# Internetworking with satellite constellations

by

**Lloyd Wood**

**Thesis submitted to the University of Surrey**
**for the degree of Doctor of Philosophy**

Centre for Communication Systems Research,
School of Electronics, Computing and Mathematics,
University of Surrey, Guildford, United Kingdom.

*for Dad*

Harry Wood
7 May 1940 – 31 July 1996

# *Acknowledgements*

The acknowledgements section of the thesis is where the casual reader gains the rare opportunity to gain some insight into the lack of personality and lack of political skill of the writer. As such, it is avidly and assiduously read; the remainder of the thesis does not offer the committed reader such a thrill.

Having at least assured your attention thus far, I now thank:

Barry Evans, for patience and for the time needed to amass this work.

Ilias Andrikopoulos, for travelling down this road just ahead of me and rarely, if ever, hesitating to offer me the benefit of all of his extensive opinions.

George Pavlou, for forcing gears into motion, making me realise what I was missing by staging frequent gladiatorial encounters with that near-mythical and largely invisible beast called ‘supervision’, and constantly setting unrealistic deadlines with warnings of impending doom to make me achieve the seemingly impossible. George was responsible for providing constant advice, regular motivation, my ongoing paranoia, and frequent adrenaline highs.

The *ns* developers, for providing incredibly useful simulation tools, increasingly comprehensive documentation, regular advice, and the welcome opportunity for me to look a gift horse in the mouth and grumble about all of the above on an ongoing basis.

The *ns* users, for increasing my understanding of the tools we choose to use.

Tom Henderson, for bringing me to VINT, showing me the way to better simulations with his own work extending *ns,* and providing thoughtful dialogue and feedback. I learned a lot about *ns* from building on Tom’s work and from extending and modifying his code. I *know* I still can’t program, but the list of languages I know I can’t program *in* is getting to be *impressive*.

Patrick Worfolk and the sadly-missed Geometry Center, for the opportunity to contribute to *SaVi* and advice on using it.

Jerome Galtier and the INRIA team, for ECOTEL ’98 and for introducing me to

Antoine Clerget and Walid Dabbous. Antoine and Walid, for helping me realise that a good journal paper begins with a small set of ideas, which are then followed by the large amounts of work needed to express those ideas clearly. With rewrites.

Gorry Fairhurst, for the joys of drafting and debate.

Tor Wisløff, for a diverse range of discussions over dinner.

Tony Sammut and Chris Meenan, sounding boards offering sound advice. And SPOC.

Adam Kirby and John Hibbitt, for keeping our computer systems going while providing copious help leavened with deadpan humour. The dubnerds team and Mike Stonebank, for their assistance with the ever-growing burden of web administration as the web becomes more and more involved with everything we do. Stephanie Evans, for gracefully handling all of the thankless administrative tasks that keep things running.

The **`breaking sate//ites`** subscribers, a continual source of news and entertaining technical discussion. Rahim Tafazolli, for suggesting the list in the first place back when Globalstar's Zenit hit its nadir, so ultimately naming it, too.

Many others in the Centre and elsewhere, for conversations and insights.

I am grateful to the DTI and SSPI, for merit awards that assisted with finance. The Theseus and Bisante projects provided funding and support for travel.

I would also like to express my gratitude to the masters students that I supervised. They helped show me the joys, the frustrations, and the rewards of taking on a teaching and supervisory role. In particular, Francisko Ninos' enthusiastic efforts and persistent questioning contributed to the analysis of multicast presented here.

Valerie, for lunch. Speros and Peter, for leaving. Roger and Ashleigh, for being there. Bruce, for bad movies. Bob, for a reason to finish. Li Hao, for bay shores and blue slopes. emma, for paris. Dan and Lesley, for a home from home.

And Mum, for home – and everything.

**Lloyd Wood**
L.Wood@surrey.ac.uk
http://www.ee.surrey.ac.uk/Personal/L.Wood/

late 2000/early 2001.

# *Abstract*

The development and growth of the Internet during the past thirty years has led to demand for and development of Internet services everywhere and over every possible communications medium. This includes the medium of satellite communications. During those same three decades, the growth in use of satellite communications to provide a widely-available wireless communications infrastructure has led to the development of broadband satellite communications using satellite constellation networks. These two technological trends have intersected.

Here, we examine networking and internetworking issues affecting satellite networking in complex satellite constellation networks, and determine what is needed in order to support services based on the TCP/IP suite well in satellite constellations.

We analyse constellation network topology. Its movement and effects on end-to-end delays experienced by network traffic travelling across the constellation are examined in detail. Analysis of the impact of cross-seam links upon delays experienced by traffic across star constellations shows that the use of cross-seam links is worthwhile.

We examine the effects of multi-path routing within the constellation upon TCP communication, and demonstrate the performance advantages of an intelligent flow-based approach to routing in the constellation network.

The desirability of implementing IP routing functionality in the space segment of the constellation is shown. The use of IP routing, to enable good support for IP QoS and IP multicast, is shown to be possible.

We present an approach to implementing IP multicast within the constellation, evaluating use of a core-based tree algorithm, and outline an architecture permitting IP routing of IP traffic in an ATM-based satellite constellation network, using MPLS.

Finally, we present and demonstrate the advantages of a novel method of managing path delay between ground terminals across a rosette constellation with intersatellite links, by using controlled handover to manage surface diversity to provide classes of service to network traffic.

# Contents

# *Figures*

## Key to *SaVi* plots

degree of coverage/diversity: 0 1 2 3 4+

single satellite footprint (area of single coverage)
showing subsatellite point (nadir) ●

## Key to intersatellite links

– **terminal-satellite uplinks/downlinks** are handed off at MEO and LEO.
– **intra-plane ISLs** are permanent and are never handed off.
– **interplane ISLs** may break as satellites near each other at speed, and may be handed off near highest latitudes as planes cross.
– **cross-seam ISLs** are temporary, regularly handed off as satellites travelling in opposing directions pass each other.

## Index to figures

# *Acronyms & abbrev.*

Definitions of **SI** units are not provided in this glossary.

| | |
|---|---|
| δ | angle of inclination to the Equator, delta. |
| Ω | right angle of ascension through the Equator, omega. |
| ° | angle in degrees. |
| **&** | and, ampersand. |
| **6Bone** | **IPv6** backbone, a virtual overlay network. |
| **AAL** | **ATM** Adaptation Layer. |
| **abbrev.** | abbreviation. |
| **ABR** | **ATM** Available Bit Rate. |
| **ack**, **ACK** | acknowledgement (‘acknowledgment’ in **US**, e.g. **Allman98**). |
| **ACM** | **IEEE** Association for Computing Machinery. |
| **ACTS** | **NASA** Advanced Communications Technologies Satellite. *Or*<br>**EU** Advanced Communications Technologies and Services framework. |
| **AIAA** | American Institute of Aeronautics and Astronautics. |
| **ALG** | **NAT** Application-Level Gateway. |
| **ARP** | Address Resolution Protocol. |
| **ARQ** | Automatic Repeat reQuest. |
| **AS** | Autonomous System. |
| **ATM** | Asynchronous Transfer Mode. |
| **b** | bit. |
| **B** | byte, generally 8**b**. |
| **BA** | **diffserv** Behaviour Aggregate. Now called **PDB**. |
| **BCP** | Best Current Practice. A class of **IETF RFC**. |
| **BGMP** | Border Gateway Multicast Protocol; not to be confused with **MBGP**. |
| **BGP** | Border Gateway Protocol. |
| **Bisante** | Broadband integrated satellite network traffic evaluations.<br>An **EU** Esprit project. |
| **bps** | bits per second. |

| | |
|---|---|
| **C-ARS** | Constellation Address Resolution Server. |
| **CBT** | Core-Based Tree, an approach to multicast spanning trees. |
| **CCSDS** | Consultative Committee for Space Data Systems. |
| **CCSR** | Centre for Communication Systems Research. At **UniS**. |
| **CDMA** | Code Division Multiple Access. |
| **CoS** | Class of Service. |
| **COST** | **EU** Cooperation in the field of Scientific and Technical Research. |
| **CR-LSP** | **MPLS** Constraint-based Routed Label-Switched Path. |
| ***cwnd*** | **TCP** implementation congestion window variable. |
| **DARPA** | **US** Defense Advanced Research Projects Agency. |
| **delack** | delayed **ack**. |
| **diffserv** | **IETF** Differentiated Services **QoS** framework. |
| **DNS** | Domain Name System. |
| **DSCP** | **diffserv** Differentiated Services Code Point. |
| **DSP** | Digital Signal Processing. |
| **DTI** | **UK** Department of Trade and Industry |
| **dupack** | duplicate **ack.** |
| **DVMRP** | Distance Vector Multicast Routing Protocol. |
| **ECOTEL** | École d’Hiver des Télécommunications de Sophia Antipolis. |
| **ECN** | Explicit Congestion Notification. |
| **ed.** | editor. |
| **e.g.** | *exempli gratia*, for example. |
| **EGP** | Exterior Gateway Protocol. |
| **ENST** | École Nationale Supérieure des Télécommunications. |
| **ER** | **ABR** Explicit Rate. |
| **et al.** | *et alia* (or variants thereof), and others. |
| **EU** | European Union. |
| **fax** | facsimile. |
| **FCC** | **US** Federal Communications Commission. |
| **FDMA** | Frequency Division Multiple Access. |
| **FTP** | File Transfer Protocol. |
| **GEO** | Geostationary Earth Orbit. |

| | |
|---|---|
| **GLONASS** | Global Navigation Satellite System<br>(Global'naya Navigatsionnaya Sputnikovaya Sistema). |
| **GPS** | **US** Global Positioning System. |
| **GSM** | Global System for Mobile communication (Groupe Spécial Mobile). |
| **handover** | handover (handoff in **US**, e.g. **Henderson99)**. |
| **HEO** | Highly Elliptical Orbit. |
| **http, HTTP** | Hypertext Transfer Protocol. |
| **ICC** | **IEEE** International Conference on Communications. |
| **ICT** | **IEE** Conference on Telecommunications. |
| **ICO** | Intermediate Circular Orbit. Often synonymous with **MEO**. |
| **ICSSC** | **AIAA** International Communications Satellite Systems Conference. |
| **ID** | identifier. *also* Internet-Draft (I-D). |
| **i.e.** | *id est*, that is. |
| **IEE** | Institution of Electrical Engineers. A **UK** organisation. |
| **IEEE** | Institute of Electrical and Electronics Engineers. A **US** organisation. |
| **IESG** | Internet Engineering Steering Group. |
| **IETF** | Internet Engineering Task Force. |
| **IFIP** | International Federation for Information Processing. |
| **IGMP** | Internet Group Management Protocol. |
| **IGP** | Internal Gateway Protocol. |
| **IMSC** | International Mobile Satellite Conference. |
| **Inc.** | Incorporated. |
| **Inmarsat** | International Maritime Satellite Organisation. |
| **intserv** | **IETF** Integrated Services **QoS** framework. |
| **ISDN** | Integrated Services Digital Network. |
| **ISL** | Inter-Satellite Link. |
| **IP** | *see* **TCP/IP**. |
| **IPv4** | **IP** version 4. Widely deployed. |
| **IPv6** | **IP** version 6. Increasing deployment, mostly on the **6Bone**. |
| **IPSec** | **IETF** IP Security framework. |
| **ISO** | International Standards Organisation. |
| **ISOC** | Internet Society. |

| | |
|---|---|
| **ITU** | International Telecommunication Union. |
| **IWQoS** | International Workshop on **QoS**. |
| **K** | 1024 (as opposed to the **SI** k, 1000). |
| **LDP** | Label Distribution Protocol. |
| **LEO** | Low Earth Orbit. |
| **LFN** | Long Fat Network. |
| **LLC** | Logical Link Control sublayer. *Or* Limited Liability Company. |
| **LOOPUS** | Loops in Orbit Occupied Permanently by Unstationary (*sic*) Satellites. |
| **LSP** | **MPLS** Label-Switched Path. |
| **LSR** | **MPLS** Label-Switching Router. |
| **MAC** | Medium Access Control layer. |
| **MARS** | Multicast Address Resolution Server. |
| **MASC** | Multicast Address Set-Claim protocol. |
| **MBGP** | Multi-protocol extensions to **BGP**. Not to be confused with **BGMP**. |
| **MBone** | Multicast Backbone, a virtual overlay network. |
| **MCS** | **ATM** Multicast Server. |
| **MEO** | Medium Earth Orbit. *Not* Middle Earth. |
| **MILCOM** | **IEEE** Military Communications conference. |
| **MOSPF** | Multicast extensions to **OSPF**. |
| **MPLS** | Multi-Protocol Label Switching. |
| **MSc** | degree of Master of Science. |
| **MSDP** | Multicast Source Discovery Protocol. |
| **MTU** | Message Transfer Unit. |
| **NASA** | **US** National Aeronautics and Space Administration organisation. |
| **NAT** | Network Address Translation. |
| **NGSO** | Non-Geostationary Orbit. |
| **NIC** | Network Information Center. |
| **no.** | number |
| **NOSSDAV** | Network and Operating System Support for Digital Audio and Video. |
| ***ns*** | network simulator. |
| **OBP** | On-Board Processing |
| **OCBT** | Ordered Core-Based Tree, an extension to **CBT**. |

| | |
|---|---|
| **OSI** | **ISO** Open Systems Interconnection; laid out 7-layer networking stack. |
| **OSPF** | Open Shortest Path First routing protocol. |
| **PDB** | **diffserv** Per-Domain Behaviour. Previously **BA**. |
| **pdf** | probability density function. |
| **PHB** | **diffserv** Per-Hop Behaviour. |
| **PhD** | degree of Doctor of Philosophy. |
| **PILC** | **IETF** Performance Implications of Link Characteristics **WG**. |
| **PIM-SM** | Protocol Independent Multicast – Sparse Mode. |
| **PNNI** | **ATM** Private Network-Network Interface architecture. |
| **pp.** | pages |
| **QoS** | Quality of Service. |
| **RFC** | **IETF** Request for Comments document approved by the **IESG**. |
| **RIP** | Routing Information Protocol. |
| **RSVP** | Resource reSerVation Protocol. |
| **RTP** | Real-Time Protocol. |
| **RTT** | Round Trip Time. |
| **SACK** | **TCP** Selective Acknowledgements. |
| **SATNET** | Atlantic Packet Satellite Network. |
| ***SaVi*** | Satellite Visualisation software from the Geometry Center. |
| **SCPS** | **CCSDS** Space Communications Protocol Standards protocol suite. |
| **SCPS-TP** | **SCPS** Transport Protocol. |
| **SEAM** | Scalable and Efficient **ATM** Multicast. |
| **SI** | International System of Units (Le Système International d'Unités). |
| **SIGCOMM** | **ACM** Special Interest Group on Data Communication. |
| **SLA** | Service Level Agreement. |
| **SLS** | Service Level Specification. Part of an **SLA**. |
| **SMDS** | Switched Multi-megabit Data Service. |
| **SONET** | Synchronous Optical Network. |
| **SPOC** | Satellite Positioning and Orbital Control software used within **CCSR**. |
| **SSPI** | Society of Satellite Professionals International. |
| ***ssthresh*** | **TCP** implementation slow start variable. |
| **STD** | Internet Standard. Defined by one or more **IETF RFC** documents. |

| | |
|---|---|
| **TCP** | *see* **TCP/IP**. |
| **TCP/IP** | Transmission Control Protocol/Internet Protocol protocol suite. |
| **TDMA** | Time Division Multiple Access. |
| **TE** | Traffic Engineering. |
| **Theseus** | Terminal at high speed for European stock exchange users.<br>An **EU ACTS** project. |
| **TOS** | Type Of Service. |
| **TLA** | **abbrev.** for Three-Letter **Abbrev**. Not an acronym. |
| **TT&C** | Telemetry, Tracking and Control. |
| **TTL** | Time To Live. |
| **UDP** | User Datagram Protocol, a fundamental part of **TCP/IP**. |
| **UK** | United Kingdom. |
| **UNI** | **ATM** User-Network Interface. |
| **UniS** | University of Surrey corporate brand. |
| **US, USA** | United States of America. |
| **VC** | Virtual Circuit. |
| **VCC** | **ATM** Virtual Channel Connections. |
| **VENUS** | Very Extensive Non-Unicast Service. |
| **VINT** | **DARPA** Virtual Internetwork Testbed project developing ***ns***. |
| **vol.** | volume. |
| **VPC** | **ATM** Virtual Path Connections. |
| **VPI** | **ATM** Virtual Path Identifier. |
| ***vs.*** | *versus*, against. |
| **VSAT** | Very Small Aperture Terminal. |
| **WG** | **IETF** Working Group. |
| **WOSBIS** | Workshop on Satellite-Based Information Services. |
| **WRC** | **ITU** World Radio Congress. |

# *1. Introduction*

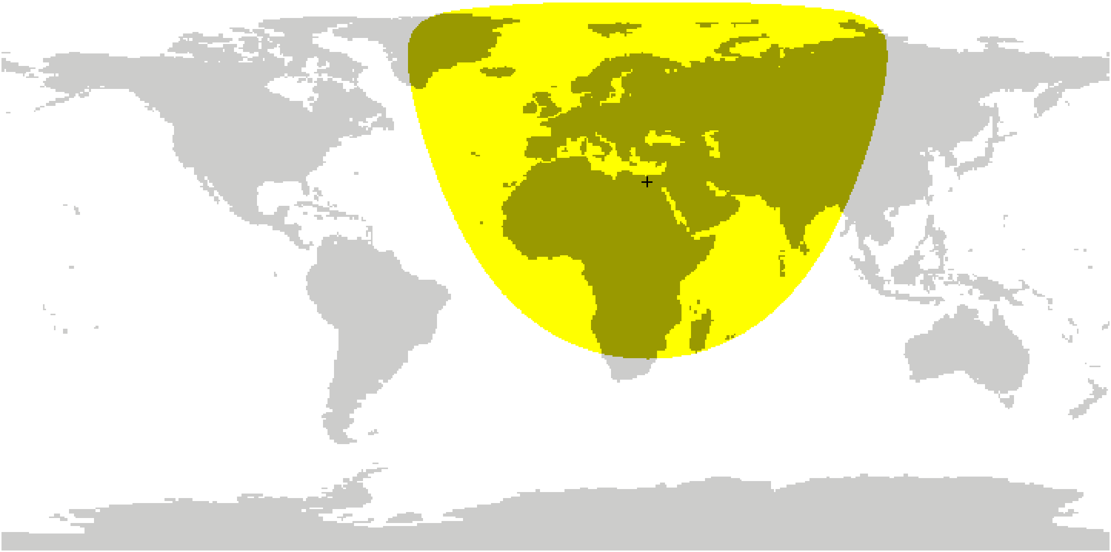
A satellite constellation can be defined as a number of similar satellites, of a similar type and function, designed to be in similar, complementary, orbits for a shared purpose, under shared control. Satellite constellations have been proposed and implemented for use in communications, including networking. This chapter introduces the satellite constellation and its networking aspects, before presenting the networking scenarios that will be examined in detail and the scope of this thesis.

## 1.1 Background

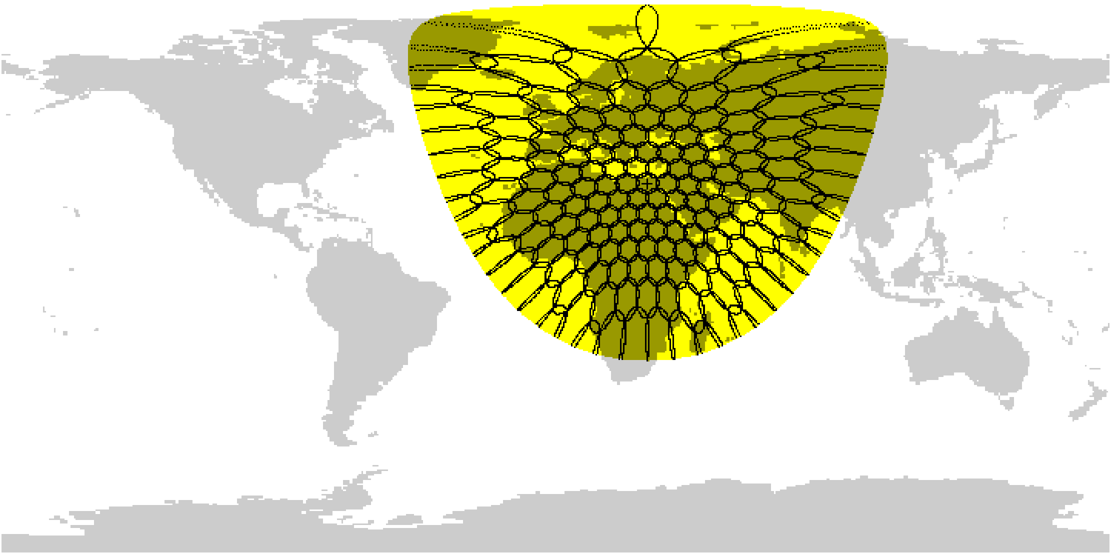
Satellite-based networking has developed in complexity over the years, rising up the network protocol stack and building upon established work at each of the various networking layers as described by the ISO OSI Reference Model [**ISO7498**].

Networking using satellites began by using individual satellites in geostationary orbit, where uplinked signals were amplified, frequency-shifted and broadcast down to a large ground area with the use of simple transparent ‘bent-pipe’ repeaters onboard the satellites. Sharing of this broadcast physical and data-link layer capacity led to the introduction of increasingly complex media-access control (MAC) schemes to use capacity effectively, most notably with slotted aloha and its variants for use with very small aperture terminal (VSAT) networks [**Maral95**]. The development of multiple spotbeams per satellite, illustrated in figure 1.1, established a requirement for on-board switching and MAC, with control of capacity allocated via circuits and a Logical Link Control (LLC) sublayer.

The idea of satellite constellations being used to provide wireless communication services to much of the Earth can be traced back to Arthur C. Clarke’s paper in *Wireless World* in 1945 [**Clarke45**]. This proposed a constellation of three geostationary satellites to provide full Equatorial coverage of the Earth, using the geostationary earth orbit (GEO). This ‘stationary orbit’ has been known about since the 16th century.



a. shared footprint area using a single antenna, with low overall capacity (unprojected map)



b. use of eight tiers of unshaped spotbeams increases footprint capacity and frequency reuse.
+ is subsatellite point (nadir)

*ICO* MEO satellite approximation (10,355km altitude, 10° minimum elevation angle)

**Figure 1.1 - distinguishing between satellite footprint and spotbeams**

(Today, some Inmarsat GEO systems approximate Clarke's idea, although not for his original suggested purpose of educational television for distance learning.)

Low-earth orbiting (LEO) and medium-earth-orbiting (MEO) satellite constellations, using orbits lower than the geostationary orbit, have been proposed, as have highly-elliptical-orbit (HEO) constellations. These give full global or targeted coverage of the Earth. Constellations make possible more reuse of limited available ground-space communication frequencies, providing higher overall network capacity as a result of this frequency reuse.

The decrease in propagation delay to LEO, MEO and some HEO when compared to GEO is a bonus for delay budgets, but can be unimportant for many applications. However, these non-GEO constellations require more satellites to provide constant continuous coverage of areas of the Earth. Their movement relative to the surface of the Earth requires management of handover and increases system complexity.

The number of satellites needed by some of these planned systems was unthinkable only a decade earlier, and was considered economically infeasible a few years later on, after a brief period of commercial optimism. Expectations have been adjusted and systems have been scaled back accordingly. Many systems proposed during that period of optimism now appear unlikely to be constructed.

As an example of this, in 1995 Teledesic LLC’s original commercial proposal of an 840-active-satellite LEO constellation became the largest seen. In 1997 that suggested proposal was scaled back to a 288-active-satellite LEO constellation designed with Boeing, illustrated in Figure 1.2.

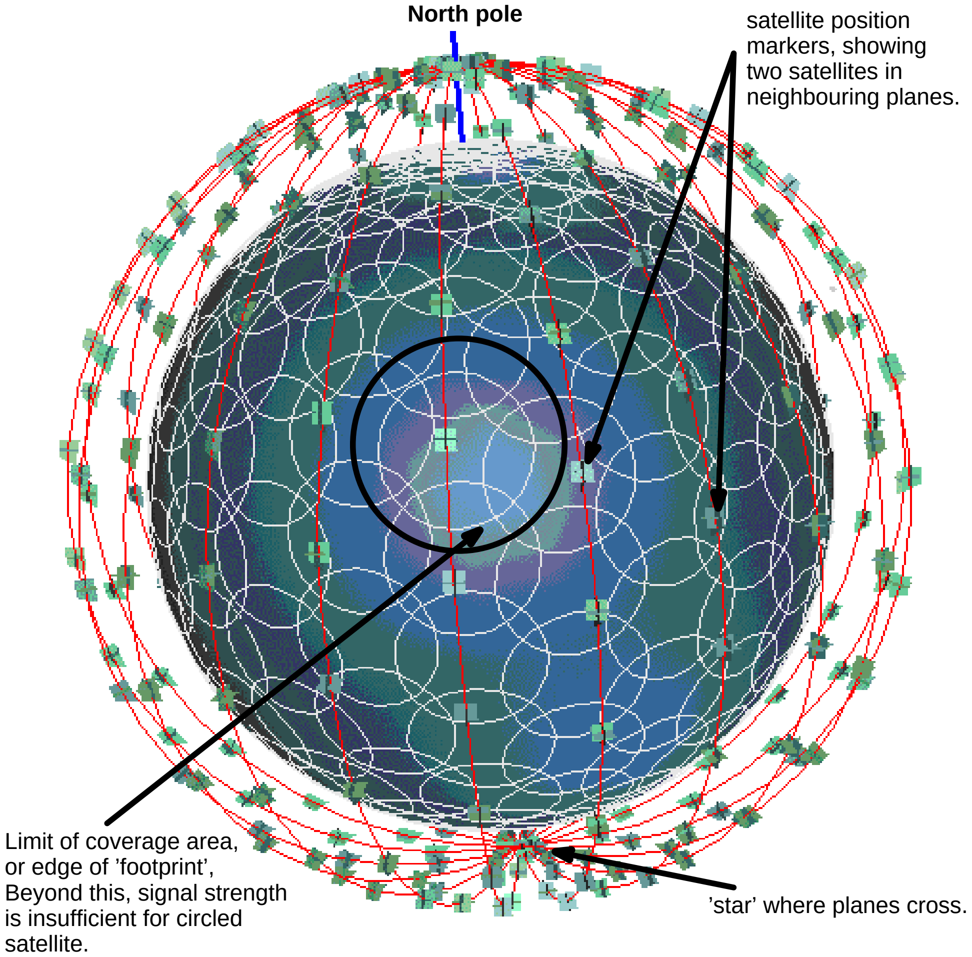


**Figure 1.2 - *Teledesic* (Boeing 288 active satellites) in *SaVi* [WorfolkThurman97]**

In 2000 Teledesic LLC gained management of ICO Global Communications and its unlaunched ten-active-satellite MEO constellation. This shift neatly reflects lowered expectations of the size of systems that can succeed technically and commercially.

## 1.2 Services provided by satellite constellations

Satellite constellations can be used for a variety of services, including:

### 1.2.1 Broadband data

Broadband satellite constellation networks have been proposed at GEO, LEO, MEO and HEO. Many are under development. These systems will provide medium- and high-capacity wireless data services, and will interconnect with existing terrestrial networks. Evans has provided comprehensive summaries and brief histories of a number of commercial proposals, overviewing this field [**Evans00a**, **Evans00b**].

The TCP/IP protocol suite used in the Internet has become established as the most popular method for computer network communication in the world. Proposed broadband satellite constellation networks, such as *Teledesic* [**Tucketal94**, **Sturza95**, **Stuart96**, **Kohn97**], *Spaceway* [**Fitzpatrick95a**, **Fitzpatrick95b**], and *SkyBridge* [**FCCSkyBridge97**, **Fraise00**], can therefore be expected to transport significant volumes of TCP/IP traffic.

Although use of satellite constellations to deliver broadband IP services is new, satellite constellation-based systems are already in use for many services, including:

### 1.2.2 Telephony applications

Constellations are providing voice telephony and low-bit-rate communication services. The *Iridium* [**Leopold91**, **LeopoldMill93**, **LeopoldMillGrubb93**] and *Globalstar* [**WiedeViterbi93, Dietrich98**] systems have made service available. These systems must interoperate with GSM and other terrestrial cellular networks, and generally offer basic fax, low-speed data and paging facilities (2400/9600 bps) alongside voice services. Of these voice-oriented constellations, only *Iridium* employs intersatellite links (ISLs), making it of interest as having demonstrated the feasibility of a standalone space subnetwork [**HutchLaurin95**]. *Iridium* is also of interest for demonstrating that the voice telephony market for satellite handsets is far smaller than originally anticipated, primarily due to the successful widespread adoption of terrestrial cellular networks during *Iridium*'s long development period, and the now near-ubiquitous GSM roaming. Later systems, including *Globalstar* and *ICO Global* [**MakSmith98**, **Ghediaetal99**], have been adversely affected by Iridium LLC's bankruptcy and are now placing more emphasis on additional markets to voice telephony, especially data services.

### 1.2.3 Navigation

Constellations are providing information to determine position and navigation and for geodesy. In particular, the MEO *GPS* [**Kruesi96**] and *GLONASS* [**BorJohDar99**] are dedicated to this purpose. (These navigation constellations do not use intersatellite links.) Other satellite systems fly additional onboard payloads to provide additional information about the reliability of the information provided by these dedicated systems, e.g. Inmarsat-3 navigation transponders.

### 1.2.4 Messaging applications

Constellations are providing low-bandwidth send, receive and broadcast services for a variety of applications, including paging, tracking, and remote data gathering. The first purpose-built commercial constellation in service in the messaging field, although not financially successful, is *Orbcomm* [**Hardman91**, **Mazur99**]. Proposed commercial systems include *Leo One* [**Goldman99**].

## 1.3 Orbit and altitude choices

The altitude of the satellites in the constellation is a significant factor in determining the number of satellites that is required to cover the Earth and the characteristics of the constellation.

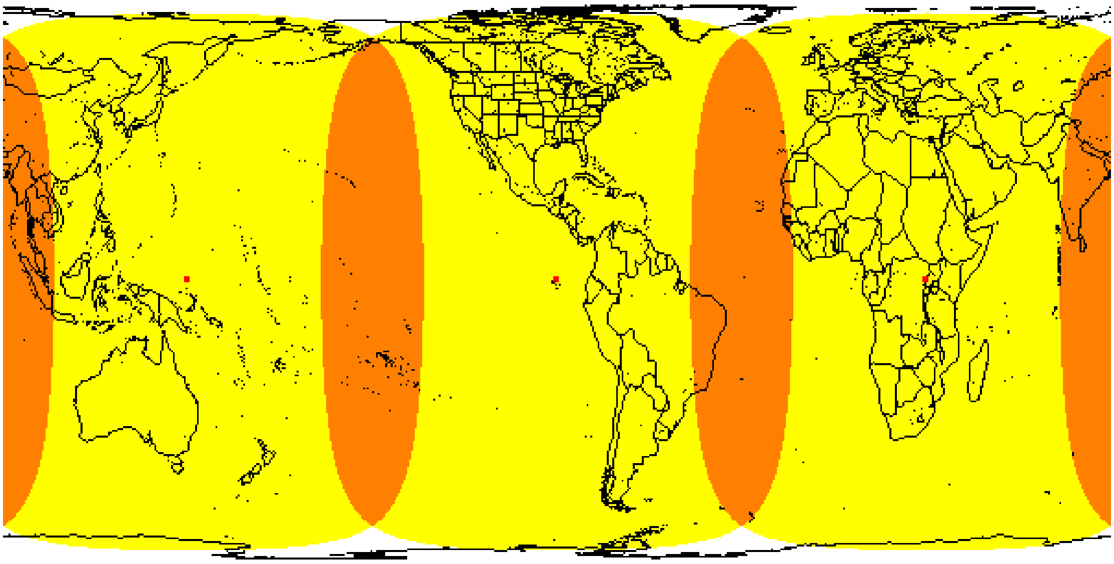
A lower altitude decreases free space loss and propagation delay, but means that the service that each satellite can offer is limited to users in a smaller area of the ground (the satellite's 'footprint'). To fully cover the globe, more satellites are needed. This increases frequency reuse and overall system capacity, but will also increase overall system construction and maintenance costs. Satellites at lower altitudes must move faster relative to the ground to stay in their orbits. As the area covered by each satellite moves with the satellite in its orbit, a lower orbit increases the frequency of handover and causes larger Doppler effects on signals between ground terminals and satellites.

Successive satellites arranged in circular orbits, sharing the same orbital plane, can provide continuous coverage of a strip of ground beneath them: the ‘street of coverage’. Many proposed systems use streets of coverage. Most use circular orbits, since a constant altitude means that satellite overhead pass times remain constant, and that each satellite can be used for network traffic throughout its orbit.

Other systems have been proposed using multiple elliptical orbits, where satellites are only intended to be available for service while moving relatively slowly around high-altitude apogee, while the communication payloads are unused at other points around the orbits. This chapter briefly introduces, outlines and illustrate these orbital choices.

### 1.3.1 Geostationary Earth Orbit (GEO)

At an altitude of 35,786km above the Equator, the angular velocity of a satellite in this orbit matches the angular rate of rotation of the Earth’s surface. This makes the satellite appear stationary to an observer on the Earth. This useful feature has resulted in the orbit becoming extremely popular, and satellite spacing in the orbit is at the limits of terrestrial antenna discrimination (the angle between orbital slots has gradually narrowed from 3° to 2° and occasionally 1.5°).



cylindrical projection, showing maximum range of visibility (minimum elevation angle<1°)

**Figure 1.3 - near-global coverage from *Clarke* geostationary constellation (*SaVi*)**

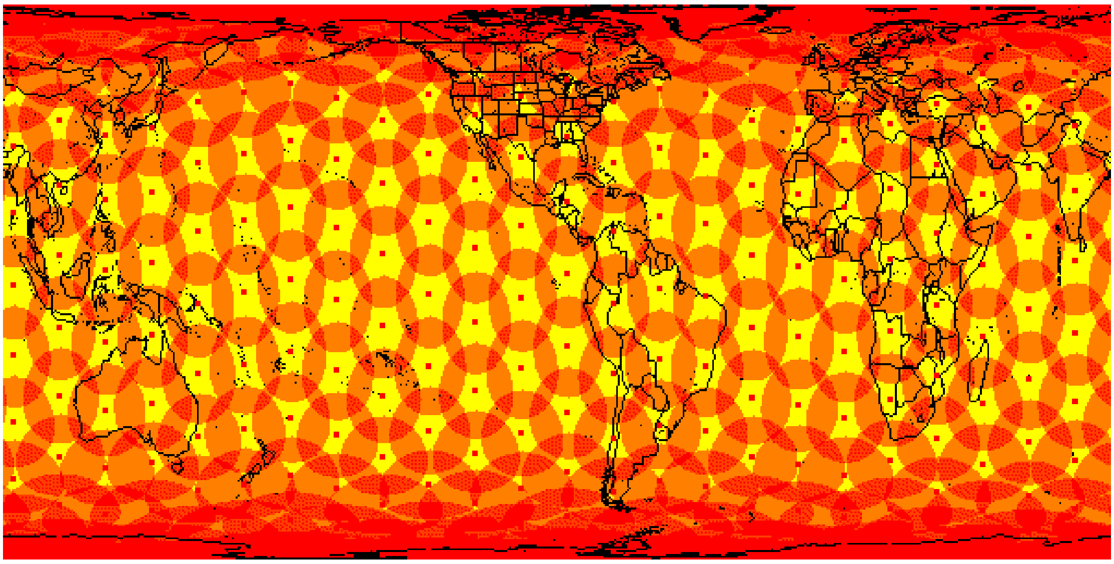
Coverage of high latitudes is impossible above 81° latitude and rarely possible above 75°, so full Earth coverage cannot be achieved by using any purely geostationary constellation. However, much of the Earth can be covered with a minimum of three geostationary satellites [Figure 1.3, **Clarke45**].

Propagation delay between an earth station and a geostationary satellite varies with the difference in position in longitude and terminal latitude, but is around 125ms (milliseconds), or around 250ms between ground stations. This leads to the widely-quoted half-second round-trip latency for communications via geostationary satellite.

### 1.3.2 Low Earth Orbits (LEO)

At altitudes of typically between 500 and 2000km, lying beyond the upper atmosphere but below the peaks of the inner Van Allen radiation belt, a large number of satellites is required to provide simultaneous global coverage in low earth orbit. The actual number of satellites used depends upon the coverage required and upon the minimum elevation angle desired for communication. These determine the degree of atmosphere-induced slant loss permitted, and dimension the resulting link budget.

With a large number of satellites and their resulting small footprint areas (shown for the Boeing *Teledesic* design in figure 1.4) and small spotbeam coverage areas, large amounts of frequency reuse become possible across the Earth, providing large system capacity.



cylindrical projection, using minimum elevation angle of 40°. Compare with Figure 1.2.

**Figure 1.4 - global coverage from Boeing *Teledesic* LEO constellation (*SaVi*)**

LEO satellites move rapidly relative to the surface of the Earth and to the ground terminals that they communicate with. Speeds at over 25,000 km/hour, with visibility of only a few minutes before handover to another satellite occurs, are the norm. Propagation delay between ground and LEO is often under 15ms, and varies rapidly as the satellite approaches and leaves local zenith while passing the ground terminal.

### 1.3.3 Medium Earth Orbits (MEO)

At altitudes of between 9,000 and 11,000km, between the inner and outer Van Allen belts, these orbits can permit full Earth coverage with fewer, larger satellites. The satellites have larger coverage footprints from the increased altitude, but also increased resulting delay. Movement is slower, with visibility times of tens of minutes before handover must take place. Propagation delay for the uplink or downlink between earth station and satellite is typically under 40ms. Hughes' *Spaceway NGSO* MEO proposal is shown in figures 1.5 and 1.6.

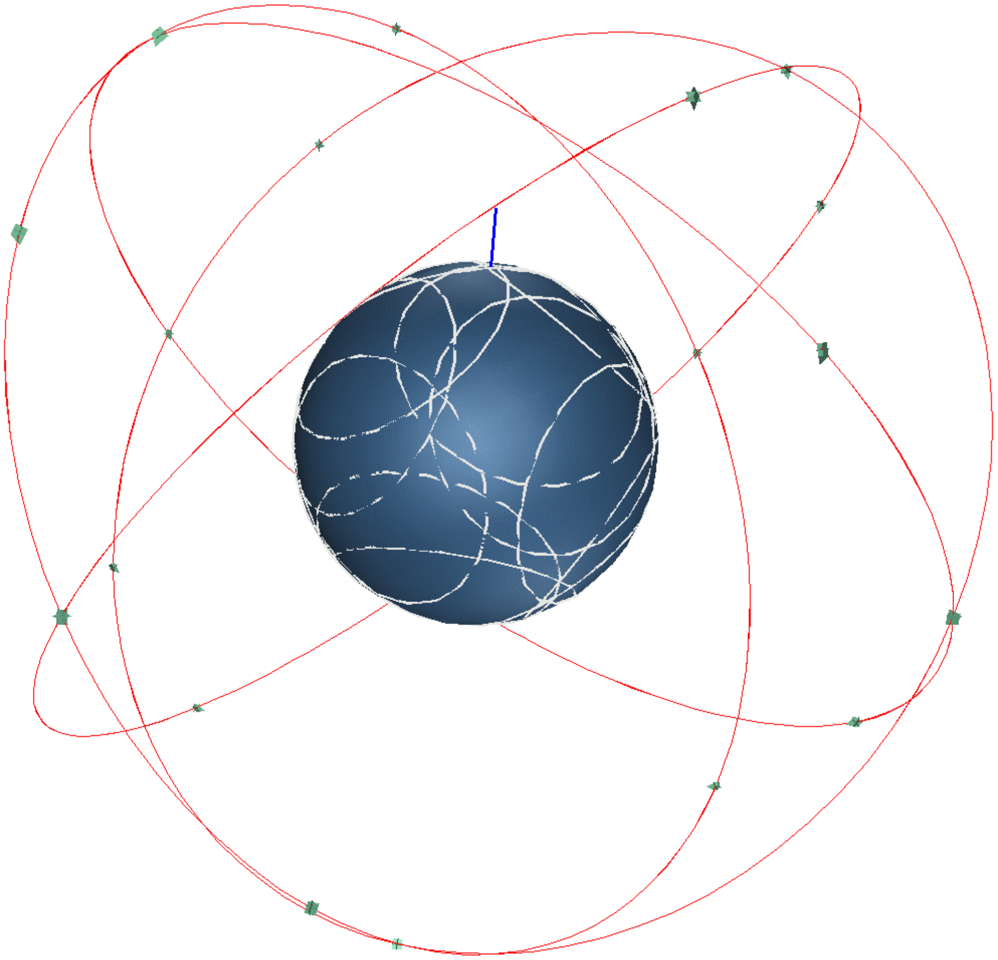
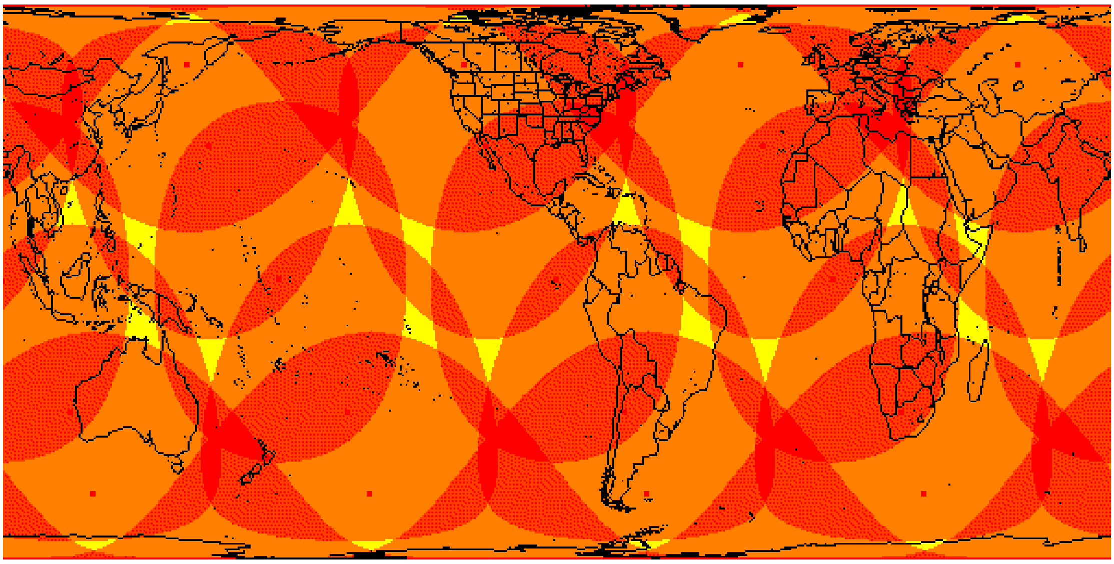


**Figure 1.5 - MEO *Spaceway NGSO* constellation at a moment in time (*SaVi*)**



cylindrical projection, using minimum elevation angle of 30° [**FCCSpacewayNGSO**]

**Figure 1.6 - global coverage from *Spaceway NGSO* MEO constellation (*SaVi*)**

### 1.3.4 Highly Elliptical Orbits (HEO)

Use of elliptical orbits differs from the continuous-though-moving coverage of circular orbits. Coverage for communications services from elliptical orbits is generally only provided when the satellite is moving very slowly relative to the ground while at apogee, furthest from the Earth's surface, and power requirements in link budgets are dimensioned for this large distance.

When the satellite moves from near high apogee to low perigee and back at varying speed in accordance with Kepler's third law, its coverage area zooms in size, and service is generally disabled. (Other satellites in the constellation are nearing apogee in their own orbits and providing service coverage in its place as the Earth rotates). The satellite's electronics may even be shut down to protect them from damage while passing through the Van Allen radiation belts.

Useful elliptical orbits are inclined at 63.4 ° to the Equator, so that orbital motion near apogee appears to be stationary with respect to the Earth's surface. High inclination and high altitude enable coverage of high latitudes.

Use of *Molnya* (or *Molniya*) and *Tundra* elliptical orbits is now well established for providing satellite television services targeted to the high-latitude states of the former Soviet Republic [Figures 1.7 and 1.8, **MaralBousquet98**, ch. 7].

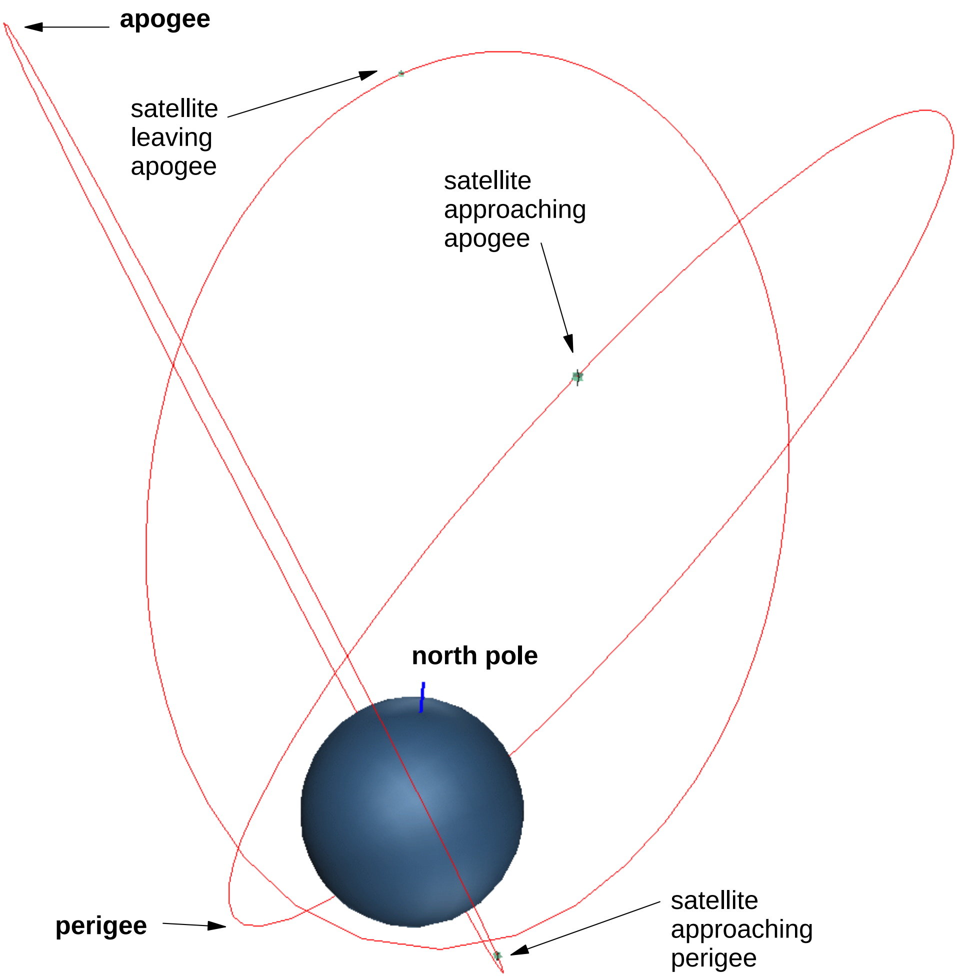
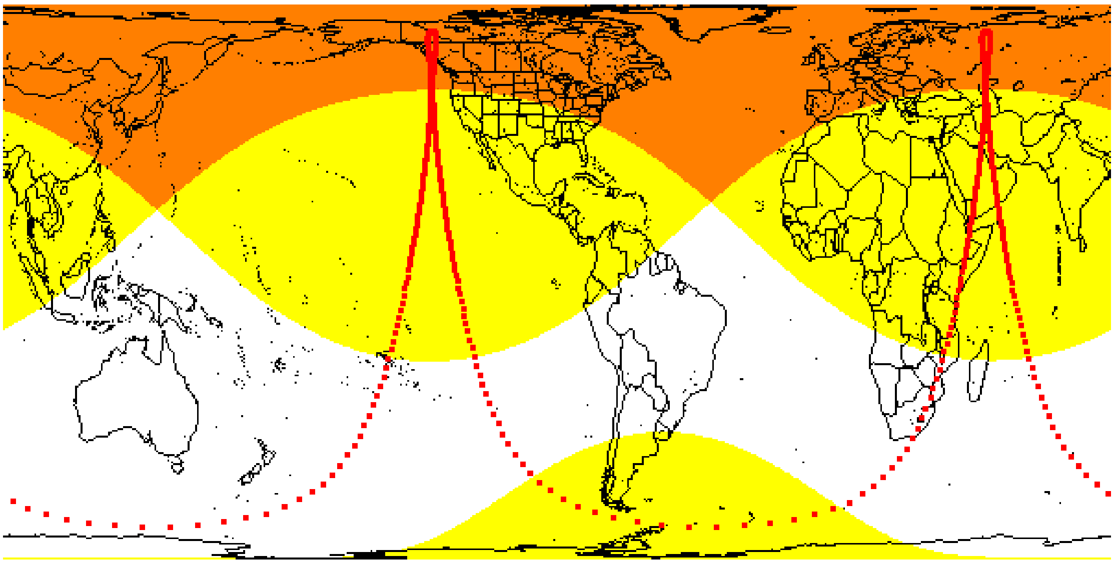


**Figure 1.7 - HEO *LOOPUS* constellation at a moment in time (*SaVi*)**



repeating shared-groundtrack plot at 60s intervals shows apogee loops and higher speeds at perigee.

One satellite is near perigee, moving to replace the satellite over Asia as it begins to leave apogee.

cylindrical projection, showing maximum range of visibility (minimum elevation angle<1°)

**Figure 1.8 - *LOOPUS* 3-satellite *Molnya* constellation [as in Dondl84] (*SaVi*)**

Proposed elliptical constellations include the hybrid *Ellipso* constellation [**CastielDraim95**] for voice services, and *Pentriad* [**FCCPentriad97**] and *Virtual GEO* [**FCCVirtualGEO99**] for broadband data. The latter two proposals use elliptical orbits with apogees beyond geostationary orbit and larger propagation delays. The *Virtual GEO* proposal also intends to establish intersatellite links between satellites nearing the apogees of different elliptical orbits at similar times.

Draim has explored the properties of elliptical orbits and optimisation of their coverage extensively since the 1970s, although his work is usually found in patents rather than in academic papers [**Draim89**, **DraimCefCas00**]. That work is used in the design of *Ellipso*.

As constellations using elliptical orbits are the exception rather than the general case, and are generally used to provide coverage to selected areas of the Earth, rather than to provide complete global coverage, they are not considered further here.

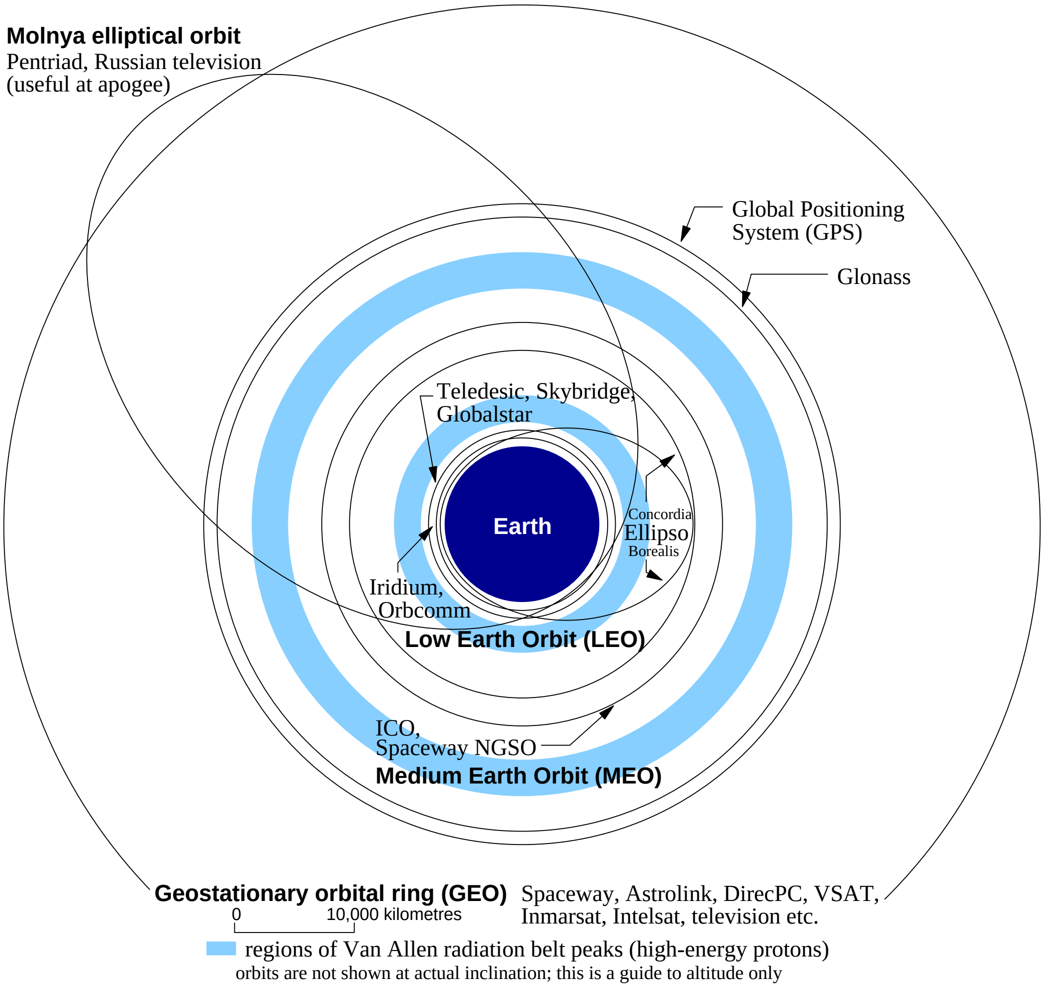


**Figure 1.9 - altitudes of orbits used by satellite constellations**

### 1.3.5 Illustrating constellations and altitudes

Figure 1.9 compares the altitudes of the systems that were discussed.

Elliptical, medium and low Earth orbits are not shown at their true inclinations, but are contrasted with the GEO orbital ring at the same inclination so that their altitudes can be compared.

Approximate positions at the Equator of the areas of peak intensities of the Van Allen belts are indicated. However, the effects of the solar wind on the magnetosphere are not represented, and the belts' following of the magnetic field lines inwards to the Earth's magnetic poles is not shown. To ensure that different LEO orbits are clearly visible, the large *Tundra* orbit is not presented in this figure.

Summary details of the constellations simulated in detail in this thesis are presented in Appendix 5 at end.

## 1.4 Networking approaches

From a networking viewpoint, there are two fundamentally different approaches taken by these broadband constellations. Each constellation consists of a number of similar satellites, where the satellite design is based around one of the following approaches:

- a ground-based approach, where network functionality is entirely terrestrial;
- a space-based approach, where the space segment possesses network functionality.

These two approaches will now be discussed in detail.

### 1.4.1 The ground-based constellation network

Here, each satellite is a space-based retransmitter – either using 'bent-pipe' frequency shifting and amplification, or using signal regeneration with baseband digital signal processing (DSP) – of traffic received from user terminals and local gateways below it on the ground, returning the traffic to the ground. This allows isolated user terminals to exchange traffic with nearby ground stations that are gateways into the terrestrial network infrastructure. The satellites provide a wireless 'last hop' to an extensive ground network. From a networking viewpoint, this poses challenges in the space segment for medium access control (MAC), logical link control (LLC) and handover.

Commercial systems taking this approach include *Globalstar*, the *ICO Global* and *SkyBridge* proposals, and many traditional geostationary satellites.

As the satellites are only used to provide last-hop connectivity, the topology of such a ground-based constellation network is entirely arbitrary, but is likely to be governed primarily by geographic, economic, political and legal considerations, which determine the locations of terrestrial gateway stations. It can be assumed that all satellite telemetry, tracking and control (TT&C) ground stations will be networked, to share information about the state of the constellation. However, beyond that there are a large number of networking possibilities and a number of different ways that the constellation gateways can be integrated with existing terrestrial networks, including the Internet. As a result, the design and topology of the terrestrial network component of a ground-based constellation such as *SkyBridge* is far more arbitrary than that of any space-based constellation whose network topology is governed by orbital geometry.

### 1.4.2 The space-based constellation network

In the space-based constellation network, each satellite has on-board processing (OBP), and is a network switch or router that is also able to communicate with neighbouring satellites by using high-frequency radio or laser intersatellite links (ISLs). This allows a user terminal on the ground below the satellite to exchange traffic with gateways to the terrestrial network or with users below distant satellites not visible to that terminal, without requiring a local gateway or significant terrestrial infrastructure to do so.

With ISLs, the space segment has now reached the network layer. This has led to consideration of network routing for communication across a mesh of ISLs between multiple satellites in orbit. Satellites in such constellations must support onboard routing as well as onboard switching. In this case, the satellite constellation is itself a true network. In conjunction with its terrestrial gateway stations the satellite constellation forms an *autonomous system* (AS). This is in contrast to the ground-based approach, where it is possible that each terrestrial gateway can be an entirely separate AS.

Commercial and proposed systems utilising ISLs include Motorola’s deployed LEO *Iridium* constellation, the proposed LEO *Teledesic* constellation, Hughes’ GEO

*Spaceway* and the MEO *Spaceway NGSO* proposals [**FCCSpacewayNGSO97**], and the GEO *Astrolink* proposal [**Elizondoetal97**].

For circular orbits, it is possible to use fixed ‘fore’ (ahead) and ‘aft’ (behind) intersatellite link equipment for intra-plane communication with satellites holding stationary relative positions within the same plane. Fixed equipment cannot be used for the interplane communication between satellites in different orbits, as the line-of-sight paths between these satellites will change angle and length as the orbits separate and converge between crossings, resulting in:

- high relative velocities between the satellites,
- challenges for tracking control as antennas must slew around,
- Doppler shift.

As a result, tracking is required as the relative positions of the satellites alter. Tracking requirements for ISLs in LEO star and rosette and MEO constellations, their feasibility and the range of slewing angles required are discussed in [**WernerJahnLutzBott95**] and [**WernerWauFringMaral99**].

In elliptical orbits, a satellite would see the relative positions of satellites fore and aft rise or fall considerably throughout the orbit, and controlled vertical pointing of the fore and aft intra-plane link antennas would be required to compensate for this. However, since elliptical orbits are only useful for large-area coverage at slow-moving apogee, there is little value in implementing intra-plane ISLs. No commercial proposals attempt this.

Interplane crosslinks between quasi-stationary apogees (as proposed for *Virtual GEO*) appear straightforward to maintain from a tracking viewpoint, although the distances between satellites are extremely large.

### 1.4.3 Comparison of ground- and space-based network approaches

The space-based intersatellite-link approach, where satellites communicate directly with each other by line of sight, can decrease ground-space traffic across the limited available air frequencies assigned to the constellation. It removes any need for multiple ground-space hops required for communication between distant ground terminals for the ground-based constellation.

However, in order to support the ISLs with network-layer functionality, the ISL-based approach requires more complex and sophisticated processing/switching/routing onboard satellite than the ground-based approach. This complexity and networking in the space segment enables the ISL network to complete communications and gives it the advantage of providing ubiquitous service in regions where the locally-overhead satellites visible to a ground terminal are not simultaneously visible from a ground gateway station.

This differs from systems using simpler 'bent-pipe' frequency amplifying/shifting satellites, which act as simple transponders. In those simpler systems, the ground-based network is limited by the distribution of its terrestrial gateway stations, and satellite terminal users must be near enough to a gateway station to share a satellite footprint.

The design of the space-based constellation network is restricted by the constraints imposed by orbital geometry and the difficulties in implementing networking in the space segment.

The ground-based constellation network separates network functionality from the space segment, allowing network-layer issues and space-segment issues to be considered separately.

The ground-based approach has the advantages of allowing reuse of the satellites for different purposes by simply upgrading or replacing easy-to-access ground equipment, and of reducing system complexity. In the more complex space-based constellation network, networking and space segment issues must be considered together.

Of the proposed broadband data constellations, *Teledesi*c and *Spaceway NGSO* are examples of space-based networks using an ISL-based approach, while *SkyBridge* takes a ground-based approach without ISLs.

If ever built, *Teledesic* can be expected to provide global service when satellites are deployed, while *SkyBridge* service deployment will be limited to the vicinity of constructed gateway stations, and will increase over time as more gateway stations are constructed. This is even though the two system designs are planned for similar orbital altitudes, at close to 1400km.

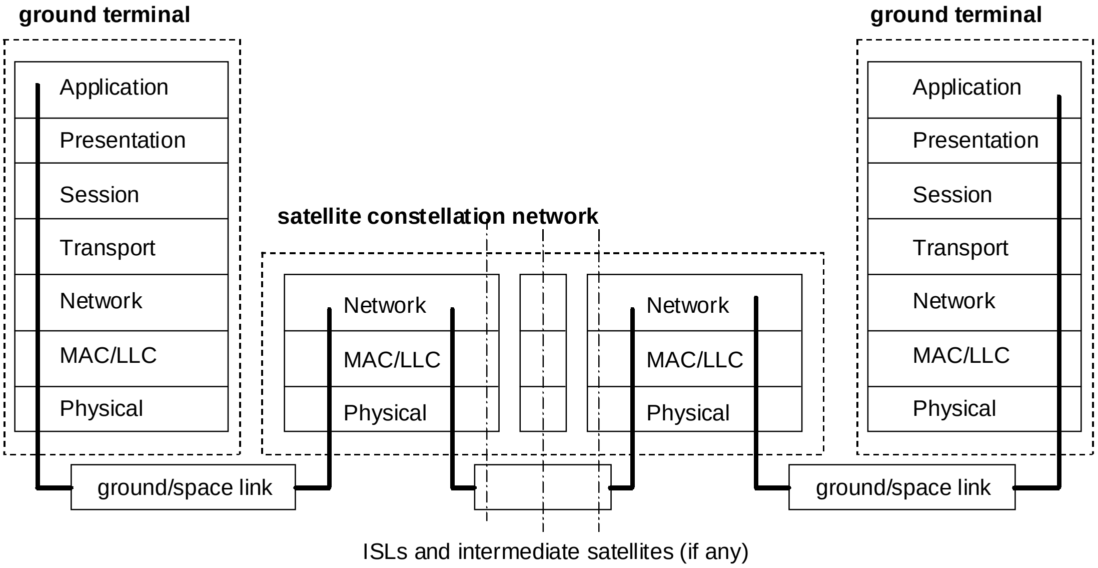


**Figure 1.10 - ISL routing approach, e.g. *Teledesic* [Bisante99]**

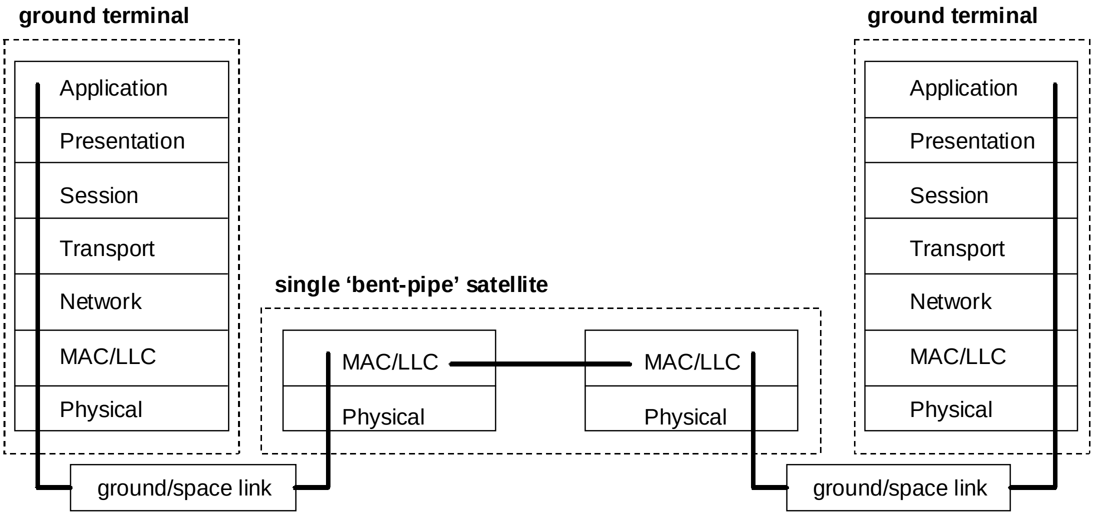


**Figure 1.11 - relay 'bent-pipe' satellite approach, e.g. *SkyBridge* [Bisante99]**

We can illustrate these two different approaches conceptually from a networking viewpoint with stack diagrams, presenting use of the various networking layers, shown in figures 1.10 and 1.11. This thesis examines networking and internetworking issues. It will therefore concentrate on the more complex case of satellite constellation networks using intersatellite links, as shown in figure 1.10, where:

- the constellation is seen from a networking viewpoint as an autonomous system;
- the space segment is part of the network layer;
- constraints on the network layer can be more clearly articulated and examined.

## 1.5 Summary

This introductory chapter has briefly overviewed the history of satellite networking and introduced the satellite constellation. It has examined, categorised, discussed and contrasted satellite constellation networks in terms of:

- services provided,
- orbital choice and altitude,
- the approach taken to networking in the constellation design,

before going on to summarise the thesis and remaining chapters.

## 1.6 About this thesis

This section presents the overall scope and contributions of this thesis, and summarises its structure.

### 1.6.1 Problem statement and scope

This thesis takes the Internet and broadband satellite constellation networks as parallel developments that are increasingly overlapping and converging. It examines how these two developments can be combined effectively to provide seamless internetworking with the terrestrial networks that constitute the Internet, in order to bring advanced Internet services to satellite users world-wide.

It is assumed that the constellation is itself a network, and that the space segment of the constellation network supports a network layer. Issues important to effective use of IP, such as how support for emerging Internet services – for IP multicast and for IP Quality of Service (QoS) – can be enabled by supporting IP routing, are examined in detail.

Taking an approach based around constellation geometry, the effects of routing choices in the constellation network on the performance of TCP and multicast traffic are examined, with an emphasis on path delay metrics.

We demonstrate how classes of service can be introduced by exploiting constellation geometry, handover and diversity to manage delay across a purpose-designed constellation network.

### 1.6.2 Contributions of this thesis

This thesis:

- examines constellation geometry and the relationship between ‘star’ and ‘rosette’ constellations, and explores their impact on network routing and on delay experienced by traffic travelling through the constellation network.
- demonstrates clearly that TCP performance (the actual throughput, or ‘goodput’, of a connection) can be degraded by poor multipath routing algorithms within the constellation, and that a flow-based approach can avoid this degradation. The throughput of TCP/IP across a mesh network with multipath routing is shown to be degraded further by use of delayed-acknowledgement implementations at receivers.
- proposes an approach to multicast within the satellite constellation network, demonstrates network capacity savings from the use of multicast, and presents and evaluates a novel algorithm for placing the core in a shared multicast tree.
- proposes an architecture enabling IP routing in the satellite constellation network, so that IP QoS and IP multicast can be supported well. This architecture is based on using MPLS over an ATM-based substructure. A method of supporting IP QoS in the satellite constellation network, based on using differentiated services for scalability, is presented, and objections to the use of IP routing are examined in detail.
- introduces a novel method of gaining controlled delay for service classes in rosette constellations with full surface coverage, where handover is carefully managed at ground terminals to exploit diversity.

