國立臺灣大學管理學院資訊管理學系

碩士論文

Department of Information Management

College of Management

National Taiwan University

Master Thesis

考量自然災害與智慧型攻擊下
確保服務持續性之冗餘及防禦資源配置演算法

Redundancy and Defense Resource Allocation Algorithms to Assure Service Continuity against Natural Disasters and Intelligent Attackers

駱睿斌

Ray, Jui-Pin Lo

指導教授：林永松 博士

Advisor: Yeong-Sung Lin, Ph.D.

中華民國九十八年八月

August, 2009

考量自然災害與智慧型攻擊下
確保服務持續性之冗餘及防禦資源配置演算法
Redundancy and Defense Resource Allocation Algorithms to Assure Service Continuity against Natural Disasters and Intelligent Attackers

本論文係提交國立台灣大學
資訊管理學研究所作為完成碩士學位
所需條件之一部分

研究生：駱睿斌　撰

中華民國九十八年八月

國立臺灣大學碩士學位論文
口試委員會審定書

考量自然災害與智慧型攻擊下
確保服務持續性之冗餘及防禦資源配置演算法

本論文係駱睿斌君（學號 R96725009）在國立臺灣大學資訊管理學系、所完成之碩士學位論文，於民國 98 年 7 月 25 日承下列考試委員審查通過及口試及格，特此證明

口試委員：林永松

林盈達

傅新彬　鍾嘉德

所　　長：蔡益坤 代

謝誌

兩年的研究生涯接近尾聲，“深刻”是對這段歷程最為貼切的形容。唯有親身經歷的冒險，事後才會感到自豪，並且永生難忘；值此微笑收穫之時，我最希望將甜美果實獻給最敬愛的父母，駱光裕先生與賴麗香女士。正是您們無怨無悔的悉心照料與寬大包容，一路溫柔推送著我向前挺進，直至終點。謝謝您們為我付出的一切，更感激您們始終相信我，並且義無反顧地支持我，我愛您們！謝謝我的哥哥子仁，總是設法使我安心並且願意與我討論問題；也感謝所有關心我的長輩與親戚，您們的關懷我都銘記在心，礙於篇幅限制，請原諒我無法一一道謝。

衷心感謝指導教授林永松老師在兩年中從未間斷的諄諄教誨與關懷。除了學術研究方面的啟發與指導之外，您更使我們親身體會待人處事的重要，也讓我們謹記應對生命中的一切抱持正確的態度。此外，也感謝輔大資工系呂俊賢教授、國立高雄第一科技大學行銷與流通管理系傅新彬教授、台大電機系鐘嘉德教授，以及交大資工系林盈達教授，因為您們的寶貴意見，才使這篇論文更臻完備。

感謝柏皓學長，百忙之中仍然願意提供協助，不僅在論文方面不吝指導，也時常給予關心與鼓勵。另外，謝謝健誠學長撥冗協助檢查論文，謝謝博士班學長姐平日幫忙處理實驗室庶務，也謝謝世昌、永斌、耀元、怡緯、子雋為我們辦妥口試當天相關事宜，使我們在撰寫論文與準備口試期間無後顧之憂。

感謝同在一條船上的冠瑋、宴毅、友仁、竣韋、培維、猷順始終同甘共苦、義氣相挺。儘管我們屢逢驚濤駭浪，結果卻總能絕處逢生、化險為夷，我想，憑藉的就是我們 OP 七俠在兩年間所建立的革命情感。數不清的歡笑、道不盡的辛酸，真的只有生死與共的你們才懂！我們熬過來了，而且一個都沒有少！

最終的感謝留給我的愛，江姿儀。雖然寫論文的這一年我倆分隔兩地，但我知道妳的心始終不曾離開，謝謝妳陪我走過這一切，接下來請讓我陪妳走，好嗎？

這篇論文的付梓，意味著學生生涯暫時告一段落；儘管即將褪去學生的身分，學習並不因而劃下句點，因為在人生的道路上，我們永遠需要不停地摸索與學習。最後，期勉自己對世間萬象長保好奇探索之慾，並永懷謙虛受教之心。

駱睿斌　謹識

于國立台灣大學資訊管理研究所

中華民國九十八年八月

論文摘要

題目：考量自然災害與智慧型攻擊下
確保服務持續性之冗餘及防禦資源配置演算法
作者：駱睿斌　　　　　　　　　　　　　九十八年八月

指導教授：林永松 博士

近年來企業持續營運管理逐漸受到企業組織的重視。於此範疇中，災害復原計畫是與資訊科技最息息相關的部分。而在實踐災害復原計畫的眾多辦法之中，冗餘的佈署是一項常被採用的有效解決方案；然而，過往探討冗餘配置問題的研究多聚焦於可靠性系統，較少著力於企業組織多所仰賴的網路系統。故本論文在考慮網路環境特性的條件之下，旨於將冗餘與額外防禦有效地搭配運用於網路系統，一方面達成其服務之持續性，另方面則提升其抵抗具備經驗累積能力之惡意攻擊的存活度。

於此論文中，我們建構了一個攻防雙方彼此角力的攻防情境，之後將其轉化為一個兩階段的非線性整數規劃問題：在內層問題（AEA 模型）中，具備絕對經驗累積能力的攻擊者透過選擇適當的目標進行攻擊，企圖以最小化的成本攻克網路中的所有核心節點；反觀外層問題（RAP-EDM 模型），防禦者則在有限的防禦資源預算之下，透過適當地佈署冗餘與額外防禦，以最大化被攻擊者最小化的總體攻擊成本。其後，我們採用以拉格蘭日鬆弛法為基礎的方法解決上述問題，並藉由電腦實驗結果證明所提出解決方案之優異性。

關鍵字：服務持續性、冗餘配置問題、存活度、攻防情境、多重核心節點、攻擊經驗累積、最佳化、數學規劃、拉格蘭日鬆弛法

THESIS ABSTRACT

GRADUATE INSTITUTE OF INFORMATION MANAGEMENT

NATIONAL TAIWAN UNIVERSITY

NAME: RAY, JUI-PIN LO   MONTH/YEAR: August/2009

ADVISOR: FRANK, YEONG-SUNG LIN

Redundancy and Defense Resource Allocation Algorithms to Assure Service Continuity against Natural Disasters and Intelligent Attackers

In recent years, Business Continuity Management (BCM) has become an important issue to organizations. Within the scope of BCM, Disaster Recovery Planning (DRP) is one of the most IT-related problems, and redundancy is a frequently used approach to implement DRP. However, previous research on Redundancy Allocation Problem (RAP) focused on dealing with the problems related to reliable systems, instead of network systems that organizations rely on. Therefore, we discuss RAP in network environments. By efficient use of redundancy together with extra defense mechanisms, we attempt to ensure a network's service continuity, and enhance its survivability against malicious attackers that utilizes accumulated experience.

We construct an attack/defense scenario, in which an attacker and a defender competing against each other, and formulate it as a two-phase nonlinear integer programming problem. In the inner problem, AEA model, the attacker that utilizes accumulated experience attempts to minimize the total attack cost of compromising all core nodes in the network by choosing appropriate targets to compromise. By contrast, in the outer problem, RAP-EDM model, the defender allocates proper redundancy and extra defense mechanisms to maximize the minimized total attack cost under the consideration of a limited defense budget. We adopt a Lagrangean Relaxation-based solution approach to resolve the problem above, and further prove the efficacy of our approach by computer experiments.

Key Words: Service Continuity, Redundancy Allocation Problem (RAP), Survivability, Attack/Defense Scenario, Multiple Core Nodes, Attack Experience Accumulation, Optimization, Mathematical Programming, Lagrangean Relaxation

Table of Contents

謝誌 I
論文摘要 III
THESIS ABSTRACT V
Table of Contents VII
List of Tables IX
List of Figures XI
Chapter 1 Introduction 1
1.1 Background 1
1.2 Motivation 5
1.3 Literature Survey 6
1.3.1 Survivability 7
1.3.2 Redundancy Allocation Problem 11
1.4 Proposed Approach 14
1.5 Thesis Organization 15
Chapter 2 Problem Formulation 16
2.1 Problem Description 16
2.2 Problem Formulation of the RAP-EDM Model 17
2.3 Problem Formulation of the AEA Model 35
Chapter 3 Solution Approach 39
3.1 Solution Approach for the AEA Model 39
3.1.1 Lagrangean Relaxation Method 39
3.1.2 Lagrangean Problem of the AEA Model 41
3.1.3 The Dual Problem and the Subgradient Method 47
3.1.4 Getting Primal Feasible Solutions 47
3.2 Solution Approach for the RAP-EDM Model 49
Chapter 4 Computational Experiments 52
4.1 Computational Experiments with the AEA Model 52
4.1.1 Simple Attack Algorithm 1 and Simple Attack Algorithm 2 52
4.1.2 Randomized Allocation Algorithm 53
4.1.3 Core Focused Allocation Algorithm 53
4.1.4 Experiment Environment 54
4.1.5 Experiment Results 56
4.1.6 Discussion of Results 61
4.2 Computational Experiments with the RAP-EDM Model 62
4.2.1 Experiment Environment 62
4.2.2 Experiment Results 62
4.2.3 Discussion of Results 69
Chapter 5 Conclusion and Future Work 71
5.1 Conclusion 71
5.2 Future Work 73
References 75

List of Tables

Table 1-1 Definition of Survivability..9
Table 2-1 Problem Assumption and Description of RAP-EDM Model..........................27
Table 2-2 Given Parameters of the RAP-EDM Model 30
Table 2-3 Decision Variables of the RAP-EDM Model..31
Table 2-4 Given Parameters of the AEA Model..35
Table 2-5 Decision Variables of the AEA Model 36
Table 4-1 Experiment Parameter Settings of the AEA Model…………………………......55
Table 4-2 Attack Results of the Experiments on Different A/D Cost Functions.............57
Table 4-3 Attack Results of the Experiments on Different Network Sizes.....................58
Table 4-4 Attack Results of the Experiments on Different Defense Budget...................59
Table 4-5 Attack Results of the Experiments on Different Choice Set Sizes 60
Table 4-6 Attack Results of the Experiments on Different Numbers of Functions.........61
Table 4-7 Experiment Parameter Settings of the RAP-EDM Model 62
Table 4-8 Defense Results of the Experiments on Different A/D Cost Functions..........63
Table 4-9 Defense Results of the Experiments on Different Network Sizes 64
Table 4-10 Defense Results of the Experiments on Different Defense Budget..............64
Table 4-11 Defense Results of the Experiments on Different Choice Set Sizes.............65
Table 4-12 Defense Results of the Experiments on Different Numbers of Functions....65

List of Figures

Figure 1-1 Percentage of Organizations with BCP [6]..3
Figure 1-2 Percentage of Losses Due to Insiders [18] 6
Figure 2-1 Initial Situation...22
Figure 2-2 Compromising a Non-core Node 23
Figure 2-3 Compromising a Core Node...24
Figure 2-4 Considering Attack Experience Accumulation 25
Figure 2-5 Achieving Attack Goal 26
Figure 2-6 Explanation of Figure 2-1~ Figure2-5...27
Figure 3-1 Process of Lagrangean Relaxation Method..41
Figure 4-1 Defense Results of the Experiments on Concave A/D Cost Function 66
Figure 4-2 Defense Results of the Experiments on Linear A/D Cost Function..............66
Figure 4-3 Defense Results of the Experiments on Convex A/D Cost Function............66
Figure 4-4 Defense Results of the Experiments on Different Network Sizes.................67
Figure 4-5 Defense Results of the Experiments on Different Defense Budget 67
Figure 4-6 Defense Results of the Experiments on Different Choice Set Sizes 68
Figure 4-7 Defense Results of the Experiments on Different Numbers of Functions 68

Chapter 1 Introduction

1.1 Background

Due to the convenience and efficiency of information technology (IT) especially in the realm of computer networks and the Internet, more and more businesses have been running their routine operations and even providing service for their customers with the help of IT in recent years. Although IT brings businesses several advantages, such as improving efficiency and reducing operating cost, there are too many potential threats to these important IT elements in the real world, such as earthquakes, tsunamis, hurricanes, typhoons, floods, abuse of employees, power outages, terrorisms, and hacker attacks. According to the sources, all of above can be mainly divided into four categories, including hazardous events, human errors, utility disruptions, and man-made malicious attacks from outsiders. These four kinds of potential threats are also known as the main causes of business interruption [1] because a business may suffer from the interruption of IT-supported operations or business processes when accidents affect its' IT [2]. Maybe all of these potential threats are quite difficult to predict and even to prevent; however, from the aspect of business management, the businesses still have to make satisfactory preparations for the worst situations. It is because every business interruption caused by whatever risks can really damage the benefits of a business, e.g., the losing profits of potential transactions during business downtime, serious damage to reputation, and the loss of customers.

For customers, one of the greatest values of a business is to provide them with the service satisfying their demands in time, i.e., businesses should provide service continuously. To achieve the ultimate and challenging goal, most businesses must try their best to prevent important business processes from interruption so as to ensure their service continuity. More concretely speaking, every business must develop an overall and concrete plan for reducing the risks caused by any potential threats, which break down the critical business operations. Moreover, every business needs to help itself recover from incidents at short. Therefore, an important issue based on what we have mentioned above is proposed afterwards. It is well known as Business Continuity Planning (BCP), or more generally known as Business Continuity Management (BCM). In the previous literature, Lam proposed a BCP cycle consisted of eight core steps [3] to provide a stepwise method for IT-related organizations in 2002.

Since more and more businesses paid much attention to BCM, British Standard Institution (BSI) established a standard named as BS25999 [4] in 2006. It is to provide practical guidelines for businesses to actually implement BCM. There are mainly two parts in BS25999. BS25999-1 provides businesses with a systematic methodology to develop and implement suitable business continuity plans. BS25999-2 is consisted of some specifications and requirements for checking and auditing the business continuity plans of businesses.

In the first part of BS25999, the concept of BCM lifecycle is of extreme importance. It consists of several main stages. First of all, a business should make use of both Business Impact Analysis (BIA) and Risk Assessment (RA) to find out the critical processes, elements, and their corresponding possible risks. After realizing what the potential risks are, the business also needs to further estimate the possible impact on itself caused by every risk. According to the results from the analysis above, the business can therefore establish its’ own BCM strategies and develop a set of business continuity plans, which is including designation of personnel, distribution of responsibilities, training and rewarding plans, backup and reaction procedures, and Disaster Recovery Planning (DRP). After producing a whole plan, the most important things are practicing, auditing regularly, and updating if necessary. Finally, the business should make BCM become a part of its' own business culture. By implementing BS25999 thoroughly, a business will tend to be much tougher while suffering from real accidents or will recover form disruptions with less cost in shorter time.

According to an electronic survey of 7,548 respondents from companies and public sector organizations around the global conducted by Frost & Sullivan and (ISC)^2 in 2008 [5], 73% of respondents view the impact of service downtime as top priority. In addition, another survey of 500 IT executives around the US conducted by AT&T in 2008 [6] also provides some convincing evidence about the importance of business continuity. It shows that BCP was seen as a priority by 71% IT executives. 80% IT executives indicated that their companies had a set of plan for business continuity as shown in Figure 1-1. This exhibits the fact that BCM is an obvious and critical issue to businesses even to date.

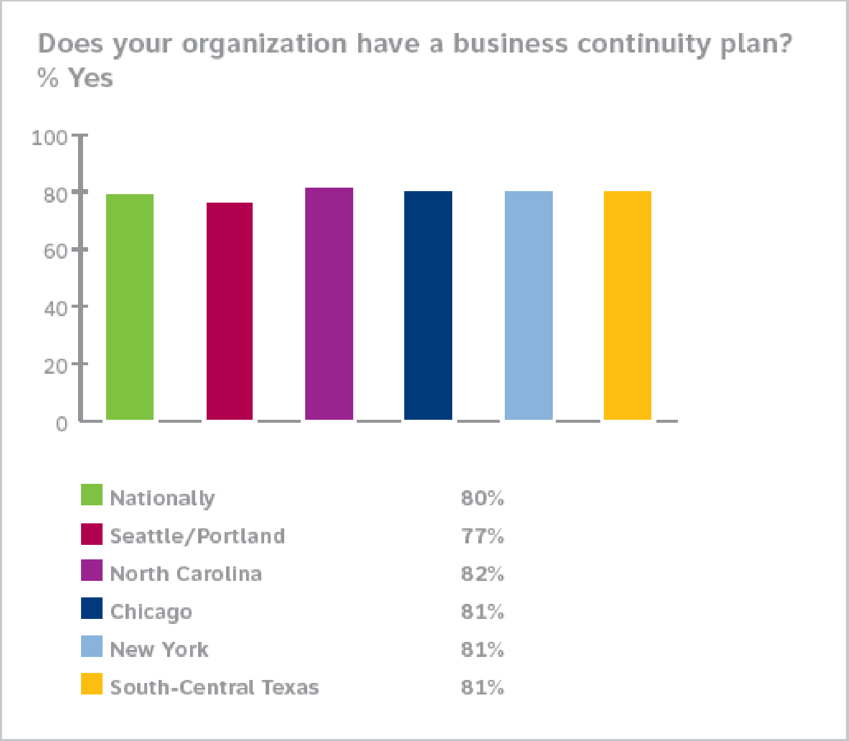

Figure 1-1 Percentage of Organizations with BCP [6]

Disaster Recovery Planning (DRP) mentioned before is not only the most IT-related portion of BCM but also a critical issue to organizations. In survey [5], the respondents either from America or Asia-Pacific all view business continuity and disaster recovery solutions as the security technology of third priority being deployed.

Some research includes BIA and RA within the scope of DRP [7] [8], but we focus on the actual planning issues of DRP here because we have finished those analysis in the first phase of BCM lifecycle. According to [8], the definition of a disaster recovery plan is *an internal control and security system which focuses on quick restoration of service for critical organizational processes when there are operational failures due to man-made or natural disasters*. There are many components can be taken into consideration of DRP, including backup methods, alternate sites, support teams, and equipments' replacement [7], and the use of existing compatible on-site equipments to replace similar equipments that have failed. This replacement is treated as the concept of redundancy as well.

Redundancy is one of the security approaches commonly used to cope with IT disaster recovery [2]. For a system, allocating redundancy is absolutely an effective solution to mitigate the potential risks of operational interruption. It is because the identical-functioned redundant components in the hot-standby state can take over each others' work immediately when any possible disaster fails some of them. This feature of redundancy sufficiently fulfills the requirements of continuous service, and in practice, there are actually many system designs using functionally similar but not exactly the same components in parallel [9]. All of these problems concerned with allocation of redundancy are generally defined as Redundancy Allocation Problem (RAP), and which is applied to several research fields [2] [9] [10] [11] [12].

The theoretical fundamentals above combine the perspective of the business with the view of IT-related departments. Based on these fundamentals, this thesis focuses on exploiting redundancy allocation to cope with the challenges of critical IT service continuity, especially for networks.

1.2 Motivation

As mentioned previously, many businesses make good use of IT to keep daily operation or to provide service, especially networks. Such kind of web-based service can be suitably applied to exchanging electronic data and even proceeding electronic transactions. Unfortunately, the risks of information leakage, real economical losses, or even being attacked to service interruption always show up together with the convenience brought by the use of IT. It is because obviously there is no perfectly safe system or network in reality. Thus, the concept of survivability, which mainly concerns the availability of a whole system during accidents rather than the capability of resisting threats in practice, is being widely discussed more often at present [13] [14] [15] [16] [17] .

