Cost Optimization and Routing for Satellite Network Constellations

by

Kenneth C. H. Kwok

B.S., Electrical Engineering, Cornell University (1999)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2001

Author ....
Department of Electrical Engineering and Computer Science
September 27, 2000

Certified by ....
Vincent W. S. Chan
Joan and Irwin M. Jacob Professor of Electrical Engineering and Computer Science, and Aeronautics and Astronautics
Director, Laboratory for Information and Decision Systems
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Cost Optimization and Routing for Satellite Network Constellations

by

Kenneth C. H. Kwok

Submitted to the Department of Electrical Engineering and Computer Science
on September 27, 2000, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

Low-earth orbit (LEO) satellite communications systems have been under rapid development in the past few years as it is predicted that they will become part of the Next Generation Internet (NGI), a global heterogeneous network that provides ubiquitous access to every part of the world. Nevertheless, very little research has been done on the cost aspect of a satellite network. In this thesis, uplink and downlink costs are ignored and a cost equation based solely on crosslinks is developed and studied closely together with a seamless constellation model. Using this cost equation, cost optimization is performed in LEO and GEO satellite systems to find the optimum constellation size with respect to the amount of uniform traffic present. Modifications of the constellations, such as the 3-crosslink-per-node mesh network, and the 1-inter-plane-crosslink mesh network, are introduced in an attempt to further reduce the cost of the system. Interaction of hotspot traffic with uniform traffic in a square mesh is also studied. We are able to find a lower bound and an upper bound of the minimum required crosslink capacity, given a stream of uniform traffic and multiple streams of hot spot traffic. We also find the properties of hot spot traffic in an infinite grid and extend the result to a fixed size grid. Finally, the notion of incorporating the satellite network into the global heterogeneous network is explored. The relationship between the satellite network and the terrestrial network is studied. In particular, the assignment of cost metrics to inter-satellite links, uplinks and downlinks, and terrestrial links is investigated. At the end a basic simulation of the traffic in a heterogeneous network is developed in MATLAB, which can be used to study the transient properties of the network.

Thesis Supervisor: Vincent W. S. Chan
Title: Joan and Irwin M. Jacob Professor of Electrical Engineering and Computer Science, and Aeronautics and Astronautics
Director, Laboratory for Information and Decision Systems

Acknowledgments

I would like to express my gratitude, first and foremost, to my advisor, Professor Vincent W.S. Chan, for continuously giving me insights on the topic. His guidance has definitely refined my approach to research and problem solving. I am also grateful to DARPA and MIT which have provided me with financial support. Without them, this piece of work would not be made possible.

Also, I would like to thank Isabella, Junko, Tommy, Xuxia, Yiu Tak, her good friend Harry, and others for making the summer of 2000 a particularly enjoyable one. Last but not least, I would like to thank my parents for their unrelenting support throughout the years.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Inspired by the successful launch of the V-2 rockets in 1942, well known writer Arthur C. Clarke, in his visionary article in *Wireless World* dated October 1945 called "Extra terrestrial relays", inspired a whole new field of communications by exploring the possibilities of using geostationary satellites for global communications. From this time onward, satellites have been used for television broadcasting, military communications, telephony trunking, data relay and mobile communications. In the past few years, research and development efforts in the low-earth orbit (LEO) satellites have increased as there are a need and a possibility of a ubiquitous global communication network. It is foreseeable that satellites can be an important part of the Next Generation Internet (NGI), due to the following reasons [2]:

- Satellite services can be provided over wide geographical areas including urban, rural, remote, and inaccessible areas. It should be noted that two-thirds of the world still does not have Internet infrastructure.

- Satellites can act as a safety valve for other parts of the NGI. Fiber failure, or network congestion problems, can be recovered easily by routing traffic through a satellite channel.

- Satellite system can be deployed faster than most terrestrial infrastructures into regions with little or no existing access.

Recent LEO satellite development can be highlighted by the development of several LEO satellite systems by different commercial parties, such as Globalstar, Teledesic and Iridium. Each system has a different architecture. For example, Globalstar uses a bent-pipe architecture with no inter-satellite links (ISLs). On the other hand, Teledesic and Iridium use ISLs but they have very different constellations. Iridium has 66 satellites with a 6×11 constellation and Teledesic has 288 satellites with a 12×24 constellation. One of the reasons for this difference is the different frequency bands used in the two systems, with the Teledesic system using higher frequencies (Ka-band) which requires higher minimum elevation angle for terminals. Higher minimum elevation angle forces the system to have more satellites for global coverage.

One of the goals of this thesis is to find out the reasons, other than the difference in frequency bands used, for this big discrepancy in system architecture among the networks, as all of them have the same basic aim of providing ubiquitous coverage to every part of the world. We will first look at the dependency of the cost of a system on crosslink costs. We will also develop a simple square mesh model for our analysis of the constellation. With the cost equation and the constellation model, we will be able to deduce the optimum constellation based on the amount of traffic in the network and crosslink considerations alone. We will also try to minimize the system cost further by making architectural changes, such as reducing the number of crosslinks per node, to the constellation.

Uniform distribution of traffic from each node is the general case considered in doing the cost optimization. However, this is not the most general case of traffic in any type of networks. A chapter will be devoted to exploring another type of traffic, hotspot traffic. Our main interest in this chapter is to find what the minimum capacity required is when multiple streams of hotspot traffic and a stream of uniform traffic are present in the network.

Finally, as previously suggested, satellites in the future will become part of the global heterogeneous network. The last part of the thesis will explore the interactions of the satellite network with the terrestrial networks. Finally, a basic simulation of the heterogeneous networks, with terrestrial links, up and down links, and inter-satellite links will be developed in MATLAB. It can be used to study the transient properties of the network, such as the effects of different assignments of cost metrics to the amount of traffic in different types of links.

The analyses done in this thesis may be very different from the analyses already done in the field. Nevertheless, this thesis hopefully could provide new insights on the satellite communications systems from a fresh perspective and shed lights on the current conditions of the satellite communications industry.

Chapter 2

The cost equation and the mesh model for constellation optimization

2.1 Introduction

Unlike other types of communications networks such as terrestrial wireless network or wireline network, a satellite system can provide true global coverage to every part of the world at all time; on the other hand, it is also a huge and complex communications system. In order to study this complex network, we will, in this chapter, develop a simplified model of a satellite communication system. One of the goals of this thesis is to study the sensitivity of the cost of the system to different constellation arrangements. We will first derive the cost equation for the crosslink part of the system, and towards the end of the chapter, we will construct a mesh model to represent the satellite constellation and this will be used in our future analysis.

2.2 Previous works

There have not been a lot of previous works done on constellation optimization in satellite communications. Werner in his paper [19] has determined a lower limit for the necessary number of satellites based on the ratio of the total surface area of the Earth to the area of a footprint of a satellite. Furthermore, for a polar orbit constellation, he found the minimum number of orbits (or planes) for global coverage, Ω, and the required number of satellites in each orbit to be:

$$\Omega = \lceil \frac{2\pi}{3\alpha} \rceil, \quad N \approx \lceil \sqrt{3}\Omega \rceil \approx 2\Omega \tag{2.1}$$

where α is the half-sided angle of the footprint, as shown in figure (3-4).

The goal of Werner's analysis is to find the minimum number of satellites required; however, having the minimum required number of satellites does not necessarily result in minimum cost, as there are other factors, such as sizes and distances of crosslinks, uplinks and downlinks, which also play a very important role in determining the overall cost of the system. Therefore, further analyses are needed to be done for the minimization of the overall system cost.

2.3 Derivation of the cost equation

In this section, we try to derive the overall cost of a satellite communication system. Optical crosslinks will represent significant portion of the total cost of a LEO system since crosslinks are a substantial fraction of the overall communications resources. Any cost can be broken down into two components: fixed cost and variable cost [7, p.220]. As a result, our cost equation for the overall system has the structure:

$$\mathcal{C} = N(\$_{fixed} + \$_{var}) \tag{2.2}$$

where $\mathcal{C}$ is the overall cost of the system,

N is the number of crosslinks in the system,

$\$_{fixed}$ is the fixed cost of a crosslink, and

$\$_{var}$ is the variable cost of a crosslink based on capacities and distances of the crosslinks.

In a crosslink, the variable cost roughly depends on the diameter of the telescope

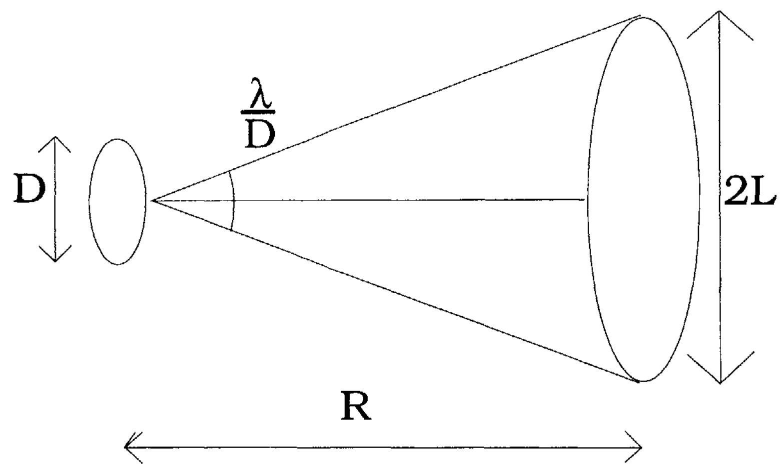

Figure 2-1: Telescope between a transmitter and a receiver

of the transmitter according to this relation:

$$\text{variable cost of the telescope} \propto D^{\alpha}, \quad 2 < \alpha < 3 \tag{2.3}$$

where α is the cost coefficient whose value depends on the technologies and manufacturers used, and D is the diameter of the lens.

D is dependent on the transmission rate, the amount of power transmitted, and the distance between a receiver and a transmitter. In order to find the variable cost of a crosslink using the aforementioned parameters, we consider figure (2-1).

First of all, because of diffraction [9], the transmitted power will propagate in free space at an angle of $\sim \frac{\lambda}{D}$; thus, at the receiver,

$$\text{radius of the receiver field area } r_r = \frac{1}{2}\frac{\lambda}{D}R \tag{2.4}$$

and therefore,

$$\text{receiver field area } A_r = \pi r_r^2 \tag{2.5}$$

$$= \frac{\pi}{4}(\frac{\lambda}{D}R)^2 \tag{2.6}$$

On the other hand, the actual area of the receiver is determined by the radius of the lens of the receiver, $\frac{D}{2}$. Therefore,

$$\text{area of the receiver} A_{r'} = \pi (\frac{D}{2})^2 \tag{2.7}$$

The amount of power received is dependent on the ratio of the two areas, A_r and $A_{r'}$.

$$\begin{aligned} \text{fractional power received} &= \frac{A_{r'}}{A_r} && (2.8) \\ &= \frac{(\frac{D}{2})^2}{\frac{1}{4}(\frac{\lambda}{D}R)^2} && (2.9) \\ &= \frac{D^4}{\lambda^2 \ R^2} && (2.10) \end{aligned}$$

Therefore, if P_T is the power transmitted from the transmitter, using equation (2.10), we have

$$\text{received power at the receiver} = \frac{P_T \ D^4}{\lambda^2 \ R^2} \tag{2.11}$$

Rate is proportional to the amount of power received. The receiver is characterized by a sensitivity parameter, and the rate depends on this parameter. Let β be this receiver sensitivity parameter in [received energy/(bits/s)], then,

$$\text{rate (C)} = \frac{P_T \ D^4}{\beta \ \lambda^2 \ R^2} \tag{2.12}$$

From equation (2.3), we have

$$\$_{var} = K \ D^{\alpha}, \quad \text{where K is a constant specific to the technology chosen} \tag{2.13}$$

From equation (2.12), we have

$$D = (\frac{C\lambda^2 R^2 \beta}{P_T})^{\frac{1}{4}} \tag{2.14}$$

Then, equation (2.3) becomes

$$\$_{var} = K \ (\frac{C\lambda^2 R^2 \beta}{P_T})^{\frac{\alpha}{4}} \tag{2.15}$$

$$= K' \, (C \, R^2)^{\frac{\alpha}{4}} \qquad (2.16)$$

$$= K' \, C^{\frac{\alpha}{4}} \, R^{\frac{\alpha}{2}} \qquad (2.17)$$

with

$$K' = \frac{K \lambda^{\frac{\alpha}{2}} \beta^{\frac{\alpha}{4}}}{P_T^{\frac{\alpha}{4}}}$$

We have found the variable cost. Therefore, the overall cost equation of the system, equation (2.2), can be expressed as:

$$\mathcal{C} = N(k_0 + k_1 C^{\frac{\alpha}{4}} \, d^{\frac{\alpha}{2}}) \qquad (2.18)$$

where $\mathcal{C}$ is the overall cost of the system,

N is the number of crosslinks in the system,

k_0 is the fixed cost of a crosslink,

k_1 is the coefficient associated with the variable cost of a crosslink (this is the same as K' in equation (2.17)),

C is the capacity of the crosslink, and

d is the distance between two crosslinks.

2.4 Mesh model

In order to study a complex network, we first need to make a large number of simplifying assumptions and approximations. The aim of this approach is to reduce complexity and break the entire complex network into simple parts and deal with the parts that are most relevant to our analysis. We can increase complexity gradually once we have formed a basic model. Adding complexity will provide a more accurate representation of the real conditions.

A low earth orbit (LEO) satellite constellation provides ubiquitous communications to everyone at any place of the world at one time. Therefore, satellites are spread out to cover every part of the Earth. Our goal in this section is to map the earth's surface into a two-dimensional plane for simpler analysis. The following simplification is made in our model.

2.4.1 Un-twisting of the planes

LEO Satellites are organized into planes. Satellites in the same plane follow the same ring-shaped orbit which goes from the north pole to the south pole and then back to the north pole. Satellites communicate to neighboring in-plane satellites through intra-plane inter-satellite links (ISLs) and communicate to neighbors that are not in the same plane through inter-plane inter-satellite links. Note that all the planes converge in the polar regions, and diverge as they go away from the polar regions. In forming the two-dimensional mesh, we break the rings formed by the orbital planes and lay them flat, as shown in figure (2-2). In this figure, different shades and shapes are used to denote nodes of different planes. Nodes of the same plane are drawn in the same shape and shade.

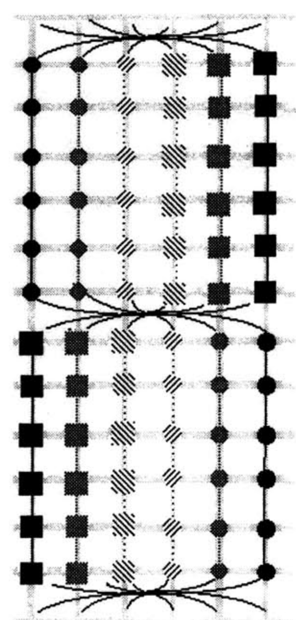

Figure 2-2: Twisted planes

Note that there is a twist of the planes in both of the poles. We further simplify the model by 'un-twisting' the planes (Figure 2-3). The positions of the nodes will change as all the nodes in the same plane are grouped vertically in this un-twisted model. This simplification can be made because we will not analyze the routing schemes of any path with specific origin and destination; rather we will analyze cases of a more general nature, such as uniform traffic and hot spot traffic (they will be defined in future chapters).

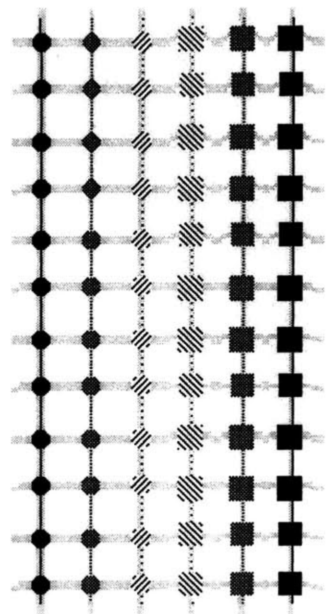

Figure 2-3: Planes un-twisted

2.4.2 Inter-satellite links

We make a general assumption about the number of inter-plane inter-satellite links. We assume that all the links are bi-directional. Therefore, there are 2 inter-plane inter-satellite links for each satellite. As a result, there are 4 inter-satellite links to each node, 2 in-plane and two cross-plane. As a result, our two-dimension model has become a two-dimensional square mesh, with each end connected to the opposite end.

2.4.3 Number of satellites in a plane

Motorola's Iridium has a constellation of 6×11 [6], and Teledesic's system has a constellation of 12×24 [14]. The number of satellites in a plane is therefore approximately twice the number of planes. In our model, we will assume

$$N = 2M \tag{2.19}$$

where N is the number of satellites in a plane and M is the number of planes in the system.

The reason behind $N = 2M$ is that with this type of constellation, the maximum distances between any two in-plane (longitudinal) neighboring satellites and between two cross-plane (latitudinal) neighboring satellites will be the same. Consider a system with M planes, and N satellites in each plane. The planes are separated from each other with the same angular distance of $\frac{2\pi}{2M}$. The satellites in a plane are separated from each other with an angular distance of $\frac{2\pi}{N}$. If we want the maximum distances between two in-plane (longitudinal) neighboring satellites and between two cross-plane (latitudinal) neighboring satellites will be the same, we need the two angles to be equal. The two angles are equal only when $N = 2M$.

2.4.4 Seamless Model

We have also assumed a seamless model. In the real model, at the edges of the network there will be two counter-rotating planes, and this is known as the seam. Implementing inter-satellite links at the seam will be very hard due to the high relative speed of the satellites moving in opposite directions. In most part of the thesis, we will ignore this effect since seamed networks are difficult to analyze.

The Walker delta network [18], a constellation which has received considerable attention, is a seamless model by placing all the ascending nodes equidistantly around the equatorial plane. Delta network is easier to analyze because of its seamless assumption; on the other hand, it will also provide double network coverage since the number of orbits is doubled when compared to a seamed network, and every place will be covered by two satellites [20]. Because of this double coverage, the cost of such a system will be higher as well.

2.4.5 Polar effects

At the polar regions, some of the inter-satellite crosslinks need to be turned off because of the interference caused in the convergence of the planes. Furthermore, because of the changing orientation of the planes and the high relative speed of satellites in polar regions, the tracking of the antennae will be very difficult. However, these polar effects are ignored in our model since they are not relevant to our studies.

Furthermore, at the polar regions, due to the spherical nature of the Earth, the distances between neighboring satellites will be substantially smaller than the distances around the equator. In our mesh model, we will neglect this spherical effect and have all the distances between neighboring satellites to be equal for all cases.

2.4.6 Homogeneous Network

In this thesis, we will consider the network to be a homogeneous network, meaning that inside the network, every satellite node is identical, considers itself to be the center of the network and sees the network in the same way. This model is a very simplistic model; nevertheless, this is a good starting point for mathematical analysis.

The model we have arrived at is very similar to the bi-directional Manhattan Street Network—a modification from the famed Manhattan Street Network (MSN) [3]. It is named for its peculiar topology, resembling the street and traffic organization in downtown Manhattan. Other constellation patterns, such as the Walker delta network [18], and various circle patterns [5, 1], have been proposed; however, they will not not be discussed in this thesis.

2.5 Summary

In this chapter, we have derived a cost equation for crosslinks, which constitute the a large proportion of the total cost of a satellite communications network. In the second part of this chapter, a simplistic mesh model is proposed to model a satellite communications network. We have explored the drawbacks and the advantages of using such a simple model. Even though there are drawbacks, this model will be used as the core model for analysis in future chapters because of its simplicity.

Chapter 3

Constellation Optimization Based on Uniform Traffic

3.1 Introduction

In this chapter, we would like to minimize the total crosslink costs of the system using the cost equation we have derived in the previous chapter. We assume the traffic pattern to be uniform traffic, which is defined as follows:

Uniform Traffic. Uniform traffic is the type of traffic in which every node sends to every node in the grid. T_u is the total amount of uniform traffic. Essentially each node sends $T_u/(NM)^2$ amount of traffic to every node, and receives the same amount of traffic from each node, including itself. A node can be both the sender and the recipient because they can both be located on the same footprint.

In order to do this optimization, we first will find the amount of capacity required for the crosslinks in order to support a stream of uniform traffic T_u. We then try to express the distance between satellites in terms of M, the number of plane in the system. We then substitute the values into the cost equation to find the minimum cost with respect to M. GEO satellites are briefly explored with this optimization model as well. Lastly, numerical examples are given to give us more insight into the planning behind real systems such as Iridium and Teledesic.

3.2 Capacity Required For Crosslinks Under Uniform Traffic

We will consider two types of constellations in this section. One is the homogeneous seamless network model which is less realistic but easier to analyze because of its symmetric pattern; the other one is the seamed network model which is more realistic but more difficult to analyze.

3.2.1 Homogeneous Seamless Network

To find the capacity required, we first try to find the average number of hops in the x and y-direction for sending traffic from all nodes to the center (0,0). To keep the mesh as general as possible, we consider a mesh of size $M \times N$, with M and N being positive odd numbers. For an arbitrary mesh with M and N not equal to each other, we would not expect each crosslink in the mesh to receive equal amount of traffic, even for uniform traffic. Consider an $M \times N$ mesh with N greater than M, there are more nodes in the y-direction; as a result, if the center sends the same amount of traffic to every node, the center will be required to send more traffic in the y-direction simply because there are more nodes in the y-direction. Therefore, for a $M \times N$ mesh with $N > M$, the amount of capacity received by the crosslinks in the y-direction will be greater than the amount received by the crosslinks in the x-direction. (Note that this may not be true in the seamed case.) Nevertheless, for uniform traffic, there is still symmetry in the x-direction and y-direction separately. All the intra-plane crosslinks (links in y-direction) will have equal capacity and all the inter-plane crosslinks (links in x-direction) will have equal capacity as well. Note that when we map a $M \times N$ spherical constellation into a two-dimensional $M \times N$ rectangular grid, some of the links will be broken and some of the routes will not be optimum as a result. This effect will be further explained in section (3.2.2).

In order to find the required crosslink capacities, we first find the total number of hops required for the center node (0,0) to send traffic to all nodes in the mesh. The sum is the following:

$$\sum_{i=\frac{-(M-1)}{2}}^{\frac{M-1}{2}} \sum_{j=\frac{-(N-1)}{2}}^{\frac{N-1}{2}} |i| + |j| \tag{3.1}$$

However, for this calculation, we have to separate the crosslinks into two distinct groups: inter-plane crosslinks (y-direction) and intra-plane crosslinks (x-direction), since they have different capacities.

$$\underbrace{\sum_{i=\frac{-(M-1)}{2}}^{\frac{M-1}{2}} \sum_{j=\frac{-(N-1)}{2}}^{\frac{N-1}{2}} |i|}_{x-direction} + \underbrace{\sum_{i=\frac{-(M-1)}{2}}^{\frac{M-1}{2}} \sum_{j=\frac{-(N-1)}{2}}^{\frac{N-1}{2}} |j|}_{y-direction} \tag{3.2}$$

We first consider the inter-plane crosslinks. The number of hops in the inter-plane crosslinks is the following:

$$\begin{aligned}
H_{inter-plane} &= \sum_{i=\frac{-(M-1)}{2}}^{\frac{M-1}{2}} \sum_{j=\frac{-(N-1)}{2}}^{\frac{N-1}{2}} |i| \\
&= 4\sum_{i=1}^{\frac{M-1}{2}} \sum_{j=1}^{\frac{N-1}{2}} i + \text{hops from nodes on the x-axis} \\
&= 4\sum_{i=1}^{\frac{M-1}{2}} \sum_{j=1}^{\frac{N-1}{2}} i + 2\sum_{k=1}^{\frac{M-1}{2}} k \\
&= 4\frac{N-1}{2}\frac{\frac{M-1}{2}\frac{M+1}{2}}{2} + \frac{M-1}{2}\frac{M+1}{2} \\
&= (N-1)\frac{M-1}{2}\frac{M+1}{2} + \frac{M-1}{2}\frac{M+1}{2} \\
&= \frac{1}{4}[(N-1)(M-1)(M+1) + (M-1)(M+1)] \\
&= \frac{1}{4}(M-1)(M+1)N
\end{aligned} \tag{3.3}$$

Similarly, by symmetry, with M and N interchanged, the number of hops in the intra-plane crosslinks is the following:

$$H_{intra-plane} = \frac{1}{4}(N-1)(N+1)M \tag{3.4}$$

There are a total of NM nodes. As a result, the average number of hops per node in the x-direction is:

$$
\begin{aligned}
H_{avg_{inter-plane}} &= \frac{1}{4MN}(M-1)(M+1)N \\
&= \frac{1}{4}(M - \frac{1}{M}) \qquad (3.5)
\end{aligned}
$$

Similarly, for the y-direction:

$$H_{avg_{intra-plane}} = \frac{1}{4}(N - \frac{1}{N}) \qquad (3.6)$$

Note that this pair of results is good even for $M = 1$ or $N = 1$, or $M, N = 1$. According to the equations above, when $M = 1$, $H_{avg_{inter-plane}} = 0$. This is true because when M is equal to one, there will be only one plane and there will not be any traffic in any of the inter-plane crosslinks. The same happens when $N = 1$. When N is equal to one, $H_{avg_{intra-plane}} = 0$ as there will be only one node per plane and there will not be any traffic in any of the intra-plane crosslinks. When M and N are all equal to one, no traffic will be sent to crosslinks at all, and both $H_{avg_{intra-plane}}$ and $H_{avg_{inter-plane}}$ are zero as a result. Nevertheless, $M = 1$ or $N = 1$ is not a valid constellation for LEO satellites since the whole Earth cannot be fully covered with such a constellation.

We would like to find the capacities for the inter-plane and intra-plane crosslinks under uniform traffic. We can find the capacity by the following equation.

$$C_{crosslink} = \frac{C_{hop}\, H_{avg}\, \#(paths)}{\#(crosslinks)} \qquad (3.7)$$

where

$C_{crosslink}$ = capacity of a crosslink,

C_{hop} = capacity per path[1],

H_{avg} = average number of hops per path,

[1]Throughout this thesis, a path means a pair of origin and destination.

$\#(paths)$ = number of paths, and

$\#(crosslinks)$ = number of crosslinks.

Consider a stream of uniform traffic T_u. For a mesh of size $M \times N$, each path will have $T_u/(NM)^2$ amount of traffic. As a result, C_{hop} is equal to $T_u/(NM)^2$. Therefore, for inter-plane crosslinks,

$$\begin{aligned} C_{inter-plane} &= \frac{T_u}{(NM)^2}\frac{1}{4}(M - \frac{1}{M})\frac{(NM)^2}{2NM} \\ &= \frac{T_u}{8MN}(M - \frac{1}{M}) \qquad (3.8) \end{aligned}$$

By symmetry, for intra-plane crosslinks,

$$\begin{aligned} C_{intra-plane} &= \frac{T_u}{(NM)^2}\frac{1}{4}(N - \frac{1}{N})\frac{(NM)^2}{2NM} \\ &= \frac{T_u}{8MN}(N - \frac{1}{N}) \qquad (3.9) \end{aligned}$$

We have found the capacities of the inter-plane crosslinks and intra-plane crosslinks under uniform traffic. If $M = N$, we will have a square mesh, and the inter-plane and intra-plane crosslinks will have equal capacity.

$$C_{M=N} = \frac{T_u}{8M^2}(M - \frac{1}{M}) \quad or \quad \frac{T_u}{8N^2}(N - \frac{1}{N}) \qquad (3.10)$$

However, since the link distances are not equal to one another in this type of $M = N$ constellation, the system cost will be different.

If $N = 2M$, as in our satellite model, then the capacities for respective crosslinks will be:

$$\begin{aligned} C_{intra-plane} &= \frac{T_u}{8MN}(N - \frac{1}{N}) \\ &= \frac{T_u}{16M^2}(2M - \frac{1}{2M}) \qquad (3.11) \\ &\approx \frac{T_u}{8M} \qquad (3.12) \end{aligned}$$

$$
\begin{aligned}
C_{inter-plane} &= \frac{T_u}{8MN}(M - \frac{1}{M}) \\
&= \frac{T_u}{16M^2}(M - \frac{1}{M}) \qquad (3.13) \\
&\approx \frac{T_u}{16M} \qquad (3.14)
\end{aligned}
$$

In this case the link distances are equal, but the capacities are not. Continuing with the calculation, the ratio of intra-plane crosslink capacity to that of inter-plane crosslink for $N = 2M$ constellation is:

$$
\begin{aligned}
\frac{C_{intra-plane}}{C_{inter-plane}} &= \frac{\frac{T_u}{16M^2}(2M - \frac{1}{2M})}{\frac{T_u}{16M^2}(M - \frac{1}{M})} \\
&= \frac{(2M - \frac{1}{2M})}{(M - \frac{1}{M})} \\
&= \frac{1}{2}\,\frac{4M^2 - 1}{M^2 - 1} \qquad (3.15) \\
&\approx 2, \quad \text{as M gets large} \qquad (3.16)
\end{aligned}
$$

Therefore, for our satellite model, under uniform traffic, we could assume $C_{intra-plane} = 2\ C_{inter-plane}$. There is no total symmetry in the arrangement. Either we can space out the satellites at equal distances but the crosslinks will not have equal capacity (as in the $M \times 2M$ case), or we do not space out the satellites equally and the crosslinks will have equal capacity (as in the $M \times M$ case). However, for commercial systems such as Teledesic and Iridium, their crosslinks are designed to have equal amount of capacity. The reason for that is that the system would want to have spare crosslinks that can replace either an inter-plane or intra-plane crosslink whenever one is out of service. As a result, we would want to size the crosslinks to have the maximum capacity needed. With this scheme, the capacity of each crosslink will have the capacity of the intra-plane crosslinks, which is $\frac{T_u}{8M}$.

3.2.2 Validity of our mesh model

According to the above analysis, the inter-plane crosslinks should have twice the capacity of the intra-plane capacities under uniform traffic. However, in the case of Teledesic and Iridium, all the crosslinks in both systems have same amount of capacities. This would raise the question of whether our mesh model is correct or not.

There are no borders in a spherical surface of finite area as the surface is wrapped around. On the other hand, there is always a border in a two-dimensional surface of finite area. Thus, there is no easy way of mapping a spherical surface into a two-dimensional surface. If we force the transformation, some of the spherical surface properties will be lost. For example, in forming our $N = 2M$ model from the spherical model, some links between nodes have to be broken and this will affect our calculation of shortest routes among nodes. On the other hand, if we want to preserve all the links, we will arrive at a $2M \times 2M$ model, but we will have double-counted the number of nodes on the spherical surface instead. In Appendix A, we will devise a scheme to eliminate the double coverage of the $2M \times 2M$ network and this will lead to a result where all crosslinks, inter-plane and intra-plane, will receive the same amount of traffic.

It therefore seems like that we should not map the spherical constellation into a two-dimensional surface; however, this would further hamper our analysis as this would increase the difficulty of the analysis tremendously. Even though some of the links are lost in the $N = 2M$ model, one fix that is introduced to our analysis is the assumption of a homogeneous network (section 2.4.6). When the network is assumed to be homogeneous, every node will consider itself to be the center of the network, and connectivity to each neighboring node at the center is always maintained. As a result, the properties of a homogeneous $N = 2M$ network will resemble the properties of a spherical network of M planes and $2M$ satellites per plane. We have made an effort in making the two-dimensional model to look like a spherical network as much as possible. Some of the traffic under this assumption will not follow its actual shortest path. The capacities of the crosslinks derived in the previous subsection is thus an upper bound.

In using a $M \times 2M$ grid and making the sender to be the center of the network, we have actually favored intra-plane crosslinks over inter-plane crosslinks. At the center of a $M \times 2M$ grid, there are $\frac{M-1}{2}$ planes on each side. However, in a sphere there are actually $M - 1$ planes on each side because of the wrapping around of planes in a sphere. As a result, a path in this $M \times 2M$ grid can at most go around a quarter of the Earth latitude-wise. For example, under this $M \times 2M$ grid, using our routing algorithm, a path whose destination is exactly at the opposite end of the Earth can only use the intra-plane crosslinks, even though in an actual sphere, going through the inter-plane crosslinks is an equally good option which results in the same number of hop counts. For this routing algorithm, we have favored the intra-plane crosslinks over the inter-plane crosslinks, and we have thus got the result that $C_{intra-plane} = 2\, C_{inter-plane}$. In Appendix A we show a better model in which all paths will follow the shortest path routes. In this model, we obtain the result that $C_{intra-plane}$ and $C_{inter-plane}$ are both equal to $T_u(\frac{1}{12M} - \frac{1}{48M^3})$.