### 1.6.3 Organisation of this thesis

**Chapter one**, this introduction to the thesis, has briefly outlined the development of satellite networking and the growing desire for provision of Internet services in the broadband satellite network. It has discussed and contrasted the various types of satellite constellation, categorising them by intended use, by orbital altitude, and by the approaches taken to networking in the space segment. Advantages and disadvantages of those approaches have been examined. Finally, the remainder of the thesis is summarised.

**Chapter two** discusses the orbital geometry unique to the satellite constellation, and examines the effect of this geometry on network topology and on end-to-end delays experienced by traffic travelling across satellite constellation networks between ground terminals. The impact of the orbital seam in Walker star constellations on path propagation delay is examined in detail by simulating a *Teledesic* proposal, and the desirability of cross-seam links as a way to reduce delay variation in network traffic is shown. Handover events and their impact on network traffic are presented in detail and analysed.

**Chapter three** examines the effects of multipath routing on end-to-end performance of TCP communications across constellation networks with ISLs. It concludes that, for the highest TCP performance, IP traffic in the constellation should be routed in engineered, ordered, flows in order to avoid invoking TCP's congestion algorithms. This consideration affects the design of routing in the constellation network, and limits choices of routing algorithms in the space segment. The impact of delayed-acknowledgement implementations on end-to-end performance across multipath routing is examined in detail.

**Chapter four** examines IP multicast, how it can be implemented in the constellation, and the benefits of doing so in terms of network capacity savings. A novel algorithm for choosing a core for a shared multicast tree in the constellation network, based around an application of the ordered core-based tree protocol, is proposed and evaluated here.

**Chapter five** examines approaches to routing in the constellation network. It discusses how IP routing functionality is necessary for the support of emerging services in the IP framework, such as multicast and QoS, and looks at how this functionality can be supported within the space segment. It examines in detail a number of approaches to supporting IP routing: tunnelling, network address translation (NAT), and exterior gateway protocols. This chapter goes on to recommend the use of a traffic engineering approach, based around use of multi-protocol label switching (MPLS), to allow ATM and IP traffic to co-exist while enabling IP routing for IP traffic.

**Chapter six** examines diversity and handover management for rosette constellation geometries, in the light of the implications of geometry and topology discussed in Chapter two. It shows how handover management can be combined with diversity and normal shortest-path routing in a novel way that provides different levels of service to traffic, with different classes of delay and reliability. It shows how existing rosette constellation proposals can be modified to support this service-oriented approach.

**Chapter seven** concludes this thesis, summarises contributions and achievements, and then outlines areas for further work.

**References** supporting statements in this thesis are then listed.

**Appendices** provide a guide to related papers and software produced during the period that the work for this thesis was carried out. Detailed results, summarised in the main body of the thesis, are also presented here. Finally, orbital parameters describing the constellations simulated in detail in this thesis are summarised.

# *2. Network geometry, topology & delay*

Simulating a satellite constellation network requires an appreciation of how the satellites move over time, and when handover between satellites can or must occur.

Orbital geometry has a considerable effect on the design of a satellite constellation network. It influences satellite coverage and visibility of satellites available for use with diversity, physical propagation considerations such as power constraints and link budgets, and – particularly important from a networking viewpoint – affects the resulting dynamic network topology and round-trip latency and variation. As a result of this, the choice of orbits and the resulting satellite network topology must be considered carefully and characterised accurately.

The overall topology of the constellation as a discrete autonomous system or private network will have a considerable effect upon network performance and delay seen by applications. It is therefore worthwhile to examine the effect of orbital geometry on network topology.

## 2.1 Geometry and constellation type

Historically, literature discussing satellite constellations has concentrated on the positioning of a minimum number of satellites in regular, similar, planes for optimal ground coverage [**AdRid87**, **Walker84**, **Ballard80**].

Circular orbits are used for constant ground coverage, or fixed satellite footprint size throughout the orbit. In considering this, the literature has come to distinguish between two basic types of constellation geometry:

### 2.1.1 Type (1) polar and near-polar star constellations

The type (1) constellation consists of orbital planes inclined at a constant angle, near 90°, to that plane of reference. Their angles of ascension, and the spacing between them, fill 180° of that plane of reference.

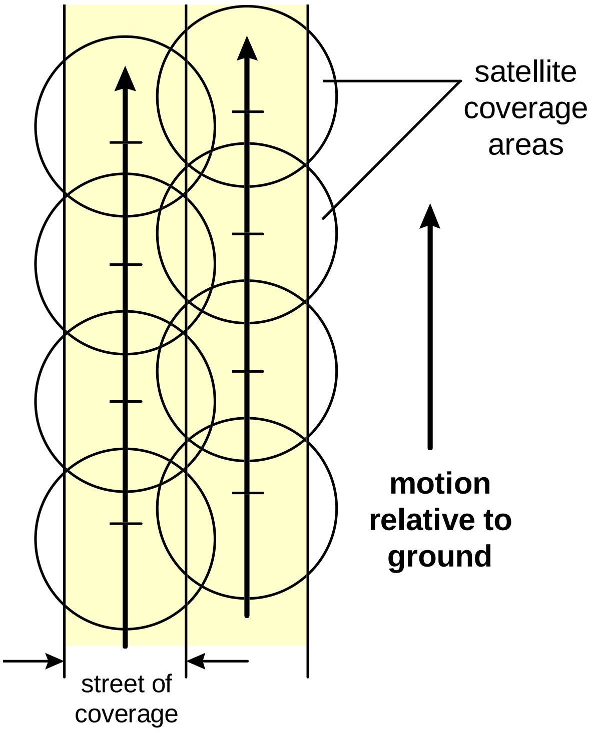


**Figure 2.1 - streets of coverage**

We can name this the π-constellation, for the angle in radians that is subtended in the plane of reference by the surface made by joining the evenly-spaced ascending nodes of the orbital planes. (The plane of reference is generally taken to be the Equator to minimise the effects of the Earth's oblateness and of differential precession between planes. When the Equator is used, the right angle of ascension through the plane of reference is the right angle of ascension through the Equator Ω, and the angle of inclination is the angle of inclination to the Equator δ.)

Many satellite constellations are based around the idea of co-rotating planes, slightly offset to provide full overlap. These planes provide *streets of coverage*, where satellites hand over communications to following satellites. This was proposed in [**Luders61**], discussed further in [**Rider85**], and is shown in figure 2.1.

Walker explored different types of constellations [**Walker71**, **Walker84**]. Type (1) near-polar constellations, with an orbital seam between ascending and descending planes, are called *Walker star patterns*. In the *star*, any point on the Earth's surface will see overhead satellites moving at regular intervals either from north to south (descending) or from south to north (ascending), except for points under the poles and the seam, which can see both.

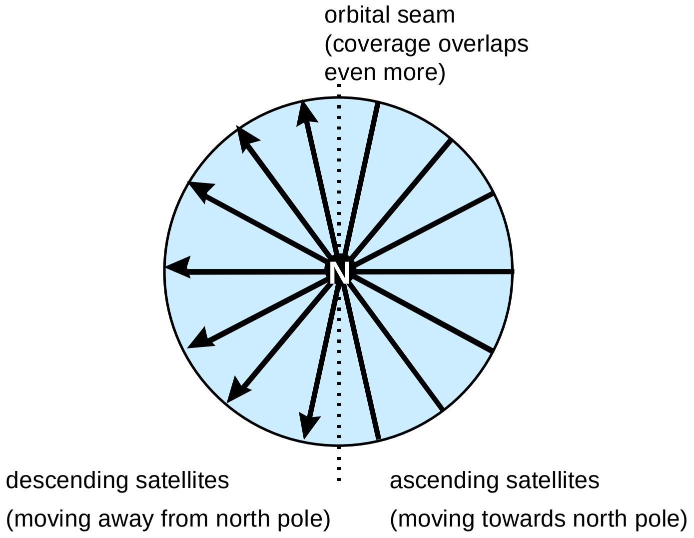
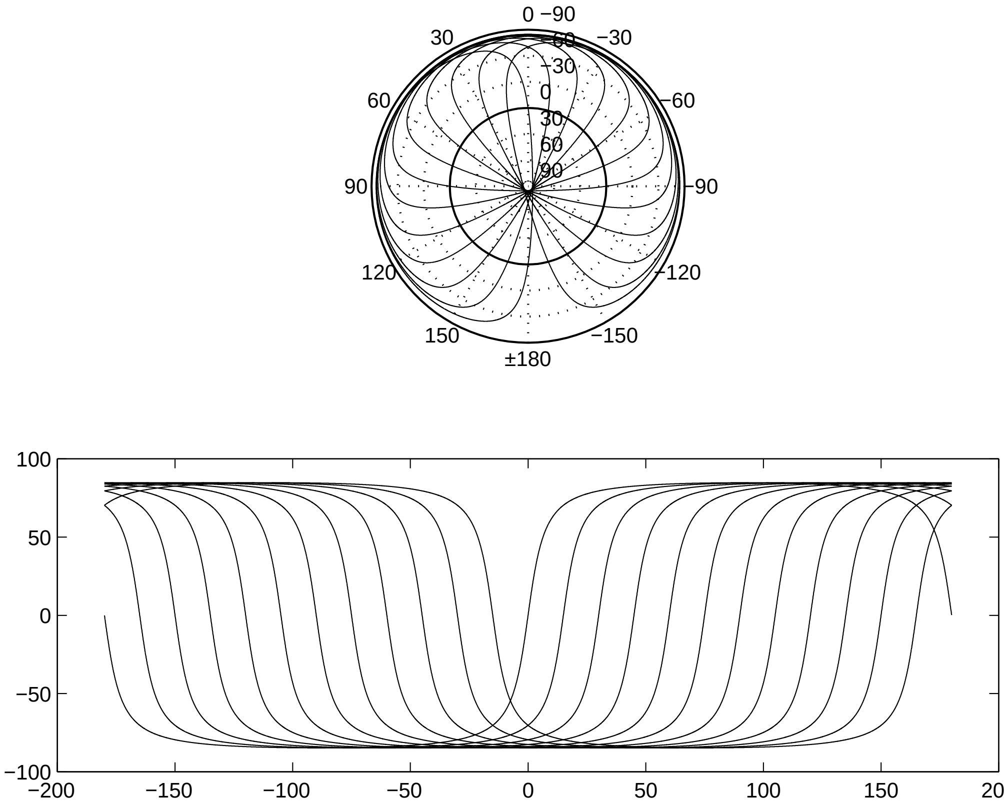


**Figure 2.2 - polar view of Walker star pattern**

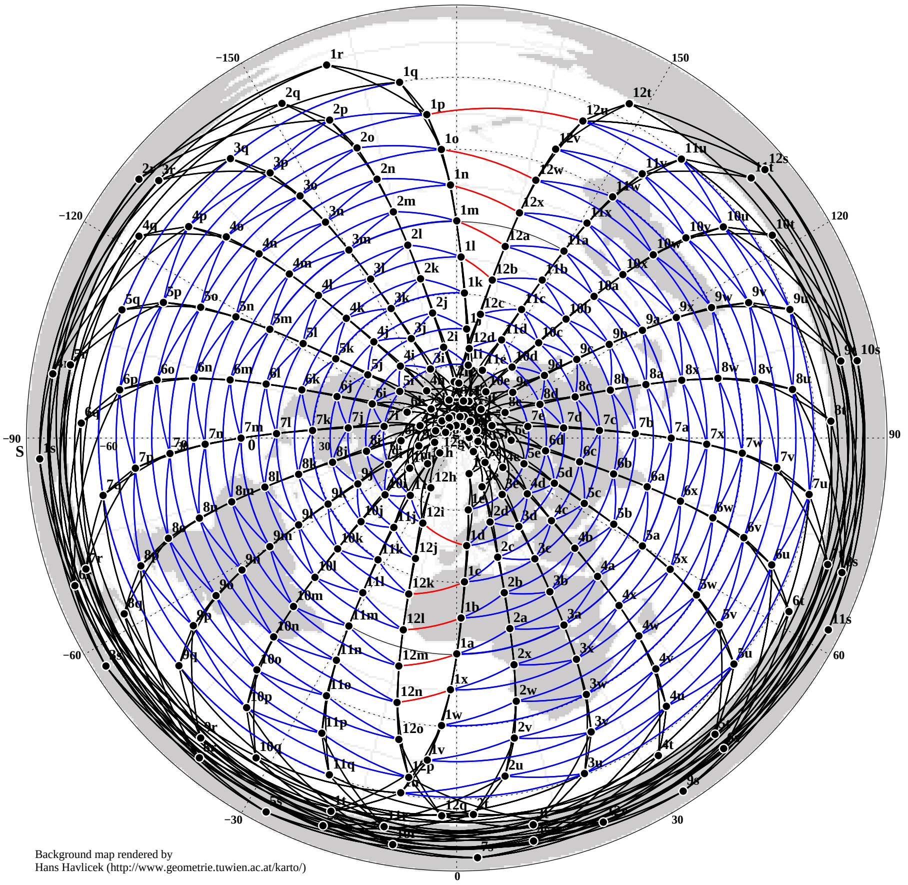


a. azimuthal equidistant projection  b. unprojected groundtracks

**Figure 2.3 - groundpaths of star at point in time (Boeing *Teledesic* design)**

The ‘star’ name comes from the fact that, if drawn on a polar map, the orbital planes intersect to make a star. This is shown conceptually in figure 2.2 and for the proposed 288-active-satellite Boeing *Teledesic* design in figure 2.3. As satellites in neighbouring planes are closer to each other at the poles than at the Equator, the star’s coverage is not even across varying latitudes, and increases towards the poles. The Equator provides the largest separation between planes and satellites that must be accommodated by footprint overlap for complete continuous coverage. The effect of the Earth’s rotation on orbital groundtracks, shown in figure 2.3, depends upon the altitude and inclination of the orbital planes in the constellation.

The Walker star adjusts the simple spacing of a polar circular-orbit type (1) between the two contra-rotating orbits nearest the orbital seam, or the edge where the ascending and descending 180° hemispheres meet [**Walker71**]. The right ascensions of the ascending nodes of the p orbital planes $\Omega_1 \ldots \Omega_p$ are evenly spaced, with the exception of a narrowed gap between the two contra-rotating planes at the seam edges. This ensures full coverage with overlap of the counter-rotating ‘streets of coverage’, or lines of footprints of the satellites in those opposing orbits as the planes of satellites move past one another. Interplane spacing is widened equally elsewhere to compensate for this seam overlap. This is illustrated in figure 2.4 for the Boeing *Teledesic* design. The orbital seam and **cross-seam links** are visible between planes of ascending and descending satellites. **Interplane** and **intra-plane** links are also visible.



Background map rendered by
Hans Havlicek (http://www.geometrie.tuwien.ac.at/karto/)

azimuthal equidistant projection

**Figure 2.4 - example constellation using Walker polar star (*Teledesic*)**

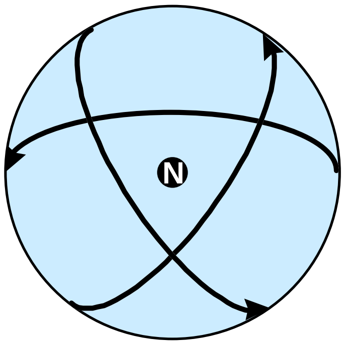
The high relative velocities of satellites travelling in neighbouring planes make maintaining ISLs at high latitudes harder due to increased Doppler shift, higher tracking rate, and the need to swap neighbours and re-establish links as orbital planes cross. Cross-seam ISLs may not be supported due to the high approach speeds (effectively twice orbital speed) of the ascending and descending satellites.

At the poles, the overlapping of satellite footprints will lead to multiple coverage and signal interference for simple frequency allocation schemes, resulting in a need for some footprints or spotbeams to be disabled. (*Iridium* disables spot beams near the poles to reduce this multiple coverage to single coverage with minimal overlap. [**YihChan93**].)

### 2.1.2 Type (2) delta constellations

A type (2) constellation consists of orbital planes inclined at a constant angle, generally less than 90°, to a plane of reference. Again, the Equator is generally used as the plane of reference for the reasons given in section 2.1.1. The even spacing of the right angles of the ascending nodes $\Omega_1 \ldots \Omega_p$ across the full 360° of longitude means that ascending and descending planes of satellites and their coverage continuously overlap, rather than being separated as with the Walker star.

We can call this the 2π-constellation, again for the angle subtended in the plane of reference by the surface made from the evenly-spaced ascending nodes. Walker named type (2) a *delta constellation*, again for the view at the pole [**Walker71**]. When (2) is at a minimum of three orbital planes, a rounded triangle, or Greek Δ, is formed by the around the pole by the planes, as shown in Figure 2.5.



no orbital seam;
ascending and descending satellites overlap

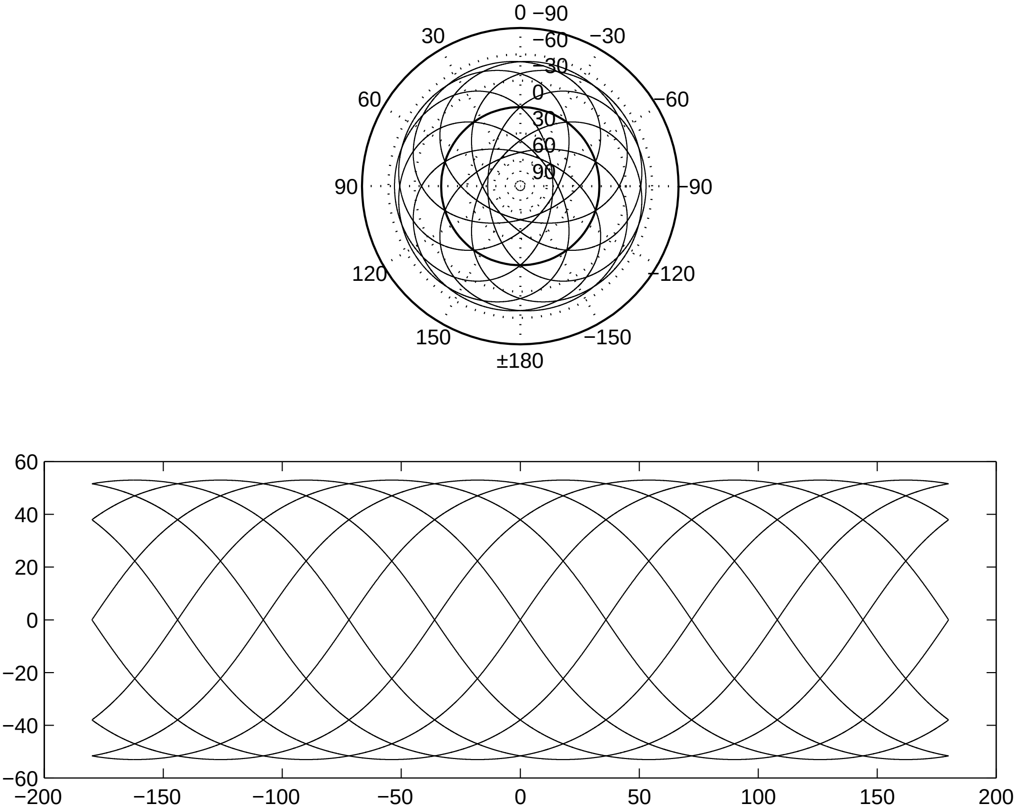
**Figure 2.5 - polar view of simplest type (2) constellation - Walker delta**

(A constellation with only two orbital planes can be thought of as a star constellation, since there an orbital seam does lie between the two planes of satellites.)

Ballard concentrates on the bounds of multiple satellite visibility by interleaving low-inclination multiple planes containing few satellites and using careful phasing to fill in the gaps between satellite footprints in the same plane [**Ballard80**]. Ballard calls the delta constellation an *inclined rosette constellation* and does not use the 'streets of coverage' approach.

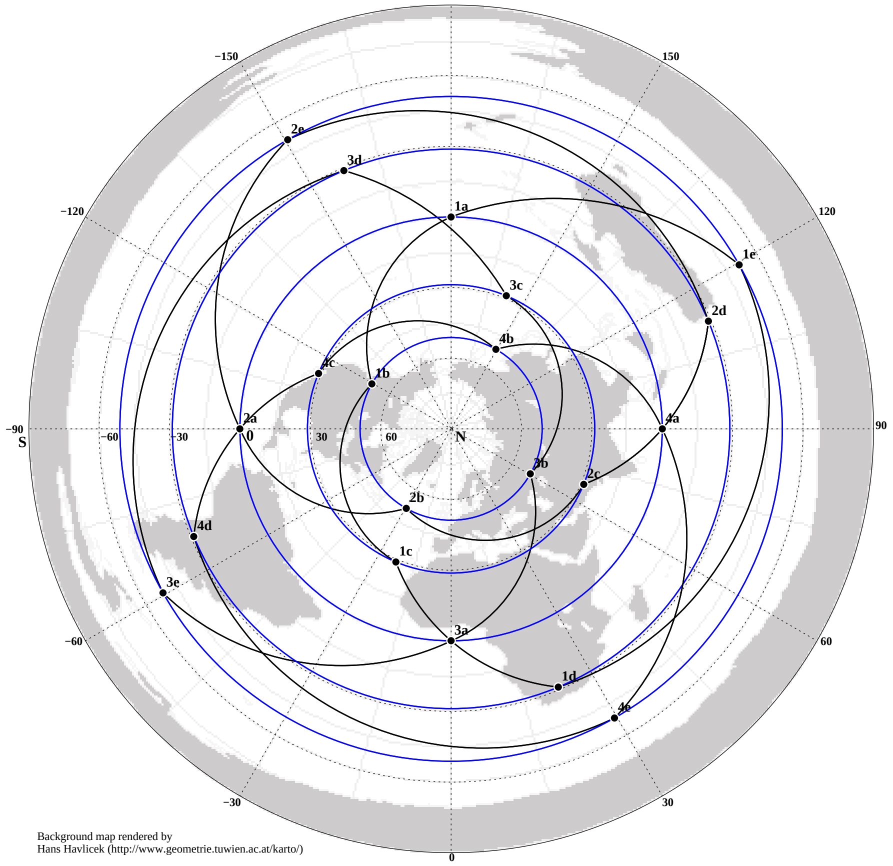
Although interleaving and careful phase alignment of planes for coverage can be adopted by rosettes, e.g. *SkyBridge* and *Globalstar*, to decrease the number of satellites required, their inclined orbits generally place more severe constraints on inter-satellite networking with intersatellite links. Relative velocities, tracking requirements, and Doppler shift are increased overall in comparison with interplane links for polar orbits. No rosette constellations have yet been implemented with ISLs.

The satellite groundpaths for a rosette constellation – a *SkyBridge* subconstellation – are shown in figure 2.6. The continuous overlapping of ascending and descending planes is clearly visible.



a. azimuthal equidistant projection  b. unprojected groundtracks

**Figure 2.6 - groundpaths of rosette at point in time (*SkyBridge* subconstellation)**



Background map rendered by
Hans Havlicek (http://www.geometrie.tuwien.ac.at/karto/)

azimuthal equidistant projection

**Figure 2.7 - example constellation using delta/rosette (*Spaceway NGSO*)**

There is no coverage above a certain latitude depending upon the value of δ; inclined rosette constellations generally neglect polar coverage, while providing the highest degree of coverage at the mid-latitudes.

The terms ‘delta’ and ‘rosette’ are effectively interchangeable. ‘Rosette’ is the more meaningful description for constellations with more than the minimum three orbital planes for which a ‘delta’ shape is visible in the polar view of orbital traces.

*Globalstar* deploys a rosette, and rosettes have been proposed for use in *SkyBridge*, *Spaceway NGSO*, *Celestri*, and other systems. The network topology of a MEO rosette constellation, the Hughes *Spaceway NGSO* proposal, is shown in figure 2.7.

### 2.1.3 Constellation notation

Today, both constellation types are known informally as Walker constellations, which can be confusing when describing constellation geometry. Type (1) is known as a ‘Walker star’ or ‘Walker polar’ constellation, while type (2) is known as an inclined constellation, a ‘Walker delta’ constellation, or as a rosette. Vargo called (1) and (2) λ-type and L-type respectively [**Vargo60**].

Constellation geometries are usually described in one of two forms in the literature:

**Walker notation:** *N/P/p*

number of satellites per plane *N*/number of planes *P*/number of distinct phases of planes *p* to control spacing offsets in planes.

Thus, the Boeing *Teledesic* design approximation shown in figure 2.4 can be described as 24/12/2.

**Ballard notation:** (*NP*,*P*,*m*)

(total number of satellites *NP*, number of planes *P*, harmonic factor *m* describing phasing between planes).

The *Spaceway NGSO* proposal shown in figure 2.7 can be described as (20,4,0).

The Walker notation is more commonly seen, although the Ballard notation can more accurately describe possible offsets between planes, especially when *m* is a fractional. Differences between the notations are discussed in [**Walker82**].

Note that the Ballard notation is assumed to describe a rosette where the ascending nodes are spaced over 360°, while in Walker notation this is unspecified and unclear. The Ballard notation is more complete and useful for its purpose.

### 2.1.4 Relationship between constellation types

Although global coverage for type (1) constellations is only achieved when the angle of inclination is on or near 90°, there is no reason that the inclination cannot be otherwise. However, the possible applications of such a type (1) constellation, with its variable latitudes of coverage as the Earth rotates, are questionable.

Similarly, there is nothing preventing a type (2) constellation, or rosette, from having a high inclination near 90°; orbital eccentricity can be adjusted to avoid collisions.

The relationship between the two constellation types that have been defined can now be stated. Two identical but opposing type (1) constellations, with the same angle of inclination and with semicircles of ascension evenly filling opposing 180° hemispheres of a common reference plane, can be combined to give us a type (2) constellation. This can be noted as:

$$(1) + (1) = (2)$$

or, naming the constellations according to the approximate angles subtended by the ascending planes in the plane of reference [**Galtier99**], while neglecting Walker’s star seam adjustment, we get:

$$\pi + \pi = 2\pi$$

This is shown graphically in Figure 2.8. A near-polar type (1) constellation using streets of coverage is sufficient to achieve global coverage.

The type (2) constellation that is made by summing two type (1) constellations of the same inclination, each with full coverage over a range of latitudes, can have two locally separate ISL network surfaces of satellites forming different parts of the network over those latitudes. One surface is of ascending satellites, the other of descending satellites.

These two surfaces are unlikely to communicate directly via ISLs established between counter-rotating satellites, due to demands on acquisition and tracking equipment caused by the high relative velocities of the satellites, and the resulting high rate of handover and Doppler shift. Ground stations in latitudes with full coverage from each type (1) constellation can see both locally-separate ascending and descending satellites, and are therefore able to communicate with the space segment using two different and locally-disjoint sets of entry points into the ISL network.

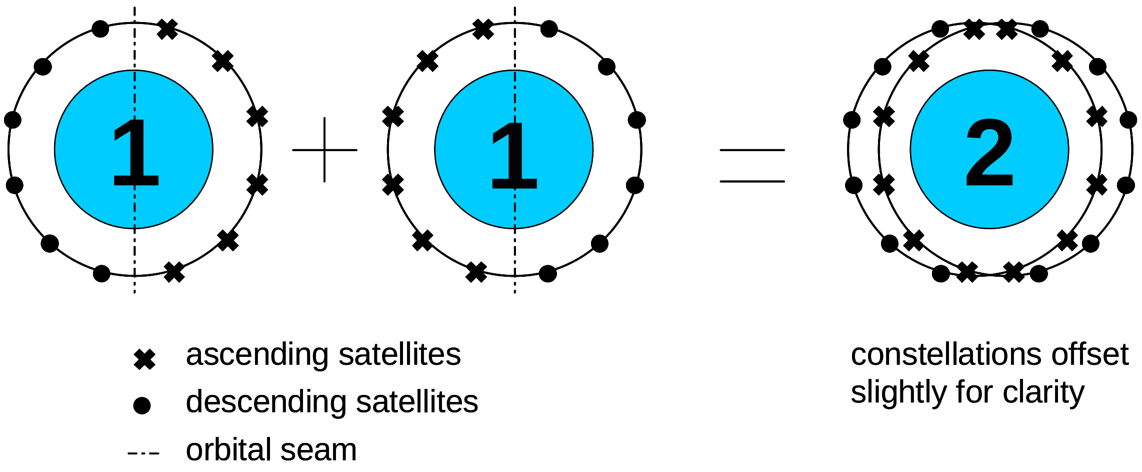


**Figure 2.8 - summing constellations through a plane of reference**

## 2.2 Intersatellite links and topology

Intersatellite links, or ISLs, act as lines (or, in graph theory terminology, as *edges*) joining the satellites (network nodes, *points* or *vertices*) to form the surface of the satellite constellation network. If each satellite is identical, and able to maintain the same number of ISLs (*degree of connectivity*), we have a regular network mesh composed of *intra-plane* and *interplane* ISLs.

Intra-plane ISLs lie within the same orbital plane between satellites following each other. They are generally permanent if the orbits are circular, as the satellites' positions remain fixed relative to each other.
Interplane ISLs are links between satellites in different orbital planes. These may not be permanent. Neighbouring orbits cross each other near highest latitudes, where each satellite's neighbours will swap sides, requiring ISL terminals between them to either physically slew through 180° to follow the neighbour and maintain the link, or that the links be broken and remade. The increased relative velocities as orbits approach also results in increased Doppler shift, which must be compensated for.

### 2.2.1 ISLs and toroidal surfaces

The type (2) network mesh loops in both intra-plane and interplane ISLs, and the mesh of ISLs can be embedded on a torus on whose surface the constellation lies.

A type (1) constellation must therefore lie on half a torus, bounded in one direction by the seam that sections the torus. Cutting the torus makes a cylinder.

Type (1) is bounded in one direction by the seam and loops in intra-plane ISLs. Its network mesh embeds on a cylinder, with half-twists at the highest latitudes where orbits cross and ascending satellites become descending satellites and vice versa. This has been shown diagrammatically [**Wood95**].

Any cross-seam links present stitch the edges of the seam together to form a spherical network. Cross-seam ISLs interconnecting the boundaries must be handed off frequently as the planes of satellites move past each other, in order to maintain connectivity. These cross-seam links have not yet been demonstrated in practice; *Iridium* – the first system constructed using ISLs – does not have cross-seam ISLs.

Adjacent orbits with inclinations near 90° have minimal Doppler shift and rate of ISL terminal slewing, and are therefore best-suited to interplane ISLs over a wide range of latitude. It is therefore unsurprising that the well-known commercial LEO constellation designs utilising ISLs, *Iridium* and *Teledesic*, are circular near-polar adjusted type (1) constellations. (*Iridium*'s few planes and wide spacing means that it only maintains three lines of cross-plane ISLs near the Equator, where Doppler shift and interplane satellite relative velocities are least.)

*Teledesic* can use its narrower spacing to maintain cross-plane ISLs over a wider range of latitudes, and also to form a geodesic mesh: each satellite maintains ISLs to its two near neighbours in each of four cardinal directions – fore, aft, port (left) and starboard (right) – for redundancy. This duplication allows seeking acquisition of a new link while a single existing cross-seam link is maintained [**Henderson99**].

The constellation network, either toroidal or semi-toroidal, effectively repeats its topology in a number of positions at regular intervals. Although the satellites move, the constellation network itself is quasi-stationary. However, movement relative to the ground, taking varying satellite visibility into account, results in the ground seeing a dynamic, changing topology, particularly for seamed semi-toroidal networks.

### 2.2.2 Manhattan network topologies

The Manhattan network is the underlying form of the ISL satellite constellation's network topology. This is a regular topology named after the regular grid street pattern established in Manhattan Island, New York. It is a toroidal mesh network with each node having four unidirectional links: two transmit and two receive [**Maxemchuk87**], as shown in figure 2.9.

Manhattan networks have been extensively discussed in computing literature in the context of parallel computing. The Manhattan network forms the basic topology of the type (2) constellation. The Manhattan network is also known as a multiaccess mesh or multimesh network [**ToddHahne97**]. The form of the Manhattan network with bi-directional links is known as either the bi-directional Manhattan network, as the HR$^4$-net [**ChungSharAg94**], or as the shufflenet.

However, orbital geometry leads to differences from previously discussed Manhattan networks. In fact, the constellation network is a slightly variant form, with bi-directional ISLs and with two half-twists where the sense of rotation changes due to each satellite seeing its neighbours swap sides while ‘down’ remains constant as the satellites reach and pass through their highest latitudes. The topologies of the space segments of *Iridium*, *Teledesic*, and *Spaceway NGSO* can be considered as variations on the type of bi-directional Manhattan network shown in figure 2.10.

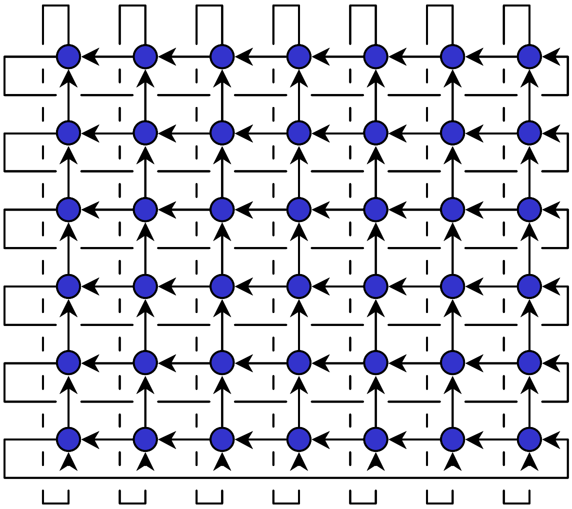
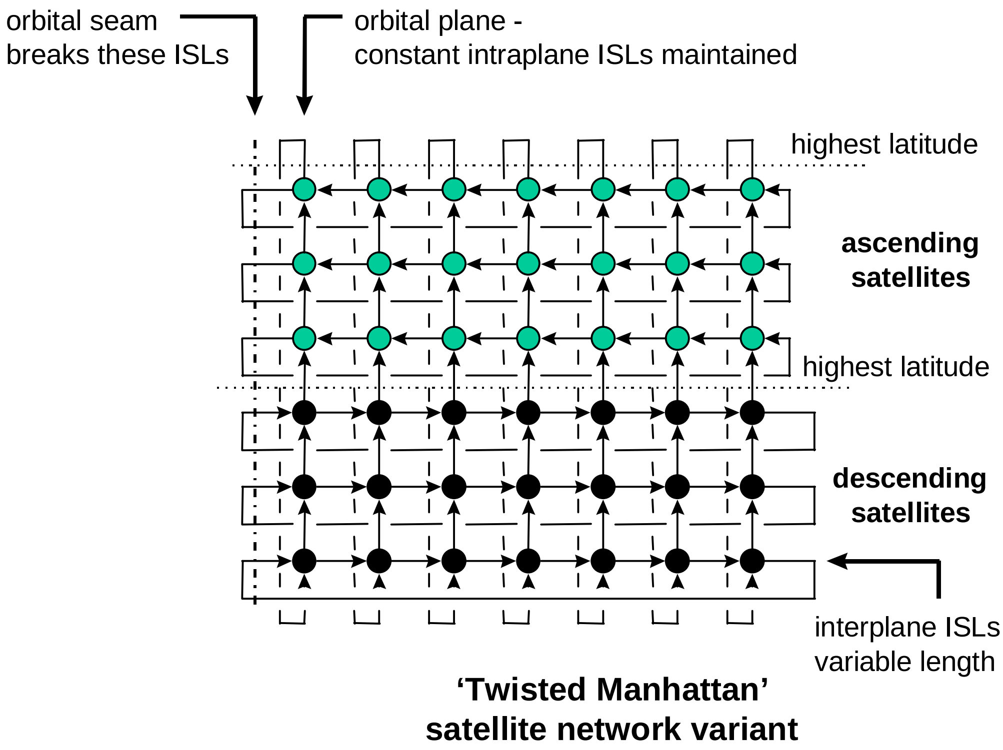


**Figure 2.9 - classic unidirectional Manhattan network**



bi-directional ISLs have added direction arrows purely to illustrate crossing of orbital planes at highest latitudes, where neighbours swap places.

**Figure 2.10 - ISL topology of LEO constellation: Walker 7/6/2**

Maxemchuk defines and discusses the Manhattan network, and provides simple packet forwarding rules, deterministic or random, that do not require the construction of complex routing tables.

Link and node failures are also discussed, although Maxemchuk's underlying assumption of node failure – that traffic will still be able to pass through failed nodes on rings – is not applicable to a switching satellite. We must consider a satellite failure as a failure of all links to and from that satellite, as well as considering partial losses in connectivity due to individual link equipment failures.

Maxemchuk assumes that a slotted-packet system is used and suggests strategies that use diversion to alternate links when a link is congested, so that packets are delayed by deflection but never dropped. This requires a link weighting metric reflecting link congestion. The cyclic structure of the network allows any node to consider itself as the centre of the network; this greatly simplifies consideration of routing schemes, such as the simple rule systems evaluated in [**GoodGreen86**].

Estimations of end-to-end packet delay, capacity and saturation are provided in [**ChungSharAg94**]. Chung et al. also provide routing algorithms for virtual circuits and datagrams across the network [**ChungRaiAg89**]. Similar work has been carried out for the bi-directional case [**WongKang90**]. A performance comparison of deterministic routing for the uni- and bi-directional cases has been provided [**AdVer94**]. [**BanBorGer92**] provides a summary of Manhattan deflection routing.

Wang recognised that the constellation network formed a Manhattan mesh and attempted to analyse delivery times across it [**Wang95**]. (Some shortcomings in his analysis are examined in [**Wood95**].)

A triangular Manhattan network, where each satellite node is the centre of six links at 60 degree intervals to form a hexagonal grid – either three incoming and three outgoing unidirectional links, or six bi-directional links – has been investigated [**MyZar90**]. From a purely topological viewpoint, this is relevant to the Motorola *Celestri* proposal, which was planned to have six ISLs per satellite, making a triangular bi-directional network. [**FCCCelestri97**, p33]. The original 77-active-satellite *Iridium* design planned up to six ISLs per satellite [**Leopold91**], although this was reduced to four in the deployed *Iridium* constellation.

### 2.2.3 Limitations of the Manhattan network approach

The recognition that the satellite constellation network topology is a variation of the Manhattan network topology has not yet enabled successful application of existing Manhattan routing strategies or deflection algorithms to the satellite network.

The satellite network has long delays, and variations of interplane link delay with latitude that mean that any satellite node selected is at the centre of a slightly different topology and sees a slightly different network. The Manhattan network is truly homogenous, while the satellite network is not. This means that simple Manhattan deflection routing strategies, based on the assumption that all nodes see the network similarly, will lead to varying network performance across the satellite network.

Although routing in the Manhattan network can be based upon a simple $(x,y)$ coordinate system that seems similar to latitude and longitude coordinates, use of geographic routing and addressing of ground terminals based on latitude and longitude has been found to be unsatisfactory in the constellation network [**Henderson00b**].

### 2.2.4 Topology and graph theory descriptions

A type (1) constellation without cross-seam links can be embedded in the plane with no overlapping edges, or on a cylinder. It forms a genus-zero graph.

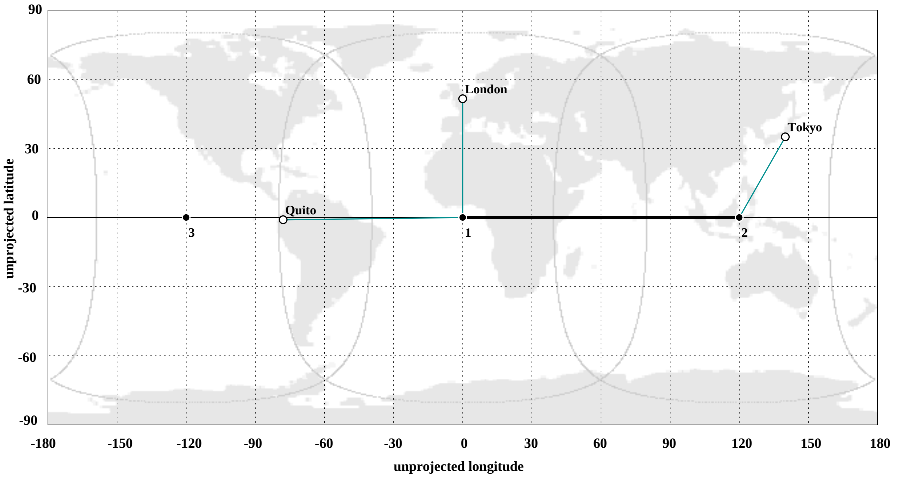
A type (2) constellation can be embedded on a torus with no overlapping edges. It forms a genus-one graph.

Let us define the constellation as having **m** planes with **n** satellites in each plane. Let it be locally a grid, where each satellite node, or vertex, is of degree 4, with connections to its nearest neighbours. Some definitions of interest to mathematicians can be stated.

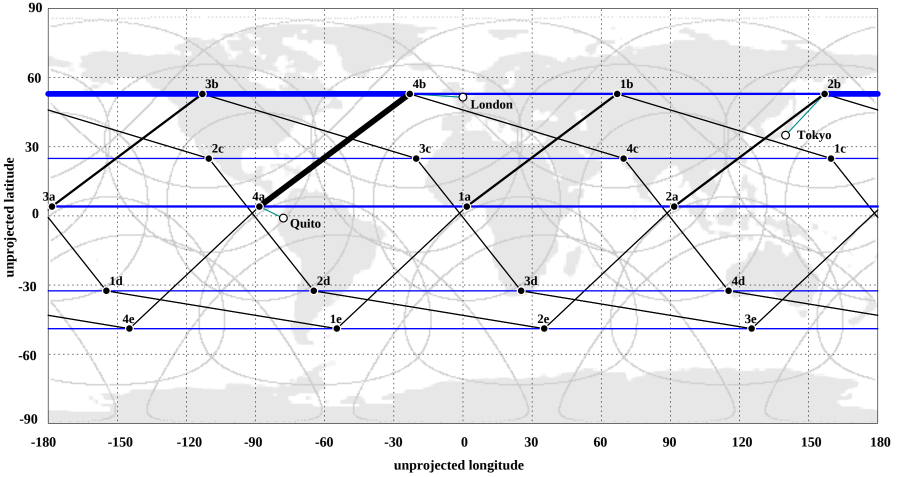
The type (1) constellation without cross-seam links approximates the product of an **n**-cycle with a path of length **m**. This neglects interplane links that are not maintained near highest latitudes.

The type (2) constellation that is produced from summing two opposing type (1) constellations similarly approximates the product of a 2**m**-cycle with an **n**-cycle, while a type (1) constellation with cross-seam links roughly approximates the product of an **n**-cycle with an **m**-cycle.

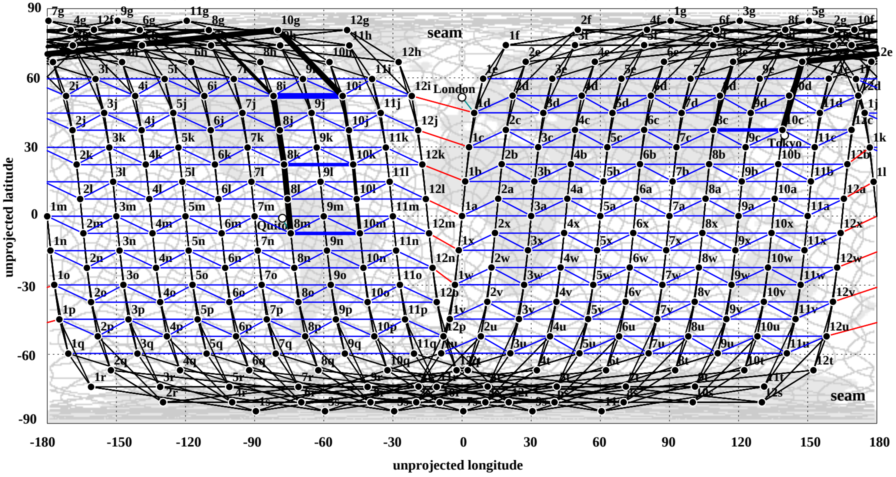
a. Minimum three-satellite Clarke GEO satellite network with ISLs, showing simulated ground terminals



b. Spaceway NGSO MEO satellite network with ISLs at a point in time, showing simulated ground terminals
ascending (moving north) and descending (moving south) satellites overlap



c. Teledesic LEO satellite network (Boeing 288-satellite design, optimised coverage), showing simulated ground terminals
descending satellites (moving south) ascending satellites (moving north)



– **terminal-satellite uplinks/downlinks** are handed off at MEO and LEO.

– **intra-plane ISLs** are permanent and are never handed off.

– **interplane ISLs** may break as satellites near each other at speed,
and may be handed off near highest latitudes as planes cross.

– **cross-seam ISLs** are temporary, regularly handed off as satellites pass.

**Thickest ISLs** show shortest path from Quito to Tokyo. **Less thick ISLs** show alternate paths with same hop metric.

**Figure 2.11 - satellite constellation networks at different altitudes**

### 2.2.5 Network topologies for example constellations

Unprojected maps showing the network topologies and coverage footprints of simulated LEO (Boeing *Teledesic* design), MEO (Hughes *Spaceway NGSO*) and GEO (*Clarke*) satellite constellation networks with ISLs are shown in figure 2.11. Here, connectivity via ISLs is shown by straight lines between satellites; these straight lines do not represent the actual paths taken by the ISLs above points on the map, and are intended only to illustrate connectivity and network topology. (Actual paths would mean that intra-plane ISLs would follow their orbital great circles, forming continuous curves.) Geostationary satellites can form a simple fixed-ring network around the Earth's Equator (figure 2.11a). At LEO and MEO altitudes (figure 2.11b and c) the ISLs between satellites form the more complex dynamic mesh topologies discussed in section 2.2.2. There, handover must occur at the terminal as satellites pass below the minimum elevation angle, interplane ISL handover may occur at highest latitudes, and cross-seam ISL handover will occur as satellites pass beyond local horizons.

Least-delay paths between Quito and Tokyo, where routing uses a delay metric for path cost and chooses the lowest-cost (shortest-delay) path across the constellation, are illustrated by the thickest lines in figure 2.11. Similar near-shortest alternate paths, using the same number of discrete wireless hops (same 'hop count' value), where available at LEO and MEO, are also indicated by slightly thickened lines.

## 2.3 Latency and delay

Network latency results from queuing, switching and link-layer MAC capacity allocation as well as from channelisation and propagation delays, and the use of link-local error recovery techniques such as automatic repeat request (ARQ). Jitter due to queuing and switching will vary considerably according to the specific system design and implementation, and not just due to orbital geometry. Those contributing delays should also be small compared to the overall path propagation delays, making them second-order effects. The speed of light is the common largest constraint on path delay, and therefore should be examined to gain a first order-approximation of end-to-end delay across the constellation between ground terminals as the constellation and Earth move. IP traffic is suited to low ARQ persistency [**FairWoodpilcdraft00**]; we assume error-free first-time delivery so that effects of loss do not interfere with results.

i. One-way propagation delay for path between ground terminals, showing smooth satellite motion and abrupt handovers

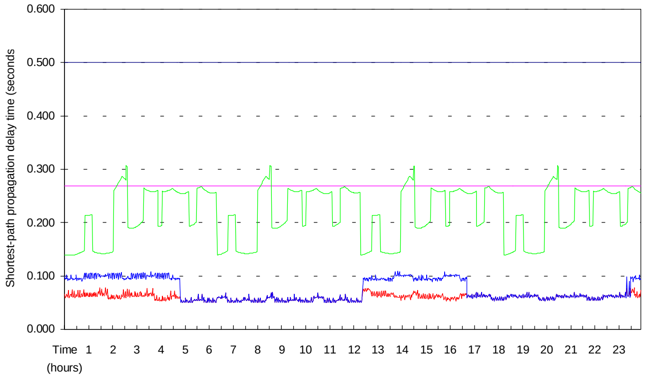


**Figure 2.12 - path propagation delay over time between Quito and London**

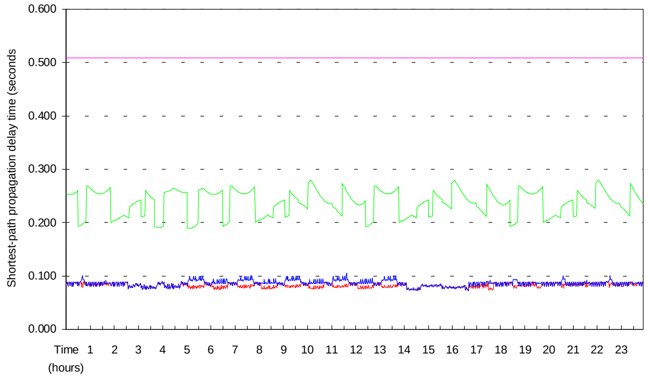


**Figure 2.13 - path propagation delay over time between Quito and Tokyo**

– Clarke geostationary (– alternate Quito-3-2-London) – *Spaceway NGSO* – *Teledesic* (seamed) – *Teledesic* (cross-seam links) up/downlink delays are included

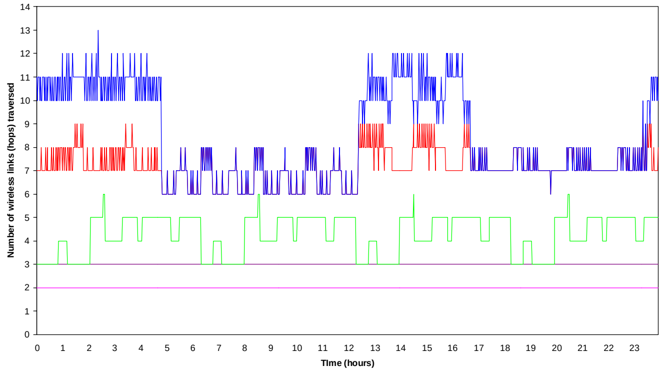


**Figure 2.14 - path length in hops over time between Quito and London**

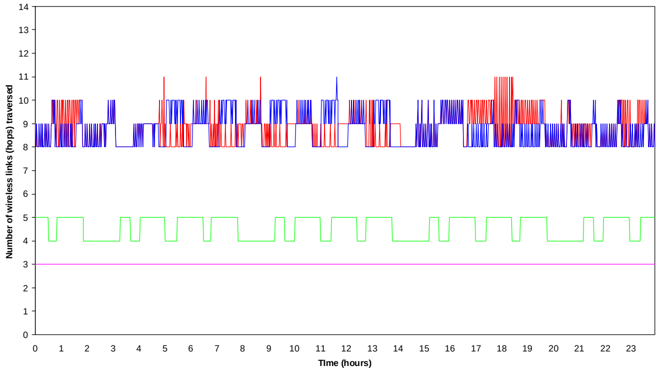


**Figure 2.15 - path length in hops over time between Quito and Tokyo**

ii. Hop counts of wireless links traversed between ground terminals, showing path changes due to handovers

### 2.3.1 Delay profiles

Figures 2.12 and 2.13 show the propagation delay experienced by traffic travelling the shortest route across constellation networks between two fixed ground terminals. Handover occurs at terminals when currently-used satellites drop below the minimum elevation angle used for the constellation. This path delay was examined over the course of a day, so that the Earth rotates fully beneath the moving satellites. The figures show gradual changes in delay due to the orbital motion of satellites, as well as abrupt changes due to handovers at terminals and path changes caused by routing updates; these changes are small compared to the granularity of a TCP timer.

As the Earth rotates under the constellation, the constellation's planes move across the surface of the Earth. Second-order and minor physical effects on the movement of the Earth's surface relative to the constellation are not considered in these simulations. These include the effect of the Earth's oblateness upon rotation of the major axis of the planes and upon drift in $\Omega$, as well as local gravitational variations resulting from the Earth's lack of homogeneity and residual atmospheric drag from the thermosphere and exosphere.

As a first approximation, delays due to queuing and switching in each constellation design can be thought of as proportional to the number of wireless links traversed, as shown in figures 2.14 and 2.15.

Examining the shortest-path propagation delay across a constellation network provides a 'best-effort' goal with which real-world performance can be compared. However, using a metric of minimum propagation delay for cost-based routing algorithms to give a minimum delay, as done in e.g. [**EkiciAkBen00**], tends to load interplane links at the highest latitudes, where distances between satellites are shortest.

### 2.3.2 Comparisons with terrestrial fibre

If the 9200km between Quito and London on a great-circle route were laid with optical fibre, the light in the fibre, travelling in glass at $2\text{x}10^8\text{ms}^{-1}$, would take a relatively constant 46ms to cover the great-circle distance. This neglects the minor effects of pulse dispersion from internal reflections at varying angles. Routing around South America would increase delay and switching times, but less than 70ms propagation delay is a reasonable expectation. Similarly, the 14450km great-circle distance between Quito and Tokyo gives 73ms fibre propagation delay. These minimum-possible fibre propagation delay values are comparable to the minimum delays computed for the constellations. In practice, given geographical and political decisions affecting fibre location, fibre delays can be expected to be larger.

### 2.3.3 Rosettes, routing and delay

Rosette or type (2) constellations with ISLs, such as the MEO *Spaceway NGSO* network shown in figures 2.6 and 2.11b, offer regular repeating routing and delay patterns over time between ground terminals, as they do not suffer disruption from the orbital seam seen in polar star constellations.

Unlike the ‘polar star’ constellations, rosette constellations are able to offer two distinct sets of paths between terminals, depending on whether the terminals’ uplink and downlink communications have been allocated to ascending or to descending satellites. The satellites forming the shortest ISL path for the traffic may not be selected if:

- the terminal is at a high latitude and does not have a wide choice of satellites;
- demand for link capacity around a terminal is high, preventing a choice of the ‘optimum’ satellite for that traffic;
- handover is not coordinated for traffic across the entire network but is a local terminal decision;
- only single satellite coverage is available for one terminal, forcing a suboptimal path using a satellite not on the shortest ISL path to provide a link.

This is the case for the simulated *Spaceway NGSO* proposal presented here, where handover takes place only when the satellite currently being used drops below the terminal’s minimum elevation angle. This is the simplest, ‘baseline’ handover procedure, and is contrasted with other, more complex, methods in [**KrewelMaral00**].

If, in figure 2.11b showing the *Spaceway NGSO* constellation network at a point in time, the terminal at Quito had been allocated to the descending satellites 2c or 2d, it would be only one or two ISL hops from Tokyo, instead of three while using 4a.

With two complete separate ISL network surfaces of coverage and bi-directional ISLs, two disjoint sets of optimum ISL routes between satellites overhead are always offered between two communicating fixed ground stations. This is useful in offering redundancy of routing, providing a degree of network diversity, as well as satellite diversity. That is discussed further in Chapter 6 and in [**Woodetal01c**].

### 2.3.4 The orbital seam and cross-seam links in star constellations

As the Earth rotates under the constellation, the type (1) constellation's orbital seam travels westward across the Earth's surface. Part of the seam passes through each longitude every twelve hours. If two fixed ground stations, or terminals, communicate with each other using ISL routing between overhead satellites, the position of the seam relative to the stations will affect the choice of available ISL routes to take, the lengths of those routes, and the path delays incurred by using those routes.

A guarantee of constant or predictable delay is an important consideration for use of a LEO constellation for bi-directional real-time multimedia applications wanting to take advantage of the reduced propagation time compared to a GEO constellation. However, regular path rerouting resulting from the seam in a type (1) constellation prevent this delay from being reasonably constant between fixed ground stations in the footprints of different satellites. The delay varies with the time of day and position of the seam. This overall variation in path delay will be considerably larger than the minor variations in propagation delay caused by the changing elevations and distances of LEO satellites communicating directly with the terminals.

The presence or absence of cross-seam ISLs will have a visible effect on network performance, by altering the disruptive effect caused by the seam. There is a sudden, large, change in total path delays whenever the seam passes over one of the ground stations, as indicated in figures 2.12 and 2.14. Introducing cross-seam links minimises this change, and also decreases the largest delay seen over the course of a day as the Earth and constellation rotate.

The duration of the seam disruption is roughly proportional to the separation in longitude between the terminals, due to the Earth's rotation; each hour of disruption to traffic between communicating ground terminals equates to 15 degrees of longitude in separation between the terminals. Similarly, the size of the change in path will be roughly inversely related to the separation between terminals. As Quito and London are relatively close in longitude, traffic between them (figures 2.12 and 2.14) shows this disruption more clearly than traffic from Quito to Tokyo (figures 2.13 and 2.15).

### 2.3.5 The impact of the orbital seam upon path propagation delays

To examine in detail the effect of the orbital seam on traffic delays across a star constellation, the network simulator *ns* [**FallVaradhan00**] and its recent satellite simulation extensions [**Henderson00a**] were used. The 288-active-satellite Boeing *Teledesic* design, both with and without cross-seam ISLs, was simulated over the 24 hours. Using a full day allows ground terminals to see the constellation from the entire range of possible positions as the Earth rotates beneath the moving satellites.

Ground terminals were placed at different latitudes: together on the Equator (0° latitude), and at 30° and 60° latitude. One terminal was moved around the Earth at that latitude, simulating a day's worth of traffic at each new terminal position in order to determine the full range of propagation delays experienced by traffic between the terminals, over 0 to 180 degrees of longitudinal separation. As the constellation is symmetrical around the Equator, results for terminals placed on e.g. 30° and -30° latitudes are identical, although shifted in time by several hours.

This methodology is illustrated in figure 2.16. The resulting path delay curves from each simulation run, each of the form first shown in figures 2.12 and 2.13, are then analysed together, summarised with statistics, and presented in figures 2.17 to 2.21.

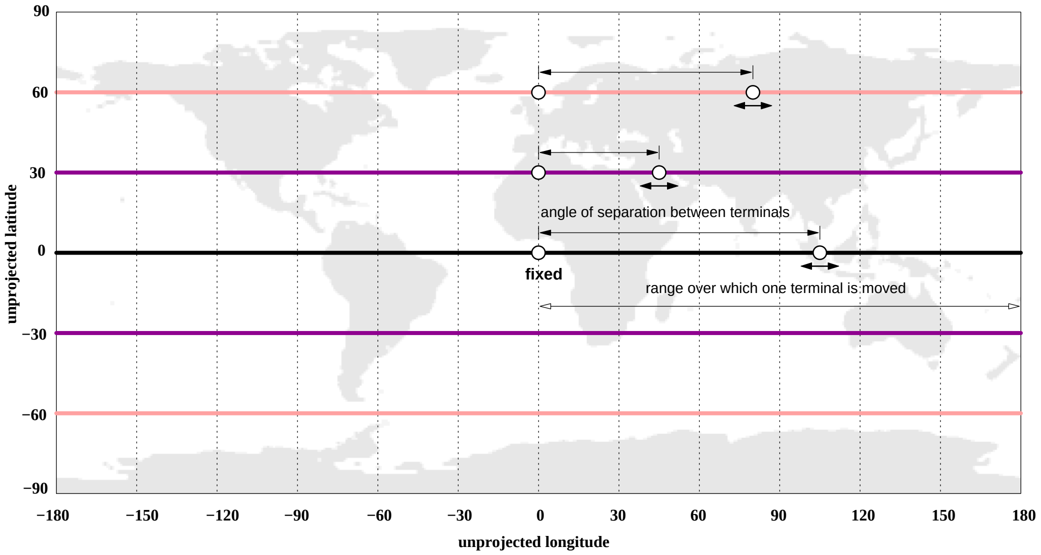


**Figure 2.16 - measurement of path delays at different latitudes between terminals**

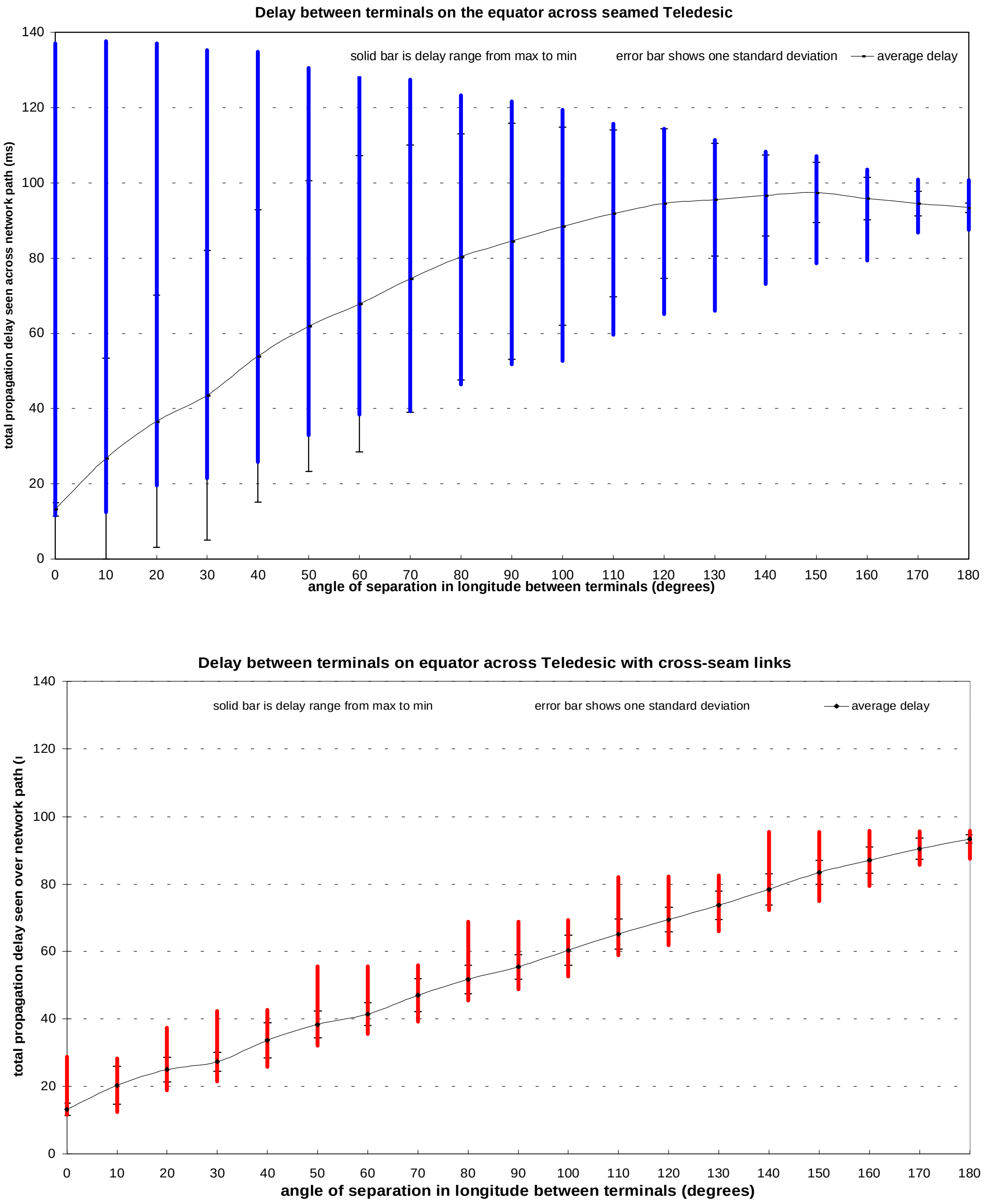
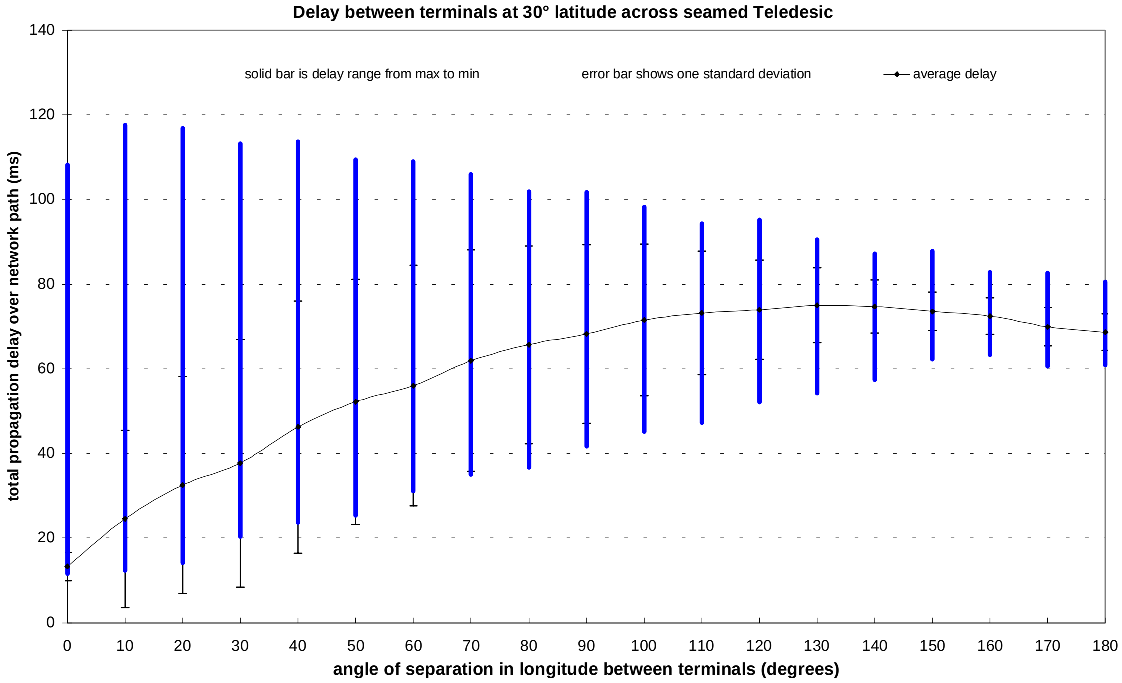


**Figure 2.17 - effect of cross-seam links in *Teledesic* on delay seen at Equator**

Figures 2.17 to 2.19 show how the introduction of cross-seam links decreases overall delay experienced by traffic between the terminals, Statistics are given for all the delay curves and separations at a given latitude. An indication of the degree of delay variation over the course of the day is indicated by the small bars. Those bars denote one standard deviation from the average delay for each simulation run, assuming that the sample distribution approximates to a normal distribution about the mean.



