การพัฒนาซอฟแวร์วิเคราะห์ประเมินประสิทธิภาพของระบบป้องกันทางกายภาพ

นายอาลิม มาร์ดี

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย
The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the University Graduate School.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีนิวเคลียร์ ภาควิชาวิศวกรรมนิวเคลียร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2558

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Development of Analytical Software for Assessing Physical Protection System.

Mr. Alim Mardhi

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Nuclear Technology
Department of Nuclear Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2015
Copyright of Chulalongkorn University

Thesis Title	Development of Analytical Software for Assessing Physical Protection System.
By	Mr. Alim Mardhi
Field of Study	Nuclear Technology
Thesis Advisor	Dr Phongphaeth Pengvanich

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

.... Dean of the Faculty of Engineering
(Professor Bundhit Eua-arporn, Ph.D.)

THESIS COMMITTEE

.... Chairman
(Associate Professor Sunchai Nilsuwankosit, Ph.D.)

.... Thesis Advisor
(Dr Phongphaeth Pengvanich)

.... Examiner
(Associate Professor Supitcha Chanyotha, Ph.D.)

.... External Examiner
(Dr Julwan Hendry Purba)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

อาลิม มาร์ดี : การพัฒนาซอฟแวร์วิเคราะห์ประเมินประสิทธิภาพของระบบป้องกันทางกายภาพ (Development of Analytical Software for Assessing Physical Protection System.) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ดร. พงษ์แพทย์ เพ่งวาณิชย์, 69 หน้า.

การประเมินประสิทธิภาพของระบบป้องกันทางกายภาพถือเป็นสิ่งที่สำคัญในลำดับต้นๆ ในการป้องกันสถานปฏิบัติการทางนิวเคลียร์จากการกระทำที่ไม่ชอบด้วยกฎหมาย เช่น การลักลอบนำวัสดุนิวเคลียร์ออกจากสถานปฏิบัติการ โดยมิได้รับอนุญาต และการก่อวินาศกรรมในสถานปฏิบัติการ อย่างไรก็ตาม การประเมินด้วยการทดสอบจริงในสถานการณ์จำลองจะมีค่าใช้จ่ายที่สูงและใช้เวลาเตรียมการมาก ดังนั้น การประเมินด้วยคอมพิวเตอร์จึงเป็นวิธีการแก้ปัญหาหนึ่งซึ่งสามารถนำไปใช้ในการวิเคราะห์การป้องกันภัยคุกคามต่างๆ ที่อาจเกิดขึ้นได้ ซึ่งในปัจจุบันแม้ว่าจะมีเครื่องมือสำเร็จรูปลักษณะนี้ที่สามารถนำมาใช้ได้ทันทีอยู่แล้ว เช่น EASI และ SAPE แต่อาจไม่เหมาะสมสำหรับใช้ในงานวิจัยนี้ซึ่งต้องการเครื่องมือที่สามารถเปลี่ยนแปลงและปรับปรุงเพิ่มเติมได้ ดังนั้นในงานวิจัยนี้จึงได้ทำการพัฒนาเครื่องมือประเมินทางคอมพิวเตอร์ขึ้นมาใหม่ โดยนำวิธีการทางระบบเครือข่ายเข้ามาใช้ในการสร้างแบบจำลองเส้นทางการโจมตีของผู้บุกรุก สามารถเรียกใช้ข้อมูลในมิติต่างๆ ที่เกี่ยวกับความมั่นคงปลอดภัยในการประเมินประสิทธิภาพของระบบในส่วนการตรวจจับ การชะลอการบุกรุก และการตอบสนองต่อการบุกรุก เครื่องมือนี้ยังสามารถประเมินเส้นทางที่เป็นจุดอ่อนและค่า Probability of Effectiveness ซึ่งเป็นตัวชี้วัดสมรรถภาพของระบบได้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาควิชา วิศวกรรมนิวเคลียร์ ลายมือชื่อนิสิต ....
สาขาวิชา เทคโนโลยีนิวเคลียร์ ลายมือชื่อ อ.ที่ปรึกษาหลัก ....
ปีการศึกษา 2558

5670583021 : MAJOR NUCLEAR TECHNOLOGY
KEYWORDS: PHYSICAL PROTECTION SYSTEM / ANALYTICAL TOOL / NETWORK / ADVERSARY PATH ANALYSIS.

ALIM MARDHI: Development of Analytical Software for Assessing Physical Protection System.. ADVISOR: DR PHONGPHAETH PENGVANICH, ., 69 pp.

Assessment of physical protection system effectiveness is the priority for ensuring the optimum protection from unlawful acts against a nuclear facility, such as unauthorized removal of nuclear materials and sabotage of the facility itself. Since an assessment based on real exercise scenarios is costly and time-consuming, the computer-based analytical tool can offer the solution for approaching the likelihood threat scenario. There are several currently available tools that can be used instantly such as EASI and SAPE, however for our research purpose it is more suitable to have the tool that can be customized and enhanced further. In this work, we have developed a computer–based analytical tool by utilizing the network methodological approach for modeling the adversary paths. The inputs are multi-elements in security used for evaluating the effectiveness of the system's detection, delay, and response. The tool has capability to analyze for the most critical path and quantify the probability of effectiveness of the system as a performance measure.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Department:	Nuclear Engineering	Student's Signature	....
Field of Study:	Nuclear Technology	Advisor's Signature	....
Academic Year:	2015		

ACKNOWLEDGEMENTS

In the Name of Allah, the Most Beneficent, the Most Merciful

All praise is to Allah, the Almighty, on whom ultimately we depend for sustenance and guidance. Second, my sincere appreciation goes to my supervisor Dr. Phongphaeth Pengvanich, whose guidance, careful reading and constructive comments was valuable. His timely and efficient contribution helped me shape this into its final.

Besides my advisor, I would like to thank the rest of my thesis committee: Associate Prof. Dr. Sunchai Nisuwankosit, Associate Prof.Dr Supitcha Chanyotha, and Dr. Julwan Hendry Purba , for their insightful comments to widen my research from various perspectives.

I would like to thank my family: my parents Abah and Umak, and to my brothers and sister for supporting me spiritually throughout writing this thesis and my life in general. Last but not the least, my wife Yenni Arisandi and my beloved sons Rasyid and Usamah for the patience and encouragement all the time during finishing this thesis.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

CONTENTS

Page

Page

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

LIST OF TABLES

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

LIST OF FIGURES

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

CHAPTER I
INTRODUCTION

1.1 BACKGROUND

There were two events that have significantly influenced the mindset of physical protection of nuclear facilities. The first event was the attack on the World Trade Centre (WTC) on September 11, 2001. The second was the Fukushima accident. Before the September 11 tragedy, the physical protection of nuclear facilities were focused on the protection of nuclear materials from being lost or stolen from facilities or during transportation. The perceptions changed after the WTC has "successfully been attacked", demonstrating the high capability and strong intention of the non-state actor to perform sabotage on high value target. The sabotage did not use the conventional weapons and tactics anymore but already considered advance method and unexpected scenario of attacks. This type of threats and scenario may happen to nuclear facility. Therefore it must be factored into the physical protection of nuclear facility[1]. The second one was the Fukushima accident on March 11, 2011. This catastrophic accident actually was initiated by an extraordinary natural disaster and caused severe consequences to the facility. However based on safety issue in the current reactor technology, the likelihood of severe accident which has consequence as worst as Fukushima accident are very low. In the matter of nuclear security point of view, such kind of consequences are still considered to have probability due to the sabotage or terrorist attack scenario on the facility[2]

Chulalongkorn University

The physical protection system (PPS) has been designed with three primary functions, which are detection, delay, and response, to protect the facility from intrusion of intentionally unauthorized persons or group to steal nuclear materials form or sabotage the facility. The objective of the PPS can be determined by threat assessment and facility characterization. For providing adequate protection to meet this objective, the PPS effectiveness needs to be assessed regularly. This routine activity become complicated when the threats to the facility escalate dynamically and each facility has been design based on their own unique site characteristic. The authority of the facility needs to establish the methodology and tools to support the PPS assessment analysis. The analysis based on the software has become one of alternative solutions. This approach offers three advantageous, i.e. reducing the cost of evaluation program compared to the real exercise scenarios, easy to modify the scenario and can be designed with many facility characteristics.

1.2 MOTIVATION

IAEA have been promoting the peaceful use of nuclear material and facilities and setting many international standards as guidance for implementation. One of the recommendation by IAEA is every state must develop its national security plan to manage the nuclear security incidents. One of the main features of this national security plan is to evaluate the physical protection of nuclear facilities. Due to the importance of the PPS establishment in nuclear facilities, International Atomic Energy Agency (IAEA) has recommended to the member states to define requirements for the physical protection of nuclear material in use and storage and during transportation and for nuclear facilities in their countries[3]. The first legally binding instrument dedicated to physical protection was the Convention on the Physical Protection of Nuclear Material (CPPNM), which was opened for signature in 1980 and came into force in 1987. Since the conclusion of the CPPNM, several other legally binding instruments related to physical protection have also come into effect, such as the International Convention for the Suppression of Acts of Nuclear Terrorism (Nuclear Terrorism Convention) and UN Security Council Resolution 1540[4].

The primary resource document for technical guidance is IAEA's guidance document The Physical Protection of Nuclear Material (INFCIRC/225) which has been amendments up to INFCIRC/225/Rev.5. IAEA has recommended that the state should establish the competent authority to control the implementation of physical protection system in nuclear facilities and recommend the operator to established and implement quality assurance policy and physical protection programs. The quality assurance should ensure that a physical protection system is designed, implemented, operated and maintained in a condition to enable to effectively respond to the threat assessment or design basis threat and that it meets the State's regulations, including its prescriptive and/or performance based requirements.

Indonesia as member of IAEA since 1957 has involved in the use of nuclear technology and application. As legal framework, nuclear research and application in Indonesia is guided by the Act 10/1997 which contains identifications and directions for how to use the nuclear material and nuclear technology application for peaceful use. Specifically, the regulation instrumentations for physical protection system (PPS) is based on the BAPETEN (Indonesia Nuclear Authority Board) Chairman Regulation Number 1 year 2009 on The Physical Protection System of Nuclear Installations and Nuclear Material. This technical guidance is adopted from IAEA international standard INFCIRC/225/Rev 5. BAPETEN is a competent authority for nuclear inspection which has mandate regularly to evaluate the PPS in nuclear facility in Indonesia.

From 2001 to 2014, there were three missions from IAEA- International Physical Protection Advisory Service (IPPAS) to visit Indonesia to determine security level and assess the implementation of physical protection at those nuclear research reactors. The third (3rd) IPPASS mission, which was in October 2014, recommended that the Indonesian government through the National Nuclear Energy Agency (BATAN) should establish assessment of the PPS design effectiveness. The PPS research is determined to be one of the priorities in nuclear research in Indonesia after the global trend of terrorism since 11 September 2001, which have possibility to attack and sabotage nuclear facility and or use the radioactive source as weapon. The based line of this research is following the already experience research in nuclear safety which has connection and similar method. So, the research in PPS in Indonesia optimistically has bright future for whom involved.

Indonesia has three nuclear research reactor facility and more than 700 facility related to nuclear technology application with different purpose as shown in Table 1.1. The nuclear utilization and facilities is not concentrating in the main city but also spread in many provinces in Indonesia as shown in Fig. 1.1. There are two main points of BAPETEN PPS inspection, the first is checking the documents related to management of facility that include in the PPS standard operating procedure, design basis threat documents, log book of maintenance and the second is verifying the performance of PPS parameters in detect, delay and response such as instruments testing, maintenance schedule, and the implementation of procedure. In fact, there is no single quantitative measurements has been done for evaluating the PPS in the scenario of adversary infiltration in facility [5]. In the other word, the integrity of PPS function due to the real scenario in facility has been evaluation yet. It is the main reason and become strong motivation for taking in advance the research of PPS in Indonesia. The first step of PPS research has been started by creating the appropriate tool for the development of PPS evaluation method that has capability to analysis PPS assessment quantitatively.

