



Asia-Pacific Economic Cooperation

APEC Secretariat

35 Heng Mui Keng Terrace Singapore 119616

Tel: (65) 6775 6012 Fax: (65) 6775 6013 Email: info@mail.apecsec.org.sg

APEC#203-TC-01.2

ISBN: 981-04-9238-3



Asia - Pacific Economic Cooperation

Optimal Topology of Testbeds and Simplified Commercial Networks in the APEC Region



Development Cooperation Steering Group
APEC Telecommunications and Information Working Group

2003



Asia-Pacific Economic Cooperation

Optimal Topology of Testbeds and Simplified Commercial Networks in the APEC Region

Jae - Whak Roh

Hansung University, Korea

Development Cooperation Steering Group
APEC Telecommunications and Information Working Group

2003

2003

Publication numbers: APEC#203-TC-01.2

ISBN: 981-04-9238-3.

Copyright © 2003 APEC Secretariat

Requests and inquiries concerning reproduction
and rights should be addressed to:

Asia-Pacific Economic Cooperation Secretariat

35 Heng Mui Keng Terrace

Singapore 119616

telephone: +65 6775 6012

facsimile: +65 6775 6013

email: info@mail.apecsec.org.sg

Foreword

The proliferation of the Internet has brought about countless changes to our lives in various sectors, and we now go so far as to say that our future depends upon the level of our Internet proficiency. As such, no one today doubts about the importance of the usage and spread of the Internet.

The problem however is the gap of technology level according to different regions, known as the digital divide. The spread of Internet particularly in the Asian region is relatively slow compared to other regions.

Behind the tardy spread of the Internet lies the issue of high cost. And this is truer in case of costly international circuits, which is the reason why we need to seek measures to promote the expansion of the Internet while keeping the cost down to the bear minimum. There are two measures that can be considered in order to reduce cost: One is optimizing the circuit configuration and second is taking advantage of the existing regional nodes.

But this is not a very simple issue: Configuring an effective regional backbone requires countries to overcome the barriers of their physical boundaries that existed in the past.

In the end, the goal of this research is to overcome the national boundary, configure effective Internet backbone through international cooperation, reduce cost and finally develop regional internet connectivity.

To achieve such goal, the research was conducted with a special focus on its cost and we hope to contribute to the development of the Internet in this region through the proposal of an effective network configuration.

The study consisted of two phases including the collection of data from various countries and relevant computer simulations.

Collecting data throughout the study posed many difficulties and data which was not updated was sometimes used, as the term of the study got extended. Computer simulations were also delayed owing to the simultaneous implementation of both new and traditional methodologies.

Throughout the terms of the study, many people have extended their support and we would like to express our deepest appreciations for Bunwhan Lee and Kyungsik Lee. We also appreciate for those who have extended indirect support including DCSG participants, APEC TEL secretariat and MIC, Korea.

Last but not least, we would like to reiterate our sincerest hope that such study will contribute to the development of the Internet in this region.

Prof. Jae-whak Roh

Hansung University, KOREA

and

Chair

Development Cooperation Facilitation Steering Group

APEC Telecommunications and Information Working Group

Table of Contents

Foreword	
Table of contents	
List of figures	
List of tables	
Executive Summary	
Chapter 1. Introduction	1
Chapter 2. Collecting Data	4
2.1: BRUNEI DARUSSALAM	4
2.2: CHINA	5
2.3: HONG KONG, CHINA	9
2.4: INDONESIA	11
2.5: JAPAN	14
2.6: KOREA	20
2.7: MALAYSIA	23
2.8: PHILIPPINES	27
2.9: SINGAPORE	29
2.10: CHINESES TAIPEI	31
2.11: THAILAND	34
2.12: VIET NAM	37
Chapter 3. Multi-Connectivity Model	40
3.1: Introduction of Multi-connectivity Model	40
3.2: Multi-connectivity Model and Solution	43
3.3: Multiple City Case in Multi-Connectivity Model	45
3.4: Multi connectivity & Capital City Scenario	51
Chapter 4. Combined Model	55

4.1: The significance of the combined model	55
4.2: Structuring Combined Model and Solution	56
4.3: Combined Model & Capital City Scenarios	59
4.4: Combined Model & Multiple City Scenarios	51
Chapter 5. Conclusions	63

List of figures

<Figure 1> China's domestic-connection	5
<Figure 2> Connection of China's CERNET	6
<Figure 3> Backbone of CERNET	6
<Figure 4> China's CSTnet	7
<Figure 5> Japan's domestic connectivity	15
<Figure 6> Network of JGN	15
<Figure 7> Network of SINET	16
<Figure 8> Selection of cities in Japan	18
<Figure 9> International Linkage Status of KOREN	20
<Figure 10> Network of TEMAN	23
<Figure 11> Domestic inter-connection of Thailand	34
<Figure 12> Domestic connectivity in Thailand (2)	34
<Figure 13> Domestic inter-connection	37
<Figure 14> example of an optical cables design	42
<Figure 15> Optical cable design that satisfies network connectivity level	42
<Figure 16> Multi-Connectivity Model-Multiple cities-N.E. Asia3-S.E. Asia2-Singapore2-all special nodes	47
<Figure 17> Multi-Connectivity Model-Multiple Cities-N.E. Asia3-S.E. Asia2-Singapore2- Partial Special node	47
<Figure 18> Multi-Connectivity Model-Multiple Cities-N.E. Asia3-S.E. Asia2-Singapore2-All Special Node	48
<Figure 19> Multiple-connectivity Model-Multiple Cities-N.E. Asia3-S.E. Asia2-Singapore2-	

Partial Special Nodes	49
<Figure 20> Multi-connectivity Model-Multi Cities- N.E. Asia3-S.E. Asia2-Singapore1-all Special nodes	50
<Figure 21> Multi-connectivity Model-Multi-Cities-N.E. Asia3-S.E. Asia2-Singapore1-Partial Special Nodes	50
<Figure 22> Multi-Connectivity Model-Multiple Cities-N.E. Asia2-S.E. Asia1-Singapore1-All Special Nodes	50
<Figure 23> Multi-Connectivity Model -Multiple Cities-N.E. Asia2-S.E. Asia1-Singapore1- Partial Special nodes	51
<Figure 24> multi-connectivity model-Capital City Scenarios-all special nodes	53
<Figure 25> Multi-Connectivity Model-Capital City Scenarios-Korea; China; Japan Special Nodes	53
<Figure 26> Multi-connectivity Model-Capital City Scenarios-Korea; Japan; China and Singapore special Nodes	54
<Figure 27> Combined Model-13 capitals (13 economies such as Korea; China; Japan; Chinese Taipei; Hong Kong, China; China; Philippines; Viet Nam; Thailand; Malaysia; Indonesia; Brunei Darussalam; Singapore; and United States)	59
<Figure 28> Combined Model-12 Capitals (Korea; China; Japan; Chinese Taipei; Hong Kong, China; China; Philippines; Viet Nam; Thailand; Malaysia; Indonesia; Brunei Darussalam; Singapore).....	60
<Figure 29> Combined Model-2 N.E. Asia-1 S.E. Asia.	62
<Figure 30> Combined Model-2 S.E. Asia Only	62
<Figure 31> Comparison of Capital Scenarios	68
<Figure 32> Frequently Recurring Figures	69
<Figure 33> Network topologies in S.E. Asia	69

List of Tables

<Table 1> Brunei Darussalam's International PSTN Traffic Volume, million minutes data from International traffic, ITU, 1999	5
<Table 2> Submarine Cables Landing in China.....	8
<Table 3> China's International PSTN Traffic Volume, million minutes	9
<Table 4> Submarine Cables Landing in Hong Kong, China	10
<Table 5> Hong Kong, China's International PSTN Traffic Volume, million minutes	11
<Table 6> Submarine Cables Landing in Indonesia	12
<Table 7> Population of Major Cities in Indonesia	13
<Table 8> Indonesia's International Traffic Volume of PSTN, million minute	14
<Table 9> Submarine Cables Landing in Japan	17
<Table 10> Population of Major Cities in Japan	18
<Table 11> Japan's International PSTN Traffic Volume, million minutes	19
<Table 12> Submarine Cables Landing in Korea	21
<Table 13> Population of Major Cities in Korea	22
<Table 14> Korea's International PSTN Traffic Volume, million minutes Data from: International traffic Index, ITU, 1999	23
<Table 15> Submarine Cables Landing in Malaysia	24
<Table 16> Population of Major Cities in Malaysia	25
<Table 17> Malaysia's International PSTN Traffic Volume, million minutes	26
<Table 18> Submarine Cables Landing in the Philippines	28

<Table 19> Population of Major Cities in the Philippines	28
<Table 20> Philippines' International PSTN traffic volume, million minutes	29
<Table 21> Submarine Cables Landing in Singapore	30
<Table 22> Singapore's International PSTN Traffic Volume, million minutes	31
<Table 23> Submarine Cables Landing in Chinese Taipei	32
<Table 24> Chinese Taipei's International PSTN Traffic Volume, million minutes	33
<Table 25> Submarine Cables Landing in Thailand	35
<Table 26> Population of Major Cities in Thailand	35
<Table 27> Thailand's International PSTN Traffic Volume, million minutes	36
<Table 28> Submarine Cables Landing in Viet Nam	38
<Table 29> Major Cities in Viet Nam	38
<Table 30> Viet Nam's International PSTN Traffic volume, million minutes	39
<Table 31> Nodes considered in Multiple City Case	45
<Table 32> Basic Scenarios	46
<Table 33> Capital City Scenarios	52

Executive Summary

1. Title

Study on the Optimal Topology of Testbeds (research networks) and Simplified Commercial Networks in the APEC Region.

2. Objectives and Importance of the Research

The purpose of this study is to find the optimal topology of Asia-Pacific regional networks with special considerations to its constructing cost, a work that we find necessary for many reasons. First, this research will provide ways to overcome the digital divide between regions. Recently, there have been discussions between North America and Europe, which focused on building a huge broadband network on the Tera bit level, while some Asian members are still stuck with very low bandwidths. It is not hard to figure out that such slow development in the Asia-Pacific region comes mainly from the region's meager financial resources. This research will provide commendable ways to overcome the digital divide by presenting the most optimal topology with minimum investment and competitive network.

Under the circumstances where the Asian members must carry the burden of paying the full circuit amount of the traffic to United States, while North Americans enjoy the free ride sharing the same circuits, and each Asian member nevertheless wishes or have no choice but to establish network links with the United States, we need to think of ways to save the regional cost burden by constructing an optimal network design which

will help Asian members avoid extra costs spent when installing it individually. This strategy would save our region from spending a huge amount of financial resources.

Moreover, regions including Europe and North America are considering the building of regional networks, which cover the entire region either for commercial or research purposes.

For all these reasons and more, it can be concluded that the faster we pursue research of this kind, the better we will be in garnering the competitive edge we need in the new digital era. And if not done now, we may always have to settle for the second best.

3. Contents and Scope of the Research

The final target of this research is establishing the topologies of Asia-Pacific region. To this end, we need to select target cities from each member and the selected cities need to be connected in a most economical way. Surely, selecting cities alone is not an easy task. It needs to consider many factors such as internal connectivity with domestic networks and demand size based on population. The accessibility of the submarine cable must also be considered. In the end, two or three candidate cities from each member economies are selected in this study for future network design.

The next issue is how we should design this network. Two ways are adopted for this purpose; the traditional way and new creative way. In chapter 3, traditional ways for constructing optical cable are adopted and 17 figures are produced as results. Traditional way here means the methodology to design networks considering only the multi-connectivity, putting priority on nodes and the supply of adequate bandwidth of each arch.

In chapter 4, new creative ways are introduced, which included the concept of both routing and multi-connectivity: Seven figures are produced as results. For the purpose

of reflecting the goal of this research, a new methodology, that takes into account expected traffic volume, is adopted in the hope that this will produce better results.

4. Research Results

When the traditional way of designing the optical cable is adopted, the result, in a nutshell, revealed that the optimal way to connect all members would be to connect them through two different rings. Those results are illustrated in Chapter 3.

The new creative way where routing and multi-connectivity are considered simultaneously is demonstrated in Chapter 4. It can be summarized in a nutshell that the optimal way of constructing the new creative network is building two-rings: South East and North East Asian ring connected through Chinese Taipei.

In the conclusion of Chapter 5, the two different results are compared and why they produced different results is analyzed. And from this, the general conclusions are drawn.

5. Policy Suggestions for Practical Use

Recently, Trans Eurasia Information Network, and APII (Asia Pacific Information Infrastructure) testbed have become one of the most important policies in Korea & AP region. The results of this research are expected to help related policies, and furthermore, be adopted in some cases. They will also hopefully help provide the necessary answers for such policy framework. For now, these results will only be applied to the policy framework of APII, and TEIN. We would definitely need more policies like this which are based on scientific analysis and can help lead the policies to be more effective.

Chapter 1

Introduction

The aim of this study is to establish the most cost-effective network with a particular focus on the Asia Pacific Economic Cooperation regions. In many cases, designing a network involves the work of an entire economy because it means connecting nodes effectively not only within a economy but also with other foreign partners.

When an economy tries to establish an international link, its usage, be it commercial or research purpose, matters little: What counts are efficiency and cost reduction, with the guarantee of its performance. With the recent surge of importance in establishing an effective regional network, this study seeks to explain the reasons for supporting the building of such network.

First reason for upholding the building of an effective international network can be explained as part of an effort to overcome the deepening digital divide. Most Asian economies are still at a developing stage and have yet to secure sufficient bandwidth if they are to compete with the advanced economies. The digital divide issue will become more apparent once Europe and North America strikes a deal on the current discussion on Terabit size networks. To cope with these circumstances, Asian members, which lack the resource, must put an effort to secure a required broadband through regional cooperation. This research will help provide directions to this end. Secondly, individual economies that generally designed their own networks can reduce the financial burden by sharing the work and the cost. While nation-wide links are tailored to individual economies, international network links are established separately through bilateral negotiations. Therefore, international networks is a complex mixture of various configurations and standards that differed according to each economy, which neither had a broader regional perspective nor took into consideration regional characteristics when designing a network. It is evident that this kind of approach will not produce a prospective result for the future. In order to simplify and enhance the network, which is particularly important for the

developing Asian members that lack resources, it is necessary to build a international or regional network where cost can be shared.

Thirdly, the Asian region will be able to manage a more lean network operation through international cooperation. Currently, Asian members are playing a less competitive game performing on their own. For example, Asian members that wish to establish connections with the U.S. generally start building a new direct link without giving considerations to an exiting regional detouring pass that could be more economical. In this case, the cost is also paid entirely by the Asian counterpart instead of being split 50 to 50 with the U.S., as is the general rule.

At the moment, Korea is leading a discussion on the regional network design for the Trans Eurasia Information Network and negotiations concerning this project are well underway. As such, an effective efficient network design or an expansion of the Asia Pacific Information Infrastructure has also emerged as an important issue in the Asian region.

This research starts by collecting data from each member economy. Therefore, Chapter 2 will touch upon the process of data collection. Data related to submarine cables, domestic infrastructure, population size, and the penetration rate of PSTN will be taken into account in selecting 2 to 3 final candidate cities of each member economy.

First, each economy's present research on networks will be reviewed in details in an attempt to identify their current network status and check domestic connectivity. Second, the submarine cable environment including landing stations will be reviewed in addition since the accessibility of submarine cables is one of the determining factors when selecting adequate cities for this project. Third, the population of each major city will also be considered as additional data in order to help the selection of cities. Finally, a comprehensive decision will be made based on the three factors mentioned above.

Referring back to the final aim of this project, we will make an assumption that the selected cities of each member are then connected via submarine cables.

Several other assumptions will also be applied to simplify our work. For instance, all APEC member economies are connected only via submarine cables and no other terrestrial cables or satellite systems are involved. When calculating the cost, different costs of each economy are not reflected. Instead, the cost function of SEA-ME-WE is adopted. Also, no factors other than distance will be considered in the cost calculation. Last assumption is to consider the technological level of each member economy to be the same.

Several new attempts will be also made in finding the optimal solution. Traditionally, only the Multi-Connectivity model is considered when designing a network. In this kind of model, a node's importance value is assigned according to the importance of nodes. And next, a network is designed to satisfy these required connectivity importance levels. In chapter 3, we will experiment the above-mentioned traditional method to a full extent. However, this type of

model is rather inappropriate for the purpose of our study, since it presupposes a sufficient supply of bandwidth, something that would not be feasible for Asian economies with limited financial resources. Therefore, a new creative method that predicts the traffic quantity between members and designs a network based on this prediction will be adopted in this study. Chapter 4 illustrates this method in more details.

Chapter 5 will include a comparison between the results of the study, along with a final conclusion. There will also be a comparison between the results of each experiment and of experiments performed under similar assumptions. This study will cover some diagrams as well and if some of them appear frequently despite different assumptions, it will consider them solid results.

The conclusion of this research will provide the overall vision of an effective Asian regional network and such vision is expected to fully contribute to the development of regional infrastructure.

Chapter 2

Data of Each Member

Recalling the final aim of this project, the first step to be taken should be selecting adequate cities. In this chapter, to select candidate cities, data regarding submarine cables, domestic infrastructure, population size, and the penetration rate of PSTN are considered and next, two or three candidate cities from each member are selected. The current network status and domestic connectivity are also reviewed in details and in addition, the submarine cable environment, including the landing station, is reviewed. Thirdly, the population of each major city is considered to help the selection of the cities. For final decision a comprehensive decision is made based on the three factors mentioned above. Through these selection processes, two or three candidate cities for each member economy are finally selected.

2.1. BRUNEI DARUSSALAM

2.1.1 Research Network

Data is not needed

2.1.2 Submarine Cables

Data is not needed

2.1.3 Population

Data is not needed

2.1.4 Selection

Data is not needed

2.1.5 Traffic Data

The data below shows the international PSTN traffic between Brunei and APEC members in million minutes. The size of the traffic volume is relatively small compared to other economies, with Malaysia ranking first and Singapore ranking second.

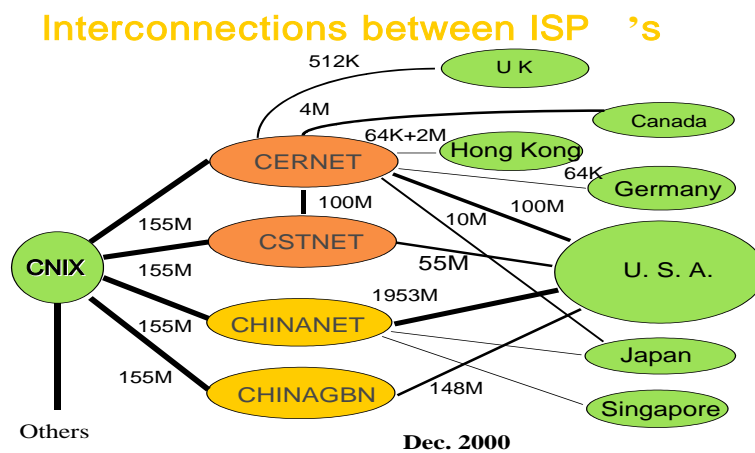
	Brunei Darussalam
China	0.05
Hong Kong, China	0.225
Indonesia	0.275
Japan	0.175
Korea	0
Malaysia	7.55
Philippines	1.75
Singapore	5.975
Chinese Taipei	0.025
Thailand	0.15
Viet Nam	0
United States	0.6
Canada	0.175
Australia	0.775
New Zealand	0.15

<Table 1> Brunei Darussalam’s international PSTN traffic volume, million minutes
Data from International traffic, ITU, 1999

2.2. CHINA

2.2.1 Research Network

As is shown in figure 1, all four important ISPs are connected to CNIX.



<Figure 1> China’s domestic-connection

China Education and Research network, which only had the capacity of between 2M and 155M in

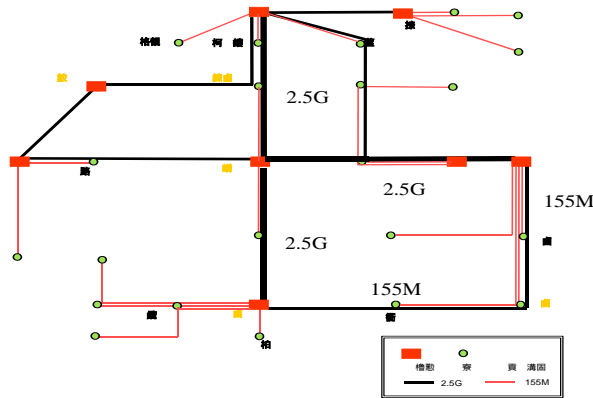


<Figure 2> Connection of China's CERNET

1999, has been upgraded in 2001 to 36 connecting cities and 800 universities through a 2.5G/155M backbone. Furthermore, CERNET2 was initiated very recently.

The following figure shows the connectivity of CERNET.

CERNET Backbone at 2.5G Mbps /155 Mbps

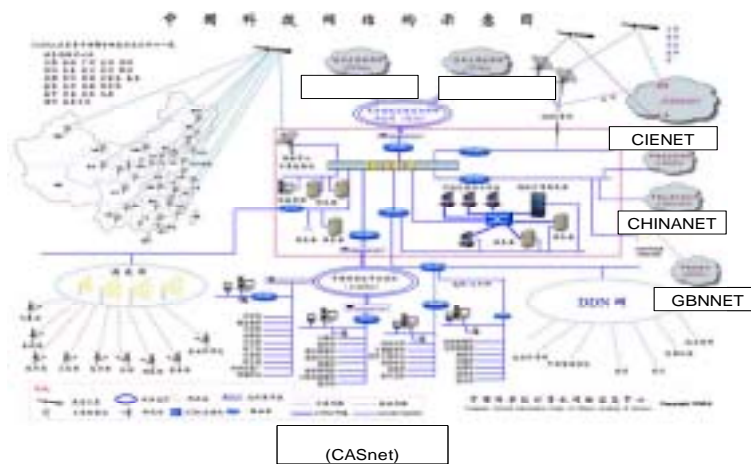


<Figure 3> Backbone of CERNET

CSTnet is a nationwide super computer network established and developed by CASnet and its foundation, the Computer Network of Chinese Academy of Sciences. It is one of the four major networks in China that received the first government authorization. It is a non-commercial, public-oriented network that is the basis of China's national information innovation program. CSTNET's main field of businesses includes science technology, science technology management, Government, and high tech companies. Its main businesses include network

communication, domain name registration, information data and supercomputer-based research. CSTnet is connected to over 1000 science research centers and science technology companies including the Chinese Academy of Science, the national natural science funding committee and other high-tech companies. It is also connected to research institutions of agriculture, forestry, earthquakes, weather, railway, electricity, electronics, aviation and environment of 45 cities nationwide. Currently CSTNET has 400,000 users that log on to CSTNET for science databases, science technology result data, science technology management, technology data, agriculture resources and documents and hundreds of GB database. Also through its homepage, CSTNET is providing various science technology information businesses and at the CSTNET super computer center, there are several supercomputers.

After reviewing two Chinese domestic research networks, it could be presumed that cities connected to the present research network and located on the coastline such as Beijing/Tianjin, Shanghai, Guangzhou and Shenyang can be considered as strong candidates in terms of domestic connectivity. As the next step, whether these cities have easy landing accessibility for submarine cables will be examined.