Unlike the traditional term “security” which includes only two extreme statuses, “safe” or “compromised,” survivability can be interpreted as a spectrum of safe degree. The extreme points on both sides are “safe” and “compromised,” respectively. From the perspective of DRP, survivability can be also treated as a measurement of the ability to keep service continuity even in the event of a threat. It seems obvious that survivability is much more practical and almost fully complies with the spirit of BCM and DRP as well. Hence we spontaneously take survivability as a major measurement for further discussion on networks.

On the other hand, the past research about RAP mainly applied it to parallel [2] [12] or series-parallel systems [9] [10] [11] instead of network structures, even though RAP has been extensively discussed for years. Besides, the major concerns of past RAP research are often the system reliability considering natural disasters or random errors, rather than the survivability of networks suffering from intelligent malicious attacks. In the 2008 CSI survey [18], there were 522 computer security practitioners as respondents in different kinds of organizations around the U.S. The result of the survey shows that the percentage of respondents that attribute losses to non-insiders jumps from 36% in 2007 to just over half (51%) in 2008, as shown in Figure 1-2. In other words, it indicates that more respondents believed their losses were due to attacks from outside of the organization; therefore, this evidence again convinces us that we should pay more attention to the impact caused by malicious attacks from outsiders, especially while considering the features of threats within network environments.

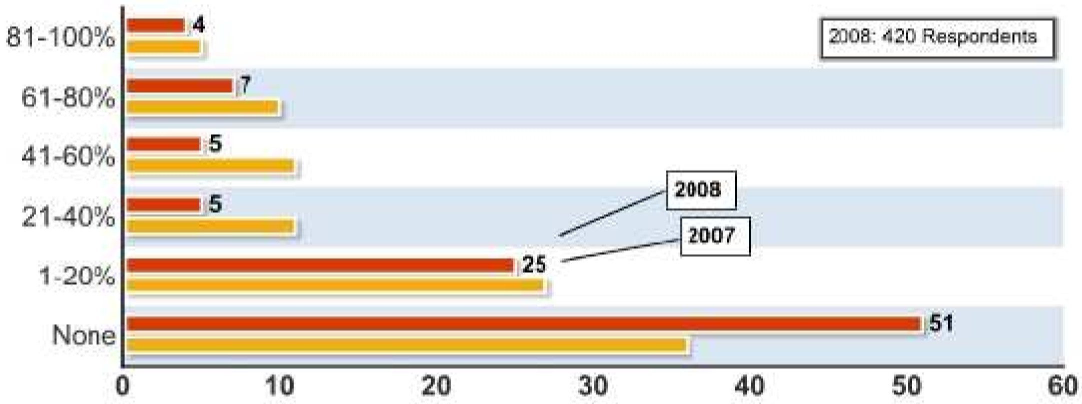

Figure 1-2 Percentage of Losses Due to Insiders [18]

To summarize all of above, this thesis attempts to compensate for insufficient RAP research about networks. More specifically, we will discuss how to make efficient use of redundancy to ensure a network’s service continuity and enhance its survivability while facing intelligent malicious attacks from outsiders. The result, therefore, could become another guideline for network operators when constructing a network with both service continuity and high survivability by implementing redundancy allocation.

1.3 Literature Survey

The subject of this thesis is how to exploit redundancy allocation to enhance a network's survivability with the regard for service continuity. Since we already deliberate upon service continuity in last two sections, our following discussion on related works will focus on the two remained topics, survivability and redundancy.

1.3.1 Survivability

Many researchers and businesses have taken survivability seriously since about 1990s; however, there is still not a consistent or standard definition of survivability according to a survey consisted of many papers concerning survivability [13]. Among all the related works, the most sited one was proposed by Ellison et al. [14] in 1997, and in which the most famous definition of survivability was also provided. Because the number of survivability-related studies is not only huge but still increasing, we pick some research on survivability in different fields and then introduce them following a timely manner below.

Jiang [15] discussed survivability from the viewpoint of military, and first of all, he defined survivability as *a measure of the degree of keeping the performances of a kind of military weaponry, equipments, or other military forces, during enemy's attacks*. He then turned to focus on the survivability of communication networks and declares that a good definition of survivability should help network operators evaluate a network's survivability and design a network with high survivability under some constraints. Finally, he proposed a new definition of survivability based on traffic flow, and that not only covered the old one based on connectivity but also revealed more correct information. When considering a network composed of N nodes and with original traffic flow A_n; thus, the generalized form of survivability s based on traffic flow can be shown as follow:

$$s = \frac{P_a A_{na} + P_b A_{nb} + \ldots + P_n A_{nn}}{\left(P_a + P_b + \ldots + P_n\right) A_n} \times 100\%$$

, where A_{na}, A_{nb}, $\cdots$ A_{nn} respectively stand for residual traffic while node A, node B, $\cdots$

node N destroyed, and P_a, P_b, $\cdots$ P_n mean destruction probability of each node, separately.

Some research in the field of wireless and mobile networks also involves the discussion of survivability, for instance, [16] completed by Malloy et al.. They defined survivability as *a network's ability to perform designated functions under the condition of some network infrastructure component failures resulting in a service outage* first, and then aimed at canvassing some issues about the outages mainly described by the number of subscribers or services affected, and the duration of the outages. They discussed some existed measurements of these outages and gave some suggestions for improving a wireless or mobile network's survivability involving architectural changes as well. In addition, they also pointed out an important fact that there is a tradeoff between extra expenditures and customers' satisfaction while service providers dealing with survivability of networks.

Besides generalizing that the definition of survivability was inconsistent, in the research [13], Westmark also found that few papers really involved in computing survivability, and even survivability has been calculated in some papers, the calculations were almost informal and not used in practice. Accordingly, the author provided a template to help people precisely define survivability, and that clearly expounded a definition of survivability should be composed of five required elements, including system, usage, minimum level of service, threats, and a business case.

Furthermore, Zhang et al. [17] proposed a network survivability analysis model based on attack graph to deal with intrusions, and there were three major steps to acquire the analytical result. At first, they gather vulnerabilities information and set their difficulty parameters in preparation step, and then detect vulnerabilities in targeted network by scanning tools to generate attack graph in the second step. Within an attack graph, the nodes stand for different states of hosts, i.e., different privileges got by attacker on hosts, in observed network environment; then, the directed links between those nodes are translations between different states with corresponding vulnerabilities which can be exploited. For a same goal, there maybe exists several routes to reach it, and these routes are defined as intrusion scenarios. In the final step, they estimate survivability of targeted network after confirming the level that cannot be tolerated, and further analyze the result for improving the survivability. The quantification of survivability *Sur* is shown below:

$$Sur = \min\left\{F_m\right\}_{m=1,\ldots M} = \min\left(1-\prod_{i=1}^{l_m}\left(1-W_{m_i}\right)\right)_{m=1,\ldots M}$$

, where W_{m_i} are levels of attack difficulty on vulnerabilities listed in the m^{th} intrusion scenario, l_m is the number of vulnerabilities under the m^{th} intrusion scenario, and M is the number of intrusion scenarios in total.

In the end of this subsection, we also get some definitions of survivability mainly from above discussed works together in the following table, and hope that will be helpful constructing a general outline of survivability for further research.

Table 1-1 Definition of Survivability

No.	Researcher(s)	Definition	Year	Origin
1	T.-Z. Jiang	● Survivability is a measure of the degree of keeping the performances of a kind of military weaponry, equipments, or other military forces, which undergoing enemy's attacks. ● Survivability of a communication network based on "connectivity" is defined as the probability of node pairs which still have one path at least, when the network being attacked by the enemy. This probability can be determined, if the topology of network and the destroyed nodes (or links) are given. ● When the nodes (or links) of a	1991	[15]

		communication net being destroyed, the remaining traffic flow (in percentage of original traffics) in the network is defined as the survivability of the network based on “traffic flow”.		
2	D.A. Fisher, H.F. Lipson, N.R. Mead, R.C. Linger, R.J. Ellison, and T. Longstaff	● Survivability is the capability of a system to fulfill its mission, in a timely manner, in the presence of attacks, failures, or accidents. ● The term system is used in the broadcast possible sense, including networks and large-scale systems.	1997	[14]
3	A.D. Malloy, A.P. Snow, and U. Varshney	● Survivability is a network’s ability to perform its designated set of functions given network infrastructure component failures, resulting in a service outage, which can be described by the number of services affected, the number of subscribers affected, and the duration of the outage.	2000	[16]
4	V.R. Westmark	● Survivability is the ability of a given system with a given intended usage to provide a pre-specified minimum level of service in the event of one or more pre-specified threats. ● Thus, to precisely define survivability requires a precise definition of: the system, the usage, the minimum level of service, and the threats.	2004	[13]
5	L. Guo, L.-J. Zhang, W. Wang, W. Yang, and Y.-T. Yang	● Survivability is the ability of a system to continue operating despite the presence of abnormal events such as intrusions.	2007	[17]

6	ATIS Telecom Glossary 2007	● Survivability is a property of a system, subsystem, equipment, process or procedure that provides a defined degree of assurance that the named entity will continue to function during and after a natural or man-made disturbance. ● Note: Survivability must be qualified by specifying the range of conditions over which the entity will survive the minimum acceptable level or post-disturbance functionality and the maximum acceptable outage duration.	2007	[19]

1.3.2 Redundancy Allocation Problem

Though redundancy means extra resource requirement, it has been still applied to different fields, for example, software programming, reliable system design, and infrastructure construction. Sometimes, redundancy is even treated as one of high priority solutions for improving reliability or survivability. Since there is no doubt about the importance of redundancy, many scholars propose different approaches to solve RAP. To better understand and effectively apply redundancy, we then briefly go through some related works handling RAP in different ways.

The research [9] by Coit and Smith can be seen as the pioneer that allowed mixing functionally similar yet different redundant components in parallel within subsystems, because they used a artificial search heuristic, Genetic Algorithm (GA), to cope with RAP in series-parallel systems. In their work, they used GA to solve two kinds of problems, which were maximizing reliability given total cost and weight constraints, and minimizing total cost given reliability and total weight constraints. Finally, the result of experiments exhibited the fact that though the natural of GA could not guarantee optimality, this approach was still better than some past benchmarks.

Hsieh [10] also considered redundancy allocation problems with multiple component choices in series-parallel systems, and notably which could be generalized typical RAP while reducing the component choice of every subsystem to 1. Instead of heuristics, he proposed a novel LP (Linear Programming) approach to cope with RAP subjected to multiple separable integer constraints in series-parallel systems, and that was consisted of two stages. In the first stage, he reformulated the original problem to a new form applying to software computing, and then actually calculated the initial redundancy allocation. By using the results from stage one, he reallocated the unused resource for best enhancement of reliability. Finally, he implemented computer experiments to proof that proposed LP approach is better than two other kinds of methods when the available resource is abundant, and most important of all, the computation time consumed by which is quite shorter.

Ramirez-Marquez et al. [11] thought GA approach proposed by Coit and Smith [9] was time consuming and cumbersome, even though that indeed provided a satisfying solution to the reliability maximization problem. Therefore, they considered RAP in series-parallel systems from another viewpoint, and that is the least reliable part of a system dominates the reliability of whole system; thus, they decided to focus on maximizing the minimum subsystem reliability. To well solve this max-min problem, they suitably transformed original problem into an equivalent linear formulation for complying with the requirement of linear programming, just like what Hsieh has done in [10], and then used a set of commercial software, LINGO, to obtain solutions. For comparison with past research, they conducted experiments on three famous RAP samples, and the result of experiments showed an interesting fact that is the max-min approach can also acquire the optimal solution of maximizing whole system reliability while problem size is small. However, taking the max-min approach as a surrogate can just get a good solution for maximizing whole system reliability, rather than an optimal one, when facing a RAP problem with large size.

From the angle of DRP, Shao [2] incorporated redundancy into a firm’s critical IT functions to maximize the overall survivability against potential disasters. In his work, he considered the situation that there is a business with many different critical IT functions, and same IT function can be implemented by a set of IT assets. Thus, the problem about getting the maximized total survivability S^* in an organization with M IT functions, while facing D potential disasters with different probability P_d, was formulated as a 0-1 integer programming problem with nonlinear objective function shown below:

$$\max S^* = \sum_{d=1}^{D} P_d \left[\sum_{m=1}^{M} w_m \left(1 - \prod_{i=1}^{n_m} v_{imd}^{\;\;x_{mi}} \right) \right]$$

, where w_m was IT function m’s importance weight, n_m stood for the number of candidate assets of IT function m, and v_{imd} meant the failure probability of m-functioned asset i during disaster d. Accordingly, x_{mi} was the decision variable, that meant whether to allocate asset i for IT function m; of course, there were some constraints should be conformed to, such as budget, redundant level, i.e., the minimum required number of assets for each IT function. Afterward the problem was transformed and solved by a procedure based on probabilistic dynamic programming, and final result of a simulation was also provided for proving effectiveness and scalability.

Unlike classical RAP, Levitina and Hausken discussed redundancy together with protection while facing intentional attacks based on a probabilistic approach [12]. They considered a system built from N identical parallel elements with a same functionality, and the vulnerability of each element was determined by a so call “attacker-defender contest success function”. Because the situation considered in their research was that the attacker could not attack precisely on certain target and the defender knew nothing about how the attacker conducts attacks, they took the expected value of total damage caused by unsatisfied utility of whole system as the main measurement. Afterward, by respective case analysis, they generalized some conclusions for defender when making the choice between deploying redundant elements and concentrating on protection of a few elements under different conditions.

1.4 Proposed Approach

To both deliberate upon how to exploit redundancy to enhance network survivability against malicious attacks and take into consideration of assuring service continuity, we construct an attack/defense scenario in which an attacker and a defender fighting against each other in a given network environment. In order to get a solution of the complex problem with dynamic changes sufficiently, we decide to adopt mathematical programming technique. Thus, we appropriately formulate it as a max min mathematical programming problem, which is Redundancy Allocation Problem with Extra Defense Mechanisms (RAP-EDM) model, and then treat it as a two-phase problem to get a satisfying solution.

In the first phase, we focus on the inner problem of RAP-EDM model, the Attack with Experience Accumulation (AEA) model, which stands for the behavior of the attacker. Therefore, we give an initial allocation of redundant components (with defense mechanisms) that meets the service continuity requirement in terms of each node's service availability assurance, and then make use of Lagrangean Relaxation method with the subgradient method to get a solution of the AEA model. Afterwards, the result of the AEA model, viewed as the result of attacks, is taken to be the input of the RAP-EDM model. Thus, we can modify the initial redundancy and defense allocation strategy against the given attacks.

Moreover, the adjusted allocation strategy can be another starting point of the next attack action. The latest attack result, of course, can be again inputted into the

RAP-EDM model to help produce better allocation strategy as well. After several cycles of attack and defense resource allocation adjustment processes, a near optimal redundancy and defense allocation strategy against the intelligent malicious attacks is finally developed, and that is exactly what we propose to achieve.

1.5 Thesis Organization

After the introduction provided in this chapter, the rest content of this thesis is organized as following: RAP-EDM model and AEA model are introduced in Chapter 2; the solution approaches of these models are presented in Chapter 3; the computer experimental result of our approach is illustrated in Chapter 4; conclusions and possible research directions are provided in Chapter 5 as the ending of this work.

Chapter 2 Problem Formulation

2.1 Problem Description

The problem we address here is how to make good use of redundancy to not only assure legitimate users of service continuity but also maximize whole network's survivability against intelligent malicious attacks at the mean time. Besides, we consider above in the situation that there exists a limitation of defense budget. Obviously, there are two main measurements we must evaluate appropriately, that are service availability of each node and survivability of whole network.

In this work, the service continuity in terms of every node's service availability is assured by the contribution of redundancy; in more precisely speaking, if we make sure that the expected number of redundant components in every node always satisfies with a predefined operating minimum requirement, each node can provide required service without interruption, even when some of redundant components in it are failed by random errors, natural disasters, or malicious attacks.

From the angle of deterring attackers from attacking a system or network, defense resource is used to increase the attack cost that an attacker has to pay for attacking successfully [20]. Based on such argument, we decide to treat the total attack cost needed to spend for achieving ultimate goal of disrupting all mission-critical service in the target network as the measurement of whole survivability in this thesis.

To fully describe the above problem, we create an attack/defense scenario where an attacker and a network operator, as a defender, competing with each other within a given network environment, and both of them are wise enough to dynamically change their own strategy according to the enemy’s action. Afterward we can utilize the concept of optimality to formulate such a complicated scenario as mathematical models, and the details of which we will discuss in the next section.

2.2 Problem Formulation of the RAP-EDM Model

For thoroughly explaining what we discuss in RAP-EDM model, we first describe the attack/defense scenario together with some figures, and then provide a list of assumptions and given conditions. Finally, we will formally introduce the mathematical part of this model.

Considering a network consisted of AS-level nodes, i.e., each node can be seen as an AS-level domain, there is just one kind of specified service function provided by each node, e.g., web server, ftp site, mail server, and the plan about which node should provide what kind of service function is predefined and consistent. Furthermore, there are multiple core nodes providing mission-critical service or storing important information in this network, but a non-core node may just provide transmission, rather than a specified service function. In Figure 2.1, different shapes of nodes are different functions nodes should provide, and the blue-colored nodes with “c” are core nodes.

Considering malicious attacks, natural disasters, and random errors simultaneously, the defender hopes to enhance the survivability of whole network with the regard for assuring service availability of every node by exploiting unified purchase to implement redundancy allocation. First of all, the defender gets a list of products from the vendor, and that lists all available kinds of redundant components providing each specified function of different brands or types, i.e., the redundant component choice set of different specified function. Besides, for each kind of redundant component, there are several extra defense mechanisms, e.g., firewall, IDS/IPS, anti-virus, anti-spam, application level firewall, of different brands or types, which are especially appropriate for being chosen to provide further protection, and such can be defined as the defense mechanism choice set of different redundant components.

Within each redundant component choice set and defense mechanism choice set, different kinds of redundant components or defense mechanisms have different prices, i.e., the costs of allocation are different for the defender. Of course, the defense abilities of different kinds of redundant components and defense mechanisms are different, and different kinds of redundant components also have different reliabilities, i.e., the probability that a redundant component operates properly. Thus, the defender as an operator of this network has to choose the appropriate redundant components to allocate in each node from the redundant component choice set of the specific function predefined for each node.

When allocating a redundant component, the defender must decide whether to deploy extra defense mechanisms to it simultaneously, and if the answer of above is yes, the defender also has to choose the appropriate ones from the corresponding defense mechanism choice set of the kind of redundant component at the meantime.

Because natural disasters and random errors may happen during operation of redundant components, the defender also has to make sure of service availability for legitimate users in each node. Thus, when allocating redundant components to each node, the number of redundant components must satisfy the requirement of service availability assurance, i.e., the expected number of redundant components must be no less than the service availability threshold for every node. On the other hand, the defender also needs to take the capacity limit of all nodes into consideration when allocating redundant components.

The situation that the defender allocates redundant components and defense mechanisms is also illustrated in our figures, for example in Figure 2-1, there are several medium and small graphs with the same shape of each node within nodes, and which are stand for redundant components and defense mechanisms, respectively. Besides, these graphs with different patterns mean different kinds of redundant components or defense mechanisms.