Table 1.1 List of Nuclear Facility in Indonesia[5]

No	Name	Type	Capacity	Application
1	RSG Swabesi	Research Reactor	30 MW	Radioisotopes Production, and Research.
2	TRIGA2000	Research Reactor	2MW	Research
3	Kartini Reactor	Research Reactor	100 kW	Research
4	Center for Sciences and Accelerator Technology	Accelerator Technology	2 Facilities	Research, and Industrial Application
5	Nuclear Industry. Corp	Nuclear Industry	2 Facilities	Radioisotopes Production, and Research.
6	Center for Nuclear Material Technology	Research Facility	1 Facility	Research
7	Center for Nuclear Waste Material Technology	Research Facility and Nuclear Waste Treatment	1 Facility	Research
8	Industries and Company	Nuclear Technology Applications	700 Facilities and Up	Industrial Application

Figure 1.1 Nuclear Technology Application Utilization Map[6]

1.3 RESEARCH OBJECTIVE

This study proposes the development of analytical software for assessing physical protection system. A clear statement of the objective of the research conducted in this thesis is presented as follows:

The objective of this study is to develop an analytical software to assess the effectiveness of the implementation of a physical protection system in a nuclear facility.

The scopes of this research work are focusing on developing the evaluation software for modeling the adversary path analysis as part of evaluation the effectiveness of PPS in a nuclear facility, conducting the evaluation for various structures or configurations of the designed PPS, and using the MATLAB program for creating PPS software. The outcome of this research is to obtain the software for evaluating the effectiveness of the PPS at a nuclear facility.

The research work has three main steps. The first step is preliminary research. It begins with the determination of the PPS objective. The facility characterization, threat definition, and target identification are the three factors to be considered in this step. Facility characterization and target identification are used to identify areas and what kinds of targets which need to be protected. Threat is defined as the potential entities with motivation, intention, and capabilities that commit for doing some malicious act. The output of this preliminary research is to generate the likelihood scenario of adversary attack into the facility. The schema of this step is shown in Fig.1.2.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

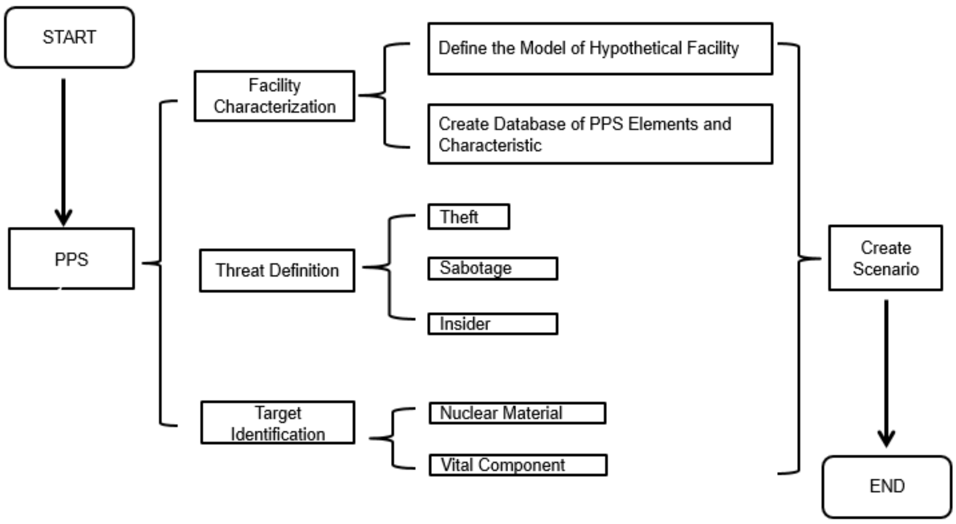


Figure 1.2 Schema of Preliminary Research

The second step is the analytical approach. It introduces the basic calculation method in evaluation PPS effectiveness. This method is very effective to calculate single path scenario. Fig 1.3 shows the analytical approach procedure for calculating probability of effectiveness. The first step is figure out the facility, paths, and PPS functions graphically in adversary sequence diagram (ASD). The next step is performed simple calculation process for measuring the PPS performance. Metrics of PPS i.e. probability of detection (Pd), adversary time delay (TR), response force time (RFT), and probability of neutralization, is required for calculating probability of intrusion and probability of effectiveness. Critical detection point (CDP) is others PPS parameter which is defined as the time difference between response force time and delay time. The relation among those PPS parameters may presents as the PPS timing diagram.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

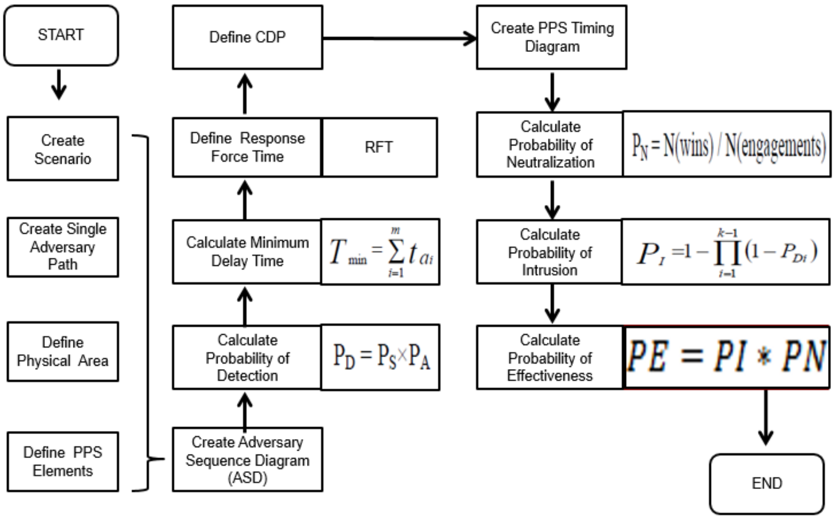


Figure 1.3 Schema of Analytical Approach

The third step as shown in Fig 1.4 is development process of adversary path analysis computer code. This step starts with writing the algorithm code based on analytical approach which use MATLAB language. This algorithm is the main components in the evaluation tool for computing probability of effectiveness. The next step is the black box of this research which consists of creating adversary path analysis model and GUI for creating model. The detail process in this black box is the main focus of this research work.

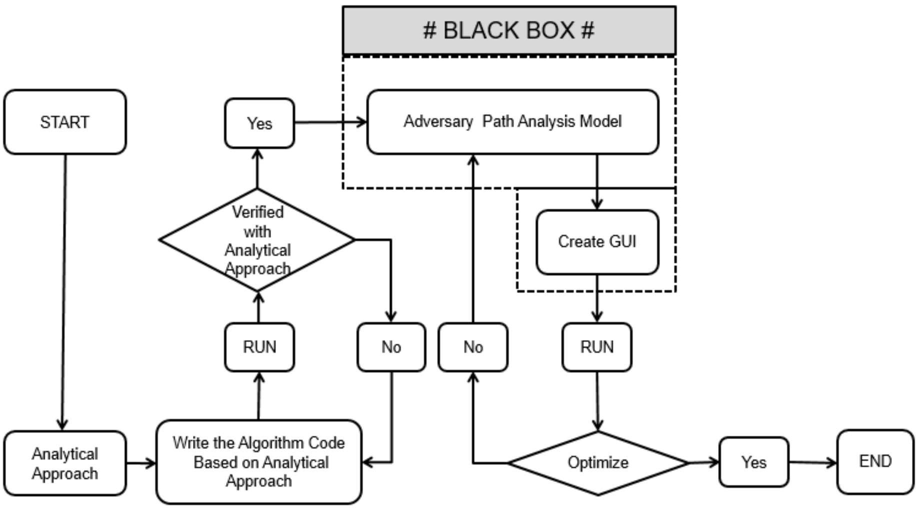


Figure 1.4 Schema of Adversary Path Analysis Computer Program

1.4 OVERVIEW OF THE MAIN SECTION

This thesis research document consists of five Chapters. Each chapter provides useful information to figure out the concept, methodology development throughout this research. Chapter I provides the background, motivation and objective of this study. Section 1.1 presents the global issue related to adversarial attack in 11 September 2001 event and its impact with the mindset of global security framework as the background of this research. The motivation of this research is stated in section 1.2 that figure out the current status of Indonesia national nuclear security framework. Section 1.3 establishes the objective of this research and described the blueprint of research planning for completing the objective of this research.

Chapter II section 2.1 is pursued the literature study of the previous research which have been done by PPS expert. This section presents the benefit of their research methodology which can be followed as the baseline of this research. Section 2.2 provides the theoretical background of PPS concept. It is continued in section 2.3 which discusses the concept of network graph to support theoretically the essential methodology used in this research.

The purpose of Chapter III is to discuss the PPS analysis tool software development. Modelling concept of this software is describe briefly in section 3.1. This section presents the general picture of this software development. The section 3.2 discusses how the network theory is used in this software development for representing adversary paths analysis. Section 3.3 discusses the calculation method which is used behind this software for measuring PPS performances. Section 3.4 describes the shortest paths finding algorithm. This algorithm has very essential function in this software for finding the kth-path. Section 3.5 presents programming code behind this software and Graphic User Interface (GUI) development. This last section provides rich information such as the guidance in how to operate this software.

Chapter IV provides the simple application of this software. The purpose of this Chapter is demonstrating the merely and capability of this software for analyzing the PPS performance due to adversary threat scenario, and verifying the analysis tool results with EASI tool. Section 4.1 and section 4.2 demonstrate the analysis of two simple practical example of adversary path scenario which adopt from previous study. The results of these sample are compared with the results from Estimate of Adversary Sequence Interruption (EASI) tool for validating the results from this analysis tool. Section 4.3 presents the example of complex PPS assessment in hypothetical nuclear facility. The last section 4.4 presents the limitation of this software and the future works. Chapter V is the last chapter which summarizes systematically the important parts in each chapter, concludes the results and discussion in short statements.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

CHAPTER II
THEORITICAL BACKGROUND

The concept for evaluation of the system effectiveness is to measure how well the three basic functions of the system (detect, delay, and response) perform to withstand a possible attack and prevent an adversary from achieving his objective. The objectives system evaluation are for verifying, that the PPS design can satisfy the requirements, identifying the system deficiencies, and improving the performance. This includes when the objective of PPS changes or the system has been upgraded, or after some accidental event, that have probability to influence the PPS design. The most effective of PPS should integrates the security devices such as sensor and barrier, security procedures including the security organization and response force procedure and facility lay out design[3]. The assessment of PPS can measure the effectiveness of the system for performing the objective of PPS in nuclear facility. The assessment results can figure out the vulnerable parts in the facility and all at once the necessary system improvement for strengthening the important parts or for relocating the overlap protection in the parts with lower security priority.

Generally, when evaluating the effectiveness, qualitative as well as quantitative analytical procedures might be used. The qualitative analysis of PPS effectiveness is applied in terms of security to less important buildings mainly, where the loss or damage to the assets is not that serious. Examples include the PPS of residential buildings, offices, smaller shops, restaurant, etc. The quantitative analysis of PPS effectiveness is then necessary, wherever the possible loss or damage to the assets has

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

generally undesirable consequences, even when assuming that the probability of an adversary attack is very low. It applies mainly to high-security systems in nuclear power plants, military installations, airports, banks, museums, etc.

There are two main perspectives in evaluating the effectiveness of a PPS. The first addresses adversary path analysis and the second deals with neutralization. The remainder of this section focuses on the quantitative analysis methods to evaluate system effectiveness for the PPS component. The concern in this research is with the former analysis. To determine the effectiveness of a PPS, path analysis is performed to evaluate adversary paths and the associated detection, delay and response timelines. The facility is characterized in terms of physical areas, protection layers, protection elements, path elements, path segments, and target locations. Each protection layer contains delay and detection protection elements that define the path elements and path segments of possible adversary paths. Due to the complexity of the system, it requires PPS performance measuring tools with to determine how well the performances.