<Figure 4> China's CSTnet

2.2.2 Submarine Cables

CKC, CUCN, SMW, FLAG are submarine cables that have landing stations in China. SMW and FLAG have landing stations near Shanghai. CKC has a landing station in Qingdao and SWM has access stations in Shantou and Taipa, while CUCN has a landing point in Shangmin. All this information regarding landing stations will be used in a comprehensive way in selecting the cities at the next step.

	China
Korea	CKC((560M) CUCN(80G) SMW3(40G) FLAG(10G)
Japan	FLAG(10G) CUCN(80G) SMW3(40G)
Hong Kong, China	SMW3(40G) FLAG(10G)
Philippines	SMW3(40G)
Viet Nam	SMW3(40G)
Thailand	Not available
Malaysia	SMW3(40G) FLAG(10G)
Singapore	SMW3(40G)
Indonesia	SMW3(40G)

<Table 2> Submarine cables landing in China.

2.2.3 Population

Not needed

2.2.4 Selection

Major cities within the Chinese research network that have landing stations can be found on the east coast of China. Guangzhou, near Hong Kong, China, Shanghai, Beijing and Tianjin are all cities that are situated on the coastline and have access to CERNET. Although Beijing is not located directly on the coast it is still considered to be a competitive candidate as it is reasonably close to the coastline. It is also found that, in China's case, all three cities above have proper landing access-abilities. The fact that Beijing, Shanghai and Guangzhou each can be ranked high in terms of population is also a very favorable factor. Therefore, Beijing will be the first choice, Shanghai second and Guangzhou third.

2.2.5 Traffic Data

The data above shows the international PSTN traffic volume between China and APEC members in million minutes. The ranking according to Traffic volume is 1.Hong Kong, China;2.United States; 3.Thailand; 4.Japan; and 5.Korea. It can be found that considering

	China
Brunei Darussalam	0.05
Hong Kong, China	956.25
Indonesia	4.675
Japan	161.45
Korea	107.5
Malaysia	8.025
Philippines	4.075
Singapore	42.5
Chinese Taipei	174.45
Thailand	6.95
Viet Nam	1
United States	218.5
Canada	6.5
Australia	27.5
New Zealand	2.5

<Table 3> China's international PSTN traffic volume, million minutes
Data from: International Traffic Index, ITU, 1999

China's unique relationship with Hong Kong, China; Northeast Asian members such as Japan and Korea make up a considerable part of the communications with China.

2.3. HONG KONG, CHINA

2.3.1 Research Network

Harnet is a wide area network, which connects the campus network of seven universities in Hong Kong, China. HARNET is operated by UPCC (The University and Polytechnic Computer Centre) and JUCC (Joint University of the Computer Centers). The JUCC members include the University of Hong Kong, the Chinese University of Hong Kong, the Hong Kong Polytechnic, the City Polytechnic of Hong Kong, the Hong Kong Baptist College, the Hong Kong University of Science and Technology, Lingnan College. While JUCC provides services for its seven members, UPCC provides network connection, along with Hong Kong, China's "Open learning institute" and other institutions including neighboring universities of Macao.

In conclusion, Hong Kong, China's research network is confined to a small geographical area and therefore Hong Kong, China can be considered as having one landing point rather than

multiple landing points.

2.3.2 Submarine Cables

It was found that APCN, HJK, SMW, FLAG land in Hong Kong, China.

	Hong Kong, China
Korea	APCN(2560G) HJK(280M) SMW3(40G) FLAG(10G)
4Japan	APCN(2560G) APC(20G) SMW3(40G) FLAG(10G) HJK(280M)
China	SMW3(40G) FLAG(10G)
Chinese Taipei	APCN(2560G) SMW3(40G)
Philippines	APCN(2560G) SMW3(40G)
Viet Nam	SMW3(40G)
Thailand	APCN(2560G)
Malaysia	SMW3(40G) APC(20G) APCN(2560G)
Singapore	SMW3(40G) APC(20G)
Indonesia	SMW3(40G) APCN(2560G)

<Table 4> Submarine cables landing in Hong Kong, China

2.3.3 Population

Not needed

2.3.4 Selection

As Hong Kong, China has a relatively small geographical area, it would be appropriate to consider Hong Kong, China as one landing point rather than considering it as multiple access points.

2.3.5 Traffic Data

The data below shows the international PSTN traffic between Hong Kong, China and other APEC members in million minutes. In terms of the amount of traffic, China, which has a unique relationship with Hong Kong, China, has the greatest figure, with the United States ranking second. The Philippines;Thailand and Japan also appear to have significance.

	Hong Kong, China
Brunei Darussalam	0.225
China	956.25
Hong Kong, China	0
Indonesia	14.525
Japan	52.775
Korea	29.75
Malaysia	18.5
Philippines	45.625
Singapore	66
Chinese Taipei	62.975
Thailand	18.5
Viet Nam	5
United States	367.25
Canada	62.25
Australia	46
New Zealand	8.5

<Table 5> Hong Kong, China's International PSTN traffic volume, million minutes

2.4. INDONESIA

2.4.1 Research Network

The network of IPTEKnet consists of 5 main nodes and each is connected through terminals, which are used via telephone lines, radio, or satellite. The purpose of IPTEKnet is to

use the Nusantara 21, which is the central backbone of the economy. Currently there are 5 IRNs, such as BPPT (Jakarta), ITB (Bandung), UGM (Yogyakarta), ITS (Surabaya) and Unhas (Ujung Padang). Revision focusing on node coverage reveals that most of the cities are located in Pulau Jawa, making the connection between Jakarta, Bandung, Semarang and Surabaya the core backbone. Taking this into account, Jakarta and Surabaya are likely candidate cities for this project as they are both coastal cities and part of the core backbone.

2.4.2 Submarine Cables

Submarine cables such as SMW, APCN land in Indonesia.

	Indonesia
Korea	SMW3(40G) APCN(2560G)
Japan	SMW3(40G) APCN(2560G)
China	SMW3(40G)
Chinese Taipei	SMW3(40G) APCN(2560G)
Hong Kong, China	SMW3(40G) APCN(2560G)
Philippines	APCN(2560G) SMW3(40G)
Viet Nam	SMW3(40G)
Thailand	APCN(2560G)
Malaysia	SMW3(40G) APCN(2560G)
Singapore	SMW3(40G) APCN(2560G)

<Table 6> Submarine Cables landing in Indonesia

Fortunately, two cables (APCN and SMW) land near Jakarta. This naturally makes Jakarta a suitable candidate for our selection that satisfies access-ability and domestic connectivity condition. If an additional selection were to be made, Malang and Surabaya would be the likely candidates. However, it should be remarked that at present, both cities do not have appropriate landing accessibilities.

2.4.3 Population size

The following table refers to the population size of major cities.

City	Pop. 2002
Jakarta	10,810.4
Bandung	2,884.3
Surabaya	2,729.3
Medan	2,059.3
Palembang	1,462.3
Tangerang	1,393.9
Ujungpandang	1,170.2
Semarang	1,134.7
Bekasi	900.7
Bogor	761.3
Malang	742.7

<Table 7> Population of major cities in Indonesia

It can be noted that the most appropriate city for this project is Jakarta in terms of population size, the implicit demanding size of data traffic. Bandung will be considered as the next appropriate city in terms of city population. A unit created by connecting the cities between, connecting cities of Surabaya and Malang forms a geographically close and relatively important region, considering the implicit demand size. Therefore, Surabaya and Malang can be the third option for this project. As three of the most important areas are located on the Jawa Island, it is acceptable to exclude other islands from the selection.

2.4.4 Final Selection of Cities

The final selection will be made by considering all above-mentioned three information: domestic connectivity, submarine accessibility, and population size. Considering Indonesia's research network, Jakarta, Bandung, Semarang and Malang/Surabaya are the likely candidates for this project since they satisfy the all three conditions. The first choice would be Jarkarta where all three selected conditions are outstanding. The second choice would be Surabaya/Malang which also satisfies the three conditions considered here. As for Bandung and Semarang, they must be connected through terrestrial research networks because of their location and are therefore inadequate for final selection.

2.4.5 Traffic Data

The following data shows the amount of international PSTN traffic between Indonesia and APEC member economies in million minutes. It shows that Malaysia; the United States and Australia are important in terms of traffic volume.

	Indonesia
Brunei Darussalam	0.275
China	4.675
Hong Kong, China	14.525
Japan	29.1
Korea	13.325
Malaysia	30.675
Philippines	4.575
Singapore	76.2
Chinese Taipei	13.7
Thailand	4.3
Viet Nam	0.3
United States	29.85
Canada	1.95
Australia	34.325
New Zealand	1.85

<Table 8> Indonesia's international traffic volume of PSTN, million minutes
Data from: International Traffic Index, ITU, 1999

2.5. JAPAN

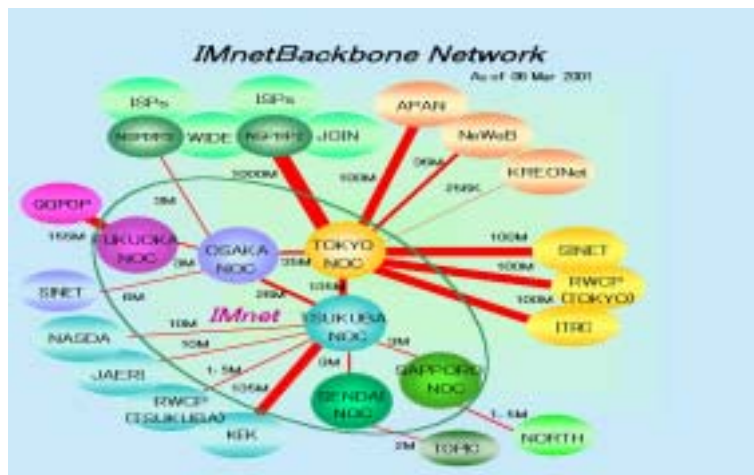
2.5.1 Research Network

Japan's main research network consists of JGN, WIDE, AI3, SINET, IMNET, RWCP, and more. These networks are all linked through IMNET. The following Figure shows the structure of the network.

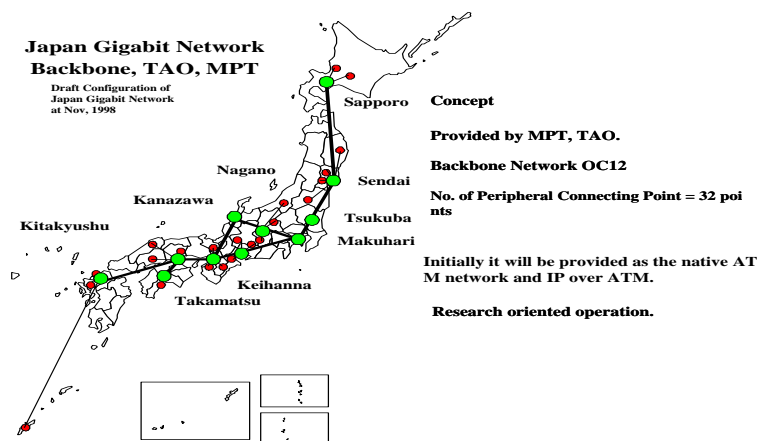
JGN is a network that is connected to universities, research institutions, enterprises, administrative organs and local governments. It aims to encourage the research and development that will eventually contribute to the early establishment of an optical-fiber-based society. The main cities of JGN are Sapporo, Sendai, Tsukuba, Nagano, Kanazawa, Kitakyushu, Takamatsu, Makuhari, and Keihanna.

SINET is an Academic information network, which was created to encourage the

exchange of information between scientists and researchers. The aim of SINET is to provide multimedia communication services, such as video. It is managed and supervised by the Ministry of Education, Culture, Sports, Science and Technology and the NII (National Institute of Informatics). SINET is currently connected to 850 universities and research institutions in Japan and internationally linked to economies such as the U.S. (STAR TAP: 300M), Thailand (ThaiSarn: 2M), and England (45M). SINET is an Information communication network designed exclusively for academic research. It connects regional nodes that are spread around the nation through broadband linkage. Furthermore, it is promoting interconnection with international networks in order to encourage the exchange of international information. The Figure below shows the network structure of SINET

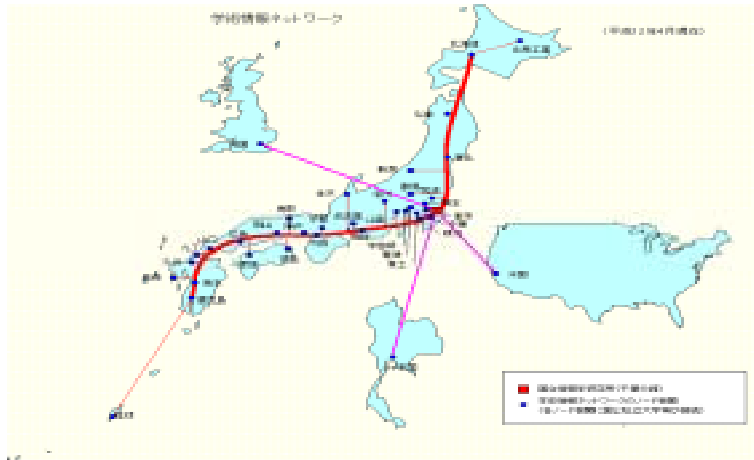


<Figure 5> Japan's domestic connectivity



<Figure 6> Network of JGN

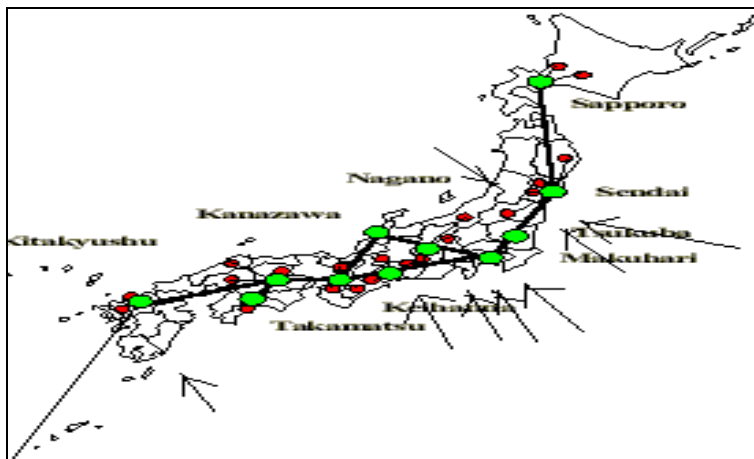
The following Figure will help understand the structure of SINET.



<Figure 7> Network of SINET

Conclusively, both SINET and JGN are composed of backbones that connect the northeast and south of Japan along the coastline. The only difference that can be found between the networks of JGN and SINET is that SINET connects Kagoshima via Fukuoka whereas JGN exclude Kagoshima and places relatively more importance on Nagano and Kanazawa. Hence, in terms of research network, Tokyo, Nagoya and Sendai seem to be suitable candidates.

2.5.2 Submarine Cables



<Figure 8> Selection of cities in Japan

	Japan
Korea	FLAG(10G) SMW3(40G) HJK(280M) RJK(560M) CUCN(80G) APCN(2560G)
China	FLAG(10G) CUCN(80G) SMW3(40G)
Chinese Taipei	SMW3(40G) APCN(2560G)
Hong Kong, China	APCN(2560G) APC(20G) SMW3(40G) FLAG(10G) HJK(280M)
Philippines	APCN(2560G) SMW3(40G)
Viet Nam	SMW3(40G)
Thailand	APCN(2560G)
Malaysia	APC(20G) APCN(2560G) SMW3(40G) FLAG(10G)
Singapore	SMW3(40G) APC(20G) APCN (2560G)
Indonesia	SMW3(40G) APCN(2560G)

<Table 9> Submarine cables landing in Japan

Many cables, such as FLAG, SMW, HJK, RJK, CUCN, APCN, have landing stations in Japan. Most of the landing stations are concentrated around Tokyo and Nagoya. And many

submarine cables have landing stations in either Tokyo or Sendai. Therefore, it would be most appropriate to select Tokyo first and then either a city left (Nagoya) or right (Sendai) of Tokyo as the next option. In terms of landing station, Miyazaki also has a landing accessibility. However, Miyazaki is not a big city with big demand and therefore is somewhat inadequate as candidate city.

2.5.3 Population

City	Pop. 2002
Tokyo	8,027.5
Yokohama	3,552.3
Osaka	2,647.0
Nagoya	2,258.0
Sapporo	1,779.7
Kyoto	1,492.5
Kobe	1,467.4
Fukuoka	1,334.2
Kawasaki	1,292.0
Hiroshima	1,131.1
Kitakyushu	1,058.8
Sendai	1,015.4
Chiba	932.8

<Table 10> Population of major cities in Japan

In terms of population, Tokyo is the largest city and next is Yokohama, Osaka, Nagoya and Sapporo in the order given. The first choice of cities, considering only the population size, will be Tokyo and Yokohama. The second choice will be Nagoya. The third option will go to Sapporo if we only consider population. However, Sapporo has a disadvantage in that it has no landing station and as it is located in the far end of the northeastern region, there seems to be little advantage in connecting Sapporo. Therefore Sendai seems to be a more logical choice.

2.5.4 Final Selection

Through the study of various factors, domestic connectivity, cable accessibility, and population size, the most clear-cut conclusion is that the first selection must be Tokyo. If a second selection were to be made, it would have to be a city near Tokyo. So Sendai, which is

located to the right of Tokyo or Nagoya, which is located to the left of Tokyo, would be the likely candidates. So the first selection will be Tokyo, the second Nagoya, and the third Sendai. However, if Sendai and Nagoya are deemed too close to Tokyo, then Miyazaki could be an alternative candidate, despite the fact that it does not qualify as a major city, since it has a landing station. Nowetsu, a city located near Niigata, also has RJK cables that connect to Pusan and Russia. However, it is too far away from the metropolitan areas to provide any significant benefits.

In conclusion, Tokyo and Nagoya are the most appropriate cities in Japan. And Sendai can be considered as the third appropriate city.

2.5.5 Traffic Data

The economy with the largest amount of traffic exchanges with Japan is undoubtedly the U. S. This proves the importance of the North East Asian region, such as Korea and China, as compared to South East Asian economies, in terms of data transaction.

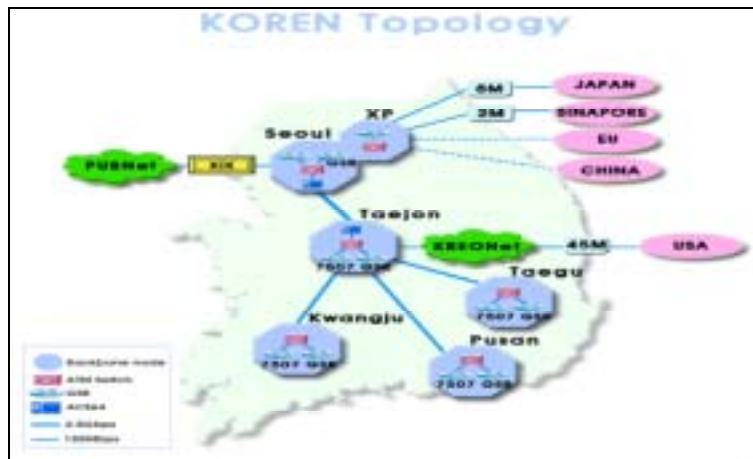
	Japan
Brunei Darussalam	0.175
China	161.45
Hong Kong, China	52.775
Indonesia	29.1
Japan	0
Korea	151.45
Malaysia	28.85
Philippines	90
Singapore	45.725
Chinese Taipei	80.2
Thailand	56.15
Viet Nam	4.275
United States	564.9
Canada	24.175
Australia	41.55
New Zealand	4.475

<Table 11> Japan's International PSTN traffic volume, million minutes
Data from: international traffic Index, ITU, 1999

2.6. KOREA

2.6.1 Research Network

KOREN was built in July 1995 with the goal of creating a collaborative research foundation through connecting national and international next generation network building projects. It aimed to achieve this by providing a high-performance network environment to users from the national R&E community, so that in turn the development of advanced network technologies and applications could flourish. It also hoped to encourage technology development businesses where national research institutions could participate together. KOREN has main nodes in Seoul, Daejeon, Daegu, Pusan and Gwangju. Seoul and Daejeon are connected at 2.5Gbps and the other cities at 155Mbps. KOREN is used as a domestic network for the APII Testbed. It is also connected to International next generation Internet research networks such as SingAREN, Abilene, vBNS, CA*net 3 through an 8Mbps link with Japan and 2Mbps link with Singapore.



<Figure 9> International linkage status of KOREN

KREONET 2 (Korea Research Environment Open Network 2) is a science technology innovation infrastructure that provides means to collaborate and exchange high performance computer resources.

The connection between Seoul and Daejeon is at 155Mbps and Pusan, Daegu, Kwangju, Chungnam, Chungbuk, Jonnam, Suwon, Incheon, and Changwon at 45Mbps. The network status for Gangwon, Ulsan is 10Mbps, and Jeju and Pohang 4Mbps. As for international lines, KREONET is connected to the U.S. through APII/KREONET·STAR TAP (45Mbps). It also has a separate line at 16Mbps with the U.S.. KREONET is linked with Japan as well at 255Kbps.

After careful revision of the research network, it is clear that the core backbone of Korea is the connection from Seoul to Daejeon, Daejeon to Kwangu, Daejeon to Daegu and Daegu to

Pusan, Seoul, Kwangju and Pusan, which are located on the coastlines, are considered to be the best candidates for this project.

2.6.2 Submarine Cables

Korea is accessible to a variety of cables such as FLAG, SMW, HJK, RJK, CUCN, and APCN. It can be noted that Pusan, Kwangju and Seoul (Incheon) are all potential landing locations that have submarine cable accessibility, as well as domestic connectivity.

	Korea
Japan	FLAG(10G) SMW3(40G) HJK(280M) RJK(560M) CUCN(80G) APCN(2560G)
China	CKC((560M) CUCN(80G) SMW3(40G)
Chinese Taipei	SMW3(40G) APCN(2560G)
Hong Kong, China	APCN(2560G) HJK(280M) SMW3(40G)
Philippines	SMW3(40G) APCN(2560G)
Viet Nam	SMW3(40G)
Malaysia	APCN(2560G) SMW3(40G) FLAG(10G)
Singapore	SMW3(40G) APCN (2560G)
Indonesia	SMW3(40G) APCN(2560G)

<Table 12> Submarine cables landing in Korea

2.6.3 Population

City	Pop. 2002
Seoul	11,153.2
Pusan	4,158.1
Daegu	2,670.2
Incheon	2,433.0
Daejeon	1,386.8
Kwangju	1,371.0

<Table 13> Population of major cities in Korea

In terms of population, Seoul and Pusan rank the highest. The next largest city is Daegu, but it is an inland city. Thus, Seoul would be the first choice and Pusan the second, when considering only the population size. Gwangju is also big enough to be considered as potential candidate and it is also a coastal city.

2.6.4 Final Selection

Seoul, Pusan and Gwangju are possible candidates which have domestic connectivity. And they all also have landing stations. In addition, all three have a big enough population. Therefore, Seoul, Pusan and Gwangju can be considered as candidates. The remaining question will be how many cities is required. If only one selection is needed, then Seoul is the first choice, if two cities are required, they would be Seoul and Pusan and naturally the third choice would be Gwangju.