After efficient allocation of redundant components in each node, the service availability of each node for legitimate users can be improved significantly. Besides, if the attacker wants to really disrupt a core node, he/she must compromise all of the same-functioned redundant components allocated in it. This approach cannot only substantially enhance the network's survivability against malicious attacks but also make every node provide more reliable service. Thus, the defender achieves the ultimate goal which is maximizing the total attack cost of compromising all core nodes regarding service availability of every node and limitation of total defense budget.

Just like the defender, the attacker also has the perfect knowledge about this target network, including either the topology of the network or the allocation of redundant components and extra defense mechanisms in each node. Furthermore, the attacker also knows the threshold of attack cost required for compromising each kind of redundant component or extra defense mechanism.

The attacker is on the known initial node at first, and he/she then compromises one node at a time until compromising all of core nodes step by step. Since the attacker's ultimate goal is making all the core nodes of the target network fail to provide any critical service with minimized attack cost, he/she prefers penetrating surreptitiously instead of destroying before actually reaching core nodes. Thus, the attacker will compromise just one redundant component, the primary one, in non-core nodes for penetrating, and then he/she can make use of such node as a hop site to reach further nodes, which are closer to core nodes. While achieving core nodes, he/she will compromise all redundant components in core nodes for whole dysfunction without doubt. Spontaneously, there are two different meanings of "compromised" produced in this network, the one is “a non-core node is penetrated”, and the other is “a core node is totally dysfunctional”. By the way, the two kinds of attack action discussed here can be found in Figure 2-2 and Figure 2-3, respectively.

Furthermore, the defender’s decision of which redundant component is primary in each node is randomized; then, there is a non-zero probability that the primary redundant component of each node is always the one that the attacker prefers attacking. Accordingly, we consider a worst case from the viewpoint of the defender, and that is the primary redundant component of each non-core node is always the one that attacker wants to compromise for minimizing total attack cost. This situation can also be seen as that, while attempting to compromise a non-core node, i.e., to penetrate it, the attacker can always arbitrarily choose the redundant component with most advantage for minimizing total attack cost to compromise. However, before actually attempting to compromise a redundant component, the attacker must compromise all of the extra defense mechanisms that have been deployed to protect it.

Because the defender implements redundancy allocation by making use of unified purchase, there are possibly same kinds of redundant components or defense mechanisms in functionally identical nodes in this target network. According to this feature, there happens an extreme situation of the attacker’s experience accumulation. If the attacker has compromised a certain kind of redundant component or defense mechanism once, he/she then found some useful methods or developed some efficient hacker tools to deal with this kind of redundant component or defense mechanism. Afterward the attacker can compromise the same kind of redundant component or defense mechanism with a comparatively low fixed attack cost. The above situation can be also considered as that the attacker continuously executes their developed methods or hacker tools to exploit existing security flaws, vulnerabilities, and bugs, to facilitate attack activities. The impact of such experience accumulation on the attacker’s decision is shown clearly in Figure 2-4.

From the perspective of the attacker, he/she has to decide which redundant components in which nodes to attack for achieving all core nodes and then actually disrupt them. Of course, the attacker we consider here is intelligent enough to sufficiently make use of the experience accumulation discussed before, while choosing appropriate attack paths to compromise all core nodes with minimized total attack cost. When all core nodes are compromised, there is an attack tree constructed, i.e., a tree rooted at node s and composed of compromised nodes, just like the attack result shown in figure 2-5.

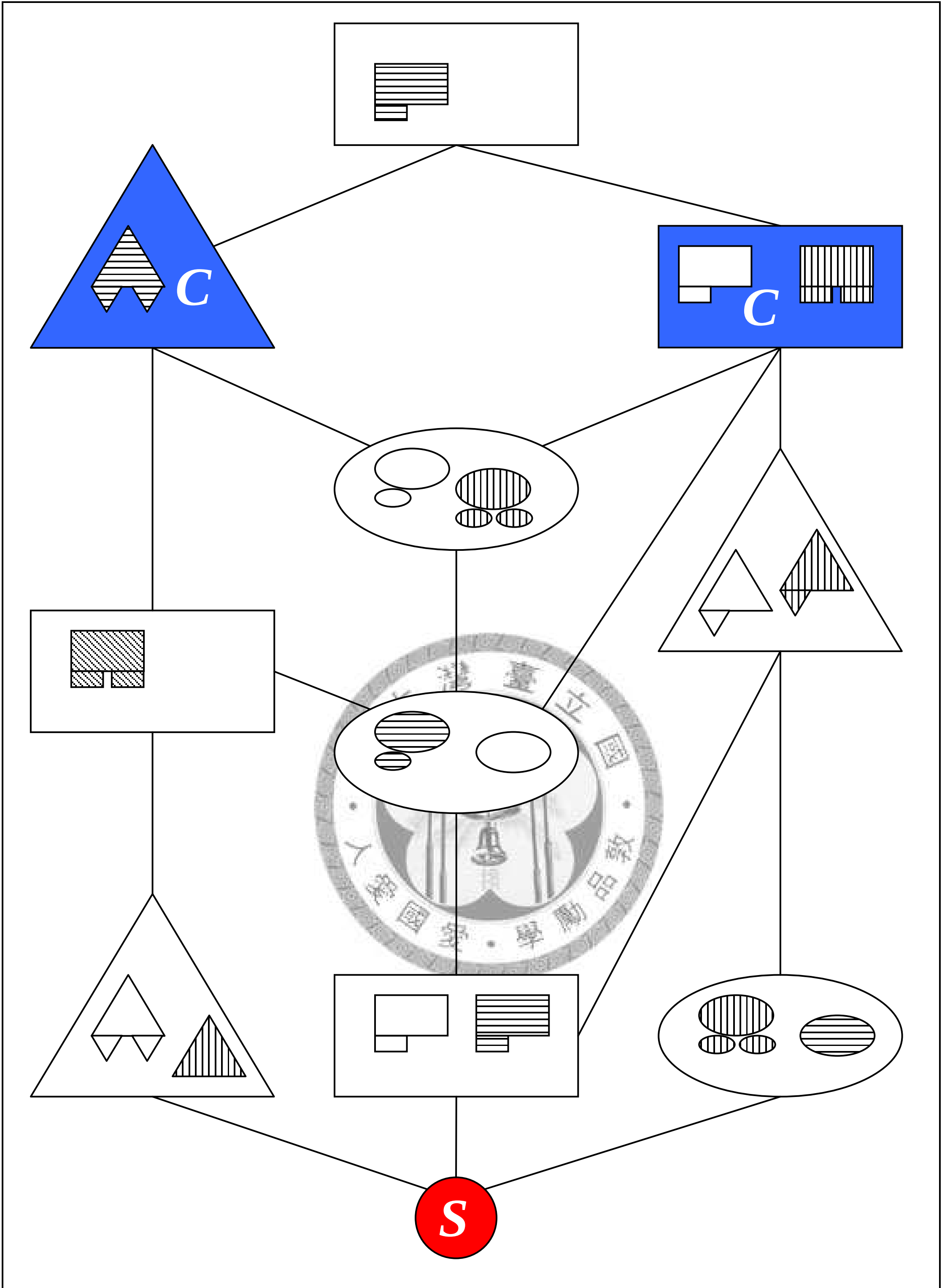

Figure 2-1 Initial Situation

The attacker is on the initial position, node *s*, and *c* means the node is one of core nodes.

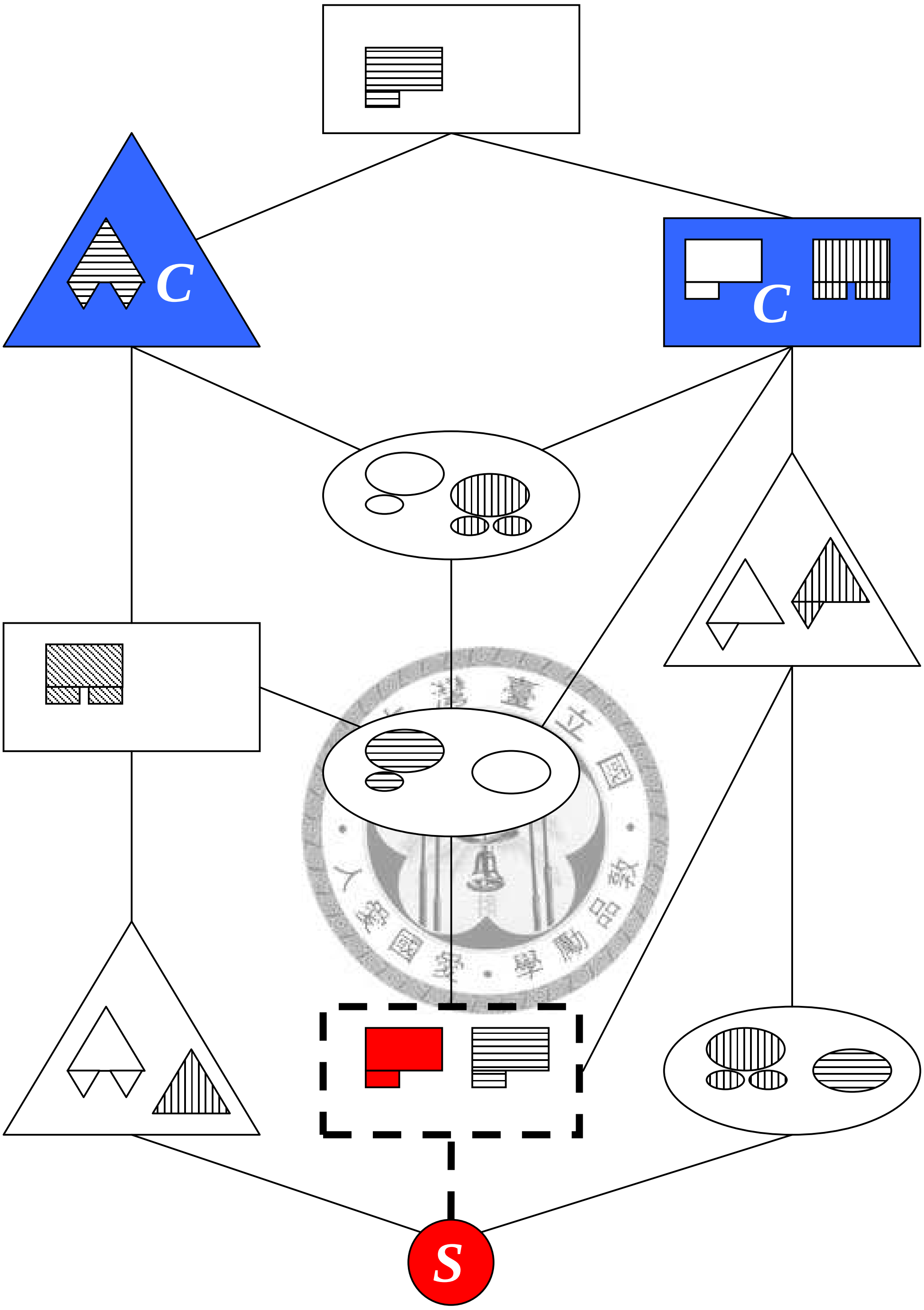

Figure 2-2 Compromising a Non-core Node

While compromising a non-core node, the attacker compromises just one of redundant components (with defense mechanisms) in such node for penetrating, and then he/she can make use of this node as a hop site to reach further nodes, which are closer to core nodes.

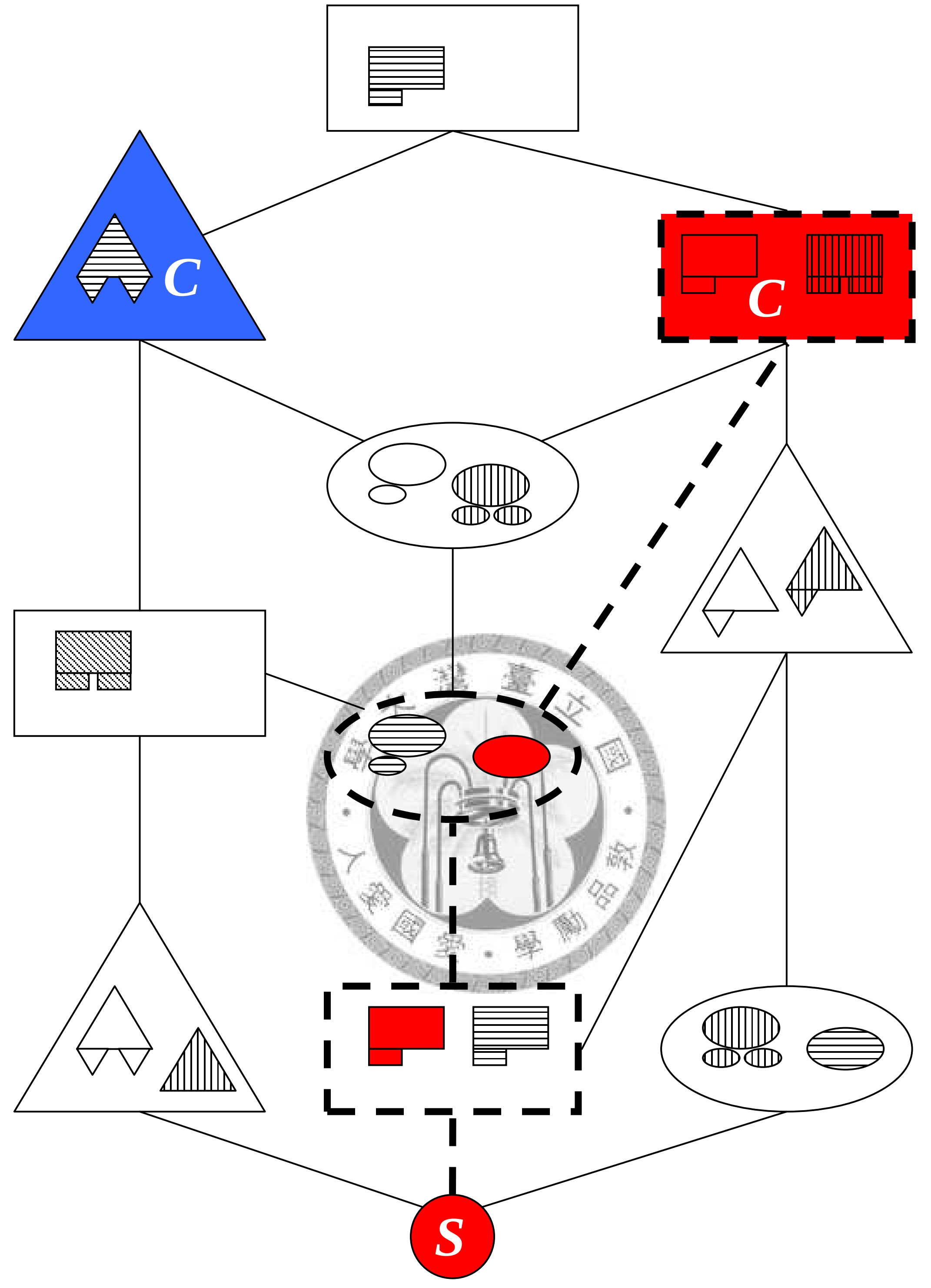

Figure 2-3 Compromising a Core Node

While compromising a core node, the attacker compromises all redundant components (with defense mechanisms) in such node for whole dysfunction.

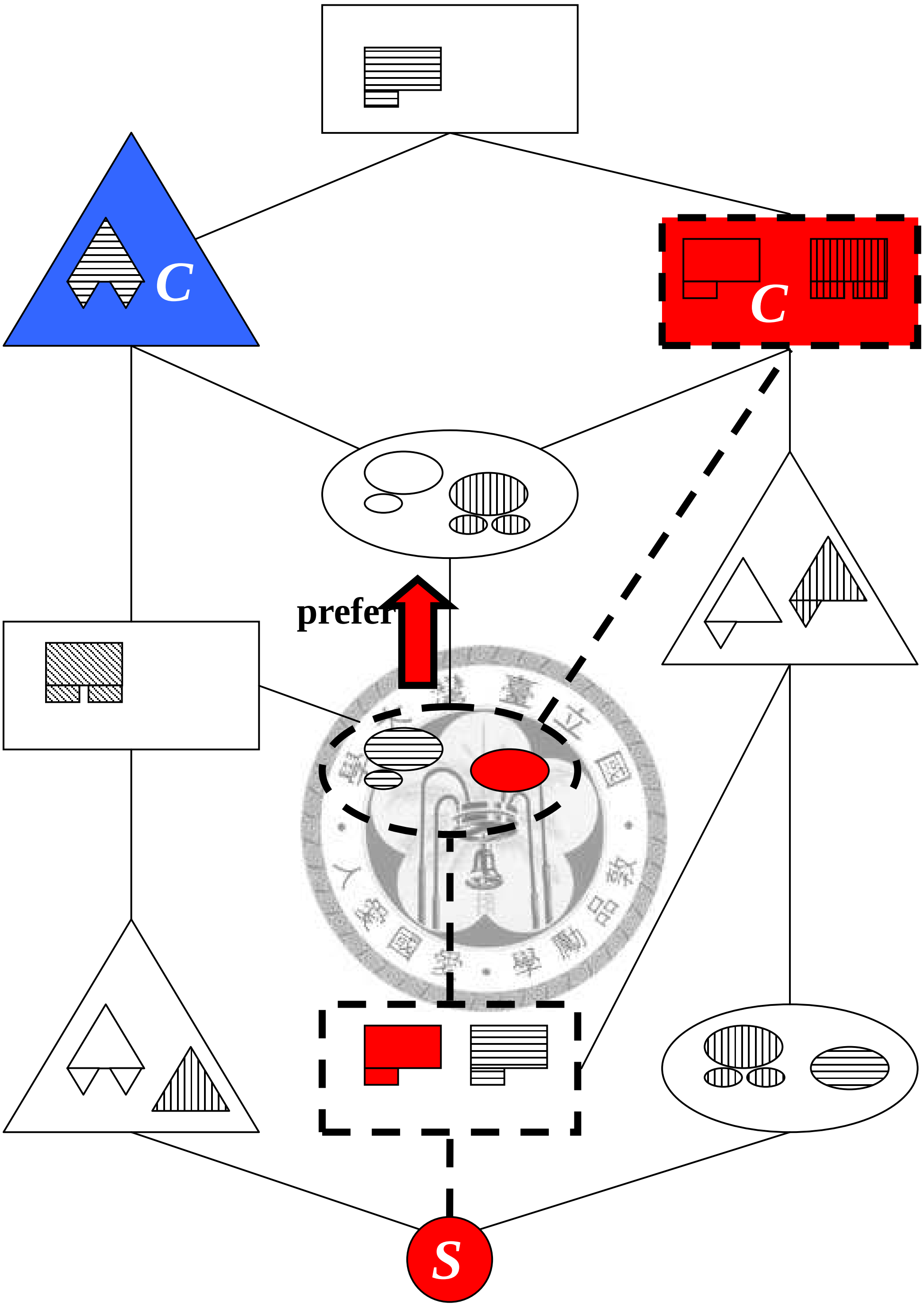

Figure 2-4 Considering Attack Experience Accumulation

After compromising some nodes, the attacker gets some experience of compromising the kinds of redundant components and defense mechanisms that he/she has compromised. Thus, he/she then prefers to compromise those nodes with more above kinds of redundant components/defense mechanisms and less ones new to him/her.