Practical exercise such as force to force exercise is the best performance measuring tools. It closes to the real time scenario and provides the authentic PPS performance. However it is limited on the time of exercise, the paths and strategic of adversary intrusion attack scenario, and costly. The other one is the tabletop exercise. By using this tool, the simulation of several controlled environment scenario with only limited human resource can be implemented with ease. Although the results are not as well as previous scenario but it is less expensive and faster. The last tool is calculation method. For estimating or projecting the PPS performance, some mathematicians have been developed formulas to approach the condition. The formulas have taken into account in computer program and developed to be software with friendly user interface.

This type of tool has capabilities to calculate single path scenario with the related safeguard elements and even the analysis of many scenario with figure out the entire of PPS elements in a facility. The limitation of this tool are need additional efforts to verify the results and expert opinion to assess it as well. The most accurate results can be provided with only put the good data on it. However, this tool is prefer to use then others due to this tool is easy to customize and can display prominently the system deficiencies and at once the necessary improvement area[7].

2.1 LITERATURE REVIEW

There are several research related to this PPS effectiveness evaluation. The research of evaluation of PPS effectiveness for nuclear facilities have been performed since 1972 when the first original systematic schema of design and evaluation process (DEPO) of physical protection system was introduced by Sandia National Laboratory (SNL) in USA[8]. It has elaborated and integrated the three functions of design basic of physical protection system (PPS), detection, delay and response with emphasis on the people, equipment, and procedures to perform the objective of physical protection in a nuclear facility. This brief concept of DEPO of PPS will be discussed later in the section 2.2.

In the same research project, Sandia has introduced the computer based PPS design evaluation tool called "Estimate of Adversary Sequence Interruption (EASI)" that can calculate the probability of interruption based on simple data in a single adversary path. EASI works under the Microsoft excel application and very easy to use. There are simple steps to calculate. User needs to create a single path scenario, and input the probability of detection in each of element. The timing location of detection when the adversary pass through the barrier can be select with three options of detection location instantly such as in the beginning, middle or the end of action. Next step is input the probability of communication and response force time (RFT). The output of this tool is probability of interruption of adversary attack. The result can be interpreted from the adversary point of view and or respect to the PPS vulnerability status. The higher value means the adversary mission can be defeated or interrupted and the PPS is adequate enough to fulfill the objective of PPS, the lower one means the adversary can success to do the mission objective and the PPS status is vulnerable to attack with this scenario[8, 9].

In the year 2008, the researchers Of Korea Institute of Nuclear Non-proliferation and Control published a two-dimensional method "Systematic Analysis of Physical Protection Effectiveness" (SAPE). The method is based on the two-dimensional map model of an analyzed building. In order to find the most vulnerable path, a heuristic algorithm A* is applied. It serves to find the "worst" path in the evaluated graph leading from an input starting node to a final one. The method is also based on the analyses of timely detection[10].

Patrick T Hester, in his PhD dissertation research, has been develop a conceptual framework for computing system effectiveness and facility protection system optimization under uncertainty methodology for facility protection against a single adversary team attack. Based on his research, the adversary path analysis method has an advantage because the assessment is based on the performance of each of the elements in PPS design. However it also has a disadvantage because the assessment requires separate enumeration of candidate adversary paths. He elaborated two methodological approaches i.e. the adversary path analysis and network analysis, and formulated the methodology in algorithm for computing the effectiveness of the system[11]. This method is used in this thesis research and will be discussed deeply in the next subsection.

2.2 EVALUATION PROCESS OF PPS EFFECTIVENESS

As mention before, the DEPO elaborated and integrated the three functions of design basic of physical protection system (PPS), detection, delay and response with emphasis on the people, equipment, and procedures to perform the objective of physical protection in a nuclear facility. The process begins with find out the objectives of PPS. In this step, the designer must study what to protect based on facility characterization and information of likelihood threat. In the design step, the system is designed to meet the objectives. The three basic principles (detect, delay and response) are taken into account and must be embedded with the system. End of process is evaluation of PPS effectiveness. The aim of this process is to verification of how well the system performs compared to the objectives. This step is the main topic and discussed clearly in this research study. The schema of design and evaluation of PPS as seen in Fig 2.1.

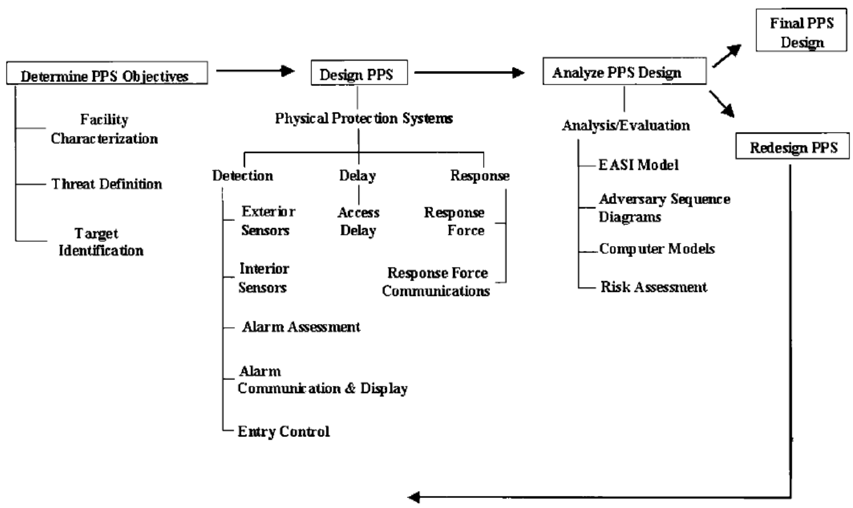


Figure 2.1 Schema of Design and Evaluation Process (DEPO)

One of the method to evaluate PPS effectiveness is adversary path analysis. The adversary path analysis is the quantitative method to measure the effectiveness of the system. This method introduced by Garcia. It focuses on the existence of adversary paths. An adversary path represents an ordered series of actions which has the successful criterion when the adversary completes the task (e.g. Theft or sabotage the facility). Fig 2.2 shows a sample for single adversary path to destroy a pump in an industrial facility.

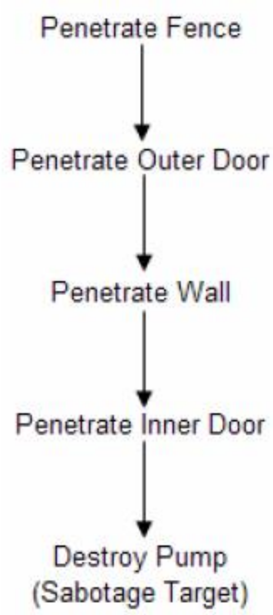


Figure 2.2 Sample Adversary Path

There are several steps to evaluate the effectiveness of the system using adversary path analysis. The first step is characterizing the facility area, defining the adversary path, and determining the PPS functions. The characterization of facility area in simple facility is shown in Fig 2.3. The adversary path is defined as physical route for the adversary to penetrate the target in facility and the time required completing the mission.

Along this path have points in the ordered series that provide PPS security elements i.e. detection and delay functions. The PPS functions are defined as the functions of system performance which include the probability of detection (Pd), detection times (Td), alarm assessment and communication, delay time in penetrating barrier, and time for the response team (TG) to interrupt the adversary.

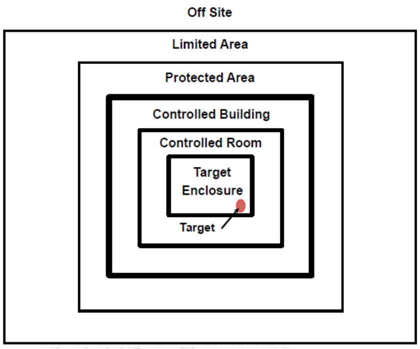


Figure 2.3 Characterization of Facility Area

The second step is modeling the facility, paths, and PPS functions graphically in the Adversary Sequence Diagram (ASD) as shown in Fig 2.4. There are three steps to make ASD. The first is modeling the facility area, the second is defining the path elements, and the last is assigning the probability of detection and delay time in each physical area and elements.

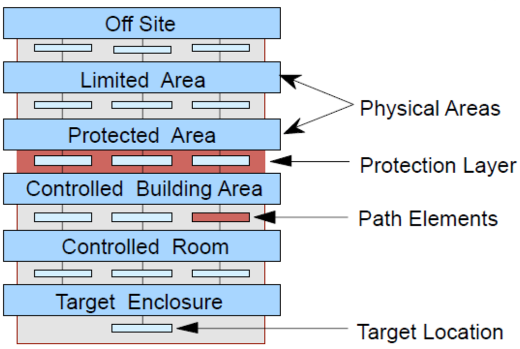


Figure 2.4 Adversary Sequence Diagram (ASD)

The last step is performing the calculation and analysis to measure the performance of the system. This performance of the system is measured by the probability of system effectiveness (PE), which is based on the system-level metrics to indicate whether the system is in good performance or not. The higher value of PE indicates the system has better performance than the lower ones. PE has two systems-level metrics that reflect how much detection, delay, and response are present. The first is the probability of interruption (PI) and the second is probability of neutralization (PN).

Probability of interruption depends on the probability of detection (Pd) of sensors, the detection time (To) of alarm systems, delay times (Td) for penetrate barriers, and the response force is probability of communication and response force time. To calculate the probability of interruption (PI), the critical detection point (CDP) must first be defined. The CDP is the last sensing opportunity along an adversary path such that the PPS response time remains less than the adversary task time remaining after the sensing. Based on this concept, the probability of interruption (PI) is the cumulative probability of detection along a path up to and including the Critical Detection Point (CDP).

Probability of neutralization (PN) is the probability that the response force can neutralize the adversary. PN can be estimated or calculated based on the expert judgments, force on force exercise or game simulation. The combination of PI and PN represents the probability of all functions to work properly or probability of effectiveness (PE).

CHULALONGKORN UNIVERSITY

2.3 GRAPH THEORY

In this subsection provides brief explanation of graph theory or sometimes called network theory interchangeably. In the previous research[11] has combined the network theory and adversaries path analysis as methodological approach for evaluation PPS effectiveness. Historically, the graph theory has been invented by a famous mathematician Leonhard Euler in 1736 in Königsberg, Prussia for solving and proving the possibility to walk and cross each of seven bridges at once and return to the starting point[12]. The graph theory and application has been used and developed in many application of science till nowadays.

Currently, the research in vertices and edges properties of graph has introduced new approach which significantly driven the availability of computers and communication networks data to be gathered and analyzed on a larger scale than previously possible [12]. This is an example of the capability of the graph methodological approach to store and transfer amount of data represented as a network in structural data analysis problems. The other research is related to the improvement of computing graph algebraic connectivity which has result in quicker communication through the network than computing node-connectivity or edge-connectivity for large networks. The results of this research is used in perimeter security application[13].

The simply definition of graph G is a collection of vertices V (also called nodes) and edges E (also called arcs or links) which connect the vertices. The general form of the graph G is represented as G = (V, E) which V={v1,v2,v3…vk} and E = (vi ,vj) where vi≠ vj and (vi ,vj) ∈ E are connected or adjacent [14]. Vertices or edges may have a variety of properties associated with them. According to the direction of flow, the

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

graph defines to be directed and undirected graph. Directed means the edges may flow the data by follow certain direction as represented by arrows. Undirected means the edges may not have orientation or unordered and the data can flow freely in both directions.

A walk in the graph G = (V, E) is a finite sequence of the form vi0 , ej1 , vi1, ej2, . . . , ejk ,vik, which consists of alternating vertices and edges of G. The walk starts at a vertex. vi0 is the initial vertex and vik is the terminal vertex. k is the length of the walk. A path is a walk without any repetition of nodes and need to be alternately followed in order to move from a start node to an end node.