3.2.3 Seamed network

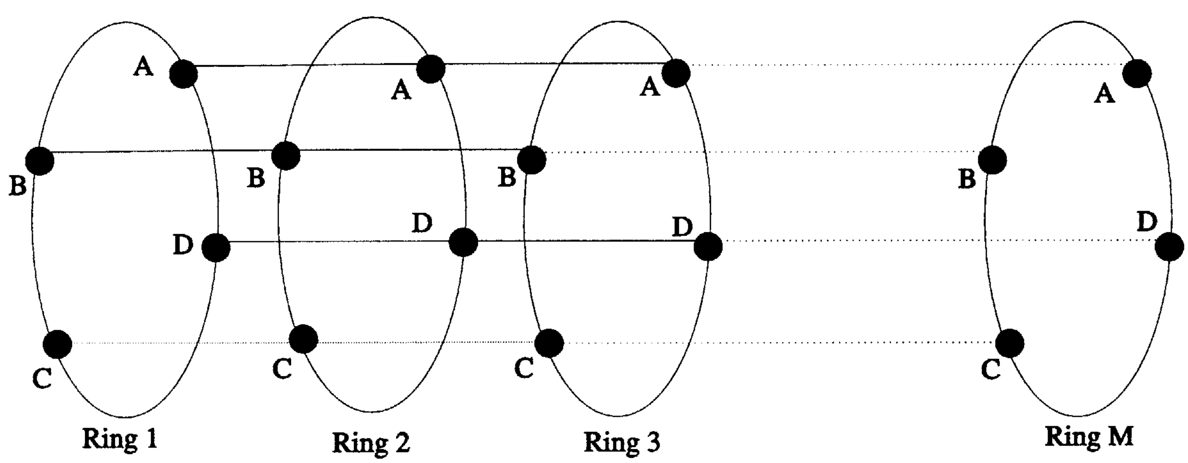

Figure 3-1: Seamed Network

In this section, we consider the case where there is no connectivity across the seam. As a result, we would expect the amount of traffic to be higher crossing the interior rings. Going cross-plane first and then intra-plane first, and going intra-plane first and then cross-plane will result in the same pattern for crosslink capacities, as we can regard one to be the reverse of the other. The paths are simply reversed and the capacities in crosslinks will not change. By symmetry, all intra-plane crosslinks in all planes will have equal capacities, and the inter-plane crosslinks between two rings will have same capacities also. Figure (3-1) is a representation of the seamed network.

We will consider traffic going cross-plane first and then intra-plane in this section, since this scheme is easier to analyze and as shown in the last paragraph, there is no loss in generality and optimality in letting all traffic to go cross-plane first and then intra-plane. Let's first consider the crosslink from node A of ring 1 to node A of ring 2, $C_{A1,A2}$. Since all the traffic will first go cross-plane, all the traffic in this crosslink must originate from the node A1. The amount of traffic originated from A1 is $\frac{T_u}{NM}$. Out of the M planes in the network, there are M planes to the right, and all the cross-plane traffic originating from A1 will need to pass through this link $C_{A1,A2}$. As a result, the amount of traffic in $C_{A1,A2}$ is:

$$\frac{T_u}{NM}\frac{M-1}{M} \tag{3.17}$$

All the crosslinks going from ring 1 to ring 2 will have this amount of capacity. We then look at the crosslink $C_{A2,A3}$. The traffic in this crosslink comes from ring 1 and ring 2 only. $\frac{1}{M-1}$ of the traffic from $C_{A1,A2}$ will be dropped in ring 2, and therefore, the amount of traffic in $C_{A2,A3}$ that is from ring 1 is:

$$\frac{T_u}{NM}\frac{M-1}{M}\frac{M-1}{M-2} = \frac{T_u}{NM}\frac{M-2}{M} \tag{3.18}$$

From the perspective of node A2, there are $M-2$ planes to the right. Therefore, similar to the argument above, the amount of traffic in $C_{A2,A3}$ that is from ring 2 is:

$$\frac{T_u}{NM}\frac{M-2}{M} \tag{3.19}$$

The total amount of traffic in $C_{A2,A3}$ is the sum of the two terms:

$$2\,\frac{T_u}{NM}\frac{M-2}{M} \tag{3.20}$$

We continue this calculation for each crosslink. For a crosslink going from plane m to plane $m+1$, there are m planes to the left, and therefore m nodes will be using that particular crosslink; there are also $M-m$ planes to the right. As a result, $\frac{M-m}{M}$ of the traffic originated from the m nodes to the left of the crosslink will use that crosslink. There is also $\frac{T_u}{NM}$ of traffic originating from each node. Therefore, in general, the amount of traffic in crosslink going from plane m to plane $m+1$, for $1 \le m \le M-1$, is:

$$C_{m,m+1} = m\frac{T_u}{NM}\frac{M-m}{M} = \frac{T_u}{NM^2}(M-m)m, \quad \text{for } 1 \le m \le M-1 \tag{3.21}$$

Similarly, by symmetry, the amount of traffic in crosslink going from plane m to plane $m-1$, for $2 \le m \le M$, is:

$$C_{m,m-1} = \frac{T_u}{NM^2}(M-m)m, \quad \text{for } 2 \le m \le M \tag{3.22}$$

We now show that the maximum capacity occurs in the center region. Although the function is an integer function, in order to find the maximum point, we can assume the function to be a continuous, reasonably smooth function and then look at the integers around the maximum point:

$$\frac{d}{dm}\left[\frac{T_u}{NM^2}(M-m)m\right] = \frac{T_u}{NM^2}(M-2m), \quad \text{for } 1 \le m \le M-1 \tag{3.23}$$

$$= 0, \quad \text{when } m = \tfrac{M}{2} \tag{3.24}$$

and

$$\frac{d^2}{dm^2}C_{m,m+1}, \frac{d^2}{dm^2}C_{m+1,m} = \frac{-2T_u}{NM^2}, \tag{3.25}$$

which is negative for all values of m.

Therefore, the maximum occurs at $\frac{M}{2}$ if M is even, or at $\frac{M}{2} - \frac{1}{2}$ and $\frac{M}{2} + \frac{1}{2}$ if M is odd. The maximum is $\frac{T_u}{NM^2}\frac{M}{2}\frac{M}{2}$ when M is even, or $\frac{T_u}{NM^2}(\frac{M}{2} - \frac{1}{2})(\frac{M}{2} + \frac{1}{2})$ when M is odd.

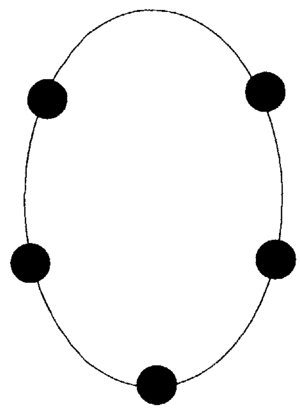

Figure 3-2: A Ring Constellation

To find the capacities in intra-plane crosslinks, we first try to find the total number of hops required for one node to send to every in-plane node. Let N be the number of satellites in the plane.
If N is odd, then

$$\text{Total number of hops} = \sum_{i=1}^{\frac{N-1}{2}} 2i = \frac{N+1}{2}\frac{N-1}{2} \quad (3.26)$$

$$= \frac{N^2-1}{4} \quad (3.27)$$

If N is even, then

$$\text{Total number of hops} = \sum_{i=1}^{\frac{N}{2}-1} 2i + \frac{N}{2} = \frac{N}{2}\frac{N-2}{2} + \frac{N}{2} \quad (3.28)$$

$$= \frac{N^2}{4} \quad (3.29)$$

There are N nodes in the plane; thus, the total number of hops for all nodes to send to every in plane node is:

$$\text{Total number of hops for all nodes} = \begin{cases} N(\frac{N^2-1}{4}) & : \text{ N odd} \\ \frac{N^3}{4} & : \text{ N even} \end{cases} \tag{3.30}$$

We will just consider the case when N is even since it is simpler to work with. There will be traffic coming from planes outside. For each in-plane node there will be exactly $M-1$ outside plane nodes which choose that particular node as an entry point into the plane. Including that particular in-plane node itself, the total number of nodes that choose that particular node as an entry point is therefore $M-1+1=M$. As a result, there will be $\frac{T_u M}{N^2 M^2} = \frac{T_u}{N^2 M}$ of in plane traffic from each in-plane node. There are $2N$ crosslinks in the system. Therefore,

$$\text{Capacity of each intra-plane crosslink} = \frac{\frac{T_u}{(N^2 M)}\frac{N^3}{4}}{2N} = \frac{T_u}{8M} \tag{3.31}$$

There is no dependency on N due to the uniform traffic assumption inside the ring. For traffic originating from each node, there will be always some which will travel through half of the ring to reach the farthest node, regardless of the number of nodes in the ring. Because of this independency on the number of nodes, the capacity inside each intra-plane crosslink depends only on the total amount of traffic, T_u.

We need to choose the maximum capacity for the size of the crosslink. Maximum traffic in inter-plane crosslinks can be found in the center region, and they have capacity:

$$C^*_{m,m-1} = \frac{T_u}{NM^2}\frac{M}{2}\frac{M}{2} = \frac{T_u}{4N}, \quad \text{for } M \text{ even} \tag{3.32}$$

Therefore, the capacity of the crosslink needs to be sized at:

$$\max\left(\frac{T_u}{4N}, \frac{T_u}{8M}\right) \tag{3.33}$$

If $N = 2M$, as in our satellite model, the crosslink capacity becomes:

$$\max\left(\frac{T_u}{8M}, \frac{T_u}{8M}\right) = \frac{T_u}{8M} \tag{3.34}$$

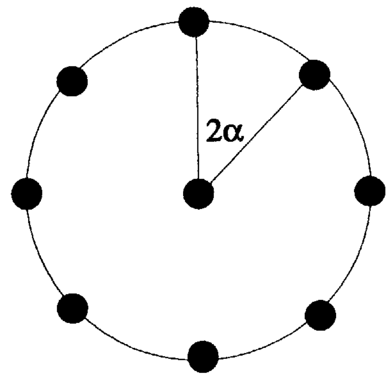

Figure 3-3: Geometry of Satellites in a Plane

The maximum capacities of inter-plane and intra-plane crosslinks are the same in this case. Furthermore, this crosslink capacity is the same as the crosslink capacity obtained in the seamless model in section (3.2.1). However in Appendix A, the crosslink capacity calculated is smaller—$\frac{T_u}{12M}$. This is because of the presence of crosslinks between the seam which allows more efficient routing. Less capacity is needed for the crosslinks as a result. However, cross-seamed crosslinks are expensive and impractical, and therefore this result of $\frac{T_u}{8M}$ for seamed network should always be used.

3.3 Expressing link distance, d, in terms of the number of planes, M

We have found the cost equation to be

$$\mathcal{C} = 4NM(k_0 + k_1 C^{\frac{\alpha}{4}} d^{\frac{\alpha}{2}}) \tag{3.35}$$

Before we could do any analysis, we would want to reduce the number of variables so that optimization can be done more easily. We have expressed C, the capacity of a crosslink, in terms of M, the number of planes in the system. In this section, we would want to express d—the distance between two satellites—in terms of M as well.

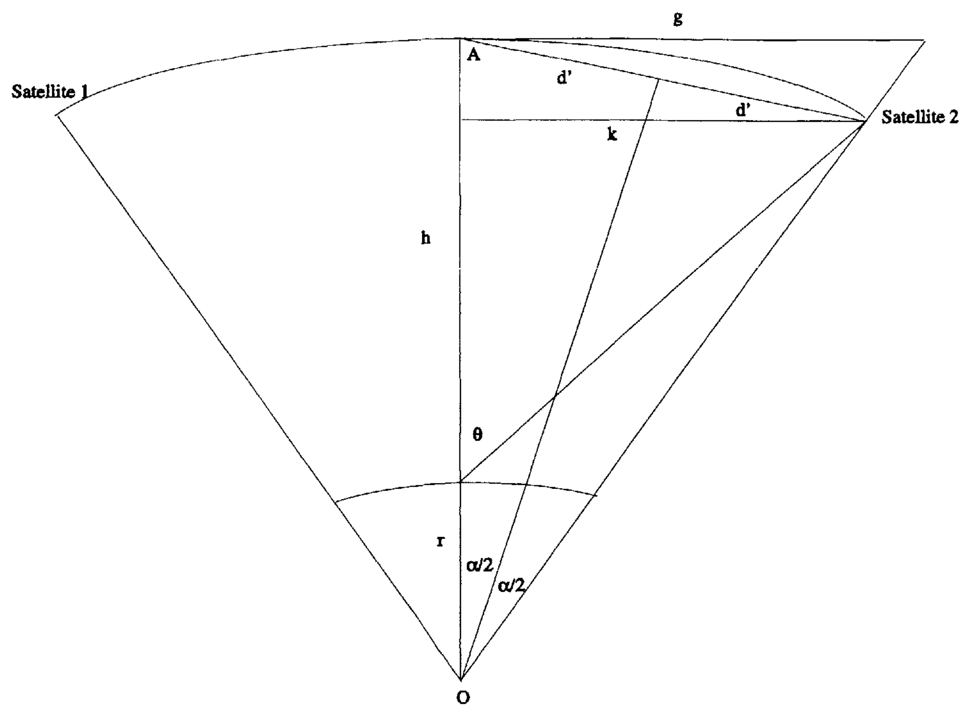

Figure 3-4: Geometry of Two Satellites in a Plane

First of all, we look at the approximations that will be made in our analysis.

3.3.1 Approximations going to be made in the calculation

In arriving at an expression of d in terms of M, we will use the following two approximations:

$$\frac{tan\theta}{tan\theta - \tan\frac{\pi}{2M}} \approx 1 \tag{3.36}$$

$$\sin\frac{\pi}{2M} \approx \frac{\pi}{2M} \tag{3.37}$$

The first approximation implies:

$$\tan\theta - \tan\frac{\pi}{2M} \approx \tan\theta\,, \quad \text{or} \tag{3.38}$$

$$\tan\theta \gg \tan\frac{\pi}{2M} \tag{3.39}$$

Using simple geometry, we have

$$\tan\theta = \frac{g}{h} \quad \text{and} \quad \tan\frac{\pi}{2M} = \frac{g}{h+r} \tag{3.40}$$

Therefore, (3.39) becomes:

$$\frac{g}{h} \gg \frac{g}{h+r} \tag{3.41}$$
$$\frac{1}{h} \gg \frac{1}{h+r} \tag{3.42}$$
$$h+r \gg h \tag{3.43}$$
$$\frac{h}{h+r} \ll 1 \tag{3.44}$$

We arbitrarily set the threshold to $\frac{1}{5}$:

$$\frac{h}{h+r} \leq \frac{1}{5} \tag{3.45}$$
$$5h \leq h+r \tag{3.46}$$
$$h \leq \frac{r}{4} = \frac{6376 \text{ km}}{4} \approx 1600 \text{ km} \tag{3.47}$$

This shows that this approximation is valid for LEO satellite systems, since the height of the LEO satellites is less than 1500 km.

Figure (3-5) is a graph of the comparison of $\sin x$ vs. x. From the graph, we can notice that $\sin x$ and x are approximately equal from $x = 0$ to 0.55. Thus, we require that $\frac{\pi}{2M}$ is between 0 and 0.55 for the approximation to be valid; this is equivalent to having M greater than 3. This clearly lies in our range of M since M is usually greater than 5. Therefore, both approximations (3.36) and (3.37) are valid in our analysis for LEO satellites.

3.3.2 Using the approximations in the cost equation

Figures (3-3) and (3-4) show the geometry of the satellites. As shown in figure (3-4), two satellites in the same plane make an angle of 2α at the center of the Earth. We would like to express d in terms of r—the radius of the Earth, h—the height of the

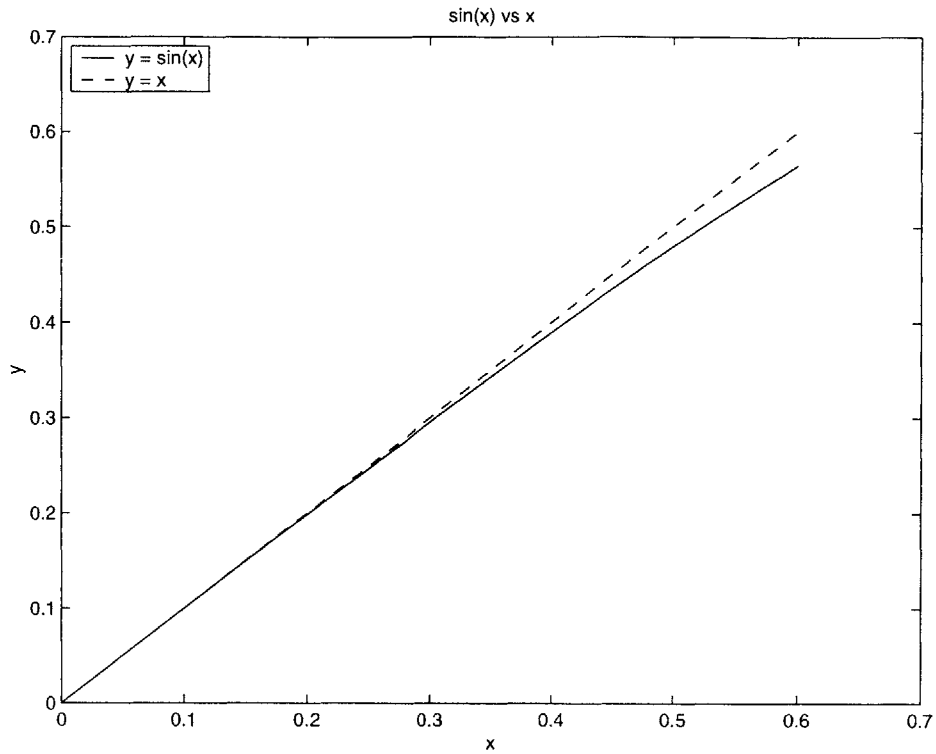

Figure 3-5: $\sin x$ vs x

satellite, and α. First, using basic trigonometry, we have:

$$\frac{d'}{r+h} = \sin(\frac{\alpha}{2}) \tag{3.48}$$

$$d' = (r+h)\sin(\frac{\alpha}{2}) \tag{3.49}$$

We then find angle A:

$$A = \frac{\pi - \alpha}{2} = \frac{\pi}{2} - \frac{\alpha}{2} \tag{3.50}$$

We then use A to find the relationship between d' and k, which is half of the distance between two satellites:

$$\sin A = \frac{k}{2d'} \tag{3.51}$$

From equation (3.51), and using equations (3.49) and (3.50), we get:

$$\begin{aligned} k &= 2d' \sin A \\ &= 2(r+h)\sin(\frac{\alpha}{2})\sin(\frac{\pi}{2}-\frac{\alpha}{2}) \\ &= 2(r+h)\sin(\frac{\alpha}{2})\cos(\frac{\alpha}{2}) \\ &= (r+h)\sin(\alpha) \end{aligned} \tag{3.52}$$

The distance between two satellites, d, is therefore:

$$d = 2k = 2(r+h)\sin(\alpha) \tag{3.53}$$

We have found d in terms of r, h and α. We then need find α in terms of h, r, and θ, which is the complementary angle of the minimum elevation angle of the satellite. The minimum elevation angle, in satellite networks, is the minimum angle between a satellite and the horizon such that a satellite can still communicate with the user on the ground. This angle is specific to a particular system. For example, the Teledesic model has a minimum elevation angle of 40°, while that of Iridium is only 8.2°. In our model, all satellites are separated with this minimum elevation angle with respect to the Earth surface. This is to ensure that the distance between two satellites will be minimized and therefore, the crosslink cost will be minimized as well. To find α, we again use simple trigonometry:

$$\frac{g}{h} = \tan(\theta) \quad , \quad \frac{g}{h+r} = \tan(\alpha) \tag{3.54}$$

Combining the two equations, we have:

$$\tan(\theta)h = \tan(\alpha)(h+r) \tag{3.55}$$

$$\alpha = \tan^{-1}[\tan(\theta)\frac{h}{h+r}] \tag{3.56}$$

We have $M = \frac{\pi}{2\alpha}$; therefore

$$M = \frac{\pi}{2\tan^{-1}[\tan(\theta)\frac{h}{h+r}]} \tag{3.57}$$

We then manipulate the above equation to get h:

$$\tan^{-1}[\tan(\theta)\frac{h}{h+r}] = \frac{\pi}{2M} \tag{3.58}$$

$$\tan(\theta)\frac{h}{h+r} = \tan(\frac{\pi}{2M}) \tag{3.59}$$

$$\frac{h+r}{h} = 1 + \frac{r}{h} = \frac{\tan(\theta)}{\tan(\frac{\pi}{2M})} \tag{3.60}$$

$$\frac{\tan(\theta)}{\tan(\frac{\pi}{2M})} - 1 = \frac{r}{h} \tag{3.61}$$

$$\begin{aligned} h &= \frac{r}{\frac{\tan(\theta)}{\tan(\frac{\pi}{2M})} - 1} \\ &= \frac{r\tan(\frac{\pi}{2M})}{\tan(\theta) - \tan(\frac{\pi}{2M})} \end{aligned} \tag{3.62}$$

We have found h in terms of M, and θ. Now put this h back in the expression of d.

$$\begin{aligned} d &= 2\sin(\alpha)(r+h) \\ &= 2\sin(\frac{\pi}{2M})(r + \frac{r\tan(\frac{\pi}{2M})}{\tan(\theta) - \tan(\frac{\pi}{2M})}) && (3.63) \\ &= 2\sin(\frac{\pi}{2M})r(1 + \frac{\tan(\frac{\pi}{2M})}{\tan(\theta) - \tan(\frac{\pi}{2M})}) && (3.64) \\ &= 2\sin(\frac{\pi}{2M})r(\frac{\tan(\theta)}{tan(\theta) - \tan(\frac{\pi}{2M})}) && (3.65) \\ &\approx 2\sin(\frac{\pi}{2M})r\frac{\tan(\theta)}{\tan(\theta)}, \quad \text{using approximation (3.36)} && (3.66) \\ &\approx 2\,\frac{\pi}{2M}\,r, \quad \text{using approximation (3.37)} && (3.67) \\ &= \frac{\pi r}{M}, \quad \text{valid for } M > 5 \text{ and } h < 1500 && (3.68) \end{aligned}$$

We have successfully reduced, d, the distance between two satellites, into a function of M only.

3.4 Cost optimization using the cost equation

We have derived an expression in terms of M for both d and C in the previous two subsections. We could now plug the expressions back into the cost equation, equation (3.35). In this analysis, we assume the constellation to be $M \times 2M$ and every crosslink will have the same capacity, $\frac{T_u}{8M}$, because this value is the capacity required for the intra-plane crosslinks, which demands more capacity than inter-plane crosslinks in a constellation of $M \times 2M$, and this value is also the capacity required for both intra-plane and inter-plane crosslinks in the seamed model.

$$\begin{aligned}
\mathcal{C} &= 4NM(k_0 + k_1 C^{\frac{\alpha}{4}}\, d^{\frac{\alpha}{2}}) && (3.69)\\
&= 8M^2(k_0 + k_1 C^{\frac{\alpha}{4}}\, d^{\frac{\alpha}{2}}) \quad (N = 2M) && (3.70)\\
&= 8M^2[k_0 + k_1\, (\frac{T_u}{8M})^{\frac{\alpha}{4}}\, (\frac{\pi r}{M})^{\frac{\alpha}{2}}], \text{ using } C = \tfrac{T_u}{8M} \text{ and equation (3.68)} && (3.71)\\
&= 8k_0M^2 + 8k_1(\frac{T_u}{8})^{\frac{\alpha}{4}}(\pi r)^{\frac{\alpha}{2}}\frac{1}{M^{\frac{3\alpha}{4}-2}} && (3.72)\\
&= k_0'M^2 + k_1'\frac{T_u{}^{\frac{\alpha}{4}}}{M^{\frac{3\alpha}{4}-2}} && (3.73)\\
&\qquad \text{with } k_0' = 8k_0 \text{ and } k_1' = 8^{(1-\frac{\alpha}{4})}k_1(\pi r)^{\alpha/2}
\end{aligned}$$

Note that there are two variables in the above cost equation—T_u and M. There are two types of optimization we can work on. Firstly, we can treat T_u as a constant and then optimize the cost equation over M, and secondly, we can treat both M and T_u as variables: we find the optimum M and T_u using profit maximization.

3.4.1 Treating T_u as a constant

Treating T_u to be a constant means in reality that the LEO system is faced with T_u amount of uniform traffic, and we try to find the optimum M such that the overall cost, $\mathcal{C}$, is minimum.

Differentiating equation (3.73) with respect to M, we have:

$$\frac{d\mathcal{C}}{dM} = 2k_0' M - (\frac{3}{4}\alpha - 2)k_1' \frac{T_u^{\frac{\alpha}{4}}}{M^{\frac{3}{4}\alpha - 1}} \tag{3.74}$$

To minimize $\mathcal{C}$, we set $\frac{d\mathcal{C}}{dM}$ to zero:

$$\begin{aligned}
2k_0' M^* - (\frac{3}{4}\alpha - 2)k_1' \frac{T_u^{\frac{\alpha}{4}}}{(M^*)^{\frac{3}{4}\alpha - 1}} &= 0 & (3.75)\\
2k_0'(M^*)^{\frac{3}{4}\alpha} &= (\frac{3}{4}\alpha - 2)k_1' T_u^{\frac{\alpha}{4}} & (3.76)\\
M^* &= [\frac{(\frac{3}{4}\alpha - 2)k_1' T_u^{\frac{\alpha}{4}}}{2k_0'}]^{1/\frac{3}{4}\alpha} & (3.77)\\
&= [k_2\,(\frac{3}{4}\alpha - 2)\;T_u^{\frac{\alpha}{4}}]^{\frac{4}{3\alpha}} & (3.78)\\
&= [k_2\,(\frac{3}{4}\alpha - 2)]^{\frac{4}{3\alpha}}\;T_u^{\frac{1}{3}} & (3.79)\\
&\quad \text{with } k_2 = \frac{k_1'}{2k_0'} & \\
&= k_3\;T_u^{\frac{1}{3}} & (3.80)\\
&\quad \text{with } k_3 = [k_2\,(\tfrac{3}{4}\alpha - 2)]^{\frac{4}{3\alpha}} &
\end{aligned}$$

Note that α needs to be greater than $\frac{8}{3}$; otherwise M^* will be negative. Taking the derivative results in minimal cost because

$$\frac{d^2\mathcal{C}}{dM^2} = 2k_0' + (\frac{3}{4}\alpha - 2)(\frac{3}{4}\alpha - 1)k_1' \frac{T_u^{\frac{\alpha}{4}}}{M^{\frac{3}{4}\alpha}} > 0 \tag{3.81}$$

for M^*, k_0', $k_1' > 0$ and $\alpha > \frac{8}{3}$.

3.4.2 Treating T_u and M as variables

In this section, we will treat T_u and M as variables, and try to find the operating point where the profit is maximized.

In order to find out the revenue, we first need to find out the revenue function of our satellite network. In the bandwidth market, since there is not one dominant player, we can assume that it is in perfect competition [7, p. 219], i.e. each network cannot affect the price of bandwidth by varying its amount of supply, but rather the price is a constant as determined by the market.

We will assume uniform traffic in this analysis. In uniform traffic every node sends to all nodes and therefore each node will generate the same amount of revenue. We can then express revenue as a linear function of T_u:

$$R(T_u) = aT_u + b \tag{3.82}$$

and

$$\frac{dR(T_u)}{dT_u} = a \tag{3.83}$$

The constant b is the fixed revenue, which may include subscription fees from the users and other source of revenue.

We then look at the cost function. We have found M^* in the last subsection. We can substitute M^* back into equation (3.73)—the cost equation—so that it will be expressed in terms of T_u only. After that, we will find $\frac{dC^*}{dT}$. Using (3.80), (3.73) becomes

$$\begin{aligned} C^* &= k_0' {M^*}^2 + k_1' \frac{{T_u}^{\frac{\alpha}{4}}}{{M^*}^{\frac{3\alpha}{4}-2}} && (3.84)\\ &= k_0'' T^{\frac{2}{3}} + k_1'' \frac{T_u^{\frac{\alpha}{4}}}{(T^{\frac{1}{3}})^{\frac{3\alpha}{4}-2}} && (3.85)\\ &= k_0'' T^{\frac{2}{3}} + k_1'' T_u^{\frac{2}{3}} && (3.86)\\ &= (k_0'' + k_1'') T_u^{\frac{2}{3}} && (3.87)\\ &\quad \text{with } k_0'' = k_0' {k_3}^2, \text{ and } k_1'' = k_1' {k_3}^{2-\frac{3\alpha}{4}} \end{aligned}$$

Then,

$$\frac{dC^*}{dT_u} = \frac{2}{3}(k_0'' + k_1'') T_u^{-\frac{1}{3}} \tag{3.88}$$

Note that $\frac{dC^*}{dT}$ is a strictly decreasing function for $T_u \geq 0$ and this suggests that the marginal cost will decrease as T_u increases. If we assume $R(T_u)$ to be a linear function, then the marginal revenue is simply a constant. This implies that we always should choose T_u to be as big as we can, in order to maximize profit. Therefore, we should always set the capacity of crosslinks to a point where it can accommodate maximum amount of uniform traffic allowable by capacities of the uplinks and downlinks, and by what the market can bear.

3.5 Geostationary orbit (GEO) satellite networks

In this section, we extend the analysis to geostationary orbit (GEO) satellite systems. Again, similar to the analysis for LEO satellite systems, we need to find expressions for C and d. We will show that the GEO satellite systems have only one orbit; therefore, we can assume the satellites to form a ring pattern, or simply a circle.

3.5.1 Finding the distance between two neighboring satellites

First of all, we can assume h, the height of a satellite, is a constant because there is only one circular orbit in the equatorial plane which is suitable for geostationary satellites. This is due to Kepler's Third Law [10, p.9]:

Theorem 3.1 (Kepler's Third Law) *The period T of a satellite and the mean distance—the semi-major axis of the elliptical orbit, a—are related as follows*

$$T^2 = \mu a^3 \tag{3.89}$$

where μ is a constant, which can be determined according to the dimensions of T and a.

With a in kilometers and T in mean solar days (a unit equal to 1.0027379 sidereal days that we use), μ has the value of 2.367×10^{-5}. It can be shown that for geostationary orbit, T is equal to 0.99727 [17]. Then, according to Kepler's third law, we obtain

$$a = \frac{1}{2.367 \times 10^{-5}} \times 0.99727^{2/3} \tag{3.90}$$

With circular orbit, a is simply the sum of the radius of Earth and the height of the Earth, $r + h$. The radius of the Earth is 6378 km. Therefore,

$$h = (42241 \times 0.99727^{2/3} - 6378) \text{ km} \tag{3.91}$$
$$\approx 35786 \text{ km} \tag{3.92}$$

Therefore, all the GEO satellites lie 35786 km above the ground. We can therefore express h as a constant.

Using the law of cosines, we can find d, the distance between two neighboring satellites, in terms of r, h, and N—the number of satellites in the system. The angle θ is equal to $\frac{2\pi}{N}$. Referring to the figure below:

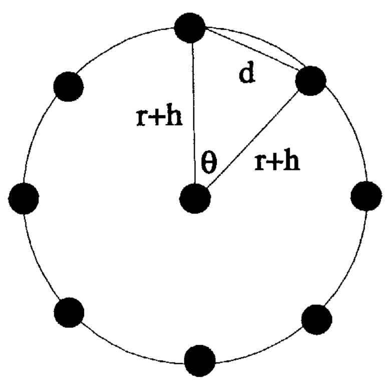

Figure 3-6: Geometry of geostationary satellites

$$d^2 = (r+h)^2 + (r+h)^2 - 2(r+h)^2 \cos\theta \tag{3.93}$$
$$= 2(r+h)^2(1-\cos\theta) \tag{3.94}$$
$$= 2(r+h)^2(1-\cos\frac{2\pi}{N}) \tag{3.95}$$

Therefore,

$$d = \sqrt{2}\,(r+h)\,(1-\cos\frac{2\pi}{N})^{\frac{1}{2}} \tag{3.96}$$
$$d^{\frac{\alpha}{2}} = 2^{\frac{\alpha}{4}}\,(r+h)^{\frac{\alpha}{2}}\,(1-\cos\frac{2\pi}{N})^{\frac{\alpha}{4}} \tag{3.97}$$

$$= \quad k_d\,(1-\cos\frac{2\pi}{N})^{\frac{\alpha}{4}}, \quad \text{with } k_d = 2^{\frac{\alpha}{4}}\,(r+h)^{\frac{\alpha}{2}} \tag{3.98}$$

3.5.2 Finding the capacity of each crosslink

We then need to find the capacity of the crosslink for uniform traffic. First of all note that this case is essentially the same as the intra-plane crosslink case in the seamed network model.