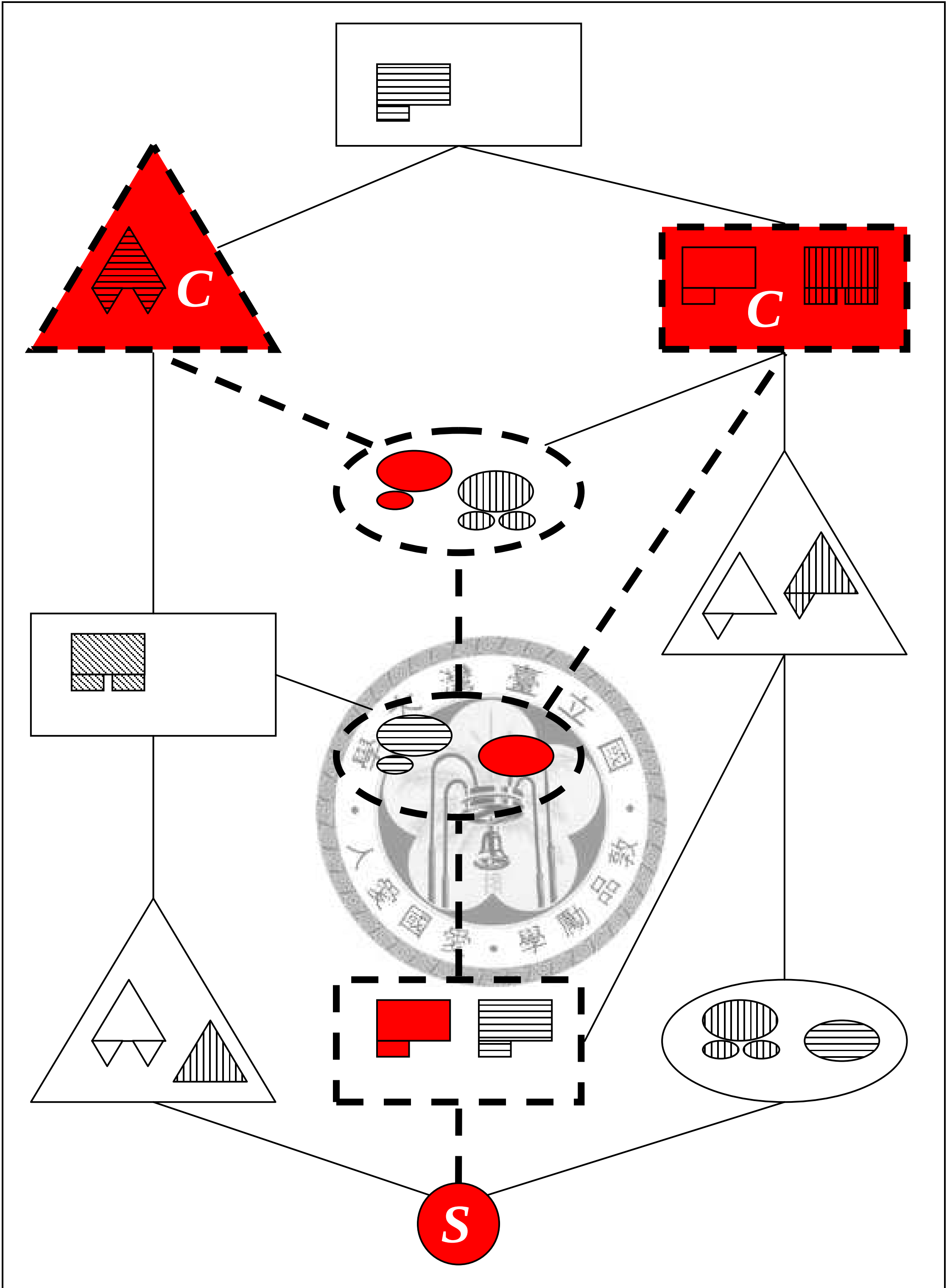

Figure 2-5 Achieving Attack Goal

The attacker keeps attacking until compromising all core nodes.

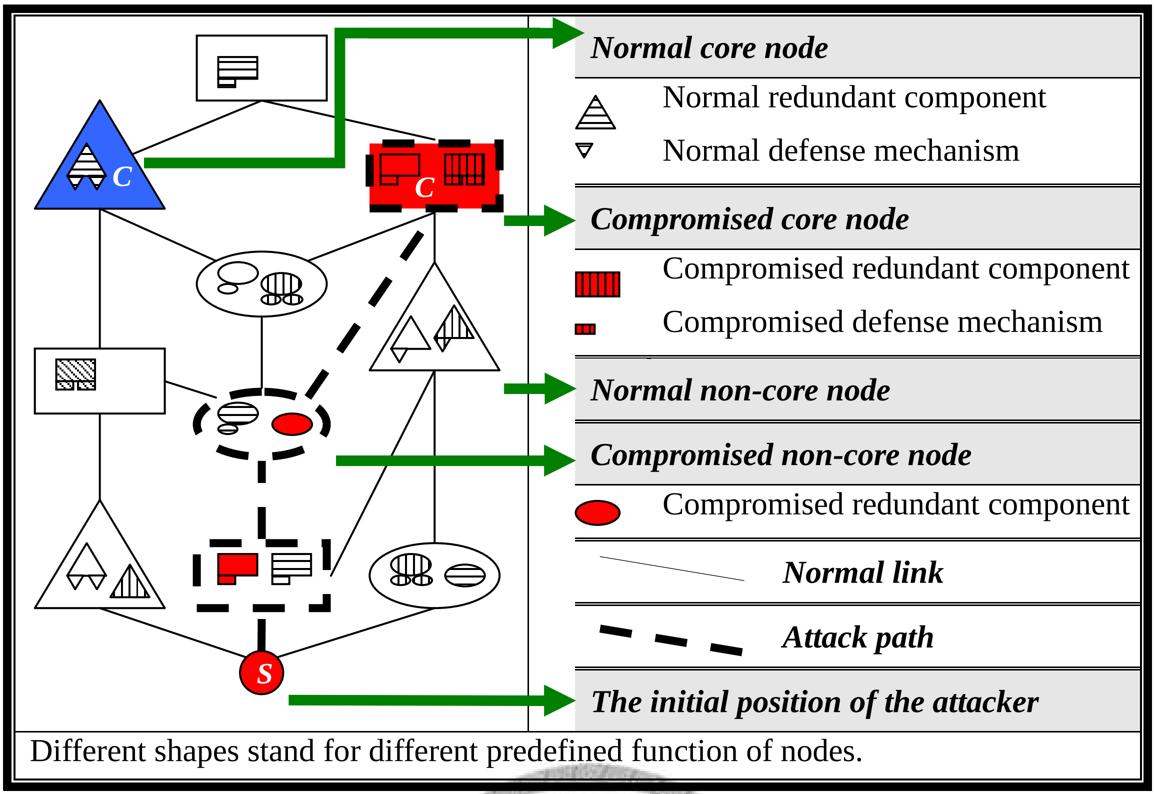
C
C
S
Normal core node
Normal redundant component
Normal defense mechanism
Compromised core node
Compromised redundant component
Compromised defense mechanism
Normal non-core node
Compromised non-core node
Compromised redundant component
Normal link
Attack path
The initial position of the attacker
Different shapes stand for different predefined function of nodes.

Figure 2-6 Explanation of Figure 2-1~ Figure2-5

After describing the scenario in detail, we then form a table below to provide all information of this problem concretely.

Table 2-1 Problem Assumption and Description of RAP-EDM Model

Assumption

- Every node in this network is at AS-level.
- No attacks on links are considered.
- No distributed denial-of-service (DDoS) attacks are considered.
- Both the defender and the attacker have perfect knowledge about this network.
- Each node in the network must provide just one kind of predefined function.
- The defender has limitation of total defense budget.
- The requirement of service availability threshold, which defines the minimum expected number of redundant components for every node, must be satisfied.
- The number of redundant components in each node cannot be more than the capacity limit.
- All kinds of redundant components in a choice set provide identical main function, and the defender selects the redundant components with the same function for different nodes from a same redundant component choice set.
- Other than providing the main function, all kinds of redundant components also

have little basic defense ability.

- All redundant components are in hot-standby state.
- All compromised redundant components are never repaired.
- There are several extra defense mechanisms available for further protecting each kind of redundant component, and the defender selects the defense mechanisms for the same kind of redundant components from a same defense mechanism choice set.
- The probability that a redundant component operates properly is independent of whether extra defense mechanisms are deployed to it.
- A node is subject to attack only if a path exists from node *s* to that node, and all the intermediate nodes on the path have been compromised.
- The attacker's ultimate goal is making all the core nodes fail to provide any critical service with minimized attack cost; thus, the attacker will compromise just one redundant component, the primary one, in non-core nodes for penetrating, and compromise all redundant components in core nodes for whole dysfunction.
- A core node is compromised, i.e., totally dysfunctional, if and only if all redundant components allocated in it have been compromised.
- While attempting to compromise a non-core node, i.e., to penetrate it, the attacker can always arbitrarily choose the redundant component with most advantage for minimizing total attack cost to compromise.
- A non-core node is compromised, i.e., penetrated, if one of redundant components allocated in it has been compromised.
- A redundant component is subject to attack only if all extra defense mechanisms allocated to protect it have been compromised.
- If the attacker has compromised the extra defense mechanism *d* of redundant component *m* once, he/she then learned some effective skills or developed some powerful hacker tools to deal with this kind of defense mechanism *d* of redundant component *m*. Hence, the attacker can compromise the same kind of defense mechanism *d* of the same kind of redundant component *m* with a comparatively low cost afterward.
- According to the same reason mentioned above, the attacker can compromise any kind of redundant component which he/she has ever compromised with a comparatively low cost.

Given:

- The Core nodes
- The initial position of attacker
- The topology and size of the network
- The total defense budget
- The threshold of service continuity assurance that defines the minimum expected number of redundant components for every node
- The capacity limit of all nodes
- The predefined function of each node
- The probability of a non-core node providing a service function
- The redundant component choice set of each kind of function
- The defense mechanism choice set of each kind of redundant component
- The cost of each kind of redundant component
- The cost of each kind of extra defense mechanism available for each kind of redundant component
- The attack threshold of compromising each kind of redundant component and defense mechanism
- The probability of each kind of redundant component operating properly
- The ratio of the fixed part of attack cost for compromising each kind of redundant component and defense mechanism

Objective

- To maximize the minimized total attack cost

Subjected to

- The total cost spending on allocating redundant components and extra defense mechanisms must be no more than the limitation of total defense budget.
- The expected number of redundant components in each node must be no less than the threshold of service continuity assurance, but the exact number of redundant components in each node cannot be more than the capacity limit.
- The node to be attacked must be connected to the existing attack paths.

To determine

Defender:

- Allocate proper redundant components with defense mechanisms to nodes

Attacker:

- Compromise proper redundant components with defense mechanisms in nodes

Therefore, all above are formulated as the proposed RAP-EDM model that is a max min integer programming problem as follow.

Table 2-2 Given Parameters of the RAP-EDM Model

Given Parameters	
Notation	**Description**
B	The total defense budgetary limitation.
N	The index set of all nodes in the network
T	The index set of all core nodes in the network
U	The index set of all non-core nodes in the network
F	The index set of all functions provided by the nodes in the network
M_f	The index set of all redundant components which can be selected to provide the same main function f, where $f \in F$
W	The index set of all Origin-Destination (O-D) pairs, where the origin is node s and the destination is the other node i, where $s, i \in N$
P_w	The index set of all candidate paths of an O-D pair w, where $w \in W$
D_m	The index set of all extra defense mechanisms available for the kind of redundant component m, where $m \in M_f$, $f \in F$
α	The threshold of service continuity assurance that defines the minimum expected number of redundant components for every node
β	The capacity limit of redundant components for each node
σ_{if}	The indicator function, which is 1 if node i provides function f, and 0 otherwise (where $i \in N$, $f \in F$)
δ_{pi}	The indicator function, which is 1 if node i is on the path p, and 0 otherwise (where $i \in N$, $p \in P_w$, $w \in W$)
c_m	The cost of the kind of redundant component m, where $m \in M_f$, $f \in F$
$\hat{a}_m(c_m)$	The threshold of the attack cost required to compromise the kind of redundant component m, where $m \in M_f$, $f \in F$
λ_m	The consistent ratio that defines the fixed part of the attack cost for compromising the kind of redundant component m, where $m \in M_f$, $f \in F$
Q_m	The probability of the kind of redundant component m that operates properly, where $m \in M_f$, $f \in F$
c_{md}	The cost of the defense mechanism d of the kind of redundant component m, where $d \in D_m$, $m \in M_f$, $f \in F$
$\hat{a}_{md}(c_{md})$	The threshold of the attack cost required to compromise the defense

	mechanism *d* of the kind of redundant component *m*, where $d \in D_m$, $m \in M_f$, $f \in F$
λ_{md}	The consistent ratio that defines the fixed part of the attack cost for compromising the defense mechanism *d* of the kind of redundant component *m*, where $d \in D_m$, $m \in M_f$, $f \in F$

Table 2-3 Decision Variables of the RAP-EDM Model

Decision Variables	
Notation	**Description**
R_{im}	1 if the redundant component *m* is allocated in node *i*, and 0 otherwise (where $m \in M_f$, $f \in F$, $i \in N$)
R_{imd}	1 if the defense mechanism *d* of redundant component *m* is allocated in node *i*, and 0 otherwise (where $d \in D_m$, $m \in M_f$, $f \in F$, $i \in N$)
y_i	1 if node *i* is compromised, and 0 otherwise (where $i \in N$)
y_{im}	1 if the redundant component *m* in node *i* is compromised, and 0 otherwise (where $m \in M_f$, $f \in F$, $i \in N$)
y_{imd}	1 if the defense mechanism *d* of redundant component *m* in node *i* is compromised, and 0 otherwise (where $d \in D_m$, $m \in M_f$, $f \in F$, $i \in N$)
z_m	Times of the kind of redundant component *m* being compromised by the attacker (where $m \in M_f$, $f \in F$)
z_{md}	Times of the kind of defensive mechanism *d* of redundant component *m* being compromised by the attacker (where $d \in D_m$, $m \in M_f$, $f \in F$)
x_p	1 if path *p* is selected as the attack path, and 0 otherwise (where $p \in P_w$, $w \in W$)

Objective:

$$\max \min_{z_m, z_{md}} \sum_{f \in F} \sum_{m \in M_f} \left[\left\lceil \frac{z_m}{|N|} \right\rceil \hat{a}_m(c_m)\left[1+(z_m-1)\lambda_m\right] + \sum_{d \in D_m} \left\lceil \frac{z_{md}}{|N|} \right\rceil \hat{a}_{md}(c_{md})\left[1+(z_{md}-1)\lambda_{md}\right] \right] \quad \textbf{(IP1)}$$

Subjected to:

$$\sum_{w \in W} \sum_{p \in P_w} x_p \delta_{pi} \le \left(|N|-1\right) y_i \qquad \forall i \in N \qquad \text{(IP 1.1)}$$

$$\sum_{p \in P_w} x_p = y_i \qquad \forall i \in N, w \in W \qquad \text{(IP 1.2)}$$

$\sum_{p\in P_w} x_p \le 1$	$\forall w\in W$	(IP 1.3)
x_p = 0 or 1	$\forall p\in P_w, w\in W$	(IP 1.4)
y_i = 0 or 1	$\forall i\in U$	(IP 1.5)
$y_i = 1$	$\forall i\in T$	(IP 1.6)
$R_{imd} \le R_{im}$	$\forall i\in N, m\in M_f, f\in F, d\in D_m$	(IP 1.7)
$\sum_{m\in M_f} R_{im}Q_m \ge \alpha$	$\forall i\in N, f\in F$	(IP 1.8)
$\sum_{m\in M_f} R_{im} \le \beta$	$\forall i\in N, f\in F$	(IP 1.9)
R_{im} = 0 or 1	$\forall i\in N, m\in M_f, f\in F$	(IP 1.10)
R_{imd} = 0 or 1	$\forall i\in N, m\in M_f, f\in F, d\in D_m$	(IP 1.11)
$y_{im} \le R_{im}$	$\forall i\in N, m\in M_f, f\in F$	(IP 1.12)
$y_{imd} \le R_{imd}$	$\forall i\in N, m\in M_f, f\in F, d\in D_m$	(IP 1.13)
$y_i = \sum_{m\in M_f} y_{im}$	$\forall i\in U, f\in F$	(IP 1.14)
$y_{im}\sum_{d\in D_m} R_{imd} \le \sum_{d\in D_m} y_{imd}$	$\forall i\in U, m\in M_f, f\in F$	(IP 1.15)
$y_{im} = R_{im}$	$\forall i\in T, m\in M_f, f\in F$	(IP 1.16)
$y_{imd} = R_{imd}$	$\forall i\in T, m\in M_f, f\in F, d\in D_m$	(IP 1.17)
y_{im} = 0 or 1	$\forall i\in N, m\in M_f, f\in F$	(IP 1.18)
y_{imd} = 0 or 1	$\forall i\in N, m\in M_f, f\in F, d\in D_m$	(IP 1.19)
$z_m = \sum_{i\in N} y_{im}$	$\forall m\in M_f, f\in F$	(IP 1.20)
$z_{md} = \sum_{i\in N} y_{imd}$	$\forall m\in M_f, f\in F, d\in D_m$	(IP 1.21)
$\sum_{i\in N}\sum_{f\in F}\sigma_{if}\sum_{m\in M_f}\left[R_{im}c_m + \sum_{d\in D_m} R_{imd}c_{md}\right] \le B$		(IP 1.22)

Explanation of RAP-EDM **model:**

- **Objective function:** The objective is to maximize the minimized total attack cost for compromising all core nodes in the network. In the inner problem, the attacker attempts to minimize the total attack cost by deciding which redundant components in which nodes to compromise. Because of the allocation of extra defense mechanisms, the attacker must also bring the attack cost of compromising these defense mechanisms of redundant components into consideration. It is worthy to mention that $\left\lceil \frac{z_m}{|N|} \right\rceil \left[1+\left(z_m-1\right)\lambda_m\right]$ and $\left\lceil \frac{z_{md}}{|N|} \right\rceil \left[1+\left(z_{md}-1\right)\lambda_{md}\right]$ stand for the impact of the attacker's experience accumulation on redundant components and extra defense mechanisms, respectively. When attacking a kind of redundant component or defense mechanism which is new to the attacker, he/she has to pay the full attack cost which equals to the threshold of it, and he/she just needs to spend a fixed small portion of original attack cost while compromising the same kind of redundant component or defense mechanism afterwards. In the outer problem, the defender tries to make best use of limited defense resource to allocate suitable redundant components (with extra defense mechanisms) for maximizing the minimized total attack cost, and also regard for the service availability assurance and capacity limit of every node at the meantime.
- **Constraint (IP 1.1)** prevents the attack paths forming loops.
- **Constraint (IP 1.2)** and **Constraint (IP 1.3)** enforces that if the attacker tries to compromise a node, there must be one attack path to that node.
- **Constraint (IP 1.4)** and **Constraint (IP 1.5)** respectively restrict the value of x_p and y_i to be 0 or 1.
- **Constraint (IP 1.1)** ~ **Constraint (IP 1.5)** jointly compose the "continuity constraints".
- **Constraint (IP 1.6)** enforces that all core nodes must be compromised.