2.6.5 Traffic Data

The following data shows the traffic volume between Korea and APEC member economies in million minutes. The amount of PSTN traffic shows that the U.S. has the most active exchange with Korea, and then follows Japan and China. Based on this information, it can be presumed that the flow of data will also be in a similar order. Therefore, it seems that the amount of traffic is much greater with economies from North East Asia than South East.

	Korea
Brunei Darussalam	0
China	107.5
Hong Kong, China	29.75
Indonesia	13.325

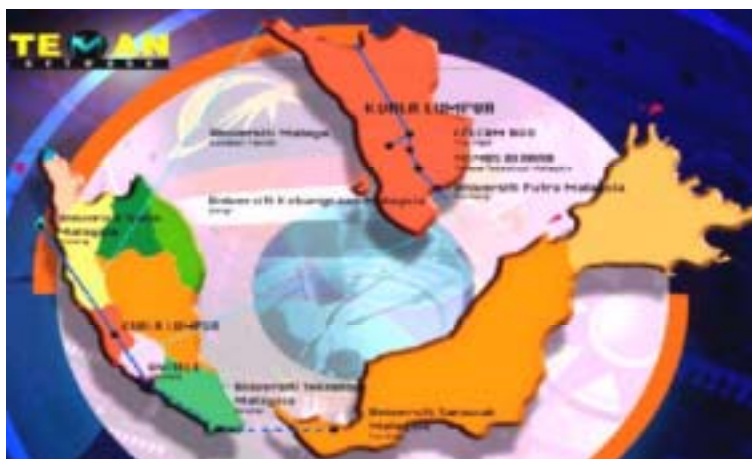
Japan	151.45
Korea	0
Malaysia	4.1
Philippines	12.75
Singapore	12.5
Chinese Taipei	10.6
Thailand	8.9
Viet Nam	5.7625
United States	317.125
Canada	8
Australia	16.5
New Zealand	3.5

<Table 14> Korea's international PSTN traffic volume, million minutes
Data from: International traffic Index, ITU, 1999

2.7. MALAYSIA

2.7.1 Research Network

TEMAN is a broadband multimedia testbed that was launched in 1997 for the development of Multimedia application-related technology, broadband communication and ATM. It is a research network that is managed by the IRPA (Intensification of Research in Priority Areas). Better known as the National Broadband Multimedia Testbed, this project was



<Figure 10> Network of TEMAN

sponsored by the IRPA. It was launched in 1997 and completed in 2000 and was initially proposed with the aim of establishing a National Broadband Multimedia Testbed that will serve as an experimental platform for integrating various kinds of networks, multifarious broadband, and multimedia applications within Malaysia. The members of TEMAN are MIMOS BERHAD, University Kebangsaan Malaysia, University Sains Malaysia, University of Malaysia, Malaysia Putra University, University Malaysia Teknologi, Cellular Communications Network (Malaysia) SDN BHD and the Multimedia University.

The connection from Kuala Lumpur to Taiping and from Kuala Lumpur to Johor Baharu seems to be the most important city network, considering the domestic connectivity. The island of Borneo seems to be less important than the Island of Java, as it does not have many major cities. Therefore, our candidate city selection will be made from the Island of Java rather than the Island of Borneo.

2.7.2 Submarine Cables

Cables landing in Malaysia include APCN, SMW, FLAG, APC. There are relatively more landing points on the Malay Peninsula. Kuantan and Mersing are cities that are situated on the Peninsula which have landing accessibilities. However, there are no landing stations in the cities such as Kuala Lumpur and Ipoh, which are situated on the western side. Thus, installing new landing points must be adopted as an option if we were to choose cities on the western side.

	Malaysia
Korea	APCN(2560G) SMW3(40G) FLAG(10G)
Japan	APC(20G) APCN(2560G) SMW3(40G)
China	SMW3(40G) FLAG(10G)
Chinese Taipei	SMW3(40G) APCN(2560G)
Hong Kong, China	SMW3(40G) APC(20G) APCN(2560G)

Philippines	APCN(2560G)
Viet Nam	SMW3(40G)
Thailand	APCN(2560G)
Singapore	APC(20G) APCN(2560G) SMW3(40G)
Indonesia	SMW3(40G) APCN(2560G)

<Table 15> Submarine cables landing in Malaysia

2.7.3 Population

City	Pop. 2002
Kuala Lumpur	1448.6
Johor Baharu	722
Ipoh	562
Kelang (Port Kelang)	534
Petaling Jaya (Bandar Petaling Jaya)	489
Kuching	368
Kuala Terengganu	303
Kuantan	293

<Table 16> Population of major cities in Malaysia

The data above shows that the cities on the Island of Borneo are not linked to the current research network and that they are smaller in size compared to the cities of Johor Baharu, Ipoh, and Kelang, situated on the Island of Java. Therefore cities located on Borneo Island, such as Kuching, will not be included in the selection

2.7.4 Final Selection

Judging from the domestic connectivity and population, cities that are situated near Kuala Lumpur are appropriate, though these cities do not presently have landing stations. Despite such disadvantage, the first city of choice should be the economy's capital, Kuala Lumpur, with the biggest population. The second choice should be Johor Baharu, which has fairly easy access to

submarine cables and is the second largest city. The last option will be Kuantan, which lacks domestic connectivity but has a landing station. If the Island of Borneo must be considered, then Kuching will be the most likely choice.

2.7.5 Traffic Data

In terms of traffic volume, Malaysia shares the largest amount of traffic with Singapore. Indonesia, the U.S. and Japan comes next.

Gathering from the traffic data, the communication network that connects Malaysia, Singapore and Indonesia is important.

	Malaysia
Brunei Darussalam	7.55
China	8.025
Hong Kong, China	18.5
Indonesia	30.675
Japan	28.85
Korea	4.1
Malaysia	0
Philippines	8.225
Singapore	369
Chinese Taipei	13.325
Thailand	13.375
Viet Nam	0.6
United States	26.15
Canada	0
Australia	32
New Zealand	4.5

<Table 17> Malaysia's international PSTN traffic volume, million minutes
Data from : international Traffic Index, ITU, 1999

2.8. PHILIPPINES

2.8.1 Research Network

PHNET was created with the support of the DOST and the Industrial Research Foundation IRF. It traces its root to June 1993 when the DOST convened a Technical Committee composed of representatives from DOST (MIS, PCASTRD, and ASTI), Ateneo de

Manila University (ADMU), De La Salle University (DLSU), and the University of the Philippines at Diliman (UPD) and the University of the Philippines at Los Banos (UPLB), to study the viability of establishing such a network. Phase 1 of the DOST project began in July 1993 with a limited electronic facility linking DOST, ADMU, DLSU, and UPD. ADMU acted as the local relay hub for the Philippines. The DOST committed additional support for the continuation of the project to Phase 2. At phase 2, the coverage of PHNET was enlarged to 10 members with the inclusion of the University of Santo Tomas (UST), University of San Carlos (USC) in Cebu, St. Louis University (SLU) in Baguio, Xavier University (XU) in Cagayan de Oro, and the Industrial Research Foundation (IRF). In addition, the Asian Development Bank (ADB), International Rice Research Institute (IRRI), Mosaic Communications (MOZCOM), and the Asian Institute of Management (AIM) joined the project. Phase 2 of the project was successfully completed when 64 Kbps leased line connections were effected among its members and full access to the Internet was established to the U.S. via a 64 Kbps international leased line in March 29, 1994.

In conclusion, based on the domestic research network of the Philippines, the connection from Manila to Davao through Cebu seems to be the core domestic connection.

2.8.2 Submarine Cables

SMW, APCN, GPT are cables that have landing stations in the Philippines. SMW and APCN have landing stations in Batangga and GPT has in Infant. These cables can be seen as having direct access to Manila due to geographical adjacencies.

In other words, there are no landing points other than the Manila region in the Philippines. This means that in order to make another link a new landing station must be considered.

	Philippines
Korea	SMW3(40G) APCN(2560G)
Japan	APCN(2560G) SMW3(40G)
China	SMW3(40G)
Chinese Taipei	G-P-T(20G) SMW3(40G) APCN(2560G)
Hong Kong, China	APCN(2560G) SMW3(40G)

Viet Nam	SMW3(40G)
Thailand	APCN(2560G)
Malaysia	APCN(2560G)
Singapore	APCN(2560G) SMW3(40G)
Indonesia	SMW3(40G) APCN(2560G)

<Table 18> Submarine cables landing in the Philippines

2.8.3 Population

In terms of population, Manila is the largest. Davao and Cebu are the next largest cities. Naturally, all three cities must be included in the selection if we consider the population to indirectly represent the demand for network.

City	Pop. 2002
Manila	10,133.2
Davao	873.8
Cebu	747.3
Bacolod	440.7
Cagayan	425.8

<Table 19> Population of major cities in the Philippines

2.8.4 Final Selection

The Philippines uses the Internet network as its research network. Therefore, unlike other economies, the Philippine does not have a separate research network. However, if a core network must be identified, then it would be the connection from Manila to Davao. The next city that appears to be most appropriate for this project is Cebu. This is because most of the available landing stations in the Philippines are concentrated near Manila. Furthermore, there is no landing station near Davao. In addition to that, Cebu has the third ranking population size and is located near South China Sea, which can help save construction cost.

In conclusion, the best city for this project is Manila. And Cebu would be the next appropriate choice.

2.8.5 Traffic Data

The following data shows that the U. S. has the largest amount of traffic with the Philippines. Hong Kong, China has the second largest and Chinese Taipei and Canada the third. Based on this data, it is possible to predict that a connection between Thailand and Hong Kong, China is appropriate.

	Philippines
Brunei Darussalam	1.75
China	4.075
Hong Kong, China	45.625
Indonesia	4.575
Japan	90
Korea	12.75
Malaysia	8.225
Singapore	19.85
Chinese Taipei	25.3
Thailand	3.875
Viet Nam	0
United States	140.075
Canada	22.025
Australia	22.05
New Zealand	0.75

<Table 20> Philippine's International PSTN traffic volume, million minutes
Data from : International traffic index, ITU, 1999

2.9. SINGAPORE

2.9.1 Research Network

The Singapore research network is a research education network that was launched in 1997 in order to develop a new broadband technology and to test interoperability. SINGAREN was built with the purpose of promoting and facilitating R&D in next generation broadband and Internet technologies, and linking Singapore's R&E community to the global information infrastructure while fostering collaboration. SINGAREN is a broadband network linked to approximately 30 universities and research institutions. SINGAREN also supports universities, research institutions and the R&D activities of various enterprises. Nationally, SINGAREN is an experimental broadband network of two regional universities and several regional research

institutions. Other institutions can connect by high-speed connection via Singapore ONE (Singapore’s Commercial broadband network). Internationally, SINGAREN maintains links with to the U.S. and Canada at 14 Mbps and also to Japan and Korea at 2Mbps for non-commercial uses. SINGAREN is connected to the U.S. through vBNS, to Canada through Canarie, and to Japan through various R&D networks. Europe is connected to Singapore via CANARIE’s Trans-Atlantic connection.

2.9.2 Submarine Cables

Singapore has landing stations for cables SMW, APCN and APC. As it is a city-sized economy multiple landing stations would not be possible.

	Singapore
Korea	SMW3(40G) APCN(2560G)
Japan	SMW3(40G) APC(20G) APCN(2560G)
China	SMW3(40G)
Chinese Taipei	SMW3(40G) APCN(2560G)
Hong Kong, China	SMW3(40G) APC(20G) APCN(2560G)
Philippines	APCN(2560G) SMW3(40G)
Viet Nam	SMW3(40G)
Thailand	APCN(2560G)
Malaysia	APC(20G) APCN(2560G) SMW3(40G)
Indonesia	SMW3(40G) APCN(2560G)

<Table 21> Submarine cables landing in Singapore

2.9.3 Population

Not needed

2.9.4 Final Selection

As there is no geographical difference between landing stations, regardless of the number chosen, all connections will be linked to one point, Singapore.

2.9.5 Traffic Data

	Singapore
Brunei Darussalam	5.975
China	42.5
Hong Kong, China	66
Indonesia	76.2
Japan	45.725
Korea	12.5
Malaysia	369
Philippines	19.85
Chinese Taipei	26.275
Thailand	35.55
Viet Nam	8.2625
United States	67
Canada	0
Australia	60
New Zealand	7

<Table 22> Singapore's International PSTN traffic volume, million minutes

Malaysia, which is geographically close, has the largest amount of traffic with Singapore. Indonesia; Hong Kong, China; China and United States each rank second, third and fourth, respectively, in terms of the amount of traffic shared with Singapore. Based only on the information above, it would be ideal to connect Singapore with both Malaysia and Indonesia. .

2.10. CHINESE TAIPEI

2.10.1 Research Network

Chinese Taipei has BARRNET (Bay Area Research and Education Network), TWIX(Chinese Taipei Academic Network) TARNet2 as research networks. TARNet2

provides network environment so that Chinese Taipei's education and research institutions can cooperate efficiently with foreign institutions, especially education and research institutions of the United States.

Managed and operated by the Ministry of Education, its foreign network maintains connection through its links with vBNS, CANARIE, SingAREN, SURFnet(Netherlands), Israel Internet 2 and APAN via its connection with America's STAR TAP(45M).

The goal of this network is as follows: 1. Creating a friendly research environment through connections with America's NSF vBNS, Canada's CANARIE CA*net2, Singapore's SingAREN and APAN of the Asia Pacific region. 2. To solve the overflow of international network traffic by increasing the network's line 3. To boost Chinese Taipei's network technology research to an international level by importing next generation network technology. 4. To participate actively in international standard agreement thus expanding the scope of national network research.

2.10.2 Submarine Cables

	Chinese Taipei
Korea	SMW3(40G) APCN(2560G)
Japan	SMW3(40G) APCN(2560G)
China	SMW3(40G)
Hong Kong, China	APCN(2560G) SMW3(40G)
Philippines	G-P-T(20G) SMW3(40G) APCN(2560G)
Viet Nam	SMW3(40G)
Thailand	APCN(2560G)
Malaysia	SMW3(40G) APCN(2560G)
Singapore	SMW3(40G) APCN(2560G)
Indonesia	SMW3(40G) APCN(2560G)

<Table 23> Submarine cables landing in Chinese Taipei

SMW, APCN, GPT have landing stations in Chinese Taipei. However, as all the cables have landing stations in one island, it is unnecessary to think of a way to have several different landing points

2.10.3 Population

Not needed

2.10.4 Final Selection

Like Singapore, it is unnecessary to make several selections in Chinese Taipei. Therefore in this project, only one city with a landing station will be considered as candidate.

2.10.5 Traffic Data

The data below shows that the United States leads in terms of the amount of Traffic with Chinese Taipei; China; Hong Kong, China; Japan; the Philippines; Singapore and Thailand also share a large amount of traffic with Chinese Taipei. It also shares a significant amount of traffic with Canada and Australia.

	Chinese Taipei
Brunei Darussalam	0.025
China	174.45
Hong Kong, China	62.975
Indonesia	13.7
Japan	80.2
Korea	10.6
Malaysia	13.325
Philippines	25.3
Singapore	26.275
Chinese Taipei	0
Thailand	17.725
Viet Nam	10.35
United States	249.45
Canada	18.675
Australia	19.65
New Zealand	5.35

<Table 24> Chinese Taipei’s International PSTN traffic volume, million minutes

Data from : international traffic Index, ITU,1999

In July 2000, 1772 schools joined Schoolnet through the Golden Jubilee network, which was owned by the royal family of Thailand. By August 2001, more than 2814 schools had joined Schoolnet. Currently, Schoolnet expects approximately 5000 schools to join its network by the end of this year.

After reviewing Thailand's domestic network, it can be noted that all major domestic connection nodes are located inland rather than on the coast. This might cause some limitations in selecting coastal cities with easy access to submarine cables.

2.11.2 Submarine Cables

It can be seen from the table below that only one cable (the APCN) has a landing station in Thailand. This landing station is situated in Phetchaburi near Bangkok, the capital city of Thailand.

	Thailand
Korea	APCN(2560G)
Japan	APCN(2560G)
China	N.A.
Chinese Taipei	APCN(2560G)
Hong Kong, China	APCN(2560G)
Philippines	APCN(2560G)
Viet Nam	APCN(2560G)
Malaysia	APCN(2560G)
Singapore	APCN(2560G)
Indonesia	APCN(2560G)

<Table 25> Submarine cables landing in Thailand

2.11.3 Population

Most of Thailand's cities are located inland, therefore, making it difficult to have submarine cable networks. So Bangkok, which is located near the coast, seems to be the most appropriate choice in terms of cable accessibility.

City	Pop. 2002
Krung Thep [Bangkok]	6,513.1
Samut Prakan	390.3
Nonthanburi	300.2

<Table 26> Population of major cities in Thailand

2.11.4 Final Selection

The difficulty in selecting a city in Thailand comes from the fact that most major cities are located inland. Therefore, the only relevant city to this project is Phetchaburi, which is located near Bangkok and also, has a landing station for APCN. Therefore Bangkok, which is near the city of Phetchaburi, is the first and the second choice should be Songkhla.

2.11.5 Traffic Data

The PSTN traffic of Thailand shows that it communicates with Japan the most. Singapore and the U.S. exchange the next largest amount of traffic. The large amount of traffic between Thailand and Chinese Taipei; Hong Kong, China and Malaysia comes from the closeness of these economies. Therefore, it can be concluded that a connection between Thailand and Malaysia is necessary.

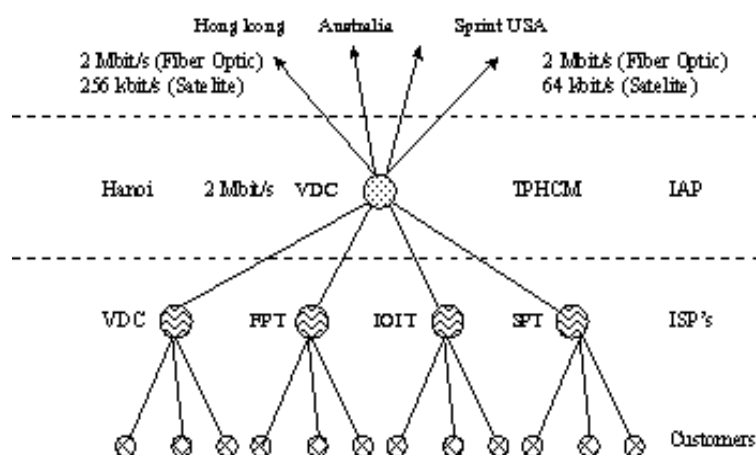
	Thailand
Brunei Darussalam	0.15
China	6.95
Hong Kong, China	18.5
Indonesia	4.3
Japan	56.15
Korea	8.9
Malaysia	13.375
Philippines	3.875
Singapore	35.55
Chinese Taipei	17.725
Thailand	0
Viet Nam	2.6
United States	31.8
Canada	0
Australia	12.65
New Zealand	1.05

<Table 27> Thailand's International PSTN traffic volume, million minutes
Data from : International traffic Index, ITU, 1999

2.12. VIET NAM

2.12.1 Research Network

The Hanoi region is connected to Hong Kong, China through cable and with Australia through satellite at 256K. It is also connected to the United States through satellite at 2M or 64 K.



<Figure 13> Domestic inter-connection

VAREnet was initiated in 1993 in conjunction with the national university of Australia. VAREnet is a computer communication network initiated for network development/expansion research which aims to provide research and education services. Currently, VAREnet is a large-scale national project where each research training institution is cooperating in maintenance and development. VAREnet aims to establish an Internet system, which coincides with the existing international standards and domestic standards, through the improvement and expansion of technology. The IOIT and the national natural science technology center functions as the main network for VAREnet. VAREnet started out with National Center for Science & Technology, National Technology Information Center, Hanoi University of Technology, Ho Chi Min city University of Technology, Ho Chi Min city University of Technology among others, as its first seven members.

In conclusion, based on the current research network, Ho Chi Min and Hanoi must be chosen. Both cities have the advantage of being near the coastline and having excellent accessibility to submarine cables.

2.12.2 Submarine Cables

SMW has a landing station in Viet Nam at the city of Danang. It should be brought to attention that Danang is located between Hanoi and Ho Chi Min. Another fact that is worth noticing is that there is no submarine cable that connects Viet Nam with Thailand. A submarine cable may possibly be needed in order to connect Viet Nam and neighboring Thailand.

	Viet Nam
Korea	SMW3(40G)
Japan	SMW3(40G)
China	SMW3(40G)
Chinese Taipei	SMW3(40G)
Hong Kong, China	SMW3(40G)
Philippines	SMW3(40G)
Thailand	N.A.
Malaysia	SMW3(40G)
Singapore	SMW3(40G)
Indonesia	SMW3(40G)

<Table 28> Submarine cables landing in Viet Nam

2.12.3 Population

The table below shows the population of the major cities in Viet Nam. Ho Chi Min and Hanoi are undoubtedly the most important cities along with Danang, which has a landing station.

City	Pop. 2002
Tha Pho Ho Chi Min	3,378.5
Ha Noi	1,372.8
Hai Phong	572.1
Da Nang	446.0
Bien Hoa	365.5

<Table 29> Major cities in Viet Nam

2.12.4 Final Selection

The research network shows that Hanoi and Ho Chi Min are important regional nodes in Viet Nam. Therefore, Hanoi and Ho Chi Min must be selected. However, the problem remains in the location of the city that has the landing station, the city of Danang. Danang is situated between Hanoi and Ho Chi Min city. Thus, if we were to disregard the landing problem, the first candidate would be Hanoi and the second Ho Chi Min.

2.12.5 Traffic Data

The table below shows the PSTN traffic between Viet Nam and other economies in million minutes. The United States shares the largest amount of traffic with Viet Nam, Chinese Taipei and Singapore share rank the second and third. Hong Kong, China; Korea and Japan also

have a significant amount of communication with Viet Nam.

	Viet Nam
Brunei Darussalam	0
China	1
Hong Kong, China,	5
Indonesia	0.3
Japan	4.275
Korea	5.7625
Malaysia	0.6
Philippines	0
Singapore	8.2625
Chinese Taipei	10.35
Thailand	2.6
Viet Nam	0
United States	42.325
Canada	1.5
Australia	6.25
New Zealand	0

<Table 30> Viet Nam's International PSTN traffic volume, million minutes

Data from : international traffic Index, ITU, 1999

Chapter 3

Multi-Connectivity Model

Chapter 3 will be guided by the following three sections: section one with a short introduction to the multiple connectivity model, section two that explains about a specific mathematical model and simulation solutions, and section three which includes network design scenarios and each of their end results. Appendix is also attached at the end to show what computer source code and input data were used in the programme.

3.1. Introduction to Multi-Connectivity Model

Optical communication allows us to simultaneously transmit vast amount of data thus enabling the exchange of various information and fulfilling a more satisfactory service that the society demands today. While optical communication has the capability to transfer data at a very high speed, its failure can lead to an enormous loss of data, resulting in a great loss for the entire society. For such reasons, the security and integrity of the network becomes an essential part in the broadband era network. Furthermore, a network with the ability to self-recover in case of a glitch will be crucial.

Automatic cross-connect that uses intelligent transferring and transmission equipment can be utilized to configure a secure network. But even before that, we must be able to guarantee diverse routing.

The physical layer of optical networks is simply a connection of physical links made up of optical cables. Transmission systems that are provided on such layers frequently require two-connectivity to ensure stability. For example, stability ensured by ADM (which is used in either SONET or SDH) or OADM (which uses WDM) assumes that there are always at least more than two routes that are secured. If the connectivity of current optical cable is poor, and a safer connectivity cannot be guaranteed, then the recovery of a damaged traffic is unthinkable,

regardless of intelligent switching and transmission equipment. This leads to the conclusion that designing a two-connectivity network is a matter that needs prior attention when implementing a stable network.