The graph may be represented by a two dimensional adjacency matrix. If G has number of vertices n, let A is an n by n matrix. Then Aij = 1, if there is connection between vertices (i , j) and Aij = 0 otherwise. The adjacency matrix has n2 elements, only m of which are nonzero. Consequently, this representation is space efficient only if the network is sufficiently dense; for sparse networks this representation wastes considerable space. Nevertheless, the simplicity of the adjacency representation permits us to use it to implement most network algorithms rather easily. As seen in Fig 2.5 is an example of a directed graph where V ={1,2,3,4,5,6,7} and E={(1,2),(1,3),(2,3)…(6,7)}. The walk is W={1, (1,2), 2, (2,3)…,7}, a path is P={1,3,6,7}. Fig 2.6 is the example of undirected graph where the edges flow is permitted in the both side E{(1,2),(2,1),(1,3),(3,1)…(6,7),(7,6)}. The adjacency matrix of the directed graph in Fig 2.5 is shown in Fig 2.7.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

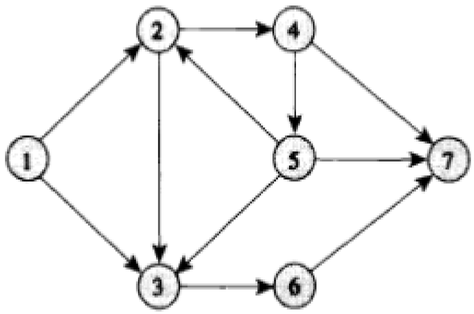


Figure 2.5 Directed Graph.

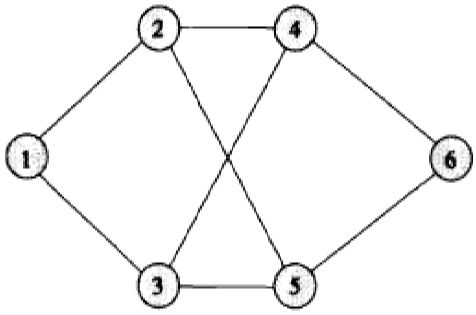


Figure 2.6 Undirected Graph

$$
A(G) = \begin{pmatrix}
0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\qquad \text{Where } A_{ij} = \begin{cases} 1 & \text{if } (i, j) \text{ as an edge} \\ 0 & \text{otherwise} \end{cases}
$$

Figure 2.7 Adjacency Matrix

Each of edge (i , j) in the graph has capability to store data or weighted that represented as aij. The edge has weight if aij > 0 and aij = 0 if unweighted. These data may represent the relation properties between vertices, it may cost, distance. For instance, if vertices 1 to 2 represent two different locations, the data or weight may represent the distance of this locations.

CHAPTER III
PPS SOFTWARE ANALYSIS TOOL DEVELOPMENT

3.1 MODELLING CONCEPT

There are three main points in this PPS software analysis tool development. The first part is how to characterize the facility and interpret the adversary path using network methodological approach. The second is how to define the available safeguards option in the network and perform the calculation for measuring the PPS performance. The third is how to build the Graphical User Interface (GUI) and write the codes of this software using the MATLAB. In this first subsection describes the general concept of this main parts and it will be discussed in detail in following separate subsection.

The main concept in the first part is based on DEPO schema. One of the requirements for obtaining the objective of PPS analysis is facility characterization. In this software, this function is solved by the network methodological approach. Network properties such as vertices, edges, weight, network adjacency matrices is created as analogous of adversary path and map of installed safeguards element in a facility. Other function related to this part is network analysis for determining and ranking the most vulnerable path in facility. Due to this last function is rated as the new development technique for analyzing PPS effectiveness, this part is the virtue of this research.

Although as mention before that the developing of this software attempt to makes an improvement by using network methodological approach for determining the vulnerable path, but the calculation methodological approach or formula to measure the

PPS performance behind this tool is mostly referred to EASI tool. It is reasonable due to this tool is the basic state of most of current PPS complex analysis tool and the simplicity of this calculation tool which can be adopted in many practice way. Since EASI has a disadvantage to perform multiple adversary path with several scenario, and the results are unable to find the most vulnerable in each of paths, the network method is covered this leakage.

In this research, the development of PPS analysis tool or PPS ETOOL is programmed by using MATLAB. In the fact that MATLAB is used by a large number of people, it provides a large amount of online free accessed set of network analysis toolboxes in m--files format. It makes the effort to design software with a large amount of algorithms in MATLAB easily by putting and dragging the file to use. In the other hand the MATLAB language enhances the code development because it supports matrix and vector operations. However, network analysis application in MATLAB has limitation due to the small memory capacity in MATLAB which influence the size of networks and the speed of the calculations. The visualization in MATLAB is not as good as others programmed but proper enough for representing the analysis data.

3.2 ADVERSARY PATHS REPRESENTATION

The first step in this software development is construct the paths network as a simple representation of a facility and describe likelihood adversary action through the arrangements of safeguard elements in the facility. Generally, this concept is same as in Adversary Sequence Diagram (ASD) but the network diagram may represent the connection better with some of advantage in find the most vulnerable path and determine the critical point detection. The previous research[11] has introduced this concept of network as connectivity diagram (CD) for representing the relationship between the elements or actions. As shown in Fig 3.1, the vertices represent the physical location such as outside facility, fence, outer door and wall, or the adversary milestone such as start point and target. The existence of an edge in between two vertices may represent the connection between two physical locations or method of adversary action such as penetrate fence and outer door. The safeguard elements properties such as probability of detection in sensors and delay time of delay elements represent by number of network capacity or weight value in edge.

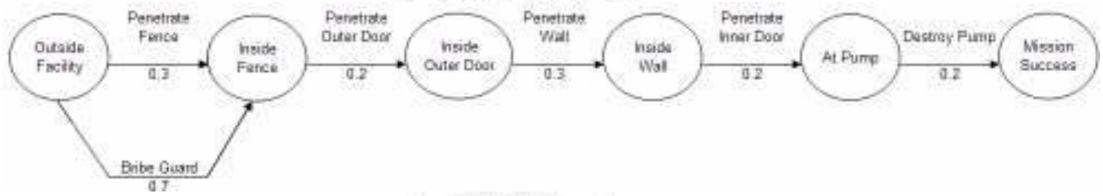


Figure 3.1 Adversary Path Representation

Furthermore, in Fig 3.2 is shown Adversary Sequence Diagram (ASD) 2D Map of a sample facility. Both represent the same physical protection system and the same adversary path physical protection system which already known well for displaying PPS of a facility. Fig 3.3 shows the sample of network for representing adversary path. This network has the same function as ASD. The correlation between network components and its representation in PPS analysis is shown in Table 3.1.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

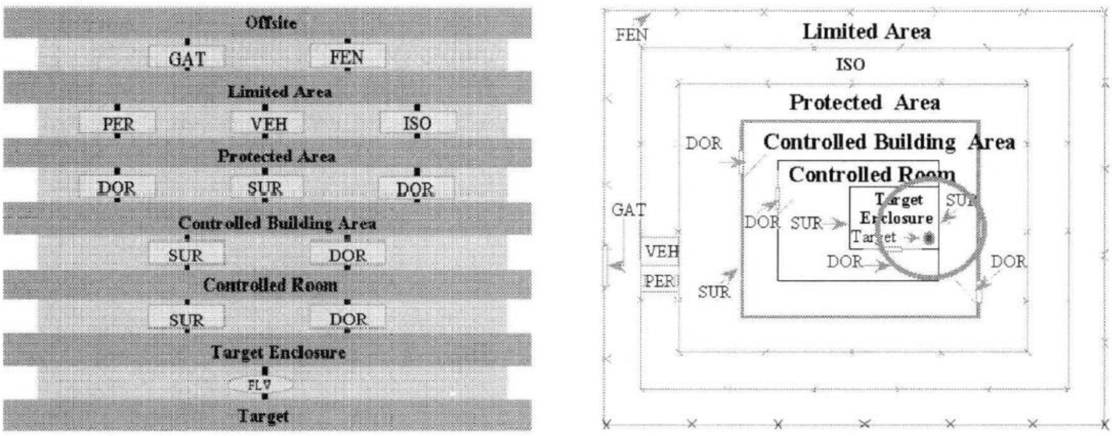


Figure 3.2 Adversary Sequence Diagram and 2D Map of Sample Facility

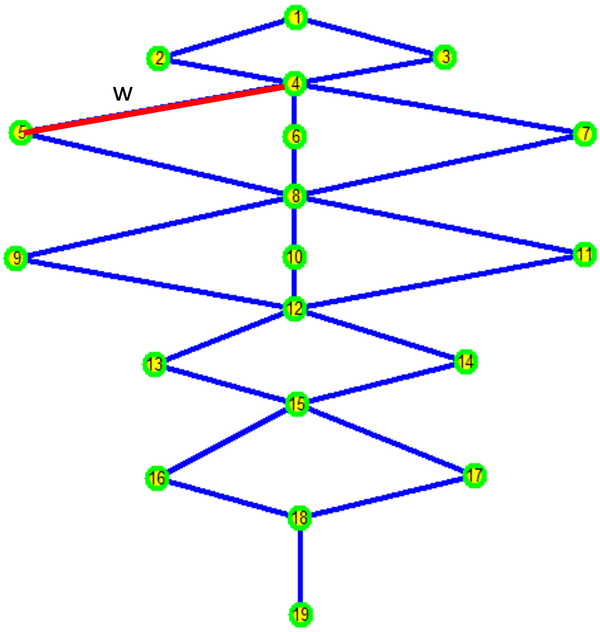


Figure 3.3 Network Diagram as Representation of Sample Facility

For the remainder of this study, the graph G = (V , E) denotes a directed network with vertices set V and edges set E. Edges are referred by a single index i or j. An adversary penetrates the kth - path in the ordering of actions or tasks from start point (s) through a sequence of safeguard elements such as detection sensors and barriers, or just transit in a point, then arrives in the target (t) for completing the mission.

If there is an adversary walk at a path k throughout s and t (with s ≠ t) indicates origin and destination, respectively , index of the current edge is i and the previous one is j, where i,j ∈ E. In Fig 3.3, a path is k1 = { V1 – V2 - V4 – V5 - V8 – V9 - V12 – V13 - V15 – V16 - V18 – V19 } where the start point is vertices 1 (V1) and end point is V19 . Alphabet w in the upside of the current edge is weight value that may represent cost, distance between vertices or probability of detection due to safeguard elements installed in this location.

Table 3.1 List of Vertices and Edge in Network Diagram of Sample Facility

No Vertices	No Edge	Name	Action
1		Offsite	
	1-2		Penetrate Gate
	1-3		Penetrate Fence
2		Inside Gate	
3		Inside Fence	
	2-4		Penetrate Limited Area From Gate
	3-4		Penetrate Limited Area From Fence
4		Inside Limited Area	
	4-5		Penetrate Vehicle Entrance
	4-6		Penetrate Personal Entrance
	4-7		Penetrate Isolation Zone
. . .	. . .	. . .	. . .
18		Target Enclosure	
	18-19		Penetrate Floor Vault
19		Target	

3.3 PPS PERFORMANCES MEASUREMENT

There are two measure has been developed for assessing the performance of the system in the term of effectiveness to withstand against adversary attack scenarios. In the first case such in theft or stealth scenario, may be approached by measuring minimum cumulative probability of detection (Pmin) in the point along adversary time lines before the goal is achieved. The acceptable value of Pmin can be achieved by having an adequate protection in the system. The disadvantage of this measure is that no consideration of delay is involved. Detection without sufficient subsequent delay is not effective because the response force may not have sufficient time to interrupt the adversary.

The second likelihood scenario is sabotage or force attack scenario. This tactic is measured by compared the minimum cumulative time delay along the path (Tmin) to guard response time (TG). For an effective system, TG must be less than Tmin. System improvements are achieved by decreasing TG or by adding protection elements with more delay to increase Tmin. The disadvantage of this measure is that no consideration of detection is involved. Combining of these two method, cumulative probability of detection (Pmin) and delay time (Tmin, and TG) defined as timely detection concept. For measuring PPS performance in this software follows the last methodological approach which has developed by Sandia National Laboratory in EASI tool.