- **Constraint (IP 1.7)** enforces that the defender must allocate a redundant component before deploying any available defense mechanisms to it.
- **Constraint (IP 1.8)** restricts that the expected number of redundant components in each node should be no less than the threshold of service availability assurance.
- **Constraint (IP 1.9)** restricts that the number of redundant components in each node should be no more than the capacity limit of redundant components.
- **Constraint (IP 1.10)** and **Constraint (IP 1.11)** respectively restrict the value of R_{im} and R_{imd} to be 0 or 1.
- **Constraint (IP 1.12)** and **Constraint (IP 1.13)** enforces that the attacker cannot compromise any nonexistent defense mechanisms or redundant components in any nodes.
- **Constraint (IP 1.14)** restricts that a non-core node is compromised, i.e., penetrated, if one of redundant components allocated in it has been compromised.
- **Constraint (IP 1.15)** restricts that the attacker can attempt to compromise a redundant component if and only if all extra defense mechanisms deployed for protecting it have been compromised.
- **Constraint (IP 1.16)** and **Constraint (IP 1.17)** restrict that a core node is compromised, i.e., totally dysfunctional, if and only if all redundant components with defense mechanisms allocated in it have been compromised.
- **Constraint (IP 1.18)** and **Constraint (IP 1.19)** respectively restrict the value of y_{im} and y_{imd} to be 0 or 1.
- **Constraint (IP 1.20)** and **Constraint (IP 1.21)** respectively restrict the value of z_m and z_{md} to be the total times of compromising each kind of redundant components and defense mechanisms.
- **Constraint (IP 1.22)** restricts that the total defense spending should be no more than the total defense budgetary limitation.

2.3 Problem Formulation of the AEA Model

Since the result of the max min problem we face in RAP-EDM model dynamically changes caused by the competition between the defender and the attacker, it is too difficult to solve such a two-leveled problem immediately. Instead, we adopt an alternative two-phase approach to cope with it. At first, we abstract the inner problem of RAP-EDM model as a maximization integer programming problem, AEA model, and deal with it to get the best attack strategy. Afterward we treat the solution of AEA model as the input of RAP-EDM model, and then solve it to develop the defense plan about how to allocate redundant components and defense mechanisms in each node.

In AEA model, it is worthy to mention that the allocation of redundant components and defense mechanisms become given parameters, but the else parts of AEA model are just similar with RAP-EDM model. Thus, we directly introduce the AEA model below:

Table 2-4 Given Parameters of the AEA Model

Given Parameters	
Notation	**Description**
B	The total defense budgetary limitation
N	The index set of all nodes in the network
T	The index set of all core nodes in the network
U	The index set of all non-core nodes in the network
F	The index set of all functions provided by the nodes in the network
M_f	The index set of all redundant components which can be selected to provide the same main function f, where $f \in F$
W	The index set of all Origin-Destination (O-D) pairs, where the origin is node s and the destination is the other node i, where $s, i \in N$
P_w	The index set of all candidate paths of an O-D pair w, where $w \in W$
D_m	The index set of all extra defense mechanisms available for the kind of redundant component m, where $m \in M_f$, $f \in F$
σ_{if}	The indicator function, which is 1 if node i provides function f, and 0 otherwise (where $i \in N$, $f \in F$)

δ_{pi}	The indicator function, which is 1 if node i is on the path p, and 0 otherwise (where $i \in N$, $p \in P_w$, $w \in W$)
c_m	The cost of the kind of redundant component m, where $m \in M_f$, $f \in F$
$\hat{a}_m(c_m)$	The threshold of the attack cost required to compromise the kind of redundant component m, where $m \in M_f$, $f \in F$
λ_m	The consistent ratio that defines the fixed part of the attack cost for compromising the kind of redundant component m, where $m \in M_f$, $f \in F$
c_{md}	The cost of the defense mechanism d of the kind of redundant component m, where $d \in D_m$, $m \in M_f$, $f \in F$
$\hat{a}_{md}(c_{md})$	The threshold of the attack cost required to compromise the defense mechanism d of the kind of redundant component m, where $d \in D_m$, $m \in M_f$, $f \in F$
λ_{md}	The consistent ratio that defines the fixed part of the attack cost for compromising the defense mechanism d of the kind of redundant component m, where $d \in D_m$, $m \in M_f$, $f \in F$
R_{im}	1 if the redundant component m is allocated in node i, and 0 otherwise (where $m \in M_f$, $f \in F$, $i \in N$)
R_{imd}	1 if the defense mechanism d of redundant component m is allocated in node i, and 0 otherwise (where $d \in D_m$, $m \in M_f$, $f \in F$, $i \in N$)

Table 2-5 Decision Variables of the AEA Model

Decision Variables	
Notation	**Description**
y_i	1 if node i is compromised, and 0 otherwise (where $i \in N$)
y_{im}	1 if the redundant component m in node i is compromised, and 0 otherwise (where $m \in M_f$, $f \in F$, $i \in N$)
y_{imd}	1 if the defense mechanism d of redundant component m in node i is compromised, and 0 otherwise (where $d \in D_m$, $m \in M_f$, $f \in F$, $i \in N$)
z_m	Times of the kind of redundant component m being compromised by the attacker (where $m \in M_f$, $f \in F$)
z_{md}	Times of the kind of defensive mechanism d of the kind of redundant component m being compromised by the attacker (where $d \in D_m$, $m \in M_f$, $f \in F$)
x_p	1 if path p is selected as the attack path, and 0 otherwise (where $p \in P_w$, $w \in W$)

Objective:

$$\min_{z_m, z_{md}} \sum_{f \in F} \sum_{m \in M_f} \left[\left\lceil \frac{z_m}{|N|} \right\rceil \hat{a}_m(c_m)\left[1+(z_m-1)\lambda_m\right] + \sum_{d \in D_m} \left\lceil \frac{z_{md}}{|N|} \right\rceil \hat{a}_{md}(c_{md})\left[1+(z_{md}-1)\lambda_{md}\right] \right] \quad \textbf{(IP 2)}$$

Subjected to:

$\sum_{w \in W} \sum_{p \in P_w} x_p \delta_{pi} \le (\lvert N \rvert - 1) y_i$	$\forall i \in N$	(IP 2.1)
$\sum_{p \in P_w} x_p = y_i$	$\forall i \in N, w \in W$	(IP 2.2)
$\sum_{p \in P_w} x_p \le 1$	$\forall w \in W$	(IP 2.3)
x_p = 0 or 1	$\forall p \in P_w, w \in W$	(IP 2.4)
y_i = 0 or 1	$\forall i \in U$	(IP 2.5)
y_i = 1	$\forall i \in T$	(IP 2.6)
$y_{im} \le R_{im}$	$\forall i \in N, m \in M_f, f \in F$	(IP 2.7)
$y_{imd} \le R_{imd}$	$\forall i \in N, m \in M_f, f \in F, d \in D_m$	(IP 2.8)
$y_i = \sum_{m \in M_f} y_{im}$	$\forall i \in U, f \in F$	(IP 2.9)
$y_{im} \sum_{d \in D_m} R_{imd} \le \sum_{d \in D_m} y_{imd}$	$\forall i \in U, m \in M_f, f \in F$	(IP 2.10)
$y_{im} = R_{im}$	$\forall i \in T, m \in M_f, f \in F$	(IP 2.11)
$y_{imd} = R_{imd}$	$\forall i \in T, m \in M_f, f \in F, d \in D_m$	(IP 2.12)
y_{im} = 0 or 1	$\forall i \in N, m \in M_f, f \in F$	(IP 2.13)
y_{imd} = 0 or 1	$\forall i \in N, m \in M_f, f \in F, d \in D_m$	(IP 2.14)
$z_m = \sum_{i \in N} y_{im}$	$\forall m \in M_f, f \in F$	(IP 2.15)
$z_{md} = \sum_{i \in N} y_{imd}$	$\forall m \in M_f, f \in F, d \in D_m$	(IP 2.16)

Explanation of AEA model:

- **Objective function:** The objective is to minimize the total attack cost for compromising all core nodes in the network. In the AEA model, the attacker tries to minimize the total attack cost by deciding which redundant components and extra defense mechanisms in which nodes to compromise. By the way, $\left\lceil \frac{z_m}{|N|} \right\rceil \left[1+\left(z_m-1\right)\lambda_m\right]$ and $\left\lceil \frac{z_{md}}{|N|} \right\rceil \left[1+\left(z_{md}-1\right)\lambda_{md}\right]$ represent the attacker's experience accumulation and still make impact on attack cost.
- **Constraint (IP 2.1) ~ Constraint (IP 2.6)** are identical to **Constraint (IP 1.1)** ~ **Constraint (IP 1.6)** of the RAP-EDM model.
- **Constraint (IP 2.7)** ~ **Constraint (IP 2.16)** equal to **Constraint (IP 1.12)** ~ **Constraint (IP 1.21) of** the RAP-EDM model, which restrict the attacker's behavior.

Chapter 3 Solution Approach

3.1 Solution Approach for the AEA Model

3.1.1 Lagrangean Relaxation Method

We adopt a powerful tool that is Lagrangean relaxation (LR) method to deal with the complicated problems addressed in this work. One of the key concepts of LR is “decomposition” that divides the primal problem into some easier subproblems, and which can be solved independently. This really helps reduce computational complexity and facilitates the process of getting solutions, especially while coping with large-scale mathematical programming problems.

Those complex and difficult problems are usually composed of a relatively simpler mathematical programming problem and a set of side constraints which are difficult to handle, and LR provides us with an efficient and effective approach, “relaxation”, to deal with these kinds of complicated problems. Thus, we can remove the limitations caused by the set of relatively troublesome side constraints; instead, we take them into the objective function of the primal problem, IP, with corresponding Lagrangean multipliers, μ. Afterward a Lagrangean relaxation problem, LR$_{\mu}$, is then constructed, and we can further make use of decomposition to separate this LR problem into several easily solvable subproblems, which are independent to each other.

Besides, LR also facilitates effective obtaining the boundary of objective in the primal problem. While dealing with a minimization problem, the objective value, $Z_D(\mu)$, of the LR problem is always a lower bound (LB) of the primal problem’s optimal solution [21], although this solution of LR problem may be infeasible for the primal problem. Based on this, we will attempt to acquire the tightest LB, i.e., let $Z_D(\mu)$ as large as possible, by continuously tuning the Lagrangean multiplier. The process of unceasingly tuning the Lagrangean multiplier is known as Lagrangean dual problem; then, we adopt one of the most popular approaches to it. That is the subgradient method.

While the LR problem is solved, we then turn to consider whether the solution is feasible to the primal problem. If the solution meets all requirements of primal constraints, a satisfying optimal solution is acquired; otherwise, we have to further develop some appropriate heuristics to make the infeasible solution become a feasible one, according to some hints from the earlier processes.

By the way, each feasible solution of the primal problem also provides an upper bound (UB) of the optimal value to it; therefore, the actually optimal solution of the primal problem can be guaranteed within the range between the UB and LB obtained before.

The Lagrangean relaxation method is illustrated in Figure 3-1, and the solution process of AEA model based on this approach is then presented in section 3.1.2.

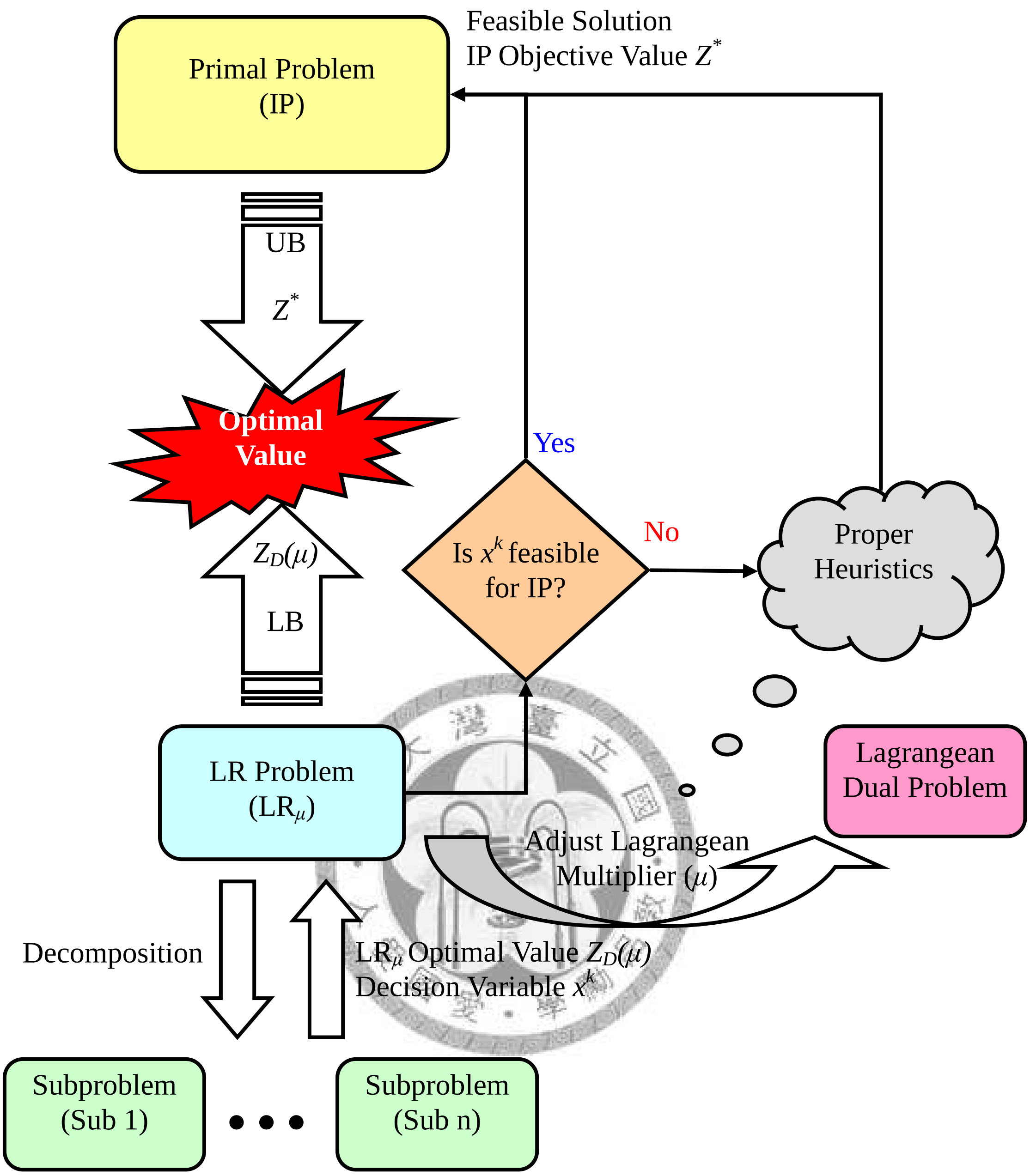

Figure 3-1 Process of Lagrangean Relaxation Method

3.1.2 Lagrangean Problem of the AEA Model

Making use of Lagrangean relaxation method, we turn the primal problem (IP 2) into the Lagrangean relaxation problem (LR 1) by relaxing the constraints (IP 2.1), (IP 2.2), (IP 2.9), and (IP 2.10). With a vector of Lagrangean multipliers, the Lagrangean relaxation problem can be shown as follows.

Optimization problem:

$$
\begin{aligned}
&Z_D(\mu_1,\mu_2,\mu_3,\mu_4)= \\
&\min_{z_m,z_{md}} \sum_{f\in F}\sum_{m\in M_f}\left[\left\lceil \frac{z_m}{|N|}\right\rceil \hat{a}_m(c_m)\left[1+(z_m-1)\lambda_m\right]+\sum_{d\in D_m}\left\lceil \frac{z_{md}}{|N|}\right\rceil \hat{a}_{md}(c_{md})\left[1+(z_{md}-1)\lambda_{md}\right]\right] \\
&+\sum_{i\in N}\mu_i^1\left[\sum_{w\in W}\sum_{p\in P_w} x_p\delta_{pi}-(|N|-1)\,y_i\right]+\sum_{i\in N}\mu_i^2\left[\sum_{p\in P_{(s,i)}} x_p-y_i\right] \\
&+\sum_{i\in U}\sum_{f\in F}\sigma_{if}\mu_{if}^3\left[y_i-\sum_{m\in M_f} y_{im}\right]+\sum_{i\in U}\sum_{f\in F}\sigma_{if}\sum_{m\in M_f}\mu_{ifm}^4\left[y_{im}\sum_{d\in D_m} R_{imd}-\sum_{d\in D_m} y_{imd}\right]
\end{aligned}
\qquad \textbf{(LR 1)}
$$

Subjected to:

$$\sum_{p\in P_w} x_p \le 1 \qquad \forall w\in W \qquad \text{(LR 1.1)}$$

$$x_p = 0 \text{ or } 1 \qquad \forall p\in P_w, w\in W \qquad \text{(LR 1.2)}$$

$$y_i = 0 \text{ or } 1 \qquad \forall i\in U \qquad \text{(LR 1.3)}$$

$$y_i = 1 \qquad \forall i\in T \qquad \text{(LR 1.4)}$$

$$y_{im} \le R_{im} \qquad \forall i\in N, m\in M_f, f\in F \qquad \text{(LR 1.5)}$$

$$y_{imd} \le R_{imd} \qquad \forall i\in N, m\in M_f, f\in F, d\in D_m \qquad \text{(LR 1.6)}$$

$$y_{im} = R_{im} \qquad \forall i\in T, m\in M_f, f\in F \qquad \text{(LR 1.7)}$$

$$y_{imd} = R_{imd} \qquad \forall i\in T, m\in M_f, f\in F, d\in D_m \qquad \text{(LR 1.8)}$$

$$y_{im} = 0 \text{ or } 1 \qquad \forall i\in N, m\in M_f, f\in F \qquad \text{(LR 1.9)}$$

$$y_{imd} = 0 \text{ or } 1 \qquad \forall i\in N, m\in M_f, f\in F, d\in D_m \qquad \text{(LR 1.10)}$$

$$z_m = \sum_{i\in N} y_{im} \qquad \forall m\in M_f, f\in F \qquad \text{(LR 1.11)}$$

$$z_{md} = \sum_{i\in N} y_{imd} \qquad \forall m\in M_f, f\in F, d\in D_m \qquad \text{(LR 1.12)}$$

The Lagrangean multipliers, μ_1, μ_2, μ_3, and μ_4, are the vectors of $\{\mu_i^1\}$, $\{\mu_i^2\}$, $\{\mu_{if}^3\}$, and $\{\mu_{ifm}^4\}$, respectively, where μ_2 and μ_3 are non-restricted, andμ_1 and μ_4 are both non negative. Afterward we decompose (LR 1) into four easily solvable independent subproblems as shown in the following pages.

Subproblem 1.1 (related to decision variable x_p)

$$Z_{Sub1.1}(\mu_1, \mu_2) = \min \sum_{i\in N} \sum_{w\in W} \sum_{p\in P_w} \mu_i^1 x_p \delta_{pi} + \sum_{i\in N} \sum_{p\in P_{(s,i)}} \mu_i^2 x_p \qquad \textbf{(Sub 1.1)}$$

Subject to:

$$\sum_{p\in P_w} x_p \le 1 \qquad \forall w \in W \qquad \text{(Sub 1.1.1)}$$

$$x_p = 0 \text{ or } 1 \qquad \forall p \in P_w, w \in W \qquad \text{(Sub 1.1.2)}$$

We decompose (Sub 1.1) into |W| independent subproblems. Since μ_1 as the associated node weights in subproblem 1.1 are non-negative, the first part of each subproblem can be efficiently resolved by Dijkstra shortest path algorithm. Afterward we add the second part, which is related to μ_2, into the result from the first part. When the destination of an O-D pair is a core node, we just add the result of this subproblem without doubt. On the other hand, when the destination of an O-D pair is a non-core node, we add the result of this subproblem only if the result is negative. Accordingly, the complexity of subproblem 1.1 is just as the one of Dijkstra's algorithm, O($|N|^2$), where |N| is the number of nodes.