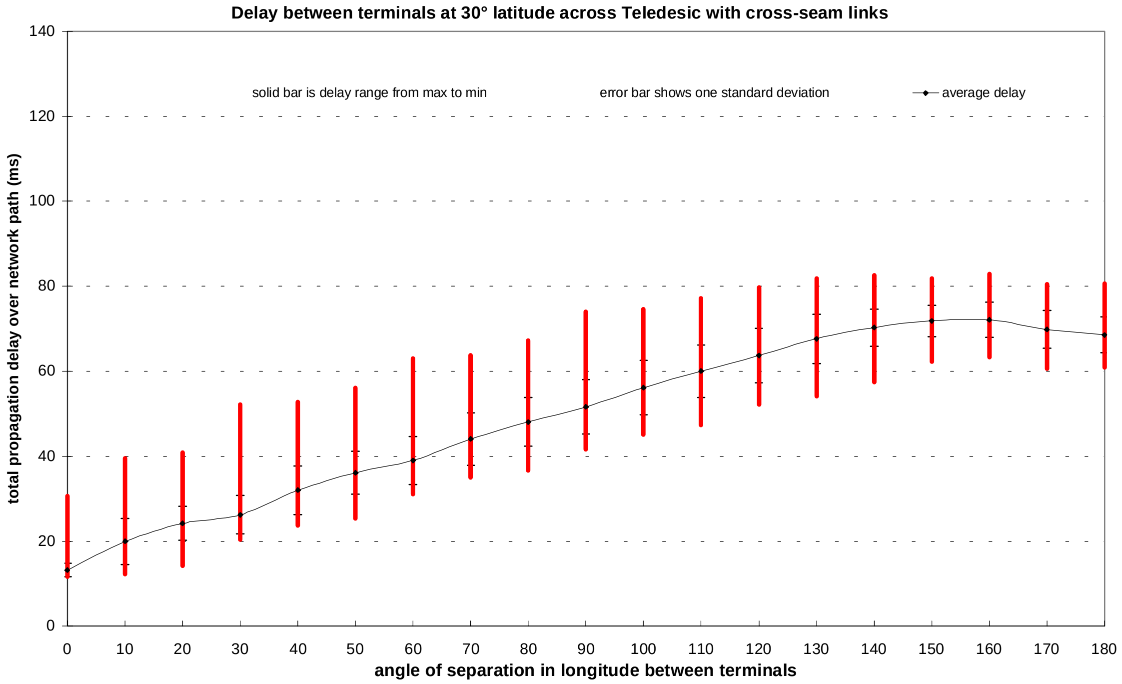
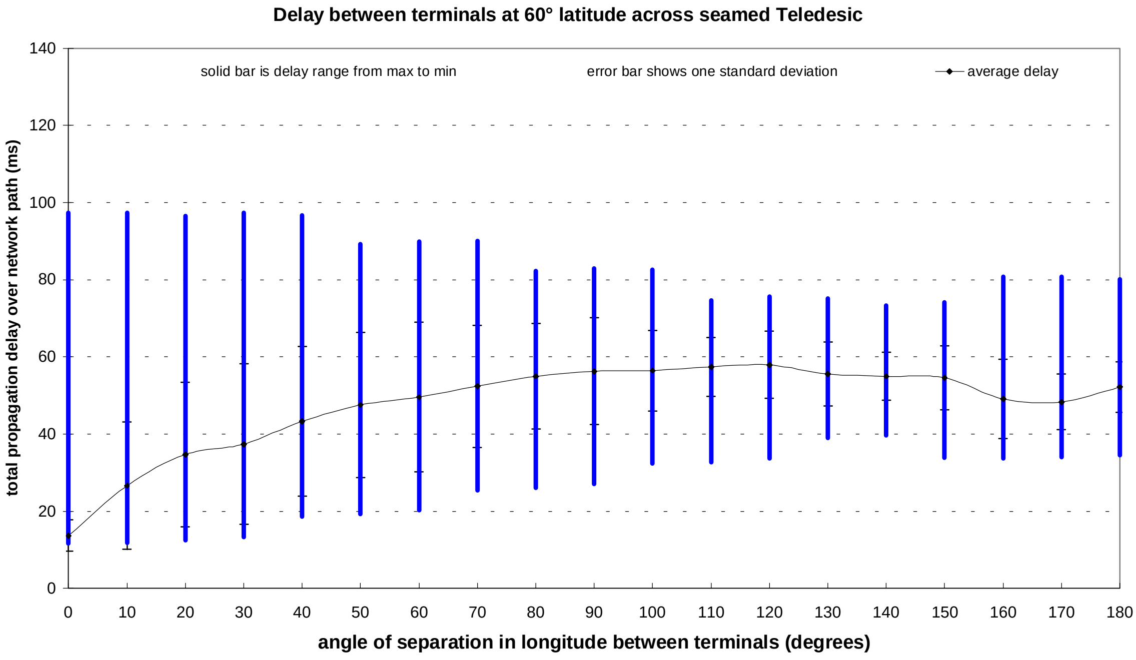


**Figure 2.18 - effect of cross-seam links in *Teledesic* on delay seen at 30° latitude**

A smaller separation between the bars indicates a smaller deviation and smaller spread of delay samples, and thus less overall variability in delay. Variability in delay and delay variation is decreased dramatically for most separation angles by the use of cross-seam links.



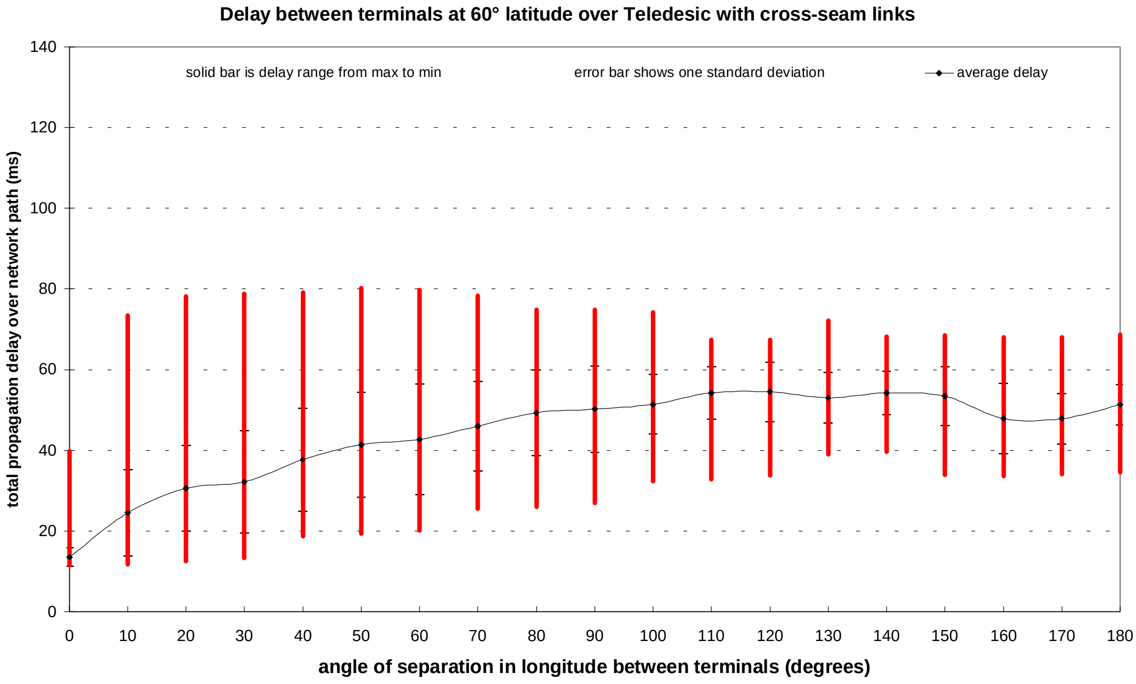


**Figure 2.19 - effect of cross-seam links in *Teledesic* on delay seen at 60° latitude**

Two nearby terminals in the footprints of neighbouring satellites connected by cross-plane ISLs will be separated by the seam only when it intersects the shortest ground distance between the two stations, leading to path rerouting over high latitudes.

Two terminals located closely together in the same footprint will be separated by the seam for the time between their handovers to counter-rotating satellites.

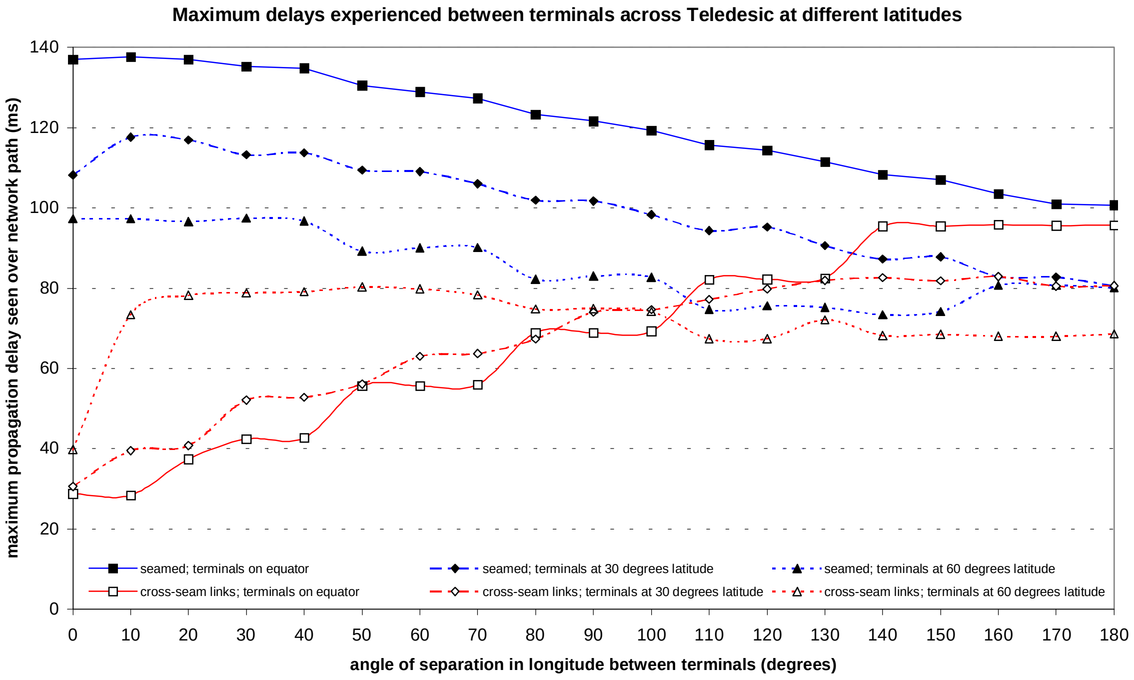
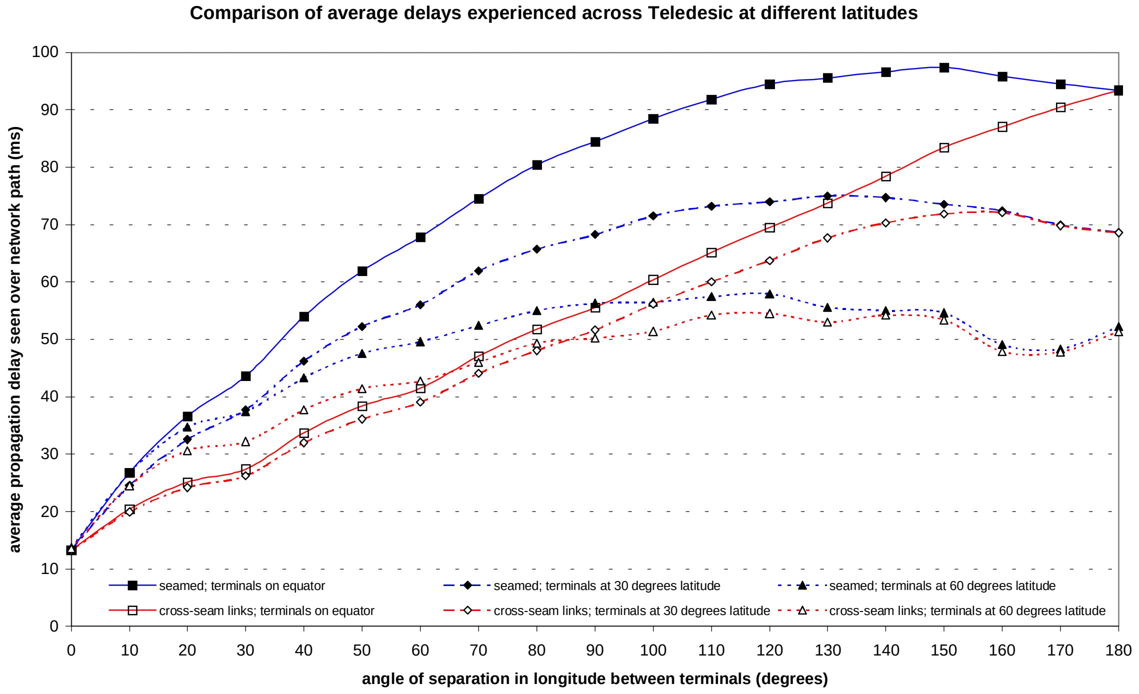


**Figure 2.20 - highest delays experienced between terminals sharing latitudes**



Note that this graph has a smaller y axis to show differences between curves more clearly.

**Figure 2.21 - average delays experienced between terminals sharing latitudes**

When cross-seam links are not available to help maintain path routes, the seam alters the path between terminals from a relatively direct short series of hops, over roughly the same latitude, to a path over the highest latitudes and across much of the satellite network. The seam is forcing rerouting of the path taken by the traffic to occur.

The decrease in delay is particularly noticeable for traffic between terminals at smaller separations, which is most affected by the seam disruption. Figure 2.17 shows the effect of the presence or absence of cross-seam links on traffic between terminals at the Equator. This figure shows a dramatic difference between ‘seamed’ and ‘seamless’ cases, and the large path delay reductions resulting from the introduction of cross-seam ISLs. The differences in delays between these two cases is less for terminals separated at 30° latitude (figure 2.18), and is much smaller for terminals at 60° latitude (figure 2.19), because the changes in path rerouting as the seam disrupts traffic decrease as terminal latitude increases. This benefit decreases for terminals placed nearer to the highest latitudes of the constellation network. At large separations, traffic over a path chosen by selecting a route using links with the smallest total delay metric will always travel via the highest latitudes.

At large longitudinal separations between the terminals, the impact of the use of cross-seam links upon traffic disrupted by the seam is at its least. This is shown most clearly by the ‘seamed’ and ‘seamless’ average- and maximum-delay curves, compared in figures 2.20 and 2.21, approaching each other as the angle of separation in longitude between terminals increases to its maximum spacing of 180°.

Figure 2.20 shows that traffic between terminals at low latitudes benefits most from the introduction of cross-seam links in terms of reducing maximum delay seen. This is clearly illustrated by the probability density functions (pdfs) of the traffic path delays presented in figures 2.22 and 2.23, showing groups of delays related to the number of ISL hops used.

The use of cross-seam links removes the ‘clump’ of large path delays, seen at around 130ms, caused when the seam passes between the communicating terminals, intersecting the shortest path.

It is clear from these results that use of cross-seam links reduces the maximum propagation delay, path lengths, path changes and visible end-to-end disruption of network traffic in star constellation networks with ISLs. Implementing cross-seam links decreases the overall variability of delay experienced by traffic at different times of the day. This is an important consideration for QoS and for the ability of the constellation to offer bounded delay guarantees for real-time multimedia applications.

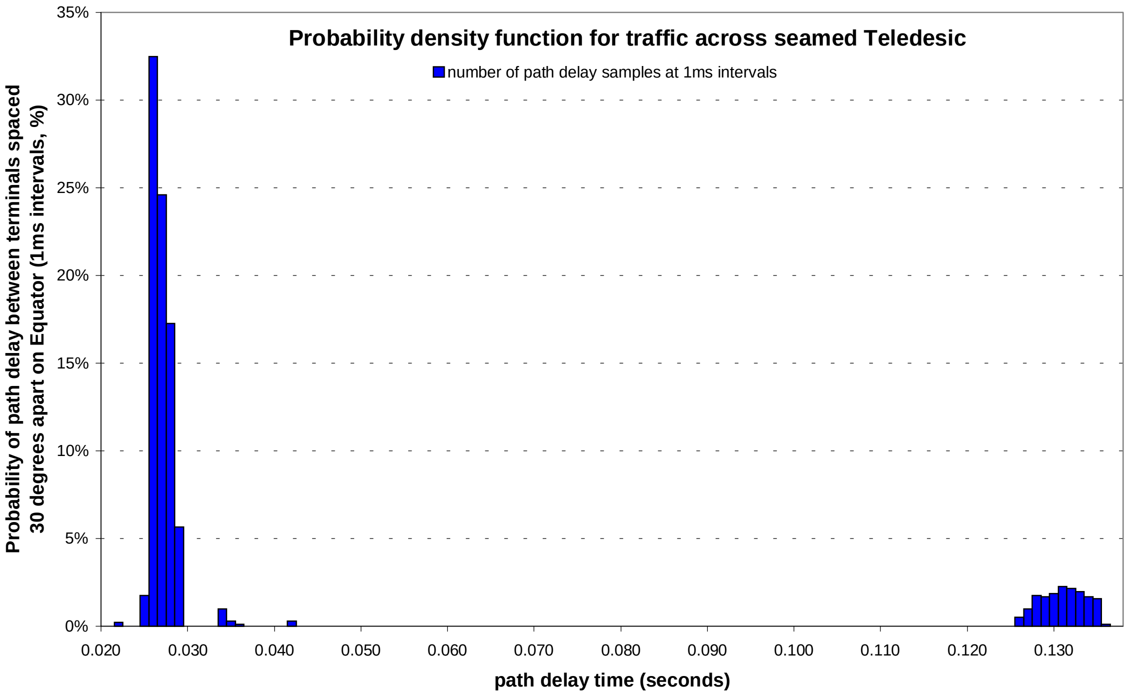


Figure 2.22 - probability density function of path delays across seamed *Teledesic*

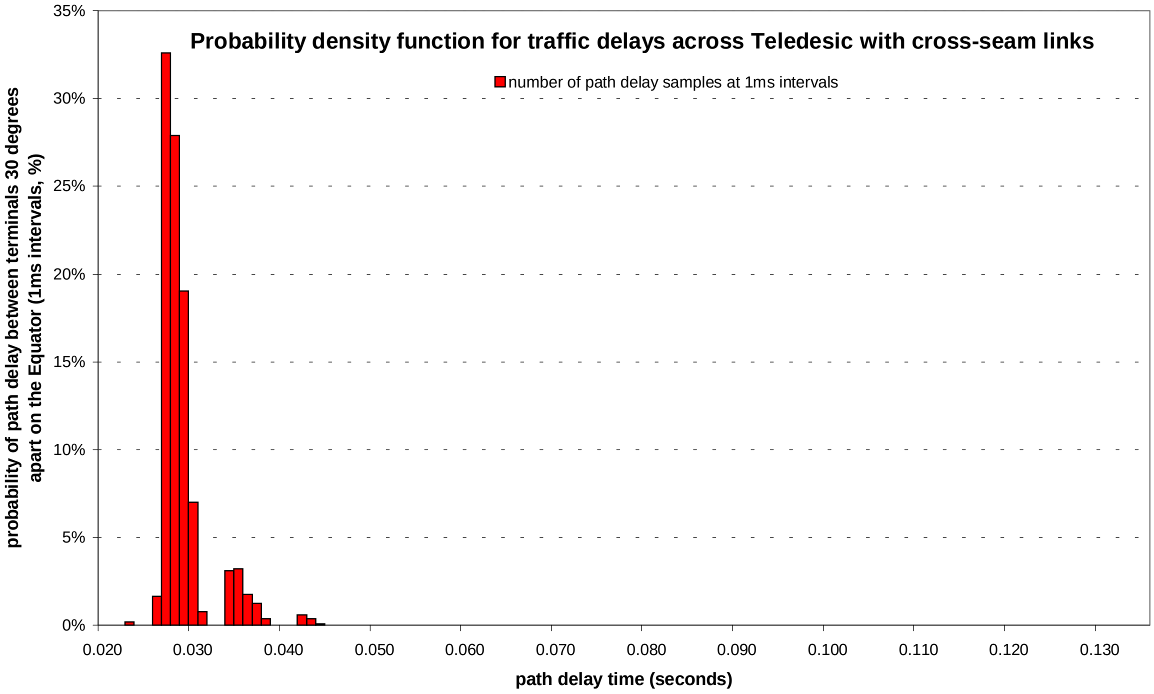


Figure 2.23 - probability density function of *Teledesic* traffic (cross-seam links)

The large sample size (>1000 samples) gives a high degree of confidence in capturing the overall effect of the impact of the seam on network traffic, although it also enabled discovery of interesting transient effects that can now be discussed.

Handover events at terminals, governed by orbital geometry and taking place in the *Teledesic* simulation when the current satellite used by each terminal drops below the minimum mask elevation angle of 40° above the horizon, are simulated and occur simultaneously for the terminals.

However, packets that are already in transit within the constellation network as the handover occurs – those that are on the uplink from one satellite or that are in the former overhead satellite's buffers – must still pass from one side of the seam to the other, once satellite routing tables are updated with the path to take to the new location of the destination terminal.

(The *ns* simulation updates routing tables immediately, with no lag. This assumes a predictive model of the constellation, and neglects routing update propagation.)

These ‘in-transit’ packets thus experience the large path delay resulting from the seam, resulting in a large maximum path propagation delay for zero degrees separation, even though the co-located terminals are always on the same side of the seam.

### 2.3.6 Handover, ISLs and transients

The preceding section discussed the effect of terminal handover on the maximum delay seen by packets sent between co-located terminals, so that the orbital seam can be experienced by packets in transit even though handover was simultaneous.

In fact, the possibility of an altered path propagation delay for packets in transit whenever terminal handover occurs at their destination is a fundamental property of the moving constellation packet network, as these ‘in flight’ packets must travel a slightly different path to reach their destination than previous or subsequent packets.

Due to the larger distances and propagation delays in the satellite network, this effect is greater than in terrestrial wireless networks, and is likely to affect more traffic.

To examine this in detail, high-speed traffic was simulated between terminals. The time between successive packets was less than half of each link propagation delay experienced by the packets. This allowed us to capture and view transients due to ISL use and terminal handover that traffic at slower rates would see rarely, if at all.

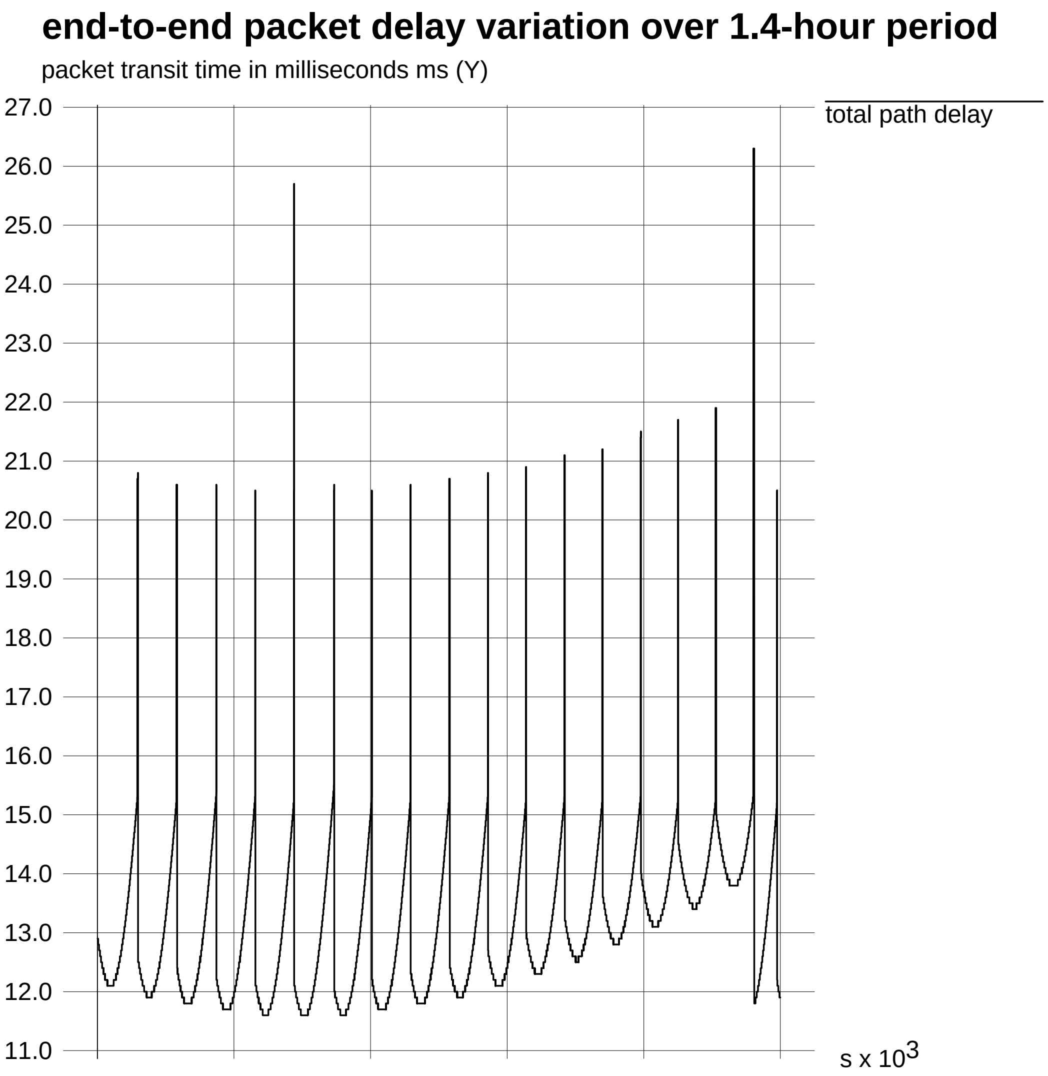


**Figure 2.24 - examination of high-rate traffic over a small timescale for *Teledesic***

Figure 2.24 shows these transients for communications between two terminals, chosen for their close location on the Equator so that the effects of multiple cross-seam or interplane ISL handovers upon the delay along the path were minimised.

The steps visible in the curves in figure 2.24 are due to the resolution of 0.1ms for recorded time within the simulator. (Compare figure 2.24 with figure 6.9 in [**Henderson99**], whose lower traffic rate does not capture these transients.)

Each satellite pass, showing a smooth decrease or increase in packet path delay as the satellite approaches or leaves its local zenith for the terminal, can be clearly seen. Larger path delays for packets in flight during handover between passes are also visible.

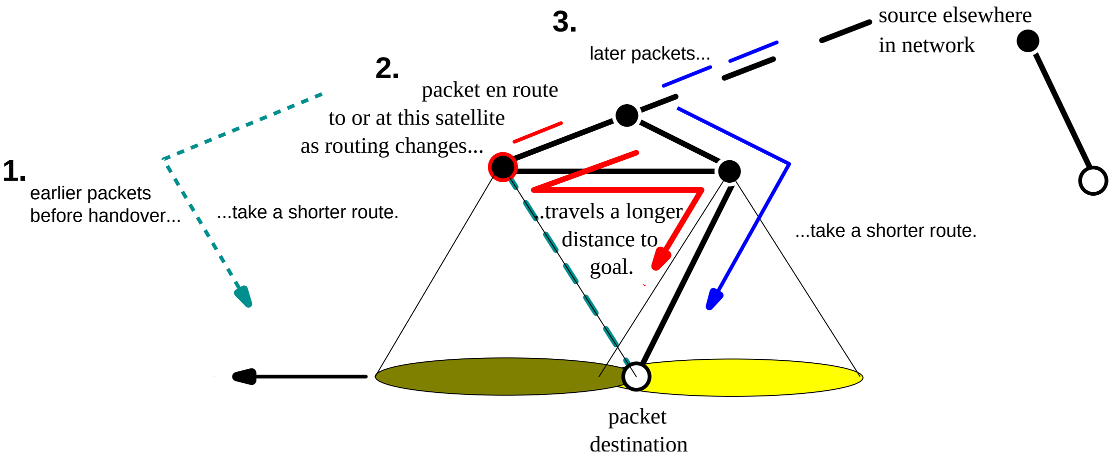


**Figure 2.25 - illustrating the process that leads to a transient spike in path delay**

These longer-path-delay packets are in the last hop in the ISL network before the downlink to their destination, or at the satellite the destination terminal is leaving, as that terminal undergoes handover from satellite to satellite. After reaching the satellite that was being used by the destination terminal before terminal handover took place, these packets must then be routed onward to the current satellite the terminal is now using.

This adds delay before those packets are received. Later packets bypass the former satellite entirely, to the path delay returns to near its previous value. This process is shown in figure 2.25.

The overall rotation of the Earth under the plane of satellites and movement of the destination terminal across that plane, showing a gradual decrease and increase in the minimum times of local satellite zenith, are also visible in figure 2.24 as a great curve that can be drawn tangentially to all satellite pass curves, encompassing them. These gradual movements under each plane are separated by larger step changes when terminal handover to another plane, and another street of coverage takes place. In-flight packets briefly travel two extra ISLs to cross the plane, rather than one, before dropping back an ISL delay as the new satellite in the new plane begins its pass.

(The large transient spike near 14,000s occurs between the two passes where the uplink and downlink satellites come nearest their terminals’ zeniths. The satellites are held onto for the longest period of time before handover occurs at each ground terminal, and the resulting handover is more dramatic as the satellite network has moved the most in its orbits and relative to the surface of the Earth. Since *Teledesic* is a redundantly-connected geodesic mesh, packets can traverse both long and short ISLs, of different delays.)

These transients are also visible in the probability density functions shown in figures 2.22 and 2.23, as the small clumps of samples to the right of and separated from other clumps that represent discrete numbers of ISL hops.

Encountering these transient delay changes by sending a packet just before a handover event occurs could lead to packet reordering, which can adversely impact applications reliant on an ordered flow of packets. (The impact of packet reordering on TCP is discussed further in Chapter 3 and in [**Woodetal01b**].)

If the satellites along the path knew that a handover was about to take place at the destination terminal, it might be possible for them to buffer packets along the path to delay the packets from reaching the last hop before downlink. However, this would impose a lot of per-flow state on the network, and is not practical for the high-rate traffic that is most likely to experience these transients. Handovers cannot always be easily predicted, particularly for mobile terminals experiencing shadowing.

Although low-rate traffic is unlikely to experience these effects, applications sending high-rate and extremely jitter-sensitive traffic can be affected, and the impact of handover on network traffic must be carefully considered in the system design.

## 2.4 Summary

A regular circular satellite constellation lies on the surface of a torus due to the effect of orbital mechanics. The two well-known ‘types’ of constellation – the Walker star and the Ballard rosette – result from this.

A type (1) constellation is a superset of a Walker star constellation and forms a section of the surface of a torus that can be described as a cylinder with two 180°-twists. This gives single network surface coverage of the ground it covers, bounded by the seam at the cut across the torus. This network cut may be crossed with cross-seam intersatellite links, although these have not yet been implemented in practice.

A type (2) constellation is the superset of a rosette and can be thought of as the sum of two type (1) constellations. The type (2) constellation can offer double-surface network coverage of each point on the ground it covers.

This chapter has shown that the network topology of a satellite constellation network with ISLs is a variant on the topology of the well-known Manhattan network, although we were not able to successfully apply and reuse existing results from Manhattan networks for parallel computing to the satellite network.

Delays between ground terminals are subject to continuous satellite motion and more abrupt handover and routing changes. This chapter has demonstrated that path delays for traffic across rosette constellations are more repeatable on a large scale, as the constellation moves, than delays across star constellations. This is due to the presence of the disruption of an orbital seam between counter-rotating planes in the star constellation.

Analysis of these simulations has shown that the impact of the orbital seam on path propagation delay experienced when traversing star constellation networks is decreased with the use of cross-seam links. Cross-seam links reduce the path delay experienced by traffic, particularly between ground stations at low latitudes and with small longitudinal separations. Use of cross-seam links also reduces the overall variation in path propagation delay seen throughout the lifetime of a communication between two terminals.

The technical difficulties of implementing cross-seam links, which will require regular handovers between satellites and high-speed tracking to compensate for the movement of the satellites in counter-rotating planes, appear to be worthwhile for the resulting benefits to network traffic of decreased overall path delay and path delay variation.

This chapter has shown that handover events are a significant feature of LEO and MEO satellite constellation networks, with considerable impact on end-to-end delay experienced by traffic. Handover events affect traffic traversing the ISL mesh as routing changes. By temporarily increasing path delay, handover events can also affect network traffic by reordering packets between source and destination. These events are more likely to affect applications dependent on high-rate traffic, while low-rate traffic is less likely to encounter them. However, the degree of the impact of these events upon traffic depends a great deal upon the network design and implementation.

# *3. TCP and routing in the constellation*

TCP/IP is the general term used for the suite of protocols upon which the Internet depends.

IP, the Internet Protocol, provides a connectionless datagram service that enables packets to be routed to their destinations by inspection of header information. TCP, the Transmission Control Protocol, makes reliable ordered communication of streams of data, such as files, possible by implementing a duplex protocol, based around sliding windows and acknowledgements, over IP. UDP, the User Datagram Protocol, provides an unreliable delivery service using IP for applications to implement their own transport methods to suit their own requirements. Detailed descriptions of these and other, related, protocols can be found in [**Stevens94**].

## 3.1 Congestion control

Network congestion results when buffers in router or switches overflow and packet or cell losses (or ‘drops’) occur. Congestion avoidance, often also known as congestion control, is an important problem for networks that are heavily utilised.

Network congestion cannot easily be controlled as a single variable in a complex network or internetwork, especially when tunnelling takes place. One merely designs a network to try and avoid the set of states that constitute being congested, by having traffic sources control their contribution to the congestion. [**Jacob88**] discusses the difference between congestion *avoidance* and congestion *control*. Avoidance involves the network signalling to the endpoints that congestion is occurring (even e.g. by packet loss) so that the endpoints can decrease traffic input to the network, while control involves allocation of network capacity, generally in a manner fair to sources.

### 3.1.1 In ATM

In the ATM world, congestion avoidance involves congested ATM switches using control mechanisms that discard cells and notify switches forwards or backwards that this has taken place. If backwards notification takes place, the source's switch will be made aware of the congestion, and it can take steps via traffic shaping to alter the flow and ramp down a contributory cause of the congestion. One such closed-loop approach is described in [**KalyJainetal98**].

### 3.1.2 In IP

In the established best-effort IP world, the network does not normally enforce traffic shaping of flows or allocate capacity used to flows. If packets are lost due to congestion there is no explicit feedback from the network to the source to provide notification that congestion has taken place – since the network is congested, there is an argument against generating even more notification traffic to add to the congestion. In fact, packets providing congestion notification may themselves be subject to congestion and lost, making explicit congestion notification (ECN) [**RFC2481**] an imperfectly reliable indicator of congestion even where implemented.

### 3.1.3 With TCP

TCP implements its own end-to-end feedback, via acknowledgements, to ensure a lossless channel and reliable delivery.

Van Jacobson's congestion-avoidance exponential-backoff/linear-increase algorithms [**Jacob88**], additions to the then-standard TCP implementation, make use of this acknowledgement feedback to make assumptions about the state of the network and its congestion, and to alter the rate of transmission accordingly. The acknowledgements effectively form a feedback control loop with the congestion control algorithms as strong dampers in that loop [**RFC2581**]. The best-effort IP network does not explicitly inform the TCP senders at each end of the TCP connection of its congestion; the senders just happen to infer network congestion from the observed behaviour of their end-to-end channel, and adapt their throughput to the congestion perceived.

For TCP/IP, loss of packets containing TCP segments or acknowledgements results in the TCP sender not receiving acknowledgements for packets sent. The TCP implementation in the sender will notice this, will assume that the cause of this is congestion in the network, and will then exponentially decrease (or 'back off') the rate at which it sends new data segments. This assumption can lead to loss of throughput on less-reliable links such as errored satellite links, where packets are lost due to errors in the channel on the satellite link, rather than due to congestion at terminals.

### 3.1.4 With UDP

In being able to make use of the minimal information available on the state of the internetwork via feedback provided by acknowledgements, TCP communication is a useful subset of all IP packet communication, providing the abstraction of reliable delivery to applications. By contrast, individual UDP packets are one-way and delivery is unreliable, and UDP does not use end-to-end feedback to alter its transmission rate. This means that UDP-based applications, including multicast applications, must implement their own congestion avoidance routines and ensure that they receive data with any necessary degree of reliability. Interaction with TCP traffic is important, and recent emphasis has been placed on ‘TCP-friendly’ congestion algorithms for use with UDP. This involves algorithms limiting their delivery rates to a limit of some constant over the square root of the packet loss rate, so as not to exceed the delivery rate of equivalent TCP flows [**SisaSchulz98**].

### 3.1.5 For IP multicast

Not being based on the single sender/receiver model used in e.g. TCP, UDP-based IP multicast does not have straightforward acknowledgements-based feedback on congestion. The feedback loop is still open and uncontrolled. Work on reliable multicast applications providing feedback, outside the scope of this thesis, is still an active research area, and strategies must be developed to avoid the *acknowledgement implosion* problem of all receivers acknowledging receipt of a packet from a single source, overwhelming and saturating it with traffic [**Floydetal97**]. ‘TCP-friendly’ congestion algorithms, limiting packet delivery rate to a constant over the root of the packet loss, are also being explored here [**ViciCrowRizzo98**].

## 3.2 A brief history of TCP/IP over satellite

The TCP/IP suite has been carried transparently over satellite ever since experiments were first conducted with SATNET in the 1970s [**RFC829, Seo88**], and TCP/IP implementations have been shown to work well over satellite links. TCP/IP can be tunnelled over satellite, as experiments with TCP/IP over ATM on the NASA ACTS satellite have successfully demonstrated [**Bajajetal96**].

### 3.2.1 GEO delay and TCP implementations

Although TCP has been carried over satellite successfully for some time, the long propagation delay to satellites in geostationary earth orbit (GEO) has imposed limitations on interactive applications and on existing TCP *implementations*, which are affected by the large latency in satellite applications. The *(latency*bandwidth)* product used for dimensioning pipes and buffers for unacknowledged data can scale beyond buffer limits in existing implementations for high-capacity channels and for the high (circa 250ms earth-station-to-earth-station, or circa 500ms round-trip to send and acknowledge) latencies found in geostationary-satellite-based applications.

(A buffer using a 16-bit memory pointer gives you access to 64 kilobytes, or KB, of memory, placing an upper limit of 64KB for unacknowledged data in flight. A half-second round-trip time or RTT leads to an upper limit of 1Mbps of application data. Early TCP window buffer sizes were typically 4KB, which would lead to an upper limit of 64kbps in the same scenario. In practice, throughput will be less than the limit on buffer size suggests, due to the need for some free memory in the buffer for churn.)

This can result in less than optimal throughput over the satellite channel, as there is only so much space to hold data that is unacknowledged. The full capacity of the satellite channel cannot be used easily by a *single* TCP connection for a single high-bandwidth application, given small buffer spaces at the sender and receiver. This limits the use of satellites for seamless internetworking. There is, however, no limit to the number of concurrent TCP/IP flows that can be multiplexed together to fill the satellite channel, or pipe.

### 3.2.2 Addressing the delay limitation for implementations

Workarounds for the limitations on a single large TCP connection have been suggested to make better use of existing satellite channels by having the application open multiple parallel connections to use multiple small buffers instead of a single large buffer to increase throughput [**AllmanOstKruse95**, **AllmanOstKruse96**]. However, rewriting the application to be more aggressive in demanding network capacity to compensate for shortcomings in the implementation of the lower layers is a short-term fix and undesirable; the longer-term and more preferable solution is to improve the lower-layer implementations.

Work on large windows suitable for Long, Fat Networks (LFNs, pronounced "elephants") [**RFC1323**] and selective acknowledgements [**RFC2018**] has been designed to overcome these implementation problems with paths that exhibit high bandwidth-delay products, such as links over geostationary satellites.

## 3.3 TCP, congestion and errors

TCP treats all losses on the journey between source and destination as an indication of congestion, or overflowing packet buffers in router queues leading to discards. TCP creates its own imperfect feedback on the state of its communication path and thus the internetwork the path traverses, as a result of its acknowledgements. The performance of an end-to-end TCP connection does not equate to the performance of any link that the TCP connection is carried across. TCP cannot be aware of the condition of an individual subnetwork or link.

To illustrate this, consider an end-to-end TCP connection across a number of joined networks, one of which is an ATM network where an available bit-rate (ABR) virtual channel is used to tunnel the TCP packets across the ATM network.

ABR is generally intended to be closed-loop, where the network communicates its congestion state back to the endpoints so that they can adjust their output [**KalyJainetal98**]. This contrasts with the more open-loop approach taken by TCP, where TCP infers network conditions, rather than being told them explicitly.

If the ATM network is congested, there is no way that ABR with explicit rate (ER) feedback can tell the TCP endpoints how much they need to decrease their output, and ABR may then make decisions about allocation which are inappropriate for the needs of the TCP senders.

The TCP endpoints do not react directly to the congestion in the ATM network, and gain a less accurate picture of congestion in that network as a result. Here, the different assumptions used in tunnelling and tunnelled protocols have acted to break any direct feedback loop from congestion through losses to TCP detection.

That is one example demonstrating that the information about the internetwork inferred from TCP’s acknowledgements feedback loop is not indicative of the state of networks through which TCP/IP traffic is tunnelled. As many (often ATM-based) satellite networks will be tunnelling IP, implementations of TCP’s congestion-avoidance mechanism at the customer endpoints may well be of little use to the satellite network operators. If the tunnels are guaranteed channels, then the user will expect to get the full channel capacity that is being paid for. This explains the satellite operator’s traditional emphasis on getting the most from fixed channels, and concern over TCP’s congestion-control reaction to packet losses caused by link errors, rather than by congestion.

A number of intertwined algorithms determine the rate at which TCP puts new data into the network, and how losses are detected and handled. In particular, a TCP sender maintains a *congestion window* variable that determines how large the size of its sliding window of unacknowledged data given to the network can be at any point in time. This *cwnd* variable is increased over time from start, or after congestion losses, as the TCP sender probes the network and discovers that new data can be sent with increasing frequency. It is decreased when the sender infers losses due to congestion. Slow-start and other algorithms control the congestion window and other TCP settings. These are described in detail in [**RFC2581**].

### 3.3.1 Errors in the satellite environment

Communication between terminals using a geostationary satellite has a comparatively long round-trip time (RTT) from sender to receiver and back of the order of half a second. This means that considerable time is needed in slow-start, on opening a new connection for TCP, to increase its throughput to what the link capacity will bear. Any losses due to errors in transmission on a less-than-perfect satellite link, rather than to congestion, cause TCP to reduce its overall sending rate to help alleviate perceived congestion. After an errored packet is inferred as lost due to congestion, and the congestion window is reduced, the return to the previous high rate of throughput is slowed by the large RTT.

Current TCP congestion control algorithms assume that packets that are lost due to satellite channel errors are lost as a result of internetwork congestion. The errors in the satellite channel can be bursty as a result of convolutional coding; interleaving and an outer Reed-Solomon code can randomise and spread the occurrence of errors, but any errors will still affect higher layers. This leads to sub-optimal use of available satellite link capacity after an error where frame loss leads to packet loss: congestion avoidance algorithms cause the TCP sender to decrease its sending rate dramatically, followed by a slow return to the previous transmission rate. This can waste expensive satellite channel capacity in the interim.

This really results from a lack of reliable explicit congestion notification [**RFC2481**], or some roughly-equivalent form of explicit corruption notification, to distinguish between network congestion and link errors. Tweaking congestion control algorithms to improve performance slightly in the satellite environment cannot completely compensate for this lack of information on the real cause of packet loss.

In both TCP slow-start and when recovering from errors, increasing the congestion window takes a number of slow half-second roundtrips, and the capacity of an expensive satellite link will not be fully used as a result. Satellite considerations for TCP are detailed in [**PartridgeShepard97**] and [**RFC2488**].

TCP’s process of treating error recovery as congestion and using timeouts has been improved by the later introduction of fast retransmit and recovery. These permit less backoff in cases where communication between sender and receiver is still clearly taking place [**RFC2581**].

### 3.3.2 Coding on the satellite link

Careful selection of physical- and data-link error-correction suitable for the space environment can minimise the errors on a link, in packets and thus seen by TCP. This can minimise the effects of errors on congestion control algorithms, without requiring changes in the TCP implementations. Improvements in coding on satellite links have altered link error characteristics to be in one of two states from a packet-level viewpoint: either error-free or extremely errored. With a correctly-dimensioned purpose-designed satellite link, making possible the low resulting error rate that TCP expects, it can be assumed that the satellite link is as reliable as any terrestrial link. This assumption is extremely useful for simulation, as it allows concentration on effects of interest, without confusing the results for the added effects on TCP resulting from packet discards or transmission errors leading to packet loss.

## 3.4 Other TCP considerations

End-to-end TCP/IP can be expected to work as is over these new satellite constellations for multiplexing of ‘thin’ TCP connections using existing implementations when data transfer rates are low, allowing communication between two earth stations via a space-based wireless route using the constellation. Implementation of the standards discussed above allows ‘fatter’ higher-capacity TCP connections and easy use of higher bandwidths by scaling TCP upwards.

### 3.4.1 Asymmetric connections

TCP may not handle connections across links with highly asymmetric capacities well, due to the backtraffic that is necessary for acknowledging that sent data has been transmitted successfully. However, lack of capacity for acknowledgements only becomes a problem when the forward sending channel capacity to backchannel acknowledgements capacity ratio exceeds 50:1 [**DurstMillTrav96**]. Although this limitation concerns asymmetric solutions, e.g. DirecPC and similar systems [**Falk94**], this limitation is unlikely to be relevant to broadband satellite constellation networks, where TCP connections can be expected to exist multiplexed within higher-capacity satellite channels and are unlikely to experience such a capacity constraint.

### 3.4.2 Variants and split TCP

The requirements and constraints of the space environment and the desire to utilise satellite capacity efficiently has also led to the development of a number of optimised-for-space-environment TCP variants with altered transmission control behaviour, e.g. SCPS-TP, the Transport Protocol that is part of the Space Communications Protocol Standards (SCPS) protocol suite [**SCPS99**] described in [**CompRam97**]. These variants run between ground terminals across the satellite link, while other TCP implementations complete the end-to-end communication across the terrestrial Internet. The end-to-end TCP connection across the congested but relatively link-error-free terrestrial Internet and the less-congested but more error-prone satellite link is *split* into individual TCP connections tailored for each environment; easy to do with an HTTP proxy cache, but less elegant in implementation when other applications are concerned. True internetworking requires end-to-end TCP, rather than split TCP.

## 3.5 TCP across multiple paths

A redundantly-connected *distributed* multipath ISL topology, as defined in [**Baran64**], allows a change in behaviour when congestion occurs – rather than routers on a congested path dropping packets, those routers can instead divert those packets to another, less congested, path as discussed for the Manhattan network in section 2.2.2. This introduces load balancing. Multiple routes between source and destination also allow load splitting, or sharing of traffic load across different paths at the same time. These are considered useful for military satellite networks where redundancy and reliability are important, and a multiple-path routing algorithm to enable these is described in [**Cainetal87**].

### 3.5.1 Path choices in the constellation

In fixed fibre terrestrial networks, there is often only one fixed path between sender and receiver, since building redundant paths adds to the cost of the network. MEO and LEO ISL meshes, such as the *Spaceway NGSO* and *Teledesic* networks, are able to offer more than one possible path of ISLs for packets to take between the two satellites communicating with the source and destination ground terminals.

When the path is longer than one ISL hop, there are likely to be a number of equivalent paths, of the same number of hops between satellites, whose queuing and framing delays will be roughly equivalent. If a ground terminal has the opportunity to communicate with more than one satellite visible to it above its local horizon (using ‘diversity’, as discussed in Chapter 6), the number of possible paths between source and destination terminals can be increased still further.

A complex scheme involving using measuring total propagation and queuing delays on these multiple paths and using that information to select a ‘best’ path for each packet has been developed for Teledesic LLC [**Liron00**].

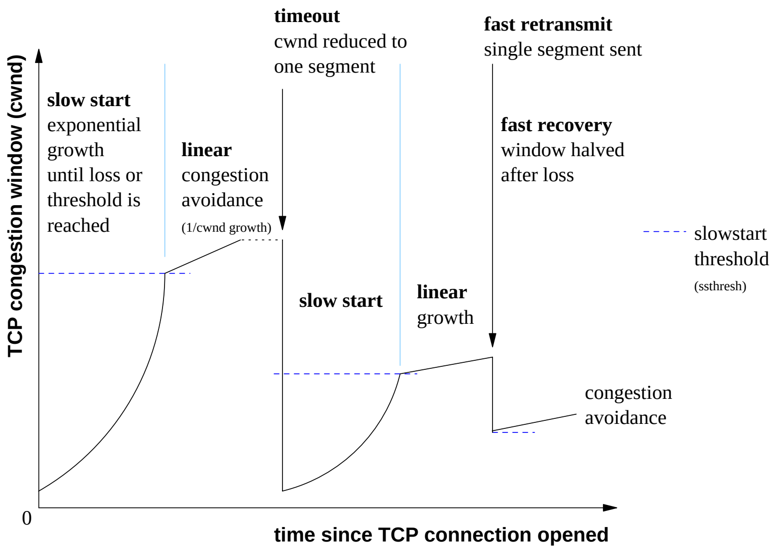


**Figure 3.1 - illustration of changes in TCP congestion window**

The inherently multipath topologies of LEO and MEO satellite constellations with ISLs allow the opportunity to examine TCP's behaviour when exposed to multipath routing in the satellite environment. It was found that two of TCP's algorithms in particular affect the performance of TCP over multiple paths: *fast retransmit and recovery*, shown in figure 3.1, and *delayed acknowledgements*.

### 3.5.2 The effects of fast retransmit and recovery

A TCP sender can infer congestion in the network causing packet loss from:

- A *timeout*, where no new acknowledgements are received for a calculated period of time, typically a multiple of half a second. The sender believes that all traffic sent to the receiver is being lost due to congestion, and attempts to avoid contributing to this perceived congestion by reducing its congestion window to one segment. It then probes the network with an exponentially-increasing flow of packets in slow-start, much as it did when first sending data when the connection was opened.

- Receipt of a number of duplicate acknowledgements or *dupacks*, where the receiver sends an acknowledgement that repeats the current position of the left edge of the TCP window. The fact that the receiver is issuing dupacks indicates that it is still receiving traffic from the sender. However, it is not receiving the data necessary to be able to deliver previous data received to its application and

> move the left edge of the window along, allowing the sender to inject more new data into the network. A loss of data just after the window position that is indicated repeatedly in the dupacks can therefore be inferred by the sender, which can do a *fast retransmission* of the lost packet. The sender assumes that the loss is due to congestion in the network and reduces its congestion window to take account of this. As the received dupacks indicate that data is still getting through to the receiver, the sender can be less conservative in reducing its window to avoid congestion than when encountering a timeout. The congestion window, controlling the rate at which new data is sent, is halved before growing slowly to its previous size as new data is acknowledged. This is *fast recovery*.

Receipt of three dupacks, after the original ack of the last in-sequence packet received, is taken to indicate that a packet has been lost in the network. Three is a reasonable limit for an ordered flow of packets between sender and receiver, since it gives the fast warning of congestion-induced losses that the retransmit and recovery processes were named for, while tolerating minor amounts of packet reordering.

However, if more than one possible path between source and destination can be used simultaneously, packets in TCP flows may be received out of order due to slight differences in latency between the paths. This will cause dupacks to be issued even when no losses have taken place. Any resulting fast retransmission and fast recovery will be entirely unnecessary and detrimental to TCP’s performance. In a mesh of ISLs between satellites, where congestion leads to dynamic rerouting on a per-packet basis rather than to straightforward discards, we can expect to see dupacks from reordering.

### 3.5.3 Dupacks in a multipath routing environment

To examine the effect of multiple paths in the constellation networks on dupacks and the fast recovery process, the network simulator *ns* [**FallVaradhan00**] was used to simulate multipath routing of traffic between Quito and Tokyo. Traffic was sent across all available routes with the same number of wireless hops, using the paths shown in the MEO and LEO topologies presented earlier in figure 2.11. Approximations of the *Teledesic* and *Spaceway NGSO* proposals were simulated, based on descriptions given in their applications to the US FCC for frequency allocation.

Shortest-path routing, based upon the path with the smallest propagation delay, was compared with arbitrary multipath routing – selecting any equal-hop path to the destination at a point in time. This was intended to approximate the behaviour of temporarily-congested satellites diverting newly-arrived packets from their current path onto the alternate path, or of simple ‘hot-potato’ local routing decisions.

The choice of path at each satellite was determined by a simple round-robin selection, updated on the arrival of each new packet. This made each multipath simulation deterministic and repeatable.

Since the aim was to examine only the effect of path diversion on TCP without confusing its effects with the effects of congestion, it was again presumed that the links were both error- and congestion-free. Any difference in propagation delays of the links would result in packet reordering as the packets took different paths to the receiver.

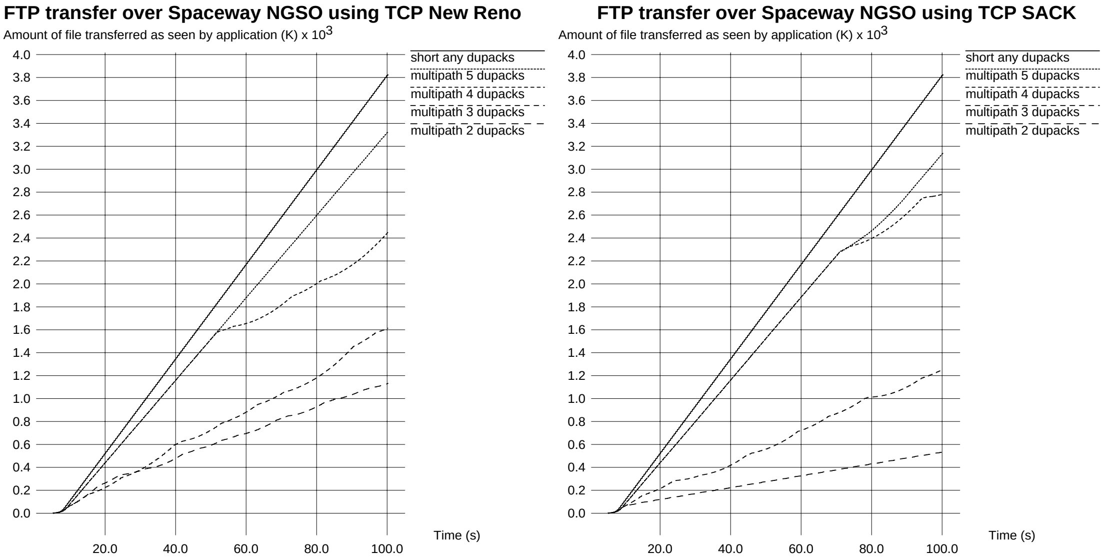
The simulations used high-capacity channels – 5Mb/s reverse and forward on the uplink and downlink and 10Mb/s ISL links – so that the time needed to receive each 1000-byte packet was small compared to the propagation delay at each wireless hop. TCP connections had small receiver windows, to simulate multiplexing of many small TCP connections rather than a single TCP connection filling the path to capacity.

A bulk FTP transfer of a large file was simulated over a period of a hundred seconds: a time that was short enough that terminal-satellite handovers did not occur in these simulations. The comparatively low rate of data transfer (governed by a small receiver window) also allowed us to ignore the handover transients discussed previously in chapter 2. Abstractions of two popular variations of the TCP algorithms that have been widely implemented were simulated in the terminals:

- New Reno, based on a stack implementation originally developed on a machine named Reno, and widely deployed since with some modifications [**RFC2582**],
- Selective Acknowledgements (SACK), an enhancement to the Reno algorithms that adds information in the option field of acks sent from receiver to sender, allowing the sender to retransmit specific data indicated as unreceived [**RFC2018**].

The goodput performance graphs in figures 3.2 and 3.3 show the effects of fast retransmit and recovery on overall TCP throughput as shown by data delivered to the receiving application. Dupacks received due to out-of-order reception of both segments and acks over the multiple paths across the ISL meshes lead to fast recovery. In each case, TCP's throughput is degraded by use of multiple paths, even though there are no losses from transmission errors or congestion in the network.

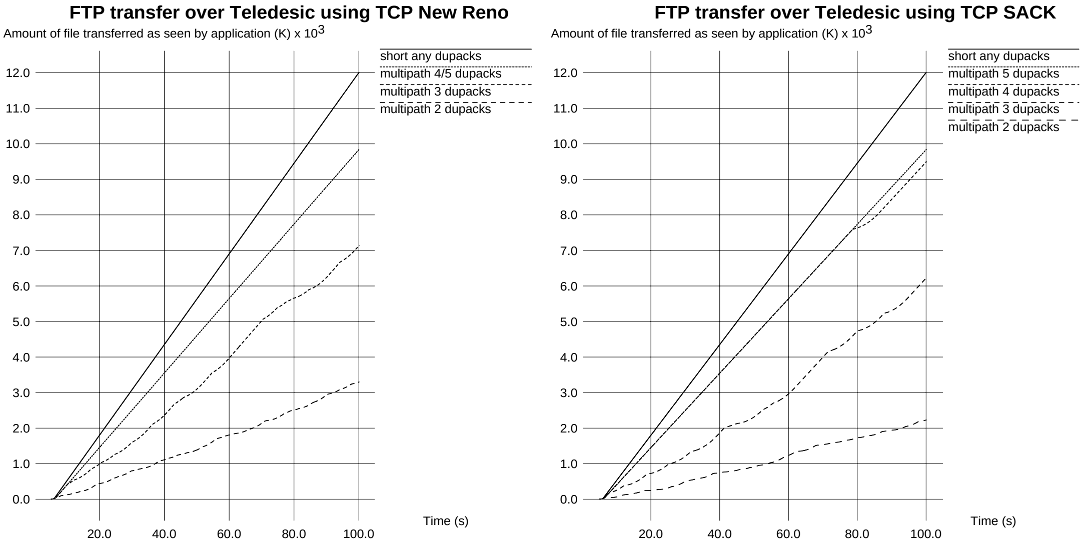
By increasing the arbitrary requirement of three dupacks before entering fast retransmit and recovery to a higher threshold, the throughput of TCP over multiple paths was gradually improved to approaching that of the throughput over the single shortest path.



**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo using *Spaceway NGSO*

**Figure 3.2 - effect of dupack threshold on file transfer over multiple MEO paths**



**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo using *Teledesic*.

**Figure 3.3 - effect of dupack threshold on file transfer over multiple LEO paths**

Similarly, lowering the dupack threshold decreased even further TCP's tolerance of misordering caused by multiple paths. Resulting throughput was lower.

The mesh topologies of these satellite networks provide multiple paths between points and are clearly multipath. However, packet reordering has been observed in the terrestrial Internet due to parallelism or load balancing [**Bennettetal99**], and can be considered a natural state of affairs rather than something to be avoided at all costs. Reordering from two parallel satellite links with varying loads is discussed in [**Seo88**].

SACK's performance when experiencing dupacks can be improved by using the ack options field to indicate what is causing each dupack to be generated, rather than just sending a dupack without extra information [**RFC2883**]. The TCP sender could use this information to infer multipath environments and to alter its behaviour dynamically to suit by varying its dupack threshold as conditions dictate.

### 3.5.4 The effects of delayed acknowledgements

In most TCP implementations, segment-bearing packets may not be acknowledged immediately when received as one might expect. Instead a nominally optional, but widespread, *delayed acknowledgements* mechanism is used. This allows the receiver to skip acknowledging TCP segments before issuing a cumulative ack covering all the segments received since the last ack was sent. This is shown in figure 3.4.

The receiver should acknowledge every second in-order TCP segment received, and should wait between 0.1 and 0.5s for new packets containing TCP segments to arrive before issuing an ack. This also allows acks to be piggybacked on any data segments sent by the receiver (as TCP is duplex), reducing overall network traffic. In practice, it is commonplace to wait for a second in-order segment before sending an ack.

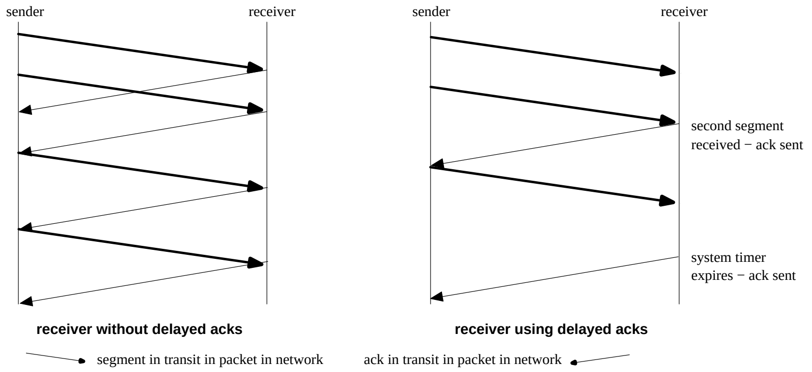


**Figure 3.4 - ladder diagrams showing TCP with and without delacks**

The receiver can wait until a system timer that goes off every 200ms, when an ack must be sent. (We included this common 200ms delay when simulating the effect of the dupack threshold on multipath throughput, as seen in the previous figures.)

Only in-order segments are subject to this delay. Out-of-order segments are generally acknowledged immediately, on the principle that this dupack information is useful to the sender in determining what to resend in the case of losses. The implementation of and effects of delayed acks are discussed in detail in [**Allman98**].

Delayed acks have the advantages of conserving processing resources at the receiver, and decreasing backtraffic and the number of packets and resulting load along the return path to the sender. This is useful for networks with asymmetric paths, such as DirecPC, where the downlink can be around 400kbps via broadcast from geostationary satellite, but the ‘uplink’ is a terrestrial dialup modem at 56kbps or less.

However, delayed acks are widely recognized as degrading TCP throughput in some situations. Since TCP slow-start at the beginning of a transfer uses the number of acknowledgements received as an indication of how much new traffic can be injected into the network, the initial slow-start phase is slowed further by the decreased number of acks sent by a delayed-ack receiver.

This is shown in figure 3.5, where an increased receiver ack delay leads to a slower slow-start. After slow-start the rate of goodput or useful throughput (the overall gradient of the curve) is identical across all receivers, regardless of the delayed-ack implementation, when there is no congestion or loss.

Delayed acknowledgements slow growth of the congestion window similarly after a timeout or in fast recovery. From the viewpoint of Internet traffic as a whole, this can be considered beneficial, as window growth is damped and individual TCP flows using delayed acknowledgements are less aggressive in gaining network capacity.

It is this damping which adversely affects TCP’s performance due to out-of-order segments received over multiple paths leading to dupacks and fast recovery. Receivers with and without delayed acknowledgements behave identically when receiving out-of-order segments. They issue acks immediately so that the sender will be able to make the same decision whether to do a fast retransmit after receiving three dupacks and enter fast recovery.

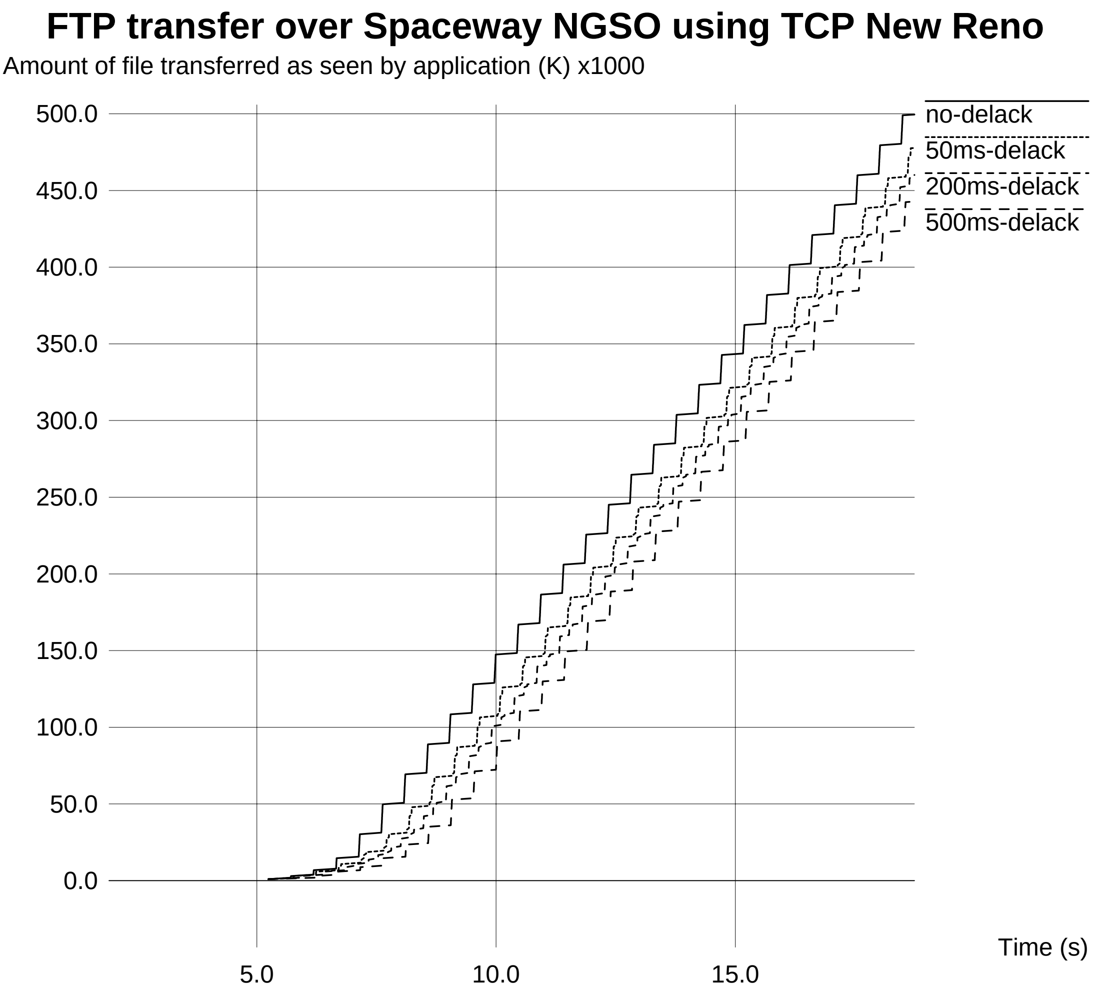


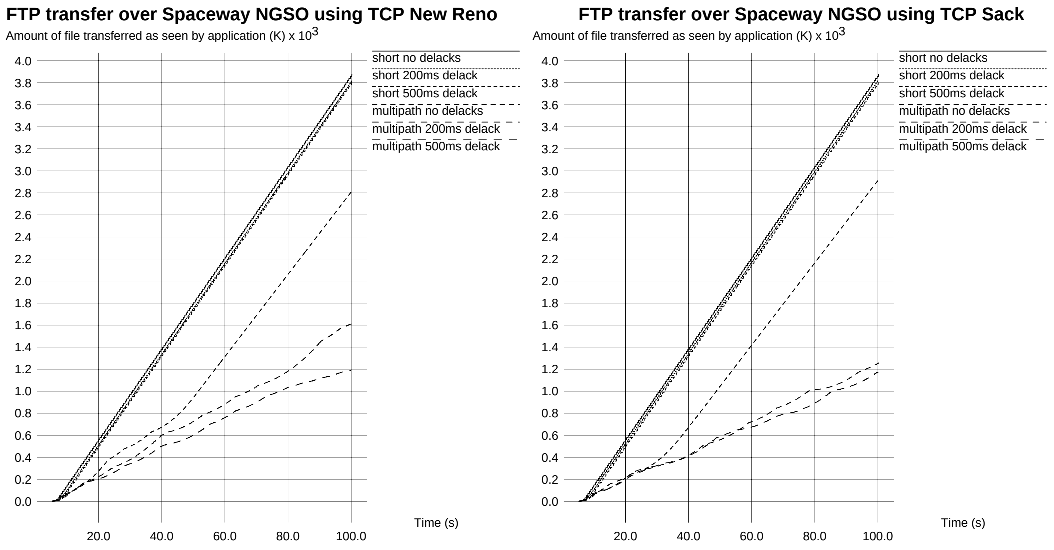
**Figure 3.5 - increased ack delay degrades performance at slow-start**

Once in fast recovery, growth of the congestion window is slowed because the sender receives fewer, later, acknowledgements to in-order packets. If entering fast recovery becomes a regular occurrence, as we have seen happen in multipath environments, then the delayed growth of the congestion window constrains overall throughput further, and considerably more time is spent in fast recovery with a low congestion window setting.