3.1.1 Related Studies

From the late 1980s to the 90s, network enterprisers in the developed economies carried out studies on designing two-connectivity networks that would secure the stability of physical optical cable network. Bellcore (currently known as Telcodia) also supported these studies.

As a result, a mathematical model and solution for designing a stable optical cables network was published[3, 4, 5, 6]. And the result was presented in the form of an optical network design software by Bellcore. [1, 2]

In this study, the optical cable network configuration model from the existing studies will be used to derive optical cable network configuration plans for economies in the Asia-Pacific region. For convenience, the two-connectivity network design and the related concepts presented in the previous studies will be explained.

A network can be explained as a set of nodes and edges. Nodes determine the connection of optical cables such as transmission systems. These nodes can be divided into three categories such as "special node", "ordinary node" and "optional node" according to the level of demand of connectivity. Special node represents nodes which must have connectivity that includes a reciprocal detour route, due to its high demand and functional importance, while ordinary node represents nodes that require simple connectivity. Optional node refers to nodes that may or may not be included in the network because of the shape, geographical location, distance or equipments of the network.

Meanwhile, edges connect these nodes using physical connectivity. It is the work of the designer to find an effective network connection figures that satisfy the demanded node connectivity level.

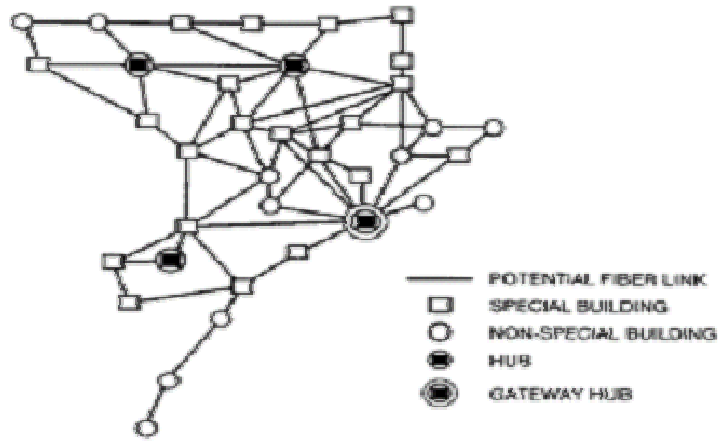
In the existing studies mentioned above, special node is given the connectivity demand level 2, ordinary node level 1 and optional node level 0. The restriction on the two connected nodes requires that it must have a larger number of routes than the minimum connectivity level of the two by the theorem.¹

In other words, if a certain node has a connectivity demand of level 1 and the second node level 2, then there must be more than one route between these two nodes. Using the widely known theorem, it can be said that if both nodes have the connectivity demand of level 2, then there must be more than 2 routes that do not share nodes or edges.

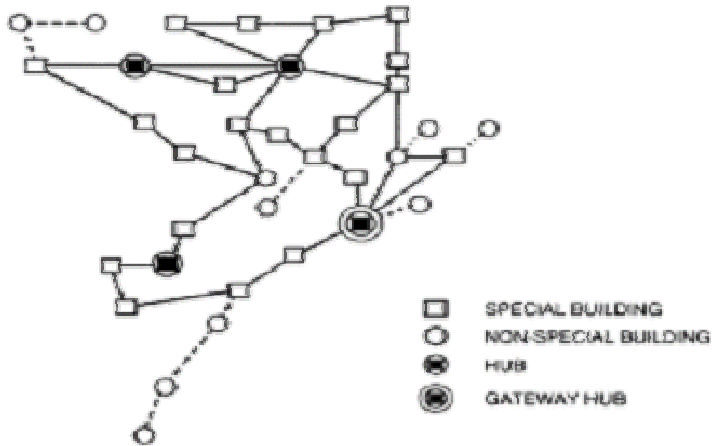
Diagram 14 shows an example of an optical cable network design. Edges represent

¹ Clyde L. Monma and David F. Shallcross, P.531, 1989.

installable optical cables, GATEWAYHUB, HUB and SPECIAL BUILDINGs represent special nodes, and other nodes represent ordinary nodes. Diagram 15 shows the optical cable network design that satisfies the demanded connectivity level.



< Figure 14> Example of an optical cable design



< Figure 15> Optical cable design that satisfies network connectivity level

The advantage of an optical cable network design model, like the one above, is that various alternative network designs can be acquired by changing the demanded connectivity

level according to the importance of the nodes. The design model illustrated above seems very convincing in that it allows the network to carry out its original functions despite changes in the present or future switching and transmission technology. In addition, it is also considered to be a realistic model in that the network engineers have applied it to the structural designing of optical cables. This chapter will mainly focus on the use of this conventional method in designing a suitable network for the Asian region.

3.2. Multi-connectivity Model and Solution

3.2.1. Mathematical Model

"Two-connectivity" can be defined as connectivity that has at least two routes that do not share the same nodes with exceptions to the starting and ending points. "Two-edge connectivity" means having at least two edges that are not shared, whereas "two-node connectivity" means there are at least two nodes that are not connected. Once the "two-node connectivity" is satisfied then the "two-edge connectivity" is in turn satisfied: But it does not work in the reverse way.

If the network keeps two-connectivity free of damages caused by the cables, then the "two-edge connectivity" will have no problem. In the same context, if it is kept free from damages caused by the nodes, the "two-node connectivity" will be satisfied.

In this research, the "two-edge connectivity" model rather than the "two-node connectivity" model is adopted, since this research implicitly assumes that doubling the node facilities will help construct networks and solve problems which are caused by nodes.

In reality, ensuring two-connectivity throughout the network is hard to achieve, due to its high cost. Thus, the nodes are generally divided into 3 types taking into account their demand and function: the "special node" which guarantees bi-connectivity, "ordinary node" which satisfy regular connectivity and "optional node" which is optional.

The physical connection of the network can be represented by a non-directed graph $G = (V, E)$. V represents a set of nodes equipped with transmission systems and repeater. E represents a set of optical cable edges that are installable between nodes. G could be a complete graph connecting all nodes. Here, E is composed of edges that are realistically installable and, thus excludes edges that are unsuitable for installation because of its high cost and natural and social conditions

It was mentioned above that V represents a set of nodes within the network and the nodes can be divided into 3 types according to the level of their importance. Main nodes must secure two-connectivity in order to prevent a breakdown that can result in a great damage of data. With

the ordinary nodes, only a simple connectivity is required. For exclusively the transmission purpose, an optional position will be sufficient. An importance level of 2, 1 or 0 can be assigned to special nodes, ordinary nodes and optional nodes respectively. When these numbers are applied to the nodes s and t , at least the minimum number of disjoint edges should exist to guarantee the demanded connectivity level, for the previously notified reason.

The main issue here is to select edges among the set of E , which satisfy the requirement condition of the node at a minimum cost. The optimal mathematical solution for the network can be calculated as follows.

In the model below, C_{ij} represents the cost of installing optical cables between node i and node j . The fixed variable x_{ij} is a binary variable which takes the value of 1 in case of installation and 0 in non-installation cases.

$$\begin{aligned}
 &\text{minimize} && \sum_{(i,j) \in E} C_{ij} x_{ij} \\
 &\text{subject to} && \sum_{(i,j) \in CUT(s,t)} x_{ij} \geq r_{st} \quad , \quad \forall CUT(s,t), \quad s \in V, \quad t \in V, \quad s \neq t \quad 1) \\
 &&& 0 \leq x_{ij} \leq 1 \quad , \quad \forall (i,j) \in E \quad 2) \\
 &&& x_{ij} : \text{integral number}
 \end{aligned}$$

In the mathematical model above, r_{st} represents the minimum number of the level of two nodes and $CUT(s,t)$ represents a set of edges which can disconnect the nodes of s and t .

In other words, if the edges included in $CUT(s,t)$ are removed, the connection between s and t will be severed.

One of the advantages of the mathematical model above is that by changing the value of $r(i)$, various alternatives can be presented. Another advantage is that this model can be used in designing networks with higher connectivity by giving $r(i)$ some other values, such as 3, instead of 0,1 or 2 used in this experiment.

3.2.2. Computer Simulation

It is widely recognized that arriving at an optimal solution regarding a network design, which satisfies two-connectivity, is extremely difficult and that the only way to reach a solution is by enlisting all possible alternatives. In fact, this is where it got its name, "Non Polynomial - hard". One of the disadvantages of such mathematical model is that it has too many restrictions. In other words, as the number of edges that can disconnect two certain nodes increases exponentially, it becomes more difficult to count and include all the possible restrictive sets in advance.

To solve this problem the “branch-and-bound” algorithm combined with the “constraint generation method”, which only includes generated restrictions when necessary, are adopted. Since the basic purpose of this research is only to find the optimal structure of the optical cables, complex algorithms will be avoided while applying simple algorithm to satisfy our basic needs. All source codes that will be used in our experiments will be enlisted in the appendix.

3.3. Multiple City Case in Multi-Connectivity Model

All possible cases of nodes are enlisted in table 31.

Case	Set of nodes
N31	Seoul, Pusan, Gwangju, Tokyo, Nagoya, Sendai, Beijing, Shanghai, Guangzhou, Jakarta, Surabaya, Kuala Lumpur, Taiping, Hanoi, Ho Chi Min, Manila, Cebu, Bangkok, Sonkhla, Singapore1, Singapore2, Bandar Seri Begawan, Taipei, Hong Kong
N32	Seoul, Pusan, Tokyo, Nagoya, Beijing, Shanghai, Jakarta, Kuala Lumpur, Hanoi, Manila, Bangkok, Singapore1, Singapore2, Bandar Seri Begawan, Taipei, Hong Kong
N33	Seoul, Pusan, Gwangju, Tokyo, Nagoya, Sendai, Beijing, Shanghai, Guangzhou, Jakarta, Surabaya, Kuala Lumpur, Taiping, Hanoi, Ho Chi Min, Manila, Cebu, Bangkok, Sonkhla, Singapore, Bandar Seri Begawan, Taipei, Hong Kong
N34	Seoul, Pusan, Tokyo, Nagoya, Beijing, Shanghai, Jakarta, Kuala Lumpur, Hanoi, Manila, Bangkok, Singapore, Bandar Seri Begawan, Taipei, Hong Kong

<Table 31> Nodes considered in Multiple City Case

In the case of N31 above, Korea; Japan and China each takes 3 subject cities, while

Indonesia; Malaysia; the Philippines; and Thailand each takes 2 and the others 1. In N32, Korea, Japan; China and Singapore each takes 2 subject cities, and other economies take 1. In case of Singapore, two different nodes, Singapore1 and Singapore 2, may be considered taking into account its social and economic importance in the region. For example, while in N31, Singapore is considered to have two different nodes, in N33, it is considered to have only one node. The same applies in the case of N34 and N32.

The node type of each set of subject nodes will be divided into the following two cases. The first case C2 consists of subject nodes that are all designated as special nodes and the second case C3 is considered only when securing connectivity. Connectivity can be secured in two ways: 1) if a single economy has several subject nodes, the corresponding node for each city is designated as an ordinary node to secure connectivity and 2) if a single economy has a single subject node, then the single node is designated as a special node to ensure connectivity.

In other words, in the case of C2, all subject nodes are assigned as nodes that must have two-connectivity. And In the case of C3, where there are several subject nodes, it is assumed that a failure in a certain node can be substituted by another node. So these nodes are designated as ordinary nodes.

Eight basic scenarios derived from the data above can be summarized into the table below.

Case	set of node	Connectivity
N31C2 (figure 16)	N31	C2
N31C3 (figure 17)		C3
N32C2 (figure 18)	N32	C2
N32C3 (figure 19)		C3
N33C2 (figure 20)	N33	C2
N33C3 (figure 21)		C3
N34C2 (figure 22)	N34	C2
N34C3 (figure 23)		C3

<Table 32> Basic Scenarios

Here, it can be said that N32and N34 are more important than N31 and N33 since the assumption itself is more realistic and the difference in the number of candidate cities between N31 and N32. N31 assumes 3 North East Asian cities and 2 South East Asian cities each as subject nodes. However, N32 assumes 2 North East Asian cities and 1 South East Asian cities

as subject node. This relationship is also the same in N33 and N34. If the network construction is to actually take place, the chances are that only one point among multiple subject nodes will work as international gateway node. Therefore, hypothetically considering many cities all at the same time is of less value.

Such being the case, N32, which has less candidate cities, is thought to be more realistic, although both cases will be experimented. The same can be applied to case N33 and N34, with N34 being considered more realistic.

With the C2 model, all subject nodes are assumed to have two-connectivity. However the C3 model assumes that another node can be used for communication in case should the main node suffer from a failure or breakdown. Therefore, a safe assumption can be made that the C3 model is more realistic than the C2 since not all nodes have to necessarily satisfy the two-connectivity.

In conclusion, due to the enlisted reasons above, the results of the N32C3 and N34C3 models should be observed with special attention.

Since all nodes in the C2 model must satisfy two-connectivity as in the case of N31C2, we can presume that this will result in a ring shape network topology. This network in fact has a round shape in Japan; Singapore and Indonesia due to the binding restriction of bi-connectivity. However, such inefficiencies disappear in the more simplified case of N31C3.

N 3 1 C 2 Total Length: 2610 . Computation time : 0.12 sec .



< Figure 16> Multi-Connectivity Model-Multiple cities-N.E. Asia3-S.E. Asia2-Singapore2-
All Special Nodes, N31C2

N 3 1 C 3 Total Length: 2020 , Computation time : 27,690.68 sec.



< Figure 17> Multi-Connectivity Model-Multiple Cities-N.E. Asia3-S.E. Asia2-Singapore2-Partial Special node, N31C3

N 3 2 C 2 Total Length: 2140 , Computation time : 0.03 sec.



< Figure 18> Multi-Connectivity Model-Multiple Cities-N.E. Asia3-S.E. Asia2-Singapore2-All Special Nodes, N32C2

In N32C3, one of the most important figures, South East Asia forms a single circle and North East Asia draws a "Y" shaped network and the two meets in Chinese Taipei. This is an extremely noteworthy point. The double edges between Singapore and Kuala Lumpur can be explained by the double nodes assumption of Singapore. The total length of edges in this figure is 16565.32.

N 3 2 C 3 Total Length : 1905 , Computation time : 0.60 sec .



< Figure 19> Multiple-connectivity Model-Multiple Cities-N.E. Asia2-S.E. Asia2-Singapore2-Partial Special Nodes, N32C3

When comparing N31C3 and N32C3, it is clear that N31 will cost less as it has more room for selection: Shorter edges between nodes can be selected. This assumption can also be proven with total circuit link, N31C3 being 14784 while N33C3 recorded 16565. This shows that the greater the number of candidate cities the cheaper the cost of total connectivity. However, it has to be reiterated that this lacks realistic approach as it ignores the possibility that the selection made can be a city other than the capital or major city.

Another important observation is that C2 shows a single ring topology, while C3 shows a divided configuration. In other words, C2 is a model in which all nodes are considered to be special nodes and therefore demands dual connection for all nodes. And hence, the topology of C2 results in a typical ring topology whereas C3 does not.

N 3 3 C 2 Total Length : 2610 , Computation time : 0.07 sec .



< Figure 20> Multi-connectivity Model-Multi Cities- N.E. Asia3-S.E. Asia2-Singapore1-All Special Nodes, N33C2

N 3 3 C 3 Total Length : 2045 . Computation time : 3.883.99 sec.



< Figure 21> Multi-connectivity Model-Multi-Cities-N.E. Asia3-S.E. Asia2-Singapore1-Partial Special Nodes, N33C3

N 3 4 C 2 Total Length : 2140 . Computation time : 0.02 sec.



< Figure 22> Multi-Connectivity Model-Multiple Cities-N.E. Asia2-S.E. Asia1-Singapore1-All Special Nodes, N34C2

N 3 4 C 3 Total Length: 1905 . Computation time : 0.48 sec .



< Figure 23> Multi-Connectivity Model -Multiple Cities-N.E. Asia2-S.E. Asia1-Singapore1-Partial Special Nodes, N34C3

Comparing C2 and C3, as in N31, N32 and N33, N34, C2 shows a complete ring topology. On the other hand, C3 shows a divided form. Among the eight cases considered here, the results of N32C3 and N34C3 are similar each other and realistic outcomes based on realistic assumptions.

The conclusion that can be derived from the eight experiments examined above is that there are two possible plans in constructing an Asian network. The first scheme involves linking the entire Asian region in a single ring topology and the second to bind the North and South East Asia separately. Furthermore, the data above demonstrates well that for North East Asian region, forming two separate connections in the shape of 'y' is more economical than a single giant ring shape.

3.4. Multi connectivity & Capital City Scenario

In the previous Section, two or three cities from North East Asia and one or two cities from South East Asia were selected and experimented in designing the network. But they are not necessarily a connection between the capital cities. In reality, however, most international gateways of a economy exist in the capital cities. This is particularly true in Asia, and so only the capital cities from each economy will have to be considered. The set of 12 selected nodes are named N5 and the subject nodes include Seoul, Tokyo, Beijing, Jakarta, Kuala Lumpur, Brunei Darussalam, Hanoi, Bandar Seri Begawan, Manila, Bangkok, Singapore, Chinese

Taipei; and Hong Kong China.

The first case (C6) is the case where all nodes are designated as special nodes($r(i)=2$). In the second case (C7), Seoul, Tokyo, Beijing are designated as special nodes($r(i)=2$), and all other nodes are ordinary nodes ($r(i)=1$). In this case, Korea; Japan and China are considered to be more important than other members.

As for the third case (C8), Singapore is added so to designate Seoul, Tokyo, Beijing and Singapore as special nodes and all other nodes as ordinary nodes.

Scenarios	Set of nodes	Connectivity
N5C6 (figure 24)	N5	C6
N5C7 (figure 25)		C7
N5C8 (figure 26)		C8

<Table 33> Capital City Scenarios

The outcome of this section is very important because it provides the keys to directly compare the results with the outcome of the next chapter, which is based on a totally different model but uses the same cities as its subject.

As is foreseen, the diagram below forms a ring topology which covers the entire Asian region. This is a predictable result since all nodes require two-connectivity. It shows that a complete ring that starts from Korea and China through Hong Kong, China; Viet Nam; Thailand and Singapore to Malaysia; Indonesia; Brunei Darussalam; and the Philippines, and back to Japan and Korea is the most effective model. In other words, if all capital cities must be linked, this type of ring topology would be the most appropriate.

If N5C7 is slightly revised so that only Korea; Japan and China are special nodes of importance level 2 and all other capitals are ordinary nodes of level 1, the result will show a form in which North and South East Asia are separated. This separation can be interpreted as the result of an effort to closely link the three important North Eastern capital cities. A more effective network can in fact be built by separating the North and South Eastern sides, creating a small ring configuration covering the Northern part.

When comparing the results of C6, C7 and C8, the hypotheses of C7 and C8 are expected to be the most realistic ones: Not only because it emphasizes the importance of Korea; Japan; China and Singapore, but also because it reflects the level of advancement in network qualities in these regions. The only difference between C7 and C8 is the level of importance given to Singapore. This is very hard to judge, and thus, equal importance will be given to the results of

C7 and C8 for the time being.

What is remarkable in C8 compared to C7 is that Singapore become included as part of the ring. It seems as though this is because under assumption C8, Singapore must have two-connectivity.



< Figure 24> Multi-Connectivity model-Capital City Scenarios-All Special Nodes, N5C6



< Figure 25> Multi-Connectivity Model-Capital City Scenarios-Korea; China; Japan - Special Nodes, N5C7

N 5 C 8 Total Length: 1975 , Computation time : 2.20 sec.



< Figure 26> Multi-connectivity Model-Capital City Scenarios-Korea; Japan; China and Singapore - special Nodes, N5C8

Comparing the three models above clearly shows how difficult it is to decide upon a model that is truly realistic. The results have all differed according to the hypotheses. And therefore, the difficulty now remains in determining realistic restrictions and assumptions. Once the restriction is set, finding the most appropriate network shape can be relatively easy.

To briefly compare the results, the expense for N5C6 and N5C7 were 17,840 and 14,683, respectively. N5C8 is expected to use an expense of 16,814. Thus, in terms of expenses, C7 is deemed to be the cheapest model.

Chapter 4

Combined Model

Section 1 of Chapter 4 introduces a Combined Model and emphasizes its significance. Section 2 delivers an explanation on a mathematical model and computer solutions and section 3 enlists the results of the experiments. In section 4, the experiments are expanded to include multiple cases. All the computer programme sources and input data are appended.

4.1. Significance of the Combined Model

In general, routing is disregarded when designing an initial network topology. Only the multi-connectivity level of the nodes is usually considered, by taking into account the importance of the node. After this kind of simple connectivity is established, then are the routing and the amount of traffic considered as next steps. Such approach presupposes that a sufficient bandwidth between the nodes will be provided when designing a network.

Such typical and mechanical methodology was already fully examined in the previous chapter. Thus, this chapter will attempt to establish a more goal-oriented, creative methodology. The model in this chapter, unlike the previous chapter, is based on the assumption that sufficient bandwidth can be provided. The traditional methodology, which only considers the importance of the node and assumes there to be enough supply of bandwidth, needs revision in a sense that we need to simply focus on the connectivity of the node.

To take into account the routing and the amount of traffic means that the potential amount of traffic which will be generated on the physical network and routing routes will be considered simultaneously along with the designing of the physical network topology. This approach, which considers two aspects at the same time, will be an entirely new approach.

It is impossible to predict accurately the traffic between nodes. Had this been possible, it would have made things a lot easier for us. To make things more complex, such data does not

exist at all at the present moment. International organizations such as APT, as well as APEC, have conducted several investigations on this matter, however, the results have been unsatisfactory.

To overcome this kind of limitation, this study starts with a very important assumption. It presumes that since the current amount of traffic between economies via PSTN has reached a saturated point, it indirectly discloses the relative size of data which will be reached by the data network under a mature state. The most realistic and optimal topology will be studied here by generating the necessary relative size of data traffic deduced from the PSTN traffic amount.

4.2. Structuring Combined Model and Solution

Before going any further, it is necessary to explain the notation and the mathematical model. They are as follows.

$$\min \sum_k \sum_{(i,j)} c_{ij} x_{ij}^k + \sum_{(i,j)} e_{ij} y_{ij} \quad (1)$$

$$s.t. \quad \sum_{j \in A(i)} x_{ij}^k - \sum_{j \in B(i)} x_{ji}^k = \begin{cases} d_k & \text{if } i = s(k) \\ -d_k & \text{if } i = t(k) \\ 0 & \text{o/w} \end{cases} \quad \forall i \quad \forall k \quad (2)$$

$$\sum_k x_{ij}^k \leq M y_{ij} \quad \forall (i, j) \quad (3)$$

$$x_{ij}^k \geq 0, \quad y_{ij} = 0/1 \quad (4)$$

Notation :

(i, j) : Edges between node i and node j

k : Demanded commodity traffic between two economies

$s(k)$: Starting node of k commodity

$t(k)$: Arriving node of k commodity

d_k : Demanded quantity of commodity k

x_{ij}^k : The amount of traffic of commodity k on physical cable (i, j)

y_{ij} : 1 if traffic exists on physical cable (i, j) , or 0 if not

$A(i)$: The set of edges from i

$B(i)$: The set of edges towards i

M : One-way capacity of physical cable

C_{ij} : Construction cost of additional 1 Mbps on physical cable (i, j)

e_{ij} : Set up cost of physical cable (i, j)

(1) is an objective function. This objective function is made up of the costs deriving from an additional 1Mbps traffic and setup cost. If the traffic of 100Mbps flows through (i, j) , the total expense becomes $100C_{ij} + e_{ij}$. On the other hand, if there is no traffic on (i, j) , node i and j do not have to be connected. Therefore, $y_{ij} = 0$ and the setup cost becomes 0.