Subproblem 1.2 (related to decision variable y_i)

$$Z_{sub1.2}(\mu_1, \mu_2, \mu_3) = \min - \sum_{i\in N} \mu_i^1 \left(|N| - 1\right) y_i - \sum_{i\in N} \mu_i^2 y_i + \sum_{i\in U} \sum_{f\in F} \sigma_{if} \mu_{if}^3 y_i \qquad \textbf{(Sub 1.2)}$$

Subject to:

$$y_i = 0 \text{ or } 1 \qquad \forall i \in U \qquad \text{(Sub 1.2.1)}$$

$$y_i = 1 \qquad \forall i \in T \qquad \text{(Sub 1.2.2)}$$

In subproblem 1.2, there are just two possibilities of decision variable y_i for every non-core node in the network. Therefore, we can use exhausted search to find out y_i should be 0 or 1 that makes the objective value minimized, for each non-core node *i*. The complexity of subproblem 1.2 is O(|N|), where |N| is the number of nodes.

Subproblem 1.3 (related to decision variable *y_{im}*, *z_m*)

$$Z_{sub1.3}(\mu_3,\mu_4)=$$

$$\min\sum_{f\in F}\sum_{m\in M_f}\left\lceil\frac{z_m}{|N|}\right\rceil\hat{a}_m\left(c_m\right)\left[1+\left(z_m-1\right)\lambda_m\right] \qquad \textbf{(Sub 1.3)}$$

$$-\sum_{i\in U}\sum_{f\in F}\sigma_{if}\sum_{m\in M_f}\mu_{if}^{3}y_{im}+\sum_{i\in U}\sum_{f\in F}\sigma_{if}\sum_{m\in M_f}\mu_{ifm}^{4}y_{im}\sum_{d\in D_m}R_{imd}$$

Subject to:

$$y_{im}\le R_{im} \qquad \forall i\in N, m\in M_f, f\in F \qquad \text{(Sub 1.3.1)}$$

$$y_{im}=R_{im} \qquad \forall i\in T, m\in M_f, f\in F \qquad \text{(Sub 1.3.2)}$$

$$y_{im} = 0 \text{ or } 1 \qquad \forall i\in N, m\in M_f, f\in F \qquad \text{(Sub 1.3.3)}$$

$$z_m=\sum_{i\in N}y_{im} \qquad \forall m\in M_f, f\in F \qquad \text{(Sub 1.3.4)}$$

In subproblem 1.3, we first let y_{im}s of core nodes equal to their R_{im}s, and calculate the corresponding Z_ms. Then, there are only two choices for every decision variable y_{im}, which's R_{im} equals to 1, in each non-core node. If a y_{im}'s corresponding Z_m equals to 1, we just determine whether the absolute value of last two parts of this y_{im}, which are related to μ_3 and μ_4, is larger than $\hat{a}_m\left(c_m\right)\lambda$ or not, where $\hat{a}_m\left(c_m\right)$ is the attack cost of *m* and λ is the consistent ratio of redundant component attack cost caused by attack experience accumulation. If the absolute value of their sum is bigger, we let this y_{im} be 1 and calculate its corresponding Z_m; otherwise, we just set this y_{im} to 0. If its corresponding Z_m equals to 0, we mark those y_{im}s that lead to negative subresults of its last two parts and sum up their subresults. If the absolute value of such sum is larger than $\hat{a}_m\left(c_m\right)\left[1+\left(x-1\right)\lambda\right]$ where *x* is the number of those y_{im}s that lead to negative subresults, we set all those marked y_{im}s to 1 and calculate the corresponding Z_m; otherwise, we just set those marked y_{im}s to 0. The complexity of **(Sub 1.3)** is O(|N|x|M|), where |N| is the number of nodes, and |M| is the number of redundant component types.

Subproblem 1.4 (related to decision variable y_{imd}, z_{md})

$$Z_{sub1.4}(\mu_4) =$$

$$\min \sum_{f\in F}\sum_{m\in M_f}\sum_{d\in D_m}\left\lceil \frac{z_{md}}{|N|}\right\rceil \hat{a}_{md}\left(c_{md}\right)\left[1+\left(z_{md}-1\right)\lambda_{md}\right] - \sum_{i\in U}\sum_{f\in F}\sigma_{if}\sum_{m\in M_f}\sum_{d\in D_m}\mu^4_{ifm}y_{imd} \qquad \textbf{(Sub 1.4)}$$

Subject to:

$$y_{imd} \le R_{imd} \qquad \forall i\in N, m\in M_f, f\in F, d\in D_m \qquad \text{(Sub 1.4.1)}$$

$$y_{imd} = R_{imd} \qquad \forall i\in T, m\in M_f, f\in F, d\in D_m \qquad \text{(Sub 1.4.2)}$$

$$y_{imd} = 0 \text{ or } 1 \qquad \forall i\in N, m\in M_f, f\in F, d\in D_m \qquad \text{(Sub 1.4.3)}$$

$$z_{md} = \sum_{i\in N} y_{imd} \qquad \forall m\in M_f, f\in F, d\in D_m \qquad \text{(Sub 1.4.4)}$$

In subproblem 1.4, we first let y_{imd}s of core nodes equal to their R_{imd}s, and calculate the corresponding Z_{md}s. Then, there are only two choices for every decision variable y_{imd}, which's R_{imd} equals to 1, in each non-core node. If a y_{imd}'s corresponding Z_{md} equals to 1, we just determine whether the absolute value of the part related to μ_4 of this y_{imd} is larger than $\hat{a}_{md}\left(c_{md}\right)\lambda$ or not, where $\hat{a}_{md}\left(c_{md}\right)$ is the attack cost of d and λ is the consistent ratio of defense mechanism attack cost caused by attack experience accumulation. If the absolute value is larger, we let this y_{imd} be 1 and calculate its corresponding Z_{md}; otherwise, we just set this y_{imd} to 0. If its corresponding Z_{md} equals to 0, we mark those y_{imd}s that lead to negative last parts and sum up their subresults. If the absolute value of such sum is larger than $\hat{a}_{md}\left(c_{md}\right)\left[1+\left(x-1\right)\lambda\right]$ where x is the number of those y_{imd}s that lead to negative subresults, we set all those marked y_{imd}s to 1 and calculate the corresponding Z_{md}; otherwise, we just set those marked y_{imd}s to 0. The complexity of subproblem 1.4 is O(|N|x|D|), where |N| is the number of nodes, and |D| is the number of all defense mechanism types.

3.1.3 The Dual Problem and the Subgradient Method

The Lagrangean Relaxation problem (LR 1) can be solved by optimally solving the above subproblems (Sub 1.1~ Sub 1.4), respectively. As mentioned in section 3.1.3, every solution of Z_D yields a lower bound on Z_{IP} in minimization problems. Therefore, we construct a dual problem and solve it by the subgradient method to acquire a LB which is as tight as possible.

Dual Problem

$Z_D = \max Z_D(\mu_1, \mu_2, \mu_3, \mu_4)$ **(D1)**

Subject to: $\mu_1, \mu_4 \geq 0$

Use a vector *m* as a subgradient of $Z_D(\mu_1, \mu_2, \mu_3, \mu_4)$, and the multiplier vector $\mu^k = (\mu_1^k, \mu_2^k, \mu_3^k, \mu_4^k)$ is updated by $\mu^{k+1} = \mu^k + t^k m^k$ in iteration *k* of the subgradient process, where

$$m^k(\mu_1^k, \mu_2^k, \mu_3^k, \mu_4^k) = (\sum_{w \in W} \sum_{p \in P_w} x_p \delta_{pi} - (|N| - 1) y_i, \sum_{p \in P_{(s,i)}} x_p - y_i, y_i - \sum_{m \in M_f} y_{im}, y_{im} \sum_{d \in D_m} R_{imd} - \sum_{d \in D_m} y_{imd})$$

The step size, *t*^{k}, is derived from $t^k = \lambda \frac{Z_{IP2}^* - Z_D(\mu^k)}{\|m^k\|^2}$, where λ is a scalar from 0 to 2. The initial value of λ is usually set as 2, and we halve it while the best value of objective function is not improved within a limited number of iterations. Furthermore, Z_{IP2}^* represents the best upper bound of primal objective function value that we have obtained by iteration *k*.

3.1.4 Getting Primal Feasible Solutions

The procedure of Lagrangean Relaxation not only provides us a lower bound for the primal objective function value, but also gives us some hints to develop proper heuristics for getting primal feasible solutions. Thus, we make use of some LR multipliers and decision variables to develop a LR-Based heuristic attack algorithm to deal with the attack problem discussed in AEA model.

First, the attacker must decide which paths to take as attack paths for compromising all core nodes, and we take advantage of the result of subproblem 1.1 **(Sub 1.1)**, which is related to attack path choice, to create the attack paths. Instead of directly using the result of subproblem 1.1 as the attack paths, we redefine the path cost of each node and run Dijkstra shortest path algorithm to construct the attack paths to all core nodes. We calculate the lowest attack cost of compromising each non-core node, i.e., the attack cost of compromising the weakest redundant component, and divide it by 1,000. This result is added to the μ_1, thus the path cost of each node is decided. Moreover, we induce the attacker to take those compromised nodes as hop sites when choosing paths to achieve the remained core nodes. The way of doing above idea is to let the path cost of each chosen node be zero to replace original node path cost after deciding any attack path, and then rerun Dijkstra shortest path algorithm for achieving the next core node. This procedure will be executed until reaching all core nodes from starting node *s*.

According to the descriptions of our attack scenario, the attacker must compromise all core nodes by compromising all redundant components with defense mechanisms allocated to them. Thus, we compromise all redundant components with defense mechanisms in core nodes; meanwhile, we have to calculate the corresponding Z_ms and Z_{md}s.

The last step of this LR-Based attack algorithm is to determine which redundant component with defense mechanisms to compromise in those non-core nodes which are on the attack tree, so we need to find out the attack cost of each redundant component with defense mechanisms for each non-core node which is on the attack tree. When considering the cost of compromising these non-core nodes, the impact of attack experience accumulation is also our concern. If there is a kind of redundant component which has been compromised before, i.e., its corresponding Z_m is already set to non zero, its attack cost will be only seen as the fixed part. Of course, this rule is also applied to defense mechanisms. At last, we choose the redundant component with defense mechanisms that totally costs least within a non-core node to compromise for penetrating each non-core node on the attack tree. During the attack processes in these non-core nodes, the corresponding Z_ms and Z_{md}s also need to be updated continuously.

3.2 Solution Approach for the RAP-EDM Model

The LR-Based heuristic attack algorithm proposed in section 3.1.4 makes the attacker achieve the ultimate goal in an efficient way, and it also provides the defender some information about the intelligent attacker’s behavior in the meantime. As the references of redundancy and defense allocation, the information really helps the defender a lot. Therefore, we design a heuristic allocation algorithm which is highly related to the LR-Based attack algorithm. We call it as LR-Based allocation algorithm and introduce it below.

The LR-Based allocation algorithm consists of two main parts, initial allocation and allocation adjustment. In the process of initial allocation, the first thing we need to deal with is satisfying the service continuity requirement, so we first allocate redundant components to all nodes following a predefined order. The core nodes have the first priority to get the types of redundant components with higher costs, so we allocate as many different kinds of more expensive redundant components to core nodes as possible considering the capacity and budget constraints. We then allocate unused types of redundant components to those non-core nodes which are 1-Hop away from starting node *s* or 1-Hop away from core nodes, and we allocate redundant components to the remained non-core nodes at last. Unlike allocating redundant components to core nodes, we allocate the required number of a same kind of redundant components, rather than different kinds of redundant components, to non-core nodes. The reason is that we give the attacker no chance to choose a more vulnerable redundant component within a non-core node to compromise for penetrating purpose. Moreover, the times of allocating each kind of redundant component should be as equivalent as possible when allocating redundant components to non-core nodes.

After satisfying the requirement of service continuity, we allocate as many different kinds of defense mechanisms to protect the redundant components in core nodes if the remained budget is abundant. Then we execute the LR-Based attack algorithm and record how many times each non-core node has been really compromised within 2,000 iterations. Those non-core nodes with higher records will be allocated with defense mechanisms first, so they will have stronger probability to get more expensive defense mechanisms. Because the attacker can arbitrarily choose the most vulnerable redundant component with defense mechanisms to attack for compromising a non-core node, we must allocate the same kind of defense mechanism to every redundant component within a non-core node if we decide to allocate any kind of defense mechanism to a redundant component in such node. For protecting each non-core node as fairly as possible, the allocation of defense mechanisms is in rotation following the priority made by the above record, i.e., a non-core node get allocated a kind of defense mechanism each time. The allocation of defense mechanisms will be executed until running out of budget, and the first part of the LR-Based allocation algorithm, initial allocation, is finished.

After finishing the initial allocation of this network, we rerun the LR-Based attack algorithm and still record how many times each non-core node has been actually compromised within 2,000 iterations. If the total allocation cost of redundant components and defense mechanisms in a non-core node with a lower record is higher than the total cost of another same functioned non-core node with a higher record, we exchange whole their allocations. This kind of adjustment will continuously executed until we let those non-core nodes that are compromised more frequently be sure to get the more expensive allocation of redundant components and defense mechanisms, and a round of adjustment is finished.

The process of allocation adjustment is comparatively simple, and it is just composed of many rounds of adjustment. We keep repeating the procedure described in last paragraph until the total number of rounds reaches the limit number of execution, and we then take the allocation that causes the highest total attack cost compared with all the other results of each round as the final allocation decision.

Chapter 4 Computational Experiments

4.1 Computational Experiments with the AEA Model

To prove how effective our algorithms proposed respectively for the AEA model and the RAP-EDM model actually are, we additionally provide two simple attack algorithms and a randomized allocation algorithm as the comparisons with our LR-Based algorithms.

4.1.1 Simple Attack Algorithm 1 and Simple Attack Algorithm 2

We adopt Dijkstra shortest path algorithm as the method of choosing attack paths reaching the core nodes in both simple attack algorithm 1 and simple attack algorithm 2, but there is a main difference between these algorithms that is the path cost of each node. We take the physical distance, i.e., the hop count to other nodes, as the path cost of each node in simple attack algorithm 1.

In simple attack algorithm 2, we calculate the attack cost of compromising each redundant component with defense mechanisms in each non-core node and let the minimum attack cost of a redundant component within a non-core node be its path cost. On the other hand, we let the path costs of all core nodes be 0.

After deciding the attack paths, the fallowing steps in these two simple attack algorithms are just the same. We compromise all core nodes by compromising all redundant components with defense mechanisms allocated in them and update the records of corresponding Z_m and Z_{md}.

The last process of two simple algorithms is compromising the cheapest redundant component with defense mechanisms in each non-core node that was chosen in the first step. Of course, the impact of attack experience accumulation must be under consideration when choosing the most appropriate redundant component with defense mechanisms to compromise, and the records of Z_m and Z_{md} also have to be kept updating continually in this procedure.

4.1.2 Randomized Allocation Algorithm

Because RAP is seldom discussed under a network attack/defense scenario, there is no suitable well-known question to evaluate the quality of our LR-Based allocation algorithm. We implement a randomized allocation algorithm that fulfills the service continuity requirement, budget constraint, and capacity constraint, to compare with our proposed allocation algorithm.

We first randomly allocate the required number of redundant components to each node for complying with the service continuity requirement, and we then each time allocate a randomly chosen defense mechanism to a redundant component in a node if there is still remained budget after satisfying the service continuity requirement. This randomized allocation process will be terminated when the budget runs out.

4.1.3 Core Focused Allocation Algorithm

Under the defense/attack scenario we discussed, it is obvious that the defender should allocate as more different kinds of redundant components and defense mechanisms as possible to enhance the total attack cost of compromising the whole network. Therefore, we propose another allocation algorithm that first focuses on allocating redundant components and defense mechanisms to the core nodes well. After taking care of the core nodes, we then randomly allocate redundant components and defense mechanisms to each non-core node. Of course, we must make this allocation comply with the service continuity requirement, the capacity constraint, and the defense budget limit.

4.1.4 Experiment Environment

We implement the AEA model on a notebook, HP 2140 which is equipped with Intel Atom^{TM} N270 1.60GHz CPU and 2GB RAM. The OS of this experiment notebook is Windows XP Home Edition SP3, and the coding environment is Dev-C++ Version 4.9.9.2.

We compare our proposed LR-Based attack algorithm with two simple attack algorithms under two different allocations in a grid network topology because of physical defense in depth provided by this kind of topology. The different functions provided by every node are randomly decided, and each non-core node has 50% probability to provide only transmission function, rather than those certain service functions provided by the core nodes. There are six predefined core nodes in the target network, and their positions are consistent. The service continuity requirement forces the expected value number of redundant components in each node not to be smaller than 2, and the maximal number of redundant components exactly allocated in each node is restricted by the capacity limit which equals to 5.

The prices of different kinds of redundant components are between 50 and 100, and their reliabilities, i.e., the probabilities that these redundant components operate properly considering natural disasters random errors, are between 85% and 99%. The prices of different kinds of defense mechanisms are between 1 and 20. We define the fixed part of attack cost for compromising different redundant components and defense mechanisms as the attack threshold multiplied by a random ratio which is between 1% and 30%.

To prove our LR-Based attack algorithm is generally suitable for use, the experiments are implemented under fifteen different parameter settings, and which are composed of the different values provided by five main parameters: Number of Nodes, Defense Budget, Number of Functions, Size of Redundant Component and Defense Mechanism Choice Sets, and the relationship between the attack costs and the prices of all redundant components and defense mechanisms.

It is most worthy to mention that we set three different kinds of relationships between the attack costs and the prices of all redundant components and defense mechanisms, and which are linear, convex, and concave. In order to bring some variations into the attack costs, we randomly add or minus a little portion of the prices in addition.

At last, we provide the detail settings of all experiment parameters mentioned above together with some important parameters of Lagrangean Relaxation in the following table, Table 4-1.