### 3.5.5 Delacks in a multipath routing environment

Figures 3.6 and 3.7, from the *ns* simulations as described earlier, show that throughput in multipath environments is decreased by receiver ack delays.

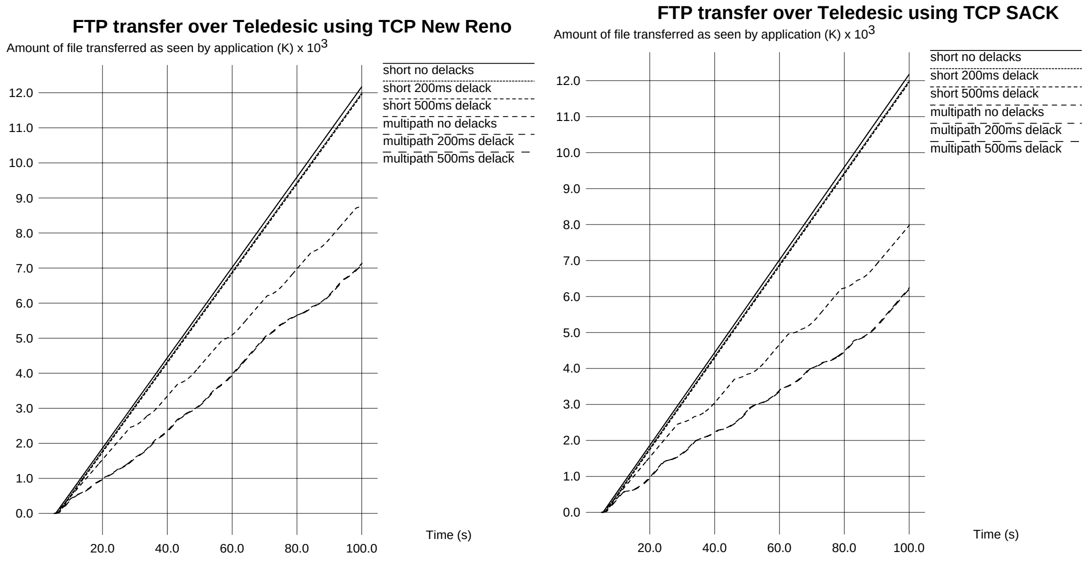
The effect on shortest-path throughput, due to the change in slow start when the connection is opened, is comparatively minimal.



**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo using *Spaceway NGSO*

**Figure 3.6 - delayed acks degrading file transfer over multiple MEO paths**



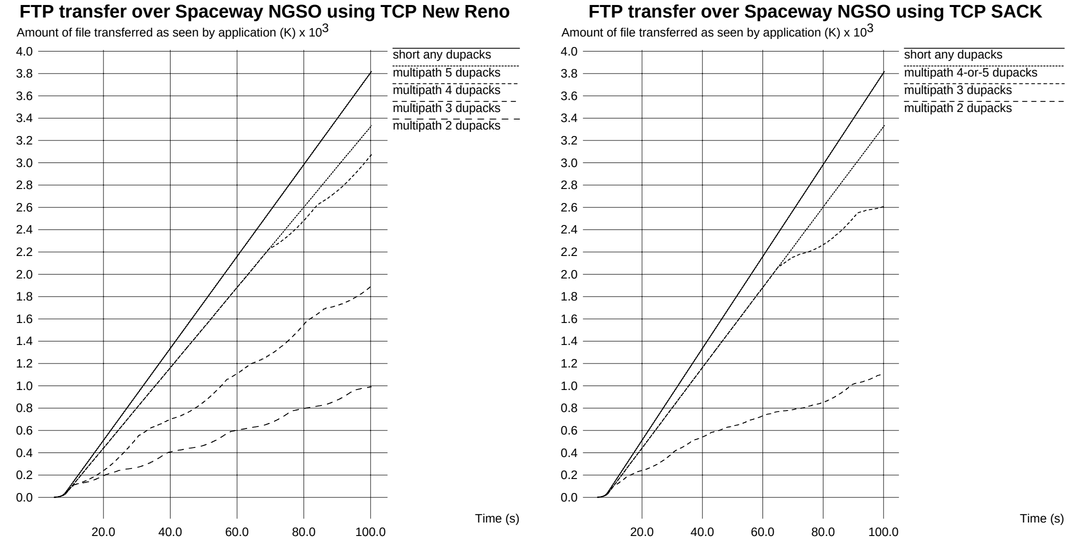
**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo using *Teledesic*

**Figure 3.7 - delayed acks degrading file transfer over multiple LEO paths**

Delayed acknowledgements contribute to the degradation of TCP's performance in multipath environments. This multipath degradation due to delacks could be reduced by selectively avoiding use of delayed acknowledgements when the TCP sender is attempting to grow its congestion window.

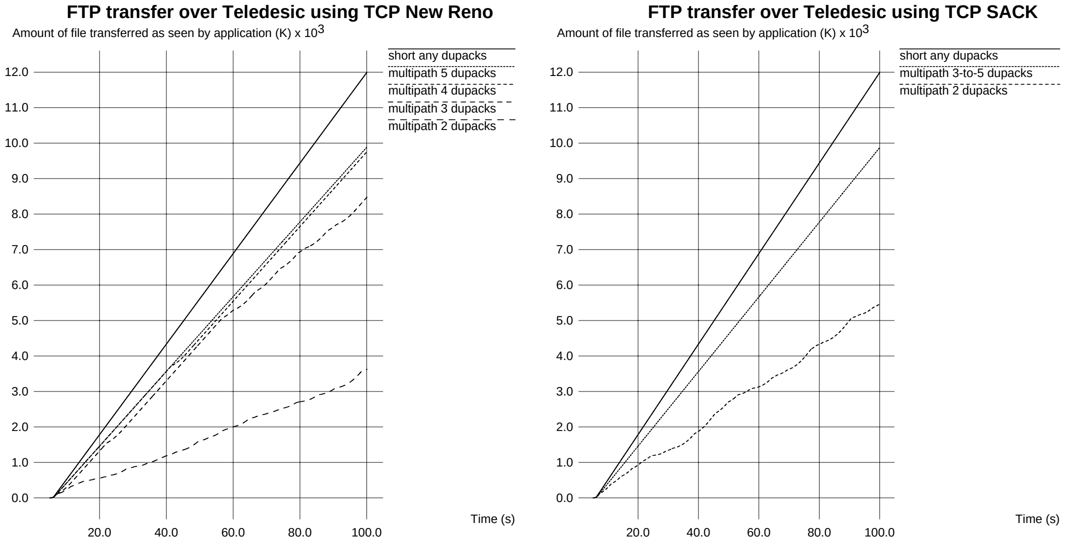
The sender would need to provide additional information to the receiver to make this possible. Such a change has already been suggested in [**Allman98**] for the slow-start algorithm, and would benefit TCP traffic over geostationary satellites by speeding up the initial startup and post-timeout phases. However, the impact of such changes on the characteristics of Internet traffic as a whole is unknown.



**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo using *Spaceway NGSO*
delacks only take place on packets received at the right edge of the window (RFC2581 SHOULD)

**Figure 3.8 - dupack threshold affecting file transfer over MEO obeying RFC2581**



**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo using *Teledesic*.
delacks only take place on packets received at the right edge of the window (RFC2581 SHOULD)

**Figure 3.9 - dupack threshold affecting file transfer over LEO obeying RFC2581**

### 3.5.6 RFC specification of delack handling

Delayed-ack use has recently been tightened by a change in specification so that only acks acknowledging the receipt of *new* in-order packets received at the right end of the window should be delayed, rather than also delaying ack responses to a series of in-sequence segments earlier in the window.

Acks to segments 'filling-in' holes in window data should be sent immediately [**RFC2001** section 3, **RFC2581** end of section 4.2].

As a *SHOULD*, this is optional in implementation, although recommended. This goes part of the way towards avoiding use of delayed acknowledgements when the TCP sender is attempting to grow its congestion window in multipath environments.

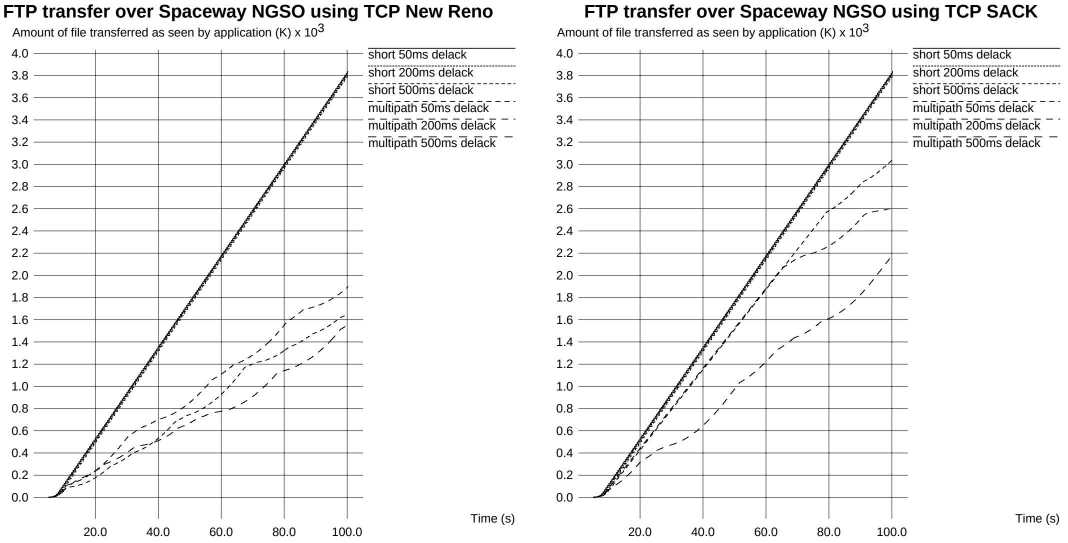
The *ns* simulations whose results are presented in figures 3.6 and 3.7 did not obey this *SHOULD*. Comparison of SACK and New Reno traffic across these simulated networks showed that SACK's performance was not noticeably better than that of New Reno's, and often seemed worse, which was surprising.

The *ns* delayed-ack receiver was modified to obey this *SHOULD* (a change later checked into the *ns* distribution), and the simulations were run again.

The results of those simulations are presented in figures 3.8 to 3.11. Those figures show that following this recommendation to only delay acknowledgements to packets at the right edge of the window increases throughput for both New Reno and for SACK.

This recommendation also goes some way towards avoiding delayed acknowledgements when the TCP sender is attempting to grow its congestion window in reordering-induced fast recovery.

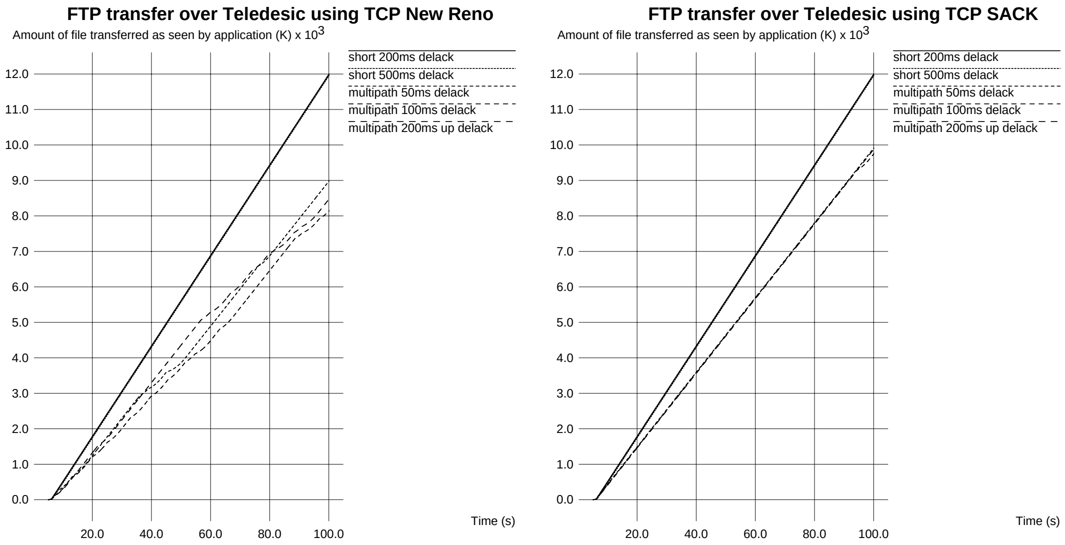
This increase in FTP throughput over that of receivers not following this recommendation in particularly noticeable for SACK's fast recovery algorithm (described in detail in [**Henderson99** Appendix A]), where the congestion window is inflated more rapidly.



**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo *using Spaceway NGSO*
delacks only take place on packets received at the right edge of the window (RFC2581 SHOULD)

**Figure 3.10 - delacks degrading rate of file transfer over MEO obeying RFC2581**



**a. Transfers using New Reno TCP** **b. Transfers using SACK TCP**

Progress of FTP transfer between terminals at Quito and Tokyo using *Teledesic*
delacks only take place on packets received at the right edge of the window (RFC2581 SHOULD)

**Figure 3.11 - delacks degrading rate of file transfer over LEO obeying RFC2581**

## 3.6 Practical considerations for the constellation

### 3.6.1 Transient effects

It is possible that dynamic routing changes, due to link failures or to handover in the satellite constellation, may result in dupacks and a drop into fast recovery at high throughputs. This is due to loss of packets at routing changes, or to interleaving of packets and acknowledgements in flight along both original and new routes as routing information is propagated. Such transient effects are discussed further in [**Varadhanetal98**], where the effects of link failures and reestablishment on traffic are looked at in detail.

The dynamics of handover and resulting transient effects in the constellation network were examined at the end of Chapter 2.

### 3.6.2 Layering of protocols

It is unlikely that any broadband satellite constellation will be a true IP packet-switching network. Instead, the need for management of frequency allocation, particularly in the terminal uplink, dictates the need for logical link control (LLC) and media access control (MAC) with a fixed frame size.

A number of proposed schemes plan a custom MAC frame encapsulating a small number of ATM cells (generally two), so that IP packets would then be tunnelled over an ATM-based network.

The recent advent of multiprotocol label switching (MPLS) can allow IP-based routing of ATM flows of IP traffic in ATM switches. This seems promising for satellite constellations as it allows support for IP multicast and QoS, as well as enabling traffic engineering. This is discussed further later in Chapter 5.

### 3.6.3 Flow-aware and unaware approaches

Although TCP is tolerant of out-of-order traffic, other none-IP-based traffic is less so, and would be more likely to require routing across a single path, with priority over best-effort Internet traffic. This consideration dictates a flow-based approach to traffic engineering in the constellation.

If multipath routing is used in the constellation without considering individual flows at each hop, it would be possible to have large-scale buffering and reordering of entire traffic flows at the edges of the satellite network, rather than just the minimal buffering needed to reconstruct higher-layer frames from MAC-level frames. However, this would add to overall latency, state held in the network, and system complexity.

### 3.6.4 A split-TCP approach

It is possible to implement split-TCP connections across the constellation, where the TCP senders communicating over the satellites are optimised for local conditions, while TCP senders talking to the terrestrial networks are set up differently. This is already done to optimise local performance over GEO satellite hops by tweaking congestion control algorithms to suit satellite link conditions and error rates e.g. by modifying slow-start and initial window sizes.

If the constellation uses multipath routing, the TCP implementation used in terminals and gateways for satellite communication would include modifications, such as a higher dupack threshold for fast recovery and disabled delayed acknowledgements. This can improve performance across the constellation relative to end-to-end TCP traffic, while the TCP implementation used for communicating with the terrestrial Internet would be configured differently.

This approach is reminiscent of many of the split-TCP implementations used with existing geostationary satellites today. Split-TCP approaches in the satellite environment and implementation details are discussed in more detail in [**Henderson99**], where considerable performance gains are shown for the split-TCP implementations against competing flows, particularly for the large round-trip times encountered for geostationary satellites.

## 3.7 Summary

This chapter has examined TCP over satellite constellation networks and have clearly shown that using multiple paths to spread network load and to avoid packet discards due to congestion can cause a TCP sender to behave just as if it was on a single, congested, path where packets are dropped. This is due to TCP’s fast recovery algorithm, which is intended for handling losses in an ordered, sequential, flow of segments.

Delayed acknowledgements can damp window growth to impair the performance of TCP over multipath routing still further, particularly when acknowledgements to an ordered sequence of packets filling in gaps in a window are delayed as well.

Although TCP tolerates the misordering that results from multipath routing and load balancing, TCP's overall performance suffers as a result of its assumptions about congestion. This performance degradation was demonstrated by looking at the effective throughput (goodput) of the TCP flow, as seen by the receiving application.

Performance considerations encourage a single ordered flow of traffic between source and destination. TCP is encouraging a circuit or flow paradigm in the underlying packet networks. The design of TCP is acting to affect and restrict the design of future networks. TCP should be examined with a view to improving its tolerance of simple load balancing and multipath routing. Algorithms to vary the dupack threshold dynamically once large amounts of packet reordering are detected would be particularly valuable.

Chapter 2 discussed various Manhattan mesh or simple forwarding approaches to routing based on local information. As a result of the design of TCP explored in this chapter, such approaches have been shown to be less desirable from a TCP performance viewpoint than a global single shortest-path routing approach or a flow-based traffic engineering approach to routing within the satellite constellation network.

Performance degradation due to simple flow-unaware routing approaches within the constellation network could be compensated for by complex reordering of packets at gateways or by a split-TCP approach, using a satellite-optimised TCP that includes more tolerance of misordering-induced duplicate acknowledgements.

A TCP sender communicating with a ground station through a single satellite avoids these mesh routing considerations, while designing constellation networks without intersatellite links or using mesh-free geostationary networks can avoid such routing strategy decisions entirely.

# *4. Multicast*

Communications using TCP work over even geostationary satellites, have done so in the past, and will continue to work over satellite in the future. However, there is more to the TCP/IP family of protocols than simply being able to support end-to-end communication using TCP. For example, interactive applications may use the connectionless User Datagram Protocol (UDP), the Real-time Transport Protocol (RTP) or communicate between groups of endhosts using IP multicast.

The trend towards complex switching and routing onboard satellite, and the network topologies created by an orbiting constellation of broadband satellites with ISLs, have given constellation networks the ability to route traffic internally through multiple satellites, from a source to a destination on the ground. Controlled duplication of packets at the satellites would permit multicast trees to be built in the constellation.

Although unicast transmissions, such as those for TCP connections, can be supported end-to-end across any network of proprietary design (including satellite networks) by simply tunnelling, implementing support for other protocols in the TCP/IP suite, particularly IP multicast protocols, is less straightforward.

## 4.1 Overview of multicast

Multicast fits between the widely-used unicast point-to-point and the broadcast communication techniques.

In unicast, data is sent from one source to one destination. In broadcast, data is sent from the source to all other hosts in the network simultaneously (or, more typically, in the same subnetwork; an Ethernet subnet is a physical broadcast medium as a result of its shared bus).

For group applications, where more than two users are exchanging messages and maintain shared state, the number of unicast connections required increases rapidly as the number of users increases. To prevent applications from needing to know about all users in the group, or needing to be responsible for maintaining all these connections, and to decrease network load, we require multicast. Multicast is the efficient emulation of a broadcast service to interested users, within the constraints of a network environment.

Multicast allows a source to send data simultaneously to all hosts on the internetwork where users are interested in receiving the data, but in a more efficient manner than simply flooding the entire internetwork with redundant broadcast packets.

The set of all hosts with interested users or participants forms a multicast group. Uninterested hosts that are not in the group do not see the data, perhaps because there is no need for the data to be sent across their subnetwork, or, if they do see it, discard it. An example of this is the logical level of Ethernet, where packets not addressed to the card's network interface or associated groups are ignored.

To communicate the data efficiently to all hosts in the group, each network or internetwork must set up a *spanning tree* connecting the subnetworks of all interested users, along which the multicast messages can be sent and replicated at tree branches. Construction of spanning trees for multicast protocols has been considered in depth previously for networks, notably in [**Deering91**] and [**BallFranCrow93**]. Group management is generally separated from tree construction, and becomes an internetwork function in Deering's IP multicast group model.

Some network technologies implement multicast support at a low level. For example, Switched Multi-megabit Data Service (SMDS) implements the concept of multicast group addresses at the data-link layer. The local router is responsible for mapping IP multicast group addresses efficiently onto this functionality.

## 4.2 Tunnelling and multicast

As we have seen in Chapter 2, a satellite constellation network using intersatellite links is a network with a dynamic, changing but quasi-stationary topology in the form of a toroidal or semi-toroidal mesh. Such a network requires internal routing strategies in order to make decisions on how to transmit data across it between widely-separated ground terminals operated by users.

Those ground stations may themselves be connected to and passing data to and from other networks; the satellite constellation network forms an intermediate connection between the ground stations. This means that the ground stations are gateways between networks, and that the constellation network must be capable of transporting data between networks and interoperating with those networks.

Data transport can be accomplished easily from point to point across the constellation network by simply *tunnelling* the traffic. Packets or cells of data sent from one ground network to another do not need to be understood in any way by the constellation network, as they are *encapsulated* within the constellation network's chosen protocol (likely to be ATM, as discussed in section 3.6.2). This has the advantage of making the satellite network transparent to terrestrial networks, by hiding it below the terrestrial networks' network layer, and allows the satellite network designers to select network protocols that they consider best-suited to the space environment or to international standards. However, tunnelling in this way has disadvantages; it only handles unicast transmissions, or point-to-point communications setting up an end-to-end connection across the constellation network.

When networks on an internetwork are not multicast-aware, multicast messages can be *tunnelled* across them to all the multicast destinations by encapsulating (and, if necessary, fragmenting) the multicast messages within network messages native to that network. However, establishing multicast tree branches within a tunnel using information in the tunnelled packet header is generally not possible.

### 4.2.1 Defining tunnelling

In brief, tunnelling is out-of-context encapsulation, where the encapsulating layer subsumes the network functionality already expressed in the encapsulated layer.

Tunnelling is often used to route packets in one network through an intermediate network that belongs to a different routing realm and that can have a differing network layer. Packet formats, addressing space, and routing paradigms in the two networks may be *entirely* different.

When IP is tunnelled, a virtual IP hop (a tunnel) is created between the two IP-capable routers at the borders of the intermediate network. When entering the tunnel, the IP packet, including the IP header, is sent as payload data to the other side of the tunnel using the network layer of the intermediate network.

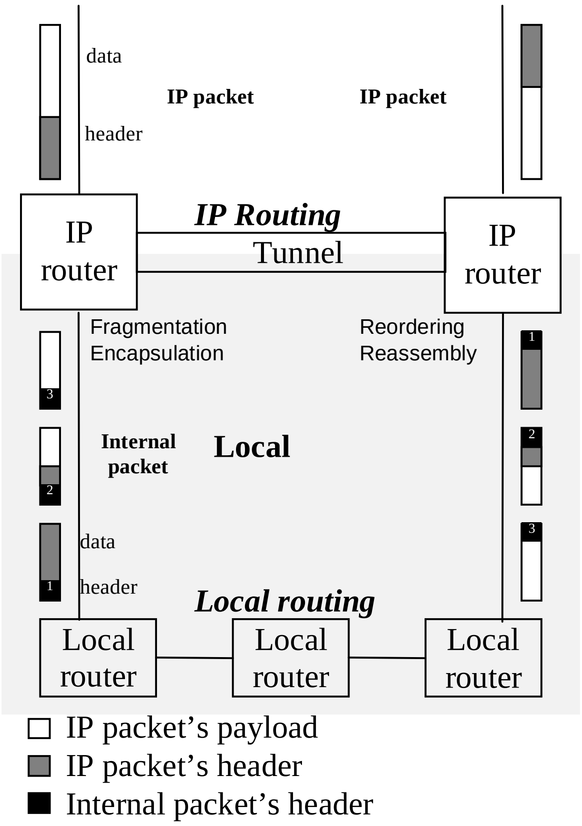


**Figure 4.1 - conceptual view of tunnelling [Woodetal01a]**

Steps in tunnelling, shown in Figure 4.1, include:

- Possible implicit fragmentation of an IP packet to fit in the payloads of packets used in the intermediate network.
- Building local packets or cells, suitable for routing or switching across the intermediate network using its routers or switches, by the addition of appropriate header information.
- Setting the destination address in these packets or cells to the internal network address of the IP-capable router at the other end of the tunnel.

When leaving the tunnels, the IP packets are reassembled as required and forwarded along the next IP hop.

Tunnelling is often used as a transitional measure to support new functionality added to parts of today's Internet without requiring the entire Internet to be upgraded at once to support new features for those who wish to adopt them (a so-called 'flag day'). It can also be used to join legacy networks across a new, alien, network infrastructure.

An example of transitional tunnelling, using IP-in-IP tunnelling to interconnect network islands using the new protocol, is the 6Bone, a testbed virtual network that is assisting in the deployment of IPv6 over IPv4 as part of the Internet's gradual move to IPv6. Examples of legacy tunnelling include Appletalk or IPX over IP.

### 4.2.2 Tunnelling group transmissions

Supporting group transmissions across an internetwork – say, *n* disparate subnetworks, each with a gateway to the constellation network and using the network as a backbone to communicate – would need to be carried out as up to:

$\sum_{k=1}^{k=n-1} n$ separate unicast transmissions across the tunnelling constellation network

(this is equivalent to ½[$n(n$-1)] bi-directional connections accomplished via e.g. TCP).

Managing the group transmission – adding or removing group members – requires either considerable intelligence on the part of either the constellation network gateways or the hosts participating in the group transmission.

If the constellation network is not multicast-aware, it is conceivable that the senders or the multicast-aware ground networks could compensate for this deficiency in the backbone by setting up *reflectors* to form their own application-level emulation of multicast across the constellation network. This moves management of the 'multicast' to the application layer and concentrates traffic at central reflectors to minimise the mesh complexity of multiple end-to-end unicast connections required across the constellation network between all interested hosts. (This is, in fact, how Cornell University originally implemented *CU-SeeMe*, a Macintosh-based videoconferencing application, across networks whose routers did not implement multicast protocols.)

However, *mirroring* the transmissions from the reflectors by duplicating unicast packets for each destination adds considerably to the overall network use and degrades overall performance. Deficiencies in the underlying network structure are being compensated for at the application level by the creation of a virtual network.

If a satellite network:

- was to implement ATM as its transport mechanism, in a world where applications increasingly speak IP,

- had to transport IP, the Internet Protocol, successfully as a requirement for seamless internetworking with the Internet,

then tunnelling of IP over ATM-over-satellite would be required. As described above, this simple tunnelling approach is sufficient for unicast connections, but does not allow effective management of multicasting as multicast packets cannot be easily both tunnelled *and* replicated within the backbone network at the same time without explicit knowledge of IP multicast routing requirements in the tunnelling network. Replication of packets for multiple destinations must take place before tunnelling, which increases the network capacity used in the tunnelling network.

Another disadvantage of tunnelling is that it can give a false impression of the latency involved in communicating across the tunnel for any protocol that uses packet life counters in its packet headers, and decrements those counters to get an idea of how far a packet has travelled. Anything relying on this counter information for link weighting gains a false picture of the network, as all tunnels appear as a single link and therefore a faster route to the destination than non-tunnelled routes through multiple routers that each decrement the packet hop counter. IPv6 replaces IPv4’s time-to-live (TTL) millisecond counter, which few IP routers could handle and decrement accurately due to difficulties in measuring elapsed time, with a simpler hop counter that standardises the *de facto* decrement-TTL-field-by-one behaviour of most IPv4 routers. All tunnels appear the same length to the packets being tunnelled through them, unless explicit packet marking is carried out at the tunnel endpoints.

Tunnelling generally does not make optimal use of the intervening network. Often the tunnel must be set up manually, and the tunnel endpoints cannot be moved easily.

## 4.3 ATM multicast

With its circuit-switching focus, ATM did not originally have the concept of multicast, and the closest it came to multicast was the concept of a multicast server (MCS), where virtual circuits from all receivers would connect to a single source, which duplicates information to each receiver. Point-to-multipoint virtual circuit connections (VCCs) are then managed by the MCS for each group. This scales extremely badly in large networks with many users, as the links adjacent to the server soon run out of both capacity and available Virtual Path Identifiers (VPIs) on local links to allocate for use by individual VCCs. An alternative to the MCS is the multicast virtual circuit mesh, where each ATM source establishes its own independent point-to-multipoint VC to recipients. (This is a form of source-based multicast.)

Member-initiated joining and leaving of a multicast group has been included in later versions of the ATM Forum's UNI, but this is not enough for convenient multipoint-to-multipoint communication.

As a result of this, new multicast architectures have been proposed for ATM, e.g. SEAM, a Scalable and Efficient ATM Multicast, which uses a single core-based tree and combines Virtual Circuits to minimise VC switching overhead [**GrossRama97**].

However, although considerable effort has taken place, no ATM multicast protocols have yet been standardised. Despite the improvements in multicast management included in the second iteration of the Private Network-Network Interface (PNNI 2.0) that allow sensible join and leave strategies for multipoint-to-multipoint connections, ATM multicasting is less mature than IP multicasting.

## 4.4 IP multicast

IP multicasting protocols rely on the Internet Group Management Protocol (IGMP) to manage multicast groups. IGMP allocates reserved Class D IP addresses to groups. There is a one-to-one relation between a multicast group and a D address at any given time [**RFC1112**, **RFC2236**].

Multicast routing protocols use this addressing abstraction. These multicasting protocols are many and varied in their approaches to handling multicasting.

[**RFC1458**] summarises early implementations, while [**Almeroth00**] and [**Ramalho00**] overview the development of multicast and provide a taxonomy of protocols.

Reliable multicast transport protocols, which add reliable delivery to a multicast tree, are categorised in [**Obraczka98**]. This thesis will limit the discussion to popular, well-established, multicast routing protocols.

## 4.4.1 IP multicast over ATM networks

### 4.4.1.1 The MBone

The MBone, or Multicast Backbone, began as a demonstration of high-speed IP multicasting over fast ATM networks. It has proven to be far more popular and well-established than originally intended.

In the MBone, isolated multicast-aware IP subnets are connected to other, remote, multicast-aware IP subnets using a hand-built hand-maintained topology using end-to-end tunnels between the subnets. To allow easy use of video and audio applications, high-capacity fixed ATM links were often used for the tunnels [**Eriksson94**].

The Mbone and its tunnels form a virtual internetwork layered on top of the physical internetwork. Multicast-capable routers (or end hosts running **mboned** software to act as routers) forward packets to neighbouring multicast routers within the MBone topology. A tunnel is configured between topologically-near routers that are separated by non-multicast-capable routers, and multicast packets pass through this tunnel.

The MBone was intended as a temporary way of getting world-wide IP multicast communication functionality up and running, while easing the transition to an IP-multicast world (much as the 6Bone, where IPv6 packets are tunnelled between IPv6 subnets within IPv4 packets, is intended). However, the high-speed fixed ATM links required for effective video and audio use and the speed of initial deployment resulted in what has now become a popular established networking implementation. The transition to a fully native non-tunnelled MBone is slow and is still ongoing.

MBone over geostationary satellite demonstrators have been carried out to demonstrate that these fixed, inflexible, hand-managed tunnels can be extended into the satellite [**AlmZhang98**] and ATM-over-satellite domains [**Zhangetal97**].

As an example of tunnelling and with a hand-maintained topology, the MBone is not a good example of the ideals of internetwork multicast, much less of viable tree construction or internetwork multicast congestion-avoidance routines. As the hand-built topology relies on fixed links, the MBone would not map well to the moving full mesh that is a LEO constellation, where regular automatic routing updates are required. The MBone relies on DVMRP, discussed later in section 4.4.2.1.

#### 4.4.1.2 MARS and VENUS

MARS, or the Multicast Address Resolution Server [**RFC2022**] is a way of supporting IP multicast over virtual-circuit-based ATM networks. MARS works by creating proxy servers at the network edges where the ATM and IP networks join. The MARS server registry holds information associating layer-3 multicast group members with ATM interfaces. Each MARS server can only serve a single IP subnet with its own logical mapping tables and support, and IP multicast routers utilising MARS to cross the ATM network must be carefully positioned to avoid seeing and using multiple MARS servers.

MARS illustrates the limitations of storing network routing information at the edges of the network, with the loss of flexibility that results from individual servers separate from the routing and from tunnelling. MARS can be useful in carefully controlled topologies, such as the MBone, but is of much less use in the general internetworking case. MARS has been extended to support more than one subnet with VENUS [**RFC2191**], but its lack of scalability is widely recognised.

### 4.4.2 Existing IP multicast routing protocols

The IP multicast routing protocols can themselves be divided into two basic sets of groups, depending upon a taxonomy predicated on the basic assumptions made about traffic use in the spanning tree, and on the distribution of multicast group members throughout the network. These are that the multicast tree is either *source-based* or *a shared tree*, and that it is either *sparse* (assumes few group members in the internetwork) or *dense* (assumes many members in the internetwork). Source-based trees are generally dense, while core-based trees are generally assumed to be sparse.

#### 4.4.2.1 Source-based trees

*Source-based tree* multicast protocols are data-driven or source-initiated. Construction of the multicast spanning tree begins top-down from the source outward as it transmits information, and data on the state of the tree is flooded to all routers. Routers on subnetworks with no interested members ‘prune back’ by requesting the tree no longer reach them. (This results in considerable soft state overhead, as uninterested routers must continually remove newly-received state concerning new multicast trees.)

The multicast trees constructed allow data to travel in one direction, from source to group, emulating broadcast. Distance Vector Multicast Routing [**Deering91**] is an example of this type. The initial flooding assumes implicitly that potential group members are densely distributed throughout the internetwork, i.e. that many subnetworks contain at least one group member and will be interested in receiving the communication, making the imposition of state concerning the multicast tree and the resulting network overhead worthwhile.

For full group-to-group communication using source-based multicasts, a separate multicast tree must be set up for each source, or *(source, group)* tuple. This scales badly for large groups and imposes considerable joining and leaving overhead (build a new tree, then destroy it) if we are considering group communication between peers. Source-based protocols include:

- Distance Vector Multicast Routing Protocol (DVMRP) [**RFC1075**], where every host on the network is initially assumed to be part of the multicast group receiving traffic from the source. The tree is then pruned to an optimal state, connecting only interested networks, via the use of Reverse Path Forwarding (which requires bi-directional links for pruning, although the resulting tree is still unidirectional from source to destinations). The spanning tree effectively begins as an uncontrolled broadcast from the source that is then cut back to a more efficient multicast state. The periodic discovery floods required by this protocol to set up trees would be undesirable in a wide-area network such as a satellite constellation. Each sender in a multi-way multicast would require its own tree to be set up, as the spanning tree is for one-way delivery of communication from a single source. DVMRP also requires that all routers receive and maintain state on every multicast group, and so can be expected to scale badly for large networks where group members are widely separated. DVMRP is widely implemented on the MBone. It is derived from and relies on characteristics of the earlier RIP, the Routing Information Protocol, its unicast equivalent [**RFC1722**].

- Multicast Open Shortest Path First (MOSPF) is based upon OSPF [**RFC1584**, **RFC2328**]. OSPF routes messages along a least-cost path, where cost is expressed in terms of a dynamic link-state metric that can represent such things as the amount of traffic on the link, or the latency involved in using the link. MOSPF relies on

OSPF and uses the Dijkstra algorithm to compute a shortest-path tree. However, MOSPF floods IGMP information across the routing domain periodically, and so does not scale well, although the requirement that all MOSPF routers have a complete topology map and know all the locations of members is not infeasible in the single autonomous system that is the satellite constellation network.

#### 4.4.2.2 Core-based trees

Core-Based Trees (CBT), with one or more central routers from which the tree branches out in all directions, have been suggested for groups where there are many active senders within the group, allowing multi-way communication over a single tree [**BallFranCrow93**]. CBT was later enhanced to handle a hierarchy of multiple cores reliably [**ShieldsGarcia97**]. An architecture for core-based trees is described in [**RFC2201**], while a related protocol is specified in [**RFC2189**].

Core-based-tree multicast protocols have receiver-initiated multicast spanning trees, where a router becomes involved in a branch of a multicast distribution tree only when one of the hosts on its subnetwork requests membership by issuing a join message. There may be one or more central core routers that receive join and leave messages, and that pass received multicast packets downstream through the tree.

The lack of any initial flooding and the assumptions of constrained capacity and fewer interested members, sparsely distributed, mean that these shared trees are more scalable for internetworks than source-based trees. In a source-based tree, every active source is associated with its own tree. This results in a scaling of

***O*(**set of sources * set of members for each source group**)**.

In a shared tree, the scaling will be ***O*(**all group members**)** with less state held in the network. However, there is a delay tradeoff in going from shortest-path trees, based directly on the underlying routing protocols, to shared trees, where all multicast traffic must travel via the core. This single shared-tree *(*, group)* approach differs from the *(source, group)* pairings of source-based trees such as DVRMP and MOSPF.

Any source wishing to send data transmits it to the core, which then multicasts it to all receivers in the group via the tree. Any source that is not already a tree member will encapsulate the multicast packet in a unicast packet addressed to the core.

| Type | Source-based trees | Shared trees |
|---|---|---|
| **Dense** | DVMRP, PIM-DM, MOSPF | |
| **Sparse** | PIM-SM | CBT, OCBT, PIM-SM |

**Figure 4.2 - a taxonomy of multicast protocols**

This prevents the need for all routers in the network to know the location of the core, and decreases the amount of multicast tree state that must be held in the network.

Choosing an appropriate position in the network for core routers for the multicast group, or changing the core positions as the receiver set/network topology changes, is a non-trivial problem. A good core position decreases the amount of multicast state routing information that needs to be stored and the number of routers involved.

Protocol-Independent Multicast – Sparse Mode (PIM-SM) [**RFC2362**] constructs a multicast tree around a chosen router, called a rendezvous point, similar to the core in CBT. (PIM-SM is distinct from Protocol-Independent Multicast – Dense Mode, or PIM-DM. PIM-DM is similar to DVMRP, but relies more upon the underlying unicast routing protocol.) However, PIM-SM allows shortest-path source-based trees as well as shared-group trees. In PIM-SM, a multicast packet is *always* encapsulated in a unicast packet sent to the rendezvous point address, whether the source of the packet is a group member or not, since PIM-SM state held in routers participating in the tree is downstream only.

#### 4.4.2.3 Exterior protocols

The sparse or dense, source- or shared-tree protocols that have already been discussed are for use within a single managed network, or domain. The space-based satellite constellation network forms one such domain.

Exchanging information about a multicast sources with other domains to enable branches of spanning trees crossing multiple administrative domains to be established is an entirely different problem, and has resulted in creation of protocols designed to address that problem. These include multiprotocol extensions to the Border Gateway Protocol [**RFC2858**], and the Multicast Source Delivery Protocol, MSDP [**Farinaccietaldraft00**].

To join shared trees of the same group in different domains together and establish a root domain in which the core of the resulting shared tree is placed, the Border Gateway Multicast Protocol (BGMP) has been proposed [**Kumaretal98**, **Thaleretaldraft00**]. In the satellite constellation network, these protocols must be implemented in terrestrial gateways interconnecting the constellation network with the terrestrial Internet, allowing group members and sources in other networks to communicate across the constellation.

## 4.5 Considerations for the constellation network

### 4.5.1 Choosing shared or source-based trees

An ISL-using satellite constellation network communicating with the terrestrial Internet possesses considerable network capacity in the space segment due to its broadband microwave or laser ISLs. There is also large network capacity in the fibre-based terrestrial Internet ground segment. The throughput constraint lies in the limited capacity of the ground-space air interface between the two, due to limited allocation of available frequencies useful for communication through the atmosphere.

Being able to duplicate IP multicast packets in the ISL network, for redistribution to all communicating ground parties involved at remote terrestrial terminals, optimises use of the ground-space interface, as no unnecessary packet duplication or repetition of connections needs to occur across the bottleneck air interface.

As it is in the interest of the satellite constellation network operator to make the most effective possible use of its available network capacity, this favours the minimal-routing-overhead approach taken by the shared-tree protocols.

However, constraining the network to a single shared-group tree reduces fault tolerance, and does not produce minimal shortest-path latency between two points. For minimum overall latency, a dense source-based approach with multiple spanning trees – one per source – would be necessary.

This raises two opposing sets of requirements:

1. A source-based tree, relying on shortest-path routing, can minimise latency between points by using shortest-path routing from source to group members. However, the

setup procedures of such protocols place a heavy state load on the network, particularly when the number of group members is low.

2. A shared tree minimises state held in the network, and is highly suitable for non-real-time shared data transmission where latency overhead is not critical. A sparse shared-tree protocol, adapted to support the changing topology of the satellite constellation, would be useful for maintaining state between multiple points while making efficient use of the network (and cheaper use from a viewpoint of capacity pricing). Overhead on the network of new group members joining and leaving the network will be minimal, as for multi-way communications all that is required is a new branch on the existing tree. This is in contrast to the source-based approach to multi-way communications that involves adding new branches to all dense-mode trees associated with the group and setting up a new multicast tree with the joiner as source.

In shared trees, minimum delay in communication, by using the shortest path between source and receiver, is sacrificed for network capacity and ease of administration of group setup and member joins and leaves.

Another concern for the constellation network is that it is likely to be used as a transit network, providing connectivity to networks in remote locations and enabling them to communicate with the terrestrial Internet.

For multicast connectivity, this can mean establishing a local multicast ‘tree’ across the constellation network, with a single source gateway to the bulk of terrestrial Internet and a single destination gateway to the remote network as the sole group member. This is necessary just to enable multicast connectivity to the remote location, and requires exterior protocols advertising that the remote network is reachable. Given the ratio of terrestrial Internet multicast groups to remote networks requiring satellite connectivity, such low-member groups are likely to be common. Minimising the multicast state overhead that is required for small multicast groups of two or more members can be considered important. This consideration favours the use of shared trees.

The constellation network is a transit network that spans the world, so it is likely to be a logical choice as the root domain for BGMP for multicasts permitted to run across its network, and is therefore likely to contain the core of any inter-domain shared tree.

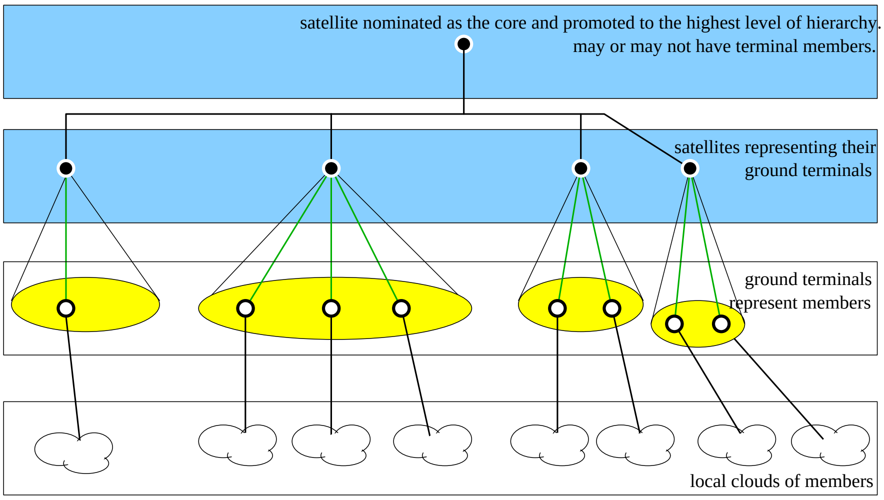


Figure 4.3 - hierarchy within the multicast tree

Multicast Address-Set Claim (MASC) [**Kumaretal98**, **RFC2909**] can be used to assign an address prefix to the constellation domain, which will ensure that the core of any inter-domain tree can be rooted in the constellation network.

### 4.5.2 Outlining use of a shared tree – placing the core

Given that the satellite constellation network is a single managed entity and is likely to contain the core of a shared tree, algorithms can be developed to determine a satellite to be nominated as that core and thus influence the shape of the tree that is built.

We can consider the hierarchy of ground users/ground terminal/satellite/core in the constellation network as suitable for the hierarchies of the ordered core-based tree protocol (OCBT) [**Shields96**, **ShieldsGarcia97**]. This hierarchy is seen in figure 4.3.

Given satellite motion over time, all satellites in the constellation will be candidate core routers for a bootstrap mechanism at some point. Any hashing functionality to determine the centre of the tree, or highest-level core, at a point in time must be based upon the locations of known group members – or rather, the known locations of their ground terminals. Core handover occurs as satellites succeed one another over a geographical position and inherit the state of their predecessors.

Algorithms to place the highest-level core (from now on, the *core*) to minimise delay overhead and network capacity used by the shared tree are of particular interest. However, finding a minimum-delay shared tree is recognised as an NP-complete problem [**SahMuk00**].

Establishing a multicast tree is of complexity $\boldsymbol{O}(n)$, where $n$ is the number of links in the spanning tree, so attempting to minimise the size of the spanning tree would be advantageous. As core handover *must* occur between satellites due to satellite motion, moving the core more dramatically to suit the requirements of the current multicast group during a handover is not unimaginable.

A truly fair core placement algorithm would attempt to minimise the differences in delays between the core and all ground terminals, giving similar delays across the spanning tree from the core to all multicast group members. However, this presumes that all group members are directly connected to the constellation network, that any delay in the terrestrial network is insignificant, that fairness is important to the multicast application, and that an optimally-fair location can be easily computed. When supporting inter-domain multicast, where terrestrial networks will be involved and traversed by multicast traffic, this is unlikely to be achievable. Instead, this chapter focuses on a simple algorithm to produce a central core that shapes the spanning tree around itself. (For failure-tolerance, satellites around the central core satellite can be nominated as secondary cores; one can assume core duties if the current core fails.)

## 4.6 A simple vector algorithm to nominate a core satellite

A method of selecting a location for a single core for a shared multicast tree in the constellation network, based on aggregating the locations of ground terminals that each represent a number of group members, can be described as follows:

### 4.6.1 Description of the algorithm

1. Let the number of interested group members be represented by $m$, where $m$ is known or a fair approximation. Let the number of ground terminals representing those $m$ group members be $n$, where $n$ is always less than or equal to $m$, and $n$ is known. Each terminal knows how many members it represents. Let the number of satellites talking to the $n$ terminals be $p$, where $p$ is always less than or equal to $n$. Each satellite knows about the terminals that it represents, and so has some idea about the number of members that it represents in the hierarchy.

   If $p$=1, the satellite talking to the $n$ terminals in its coverage area is immediately nominated as the core and promoted to the highest level of the hierarchy.

If $p=2$, the constellation network is currently being used to reach a remote network. Any satellite currently talking directly to a known terminal can immediately be nominated as the core and promoted, to ensure that shortest-path routing is used, although promoting the satellite with the largest membership is preferable. If $p>2$, continue with this algorithm.

2. For each ground terminal $k$ known to represent interested group members, take spherical coordinates ($\mathrm{lat}_k$, $\mathrm{long}_k$) of that terminal. Repeat for all $n$ terminals. Convert each set of terminal coordinates to a vector $\mathbf{v}_k = [x_k, y_k, z_k]$, of magnitude 1.

   The vector is directed out from the centre of the earth as origin O. Multiply the vector by the scalar $j$ representing the $j$ group members at that position. This can be done locally in each satellite – the next logical level up – before the information is pooled across the network and used to find the core.

   (An optimisation for large groups would be to only consider the current coordinates of each of the $p$ satellites that the set of group members in each satellite footprint communicates with. Each satellite will be told the weightings of the $q$ ground terminals that are using that satellite to communicate, where each weighting represents interested group members at that ground terminal. Make the magnitude of each satellite vector representing its position $\Sigma q$, thus representing the number of group members that the satellite knows about and weighting the satellite vector. So, rather than summing one vector per terminal representing local members, there is now one vector per satellite, representing local terminals multiplied by a sum of magnitudes. Since satellite movement results in a gradual change in the membership for the satellite as terminals execute handovers, this approximation is only useful if the algorithm is run at regular intervals.)

3. Sum all the vectors for the group to get a resultant vector, as shown in figure 4.4:

   $$\mathbf{c} = [x_{\mathrm{sum}}, y_{\mathrm{sum}}, z_{\mathrm{sum}}] = [\Sigma_k x_k,\ \Sigma_k y_k,\ \Sigma_k z_k].$$

4. Convert the direction of **c** into ($\mathrm{lat}_c$, $\mathrm{long}_c$) to determine the position of the core. Find the satellite currently nearest this position from known satellite positions, and nominate this satellite as the core of the group's multicast tree. Repeat this last step at regular intervals, moving the core to a new nearest satellite when required.

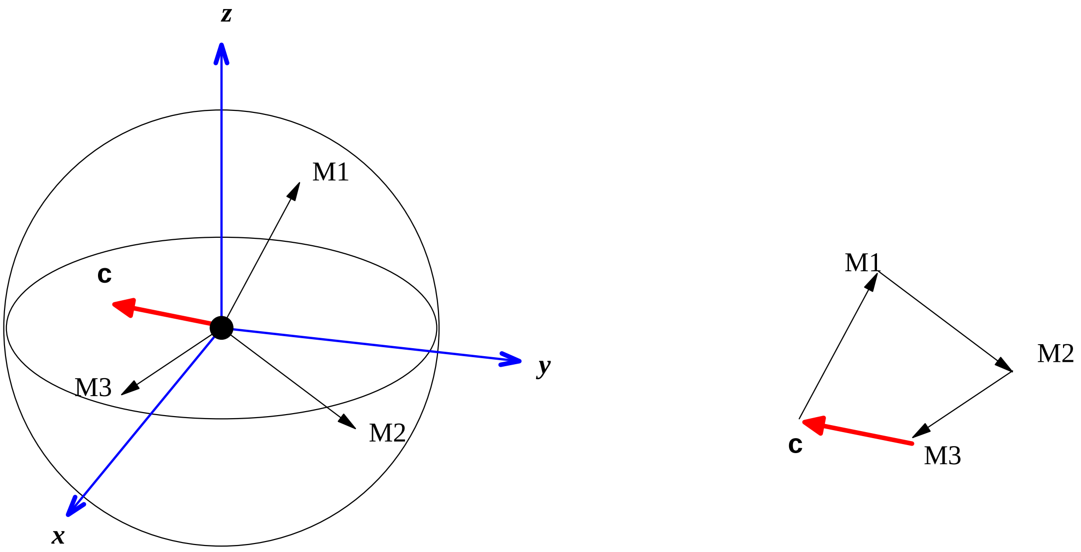


**Figure 4.4 - vector summation to locate the core position**

If the terminals regularly supply updates on the size of their pools of interested members, this is sufficient to provide a core location. However, explicitly adjusting the core location when members join or leave the multicast group can be as follows:

### 4.6.2 Handling new member joins

A new member wishes to join the multicast group, and has sent a join message to its local terminal to do so. If that member is the only local group member, the ground terminal is itself new to the group's tree and propagates its request to the core. If the ground terminal is not new to the tree, it merely has to update the weight of its associated vector to reflect the new group member.

There are $n$ ground terminal positions and $p$ satellites currently involved in the multicast group; the $m$+1th member of the unordered set of members is about to join.

1. If, before joining, $p$=1, do nothing further; the core can stay where it is at the current satellite, since $n$ can only stay 1 or become 2. If, before joining, $p$=2, compute $\mathbf{c}_{m+1}$ as already described in the preceding section for **c**. If $p$>2, continue.

2. If **c** has been discarded, convert ($\text{lat}_c$, $\text{long}_c$) into vector **c**, and make the magnitude of **c** a scalar representing the $m$ current group members.

3. Take spherical coordinates ($\text{lat}_{m+1}$, $\text{long}_{m+1}$) of the ground terminal for the new member joining the group (or, as an approximation, the coordinates of the satellite

that the new member currently belongs to, i.e. that its terminal communicates with. If the terminal or the satellite is already represented in the group, merely increase the weighting of the terminal or satellite by increasing the magnitudes of their vectors to reflect the new member.)

4. Convert to a vector $\mathbf{v}_{m+1} = [x_{m+1},y_{m+1},z_{m+1}]$ of magnitude 1, directed out from the centre of the Earth as origin O. If the satellite is already a member of the group, so that $p$ is not increased, the information on the new member merely increases the magnitude of that satellite's vector. The new member is added as magnitude 1 regardless of whether the ground terminal or satellite coordinates are used, since other members associated with the magnitude of the previous vector are already represented in the magnitude of **c**.
5. let $\mathbf{c}_{m+1} = \mathbf{c} + \mathbf{v}_{m+1}$. This is the new value of **c**.
6. Convert the direction of **c** into ($\text{lat}_c$, $\text{long}_c$) to determine the position of the core. Take the satellite currently located nearest this position, and make it the core.

### 4.6.3 Handling member leaves

A member of the multicast group wishes to leave the group, and has informed its ground terminal. If that member is the only local group member, the ground terminal can leave the group by communicating a vector magnitude zero to its satellite. Otherwise, the change in size of the vector magnitude must be communicated to the satellite and to the core. There are $n$ ground terminal positions and $p$ satellites currently involved in the multicast group; the $m$th member of the unordered set of members is about to leave.

1. If, before leaving, $n$=1, simply discard the group information at higher levels and end. If $n$=2, nominate the satellite currently above and communicating with the remaining member ground terminal as the core. If $n$>2, continue with this algorithm. If, after leaving, once the local satellite and core vectors have been modified, if $p$=1 and $n$>1, nominate that satellite as the core. If $p$=2, nominate the remaining satellite with the largest membership as the core.
2. If **c** has been discarded, convert ($\text{lat}_c$, $\text{long}_c$) into **c**, and make the magnitude of **c** a scalar representing the $m$ current group members.

3. Take spherical coordinates ($\text{lat}_m$, $\text{long}_m$) of the ground terminal representing the member *m* leaving the group (or, as an approximation, the coordinates of the satellite through which the member communicates).
4. Convert to a vector $\mathbf{v}_m = [x_m, y_m, z_m]$, of magnitude 1, directed from the centre of the Earth as origin O.
5. let $\mathbf{c}_{m-1} = \mathbf{c} - \mathbf{v}_m$. This is the new value of **c**.
6. Convert the direction of **c** into ($\text{lat}_c$, $\text{long}_c$) to determine the new core position. Take the satellite currently located nearest this position, and make it the core.

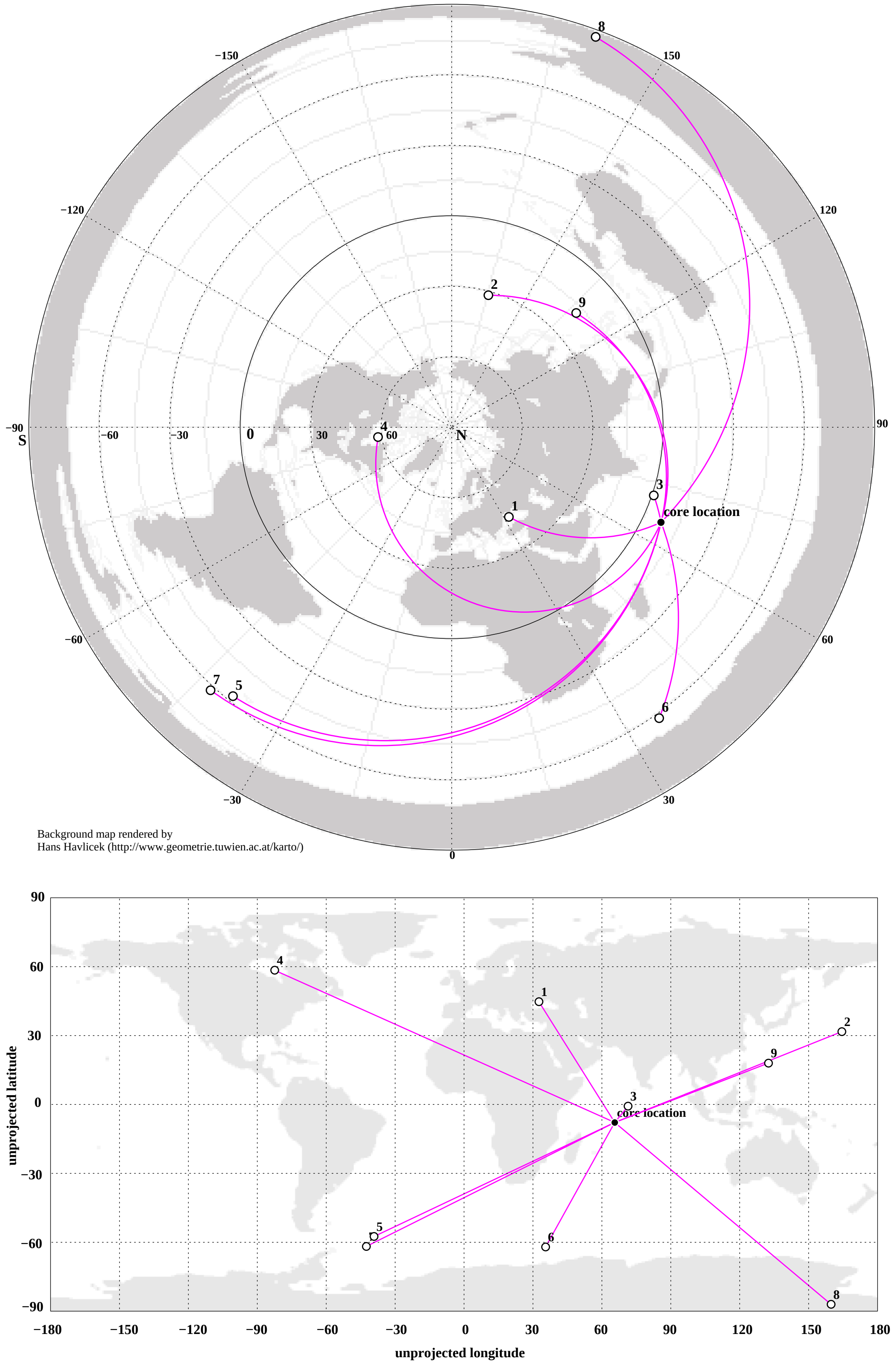
As far as this algorithm is concerned, member/ground terminal/satellite/core forms a hierarchy of cores, of approximations of positions and of summation of weights of members. The OCBT hierarchy maps cleanly onto this hierarchy; join requests always travel up the OCBT hierarchy and up the satellite hierarchy.

Ground terminals – particularly terrestrial gateways to the Internet – may be heavily weighted by exterior multicast protocols to represent large numbers of members in other networks, and this will be reflected in the magnitudes of their vector representations.

Individual terminal locations can be approximated by satellite member locations. Use the current coordinates of each satellite to obtain the satellite member vector, but multiply each satellite member vector's magnitude by the number of members in its subnetwork to give the appropriate weighting for the core location, so that vector magnitudes, rather than vectors, are summed. It is reasonable to assume that each satellite must know the number of ground terminals in its subnetwork interested in a particular multicast group, since this is a necessary part of establishing communications, allocating multicast at the link layer, and so on. In OCBT, each level of hierarchy knows about the level below it in the hierarchy.

### 4.6.4 Use of this vector algorithm

This algorithm results in a core placement that is reasonable for star constellations with cross-seam links, where the sphere accessible to the vector corresponds to the sphere of the constellation network. An example of the algorithm is illustrated conceptually in figure 4.5.



Background map rendered by
Hans Havlicek (http://www.geometrie.tuwien.ac.at/karto/)

links joining points show overall connectivity and show tree spanning the Earth around the Equator. Each terminal location has equal weighting; the satellite aggregation step in the hierarchy is skipped.

**Figure 4.5 - vector summation chooses core location using terminal locations**

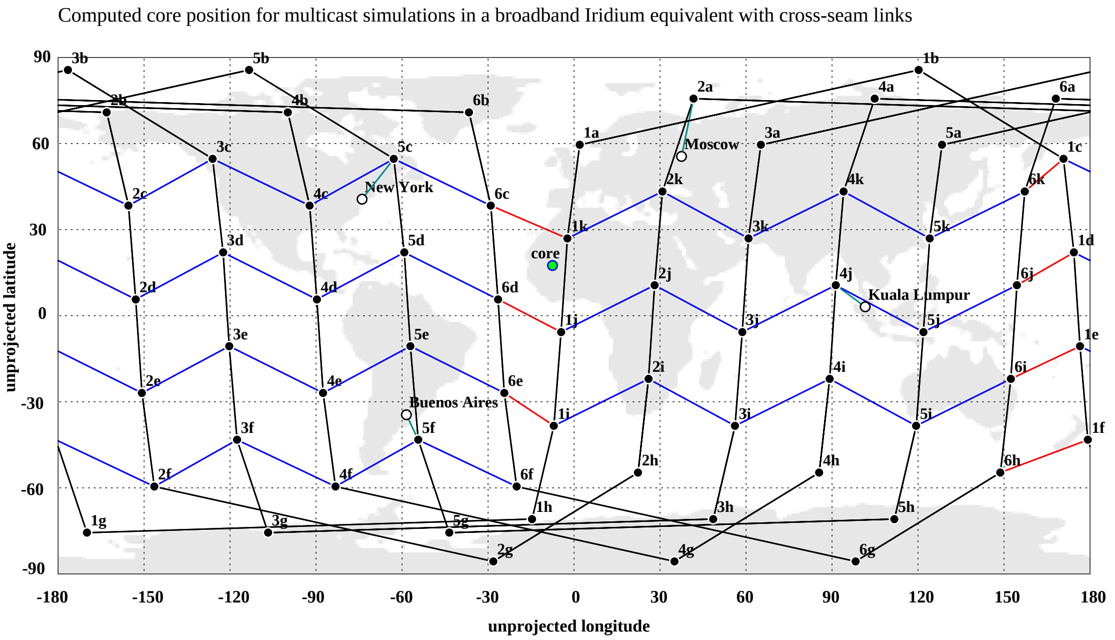


**Figure 4.6 - terminal and computed core locations under broadband *Iridium***

### 4.6.5 Evaluating this simple vector summation algorithm

To gain an idea of the performance of a group application using a core-based multicast tree to communicate across the satellite constellation, with the core location computed as previously described, use of the application was simulated. To understand the tradeoffs involved in going from unicast to a core-based multicast approach, that was then compared to an equivalent application using multiple unicast flows to achieve the same communications between endhosts located at ground terminals.

Group applications communicating between four terminals, in Buenos Aires, Kuala Lumpur, Moscow and New York, were evaluated over MEO and LEO scenarios. Simulations were conducted by using the *ns* satellite extensions, described previously, to generate a network topology at each point in time. This topology was then described as a script and exported so that an *ns* core-based tree could be run across the network to simulate traffic; in *ns*, multicast is not yet interoperable with wireless or satellite simulation. For the MEO scenario, the *Spaceway NGSO* proposal described earlier in Chapter 2 was used. For LEO, the *Teledesic* proposal posed simulation and multicast routing scalability problems for the simulation approach taken, due both to its large number of satellites and its geodesic mesh. Rather than attempt to evaluate a group application running across *Teledesic*, simulations were constrained to the similar but simpler well-known *Iridium* geometry, assuming broadband capacity and adding cross-seam links as *Teledesic* has to approximate the *Teledesic* design.

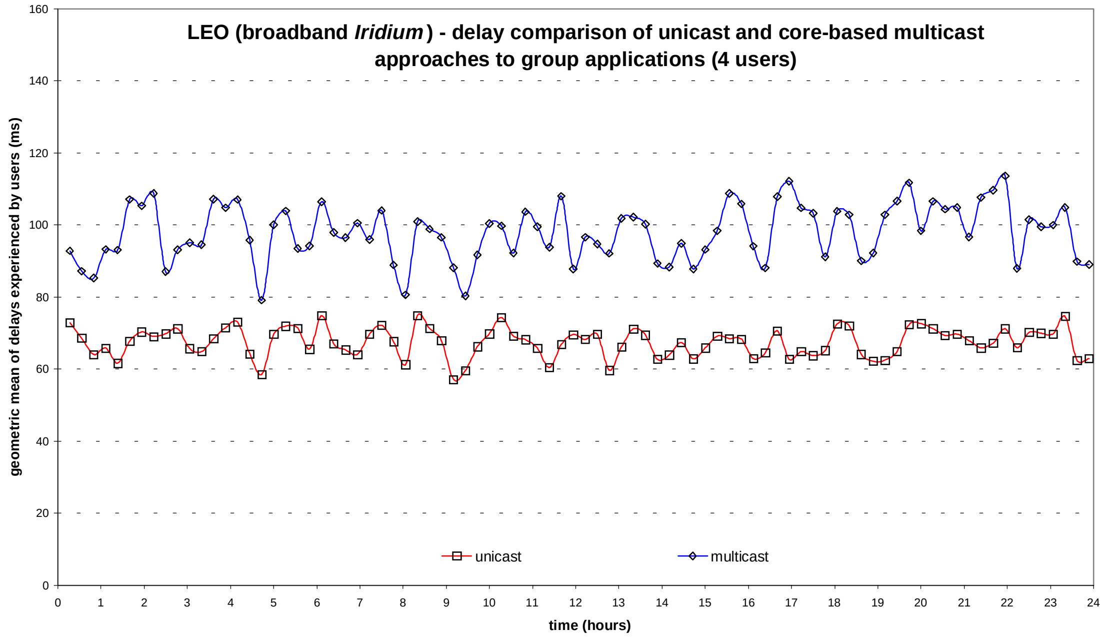


**Figure 4.7 - comparison of mean group delays for 4-user application over LEO**

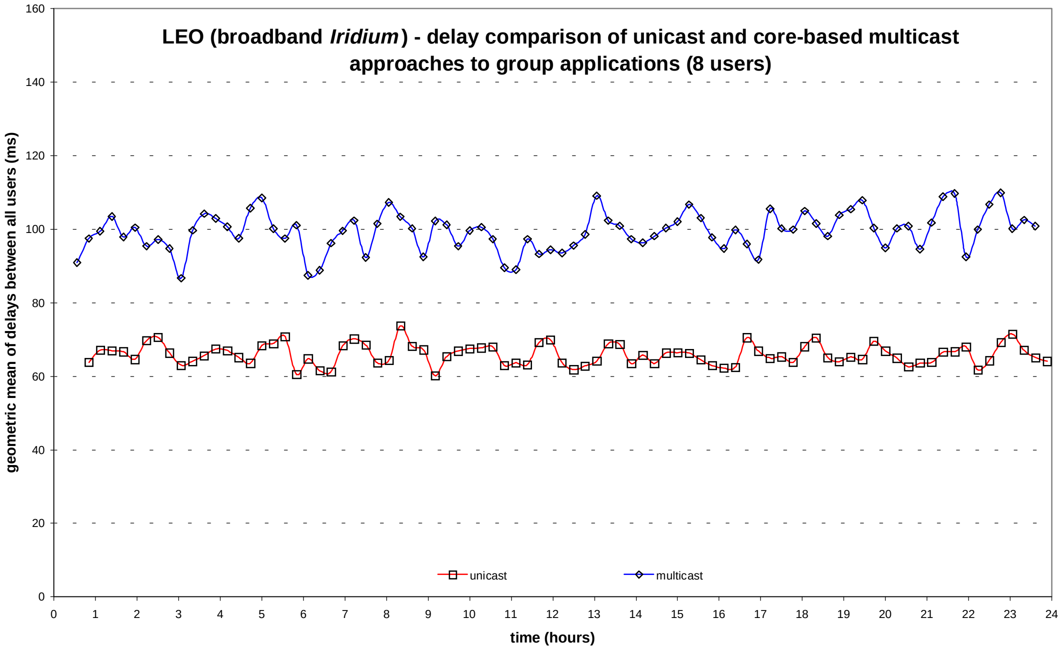


**Figure 4.8 - comparison of mean group delays for 8-user application over LEO**

Although the *Iridium* geometry uses a lower altitude for its fewer satellites (780km vs. 1400km for the Boeing *Teledesic* proposal), the overall star geometries are broadly similar. The smaller, less complex, LEO constellation is more tractable for simulation, while still giving an idea of group application performance at LEO. The simulation scenario for a group application over the broadband *Iridium* is shown in figure 4.6.

