms RTT between a background source and sink in the experimental topology. . . . .	63
4.21	TCP latency of a 3 segment server reply using standard TCP. . . . .	64
4.22	Future satellite networking topology in which a satellite-based host communicates with a server in the Internet through a satellite protocol gateway. . . . .	66
4.23	Forward throughput performance of split TCP in the presence of a short-delay competing connection. TCP SACK NewReno with large windows was used on both connection portions. . . . .	67
4.24	Reverse channel utilization of split TCP in the presence of a short-delay competing connection. TCP SACK NewReno with large windows was used on both connection portions. . . . .	68
5.1	Illustration of the main packet formats in STP. . . . .	72
5.2	Basic STP operation for bulk data transfer. . . . .	73
5.3	Basic STP operation for short transactions. . . . .	74
5.4	Overview of STP state transitions. . . . .	75
5.5	Simulation configuration for GEO topology. Simulations involved measuring the performance of file transfers between "foreground" hosts against background Web-like traffic generated by "background" hosts. . . . .	78

5.6	Simulation configuration for LEO topology. Simulations involved measuring the performance of file transfers between “foreground” hosts, across four LEO satellite hops, against background Web-like traffic injected at various points in the topology.	79
5.7	Simulation results of the forward throughput performance of STP on a 1 Mb/s channel with a variable BER.	83
5.8	Reverse channel bandwidth required for a large STP file transfer as a function of BER (simulation results corresponding to Figure 5.7).	83
5.9	Future satellite networking topology in which a satellite-based host communicates with a server in the Internet through a satellite protocol gateway.	84
5.10	Comparison of forward throughput performance of split TCP and split STP. For fair comparison, both TCP and STP used identical congestion control policies.	85
5.11	Comparison of reverse channel usage of split TCP and split STP, for the forward transfers illustrated in Figure 5.10.	86
6.1	Example of a polar-orbiting satellite constellation (figure reproduced from Section 2.2).	91
6.2	Snapshot of the Iridium constellation, illustrating active ISLs.	93
6.3	Snapshot of the Teledesic constellation, illustrating active ISLs.	94
6.4	Scatter plot of the one-way delay experienced by 10,000 different pings between random locations on the Earth’s surface, when global min-delay shortest path routing is used (Teledesic constellation configuration).	95
6.5	Histogram of values corresponding to Figure 6.4. Fewer than 1% of the delays exceeded 100 ms.	96
6.6	Scatter plot of the one-way delay experienced by 10,000 different pings between random locations on the Earth’s surface, when global min-hop shortest path routing is used (Teledesic constellation configuration).	97
6.7	Scatter plot of the delay experienced by 10,000 different pings, when global min-delay shortest path routing is used (Iridium constellation).	98
6.8	Delay variation between New York and San Francisco over the course of one day, for the Teledesic constellation.	99
6.9	A view of the previous plot at a smaller timescale. End-to-end delays are characterized by slow variations over tens of seconds punctuated by step increases and decreases in the delay of generally no more than 8 ms.	99
6.10	Delay variation between New York and San Francisco over the course of one day, for the Iridium constellation without cross-seam ISLs.	100
6.11	Average and maximum delay difference between min-hop and min-delay shortest path routing, as a function of the great-circle distance between terminals (Teledesic constellation).	100
6.12	Average and maximum delay difference between min-hop and min-delay shortest path routing, as a function of the great-circle distance between terminals (Iridium constellation).	101
6.13	An example of how min-hop routing can occasionally select a route with a much larger delay than that of min-delay routing. Both routes contain eight satellite hops, but one route uses hops in the higher latitudes which incur much less propagation delay.	102

6.14	Alternatives for dividing the Earth's surface into cells. . . . .	103
6.15	A cellular geometry consisting of roughly equal-sized trapezoidal cells [118]. . . .	104
6.16	A logical network topology: fixed zones on the Earth's surface are assigned a logical address, and a satellite serving a particular zone embodies the logical node serving that region (from [84]). . . . .	106
6.17	Hybrid routing strategy based on geographic packet forwarding for distant destinations and locally-scoped shortest path routing for local destinations. The figure denotes a subgraph of the satellite mesh and a hypothetical packet trace. A packet sourced at $S$ is forwarded based on geographic information to the satellite numbered 4. Satellites use shortest-path routing information to complete the routing to destination $D$ , which is served by satellite 6. . . . .	108
6.18	View of the Iridium topology above the North pole. Satellites closest to the pole have interplane ISLs turned off. The "polar region" is bounded by the set of satellites closest to the pole that have all of their interplane ISLs active. . . . .	110
6.19	Illustration of the intersection of counter-rotating planes. . . . .	111
6.20	An illustration of how deviations from pure polar orbits cause the latitude at which the counter-rotating planes intersect to degrade. This plot assumes a $15^\circ$ degree plane separation such as used in the Teledesic design. . . . .	112
6.21	Average and maximum delay difference between using geographic forwarding and minimum-hop shortest path routing as a function of terminal separation (Teledesic constellation). Error bars denote one sample standard deviation from the sample mean. . . . .	113
6.22	Average and maximum delay difference between using geographic forwarding and minimum-delay shortest path routing as a function of terminal separation (Teledesic constellation). Error bars denote one sample standard deviation from the sample mean. . . . .	114
6.23	Example of the cellular numbering strategy in one dimension. Two potential size-4 aggregations are illustrated. . . . .	117
6.24	Comparison of temporal and geographic consistency of forwarding tables across topology states (Teledesic constellation). . . . .	119
6.25	A hypothetical aggregation of 35 cells into 7 forwarding table entries. . . . .	121
6.26	Benefit of aggregating contiguous forwarding table entries. The number of aggregated entries is roughly a tenth (or fewer) of the total number of cells that must be represented (Teledesic constellation). . . . .	124