The concept of timely detection as shown in Fig 3.4 describes that system effectiveness is measured by the cumulative probability of detection (PI) at the point where there is still enough time remaining for the response force to interrupt the adversary. This point defined as Critical Detection Point (CDP). Enough time remaining means response force time (TG) should be quicker than time remaining for the adversary to complete the mission (TR).

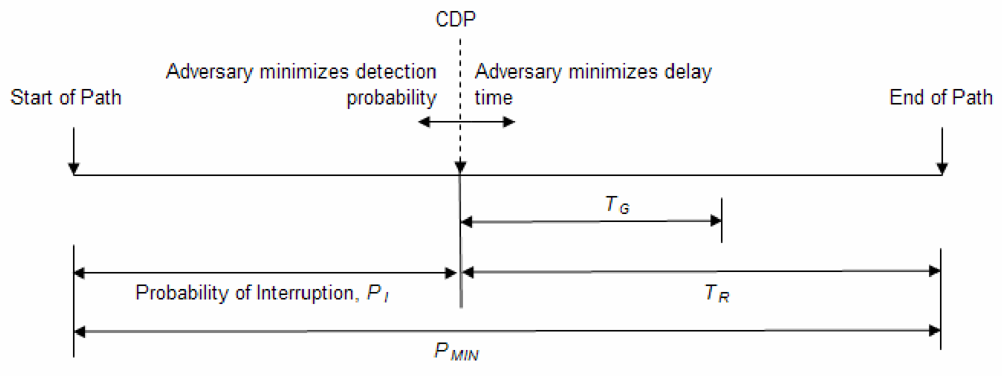


Figure 3. 4 Concept of Timely Detection

The steps to calculate PPS performances in this software describe as follows

1. Calculate Probability of Detection (PD).
2. Calculate Adversary Time Remaining (TR) at Certain Point
3. Calculate Probability of Response Force Arrival P(R I A).
4. Calculate Probability of Interruption (PI).
5. Calculate Probability of Effectiveness (PE).

3.3.1 Probability of Detection

Probability of detection (PD) is product of intrusion detection systems. Intrusion detection is defined as the detection of a person or vehicle attempting to gain unauthorized entry into an area that is being protected by someone who is able to authorize or initiate an appropriate response[8]. It consists of intrusion detection system such as exterior and interior intrusion sensors, video alarm assessment, entry control, and alarm communication systems. It should be installed along the adversary path for gaining optimum detection. The ideal detection system has PD value one (1.0). It means the performance of this system is 100% may detect the adversary movement. However, PD is always less than 1 due to the imperfection of the system. This value is depend on others parameter such as type of adversary method to do action (i.e., a walking, running, or crawling intruder; tunneling, etc.), sensor hardware design, installation conditions, sensitivity adjustment, weather conditions and condition of the equipment.

In this research, the PD values for each of safeguard elements instantly can be used in analysis. It has assumption that this value is obtained from the trust research center in PPS design. However, The PD values in particular edge may be varied depend on selection method of adversary in perform the action and type of detection system installed. Considering the purpose of installing detection system for detecting as early as possible the adversary intrusion, the calculation of PD in this research is performed in each particular location before target point in a kth-path. The formula is given as follows

$$P_{Dik} = \left[1-(1-P_{Do}) \prod_{l \in S}(1-P_{Dli}\ y_{li})\right] \prod_{j=1}^{i-1}(1-P_{Dj}\ x_{jk}) \ldots\ldots.(1)$$

PDik is the total probability of detection in current edge i on the kth-path. PDO is the total probability of detection due to the un-interdicted (inherent) detection. This probability is caused by movement of adversary for example the sound when the door is opened. S is defined as set of safeguard types and l is safeguard element. Then PDli is probability of detection due to safeguard element type l in current edge i. yli is binary variable as an indicator whether or not a safeguard of type l is installed. Since the probability of detection in the previous edges may have contribution for increase the probability of detection in kth – path, so per-edge value is then normalized by the probability that the adversary has not been detected until the current edge. PDj is the probability of detection across edge j, and xjk is a binary variable indicating whether or not flow travels across edge j on kth - path.

3.3.2 Adversary Time Remaining (TR) at Certain Point

TR is the time remaining for the adversary to reach the end point of sequence when a sensor activates. All of the performance evaluation at certain point along kth – path is addressed in and focused on this end point due to the objective of PPS is for protecting this target point. The time remaining at current point i in the kth - path is defined as follows

$$TR_{mk} = \sum_{m=i+1}^{j} [\ Tm + \sum_{l \in S} T_l\ y_{lmk}\ ] + \theta \sum_{m=i}^{i} [T_m + \sum_{l \varepsilon S} T_l\ y_{lmk}] \ldots\ldots.(2)$$

Assuming each task to be independent, the variance of the path time remaining between point m and the terminal point j is calculated by:

$$Var\ (TR_{mk}) = \sum_{m=i+1}^{j} Var\ [\ Tm + \sum_{l \in S} T_l\ y_{lmk}\ ] + \theta \sum_{m=i}^{i} Var\ [\ Tm + \sum_{l \in S} T_l\ y_{lmk}\ ] \ldots.(3)$$

Where Tm is the adversary travel time to penetrate edge m, Tl = delay time due to performance of installed barrier or safeguard elements type l. ylmk is binary variable as an indicator whether or not a safeguard of type l is installed.□ ɵ is the location in the current task at which detection occurs, ɵ = 1 indicates that detection occurs at the beginning of the task. ɵ = 0 indicates that detection occurs at the end of the task. For location in middle of task in time remaining (the first equation) ɵ = 0.5, and for variance ɵ = 0.25.

3.3.3 Probability of Response Force Arrival (PR I A).

Probability of response force arrival (PR|A) is defined as the probability of response force (R) to interrupt an adversary task (A) after the first alarm notification activated. Probability of response force arrival (PR|A) is the function of response force time (TG) and adversary remaining time (TR) after detection point (CDP). For successful of adversary interruption, it is necessary that

TR - TG > 0 …………………………………………………………………….(4)

Since TR and TG has standard deviations, it can be assumed that TR and TG as sums of random variables (X) and normally distributed. It is defined as follows:

X = TR – TG…………………………………………………………………......(5)

X is normally distributed with mean and variance as follows :

$\mu_x = E(TR\text{-}TG) = E(TR)\text{-}E(TG)$ ………………………………………..(6)

$\sigma_x^2 = Var(TR\text{-}TG) = Var(TR) + Var(TG)$…………………………...(7)

Then $P_{R|A}$ is calculated as follows

$P_{R|A} = P(TR\text{-}TG) = P(X > 0)$…………………..………………………(8)

$P_{R|A} =$

$\int_0^{\sim} \frac{1}{\sqrt{2\mu\sigma_x^2}} \exp\left[-\frac{(x-\mu_x)^2}{2\sigma_x^2}\right] dx$……………………………………………..……(9)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.3.4 Probability of Interruption (PI)

Probability of interruption is defined as the probability that the response force will be in the right place and have enough time to interrupt an adversary before he or she raises the target or completes the task to thief or sabotage. Specifically, it is the function of the probability of detection, the delay time, and the response force arrival time. The response force will be notified by an alarm when there is sufficient time remaining in the sequence for the force to respond. The probability of alarm (PA) due to the activation of single detection system with PDi as probability of detection and PCi as probability of communication to the response force is defined as follow :

$P_{Ai} = P_{Di} * P_{Ci}$ ……………………………………………………(10)

If (PR|A) is defined as probability of response force arrival The probability of interruption in this case is calculated as follows:

$PI = P_{R|A} * P_A$ ....(11)

For the cumulative probability of interruption along the adversary's path from the starting point, must consider detection at the first location, at the second, and so on. However PC1 is included in the first term only because if we do detect at the first location, but do not communicate to the response force based on that detection (due to jamming, etc.), we will probably not get a second chance to communicate at the second location just by the virtue of being detected there. (The probability of this event is P (D1) * (1 – PC1)), which represents the difference between P(D1)*P(C1) in the first probability term and P(D1) used in the first part of the second probability term in. The general formula for P(I) based on similar reasoning is

$PI_{ik} = PA_1 * PR_{|A1} + \sum_{i=2}^{n} PA_i * PR_{|Ai} \prod_{j=1}^{i-1}(1- PD_j)$ ………………(12)

n is the number of elements (edges) on the kth path, and the probability that a response force (R) dispatched at the time of detection at element i interrupts the adversary before his task (Ai) is P(R|Ai). Due to this detection and interruption occurs only when all the previous detection opportunities fail so the joint probability of detection failure from previous edges (PDj) is taken into account. The lowest probability of interruption may indicate the path is the most vulnerable in the system.

3.3.5 Probability of Effectiveness (PE)

Calculation of probability of system effectiveness (PE) is the last step for measuring the performance of the system. PE is one of indicator whether the system is in good performance or not. The higher value of PE indicates the system has better performance than the lower ones. In this research, probability of effectiveness (PE) in kth - path is the function of probability of interruption (PI) and probability of neutralization (PN) which can be put as following:

$$P_{Ek} = \sum_{i=0}^{n} \left[P_{Iik} * P_{Nk/TDk}\right] x_{ik} \text{ ....(13)}$$

Probability of neutralization (PN) is the probability that the response force can neutralize the adversary. PN can be estimated or calculated based on the expert judgments, force on force exercise or game simulation. Due to the complexity of analytical calculation, the probability of neutralization in this research is obtained from the available analysis tool developed by Garcia.

3.4 SHORTEST PATHS FINDING ALGORITHM

The problem of adversary path analysis in the context of paths finding in network is how to find and analyze the probable path that can be used by adversary to achieve the target. This case is known well in the network theory as path analysis problem. One of the famous problem is find the shortest path through a network. It is important to realize that the term 'shortest' does not just apply to distance between vertices, but can be implement in one or more factors, including cost, metric performance of equipment, or time that put a weighting on the route. K-shortest path (KSP) algorithms are thus widely used in the fields of data structure analysis, computer science and transportation science.

The shortest path through a network is the minimum cost used to travel between a vertices to another vertices, and this path will usually be the preferred route between those two vertices. When the shortest path between two vertices is not available, it is necessary to determine the second shortest path. If this too is not available, a third path

Chulalongkorn University

may be needed. The sequences of path which is arranged from the least length thus derived are known collectively as the k-shortest path (KSP), and represent the first, second, third, till kth –path.

In order to calculate the KSPs, it is necessary to force the algorithms to choose different paths through the network. This is conveniently done by marking the vertices and edges as not being allowed in further path calculations (between the two specified nodes). This operation is often referred to as removal of vertices and removal of an edge respectively. Edge marking is implemented by setting edge to have infinite length in some algorithms. This means that they will never be selected to form a path (or if they are, all valid paths have already been found) a path of infinite length is useless.

In this research, Yes's algorithm is used for solving finding the all of kth - shortest path problem in adversary path analysis. The shortest path is found using a standard shortest path algorithm and placed in the result list (Yen's list A). Yen takes every vertices in the shortest path, except the terminating vertices and calculates another shortest path (spur) from each selected vertices to the terminating vertices. For each such vertices, the path from the start point to the current vertices is the root path.

Vertices and edge marking is used to prevent the spur paths from looping or simply following the route of a previous k-shortest path. If a new spur path is found, it is appended to the root path for that node, to form a complete path from start to end node, which is then a candidate for the next KSP. All such paths are stored (Yen's list B) and the shortest remaining unselected path is selected as the next KSP, and transferred to the result list (Yen's list A). The same process is repeated, calculating a spur path from each node in each new KSP, until the required number of KSP have been found.

3.5 GRAPHICAL USER INTERFACE (GUI)

A graphical user interface (GUI) is a graphical display in one or more windows containing controls, called components such as menus, toolbars, push buttons, radio buttons, list boxes, and sliders which is enable a user to perform interactive tasks. MATLAB provides two application program for design GUI.

The first application is GUIDE (GUI Development Environment), an interactive GUI construction kit. This approach starts with a figure that you populate with components from within a graphic layout editor. GUIDE creates an associated code file containing callbacks for the GUI and its components. GUIDE saves both the figure (as a FIG-file) and the code file. Opening either one also opens the other to run the GUI.