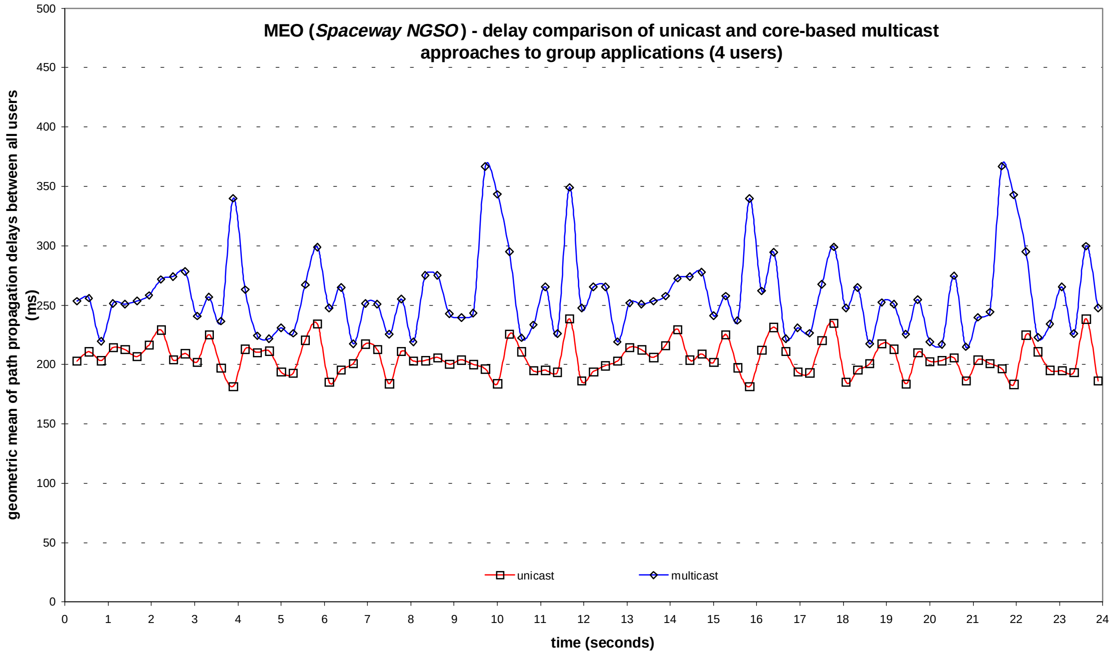


**Figure 4.9 - comparison of mean group delays for 4-user application over MEO**

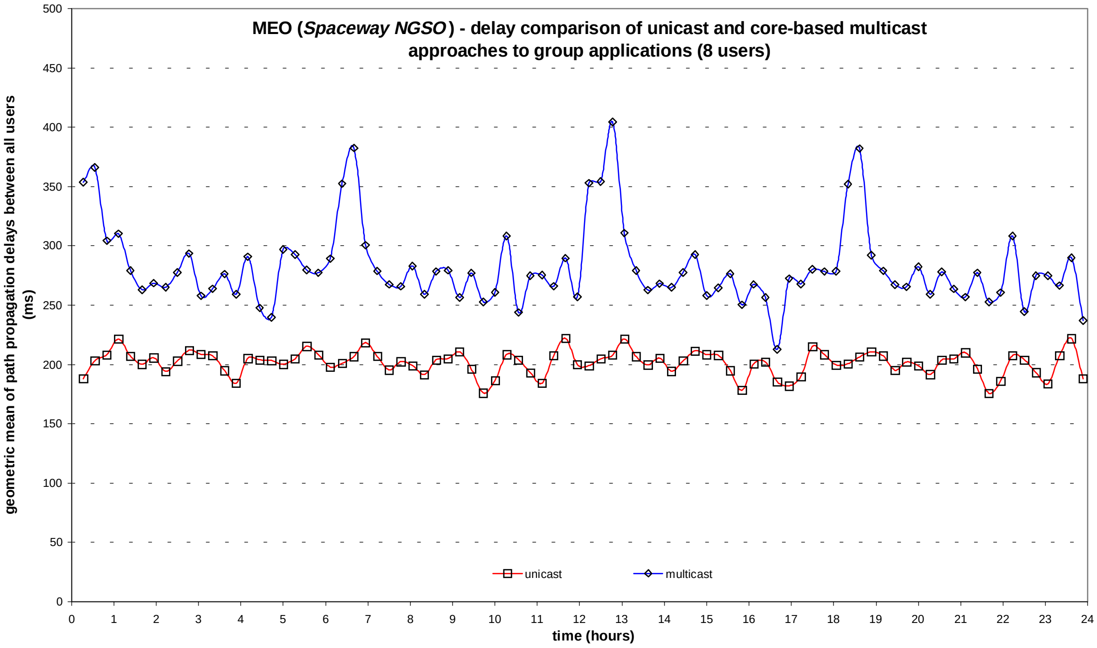


**Figure 4.10 - comparison of mean group delays for 8-user application over MEO**

Unicast and multicast use was simulated over these scenarios for the group of four users, and for a group of eight users where users at London, Tokyo, Quito and Sydney were added. Detailed results are given in Appendix 4, while summary results are given here for comparison. Figures 4.7 to 4.10 show unicast's delay advantage over a core-based multicast approach, where all communications must travel via the core.

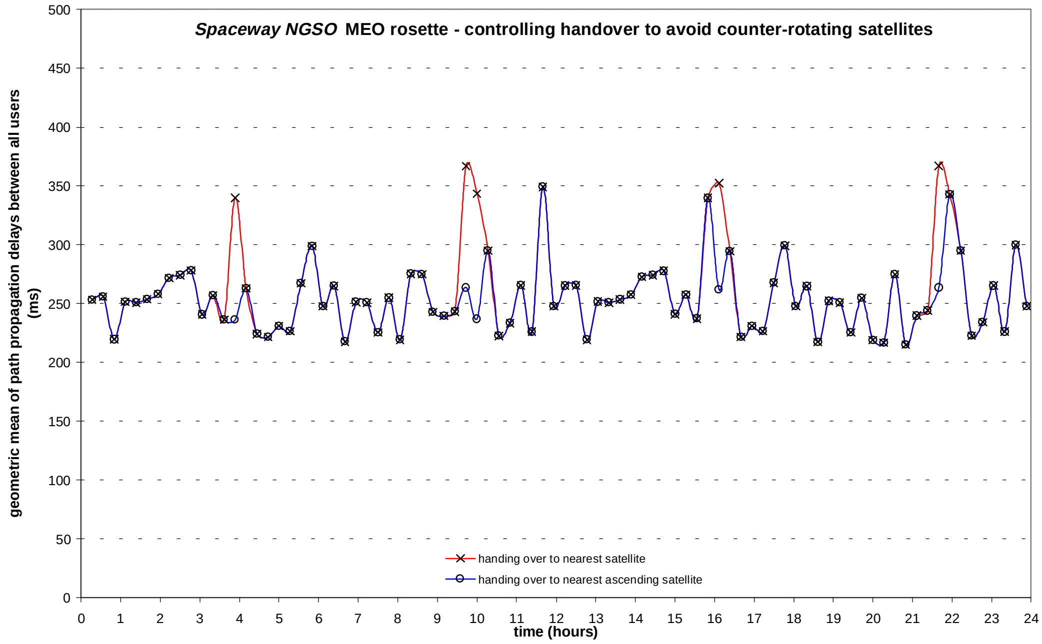


**Figure 4.11 - path rerouting affecting 4-user group delays in the MEO rosette**

We measured the path delays between all group members at regular points in time for the Earth's rotation over the course of a day, for both unicast shortest-path routing and multicast routing via the selected satellite nearest to the core location. The large spikes seen in the mean multicast delays for MEO simulations are due to ground terminals handing over between ascending and descending satellites, changing their positions in the ISL mesh and increasing their path distances to the core.

These increases in delay are visible in the detailed results presented in figures A3.6 and A3.8. In the smaller four-user group, a terminal will occasionally be forced to switch from use of an ascending MEO satellite and use a descending satellite to provide coverage, moving the terminal a long distance from the core and leading to a spike in the maximum path delay seen between group members. For the larger eight-member group, at least one terminal is always forced onto the descending surface, with the resulting permanently-high maximum path delay seen. This high maximum delay is, relatively speaking, much larger in proportion to the mean delays and to the MEO unicast delays than the maximum delays in the LEO constellation are, for results given in figures A3.1 to A3.4. The LEO star constellation has streets of coverage and a single ascending or descending layer of satellites that is used by each terminal. This was later examined by forcing handover to nearest *ascending* satellite whenever possible, rather than simply using the nearest satellite to the terminals or core location.

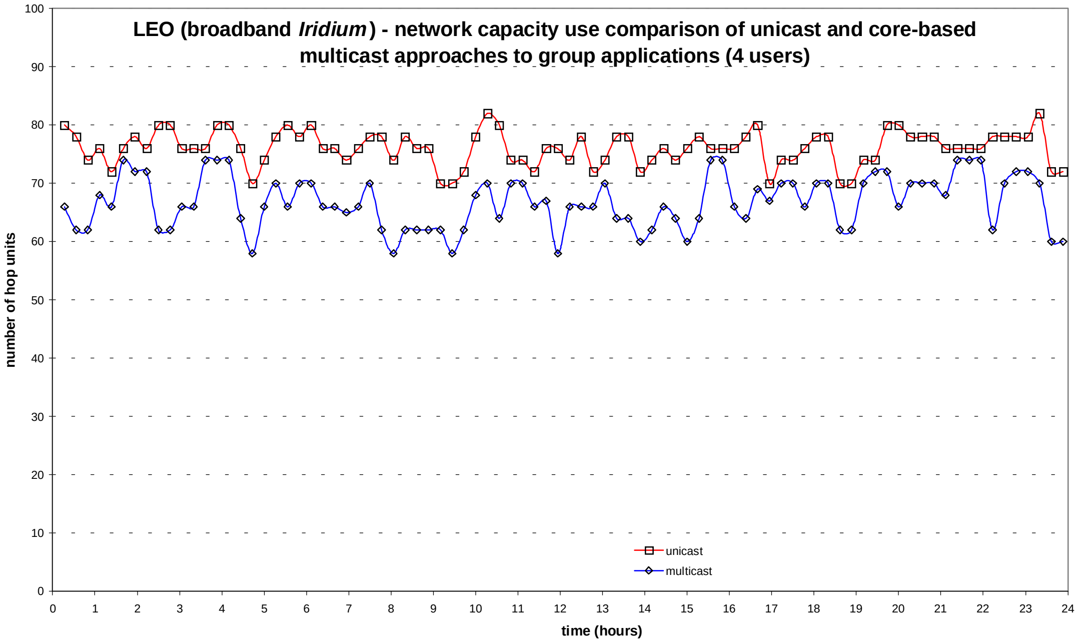


Figure 4.12 - comparison of capacity use for 4-user application over LEO

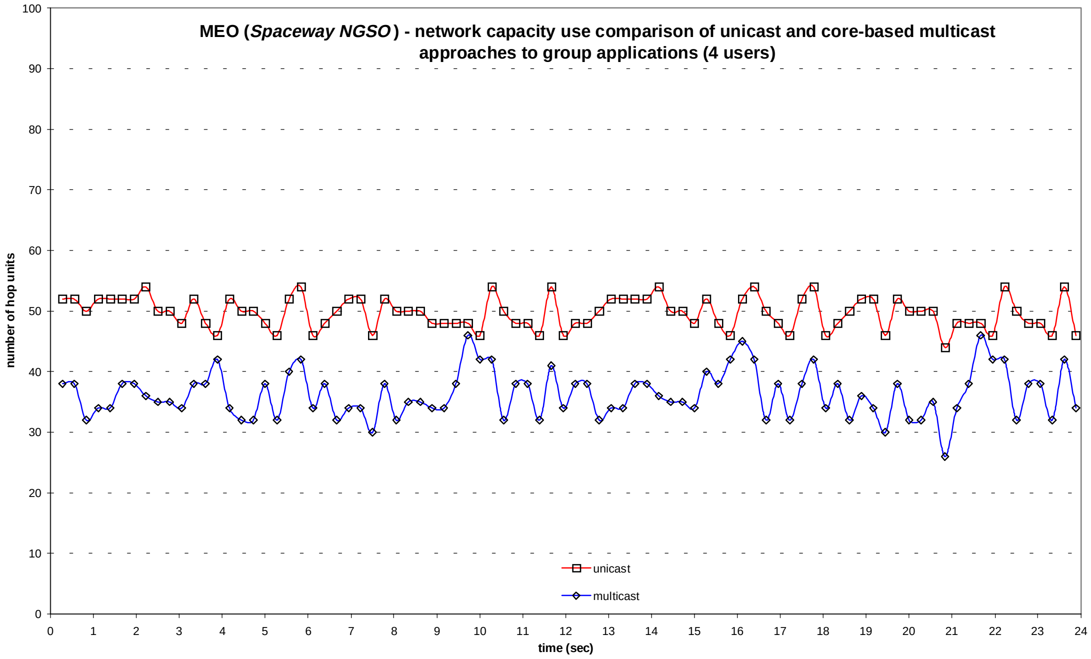


Figure 4.13 - comparison of capacity use for 4-user application over MEO

A comparison of these terminal handover strategies for the rosette, showing the decrease in mean group delay resulting from use of ascending satellites when possible, is shown in figure 4.11. Due to the *Spaceway NGSO* proposal's coverage, terminal use of counter-rotating satellites when nothing else is visible is unavoidable (as shown in Figure 1.6). This leads to rerouting of traffic along different paths, reflected in spikes that are visible in both mean delay curves in figure 4.11.

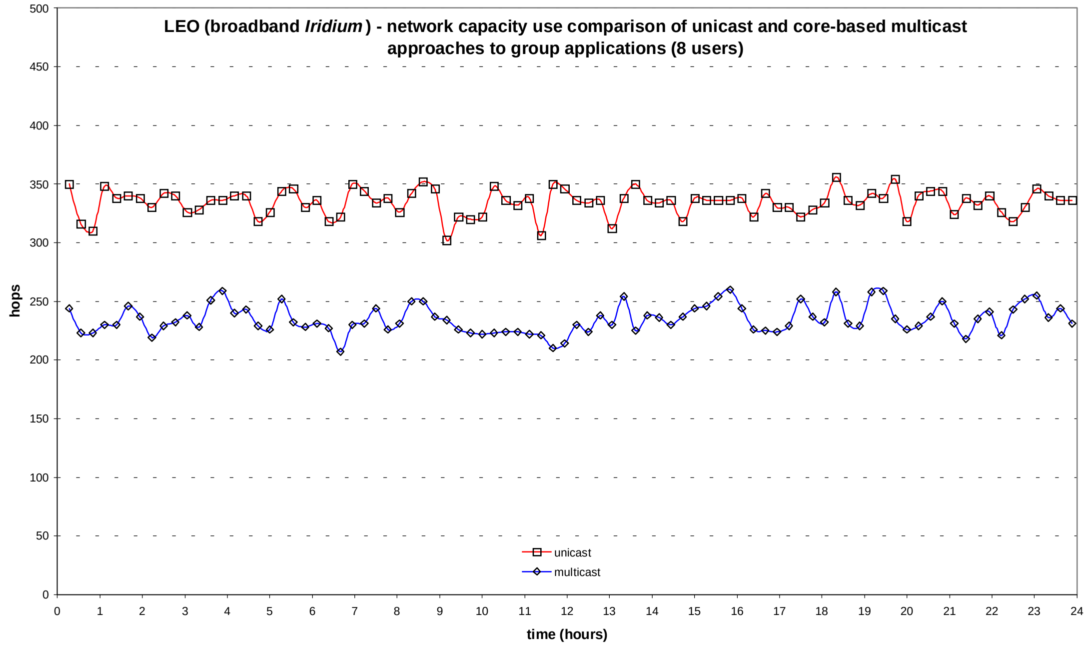


Figure 4.14 - comparison of capacity use for 8-user application over LEO

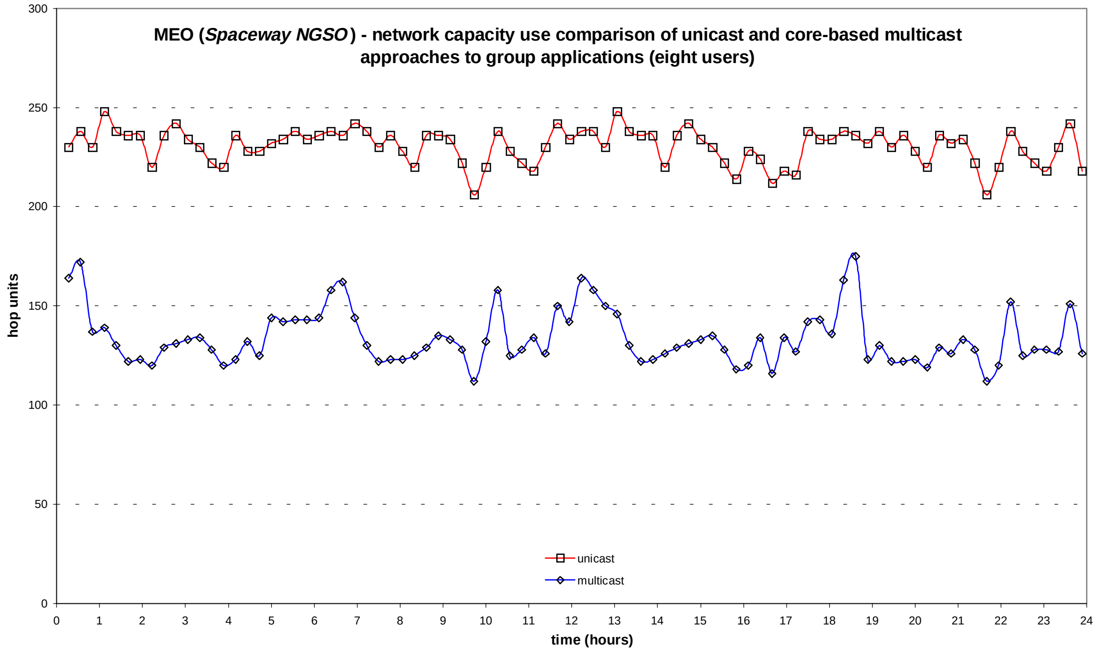


Figure 4.15 - comparison of capacity use for 8-user application over MEO

Use of multicast over unicast saves network capacity, as seen in figures 4.12 to 4.15. Here, the y axis represents *hop units*, where a hop unit is the use of a single link between two points by a single flow of traffic. Unicast traffic results in use of multiple hop units per shared link, one per group member, while multicast can decrease this to as little as two hop units per shared bidirectional link – one to the core, and one back.

Even given the added overhead on MEO multicast trees of individual terminals handing over to counter-rotating satellites, and the resulting longer paths between group members via the core, we can still see from these figures that multicast traffic requires less network capacity than equivalent unicast traffic for the same purpose. (We assume that network use is symmetrical and that all group members generate the same amounts of traffic.)

At LEO, with four users, the capacity savings are minimal, as traffic must be routed more indirectly to pass through the core and there is little overlap between paths used between terminals. As group size increases, overlap between unicast paths between members increases, leading to increased savings when multicast is implemented to reduce that overlap.

By simulating individual users, widely spaced, we are really showing the multicast capacity savings resulting from use of shared paths in the ISL meshes. This differs from any savings in the shared medium of the ground/space interface when terminals share the same spotbeam. In unicast, use of the ground/space interface depends on group size. The ground/space interface for each ground terminal taking part in the group application for eight-user unicast will have eight uplink and eight downlink streams, or use sixteen hop units. The multicast equivalent for a similar member terminal will have one individual uplink and one shared downlink stream for all local terminals in the same spotbeam or footprint, or two hop units' use. Capacity savings with multicast in the ground/space interface can be expected to be greater at MEO than at LEO, as MEO's larger footprints and spotbeams allow increased sharing of downlink capacity.

## 4.7 Capacity saving and the Chuang-Sirbu scaling law

We examined capacity savings in the ISL mesh in detail, by simulating multicasts between a varying number of ground terminals at a point in time. We increased the number of users for each simulation by adding another user at a major city. [**ChuangSirbu98**] examines multicast capacity savings in detail. It predicts that the amount of network capacity saved by use of multicast over the unicast equivalent is dependent upon the size of the group, and that the ratio of use is:

$$L_m/L_u = k \cdot \mathbf{N}^{0.8}$$

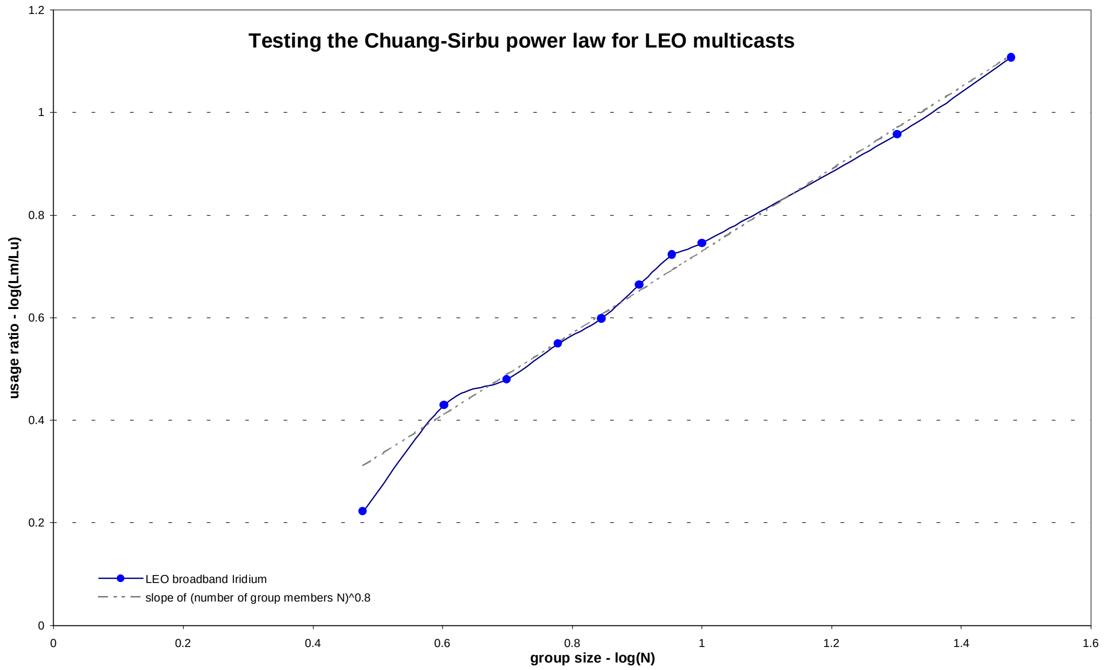


Figure 4.16 - multicast/unicast capacity gain for groups using LEO

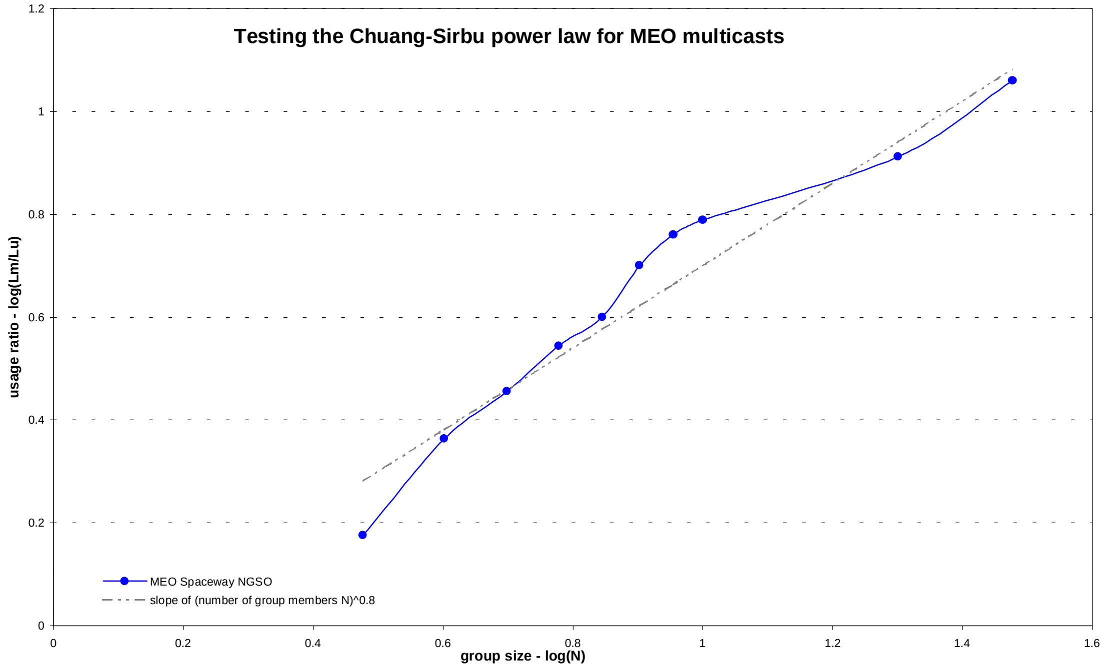


Figure 4.17 - multicast/unicast capacity gain for groups using MEO

Where: $L_m$ is the total number of hop units involved in use of the multicast tree;

$L_u$ is the number of hop units involved in the equivalent unicast flows;

$k$ is a proportional constant;

**N** is the size of the multicast group membership.

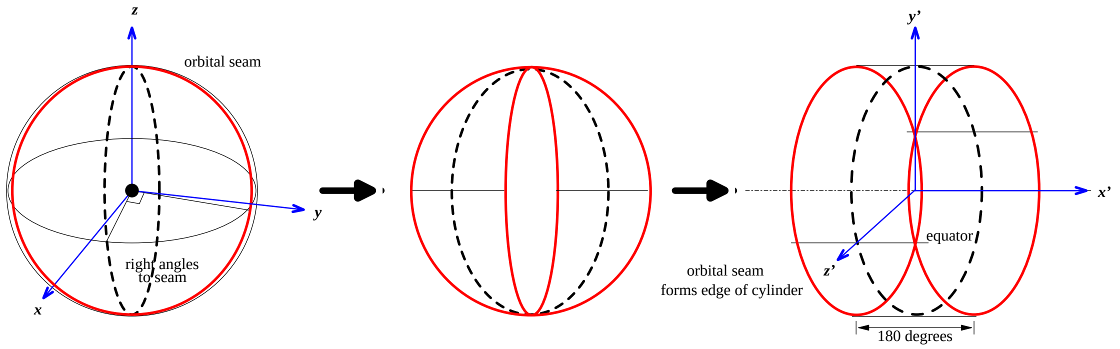
The applicability of this scaling law to the capacity savings in ISL meshes was tested by computing the number of hop units, or link-equivalent units, used by both multicast and unicast applications for a given group size at a point in time. This was then graphed against the group sizes, as shown in figures 4.16 and 4.17, showing good agreement with this law. Removing the unused parts of a mesh network creates a tree network, and the degree of each node is within the bounds considered by Chuang and Sirbu, so this seems unsurprising in retrospect.

By considering a small number of users, widely spaced, where ground/space capacity was not shared, we did not see saturation resulting in a levelling-off of the curve for large groups, as described in [**ChuangSirbu98**]. [**ChalmersAlmeroth00**] criticises the scaling law for considering the total number of endhost receivers to be equivalent to the number of last-hop routers, which leads to this saturation as the number of receivers exceeds the number of individual broadcast subnets in the network. At that point, the capacity advantage of multicast over unicast is obvious.

The Chuang-Sirbu law provides a useful rule of thumb for the capacity savings gained from using multicast for small multicast groups. [**PhilShenTang99**] gives a more complete analysis of the general network properties involved, and derives, from first principles, a more complex formula that provides very similar end results.

## 4.8 A case for a variation on the vector algorithm

In using a vector coordinate system with $x$ and $y$ planes that are in the plane of the Earth’s Equator, and in lacking explicit detailed knowledge of the constellation network’s geometry, the simple algorithm already discussed has a tendency to place the core around the Equator in-between ground terminals. It does not move the core near the pole when that would be more appropriate. In particular, this makes the algorithm less useful for star constellations without cross-seam links or for terminals that are widely separated. However, the simple optimisations to move the core to the satellite on the other end of a ground-to-space link from a ground terminal, when $n$=2, allow the algorithm to directly use shortest-path routing from the underlying unicast routing protocol when the network is being used for transit only. Those optimisations are particularly useful when the constellation network is merely being traversed to reach distant group members on other external networks.



**absolute coordinates** **twisted cylinder** **cylindrical coordinates relative to orbital seam**

**Figure 4.18 - transformation of coordinate systems to cylindrical**

The optimisations remove some of the worst core placement/terminal separation cases.

To handle the positioning the core relative to the orbital seam of the star constellation without cross-seam links well, a change in coordinate systems was found to be necessary.

As the orbital seam moves around the earth, the position of the core must be regularly recomputed and moved relative to the orbital seam. The coordinate system in which the vectors are summed must be relative to the orbital seam, rather than one representing absolute position. To do this, a transformation to cylindrical coordinates is used. The cylinder's central axis (the new $x$ plane, or $x'$) lies parallel to the Equator.

The cylinder's horizontal axis, on which the new $y'$ and $z'$ planes lie, is perpendicular to the Equator at 90° to the seam between counter-rotating planes, through the centre of the constellation network. The choice of axes is arbitrary and chosen for programming convenience, but lefthandedness of rotation between the axes should be preserved.

It is also necessary to handle the twists in the cylinder at the highest latitudes. To do this, the orbital planes are untwisted. $x'$ represents the angle from the central great circle, removing bias in the coordinate system with respect to latitude.

Figure 4.18 illustrates this transformation.

After the core position is computed by vector summation in the new coordinate system using the method described earlier, the core vector is transformed back to the absolute coordinate system to give the core's position in terms of latitude and longitude. Explicit joins and leaves are handled roughly as described earlier. One significant difference is that the vector addition and subtraction must take place while transformed in the cylindrical coordinate system.

### 4.8.1 Description of the seamed algorithm

1. As before, let the number of interested group members be known, and represented by $m$. Let the number of ground terminal positions representing those $m$ group members be $n$, where $n$ is always less than or equal to $m$. Let the number of satellites talking to the $n$ terminals be $p$, where $p$ is always less than or equal to $n$. If $p$=1, the satellite currently talking directly to the $n$ terminals is immediately nominated as the core and is promoted to the highest level in the hierarchy. If $p$=2, the constellation network is merely being used to reach a remote network. Any satellite currently talking directly to a known terminal can immediately be nominated as the core and promoted, although the satellite with the largest local membership is preferable. If $p$>2, continue. For each ground terminal known to represent interested group members, take spherical coordinates ($\text{lat}_k$, $\text{long}_k$) of that terminal. Repeat for all $n$ terminals.

2. Convert each set of coordinates to a vector $\mathbf{v}_k = [x_k, y_k, z_k]$ directed out from the centre of the Earth as origin O. Transform each vector into an equivalent vector $\mathbf{v_k}' = [x'_k, y'_k, z'_k]$ representing the equivalent position on the cylinder. Multiply each vector $\mathbf{v_k}'$ by the scalar $s$ representing the $s$ group members at that position.

   (Again, an optimisation, useful for large groups, is to only consider the current coordinates of each satellite that the set of group members in the satellite footprint communicates with, and to use the satellite coordinates to generate an approximate vector with a magnitude representing the set. This assumes that the algorithm is run at regular intervals to update the core position to handle gradual movement of multicast members to succeeding satellites.)

3. Sum all the vectors for the group to get a resultant vector in cylindrical coordinates:

$$\mathbf{c}' = [x'_{\text{sum}}, y'_{\text{sum}}, z'_{\text{sum}}] = [\Sigma_k x'_k,\ \Sigma_k y'_k,\ \Sigma_k z'_k].$$

4. Transform $\mathbf{c}'$ back to spherical coordinates to give $\mathbf{c}$, and convert the direction of $\mathbf{c}$ into ($\text{lat}_c$, $\text{long}_c$) to determine the position of the core.

5. Find the satellite currently located nearest this position from known satellite positions, and nominate this satellite as the core of the multicast tree for that group. Repeat at regular intervals and nominate the new satellite as the core.

This entire algorithm must be repeated regularly to adjust for the movement of the seam. As the seam moves slowly, travelling through 15 degrees of longitude in an hour as the Earth rotates beneath the constellation, this calculation does not have to be done as often as the last step of choosing a satellite as core.

Adjusting the core location for dealing with members joining or leaving the multicast group can be dealt with as follows:

### 4.8.2 Handling new member joins for the seamed algorithm

A new member wishes to join the multicast group, and has sent a join message.

1. If $m$=1, do nothing further; the core need not move, since $n$ will become 1 or 2. If $n$=2, compute $\mathbf{c}_{m+1}$ as already described in the preceding section for **c**. If n>2, continue with this algorithm.
2. Compute $\mathbf{c}'$ as described in the preceding section (transformation to cylindrical coordinates to compute $\mathbf{c}'$ by vector summation). Due to movement of the orbital seam, the core vector cannot be cached as it was for the simple summation. The magnitude of $\mathbf{c}'$ will be a scalar representing the $m$ current group members.
3. Take spherical coordinates ($\text{lat}_{m+1}$, $\text{long}_{m+1}$) of the ground terminal for the new member joining the group (or, as an approximation, the coordinates of the satellite that the new member belongs to. If the terminal or satellite is already represented in the group, just increase its weighting to reflect its new member.)
4. Convert to a vector $\mathbf{v}_{m+1} = [x_{m+1}, y_{m+1}, z_{m+1}]$, of magnitude 1, heading away from the centre of the earth as origin O. Transform to $\mathbf{v}'_{m+1} = [x'_{m+1}, y'_{m+1}, z'_{m+1}]$. (If the satellite is already a member of the group, the information on the new member would increase the magnitude of that satellite's vector. We add the new member as magnitude 1, regardless of whether we use the ground terminal or satellite coordinates, since other members forming the previous satellite vector are already included in **c** and $\mathbf{c}'$.).
5. Let $\mathbf{c}'_{m+1} = \mathbf{c}' + \mathbf{v}'_{m+1}$
6. Transform $\mathbf{c}'$ to **c**, and convert the direction of **c** into ($\text{lat}_c$, $\text{long}_c$) to determine the position of the core. Take the satellite currently located nearest this position, and

nominate it as the core.

### 4.8.3 Handling member leaves for the seamed algorithm

A member of the multicast group wishes to leave the group, and has sent a leave request to the core. There are $n$ ground terminal positions currently involved in the multicast group.

1. If, before leaving, $n$=1, simply discard the group information at higher levels and end. If $n$=2, the core can be moved to the satellite currently above and communicating with the remaining member ground terminal. If $n$>2, continue with this algorithm. If, after leaving when the vector representations in the satellite and in the core have been modified, $p$=1 and $n$>1, nominate that single satellite in the group as the core. If $p$=2, nominate the remaining satellite with the largest membership as the core.

2. Compute **c** as described earlier (transformation to cylindrical coordinates to compute **c′** by vector summation). The magnitude of **c′** will be a scalar representing the $m$ current group members.

3. Take spherical coordinates ($\mathrm{lat}_m$, $\mathrm{long}_m$) of the ground terminal representing the member $m$ leaving the group (or, as an approximation, the coordinates of the satellite with which the member communicates).

4. Convert to a vector magnitude 1 $\mathbf{v}_m = [x_m,y_m,z_m]$ heading away from the centre of the Earth as origin O. Transform to $\mathbf{v}'_m = [x'_m,y'_m,z'_m]$.

5. Let $\mathbf{c}'_{m-1} = \mathbf{c}' - \mathbf{v}'_m$

6. Convert the direction of $\mathbf{c}_{m-1}$ into ($\mathrm{lat}_c$, $\mathrm{long}_c$) to determine the new core position. Take the satellite currently located nearest this position, and make it the core.

### 4.8.4 Evaluating the seamed algorithm

When taking the position of the seam into account for a star constellation without cross-seam links, the computed position of the core varies with the position of the seam over the course of a day. Figures 4.19 and 4.20 show the core locations this algorithm produces for the same ground terminals communicating as in figure 4.5, with the seam at different longitudes at different times of the day.

The movement of the core as the seam moves over the course of a day is shown in figure 4.21. Since the seam is symmetrical, the core movement repeats every 12 hours, or 180° of seam movement.

Here, the core position is altered dramatically whenever the orbital seam crosses a terminal position, leading to nine core positions as the seam crosses each of the nine terminal positions. Each core position is labelled with the latitude at which it becomes the position, as the Earth rotates eastward under the seam.

A more detailed analysis of this variation on the original algorithm requires that a decision be made in the simulation as to when the core position should be moved and recomputed; this was not implemented in the simulations presented here.

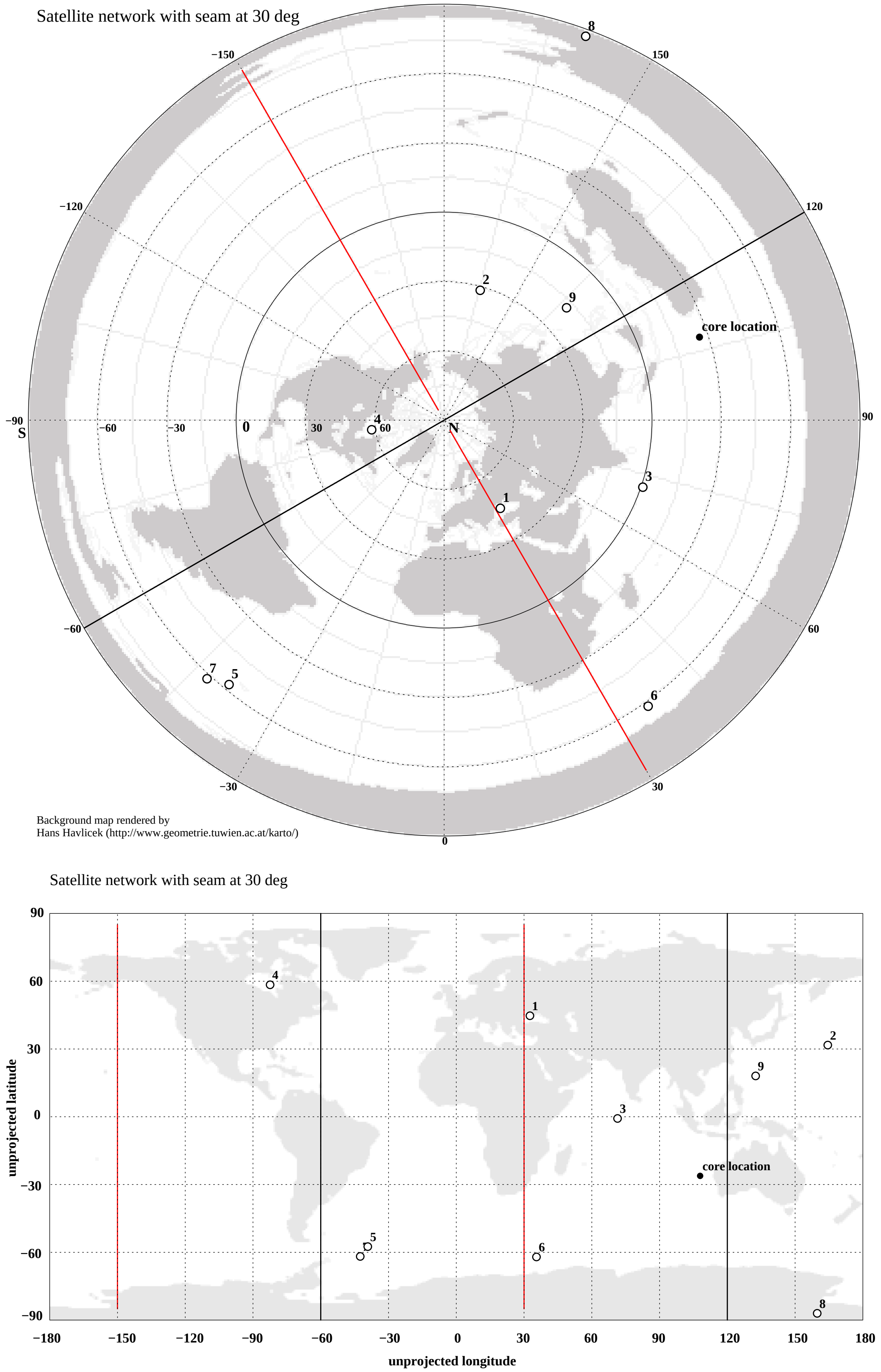
## 4.9 Updating and moving the core

Although computing the core location whenever a change occurs in the group membership is straightforward, moving the core around the network regularly is likely to be disruptive and to incur considerable network overhead. This is undesirable, particularly when the number of involved satellites is small and changing.

It is worth computing the difference between the currently-computed core position and the position that the currently nominated core satellite maintains as a result of handover, and to only update and move the actual core position if:

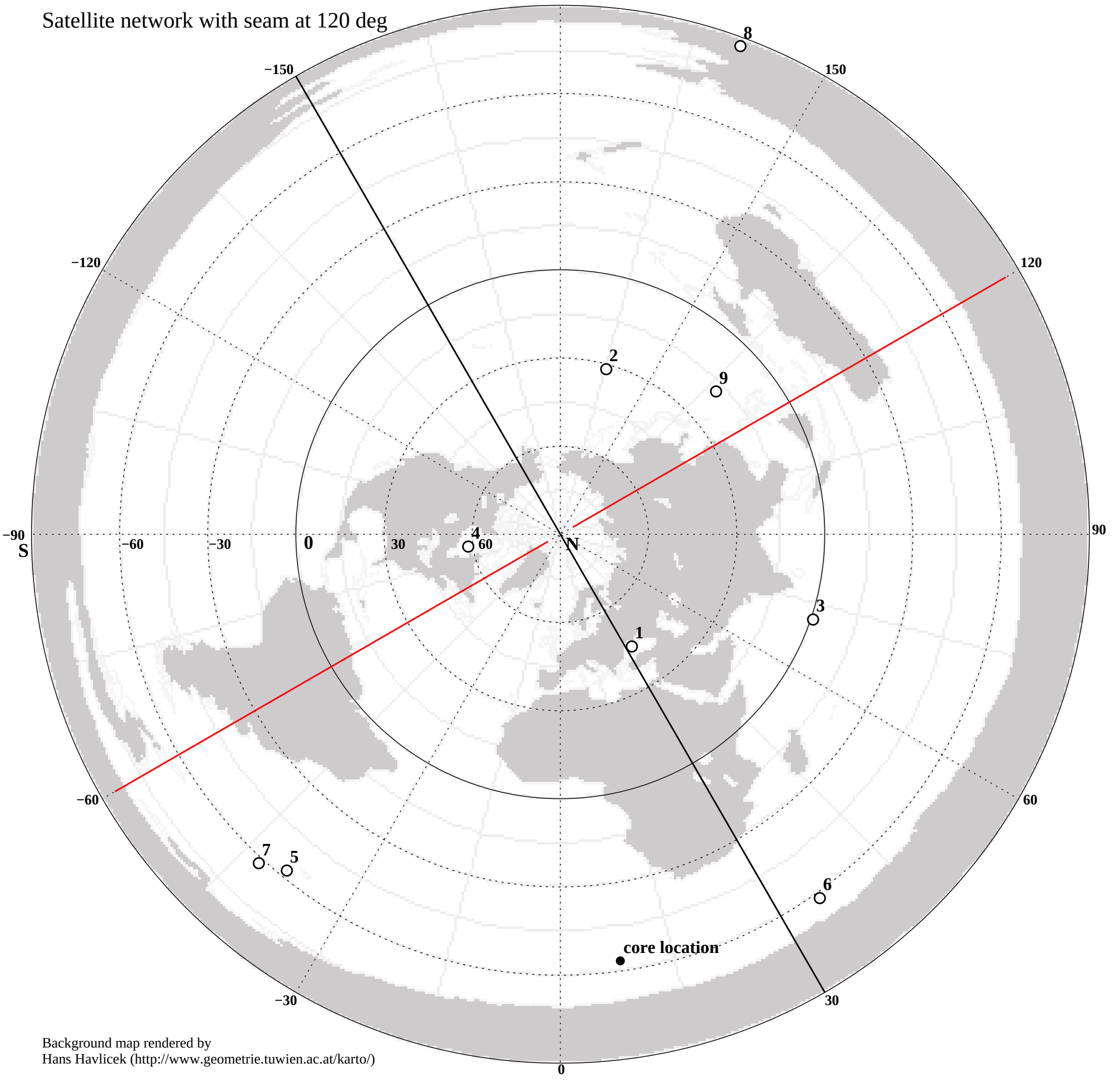
- the difference between positions (i.e. the dot product of the new computed and current core vectors) is significant and likely to remain so;
- group membership is stable, large and not expected to change significantly in future so that further position movements are not expected, requiring stochastic knowledge of the multicast application and its expected behaviour;
- enough time has elapsed since the previous core move; in the seamed constellation the core position might be updated every few hours;
- the needs of the group application (e.g. delay constraints) demand it.

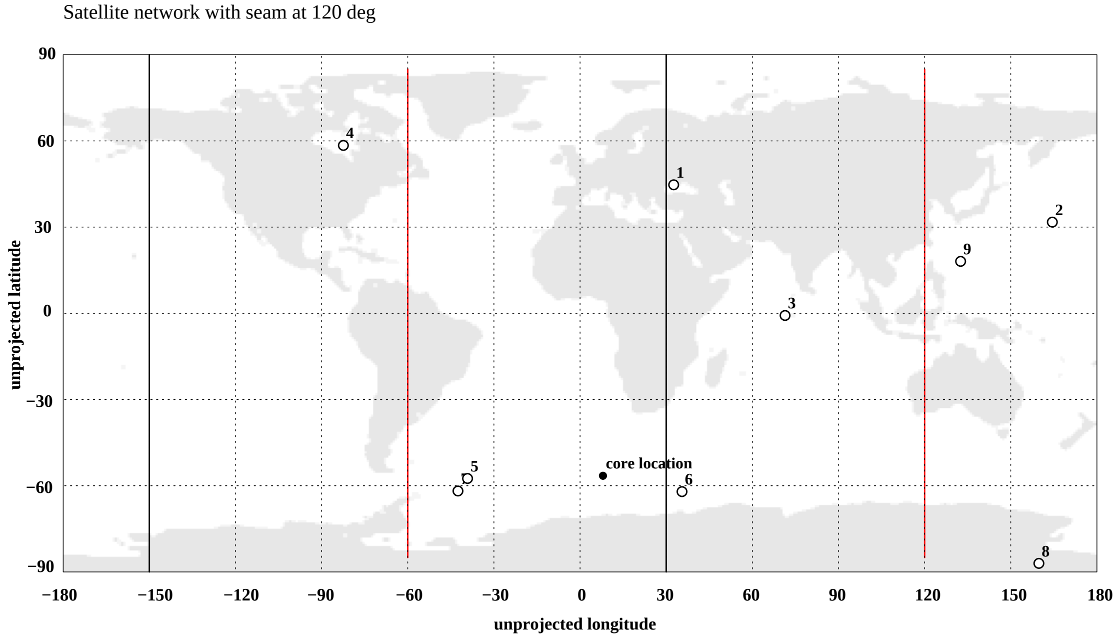
Deciding when the core should be moved is a traffic-engineering problem specific to the operation of a particular constellation network, and will affect the performance of any algorithm considerably.



latitudes of seam and **cylindrical Equator** are shown

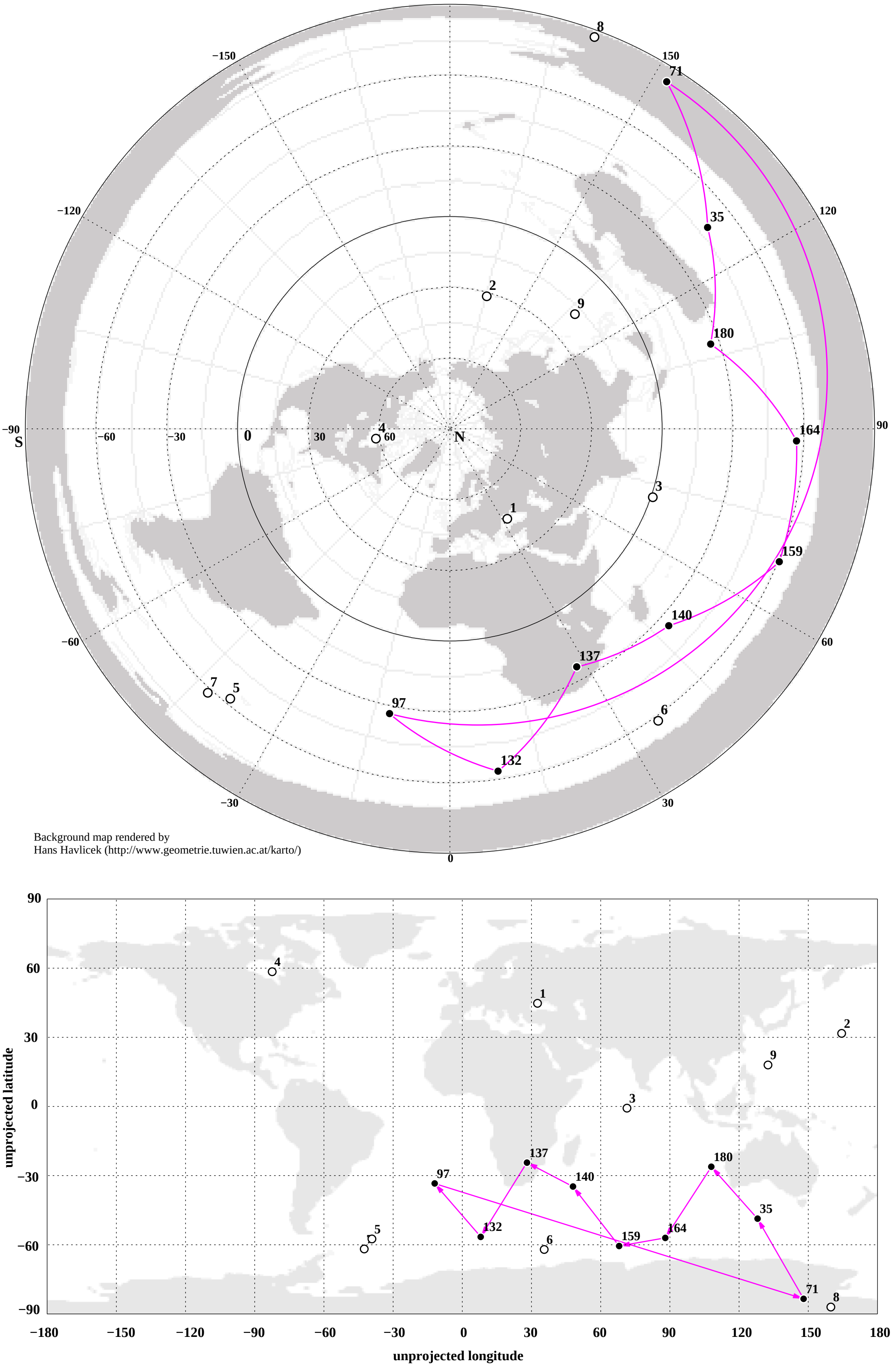
**Figure 4.19 - core computation for star constellation with seam at 30° latitude**





latitudes of seam and **cylindrical Equator** are shown

**Figure 4.20 - core computation for star constellation with seam at 120° latitude**



Background map rendered by
Hans Havlicek (http://www.geometrie.tuwien.ac.at/karto/)

core assumes each new position at the seam latitude noted by the core.

**Figure 4.21 - core movement as the seam moves, showing a twelve-hour loop**

## 4.10 Implementing IP multicast in commercial constellations

Given the maturity and increasingly widespread adoption of IP multicast protocols, there will be a demand for support for IP-multicast-aware group applications in wireless environments, including satellite networks.

Support for multicast within the broadcast environment of a single geostationary satellite with OBP is relatively straightforward, and can be addressed at the link layer. The network layer in the constellation network's space segment and the need to establish a tree for the multicast group within that make this less easy.

For a constellation network, support for a multicast application, e.g. satellite-based videoconferencing, could be accomplished by tunnelling the IP-multicast messages through the satellite gateways. However, this would require the setting up of multiple tunnelled virtual circuits between geographically-separate users to build a virtual network. This makes group management more difficult and uses more satellite capacity and limited ground-space air capacity than would be necessary if the satellites' on-board switches were to directly support IP multicast and packet replication.

At this point in time, no proposed commercial constellation appears to be in a position to support IP multicast directly. Proposed implementations of satellite constellation networks utilise internetwork tunnelling, and do not support multicasting.

Literature on *Teledesic* discusses a proprietary network stack, with fixed-length cells and connectionless routing [**Sturza95**]. This means that *Teledesic* would be tunnelling other protocols, including ATM, through its network from end to end, precluding easy native support for IP multicast via support for IP routing.

Each *SkyBridge* satellite is a simple transparent repeater connecting the terminal with a terrestrial gateway [**FCCSkyBridge97**], resulting in each satellite connection acting as a simple short one-hop ATM tunnel that is transparent to IP multicast. The problem of internetwork multicast is moved away from the space segment and back into the terrestrial networks that are utilising *SkyBridge* satellites for connectivity.

The LEO component of *Celestri* was intended to use cell relay routing, where each satellite carries a ATM switch [**FCCCelestri93**, p33]. *Celestri*'s GEO component supported broadcasting; this allowed the issue of implementation of support for multicasting in the moving *Celestri* LEO constellation to be effectively avoided. However, the latency of GEO discourages the use of many of the interactive group applications to which use of multicast would be appropriate.

Other networks, e.g. Hughes *Spaceway*, also plan an ATM-based approach, and can therefore be expected to inherit many of ATM's limitations regarding multicast.

## 4.11 Summary

There is a need for true multicast support in satellite constellations, to increase use of network capacity, to facilitate real-time group applications, and to make internetworking with the satellite constellation network as attractive as possible.

True multicast support is desirable, as is the network flexibility that multicast requires. Tunnelling is not desirable, as it increases the difficulty of implementing that network flexibility.

The way to enable true IP multicast support, and to leave network flexibility as open as possible from an Internet viewpoint, is to embed IP routing functionality, or a minimal subset of IP routing functionality, within every satellite that is a network node, and to enable satellites to replicate multicast packets when required. (How this can be done using MPLS is discussed in Chapter 5.)

This chapter has shown how the ordered core-based tree protocol can be adapted to the hierarchy of the constellation network. It has developed and presented a simple vector algorithm, using the regular topology of the network and its relationship with the surface of the Earth, to locate a core in the constellation network to produce a reasonable spanning tree. Evaluation of use of this algorithm has shown the delay overhead imposed by use of the core-based multicast, and the capacity savings resulting from the use of multicast. We have shown that the choice of ascending or descending satellite plays an important part in determining path delays across the rosette constellation (and we will return to this in Chapter 6). We have demonstrated that use of core-based multicast within the satellite constellation network saves network capacity over its unicast equivalent, and that that capacity saving obeys the Chuang-Sirbu scaling law.

# 5. *Implementing IP routing within the constellation network*

The high mobility of LEO satellites leads to a rapidly and regularly-changing network topology for the constellation, and raises a number of issues for the networking layer with respect to routing, particularly when considering adopting Internet protocols. Implementing IP routing for IP traffic is extremely desirable for reasons discussed here. However, it is necessary to avoid propagating IP routing updates between the separate routing realms of the constellation network and the Internet. There are a number of different approaches to implementing IP routing in the constellation network while achieving routing separation. Several approaches are examined here.

## 5.1 Satellite mobility and routing issues

Routing protocols, including terrestrial Internet routing protocols such as Open Shortest Path First (OSPF) [**RFC2328**] and Routing Information Protocol (RIP) [**RFC2453**], rely on exchanging topology information when network connections are established or changed.

A link-state routing approach has been identified as suitable for use in the constellation. The management of the constellation as a single entity and its fixed network size suggests that a link-state approach is very applicable [**Schacham88**].

In LEO constellations, this topology information quickly becomes obsolete and must constantly be refreshed and updated with new information. The overhead of regularly providing and updating this information is an obstacle to considering satellites as conventional Internet routers.

However, the topology of these constellations exhibits interesting and useful properties:

- predictability;
- periodicity in the space segment;

- regularity;
- a constant number of satellite nodes.

To perform routing in this highly dynamic yet tractable context, several strategies have been proposed:

### 5.1.1 Path maintenance via Virtual Topology Routing

The idea behind Dynamic Virtual Topology Routing [**Werner97**] is to exploit the periodic and predictable nature of the constellation topology.

Time intervals [$t_0$=0,$t_1$], [$t_1$,$t_2$], ... , [$t_{n-1}$,$t_n$=T], where T is the period, are chosen so that:

- Over an interval [$t_i$,$t_{i+1}$], the topology can be modelled as a constant graph $G_i$, i.e. link activation and deactivation take place only at discrete times $t_0$, $t_1$, ..., or $t_n$.
- The interval [$t_i$, $t_{i+1}$] is small enough to consider the costs of individual ISLs as constant over this time interval. The costs of these links could be computed from a function of inputs such as distance between the satellites, duration before link deactivation, geographic position, or other factors – assigning higher cost to high-latitude ISLs with a short time remaining before deactivation, for example.

Over these time intervals, the 'instantaneous' topology, $G_i$, is fixed. Optimal shortest paths and alternate paths can be established across the network graph between all pairs of satellites, using well-known methods such as the Dijkstra shortest-path algorithm. These optimal paths can be calculated in advance for the topology on the ground and then uploaded to all satellites via broadcast command.

It is also possible to add an optimisation procedure to choose among alternate paths between two satellites in order to minimise the number of satellite-to-satellite handovers required over the period [**WernerDelVogetal97**].

This path-based approach makes it possible to attempt to hide the mobility of satellites from virtual-circuit-based network protocols, such as ATM, that may be running over the constellation. This simplifies their view of the constellation and thus the routing complexity that they are exposed to.

### 5.1.2 The virtual node concept

The virtual node concept [**MaugRosen97**] aims to exploit the regularity of the constellation's topology. Again, the goal is to hide the mobility of satellites from routing protocols running over the constellation.

In this scheme, information concerning terrestrial constellation users, and how to communicate with them, is state that relates to a region of the Earth and is maintained in a fixed position relative to the surface of the Earth. Constellation users communicate with the virtual entity containing state pertaining to them: this is the virtual node. This virtual node is embodied at any given time by a satellite, and a virtual network of these nodes is embodied at any time by the satellite constellation.

As the satellites move and as terminals undergo handovers, state, such as routing table entries or channel allocation information, must be transferred continuously from one satellite to another; virtual node states passes between physical nodes. Routing is performed in the fixed virtual network, by using a common routing protocol.

Since we are considering carrying connectionless IP traffic in the constellation, we will consider this path-independent strategy further.

### 5.1.3 Strategies dependent on topology

These strategies use proprietary routing protocols that have explicit knowledge of the constellation topology and the satellite mobility. Such protocols require that there is always a path between two communicating ground hosts, and that routing is loop-free.

Each proprietary protocol will be very specific to the design of a certain type of constellation. The Footprint Handover Routing Protocol is a simple example of such a protocol for polar Walker star constellations [**Uzunaliogluetal97**].

## 5.2 ATM switching onboard satellite

Literature discussing ATM switching in satellite constellations is dominated by Markus Werner and colleagues. In their work, set-up of ATM virtual paths and virtual circuits over the intersatellite links is considered, rather than datagram routing. Network management is assumed to be centralised; this is appropriate for commercial use, while military use generally assumes decentralised management for battlefield reliability. The work takes advantage of the periodicity of the constellation.

It is assumed that interplane ISLs are maintained permanently [**WernerDelBurch97**]. Due to the swapping-over-of-interplane-neighbours that must always take place for each satellite at highest latitudes reached, and the need rapid slewing, this assumption may not hold for physical implementations.

In a cylindrical constellation with an orbital seam (*Iridium*, *Teledesic*), through-traffic load on satellites on the edge of the network, near the seam, is shown to be less than the load on centrally-placed satellites [**WernerKroMar97**].

[**WernerMaral97**] notes that there is little published work in the field of routing traffic over the dynamic network topologies of satellite constellations, and discusses adaptive routing based on metrics from cost computations applied to each link.

The purely ATM-based approach advocated in that work is, however, not the only possible method for implementing routing within a satellite constellation network.

## 5.3 Reasons for considering IP routing onboard satellite

Assuming that the constellation network has ISLs, and that it will be expected to carry IP traffic, there are a number of compelling reasons for wanting to support IP routing of that same IP traffic in the constellation network’s space segment:

### 5.3.1 Supporting IP multicast within the constellation network

Supporting multicast allows us to provide group applications while using less network capacity than would be required by the equivalent applications using multiple unicasts to communicate.

Without IP routing and native support for IP multicast, implementing any support for IP multicast becomes increasingly problematical, as the IP multicast group and tree will need to be projected with difficulty onto other network layers and routing paradigms, where IP multicast routing functionality must be duplicated.

The ability to support IP multicast by duplicating packets in the mesh of ISLs requires explicit knowledge of IP routing requirements and of the IP multicast tree in the network layer of the constellation.

### 5.3.2 Supporting IP QoS within the constellation network

The traditional network service on the Internet is best-effort datagram transmission. IP packets are sent from a source to a destination without any guarantee that the packet will be delivered.

For traditional two-way data applications that are elastic in nature in that they tolerate packet delays and packet losses, this best-effort model is satisfactory, and any necessary reliability can be implemented without redundancy at the end-points, e.g. via acknowledgements in TCP [**Saltzeretal84**].

However, the emerging real-time applications have very different characteristics and requirements to data applications. They are less elastic, less tolerant of delay variation and need specific network conditions in order to perform well. The Internet protocol architecture is being extended to provide support for real-time services by adding Quality of Service (QoS) models to meet these application requirements.

There are two architectures that are being defined in this context: *Integrated Services* (commonly known as intserv) and *Differentiated Services* (commonly known as diffserv). Support for IP routing within the satellite constellation would make it possible to support IP QoS via one of these architectures.

#### 5.3.2.1 Integrated Services

The primary goal of the Integrated Services architecture and QoS model is to provide IP applications with end-to-end ‘hard’ QoS guarantees, where the application may explicitly specify its QoS requirements and these will be guaranteed and met by the network [**RFC1633**].

For this to be accomplished, the Resource Reservation Protocol (RSVP) is used to signal the resource requirements of the application to the routers situated on the transit path between the source and destination hosts [**RFC2205**, **RFC2210**].

RSVP implements minimal quality of service guarantees for IP networks for multicast and unicast flows. It associates QoS information with the destination information in each packet’s network and transport headers. Packet headers are examined at each router along the flow’s path, to determine how to treat the packet. Each router must be RSVP-aware to be able to provide QoS to the flow.

As it uses information within the transport header within the packet (UDP/TCP port information), RSVP also has the problem of being invisible inside the tunnelled IP packet in non-IP networks, again requiring interpretation of Quality of Service (QoS) and translation between different networks' QoS at gateways. This places more load on those gateways.

Although RSVP can be deployed on small private or access networks, it is recognised that RSVP does not scale well to large backbones or large-capacity connections such as the intersatellite links considered in a data network constellation. This is due to the management overhead required, and aggregating individual RSVP flows at gateways is suggested to reduce this overhead [**RFC2208**]. This aggregation will certainly not scale well to a multiple-multicasts scenario.

The Integrated Services architecture supports two new classes of service, in addition to the existing best-effort class:

- The *Guaranteed Service* guarantees both delay bounds and capacity availability, setting a maximum queuing delay.
- The *Controlled Load Service* approximates the end-to-end behaviour provided by best-effort service under unloaded conditions. The network ensures that adequate link capacity and packet processing resources are available to handle the requested level of traffic.

The major drawback of Integrated Services is that the amount of state information, which must be maintained at each node, is proportional to the number of application flows. Resource requirements must be negotiated over a set path or a set spanning tree, where the routers in the path or tree maintain soft state pertaining to the flows passing through them. Support for the regularly-changing paths resulting from satellite mobility would act to invalidate RSVP guarantees, as the path between endpoints for which an RSVP guarantee is set up does not remain constant. RSVP renegotiation when this happens would be extremely undesirable. Maintaining RSVP's QoS guarantees for flows during handover and route changes would be a difficult problem.

RSVP is not suited for deployment on high-capacity backbones or on transit networks due to its reliance on per-flow state and complex per-flow and per-packet processing. Aggregation of flows will be necessary to make large-scale RSVP tractable.

#### 5.3.2.2 Differentiated Services

The Differentiated Services (diffserv) architecture has been proposed to overcome perceived limitations of Integrated Services [**RFC2475**]. Diffserv allows IP traffic to be classified into a finite number of priority and/or delay classes. Traffic classified as having a higher priority and/or delay class receives some form of preferential treatment over traffic classified into a lower class.

The Differentiated Services architecture does not attempt to give explicit ‘hard’ end-to-end guarantees. Instead, at congested routers, the aggregate of traffic flows with a higher class of priority has a higher probability of getting through. Traffic with a marked delay priority is scheduled for transmission before traffic that is less delay-sensitive.