The k-commodity traffic between two economies passes through node i and j . The construction cost is divided into 2 parts. The fixed and variable cost varies according to the amount of traffic.

x_{ij}^k is the amount of traffic of a certain k-commodity between node i and j . Hence, C_{ij}^k represents the cost derived by 1 size of commodity traffic, which can be explained by a marginal cost as well. Thus $C_{ij}^k x_{ij}^k$ made up of variable cost of commodity traffic k.

To get the cost function of C, the real cost data used in constructing SE-ME-WE² is adopted. Refer also to this information in the web-site. Since this cost function is a function of distance, real cost can be calculated using the measured distance data.

To calculate the setup cost, the information from SE-ME-WE System is used. Using the discount rate below, set up cost can be calculated.

Distance	Discount Rate
200 Km below	Below 200K MIU per Km 120\$/MIU*km
Between 200Km and 600Km	Above 200K MIU per Km, 60\$/MIU*km
Above 600K	Above 600K MIU per Km, 30\$/MIU*km

² <ftp://www.marine.francetelecom.fr/english/frames/offreglo/pose/pose.htm>.

To get the cost function, regression method is used to first figure out the estimation line. The resulted estimation line shows that $30.381*1 + 27435$ (\$) cost is calculated when the cable of 2 MIUs are constructed in 1000 Km.

Accordingly, $(30.381x_{ij} + 27435)*l_{ij}/1000$ is the cost of constructing x_{ij} MIU of distance l_{ij} . Lastly, $c_{ij} = \frac{30.381*l_{ij}}{500}$, $e_{ij} = \frac{27435*l_{ij}}{500}$ is the cost of 1Mbps per Km. This number will also be used in the computer simulation.

Considering the characteristics of the first part of the object function $\min \sum_k \sum_{(i,j)} c_{ij}^k x_{ij}^k + \sum_{(i,j)} e_{ij} y_{ij}^k$, the cost will increase by traffic quantity and distance due to c_{ij}^k .

Thus to minimize the cost, since direct connection is always cheaper than any other alternatives, the mesh type topology is expected to show up as a result.

If the object function is made up only of the second part, the model has to encounter only the setup cost which naturally leads to the same product from “spanning tree” since the spanning tree is the optimal solution for minimizing the set up cost.

However, the model adopted in this chapter covers both the variable and set up cost, thus this model continues on to compare the cost. It helps to decide whether to use the existing edge or to build new edges.

$$\text{Formula 2} \quad \sum_{j \in A(i)} x_{ij}^k - \sum_{j \in B(i)} x_{ij}^k = \begin{cases} d_k & \text{if } i = s(k) \\ -d_k & \text{if } i = t(k) \\ 0 & o/w \end{cases}$$

conservation. This determines the amount of incoming and outgoing traffic at a certain node. If that node is the starting node of k-commodity then the difference in the incoming and outgoing traffic represents the total commodity's demand d_k . If such node is the arriving node $t(k)$

then it will be $-d_k$ since arriving can be interpreted as a negative starting of traffic. The rest of the nodes have the value 0 as they act only as intermediaries and the same amount eventually comes and goes out.

In the computer simulation, the proxy variable calculated from PSTN is taken instead of the real number of d_k as was described previously: This research attempts to overcome data shortage by using a proxy variable that is believed to reflect the needed amount of data traffic. PSTN data is used because the amount of PSTN traffic between economies rightly

represents the amount of data traffic that must ultimately be reached.

Formula 3 is a restriction formula that limits the total capacity of physical cables (i, j) , so that it does not overflow the cable capacity M . Due to formula 4, y_{ij} becomes $y_{ij} = 0$ when there is no traffic on (i, j) and $y_{ij} = 1$ when there is traffic.

4.3. Combined Model & Capital City Scenarios

First of all, the capital cities will be connected using the combined model. The subject economies are Korea; China; Japan; Chinese Taipei; Chinese Hong Kong, the Philippines, Viet Nam; Thailand; Malaysia; Indonesia; Brunei; and Singapore and perhaps the U.S. as an additional option. The U.S. is included as an option because of the fact that most of the important traffic has close links with the U.S.. Thus, the network configuration will be examined with this in mind.

The differences between the models that include the U.S. and that do not will be observed and compared.



< Figure 27> Combined Model-13 capitals (13 members such as Korea; China; Japan; Chinese Taipei; Hong Kong, China; Philippines; Viet Nam; Thailand; Malaysia; Indonesia; Brunei Darussalam; Singapore; and United States)

The result shows that North and South East Asia are separated by Chinese Taipei. North East Asia forms a triangular shape, whereas South East Asia forms a ring type topology with Singapore in the center.

One important result is that Japan, which delivers the traffic that stems from South East

Asia to the United States, plays the role of an important regional gateway to the United States and is directly connected to Chinese Taipei. This may be because Japan is closer to the United States than any other economies considered in this study. But these edges disappear when the United States is excluded from the option.

The fact that the shape of South East Asia in both figure 27 and 28 does not change implies that the traffic between the United States and South East Asia is still not big enough to change the South East Asia's topology .



< Figure 28> Combined Model-12 Capitals (12 members such as Korea; China; Japan; Chinese Taipei; Hong Kong; China; Philippines; Viet Nam; Thailand; Malaysia;,Indonesia; Brunei Darussalam; Singapore)

When the traffic related to the United States is excluded from the option, the resulting topology shows that the South East Asian region formed a topology type that was similar to a ring with Singapore in the center and the North East Asian region forms a "Y" shape with Korea in the center. In details, South East Asia basically formed a ring, with a couple of branches that spread out to Malaysia; Indonesia; and Thailand.

One of the advantages of the Combined Model is that it shows the relative size of the edges. Considering the fact that the numbers presents in figure 28 delivers the relative bandwidth size of the edges, and not the size of the bandwidth itself, it would be worthy to notice the different bandwidths between the two regions.

The result of this study shows that the bandwidth size between Chinese Taipei and Korea are similar to that of Korea and Japan, whereas the link between Korea and China needs to be greater than the two mentioned earlier. It also seems that half the bandwidth size of Korea and Japan will be sufficient for the connections among other South East Asian economies.

4.4. Combined Model & Multiple City Scenarios

The result of the previous section is a model that takes into account the capital city of each economy and limits the number of connecting points to one point, while the model to be examined in this section deals with several cities. In other words, it is assumed that international traffic generated in each economy could be generated not only in capital cities but also in other cities in a simultaneous manner. In such case, the total amount of traffic from several cities within an economy must be the same as the economy's total demand.

The above-mentioned model is shown below in a mathematical formula. When (i, j) represents the cables between two cities (not two economies), z_k represents the demand between the two cities—which will be called inter-city commodity—and $ncom(k)$ represents the economy commodity number of the economy in which the two cities are included .

Model – TCM 2.

$$\min \sum_k \sum_{(i,j)} c_{ij} x_{ij}^k + \sum_{(i,j)} e_{ij} y_{ij} \quad (1)$$

$$s.t. \quad \sum_{j \in A(i)} x_{ij}^k - \sum_{j \in B(i)} x_{ji}^k = \begin{cases} z_k & \text{if } i = s(k) \\ -z_k & \text{if } i = t(k) \\ 0 & \text{o/w} \end{cases} \quad \forall i \quad \forall k \quad (2)$$

$$\sum_{i | ncom(i)=k} z_k = d_k \quad \forall k \quad (3)$$

$$\sum_k x_{ij}^k \leq M y_{ij} \quad \forall (i, j) \quad (4)$$

$$x_{ij}^k, z_k \geq 0, \quad y_{ij} = 0/1 \quad (5)$$

The constraint of (3) implies that all the traffic gathered among the cities within an economy equals the traffic between economies. All other conditions are similar to those in the previous section.

When considering two cities from Korea;China and Japan and one from another economy, the result is as shown in figure 29.

Since two cities within the North East Asian economies are selected, a capital city with an inefficient location can be excluded, as in the case of Beijing and Seoul. Thus, Shanghai is selected for China, as Pusan is for Korea, and the two cities are then connected.

As expected, no change took place in South East Asia and the total length of 11000.02 also stays the same as before. The total length in North East Asia, however,

decreases from 3813.8 to 2956.52, since relatively effective cities are selected in terms of location.



< Figure 29> Combined Model-2 N.E. Asia-1 S.E. Asia.

As for South East Asia, only one city is taken into account and the topology of the network is as shown in the diagram above. However, if two cities from each member economy are taken into account, the resulting topology would be a more complete, smooth, and short ring shape.

The ring, as we predict, turned out to be more complete when we select two cities than just one. The most important fact here is that Jakarta and Kuala Lumpur are given independent connections from a ring due to their difficulties in accessing via Singapore, as is the case for Hong Kong, China and Chinese Taipei.

Another significant observation is the establishment of a ring type connection that ran from Hong Kong, China through Viet Nam; Thailand; Singapore; Brunei Darussalam; and the Philippines and back to Hong Kong, China.



< Figure 30> Combined Model-2 S.E. Asia Only

Chapter 5

Conclusions

The purpose of this study is to construct a regional research or a simplified Internet network among APEC members at a minimum cost. The primary goal in this study is to examine the most effective way of constructing a network by taking into account following three aspects: First, Asian member economies are still at a developing stage. Secondly, it is difficult for the members to mobilize enough financial resources to establish a network on their own and lastly, international cooperation is greatly demanded as members seek to build an effective regional network.

When one economy establishes an international linkage, it only considers its national circumstances, based on this performs bilateral negotiations with targeted international partners.

In general, the cost of connecting a link is shared equally between two partners. However, there are often exceptions where an Asian member economy that wishes to establish a connection with the United States has to pay for the full circuit, contrary to the general cost-sharing rule. To solve this problem, economies without sufficient financial resources can review ways to use partial detouring connections through regional cooperation, instead of individual direct connections with the United States.

This study will try to design an effective network, which surpasses national boundaries and focuses on regional interests. For example, developing economies mainly concerned with economic aspects will be encouraged to construct one or multi-collecting nodes in the region, connect the United States via these collecting nodes, and share the cost for increased cost-effectiveness.

In reality, however, what we stated above is no easy problem to tackle. If, for example, a economy's traffic had a detour route through a third economy, a new question of how the expenses will be shared would be raised and the problem will become much more complex. But

for the simplicity of this study, we will set aside this particular problem for the time being and dwell upon the ways to reduce cost while taking into consideration the traffic in the Asian region

In the end, the keys to solve these problems lie in the hands of international organizations. Research such as the one done here will help surpass national boundaries, present appropriate plans for network-related policies, and give a sense of direction for regional developments realized through cooperation between member economies.

This study aims to find a plan to establish the most suitable Asian network at a minimum cost. It is, however, not aimed at emphasizing the network's usage, for example, the question of is it a research or commercial purpose. Recalling the final aim of this project, the first step taken is selecting cities. To select candidate cities, data regarding submarine cables, domestic infrastructure, population size, and the penetration rate of PSTN are considered and based on such criteria, two to three finally chosen cities from each member are selected for later simulation.

The current network status and domestic connectivity are also reviewed, as well as the submarine cable environments including landing station. Next, to help select the cities, the population of each major city is considered as additional data. For the final decision, a comprehensive judgment is made based on the three factors mentioned above. Through these selection processes, two to three candidate cities for each member economy are finally chosen. In the next phase, submarine cables are connected with the cities using the computer simulation.

As for China, Beijing is selected as first choice, Shanghai as second and Guangzhou as third. For Indonesia, the first choice is Jakarta where all three conditions met our requirements. The second choice is Surabaya/Malang, which also satisfies the three conditions mentioned here. In case of Bandung and Semarang, the two cities required connection through terrestrial research networks rather than submarine cables due to geographical reasons and therefore are deemed inadequate as candidate cities. For Japan, Tokyo and Nagoya are selected as adequate cities and Sendai joined the last in rank. For Korea, if only one city is to be selected, then it is definitely Seoul. If including a second choice, they are Seoul and Pusan and the third choice would eventually be Gwangju. For Malaysia, despite its disadvantage of not currently possessing any landing stations, the first city of choice is the economy's capital, Kuala Lumpur, with the biggest population. The second choice is Johor Baharu, which has fairly easy access to submarine cables and is the second largest city. The last option would be Kuantan, which lacks domestic connectivity but has a landing station. For the Philippines, the best city for this project is Manila. Cebu would be the next appropriate choice since Cebu ranks third in terms of the size

of the population and is located near South China Sea, which would help save construction cost. Davao, despite its population size that ranks second, is excluded from the choice since it has no landing accessibility and is geographically far away from China Sea. For Thailand, Bangkok, which is near the city of Phetchaburi, is the first choice. The second choice is Songkhla. The research network revealed that Hanoi and Ho Chi Min are the two important regional nodes in Viet Nam. But there is a problem with the landing station, the city of Danang, situated between Hanoi and Ho Chi Min. Disregarding this problem the first candidate would be Hanoi and the second, Ho Chi Min.

Computer simulation process is continued with the completed selection of cities. To fulfill our aim, the problem has to be simplified and computerized solution has to be based on these simplified assumptions, rendering to some extent a limitation on the study.

The first assumption considered all APEC member economies to be connected only via submarine cables. If this research included connections also via terrestrial cables and satellite systems, multiple scenarios would have to be considered, which would further complicate the program. In order to avoid computer simulation failure, such complexity is ruled out from the first place and the research is carried out with the assumption that there are only submarine cable connections, with no other connections such as satellite or terrestrial cables.

The second assumption considered distance to be the most important factor in determining cost. In the real world, however, distance is not the only important factor in determining the cost: Market price, which may differ according to market structures, also plays an essential role when it comes to cost. That is to say that the price of the two circuits in the same distance can vary depending on the monopoly or market situation. However, for the sake of simplicity, this research will only consider distance to be the main factor in determining cost.

The third assumption considered all member economies to have the same technological level. In reality, the circuit cost can differ according to the technology level of each economy and enterpriser. However, in this study, the same technological level is applied for all members.

In the fourth assumption, the cost function of SEA-ME-WE is adopted to reflect the real cost scheme and the cost is calculated according to distance. This assumption is explained in details in Chapter 4.

Last assumption considered a new way to find an optimal solution in designing a network that is cost-effective. As mentioned earlier in Chapter 3, the most important factor when designing a network according to the traditional approach is to consider the multi-connectivity of nodes. But since it is possible to secure sufficient bandwidth, this approach risks the problem of not taking into account the routing issue. This traditional approach can be valid when a single

member economy/company designs the network and aims to handle all possible traffic with sufficient bandwidth. But it is not a realistic scenario and cannot be applied to the Asian region, where most member economies/companies rather prefer to minimize cost than secure links for future traffic.

Therefore this study takes a comprehensive approach which includes both the traditional approach mentioned in Chapter 3 and the creative approach devised in this study to consider necessary future routing aspects (Chapter 4 has details on this combined model). The new approach adopted in Chapter 4 is different from the existing, traditional one in that it takes into account the possible amount of traffic and routing. For this purpose, an appropriate model is newly created to reflect the amount of traffic between nodes.

Despite certain limitations in the study, as mentioned earlier, Chapter 3 and 4 produced results that are worth noting.

When examining the Multi-Connectivity model in Chapter 3, our assumption produced two different kinds of results. The first result came from the two-connectivity assumption, which binds together the entire region into one ring topology. In fact, these figures are not new in the two-connectivity assumption cases. The second assumption forms a complete ring in South East Asia and a 'Y' shape in North East Asia. The second observation is the result of the assumption that considers two regional networks: South and North East Asia. It shows that Chinese Taipei and Hong Kong, China could serve as partial hub points. Of the two cities, Chinese Taipei, because of its geographical importance, stood out. Observation also shows that the size of the ring became smaller and smoother as our assumption shifts from one that focuses only on connecting the capital cities to one that considers the connection of any candidate cities that are competitive, thus, naturally canceling out those capital cities with inadequate locations.

The three following cases are considered when the computer simulation is run with only capital cities using traditional Multi-connectivity algorithm: First, all capital nodes are considered as special nodes. Second, Korea; China; and Japan are considered as special nodes, while others are all ordinary nodes. Last, Korea; China; Japan; and Singapore are considered special nodes, while others are ordinary nodes.

The first case produced nodes that are connected as a single ring while the second case separated the network into two parts, a single ring in the South East and a triangle in the North East Asia. The last case showcased a partial ring with edges that branched out at Singapore. Among the three cases, the second showed the most similar configuration to Chapter 4 and is most important in that the assumption is very realistic.

The Combined model in Chapter 4, unlike the Multi-connectivity model in Chapter 3,

presents a model that does take into account the flow of traffic and routing. Results generally show a topology that divides the North East and South East Asian connection with Chinese Taipei in the center. Another notable result is the South East Asian topology from the Combined Model. It is somewhat different from the two (Multi)-connectivity model in that the South East Asian connection shows a relatively incomplete ring. In other words, a ring starts from Hong Kong, China; China through Viet Nam; Thailand; Singapore; Brunei Darussalam; and the Philippines to Chinese Taipei and back to Hong Kong, China with two edges spreading from Singapore to Jakarta and from Singapore to Kuala Lumpur. This configuration becomes more distinct as alternative cities that have better locations are considered. For instance, a ring type is formed in the South East Asian region while either a triangular or “Y” shape is formed in North Eastern side. The fact that Singapore had two edges spreading separately, one to Indonesia and the other to Brunei Darussalam, rather than a single edge that connected from Singapore through Jakarta to Brunei Darussalam shows how the closeness of Singapore and Brunei Darussalam played an important role in shaping the outcome.

Another noteworthy observation is that there are no connecting edges between Hanoi and Bangkok. When considering the cost-effectiveness of the connection in terms of the distance, there should have been a link between these two cities, however, such connection is ruled out because of the lack of exchanges of data between the two cities.

The same applies for Chinese Taipei. When considering the distance, Chinese Taipei makes a perfect connecting point for North and South East Asia. However, the Philippines got selected for the same reason that it has more data transactions than Chinese Taipei.

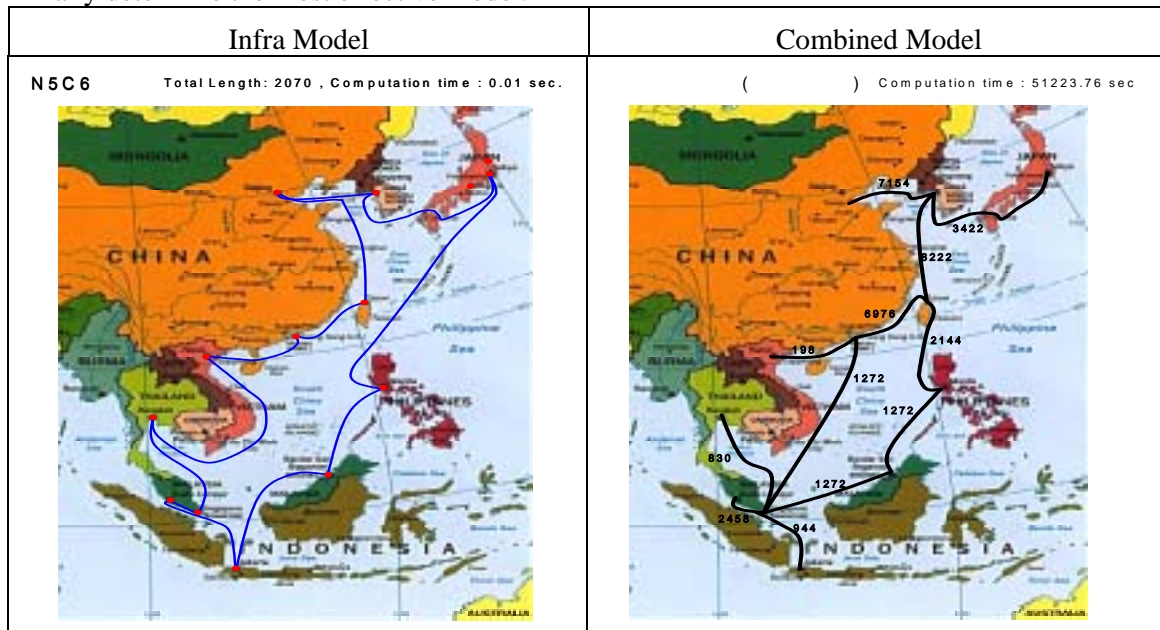
This study also compared each case to find out if there are any similarities between the cases. The results of the Multi-connectivity model and the Combined model are good tools to compare the similarities and differences of such cases. Here, the results of the scenario that only focused on the connections among capital cities are compared. Multi-connectivity and Capital city case of N5C7 (Multi-connectivity Model-Capital scenario 2) and N5C8 (Multi-connectivity Model-Capital scenario 3) are hence selected and the diagram from the Combined Model is compared since both are based on the same assumption.

The result showed that close similarities are demonstrated by the figure N5C7 and the Combined Model. In both cases, Singapore and Hong Kong, China played an important role as partial hubs. The only two differences included the edge between Honk Kong, China and Singapore, which is present in N5C7 but not in the Combined Model and the edge between the Philippines and Hong Kong, China versus the edge between the Philippines and Chinese Taipei.

Another difference between the two models is the difference in their presumptions. The

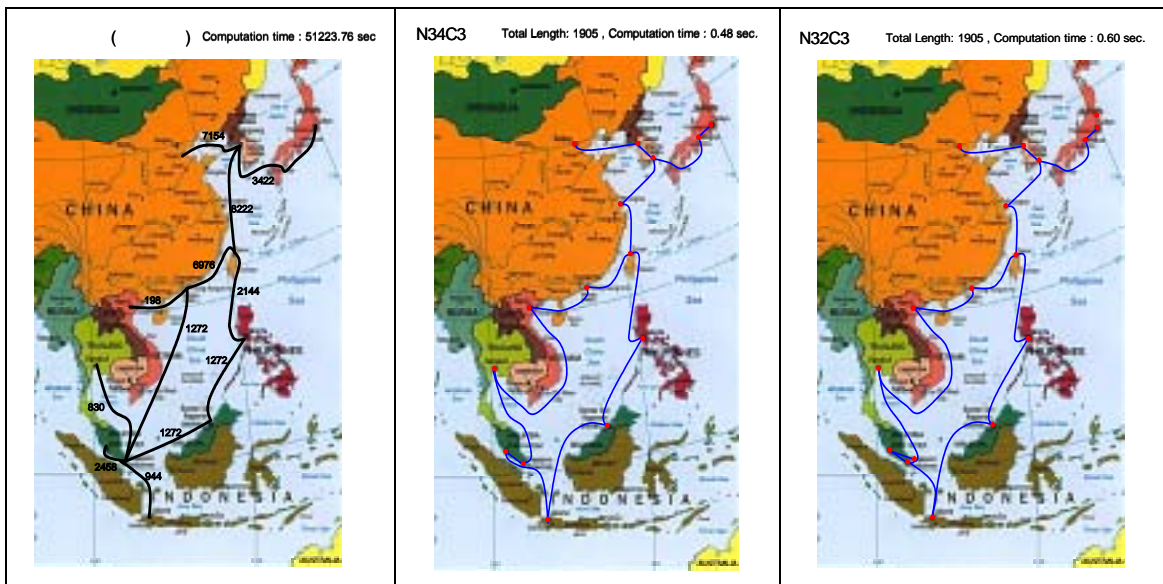
Combined Model shows the connectivity and the size of the connectivity, whereas the Multi-Connectivity Model only shows the connectivity. This is already explained above in details.