The second is create code files that generate GUIs as functions or scripts (programmatic GUI construction). Using this approach, you create a code file that defines all component properties and behaviors. When a user executes the file, it creates a figure, populates it with components, and handles user interactions. Typically, the figure is not saved between sessions because the code in the file creates a new one each time it runs. The last method is used for design GUI in this research.

Design GUI for PPS evaluation analysis tool in this research is concerned with three parts as follows:

1. How to generate networks.

2. How to perform PPS calculation

3. How to design GUI components

Due to design GUI is integrated in a process with generate network and create PPS calculation codes, the discussion of design GUI components is presented together with the first two topics. The following discussion is started by introducing PPS analysis tool main interface.

The GUI application window, as seen in the Fig 3.5, is launched by open the file name PPS_ETOOL.m with MATLAB. In MATLAB, The application window is categorized as main Figure which is allowed the user to add more panel and toolboxes as interface to the program code behind. As default, it contains standard menu in MATLAB for the File management, Edit, Save, Tools and Help. The main figure is important part in GUI due to its functioned as a parent in this program where all of data structure address in. The following is the code to create this main figure.

```
%-----------------------------------------------------------------------------------------------------------------
        h.f = figure ('Visible','on','Position', [30, 5, 1300, 700]);
%-----------------------------------------------------------------------------------------------------------------
```

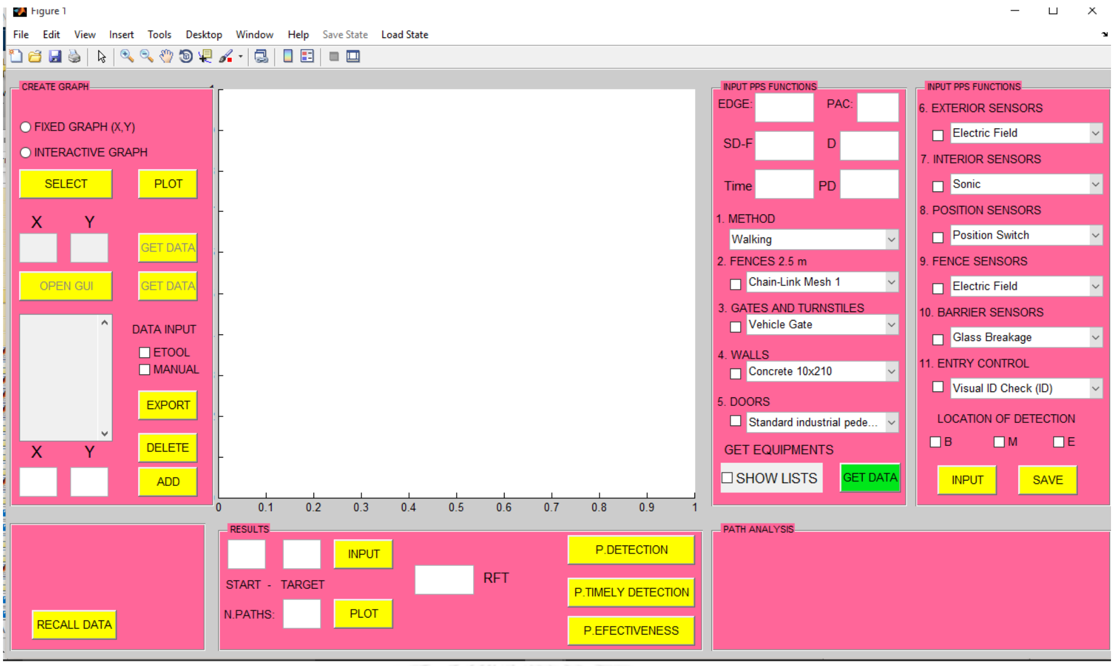


Figure 3.5 Main Application Window of PPS Analysis Tool.

3.5.1 Generate Network

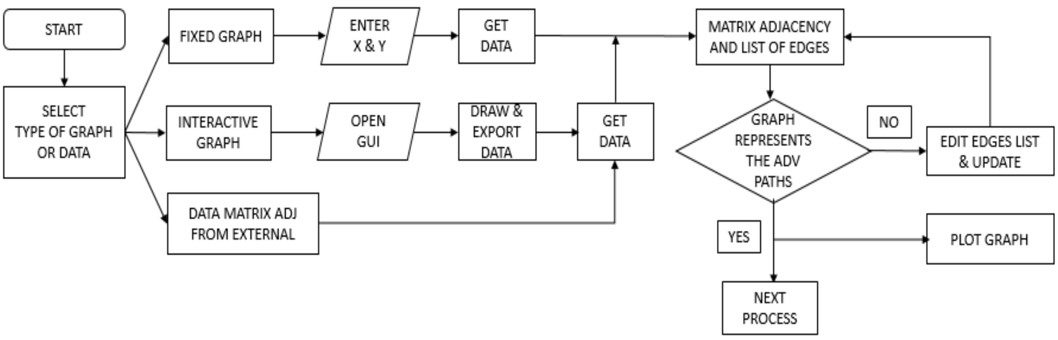


Figure 3.6 Flow Diagram of Generating Network

The flow diagram of generating network in Fig 3.6 shows the starting point to generate a network is by selecting type of graph. There are two methods available i.e. fixed graph and interactive graph. Other method to generate graph is use external data in the form matrix adjacency. Fixed graph is generated by enter the X and Y value in the textbox as number of rows and coulombs. For example in the Fig 3.7 is a two and two (2 rows x 2 coulomb) fixed graph.

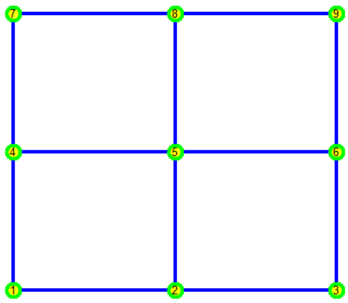


Figure 3.7 Grid Model Graph

This graph uses function code as follow:

```
%----------------------------------------------------------------------------------------------------------------
   [X, Y] = gen_mesh(R, C);              % create mesh with input Row and Colom
      A    = gen_adjacency(X, Y, 1);     % gen_adjacency matrix
    Asp   = sparse (A);
%----------------------------------------------------------------------------------------------------------------
```

Interactive graph or dynamic graph is graph which draw by independent GUI specialized for network drawing. It has advantage to draw graph freely and easy to use. It is recommended for analysis the real adversary path scenario in facility due to it is flexible to draw.

Chulalongkorn University

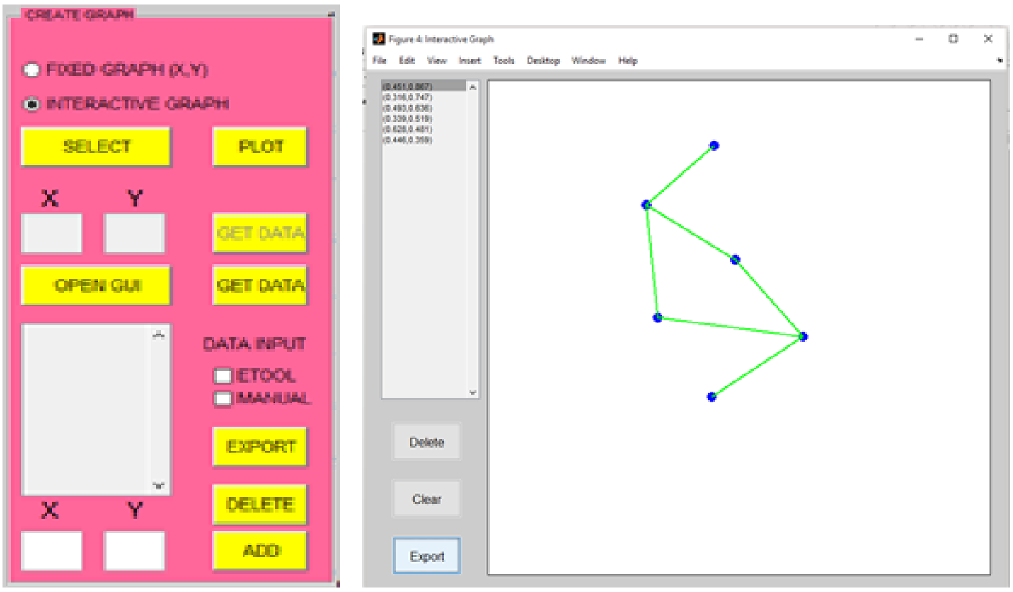


Figure 3.8 Network Generation Tool Box (Left) and GUI Interactive Model Graph (Right)

The interactive graph is opened by pressing OPEN GUI button. The user may interact with this GUI by using mouse. Left click creates one vertices and right click connects pair of vertices. The direction of edge follow the order of click. User can edit the graph with delete and clear button. The output of this GUI are matrix adjacency and xy location. Export button sends this output matrix to MATLAB workspace. This data called by GET DATA button in PPS ETOOL for further analysis.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

The output of this process is matrix adjacency and list of edges as standard graph properties. Since the graph represents the path adversary, it is necessary to check whether the graph provides the same path as scenario. This tool is designed to compromise with this thing. The edge of graph can be edited by adding or deleting an edge. When adding the edge, the user needs to fill text box X and Y as the first and second vertices before press add button. In the other hand for deleting ones, the user may select only edges list in Edges List Box and press delete button. The concept of code behind this toolbox is just change the adjacency matrix property from 1 if pair of vertices has connection and 0 if other than replot the current adjacency matrix for updating the display.

The graph is displayed as connecting lines between pair of vertices. It is plotted based on xy positions of vertices in the general axes coordinate. The connection of vertices is followed the graph adjacency matrix (Af) in the form of sparse matrix (Asp). The plotting function is shown as follows.

```
%----------------------------------------------------------------------------------------------------------------
                        Asp  =  sparse (Af);
            edge_list_prime  =  plot adjacency (Af,X, Y);
%----------------------------------------------------------------------------------------------------------------
```

3.5.2 Input Data Processing

The next step is input data processing. The data is defined as metric of performance in each of PPS elements in the system, such as probability of detection, time delay, response force time and so on. The flow diagram in Figure 19 shows the process of input data in this analysis tool. After select the edge, the next process is input the data one by one and store the data in memory as weight value. This value is represented as matrix of weight that can be used in the next analysis. In the following Fig 3.9 shows the user needs to input some primary data i.e. probability of communication, standard deviation factor, length of edge as a distance, detection location, and response force time. Others data such as probability of detection and delay time due to delay elements have option to input by using data from database (e-tool) or manually. This option menu is shown in Fig 3.8 placed in above export button .

The next process is selection one of the method of adversary to perform an action in current edge. In the adversary method list box is listed twelve common methods which is used during movement or intrusion to PPS of facility. The methods are walking, running, crawling, rolling, jumping, tunneling, and trenching, bridging, cutting, climbing and land vehicle. In adversary point of view, selection of this method plays important role for completing the mission due to the contribution of this method to time travel and likelihood of being detected along the way.