Recalling from section (3.2.3), the total number of hops for all nodes to send to every node is:

$$\text{Total number of hops for all nodes} = \begin{cases} \frac{N(N^2-1)}{4} & : \text{ N odd} \\ \frac{N^3}{4} & : \text{ N even} \end{cases} \tag{3.99}$$

For each path, there is $\frac{T_u}{N^2}$ of traffic going through, and there are $2N$ crosslinks in the system. Therefore,

$$\text{Capacity of each crosslink} = \begin{cases} \frac{\frac{T_u}{N^2}\frac{N(N^2-1)}{4}}{2N} = \frac{T_u N(N^2-1)}{8N^3} & : \text{ N odd} \\ \frac{\frac{T_u}{N^2}\frac{N^3}{4}}{2N} = \frac{T_u}{8} & : \text{ N even} \end{cases} \tag{3.100}$$

We will just consider the case when N is even since it is simpler to work with. Surprisingly, the capacity of a crosslink only depends on the total amount of traffic, T_u. It does not depend on the number of satellites in the system. This is due to the uniform traffic assumption in a ring discussed in section (3.2.3).

3.5.3 Plugging the values back into the cost equation

We have found an expression for both C and d. We can now substitute these expression in the cost equation. The cost equation for a GEO satellite system is

$$\mathcal{C} = 2N(k_0 + k_1 C^{\frac{\alpha}{4}}\, d^{\frac{\alpha}{2}}) \tag{3.101}$$

since there is only 2 crosslinks per satellite.

Substituting C and d into the equation, we have:

$$\mathcal{C} = 2N(k_0 + k_1\, C^{\frac{\alpha}{4}}\, d^{\frac{\alpha}{2}}) \tag{3.102}$$

$$= 2k_0\, N + 2k_1\, N\, \frac{T_u}{8}^{\frac{\alpha}{4}}\, k_d\, (1 - \cos\frac{2\pi}{N})^{\frac{\alpha}{4}} \tag{3.103}$$

The expression is very complex, and the derivative $\frac{d\mathcal{C}}{dN}$ will be more complex since it involves products and trigonometric functions. We cannot apply the approximations as in the LEO case because the approximations are no longer valid in the operation region. It would be very hard to solve $\frac{d\mathcal{C}}{dN} = 0$ analytically. Nevertheless, since the number of satellites in a GEO system cannot be that big (otherwise a LEO or a MEO would make more sense), we can simply plug in values ranging from 3 to ~10 (or some other number) for N in the cost equation and choose the N with the minimum cost. It can be seen that the optimum number of satellites in a GEO system, based on the cost equation (3.101), depends on the constants in the cost equation.

3.6 Numerical examples

In this section, a few numerical examples for LEO and GEO satellites are provided. We set α to 3 and have three sets of values for k_0 and k_1:

1. $k_0 = 1.000 \times 10^3, k_1 = 5.654 \times 10^{-9}$
2. $k_0 = 5.000 \times 10^5, k_1 = 4.243 \times 10^{-9}$
3. $k_0 = 1.000 \times 10^6, k_1 = 2.828 \times 10^{-9}$

All the above k_0's and k_1's give a total cost of 2 million dollar per crosslink with 10 Gbps and 50000 km. For each LEO case, the following graphs are produced to show the sensitivity of cost, crosslink capacity and constellation size to k_0 and k_1:

1. M vs. amount of traffic, using equation (3.80)
2. Capacity of a crosslink vs. amount of traffic, with capacity of a crosslink equal to $\frac{T_u}{8M^*}$

3. Total Cost vs. amount of traffic, using the cost equation, equation (3.35)

Detailed analysis of the LEO results will be provided in the next section.

For each GEO case, graphs are produced to show the sensitivity of total system cost to k_0 and k_1, with T_u equal to 1.0×10^9, 1.0×10^{12}, and 1.0×10^{15}. From the graphs, it can be seen that the minimum cost of a system depends very much on the total amount of traffic and the constants, k_0 and k_1. When T_u is equal to 1.0×10^9, fewest number of satellites results in lowest cost. However, when T_u is equal to 1.0×10^{12}, or 1.0×10^{15}, fewest number of satellites results in maximum cost.

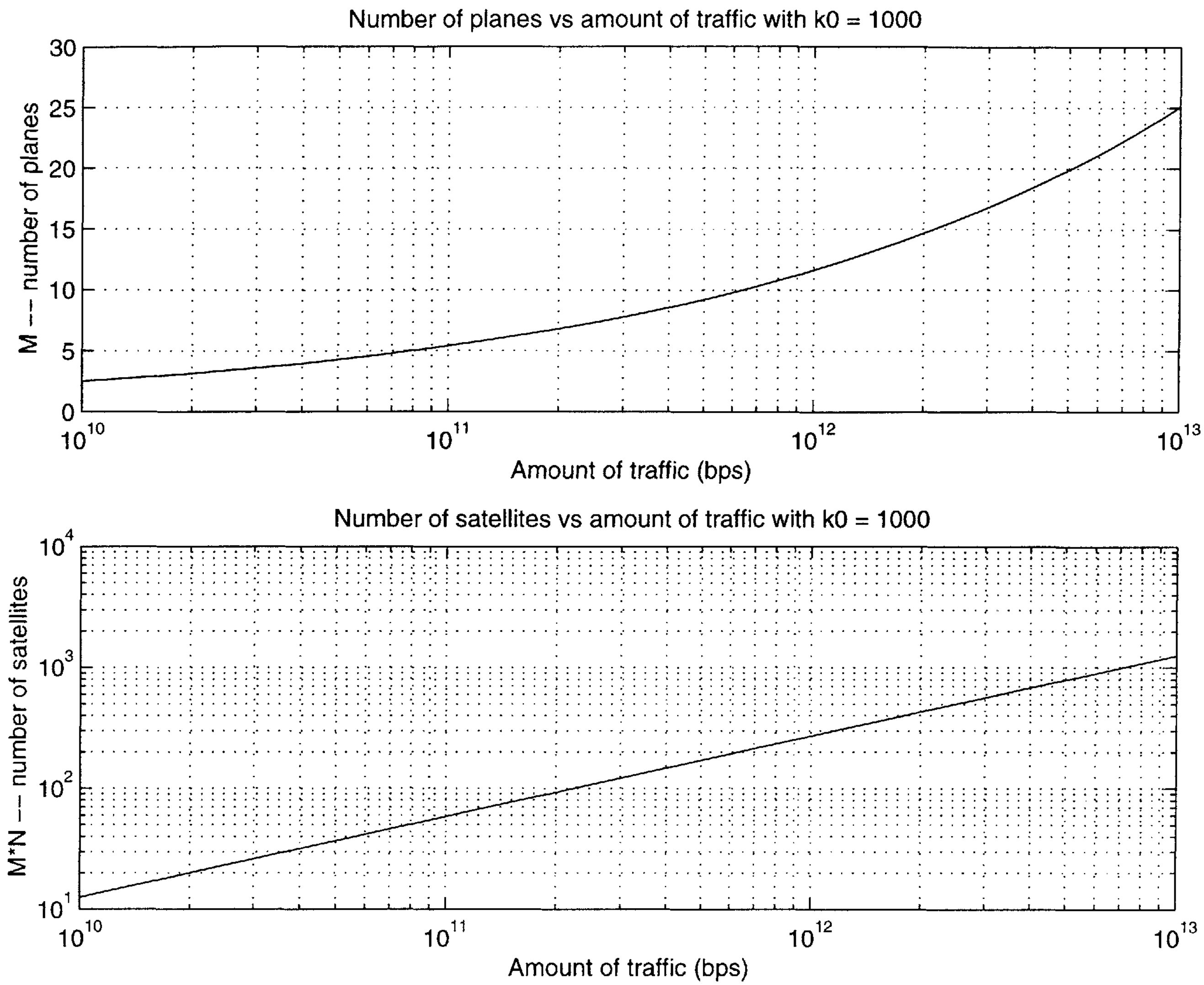

Figure 3-7: LEO: M & MN vs. amount of traffic, for $k_0 = 1000$

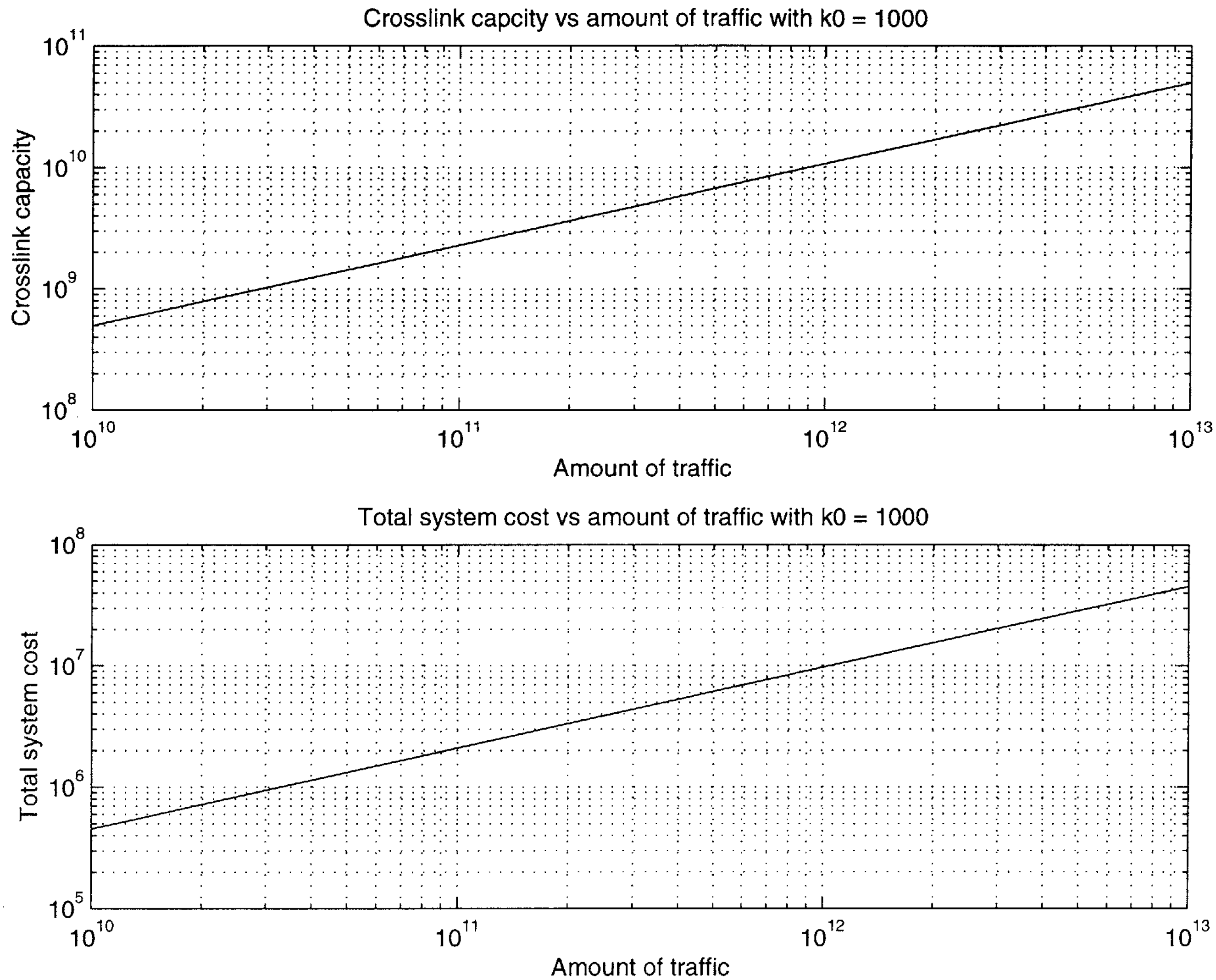

Figure 3-8: LEO: Crosslink capacity and total system cost vs. amount of traffic, for $k_0 = 1000$

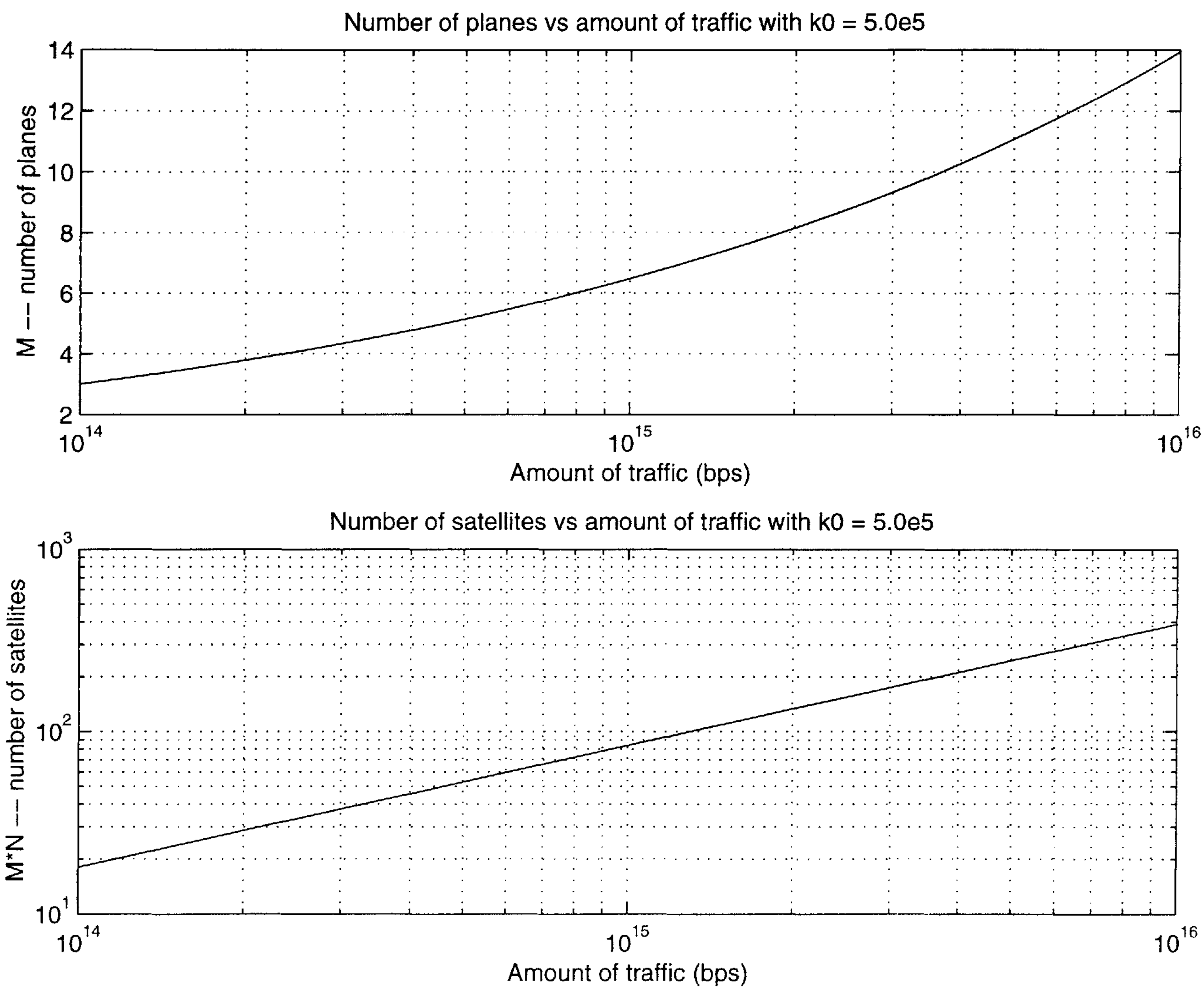

Figure 3-9: LEO: M & MN vs. amount of traffic, for $k_0 = 5.0 \times 10^5$

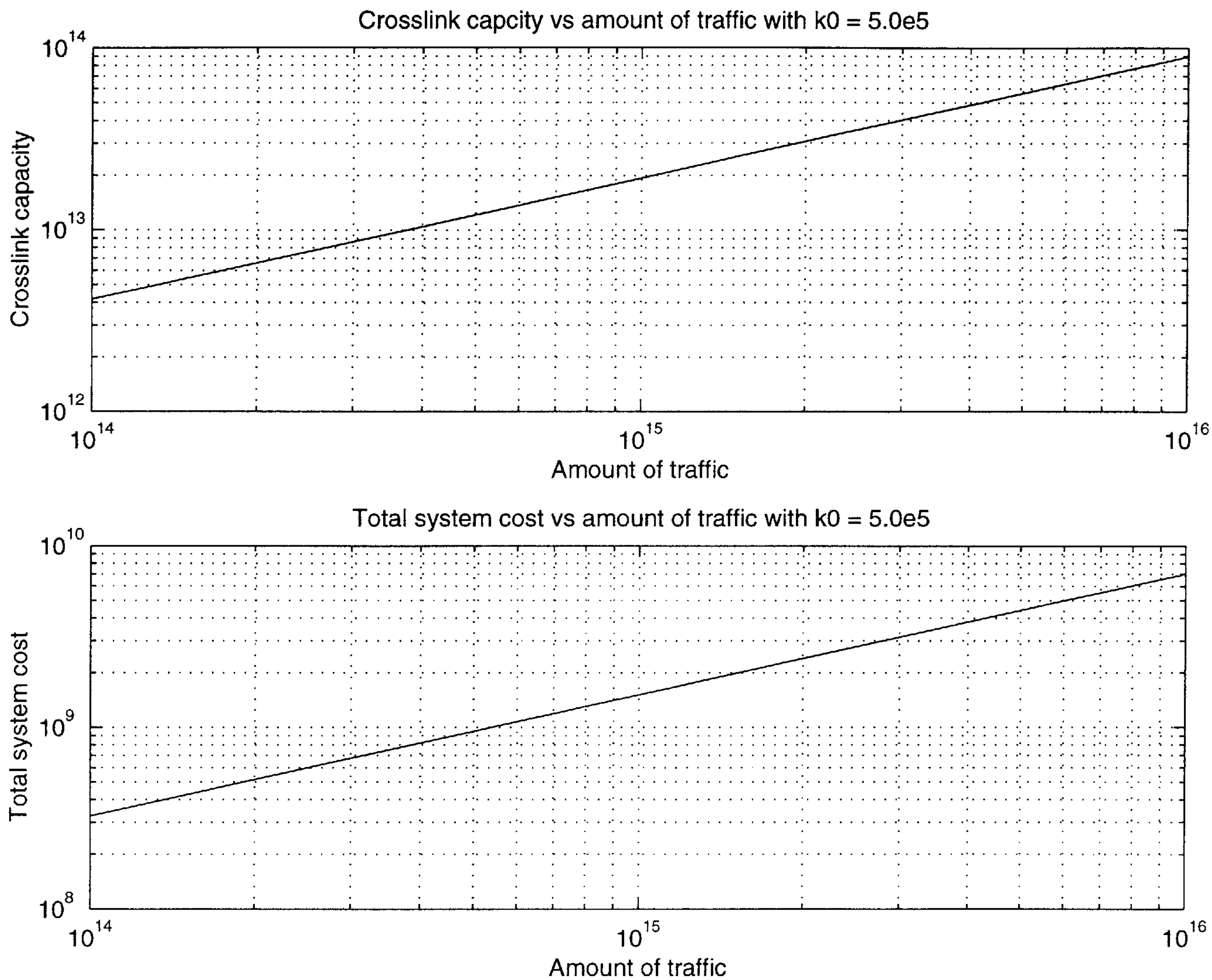

Figure 3-10: LEO: Crosslink capacity and total system cost vs. amount of traffic, for $k_0 = 5.0 \times 10^5$

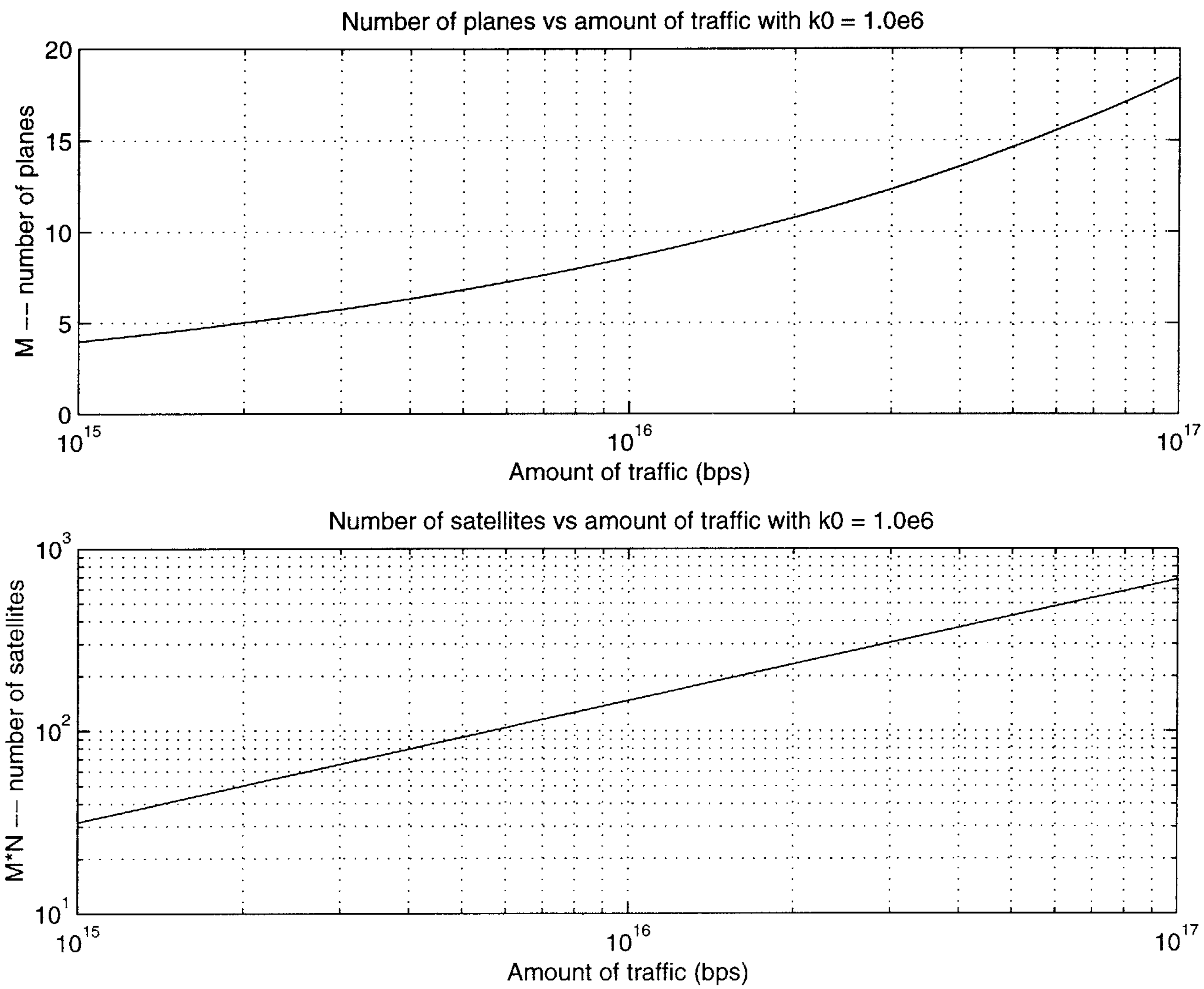

Figure 3-11: LEO: M & MN vs. amount of traffic, for $k_0 = 1.0 \times 10^6$

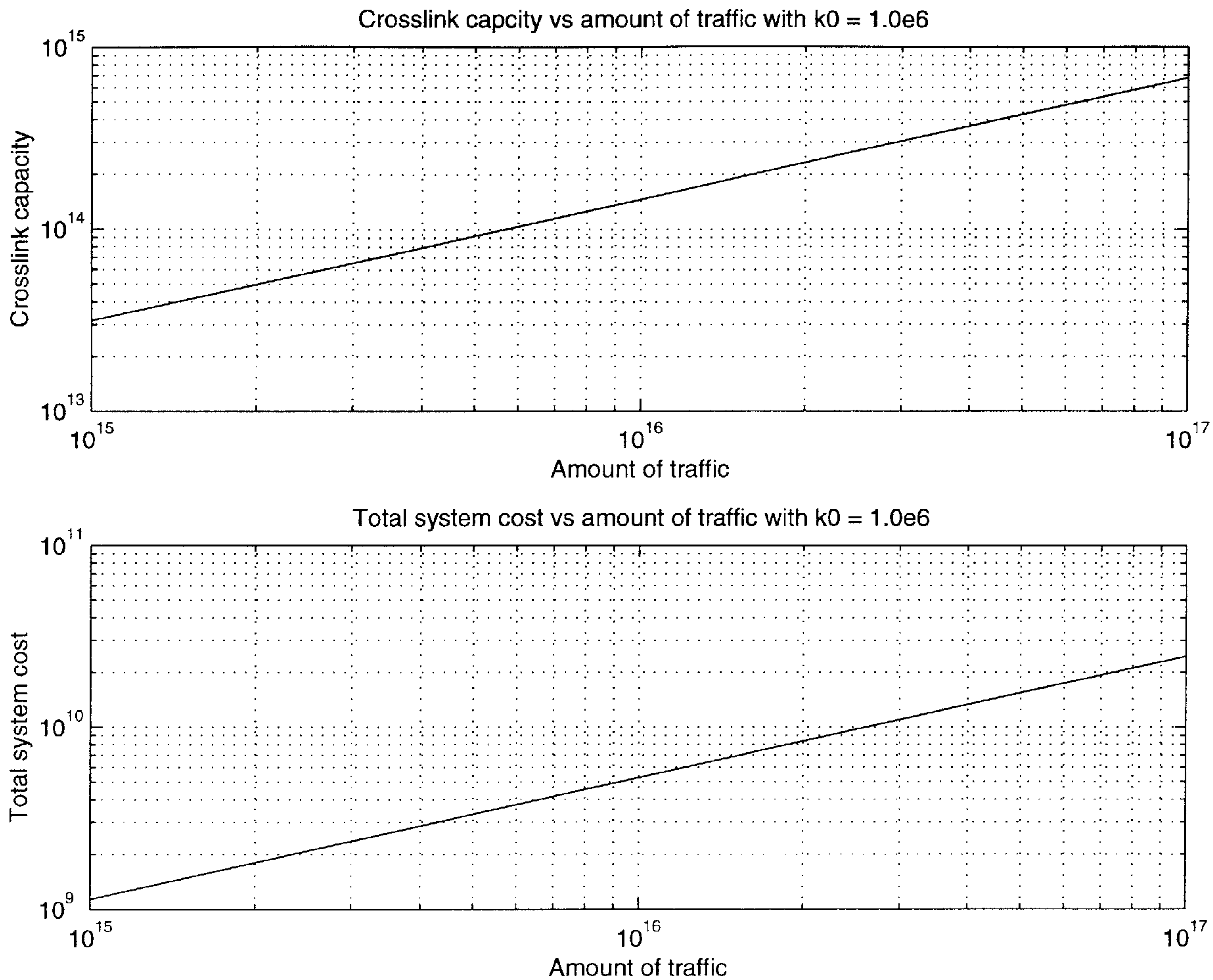

Figure 3-12: LEO: Crosslink capacity and total system cost vs. amount of traffic, for $k_0 = 1.0 \times 10^6$

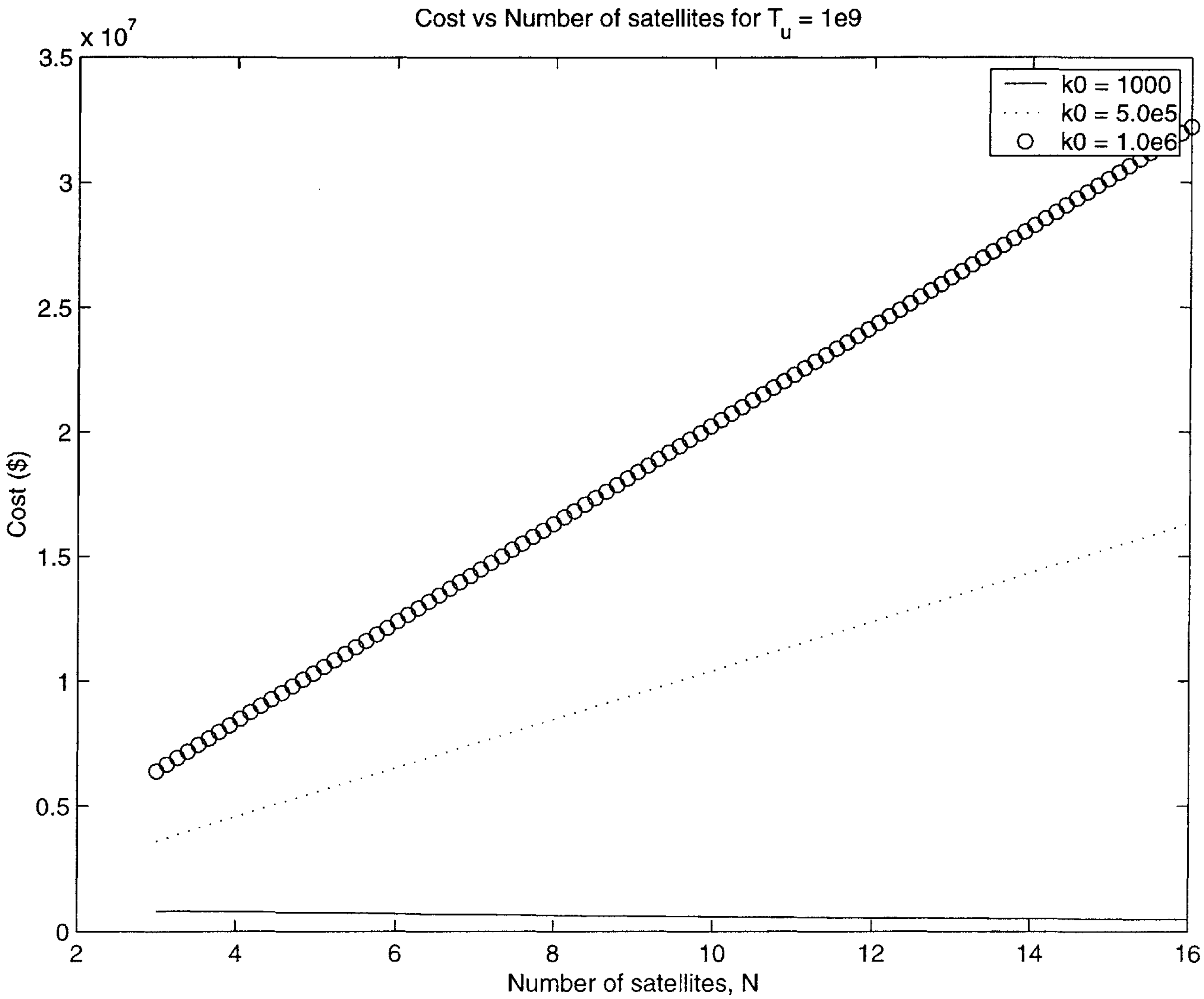

Figure 3-13: GEO: Total cost vs. number of satellites, with $T_u = 1.0 \times 10^9$