Our main goal of this study is to find the most frequently recurring, robust diagram from a number of results that we derive from our hypotheses. This kind of approach is taken up because it is not easy to decide which assumption would be most realistic and applicable when all of them are logically convincing. Thus, it seems only appropriate to seek similarities and examine the most frequently recurring and robust diagram from the results of our hypothesis to finally determine the most effective model.



<Figure 31> Comparison of Capital Scenarios

The three diagrams below look very similar. The most frequently recurring diagram with reasonable conditions is chosen as the most robust and appropriate network configuration. In all three diagrams below, a circle starting down from Chinese Taipei extending to the South East Asian region is formed and although the North East Asian region does not form a perfect ring, there exists a connection centered round Korea. The only difference in the diagrams is the connection in Viet Nam and Indonesia.



<Figure 32> Frequently Recurring Figures

The most imminent issue at hand is the issue of designing a South East Asian network. This is particularly true at a time when many discussions on the construction of a regional network led by Trans Eurasia Information Network and the expansion of APII testbeds are underway. The problem is constructing a regional network in a practical way. In this respect, this study tries to offer a clear solution to these problems. The results of these experiments clearly show how a partial ring topology in the South East Asian region could prove to be the most effective model.



< Figure 33> Network topologies in S.E. Asia

The suggested ideal topology is a deviation from a complete ring with direct connections from Singapore to Kuala Lumpur and from Singapore to Jakarta, places that are located relatively far away from each other to form a convenient ring. The rest of the cities can be connected through a ring configuration.

Another conclusion derived from several final tests show that the network configuration may change according to different purposes and assumptions. Therefore, if the purposes and limitations can be set variables when designing a network, a more clear-cut solution will be available. However, it is hard to provide clear conditions and assumptions in most cases. Thus, in the early designing phase, limitations must be taken into full consideration and meticulous comparisons must to be carried out.

Overall, the Asian network configuration becomes effective when the South East Asia forms a partial ring and North East Asia groups into a 'y' form. Such forms are chosen because they showcase the best average score under many different assumptions. A more efficient design can be derived if results from the combined Model, in which branching out at important nodes are permitted, is applied.

Reference

1. Ahuja, R.K., Magnant, T.L. and Orlin, J.B., Network Flows: Theory, Algorithm, and Applications, Prince-Hall, 1992
2. Cardwell, R.H., C.L. Monma, T. Wu, 1989, "Computer-aided design procedures for survivable fiber optic networks", IEEE JSAC 7, 1188-1197.
3. Cosares, S., N.D. Deutsch, I. Saniee, O.J. Wasem, 1995, "SONET toolkit: A decision support system for designing robust and cost-effective fiber-optic networks", Interfaces 25, 20-40.
4. Garey, M.R. and Johnson, D.S., Computer and Intractability: A Guide to the Theory of NP-completeness, Bell Lab, Murray Hill, 1979
5. Grotschel, M., C.L. Monma, 1990, "Integer polyhedra arising from certain network design problems with connectivity constraints", SIAM J. Disc. Math. 3, 502-524.
6. Grotschel, M., C.L. Monma, M. Stoer, 1992, "Facets for polyhedra arising in the design of communication networks with low-connectivity constraints", SIAM J. Optim. 2, 474-504.
7. Grotschel, M., C.L. Monma, M. Stoer, 1992, "Computational results with a cutting plane algorithm for designing communication networks", Operations Research 40, 309-330.
8. Monma, C.L., D.F. Shallcross, 1989, Methods for designing communication networks with certain two-connected survivability constraints", Operations Research 37, 531-541.
9. Park, SoonDal, A theory of OR and Practices, Min Young Sa, 1996

Websites

1. <http://www.apectel.org>
2. <http://www.aptsec.org>
3. <http://www.canarie.ca>
4. <http://www.ccert.edu.cn>
5. <http://www.co.jp> and <http://www.nii.ac.jp>
6. <http://www.koren21.net>
7. <http://www.marine.francetelecom.fr/english/frames/offreglo/pose/pose.htm>
8. <http://www.mic.go.kr>
9. <http://www.singaren.net.sg>

10. <http://www.transeurasia.org>
11. <http://www.transeurasia.org/expert/back.asp>
12. <http://www.ucaid.edu/ebilene>
13. <http://www.vbns.net>

Appendix

1. Combined Model

source code : tcm.c , input file : apec2.dat , output file : result.out

```
#include "cplex.h"
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include <sys/times.h>
#include <unistd.h>
#define BIGNUM 2.0E+4
#define TRUE 1
#define FALSE 0
#define SCALE 32768.0
#define MAXNODE 20
#define MAXARC 300
#define MAXCOMM 300
#define float double
#define colspace 70000
#define rowSPACE 5000
#define nzspace 400000
/* variable for CPLEX */
CPXENVptr env = NULL;
CPXLPptr lp = NULL;
int status;
char *probname = "colgen";
int numcols = 0;
```



```

int numrows = 0;
int objsen = 0;
double obj[colspace]={0,};
double rhs[rowspace]={0,};
char sense[rowspace]={0,};
int matbeg[colspace]={0,};
int matcnt[colspace]={0,};
int matind[nzspace]={0,};
double matval[nzspace]={0,};
double lb[colspace]={0,};
double ub[colspace]={0,};
char ctype[colspace]={0,};

int head[MAXARC], tail[MAXARC], N , NEdge, PATHIDX,
source[MAXCOMM],sink[MAXCOMM] ;
int count, NoComm ;
float sol[MAXARC],slack[MAXARC];
double obj_value,totaltime;
float cost[MAXARC],demand[MAXCOMM], fixedcost[MAXARC] ;
FILE *ft ;
FILE *fin ;

void setPD(int *numcols_p , int *numrows_p , int *objsen_p , double *obj , double *rhs , char *sense,int
*matbeg,int *matcnt , int *matind , double *matval ,double *lb , double *ub , char *ctype);

float timer(mode)
int mode;
{
static long clktck=0L;
static struct tms tstart,tend;
static clock_t start,end;

if(clktck == 0L) {
clktck = sysconf(_SC_CLK_TCK);
if(clktck < 0) {
fprintf(stdout, "\nError SYS TIME CONFIG.\n");
}
}
}

```

```

        exit(1);
    }
}
if(mode == 1) {
    if((start = times(&tstart)) == -1) {
        fprintf(stdout, "\nError in TSTART\n");
        exit(1);
    }
}
else if(mode == 2) {
    if((end = times(&tend)) == -1) {
        fprintf(stdout, "\nError in TEND\n");
        exit(1);
    }
    return (tend.tms_utime-tstart.tms_utime)/(double)clktck;
}
}

main()
{
    int k, i, j;
    /* START OF INPUT DATA */
    env = CPXopenCPLEX(&status);
    if ( env == NULL )
    {
        char errmsg[1024];
        fprintf (stderr, "could not open CPLEX environment.\n");
        CPXgeterrorstring (env, status, errmsg);
        fprintf (stderr, "%s", errmsg);
    }

    if (status !=0)
    {
        fprintf(stderr, "Failure to turn on screen indicator, error %d.\n", status);
        exit(1);
    }
}

```

```

}

if (status != 0)
{
    fprintf(stderr, "Failure to set tree memory limit, error %d.\n", status);
    exit(1);
}

fin=fopen("apec2.dat", "rt") ;
fscanf(fin, "%d %d %d", &N, &NEdge, &NoComm);

for (i=1 ; i <=NEdge ; i++)
{
    fscanf(fin, "%d", &tail[i]);
    fscanf(fin, "%d", &head[i]);
    fscanf(fin, "%lf", &cost[i]);
    fscanf(fin, "%lf\n", &fixedcost[i]);
}
for (k=NEdge+1; k <=2*NEdge ; k++) /* Symmetric arc cost */
{
    tail[k] = head[k-NEdge];
    head[k] = tail[k-NEdge];
    cost[k] = cost[k-NEdge];
    fixedcost[k] = fixedcost[k-NEdge];
}
for (k=1 ; k <=NoComm ; k++)
{
    fscanf(fin, "%d", &source[k]);
    fscanf(fin, "%d", &sink[k]);
    fscanf(fin, "%lf\n", &demand[k]);
}
for (k=NoComm+1; k <=2*NoComm ; k++) /* Symmetric demand */
{
    source[k] = sink[k-NoComm];

```

```

        sink[k] = source[k-NoComm];
        demand[k] = demand[k-NoComm];
    }
    fclose(fin);
    ft=fopen("result.out","wt");
    NEdge = 2*NEdge ;
    NoComm = 2*NoComm ;
    setPD(&numcols,&numrows,&objsen,obj,rhs,sense,matbeg,matcnt,matind,matval,lb,ub,ctype);
    lp=
CPXloadlp(env,probname,numcols,numrows,objsen,obj,rhs,sense,matbeg,matcnt, ,matind,matval, lb,ub,
        NULL,colspace, rowspace,nzspace);
    status = CPXloadctype(env, lp, ctype);
    CPXlpwrite(env, lp, "colgen.lp");
    status = CPXsetintparam(env, CPX_PARAM_SCRIND,CPX_ON);
    timer(1);
    status = CPXmipoptimize(env ,lp);
    totaltime = timer(2);
    status = CPXgetmx(env, lp, sol, NEdge*NoComm , NEdge+NEdge*NoComm-1);
    status = CPXgetmslack(env,lp,slack,N*NoComm,N*NoComm+NEdge-1);
    status = CPXgetmipobjval(env,lp,&obj_value);
    for(i=0 ; i<=NEdge-1 ;i++)
        {
            if(sol[i]>0.001)
                {
                    fprintf(ft,"the arc from %d to %d , the bandwidth is %lf\n",tail[i+1],head[i+1],BIGNUM-
slack[i]);
                }
        }
    fprintf(ft,"the objective value is %lf\n",obj_value);
    fprintf(ft,"the time is %lf\n",totaltime);
    fclose(ft);
    if ( lp != NULL )
    {
        status = CPXfreeprob (env, &lp);
        if ( status )
        {

```

```

        fprintf(stderr, "CPXfreeprob failed, error code %d.\n", status);
    }
}

if ( env != NULL )
{
status = CPXcloseCPLEX (&env);
if ( status )
    {
        char  errmsg[1024];
        fprintf (stderr, "Could not close CPLEX environment.\n");
        CPXgeterrorstring (env, status, errmsg);
        fprintf (stderr, "%s", errmsg);
    }
}
return (status);
}

void setPD(int *numcols, int *numrows, int *objsen, double *obj, double *rhs, char *sense, int *matbeg,
int *matcnt, int *matind, double *matval, double *lb, double *ub, char *ctype)
{
    int i=0, j=0, k=0 , s=0 ;
    int colindex, colindex2, rightindex, nzcom ;
    *numcols= NEdge + NEdge*NoComm;
    *numrows= NEdge + NoComm*N ;
    *objsen=1;

    nzcom=0;
    for(k=1;k<=NoComm ; k++)
        for (i=1;i<=NEdge;i++)
            {
                obj[nzcom]=cost[i] ;
                nzcom++;
            }
    for (i=1;i<=NEdge;i++)
        obj[nzcom+i-1]= fixedcost[i] ;
}

```

```

nzcom = 0 ;
colindex = 0 ;

for(k=1 ; k<=NoComm ; k++)
  for(i=1 ; i<=NEdge ; i++)
    {
      matbeg[colindex] = 3*colindex ;
      matcnt[colindex] = 3 ;
      if(tail[i] < head[i])
        {
          matind[nzcom]=(k-1)*N+tail[i]-1 ;
          matval[nzcom]=1.0 ;
          matind[nzcom+1]=(k-1)*N+head[i]-1 ;
          matval[nzcom+1]=-1.0 ;
        }
      if(tail[i] > head[i])
        {
          matind[nzcom]=(k-1)*N+head[i]-1 ;
          matval[nzcom]=-1.0 ;
          matind[nzcom+1]=(k-1)*N+tail[i]-1 ;
          matval[nzcom+1]=1.0 ;
        }
      colindex++;
      matind[nzcom+2] = NoComm*N + i -1 ;
      matval[nzcom+2] = 1.0 ;
      nzcom = nzcom + 3 ;
    }
colindex2 = 3*colindex;
for(i=1;i<=NEdge;i++)
{
  matbeg[colindex] = colindex2 ;
  matcnt[colindex] = 1 ;
  matind[nzcom] = NoComm*N + i -1 ;
  matval[nzcom] = -BIGNUM ;
  colindex2++;
}

```

```

        colindex++;
        nzcom++;
    }

    for(i=0;i<=NoComm*NEdge-1 ; i++)
    {
        lb[i] = 0.0;
        ub[i] =INFBOUND;
        ctype[i]='C' ;
    }

    for(i=0 ; i<= NEdge-1 ; i++)
    {
        lb[NoComm*NEdge+i] = 0.0 ;
        ub[NoComm*NEdge+i] = 1.0 ;
        ctype[NoComm*NEdge+i]='B' ;
    }
    rightindex = 0;
    for(k=1;k<=NoComm ;k++)
        for(i=1 ; i<=N ; i++)
            {
                sense[rightindex] = 'E' ;
                rhs[rightindex]=0;
                if(i==source[k])
                    rhs[rightindex] = 0+demand[k] ;
                if(i==sink[k])
                    rhs[rightindex] = 0-demand[k] ;
                rightindex++;
            }

    for(i=NoComm*N ; i<= NoComm*N + NEdge -1 ; i++)
    {
        sense[i] = 'L';
        rhs[i] = 0.0;
    }
}

```

2. Combined Model/Multi City Model

source code : tcm2.c , input file : apec2.dat , output file : result.out

```
#include "cplex.h"
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include <sys/times.h>
#include <unistd.h>
#define BIGNUM 2.0E+4
#define TRUE 1
#define FALSE 0
#define SCALE 32768.0
#define MAXNODE 15
#define MAXARC 200
#define MAXCOMM 200
#define float double
#define colspace 35000
#define rowspace 4000
#define nzspace 120000
```

```
/* variable for CPLEX */
```

```
CPXENVptr env = NULL;
```

```
CPXLPptr lp = NULL;
```

```
int status;
```

```
char *probname = "colgen";
```

```
int numcols = 0;
```

```
int numrows = 0;
```

```
int objsen = 0;
```

```
double obj[colspace]={0,};
```

```
double rhs[rowspace]={0,};
```

```
char sense[rowspace]={0,};
```



```

int matbeg[colspace]={0,};
int matcnt[colspace]={0,};
int matind[nzspace]={0,};
double matval[nzspace]={0,};
double lb[colspace]={0,};
double ub[colspace]={0,};
char ctype[colspace]={0,};

/*General Variables */

int head[MAXARC], tail[MAXARC], N , NEdge, PATHIDX, source[MAXCOMM],sink[MAXCOMM] ,
NoNationComm ;
int      count,      NoComm      ,Npoint,Ngroup,nsource[MAXCOMM],nsink[MAXCOMM],
nationlabel[MAXNODE] ;
float sol[MAXARC],slack[MAXARC];
double obj_value,totaltime;
float cost[MAXARC],demand[MAXCOMM],fixedcost[MAXARC] ;
int demandlabel[MAXNODE][MAXNODE];

FILE *ft ;
FILE *fin ;

void setPD(int *numcols_p , int *numrows_p , int *objsen_p , double *obj , double *rhs , char *sense,int
*matbeg,int *matcnt , int *matind , double *matval ,double *lb , double *ub , char *ctype);

float timer(mode)
int mode;
{
    static long clktck=0L;
    static struct tms tstart,tend;
    static clock_t start,end;

    if(clktck == 0L) {
        clktck = sysconf(_SC_CLK_TCK);
        if(clktck < 0) {

```

```

    fprintf(stdout, "\nError SYS TIME CONFIG.\n");
    exit(1);
}
}
if(mode == 1) {
    if((start = times(&tstart)) == -1) {
        fprintf(stdout, "\nError in TSTART\n");
        exit(1);
    }
}
else if(mode == 2) {
    if((end = times(&tend)) == -1) {
        fprintf(stdout, "\nError in TEND\n");
        exit(1);
    }
    return (tend.tms_utime-tstart.tms_utime)/(double)clktck;
}
}

main()
{
    int k, i, j, demandindex ;
    /* START OF INPUT DATA */
    env = CPXopenCPLEX(&status);
    if ( env == NULL )
    {
        char errmsg[1024];
        fprintf (stderr, "could not open CPLEX environment.\n");
        CPXgeterrorstring (env, status, errmsg);
        fprintf (stderr, "%s", errmsg);
    }

    if (status !=0)
    {
        fprintf(stderr, "Failure to turn on screen indicator, error %d.\n", status);
    }
}

```

```

        exit(1);
    }

if (status != 0)
{
    fprintf(stderr, "Failure to set tree memory limit, error %d.\n", status);
    exit(1);
}

/*Input Data */

fin=fopen("apec2.dat", "rt") ;
fscanf(fin, "%d %d %d %d", &N, &NEdge, &NoComm, &NoNationComm);
for (i=1 ; i <=NEdge ; i++)
{
    fscanf(fin, "%d", &tail[i]);
    fscanf(fin, "%d", &head[i]);
    fscanf(fin, "%lf", &cost[i]);
    fscanf(fin, "%lf\n", &fixedcost[i]);
}
for (k=NEdge+1; k <=2*NEdge ; k++) /* Symmetric arc cost */
{
    tail[k] = head[k-NEdge];
    head[k] = tail[k-NEdge];
    cost[k] = cost[k-NEdge];
    fixedcost[k] = fixedcost[k-NEdge];
}
for (k=1 ; k <=NoComm ; k++)
{
    fscanf(fin, "%d", &source[k]);
    fscanf(fin, "%d", &sink[k]);
}
for (k=NoComm+1; k <=2*NoComm ; k++)
{
    source[k] = sink[k-NoComm];
}

```

```

        sink[k] = source[k-NoComm];
    }

    for (k=1; k <= N ; k++)
    {
        fscanf(fin,"%d\n",&nationlabel[k]);
    }

    demandindex = 1;
    for (k=1 ; k <=NoNationComm ; k++)
    {
        fscanf(fin,"%d %d %f\n",&nsource[k],&nsink[k],&demand[k]);
        demand[NoNationComm+k] = demand[k];
        demandlabel[nsource[k]][nsink[k]] = demandindex ;
        demandlabel[nsink[k]][nsource[k]] = NoNationComm + demandindex ;
        demandindex++;
    }
    fclose(fin);
    ft=fopen("result.out","wt") ;
    NEdge = 2*NEdge ;
    NoComm = 2*NoComm ;
    NoNationComm = 2*NoNationComm ;
    /* Solve Problem */
    setPD(&numcols,&numrows,&objsen,obj,rhs,sense,matbeg,matcnt,matind,matval,lb,ub,ctype);
    lp
CPXloadlp(env,probname,numcols,numrows,objsen,obj,rhs,sense,matbeg,matcnt,matind,matval,lb,ub,
NULL,colspace,rowspace,nzspace);
    status = CPXloadctype(env, lp, ctype);
    CPXlpwrite(env, lp, "colgen.lp");
    status = CPXsetintparam(env, CPX_PARAM_SCRIND,CPX_ON);
/*    status = CPXsetintparam(env,CPX_PARAM_NDLIM,500);
    status = CPXsetdblparam(env,CPX_PARAM_EPGAP,0.05);
    status = CPXsetdblparam(env,CPX_PARAM_EPINT,0.00001);
    status = CPXsetdblparam(env,CPX_PARAM_CUTUP,1366545);*/
    timer(1);
    status = CPXmipoptimize(env ,lp);

```

```

totaltime = timer(2) ;
status = CPXgetmx(env, lp, sol, NEdge*NoComm , NEdge+NEdge*NoComm-1);
status = CPXgetmslack(env,lp,slack,N*NoComm,N*NoComm+NEdge-1);
status = CPXgetmipobjval(env,lp,&obj_value);

/* Output the Result */
for(i=0 ; i<=NEdge-1 ;i++)
    {
        if(sol[i]>0.001)
            {
                fprintf(ft,"the arc from %d to %d , the bandwidth is %lf\n",tail[i+1],head[i+1],BIGNUM-
slack[i]);
            }
    }
    fprintf(ft,"the objective value is %lf\n",obj_value);
    fprintf(ft,"the time is %lf\n",totaltime);
    fclose(ft);
    if ( lp != NULL )
    {
        status = CPXfreeprob (env, &lp);
        if ( status )
        {
            fprintf(stderr, "CPXfreeprob failed, error code %d.\n", status);
        }
    }

    if ( env != NULL )
    {
        status = CPXcloseCPLEX (&env);
        if ( status )
        {
            char  errmsg[1024];
            fprintf (stderr, "Could not close CPLEX environment.\n");
            CPXgeterrorstring (env, status, errmsg);
            fprintf (stderr, "%s", errmsg);
        }
    }

```

```

    }
    return (status);
}

void setPD(int *numcols, int *numrows, int *objsen, double *obj, double *rhs, char *sense, int *matbeg,
int *matcnt, int *matind, double *matval, double *lb, double *ub, char *ctype)
{
    int i=0, j=0, k=0, s=0 ;
    int colindex, colindex2, rightindex, nzcom, cntindex ;
    *numcols= NEdge + NEdge*NoComm + NoComm ;
    *numrows= NEdge + NoComm*N + NoNationComm ;
    *objsen=1;

    nzcom=0;
    for(k=1; k<=NoComm ; k++)
        for (i=1; i<=NEdge; i++)
            {
                obj[nzcom]=cost[i] ;
                nzcom++;
            }
    for (i=1; i<=NEdge; i++)
        {
            obj[nzcom+i-1]= fixedcost[i] ;
        }
    for (i=1; i<=NoComm; i++)
        obj[NEdge + NEdge*NoComm+i-1]= 0 ;

    nzcom = 0 ;
    colindex = 0 ;

    for(k=1 ; k<=NoComm ; k++)
        for(i=1 ; i<=NEdge ; i++)
            {
                matbeg[colindex] = 3*colindex ;
                matcnt[colindex] = 3 ;
                if(tail[i] < head[i])

```

```

    {
        matind[nzcom]=(k-1)*N+tail[i]-1 ;
        matval[nzcom]=1.0 ;
        matind[nzcom+1]=(k-1)*N+head[i]-1 ;
        matval[nzcom+1]=-1.0 ;
    }
    if(tail[i] > head[i])
    {
        matind[nzcom]=(k-1)*N+head[i]-1 ;
        matval[nzcom]=-1.0 ;
        matind[nzcom+1]=(k-1)*N+tail[i]-1 ;
        matval[nzcom+1]=1.0 ;
    }
    colindex++;
    matind[nzcom+2] = NoComm*N + i -1 ;
    matval[nzcom+2] = 1.0 ;
    nzcom = nzcom + 3 ;
}
colindex2 = 3*colindex;
for(i=1;i<=NEdge;i++)
{
    matbeg[colindex] = colindex2 ;
    matcnt[colindex] = 1 ;
    matind[nzcom] = NoComm*N + i -1 ;
    matval[nzcom] = -BIGNUM ;
    colindex2++;
    colindex++;
    nzcom++;
}

for(k=1;k<=NoComm ;k++)
{
    cntindex = 0 ;
    matbeg[colindex] = colindex2 ;
    for(i=1 ; i<=N ; i++)
    {