The class-of-service (CoS) information needed to perform actual differentiation in the network elements is carried in reserved bits in the Type of Service (TOS) field of the IPv4 packet headers, or the Traffic Class field of the IPv6 packet headers. This is referred to as the differentiated services codepoint (DSCP) [**RFC2474**].

Since the information required by the buffer management and scheduling mechanisms is carried within the packet, no complex per-flow signalling protocols are required. As a result, the amount of state information, which is required to be maintained per node, is only proportional to the small overall number of service classes and is not proportional to the large number of application flows.

The Differentiated Services architecture is composed of a number of functional elements, namely packet classifiers, traffic conditioners and per-hop forwarding behaviours (PHB). A PHB describes the forwarding behaviour of a differentiated services node that is externally visible, as it is applied to a collection of packets with the same DSCP that are traversing a link in a particular direction.

Each service class is associated with a PHB. A per-domain behaviour (PDB) is a collection of packets with the same DSCP, thus receiving the same PHB, traversing from edge to edge of a single diffserv network or domain. (The PDB was previously known as a Behaviour Aggregate, or BA.)

PHBs are defined in terms of behaviour characteristics relevant to service provisioning policies, and not in terms of particular implementations. PHBs may also be specified in terms of their resource priority relative to other PHBs, or in terms of their relative observable traffic characteristics. These PHBs are normally specified as group PHBs and are implemented by means of buffer management and packet scheduling mechanisms. A number of PHBs are being standardised [**RFC2597**, **RFC2598**].

According to the basic differentiated-services architecture definition, these elements are normally placed in ingress and egress boundary nodes of a differentiated-services domain and in interior DS-compliant nodes. However, it is not necessary for all the elements to be present in all the DS-compliant nodes; that depends on the functionality required at each node.

At each differentiated services user/provider boundary, the service provided is defined by means of a Service Level Specification (SLS). The SLS specifies the overall technical performance and features that can be expected by a customer or by another provider network, and forms part of the overall legal Service Level Agreement (SLA) with the latter.

#### 5.3.2.3 An implementation of IP QoS in the constellation

The satellite constellation network is effectively a high-bandwidth mobile backbone, and the Integrated Services model simply does not scale for use within the constellation. Aggregation of RSVP-specified flows would be extremely complex to implement, and mobility would be difficult to overcome even with use of virtual nodes.

Internetworking considerations dictate support for RSVP applications at the edges of the network in the border routers at the gateway earth stations, while supporting the more scaleable differentiated-services model within the constellation.

From a differentiated-services viewpoint, the satellites in a constellation can be expected to be mass-manufactured and identical, with identical routing functionality. The constellation can be considered as both a single diffserv domain and as a single diffserv region, where a common, single, set of per-hop behaviour (PHB) groups is implemented within the routers in every satellite and in every ground station. PHBs specific to the capabilities of the satellite constellation can be defined.

Marking the DSCP in order to select an appropriate PHB will be carried out at the border gateway routers, which are also diffserv ingress and egress boundary nodes that will undertake any necessary traffic conditioning.

As a single region, domain and PHB behaviour set, there should be no need for additional PHB mapping within the constellation between satellites. This will greatly simplify meeting service-level specifications in comparison to other, less homogeneous, diffserv-capable networks.

## 5.4 Overcoming objections to IP routing onboard satellite

The preceding section has given reasons why implementing IP routing onboard satellite is extremely desirable – for IP multicast and for IP QoS.

This section discusses reasons often stated for not considering IP routing onboard satellite, and explains why these reasons are not insurmountable.

### 5.4.1 Variable-size IP packets

A common misconception is that, as the satellite air interface must allocate channel capacity in some predefined manner via FDMA/TDMA, fixed datagram sizes are needed to fit neatly into the frame structures for the allocated slots in the wireless channel. As IP packets are of variable length, an objection to the use of IP is raised.

It is possible to fit IP packets into any fixed-length frame structure by the use of either:

- explicit visible IP-level fragmentation, where the packet is broken into sections at the internetwork layer. Each section has a fragmentation identifier (ID), small enough to transmit across the fixed-length interface as IP packets themselves.
- implicit transparent lower-level fragmentation, where the IP packet is broken up in order to be carried by a MAC-level or a tunnelling protocol. Padding can be used where appropriate to fill up frame structures not completely filled with all or part of an IP packet payload.

IP-level fragmentation is generally undesirable, and its occurrence can be minimised by use of path message transfer unit (path MTU) discovery [**RFC1191**] and the setting of a maximum MTU size for IP packets. Although use of path MTU discovery with IPv4 and TCP has implementation problems [**RFC2923**], its use is mandatory with IPv6 [**RFC1981**].

### 5.4.2 Routing table management

Routing table size and complexity is often cited as an obstacle to performing IP routing onboard satellite.

For performance equivalent to terrestrial equipment, space-qualified computing hardware is generally considerably more difficult and more expensive to produce, and satellite computing performance (and thus routing performance) can be expected to lag behind equivalent terrestrial performance at any point in time.

Given the expense and difficulty of launching satellites, a long satellite lifetime is desirable. Once a satellite is launched it cannot be upgraded for the duration of its expected lifetime, meaning that the satellite performance can be expected to fall increasingly behind terrestrial performance, and must be designed with a margin to meet expected needs at the end of the satellite lifetime. This is a clear argument for placing as much as possible of the complexity of the satellite constellation network in the ground segment in order to future-proof it. This argument can be taken to the extent of limiting the space segment to nothing above the data-link layer and rejecting use of ISLs, in order to make the space segment as flexible as possible to meet changing terrestrial needs.

The space environment, with radiation and temperature variations, is harsh on processors. This limits available processing capacity in comparison with equivalent terrestrial Internet routers. The available power budget, from on-board batteries recharged by solar cells, is also limited.

Any one LEO satellite is unlikely to be able to hold information on how to route towards all of the necessary connected networks in the world. Handing its entire routing table for the Internet over to the next satellite, as the satellites move and assume new virtual nodes or management of earth-fixed cells, is harder still. This creates a scalability problem for on-board routing tables and processing. This is the problem commonly cited by people with an understanding of terrestrial Internet routing protocols. The constellation would be overwhelmed with information about the terrestrial Internet. The satellites can be seen to function better if they do not need to know about terrestrial Internet addresses or about terrestrial routing.

The movement of the onboard IP routers in the non-geostationary satellite constellation can also create a similar scaling problem if information concerning their continuous motion is propagated continuously as routing updates within the terrestrial Internet. The terrestrial Internet functions well without knowing about the motion of satellites in the space segment.

In both cases, the problem is not with IP routing, but with full integration and the rate at which large amounts of routing information must be updated for both the terrestrial Internet and the constellation network. Keeping routing updates from propagating from one to the other is a way to prevent these problems.

There is also a secondary problem with the growth of backbone routing tables as the Internet increases in size. This continual growth in routing table size means that the tables might eventually outstrip the availability of the satellites' onboard routers to hold them, leading to loss of service to parts of the terrestrial Internet. Field upgrades to e.g. upgrade processors or add more table memory are not feasible for satellites; again, excess table capacity must be designed in for the satellites' lifetimes, in order to meet the expected size of routing tables at the end of the satellites' expected lifetimes before replacement. This excess capacity is undesirable from an engineering viewpoint, as it relies on matching the expected satellite lifetimes with uncertain projections of routing table sizes. The satellite network itself is relatively fixed in size, requiring a fixed-size table; removing Internet-related information removes this routing problem.

In all cases, the limited ground-space capacity also benefits if routing updates do not need to be unnecessarily propagated across it, and the amount of routing information and state held in the satellites, and resulting size of the routing tables, is minimised.

These goals can be achieved by separating and isolating satellite and Internet routing updates to their respective routing realms, using one of the methods discussed later in section 5.5.

### 5.4.3 Speed of routing vs. switching

IP routing involves examining packet headers for a global destination address and then executing a table lookup for the correct forwarding action to take for the packet, rather than simply transmitting the packet from one switched interface to another. As a result, IP routing is generally perceived as requiring more processing power than, and being slower in operation than, simpler switches that do not need to consider anything beyond the local interfaces and that do not need to examine global addresses in headers or perform complex lookups.

As on-board processing capabilities are constrained by the limitations of the space environment, this is often cited as a reason why IP routing is not suitable onboard satellite.

However, continual advances in processing power, better lookup algorithms and the move from simple bus-based routers to crossbar and shared-memory switching fabric designs have acted to improve IP routing performance [**KeshavSharma98**]. This raises the bar (to coin a pun) on what is feasible for onboard processing.

Work on ‘shortcut’ IP switching techniques such as MPLS demonstrates increased throughput and routing performance with shorter queuing delays and fewer local state overheads [**RFC3031**].

## 5.5 Approaches to separating routing

The approaches to separating routing in the constellation network discussed here are:

- *tunnelling* over another network protocol or over IP in the constellation network.
- *network address translation* (NAT) using twice NAT, which separates internal and external IP addressing and routing;
- *exterior routing protocols*, where border routing is managed at the edges of the network using e.g. the Border Gateway protocol (BGP), in order to use different internal routing protocols or network layers while controlling internal and external propagation of routing updates.

Combining IP routing of IP traffic with routing of other types of network traffic is then discussed.

## 5.6 Tunnelling approaches

With tunnelling, end-to-end virtual-circuit communications are supported well, and clear separation of the tunnelling network layer and the external network prevents communication of routing updates between the two.

A tunnelling network layer must be selected to carry the internetwork's IP communications. The common alternatives are briefly described below.

### 5.6.1 IP over ATM

Many satellite constellation operators and manufacturers have focused on ATM as the network protocol for the constellation, but with use of proprietary ATM signalling protocols and MAC layers. Support for ATM and interworking with ATM networks is a commercial goal.

ATM virtual path connections (VPCs) can be maintained between all pairs of satellites, using for example the Dynamic Virtual Topology Routing concept that is discussed in Chapter 5. All the ATM virtual channel connections (VCCs) that share the same pair of satellite entry and exit points can be aggregated into the same VPC. Switching is done onboard only according to the VPC label.

If an ATM service is provided to interconnect two constellation users, IP packets can be tunnelled and carried by ATM cells, using e.g. classical IP-over-ATM encapsulation [**RFC2225**].

### 5.6.2 IP over a proprietary protocol

A proprietary network layer and routing protocol can be specifically optimised for the constellation. Such a protocol can avoid transmitting unnecessary routing information, while propagating other useful network-specific information such as internal delay, expected traffic load or instantaneous traffic load.

This appears to be the approach adopted by the early proposed *Teledesic* designs. The overhead of reassembling IP packets from custom protocol frames at each satellite just so that IP header processing can be done would be undesirable.

### 5.6.3 IP over IP

It may sound curious to suggest the tunnelling of IP over IP [**RFC1853**] in the constellation network, but this approach does allow separate addressing, separate routing realms and avoids propagation of routing information between the constellation network and the terrestrial Internet. This approach has the advantage of using existing IP routing protocols, with the possibility of relatively straightforward support for IP routing features such as IP multicast or IP QoS, unlike tunnelling over non-IP protocols. Implicit fragmentation is still likely to be required for the wireless links.

However, IP-in-IP encapsulation imposes a header overhead. One of the few advantages of NAT, discussed in section 5.7, is that NAT avoids this header overhead.

A second disadvantage of supporting only IP routing in the constellation is the tunnelling of non-IP communications over the IP layer. This has unacceptable overheads e.g. for ATM traffic, and an alternative approach is necessary if the constellation is to support more than just IP.

### 5.6.4 Tunnelling in the satellite constellation

To isolate the constellation network’s routing realm from external networks and from external routing realms, or to send IP traffic across a constellation network where IP routing is not implemented or is not supported in the network layer, tunnels can be used to link IP-capable entities on the ground across the network, namely:

- isolated ground hosts using the constellation for connectivity to the terrestrial Internet or to other isolated ground hosts or networks.
- small routers using the constellation network to interconnect a local area network (LAN) with the terrestrial Internet or to other ground hosts or networks.
- large border gateways interconnected with the rest of the Internet, through which traffic from the previously-listed entities would travel to reach the terrestrial Internet.

The network topology seen from the IP level is as illustrated in figure 5.1. The border gateways, routers, and ground terminals, shown in black, are where tunnelling would occur. As tunnels must be established between all pairs of hosts, these IP ground entities create a fully connected graph, thus creating a virtual network across the satellite network.

For destinations outside the constellation, small routers and border gateways can reach the egress border gateway in one hop over the constellation, whereas isolated ground constellation users need to send their packets to an IP routing entity (e.g. a border gateway), that will tunnel the packet to the egress point.

One significant difference between tunnelling across the constellation network, and the use of tunnelling described earlier for the MBone and 6Bone, is that for the constellation network the tunnelling is implemented as a permanent, rather than a transitional, measure, with no future benefits for the lifetime of the constellation.

### 5.6.5 Constellation Address Resolution Servers

To send a packet through a tunnel from one edge router to another, it is necessary to know the constellation address (the address in the constellation realm) of the communicating peer on the other side of the tunnel.

This constellation address could be pre-configured in the tunnelling entity, but since the number of possible peers is potentially very large and the virtual network is a fully-connected graph, this approach does not scale well with size or adapt well over time as new constellation user networks join the constellation. An on-demand strategy for retrieving these constellation addresses appears more reasonable.

On an Ethernet local network, an IP entity that wants to send a packet to another local IP entity first needs to retrieve its Ethernet address using an Address Resolution Protocol (ARP) [**RFC826**].

Here, we can use a similar strategy, except that we do not resolve the address by broadcasting a request, but by interrogating a Constellation Address Resolution Server (C-ARS), in a similar fashion to ATM ARP [**RFC2225**].

All that a ground IP edge router needs in order to communicate with other IP hosts at the edges of the constellation’s autonomous system is the destination IP address, and the constellation address of a C-ARS. This information would be situated in a gateway station that ideally also controls the satellites and has detailed knowledge of the satellite constellation.

To avoid concentrating all the address resolution traffic around a single C-ARS and to provide redundancy, multiple peered C-ARS servers are situated in other ground stations, communicating network updates to one another. This is shown in figure 5.1.

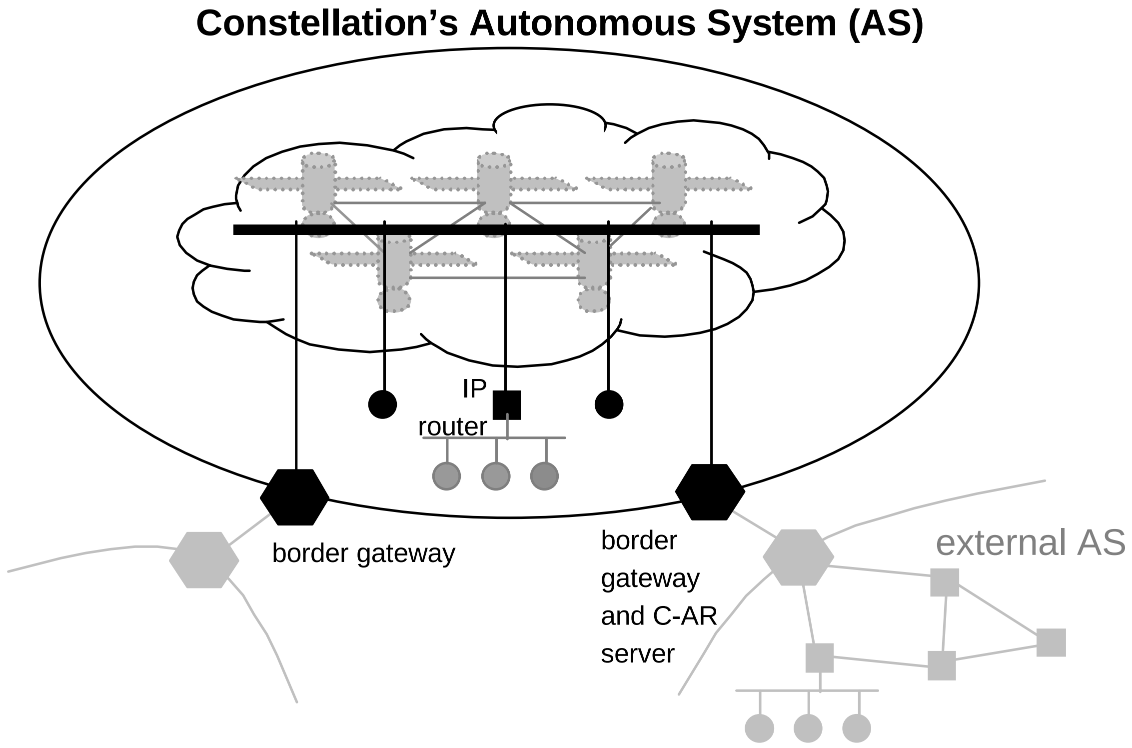


**Figure 5.1 - tunnelling in the constellation network [from Narvaezetal98]**

Having the C-ARS servers all linked over the satellite network may result in dangerous failure modes. A possible option would then be to have a core of primary C-ARS synchronized by sharing state over dedicated redundant terrestrial links. Additional secondary C-ARS could be added in remote regions and synchronized by communicating over the constellation itself with a primary C-ARS. A host whose Address Resolution Request to a secondary C-ARS failed would switch to a safer primary C-ARS.

Allocation of IP ground hosts to C-ARS servers can be dynamic, based on geographical position, or static, which leads to inefficiency in the case of large-scale roaming of mobile ground hosts. (Any localised movement of mobile ground terminals can be neglected compared to the larger satellite movement.)

### 5.6.6 The constellation realm

A specific routing scheme can hide the high mobility of the constellation from the ground users and from the rest of the Internet. In particular, we want to avoid generating huge amount of routing traffic between the separate routing realms of the Internet and the constellation, while still propagating necessary updates concerning route changes internally.

In the constellation realm, we can expect to exchange very little information on the dynamic topology of the constellation, since the topology changes are mostly predictable.

The necessary information for routing, i.e. the position of a node within that topology, can be deduced from its constellation address.

For the satellite interfaces:

- Satellite ID, ISL Interface ID
- or Virtual node and interface ID

For the ground host interfaces:

- Fixed geographical position

  Earth-fixed cell ID, related to latitude and longitude,

  or Virtual Node ID.

  or Moving geographical position

  Moving cell ID, current satellite and downlink interface ID.
- MAC address (code, time slot, and/or frequency ) or Host ID

A simple mapping between address and position within the topology allows us to use routing protocols that use very little network capacity to exchange topology information.

### 5.6.7 Advantages of tunnelling

- Tunnelling can allow us to adapt the tunnelling network layer and routing protocols inside the constellation network to the needs and constraints of that network.
- Tunnelling decouples network technologies inside and outside the constellation network.

- Tunnelling is a simple solution to separate routing updates and addressing in the constellation network from routing updates and addressing in the Internet.

### 5.6.8 Disadvantages of tunnelling

- Tunnelling imposes some processing overhead – dealing with headers, encapsulation and fragmentation.
- Tunnelling can give a false picture of the number of hops between two points, as discussed in Chapter 4. Decrementing hop counters can require explicit handling at the edges of the tunnel (as is done in MBone tunnels for thresholds to set the scope of multicasts.)
- As discussed earlier, mapping IP QoS and IP multicast onto the tunnelling network and supporting them in that network is a non-trivial problem.
- Events in the constellation network are not visible to the terrestrial Internet, making it difficult to notify the Internet about local network conditions that may have an impact on IP QoS. Notification to the Internet of congestion in the constellation network via e.g. Explicit Congestion Notification (ECN) [**RFC2481**] would be difficult without complex handling of notification events that are passed between the tunnelling layer and the terrestrial Internet.

### 5.6.9 Tunnelling in brief

This section has illustrated how:

- internetwork tunnelling restricts network flexibility.
- internetwork tunnelling makes internetwork congestion avoidance more difficult, particularly when network stacks with different assumptions about how congestion notification can take place collide.
- internetwork tunnelling is usually intended as transitional get-it-working step that becomes entrenched as a permanent implementation.
- internetwork tunnelling makes internetwork multicast difficult.

In short, tunnelling is undesirable. The endpoints of any internetwork are increasingly likely to be IP-based, and using IP routing end-to-end does not limit network flexibility or multicast implementations as tunnelling does. Suggesting implementing IP routing functionality onboard satellite, for flexible satellite multicast and to enable effective internetwork congestion avoidance, is a logical conclusion for internetworking with satellite constellations.

## 5.7 Network Address Translation

Network Address Translation (NAT) is the term for techniques used in private IP networks to manage internal address space by separating it from the global Internet address space. NAT translates the internal-realm addresses in every IP packet to new addresses suitable for use in the external realm.

NAT has been used to avoid having to renumber a private network when topology outside the network changes, for firewalls, and for a variety of other reasons. As an increasingly popular technique, its use is being documented by the IETF NAT WG (working group) [**RFC2663**].

### 5.7.1 NAT in the constellation network

The satellite constellation network can be viewed as a private network with a single operator controlling both the space segment and the terrestrial gateways interfacing to the terrestrial Internet. The external topology of the terrestrial Internet changes from moment to moment as far as non-geostationary satellites in the network are concerned, thanks to their orbital motion; if the satellites support IP routing, routing updates become a problem, as discussed earlier.

NAT can be considered as a useful way of separating constellation network addressing from global Internet addressing to provide separate address realms. It can remove the need for propagation of routing updates between the constellation and the terrestrial Internet.

NAT would be implemented in the terrestrial gateway stations interconnecting the satellite constellation network with the terrestrial Internet. These gateways translate between internally-visible addresses of constellation network users and externally-visible addresses associated with that gateway. The gateway is viewed as the endpoint of all communications for routers and users internal and external to the constellation.

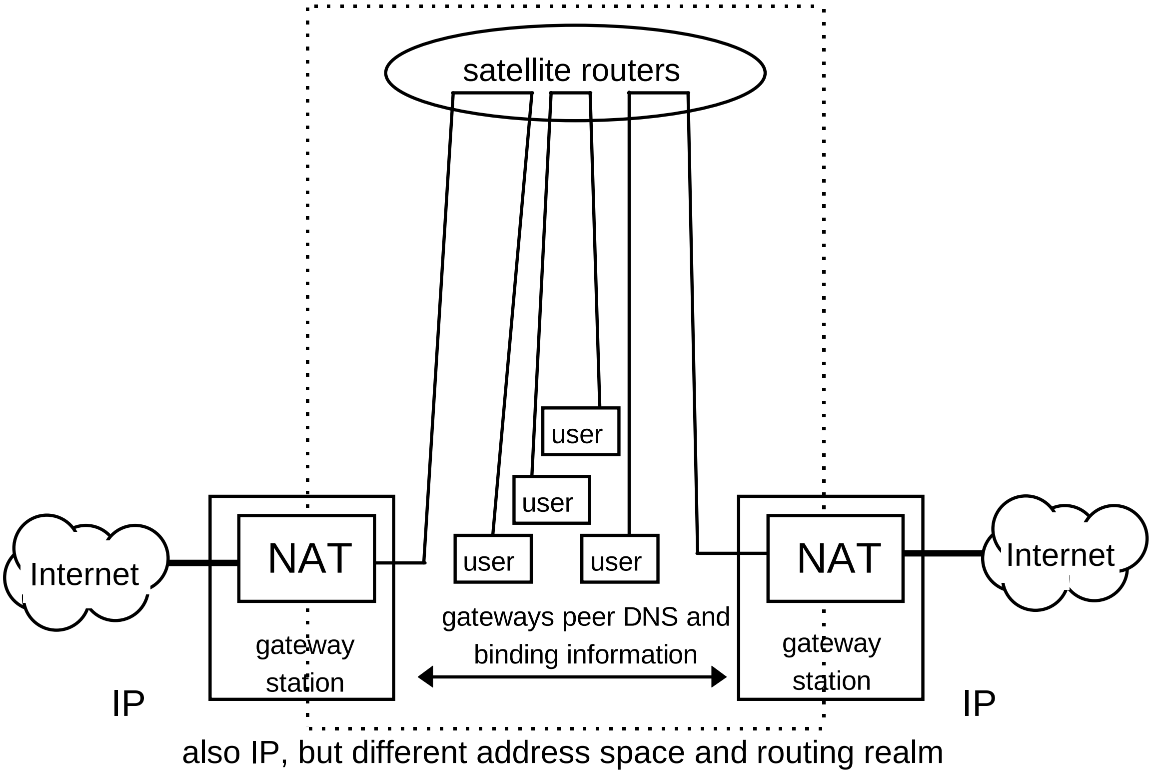


**Figure 5.2 - NAT in the constellation network [Woodetal01a]**

The gateway provides transparent routing by straddling and having knowledge of both addressing realms.

It would be necessary to propagate translations and address bindings between all the gateways in the terrestrial segment of the network, in order to be able to handle e.g. gateway failures or link outages.

The NAT implementation is illustrated in figure 5.2.

### 5.7.2 Types of NAT

There are a number of established varieties of NAT.

*Traditional or Outbound NAT* hides the private address space from global visibility by translating private ports and addresses seen in the headers of outbound packets. It removes the need for internal routing updates to propagate outside the private network to the global Internet. However, routing updates from the global Internet are still propagated within the private network, which is undesirable in this scenario. Since translation is only carried out on the way out, only hosts within the private network can initiate sessions.

*Bi-directional or two-way NAT* adds DNS support via a DNS-specific Application-Level Gateway (ALG) [**RFC2694**] and address binding to allow sessions to be initiated externally as well; routing updates from outside are still propagated internally.

*Twice NAT* translates addresses in both directions, rewriting internal addresses in packet headers to addresses associated with the gateway externally, and rewriting external addresses to addresses associated with the gateway internally. This means that the external Internet and internal network only need to know how to route to the appropriate gateway at the edges between the networks; it is no longer necessary to propagate routing updates into what are now separate routing realms. This is useful for the satellite network to decrease the routing state held onboard satellite to that of only the private satellite network. A DNS ALG, where names bind to different addresses depending on whether the source of the DNS request is inside or outside the private network, is also necessary.

Twice NAT offers the ability to abstract from a global *physical network address* to a *logical network address* that is used only within the constellation to identify and route to the translating earth station gateway.

As IP multicast communication is already abstracted to a logical group address, it is not necessary to translate multicast packet headers, and multicast can be handled as it is for terrestrial networks.

Routing table lookup within the satellite network can then become as simple as masking the destination address to determine in which block of addresses it lies, and with which destination gateway or constellation user network that block of addresses is associated.

The translation of packet headers takes place in the gateway, where global terrestrial routing tables are held. This moves complexity from the space segment into the ground segment, where more processing power is available to do header translation.

### 5.7.3 Implementation problems with NAT

NAT is widely regarded as undesirable as it removes clear abstraction and adversely affects the implementation and functionality of many existing applications and security services. As well as rewriting IP packet headers, it becomes necessary to rewrite in-band information in packet payloads that duplicates or relies on the header address or port information, using specific ALGs for each protocol to do so. This can add considerable implementation difficulty and processing overhead. As a result of this, security using IPSec [**RFC2401**] is heavily affected by the use of NAT [**RFC2709**].

From an operator mindset, having some applications explicitly supported by ALGs, while breaking all others and making them unworkable, may be desirable from a network- or use-control perspective, or even to allow charging for individual services that are enabled by use of an ALG. This selfish perspective is, however, contrary to the spirit of the Internet, and increasingly contrary to the expectations of users.

NAT also breaks explicit IP fragmentation, since only the first fragment of a packet possesses information identifying the protocol and the source and destination port used by the applications, while the remaining fragments are assigned fragmentation IDs that are not unique. This makes tracking of multiple simultaneous connections from the same end host difficult.

Use of Path MTU discovery can decrease the amount of fragmentation seen, and is essential in IPv6.

### 5.7.4 NAT with QoS

From a QoS point of view, NAT interacts badly with RSVP, the resource reservation signalling protocol associated with IP Integrated Services (intserv). The use of NAT invalidates RSVP Integrity Objects among other issues, due to the use of in-band information in those objects [**RFC3027**].

This does not entirely prevent the use of RSVP with NAT. Since a broadband constellation network acts as a high-capacity backbone or transit network for its customers, RSVP integrated services would in any case experience the scalability problems described earlier; RSVP-capable NAT gateways would need to map RSVP flows to roughly-equivalent differentiated-services PHBs within the constellation network.

### 5.7.5 Other NAT problems

NAT for multihomed constellation network users – where a user's small terrestrial network has a satellite link for redundancy in case their terrestrial connection to the Internet fails – also poses implementation problems.

The user's terrestrial network should not have to dynamically renumber itself into the constellation's realm if the terrestrial connection fails – avoiding renumbering is a motivation for implementing NAT in subnetworks on the Internet. One might assign the user's network addresses within the constellation's realm and require NAT at the router on the user's outbound terrestrial link. This NAT gateway could peer with and exchange bindings with constellation NAT gateways across the constellation network. However, this means that renumbering into the constellation's realm is necessary when the satellite link is enabled, making adding satellite redundancy to an existing network difficult.

### 5.7.6 NAT in brief

At first glance, twice NAT is an attractive way of decreasing routing table overhead and managing mobility by gaining address space separation into separate realms at the IP level without the need for tunnelling or the overhead of additional encapsulation.

However, NAT's considerable technical implementation problems mean that any use of NAT must be carefully evaluated.

From a non-technical viewpoint, the problems of NAT can arguably be considered a feature as far as the constellation network operator is concerned. IP-level support for services can be enabled and disabled on a per-ALG basis. Being seen to discourage implementation of IPSec might be viewed as helpful in meeting the security concerns of international governments. Potential customers may, however, disagree.

Implementing protocol support at the edges of the constellation network in the ground segment, which is easier to manage and change, is an advantage of NAT. However, this is an advantage that is shared by tunnelling and by exterior routing, which do not have NAT's protocol-specific implementation problems or processing and state overhead.

## 5.8 Exterior routing for constellation networks with BGP

For administrative purposes, today's Internet is divided into many different autonomous systems (AS). An AS is a collection of networks under a common administration using a consistent routing protocol. An AS is identified by a unique 16-bit number that is assigned by the Network Information Center (NIC).

Splitting the Internet into ASs makes it possible for groups of networks using different routing strategies to cohabit without increasing their interdependency to completely unmanageable proportions. When packets travel between ASs, they must cross a pair of connected border gateways.

The satellite constellation network can be viewed as an AS. Autonomous systems must communicate and exchange routing information to make global routing possible. Border gateways run an exterior routing protocol that enables them to determine routes to other AS. These routes are then propagated in the autonomous system through the internal routing protocol. Decisions about which external routes can be propagated internally, used externally, and advertised to other ASs are policy decisions made by the operators of the AS.

The Border Gateway Protocol (BGP) is an example of an exterior routing protocol that is widely deployed in the Internet, having mostly supplanted the older Exterior Gateway Protocol (EGP) [**RFC1771**, **RFC1772**].

Different types of BGP connections (communicating over TCP) are established between the border gateways:

- *External BGP connections,* between the border gateways of neighbour autonomous systems, are used to advertise routes to networks of the autonomous system, and routes to other autonomous systems' networks. The border gateway should only advertise routes that it itself uses to prevent pathological routing, but it is possible to restrict these exported routes further for administrative reasons.

- *Internal BGP connections,* between all the border gateways of the same autonomous system, are used to exchange routes learned from external connections. They then decide on an egress point for networks outside the AS by minimising the *external metric*, evaluated locally by the border routers using criteria such as the length of the AS path. Multiple gateways connected to a neighbouring AS may be chosen between for communication using the *border gateway preference value* (awkwardly called the Inter-AS metric in BGP) that is advertised by this neighbouring AS.

Routes to external networks via border gateways are then imported to all the routers of the autonomous system. This importing is carried out using the AS's internal routing protocol, whatever that routing protocol may be [**RFC1403**], or by using an internal gateway protocol (IGP) to indicate the choice of border gateway to be used for routes to external networks.

### 5.8.1 BGP traffic

The border gateways of an autonomous system must be interconnected with BGP connections internal to the AS to form a completely connected graph. Routing updates received at one of the border gateways must be propagated to the rest of the border gateways, in a similar fashion as discussed for NAT. This generates a lot of traffic, as the updates concern information on routes to all networks in the Internet external to the AS. This traffic is best handled in the constellation network by using dedicated terrestrial links, to avoid large amounts of routing table state update traffic passing through the satellite constellation and to separate reachability from satellite reliability.

Information about networks that use the same AS route can be aggregated so that fewer updates are sent to internal peers.

### 5.8.2 Choosing ingress and egress points

Routing from or to an external network is done hierarchically: the ingress or egress point is chosen by the border gateways, and the shortest route or most appropriate route to that point is determined by the internal routing protocol. External metric is therefore given more importance than internal metric.

In many terrestrial networks, this is reasonable, since the external metric represents the cost of sending a packet through other ASs over a large geographic distance, whereas the internal metric measures the lesser cost of the transmission inside a single AS.

However, in the constellation network, the length of the internal path may easily be as large as or larger than that of the external one, since this autonomous system itself covers the entire surface of the Earth. This discrimination between external and internal metric may therefore be inverted.

It is possible to force incoming traffic for a given constellation user network to come through one of the nearest border gateways by advertising a route to this network only from these gateways. For outgoing traffic, having all the border gateways import their best route inside the AS, mapping the external metric onto the internal one, enables us to optimise the path based on a combination of external and internal metrics.

Note that this may lead to asymmetrical routing, and that this asymmetrical routing may not work with NAT, where incoming and outgoing traffic must cross the same NAT gateways to avoid propagation of per-flow state between peered NAT gateways.

## 5.9 Co-existing with IP routing

Support for IP routing is extremely useful for handling IP traffic well, but is not useful for routing non-IP traffic, such as ATM or frame relay.

Given the large amount of work completed on wireless and satellite ATM links, it is likely that ATM will provide an underlying link-layer protocol over which IP traffic will be carried within the satellite constellation. A satellite-specific MAC layer for ATM must be defined for the ISLs and for the ground/space air interface, much as a MAC layer would be needed for IP.

With the use of ATM, the interworking of IP and ATM poses a number of interesting problems, particularly with respect to routing and support for IP multicast and QoS.

One solution for IP multicast over ATM is the Multicast Address Resolution Server (MARS), which maps IP multicast addresses into ATM server addresses [**RFC2149**]. However, the MARS family does not cope with mobility, does not scale easily to multiple servers that must share state, and is difficult to implement because ATM's routing paradigm and resulting multicast model differ considerably from those of IP.

Support for IP QoS over ATM is a non-trivial problem due to difficulties in mapping the IP QoS models accurately to available ATM service classes.

### 5.9.1 IP routing on ATM with MPLS

Multi-Protocol Label Switching (MPLS) is a developing technology, which warrants serious consideration for the IP-over-ATM scenario [**Callonetalmplsdraft99**]. It is being standardised by the ITU as the IP-over-ATM transport method of choice.

MPLS uses a label-swapping paradigm to integrate the flexibility and efficiency of Layer-3 IP routing with the fast Layer-2 switching of ATM. Fixed-size labels are assigned to the IP packets according to their respective egress destination nodes in the satellite network. Labelled packets that are destined for the same egress traverse a label-switched path (LSP) that is bound to an equivalent Layer-3 route. Every MPLS-enabled ATM switch, or Label-Switching Router (LSR), along the LSP checks the label and rapidly forwards the packet to the appropriate output interface with little lookup overhead.

Labels are exchanged by means of a label distribution protocol (LDP) [**RFC3036**]. LSPs can be pre-established for reserved traffic, or created when required after initial layer-3 IP routing of a flow, and may be updated according to the IP routing tables.

Relative to the interconnection of IP edge routers tunnelled over an ATM core, MPLS improves the scalability of routing. This is due to the reduced number of immediate peers and elimination of the '*n*-squared' logical links between the *n* IP routers at the edge of the ATM core that are operating IP routing protocols.

A major feature of MPLS is that all ATM software above the ATM adaptation layer (AAL), including signalling, does not need to be involved in the routing of IP traffic, and does not need to be present or even defined. As the MPLS flows depend upon the IP routing tables, IP routing has full control of IP traffic. It is possible to use MPLS control plane to provide IP routing of IP traffic in parallel with native ATM-Forum control plane for ATM traffic without interference (so-called 'ships in the night' operation).

MPLS provides the following significant advantages for IP over ATM, which are of benefit to IP traffic in constellation networks:

- MPLS forwarding can be accomplished with little computational overhead or network knowledge – of benefit to on-board processing.

- Layer-3 IP routing can be used as is, without any need for interoperability with ATM switching or for explicit circuit setup.
- Hierarchical MPLS can be used to hide external routing information from internal nodes. Multiple layers of tunnelling are possible via label stacks, allowing e.g. BGP information to be easily distributed through nodes that remain unaware of routing outside the constellation AS.
- MPLS can be configured to use explicit routing controlled by egress switches, in order to divert traffic from a congested part of the network, for example. MPLS can allow the egress ground-based gateways and user terminals to explicitly control routing across the constellation network based on the visible satellites and distributed topology information.
- IP multicast spanning trees can be mapped in a straightforward manner onto the ATM network by mapping branches of the multicast trees directly to the relevant LSPs.
- Downstream merge to a node supports multipath communications across a mesh network well. Layer-3 forwarding and hashing functions on source and destination addresses can be used to prevent out-of-order packets in individual source/ destination flows, or individual label stacks can be used;
- IP QoS, whether it be an implementation of integrated or differentiated services [**AndrikopoulosPavlou99**], can be supported using the available IP routing and traffic engineering functionality. Mapping differentiated services behaviour aggregates to available label values is possible [**Facheuretalmplsdraft00**].

### 5.9.2 Constraint-based routing

QoS routing can be important for LEO constellations, due to their redundant mesh topologies and choice of available paths to a destination. Given a QoS request for a flow, QoS routing could return the route that is most appropriate to the QoS requirements.

Constraint-based routing considers not only the topology of the network and the QoS requirements of the flow, but also resource availability of the links, and possibly other information specified by the network administrators – such as assigned link costs from Virtual Topology Routing. By taking all these factors into consideration, constraint-based routing may find a longer and lightly-loaded path better than the heavily-loaded minimum-delay or -hop path, distributing network traffic more evenly, avoiding congestion and improving network utilisation.

The primary components of a constraint-based routing scheme are the advertisement of link state information and the selection of metrics and route computation algorithms. This is applicable to MPLS-based architectures. MPLS LSPs allow constraint-based routing with per-LSP statistics, and LDP can be extended to provide constraint-based routed label switched paths (CR-LSPs) [**Janoussietalmplsdraft99**]. Given precise information on how traffic flows through the network, constraint-based-routing can determine how to dynamically configure LSPs for explicit routing to carry the traffic through the network more efficiently and provide effective QoS. MPLS enables *traffic engineering* (TE) to be undertaken within the network [**Armitage00**, **RFC2702**].

## 5.10 Approaches taken by the commercial constellations

IP QoS and IP multicast implementations for internetworks are still being defined at present, and are not yet widely implemented. A number of commercial broadband constellations are finishing their design stages. It is clear that implementing support for IP multicast and for IP QoS cannot figure strongly in the designs of these commercial constellations. In fact, onboard IP routing does not figure in commercial designs either.

Of the most visible of the proposed broadband constellations, Hughes' *Spaceway* and Lockheed Martin's *Astrolink* are fixed GEO constellations that appear to be adopting ATM-based switched communication across ISLs and in the ground/space interface, with custom MAC/LLC layers for the satellite environment and custom signalling. Their protocol architectures may resemble those described in [**Mertzanisetal99**].

Of the LEO constellations, Teledesic LLC is understood to have designed its own custom- protocols for use over ISLs and in the earth/space interface in its constellation design, while Alcatel's *SkyBridge* is taking a ground-based ATM approach without the use of either onboard routing or ISLs.

These commercial constellation networks can be expected to support end-to-end communication of unicast IP traffic via tunnelling, as described in Chapter 4 and in [**Woodetal98**], but support for IP beyond that is an open question.

Given the use of either a custom protocol or ATM with classic AAL encapsulation, implementation of IP QoS, IP multicast, or of future enhancements to the IP architecture in the proposed commercial constellations looks to be problematic. At this point in time, no commercial constellation has been proposed using MPLS.

## 5.11 Summary

We have presented compelling reasons for wanting IP routing in the satellite constellation network, and have shown that objections to implementing IP routing are not insurmountable. Use of IP routing offers benefits to the IP traffic that is routed, allowing straightforward implementation of support for IP multicast and for IP quality of service. It avoids the difficulties or complexities of attempting to map necessary IP routing state and IP assumptions about the network onto a different network layer.

IP routing can be implemented in a manageable fashion in satellite constellation networks by selecting from a combination of border routing protocols, tunnelling, NAT and MPLS.

We propose the use of an architecture that is based on a combination of BGP and MPLS within the constellation network. MPLS appears to be a realistic and promising method for implementing support of advanced IP routing functionality on an ATM-based backbone likely to be found in the environment of a satellite constellation network. MPLS also permits good support for non-IP protocols i.e. it is capable of co-existing with ATM routing of ATM traffic. Future broadband constellation network designers should evaluate the use of MPLS within their constellation.

# 6. *Managing diversity with handover to provide service classes*

The rosette constellation network with intersatellite links presents unique properties, in providing locally separate ascending and descending surfaces of interconnected satellites for the ground terminal to communicate with. This chapter discusses an approach exploiting this rosette geometry by using control of handover decisions and management of satellite diversity to decide which surface the ground terminal communicates with. This effectively provides a degree of ingress control. By allocating traffic separate paths through the network with different degrees of delay, different levels of service become available for traffic between ground terminals. This allows us to use the rosette's network topology to provide varying classes of services, and thus a degree of support for QoS (Quality of Service) from a delay-oriented viewpoint.

## 6.1 Introducing diversity

*Diversity* is a widely-used term. It simply indicates that a diversity of communication methods is available, i.e. that for a satellite user, there is more than one satellite available for a ground terminal to communicate with at all times. At the physical level, diversity and multiple satellite visibility can be used to combat shadowing by buildings or terrain, to provide redundancy and increase the likelihood of a satellite always being in view of the terminal. The percentage of time that diversity is available for a number of commercial proposals has been analysed in detail [**GkizeliTaagEv00**].

Two basic implementations of satellite diversity are *switched diversity* and *combined diversity*.

**Switched diversity** simply means that the ground terminal has a choice of multiple visible satellites with which it can communicate with, and that it switches between use of individual satellites as conditions dictate.

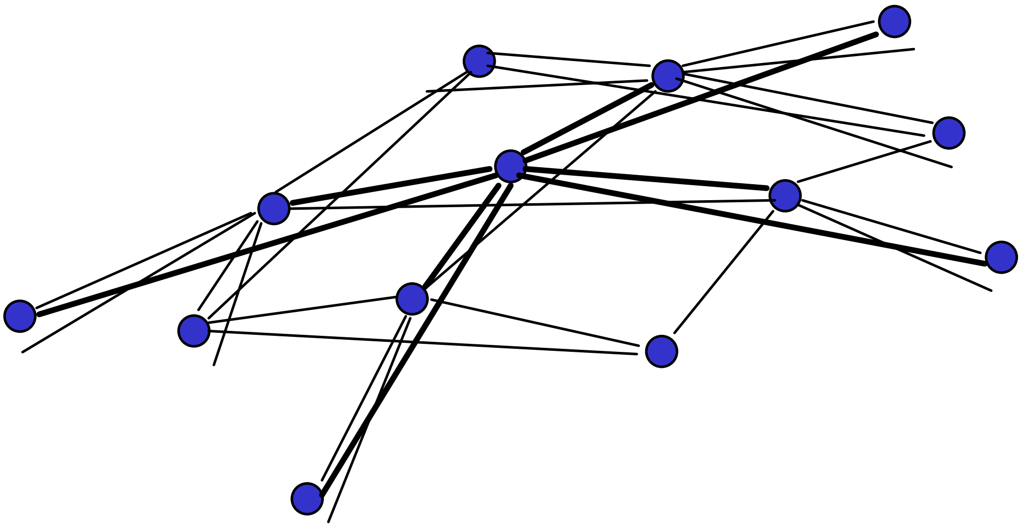
**Combined diversity** is when the ground terminal communicates across multiple satellites simultaneously. This is also referred to as artificially-introduced multipath

[**WernerBischLutz95**]. This is exploited in CDMA-based systems such as *Globalstar*, which recombine signals passed through more than one satellite at a shared gateway station as a way of combating shadowing. This combined diversity can be exploited to ensure soft handovers.

Ground-to-space diversity can be exploited at various layers of the network protocol stack. Physical diversity can be exploited in rosette constellations without ISLs, e.g. *Globalstar*'s use of CDMA and recombination of signals across multiple satellite transponders. It can be exploited at the data-link layer, via TDMA management as in *ICO* [**Wisløff96**].

Beyond that, coding diversity and network-layer diversity between a terminal and multiple satellites are not currently proposed for exploitation in commercial constellations, although [**Schacham88**] identified multihoming of ground terminals – using available satellite diversity to increase redundancy and thus fault tolerance – as desirable from a networking viewpoint.

Network-layer path diversity for satellites communicating via ISLs is already present in the original 840-active-satellite design and the Boeing redesign of *Teledesic*, due to the large number of satellites and their close spacing [**Sturza95**]. This was shown in figures 2.4 and 2.11c and is illustrated conceptually in figure 6.1. The multipath ISL routing discussed in Chapter 3 and in [**Woodetal01b**] can also broadly be considered a form of routing diversity. Use of diversity is fundamental to traffic engineering.



all links from the central satellite are shown darkened.

**Figure 6.1 - part of *Teledesic* network mesh showing network diversity**

Although the *Teledesic* designs that have been discussed have diversity in the space segment, they have no diversity or redundancy in the ground-space interface; a ground terminal is simply allocated to a single satellite. If that single satellite is shadowed or has failed, the ground terminal is unable to communicate with the satellite network.

A different form of ground-to-space diversity use is planned in *SkyBridge*, which uses dual switched diversity between neighbouring satellites in its two subconstellations to avoid sending transmissions from satellites seen as being in the part of the sky already inhabited by the geostationary arc. This is necessary because *SkyBridge* is based around reusing Ku-band frequencies already in use by geostationary satellites [**FCCSkyBridge97**]. Since *SkyBridge* does not have ISLs, its use of diversity in this way means only a minor alteration in end-to-end path delay between terminal and terrestrial gateway, with no other visible effects from this diversity from a networking viewpoint. This dual diversity could be exploited even when satellites are in another part of the sky, providing a degree of redundancy in the communication infrastructure.

## 6.2 Managing diversity in rosettes with ISLs

Rosette constellations with ISLs offer the opportunity to implement diversity, not just across multiple ascending and descending satellites, but across multiple ascending and descending satellites that form part of ascending and descending ISL meshes. These meshes, though overlapping and part of the same constellation mesh outlined in Chapter 2, are locally separate; to reach the local ascending mesh from the local descending mesh traffic must travel over the highest latitudes using intra-plane ISLs. To investigate the effect of this choice of meshes on constellation traffic, the satellite simulations introduced in Chapter 2 were modified so that ground terminals could express a preference whether handover would take place to ascending or to descending satellites, if both were available. The network could accept and follow this preference, permitting the terminals some influence over the paths that network traffic would take.

A simple terminal handover function was introduced to the *ns* satellite extensions described in [**Henderson00a**]. When the current satellite drops below the minimum elevation angle deemed necessary for reliable communications by a terminal and handover is required, the terminal looks for the highest visible satellite and establishes a new link to it.

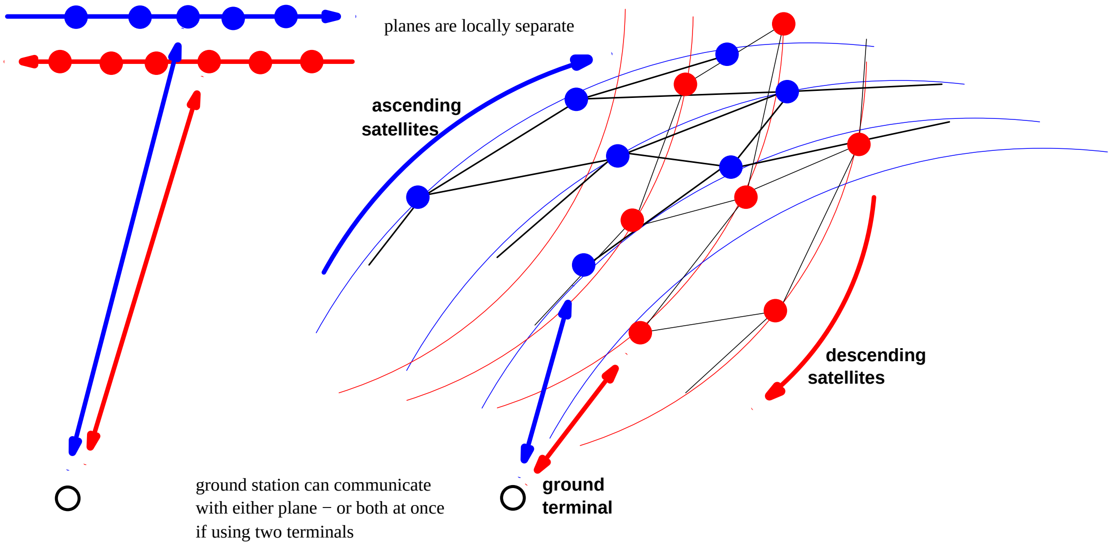


**Figure 6.2 - choice of ascending and descending ISL mesh surfaces**

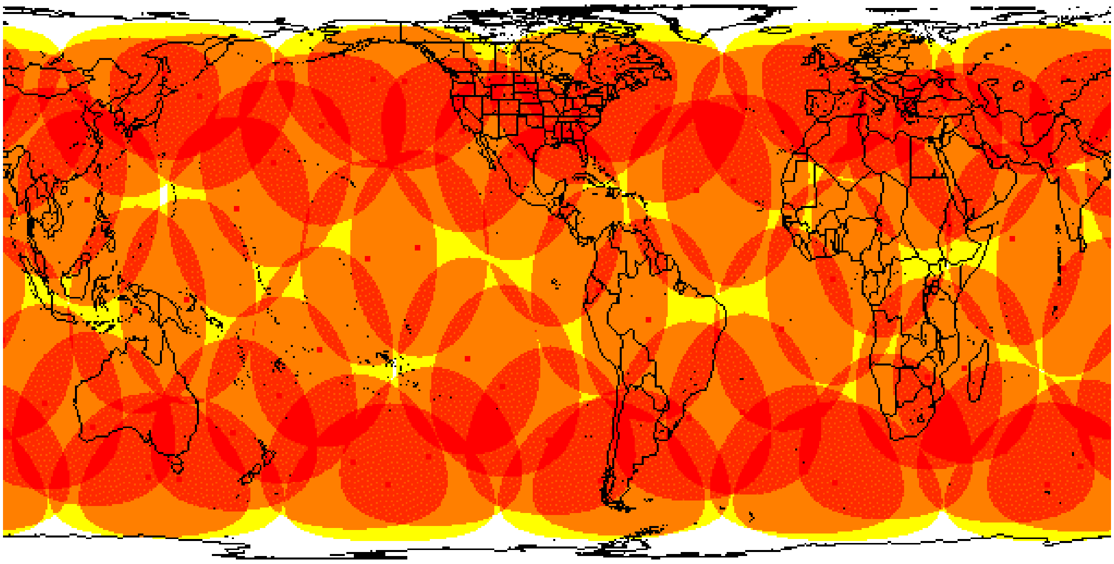
We further modified this procedure so that the terminal can find and select between the highest visible ascending and highest visible descending satellites.

The terminal will then select one satellite for communication, so that it will always use an ascending (or descending) satellite if physically possible (and permitted by the network, which must consider available spotbeam capacity and other issues). This careful use of available diversity offers the opportunity to dictate the degree of end-to-end path delay across the ISL mesh to a considerable extent, depending on whether the two terminals that are communicating are using the same or different ascending or descending surfaces.

Packets can be injected into the ISL mesh consistently on one ‘mesh surface’ of the rosette chosen by the ground terminal, before normal shortest-path routing, based around a delay metric, routes them to their destinations across the ISL mesh. This choice of surfaces is illustrated in figure 6.2.

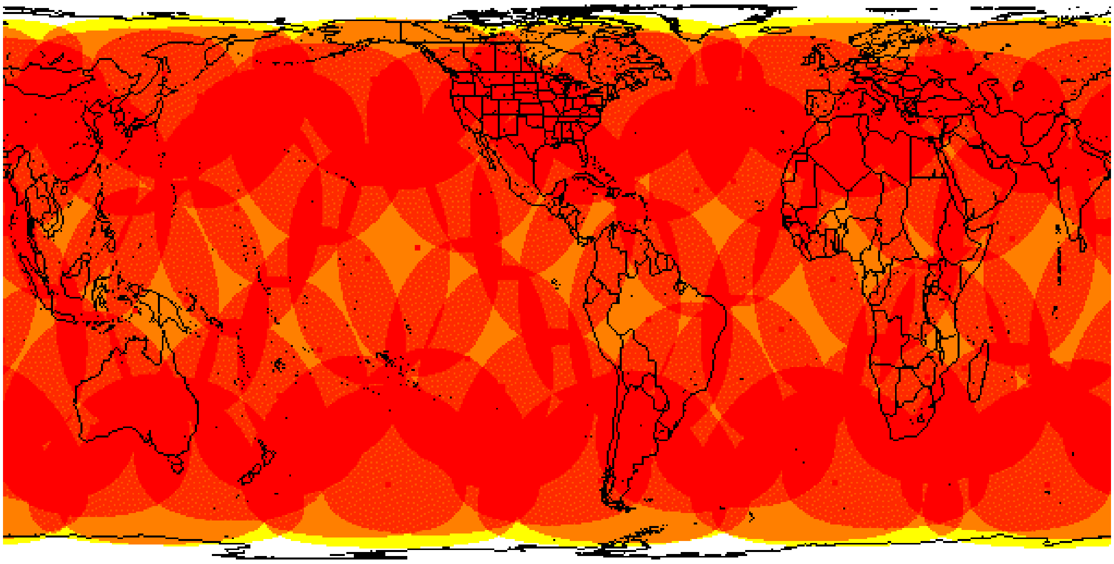
Communications between two co-located terminals using the same (ascending or descending) surface will have very short path delay times, while communications between co-located terminals using different surfaces will need to travel over intra-plane links across highest latitudes, and will have longer path delay times.

For terminals separated by 180° of longitude, the path over highest latitudes is likely to be shorter and more direct than the path across interplane links around the Equator.



cylindrical projection, using minimum elevation angle of 16°. compare with Figure 7.4.

**Figure 6.3 - *Celestri* with minimum elevation angle of 16° [FCCCelestri97] (*SaVi*)**



cylindrical projection, using minimum elevation angle of 10°. compare with Figure 7.3.

**Figure 6.4 - *Celestri* with minimum elevation angle lowered to 10° (*SaVi*)**

## 6.3 Providing double surface diversity

To examine the effects on the communications between ground terminals of selecting ascending or descending surfaces, it was necessary to use a constellation geometry providing not just double coverage, but double *surface* coverage, throughout.

Having more than one satellite visible at all times from covered areas is not sufficient; a minimum of one satellite from each of the ascending and descending surfaces must be visible to provide double surface diversity. Commercial proposals that combine rosette constellation geometries with ISLs are few, but do include Motorola's *M-Star* proposal that predated their *Celestri* proposal [described in **WernerDelBurch97**].

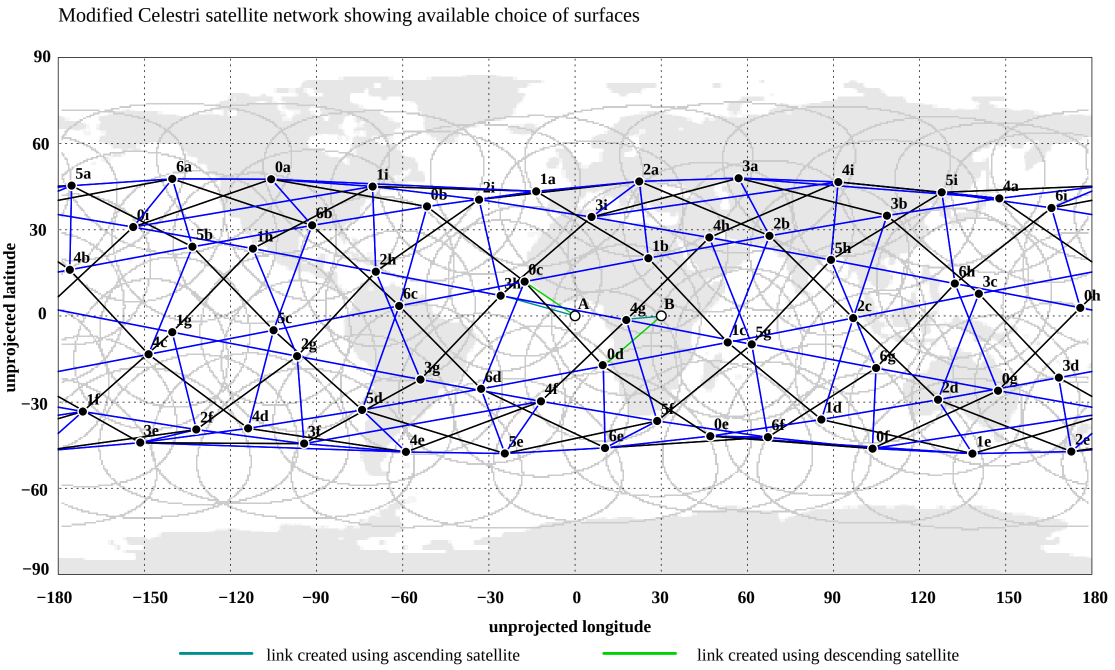


**Figure 6.5 - *Celestri* network (lowered elevation angle) showing handover choices**

The *Celestri* proposal [**FCCCelestri97**] and the Hughes *Spaceway NGSO* proposal [**FCCSpacewayNGSO97**] are rosette constellations with ISLs. However, these systems do not provide double surface coverage, and therefore do not allow a terminal to select between ascending and descending satellites at every point in time.

The geometry of *Celestri* was examined in a simulation, which was then modified to produce a rosette constellation capable of supporting full double surface coverage.

### 6.3.1 Using *Celestri*

The *Celestri* geometry is described in [**FCCCelestri97**] as giving single coverage of the Earth at elevation angles above 16° between 60° of latitude. A simulation based on that description is illustrated in figure 6.3. The simulation shows coverage from a single satellite in places, as well as the occasional (but small) gap in coverage; terminal handovers between surfaces would be dictated by this low varying coverage.

The proposed *Celestri* design was used as a base. Lowering the minimum mask elevation angle of *Celestri* ground terminals from 16 to 10 degrees gave double surface coverage, with the separate layers of ascending and descending satellites covering the Earth entirely between 60° of latitude, as shown in figure 6.4. This presumes that link budgets could be recalculated and equipment dimensioned to meet this change.

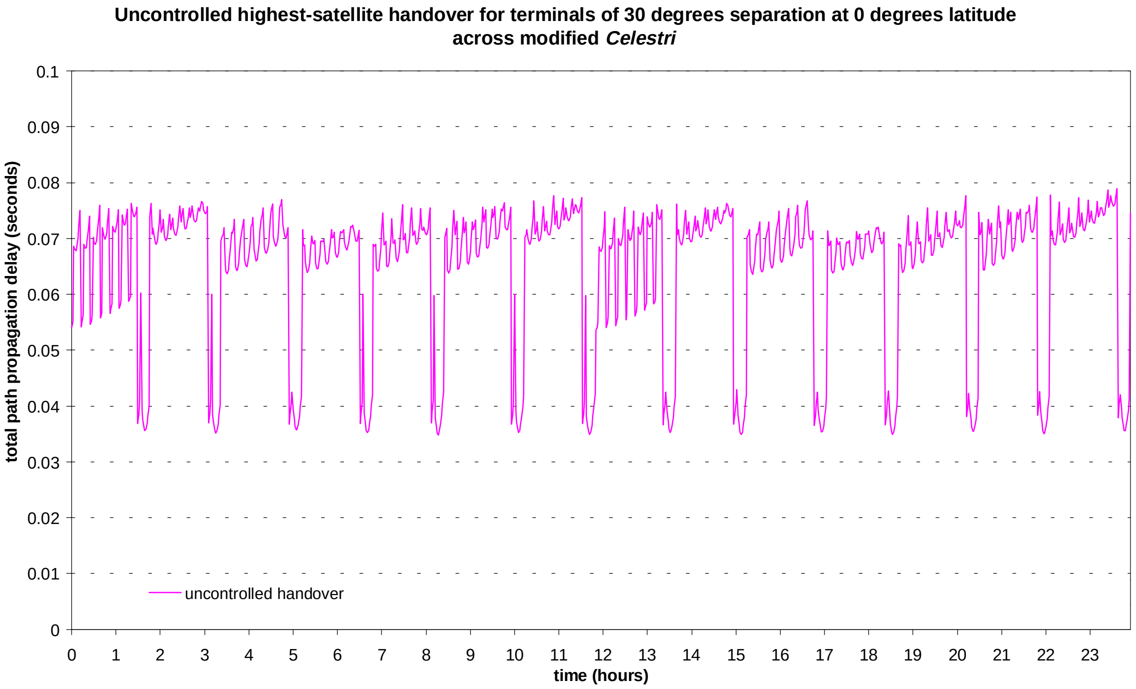


**Figure 6.6 - handover to highest-elevation satellite, ignoring satellite movement**

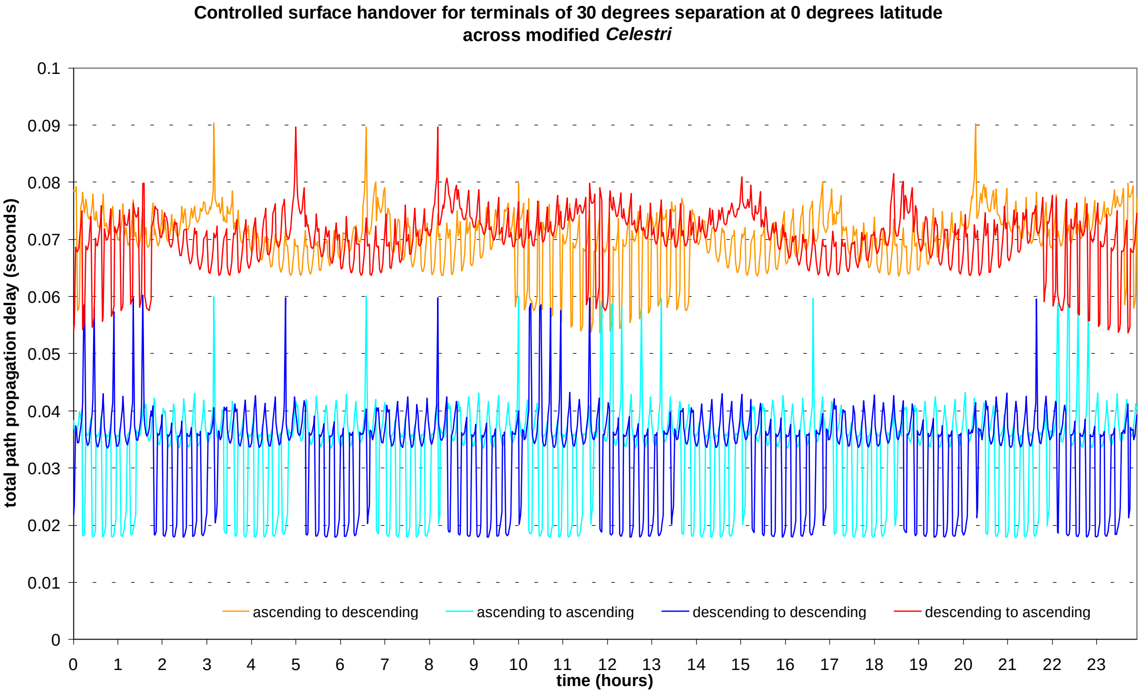


**Figure 6.7 - handover to highest satellite sharing direction of former satellite**

We were then able to compare terminal handover strategies: choosing the highest visible satellite versus a preference of highest visible ascending or descending surface, as illustrated for the terminals shown with the satellite network topology in figure 6.5.

Resulting path delay curves for traffic between the terminals are illustrated in figures 6.6 and 6.7. Figure 6.6 shows use of simple seek-highest-satellite handover, which does not discriminate between ascending and descending surfaces, and therefore wanders between them. This shows considerable variation in the path delay between terminals over the course of a day. When the terminals happen to both be using the same ascending or descending surface, where satellites are nearby in the ISL mesh topology, the delay across the path is low (<40ms). However, most of the time the terminals are using satellites on different surfaces, where the satellites are distant from each other in the ISL mesh, and path delay is higher (54-78ms).

The four different delay curves resulting from use of preferred-surface handover between the same terminals, where the network ensures that terminals remain on the same ascending or descending surface of satellites, are shown in figure 6.7. The conceptual layout of the different routes is shown in figure 6.8.

Use of this careful handover exhibits lower delays for the shorter paths between terminals, when the satellites the terminals are using are near each other in the ISL mesh. Following handover preferences also provides a second set of larger delay curves, due to longer paths over highest latitudes in the ISL mesh, when the satellites that the terminals are using are further apart in the mesh.

As with *Teledesic* in Chapter 2, some interplane ISL handover transients were encountered and are clearly visible in both plots; they are the longer two-ISL transients resulting from switching to using a satellite in a neighbouring plane.

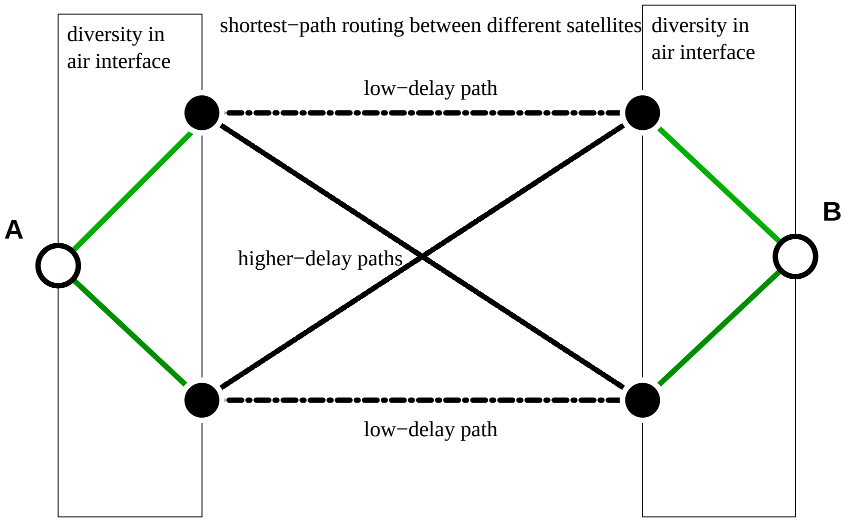


**Figure 6.8 - diversity permits four shortest-path routes across the constellation**

The ISL mesh is still using shortest-path routing based on a delay metric; by influencing where the sources and destinations of traffic connect to the ISL mesh – a form of network ingress control – we are effectively able to choose different paths across the ISL mesh for different types of traffic.