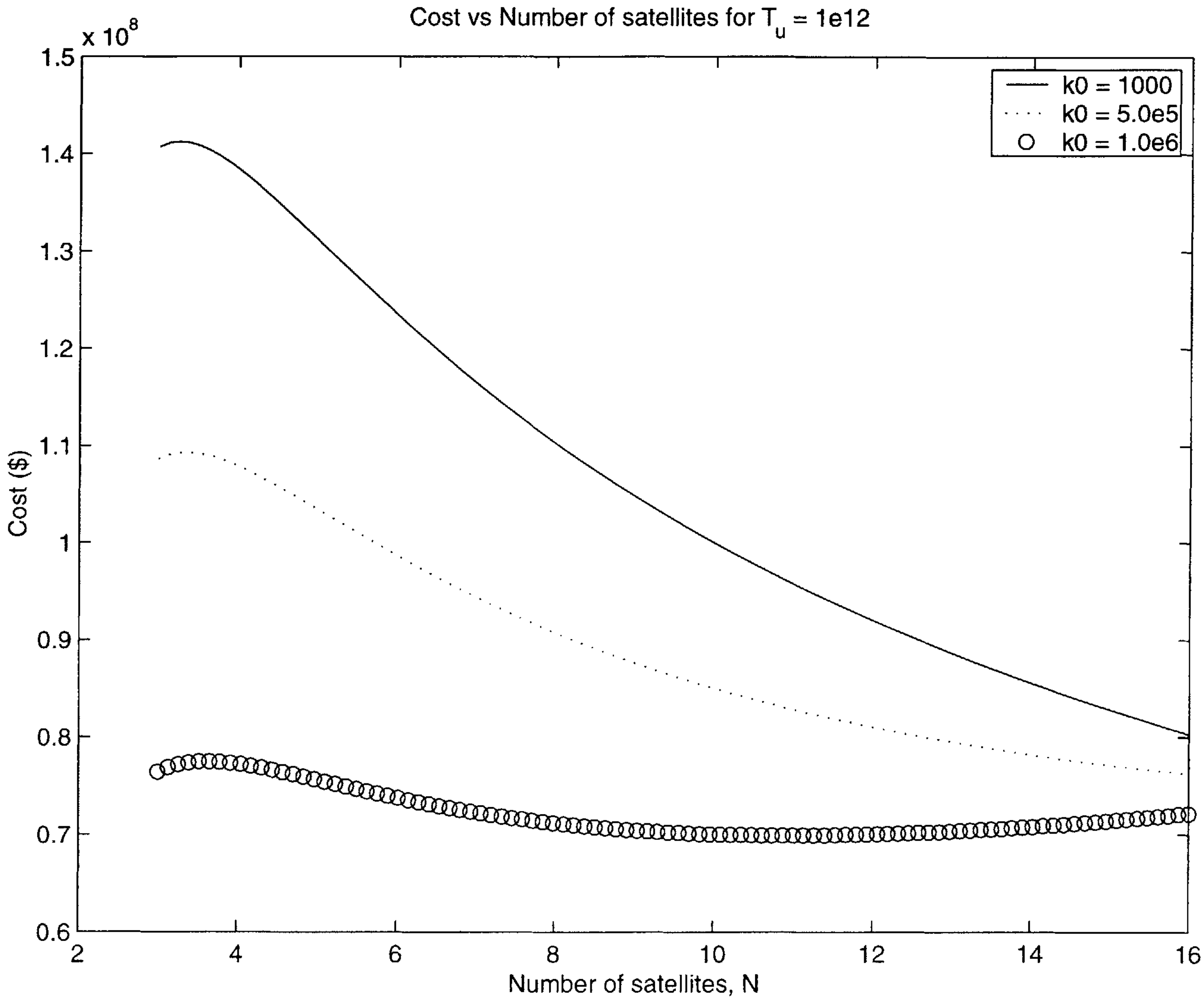

Figure 3-14: GEO: Total cost vs. number of satellites, with $T_u = 1.0 \times 10^{12}$

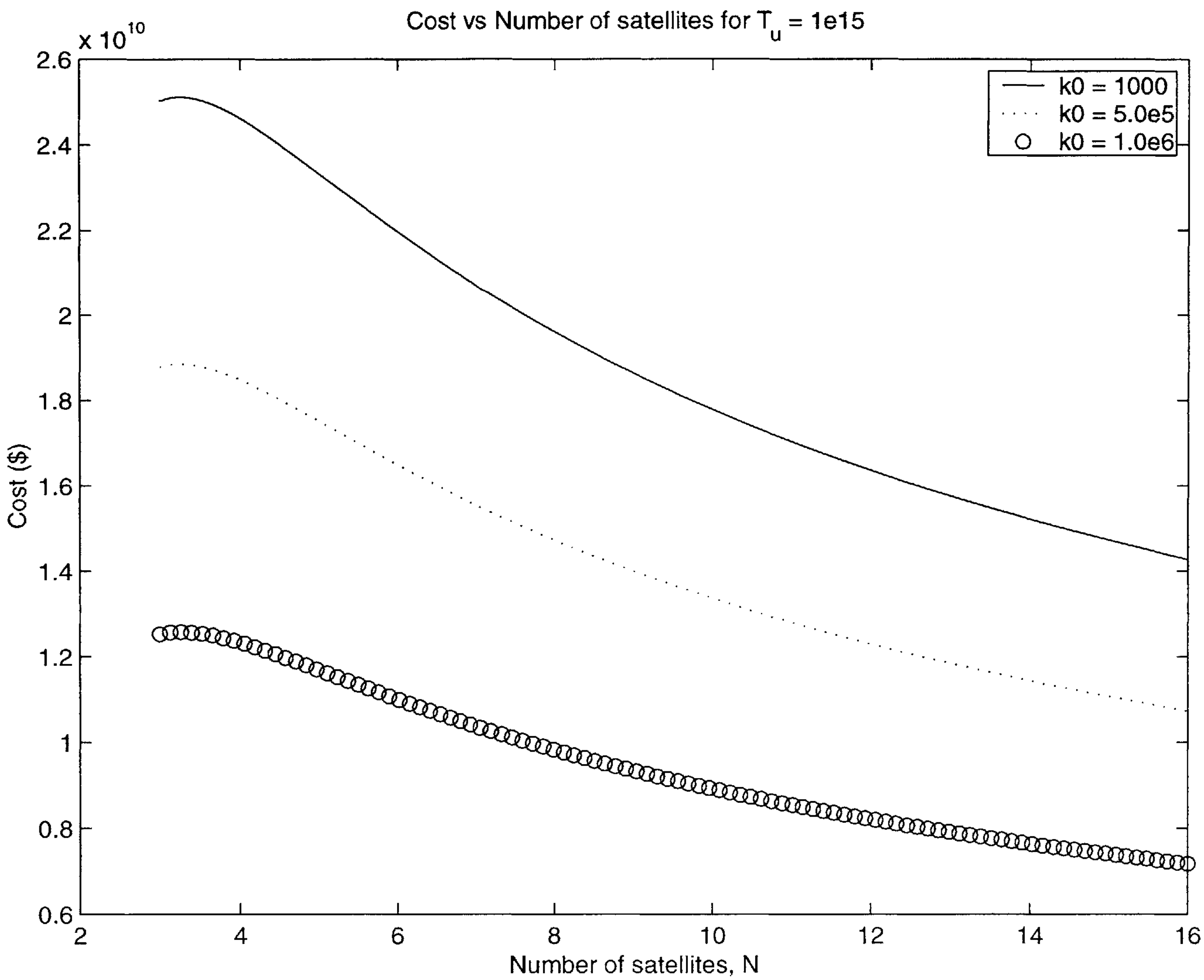

Figure 3-15: GEO: Total cost vs. number of satellites, with $T_u = 1.0 \times 10^{15}$

3.7 Interpretation of k_0 and k_1 in the cost equation for LEO satellite systems

We have plotted the results for the following three scenarios. The first scenario is the case where fixed cost is a very small proportion of the total cost; the third scenario is the case where fixed cost is half of the total cost; and the second scenario is the case where fixed cost is between the two extremes in the first and third scenarios.

Scenario	k_0	k_1	Fixed cost	Variable cost	Total cost
1	1.000×10^3	5.654×10^{-9}	1.000×10^3	1.999×10^6	2.000×10^6
2	5.000×10^5	4.243×10^{-9}	5.000×10^5	1.500×10^6	2.000×10^6
3	1.000×10^6	2.828×10^{-9}	1.000×10^6	1.000×10^6	2.000×10^6

Table 3.1: Three sets of values for k_0 and k_1

k_0 is essentially the fixed cost of a satellite, which includes the cost of the components which can be found in every satellite, such as batteries, transponders, and constant parts of the communication payload such as the acquisition and tracking system, payload control processors, etc., and this cost is independent of the distance and the capacity served. Intuitively, we would want to minimize the number of satellites when k_0 is large.

k_1 is associated with the variable cost of the crosslink, which is dependent on the distance served and the capacity of the crosslink. With $\alpha = 3$, the variable cost becomes $k_1 C^{3/4} d^{3/2}$. This term is rather insensitive to distance and capacity, since the exponentials to the term are very close to 1. Therefore, it seems to suggest that the fixed cost (k_0) plays a more important role than the variable cost in determining the optimized number of satellites.

The total amount of traffic in the network desired in current environment is between 100 Gbps ($1 \times 10^{11} bps$) to around 1 terabits per second ($1 \times 10^{12} bps$). When k_0 is in the range of 10^5 to 10^6, the number of satellites required to support this range of traffic is very small (M is equal to less than 3). This is infeasible for LEO satellites because with low altitudes this small number of satellites cannot possibly cover the entire surface of the Earth. This suggests that when k_0 and k_1 are found to have values similar to the last three cases, the LEO system would not have optimum constellation size if it were used to carry current traffic demand (100 Gbps to 1 Tbps). With similar $k0$ and $k1$, and with the number of satellites very close to that of Iridium or Teledesic, the system will be optimized if it were used to carry traffic that is in the range of 10^{14} to 10^{17}. It implies that a LEO system such as Teledesic and Iridium would be very costly to support today's traffic demand. However, as demand of bandwidth keeps on doubling every half a year (Moore's Law of Bandwidth) [8], such a system may make sense in a few years time. In fact, it is foreseeable that by 2001 Internet traffic will exceed 30 Tbps.

We have found that the fixed cost has to be ridiculously small (such as 1000 per crosslink) in order to support total traffic that is in line of current demand of bandwidth (100 Gbps to 1 Tbps). When k_0 is 5.00×10^5 or 1.00×10^6, the optimal range of traffic that a system should support is between 1.00×10^{14} bps to 1.00×10^{17} bps, as shown in figures (3-9) and (3-11). This finding suggests that the high fixed cost of satellites has made a LEO system very costly to support low latency Quality of Service in current situation. Note that the equation for the optimized number of satellites depends on the ratio between k_0 and k_1. When k_0 is 1000, the ratio (k_0/k_1) is 1.77×10^{11}; when k_0 is 10000, the ratio is 1.77×10^{12}. Therefore, for a LEO satellite system with the number of satellites similar to that of Iridium and Teledesic, we have found in this analysis a necessary ratio between the two constants such that their number of satellites is optimized—k_0 and k_1 must have a ratio in the magnitude of 10^{11} to 10^{12}. However, this ratio holds only when the fixed cost of a satellite is very low.

With this analysis, we can make the following conclusions about the constellations of Iridium and Teledesic:

- The Teledesic architecture is more efficient when it supports traffic that far exceeds current traffic demand. They are designed to handle traffic of the future. However, their system specifications seem to suggest the contrary.

- The designs of their systems have not taken into account the cost of the crosslinks. Their systems are not optimized with the number of satellites required. They have wasted too much capital in building extra satellites and crosslinks.
- They may have a cost equation different from the one we have.
- LEO satellite systems may be too costly to support current level of traffic. Alternatives, such as terrestrial optical cables, may provide a more cost-efficient solution to support global traffic.
- Previous studies on the optimization of the required number of satellites seem to focus solely on height as the only variable for the cost of the satellites. The capacity and distance served of the crosslinks are insignificant to the total manufacturing costs in their calculations. In Richharia [16], it was assumed that minimum number of satellites would result in the most minimum network cost and as a result, orbital height which governs the total number of satellites and hence the total cost of the network is used as an important optimization parameter. Our results, on the other hand, do not rely very much on the orbital height but instead rely on the the crosslink capacities and link distances. Different assumptions will lead to different results and therefore, our results may not be totally consistent with other people's results.

3.8 Summary

In this chapter, we have found an optimum constellation based on crosslink costs given any particular amount of uniform traffic T_u. In the process of getting this, we first have found the amount of capacity in a crosslink with uniform traffic, in the un-seamed and seamed model. We then found a way to express d in terms of M, using some valid approximations. The values we have obtained are plugged back into the cost equation for optimization. We have extended this optimization exercise to GEO satellite systems as well. Lastly, numerical results are plotted to explore the sensitivity of the system cost to the constants k_0 and k_1 in the cost equation.

Chapter 4

Reducing the number of crosslinks per node

4.1 Introduction

In previous chapter, we have derived the cost equation of the network to be:

$$\mathcal{C} = 4NM[k_0 + k_1 C^{\alpha/4} d^{\alpha/2}] \tag{4.1}$$

In this cost equation, we have assumed that there are four crosslinks (two inter-plane crosslinks and two intra-plane crosslinks) per node to give a symmetric (with respect to connectivity) constellation. Since the total cost depends very much on the number of crosslinks, it may be sensible to reduce the number of crosslinks per node. This chapter will explore this idea of reducing the number of crosslinks to minimize system cost.

4.2 Will this scheme actually result in cost reduction?

Fixed cost can contribute to as much as half of the total cost of the network, even in future systems. As a result, one logical approach is to reduce the number of crosslinks

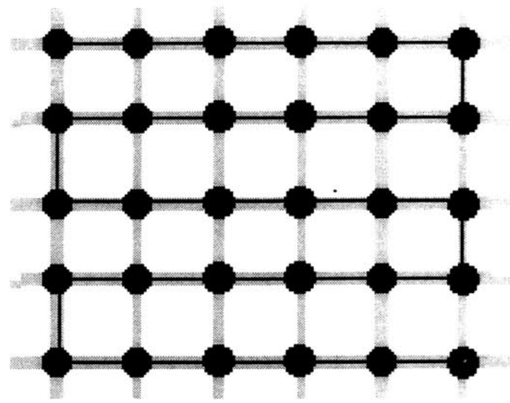

Figure 4-1: One crosslink per node network

per node as much as we can, while still maintaining continuous global coverage. We could reduce the average number of crosslinks per node to one and are still able to maintain continuous global coverage (Figure 4-1). However, this would increase the average number of hops per path since most of the links are no longer available. If we want minimal effect on the increase of path lengths while trying to decrease the number of crosslinks per node, we may want each node to have 3 links to neighboring nodes in order to maintain a mesh-like coverage. This 3-crosslink-per-node type of network is constructed from our original square constellation in which every node has 4 crosslinks to its neighbors. Each node has two inter-plane crosslinks as in the original constellation. However, in every plane, we first remove the left inter-plane crosslinks from all the odd-numbered nodes, and then the right inter-plane crosslinks from all the even-numbered nodes, and the result will be a 3-crosslink-per-node network (Figure 4-2). Its new cost equation will then be:

$$\mathcal{C} = 2NM[k_0 + k_1 C'^{\alpha/4} d^{\alpha/2}] + NM[k_0 + k_1 (C'')^{\alpha/4} d^{\alpha/2}] \tag{4.2}$$

Let's take a close look at the new cost equation. We first look at the variable cost part (the terms with k_1). The distance between crosslinks remains to be the same as the original square constellation since no nodes have been moved around or removed. The only variables that have changed are therefore C' and C'', the capacity of the intra-plane and inter-plane crosslinks. Since α usually ranges from 2

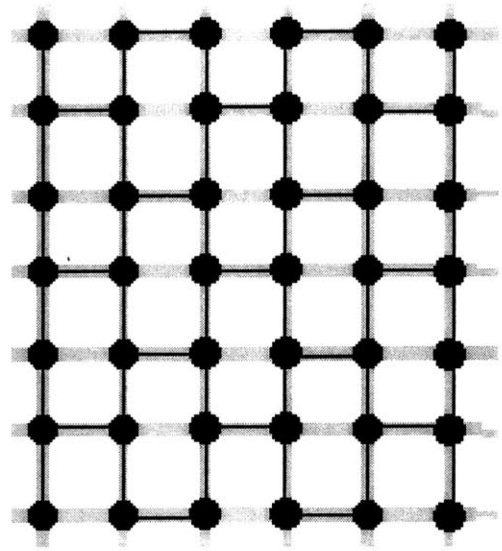

Figure 4-2: 3-crosslink-per-node mesh network

to 3, the exponent to C' and C'', $\alpha/4$, is therefore less than 1. As a result, the cost function with respect to the capacity is a concave function. Therefore, the variable cost exhibits increasing return to scale, since the exponent associated with capacity is less than 1. When the crosslink doubles its capacity, the variable cost associated with this doubling of capacity is less than the variable cost of having two crosslinks of the original capacity to support the same amount of traffic (Figure 4-3). Furthermore, reducing the number of crosslinks will result in a reduction of total fixed cost (the terms with k_0). Therefore, it seems that reducing the number of crosslinks is always a good idea.

However, the question of whether the total cost will decrease depends also on the amount of capacity increased per crosslink when the number of crosslinks is reduced. The capacity of each crosslink will increase due to the reduction of the number of crosslinks available. In fact, if the original constellation carries C amount of uniform traffic per crosslink, the new constellation will require each crosslink to carry at least $4/3C$ amount of uniform traffic, due to, firstly, the decrease in the number of crosslinks, and secondly, due to the re-routing required when some of the crosslinks are no longer available. Some of the paths will need additional hops.

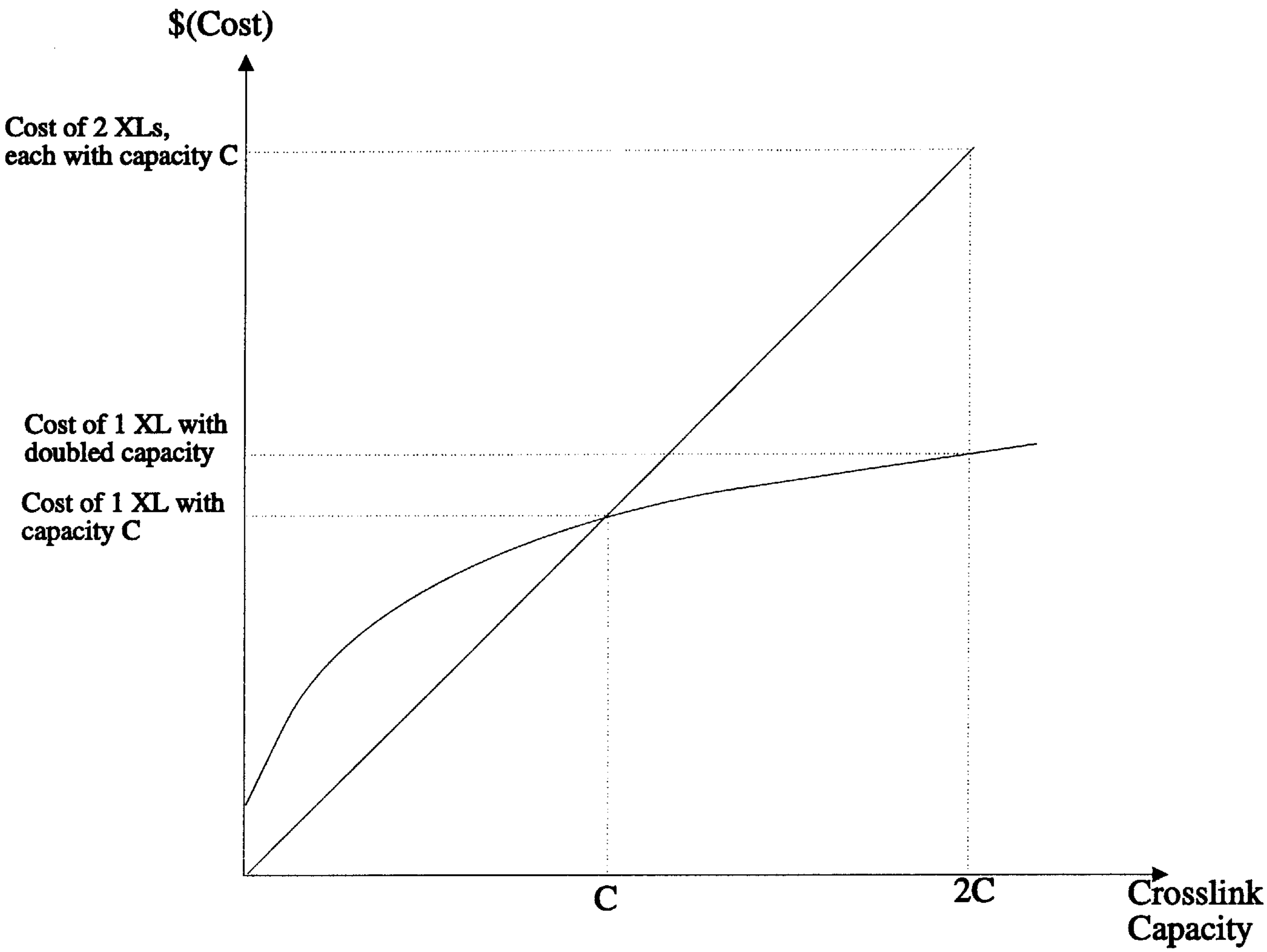

Figure 4-3: Cost difference between a crosslink with capacity C and a crosslink with capacity 2C

As a result, there are two opposing factors in the new cost equation:

1. There are fewer crosslinks, and this lowers the fixed cost.

2. The capacity of each crosslink will increase as a result of the decrease in the number of crosslinks. This will increase cost.

Therefore, it is not clear whether reducing the number of crosslinks per node will actually decrease the total cost of the system. In the next section, we will find the necessary conditions such that reducing the number of crosslinks will actually result in the decrease of total system cost.

4.3 Necessary conditions for reduction of total system costs when reducing the number of crosslinks per node

There are four variables in this discussion of the reduction of crosslinks per node. The first two are the number of inter-plane and intra-plane crosslinks per node, and the third and fourth variables are the amount of increased capacity per inter-plane crosslink and intra-plane crosslink. Since we would want to achieve maximum cost reduction, we assume in this analysis that the new crosslinks do not need to have equal capacities—intra-plane and inter-plane crosslinks can have different capacities. We will try to find the necessary relationship among the four variables for actual cost reduction. Since the intra-plane crosslinks and the inter-plane crosslinks have different capacities, and the amount of increased capacities is different in the two cases, we cannot bundle the inter-plane and intra-plane crosslinks together as one case. We also assume the seamless model in this analysis. Even though the seam of the network is ignored, our results, for capacity increased in crosslinks, will still be a lower bound to the results from the seamed network. This is a lower bound because with a seamed model, traffic can no longer go through the seam and as a result, there will be more traffic for the inter-plane crosslinks in the seamed case.

First of all, we let E_0 to be the number of inter-plane crosslinks to be removed per node, and E_1 be the number of intra-plane crosslinks to be removed per node.

Secondly, we let C' and C'' to be the new required capacities of an inter-plane crosslink and an intra-plane crosslink respectively, when E_0 of the inter-plane crosslinks and E_1 of the intra-plane crosslinks are reduced, the capacities in the crosslinks will change:

$$C' = \mu_0 C, \quad and \quad C'' = \mu_1 C, \quad where\ \mu_0, \mu_1 \geq 1 \tag{4.3}$$

μ_0 and μ_1 are always greater than or equal to 1 because C' and C'' are always greater than or equal to C for all cases. With these new variables, the new cost equation has become:

$$
\begin{aligned}
\mathcal{C}' &= (2 - E_0)\ NM\ [k_0 + k_1 C''^{\alpha/4} d^{\alpha/2}] + (2 - E_1)\ NM\ [k_0 + k_1 C''^{\alpha/4} d^{\alpha/2}] \\
&= (4 - E_0 - E_1)\ NM\ k_0 + (2 - E_0)\ NM\ k_1\ d^{\alpha/2} C'^{\alpha/4} + \\
&\quad (2 - E_1)\ NM\ k_1\ d^{\alpha/2} C''^{\alpha/4} \qquad (4.4)
\end{aligned}
$$

Using equation (4.3), we get:

$$
\begin{aligned}
\mathcal{C}' &= (4 - (E_0 + E_1))\ NM\ k_0 + (2 - E_0)\ NM\ k_1\ d^{\alpha/2} C^{\alpha/4} \mu_0^{\alpha/4} + \\
&\quad (2 - E_1)\ NM\ k_1\ d^{\alpha/2} C^{\alpha/4} (\mu_1)^{\alpha/4} \\
&= (4 - (E_0 + E_1))\ NM\ k_0 + NM\ k_1\ d^{\alpha/2} C^{\alpha/4} [(2 - E_0)\mu_0^{\alpha/4} + \\
&\quad (2 - E_1)(\mu_1)^{\alpha/4} \qquad (4.5)
\end{aligned}
$$

We want the reduction of the number of crosslinks to result in a decrease in cost, i.e.

$$
\mathcal{C} - \mathcal{C}' > 0 \qquad (4.6)
$$

Expanding $\mathcal{C}$ and $\mathcal{C}'$ using equations (4.1) and (4.5), we obtain:

$$
\begin{aligned}
&4NM[k_0 + k_1 C^{\alpha/4} d^{\alpha/2}] - NM\ k_0[4 - (E_0 + E_1)] - \\
&\quad NM\ k_1\ d^{\alpha/2} C^{\alpha/4} [(2 - E_0)\mu_0^{\alpha/4} + (2 - E_1)\mu_1^{\alpha/4}] > 0 \qquad (4.7)
\end{aligned}
$$

$$
(E_0 + E_1)\ NM\ k_0 + NM\ k_1\ d^{\alpha/2} C^{\alpha/4} [4 - (2 - E_0)\mu_0^{\alpha/4} - (2 - E_1)\mu_1^{\alpha/4}] > 0 \qquad (4.8)
$$

$$
\begin{aligned}
-(E_0 + E_1)\ k_0 &< k_1\ d^{\alpha/2} C^{\alpha/4} [4 - (2 - E_0)\mu_0^{\alpha/4} - (2 - E_1)\mu_1^{\alpha/4}] \qquad (4.9) \\
\frac{-(E_0 + E_1)\ k_0}{k_1\ d^{\alpha/2} C^{\alpha/4}} &< [4 - (2 - E_0)\mu_0^{\alpha/4} - (2 - E_1)\mu_1^{\alpha/4}] \qquad (4.10)
\end{aligned}
$$

Note that the left hand side of the inequality is proportional to the negative of $(E_0 + E_1)$ times the ratio of fixed cost to variable cost of an inter-plane crosslink. We denote this ratio to be θ.

$$-(E_0 + E_1)\theta - 4 \;<\; -(2 - E_0)\mu_0^{\alpha/4} - (2 - E_1)\mu_1^{\alpha/4} \tag{4.11}$$

$$(E_0 + E_1)\theta + 4 \;>\; (2 - E_0)\mu_0^{\alpha/4} + (2 - E_1)\mu_1^{\alpha/4} \tag{4.12}$$

We have therefore determined the relationship of how much capacity per inter-plane crosslink and intra-plane crosslink is allowed to increase such that the total system cost will decrease. We first need to find μ_0 and μ_1 for that particular network, and then plug it into the equation (4.12) and determine whether the condition is satisfied or not.

4.4 3-crosslink-per-node mesh network

In this section we will examine the 3-crosslink-per-node seamless mesh network as discussed at the beginning of the section, and determine whether the additional capacity required per crosslink will exceed the value as determined by equation (4.12). We consider a $(M + 1) \times (2M + 1)$ network, M even, for reason of symmetry. We will further assume that the traffic in intra-plane crosslinks is twice as much as the traffic in inter-plane crosslinks. In this constellation, E_0 is equal to 1 because there is one inter-plane crosslink removed at every node. E_1 is zero because no intra-plane crosslinks are removed. Therefore, equation (4.12) becomes:

$$\theta + 4 > \mu_0^{\alpha/4} + 2\mu_1^{\alpha/4} \tag{4.13}$$

We first need to find μ_0, the ratio of increased capacity to original capacity in each inter-plane crosslink. Consider an s-t cut across two planes that cuts the nodes into two groups, group $\mathcal{S}$ and group $\mathcal{T}$ (Figure 4-4). In the original $(M + 1) \times (2M + 1)$ network with no deletion of crosslinks, there are $2M + 1$ inter-plane crosslinks across that cut. Since all the crosslinks have equal capacity, and if we let C be the capacity

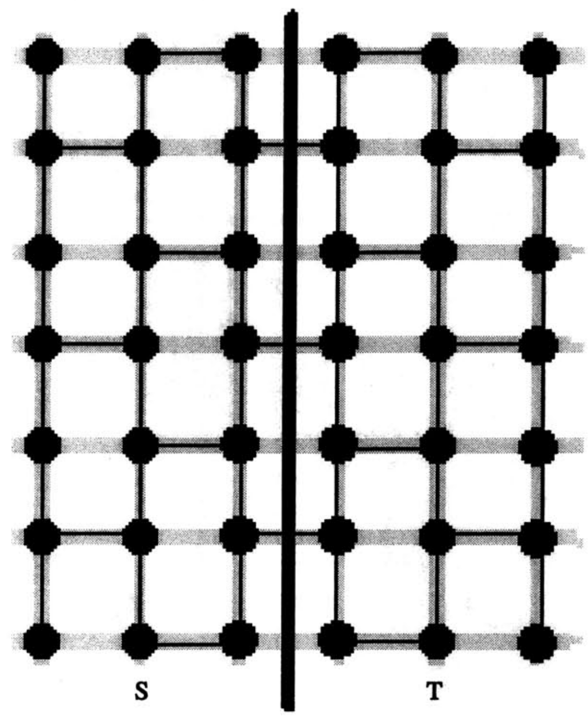

Figure 4-4: An s-t cut across two planes

in one of these crosslinks, the amount of traffic going from $\mathcal{S}$ to $\mathcal{T}$ is $(2M+1)C$. If we now remove one inter-plane crosslink from each node, following the scheme just described before, there will be either M or $M+1$ inter-plane crosslinks remaining across the cut. Since we have assumed uniform traffic, each inter-plane crosslink will again receive equal amount of traffic. The amount of traffic in each of new crosslinks will be $\frac{(2M+1)C}{M}$, or $\frac{(2M+1)C}{M+1}$. Then,

$$\begin{aligned} \frac{\text{new amount of traffic}}{\text{original amount of traffic}} &= \frac{\frac{(2M+1)C}{M}}{C} \quad or \quad \frac{\frac{(2M+1)C}{M+1}}{C} \\ &\approx 2, \quad \text{when M is large} \end{aligned} \tag{4.14}$$

However, since we have discovered that only half of the capacity is used in each inter-plane crosslink in the original constellation, half of the capacity in each inter-plane crosslink is free. As a result, we only need to increase the inter-plane crosslink capacity by half of the capacity increased, i.e., μ_0 is equal to half of (4.14), which is 1. (Note that this result, $\mu_0 = 1$, is only valid for the seamless case, not for the seamed case.)

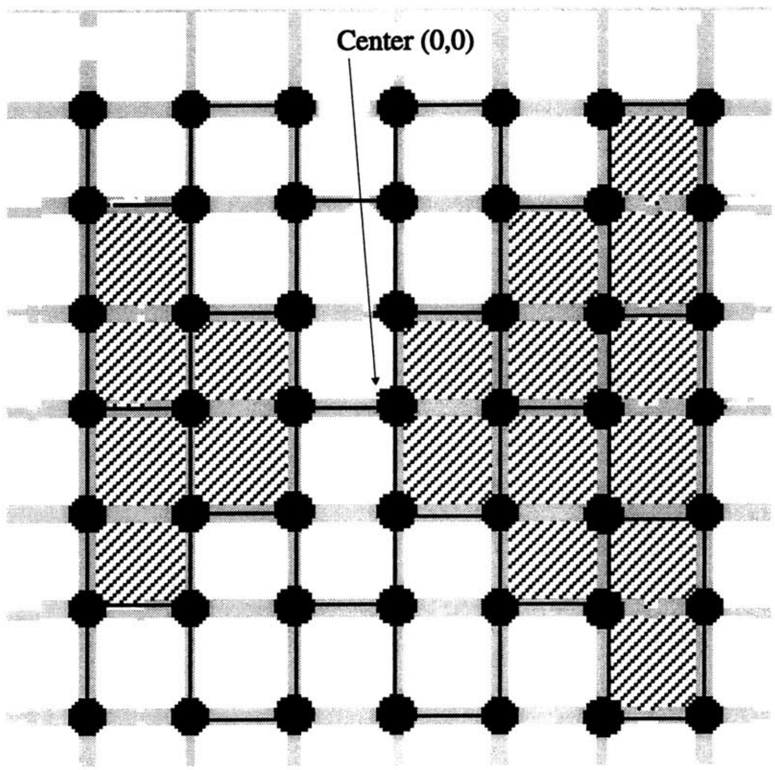

Figure 4-5: Paths that require additional hops

We have found μ_0, and our next step is to find μ_1. First of all, we find the paths which require additional number of hops as a result of the change of constellation. We first consider traffic which originates from the center and travel to every node. As shown in the figure (4-5), paths which require additional number of hops are those not colored in grey. We first consider the right side of the diagram. At one horizontal hop away from the origin, the only destination node which requires additional hop is (1, 0), and it requires two additional vertical hops. At two horizontal hops away, the destination nodes which require additional hops are (2, -1), (2, 0) and (2, 1). Each path needs two additional vertical hops, and therefore the total sum of extra hops is 6. We then look at destinations which are three horizontal hops away. There are five nodes which require additional hops: (3,0) requires four additional hops and the rest require two additional hops. The total number of additional hops required is therefore $4 + 2 + 2 + 2 + 2 = 12$. We can show, by mathematical induction, that for paths on the right hand side of the figure, the number of additional vertical hops for paths that terminate at i horizontal distance away is:

$$a_i = \sum_{j=1}^{i} 2i = i^2 + i \tag{4.15}$$

Proof: First of all, at one unit of horizontal distance away from the center, as mentioned in the last paragraph, the number of additional vertical hops required is 2, and

$$a_1 = \sum_{j=1}^{1} 2i = 1^2 + 1 = 2 \tag{4.16}$$

Suppose that the number of additional vertical hops for paths that terminate at $i - 1$ horizontal distance away is $(i-1)^2 + (i-1)$, as according to equation (4.15). We now consider paths at i horizontal distance away. If we compare the destinations at horizontal distance i to those at horizontal distance $i - 1$, there will be two more nodes in the grey shaded region (notice that the grey shaded region is like a triangle; there will be one additional node at the top and one additional node at the bottom as we increase the horizontal distance by 1) and each of them requires two additional vertical hops, and $i - 2$ of the nodes will require two more additional hops. As a result, as we increase the horizontal distance by one, there will be an increase of $2 \times 2 + 2 \times (i - 2) = 2i$. Therefore, the number of additional hops for paths that terminate at i horizontal distance away is:

$$(i-1)^2 + (i-1) + 2i = i^2 + i = \sum_{j=1}^{i} 2i = a_i \tag{4.17}$$

Therefore, by mathematical induction, the number of additional vertical hops for paths that terminate at i horizontal distance away is determined by equation (4.15).