```

```

    if(i==source[k])
    {
        matind[nzcom] = N*(k-1)+source[k] -1 ;
        matval[nzcom] = -1.0 ;
        nzcom++;
        colindex2++;
        cntindex++;
    }
    if(i==sink[k])
    {
        matind[nzcom] = N*(k-1) + sink[k] -1 ;
        matval[nzcom] = 1.0 ;
        nzcom++;
        colindex2++;
        cntindex++;
    }
}
if(nationlabel[source[k]] != nationlabel[sink[k]])
{
    matind[nzcom] = NoComm*N + NEdge +
demandlabel[nationlabel[source[k]]][nationlabel[sink[k]]]-1 ;
    matval[nzcom] = 1.0 ;
    nzcom++;
    colindex2++;
    cntindex++;
}
matcnt[colindex] = cntindex ;
colindex++;
}

for(i=0;i<=NoComm*NEdge-1 ; i++)
{
    lb[i] = 0.0;
    ub[i] =INFBOUND;
    ctype[i]='C' ;
}

```



```

for(i=0 ; i<= NEdge-1 ; i++)
{
    lb[NoComm*NEdge+i] = 0.0 ;
    ub[NoComm*NEdge+i] = 1.0 ;
    ctype[NoComm*NEdge+i]='B' ;
}

for(i=0 ; i<= NoComm-1 ; i++)
{
    lb[NoComm*NEdge+NEdge+i] = 0.0 ;
    ub[NoComm*NEdge+NEdge+i] = INFBOUND ;
    ctype[NoComm*NEdge+NEdge+i]='C' ;
}

rightindex = 0;
for(k=1;k<=NoComm ;k++)
    for(i=1 ; i<=N ; i++)
        {
            sense[rightindex] = 'E' ;
            rhs[rightindex]=0;
            rightindex++;
        }
for(i=NoComm*N ; i<= NoComm*N + NEdge -1 ; i++)
{
    sense[i] = 'L';
    rhs[i] = 0.0;
}

for(i=NoComm*N + NEdge ; i<= NoComm*N + NEdge + NoNationComm -1 ; i++)
{
    sense[i] = 'E';
    rhs[i] = demand[i - NoComm*N - NEdge +1];
}
}

```

3. Multi-connectivity Model

A. 2CON.C

```
#include <ilcplex\cplex.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>

#define MAX    100
#define MAXEDGE 2000
#define MAC    100
#define MAR    100
#define MACSZ  MAC*20
#define MARSZ  MAR*100
#define MATSZ  1000000
#define TLNC   1e-06
#define NDLIMIT 500000

#define YES 1
#define NO 0

// #define LP FORM // listing LP formulation
// #define CHECK // checking CPLEX error
// #define DEBUG // Branch-and-cut node

/* CPLEX - part I arguments */

char *probname = "edge con";
int mac ;
int mar ;
int objsen = 1; /* minimize */
double objx[MACSZ] ;
double rhsx[MARSZ] ;
char senx[MARSZ] ;
int matbeg[MACSZ] ;
int matcnt[MACSZ] ;
int matind[MATSZ] ;
```

```

double  matval[MATSZ] ;
double  bd[MACSZ] ;
double  bdu[MACSZ] ;
int     lpstat;
double  obj;
double  x[MACSZ];
double  pi[MARSZ];
double  slack[MARSZ];
double  dj[MACSZ];

double xx[MACSZ] ; /* inclumbant solution's home */
char  cstore[MATSZ];  char *cname[MACSZ];

/**** add row argument *****/
int  ccnt = 0 , rcnt = 1, nzcnt = 0, rmatbeg = 0, rmatind[MAXEDGE] ;
double  rhs, rmatval[MAXEDGE] ;
char  sense ;

/***** main definitions *****/

int  G[MAX][MAX] ;    /* underlying network incidence matrix */
int  Cij[MAX][MAX] ; /* link costs */
int  Fixed[MAXEDGE] ;
int  req[MAX] ;      /* node requirements : can be used for node classification */
int  no of nodes ;
int  no specials ;
int  no steiners ;
int  no of links ;
int  no fixed links ;

int  no of cons; /* no of constraints */
int  orig cons;
int  no lp;
int  no of vars ;
int  no of cuts = 0 ;
int  no lp0 ;

```

```

int no bnc nodes ;

char city[MAX][25];
char node option[10];

double upper sol = -1000000, lower sol, usol = -1000000;
double locut = 0.0, first, second ;

clock t time start; // checking time.

/* Main Constraints Separation argument */

typedef struct arc {
    double cij, fij ;
} Arc ;

typedef struct t node {
    int n, pred, label ;
} T node ;

Arc NetOrig[MAX][MAX] ; /* original network with capacity and flow */
Arc Net[MAX][MAX] ; /* temporary net in G-q, q in E */
T node Tree[MAX] ; /* Tree used in tree growing procedure */
int labelled[MAX] ; /* nodes which are labelled in that stage */
int unscanned[MAX] ; /* nodes which are not scanned in that stage */
int X[MAX] ; /* node set which contains source node s in min cut */
double flow ; /* flow value */
int MINCUTS[MAX][MAX][MAX] ; /* incidence matrices for min cut

/***** branch and cut part *****/

typedef struct {
    int lower, upper ;
} Var bound ;

typedef struct node {

```

```

double lpvalue ;
Var bound VB[MAXEDGE] ; /* mst be modified ! */
struct node *next ;
} Node ;

typedef struct {
    Node *front ;
    Node *rear ;
} Queue ;

Node *front, *rear ;
Node *branch[2], *currprob = NULL ;
double incumobj = 200000.0, parentobj ; /* obj. of current integer solution and parents obj */
double currobj ;
int currstate ;

// Functions -----
void terminate();
void set city names (char node option[])
{
    if( !strcmp (node option, "N31") || !strcmp (node option, "N33")) {
        strcpy(city[1], "Seoul"); strcpy(city[2], "Pusan"); strcpy(city[3], "Gwangju");
        strcpy(city[4], "Tokyo"); strcpy(city[5], "Nagoya"); strcpy(city[6], "Sendai");
        strcpy(city[7], "Beijing"); strcpy(city[8], "Shanghai"); strcpy(city[9], "Guangzhou");
        strcpy(city[10], "Jakarta"); strcpy(city[11], "Surabaya");
        strcpy(city[12], "KualaLumpur"); strcpy(city[13], "Taiping");
        strcpy(city[14], "Hanoi"); strcpy(city[15], "Hochimin");
        strcpy(city[16], "Manila"); strcpy(city[17], "Cebu");
        strcpy(city[18], "Bangkok"); strcpy(city[19], "Singapore1");
        strcpy(city[20], "Bandar"); strcpy(city[21], "Taipei");
        strcpy(city[22], "Sonkhla"); strcpy(city[23], "Hongkong");
        strcpy(city[24], "Singapore2");
    }
    if( !strcmp (node option, "N32") || !strcmp (node option, "N34")) {
        strcpy(city[1], "Seoul"); strcpy(city[2], "Pusan");
        strcpy(city[3], "Tokyo"); strcpy(city[4], "Nagoya");
    }
}

```

```

    strcpy(city[5], "Beijing"); strcpy(city[6], "Shanghai");
strcpy(city[7], "Jakarta"); strcpy(city[8], "KualaLumpur");
    strcpy(city[9], "Hanoi"); strcpy(city[10], "Manila");
    strcpy(city[11], "Bangkok"); strcpy(city[12], "Singapore1");
strcpy(city[13], "Bandar");  strcpy(city[14], "Taipei");
    strcpy(city[15], "Hongkong"); strcpy(city[16], "Singapore2");
} //=====

```

=

```

if( !strcmp (node option, "N41") || !strcmp (node option, "N43")) {
    strcpy(city[1], "Seoul"); strcpy(city[2], "Pusan"); strcpy(city[3], "Gwangju");
strcpy(city[4], "Tokyo"); strcpy(city[5], "Nagoya"); strcpy(city[6], "Sendai");
strcpy(city[7], "Beijing"); strcpy(city[8], "Shanghai"); strcpy(city[9], "Guangzhou");
strcpy(city[10], "Jakarta"); strcpy(city[11], "Surabaya");
    strcpy(city[12], "KualaLumpur"); strcpy(city[13], "Taiping");
    strcpy(city[14], "Hanoi"); strcpy(city[15], "Hochimin");
strcpy(city[16], "Manila"); strcpy(city[17], "Cebu");
    strcpy(city[18], "Bangkok"); strcpy(city[19], "Singapore1");
    strcpy(city[20], "Bandar"); strcpy(city[21], "Taipei");
strcpy(city[22], "Sonkhla"); strcpy(city[23], "Hongkong");
    strcpy(city[24], "Korea"); strcpy(city[25], "Japan"); strcpy(city[26], "China");
    strcpy(city[27], "Indonesia"); strcpy(city[28], "Malaysia");
    strcpy(city[29], "Vietnam"); strcpy(city[30], "Philippine");
strcpy(city[31], "Thailand"); strcpy(city[32], "Singapore");
    strcpy(city[33], "Singapore2");
}
if( !strcmp (node option, "N42") || !strcmp (node option, "N44")) {
    strcpy(city[1], "Seoul"); strcpy(city[2], "Pusan");
    strcpy(city[3], "Tokyo");  strcpy(city[4], "Nagoya");
    strcpy(city[5], "Beijing"); strcpy(city[6], "Shanghai");
strcpy(city[7], "Jakarta"); strcpy(city[8], "KualaLumpur");
    strcpy(city[9], "Hanoi"); strcpy(city[10], "Manila");
    strcpy(city[11], "Bangkok"); strcpy(city[12], "Singapore1");
strcpy(city[13], "Bandar");  strcpy(city[14], "Taipei");
    strcpy(city[15], "Hongkong"); strcpy(city[16], "Korea");
    strcpy(city[17], "Japan"); strcpy(city[18], "China");
strcpy(city[19], "Singapore2"); strcpy(city[20], "Singapore");

```

```

}//=====
=====
if( !strcmp (node option, "N5")) {
    strcpy(city[1] , "Seoul"); strcpy(city[2] , "Tokyo");
    strcpy(city[3] , "Beijing"); strcpy(city[4] , "Jakarta");
    strcpy(city[5] , "KualaLumpur"); strcpy(city[6] , "Hanoi");
    strcpy(city[7] , "Manila"); strcpy(city[8] , "Bangkok");
    strcpy(city[9] , "Singapore"); strcpy(city[10] , "Bandar");
    strcpy(city[11] , "Taipei"); strcpy(city[12] , "Hongkong");
}//=====
=====
return;
}
/*****/

void data read(char *file1, char *file2) /* read data file */
{
    FILE *fin ;
    int i, j, k, temp, ct;

    fin = fopen(file1, "r") ;

    if ( fin == NULL ) {
        printf("error : data file does not exist!\n") ;
        exit(1) ;
    }
    fscanf(fin, "%s\n", node option);
    set city names (node option);
    fscanf(fin, "%d %d", &no of nodes, &no of links) ;
    for(i = 0; i <= no of nodes; i++)
        for(j = 0; j <= no of nodes; j++)
            G[i][j] = 0; // initialization

    for (k = 1; k <= no of nodes; k++)
        fscanf(fin, "%d", &req[k]);
}

```

```

ct = 0;
for ( k = 1 ; k <= no of links ; k++ ) {
    fscanf(fin, "%d %d", &i, &j) ;
    fscanf(fin, "%d", &temp) ;
    if(temp == -1) G[i][j] = G[j][i] = -1;
    else {
        Cij[j][i] = Cij[i][j] = temp;
        if ( Cij[i][j] == 0 ) {
            no fixed links++;
            Fixed[k] = 1 ;
        }
        ct++;
        G[i][j] = G[j][i] = ct ; /* link numbering */
    }
}
no of links = ct;
fclose(fin) ;

if( strcmp ("NULL", file2)) {
    fin = fopen(file2, "r") ;
    fscanf(fin, "%d", &ct) ;
    for ( k = 1 ; k <= ct ; k++ ) {
        fscanf(fin, "%d %d", &i, &j) ;
        xx[G[i][j]-1] = 1.0 ;
    }
    fscanf(fin, "%lf", &incumobj) ;
    fclose(fin) ;
}

}

int time check(FILE *out)
{
    clock t stop;
    static double time limit = (3.0) * 3600.0; // (x) hours
    double duration;

```



```

stop = clock();
duration = ((double) (stop - time start)) / CLOCKS_PER_SEC;

if(duration > time limit + TLNC) {
    fprintf(out, " *** %.2lf    Time Limit .. END.. *** \n", time limit);
    return 1;
}
return 0;
}

void set arg(char *file1, char *file2) /* argument setting for LP */
{
    int i, j, k;
    int ct, dt, n;
    char string2 = 'x', string10[5];

    data read(file1, file2) ;

    no of vars = no of links ;
    no of cons = no of nodes ;
    mar = no of cons ;
    mac = no of vars ;

    for ( i = 0 ; i < no of nodes ; i++ ) {
        rhsx[i] = req[i+1] ;
        senx[i] = 'G' ;
    }

    ct = dt = n = 0 ;
    for ( i = 1 ; i < no of nodes ; i++ )
        for ( j = i+1 ; j <= no of nodes ; j++ ){
            if ( G[i][j] <= 0 ) continue ;
            // name x(i, j);
            cstore[ct] = string2;  cname[dt++] = &cstore[ct++];  cstore[ct++] = '(';
            itoa(i, string10, 10);  for(k = 0; string10[k] != '\0'; k++) cstore[ct++] = string10[k];

```

```

    cstore[ct++] = ',';
    itoa(j, string10, 10); for(k = 0; string10[k] != '\0'; k++) cstore[ct++] = string10[k];
    cstore[ct++] = ')';
    cstore[ct++] = '\0';

    objx[n] = (double)Cij[i][j] ;
    bdl[n] = (Cij[i][j] == 0) ? 1.0 : 0.0 ; /* lower bd : fixed edge = 1, ordinary edge = 0 */
    bdu[n++] = 1.0 ;
}

ct = 0 ; i = 0 ;
for ( j = 1 ; j <= no of nodes ; j++ )
    for ( k = j+1 ; k <= no of nodes ; k++ ) {
        if ( G[j][k] <= 0 ) continue ;

        if ( i == 0 )
            matbeg[i] = 0 ;
        else
            matbeg[i] = matbeg[i-1] + matcnt[i-1] ;
        matcnt[i] = 2 ;
        matval[ct] = 1 ;
        matind[ct++] = j - 1 ;
        matval[ct] = 1 ;
        matind[ct++] = k - 1 ;
        i++ ;
    }
return;
}

CPXENVptr    env = NULL;
CPXLPptr    lp = NULL; /* pointer that indicate LP of master problem */

int init lp(char *file1, char *file2)
{
int    status;

```

```

/* Part I - Loading, optimizing and obtaining a solution to
the original problem. */

set arg(file1, file2) ;

lp = CPXcreateprob(env, &status, probname);
if ( lp == NULL ) {
    printf ("Failed to load LP. .. in init lp() \n");
    goto TERMINATE;
}

status = CPXcopylpwnames(env, lp, mac, mar, objsen, objx, rhsx, senx, matbeg,
    matcnt, matind, matval, bdl, bdu, NULL, cname, NULL);
if(status) {
    printf("some error happened at CPXcopylp\n");
    terminate(); exit(0); exit(0);
}
#ifdef CHECK
status = CPXcheckcopylp (env, lp, mac, mar, objsen, objx, rhsx,
    senx, matbeg, matcnt, matind, matval,
    bdl, bdu, NULL) ;
if(status) {
    printf("some error happened in init master formulation\n");
    terminate(); exit(0);
}
#endif

#ifdef LP FORM
status = CPXwriteprob(env, lp, "edgecon.lp", NULL) ;
if ( status ) {    fprintf (stderr, "Failed to write LP to disk. in init lp master() \n");    goto
TERMINATE; }
#endif

/* Optimize the problem and obtain solution. */

```

```

status = CPXprimopt(env, lp);
if ( status ) goto TERMINATE;

status = CPXsolution (env, lp, &lpstat, &obj, x, pi, slack, dj);
if ( status ) goto TERMINATE;
#ifdef DEBUG
printf ("\nSolution status = %d\n", lpstat);
printf ("Solution value = %f\n\n", obj);
#endif

TERMINATE:

return (0);

} /* END init lp */
/*****
/***** Maximal Flow Routine *****/

void InitNet(int node)
{
    int i, j;

    for ( i = 0 ; i <= node ; i++ )
        for ( j = 0 ; j <= node ; j++ ) {
            Net[i][j].cij = 0 ;
            Net[i][j].fij = 0 ;
        }
}

void IniTemp(int node)
{
    int i ;

    for ( i = 0 ; i <= node ; i++ ) {
        Tree[i].n = 0 ;
        Tree[i].pred = 0 ;
    }
}

```

```

    Tree[i].label = 0 ;

    labelled[i] = 0 ;

    unscanned[i] = 0 ;

    X[i] = 0 ;
    }
}

void MakeNet(int node) /* make the original graph */
{
    int i, j, ct ;

    ct = 0 ;
    for ( i = 1 ; i <= node ; i++ )
        for ( j = i+1 ; j <= node ; j++ ){
            if ( G[i][j] <= 0 ) continue ;
            Net[i][j].cij = x[ct++] ;
            Net[j][i].cij = Net[i][j].cij ;
        }
}

void PlanTree(int so)
{
    labelled[so] = 1 ;
    unscanned[so] = 1 ;

    Tree[so].pred = 0 ;
    Tree[so].label = 1 ; /* source:1, forward:2, backward:3 */
}

int SelectNode(int node)
{
    int i ;

```

```

for ( i = 1 ; i <= node ; i++ )
    if ( unscanned[i] == 1 ) {
        unscanned[i] = 0 ;
        return ( i ) ;
    }

return ( 0 ) ; /* termination sign */
}

int GrowTree(int node, int so, int te)
{
    int scan ;
    int i ;

    IniTemp(node) ;
    PlanTree(so) ;

while(1) {
    scan = SelectNode(node) ;
    if ( scan == 0 )
        return ( 0 ) ; /* there is not a augmenting path */

/* labelling */
for ( i = 1 ; i <= node ; i++ ) {
    if ( labelled[i] == 0 ) {
        if ( Net[scan][i].cij > Net[scan][i].fij ) { /* when sparse, modify! */
            labelled[i] = 1 ;
            unscanned[i] = 1 ;

            Tree[i].pred = scan ;
            Tree[i].label = 2 ;
        }
        else if ( Net[i][scan].fij > 0 ) { /* when sparse, modify! */
            labelled[i] = 1 ;
            unscanned[i] = 1 ;

```

```

        Tree[i].pred = scan ;
        Tree[i].label = 3 ;
    }
}
if ( labelled[te] == 1 )
    return (1); /* there is a flow augmenting path */
}
}

```

```

void Augment(int so, int te)

```

```

{
    int i ;
    double min = 32767., temp ;
    int from, to ;

    /* find minimum residual capacity */
    for ( i = te ; i != so ; i = Tree[i].pred ) {
        from = Tree[i].pred ;
        to = i ;
        if ( Tree[to].label == 2 )
            temp = Net[from][to].cij - Net[from][to].fij ;
        else if ( Tree[to].label == 3 )
            temp = Net[to][from].fij ;

        if ( temp <= min )
            min = temp ;
    }

    /* augment minimum residual capacity */
    for ( i = te ; i != so ; i = Tree[i].pred ) {
        from = Tree[i].pred ;
        to = i ;
        if ( Tree[to].label == 2 )
            Net[from][to].fij += min ;
        else if ( Tree[to].label == 3 )

```

```

        Net[to][from].fij -= min ;

    }
}

void MakeX(int node)
{
    int i ;

    for ( i = 1 ; i <= node ; i++ )
        if ( Tree[i].label > 0 )
            X[i] = 1 ;
}

void CalcFlow(int node, int so)
{
    int i ;

    flow = 0.0 ;
    for ( i = 1 ; i <= node ; i++ ) {
        flow += Net[so][i].fij ;
        flow -= Net[i][so].fij ;
    }
}

int CUT[MAX][MAX] ;

void MakeCut()
{
    int i, j ;

    for ( i = 0 ; i <= no of nodes ; i++ )
        for ( j = 0 ; j <= no of nodes ; j++ )
            CUT[i][j] = 0 ;
}

```



```

for ( i = 1 ; i <= no of nodes ; i++)
    for ( j = 1 ; j <= no of nodes ; j++) {
        if ( X[i] == 1 && X[j] == 0 ) {
            CUT[i][j] = 1 ;
            CUT[j][i] = 1 ;
        }
        if ( G[i][j] <= 0 ) {
            CUT[i][j] = 0 ;
            CUT[j][i] = 0 ;
        }
    }
}

void StoreCut(int num)
{
    int i, j ;

    for ( i = 0 ; i <= no of nodes ; i++)
        for ( j = 0 ; j <= no of nodes ; j++)
            MINCUTS[num][i][j] = 0 ;

    for ( i = 0 ; i <= no of nodes ; i++)
        for ( j = 0 ; j <= no of nodes ; j++)
            MINCUTS[num][i][j] = CUT[i][j] ;
}

void MaxFlow(int node, int so, int te) /* only if MakeNetOrig() run before */
{
    int flag ;

    InitNet(node) ;
    MakeNet(node) ; /* make the original network using current solution x[i]*/

    while(1) {
        flag = GrowTree(node, so, te) ;
    }
}

```

```

if ( flag == 1 ) {
    Augment(so, te) ;
}
else if ( flag == 0 ) {
    MakeX(node) ;
    MakeCut() ; /* make corresponding cut */
    CalcFlow(node, so) ;
    return ;
}
}
}

```

/****** cut tree routine *****/

```
int p[MAX] ;
```

```
double fl[MAX] ;
```

```
void CutTree(int node)
```

```
{
```

```
    int s, t, i ;
```

```
    for ( i = 1 ; i <= node ; i++ )
```

```
        p[i] = 1 ;
```

```
    for ( s = 2 ; s <= node ; s++ ) {
```

```
        flow = 0.0 ;
```

```
        t = p[s] ;
```

```
        MaxFlow(node, s, t) ;
```

```
        fl[s] = flow ;
```

```
        for ( i = 1 ; i <= node ; i++ )
```

```
            if ( ( i != s ) && ( X[i]==1 ) && ( p[i]==t ) )
```

```
                p[i] = s ;
```

```
        if ( X[p[t]] == 1 ) {
```

```
            p[s] = p[t] ;
```

```

        p[t] = s ;
        fl[s] = fl[t] ;
        fl[t] = flow ;
    }
}

/* make incidence matrix of a graph and find path using Breadth first search */
int gra[MAX][MAX], mami[MAX] ;
double cutval[MAX][MAX] ;

void make_graph(int node)
{
    int i, j, k = 1 ;

    for ( i = 0 ; i <= no of nodes ; i++ )
        for ( j = 0 ; j <= no of nodes ; j++ ){
            gra[i][j] = 0 ;
            cutval[i][j] = 0.0 ;
        }

    for ( i = 2 ; i <= node ; i++ ) {
        gra[i][p[i]] = 1 ;
        gra[p[i]][i] = 1 ;
        cutval[i][p[i]] = fl[i] ;
        cutval[p[i]][i] = fl[i] ;
        /*printf("%d - %d : %lf\n", i, p[i], fl[i]) ;*/
    }

    for ( i = 1 ; i <= node ; i++ )
        for ( j = i+1 ; j <= node ; j++ ) {
            if ( gra[i][j] == 1 ) {
                gra[i][j] = k ;
                gra[j][i] = k ;
                MaxFlow(node, i, j) ;
                StoreCut(k) ;
                k++ ;
            }
        }
}