By having ground terminals use the same or different surfaces, we get two clear classes of delay between them. By having multihomed gateway stations communicate with both ascending and descending surfaces in the satellite network, these delay classes become accessible to single-link ground terminals that do not themselves exploit diversity simultaneously. Although the terminal is only communicating with one surface, it can send traffic to the multihomed gateway using two different routes.

To use both classes of service when communicating with other ground terminals that do not exploit diversity, the terminal must itself exploit diversity and communicate with both surfaces; this combined switched network-layer diversity requires multihoming by maintaining two communication links, using two sets of terminal equipment.

Traffic between ground terminals not exploiting diversity will not have its choice of path influenced. Terminals exploiting diversity could allocate lowest-priority traffic to either surface to use up available air-interface spotbeam capacity after higher-priority traffic has been allocated; this low-priority traffic will experience terminal handover that does not consider the resulting path the traffic takes across the network.

The delay properties of handover preference were examined, using a similar approach to the *Teledesic* seam delay assessment presented in Chapter 2. As the *Celestri* geometry provides coverage between 60° of latitude, we examined a range of latitudes at 15° intervals, as shown in figure 6.9. The rosette constellation lacks the obvious symmetry of the star constellation's seam, so results are presented for the whole 360° of longitude. Each simulation run was repeated for a variety of handover choices:

- terminal A and B have surface-unaware handover (pick highest available satellite).
- terminals A and B prefer ascending satellites.
- terminals A and B prefer descending satellites.
- terminal A prefers ascending satellites, while B prefers descending.
- terminal A prefers descending satellites, while B prefers ascending.

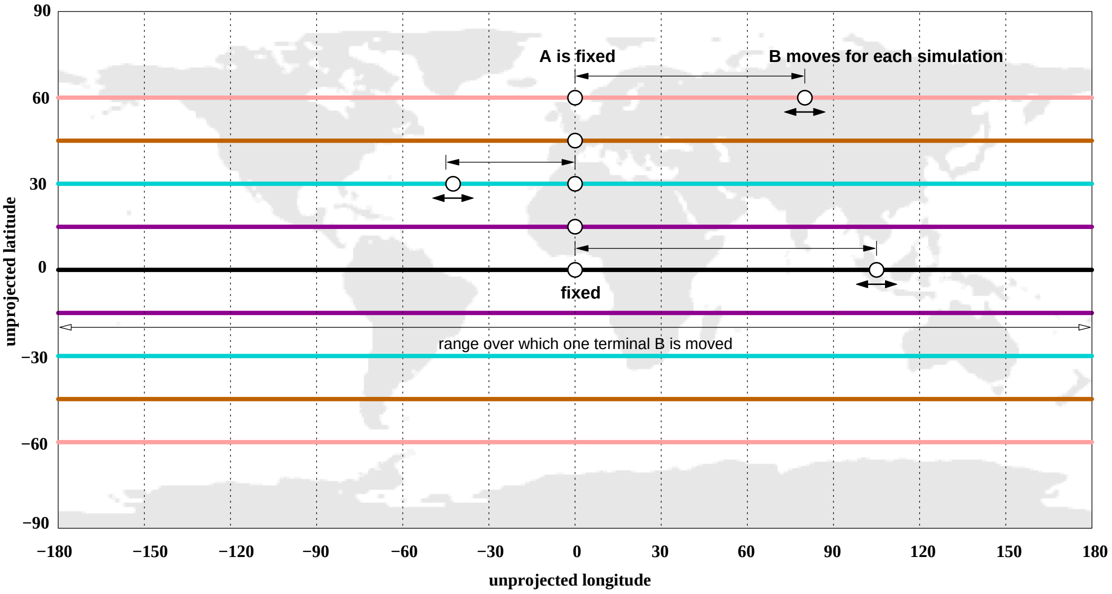


**Figure 6.9 - measurement of path delays at different latitudes between terminals**

Summary results are presented here in the form of comparisons of average path delays experienced across latitudes for the various terminal handover scenarios. Detailed simulation results are provided in Appendix 4.

The differences in path delay between the first class of service and the second, longer delay class of service are clearly visible for terminals at the Equator in figure 6.10. The differences between the classes lessen as the spacing between the terminals increases, until there is no appreciable difference at 180° separation.

Traffic from terminal handovers not considering network paths varies between the two classes and is extremely sensitive to minor differences in longitude and to initial handover choices at the start of the simulation. As the latitude increases, the difference in delay between the two service classes becomes less marked. It is possible for the effects of poor path choices resulting from simple handover at both terminals to cancel each other out; this explains the lack of variation for that at 15° latitude (figure 6.11).

As the latitude of the communicating terminals increases towards the limits of coverage of the constellation (figures 6.12-6.14), the differences in path delays for the service classes drops to nothing. This is because terminal handover is increasingly dictated by available coverage, and different paths overlap and share common routes.

Having one terminal use preferred-surface handover, while simple surface-unaware handover is used at the terminal it is communicating with, was not examined.

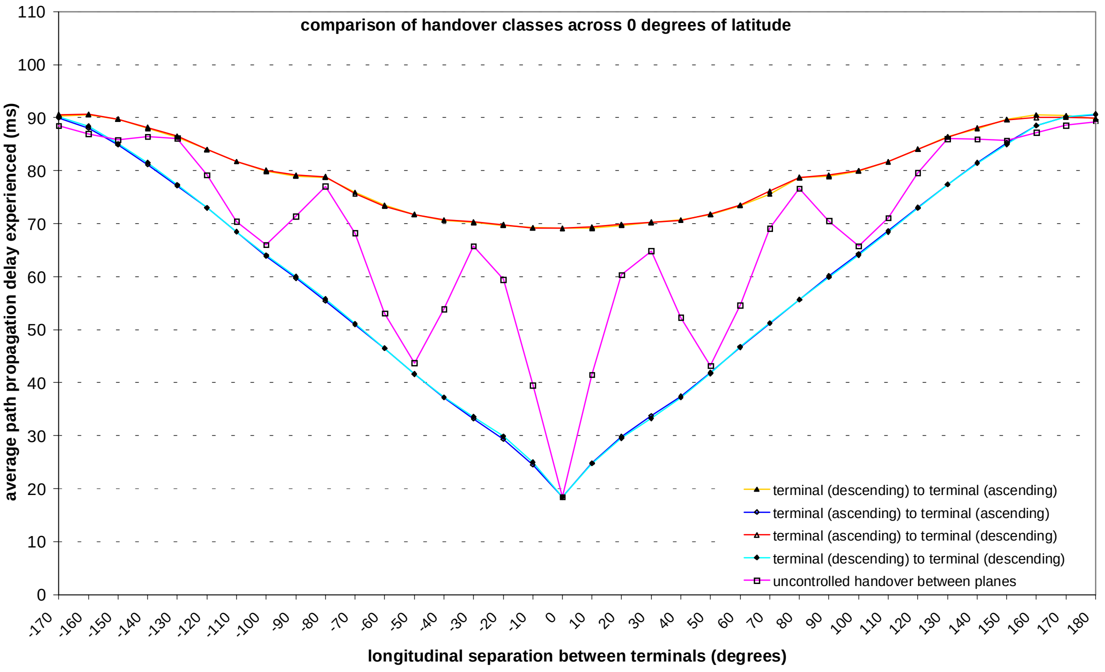


Figure 6.10 - different path delays between terminals on Equator

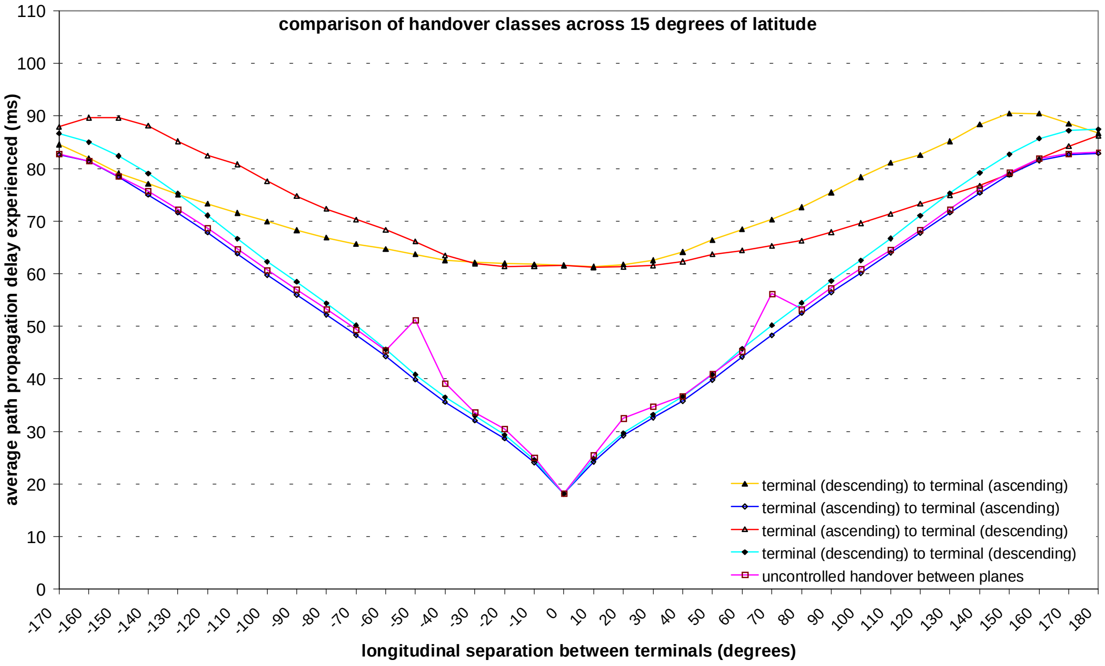


Figure 6.11 - different path delays between terminals at 15° latitude

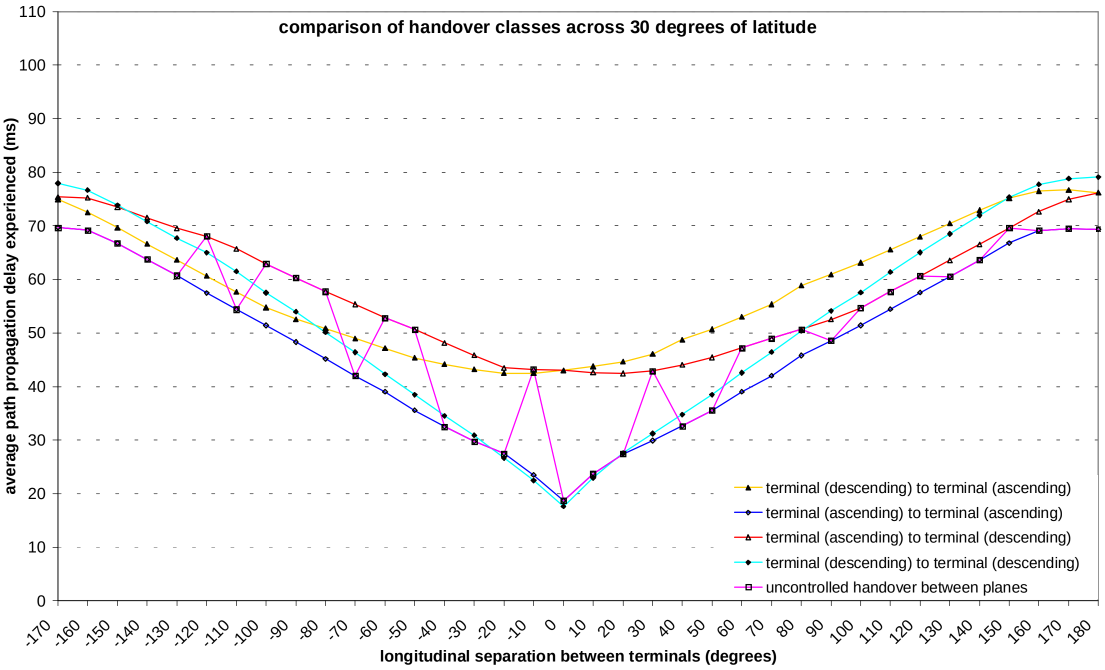


**Figure 6.12 - different path delays between terminals at 30° latitude**

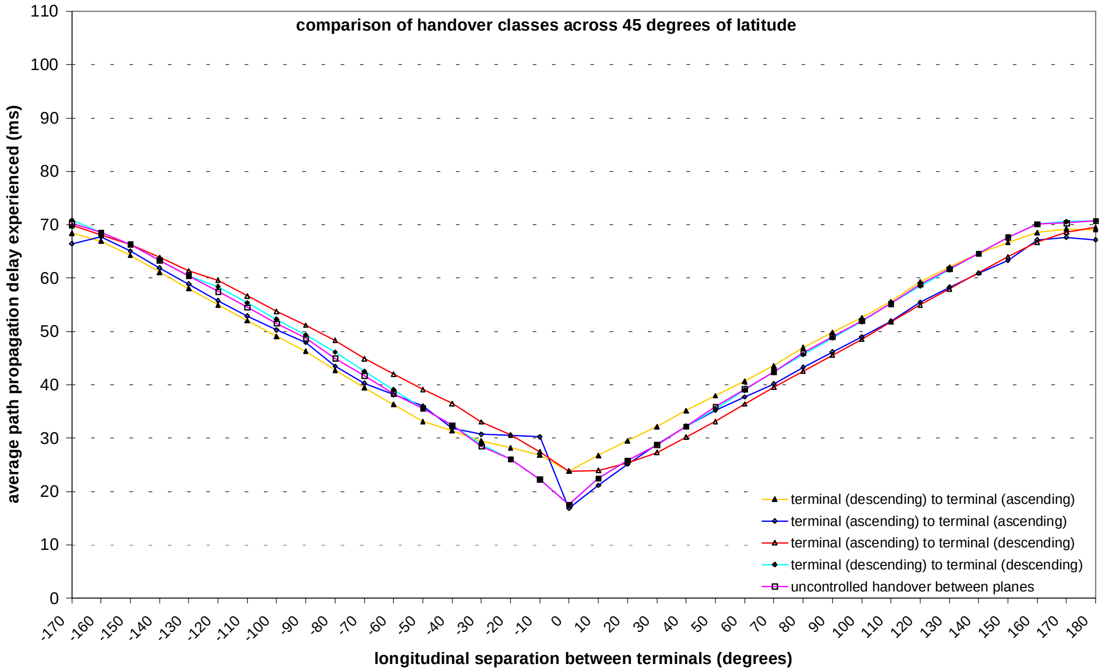


**Figure 6.13 - different path delays between terminals at 45° latitude**

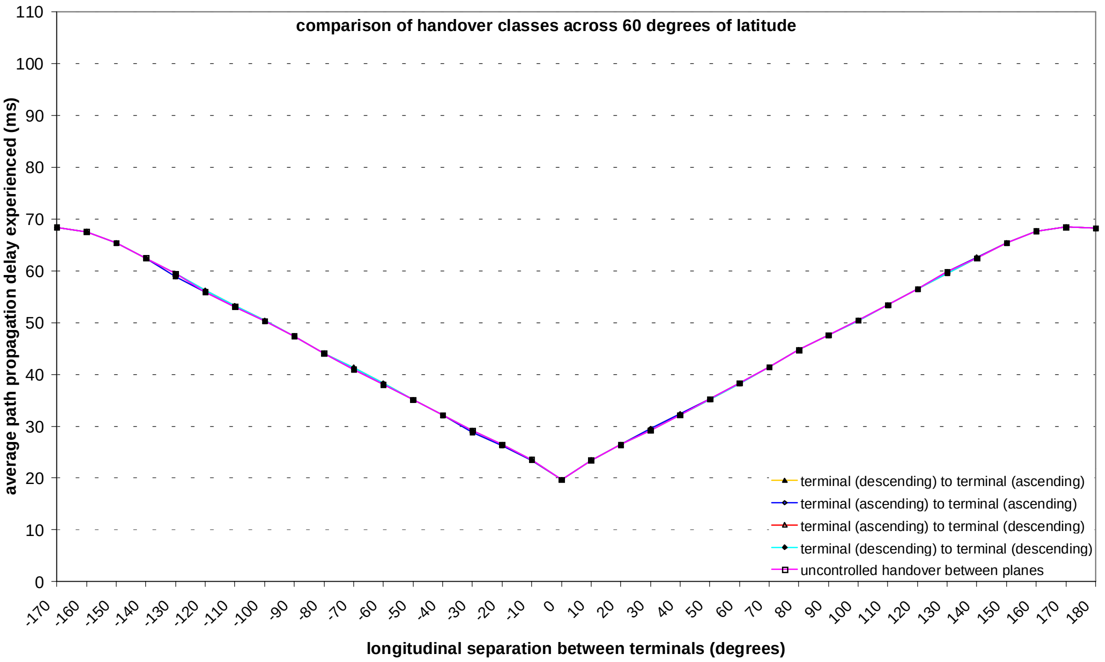


Figure 6.14 - different path delays between terminals at 60° latitude

As well as showing the difference in average delays, it is also worthwhile to examine the probability density functions (pdfs) for path delay samples recorded throughout the 24-hour simulation runs. These are shown in figures 6.15, 6.16 and 6.17.

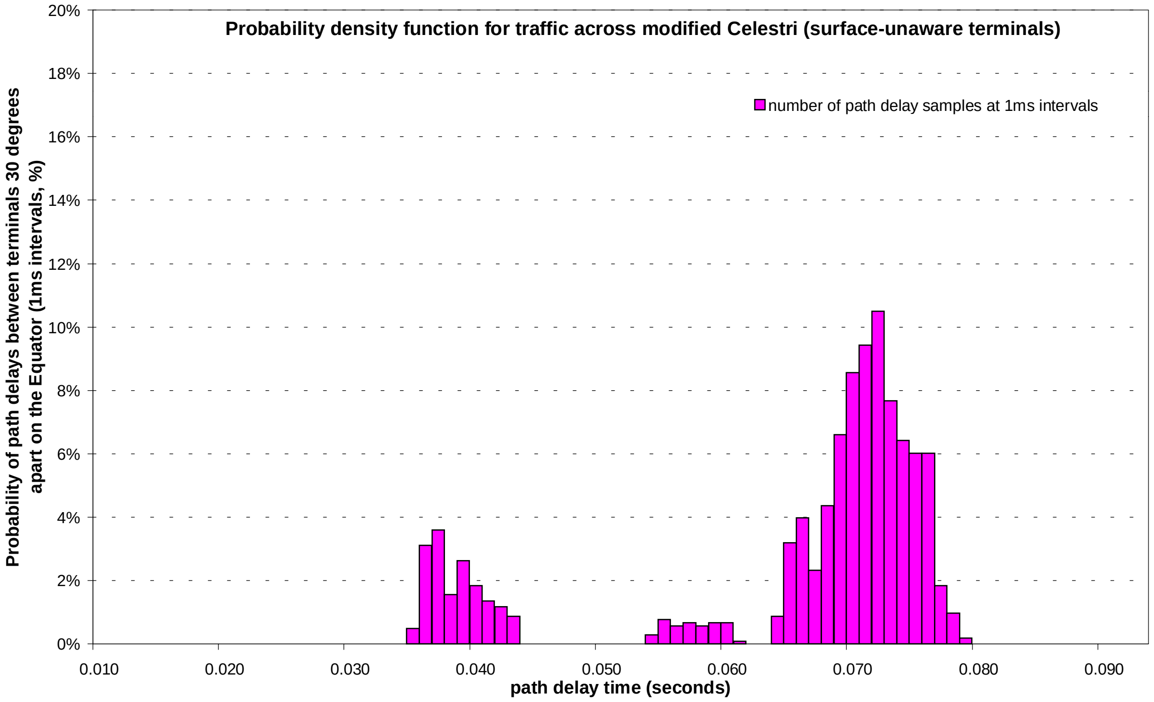


Figure 6.15 - probability density function of traffic (suface-unaware terminals)

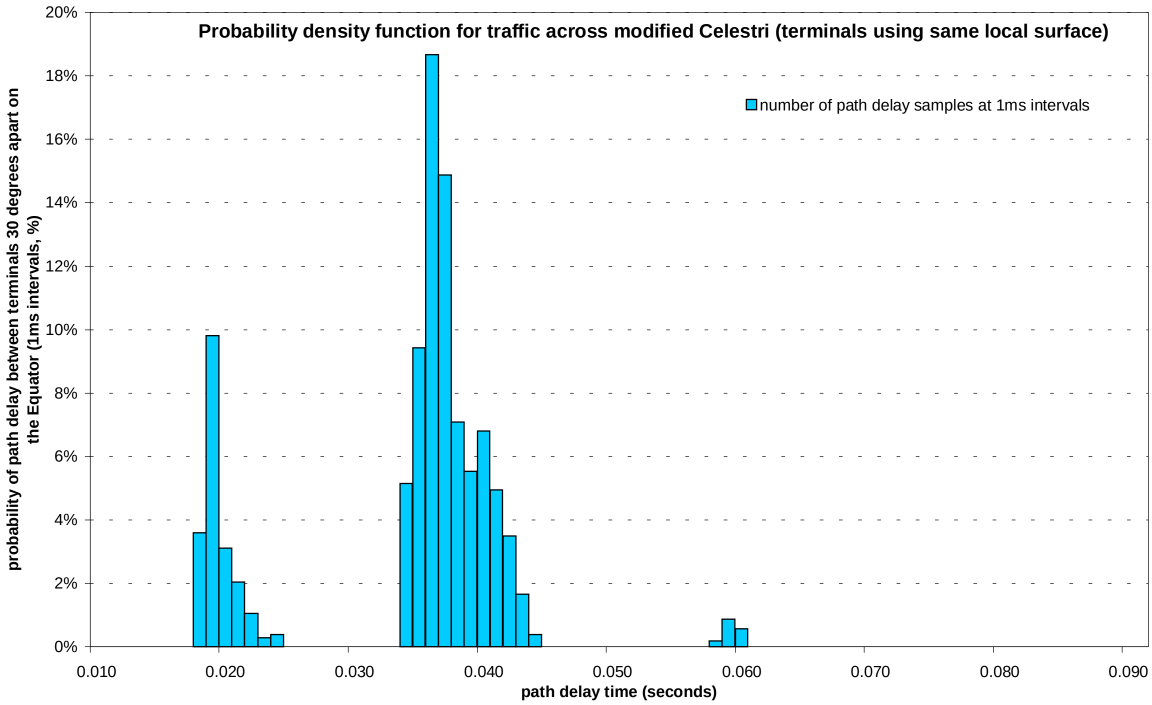


Figure 6.16 - probability density function of traffic (terminals sharing surface)

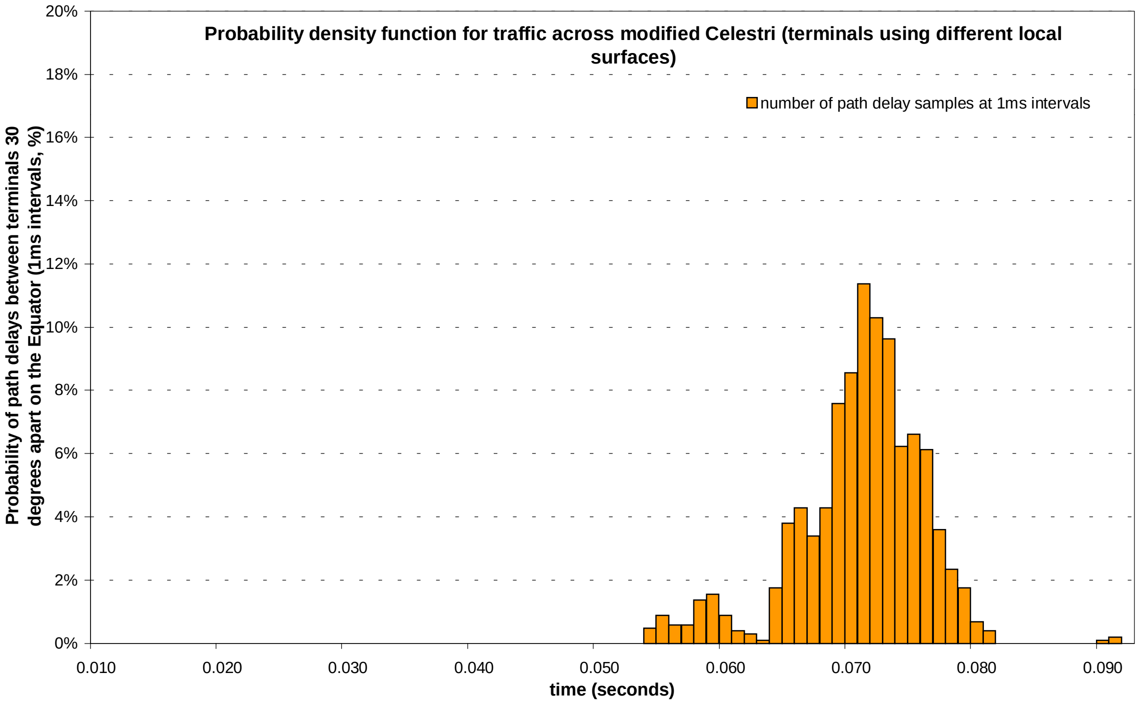


Figure 6.17 - probability density function of traffic (different-surface terminals)

It is clear from these pdf graphs that the overall set of path delay times experienced when handover does not discriminate between surfaces is split into two distinct low-and high-delay groups when a distinction is made between ascending and descending surfaces. Coordination of surface use permits access to an even lower set of delays not experienced when handover at both source and destination terminal is handover-agnostic, when the terminals share use of the same, single, satellite.

Samples showing transient spikes in delay due to traffic on the last ISL hop as handover takes place, as discussed in Chapter 2, are visible to the right of these graphs.

### 6.3.2 A comparison of *Celestri* and *Teledesic* designs

Having computed average delays between ground terminals across the latitudes of 0°, 30° and 60° for the modified *Celestri* design (this chapter) and for the Boeing *Teledesic* design (Chapter 2), the path delays for the two designs can now be compared. This is a reasonable comparison to make, since the constellation designs do share similar proposed altitudes: 1375km (*Teledesic*) vs 1400km (*Celestri*).

To show each proposal at its best, we can compare results achieving lowest average delays: the *Teledesic* design using cross-seam links against a modified *Celestri* design where a choice of surface preferences is expressed at terminal handover and both terminals share a surface. The comparison is shown graphically in Figure 6.18. At 60° latitude, the *Teledesic* design benefits from being able to use intra-plane links over the highest latitudes for large terminal separations.

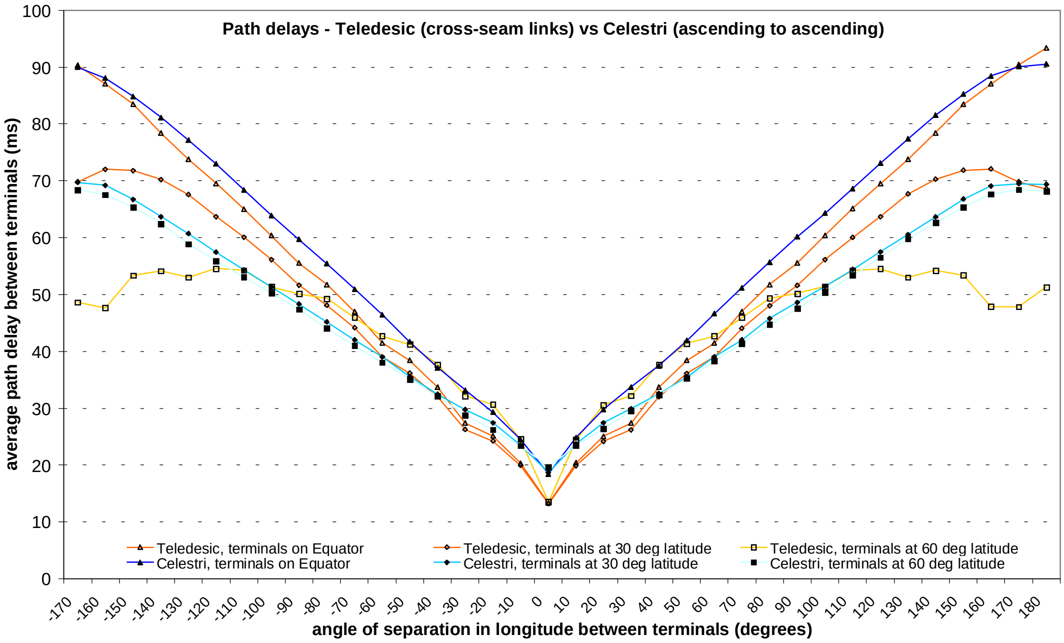


**Figure 6.18 - a comparison of average delays for *Teledesic* and *Celestri* variants**

Otherwise, the overall average path delays, over the course of a day, are quite similar for the non-polar latitudes where both systems provide coverage.

The larger number of active satellites in the *Teledesic* constellation (288 versus 63 for *Celestri*) and its geometry do not appear to give it a particular advantage in the range of average delays encountered. (Compare figures 2.17-2.19 with Appendix 4.)

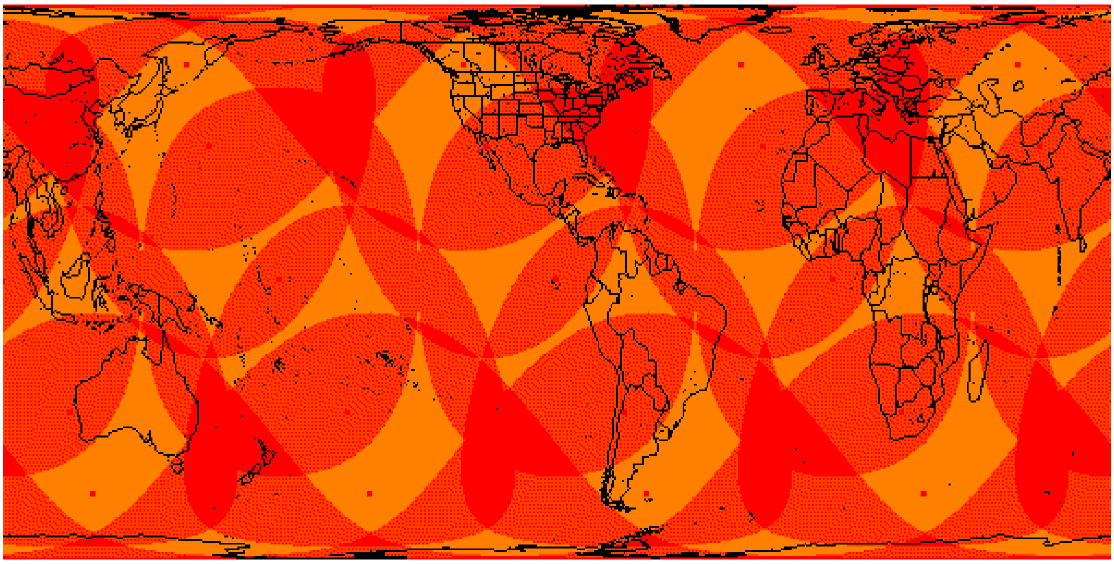
However, path delays for the *Teledesic* design are dependent on how cross-seam links are used, and are extremely sensitive to changes in their implementation, while handover in the modified *Celestri* design simply exploits available diversity carefully.

### 6.3.3 Using *Spaceway NGSO*

We also attempted to achieve full double surface coverage for a MEO constellation by modifying the proposed Hughes *Spaceway NGSO* constellation, described in [**FCCSpacewayNGO97**].

By lowering the minimum elevation angle of the constellation's ground terminals from the stated value of 30°, as shown in Figure 1.6, to 25°, a minimum of dual satellite coverage was achieved across the Earth's surface. This is shown in figure 6.19.

This, however, turns out not to be full double surface coverage throughout the course of a day; there are gaps in a surface's coverage between individual planes that are visible in path delays when terminals pass from one plane of satellites to another.



cylindrical projection, using minimum elevation angle of 25°. compare with Figure 1.6.

**Figure 6.19 - *Spaceway NGSO* constellation with minimum dual coverage (*SaVi*)**

This gap in surface coverage between planes forces terminal handover to be initiated from the current satellite to a satellite in a counter-rotating plane, from ascending to descending satellites or vice versa, so that some communication can be maintained. This alters delay significantly, and interrupts the otherwise clearly distinct delays resulting from different path lengths across the ISL mesh.

The effects of these gaps in full double surface coverage are shown in the delay traces in figure 6.20. As the Earth rotates under the orbital planes, one terminal reaches the edge of a plane's coverage and is forced to hand over to a counter-rotating mesh surface as the current surface that its plane is on can no longer provide it with coverage. This dramatically changes total path delay between terminals, even when a handover preference is obeyed to ensure handover to a corotating plane if one is available. Without diversity, handover is dictated by coverage.

The four gaps in double coverage between the four planes of *Spaceway NGSO* are clearly visible in the delay traces shown in figure 6.20. To compensate for this by making minimal changes to the *Spaceway NGSO* geometry, either:

- the altitude of the satellites can be increased to increase coverage, which is likely to be undesirable for MEO due to the proximity of the Outer Van Allen belt, or
- another orbital plane can be introduced so that the existing planes can be brought closer together and the gaps in double coverage between planes can be closed.

Closer planes were needed, and the second option directly addressed this. Adding an extra plane of five satellites to the existing *Spaceway NGSO* proposal's geometry of four planes increases the number of active satellites used by its constellation from 20 to 25, but also ensures that handover can always takes place between co-rotating planes.

The effect of this change on total path propagation delay between terminals is shown in figure 6.21. Five slight patterns of disruption are still visible in each delay trace at the times when terminals hand over from one plane to another, but the two sets of delay traces are clearly separated, showing distinct classes of path delay and thus service.

Any such alteration of an existing constellation design will require a reworking of link budgets, equipment requirements, frequency mappings and allocations for overlapping spotbeams, and so on.

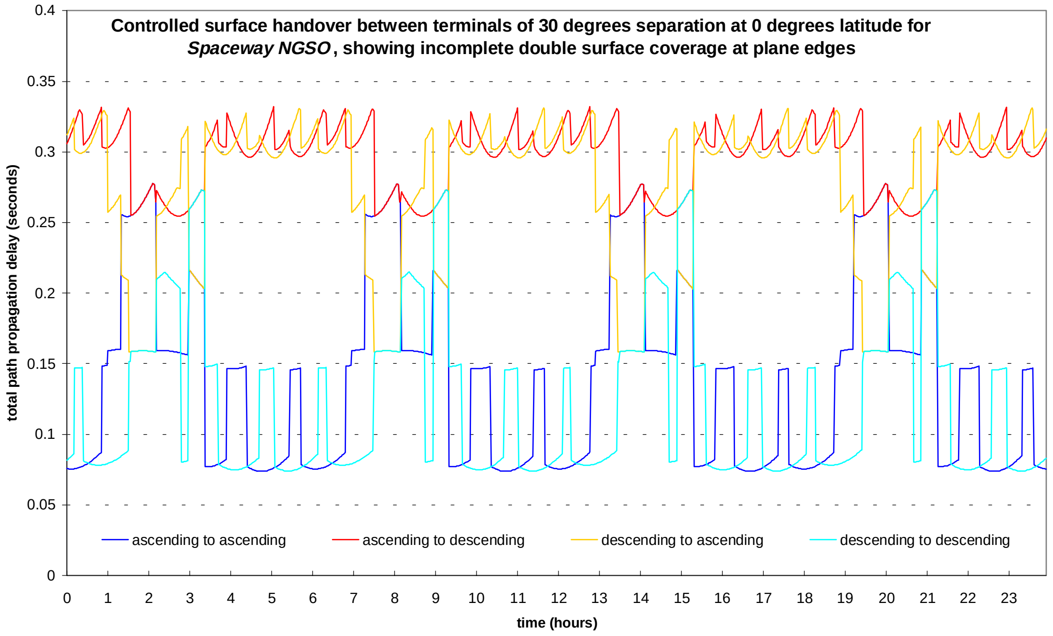


Figure 6.20 - *Spaceway NGSO* constellation showing lack of true double coverage

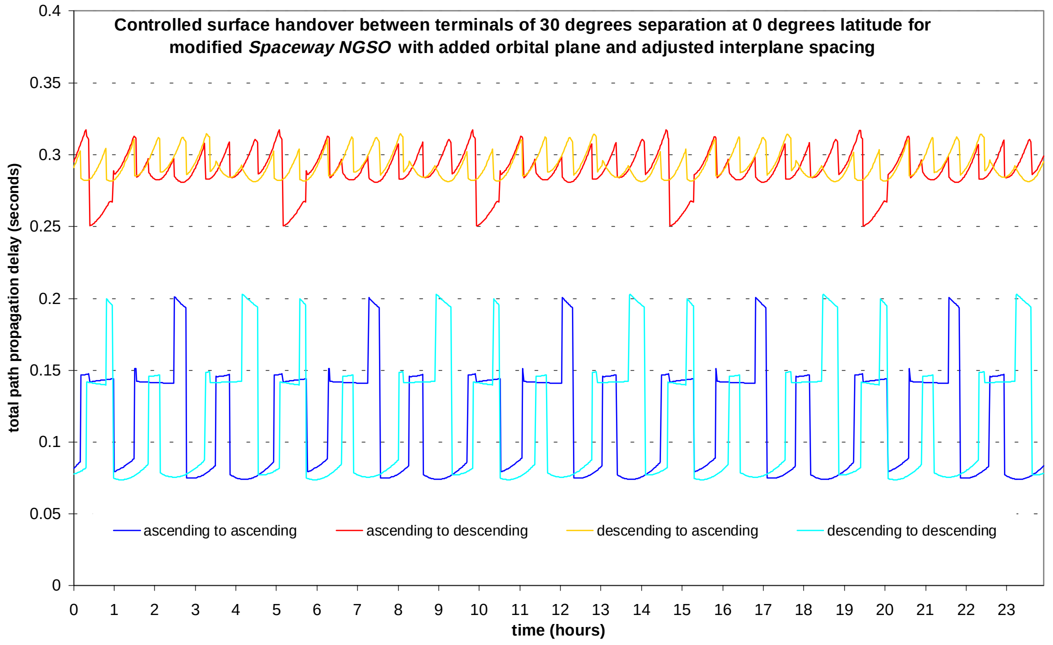


Figure 6.21 - adjusted *Spaceway NGSO* constellation with clear classes of service

From a networking and services viewpoint, it would seem sensible to begin with acceptable application latencies and traffic delay bounds as design constraints, and then to work from there to dimension classes of service for traffic, achievable path delays and then an acceptable constellation geometry and routing that meets the delay requirements. Equipment able to meet the needs of this top-down design specification can then be dimensioned.

However, the commercial proposals that have been examined are described in FCC applications for radio frequency allocation. The FCC application is often the first step towards obtaining worldwide allocations of frequency at a WRC. (FCC applications are also legal, rather than technical, documents, and this is reflected to some extent in their content.)

Given this, commercial proposals are driven by the need to meet power and interference requirements while working at specific frequencies in the ground-space interface, leading to a bottom-up approach to constellation geometry, the protocol stack, and the resulting constellation network design.

The discovery that these commercial proposals are not optimised for the network traffic they are intended to carry, and that they offer widely-varying delays, is not surprising.

## 6.4 Classes of service and reliability

The constellation network is homogenous in terms of ISL design and capacity. We know that the total path delay is roughly proportional to the number of hops the path traverses $n$ (as discussed in Chapter 2 and shown in figures 2.12-2.15). As there is always an uplink and downlink, $n$ is 2 or greater.

As the reliability of transmission across each hop is imperfect, we can express the probability of a packet successfully traversing the path as:

$$P_{\text{success}} = (1-d_{\text{uplink}})(1-d_{\text{isl}})^{n-2}(1-d_{\text{downlink}})$$

where $d_{\text{link}}$ is an estimation of the probability of the packet being discarded due to errors in transmission along the link or to any congestion at the queuing head of the link.

This introduces a reliability dimension to the use of two sets of paths, allowing different levels of guarantee of delivery. The longer set of paths constituting the second class of service can be expected to be generally slightly less reliable than the shorter, first, class of service; this must be considered in service specifications.

## 6.5 Handover, transients and network state

The terminal handover mechanism is also worth examining. Consider a single-homed ground terminal that does not exploit ground-space diversity, and which is undertaking handover to a co-rotating satellite, perhaps as part of a street of coverage so that the satellites are adjacent neighbours in the network mesh. Such a terminal handover will involve moving soft or hard network state from one satellite to another. This movement of state is clearly more straightforward for adjacent satellites than for when the terminal handover must be carried out to a counter-rotating satellite that is at some distant position in the rosette ISL mesh, particularly when e.g. assignment of new spotbeam capacity and coordinating timings for soft handover are taken into account. This must be considered for the hard state associated with virtual circuits, or for the soft state associated with multicast cores and addressing that has been discussed previously in Chapters 4 and 5.

Consider also the transient effects that were discussed at the end of Chapter 2. Any handover between counter-rotating planes will lead to large transient spikes for traffic in flight on the last ISL hop as handover takes place – either increased delays, or temporary losses as the in-flight traffic is discarded. Assuming no losses, that in-flight traffic will traverse a significant amount of the ISL mesh to reach its new destination terminal after handover. A rosette constellation with gaps in double coverage is analogous to a star constellation without cross-seam links as far as these transients are concerned. In fact, the rosette constellation is worse, in that seam-like disruptive handovers can take place at any point. (A rosette constellation relying on cross-seam links to join and interweave all neighbouring counter-rotating planes has not been considered in detail, and the topology of such a constellation would be constrained by the limitations of the cross-seam links.)

Full double coverage, where terminals handover never takes place to a counter-rotating satellite, minimises the difficulties and disruptions caused by handover in a rosette satellite constellation network with ISLs.

Use of ISLs in the rosette encourages a streets-of-coverage approach to constellation geometry, so that full double coverage can be achieved and disruptions in the satellite state and the ISL network due to terminal handover can be minimised.

## 6.6 Summary

Use of diversity and a careful approach to terminal handover in rosette constellations can create sets of multiple paths of different delays between ground terminals. Control of handover can go a very large way towards control of traffic in the constellation; obeying a handover preference, or exploiting multihoming, can provide ingress control for the rosette constellation network.

Detailed simulation of rosette constellation networks has shown that using terminal handover to exploit surface diversity is particularly useful for nearby terminals, with a considerable difference in delays between the choices of available path. This difference in delays decreases as the terminal locations approach the limits of coverage for the constellation, and as terminal separation approaches its maximum of 180° of longitude.

The commercial rosette constellations that were examined were not dimensioned to be able to offer this flexibility of service to traffic. This chapter showed how this flexibility of classes of delay for network traffic could be easily introduced by modifying the geometries of the commercial *Celestri* and *Spaceway NGSO* proposals slightly.

Average path delays across the simulated LEO *Celestri* and *Teledesic* constellation networks were compared. This showed that broadly similar path delays are offered to communicating ground terminals; a managed-handover *Celestri* rosette is similar to a *Teledesic* design with cross-seam links in average offered delay.

This novel approach relating terminal handover, diversity, and delay can significantly reduce path delays for traffic across the rosette constellation, as well as allowing sets of delays that permit clear classes of service. This approach minimises the disruption to network traffic and to the movement of network state between satellites that is introduced when terminal handover occurs.

# 7. Conclusions

Here, conclusions resulting from this work are summarised and the achievements of the work are presented, before further work that would build on this is discussed.

## 7.1 Summary

In examining satellite constellation networks with intersatellite links, this thesis has focused on measuring path propagation delays across simulated constellation networks, between ground terminals as endpoints, as a way of characterising the perceived performance of the constellation. These path delays reflect the properties of the constellation well, as they show continuous satellite motion as well as more abrupt handover and routing changes. Analysis of the resulting path delay statistics has shown that the impact of the orbital seam on path propagation delays experienced when traversing star constellation networks is decreased dramatically with the use of cross-seam links. Total delay and variation in delay are both reduced by the use of cross-seam links, making cross-seam links worthwhile to implement despite any technical or practical difficulties in doing so.

Handover events form a significant feature of LEO and MEO satellite constellation networks, although the degree of their impact on traffic and on movement of network state between satellites depends a great deal upon the design and implementation of the satellite constellation network. Use of terminal handover together with exploitation of diversity in rosette constellations can permit use of multiple paths of different delays between ground terminals. Terminal handover can be used to provide ingress control of traffic for the constellation network.

An examination of TCP traffic across satellite constellation networks has shown that using multiple paths to spread network load and to avoid packet drops due to congestion can adversely affect TCP throughput. This is due to interactions with the TCP congestion control algorithms, which tend to assume and are optimised for use with a single, ordered, flow of packets from sender to receiver. Delayed receiver acknowledgements can also affect TCP performance in this situation, although this is implementation-dependent. Avoiding any degradation of TCP performance requires a single, ordered, flow of traffic between source and destination, encouraging adoption of a complex flow-aware traffic engineering approach within the constellation network.

Use of IP routing in the constellation network offers benefits to the IP traffic that is routed. IP routing enables support for IP multicast and for IP quality of service, without the difficulties or complexities of attempting to map necessary IP router state and IP assumptions about the network onto an entirely different network layer. An approach to implementing IP multicast within the constellation network, using an ordered core-based-tree protocol, has been outlined, and an algorithm for placing the core has been presented and evaluated.

IP routing can be implemented in satellite constellation networks in a manageable fashion by selecting from a combination of border routing protocols, tunnelling, NAT and MPLS. As in terrestrial networks, use of NAT is problematic and offers few advantages. However, MPLS in particular appears to be a realistic and highly practical way of implementing support of advanced IP routing functionality on an ATM-based backbone likely to be found within a satellite constellation network, without adversely affecting support for other, non-IP, network protocols. MPLS decreases onboard routing complexity and removes the overhead of IP packet reassembly for header inspection.

## 7.2 Achievements

This work has analysed and simulated ‘star’ and ‘rosette’ constellation networks in detail, and explored their impact on network routing and on delay experienced by traffic travelling through the constellation network. This work has shown the desirability of cross-seam links for their ability to minimise delay and expected delay variation in star constellations. We have demonstrated the existence of interesting transient effects on traffic due to handover.

We have examined the performance of TCP traffic across the constellation, and determined the effects of multipath routing in the constellation network upon TCP.

We have proposed the novel use of an MPLS-based architecture to enable IP routing in the satellite constellation network, so that IP QoS and IP multicast can be supported. We have proposed an approach to multicast in the constellation, using an ordered core-based tree approach to shared multicast trees, and we have developed a novel algorithm that can be used to position a primary tree core to minimise delay and network use. We have evaluated use of this algorithm and shown the delay overhead imposed by use of a core-based multicast, and the capacity savings in network use resulting from the use of multicast. We have shown that use of core-based multicast within the satellite constellation network obeys the Chuang-Sirbu scaling law.

We have developed and analysed a novel method of controlling delay and traffic rerouting by managing handover decisions for terminals in a particular subset of the rosette constellation to provide service classes to traffic. It has been shown that existing commercial proposals could benefit from minor modifications in geometry in order to take advantage of this method.

A list of publications is provided in Appendix 1.

Programs and code used for the simulations presented here are publicly available for use by others. Contributions have been made to *SaVi* and to *ns*, as detailed in Appendix 2.

## 7.3 Further work

Presenting this work has merely scratched the surface of a variety of interesting fields.

In an attempt to inject a degree of realism into this work, this thesis has concentrated on simulating aspects of commercial proposals that are already described in some detail in the literature. No attempt has been made to simulate large classes of constellation networks, or to optimise constellation design or coverage to meet desired fixed constraints, such as total or maximum end-to-end delay, or degree of wander or jitter in delay, over a time period.

This thesis has established the relationship between the topology of the constellation network with ISLs and the Manhattan network, but has not been able to build on this to exploit this relationship. Further work would take existing results for Manhattan and toroidal networks from the fields of parallel computing and mathematics and apply them successfully to the constellation network.

Although an approach to handover decisions for terminals and multihoming for controlled delay across rosette constellations has been outlined, this has only been considered with shortest-path routing to provide two broad classes of delay via the four possible routes between endpoints. Further work would look at optimising the design of the constellation to meet particular delay requirements between points on the globe. It would combine these handover decisions for ground terminals with traffic engineering across multiple ISL paths, in an IP differentiated-services framework, to manage the IP traffic load on the constellation and provide a far wider range of specified classes of service.

This thesis has shown the degradation in TCP’s performance when faced with multipath routing that leads to reordering of received packets. Increasing TCP’s tolerance of packet reordering when packet loss does not occur is of interest. Development of algorithms to selectively raise and lower TCP’s dupack threshold safely, as large amounts of packet reordering are encountered or cease, would improve TCP’s performance.

The overheads of and complexities behind placing a multicast tree in the constellation network have not been examined in detail. Detailed analysis of state to be handed over and the disruption to the tree of moving the core, and how that disruption can be minimised, can be carried out. An algorithm to place a primary core in the rosette constellation network, where member satellites are both ascending and descending due to the handover decisions made by the ground terminals, has not been worked out in full successfully. It is likely to require the use of four-dimensional vectors to compute the core vector, as well as requiring the full double coverage that is discussed in Chapter 6.

The approach of using network topology and member positions to generate a reasonable location for the core of a multicast tree, with or without transformations to other coordinate systems, may be more generally applicable to a wider variety of types of networks. Projecting a network topology of a genus higher than the topology needed to specify user locations onto those user locations to ensure multiple redundant separate paths between users may be generally useful; Chapter 6 has shown the benefits resulting from mapping a toroidal network (of genus 1) onto users located on a sphere (of genus 0).

Interactions between the constellation network, as an autonomous system, and an Earth's surface that is filled with networks that are themselves autonomous systems are extremely interesting from an operational and management perspective. How BGP implementations in constellation gateway stations can peer, share information, aggregate and advertise routes, and how they can do so gracefully in the presence of failure of parts of the constellation network, is easily a research area in itself.

This thesis has presented a network architecture for the constellation, based around the use of MPLS. Also of interest are how MPLS might be implemented across the ground-space interface, so that traffic engineering can be carried out directly at ground terminals. Ingress control methods, using detailed knowledge of the constellation's properties and traffic load to optimise use of the limited capacity in the all-important ground-space interface, are worthy of investigation.

In taking an 'abstract' network-layer view of the constellation, this thesis has neglected a number of interesting topics at lower layers, such as optimising satellite footprint or spotbeam capacity management and media access and allocation techniques in the uplink and downlink.

Issues at lower layers in the protocol stack, such as spotbeam frequency assignment schemes to avoid interference and gain optimal frequency reuse, are out of the scope of this thesis, but will be affected by the work in this thesis. The effects on spotbeam frequency assignment of e.g. a rosette constellation designed for use with the handover surface-selection method, with its overlapping spotbeams giving the necessary diversity, have not been examined. Work is needed on ascertaining how these lower layers are affected by changes in the constellation that are dictated by the needs of higher layers.

That is future work. This work must be said to be completed at some point, and here is a better place than most.

# *References*

**AdRid87**, W. S. Adams and L. Rider, 'Circular polar constellations providing continuous single or multiple coverage above a specified latitude', *Journal of the Astronautical Sciences*, vol. 35 no. 2, pp. 155-192, April-June 1987.

**AdVer94**, V. S. Adve and M. K. Vernon, 'Performance analysis of mesh interconnection networks with deterministic routing', *IEEE Transactions on Parallel and Distributed Systems*, vol. 5 no. 3, pp. 225-246, March 1994.

**Allman98**, M. Allman, 'On the Generation and Use of TCP Acknowledgments', ACM SIGCOMM *Computer Communication Review*, vol. 28 no. 5, pp. 4-21, October 1998.

**AllmanOstKruse95**, M. Allman, S. Ostermann and H. Kruse, 'Data transfer efficiency over satellite circuits using a multi-socket extension to the file transfer protocol (ftp)', ACTS Results Conference, September 1995.

**AllmanOstKruse96**, M. Allman, S. Ostermann and H. Kruse, 'An application-level solution to TCP's satellite inefficiencies', Proceedings of the First ACM/IEEE International Workshop on Satellite-based Information Services (WOSBIS), pp. 100-107, November 1996.

**Almeroth00**, K. C. Almeroth, 'The Evolution of Multicast: From the MBone to Interdomain Multicast to Internet2 Deployment', *IEEE Network*, vol. 14 no. 1, pp. 37-52, January/February 2000.

**AlmZhang98**, K. C. Almeroth and Y. Zhang, 'Using Satellite Links as Delivery Paths in the Multicast Backbone', Proceedings of the Third ACM/IEEE International Workshop on Satellite-based Information Services (WOSBIS '98), pp. 47-54, Dallas, Texas, October 1998.

**AndrikopoulosPavlou99**, I. Andrikopoulos and G. Pavlou, 'Providing Differentiated Services to MPLS Networks', Proceedings of the 7th IEEE/IFIP International Workshop on Quality of Service (IWQoS '99), pp. 207-215, London, June 1999.

**Armitage00**, G. Armitage, 'MPLS: the magic behind the myths', *IEEE Communications Magazine*, vol. 38 no. 1, pp. 124-131, January 2000.

**ArSupBarDil96**, V. Arora, N. Suphasindhu, J. S. Baras and D. Dillon, ‘Effective Extensions of Internet in Hybrid Satellite-Terrestrial Networks’, technical report TR 96-20, Center for Satellite and Hybrid Communication Networks, University of Maryland. Revised version published in Proceedings of the 1st Conference of Commercial Development of Space, Part One, pp. 339-344, Albuquerque, New Mexico, 7-11 January 1996.

**Bajajetal96**, S. M. Bajaj, C. Brazdziunas, D. E. Brooks, D. F. Daly, S. M. Srinidhi, T. Robe and F. Vakil , ‘Performance characterization of TCP/IP-on-ATM over an ATM/SONET high data rate ACTS channel’, Proceedings of the 16th AIAA International Communications Satellite Systems Conference (ICSSC), pp. 1008-1019, February 1996.

**Ballard80**, A. H. Ballard, ‘Rosette Constellations of Earth Satellites’, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 16 no. 5, pp. 656-673, September 1980.

**BallFranCrow93**, A. J. Ballardie, P. F. Francis and J. Crowcroft, ‘Core-based trees (CBT): An Architecture for Scalable Inter-Domain Multicast Routing’, Proceedings of ACM SIGCOMM, pp. 85-95, 1993.

**BanBorGer92**, J. Bannister, F. Brogonova and M. Gerla, ‘A procedure to evaluate the mean transport time in multibuffer defelction-routing networks with non-uniform traffic’, Proceedings of INFOCOM ’92, pp. 1069-1078, Florence, Italy, May 1992.

**Baran64**, P. Baran, ‘On distributed communications: I. Introduction to distributed communications network’, memorandum RM-3420-PR, the Rand corporation, August 1964.

**Bennettetal99**, J. C. R. Bennett, C. Partridge and N. Shectman, ‘Packet Reordering is not Pathological Network Behaviour’, *IEEE/ACM Transactions on Networking*, vol. 7. no. 6, pp. 789-798, December 1999.

**Bisante99**, H. Cruickshank (ed.), Z. Sun, T. Örs, L. Wood, R. Dhaou, M. Becker and C. Xenakis, ‘LEO Satellite Network Characteristics’, public deliverable 2.3, EU Esprit Bisante project, August 1999.

**BorJohDar99**, J. Börjesson, J. Johansson and F. Darin, 'GLONASS: experiences from the first global campaign', Radio Vetenskap och Kommunikation 1999 (RVK '99), Karlskrona, Sweden, June 1999.

**Cainetal87**, J. B. Cain, S. L. Adams, M. D. Noakes, T. Kryst and E. L. Althouse, 'A "near-optimum" multiple path routing algorithm for space-based SDI networks', Proceedings of MILCOM '87, pp. 578-584, October 1987.

**Callonetalmplsdraft99**, R. Callon et al., 'A Framework for Multiprotocol Label Switching', work in progress as internet-draft, IETF MPLS working group, September 1999.

**CastielDraim95**, D. Castiel and J. E. Draim, 'The Ellipso mobile satellite system', Proceedings of the International Mobile Satellite Conference 1995 (IMSC '95), pp. 409-419, Ottawa, Canada, 6-8 June 1995.

**ChalmersAlmeroth00**, R. C. Chalmers and K. C. Almeroth, 'Developing a multicast metric', Proceedings of IEEE Globecom 2000, San Francisco, December 2000, pp. 382-386. An earlier version appeared as a Stardust.com white paper.

**ChuangSirbu98**, I. J. Chuang and M. A. Sirbu, 'Pricing Multicast Communication: A Cost Based Approach', Proceedings of INET '98, Geneva, Switzerland, July 1998.

**ChungRaiAg89**, T. Y. Chung, S. Rai and D. P. Agrawal, 'A routing scheme for datagram and virtual circuit services in the Manhattan Street Network', Proceedings of the 8th International Phoenix Conference on Computers and Communications, March 1989.

**ChungSharAg94**, T. Y. Chung, N. Sharma and D. P. Agrawal, 'Cost-performance trade-offs in Manhattan Street Network versus 2D Torus', *IEEE Transactions on Computers*, vol. 43 no. 2, pp. 240-243, February 1994.

**Clarke45**, A. C. Clarke, 'Extra-terrestrial relays', *Wireless World*, pp. 305-308, October 1945.

**CompRam97**, G. Comparetto and R. Ramirez, 'Trends in Mobile Satellite Technology', IEEE Computer, vol. 30 no. 2, pp. 44-52, February 1997.

**Deering91**, S. Deering, 'Multicast routing in a datagram internetwork', Stanford University PhD thesis, December 1991.

**Dietrich98**, F. Dietrich, The Globalstar Satellite Cellular Communication System - Design and Status', Proceedings of the 17th AIAA International Communications Satellite Systems Conference (ICSSC), pp. 47-53, Yokohama, Japan, February 1998, AIAA-98-1213.

**Dondl84**, P. Dondl, 'LOOPUS opens a new dimension in satellite communications', *International Journal of Satellite Communications*, vol. 2 no. 4, pp. 241-250, October-December 1984.

**Draim89**, J. E. Draim, 'Satellite continuous coverage constellations', US patent 4,809,935, 'Elliptical satellite system which emulates the characteristics of geosynchronous satellites', issued 7 March 1989, and related patents.

**DraimCefCas00**, J. E. Draim, P. J. Cefola and D. Castiel, 'Elliptical orbit constellations – a new paradigm for higher efficiency in space systems?', Proceedings of 2000 IEEE Aerospace Conference, Big Sky, Montana, 18-25 March 2000.

**DurstMillTrav96**, R. C. Durst, G. J. Miller and E. J. Travis, 'TCP Extensions for Space Communications', Mobicom '96, November 1996.

**EkiciAkBen00**, E. Ekici, I. F. Akyildiz and M. D. Bender, 'Datagram routing algorithm for LEO satellite networks', Proceedings of IEEE INFOCOM 2000, pp. 500-508, Tel Aviv, March 2000.

**Elizondoetal97**, E. Elizondo, R. Gobbi, A. Modelfino and F. Gargione, 'Evolution of the Astrolink System', Proceedings of the third Ka-band utilization conference, pp. 3-7, 15-18 September 1997.

**Eriksson94**, H. Eriksson, 'MBone: The Multicast Backbone', *Communications of the ACM*, vol. 37, pp. 54-60, August 1994.

**Evans00a**, J. V. Evans, 'The US proposed new multimedia communications satellite systems', Proceedings of 2000 IEEE Aerospace Conference, Big Sky, Montana, 18-25 March 2000.

**Evans00b**, J. V. Evans, 'The US filings for multimedia satellites: a review', *International Journal of Satellite Communications*, vol. 18 no. 3, pp. 121-160, May/June 2000.

**Facheuretalmplsdraft00**, F. Facheur et al., 'MPLS support of differentiated services', work in progress as internet-draft, IETF MPLS working group, February 2001.

**FairWoodpilcdraft00**, G. Fairhurst and L. Wood, ‘Link ARQ issues for IP traffic’, work in progress as internet-draft, IETF PILC working group, March 2001.

**FallVaradhan00**, K. Fall and K. Varadhan (editors), *ns* manual / ‘ns Notes and Documentation’, VINT project documentation, available with the network simulator *ns* from:

http://www.isi.edu/nsnam/

**Falk94**, A. D. Falk, ‘A system design for a hybrid network terminal using asymmetric TCP/IP to support Internet applications’, Master’s thesis, Institute for Systems Research, University of Maryland, 1994.

**Farinaccietaldraft00**, D. Farinacci et al., ‘Multicast Source Discovery Protocol’, work in progress as internet-draft intended as standards track, MSDP working group, July 2000.

**FCCCelestri97**, M. D. Kennedy, B. Lambergman et al., ‘Application to construct, launch and operate the Celestri multimedia LEO system’, filing with the US Federal Communications Commission, Motorola Global Communications, Inc., June 1997.

**FCCPentriad97**, D. J. Burnett et al., ‘Application of Denali Telecom LLC – Consolidated system proposal for authority to launch and operate thirteen satellites in the Pentriad system’, filing with the US Federal Communications Commission, Denali Telecom LLC, 26 September 1997.

**FCCSkyBridge97**, P. L. Spector, J. H. Olson et al., ‘Application of SkyBridge LLC for authority to launch and operate the SkyBridge system’, filing with the US Federal Communications Commission, SkyBridge LLC, 28 February 1997.

**FCCSpacewayNGSO97**, F. A. Taormina et al., ‘Application of Hughes Communications, Inc. for authority to launch and operate Spaceway NGSO, an NGSO expansion to the Spaceway global broadband satellite system’, filing with the US Federal Communications Commission, Hughes Communications, Inc., 22 December 1997.

**FCCVirtualGEO99**, D. Castiel et al., ‘Application of Virtual Geosatellite LLC for authority to launch and operate the Virtual GEO satellite system’, filing with the US Federal Communications Commission, Virtual Geosatellite LLC, 8 January 1999.

**Fitzpatrick95a**, E. J. Fitzpatrick, ‘Spaceway system summary’, *Space Communications*, vol. 13, pp. 7-23, 1995.

**Fitzpatrick95b**, E. J. Fitzpatrick, ‘Spaceway: Providing Affordable and Versatile Telecommunication Solutions’, *Pacific Telecommunications Review*, vol. 17 no. 1, September 1995.

**Floydetal97**, S. Floyd, V. Jacobson, C. Liu, S. McCanne and L. Zhang, ‘A reliable multicast framework for light-weight sessions and application level framing’, *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 784-803, December 1997.

**Fraiseetal00**, P. Fraise. B. Coulomb, B. Monteuuis and J.-L. Soula, ‘SkyBridge LEO satellites: optimized for broadband communications in the 21st century’, Proceedings of 2000 IEEE Aerospace Conference, Big Sky, Montana, 18-25 March 2000.

**Galtier99**, J. Galtier, ‘Geographical reservation for guaranteed handover and routing in low earth orbit constellations’, Proceedings of Workshop de Comunicacao Sem Fio Belo Horizonte 1999, (WCSF ’99), Brazil, July 1999. (The author suggested the nomenclature used to Jérôme while at ECOTEL ’98.)

**Ghediaetal99**, L. Ghedia, K. Smith and G. Titzer, ‘Satellite PCN - the ICO system’, *International Journal of Satellite Communications*, Special Issue: LEOs – Little and Big, vol. 17 no. 4, pp. 273-289, July/August 1999.

**GkizeliTaagEv00**, M. Gkizeli, Payam Taaghol and B. G. Evans, ‘Service availability of LEO/MEO satellite systems’, *International Journal of Satellite Communications,* Special Issue: LEOs – Little and Big, vol. 17. no. 4, pp. 291-302, July/August 1999.

**Goldman99**, E. Goldman, ‘Little LEOs serve an unmet demand: Leo One system architecture optimized to meet market requirements’, *International Journal of Satellite Communications,* Special Issue: LEOs – Little and Big, vol. 17 no. 4, pp. 225-242, July/August 1999.

**GoodGreen86**, J. Goodman and A. G. Greenberg, ‘Sharp approximate models of adaptive routing in mesh networks’, preliminary report, Teletraffic analysis and computer performance evaluation, O. J. Boxma et al. (ed), Elsevier Science, pp. 255-270, 1986.

**GrossRama97**, M. Grossglauser and K. K. Ramakrishnan, ‘SEAM: Scalable and Efficient ATM Multicast’, Proceedings of IEEE INFOCOM ’97, Kobe, Japan, 7-11 April, 1997.

**Hardman91**, G. E. Hardman, ‘Engineering Orbcomm: a digital satellite communications system exploiting a range of modern technologies’, Proceedings of the third IEE Conference on Telecommunications, pp. 251-256, Edinburgh, 1991.

**Henderson99**, T. R. Henderson, ‘Networking over Next-Generation Satellite Systems’, PhD dissertation, Computer Science Division, University of California at Berkeley, Fall 1999.

**Henderson00a**, T. R. Henderson and R. H. Katz, ‘Network Simulation for LEO Satellite Networks’, Proceedings of the 18th AIAA International Communications Satellite Systems Conference (ICSSC), Oakland, California, 10-14 April, 2000.

**Henderson00b**, T. R. Henderson and R. H. Katz, ‘On distributed, geographic-based packet routing for LEO satellite networks’, Satellite Communications for the New Millennium symposium, Proceedings of IEEE Globecom 2000, San Francisco, November 2000.

**HutchLaurin95**, J. Hutcheson and M. Laurin, ‘Network flexibility of the Iridium global mobile satellite system’, Proceedings of the International Mobile Satellite Conference 1995 (IMSC ’95), pp. 503-507, Ottawa, Canada, 6-8 June 1995.

**ISO7498**, ISO (1984), ‘Basic Reference Model for Open Systems Interconnection’, Recommendation X.200 of the International Telecommunications Union.

**Jacob88**, V. Jacobson, ‘Congestion avoidance and control’, Proceedings of SIGCOMM ’88, Palo Alto, August 1988. Also in *ACM Computer Communication Review*, vol. 18, pp. 314-329, August 1988. (Revised slightly with M. J. Karels, 1992.)

**Jamoussietalmplsdraft99**, B. Jamoussi et al., ‘Constraint-based LSP setup using LDP’, work in progress as internet-draft, IETF MPLS working group, July 2000.

**KalyJainetal98**, S. Kalyanaraman, R. Jain, R. Goyal, S. Fahmy and S.-C. Kim, ‘Use-it or Lose-it Policies for the Available Bit Rate (ABR) Service in ATM Networks’, *Computer Networks and ISDN Systems*, vol. 30, no. 24, pp. 2293-2308, 14 December 1998.

**KeshavSharma98**, S. Keshav and R. Sharma, ‘Issues and Trends in Router Design’, *IEEE Communications Magazine*, vol. 36 no. 5, pp. 144-151, May 1998.

**Kohn97**, D. M. Kohn, ‘Providing global broadband Internet access using low-earth-orbit satellites’, *Computer Networks and ISDN Systems*, vol. 29 no. 15, pp. 1763-1768, November 1997.

**KrewelMaral00**, W. Krewel and G. Maral, ‘Analysis of the impact of handover strategies on the QoS of satellite diversity based communications systems’, Proceedings of the 18th AIAA International Communications Satellite Systems Conference (ICSSC), pp. 393-403, Oakland, 11-14 April 2000, AIAA-2000-1220.