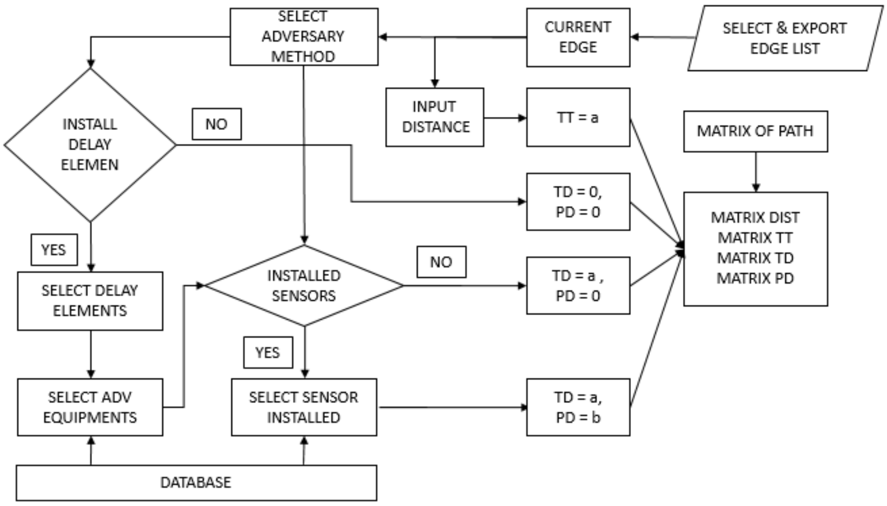


Figure 3.9 Flow of Input Data Processing

จุฬาลงกรณ์มหาวิทยาลัย

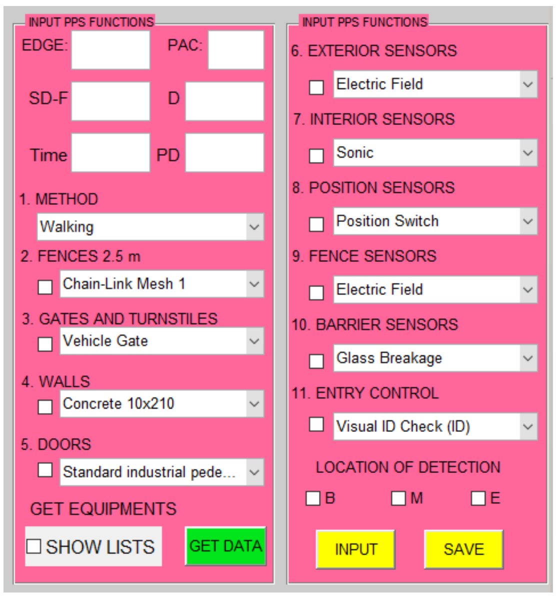


Figure 3.10 Interface of Input Data Processing

As mention before, there are two ways for input safeguard elements i.e. manually by user defined or using the PPS database data. In the former way, the user just input delay time and probability of detection value in textbox which is used directly for analysis. The second option is the user may select delay and detection elements which is provided in this tool. In Fig 3.10 shows eleven items of delay a detection elements. In delay elements toolbox, the user may select only one type of element in five groups available i.e. fences, doors, gates and turnstile, walls, and doors. Some of appropriate equipment in performing action are also available for selected. In the following Fig 3.11 is list of selected equipment tool in Get Equipment Toolbox.

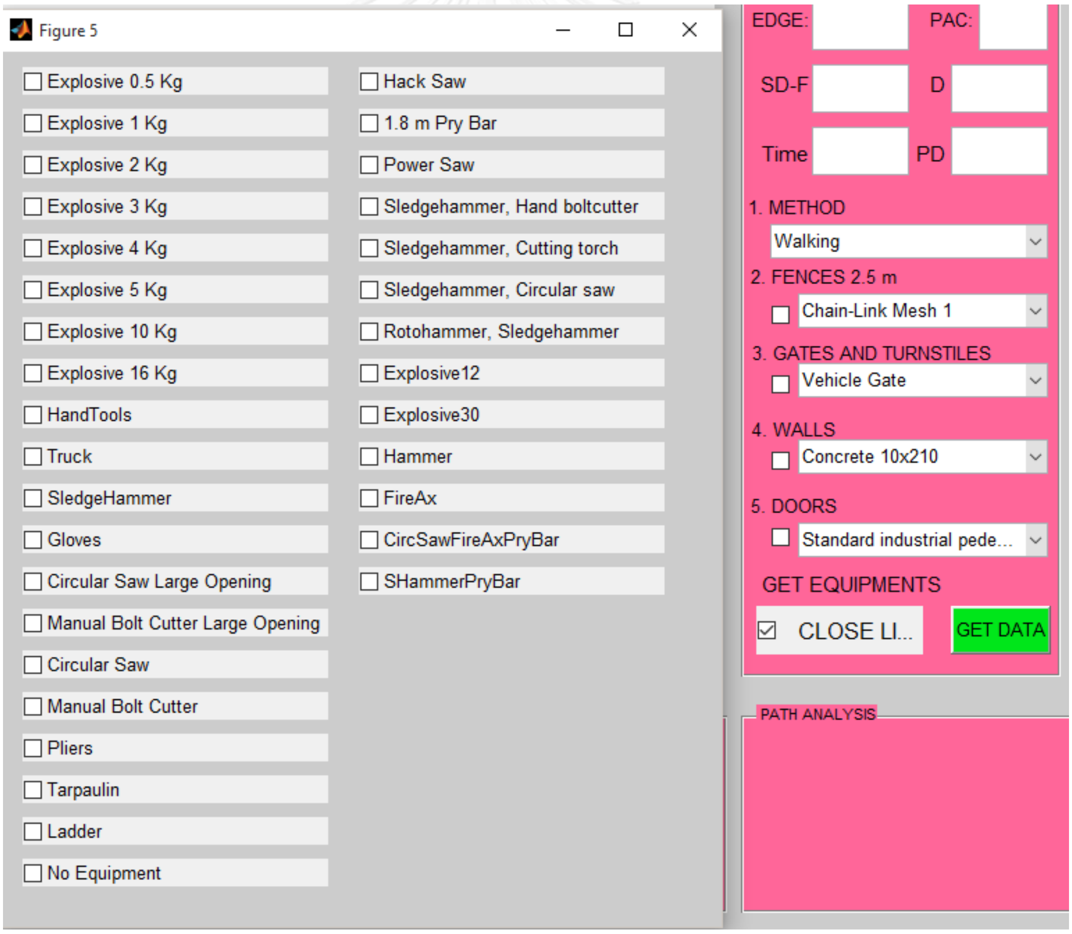


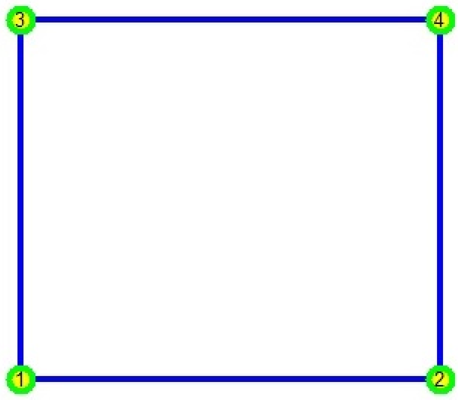
Figure 3.11 Delay Elements Toolbox

In detection elements toolbox provides list of available detector system. It is divided into six groups based on location in the system i.e. exterior sensors, interior sensors, position sensors, fences sensors, barriers sensors and entry control system. The probability of detection which is used in analysis is cumulative value due to detection sensors and method of adversary if using the equipment. It is calculated as follows:

```
%---------------------------------------------------------------------
x = length(s);                      % s is number of sensors
Pdprime = zeros(1,x);
z=zeros(1,x);
Pdo = 0;                            % PDo is assumed 0

for   i=1:x;
      z(i) = 1-s(i);                % P1 = 1- PD1
  PDiPrime = 1-(1-Pdo)*prod(z);     % Left side of equation 1 and
                                    % still need to count with
                                    previous arcs
                                    % after Path analysis.
end
%--------------------------------------------------------------------
```

All of data input are in matrix form. The data is located based on location of edge in matrix. As an example in data which is input in edge 1 – 2 is located in V1 – V2. In Fig 3.12 shows variation of matrix data graph G. A is matrix adjacency, TD is matrix of time delay, TDsd is matrix of standard deviation time delay, PD is matrix of probability of detection. These matrix is ready to use in calculation process in this analytical tool.



Graph G

$$A = \begin{array}{c} \\ V_1 \\ V_2 \\ V_3 \\ V_4 \end{array} \begin{array}{c} \begin{array}{cccc} V_1 & V_2 & V_3 & V_4 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$TD = \begin{bmatrix} 0 & 30 & 50 & 0 \\ 0 & 0 & 0 & 55 \\ 0 & 0 & 0 & 25 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad TDsd = \begin{bmatrix} 0 & 9.0000 & 15.0000 & 0 \\ 0 & 0 & 0 & 16.5000 \\ 0 & 0 & 0 & 7.5000 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad PD = \begin{bmatrix} 0 & 0.4750 & 0.6650 & 0 \\ 0 & 0 & 0 & 0.5700 \\ 0 & 0 & 0 & 0.8550 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 3.12 Example of Matrix Data of Graph G

3.5.3 Database System

There are two database in this software i.e. probability of detection database and time delay database. All of data in this database is assumed as performance metric of components from product manufacture and ready to use. Following in the Table 3.2 is the example of raw data probability of detection in different type of sensors according to the adversary method.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Table 3.2 Probability of Detection Different Type of Sensors Due To Adversary Method.

Intruder	**System Type**					
	Electric Field	Microwave	Active Infrared	Passive Infrared	Fence Motion	Taut Wire
Walking	VH	VH	VH	H	N/A	N/A
Slow Walk	VH	H	VH	M-H	N/A	N/A
Running	VH	H	VH	H	N/A	N/A
Crawling	H	M-H	M-H	L-M	N/A	N/A
Rolling	VH	M-H	M-H	L-M	N/A	N/A
Jumping	VH	M-H	H	H	VH	VH

Tunneling	VL	VL	VL	VL	L	VL
Trenching	L	L-M	L	L-M	L	VL
Bridging	L	L	VL	M	VL	VL
Cutting	N/A	N/A	N/A	N/A	M-H	H
Climbing	N/A	N/A	N/A	N/A	H	H

Where :

0.95	**VH**	**= Very High**
0.8-0.9	H	= High
0.5-0.7	M-H	= Medium to High
<0.5	M	= Medium
	L-M	= Low to Medium
	L	= Low
	VL	= Very Low
	L-H	= Low to High
	N-A	= Not Applicable

Another example of raw data of probability of detection is shown in Table 3.3. This is the example of probability of detection in different type of sensors related to the selection of adversary equipment. The probability of detection in this analysis tool is obtained from two of this source and may be used in application one or both of them depend on the input data.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

The example of raw data of time delay is presented in Table 3.4. This Table is provided data of time delay from fence type 2.5 m Chain-link mesh with outriggers 4-mm x 50-mm mesh. The time delay varies according to the selection of adversary equipment and represents in four type of data i.e. minimum time, mean, maximum time, and standard deviation. The data which is input in database is in mean format. Standard deviation for each of data is depend on user input in standard deviation factor (SD-F).

Table 3.3 Probability of Detection of Different Type of Sensors Due To Equipment

Exterior Sensors	**No Equip**	**Hand Tolls**	**Power Tools**	**High Explosives**	**Land Vehicle**
Seismic Buried Cable	0.5	0.5	0.5	0.5	0.9
Electric Field	0.5	0.3	0.3	0.5	0.9
Infrared	0.8	0.4	0.4	0.5	0.8
Microwave	0.8	0.7	0.7	0.7	0.9
Video Motion	0.8	0.6	0.6	0.7	0.9
Multiple Non Complementary	0.9	0.8	0.8	0.8	0.99
Multiple Complementary	0.99	0.95	0.95	0.99	0.99

Table 3.4 Time Delay of Safeguard Element Type Fence 2.5 m Chain-link Mesh

			Penetration Time (Seconds)			
Description	Adversary Equipment	Equipment Weight (Kg.)	Min	Mean	Max	Standard Deviation
2.5-m Chain-link mesh with outriggers 4-mm x 50-mm mesh	Ladder	5.0	6	12	18	2.4
	Tarpaulin	2.0	6	12	18	2.4
	Pliers	1.0	60	120	180	24.6
	Manual Boltcutters	3.0	30	60	90	12
	Circular Saw	10	30	60	90	12
	Manual Boltcutters (Larger Opening)	3.5	45	90	135	18.6
	Circular Saw (Larger Opening)	11	45	90	135	18.6
	Gloves	0.5	6	12	18	2.4

All of the raw data for database application are represented in MATLAB as structure file as shown in Fig 3.13. Structure file format is prefer to use in programming database due to its capability as data storage and be able to transform to other file form such as table, or export to Microsoft Excel for others application. Following is the way how to use and implement this structure data.