## List of Tables

1.1	Summary of major LEO system proposals (data from various sources, but mainly [143]). . . . .	5
2.1	Parameters for the Iridium and Teledesic systems. Both systems are examples of polar orbiting constellations. . . . .	25
4.1	Effect of the IBK policy on throughput (Topology 2, $K = 4$ ). 99% confidence intervals are shown in parentheses. . . . .	51
4.2	TCP latency effects on HTTP transfers for GEO and LEO satellite connections. Trace data is taken from [27]. All latencies are in seconds. For the experimental results, 95% confidence intervals are shown in parentheses. . . . .	65
5.1	Performance comparison of STP, TCP SACK, and TCP Reno over the simulated GEO topology. . . . .	81
5.2	Performance comparison of STP, TCP SACK, and TCP Reno over the simulated LEO topology. . . . .	82
5.3	Comparison of TCP, T/TCP, and STP performance for HTTP traffic. The results are averages of 1000 HTTP transfers, where the traffic generated was drawn from an empirical distribution based on traces described in [27]. . . . .	87
5.4	Results of file transfer experiments over the DirecPC DBS system and Ricochet packet radio network. The throughputs listed are the averages of 25 file transfers. The file sizes were 1 MB for DirecPC and 100 KB for Ricochet. . . . .	87
6.1	Key constellation parameters for the Iridium and Teledesic systems. Both systems are examples of polar orbiting constellations. . . . .	92

## Acknowledgements

Pursuing a doctoral degree is a long journey that one cannot make alone, and in this space I'd like to thank the many people who have contributed to my particular journey.

First, I thank my parents for all of the love, support, and encouragement they have provided me throughout my life. I am forever grateful for the sacrifices they made to give me the wonderful opportunities that I have had.

Randy Katz has been a fantastic research advisor on many levels. I have been continually impressed with his ability and dedication in leading a very large group of students towards their degrees, while also teaching and serving as department chair. Randy gave me a lot of freedom to pursue my research, while also providing wise advice and helpful discussions along the way. I feel very fortunate to have had Randy as an advisor and role model during my time at Berkeley.

I am grateful to the many members of the BARWAN research group for their help, friendship, and support. Giao Nguyen helped me a great deal in programming related to the *ns* simulator. Venkat Padmanabhan contributed device drivers used in many of my network experiments. Keith Sklower helped out tremendously by porting satellite card drivers from the Windows platform to Unix. Hari Balakrishnan wrote the initial TCP SACK implementation that I used in experiments. I also especially thank Hari, Venkat, Giao, Elan Amir, and Tina Wong for several technical discussions and reviews of draft versions of this research. For everyone else associated with BARWAN, I thank you for your contributions to my work and to a fun and challenging research project.

I am grateful to the other two professors on my committee, Steve McCanne and Dorit Hochbaum, for their interest and concern along the way and for their help in writing this dissertation. Both provided me with technical assistance, and the classes that they taught on Computer Networks and Network Flows were especially helpful to my work. I would also like to thank Jean Walrand for serving on my quals committee, and Joseph Kahn and Avidesh Zakhori for sponsoring my research in my first year at Berkeley.

I thank Sally Floyd for her help with my initial TCP research by providing unpublished research results as well as several discussions on different aspects of TCP. The TCP fairness work reported herein derives from a class project that I undertook with Emile Sahouria, who in particular wrote several components of the background HTTP traffic generators used herein. Bruce Mah and Steve Gribble provided HTTP traffic traces that drove that traffic generator.

The *ns* simulator has been invaluable for my research, and I would like to thank everyone in the VINT project who has made *ns* such a powerful tool for networking research.

I thank Son Dao at HRL Laboratories for supporting my work both through sponsoring BARWAN and employing me as an intern over two summers. I enjoyed working with Son and the other members of HRL, including Yongguang Zhang, Bo Ryu, Dennis Connors, and Dante DeLucia. Hughes DirecPC also graciously agreed to host a Berkeley computer at their network uplink center that was used for some of my satellite network experiments.

I would also like to thank the many people who have provided me with unpublished technical data or assistance, including Don Hoffman and Marie-Jose Montpetit at Teledesic, and Lloyd Wood.

Finally, I thank my wife Sue, whose love, support, and dedication throughout this whole journey has helped get me through it all. From putting up with long hours to helping me celebrate

little victories along the way, I couldn't have asked for better companionship.

*Tom Henderson*  
*December, 1999*