**Kruesi96**, F. Kruesi, ‘The Global Positioning System: a DOT perspective of where we are and where we are going’, Proceedings of the Institute of Navigation GPS-96, Kansas City, Missouri, pp. 3-6, September 1996.

**Kumaretal98**, S. Kumar, P. Radoslavov, D. Thaler, C. Alaettinoglu, D. Estrin and M. Handley, ‘The MASC/BGMP Architecture for Inter-Domain Multicast Routing’, Proceedings of ACM SIGCOMM ’98, Vancouver, September 1998.

**Leopold91**, R. J. Leopold, ‘Low-earth orbit global cellular communications network’, Proceedings of ICC ’91, pp. 1108-1111, 1991.

**LeopoldMill93**, R. J. Leopold and A. Miller, ‘The Iridium communications system’, *IEEE Potentials*, vol. 12 no. 2, pp. 6-9, April 1993.

**LeopoldMillGrubb93**, R. J. Leopold, A. Miller and J. L. Grubb, ‘The Iridium system: a new paradigm in personal communications’, *Applied Microwave and Wireless*, vol. 5 pt 4, pp. 68-78.

**Liron00**, M. L. Liron, ‘Traffic routing for satellite communication system’, US patent 6,084,864, issued 4 July 2000.

**Luders61**, R. D. Lüders, ‘Satellite networks for continuous zonal coverage’, *American Rocket Society Journal*, vol 31, pp. 179-184, February 1961.

**MakSmith98**, F. Makita and K. Smith, ‘Design and implementation of ICO system’, Proceedings of the 17th AIAA International Communications Satellite Systems Conference (ICSSC), pp. 57-65, Yokohama, Japan, February 1998, AIAA-98-1216.

**Maral95**, G. Maral, ‘VSAT Networks’, J. Wiley & Sons, 1995.

**MaralBousquet98**, G. Maral and M. Bousquet, ‘Satellite communications systems’, third edition, J. Wiley & Sons, 1998.

**MaugRosen97**, R. Mauger and C. Rosenberg, ‘QoS Guarantees for Multimedia Services on a TDMA-Based Satellite Network’, *IEEE Communications Magazine*, pp. 56-65, July 1997.

**Maxemchuk87**, N. F. Maxemchuk, ‘Routing in the Manhattan street network’, *IEEE Transactions on Communications*, vol. 35 no. 5, pp. 503-512, May 1987.

**Mazur99**, S. Mazur, ‘A description of current and planned location strategies within the ORBCOMM network’, *International Journal of Satellite Communications*, Special Issue: LEOs – Little and Big, vol. 17 no. 4, pp. 209-223, July/August 1999.

**Mertzanis99**, I. Mertzanis, G. Sfikas, R. Tafazolli and B. G. Evans, ‘Protocol architectures for satellite ATM broadband networks’, *IEEE Communications Magazine*, vol. 37 no. 3, pp. 46-54, March 1999.

**MyZar90**, G. E. Myers and M. El Zarki, ‘Routing in TAC - a triangularly-arranged network’, Proceedings of IEEE INFOCOM ’90, pp. 481-486, San Francisco, June 1990.

**Narvaezetal98**, P. Narvaez, A. Clerget, and W. Dabbous, ‘Internet Routing over LEO Satellite Constellations’, Proceedings of the Third ACM/IEEE International Workshop on Satellite-Based Information Services (WOSBIS ’98), pp. 89-95, October 1998.

**Obraczka98**, K. Obraczka, ‘Multicast transport protocols: a survey and taxonomy’, *IEEE Communications Magazine*, vol. 36 no. 1, pp. 94-102, January 1998.

**PartridgeShepard97**, C. Partridge and T. Shepard, ‘TCP Performance over Satellite Links’, *IEEE Network*, vol. 11 no. 5, pp. 44-49, September/October 1997.

**PhilShenTang99**, G. Philips, S. Shenker and H. Tangmunarunkit, ‘Scaling of Multicast Trees: Comments on the Chuang-Sirbu scaling law’, Proceedings of ACM SIGCOMM ’99, August 1999.

**Ramalho00**, M. Ramalho, ‘Intra- and Inter-Domain Multicast Routing Protocols: A Survey and Taxonomy’, *IEEE Communication Surveys*, vol. 3 no. 1, first quarter 2000.

**RFC826**, D. Plummer, ‘An Ethernet address resolution protocol, or converting network protocol addresses to 48-bit Ethernet address for transmission on Ethernet hardware’, RFC 826, standard, also STD 0037, November 1982.

**RFC829**, V. G. Cerf, ‘Packet satellite technology reference sources’, IETF RFC 829, DARPA, November 1982.

**RFC896**, J. Nagle, ‘Congestion control in IP/TCP internetworks’, IETF RFC 896, January 1984.

**RFC1075**, D. Waitzman, C. Partridge and S. Deering, ‘Distance Vector Multicast Routing Protocol’, IETF RFC 1075, experimental, November 1988.

**RFC1112**, S. Deering, ‘Host extensions for IP multicasting’, IETF RFC 1112, also STD 0005, standard, August 1989.

**RFC1191**, J. Mogul and S. Deering, ‘Path MTU Discovery’, IETF RFC 1191, draft standard, November 1990.

**RFC1301**, S. Armstrong, A. Freier and K. Marzullo, ‘Multicast transport protocol’, IETF RFC 1301, informational, February 1992.

**RFC1323**, V. Jacobson, R. Braden and D. Borman, ‘TCP extensions for high performance’, IETF RFC 1323, proposed standard, May 1992.

**RFC1403**, K. Varadhan, ‘BGP OSPF Interaction’, IETF RFC 1403, proposed standard, January 1993.

**RFC1458**, R. Braudes and S. Zabele, ‘Requirements for multicast protocols’, IETF RFC 1458, informational, May 1993.

**RFC1584**, J. Moy, ‘Multicast Extensions to OSPF’, IETF RFC 1584, proposed standard, March 1994.

**RFC1633**, R. Braden, D. Clark and S. Shenker, ‘Integrated Services in the Internet Architecture: an Overview’, IETF RFC 1633, informational, June 1994.

**RFC1722**, G. Malkin, ‘RIP Version 2 Protocol Applicability Statement’, IETF RFC 1722, standard, also STD 0057, November 1994.

**RFC1771**, Y. Rekhter and T. Li, ‘A Border Gateway Protocol 4 (BGP-4)’, IETF RFC 1771, draft standard, March 1995.

**RFC1772**, Y. Rekhter and P. Gross, ‘Application of the Border Gateway Protocol in the Internet’, IETF RFC 1772, draft standard, March 1995.

**RFC1853**, W. Simpson, ‘IP in IP Tunnelling’, IETF RFC 1853, informational, October 1995.

**RFC1981**, J. McCann, S. Deering and J. Mogul, ‘Path MTU Discovery for IP version 6’, IETF RFC 1981, proposed standard, August 1996.

**RFC2001**, W. Stevens, ‘TCP Slow Start, Congestion Avoidance, Fast Retransmit and Fast Recovery algorithms’, IETF RFC 2001, proposed standard later obsoleted by IETF RFC 2581, January 1997.

**RFC2018**, M. Mathis, J. Mahdavi, S. Floyd and A. Romanov, ‘TCP selective acknowledgement options’, IETF RFC 2018, proposed standard, October 1996.

**RFC2022**, G. Armitage, ‘Support for multicast over UNI 3.0/3.1 based ATM networks’, IETF RFC 2022, proposed standard, October 1996.

**RFC2149**, R. Talpade and M. Ammar, ‘Multicast Server Architectures for MARS-based ATM multicasting’, IETF RFC 2149, informational, May 1997.

**RFC2189**, A. Ballardie, ‘Core Based Trees (CBT version 2) Multicast Routing Protocol’, IETF RFC 2189, experimental, September 1997.

**RFC2191**, G. Armitage, ‘VENUS - Very Extensive Non-Unicast Service’, IETF RFC 2191, informational, September 1997.

**RFC2201**, A. Ballardie, ‘Core Based Trees (CBT) Multicast Routing Architecture’, IETF RFC 2201, experimental, September 1997.

**RFC2205**, R. Braden, L. Zhang, S. Berson, S. Herzog and S. Jamin, ‘Resource ReSerVation Protocol (RSVP) Version 1 Functional Specification’, IETF RFC 2205, proposed standard, September 1997.

**RFC2208**, A. Mankin, F. Baker, B. Braden, S. Bradner, M. O’Dell, A. Romanow, A Weinrib and L. Zhang, ‘Resource ReserVation Protocol (RSVP) Version 1 Applicability Statement’, IETF RFC 2208, informational, September 1997.

**RFC2210**, J. Wroclawski, ‘The Use of RSVP with IETF Integrated Services’, IETF RFC 2210, proposed standard, September 1997.

**RFC2225**, M. Laubach and J. Halpern, ‘Classical IP and ARP over ATM’, IETF RFC 2225, proposed standard, April 1998.

**RFC2236**, W. Fenner, ‘Internet Group Management Protocol, Version 2’, IETF RFC 2236, proposed standard, November 1997.

**RFC2328**, J. Moy, ‘OSPF Version 2’, IETF RFC 2328, standard, also STD 0054, April 1998.

**RFC2362**, D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma and L. Wei, ‘Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification’, IETF RFC 2362, experimental, June 1998.

**RFC2401**, S. Kent and R. Atkinson, ‘Security Architecture for the Internet Protocol’, IETF RFC 2401, proposed standard, November 1998.

**RFC2453**, G. Malkin, ‘RIP version 2’, IETF RFC 2453, Internet standard STD0056, November 1998.

**RFC2474**, K. Nichols, S. Blake, F. Baker and D. Black, ‘Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers’, IETF RFC 2474, informational, December 1998.

**RFC2475**, S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, ‘An Architecture for Differentiated Services’, IETF RFC 2475, informational, December 1998.

**RFC2481**, K. Ramakrishnan and S. Floyd, ‘A Proposal to add Explicit Congestion Notification (ECN) to IP’, IETF RFC 2481, experimental, January 1999.

**RFC2488**, M. Allman, D. Glover and L. Sanchez, ‘Enhancing TCP Over Satellite Channels using Standard Mechanisms’, IETF RFC 2488, best current practice (BCP), January 1999.

**RFC2581**, M. Allman, V. Paxson and W. Stevens, ‘TCP Congestion Control’, IETF RFC 2581, proposed standard, April 1999.

**RFC2582**, S. Floyd and T. R. Henderson, ‘The NewReno Modification to TCP’s Fast Recovery Algorithm’, IETF RFC 2582, experimental, April 1999.

**RFC2597**, J. Heinanen, F. Baker, W. Weiss and J. Wroclawski, ‘Assured Forwarding PHB Group’, IETF RFC 2597, proposed standard, June 1999.

**RFC2598**, V. Jacobson, K. Nichols and K. Poduri, ‘An Expedited Forwarding PHB’, IETF RFC 2598, proposed standard, June 1999.

**RFC2663**, P. Srisuresh and M. Holdrege, ‘IP Network Address Translator (NAT) Terminology and Considerations’, IETF RFC 2663, informational, August 1999.

**RFC2694**, P. Srisuresh, G. Tsirtsis, P. Akkiraju and A. Heffernan, ‘DNS extensions to Network Address Translators (DNS_ALG)’, IETF RFC 2694, informational, September 1999.

**RFC2702**, D. Awduche, J. Malcolm, J. Agogbua, M. O’Dell and J. McManus, ‘Requirements for Traffic Engineering Over MPLS’, IETF RFC 2702, informational, September 1999.

**RFC2709**, P. Srisuresh, ‘Security Model with Tunnel-mode IPsec for NAT Domains’, IETF RFC 2709, informational, October 1999.

**RFC2858**, T. Bates, Y. Rekhter, R. Chandra and D. Katz, ‘Multiprotocol Extensions for BGP-4’, IETF RFC 2858, standards track, June 2000.

**RFC2883**, S. Floyd, J. Mahdavi, M. Mathis, M. Podolsky and A. Romanow, ‘An Extension to the Selective Acknowledgement (SACK) Option for TCP’, IETF RFC 2883, standards track, July 2000.

**RFC2909**, P. Radoslavov, D. Estrin, R. Govindan, M. Handley, S. Kumar and D. Thaler, ‘The Multicast Address-Set Claim (MASC) Protocol’, IETF RFC 2909, experimental, September 2000.

**RFC2923**, K. Lahey, ‘TCP Problems with Path MTU Discovery’, IETF RFC 2923, informational, September 2000.

**RFC3027**, M. Holdrege and P. Srisuresh, ‘Protocol complications with the IP Network Address Translator (NAT)’, IETF RFC 3027, informational, January 2001.

**RFC3031**, E. C. Rosen, A. Viswanathan and R. Callon, ‘Multiprotocol Label Switching Architecture’, standards track, January 2001.

**RFC3036**, L. Andersson et al., ‘LDP Specification’, IETF RFC 3036, standards track, January 2001.

**Rider85**, L. Rider, ‘Optimised polar orbit constellations for redundant earth coverage’, *Journal of the Astronautical Sciences*, vol. 33, pp. 147-161, April-June 1985.

**SahMuk00**, L. H. Sahasrabuddhe and B. Mukherjee, ‘Multicast routing algorithms and protocols: a tutorial’, IEEE Network, vol. 14 no. 1, pp. 90-102, January/February 2000.

**Saltzeretal84**, J. Saltzer, D. Reed and D. Clark, ‘End-to-End Arguments in System Design’, *ACM Transactions in Computer Systems*, pp. 277-288, November 1984.

**Schacham88**, N. Schacham, ‘Protocols for multi-satellite networks’, Proceedings of MILCOM ’88, pp. 501-505, 1988.

**SCPS99**, Consultative Committee for Space Data Systems, ‘Space Communications Protocol Specification (SCPS) Transport Protocol’, CCSDS-714.0-B-1, Blue Book, May 1999.

**Seo88**, K. Seo, J. Crowcroft, P. Spilling, J. Laws and J. Leddy, ‘Distributed Testing and Measurement across the Atlantic Packet Satellite Network (SATNET)’, Proceedings of SIGCOMM ’88, Palo Alto, August 1998.

**Shields96**, C. Shields, ‘Ordered Core Based Trees’, Masters thesis, University of California at Santa Cruz, June 1996.

**ShieldsGarcia97**, C. Shields and J. J. Garcia-Luna-Aceves, ‘The Ordered Core-based Tree Protocol’, Proceedings of IEEE INFOCOM ’97, Kobe, Japan, 7-11 April, 1997.

**SisaSchulz98**, D. Sisalem, H. Schulzrinne, ‘The loss-delay adjustment algorithm: a TCP-friendly adaptation scheme’, proceedings of Network and Operating System Support for Digital Audio and Video (NOSSDAV), Cambridge, UK, July 8-10, 1998.

**Stevens94**, W. R. Stevens, ‘TCP/IP Illustrated, Volume 1: The Protocols’, Addison-Wesley, 1994.

**Stuart96**, J. R. Stuart, ‘The wireless communications and small satellite revolutions: next generation communications concepts’, Space Horizons Summit, Boston, 4 May 1996.

**Sturza95**, M. A. Sturza, ‘Architecture of the Teledesic satellite system’, Proceedings of the International Mobile Satellite Conference, pp. 212-218, 1995.

**Thaleretaldraft00**, D. Thaler et al, ‘Border Gateway Multicast Protocol (BGMP): Protocol Specification’, work in progress as internet draft, BGMP working group, November 2000.

**ToddHahne97**, T. D. Todd and E. L. Hahne, ‘Multiaccess mesh (multimesh) networks’, *IEEE/ACM Transactions on Networking*, vol. 5 no. 2, pp. 181-189, April 1997.

**Tucketal94**, E. F. Tuck, D. P. Patterson, J. R. Stuart and M. H. Lawrence, ‘The Calling Network: a global wireless communications system’, *International Journal of Satellite Communications*, vol. 12 no. 1, pp. 45-61, January/February 1994.

**Uzunaliogluetal97**, H. Uzunalioglu, W. Yen and I. Akyildiz, ‘A Connection Handover Protocol for LEO Satellite ATM Networks’, Proceedings of ACM Mobicom ’97, pp. 204-214, October 1997.

**Varadhanetal98**, K. Varadhan, D. Estrin and S. Floyd, ‘Impact of Network Dynamics on End-to-End Protocols: Case Studies in TCP and Reliable Multicast’, Technical Report 98-672, Department of Computer Science, University of Southern California, March 1998. Edited version published as ‘Impact of network dynamics on end-to-end protocols: Case studies in reliable multicast’, in International Symposium on Computers and Communications, 1998.

**Vargo60**, L. G. Vargo, ‘Orbital patterns for satellite systems’, American Astronautical Society preprint 60-48, January 1960.

**ViciCrowRizzo98**, L. Vicisano, J. Crowcroft and L. Rizzo, ‘TCP-like congestion control for layered multicast data transfer’, Proceedings of IEEE INFOCOM ’98, San Francisco, March-April 1998.

**Walker71**, J. G. Walker, ‘Some circular orbit patterns providing continuous whole earth coverage’, *Journal of the British Interplanetary Society*, vol. 24, pp. 369-384, 1971.

**Walker82**, J. G. Walker, ‘Comments on “Rosette constellations of earth satellites”’, *IEEE Transactions on Aerospace and Electronic Systems*, vol. 18 no. 4, pp. 723-724, November 1982.

**Walker84**, J. G. Walker, ‘Satellite constellations’, *Journal of the British Interplanetary Society*, vol. 37, pp. 559-571, 1984.

**Wang95**, C.-J. Wang, ‘Delivery time analysis of a low earth orbit satellite network for seamless PCS’, IEEE Journal on Selected Areas in Communications, vol. 13 no. 2, pp. 389-396, February 1995.

**Werner97**, M. Werner, ‘A dynamic routing concept for ATM-based satellite personal communication Networks’, *IEEE Journal on Selected Areas in Communications*, vol. 15 no. 8, pp. 1636-1648, October 1997.

**WernerBischLutz95**, M. Werner, H. Bischl and E. Lutz, ‘Mobile User Environment and Satellite Diversity for NGSO S-PCNs’, Proceedings of the International Mobile Satellite Conference (IMSC’95), pp. 476-481, Ottawa, Canada, 6-8 June 1995.

**WernerDelBurch97**, M. Werner, C. Delucchi and K. Burchard, ‘ATM Networking for Future ISL-based LEO satellite constellations’, Proceedings of the International Mobile Satellite Conference 1997 (IMSC ’97), pp. 295-300, Pasadena, California, 16-18 June 1997.

**WernerDelVogetal97**, M. Werner, C. Delucchi, H.-J. Vögel, G. Maral and J.-J. De Ridder, ‘ATM-based routing in LEO/MEO satellite networks with intersatellite links’, *IEEE Journal on Selected Areas in Communications*, vol. 15 no. 1, pp. 69-82, January 1997.

**WernerJahnLutzBott95**, M. Werner, A. Jahn, E. Lutz and A. Böttcher, ‘Analysis of System Parameters for LEO/ICO-Satellite Communication Networks’, *IEEE Journal on Selected Areas in Communications*, vol. 13 no. 2, pp. 371-381, February 1995.

**WernerKroMar97**, M. Werner, O. Kroner and G. Maral, ‘Analysis of intersatellite links load in a near-polar LEO satellite constellation’, Proceedings of the International Mobile Satellite Conference 1997 (IMSC ’97), pp. 289-294, Pasadena, California, June 1997.

**WernerMaral97**, M. Werner and G. Maral, ‘Traffic flows and dynamic routing in LEO intersatellite link networks’, Proceedings of the International Mobile Satellite Conference 1997 (IMSC ‘97), pp. 283-288, Pasadena, California, 16-18 June 1997.

**WernerWauFringMaral99**, M. Werner, J. Frings, F. Wauquiez and G. Maral, ‘Capacity Dimensioning of ISL Networks in Broadband LEO Satellite Systems’, Proceedings of the Sixth International Mobile Satellite Conference (IMSC ’99), pp. 334-341, Ottawa, Canada, 16-18 June, 1999.

**WiedeViterbi93**, R. A. Wiedeman and A. J. Viterbi, ‘The Globalstar mobile satellite system for worldwide personal communications’, Proceedings of the International Mobile Satellite Conference 1993 (IMSC ‘93), pp. 291-296, Pasadena, California, 16-18 June 1993.

**Wisløff96**, T. E. Wisløff, ‘Dual satellite path diversity in non-geostationary satellite systems’, Dr. Ing. Thesis 1996:84, Department of Telematics, Norwegian University of Science and Technology.

**WongKang90**, J. S. K. Wong and Y. Kang, ‘Distributed and fail-safe routing algorithms in toroidal-based metropolitan area networks’, *Computer Networks and ISDN Systems*, vol. 18, pp. 379-391, 1990.

**Wood95**, L. Wood, ‘Network performance of non-geostationary constellations equipped with intersatellite links’, MSc thesis for University of Surrey, Rapport 95-9, ENST Toulouse, November 1995.

**Woodetal98**, L. Wood, H. Cruickshank and Z. Sun, ‘Supporting group applications via satellite constellations with multicast’, Proceedings of the Sixth IEE Conference on Telecommunications (ICT ’98), pp. 190-194, IEE Conference Publication no. 451, Edinburgh, UK, March/April 1998.

**Woodetal01a**, L. Wood, A. Clerget, I. Andrikopoulos, G. Pavlou and W. Dabbous, ‘IP routing issues in satellite constellation networks’, *International Journal of Satellite Communications*, special issue on IP, vol. 19 no. 1, pp. 69-92, January/February 2001.

**Woodetal01b**, L. Wood, G. Pavlou and B. G. Evans, ‘Effects on TCP of routing strategies in satellite constellations’, *IEEE Communications Magazine*, special issue on Satellite-Based Internet Technology and Services, vol. 39 no. 3, pp 172-181, March 2001.

**Woodetal01c**, L. Wood, G. Pavlou and B. G. Evans, ‘Managing diversity with handover to provide classes of service in satellite constellation networks’, Proceedings of the 19th AIAA International Communications Satellite Systems Conference (ICSSC), Toulouse, France, April 2001.

**WorfolkThurman97**, P. A. Worfolk and R. E. Thurman, '*SaVi* - software for the visualization and analysis of satellite constellations', developed at The Geometry Center, University of Minnesota. Available from: http://www.geom.umn.edu/locate/SaVi

**YihChan93**, G. J. Yih and K. M. Chandler, 'Communications system employing spectrum reuse on a spherical surface', US patent 5,268,694, issued 7 December 1993.

**Zhangetal97**, Y. Zhang, D. De Lucia, B. Ryu and S. K. Dao, 'Satellite Communications in the Global Internet: Issues, Pitfalls, and Potential', proceedings of The Seventh Annual Conference of the Internet Society (ISOC), Kuala Lumpur, 24-27 June 1997.

# *Appendix 1. Publications authored*

Publications produced during the period that work for this PhD thesis was carried out:

## A1.1 Peer-reviewed journal papers

**L. Wood**, G. Pavlou and B. G. Evans, 'Effects on TCP of routing strategies in satellite constellations', *IEEE Communications Magazine*, special issue on Satellite-Based Internet Technology and Services, vol. 39 no. 3, pp. 172-181, March 2001.

**L. Wood**, A. Clerget, I. Andrikopoulos, G. Pavlou and W. Dabbous, 'IP routing issues in satellite constellation networks', *International Journal of Satellite Communications*, special issue on IP, vol. 19 no. 1, pp. 69-92, January/February 2001.

## A1.2 Conference papers

**L. Wood**, G. Pavlou and B. G. Evans, 'Managing diversity with handover to provide classes of service in satellite constellation networks', Proceedings of the 19th AIAA International Communications Satellite Systems Conference (ICSSC), Toulouse, France, April 2001.

I. Andrikopoulos, **L. Wood** and G. Pavlou, 'A fair traffic conditioner for the assured service in a differentiated services Internet', Proceedings of the IEEE International Conference on Communications (ICC '00), New Orleans, USA, June 2000.

**L. Wood**, H. Cruickshank and Z. Sun, 'Supporting group applications via satellite constellations with multicast', Proceedings of the Sixth IEE Conference on Telecommunications (ICT '98), pp. 190-194, IEE Conference Publication no. 451, Edinburgh, UK, March/April 1998.

## A1.3 Internet drafts

G. Fairhurst and **L. Wood**, 'Link ARQ issues for IP traffic', IETF internet draft accepted as a PILC WG working item, November 2000, revised March 2001.

## A1.4 COST contributions

T. Örs, A. Sammut, **L. Wood** and B. G. Evans, ‘An overview of future satellite communication options for LAN interconnection’, Third COST 253 Management Committee meeting, Brussels, Belgium, March 1998.

**L. Wood**, ‘Multicast in satellite constellations’, Third COST 253 Management Committee meeting, Brussels, Belgium, March 1998.

Contributions to the COST 253 Final Report.

# *Appendix 2. Software authored*

Software that is related to this PhD work and that has been made publicly available:

## A2.1 for the satellite visualisation tool *SaVi*

Scripts generating simulations of a variety of satellite constellations, including:

*Clarke* geostationary, *Molnya* and *Tundra* high-latitude elliptical, *Celestri*, *Globalstar GS-2*, *LEqO*, *Macrocell*, *SkyBridge*, *Orbcomm*, *@contact*, *ICO*, *Odyssey*, *Orblink*, *Spaceway NGSO*, *Teledesic*, and the Global Positioning System (*GPS*).

Several of these scripts (for *LEqO*, *SkyBridge* and *GPS*) shipped with *SaVi* release 1.0.

Scripts are available from:

http://www.ee.surrey.ac.uk/Personal/L.Wood/software/SaVi/

Results from this work are presented in Chapters 1 and 6.

## A2.2 for the network simulator *ns*

A range of enhancements to the *ns* satellite simulation extensions, including scripts drawing satellite constellation networks. Those scripts are available from:

http://www.ee.surrey.ac.uk/Personal/L.Wood/ns/sat-plot-scripts/

Results from this work are presented in Chapters 2, 4, and 6.

## A2.3 for use of the satellite footprint generator

A user-friendly interface to a modified version of the Centre’s SPOC (Satellite Positioning and Orbital Control) simulator is available for use at:

http://www.ee.surrey.ac.uk/Personal/L.Wood/footprint/

Unprojected maps showing constellation footprints rendered by SPOC are shown in Chapters 1, 2 and 6.

# *Appendix 3. Use of multicast and unicast*

This appendix presents detailed results, summarised in Chapter 4, comparing delays for equivalent multi-way group applications using unicast or core-based multicast to communicate between ground terminals. Range and averages are shown. Delays for core-based multicast are higher, and subject to more variation over time than unicast delays, as shortest-path routes between the core satellite and member satellites change.

## A3.1 Group applications across broadband *Iridium*

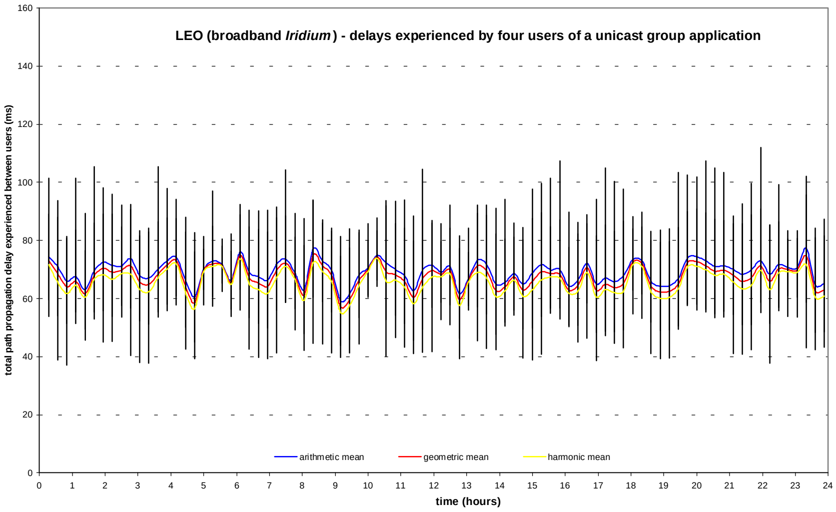


**Figure A3.1 - unicast group application for four users over broadband *Iridium***

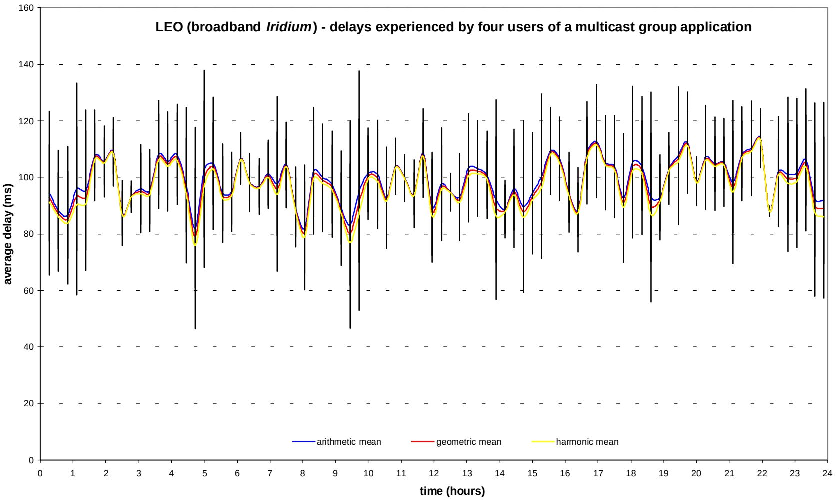


**Figure A3.2 - multicast group application for four users over broadband *Iridium***

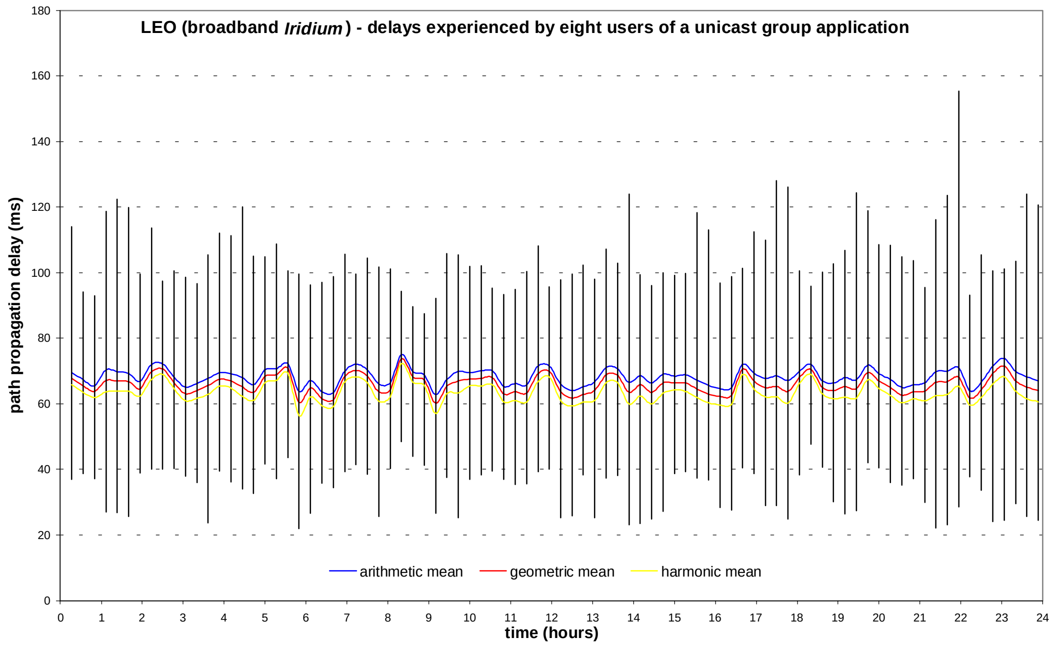


**Figure A3.3 - unicast group application for eight users over broadband *Iridium***

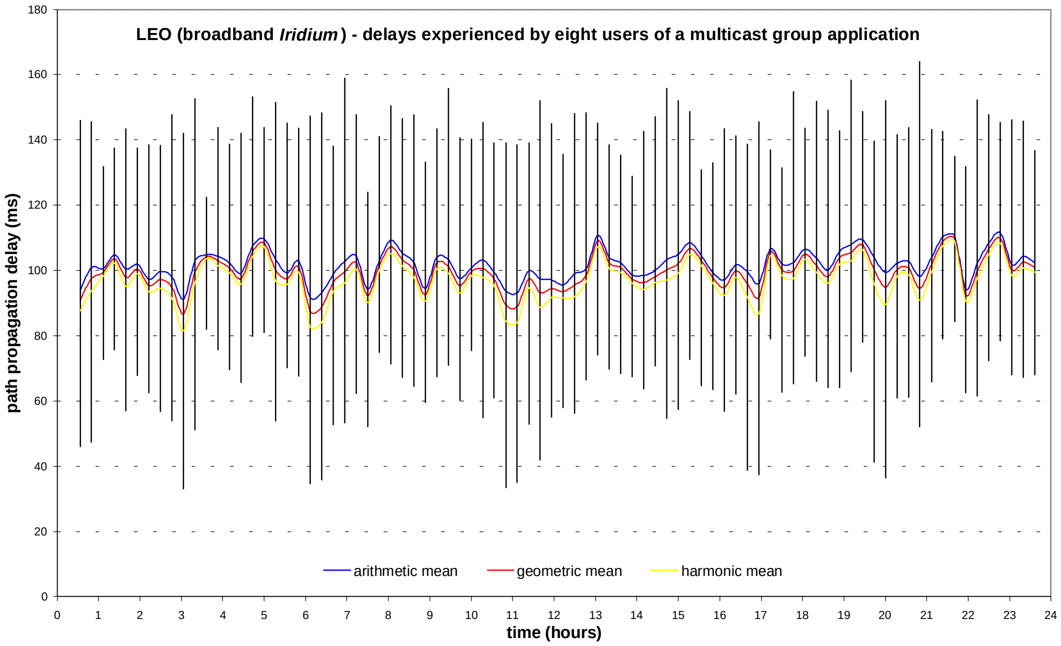


**Figure A3.4 - multicast group application for eight users over broadband *Iridium***

A larger number of users in the group decreases mean delay variation for the group over time, but the delay advantage of unicast shortest-path routing over core-based multicast is still visible. Showing standard deviations for such small sample sizes (4 and 8 users) did not seem worthwhile. Instead, arithmetic, geometric and harmonic means are given, where the distances between the means gives an idea of variation.

## A3.2 Group applications across *Spaceway NGSO*

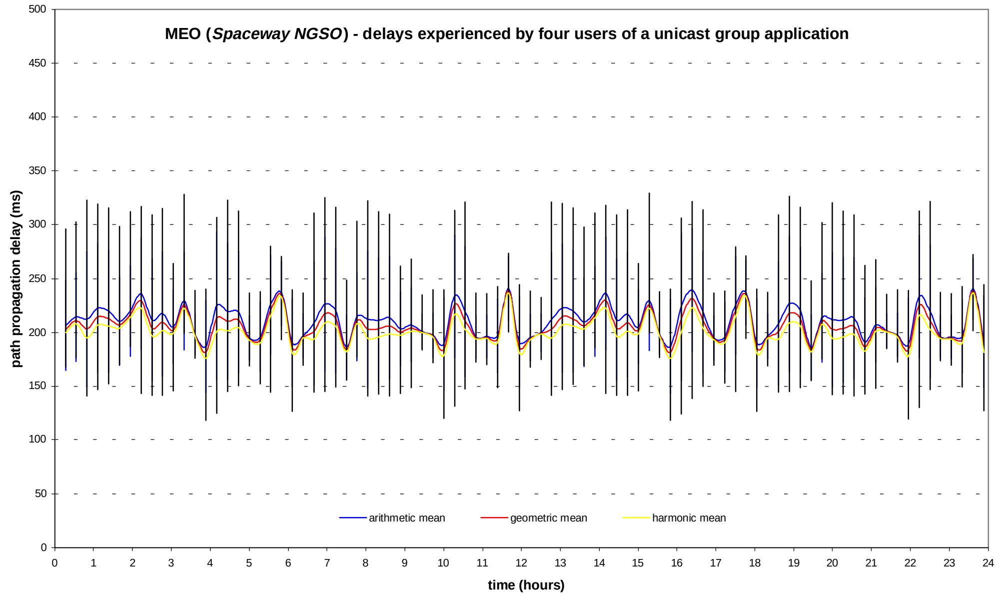


**Figure A3.5 - unicast application for four users over *Spaceway NGSO***

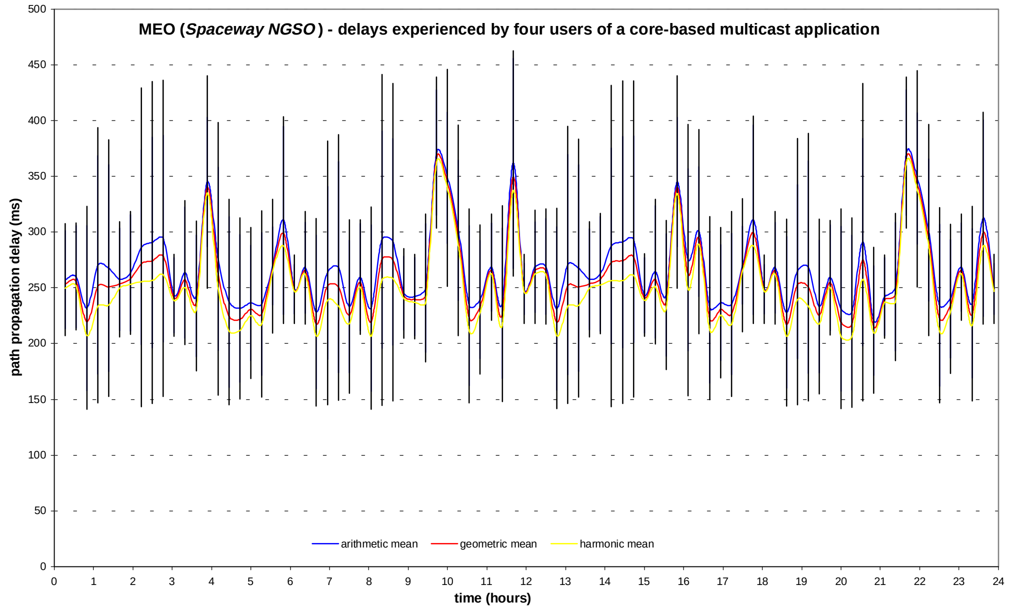


**Figure A3.6 - multicast application for four users over *Spaceway NGSO***

At MEO we see wider variation in delay for the multicast application than for unicast, as well as more visible peaks in delay due to the limited number of satellites and their increased range of distances from the core location. Some of this variation is due to the incomplete double coverage of *Spaceway NGSO*, which is discussed further in Chapter 6; user terminals were not always able to remain on a single surface.

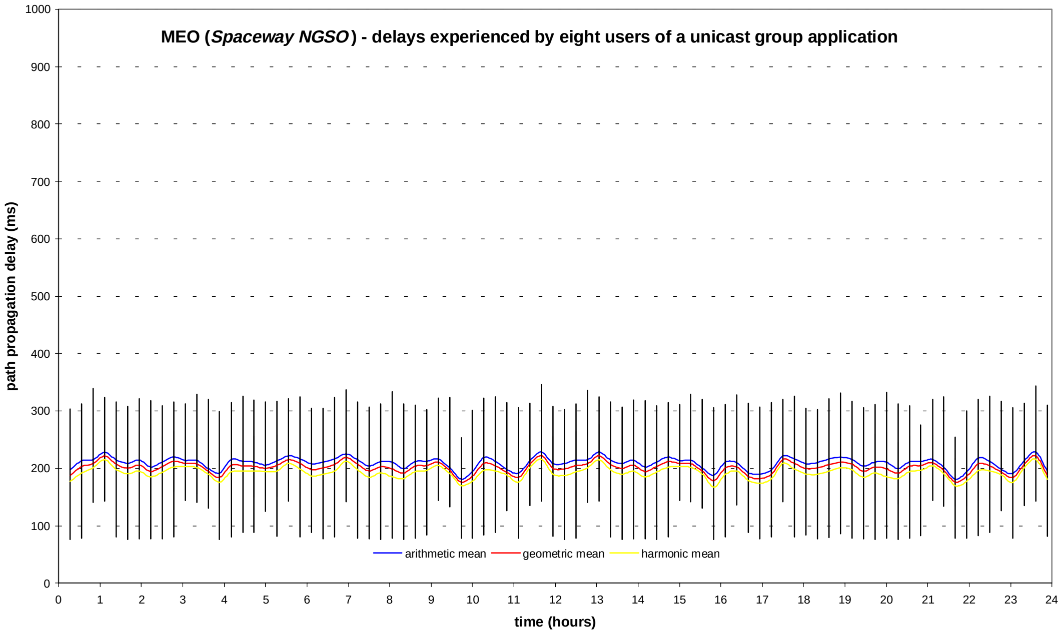


**Figure A3.7 - unicast application for eight users over *Spaceway NGSO***

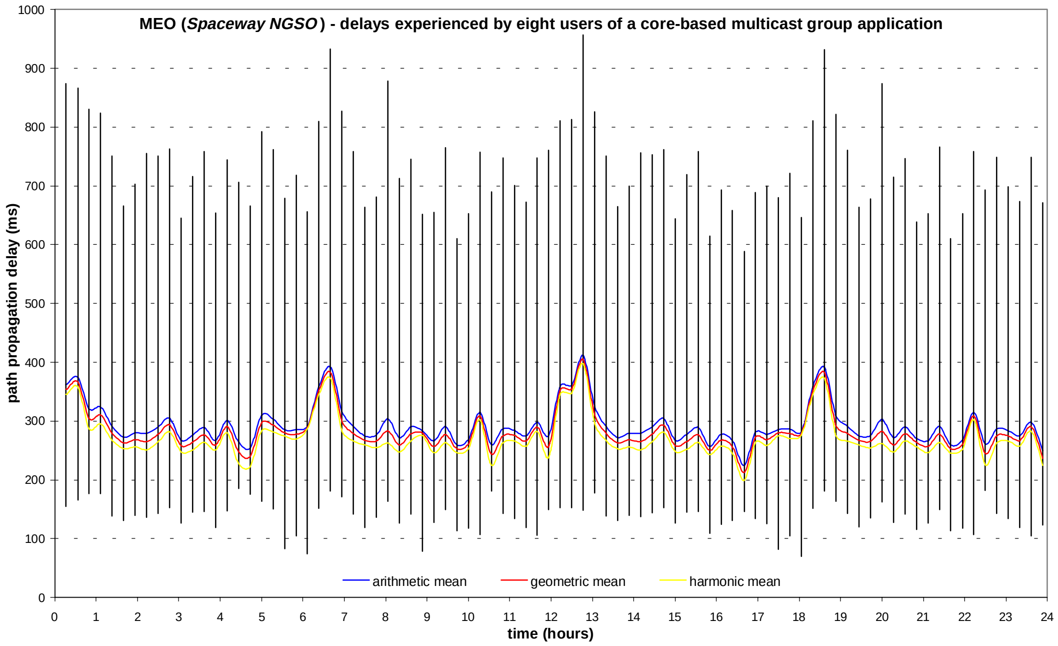


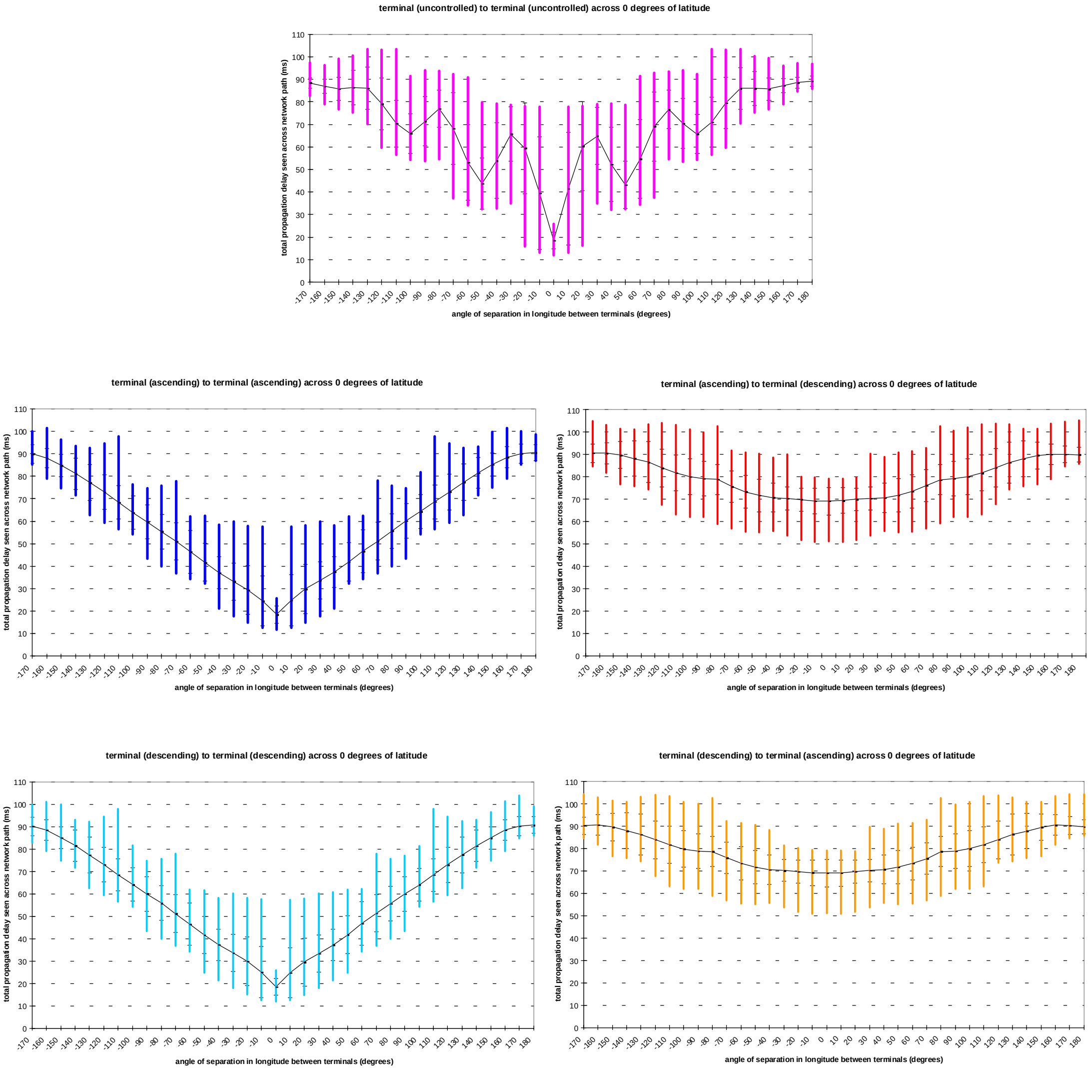
**Figure A3.8 - multicast application for eight users over *Spaceway NGSO***

Use of multicast always results in more delay variation for the group than use of unicast. With the larger number of users scattered over a larger area, the range of delays experienced increases dramatically when multicast is used, while mean delay from the group is increased slightly. These delay increases are due to terminals occasionally being unable to maintain communication with an ascending satellite.

# Appendix 4. Diversity exploration results

This appendix presents detailed results, summarised in Chapter 6, that compare delay curves for managed and unmanaged handover in the modified *Celestri* constellation. These results follow the same methodology as and are similar in form to the results presented for the analysis of the impact of cross-seam links on *Teledesic* in Chapter 2.

## A4.1 Across 0 degrees of latitude


terminal (uncontrolled) to terminal (uncontrolled) across 0 degrees of latitude

terminal (ascending) to terminal (ascending) across 0 degrees of latitude

terminal (ascending) to terminal (descending) across 0 degrees of latitude

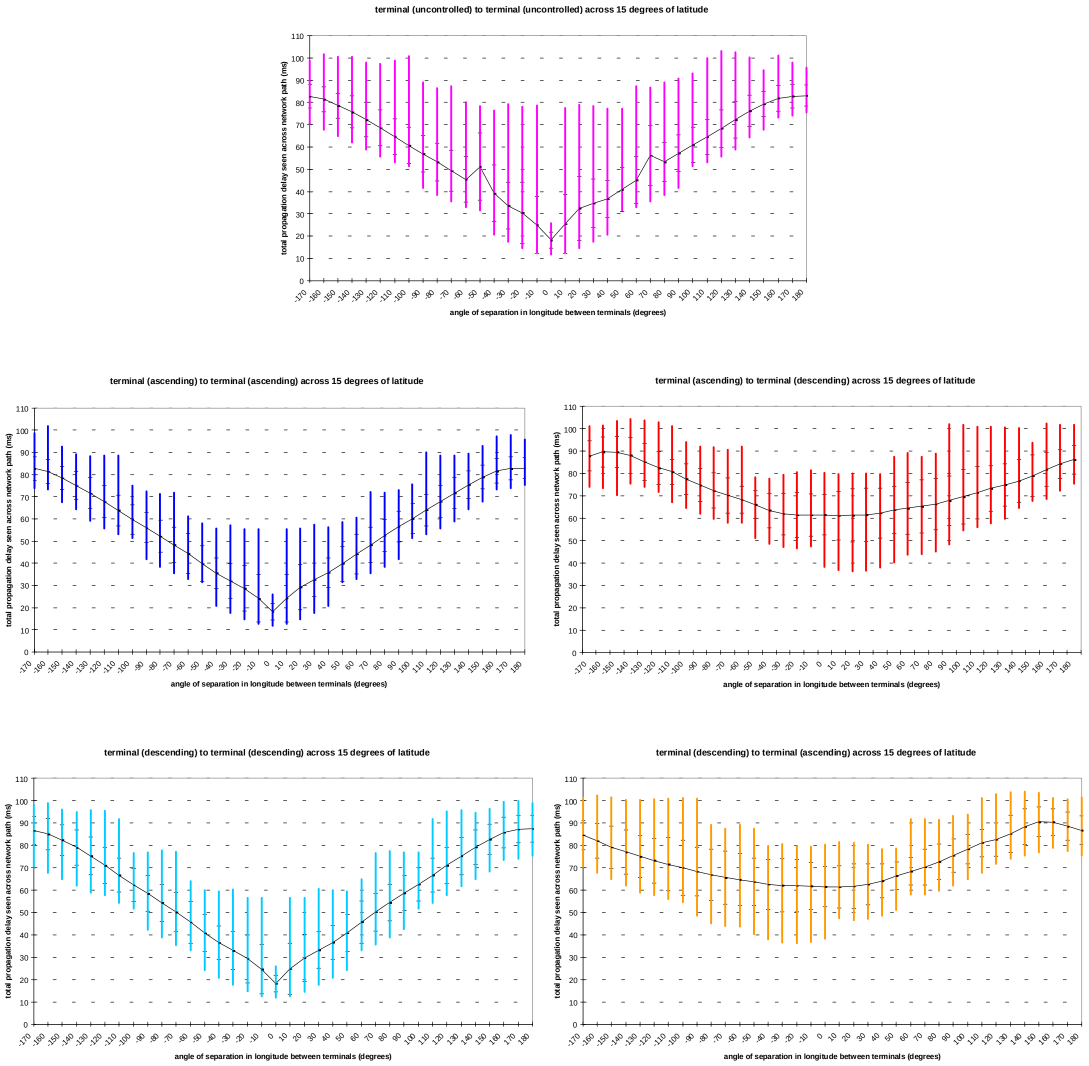
terminal (descending) to terminal (descending) across 0 degrees of latitude

terminal (descending) to terminal (ascending) across 0 degrees of latitude

**Figure A4.1 - detailed handover simulation results at 0° latitude**

Differences in path delay between the two sets of service classes are clear and large.

## A4.2 Across 15 degrees of latitude


terminal (uncontrolled) to terminal (uncontrolled) across 15 degrees of latitude

terminal (ascending) to terminal (ascending) across 15 degrees of latitude

terminal (ascending) to terminal (descending) across 15 degrees of latitude

terminal (descending) to terminal (descending) across 15 degrees of latitude

terminal (descending) to terminal (ascending) across 15 degrees of latitude

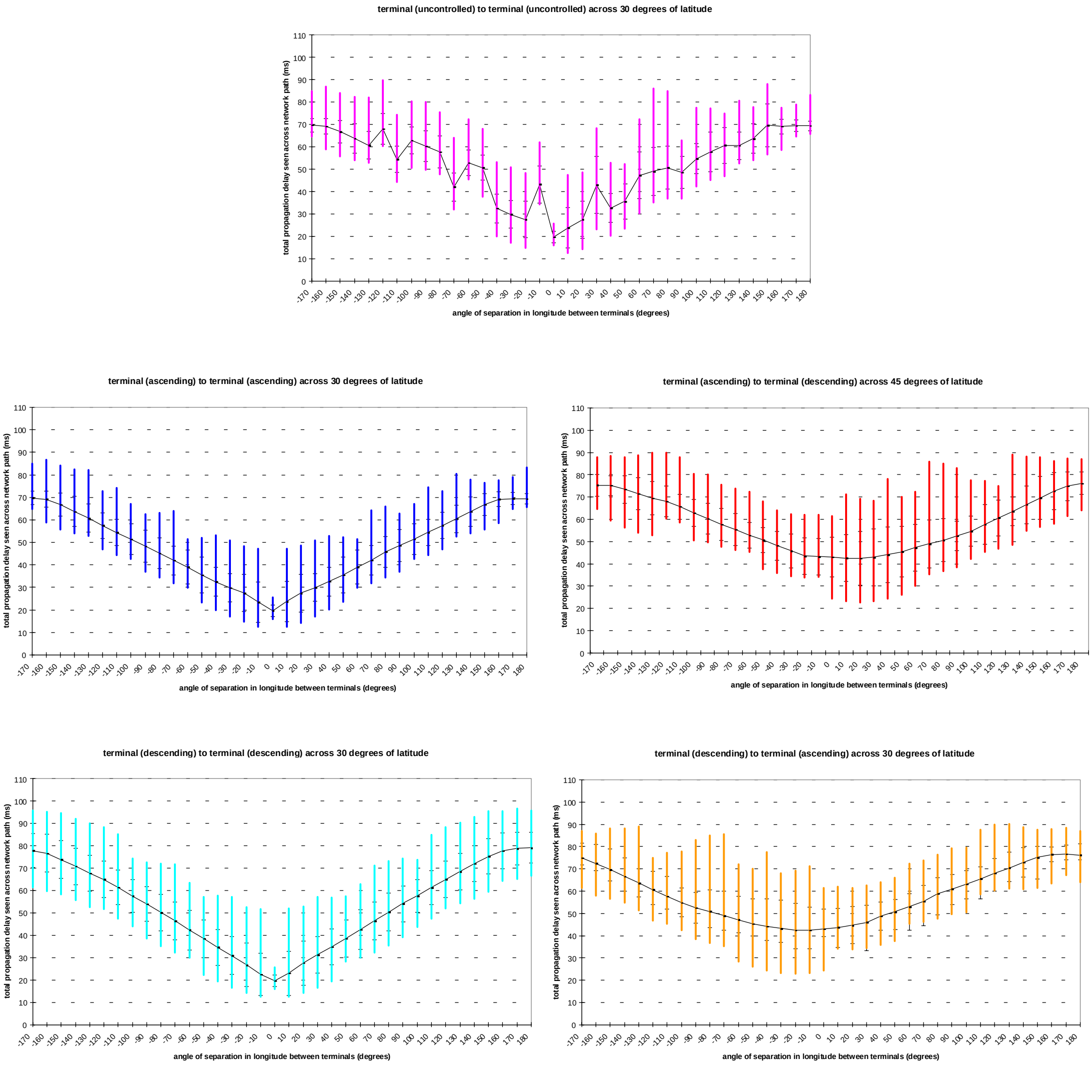
**Figure A4.2 - detailed handover simulation results at 15° latitude**

As the latitude of the terminals increases to 15°, differences between the two service classes remain clearly visible. Some oscillations are visible in the average delays for surface-agnostic handover, due to dependence on initial choice of satellite. Surface-agnostic handover has the largest spread of delays experienced by each simulation.

The ascending-to-descending and descending-to-ascending longitude/delay curves show an interesting reflection.

The difficulty in capturing transient peak delays when the terminals are co-located with zero degrees longitudinal separation, as discussed in Chapter 2, is reflected in some of the graphed results.

## A4.3 Across 30 degrees of latitude


terminal (uncontrolled) to terminal (uncontrolled) across 30 degrees of latitude

terminal (ascending) to terminal (ascending) across 30 degrees of latitude

terminal (ascending) to terminal (descending) across 45 degrees of latitude

terminal (descending) to terminal (descending) across 30 degrees of latitude

terminal (descending) to terminal (ascending) across 30 degrees of latitude

Figure A4.3 - detailed handover simulation results at 30° latitude

Differences between the two service classes, though less, are still clearly marked, and very visible in the variation of the sample uncontrolled surface-agnostic handover, which oscillates between the two sets of curves. Again, there is interesting symmetry and reflection in the range of delays experienced the second class of longer delays.

## A4.4 Across 45 degrees of latitude

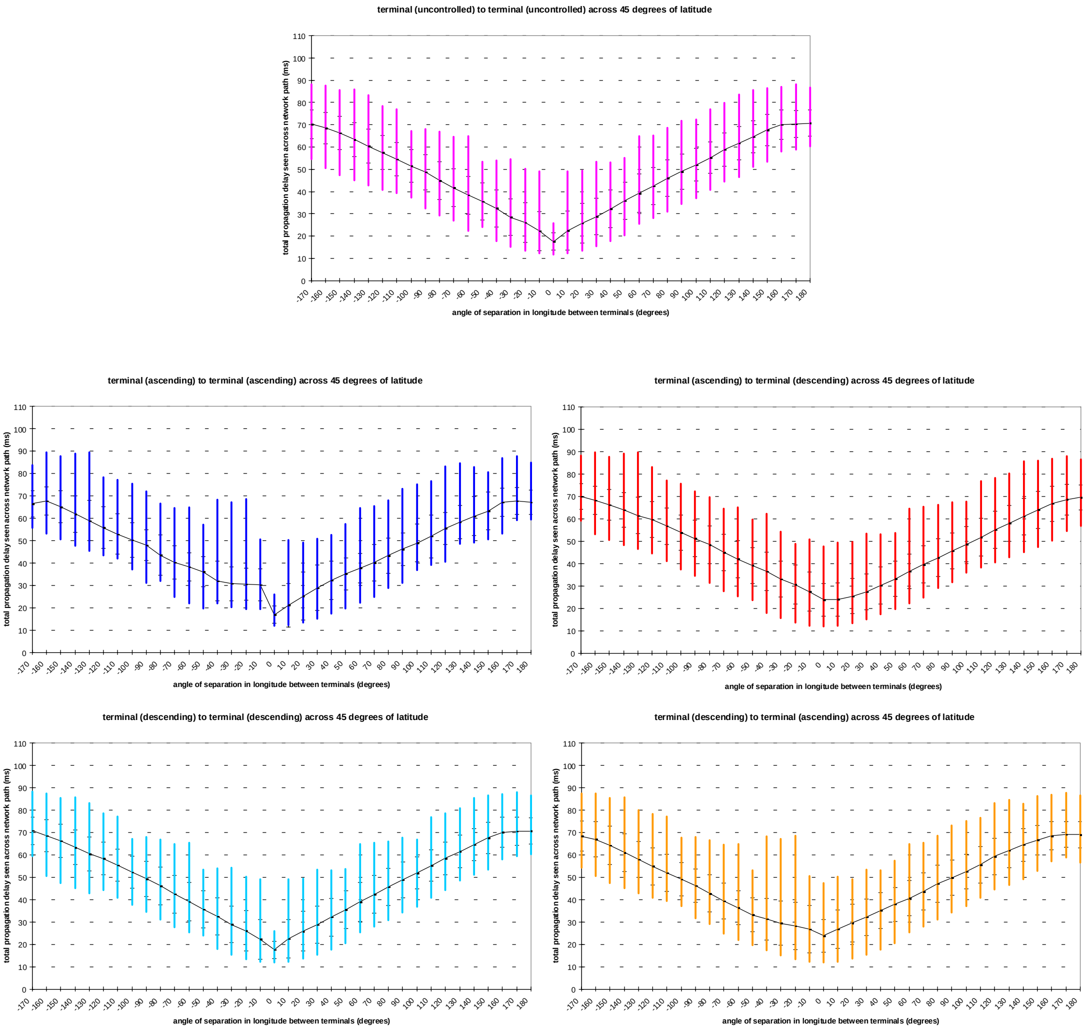


**Figure A4.4 - detailed handover simulation results at 45° latitude**

As the latitude increases to 45°, the differences between the path delays resulting from the different handover classes are less marked; there is very little difference in average delays, and ranges of delays repeat across classes. This small difference in average delay is reflected in the uncontrolled surface-agnostic handover, whose delay variation is now near that of the other classes.

## A4.5 Across 60 degrees of latitude

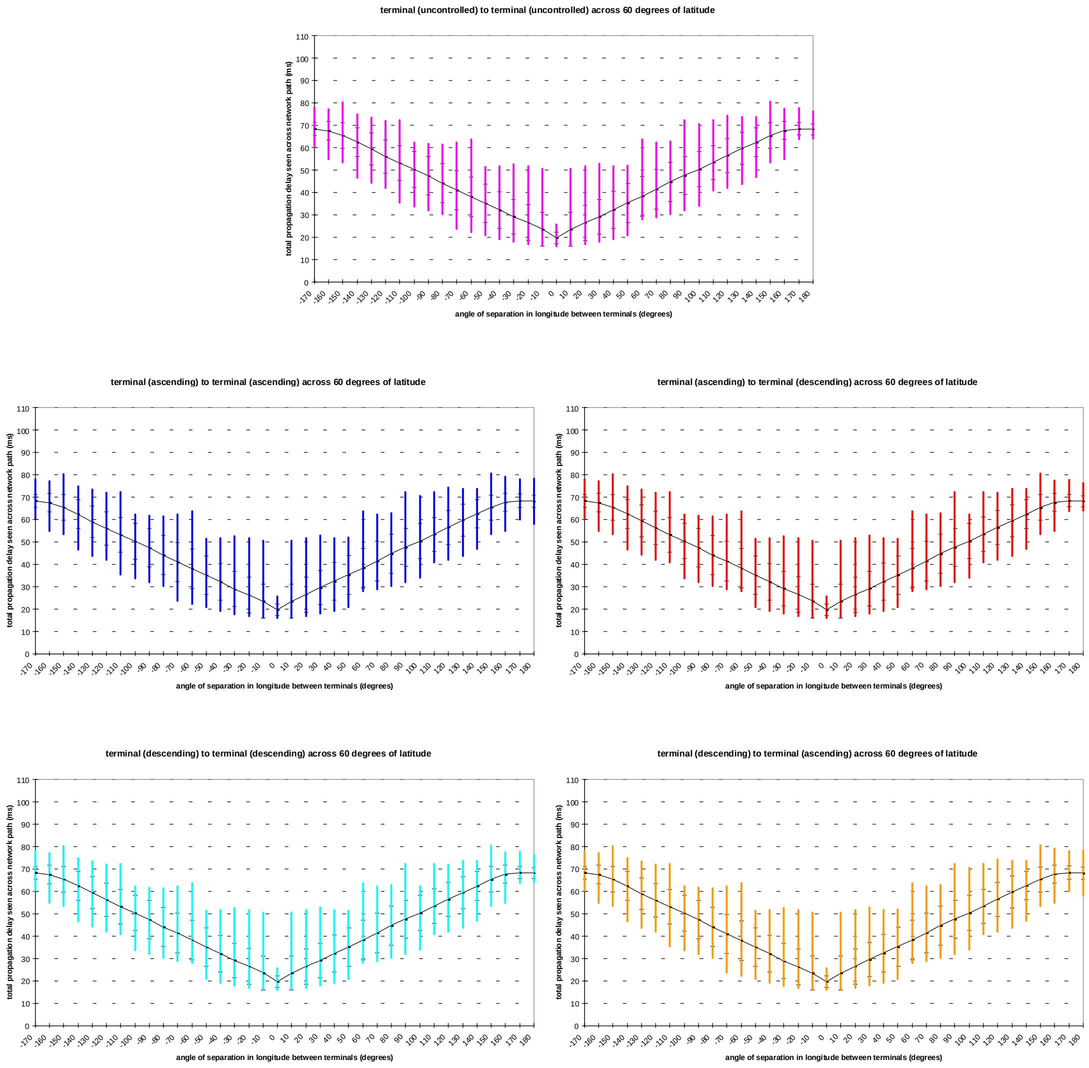


**Figure A4.5 - detailed handover simulation results at 60° latitude**

At the highest latitudes that constitute the limits of coverage these is virtually nothing to differentiate between the service classes. Handover choices are dictated by available coverage.

# *Appendix 5. Summary details outlining satellite constellation proposals*

Data describing the satellite constellations used for network simulation is presented here. Much of the orbital information presented here is used and encoded in the *SaVi* simulation scripts discussed in Appendix 2. The minor degree of orbital eccentricity present in reality is neglected in the *ns* simulations; orbits are assumed to be perfectly circular. All orbits are prograde. Satellites are evenly spaced around an orbital plane.

In the *Teledesic* and broadband *Iridium* simulations, interplane and cross-plane satellite links are assumed not to be functional above 70° of latitude, due to the rapid approach of satellites and high Doppler shift and tracking requirements. The phase offset between neighbouring *Celestri* planes uses the fifth harmonic factor [**Ballard80**]. Modifications to these parameters, when made, are described in the main text.

| **Constellation** | ***Spaceway NGSO*** | ***Teledesic*** (Boeing design) | ***Iridium*** | ***Celestri*** |
|---|---|---|---|---|
| Orbital altitude (km) | 10352 | 1375 | 780 | 1400 |
| Inclination of orbital plane to the equator (°) | 55 | 84.7 | 86.4 | 48 |
| No. of planes | 4 | 12 | 6 | 7 |
| No. of satellites per plane | 5 | 24 | 11 | 9 |
| Spacing between planes (°) | 360/4 = 90 | 15 | 31.7 | 360/7 = 51.43 |
| Phase offset between satellites in neighbouring planes (°) | none | 360/24/2<br>=7.5$^+$ | 360/11/2<br>= 32.73 | 360/9/7*5<br>= 28.57 |
| Minimum elevation angle of satellite from ground terminal for link (°) | 30 | 40 | 8.2 | 16 |
| No. of ISLs per satellite | 4 | 8 (see fig. 6.1) | 4 | 6 |
| Cross-seam links? | no (no seam) | yes | added for simulation | no (no seam) |

**Figure A5.1 - parameters describing constellation network simulations**

[+] The design of the *Teledesic* constellation was overdimensioned, with large coverage overlap between planes to permit random phasing between neighbouring planes. This was judged necessary due to the difficulty of injecting hundreds of satellites into different orbits at exact intervals and the problems of differential precession between planes and maintaining the constellation geometry over time. Simulation does not have to consider these factors, so phasing between planes was set to its optimum offset of 7.5°, or half the spacing between neighbouring satellites in the same orbital plane, in order to ensure predictable constellation behaviour across multiple simulation runs.