Table 4-1 Experiment Parameter Settings of the AEA Model

Experiment Platform	CPU: Intel Atom ^{TM} N270 1.60GHz RAM: 2GB OS: Windows XP Home Edition SP3
Programming Environment	Dev-C++ Version 4.9.9.2
Parameters of LR	
Parameters	Value
Limit of Iteration Numbers	2000
Limit of Improve Count	80
Initial Value of LB	0
Initial Value of Multipliers	μ_1, μ_2, μ_3, $\mu_4 = 0$
Initial Scalar of Step Size	2
Parameters of the AEA Model	
Parameters	Value
Testing Topology	Grid Networks
Number of Core Nodes	6
Service Continuity Requirement	2
Capacity Limit of Redundant Components	5
Probability of the Non-Core Nodes	50%

Providing Service Functions	
Number of Nodes	49, 100, 400
Defense Budget	30000, 50000, 100000
Number of Functions	3+1, 5+1, 10+1(add a transmission function)
Redundant Component and Defense Mechanism Choice Set Size	10, 15, 20
Costs of Redundant Components	50 ~ 100
Probabilities of redundant components operating properly	85% ~ 99%
Costs of Defense Mechanisms	1 ~ 20
Defense Capability Functions (Attack/Defense Cost Functions)	$\hat{a}(c) = c$ ^ 0.5 ± (1~5%) * c, $\hat{a}(c) = c$ ^ 1 ± (1~5%) * c, $\hat{a}(c) = c$ ^ 2 ± (1~5%) * c
Ratios of the Fixed Attack Costs	1%~30%
Attack Strategy	LR-Based Attack Algorithm, SA1, SA2
Redundancy and Defense Allocation Algorithm	LR-Based Allocation Algorithm, Core Focused Allocation Algorithm, Randomized Allocation Algorithm

4.1.5 Experiment Results

The metric we use for evaluating the efficiency of an attack algorithm is the total attack cost that the attacker needs to pay for achieving the ultimate goal, compromising all core nodes, in the target network. For an attack algorithm, the fewer total attack cost means it is more effective to achieve the attack goal.

The LR value is the result calculated by our proposed LR-Based attack algorithm during the getting primal feasible solution process, and SA1 and SA2 are the results obtained form simple attack algorithm 1 and simple attack algorithm 2, respectively. The LB is the lower bound got from LR procedure, and the GAP is calculated as $\frac{LR-LB}{LB}\times 100\%$ to evaluate the quality of LR. For clearly demonstrating the difference between LR-Based attack algorithm and the others, we calculate the improvement ratio of LR to SA1 and SA2 as $\frac{SA1-LR}{SA1}\times 100\%$ and $\frac{SA2-LR}{SA2}\times 100\%$, respectively. The results of experiments are demonstrated in the following pages.

Table 4-2 Attack Results of the Experiments on Different A/D Cost Functions

100 Nodes, 50000 Budget, 3+1 Functions, 10 Types					
Attack cost = Defense cost ^ 0.5		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	1191.99	44.36	19.28	6.20
	Core Focused Allocation	934.82	16.46	7.00	2.54
	Randomized Allocation	534.719	28.48	15.49	12.92
Attack cost = Defense cost ^ 1		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	5556.23	49.89	25.24	7.99
	Core Focused Allocation	4409.95	21.31	11.46	8.03
	Randomized Allocation	2770.61	32.15	11.29	6.31
Attack cost = Defense cost ^ 2		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	203793	49.39	35.29	7.58
	Core Focused Allocation	162920	22.70	23.92	2.82
	Randomized Allocation	154362	19.57	14.38	1.44

Table 4-3 Attack Results of the Experiments on Different Network Sizes

Attack cost = Defense cost ^ 1, 3+1 Functions, 10 Types						
49 Nodes 25000 Budget		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	4491.73	20.13	26.53	5.43
	Core Focused Allocation	3646.68	9.61	8.08	5.68
	Randomized Allocation	2529.51	10.71	12.34	3.35
100 Nodes 50000 Budget		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	5556.23	49.89	25.24	7.99
	Core Focused Allocation	4409.95	21.31	11.46	8.03
	Randomized Allocation	2770.61	32.15	11.29	6.31
400 Nodes 200000 Budget		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	6219.5	59.89	19.36	1.93
	Core Focused Allocation	4907.68	39.60	4.46	2.51
	Randomized Allocation	3436.64	52.92	32.09	4.68

Table 4-4 Attack Results of the Experiments on Different Defense Budget

Attack cost = Defense cost ^ 1, 100 Nodes, 3+1 Functions, 10 Types					
30000 Budget		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	5093.43	30.88	19.36	3.27
	Core Focused Allocation	3170.80	27.37	9.22	3.63
	Randomized Allocation	2134.22	20.93	12.77	9.28
50000 Budget		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	5556.23	49.89	25.24	7.99
	Core Focused Allocation	4409.95	21.31	11.46	8.03
	Randomized Allocation	2770.61	32.15	11.29	6.31
100000 Budget		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	5565.7	44.17	27.91	11.16
	Core Focused Allocation	4783.47	35.71	15.14	10.04
	Randomized Allocation	4516.21	25.68	18.42	4.22

Table 4-5 Attack Results of the Experiments on Different Choice Set Sizes

Attack cost = Defense cost ^ 1, 100 Nodes, 50000 Budget, 3+1 Functions					
10 Types		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	5556.23	49.89	25.24	7.99
	Core Focused Allocation	4409.95	21.31	11.46	8.03
	Randomized Allocation	2770.61	32.15	11.29	6.31
15 Types		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	7440.05	43.50	26.33	15.55
	Core Focused Allocation	5664.00	20.51	9.77	10.82
	Randomized Allocation	2932.09	33.15	21.24	13.67
20 Types		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	9717.51	44.23	22.35	15.59
	Core Focused Allocation	7038.98	18.98	6.68	5.02
	Randomized Allocation	3581.61	41.48	16.44	13.90

Table 4-6 Attack Results of the Experiments on Different Numbers of Functions

Attack cost = Defense cost ^ 1, 100 Nodes, 50000 Budget, 10 Types					
3+1 Functions		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	5556.23	49.89	25.24	7.99
	Core Focused Allocation	4409.95	21.31	11.46	8.03
	Randomized Allocation	2770.61	32.15	11.29	6.31
5+1 Functions		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	6578.06	49.88	30.98	14.57
	Core Focused Allocation	4719.81	26.27	16.12	10.60
	Randomized Allocation	3406.18	22.40	12.89	3.33
10+1 Functions		LR-Based Attack Cost	GAP (%)	Improvement Rate to SA1 (%)	Improvement Rate to SA2 (%)
	LR-Based Allocation	7538.73	47.84	22.13	14.60
	Core Focused Allocation	4963.68	26.31	12.08	17.67
	Randomized Allocation	3442.62	33.96	17.95	9.60

4.1.6 Discussion of Results

Our proposed LR-Based attack algorithm always helps the attacker achieve the ultimate attack goal, i.e., compromise all core nodes in the target network, with a lower total attack cost compared with the simple attack algorithm 1and 2. The total attack cost achieved by the LR-Based attack algorithm is about 10%~35% lower than which achieved by the simple attack algorithm 1 and about 10% lower than which achieved by the simple attack algorithm 2 in average.

Moreover, there is a fact that the LR-Based attack algorithm seems to provide more help when the attacker attempts to compromise LR-Based allocated networks in most cases.

4.2 Computational Experiments with the RAP-EDM Model

4.2.1 Experiment Environment

The experiment environment and parameter settings of the RAP-EDM model are similar to which of the AEA model, thus we directly provide them in the table below.

Table 4-7 Experiment Parameter Settings of the RAP-EDM Model

Experiment Platform	CPU: Intel Atom ^{TM} N270 1.60GHz RAM: 2GB OS: Windows XP Home Edition SP3
Programming Environment	Dev-C++ Version 4.9.9.2
Parameters of the RAP-EDM Model	
Parameters	Value
Testing Topology	Grid Networks
Number of Core Nodes	6
Service Continuity Requirement	2
Capacity Limit of Redundant Components	5
Probability of the Non-Core Nodes Providing Service Functions	50%
Discount Rate of Attack Threshold	10%
Number of Nodes	49, 100, 400
Defense Budget	30000, 50000, 100000
Number of Functions	3+1, 5+1, 10+1(add a transmission function)
Redundant Component and Defense Mechanism Choice Set Size	10, 15, 20
Costs of Redundant Components	50 ~ 100
Probabilities of redundant components operating properly	85% ~ 99%
Costs of Defense Mechanisms	1 ~ 20
Defense Capability	$\hat{a}(c) = c$ ^ 0.5 ± (1~5%) * c, $\hat{a}(c) = c$ ^ 1 ± (1~5%) * c, $\hat{a}(c) = c$ ^ 2 ± (1~5%) * c
Attack Strategy	LR-Based Attack Algorithm, SA1, SA2
Redundancy and Defense Allocation Strategy	LR-Based Allocation Strategy, Randomized Allocation Algorithm
Number of Allocation Adjustment	80

4.2.2 Experiment Results

The metric we use for evaluating the robustness of an allocation is the total attack cost that the attacker needs to spend on compromising all core nodes in the target network. For an allocation, the more total attack cost that an allocation forces the attacker to pay represents it is more robust when facing malicious attacks.

The LR value is the total attack cost that the attacker needs to spend on compromising all core nodes when attacking a LR-Based allocation network. The RA value and the CF value are the total attack cost that a randomized allocation network and a core focused allocation network force the attacker to pay for compromising it, respectively. For clearly demonstrating the difference between LR-Based allocation algorithm and the randomized allocation algorithm, we calculate the improvement ratio to RA as $\frac{LR-RA}{RA}\times 100\%$. On the other hand, we also calculate the improvement ratio to CF as $\frac{LR-CF}{CF}\times 100\%$ to show how better the LR-Based allocation algorithm is, compared with the core focused allocation algorithm. The data obtained from experiments are provided below.

Table 4-8 Defense Results of the Experiments on Different A/D Cost Functions

100 Nodes, 50000 Budget, 3+1 Functions, 10 Types				
Attack cost = Defense cost ^ 0.5		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	1191.99	1421.84	1265.93
	Improvement Rate to CF (%)	27.51	42.15	32.07
	Improvement Rate to RA (%)	122.92	130.24	109.66
Attack cost = Defense cost ^ 1		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	5556.23	6958.69	6000.25
	Improvement Rate to CF (%)	25.99	41.57	25.95
	Improvement Rate to RA (%)	100.54	125.69	103.72
Attack cost = Defense cost ^ 2		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	203793	275719	219237
	Improvement Rate to CF (%)	25.09	36.57	30.87
	Improvement Rate to RA (%)	32.02	56.16	40.01

Table 4-9 Defense Results of the Experiments on Different Network Sizes

Attack cost = Defense cost ^ 1, 3+1 Functions, 10 Types				
49 Nodes 25000 Budget		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	4491.73	5683.16	4735.4
	Improvement Rate to CF (%)	23.17	44.20	22.87
	Improvement Rate to RA (%)	77.57	100.00	81.15
100 Nodes 50000 Budget		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	5556.23	6958.69	6000.25
	Improvement Rate to CF (%)	25.99	41.57	25.95
	Improvement Rate to RA (%)	100.54	125.69	103.72
400 Nodes 200000 Budget		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	6219.5	7423.57	6339.37
	Improvement Rate to CF (%)	26.73	44.81	26.01
	Improvement Rate to RA (%)	80.98	63.54	76.21

Table 4-10 Defense Results of the Experiments on Different Defense Budget

Attack cost = Defense cost ^ 1, 100 Nodes, 3+1 Functions, 10 Types				
30000 Budget		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	5093.43	6079.39	5259.85
	Improvement Rate to CF (%)	60.64	75.54	60.07
	Improvement Rate to RA (%)	138.66	152.59	125.53
50000 Budget		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	5556.23	6958.69	6000.25
	Improvement Rate to CF (%)	25.99	41.57	25.95
	Improvement Rate to RA (%)	100.54	125.69	103.72
100000 Budget		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	5565.7	7119.3	6186.63
	Improvement Rate to CF (%)	16.35	29.26	17.53
	Improvement Rate to RA (%)	23.24	33.12	31.45

Table 4-11 Defense Results of the Experiments on Different Choice Set Sizes

Attack cost = Defense cost ^ 1, 100 Nodes, 50000 Budget, 3+1 Functions				
10 Types		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	5556.23	6958.69	6000.25
	Improvement Rate to CF (%)	25.99	41.57	25.95
	Improvement Rate to RA (%)	100.54	125.69	103.72
15 Types		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	7440.05	9398.83	8597
	Improvement Rate to CF (%)	31.36	51.17	36.97
	Improvement Rate to RA (%)	153.75	164.39	157.94
20 Types		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	9717.51	11889.4	11232.8
	Improvement Rate to CF (%)	39.36	59.94	50.98
	Improvement Rate to RA (%)	171.32	185.08	175.36

Table 4-12 Defense Results of the Experiments on Different Numbers of Functions

Attack cost = Defense cost ^ 1, 100 Nodes, 50000 Budget, 10 Types				
3+1 Functions		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	5556.23	6958.69	6000.25
	Improvement Rate to CF (%)	25.99	41.57	25.95
	Improvement Rate to RA (%)	100.54	125.69	103.72
5+1 Functions		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	6578.06	8616.26	7536.19
	Improvement Rate to CF (%)	39.37	57.21	44.36
	Improvement Rate to RA (%)	93.12	124.07	114.12
10+1 Functions		LR-Based Attack Cost	SA1 Attack Cost	SA2 Attack Cost
	LR-Based Allocation	7538.73	9206.93	8639.67
	Improvement Rate to CF (%)	51.88	65.49	47.91
	Improvement Rate to RA (%)	118.98	126.73	128.99