There are $\frac{M}{2}$ planes to the right of the center node; as a result, the total number of additional hops on the right hand side of the diagram is:

$$\sum_{i=1}^{\frac{M}{2}} a_i \tag{4.18}$$

The left hand side of the diagram follows the same pattern, except that the pattern starts at two horizontal hops, not at one horizontal hop. Therefore, the total number of additional hops on the left hand side of the diagram is:

$$\sum_{i=1}^{\frac{M}{2}-1} a_i \tag{4.19}$$

The total number of additional hops required is thus equal to:

$$\sum_{i=1}^{\frac{M}{2}} a_i + \sum_{i=1}^{\frac{M}{2}-1} a_i \tag{4.20}$$

We first examine the first term:

$$\begin{aligned}
\sum_{i=1}^{\frac{M}{2}} a_i &= \sum_{i=1}^{\frac{M}{2}} i^2 + \sum_{i=1}^{\frac{M}{2}} i &(4.21)\\
&= \frac{1}{6}\frac{M}{2}(\frac{M}{2}+1)(M+1) + \frac{1}{2}(\frac{M}{2}+1)\frac{M}{2} &(4.22)\\
&= (\frac{M}{2}+1)M[\frac{1}{12}(M+1)+\frac{1}{4}] &(4.23)\\
&= (\frac{M}{2}+1)M(\frac{M}{12}+\frac{1}{3}) &(4.24)
\end{aligned}$$

The second term is:

$$\begin{aligned}
\sum_{i=1}^{\frac{M}{2}-1} a_i &= \sum_{i=1}^{\frac{M}{2}} a_i - a_{\frac{M}{2}} &(4.25)\\
&= (\frac{M}{2}+1)M(\frac{M}{12}+\frac{1}{3}) - (\frac{M}{2})^2 - \frac{M}{2} &(4.26)\\
&= M(\frac{M}{12}+\frac{1}{3})(\frac{M}{2}+1) - M(\frac{M}{4}+\frac{1}{2}) &(4.27)
\end{aligned}$$

$$= M(\frac{M^2}{24} + \frac{M}{6} + \frac{M}{12} + \frac{1}{3} - \frac{M}{4} - \frac{1}{2}) \quad (4.28)$$

$$= M(\frac{M^2}{24} - \frac{1}{6}) \quad (4.29)$$

The total number of additional hops is therefore

$$H_{add} = (\frac{M}{2} + 1)M(\frac{M}{12} + \frac{1}{3}) + M(\frac{M^2}{24} - \frac{1}{6}) \quad (4.30)$$

$$= M(\frac{M^2}{12} + \frac{3M}{12} + \frac{1}{12}) \quad (4.31)$$

$$= \frac{M}{12}(M+2)(M+1) \quad (4.32)$$

$$= \frac{M^3 + 3M^2 + 2M}{12} \quad (4.33)$$

In chapter 3, we have derived the average number of hops in intra-plane crosslinks in the original constellation is:

$$H_{intra-plane} = \frac{1}{4}(N' - 1)(N' + 1)M' \quad (4.34)$$

In our constellation, $M' = M + 1$, and $N' = 2M + 1$, then

$$H_{intra-plane} = \frac{1}{4}2M(2M+1)(M+1) \quad (4.35)$$

$$= \frac{12M^3 + 18M^2 + 6M}{12} \quad (4.36)$$

μ_1 is therefore:

$$\frac{H_{intra-plane} + H_{add}}{H_{intra-plane}} = \frac{13M^3 + 21M^2 + 8M}{12M^3 + 18M^2 + 6M} \quad (4.37)$$

$$\approx \frac{13}{12}, \quad \text{when M is large} \quad (4.38)$$

We have found μ_0 and μ_1, and can plug them back into the inequality (4.13). The right hand side of (4.13) becomes:

$$\mu_0^{\alpha/4} + 2\mu_1^{\alpha/4} = 1 + 2(\frac{13}{12})^{\frac{\alpha}{4}} \quad (4.39)$$

The left hand side of (4.13) is $\theta + 4$. Since θ is a ratio of fixed cost and variable cost, it is positive, and since α ranges from 2 to 3, the right hand side will never be greater than 4; therefore, the inequality will always be satisfied in this new constellation. Cost will definitely be reduced by using this type of 3-crosslink-per-node mesh network.

4.5 1-inter-plane-crosslink mesh network

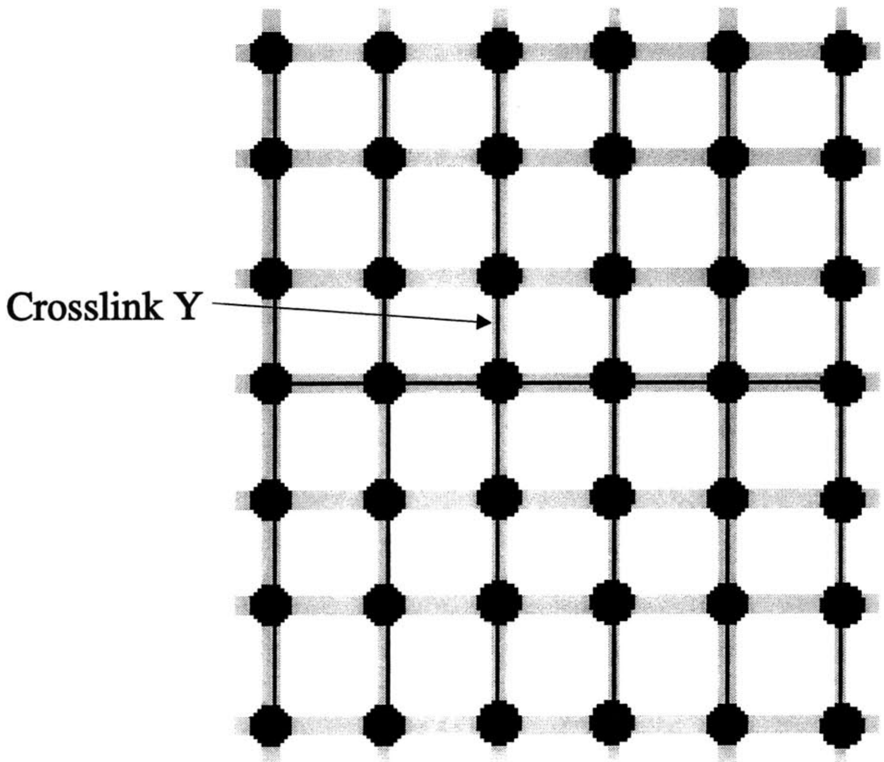

Figure 4-6: 1-inter-plane-crosslink mesh network

We then consider another case at the other end of the spectrum. In this case, only one inter-plane crosslink is remained between two planes. All the other inter-plane crosslinks are removed (Figure 4-6). Clearly the traffic will not be uniformly distributed among the inter-plane crosslinks or the intra-plane crosslinks. The intra-plane crosslinks nearest to the inter-plane crosslinks will have the most traffic among all the intra-plane crosslinks. The crosslinks which will experience the most traffic increase will be the inter-plane crosslinks because there is only one inter-plane crosslink remaining.

In this new constellation, E_1 is zero because no intra-plane crosslinks are removed. There are now only 2 intra-plane crosslinks per $2M+1$ nodes. As a result, we have

$$2 - E_0 = \frac{2}{2M+1} \tag{4.40}$$

$$E_0 = 2 - \frac{2}{2M+1} \tag{4.41}$$

Therefore, inequality (4.12) becomes:

$$(2 - \frac{2}{2M+1})\theta + 4 > \frac{2}{2M+1}\mu_0^{\alpha/4} + 2\mu_1^{\alpha/4} \tag{4.42}$$

Again, we consider a $(M+1)\times(2M+1)$ network, M even, for reason of symmetry. We consider the uniform traffic case, where each node sends out $\frac{T_u}{((M+1)(2M+1))^2}$ traffic to every node.

We first need to find μ_0, the ratio of increased capacity to original capacity in each inter-plane crosslink. Consider an s-t cut across two planes that cuts the nodes into two groups, group $\mathcal{S}$ and group $\mathcal{T}$ (Figure 4-4). In the original $(M+1)\times(2M+1)$ network with no deletion of crosslinks, there are $2M+1$ inter-plane crosslinks across that cut in one direction. Since all the crosslinks have equal capacity, and if we let C to be the capacity in one of these crosslinks, the amount of traffic going from $\mathcal{S}$ to $\mathcal{T}$ is $(2M+1)C$. In the new constellation, there is only one inter-plane crosslink across the cut, but the amount of traffic going across the cut remains the same; as a result, this lone crosslink needs to absorb all the traffic flowing from $\mathcal{S}$ to $\mathcal{T}$. Therefore, the new capacity required is $(2M+1)C$. Therefore,

$$\frac{\text{new amount of traffic}}{\text{original amount of traffic}} = \frac{(2M+1)C}{C} = 2M+1 \tag{4.43}$$

However, since we have discovered that, in the seamless case, half of the capacity is free in each inter-plane crosslink in the original constellation, μ_0 is thus equal to half of $2M + 1$:

$$\mu_0 = \frac{1}{2}(2M + 1) \tag{4.44}$$

We have found μ_0, and our next step is to find μ_1. We have noticed that the intra-plane crosslinks nearest to the inter-plane crosslinks will have the most traffic among all the intra-plane crosslinks. The traffic distribution of the intra-plane crosslinks will be the same in every plane because of the symmetry. For convenience's sake, we denote one of the intra-plane crosslinks nearest to the inter-plane crosslinks to be crosslink Y, as shown in figure (4-6). To find μ_1, we need to find the new required capacity in crosslink Y, since μ_1 needs to be the largest ratio of new to old capacities among all intra-plane crosslinks.

First of all, traffic in the crosslink can be divided into two groups: inter-plane traffic and intra-plane traffic. We will first find the amount of inter-plane traffic in crosslink Y, and then the amount of intra-plane traffic in crosslink Y.

To find the amount of inter-plane traffic, we notice that only the inter-plane traffic which terminates at the nodes above crosslink Y will use crosslink Y. There are M of these in-plane nodes. Furthermore, in a $(2M + 1) \times (M + 1)$ constellation, there are a total of $(2M + 1)M$ nodes that are not in the same plane. As a result, there are a total of $(2M + 1)M \times M = (2M + 1)M^2$ inter-plane paths that use crosslink Y. Each path has $\frac{T_u}{((M+1)(2M+1))^2}$ amount of traffic. Therefore, the amount of inter-plane traffic in crosslink Y, Y_{inter}, is equal to:

$$Y_{inter} = \frac{T_u}{((M + 1)(2M + 1))^2}(2M + 1)M^2 \tag{4.45}$$

We next calculate the amount of intra-plane traffic in crosslink Y. Intra-plane traffic follows a symmetric pattern as every node is sending the same amount of traffic to every in-plane node. Therefore, the amount of intra-plane traffic in each intra-plane crosslink is equal to

$$\frac{T_{path} NH}{CL} \tag{4.46}$$

where T_{path} is the amount of traffic in each path,

N is the number of nodes in a plane,

H is total number of hops required per node,

and CL is the number of intra-plane crosslinks.

There are $2M+1$ nodes in a plane. For a node to send to every in-plane node,

$$\text{Total number of hops required} = 2\sum_{i=1}^{M} i = (M+1)M \tag{4.47}$$

There are $2M+1$ nodes sending traffic to each other, and there are a total of $2(2M+1)$ intra-plane crosslinks.

Using the above information, equation (4.46) becomes:

$$\begin{aligned} & \frac{T_u}{((M+1)(2M+1))^2}\frac{(2M+1)(M+1)M}{2(2M+1)} \\ = & \frac{T_u}{((M+1)(2M+1))^2}\frac{(M+1)M}{2} \end{aligned} \tag{4.48}$$

The combined amount of traffic in crosslink Y is therefore equal to

$$\begin{aligned} & \frac{T_u}{((M+1)(2M+1))^2}(2M+1)M^2 + \frac{T_u}{((M+1)(2M+1))^2}\frac{(M+1)M}{2} \\ = & \frac{T_u}{((M+1)(2M+1))^2}[M^2(2M+1) + \frac{(M+1)M}{2}] \\ = & \frac{T_u}{((M+1)(2M+1))^2}(2M^3 + \frac{3}{2}M^2 + \frac{M}{2}) \end{aligned} \tag{4.49}$$

In previous section, we have found that under uniform traffic, the capacity of an intra-plane crosslink in the original square mesh to be $\frac{T_u}{8M'N'}(N' - \frac{1}{N'})$. With $M' = M+1$ and $N' = 2M+1$, this capacity becomes:

$$\begin{aligned} & \frac{T_u}{8(M+1)(2M+1)}((2M+1) - \frac{1}{2M+1}) \\ = & \frac{T_u}{((M+1)(2M+1))^2}\frac{1}{8}(M+1)(2M+1)((2M+1) - \frac{1}{2M+1}) \\ = & \frac{T_u}{((M+1)(2M+1))^2}\frac{1}{8}\frac{8M^4+20M^3+16M^2+4M}{2M+1} \end{aligned} \tag{4.50}$$

As a result,

$$\begin{aligned}
\mu_1 &= \frac{\text{new amount of traffic}}{\text{original amount of traffic}} \\
&= \frac{(2M^3 + \frac{3}{2}M^2 + \frac{M}{2})}{\frac{1}{8}\frac{8M^4+20M^3+16M^2+4M}{2M+1}} \\
&= \frac{8(2M^3 + \frac{3}{2}M^2 + \frac{M}{2})(2M+1)}{8M^4 + 20M^3 + 16M^2 + 4M} \\
&= \frac{32M^4 + 40M^3 + 20M^2 + 4M}{8M^4 + 20M^3 + 16M^2 + 4M} \\
&\approx 4, \quad \text{when M is large} \qquad (4.51)
\end{aligned}$$

Therefore, we have found μ_0 to be $\frac{2M+1}{2}$, and μ_1 to be 4. We substitute these values into inequality (4.42). The right hand side becomes

$$\frac{2}{2M+1}(\frac{2M+1}{2})^{\alpha/4} + 2\,(4^{\alpha/4}) = (\frac{2}{2M+1})^{1-\frac{\alpha}{4}} + 2^{\frac{\alpha}{2}+1} \qquad (4.52)$$

α is between 2 and 3. Therefore, the right hand side will definitely be greater than 4. The left hand side of the inequality (4.42) is $(2 - \frac{2}{2M+1})\theta + 4$. Depending on the values of θ, α, and M, the inequality may be satisfied. It is unclear whether there will actually be a cost reduction by using this constellation. As shown in this example, reducing too many crosslinks may not necessarily result in a reduction of total cost.

4.6 Summary

In this chapter, we have devised ways to reduce the overall cost of a system by reducing the number of crosslinks per node. We have shown that changing the original square mesh network into a 3-crosslink-per-node mesh network will definitely lead to a reduction of cost. However, further reducing the number of crosslinks does not necessarily equate to reducing cost, as shown in the 1-inter-plane-crosslink mesh model. This is because the crosslink capacity will be forced to increase because of the reduction of the number of crosslinks available, resulting in longer average paths.

Chapter 5

Hot Spot Traffic

5.1 Introduction

In this chapter we will extend the analysis to hot spot traffic. In previous chapters, we have only used uniform traffic, a very fundamental way to characterize traffic, as the basis of our analysis; however, this assumption of uniform traffic seems arbitrary, as it is obvious that uniform traffic will not occur all the time. As a result, hot spot traffic is introduced in this chapter in order to make our analysis more realistic. We define hot spot traffic as follows:

Hot Spot Traffic. Hot spot traffic is the traffic that originates from one place, which can be covered by one or more nodes, and transmits to every node in the mesh. T_h denotes the total amount of hot spot traffic that originates from a hot spot, and every node in the mesh receives $\frac{T_h}{NM}$ amount of traffic from the hot spot.

5.2 Lower and upper bounds of minimum crosslink capacities with uniform and hot spot traffic

In this chapter we consider the case where there is one volume of uniform traffic and one or multiple origin(s) of hot spot traffic in a square symmetric grid of size $M \times M$. A square symmetric grid of size $M \times M$ is the framework used in this chapter because it can be analyzed more easily but is still able to provide us with a general understanding of the interaction between hotspot and uniform traffic.

5.2.1 Functions for mesh grids

First of all, we define a set of functions for the mesh grid. We assume the grid to be of infinite size and therefore, we can ignore the effect of boundaries on those functions. We denote the node at the center of the grid to be the origin of the grid. $f(n)$ is the number of nodes at distance n from the origin (Figure 5-1).

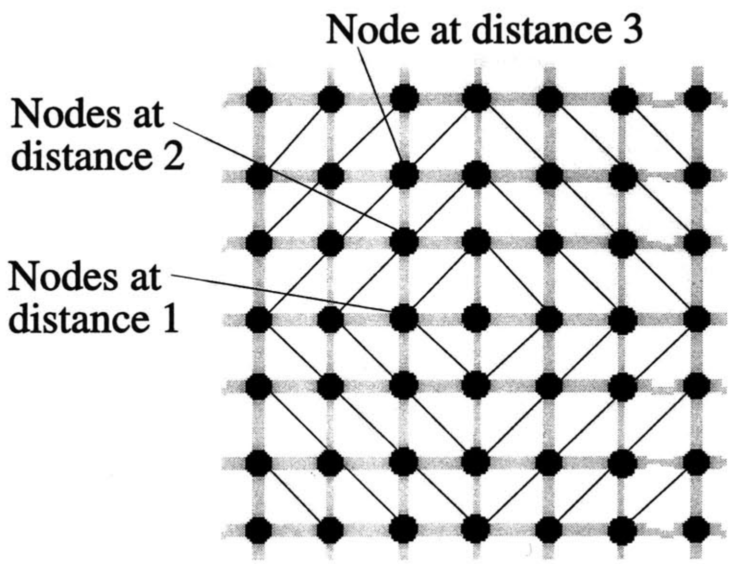

Figure 5-1: Definition of $f(n)$

By observation from a sample grid, we obtain the follow values for $f(n)$.

n	$f(n)$
0	1
1	4
2	8
3	12

Table 5.1: Values of $f(n)$

By simple observation, we obtain a closed form for $f(n)$.

$$f(n) = \begin{cases} 1 & : \quad n = 1 \\ 4n & : \quad n > 1 \end{cases} \tag{5.1}$$

Another function, $F(n)$, is the cumulative function of $f(n)$. It is the summation of the number of nodes at distances 1 to n from the origin.

$$\begin{aligned} F(n) &= \sum_{i=0}^{n} f(n) \\ &= 1 + \sum_{i=1}^{n} f(n) \\ &= 1 + \sum_{i=1}^{n} 4n = 1 + 4\left[\frac{n(1+n)}{2}\right] \\ &= 1 + 2n + 2n^2 \end{aligned} \tag{5.2}$$

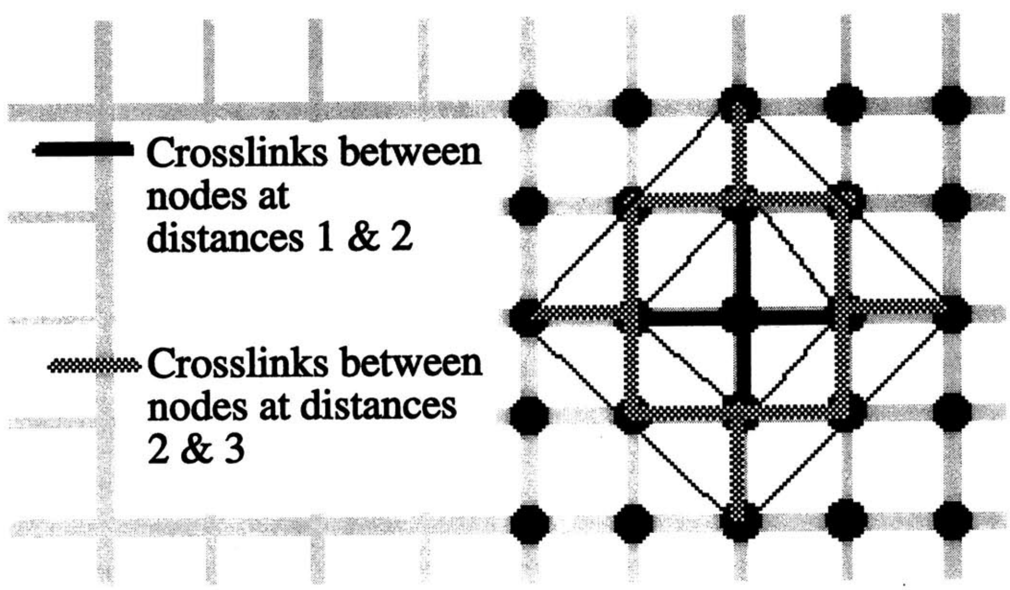

Figure 5-2: Definition of $g(n)$

We have another function, $g(n)$, for the number of crosslinks serving between nodes with distance n from the origin and nodes with distance $n+1$ from the origin (Figure 5-2).

Similarly, by observation from figure (5-2), we obtain the following values for $g(n)$:

n	g(n)
0	4×2
1	12×2
2	20×2

Table 5.2: Values of $g(n)$

In general, $g(n)$ has the following general form:

$$g(n) = 8 + 16n \tag{5.3}$$

With these functions set up, we could analyze hot spot and uniform traffic more easily.

5.2.2 Relationship between pass-through and non-pass-through traffic in uniform traffic; lower and upper bounds for crosslink capacity

With uniform traffic in a square symmetric grid of size $M \times M$, every node in the network can be regarded as identical. Since each node is identical, each crosslink surrounding a node must have equal amount of non-pass-through and pass-through traffic. As defined before, each node sends out T_u/M^2 of traffic to every node. There are four outgoing crosslinks for each node; as a result, the amount of traffic in a crosslink that originates from the parent node is:

$$\frac{1}{4}\frac{T_u}{M^2}\frac{M^2-1}{M^2} \tag{5.4}$$

$1/M^2$ of the originating traffic will not be passed to the crosslinks because this amount of traffic will be released down back to the earth because the destination is in the same footprint as the origin. Because of the uniform assumption, by symmetry, there will the same amount of traffic coming into the node.

Derived from chapter 3, the capacity for each crosslink in the seamless and seamed cases is $T_u/8M$. As a result, the fraction of non-pass-through traffic in each crosslink is:

$$
\begin{aligned}
\frac{\frac{1}{4}\frac{T_u}{M^2}\frac{M^2-1}{M^2}}{\frac{T_u}{8M}} &= \frac{8M(M^2-1)}{4M^4} \\
&= \frac{2(M^2-1)}{M^3} \qquad (5.5)
\end{aligned}
$$

Since non-pass-through traffic cannot be re-routed, for a system that serves T_u uniform traffic and T_h hot spot traffic, a lower bound of the minimum required capacity of any crosslink is one that is able to hold at least all the non-pass-through traffic and the hot spot traffic in its region. Because we have assumed all the crosslinks to have equal capacity, the lower bound is constrained by the required capacity of the crosslinks which experience the most amount of traffic. The theorem below shows that the crosslinks originating from the center of the hot spot experience the most amount of traffic.

Theorem 5.1 (Center Crosslinks) *In a square mesh grid with a stream of hot spot traffic and a stream of uniform traffic, assuming no re-routing of traffic, the crosslinks which experience the most amount of traffic are the ones which originates from the center of the hot spot, which is also known as the origin.*

Proof:

There are two types of traffic—uniform traffic and hot spot traffic. As discussed above, assuming uniform traffic is not re-routed, all crosslinks will have an equal amount of uniform traffic, which is $\frac{T_u}{8M}$. However, each crosslink will receive a different amount of hot spot traffic, depending on the location of the crosslink relative to the hot spot center. Let's denote $T_h'(n)$ to be the amount of hot spot traffic that is in a crosslink whose origin is n hops away from the center. Because the square mesh grid is symmetric, the amount of hot spot traffic in the network decreases gradually as it moves away from the center. When a stream of hot spot traffic has arrived at a node, that node will absorb T_h/M^2 amount of traffic and pass the rest to the outer crosslinks. Because of the symmetry involved, all the crosslinks with the same distance away from the center will have the same amount of traffic. Hence, $T_h'(n)$ can be expressed as follows:

$$T_h'(n) = T_h[1 - \frac{\#(\text{nodes served})}{\#(nodes)}]\frac{1}{\#(crosslinks_n)} \tag{5.6}$$

where

$\#$(nodes served) = number of nodes that have already received the hot spot traffic,

$\#(nodes)$ = total number of nodes, and

$\#(crosslinks_n)$ = number of crosslinks that are at distance n from the center of the hot spot and have the same direction flow as the hot spot traffic. Using the functions previously defined in the section (5.2.1), $T_h'(n)$ becomes:

$$T_h'(n) = T_h\ [1 - \frac{F(n)}{M^2}]\ \frac{1}{g(n)/2} \tag{5.7}$$

From equation (5.2), $F(n)$ equals to $1 + 2n + 2n^2$, and from equation (5.3), $g(n)$ equals to $8 + 16n$. Therefore, $F(n)$ has its minimum at $n = 1$, and $1 - \frac{F(n)}{M^2}$ has its maximum at $n = 1$. Similarly, the minimum of $g(n)$ occurs at $n = 1$, and therefore $\frac{1}{g(n)/2}$ is maximum for $n = 1$. Therefore, $T_h'(n)$ has its maximum when $n = 1$. This means that the crosslinks that originates from the center of the hot spot receive the maximum amount of hot spot traffic. Subsequently, they also receive the maximum amount of traffic, when uniform traffic is included.

Because hot spot traffic highly concentrates near the origin, the crosslinks around the origin will require the most capacity. The capacity required for hot spot traffic is thus constrained by the capacity required at the center crosslinks. The lower bound for the required capacity of the crosslinks is thus:

$$\frac{T_u}{8M}\frac{2(M^2-1)}{M^3}+\frac{T_h}{4} \quad or \quad \frac{1}{4}\frac{T_u}{M^2}\frac{M^2-1}{M^2}+\frac{T_h}{4} \tag{5.8}$$

If the crosslinks are set at this lower bound capacity, the crosslinks at the center can only accept the non-pass-through traffic and all the pass-through traffic will then need to be pushed to outer less crowded crosslinks. Similarly for multiple hot spots, a lower bound for the required crosslink capacity would be one that allows crosslinks to hold all the non-pass-through traffic for all hot spots, plus the non-pass-through traffic from uniform traffic. Thus, the lower bound for crosslink capacity for multiple hot spots is:

$$\frac{T_u}{8M}\frac{2(M^2-1)}{M^3}+\max(\text{combined hot spot traffic}) \tag{5.9}$$

where max(combined hot spot traffic) is the maximum of combined hot spot traffic from different streams in a crosslink, among all crosslinks.

A sensible upper bound for the minimum capacity is one that just provides enough bandwidth such that all the traffic will not need to be re-routed because of congestion. This would mean that the crosslinks must need to provide enough capacity for all the non-pass-through and pass-through traffic for both uniform and hot spot traffic. This upper bound is therefore:

$$\frac{T_u}{8M}+\frac{T_h}{4} \tag{5.10}$$

However, for multiple hot spot streams, there is a constraining case in which every node has a stream of hot spot traffic of equal size. In this case, all the hot spot traffic has essentially become another instance of uniform traffic. The upper bound of minimum capacity must be able to support these combined streams of hot spot traffic. In this case, M^2 streams of hot spot traffic combine together to form an additional stream of uniform traffic, T_u':

$$T_u' = T_h M^2 \tag{5.11}$$

For uniform traffic, the crosslink capacity needs to be $\frac{T_u}{8M}$. As a result, the capacity required for these streams of hot spot traffic is:

$$\frac{T_u'}{8M} = \frac{T_h M^2}{8M} = \frac{T_h M}{8} \tag{5.12}$$

As a result, the upper bound for minimum capacity of crosslinks in the multiple hot spot case is:

$$\frac{T_u}{8M} + \frac{M}{8} max(T_h) \tag{5.13}$$

where $max(T_h)$ is the traffic from the busiest hot spot.

Below is a table to summarize the upper and lower bounds of minimum crosslink capacity for the two cases:

	Lower Bound	Upper Bound
Single hot spot and uniform traffic	$\frac{1}{4}\frac{T_u}{M^2}\frac{M^2-1}{M^2} + \frac{T_h}{4}$	$\frac{T_u}{8M} + \frac{T_h}{4}$
Multiple hot spots and uniform traffic	$\frac{T_u}{8M}\frac{2(M^2-1)}{M^3} + max(comb.\ hot\ spot)$	$\frac{T_u}{8M} + \frac{M}{8} max(T_h)$

Table 5.3: Upper and lower bounds of crosslink capacity

The next sections discuss about how much uniform and hot spot traffic one system can serve given the capacity of the crosslinks. We will first consider a grid of infinite size as the initial case, and will go on to explore grids of arbitrary sizes. However, before all these, we should consider how re-routing of displaced traffic should be done to enhance the performance of the network.

5.3 Re-routing of displaced pass-through traffic

If we set the capacity of the crosslinks to be smaller than the upper bound described in equation (5.10), then there will be at least some pass-through traffic that needs to be re-routed. When the traffic is re-routed, it will then need to follow a non-optimized route which will result in an increase in hop counts in each route. As a result, the overall capacity required will increase. The number of hop counts increased depends on how the traffic is re-routed. Nevertheless, there are two general sensible objectives that we should keep in mind:

1. In order to minimize the utilization of crosslink capacity, when a traffic stream needs to be re-routed, it should use a path with the minimum number of increased hop counts among all the alternate paths. This optimization of crosslink capacity will lead the network to support as much traffic as possible.

2. Since uniform traffic is symmetric for every node, and hot spot traffic is symmetric across the x and y axes, the re-routed path should also be symmetric across the x and y axes; otherwise there will be unused capacity that got trapped by fully occupied crosslinks. We will show that this is a good configuration to minimize the number of hop counts for all paths.

5.3.1 Proof of the first objective

The first objective is simply a re-statement that a local optimization of hop counts for each path will lead to a global optimization of hop counts for the entire network. Suppose there are K paths in the network, and x_i is the number of hop counts for path i. The first objective is essentially:

$$\min \sum_{i=1}^{k} x_i \equiv \sum_{i=1}^{k} \min x_i \tag{5.14}$$

subject to crosslink capacity constraints.

In order to prove this statement, we first need to show that path lengths, as a cost metric, exhibit the Markovian property.