```

```

    }
}

void bfs(int root, int node)
{
    int i, s ;
    int visit[MAX] ;

    for ( i = 1 ; i <= node ; i++ ) {
        mami[i] = 0 ;
        visit[i] = 0 ;
    }
    visit[root] = 1 ;
    s = root ;
UP: ;
    for ( i = 1 ; i <= node ; i++ )
        if ( visit[i] == 0 && gra[s][i] > 0 ) {
            mami[i] = s ;
            visit[i] = 1 ;
        }
    visit[s] = -1 ;
    for ( i = 1 ; i <= node ; i++ )
        if ( visit[i] == 1 ) {
            s = i ;
            goto UP ;
        }
}

int check viol(int s, int t)
{
    int minreq ;
    double min cut val, min = 32767 ;
    int left, right, i, ct ;
    int temp[MAX] ;

    bfs(s, no of nodes) ;

```

```

for ( i = 0 ; i < no of nodes ; i++ )
    temp[i] = 0 ;
ct = 0 ;
for ( i = t ; i != 0 ; i = mami[i] )
    temp[ct++] = i ;

for ( i = 0 ; i < ct ; i++ ) {
    if ( i == (ct-1) ) continue ;
    if ( cutval[temp[i]][temp[i+1]] <= min ) {
        min = cutval[temp[i]][temp[i+1]] ;
        min cut val = min ;
        left = temp[i] ; right = temp[i+1] ;
    }
}
minreq = (req[s] > req[t]) ? req[t] : req[s] ;

if ( min cut val > (double) minreq - TLNC )
    return 0 ;
return (gra[left][right]) ;
}

// to check if the same rows is given already
double c rmatval[MAX*(MAX-1)/2][MAXEDGE];
int c nzcnt[MAX*(MAX-1)/2];
double c rhs[MAX*(MAX-1)/2];
int c rmatind[MAX*(MAX-1)/2][MAXEDGE];
int row num here;

void add to c r(double rmatval[], int rmatind[], int nzcnt, double rhs, int *row num here)
{
    int i;

    c nzcnt[*row num here] = nzcnt;
    c rhs[*row num here] = rhs;
    for(i = 0; i < nzcnt; i++) {
        c rmatind[*row num here][i] = rmatind[i];
    }
}

```

```

        c rmatval[*row num here][i] = rmatval[i];
    }
    (*row num here)++;
    return;
}

```

short int check the same(double rmatval[], int rmatind[], int nzcnt, double rhs, int row num here)

```

{
    int i, j;

    for(i = 0; i < row num here; i++) {
        if(nzcnt != c nzcnt[i]) continue;
        for(j = 0; j < nzcnt; j++) {
            if(rmatind[j] != c rmatind[i][j]) break;
            if(rmatval[j] > c rmatval[i][j] + TLNC || rmatval[j] < c rmatval[i][j]) break;
        }
        if(j == nzcnt) {
            if(rhs < c rhs[i] + TLNC && rhs > c rhs[i] - TLNC) return YES;
        }
    }
    return NO;
}

```

int add main con()

```

{
    int i, j, k, l, ct ;
    int flag = 0, status ;
    int cut num ;

    InitNet(no of nodes) ;
    MakeNet(no of nodes) ; /* make the original network using current solution x[i]*/
    CutTree(no of nodes) ; /* make cut tree */
    make_graph(no of nodes) ; /* make the incidence matrix of the cut tree */

    row num here = 0;
}

```

```

for ( i = 1 ; i <= no of nodes ; i++ )
    for ( j = i+1 ; j <= no of nodes ; j++ ) {

        if ( req[i] == 0 || req[j] == 0 ) continue ;
        if (G[i][j] == -1) continue;

        if ( (cut num = check viol(i,j)) > 0 ) {
            flag++ ;
            rhs = (req[i] > req[j]) ? (double) req[j] : (double) req[i] ;
            sense = 'G' ;
            nzcnt = 0 ;
            ct = 0 ;
            for ( k = 1 ; k < no of nodes ; k++ )
                for ( l = k+1 ; l <= no of nodes ; l++ ) {
                    if ( G[k][l] <= 0 ) continue ;
                    if ( MINCUTS[cut num][k][l] == 1 ) {
                        rmatval[ct] = 1 ;
                        rmatind[ct++] = G[k][l] - 1 ;
                        nzcnt++ ;
                    }
                }
            if(check the same(rmatval, rmatind, nzcnt, rhs, row num here) == YES)
continue; // The same row has implication.
                else add to c r(rmatval, rmatind, nzcnt, rhs, &row num here);

                status = CPXaddrows(env, lp, ccnt, rcnt, nzcnt, &rhs, &sense, &rmatbeg,
rmatind, rmatval, NULL, NULL) ;
                if(status){ printf("addrows failed:main cut\n"); terminate(); exit(1) ;}
#ifdef CHECK
                status = CPXcheckaddrows(env,
lp,ccnt,rcnt,nzcnt,&rhs,&sense,&rmatbeg,rmatind, rmatval,NULL,NULL);
                // printf("ccnt = %d, rcnt = %d, nzcnt = %d, sense = %c, rmatbeg = %d\n",
ccnt, rcnt, nzcnt, sense, rmatbeg);

```

```

        if(status){ printf("Error in addrows... \n"); terminate(); exit(0); }
#endif

        no of cuts++;
    } /* outer if */
} /* inner for loop */
if ( (flag == 0) ) return 1; /* no cuts are added*/
return 0 ;
}

int solve with cut()
{
    int status ;

    if ( (add main con() == 1) ) return 0 ;

#ifdef LP FORM
    status = CPXwriteprob(env, lp, "edgecon.lp", NULL) ; if ( status ) { fprintf (stderr, "Failed to
write LP to disk.\n"); exit(0); }
#endif

    /* Optimize the problem and obtain solution. */

    status = CPXdualopt (env, lp);
    if ( status ) goto TERMINATE;

    status = CPXsolution (env, lp, &lpstat, &obj, x, pi, slack, dj);
    if ( status ) goto TERMINATE;

    currstate = lpstat ; /* check cuurent proble state for fathoming */
    return 1;
TERMINATE:
    terminate();
    exit(1);

```



```

} /* END solve with cut */

int solve once()
{
    int status;

#ifdef LP FORM
    status = CPXwriteprob(env, lp, "edgecon.lp", NULL);
    if ( status ) { fprintf (stderr, "Failed to write LP to disk. in init lp master() \n"); goto
TERMINATE; }
#endif

    /* Optimize the problem and obtain solution. */

    status = CPXdualopt (env, lp);
    if ( status ) goto TERMINATE;

    status = CPXsolution (env, lp, &lpstat, &obj, x, pi, slack, dj);
    if ( status ) goto TERMINATE;

    currstate = lpstat ; /* check cuurent proble state for fathoming */
    if(currstate == 2) return 0;

    return 1;

TERMINATE:
    terminate();
    exit(0);

} /* END solve once */

/*****

```

```

/***** fuctions for banch and cut *****/
/*****/

int chk intg()
{
    int i, k = 0 ;

    for ( i = 0 ; i < no of links ; i++ )
        if ( x[i] > TLNC && x[i] < (1 - TLNC) )
            continue ;
        else
            k++ ;

    if ( k == no of links ) return 1 ; /* integer */
    return 0 ;
}

int IsEmpty()
{
    if ( front == NULL )
        return 1 ; /* empty */
    return 0 ;
}

void GetProb()
{
    if (front == NULL )
        printf("Empty\n") ;
    else {
        currprob = front ;
        front = currprob->next ;
        if ( front == NULL )
            rear = NULL ;
    }
}

void InsProb best(Node *np) /* Insert a node before a node */
{

```

```

Node *temp;

if ( np == NULL )
    printf("Node nonexistent\n");
else if ( front == NULL ) {
    front = np ;
    rear = np ;
} else {
    if ( (front->lpvalue) > (np->lpvalue) + TLNC) {
        np->next = front ;
        front = np ;
    }
    else {
        for (temp = front ; temp != rear ; temp = temp->next) {
            if ( ((temp->next)->lpvalue) > (np->lpvalue) + TLNC) {
                np->next = temp->next ;
                temp->next = np ;
                return ;
            }
        }
        rear->next = np ;
        rear = np ;
    }
}
return;
}

```

```

void InsProb dfs(Node *np) /* Maintain ordered list : depth first */
{

```

```

    if ( np == NULL )
        fprintf(stderr, "insertion failed\n");
    else if ( front == NULL ) {
        front = np ;
        rear = np ;

```

```

    }
    else {
        np->next = front ;
        front = np ;
    }
    return;
}

```

Node *MakeNode(FILE *out)

```

{
    Node *p ;
    int i, j, ct ;

    if ( (p = (Node *)malloc(sizeof(Node))) == NULL ) {
        fprintf(out, "Memory exhausted\n") ;
        printf("Memory exhausted\n") ;
        return NULL;
    }
    else {
        p->next = NULL ;
        p->lpvalue = parentobj ;
        if ( currprob == NULL ) {
            ct = 0 ;
            for ( i = 1 ; i <= no of nodes ; i++ )
                for ( j = i+1 ; j <= no of nodes ; j++ ) {
                    if (G[i][j]<=0) continue ;
                    p->VB[ct].lower = (Cij[i][j]==0) ? 1 : 0 ;
                    p->VB[ct++].upper = 1 ;
                }
        }
        else {
            for ( i = 0 ; i < no of links ; i++ ) {
                p->VB[i].lower = currprob->VB[i].lower ;
                p->VB[i].upper = currprob->VB[i].upper ;
            }
        }
    }
}

```

```

        return (p) ;
    }
}
/*****
*****/
typedef struct zee zee ;
struct zee {
    int z no ;
    double key ;
} ;

zee zeas[100] ;

void init zeas()
{
    int i ;

    for ( i = 0 ; i < no of links ;i++) {
        zeas[i].z no = 0;
        zeas[i].key = 0.0 ;
    }
}

/* swap: interchange v[i] and v[j] */
void swap(zee zeta[], int i, int j)
{
    int zn;
    double ky;

    zn = zeta[i].z no;
    ky = zeta[i].key;

    zeta[i].z no = zeta[j].z no;
    zeta[i].key = zeta[j].key;

    zeta[j].z no = zn;

```

```

    zeta[j].key = ky;
}

/* qsort sort v[left] ... v[right] into decreasing order */
void dec sort(zeta zeta[], int left, int right)
{
    int i, last ;
    void swap();

    if (left >= right) /* do nothing if array contains */
        return; /* fewer than two elements */
    swap(zeta, left, (left+right)/2); /* move partition elements */
    last = left ; /* to zeta[0] */
    for ( i = left + 1 ; i <= right ; i++ ) /* partition */
        if (zeta[i].key > zeta[left].key)
            swap(zeta, ++last, i);
    swap(zeta, left, last) ; /* restore partition elem */
    dec sort(zeta, left, last-1) ;
    dec sort(zeta, last+1, right) ;
}

/*****
***/

int VarSelect old()
{
    int i, count = 0 ;

    init zees() ;
    for ( i = 0 ; i < mac ; i++ )
        if ( x[i] > 0.0001 && x[i] < 0.9999 ) {
            zees[count].z no = i ;
            zees[count].key = x[i] ;
            count++ ;
        }
    if ( count == 0 ) return -1 ;
}

```

```

printf("count = %d\n", count) ;
dec sort(zees, 0, count-1) ;
printf("x[%d] = % 12.10f \n", zees[0].z no, x[zees[0].z no]) ;
return (zees[0].z no) ;

}

int VarSelect()
{
    int i ;

    for ( i = 0 ; i < no of links ; i++ ) {
        if ( x[i] > TLNC && x[i] < (1 - TLNC) ) {
            /*printf("x[%d] = % 1f\n", i, x[i]) ;*/
            return (i) ;
        }
    }
    return -1;
}

int Branch(FILE *out)
{
    Node *a, *b ;
    static int v ;

    a = MakeNode(out) ; /* branch change upper */
    if(a == NULL) return 0;
    b = MakeNode(out) ; /* branch change lower */
    if(b == NULL) return 0;

    a->lpvalue = parentobj ;
    b->lpvalue = parentobj ;

    v = VarSelect() ;

    /*printf("v = %d : % 1f\n", v, x[v]) ;*/
}

```

```

    if ( v == -1 ) {
        fprintf(out, "Cannot branch\n"); return 0;
    }
    a->VB[v].upper = 0 ;
    b->VB[v].lower = 1 ;
// Select one option out of the following three alternatives
    InsProb best(a); InsProb best(b) ; // Best bound rule
//  InsProb best(a); InsProb dfs(b) ; // Depth-first and best when backtracking
//  InsProb dfs(a);    InsProb dfs(b) ; // Depth-first rule
    return 1;
}

void FreeSolvedProb()
{
    free(currprob) ;
}

int CanPrune()
{
    int i ;

    double temp = 0.0 ;

    temp = (currobj - (double)(int)currobj < TLNC) ? currobj : (int)currobj + 1 ;

    if ( currstate == 2 )
        return (1) ;
    if ( temp > incumobj - 1.0 + TLNC)
        return (1) ;
    if ( chk intg() ) {
        if ( currobj < incumobj - TLNC) {
            incumobj = currobj ;
            for ( i = 0 ; i < no of links ; i++)
                xx[i] = x[i] ;
        }
    }
}

```



```

        return (1) ;
    }
    return (0) ;
}
int getsu, zari[MACSZ] ;
char luu[MACSZ], lul[MACSZ] ;
double bduu[MACSZ], bdll[MACSZ] ;

void ConstructLP()
{
    int i, st ;

    getsu = no of links ;
    for ( i = 0 ; i < no of links ; i++ ) {
        luu[i] = 'U' ; lul[i] = 'L' ;
        zari[i] = i ;
        bduu[i] = currprob->VB[i].upper ;
        bdll[i] = currprob->VB[i].lower ;
    }

    st = CPXchgbds(env, lp, getsu, zari, luu, bduu ) ;
    if ( st == 1 ) {
        printf("Error in chgbds() : upper\n") ;
        exit(1) ;
    }
    st = CPXchgbds(env, lp, getsu, zari, lul, bdll ) ;
    if ( st == 1 ) {
        printf("Error in chgbds() : lower\n") ;
        exit(1) ;
    }
}

void RestoreLP()
{
    int i, j, ct, st ;

```

```

ct = 0 ;
for ( i = 1 ; i <= no of nodes ; i++ )
    for ( j = i+1 ; j <= no of nodes ; j++ ) {
        if (G[i][j]<=0) continue ;
        bdll[ct] = 0.0 ;
        bduu[ct++] = 1.0 ;
    }
st = CPXchgbds(env, lp, getsu, zari, luu, bduu ) ;
if ( st == 1 ) {
    printf("Error in chgbds()\n") ;
    exit(1) ;
}
st = CPXchgbds(env, lp, getsu, zari, lul, bdll ) ;
if ( st == 1 ) {
    printf("Error in chgbds()\n") ;
    exit(1) ;
}
}

void stop here()
{
    while(front) { // free all nodes
        GetProb();
        FreeSolvedProb();
    }
}

void BranchandCut(FILE *out)
{
    int quit, c = 1, k;
    double temp ;
    int no of nodes = 2;

    while ( front != NULL ) {
        if(time check(out) == 1) {stop here(); goto SKIP2;}
    }
}

```

```

temp = 0.0 ;
GetProb();

temp = currprob->lpvalue ;
temp = (temp - (double)(int)temp < TLNC) ? temp : (int)temp + 1 ;
if ( temp > incumobj - TLNC) {
    FreeSolvedProb() ;
    continue ;
}
#ifdef DEBUG
printf("%6d  ", c ) ;
printf("%10.4lf  ", currprob->lpvalue) ;
#endif
c++;

if(c > NDLIMIT) {
    fprintf(out, "Node Limit!!! %d nodes are allowed\n", NDLIMIT);
    stop here(); break;
}

ConstructLP() ;

k = solve once() ; no lp++ ;
if(k == 0) goto SKIP;
quit = 1 ;
while(quit) {
    quit = solve with cut() ;
    no lp++ ;
    if ( currstate == 2 ) {
#ifdef DEBUG
printf("%10.4lf %10s  ", incumobj, "infeasible") ;
#endif
        goto SKIP;
    }
}
}

```

```

    currobj = obj ;
    parentobj = currobj ;
    if ( (k=CanPrune()) == 0) {
        if( Branch(out) == 0) {stop here(); goto SKIP2;} ;
        no of nodes += 2;
#ifdef DEBUG
        printf("%10.4f %10.4f      ", incumobj, currobj) ;
#endif
    }
#ifdef DEBUG
    else if ( k == 1 )
        printf("%10.4f %10.4f cut off ", incumobj, currobj) ;

#endif

SKIP: RestoreLP() ;

    FreeSolvedProb() ; no of nodes--;

#ifdef DEBUG
    printf("%6d %6d\n", no of cuts, no of nodes) ;
#endif

}

SKIP2: no bnc nodes = c-1 ;
    return;
}

void prnt sol(FILE *fout, char inf[], double duration)
{
    int i, j, count = 0 ;

    fprintf(fout, "Problem: %s\n", inf) ;

```

```

for ( i = 1 ; i <= no of nodes ; i++ )
    for ( j = i+1 ; j <= no of nodes ; j++ )
        if ( G[i][j] > 0 ) {
            if ( xx[G[i][j]-1] > 1.0 - TLNC ) {
                fprintf(fout, "%s(%d) - %s(%d) : %d\n", city[i], i, city[j], j, Cij[i][j]) ;
            }
        }
    fprintf(fout, "\n\n") ;

    fprintf(fout, "%d\t%lf\t%lf\t%.2lf\n", no bnc nodes, incumobj, (-upper sol + incumobj)/incumobj,
duration ) ;

    fprintf(fout, "-----\n");

    return;
}

void main(int argc, char *argv[])
{
    int quit = 1, i;
    clock t stop;
    double duration;
    FILE *fout ;

    void open CPLEX();

    if (argc < 4 ) {
        printf("main argument missing\n") ;
        exit(1) ;
    }

    open CPLEX();

    fout = fopen(argv[3], "a") ;

```

```

if(fout == NULL) { printf("no memory for ouput file \n"); exit(0);}

time start = clock();

init lp(argv[1], argv[2]) ; no lp0++ ;
#ifdef DEBUG
printf("===== ROOT NODE =====\n") ;
printf("TOTAL ADDED CUTS : %5d, OBJECTIVE VALUE : %10.4lf\n", no of cuts, obj) ;
#endif
while (1) {
    quit = solve with cut() ;
    no lp0++ ;
#ifdef DEBUG
printf("TOTAL ADDED CUTS : %5d, OBJECTIVE VALUE : %10.4lf\n", no of cuts, obj) ;
#endif
    if ( quit==0 ) break ;

}

upper sol = obj ;
usol = obj ;
parentobj = obj ;
currobj = obj ;

if (i = CanPrune()) goto END ;
#ifdef DEBUG
printf("\n===== BRANCH AND CUT =====\n\n") ;
printf(" NODE Best bound Best Integer Curr obj Status Tot cuts Num LP\n") ;
#endif
Branch(fout) ;

BranchandCut(fout) ;

END :
stop = clock();

```

```

duration = ((double) (stop - time start)) / CLOCKS_PER_SEC;

prnt sol(fout, argv[1], duration);
fclose(fout);
terminate();
return;
}

void open CPLEX()
{
    int status;
    env = CPXopenCPLEX (&status);
    if ( env == NULL ) {
        char  errmsg[1024];
        fprintf (stderr, "Could not open CPLEX environment.\n");
        CPXgeterrorstring (env, status, errmsg);
        fprintf (stderr, "%s", errmsg);
        exit(0);
    }
    return;
}

void terminate()
{
    /* If you want to finish the programme */
    /* Free up the problem as allocated by CPXloadlp, if necessary */

    int status;

    if ( lp != NULL ) {
        status = CPXfreeprob (env, &lp);
        if ( status ) {
            fprintf (stderr, "CPXfreeprob failed, error code %d.\n", status);
        }
    }
}

```

```

/* Free up the CPLEX environment, if necessary */

if ( env != NULL ) {
    status = CPXcloseCPLEX (&env);

    /* Note that CPXcloseCPLEX produces no output,
       so the only way to see the cause of the error is to use
       CPXgeterrorstring. For other CPLEX routines, the errors will
       be seen if the CPX PARAM SCRIND indicator is set to CPX ON. */

    if ( status ) {
        char  errmsg[1024];
        fprintf (stderr, "Could not close CPLEX environment.\n");
        CPXgeterrorstring (env, status, errmsg);
        fprintf (stderr, "%s", errmsg);
    }
}
return;
}

```


B. Data Input Procedures

The names of the nodes should be written first and the number of nodes and edges next. Spaces thereafter indicate node types from the first to the last nodes. Finally, the cost of the nodes between i and j are listed accordingly.

The estimated construction cost was calculated from the integer number of distance measured using the 1: 8,500,000 scale map and multiplied by 10.

[Table] City Numbers according to Scenarios

Member	City	N31	N32	N33	N34	N41	N42	N43	N44	N5
Korea	Seoul	1	1	1	1	1	1	1	1	1
	Pusan	2	2	2	2	2	2	2	2	
	Gwangju	3		3		3		3		
Japan	Tokyo	4	3	4	3	4	3	4	3	2
	Nagoya	5	4	5	4	5	4	5	4	
	Sendai	6		6		6		6		
China	Beijing	7	5	7	5	7	5	7	5	3
	Shanghai	8	6	8	6	8	6	8	6	
	Guangzhou	9		9		9		9		
Indonesia	Jakarta	10	7	10	7	10	7	10	7	4
	Surabaya	11		11		11		11		
Malaysia	Kuala Lumpur	12	8	12	8	12	8	12	8	5
	Taiping	13	0	13	0	13		13		
Viet Nam	Hanoi	14	9	14	9	14	9	14	9	6
	Ho Chi Min	15		15		15		15		
Philippines	Manila	16	10	16	10	16	10	16	10	7
	Cebu	17		17		17		17		
Thailand	Bangkok	18	11	18	11	18	11	18	11	8
	Sonkhola	22		22		22		22		
Singapore	Sinpoapore1	19	12	19	12	19	12	19	12	9
	Singapore2	24	16			33	19			
Brunei Darussalam	Bandar Seri Begawan	20	13	20	13	20	13	20	13	10
Chinese Taipei	Chinese Taipei	21	14	21	14	21	14	21	14	11
Hong Kong, China	Hong Kong, China	23	15	23	15	23	15	23	15	12
Total Nodes		24	16	23	15	33	20	31	18	12