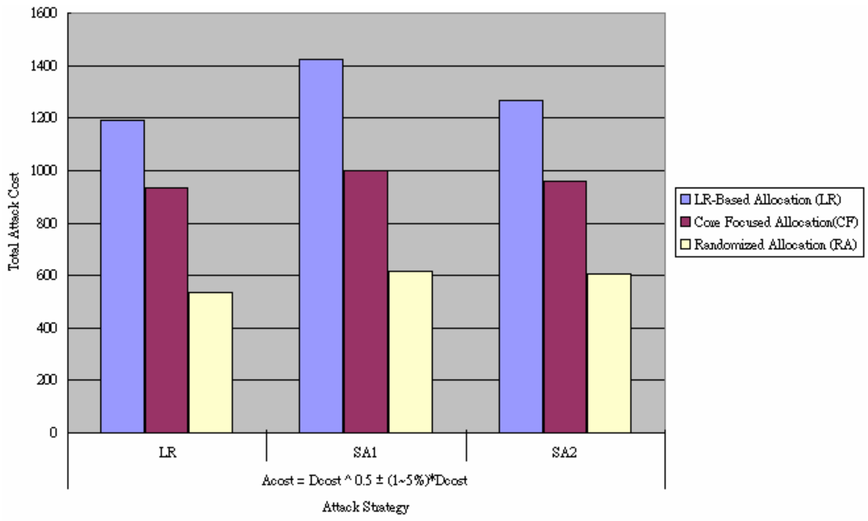

Figure 4-1 Defense Results of the Experiments on Concave A/D Cost Function

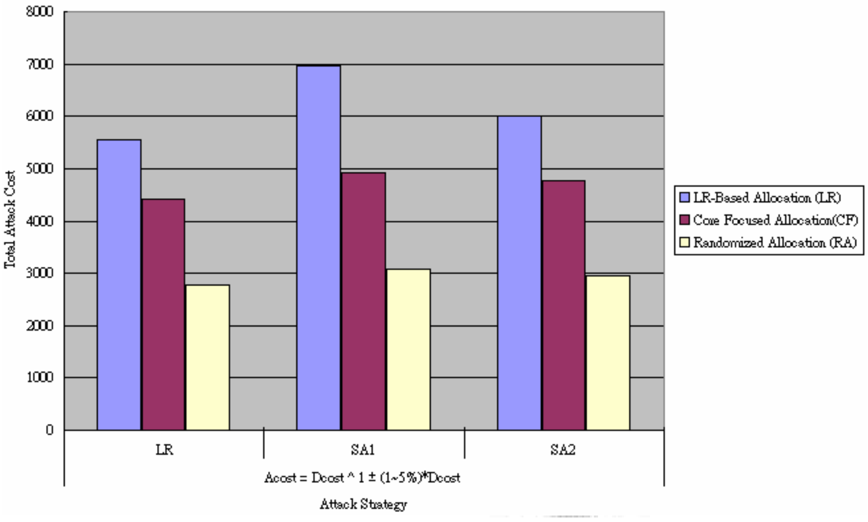

Figure 4-2 Defense Results of the Experiments on Linear A/D Cost Function

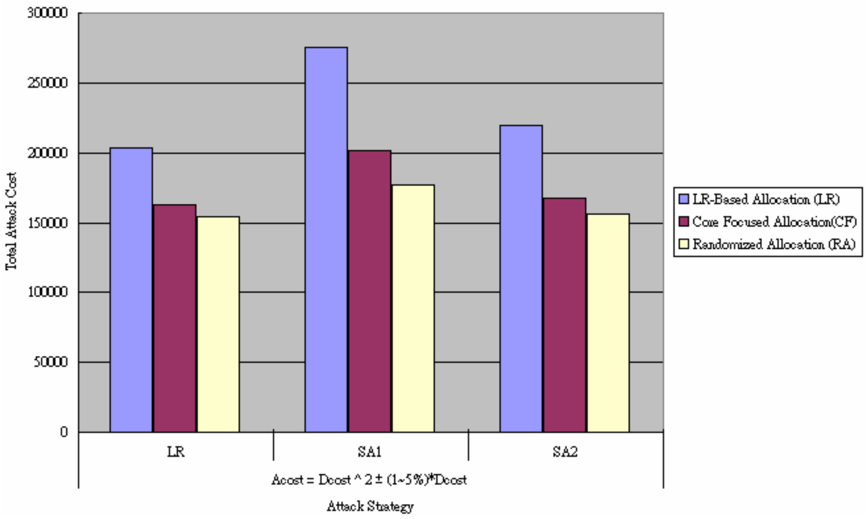

Figure 4-3 Defense Results of the Experiments on Convex A/D Cost Function

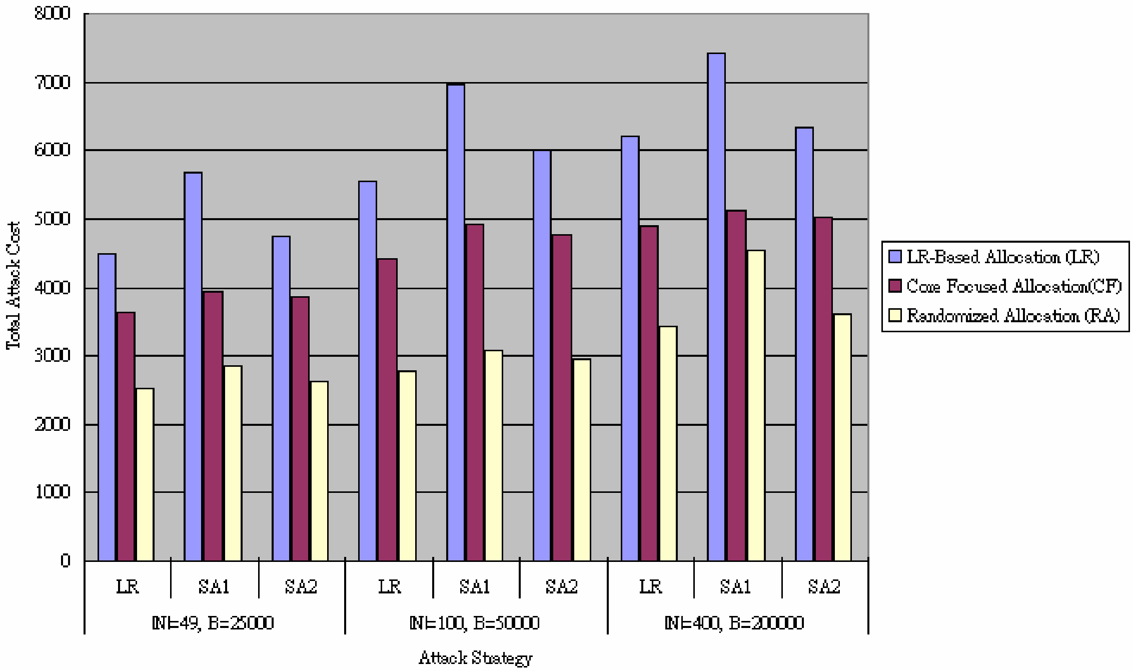

Figure 4-4 Defense Results of the Experiments on Different Network Sizes

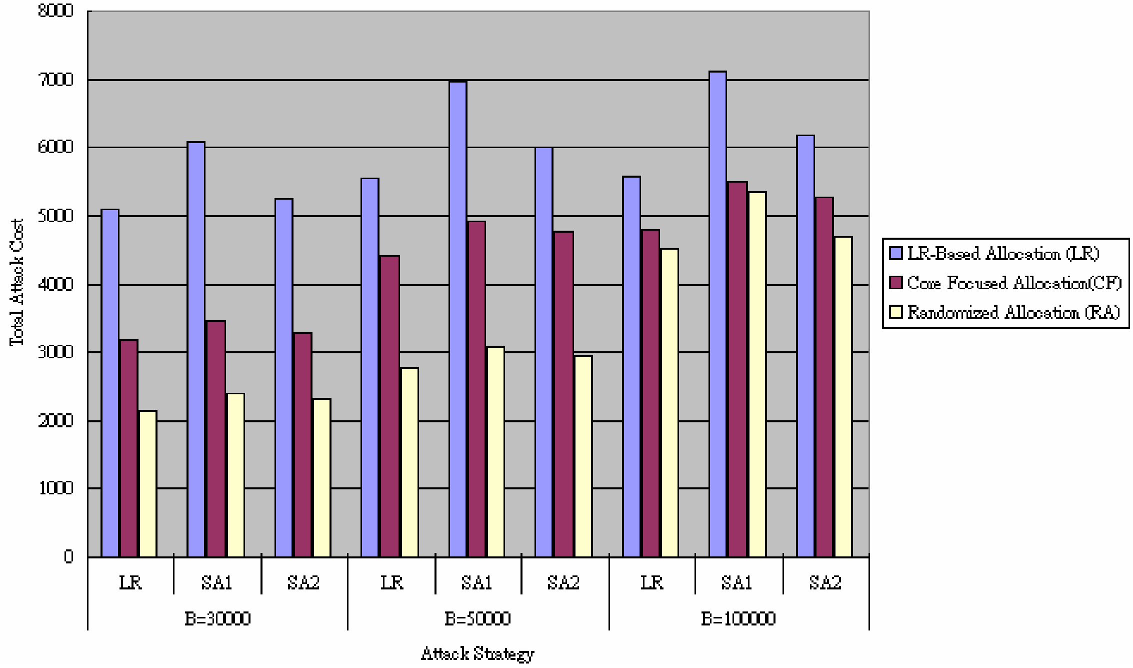

Figure 4-5 Defense Results of the Experiments on Different Defense Budget

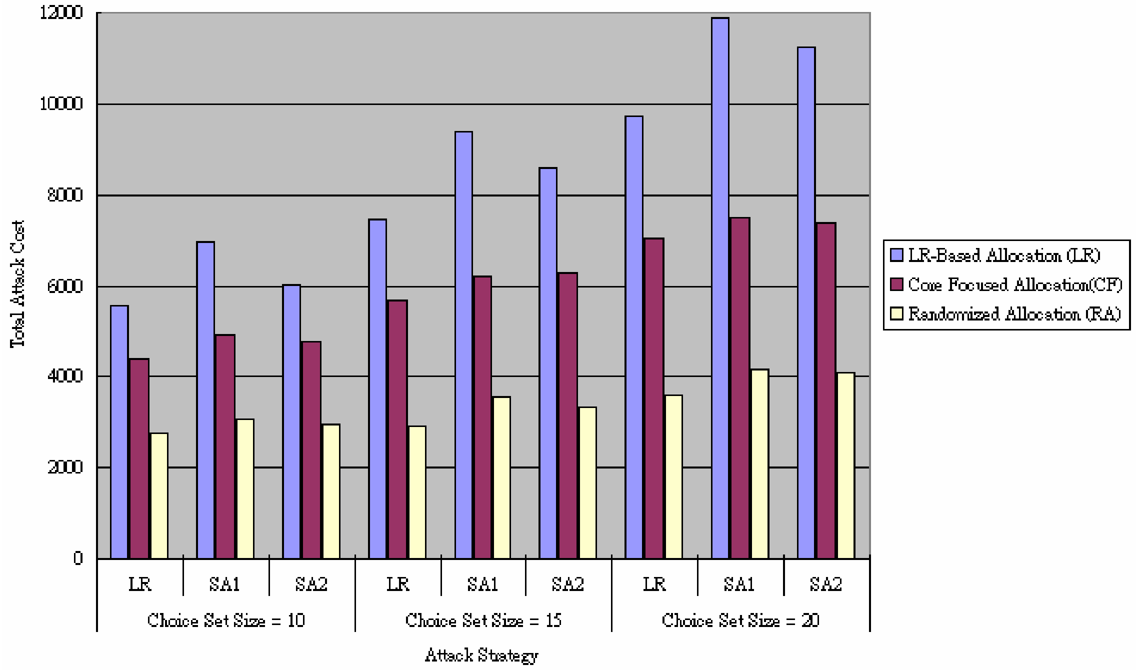

Figure 4-6 Defense Results of the Experiments on Different Choice Set Sizes

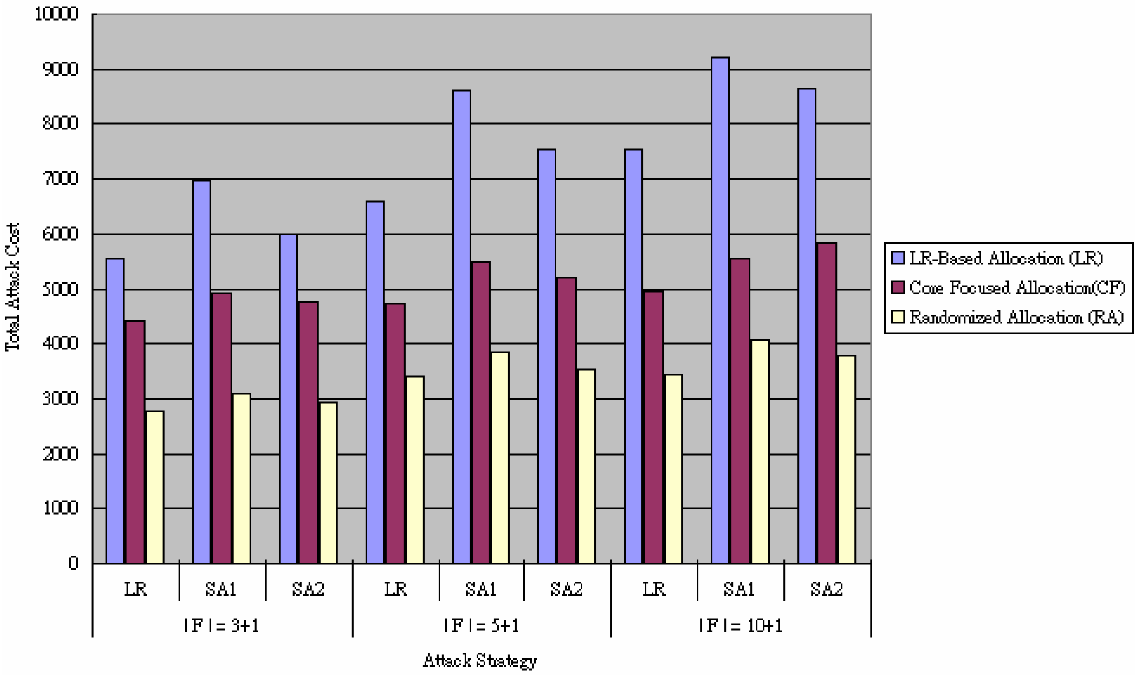

Figure 4-7 Defense Results of the Experiments on Different Numbers of Functions

4.2.3 Discussion of Results

No matter adopting what kind of relation function between the attack costs and defense costs of redundant components and defense mechanisms, the LR-Based allocation algorithm can always provide much better defense capability compared with the randomized allocation algorithm and the core focused allocation algorithm.

The different attack cost of compromising the networks in different sizes is shown in Figure 4-4. The larger networks cost the attacker more for compromising the whole network, but the enhancement of total attack cost tends to be smaller while the choice set sizes of redundant components and defense mechanisms remain the same rather than expanding with the network size. This improves the impact of the attack experience accumulation. Furthermore, the improvement ratio slightly drops down when the number of nodes increases from 100 to 400. This can be explained as that larger network provides the defender more space to randomly allocate different kinds of redundant components and defense mechanisms in the randomized allocation process.

We observe the impact on the total attack cost caused by the amount of defense budget from Figure 4-5. When the defense budget is not that abundant, i.e., 30,000, the LR-Based allocation algorithm can produce great improvement ratio compared with the randomized allocation algorithm and the core focused allocation algorithm; however, the difference between the results caused by the allocation algorithms becomes smaller when the defense budget is relatively ample, i.e., 100,000. The reason is that great amount of defense budget lets the defender has more probabilities of randomly allocating different kinds of redundant components and defense mechanisms into nodes when adopting the randomized allocation algorithm or executing the random allocation procedure of the core focused allocation algorithm.

In addition, the marginal effect on the total attack cost produced by doubling the defense budget from 50,000 to 100,000 is comparatively slight, and we attribute this to the capacity limit and the attack experience accumulation. In other words, the defense budget of 50,000 seems enough to uniformly allocate all different kinds of redundant components and defense mechanisms to the 100-noded grid network under such experiment parameter settings.

In Figure 4-6, the expansion of the choice set sizes brings more kinds of products, i.e., redundant components and defense mechanisms, into this problem. When the defender adopts the LR-Based allocation algorithm, which gives consideration to making the times of allocating each kind of product uniform as far as possible, the greater diversity of products really helps enhance the total attack cost. On the other hand, the randomized allocation algorithm cannot make good use of the diversity of products to enhance the total attack cost. Therefore, the difference between the LR-Based allocation and the randomized allocation becomes more obvious when the diversity of products getting larger. This phenomenon can also be observed from Figure 4-7 because the more functions a network needs to provide, the more kinds of different redundant components and defense mechanisms the defender can choose for allocation.

At the end of this chapter, we provide some guidelines for redundancy and defense allocation according to the results of experiments above. The defender can achieve much better defense results by adopting sophisticated allocation methods, e.g., the LR-Based allocation algorithm, in three situations. Firstly, the defense budget is not that abundant. Secondly, the choices of redundant components and defense mechanisms are rich. Thirdly, the target network provides many kinds of functions. Moreover, concentrating on strengthening the core nodes first really enhances the total attack cost under the defense/attack scenario we discussed. At last, the best way to enhance the total attack cost is finding more non existing types of redundant components and defense mechanisms to allocate while the marginal effect of defense investment on the total attack cost becomes slight.

Chapter 5 Conclusion and Future Work

5.1 Conclusion

Service continuity and networks are vital to businesses nowadays without a doubt. Redundancy is obviously an effective way to facilitate service continuity; however, there is little research that studies RAP in network environments. In addition, the principal concern of past RAP related works is the reliability of whole system considering only non man-made failures (e.g., natural disasters, random errors), but the fact is that malicious attacks are ubiquitous, especially in network environments.

In this thesis, we discuss RAP in a network topology existing intelligent attackers who attempt to compromise whole network in terms of making all the core nodes dysfunctional. Thus the network operator as a defender needs to prepare for the worst case that the attacker can always compromise the target network eventually. All the defender can do is making the attacker pay for it as much as possible, and the defender also has to concern about the service continuity assurance for legitimate users considering non man-made failures simultaneously.

In order to fully describe the above attack/defense scenario in a clear way, we formulate it as two mathematical models, RAP-EDM model and AEA model, to represent the behavior of the defender and the attacker, respectively. We successfully model the competition between both sides into a mathematical problem and further solve it by proposed mathematical methodology based heuristics. This can be regarded as the key contribution of this work.

Furthermore, current cyber attackers are brilliant enough to find out security vulnerabilities and then develop hacker tools for those vulnerabilities to facilitate follow-up attack activities. In other words, the intelligent attackers have the ability to accumulate useful experience during their attack processes. We well transform this feature of real attacks into mathematical formulations in proposed models, and this is another contribution provided by this thesis.

We experiment with our proposed attack strategy and allocation algorithm in fifteen different grid network environments, which are resulted from different parameter configurations. According to the results of computational experiments, our proposed attack strategy can always help the attacker compromise the whole network with a lower total cost compared with the other two simple attack algorithms. On the other hand, our proposed allocation algorithm conspicuously enhances the total attack cost that the attacker has to spend for achieving attack goal compared with randomized allocation.

Most importantly, we provide guidelines for allocation of redundancy and extra defense. When the defense budget is not abundant, the defender especially needs to choose and allocate well the redundancy and defense. In addition, the defender also has to carefully consider about how to well allocate redundancy and defense under the following two circumstances. Firstly, the target network provides many kinds of different functions. Secondly, the choices of products, i.e., redundant components and defense mechanisms, become more. When the marginal effect of defense investment on the total attack cost tends to become small, the best way to further enhance the survivability is expanding the sizes of product choice sets rather than keeping investing money in allocating more existed products.

5.2 Future Work

As the thesis progressing to the end, there are some possible extensions of our research provided below for reference.

According to the results of experiments, allocating as many different kinds of redundancy and defense as possible really helps the defender enhance the survivability in terms of the total attack cost that the attacker spends on compromising whole network; however, there must exist a fact in reality that purchasing a great number of identical products at one time often gets extra discount on price. Thus, the tradeoff between the diversity of purchased products (including redundant components and defense mechanisms) and the bargaining power becomes interesting and is worthy to be discussed.

Furthermore, the maintenance cost of 10 different equipments is absolutely much higher than which of 10 identical equipments because great diversity of products brings more complicated maintenance works to the defender afterwards. Therefore, the maintenance cost can be calculated into the defense budget in the future, and the impact on maintenance cost resulted from allocated product diversity can be further described in this problem.

In our current scenario, the whole plan about which node providing what kind of function is pre-specified and consistent. The positions of core nodes are also predefined. When we turn the role of the defender into the network planner, he/she may have to determine that the service functions should be provided by which nodes and decide where to allocate core nodes.

Since each node providing what kind of function becomes one of the defender's decisions, the service continuity requirement needs to be redefined at the whole network level instead of the single node level based on the real service requirements of network users. Therefore, the defender as a network planner can exploit redundancy to assure the service continuity of the whole network in a more economical fashion.

In this research, we model RAP in network environments as a defense/attack scenario considering both non man-made failures and malicious attacks. Moreover, we also clearly describe the real attackers’ capability of experience accumulation in a mathematical way within our model. None the less, all the issues of future research mentioned above, e.g., the tradeoff between the diversity of products and the bargaining power, the concern of maintenance cost, have the potential to make this research conform to the reality more. Therefore, the follow-up research is worthy to be conducted for enriching this work in the future.

References

[1] M.J. Cerullo and V. Cerullo, “Business Continuity Planning: A Comprehensive Approach,” *Information Systems Management,* Volume 21, No. 3, pp. 70-78, June 2004.

[2] B. B.M. Shao, “Optimal Redundancy Allocation for Information Technology Disaster Recovery in the Network Economy,” *IEEE Transactions on Dependable and Secure Computing,* Volume 2, No. 3, pp. 262-267, July-September 2005.

[3] W. Lam, “Ensuring Business Continuity,” *IT Professional,* Volume 4, Issue 3, pp. 19-25, May-June 2002.

[4] British Standard Institution (BSI), “BS25999,” November 2006.

[5] Frost & Sullivan and (ISC)^2, “The 2008 (ISC)^2 Global Information Security Workforce Study,” April 2008.

[6] AT&T, “AT&T’s Business Continuity Survey: 2008,” July 2008.

[7] P. Fallara, “Disaster Recovery Planning,” *IEEE Potentials,* Volume 22, Issue 5, pp. 42-44, December 2003-January 2004.

[8] A. Joseph, A. Mobolurin, H. Millar, and K.-M. Bryson, “Using Formal MS/OR Modeling to Support Disaster Recovery Planning,” *European Journal of Operational Research,* Volume 141, pp. 679-688, 2002.

[9] A.E. Smith and D.W. Coit, “Reliability Optimization of Series-Parallel Systems Using a Genetic Algorithm,” *IEEE Transactions on Reliability,* Volume 45, No. 2, pp. 254-260, 266, June 1996.

[10] Y.-C. Hsieh, “A Linear Approximation for Redundant Reliability Problems with Multiple Component Choices,” *Computers and Industrial Engineering,* Volume 44, Issue 1, pp. 91-103, January 2003.

[11] A. Konak, D.W. Coit, and J.E. Ramirez-Marquez, “Redundancy Allocation for

Series-Parallel Systems Using a Max-min Approach," *IIE Transactions,* Volume 36, Issue 9, pp. 891-898, September 2004.

[12] G. Levitina and K. Hausken, "Protection vs. Redundancy in Homogeneous Parallel Systems," *Reliability Engineering and System Safety*, Volume 93, Issue 10, pp. 1444-1451, November 2007.

[13] V.R. Westmark, "A Definition for Information System Survivability," *Proceedings of the 37th IEEE Hawaii International Conference on System Sciences*, January 2004.

[14] D.A. Fisher, H.F. Lipson, N.R. Mead, R.C. Linger, R.J. Ellison, and T. Longstaff, "Survivable Network Systems: An Emerging Discipline," *Technical Report CMU/SEI-97-TR-013,* November 1997 (Revised: May 1999).

[15] T.-Z. Jiang, "A New Definition on Survivability of Communication Networks," *Conference Record of IEEE Military Communications Conference 1991 (MILCOM'91),* Volume 3, pp. 901-904, November 1991.

[16] A.D. Malloy, A.P. Snow, and U. Varshney, "Reliability and Survivability of Wireless and Mobile Networks," *Computer,* Volume 33, Issue 7, pp. 49-55, July 2000.

[17] L. Guo, L.-J. Zhang, W. Wang, W. Yang, and Y.-T. Yang, "A Survivability Quantitative Analysis Model for Network System Based on Attack Graph," *International Conference on Machine Learning and Cybernetics 2007,* Volume 6, pp. 3211-3216, August 2007.

[18] R. Richardson, "2008 CSI Computer Crime and Security Survey," 2008.

[19] ATIS Telecom Glossary 2007, *http://www.atis.org/glossary/definition.aspx?id=1039* (Original: "Federal Standard 1037C," August 1996).

[20] M.N. Azaiez and V.M. Bier, "Optimal Resource Allocation for Security in

Reliability Systems,” *European Journal of Operational Research,* Volume 181, Issue 2, pp. 773-786, September 2007.

[21] A.M. Geoffrion, “Lagrangean Relaxation for Integer Programming,” *Mathematical Programming Study,* Volume 2, pp. 82-114, 1974.

簡歷

姓名：駱睿斌

出生地：臺灣省宜蘭縣

出生日：中華民國七十四年五月二十四日

學歷：九十二年九月至九十六年六月
國立中央大學資訊管理學系

九十六年九月至九十八年八月
國立臺灣大學資訊管理研究所