Definition 1 (Markovian Property) *[12, p. 22]*

Consider a cost function

$J = J(x(0), x(1), ...x(N); u(0), u(1), ...u(N))$

where $x(i)$ *is the i-th state, and*

$u(i)$ *is the decision chosen at stage* i*,*

we say that J *has the Markovian property if, given the two decision sequences,* $\mathcal{U}$ *and* $\overline{\mathcal{U}}$*, with*

$$\mathcal{U} = [u(0), u(1), ..., u(k), u(k+1), ..., u(N)], \quad \textit{and} \tag{5.15}$$

$$\overline{\mathcal{U}} = [\overline{u}(0), \overline{u}(1), ..., \overline{u}(k), \overline{u}(k+1), ..., \overline{u}(N)] \tag{5.16}$$

then, whenever

$$x(k) = \overline{x}(k), \tag{5.17}$$

$$u(i) = \overline{u}(i), i = 0, 1, ..., k-1, \tag{5.18}$$

and

$$J(x(k), x(k+1), ..., x(N); \mathcal{U}) \leq J(x(k), \overline{x}(k+1), ..., \overline{x}(N); \overline{\mathcal{U}}), \tag{5.19}$$

then

$$J(x(0), x(1), ..., x(N); \mathcal{U}) \leq J(x(0), \overline{x}(1), ..., \overline{x}(N); \overline{\mathcal{U}}) \tag{5.20}$$

Let $x_p(a_1, a_3)$ be the number of hop counts from node a_1 to node a_3 using path p, and $x_{p'}(a_1, a_3)$ be the number of hop counts from node a_1 to node a_3 using path p'. Suppose that to go from node a_1 to node a_3, node a_2 must be visited. Therefore, we have

$$x_p(a_1, a_3) = x_p(a_1, a_2) + x_p(a_2, a_3), \quad \text{and} \tag{5.21}$$

$$x_{p'}(a_1, a_3) = x_{p'}(a_1, a_2) + x_{p'}(a_2, a_3) \tag{5.22}$$

If $x_p(a_1, a_2) = x_{p'}(a_1, a_2)$ and $x_p(a_2, a_3) > x_{p'}(a_2, a_3)$, then by the additive nature of hop counts (equation (5.22)), then $x_p(a_1, a_3) > x_{p'}(a_1, a_3)$.

In simple English, this means that if the path lengths from node a_1 to node a_2 using paths p and p' are the same, and the path length from a_2 to a_3 using path p' is shorter than the path length using path p, then the overall path length from a_1 to a_3 is shorter using path p'. Therefore, path length as cost metric exhibits the Markovian property.

Markovian property has the property that current decisions only depend on the current state, not past decisions and states. We then use this Markovian property to examine the re-routing of a path. Consider a simple network in figure (5-3):

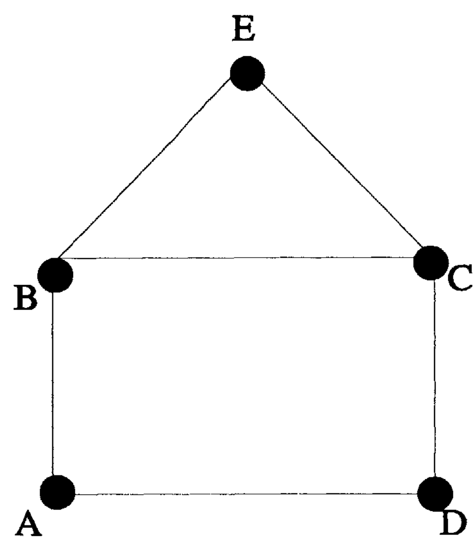

Figure 5-3: A Simple Network

We would like to find a path with minimum number of hop counts from A to D. When the link (A,D) is not full, the optimum path is clearly A-D. However, we now consider the case when the link (A,D) is full. The optimum path cannot be used and therefore we must use the path (A,B) and the path (C,D). The problem then becomes

$$\min_{poss.path} x(A, D) \equiv x(A, B) + \min_{poss.path} x(B, C) + x(C, D) \tag{5.23}$$

subject to crosslink capacity constraints.

Because of the Markovian nature of path length, this problem has become simply to find the optimum path between B and C, subject to the crosslink capacity constraints.

Theorem 5.2 (Bellman's Principle of Optimality) *[12, p. 55] An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state from the state resulting from the first decision.*

If we consider every path as one stage of the entire network decision process, then, according to this principle, in a global optimum solution, every path should itself be an optimum solution, subject to the capacity constraints of the links in the network. Therefore, a local optimization of path lengths for each path will result in the global optimization of path lengths for the whole network.

Therefore, if we had more than one option for the alternate path, we would choose the one with the minimum number of hop counts such that there will be more excess capacity available to other traffic. We would want to conserve as much capacity as possible. The consequence of this is that we will try to start with paths that has one more hop count than the original path, and if this is not possible, we would search for paths with 2 more hop counts and continue doing that until we have found a valid path. Graphically, we start with crosslinks that is 1 distance away from the center and increase the distance unit by unit. In a sense this is a radial search in which we extend the radius gradually until we have found a valid path.

5.3.2 Proof of the second objective

The second objective deals with the symmetry of the traffic pattern. For hot spot traffic, the 8 ($g(0)$) crosslinks (both inward and outward) around the origin will be the most congested and $g(1)$ crosslinks between nodes at distances 1 and 2 from the center will be the second most congested and so on. There is a symmetry in capacity usage across the two major axes. As a consequence from the first objective, we would want to divert traffic to the shortest alternate paths. We thus should occupy the crosslinks that are closer to the origin first than those farther away. Furthermore, we should try, as much as we can, to design a re-routed path such that its crosslinks should have equal amount of excess capacity. This is to ensure that all the crosslinks are used to their fullest capacity. If the paths are not designed this way, there will be some unused capacity that will be trapped by crosslinks which have no extra capacity. Since all surrounding crosslinks have been used to their fullest capacity, the trapped excess capacity cannot be used. To further elaborate this idea, consider the two paths as shown in figure (5-4).

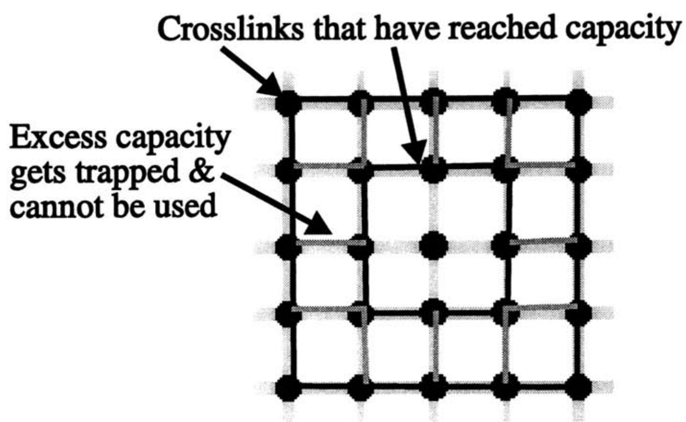

Figure 5-4: Trapped excess capacity

As shown in the figure, there are some crosslinks between the two paths which have excess capacity. However, no traffic can utilize this excess amount of capacity because all the surrounding crosslinks have reached their capacity already. As a result, some bandwidth is wasted and this configuration is definitely not optimal. The other scheme should give better performance since no bandwidth in the center region will be wasted.

5.4 Infinite Mesh

An infinite mesh is essentially a two dimensional square grid which extends indefinitely in both x and y directions. This is just a theoretical, unrealistic scenario because no such constellations exist in real world. However, since there are no corners or end points in this theoretical case, this simplifies the analysis a lot.

Theorem 5.3 (Infinite Mesh, Single Hot Spot) *In an infinite mesh that supports one stream of hot spot traffic and one stream of uniform traffic, the minimum capacity required of the crosslinks is the maximum of* $\frac{T_u}{8M} + \delta$ *and the amount of non-pass-through traffic in the most congested part of the network, which is the center.* δ *is just an arbitrary positive small number. The minimum capacity is*

$$max\{\frac{T_u}{8M} + \delta, \frac{T_u}{8M}\frac{2(M^2-1)}{M^3} + \frac{T_h}{4}\} \tag{5.24}$$

When a network supports only uniform traffic, all crosslinks will have identical distribution of traffic. Each crosslink will have an equal amount of pass-through and non-pass-through traffic. Previously, we have established that given the total amount of uniform traffic to be T_u, each crosslink will be required to serve $\frac{T_u}{8M}$ amount of uniform traffic. This is also the minimum capacity requirement for each crosslink under uniform traffic condition.

When there is an additional stream of hot spot traffic added to the network, it is clearly that the minimum capacity requirement has to be greater than the requirement solely for uniform traffic. This idea is incorporated into the first part of the theorem, which says that the capacity of the crosslink has to be at least greater than the required crosslink capacity requirement for uniform traffic. Nevertheless, this is not the absolute minimum. Another criterion for the required capacity is that it has to at least be able to support all the non-pass-through traffic at the most congested part of the network, which is the origins of hot spot traffic. The network does not need to support the overall sum of uniform traffic and hot spot traffic, because the non pass through traffic can be re-routed to paths with less congested crosslinks. We will prove next that the less crowded crosslinks are able to absorb all the displaced re-routed pass-through traffic and the minimum capacity required is thus the maximum of $\frac{T_u}{8M} + \delta$ and the amount of non-pass-through traffic in the most congested part of the network.

Hot spot traffic is most concentrated at the node of origin. The amount of hot spot traffic will decay gradually as the traffic has spread to more nodes. The amount of hot spot in a crosslink that is n units of distance from the center is:

$$T_h \frac{M^2 - F(n)}{M^2} \frac{g(n-1)}{g(n)} \tag{5.25}$$

which is equal to:

$$T_h \frac{M^2 - 2(n^2+n) - 1}{M^2} \frac{8 + 16(n-1)}{8 + 16n} \tag{5.26}$$

Take a close look at the second fraction. The maximum value of $F(n)$ for any arbitrary grid is M^2, since $F(n)$ is the aggregate sum of the number of nodes 0 to n distances away from the center, and there are only M^2 nodes in the grid. As a result, this term will get very close to zero if n is large. For an infinite grid, we can assume M^2 to be a very large number, and the amount of hot spot traffic, when it is very far away from the center of origin, is very close to zero. Therefore in crosslinks that are far away from the center, the traffic inside is mostly uniform traffic and hot spot traffic is negligible. As a result, there will actually be extra capacity inside those crosslinks because the capacity is sized to be at least greater than the required capacity for uniform traffic.

Since the crosslinks far away is definitely able to support all the uniform traffic, it can handle all the non-pass-through and pass-through uniform traffic. As a result, there will not be any overflowed traffic from those crosslinks. The overflowed traffic must therefore come from crosslinks in the center. As a result, the crosslinks that produce overflowed traffic are surrounded by crosslinks that have excess capacity. The number of crosslinks that have overflowed traffic is thus finite, and this implies that the amount of overflowed traffic is also finite.

On the other hand, for an infinite mesh, we have an infinite number of crosslinks. There will be an infinite number of crosslinks that will have at least δ excess capacity. This infinite excess capacity can therefore be used to absorb the re-routed overflowed pass-through traffic. Since the amount of overflowed traffic is finite, and the number of hops is only doubled each time when it goes one unit of distance away from the center, an infinite mesh can definitely absorb the displaced pass-through traffic.

The minimal capacity for crosslinks is the bigger of the two values, as defined in equation (5.24). For the two values, it would be nice to find a condition to determine which one is actually the minimal capacity. We would choose the capacity requirement to be the amount of all non-pass-through traffic if it is greater than the overall uniform traffic:

$$\begin{aligned}
\frac{T_u}{8M}\frac{2(M^2-1)}{M^3}+\frac{T_h}{4} &> \frac{T_u}{8M} \\
\frac{T_h}{4} &> \frac{T_u}{8M}[1-\frac{2(M^2-1)}{M^3}] \\
\frac{T_h}{T_u} &> \frac{1}{2M}[1-\frac{2(M^2-1)}{M^3}]
\end{aligned} \tag{5.27}$$

Therefore, we have the following modified theorem.

Theorem 5.4 (Modified Infinite Mesh, Single Hot Spot) *In an infinite mesh that supports one stream of hot spot traffic and one stream of uniform traffic, if T_h is greater than $T_u\frac{1}{2M}[1-\frac{2(M^2-1)}{M^3}]$, then the minimum capacity required for each crosslink is*

$$\frac{T_u}{8M}\frac{2(M^2-1)}{M^3}+\frac{T_h}{4} \tag{5.28}$$

otherwise it is:

$$\frac{T_u}{8M}+\delta \tag{5.29}$$

We can extend this theorem for multiple hot spot traffic. If there are k hot spots in the network, the amount of overflowed traffic is still finite, since the crosslinks that produce overflowed traffic are still surrounded by crosslinks that have excess capacity. There will still be an infinite number of crosslinks that have at least δ excess capacity. This infinite amount of excess capacity can therefore be used to absorb the re-routed overflowed pass-through traffic. Since the amount of overflowed traffic is finite, and the number of hops is only doubled each time when it gets re-routed to a path that is one unit of distance further away from the center, an infinite mesh can definitely absorb the displaced pass-through traffic. Therefore, if all the crosslinks have enough capacity to hold the uniform traffic (i.e. it has to be greater than $\frac{T_u}{8M}$), and also have the capacity to hold all the non-pass-through traffic in every part of the network, then an infinite mesh can support the uniform traffic and multiple hot spot streams. This gives us the next theorem.

Theorem 5.5 (Infinite Mesh, Multiple Hot Spots) *If each crosslink in every part of the network has the capacity to handle all the non-pass-through traffic, and it has some excess capacity when handling one stream of uniform traffic, then, regardless of the number of hot spots in the network, the network will be able to support all the traffic because all the pass-through can definitely be re-routed to links at outer perimeter.*

5.5 Grid of Fixed Dimensions

In previous sections, we have considered an imaginary infinite grid in which we can ignore boundary conditions in the calculation. However, this type of grid does not exist and therefore we should examine a more realistic case with a grid of fixed dimensions. We first examine few basic properties of a grid of fixed size $M \times M$.

For a grid with dimension $M \times N$ (M is odd number), the functions described in the previous section, $f(n)$, $F(n)$ and $g(n)$, will all have the same values as the infinite case from $n = 1$ to $(M-1)/2$. However as n becomes greater than $(M-1)/2$, the values will be smaller than those in the infinite case because some of the nodes are cut off at the boundaries. $f(n)$ has its maximal at $n = (M-1)/2$ and $(M-1)/2+1$. Furthermore, $f(n)$ will also be symmetric across $n = 1/2M/2$, i.e. $f(M-1)$ will be equal to $f(1)$, $f(M-2)$ will be equal to $f(2)$, and so on. Therefore, $f(n)$ for fixed size grid will typically look like a plateau function (See figure (5-5)).

Furthermore, we can define a s-t cut as a subset $\mathcal{S}$ of the set of nodes $\mathcal{N}$ such that $s \in \mathcal{S}$ and $t \notin \mathcal{S}$. Let $\mathcal{C}$ be the set of crosslinks connecting between the two sets. In every crosslink, there are the pass-through traffic and the non-pass-through traffic.

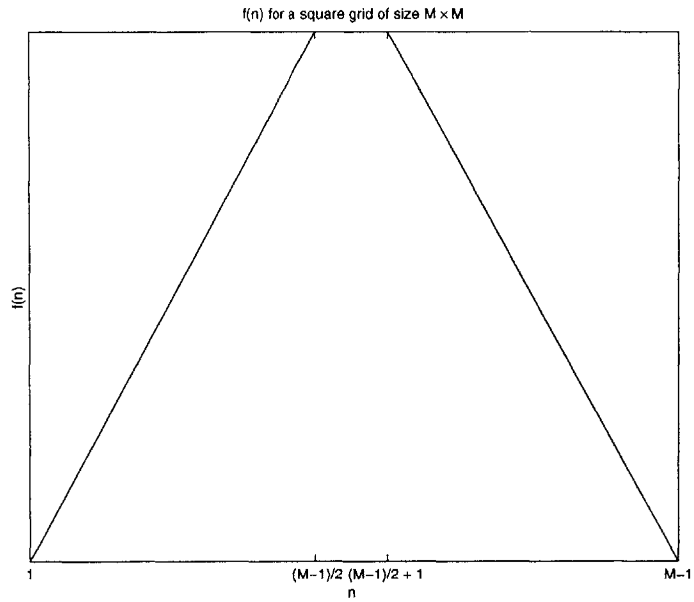

Figure 5-5: $f(n)$ for a square mesh of size $M \times M$

For uniform traffic, the non-pass-through traffic, T_{npt}, is equal to

$$\frac{2\frac{T_u}{M^2}(\text{number of nodes in } \mathcal{S})(\text{number of nodes in } \mathcal{T})}{(\text{number of crosslinks in } \mathcal{C})} \tag{5.30}$$

There is a factor of 2 because there are incoming traffic going into and outgoing traffic going from set $\mathcal{S}$. Pass-through uniform traffic, T_{pt}, is thus:

$$\frac{T_u}{8M} - T_{npt} \tag{5.31}$$

If we define $\mathcal{S}$ to be the set of nodes with distance less than or equal to n from the origin, $g(n)$ will then become the number of crosslinks in $\mathcal{C}$, $F(n)$ will become the number of nodes in $\mathcal{S}$, and T_{npt} will become:

$$\frac{2\frac{T_u}{M^2}F(n)(M^2 - F(n))}{g(n)} \tag{5.32}$$

T_{pt} is then:

$$\frac{T_u}{8M} - \frac{2\frac{T_u}{M^2}F(n)(M^2 - F(n))}{g(n)} \tag{5.33}$$

Lastly, similar to previous argument, the amount of hot spot traffic in each crosslink, $T'_{hot}(n)$, in $\mathcal{C}$ is

$$\frac{T_{hot}(1 - \frac{F(n)}{M^2})}{g(n)/2} \tag{5.34}$$

We will now introduce a systematic and iterative method to find a close lower bound to the possible hot spot and uniform capacities of the grid.

As discussed before in the previous section, when the load of the crosslinks at some distance n from the center has exceeded its capacity, the traffic will need to be diverted to the outer perimeter with additional hop counts. We have shown that if we choose a path that is one more hop away from the center, the number of hops will be doubled. As a result, we have the following difference equation:

$$\begin{aligned}&(\text{overflow capacity})_{i+1} = \\ &\quad max\{2(\text{overflow capacity})_i - (\text{excess capacity})_{i+1}, 0\}\end{aligned} \tag{5.35}$$

where

(overflow capacity)$_n$ (OC_n) is the amount of traffic that needs to be diverted in crosslinks that is n distance from the origin;

(excess capacity)$_n$ (EC_n) is the amount of unused capacity in crosslinks that are n units of distance away from the origin; $(EC)_n$ is equal to

$$g(n) \times (\text{capacity of crosslink (C)} - T'_{hot_n} - T_{npt_n} - T_{pt_n}) \tag{5.36}$$

Negative $(EC)_n$ means that current demand of capacity cannot be satisfied by the crosslinks and needs to be diverted to outer levels.

In any type of mesh networks, if the amount of overflowed traffic is equal to zero when it reaches any of the boundaries, the network will then be able to handle this type of traffic demand. Therefore, for a mesh of size $M \times M$ (M odd), if $(OC)_{(M-1)/2+1}$ is equal to zero, then all the overflow traffic can be absorbed in outer crosslinks and the network is thus able to handle this particular type of traffic. Using the difference equation, equation (5.35), starting with $EC_{(M-1)/2+1}$, and continuing down to $(OC)_1$, we would get the following:

$$\begin{aligned} 0 \; >= \; & 2(\frac{M-1}{2})OC_1 - 2(\frac{M-1}{2} - 1)EC_2 \\ & -2(\frac{M-1}{2} - 2)EC_3 - \cdots \\ & -2EC_{\frac{M-1}{2}} - EC_{\frac{M-1}{2}+1} \end{aligned} \tag{5.37}$$

or,

$$\begin{aligned} 2(\frac{M-1}{2})OC_1 \; <= \; & 2(\frac{M-1}{2} - 1)EC_2 \\ & +2(\frac{M-1}{2} - 2)EC_3 + \cdots \\ & +2EC_{\frac{M-1}{2}} + EC_{\frac{M-1}{2}+1} \end{aligned} \tag{5.38}$$

Using this scheme, the crosslinks at the center are filled up first, and then traffic gets spread out radially until it has reached one of the boundaries. As a result, there is actually excess capacity at the outer edges of the mesh which we have not used. As a result, it would be ingenious if we could use those unused capacity to increase the amount of traffic it can handle.

If the boundaries are real boundaries, then there is nothing we could do since the filled crosslinks have spread to the edges and they have partitioned the set of non-filled crosslinks into two sub-groups, and these two sub-groups cannot transfer traffic to each other anymore since the traffic must go through the section with filled crosslinks, which has no more excess capacity. Nevertheless, the boundaries in satellite mesh network are not rigid boundaries. The boundaries are in fact interconnected since the network wraps around the globe to form a sphere. Therefore, those unused capacity at the outer edges can definitely be used to support more traffic since traffic can go around the boundaries to the other side of the mesh grid. As a result, what we have found is actually a lower bound of the capacity that can be served by that particular mesh.

We could therefore find a tighter lower bound if we include the crosslinks that are not utilized in our previous scheme for re-routing of overflowed traffic. Note that there are more excess capacity in crosslinks as we move further away from the center. Therefore, if we want to move traffic from one part of the mesh to the other side through the boundaries, the bottleneck will be the starting crosslinks since they are the ones with the least excess capacity. Once the starting crosslinks is able admit the traffic, the traffic will be guaranteed to arrive at the other side of the network mesh through the boundaries without experiencing any congestion problems. As a result, if the excess capacity at the next level is greater than the overflowed capacity at the boundary, then the network will be able to support that level of traffic. We have therefore found a tighter lower bound. The new equation, modified from (5.37), has become:

$$\begin{aligned} EC_{\frac{M-1}{2}+2} \; >= \; & 2(\frac{M-1}{2})OC_1 - 2(\frac{M-1}{2} - 1)EC_2 \\ & -2(\frac{M-1}{2} - 2)EC_3 - \cdots \\ & -2EC_{\frac{M-1}{2}} - EC_{\frac{M-1}{2}+1} \end{aligned} \tag{5.39}$$

or,

$$\begin{aligned} 2(\frac{M-1}{2})OC_1 \; <= \; & 2(\frac{M-1}{2} - 1)EC_2 \\ & +2(\frac{M-1}{2} - 2)EC_3 + \cdots \\ & +2EC_{\frac{M-1}{2}} + EC_{\frac{M-1}{2}+1} + EC_{\frac{M-1}{2}+2} \end{aligned} \tag{5.40}$$

Therefore, for a mesh size $M \times M$, in order to determine how much uniform and hot spot traffic it can hold, we can do the following:

1. Find the values of the functions, $f(i), F(i)$, and $g(i)$, for all i's ranging from 1 to $\frac{M-1}{2} + 2$.

2. Determine $T'_{hot_i} - T_{npt_i} - T_{pt_i}$ for all i's from 1 to $\frac{M-1}{2} + 2$.

3. With those values, obtain EC_i and OC_i.

4. Lastly substitute the values into equation (5.40) and determine if the condition is satisfied or not. If the condition is satisfied, it means the network can hold that particular amount of hot spot and uniform traffic.

The above method will work even if there are multiple hot spot traffic streams. The tricky part is to calculate T'_{hot_i} for different hot spot traffic streams. Nevertheless, we can deal with the hot spot traffic stream one by one. For each hot spot traffic stream, we should just translate the center of hot spot to be the origin of the grid, and then find T'_{hot} for all the nodes. We repeat this step for all the hot spot traffic streams and add all the $T_{hot'}$ from different streams together to form the overall T'_{hot}. With this overall T'_{hot}, we are able to use the above method for multiple hot spot traffic without much modification.

5.6 Summary

In this chapter we have incorporated hotspot traffic into our analysis. We first have found an upper bound and a lower bound for the minimum capacity of a crosslink to serve a stream of uniform traffic T_u and stream(s) of hotspot traffic T_h. We then explored ways to displace non-pass-through traffic when links become congested. We have also considered a theoretical infinity mesh with uniform and hotspot traffic. Lastly, we looked at a grid of size $M \times M$ and find a condition to check whether such a network can support particular amount of hotspot and uniform traffic, T_h and T_u.

Chapter 6

Interaction with terrestrial fiber networks

6.1 Introduction

In previous chapters, we have explored the concept of serving both hotspot and uniform traffic in satellite communication systems. We have found out that if the system serves only one arbitrary stream of uniform traffic, T_u, the capacity of the crosslinks can be set to $T_u/8M$ and there will not be any overflowed traffic. We have also derived the necessary conditions for crosslink capacity so that the system will be able to support an arbitrary amount of hot spot and uniform traffic. However, we have always assumed the static case: in real world there can be a sudden surge of traffic which exceeds the designed capacity of the system. As a result, congestion will occur and the satellite system will not be able to support some of the surged traffic.

Nevertheless, in next generation Internet (NGI) which is conceived to be a heterogeneous global network, the satellite communication system we have been exploring constitutes only one section of the entire network. There will also be a terrestrial fiber network which forms the core of the network. As a result, if a satellite network has reached its maximal capacity and there is some overflowed traffic, the overflowed traffic does not need to be dropped. In fact, the terrestrial fiber network can be used as a buffer to alleviate congestion in the satellite network when congestion occurs.

Connecting the satellite network and the terrestrial networks are the uplinks and downlinks. Uplinks are links for transporting data from ground to space, while downlinks are for transporting data in the reverse direction. As a result, there are three major elements of a global heterogeneous network: inter-satellite crosslinks; uplinks, and downlinks; and lastly, terrestrial networks. In the following section we will first discuss about the general assumptions that can be made on a global heterogeneous network. Using these assumptions, a simulation in MATLAB will be built in the next chapter. We will then discuss about the cost metrics and the necessary relationships between the different components of the global heterogeneous network.

6.2 Assumptions on link capacities

The following assumptions about link capacities can be made:

1. The capacity of the satellite network is significantly smaller than that of terrestrial networks.

2. The capacities of the inter-satellite links and the terrestrial fiber links are constant and are not affected by weather.

3. The only links with time-varying capacities are the uplinks and downlinks of the satellite network, whose capacities can be described in step functions. The uplinks and downlinks have smaller capacities than the inter-satellite links or the terrestrial fiber links.

6.2.1 Overall capacities of satellite and terrestrial networks

The overall capacity of the satellite network is significantly smaller than the capacity of the terrestrial network:

$$C_{satt} \ll C_{terr} \tag{6.1}$$

This assumption is valid with current existing condition as the bandwidth of the satellite network is in the Mbps range, while that of terrestrial networks is in the Gbps range. With this assumption, it could be assumed that when the satellite network experiences congestion, the terrestrial network could always absorb most packets from the space network without affecting the traffic in terrestrial networks greatly.

6.2.2 Capacities of the links inside the satellite and terrestrial networks

The capacities of the fiber links in terrestrial network and the inter-satellite links (ISLs) are static. They are not affected by weather or any other time variables. This is true since all types of fiber cables have constant data rates under some maximal distances. Furthermore, weather only affects conditions in the Earth's atmosphere. The ISLs are well above the atmosphere and therefore are not affected by the turbulence of weather conditions on the Earth's surface.

$$C_{ISL} = constant \tag{6.2}$$

$$C_{Fiber} = constant \tag{6.3}$$

6.2.3 Capacities of uplinks and downlinks

The uplinks and downlinks of the satellite network are the only links with varying capacities. The capacities of the links depend on the weather condition at the particular location where the uplinks and downlinks are situated.

Step functions can be used to describe the capacities of the uplinks and downlinks. Different modulation rates and coding rates will be used in the links given various weather conditions. Given a good weather condition, a better channel with less noise will be available. As a result, with constant bit error rate (BER), we could increase the modulation and coding rates to achieve higher capacities. Since modulation and coding rates are discrete in nature, the capacities of the uplinks and downlinks vary in a step wise manner. Jihwan Choi has developed a Markov atmospheric model and

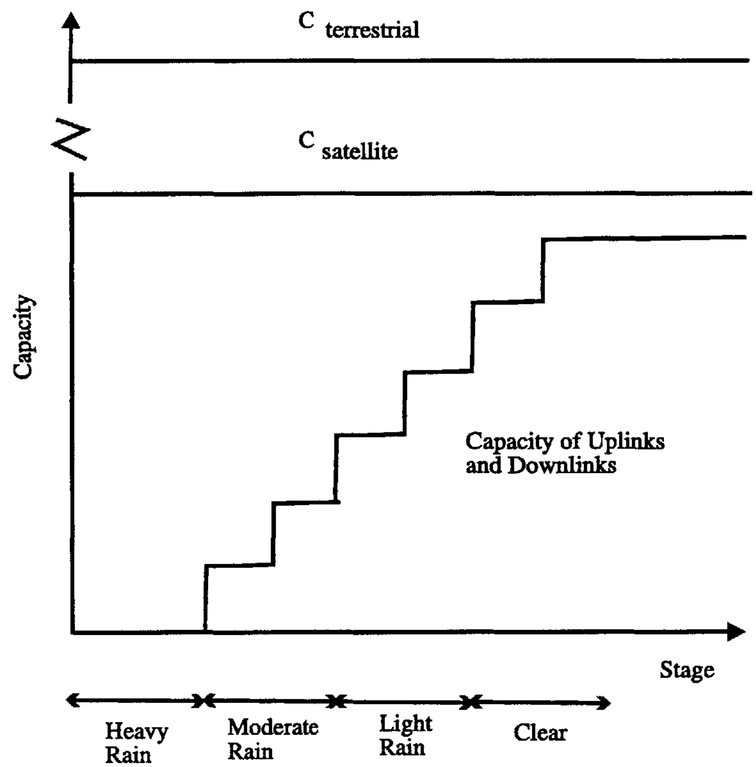

Figure 6-1: Graph of the capacity step function for uplinks and downlinks

it consists of four stages, namely Clear, Light Rain, Moderate Rain, and Heavy Rain [4]. Each stage occupies a region of the capacity step function as shown in figure (6-1).

Graphically, the entire global heterogeneous system can be conceived as two big meshes (one for the satellite network, the other for the terrestrial network) which are inter-connected by uplinks and downlinks which have smaller capacity.

6.3 Cost metrics in the heterogeneous network

Cost metrics are one-dimensional variables for indicating the preference over different systems. Depending on the value of cost metrics on different types of links, traffic will be routed to different paths. Cost metrics are usually assigned arbitrarily; however they can also be used as a measure of monetary cost, delay, jitter, bandwidth, reliability or hop count of a link or a system. Monetary cost has been the main focus

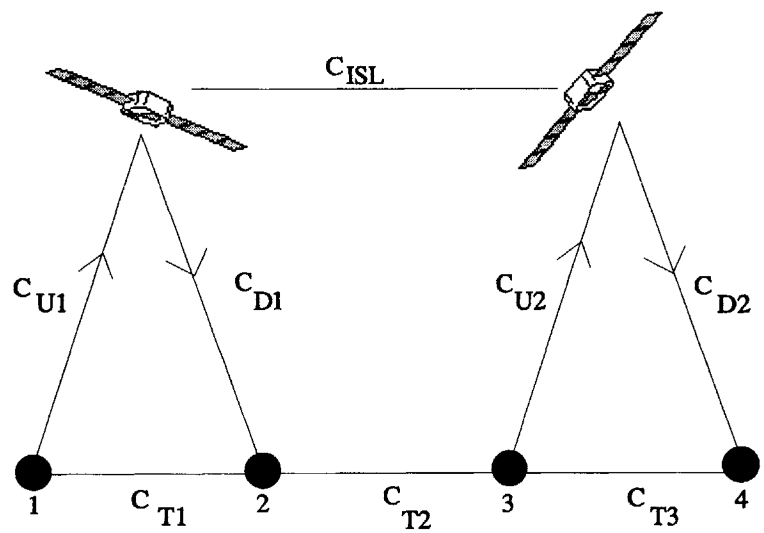

Figure 6-2: Heterogeneous Global Network

throughout the thesis. As a result, we will focus on this aspect of the cost metric in this section.

Three major components—terrestrial links, uplinks and downlinks, and inter-satellite links—form our heterogeneous global network. This heterogeneous network will provide an enormous number of route choices. The question of whether the satellite network, or the terrestrial network will be used depends on the cost metrics assigned to different types of links. Different assumptions of the links will lead to different utilization rate of the terrestrial and satellite systems. For a simple illustration, we consider the traffic going from point 1 to point 4 in figure (6-2). We assume the cost metric to be additive in this case since the traffic cost depends on the number of links it has passed through and the number of links is additive.

There are three different available paths as shown in the table below.

Case A reflects the case where traffic will never be sent up to the sky. Case B is

Case	Choice of path	Cost metrics associated
A	Terrestrial links only	$C_{T1} + C_{T2} + C_{T3}$
B	Up & downlinks but no ISLs	$C_{U1} + C_{D1} + C_{T2} + C_{U2} + C_{D2}$
C	ISLs with up & downlinks	$C_{U1} + C_{ISL} + C_{D2}$

Table 6.1: Three choices of path

the Globalstar case where only uplinks and downlinks, together with terrestrial links, are used. Case C resembles the Iridium or Teledesic case where inter-satellite links exist. The chosen path will be the one with the lowest cost metrics. We will examine various reasons behind why one system is chosen over the two other.

6.3.1 Terrestrial links

According to the table, case A will be chosen over case B if $C_{U1}+C_{D1}+C_{U2}+C_{D2} > C_{T1} + C_{T2}$. And if $C_{U1} + C_{ISL} + C_{D2} > C_{T1} + C_{T2} + C_{T3}$, case A will be chosen over case C.

With the assumption that the terrestrial links are essentially fiber optics links and can be manufactured and set up in low cost (except for undersea long fiber links), the cost metrics of terrestrial links will definitely be smaller than the other two cases. Terrestrial networks will always be used with this assumption.

6.3.2 Globalstar

Globalstar differs from the Iridium/Teledesic case for it does not have any inter-satellite links. Case B is chosen over Case C when $C_{ISL} > C_{D1} + C_{T2} + C_{U2}$. This will be true when the inter-satellite link is more expensive than an uplink, a downlink and a terrestrial link combined altogether, and this largely depends on the design assumptions and manufacturers of the network.

Case B is chosen over Case A when $C_{U1}+C_{D1}+C_{U2}+C_{D2} < C_{T1}+C_{T2}$. The cost of terrestrial links can be very expensive in the ocean since undersea cables cost a lot more than normal ground fiber optic cables, or in un-populated areas or mountain regions where it is not economically sensible to cover those regions with fiber. As a result, the Globalstar network is good for traffic which will at least pass through some regions where there is a scarcity of cheap terrestrial links such as oceans and mountains. However, this type of communications requires gateways that serve as the connecting points between the satellite network and the fiber networks.

6.3.3 Iridium/Teledesic

Case C is chosen over Case B when $C_{ISL} < C_{D1} + C_{T2} + C_{U2}$. When T2 is located in some remote areas or in the ocean, link T2 is very costly, and using inter-satellite-links will be a sound solution. Case C is chosen over Case A when $C_{U1} + C_{ISL} + C_{D2} < C_{T1} + C_{T2} + C_{T3}$. This can happen when all the terrestrial links are located in remote areas and therefore have become very expensive. Overall, Case C will be good for cases where traffic will only pass through areas with very limited supply of inexpensive terrestrial links.

Other considerations, such as delay, and congestion level, are usually used in assigning cost metrics to links. However, they will not be covered in the thesis since they are not the main focus here. Nevertheless, these cases can be simulated using the simulator in the next chapter by setting some arbitrary values to the link metrics.

6.4 Relations between uplinks, downlinks, and inter-satellite links

Traffic in most cases originates from the ground; therefore, when traffic is admitted into the satellite network system, traffic has to first pass through the uplinks and downlinks. The amount of traffic admitted into the satellite network is therefore dependent very much upon the capacity of uplinks and downlinks. In previous sections, we have neglected the effects of uplinks and downlinks; now we will examine the relationships in capacity between up/downlinks and the inter-satellite links.

6.4.1 Required capacity for uplinks and downlinks

Consider there is T_u amount of uniform traffic in the satellite network of size $M \times 2M$. By our earlier definition of uniform traffic, there will be $\frac{T_u}{2M^2}$ of traffic originating from each node. As a result, if the system wants to support T_u amount of uniform traffic, each uplink and downlink should at least have capacity $\frac{T_u}{2M^2}$.

We now extend this argument to hot spot traffic. Assuming that a stream of hot spot traffic is served by only one node, in order to support an outgoing (or incoming) hot spot traffic stream of T_h, the uplink (or downlink) of that node must therefore have a capacity of at least T_h. Similarly, extending to the case where multiple hot spot traffic streams originating from different nodes, the capacity of the up/downlink therefore must be greater or equal to the maximum hot spot traffic, $max\{T_h\}$. Therefore, with uniform traffic and hot spot traffic, the capacity of each uplink / downlink has to be at least $\frac{T_u}{2M^2} + max\{T_h\}$. Therefore, we have the following theorem:

Theorem 6.1 (Capacity of uplinks and downlinks) *Assuming all uplinks and downlinks have equal capacities, in order for the satellite network to serve a stream of uniform traffic in the crosslinks, the uplinks and downlinks have to at least have a capacity of:*

$$\frac{T_u}{2M^2} \tag{6.4}$$

On the other hand, in order to serve both uniform traffic and hot spot traffic, the uplinks and downlinks have to at least have a capacity of:

$$\frac{T_u}{2M^2} + \max\{T_h\} \tag{6.5}$$

of capacity. $\max\{T_h\}$ *is the traffic from the busiest hot spot.*

6.4.2 Necessary relationships between uplinks, downlinks, and inter-satellite links

Previously we have shown that the crosslink capacity has to be at least $\frac{T_u}{8M}$, in order to support a uniform traffic stream of T_u. As a result, the ratio of capacity between crosslink and up/downlink for maximum utilization of crosslinks under uniform traffic:

$$\frac{\text{capacity of a crosslink}}{\text{capacity of an up/downlink}} \leq \frac{\frac{T_u}{8M}}{\frac{T_u}{2M^2}} = \frac{2M^2T_u}{8MT_u} = \frac{M}{4} \tag{6.6}$$

As a result, we have the following theorem:

Theorem 6.2 (Constraint for uplinks and downlinks under uniform traffic) *If each inter-satellite links is sized at $\frac{T_u}{8M}$ to support T_u amount of uniform traffic, each uplink or downlink has to have this minimum amount of capacity:*

$$\frac{4}{M} \times \textit{Capacity of crosslink} \tag{6.7}$$

If the uplinks and downlinks have more capacity than the amount described in equation (6.7), the uplinks and downlinks are then able to support extra traffic, such as the bent-pipe traffic in Globalstar, that does not go through the inter-satellite links.

From equation (6.7), the capacity of a crosslink is greater than the capacity of an uplink and downlink if and only if $\frac{4}{M} < 1$, or $M > 4$. The constellation size of Teledesic is 12×24 [14]. The M for the network is thus 12. Therefore, using equation (6.7), the capacity of the uplink or downlink has to be at least $\frac{4}{12}$, or $\frac{1}{3}$, of the capacity of the ISL. On the other hand, for Iridium, whose M is 6 [6], the capacity of the an uplink or a downlink has to be at least $\frac{4}{6}$, or $\frac{2}{3}$ of the capacity of an ISL in order for the system to be able to admit enough traffic for maximal performance of the crosslinks.

From this calculation, the uplinks and downlinks of Iridium should have more similar capacities to its inter-satellite links when compared to the Teledesic system. This is in fact true since Iridium's inter-satellite links, uplinks and downlinks are all RF links and they have similar capacities. It uses RF ISLs because it aims to provide low-rate voice communications. On the other hand, Teledesic aims to provide higher rate data communications and therefore uses higher capacity optical laser crosslinks. Because of that, the crosslinks and up/downlinks of Teledesic have a bigger difference in capacity. This agrees with our analysis that M should be made larger if there is a large disparity between the up and downlink capacity and the crosslink capacity.

6.5 Summary

In this chapter we have explored the idea of incorporating the satellite communication network into part of the global heterogeneous network. The differences between terrestrial links, up and downlinks, and inter-satellite links have been discussed. Depending on the values of cost metrics for different types of links, traffic will take different paths, and we have discussed about different scenarios in which one type of links is more favorable than the others. At the end of the chapter, we have found a necessary condition between the capacity of an up/downlink and an inter-satellite link so that the satellite network can be used at its maximal potential for uniform traffic.

Chapter 7

Simulation of routing in a heterogeneous network

In this chapter a framework to simulate the routing of packets in a heterogeneous network is developed using MATLAB. This framework uses the assumptions we have made in the previous chapter. Using this framework, we could explore the transient properties of the network that are very hard to be analyzed mathematically. Those properties can be the stability issues of a particular routing algorithm, or the sensitivities of various parameters, such as amount of traffic; traffic distribution; capacities of crosslinks, terrestrial links and up/downlinks; routing table update interval; and cost metrics of the links, to the performance to different routing algorithms.

In this framework the satellite network and the terrestrial network are modeled as two square mesh grids of the same size, and the two meshes are linked together by uplinks and downlinks. There is There are a few variables that a user could set:

1. size of the mesh ($M \times N$)

2. capacities of crosslinks, terrestrial links and up/downlinks

3. buffer sizes of terrestrial and satellite nodes

4. time and amount of uniform traffic

5. time and amount of hot spot traffic originated from a particular node

6. time interval to update routing table

Users must input their own routing algorithms (tables) to the framework according to a format described in the program; this adds to the robustness by making the framework able to simulate any routing algorithms defined by the user. The source code of the framework can be found in Appendix B.

As an example, we have produced results for a simple 3×3 grid. We have plotted the amount of traffic in crosslinks versus time for the following cases:

1. $T_u = 81$ and $T_u = 162$

2. $T_u = 81$ with $T_h = 9$ originating from the center

In the graphs, we are able to notice how the traffic is built up before reaching its steady state. Furthermore, as shown in the first graph, the amount of traffic in a crosslink in the simulation matches to the result we have got in equation (3.10), with $M, N = 3$. The second graph shows the difference in the amount of traffic among crosslinks at different locations.

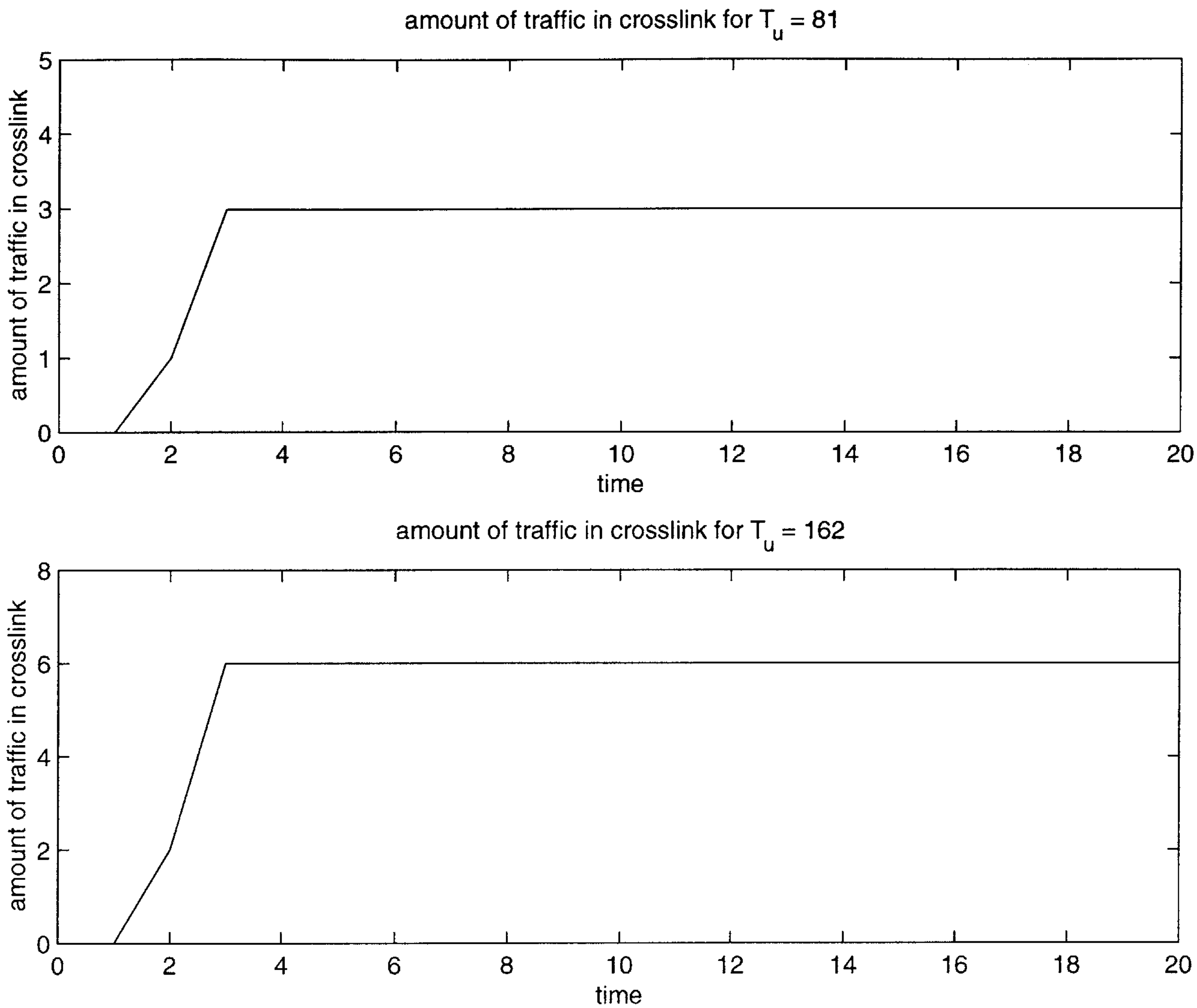

Figure 7-1: Amount of traffic in crosslink vs. time for different values of T_u

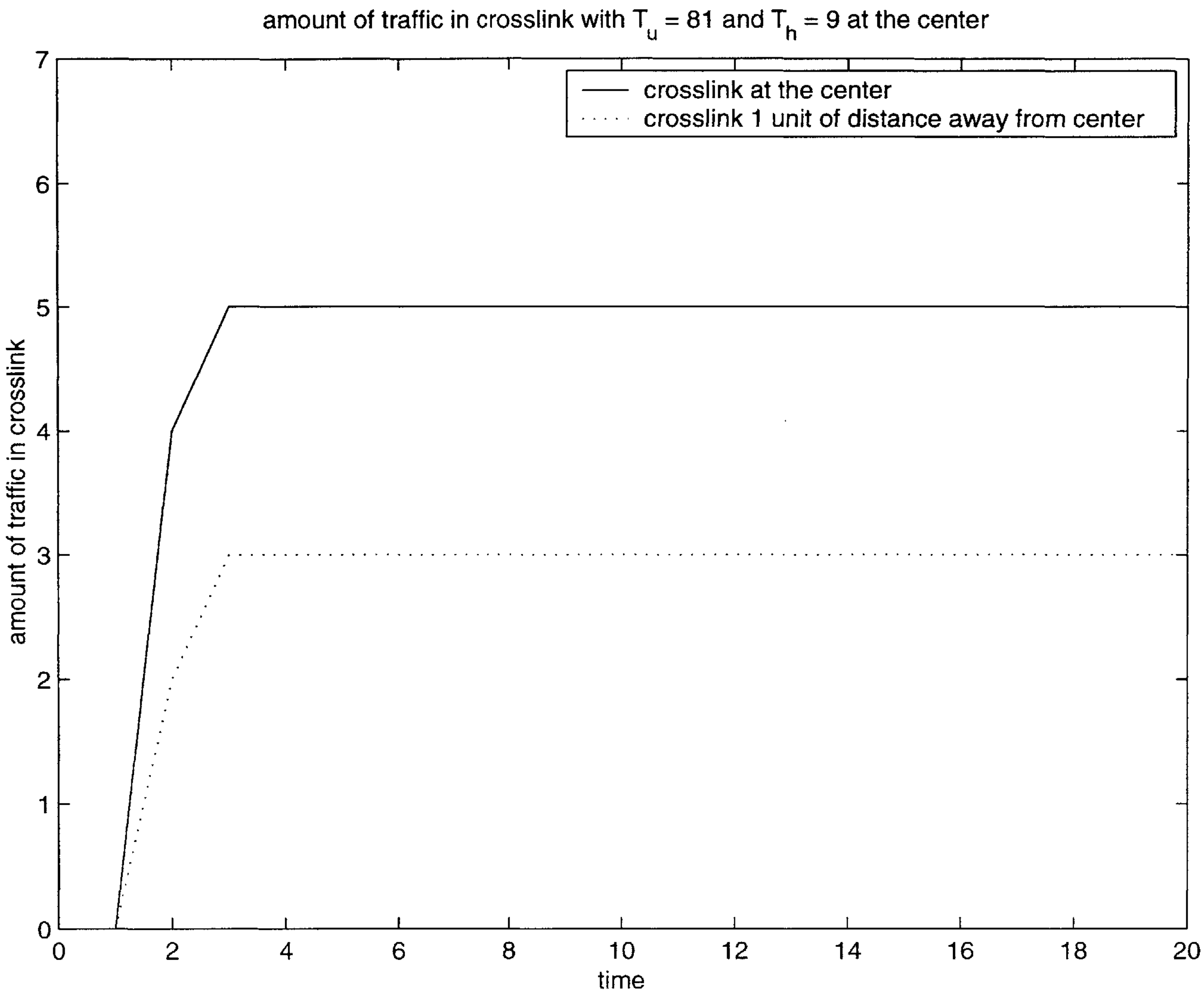

Figure 7-2: Amount of traffic in crosslink with $T_u = 81$ and $T_h = 9$ at the center

Chapter 8

Conclusion and Future Research

8.1 Conclusion

Building a satellite communications system is very costly; as a result, analyses on the system costs should definitely be carried out before actually constructing any system. In this thesis, we have done some preliminary works on the cost of a satellite communications system. Since crosslinks constitute a significant part of the entire system cost, particularly in LEO satellite systems, we first derived a cost equation of the system based on the crosslink cost alone. We then developed a simple seamless model for the constellation of the satellites. Even though a seamless model is unrealistic, it nevertheless provides us with a basic understanding of the constellation. Using the cost equation and the constellation model we had developed, we tried to optimize the cost of the crosslinks by assuming a uniform distribution of traffic from each satellite node. We then extended the analysis to GEO satellites as well. Lastly, at the end of Chapter 3, we made some remarks on Iridium and Teledesic networks based on our analysis.

We then looked into further cost reduction by reducing the number of crosslinks per node. We looked at two possible new constellations, the 3-crosslink-per-node mesh network and the 1-inter-plane-crosslink mesh network. We have found the 3-crosslink-per-node mesh network will always result in further cost reduction; on the other hand, cost reduction may or may not happen in the 1-inter-plane-crosslink mesh network, depending on the ratio of fixed cost to variable cost, and other constants.

Hot spot traffic is another traffic pattern that we have studied closely, together with uniform traffic. In Chapter 5, we analyzed in the interaction of hot spot traffic and uniform traffic in a square mesh. We first made some observations about traffic in an infinite mesh, then continued onto analysis for fixed sized grids. We also made observations on how pass through traffic should be re-routed to minimize the capacity required for crosslinks.

We lastly incorporated the satellite communications network into the global heterogeneous network. We first compared the different systems, with an emphasis on the difference in capacities among inter-satellite links, up/down links and terrestrial links. We then studied the cost metrics for links in the heterogeneous system. By assuming different cost metrics for different links, we made some comments on the three most prominent satellite systems nowadays—Iridium, Teledesic and Globalstar. At the end, a framework to study the transient properties of the heterogeneous network was built, using the assumptions we had discussed in Chapter 6.

8.2 Directions for Future Research

Our cost equation relies on the assumption that crosslinks form a significant portion of the cost in a satellite systems. While this assumption may be true at present, this may change as new evolving technologies are developed. Furthermore, this cost equation can vary a lot as it depends very much on the vendors and the technologies used. There has been little research on this aspect of satellite communications engineering since this is very much vendor-specific. Therefore, one possible direction of future research is to derive a more complex and realistic cost equation for the total system, by incorporating more components of a satellite system, or by making the cost equation more technology-dependent, rather than vendor-dependent.

The mesh model that most of the analyses in this thesis is based on is a seamless model that contradicts with the seamed model that Iridium or Teledesic has. Even though the seamless model provides a very close approximation to the seamed model, it is possible to refine the analysis of this thesis by using a more complex seamed model, as in section (3.2.3).

In this thesis, two forms of traffic flows—uniform traffic and hot spot traffic—are investigated. These traffic patterns are fictional; they are created merely to facilitate our analysis of traffic in a mesh model. Researchers have been trying to model traffic patterns in the Internet using Poisson distribution, heavy tailed distributions, fractals [15, 13], or Markov-modulated Poisson Process (MMPP) [11]. An improvement of this thesis can be made by using these distributions in the analysis; however, this may be a very difficult task to complete.

Lastly, as of September 2000, with the Iridium system to be de-orbited, the future of satellite communications networks remains uncertain. It is definitely more sensible to do more cost analysis before actually constructing a network in the sky. Further analysis on the market demand for such an expensive communications system should be carried out; otherwise, another setback for satellite networks, like the de-orbiting of the Iridium satellites, would definitely happen again.

Appendix A

Uniform capacity in crosslinks

As mentioned in section (3.2.2), when we map a spherical surface of size $M \times 2M$ into a $2M \times 2M$ model, we will maintain all the link connectivities but at the same time double-count the number of nodes on the spherical surface. In this section, we will try to modify this $2M \times 2M$ grid into a grid with no double coverage, and find the capacities of the crosslinks on this modified grid. With this modified grid, intra-plane crosslinks will not be favored over inter-plane crosslinks and as a result, all crosslinks will receive equal amount of capacity.

There are two images of the same node in the $2M \times 2M$ grid. We modify the $2M \times 2M$ grid by comparing the distances of each pair of image nodes from the center and removing the one that is farther away from the center. We first choose an arbitrary node to be the center of the grid and then examine the other $(M \times 2M) - 1$ pairs of image nodes one by one. For each pair, we delete one of the nodes which is farther away from the center. After this process, we will arrive at a mesh which bears the shape of a rhombus with the lengths of both of the diagonals equal to $2M$ hops. The outer edges of the rhombus are two sets of duplicated nodes in which both images of the nodes are equidistant from the center. Also, the four corner of the rhombus is in fact the same node—the node that is exactly at the opposite side of the Earth. We will assume that traffic to the duplicated nodes at the outer edges from the center will use all the available equal-length paths at equal probabilities.

With uniform traffic, every node is a sender and the total traffic inside the grid is

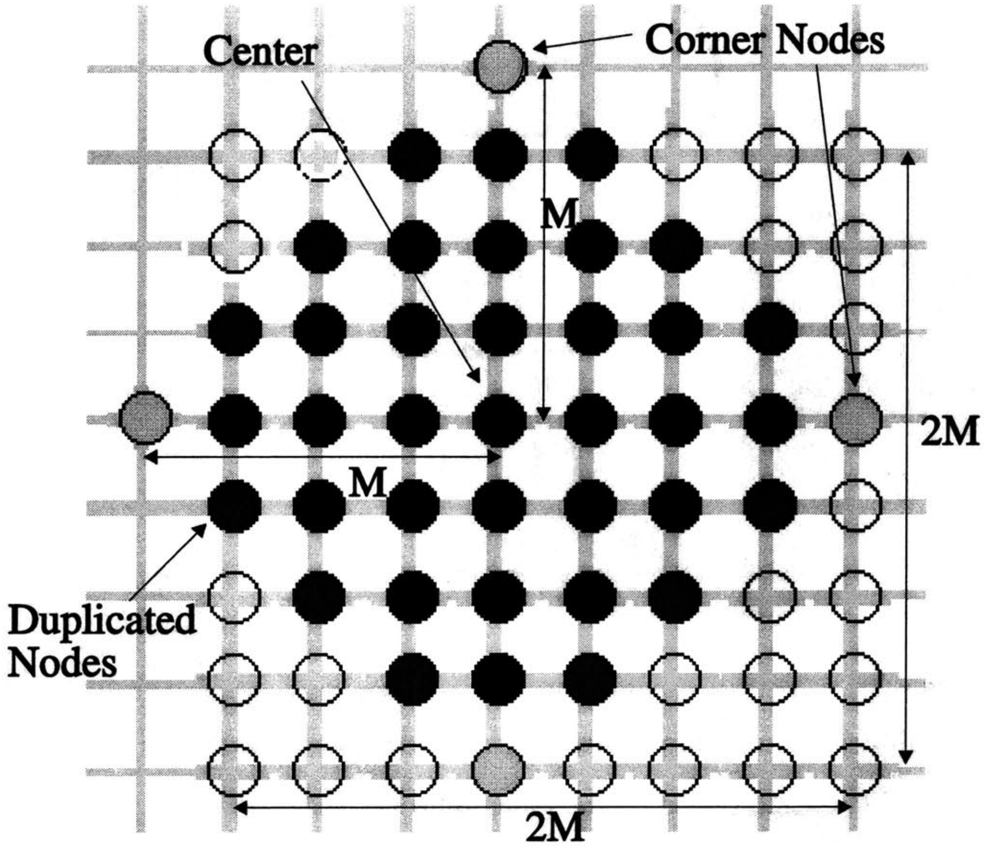

Figure A-1: Rhombus on a $2M \times 2M$ Grid

thus the superposition of $M \times 2M$ rhombi. With this superposition and the symmetry of a rhombus, all crosslinks will have equal amount of traffic. We will just look at the inter-plane crosslinks in this section, since intra-plane and inter-plane crosslinks will have the same capacities and the calculation for the amount of traffic in either of them is identical.

Since the outer edges of the rhombus contain duplicated nodes, we will break the calculation into two parts: we will first consider the nodes inside the edges (interior nodes), and then the nodes on the outer edges (exterior nodes).

We will first find the total number of horizontal hops needed for the center to reach every interior node. On the left hand side of the rhombus, there are 1 node at $M-1$ units of distance away from the center, 3 nodes at $M-2$ units of distance away from the center, and so on. This is the same for the right hand side as well. To find the total number of horizontal hops, we add up the horizontal distances of each node from the center. The total number of horizontal hops required is therefore:

$$
\begin{aligned}
& 2 \sum_{i=1}^{M-1} (M-i)(2i-1) && \text{(A.1)} \\
= & \; 2 \sum_{i=1}^{M-1} [-2i^2 + (2M+1)i - M] && \text{(A.2)} \\
= & \; 2\,[-2 \sum_{i=1}^{M-1} i^2 + (2M+1) \sum_{i=1}^{M-1} i - \sum_{i=1}^{M-1} M] && \text{(A.3)} \\
= & \; 2\,[-2 \frac{(M-1)(M)(2M-1)}{6} + \frac{(2M+1)(M-1)M}{2} - M(M-1)] && \text{(A.4)} \\
= & \; 2\,[\frac{-1}{3}(2M^3 - 3M^2 + M) + \frac{2M^3 - M^2 - M}{2} - M^2 + M] && \text{(A.5)} \\
= & \; \frac{1}{3}(-4M^3 + 6M^2 - 2M + 6M^3 - 3M^2 - 3M - 6M^2 + 6M) && \text{(A.6)} \\
= & \; \frac{1}{3}(2M^3 - 3M^2 + M) && \text{(A.7)}
\end{aligned}
$$

We then consider the exterior nodes. There are two paths to each node, except for the corner nodes (which are actually duplicated images of the same node at the exact opposite end of the Earth) which have 4 paths. Again, on the left side of the rhombus, there is one exterior node at horizontal distance M from the center, and there are 2 exterior nodes at horizontal distances $M-1$ to 0 from the center. Similar to finding the total number of horizontal hops for the interior nodes, we add up the horizontal distances of each exterior node from the center. However, since there are two different paths for each actual node (except the corner node) on the exterior edges, we multiply the total number of hops by $\frac{1}{2}$. And for the corner nodes, since there are 4 paths, we multiply the number by $\frac{1}{4}$. Therefore, the total number of horizontal hops required for the exterior nodes is:

$$
\begin{aligned}
& \frac{1}{2}\,(\text{number of hops required for the non-corner exterior nodes}) + \\
& \frac{1}{4}\,(\text{number of hops required for the corner nodes}) && \text{(A.8)} \\
= & \; \frac{1}{2}\,(4 \sum_{i=1}^{M-1} i) + \frac{1}{4}\,2M && \text{(A.9)}
\end{aligned}
$$

$$= 2\sum_{i=1}^{M-1} i + \frac{M}{2} \quad \text{(A.10)}$$

$$= 2\frac{(M-1)M}{2} + \frac{M}{2} \quad \text{(A.11)}$$

$$= M^2 - \frac{1}{2}M \quad \text{(A.12)}$$

The total number of horizontal hops required is therefore:

$$(\frac{2}{3}M^3 - M^2 + \frac{1}{3}M) + M^2 - \frac{1}{2}M \quad \text{(A.13)}$$

$$= \frac{2}{3}M^3 - \frac{1}{6}M \quad \text{(A.14)}$$

There are a total of $2M^2$ nodes. As a result, the average number of horizontal hops per node is:

$$H_{avg_{inter-plane}} = \frac{1}{2M^2}(\frac{2}{3}M^3 - \frac{1}{6}M) \quad \text{(A.15)}$$

$$= \frac{M}{3} - \frac{1}{12M} \quad \text{(A.16)}$$

As in section (3.2.1), to find the capacities for the inter-plane and intra-plane crosslinks under uniform traffic, we use the following equation:

$$C_{crosslink} = \frac{C_{hop}\, H_{avg}\, \#(paths)}{\#(crosslinks)} \quad \text{(A.17)}$$

where

$C_{crosslink}$ = capacity of a crosslink,

C_{hop} = capacity per path,

H_{avg} = average number of hops per path,

$\#(paths)$ = number of paths, and

$\#(crosslinks)$ = number of crosslinks.

For a mesh of size $M \times 2M$, each path will have $\frac{T_u}{(2M^2)^2}$ amount of traffic. As a result, C_{hop} is equal to $T_u/4M^4$. There are $4M^2$ inter-plane crosslinks in the mesh.

Therefore, for inter-plane crosslinks,

$$
\begin{aligned}
C_{inter-plane} &= \frac{1}{4M^2}\,\frac{T_u}{4M^4}\,(\frac{M}{3}-\frac{1}{12M})\;4M^4 && \text{(A.18)}\\
&= \frac{T_u}{4M^2}\,(\frac{M}{3}-\frac{1}{12M}) && \text{(A.19)}\\
&= T_u\,(\frac{1}{12M}-\frac{1}{48M^3}) && \text{(A.20)}\\
&\approx \frac{T_u}{12M}, \quad \text{when M is large} && \text{(A.21)}
\end{aligned}
$$

We have found the capacity of the inter-plane crosslinks. Intra-plane crosslinks will have the same amount of capacity because of symmetry. As we should expect, this result is between the inter-plane crosslink capacity and the intra-plane crosslink capacity in the $M \times 2M$ seamless case, which are $\frac{T_u}{16M}$ and $\frac{T_u}{8M}$ respectively. Also, this result is bigger than the result obtained in the seamed case in section (3.2.3). This is because a seamless model is used here. When compared to the seamed model, the crosslinks across the seam in this model allow more efficient routing, and as a result, the capacity required for crosslinks in this case is lower.

Appendix B

Source Code

Source code of the simulation is enclosed here.

```
% CONSTANTS

% M = size of grid in x-dimension
% N = size of grid in y-dimension (M & N are odd)
% MAX_NUM_PACKET  = maximum number of allowable traffic stream
%                   should be set to a very large number to ensure maximum will not be reached
% TIME_OF_SIMULATION = total time of simulation
% routing_update_interval = update interval of routing table
% sky_buffer_size = buffer size of satellite nodes
% terr_buffer_size = buffer size of terrestrial nodes
% ISL_size  = capacity of the ISLs
% uplink_size = capacity of the uplinks
% downlink_size = capacity of the downlinks
% terr_link_size = capacity of the terrestrial links
% ISL_metrics = initial metric value for the ISLs
% updown_metrics = intial metric value for the uplinks and downlinks
% terr_metrics = intial metric value for the terrestrial links


% DATA STRUCTURES

% nodes -- terrestrial and satellite nodes
% total number of nodes for each type= NM
% (M(x-1) + y)-th node = satellite node of (x,y)
% (MN + M(x-1) + y)-th node = terrestrial node of (x,y)
% fields inside nodes
% (1,2) location of the node
```

```
% (3) amount of traffic arrived at the node
% (4) available buffer size in node

% links -- ISLs, terrestrial links, and up/downlinks
% total number of ISLs = 4MN
% total number of terrestrial links = 4MN
% total number of up/downlinks = 2MN
% (M(x -1)+ (y-1)) * 4 + 1-th link = left ISL of (x,y)
% (M(x -1)+ (y-1)) * 4 + 2-th link = upper ISL of (x,y)
% (M(x -1)+ (y-1)) * 4 + 3-th link = right ISL of (x,y)
% (M(x -1)+ (y-1)) * 4 + 4-th link = lower ISL of (x,y)
% (4MN + M(x -1)+ (y-1)) * 4 + 1-th link = left terrestrial link of (x,y)
% (4MN + M(x -1)+ (y-1)) * 4 + 2-th link = upper terrestrial link of (x,y)
% (4MN + M(x -1)+ (y-1)) * 4 + 3-th link = right terrestrial link of (x,y)
% (4MN + M(x -1)+ (y-1)) * 4 + 4-th link = lower terrestrial link of (x,y)
% (8MN + M(x -1)+ (y-1)) * 2 + 1-th link = uplink at (x,y)
% (8MN + M(x -1)+ (y-1)) * 2 + 2-th link = downlink at (x,y)
% fields inside links
% (1)capacity available in link
% (2,3)origin and (4,5)destination of the link
% (6) cost metric of the link
% (7) flag to indicate whether origin is in the sky or in the ground -- 1 = in sky; 0 = in ground
% (8) flag to indicate whether destination is in the sky or in the ground -- 1 = in sky; 0 = in ground

% fields inside packets
% (1) flag to signal whether this place is used -- 0 = not in used; 1 = in used
% (2,3) current node
% (4,5) origin
% (6,7) destination
% (8) time started
% (9, 10) transit information
% (11) flag to determine whether the switching is done for the current time interval or not -- 1 = not done; 0 = done
% (12) flag to indicate whether current node is in the sky or in the ground -- 1 = in sky; 0 = in ground
% (13) flag to indicate whether origin is in the sky or in the ground -- 1 = in sky; 0 = in ground
% (14) flag to indicate whether destination is in the sky or in the ground -- 1 = in sky; 0 = in ground

% fields inside routing_table
% (1,2) origin
% (3) flag to indicate whether origin is in the sky or in the ground -- 1 = in sky; 0 = in ground
% (4,5) destination
% (6) flag to indicate whether destination is in the sky or in the ground -- 1 = in sky; 0 = in ground
% (7) number of intermediate nodes
% (8,9,10) (11,12,13) (14, 15, 16) ... = triplets of intermediate nodes

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% process.m
% main backbone of the simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all;

% set time to start at 1
global time
time = 1;

%initialization
init;

% generation of empty packets to hold traffic information
global MAX_NUM_PACKET
global packets
MAX_NUM_PACKET = 1000;
packets = zeros(MAX_NUM_PACKET,17);

% initial generation of uniform and hot spot traffic put here
gen_uniform_traffic;
%gen_hot_spot_traffic([2 2]);

% user-supplied routing table put here to replace gen_routing_table
gen_routing_table;

% specifiy time of simulation and routing update interval here
TIME_OF_SIMULATION = 100;
routing_update_interval = 10;

while (time <= TIME_OF_SIMULATION)

   if (mod(time, routing_update_interval) == 0)
      % user-supplied routing table put here to replace gen_routing_table
      gen_routing_table;
   end

   % procedure for switching of packets put here
   packet_switching1;

   %increase time by 1
   time = time + 1;

   % further generation of traffic put here
   %gen_uniform_traffic;
```

```
   % can put data collection routines here
   nodes
   time
   pause

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% init.m
% initialization of the simulation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% initialization of the grid
% M = size of grid in x-dimension
% N = size of grid in y-dimension (M & N are odd)

global M
global N
global nodes
global links

M = 3;
N = 3;

% node -- terrestrial and satellite nodes
% total number of nodes for each type= NM
% (M(x-1) + y)-th node = satellite node of (x,y)
% (MN + M(x-1) + y)-th node = terrestrial node of (x,y)
% information about nodes
% (1,2) location of the node
% (3) amount of traffic arrived at the node
% (4) available buffer size in node

% satellite nodes
nodes = zeros(2*M*N, 4);
sky_buffer_size = 1000;
terr_buffer_size = 10000;
for i = 1:M
   for j = 1:N
      nodes(M*(i-1) + j, 1) = i;
      nodes(M*(i-1) + j, 2) = j;
      nodes(M*(i-1) + j, 3) = 0;
      nodes(M*(i-1) + j, 4) = sky_buffer_size; % in case for nodes having different buffer sizes
   end
 end
```

```
% terrestrial nodes
for i = 1:M
   for j = 1:N
      nodes(M*N+ M*(i-1) + j, 1) = i;
      nodes(M*N+ M*(i-1) + j, 2) = j;
      nodes(M*N+ M*(i-1) + j, 3) = 0;
      nodes(M*N+ M*(i-1) + j, 4) = terr_buffer_size; % in case for nodes having different buffer sizes
   end
end

% links -- ISLs, terrestrial links, and up/downlinks
% total number of ISLs = 4MN
% total number of terrestrial links = 4MN
% total number of up/downlinks = 2MN
% (M(x -1)+ (y-1)) * 4 + 1-th link = left ISL of (x,y)
% (M(x -1)+ (y-1)) * 4 + 2-th link = upper ISL of (x,y)
% (M(x -1)+ (y-1)) * 4 + 3-th link = right ISL of (x,y)
% (M(x -1)+ (y-1)) * 4 + 4-th link = lower ISL of (x,y)

% (4MN + M(x -1)+ (y-1)) * 4 + 1-th link = left terrestrial link of (x,y)
% (4MN + M(x -1)+ (y-1)) * 4 + 2-th link = upper terrestrial link of (x,y)
% (4MN + M(x -1)+ (y-1)) * 4 + 3-th link = right terrestrial link of (x,y)
% (4MN + M(x -1)+ (y-1)) * 4 + 4-th link = lower terrestrial link of (x,y)

% (8MN + M(x -1)+ (y-1)) * 2 + 1-th link = uplink at (x,y)
% (8MN + M(x -1)+ (y-1)) * 2 + 2-th link = downlink at (x,y)


% fields of links
% (1)capacity available in link
% (2,3)origin and (4,5)destination of the link
% (6) cost metric of the link
% (7) flag to indicate whether origin is in the sky or in the ground -- 1 = in sky; 0 = in ground
% (8) flag to indicate whether destination is in the sky or in the ground -- 1 = in sky; 0 = in ground

links = zeros(10*M*N, 8);
terr_link_size = 10000;
ISL_size =1000;
uplink_size = 800;
downlink_size = 800;
ISL_metrics = 5;
updown_metrics = 10;
terr_metrics = 1;
for i = 1:M
```

```
   for j = 1:N
      for k = 1:4
         links((M*(i-1) + (j-1))*4 + k, 1) = ISL_size;
         links((M*(i-1) + (j-1))*4 + k, 2) = i;
         links((M*(i-1) + (j-1))*4 + k, 3) = j;
         links((M*(i-1) + (j-1))*4 + k, 6) = ISL_metrics;
         links((M*(i-1) + (j-1))*4 + k, 7) = 1;
         links((M*(i-1) + (j-1))*4 + k, 8) = 1;

      end

      if (i == 1)
         links((M*(i-1) + (j-1))*4 + 1, 4) = M;
      else
         links((M*(i-1) + (j-1))*4 + 1, 4) = i-1;
      end
      links((M*(i-1) + (j-1))*4 + 1, 5) = j;

      links((M*(i-1) + (j-1))*4 + 2, 4) = i;
      if (j == N)
         links((M*(i-1) + (j-1))*4 + 2, 5) = 1;
      else
         links((M*(i-1) + (j-1))*4 + 2, 5) = j+1;
      end

      if (i == M)
         links((M*(i-1) + (j-1))*4 + 3, 4) = 1;
      else
         links((M*(i-1) + (j-1))*4 + 3, 4) = i+1;
      end
      links((M*(i-1) + (j-1))*4 + 3, 5) = j;

      links((M*(i-1) + (j-1))*4 + 4, 4) = i;
      if (j == 1)
         links((M*(i-1) + (j-1))*4 + 4, 5) = N;
      else
         links((M*(i-1) + (j-1))*4 + 4, 5) = j-1;
      end


   end
end

for i = 1:M
   for j = 1:N
```

```
      for k = 1:4
         links(4*M*N+(M*(i-1) + (j-1))*4 + k, 1) = terr_link_size;
         links(4*M*N+(M*(i-1) + (j-1))*4 + k, 2) = i;
         links(4*M*N+(M*(i-1) + (j-1))*4 + k, 3) = j;
         links(4*N*M+(M*(i-1) + (j-1))*4 + k, 6) = terr_metrics;
         links(4*N*M+(M*(i-1) + (j-1))*4 + k, 7) = 0;
         links(4*N*M+(M*(i-1) + (j-1))*4 + k, 8) = 0;

      end

      if (i == 1)
         links(4*M*N+(M*(i-1) + (j-1))*4 + 1, 4) = M;
      else
         links(4*M*N+(M*(i-1) + (j-1))*4 + 1, 4) = i-1;
      end
      links(4*M*N+(M*(i-1) + (j-1))*4 + 1, 5) = j;

      links(4*M*N+(M*(i-1) + (j-1))*4 + 2, 4) = i;
      if (j == N)
         links(4*M*N+(M*(i-1) + (j-1))*4 + 2, 5) = 1;
      else
         links(4*M*N+(M*(i-1) + (j-1))*4 + 2, 5) = j+1;
      end

      if (i == M)
         links(4*M*N+(M*(i-1) + (j-1))*4 + 3, 4) = 1;
      else
         links(4*M*N+(M*(i-1) + (j-1))*4 + 3, 4) = i+1;
      end
      links(4*M*N+(M*(i-1) + (j-1))*4 + 3, 5) = j;

      links(4*M*N+(M*(i-1) + (j-1))*4 + 4, 4) = i;
      if (j == 1)
         links(4*M*N+(M*(i-1) + (j-1))*4 + 4, 5) = N;
      else
         links(4*M*N+(M*(i-1) + (j-1))*4 + 4, 5) = j-1;
      end
   end
end


for i = 1:M
   for j = 1:N
      links(8*M*N+(M*(i-1) + (j-1))*2 + 1, 1) = uplink_size;
      links(8*M*N+(M*(i-1) + (j-1))*2 + 2, 1) = downlink_size;
```

```
      links(8*M*N+(M*(i-1) + (j-1))*2 + 1, 7) = 0;
      links(8*M*N+(M*(i-1) + (j-1))*2 + 1, 8) = 1;
      links(8*M*N+(M*(i-1) + (j-1))*2 + 2, 7) = 1;
      links(8*M*N+(M*(i-1) + (j-1))*2 + 2, 8) = 0;

      for k = 1:2

         links(8*M*N+(M*(i-1) + (j-1))*2 + k, 2) = i;
         links(8*M*N+(M*(i-1) + (j-1))*2 + k, 4) = i;

         links(8*M*N+(M*(i-1) + (j-1))*2 + k, 3) = j;
         links(8*M*N+(M*(i-1) + (j-1))*2 + k, 5) = j;

         links(8*N*M+(M*(i-1) + (j-1))*4 + k, 6) = ISL_metrics;

      end
   end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% gen_hot_spot_traffic.m
% function to generate hot spot traffic
% parameter: origin -- origin of the hot spot traffic stream
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function gen_hot_spot_traffic(origin)

global M
global N
global nodes
global packets
global time

for i = 1:M
   for j = 1:N

      if (nodes(M*(origin(1)-1) + origin(2), 4) > 0)
         nodes(M*(origin(1)-1) + origin(2), 4) = nodes(M*(origin(1)-1) + origin(2), 4) - 1;
      else
         error('buffer reached capacity');
      end

      slot = placenewpacket(packets);

      % path of the packet = (origin(1),origin(2)) to (i,j)
```

```
      packets(slot, 1) = 1;
      packets(slot, 2) = origin(1);
      packets(slot, 3) = origin(2);
      packets(slot, 4) = origin(1);
      packets(slot, 5) = origin(2);
      packets(slot, 6) = i;
      packets(slot, 7) = j;
      packets(slot, 8) = time;
      packets(slot, 9) = 0;
      packets(slot, 12) = 1;
      packets(slot, 13) = 1;
      packets(slot, 14) = 1;
   end
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% gen_uniform_traffic.m
% script file to generate uniform traffic for each node
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global time;

for i = 1:M
   for j = 1:N
      for k = 1:M
         for p = 1:N
            if (nodes(M*(i-1) + j, 4) > 0)
               nodes(M*(i-1) + j, 4) = nodes(M*(i-1) + j, 4) - 1;
            else
               error('buffer reached capacity');
            end

            slot = placenewpacket(packets);
            % packet path = (i,j) to (k,p)
            packets(slot, 1) = 1;
            packets(slot, 2) = i;
            packets(slot, 3) = j;
            packets(slot, 4) = i;
            packets(slot, 5) = j;
            packets(slot, 6) = k;
            packets(slot, 7) = p;
            packets(slot, 8) = time;
            packets(slot, 9) = 0;
            packets(slot, 12) = 1;
```

```
            packets(slot, 13) = 1;
            packets(slot, 14) = 1;

         end
      end
   end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% placenewpacket.m
% function to an empty slot to place new traffic in data
% structure packets
% parameter: packets -- data structure for holding traffic
% return value: the index of the empty slot
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function place = placenewpacket(packets)
% function to look for the next available slot for packet
global MAX_NUM_PACKET

for i = 1:MAX_NUM_PACKET
   if (packets(i,1) == 0)
      place = i;
      return
   end
end

error('max number of packets reached');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% packet_switching1.m
% this is a script file which looks at the routing table for each node and performs packet switching
% for all the packets currently in transit
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i = 1:MAX_NUM_PACKET
   if (packets(i,1) ~= 0)
      packets(i,11) = 1;
   end
end


for i = 1:MAX_NUM_PACKET

   if (packets(i,1) ~= 0)
```

```
% look up routing table -- find next hop
r_table_loc = M*(packets(i,4)-1)+ packets(i,5);
entry_loc = find(routing_table(r_table_loc, :, 4) == packets(i,6) & ...
   routing_table(r_table_loc, :, 5) == packets(i,7) & routing_table(r_table_loc, :, 6) == packets(i,14));

if (packets(i,2) == packets(i,6)) & (packets(i,3) == packets(i,7)) & (packets(i,12) == packets(i,14))
   % reached destination already
   packets(i,11) = 0;

else
   % find next hop
   %nexthop = zeros(1,3);
   if packets(i,9) >= routing_table(r_table_loc, entry_loc,7)
      packets(i,15) = routing_table(r_table_loc, entry_loc,4);
      packets(i,16) = routing_table(r_table_loc, entry_loc,5);
      packets(i,17) = routing_table(r_table_loc, entry_loc,6);
   else

      packets(i,15) = routing_table(r_table_loc, entry_loc, 3*packets(i,9)+8);
      packets(i,16) = routing_table(r_table_loc, entry_loc, 3*packets(i,9)+9);
      packets(i,17) = routing_table(r_table_loc, entry_loc, 3*packets(i,9)+10);
   end

   if (packets(i,11) ~= 0)
      % check crosslink is full or not
      if (links(find(links(:,2) == packets(i,2) &  links(:,3) == packets(i,3) & ...
            links(:,4) == packets(i,15) & links(:,5) == packets(i,16) & ...
            links(:,7) == packets(i,12) & links(:,8) == packets(i,17)), 1) > 0)

         % decrease XL capacity by one
         links(find(links(:,2) == packets(i,2) &  links(:,3) == packets(i,3) & ...
            links(:,4) == packets(i,15) & links(:,5) == packets(i,16) & ...
            links(:,7) == packets(i,12) & links(:,8) == packets(i,17)), 1) = ...
            links(find(links(:,2) == packets(i,2) &  links(:,3) == packets(i,3) & ...
            links(:,4) == packets(i,15) & links(:,5) == packets(i,16) & ...
            links(:,7) == packets(i,12) & links(:,8) == packets(i,17)), 1) - 1;

         % leaving current node
         nodes(M*N*(1-packets(i,12))+M*(packets(i,2)-1)+ packets(i,3), 4) = ...
            nodes(M*N*(1-packets(i,12))+M*(packets(i,2)-1)+ packets(i,3), 4) + 1;
      else
         % do nothing; since crosslink is full
         packets(i,11) = 0;
      end
   end
```

```
      end
    end
  end


  for i = 1:MAX_NUM_PACKET
    if (packets(i,1) ~= 0) & (packets(i,11) ~= 0)

      % arriving at node
      if (nodes(M*N*(1-packets(i,17))+M*(packets(i,15)-1) + packets(i,16), 4) > 0)
        nodes(M*N*(1-packets(i,17))+M*(packets(i,15)-1) + packets(i,16), 4) = ...
          nodes(M*N*(1-packets(i,17))+M*(packets(i,15)-1) + packets(i,16), 4) - 1;
      else
        error('buffer reached capacity'); % assuming infinite buffer size
      end

      % increase link capacity by one; since traffic has arrived at node
      links(find(links(:,2) == packets(i,2) &  links(:,3) == packets(i,3) & ...
        links(:,4) == packets(i,15) & links(:,5) == packets(i,16) & ...
        links(:,7) == packets(i,12) & links(:,8) == packets(i,17)), 1) = ...
        links(find(links(:,2) == packets(i,2) &  links(:,3) == packets(i,3) & ...
        links(:,4) == packets(i,15) & links(:,5) == packets(i,16) & ...
        links(:,7) == packets(i,12) & links(:,8) == packets(i,17)), 1) + 1;

      % update packet information
      packets(i,2) = packets(i,15);
      packets(i,3) = packets(i,16);
      packets(i,12) = packets(i,17);
      % increase number of hop count by one
      packets(i,9) = packets(i,9) + 1;

      packets(i,11) = 0;


    end
  end

  for i = 1:MAX_NUM_PACKET
    if (packets(i,1) ~= 0)

      % check whether destination has been reached
      if (packets(i,2) == packets(i,6)) & (packets(i,3) == packets(i,7)) & (packets(i,12) == packets(i,14))

        % remove from packet list
        packets(i,1) = 0;
```

```
        % leaving current node
        nodes(M*N*(1-packets(i,12))+M*(packets(i,2)-1)+ packets(i,3), 4) = ...
           nodes(M*N*(1-packets(i,12))+M*(packets(i,2)-1)+ packets(i,3), 4) + 1;

        % can check time needed for completion
        % can put old packet into new array
      end

   end
end

for i = 1:MAX_NUM_PACKET
   if (packets(i,11) ~= 0)
      error('should have done switching')
   end
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% gen_routing_table.m
% script file to generate routing table for a 3X3 grid
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

M = 3;
N = 3;
routing_table = zeros(2*M*N,2*M*N,13);

% (M(x-1) + y)-th routing table = routing table of satellite nodes(x,y)
% (MN + M(x-1) + y)-th routing table = routing table of terrestrial nodes(x,y)

for i = 1:M
   for j = 1:N
      for k = 0:1
         routing_table(M*N*(1-k)+M*(i-1)+j,:, :) = router5([i, j, k], links);
      end
   end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% router5.m
```

```
% This is a function to generate routing table for a 3 X 3 system
% parameter: current_loc -- origin of the route
%            links -- data structure holding information of links
% returns a routing table of 9 entries
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function routing_table = router5(current_loc, links)

M = 3;
N = 3;

% map all points to a square grid
%
% 7 --- 8 --- 9
% |     |     |
% |     |     |
% 4 --- 5 --- 6
% |     |     |
% |     |     |
% 1 --- 2 --- 3

points = zeros(9, 3);

points(:,3) = current_loc(3);
points(5,:) = current_loc;

if (current_loc(1) == M)
   points(3,1) = 1;
   points(6,1) = 1;
   points(9,1) = 1;
else
   points(3,1) = current_loc(1) + 1;
   points(6,1) = current_loc(1) + 1;
   points(9,1) = current_loc(1) + 1;

end

if (current_loc(1) == 1)
   points(1,1) = M;
   points(4,1) = M;
   points(7,1) = M;
else
   points(1,1) = current_loc(1) - 1;
   points(4,1) = current_loc(1) - 1;
   points(7,1) = current_loc(1) - 1;
end
```

```
points(8,1) = current_loc(1);
points(2,1) = current_loc(1);

if (current_loc(2) == N)
   points(7,2) = 1;
   points(8,2) = 1;
   points(9,2) = 1;
else
   points(7,2) = current_loc(2) + 1;
   points(8,2) = current_loc(2) + 1;
   points(9,2) = current_loc(2) + 1;
end

if (current_loc(2) == 1)
   points(1,2) = N;
   points(2,2) = N;
   points(3,2) = N;
else
   points(1,2) = current_loc(2) - 1;
   points(2,2) = current_loc(2) - 1;
   points(3,2) = current_loc(2) - 1;
end

points(4,2) = current_loc(2);
points(6,2) = current_loc(2);


% populating routing table

routing_table = zeros(18, 13);

% 5 to 1
path1metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2) ...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(4,1) & links(:,5) == points(4,2) & links(:,8) == points(4,3)),1) +...
   links(find(links(:,2) == points(4,1) & links(:,3) == points(4,2) & links(:,7) == points(4,3)...
& links(:,4) == points(1,1) & links(:,5) == points(1,2) & links(:,8) == points(1,3)),1));

path2metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2) ...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(2,1) & links(:,5) == points(2,2) & links(:,8) == points(2,3)),1) + ...
   links(find(links(:,2) == points(2,1) & links(:,3) == points(2,2) & links(:,7) == points(2,3)...
& links(:,4) == points(1,1) & links(:,5) == points(1,2) & links(:,8) == points(1,3)),1));
```

```
if (path1metric > path2metric)
   routing_table(1,:) = [current_loc(1), current_loc(2), current_loc(3), points(1,1), ...
         points(1,2), points(1,3), 1, points(2,1), points(2,2), points(2,3), 0, 0, 0];
   routing_table(9+1,:) = [current_loc(1), current_loc(2), current_loc(3), points(1,1), ...
         points(1,2), ~points(1,3), ...
         2, points(2,1), points(2,2), points(2,3), points(1,1), points(1,2), points(1,3)];

else
   routing_table(1,:) = [current_loc(1), current_loc(2), current_loc(3), points(1,1), ...
         points(1,2), points(1,3), 1, points(4,1), points(4,2), points(4,3), 0, 0, 0];
   routing_table(9+1,:) = [current_loc(1), current_loc(2), current_loc(3), points(1,1), ...
         points(1,2), ~points(1,3), ...
         2, points(4,1), points(4,2), points(4,3), points(1,1), points(1,2), points(1,3)];

end

% 5 to 2
routing_table(2,:) = [current_loc(1), current_loc(2), current_loc(3), points(2,1), points(2,2), ...
      points(2,3), 0, 0, 0, 0, 0, 0, 0];
routing_table(9+2,:) = [current_loc(1), current_loc(2), current_loc(3), points(2,1), points(2,2), ...
      ~points(2,3), 1, points(2,1), points(2,2), points(2,3), 0, 0, 0];

% 5 to 3
path1metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2)...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(6,1) & links(:,5) == points(6,2) & links(:,8) == points(6,3)),1) + ...
   links(find(links(:,2) == points(6,1) & links(:,3) == points(6,2) & links(:,7) == points(6,3)...
   & links(:,4) == points(3,1) & links(:,5) == points(3,2) & links(:,8) == points(3,3)),1));

path2metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2)...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(2,1) & links(:,5) == points(2,2) & links(:,8) == points(2,3)),1) + ...
   links(find(links(:,2) == points(2,1) & links(:,3) == points(2,2) & links(:,7) == points(2,3)...
   & links(:,4) == points(3,1) & links(:,5) == points(3,2) & links(:,8) == points(3,3)),1));

if (path1metric > path2metric)
   routing_table(3,:) = [current_loc(1), current_loc(2), current_loc(3), points(3,1), points(3,2),...
         points(3,3), 1, points(2,1), points(2,2), points(2,3), 0, 0, 0];
   routing_table(9+3,:) = [current_loc(1), current_loc(2), current_loc(3), points(3,1), points(3,2),...
         ~points(3,3), ...
         2, points(2,1), points(2,2), points(2,3), points(3,1), points(3,2), points(3,3)];

else
   routing_table(3,:) = [current_loc(1), current_loc(2), current_loc(3), points(3,1), points(3,2),...
         points(3,3), 1, points(6,1), points(6,2), points(6,3), 0, 0, 0];
```

```
   routing_table(9+3,:) = [current_loc(1), current_loc(2), current_loc(3), points(3,1), points(3,2),...
         ~points(3,3), 2, points(6,1), points(6,2), points(6,3), points(3,1), points(3,2), points(3,3)];

end

% 5 to 4
routing_table(4,:) = [current_loc(1), current_loc(2), current_loc(3), points(4,1), points(4,2),...
      points(4,3), 0, 0, 0, 0, 0, 0, 0];
routing_table(9+4,:) = [current_loc(1), current_loc(2), current_loc(3), points(4,1), points(4,2),...
      ~points(4,3), 1, points(4,1), points(4,2), points(4,3), 0, 0, 0];



% 5 to 5
routing_table(5,:) = [current_loc(1), current_loc(2), current_loc(3), points(5,1), points(5,2),...
      points(5,3), 0, 0, 0, 0, 0, 0, 0];
routing_table(9+5,:) = [current_loc(1), current_loc(2), current_loc(3), points(5,1), points(5,2),...
      ~points(5,3), 1, points(5,1), points(5,2), points(5,3), 0, 0, 0];

% 5 to 6
routing_table(6,:) = [current_loc(1), current_loc(2), current_loc(3), points(6,1), points(6,2),...
      points(6,3), 0, 0, 0, 0, 0, 0, 0];
routing_table(9+6,:) = [current_loc(1), current_loc(2), current_loc(3), points(6,1), points(6,2),...
      ~points(6,3), 1, points(6,1), points(6,2), points(6,3), 0, 0, 0];



% 5 to 7
path1metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2)...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(8,1) & links(:,5) == points(8,2) & links(:,8) == points(8,3)),1) + ...
   links(find(links(:,2) == points(8,1) & links(:,3) == points(8,2) & links(:,7) == points(8,3)...
   & links(:,4) == points(7,1) & links(:,5) == points(7,2) & links(:,8) == points(7,3)),1));

path2metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2)...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(4,1) & links(:,5) == points(4,2) & links(:,8) == points(4,3)),1) + ...
   links(find(links(:,2) == points(4,1) & links(:,3) == points(4,2) & links(:,7) == points(4,3)...
   & links(:,4) == points(7,1) & links(:,5) == points(7,2) & links(:,8) == points(7,3)),1));

if (path1metric > path2metric)
   routing_table(7,:) = [current_loc(1), current_loc(2), current_loc(3), points(7,1), points(7,2),...
         points(7,3), 1, points(4,1), points(4,2), points(4,3), 0, 0, 0];
   routing_table(9+7,:) = [current_loc(1), current_loc(2), current_loc(3), points(7,1), points(7,2),...
         ~points(7,3), 2, points(4,1), points(4,2), points(4,3), points(7,1), points(7,2), points(7,3)];

else
```

```
    routing_table(7,:) = [current_loc(1), current_loc(2), current_loc(3), points(7,1), points(7,2),...
          points(7,3), 1, points(8,1), points(8,2), points(8,3), 0, 0, 0];
    routing_table(9+7,:) = [current_loc(1), current_loc(2), current_loc(3), points(7,1), points(7,2),...
          ~points(7,3), 2, points(8,1), points(8,2), points(8,3), points(7,1), points(7,2), points(7,3)];
end


% 5 to 8
routing_table(8,:) = [current_loc(1), current_loc(2), current_loc(3), points(8,1), points(8,2),...
      points(8,3), 0, 0, 0, 0, 0, 0, 0];
routing_table(9+8,:) = [current_loc(1), current_loc(2), current_loc(3), points(8,1), points(8,2),...
      ~points(8,3), 0, points(8,1), points(8,2), points(8,3), 0, 0, 0];

% 5 to 9
path1metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2)...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(6,1) & links(:,5) == points(6,2) & links(:,8) == points(6,3)),1) + ...
   links(find(links(:,2) == points(6,1) & links(:,3) == points(6,2) & links(:,7) == points(6,3)...
   & links(:,4) == points(9,1) & links(:,5) == points(9,2) & links(:,8) == points(9,3)),1));

path2metric =  (links(find(links(:,2) == current_loc(1) & links(:,3) == current_loc(2)...
   & links(:,7) == current_loc(3)...
   & links(:,4) == points(8,1) & links(:,5) == points(8,2) & links(:,8) == points(8,3)),1) + ...
   links(find(links(:,2) == points(8,1) & links(:,3) == points(8,2) & links(:,7) == points(8,3)...
   & links(:,4) == points(9,1) & links(:,5) == points(9,2) & links(:,8) == points(9,3)),1));

if (path1metric > path2metric)
   routing_table(9,:) = [current_loc(1), current_loc(2), current_loc(3), points(9,1), points(9,2),...
         points(9,3), 1, points(8,1), points(8,2), points(8,3), 0, 0, 0];
   routing_table(9+9,:) = [current_loc(1), current_loc(2), current_loc(3), points(9,1), points(9,2),...
         ~points(9,3), 2, points(8,1), points(8,2), points(8,3), points(9,1), points(9,2), points(9,3)];
else
   routing_table(9,:) = [current_loc(1), current_loc(2), current_loc(3), points(9,1), points(9,2),...
         points(9,3), 1, points(6,1), points(6,2), points(6,3), 0, 0, 0];
   routing_table(9+9,:) = [current_loc(1), current_loc(2), current_loc(3), points(9,1), points(9,2),...
         ~points(9,3), 2, points(6,1), points(6,2), points(6,3), points(9,1), points(9,2), points(9,3)];
end

return
```

Bibliography

[1] W.S. Adams and L. Rider. Circular polar constellations providing continuous single or multiple coverage above a specified latitude. *Journal of the Astronautical Sciences*, 35(2):155–192, Apr - Jun 1984.

[2] I. F. Akyildiz and S. Jeong. Satellite atm networks: A survey. *IEEE Communications Magazine*, 35(7):30–34, July 1997.

[3] G. Albertengo et al. Optimal routing algorithms for the bidirectional manhattan street network. *ICC '91, Conference Record. IEEE International Conference on Communications, 1991*, 3:1676–1680, 1991.

[4] Jihwan Patrick Choi. Channel prediction and adaptation over satellite channels with weather-induced impairments. Master's thesis, Massachusetts Institute of Technology, 2000.

[5] L.P. Clare. A traffic analysis of satellite constellation networks. *Proceedings of IEEE Infocom '87*, pages 918–925.

[6] C.E. Fossa et al. An overview of the iridium low earth orbit satellite system. *Proceedings of the IEEE Aerospace and Electronics Conference, NAECON*, pages 152–159, 1998.

[7] David D. Friedman. *Price theory: an intermediate text.* South-Western Publishing Co., Cincinnati, Ohio, second edition, 1990.

[8] G. Gilder. The bandwidth tidal wave. *Forbes ASAP*, 5 December 1994. http://www.forbes.com/asap/gilder/telecosm10a.htm.

[9] Louis J. Ippolito. *Radiowave propagation in satellite communications.* Van Nostrand Reinhold Co., New York, NY, 1986.

[10] Abbas Jamalipour. *Low Earth Orbital Satellites for Personal Communication Networks.* Artech House Mobile Communications Library. Artech House, Norwood, MA, 1998.

[11] Christopher Karpinsky. Uplink multiple access techniques for satellite communication systems. Master's thesis, Massachusetts Institute of Technology, 1998.

[12] Robert Larson. *Principles of Dynamic Programming. Part I. Basic Analytic and Computational Methods.*, volume 7 of *Control and Systems Theory.* Marcel Dekker, Inc., New York, New York, 1978.

[13] W. Leland et al. On the self-similar nature of ethernet traffic. *IEEE/ACM Transactions on Networking*, 2, February 1994.

[14] D.P. Patterson. Teledesic: a global broadband network. *IEEE Aerospace Conference*, 4:547–552, 1998.

[15] V. Paxson and S. Floyd. Wide-area traffic: the failure of poisson modeling. *Proceedings of ACM SIGCOMM '94*, February 1994.

[16] M. Richharia et al. A feasibility study of a mobile communication network using a constellation of low earth orbit satellites. *IEEE Global Telecommunications Conference and Exhibition. Communications Technology for the 1990s and Beyond. GLOBECOM '89.*, 2:773–777.

[17] D. Roddy. *Satellite Communications.* Prentice-Hall, Englewood Cliffs, NJ, 1989.

[18] J.G. Walker. Satellite constellation. *Journal of British Interplanetary Society*, 37:559–571, 1984.

[19] M. Werner et al. Analysis of system parameters for leo/ico-satellite communication networks. *IEEE Journal on Selected Areas in Communications*, 13(2):371–381, 1995.

[20] Lloyd Wood. Network performance of non-geostationary constellations equipped with intersatellite links. Master's thesis, University of Surrey, 1995.