Variables - PD

PD

1x16 struct with 35 fields

Fields	ProbOfDetection	EF	MW	AIR	PIR	FM	TW
1	'Walking'	0.9500	0.9500	0.9500	0.8500	0.5000	0.5000
2	'Slow Walk'	0.9500	0.8500	0.9500	0.6000	0.5000	0.5000
3	'Running'	0.9500	0.8500	0.9500	0.8500	0.5000	0.5000
4	'Crawling'	0.8500	0.6000	0.6000	0.4000	0.5000	0.5000
5	'Rolling'	0.9500	0.6000	0.6000	0.4000	0.5000	0.5000
6	'Jumping'	0.9500	0.6000	0.8500	0.8500	0.9500	0.9500
7	'Tunneling'	0.1000	0.1000	0.1000	0.1000	0.3000	0.1000
8	'Trenching'	0.3000	0.4000	0.3000	0.4000	0.3000	0.1000
9	'Bridging'	0.3000	0.3000	0.1000	0.5000	0.1000	0.1000
10	'Cutting'	0.5000	0.5000	0.5000	0.5000	0.6000	0.8500
11	'Climbing'	0.5000	0.5000	0.5000	0.5000	0.8500	0.8500
12	'No Equipment'	0.5000	0.5000	0.8000	0.8000	0.5000	0.5000
13	'Hand Tools'	0.5000	0.8000	0.8000	0.8000	0.1000	0.2500
14	'Power Tools'	0.3000	0.6000	0.4000	0.4000	0.1000	0.2500
15	'High Explosives'	0.5000	0.7000	0.5000	0.5000	0.7500	0.7500
16	'Land Vehicle'	0.9000	0.9000	0.8000	0.8000	0.8500	0.8500

Figure 3.13 Probability of Detection in PPS Analysis Tool Database

The command to call data from structure file is in the format:

X = FileName (FieldNumber).FieldName

For example, following is command to call probability of detection of Active Infrared (AIR) sensor in detection running activity. Name of structure file is PD then:

PD (AIR) = PD (3). AIR

PD (AIR) = 0.95

3.5.4 Calculation Process

Fig 3.14 shows the flow diagram of calculation process. There are two main input in this process i.e. matrix data from input data processing results and matrix of path from path finding algorithm. Following is discussed how to determine matrix of path.

Path finding algorithm needs sparse matrix (Asp) of graph as input, starting point, end point, and number of path to be analyzed (K). The sparse matrix of graph is the other forms of matrix adjacency. Starting and end point of paths is input in the interface. Number of probable path is estimated by square of sparse matrix size. The output of this algorithm is list of probable paths from start point to the end. The command function is defined as follows.

[PATHS] = graphkshortestpaths (Asp, start, target, K);

The calculation of adversary path analysis is based on path sequence so it is required to transform data from matrix form to list of path. The command to perform this transformation as follows.

```
%--------------------------------------------------------------------------------------------------------------------
% Data structure
A = myhandles.A;              % Matrix Adjacency
PATHS = myhandles.PATHS;      % List of Paths from Path Finding
Algorithm
nPath = myhandles.nPath;      % Total Number of Path
PM= myhandles.PM ;            % List of Path in Matrix Form
MPd = myhandles.A1;           % Matrix Data Probability of Detection
%--------------------------------------------------------------------
% 1. Prepare to convert data in Matrix form to List of Path

L=size(A);
for i=1:nPath
    d(i)=length(cell2mat(PATHS(i)));      % Total Length of Path
    v = PM(i,1:(d(i)-1));                 % V1 (Vertices 1) for
create edge
```

```
    w = PM(i,2:(d(i)));                    % V2
    lin{i} = sub2ind(L,v,w);               % Linear Index The edge in
matrix
    J(i) = {MPd(cell2mat(lin(i)))};        % convert matrix MPd to
list J
end
%------------------------------------------------------------------------
```

Following is the example of commands program to solve the calculation of probability of detection. All of equations is calculated based on list of path. This concept of calculation is same for others parameter in PPS performance calculation. The difference is on applied equation to use.

```
%------------------------------------------------------------------------
% 2. Calculation process based on Equation 1.
dl=d-1;                                    % Total Length of Path - 1
for j=1 :( length (PATHS))
L(j,(1: length(cell2mat(J(j)))))= cell2mat(J(j));
% Left Side of Eq. 1
Pdj(j,(1: length(cell2mat(J(j)))))=(1-L(j,(1:
length(cell2mat(J(j))))));
% (1- PDj); previous edge
R(j,(1: length(cell2mat(J(j)))))=
cumprod(Pdj(j,(1:length(cell2mat(J(j))))));
% Sigma (1- PDj); Right Side of Eq. 1
end


% 3. Final Calculation

for i=1:nPath
for j=2:dl(i)
    PD(i,j) = L(i,j)*R(i,j);         %Probability of Detection (Final
Results)
    PD(i,1)  = L(i,1)                %
    PTD(i,j) = R(i,j);               % ( 1 - PD )
    PTD(i,1) = 1-L(i,1)              %
    PFD(i,1) = L(i,1)                % Need for PI calculations
    PFD(i,j) = L(i,j)*PTD(i,j-1)
end
end
%------------------------------------------------------------------------
```

The rest process is calculate all of the PPS performance parameter. Detail of formula behind this calculation is already discussed in previous section.

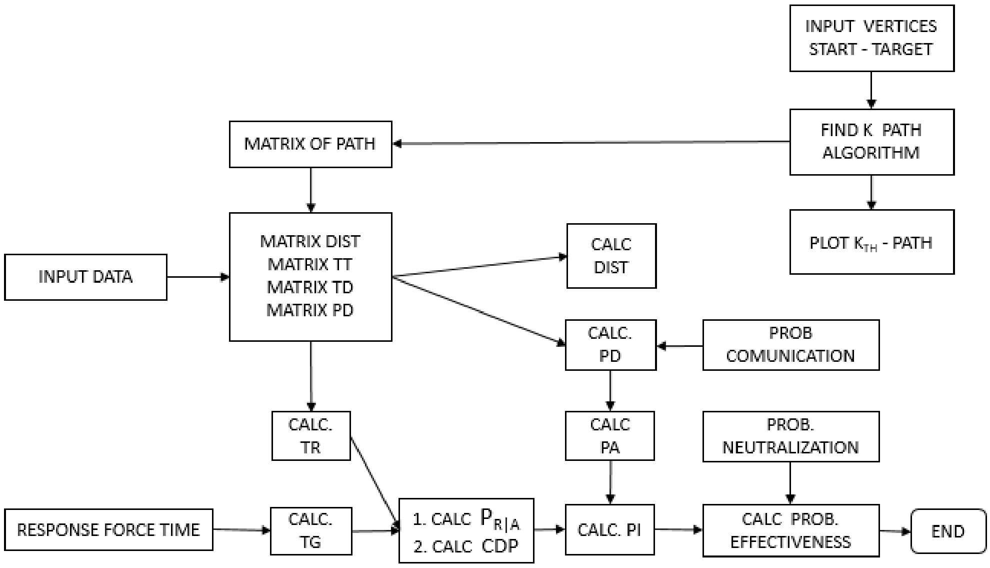


Figure 3.14 Flow Diagram of Calculation Process

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

CHAPTER IV
RESULTS AND DISCUSSION

The purpose of this Chapter is to demonstrate the use of this analysis tool in measure the PPS performance due to adversary threat scenario. There are three example problems i.e. two simple practical examples of adversary path analysis, and one example of PPS assessment in hypothetical nuclear facility.

4.1 SIMPLE PRACTICAL EXAMPLE 1

In this section discusses the example of adversary attack scenario in simple facility as seen in Fig 4.2. The facility protects the vital area with a building and a wall inside to cover it. Detection elements is installed in each of entry points such as in the door. There is a fence surround the facility.

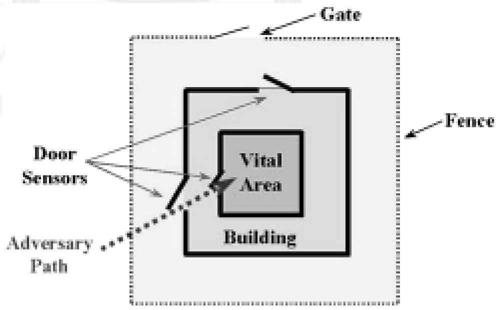


Figure 4.1 Adversary Path, Simple Practical Example

The probable task scenario is the adversary penetrate the fence, travel to the building, force open a door, travel to the vital area, force open another door, and set and detonate an explosive device on the critical asset. This adversary path is translated into the following network:

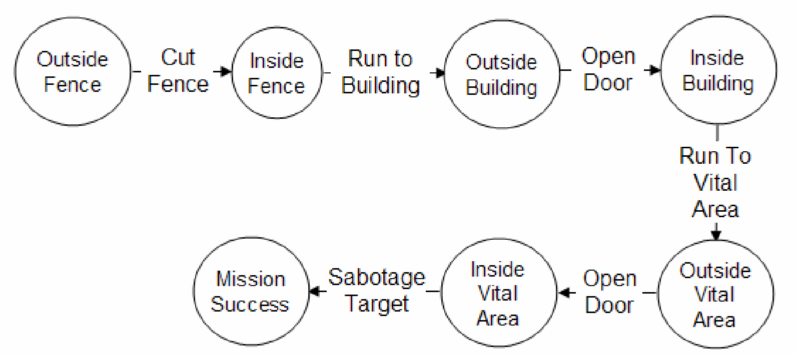


Figure 4.2 Adversary Path Scenario of Simple Practical Example

The probable task scenario is enhanced into six different detection value arrangement as shown in Table 6, and 7. Others data which is required for PPS analysis present in table 8. As a note, the time units is used second. The experiments focus on calculating probability of interruption as PPS for a given scenario.

Table 4. 1 Probability of Detection in Mean Values

Task	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6
Cut Fence	0	0.9	0.9	0.9	0	0
Run to Building	0	0	0	0	0	0
Open Door	0.9	0.9	0.9	0.9	0.9	0.9
Run to Vital Area	0	0	0	0	0	0
Open Door	0.9	0.9	0.9	0.9	0.9	0.9
Sabotage Target	0	0	0	0	0.9	0

Table 4. 2 Delay Time in Mean Values (second)

Task	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6
Cut Fence	10	10	10	10	10	30
Run to Building	12	12	12	12	12	12
Open Door	90	90	90	90	90	90

Run to Vital Area	10	10	10	10	10	10
Open Door	90	90	90	90	90	90
Sabotage Target	120	120	120	240	120	120

Table 4.3 PPS Complement Data Example 1.

Parameters	Value
Response Force Time	300
Probability Of Communication	0.95
Standard Deviation Factor	0.3
Probability Of Neutralization	1

The network of these scenarios in Figure 27, is generated in PPS analysis tool by using fixed graph method with modification. Start and target points is Vertices 4 and 46.

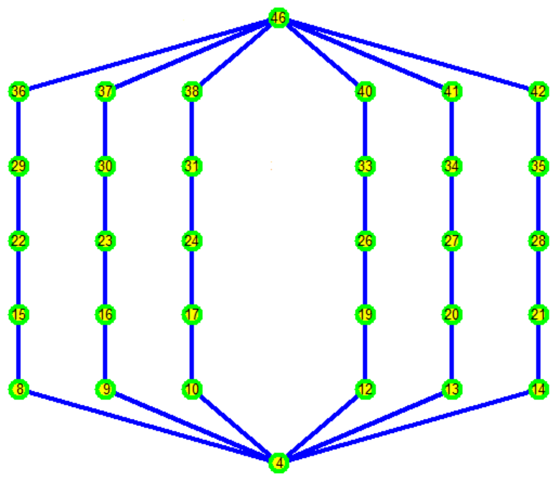


Figure 4.3 Network Graph Model of Adversary Path

Table 4.4 shows the calculation results of above scenario. This results presents critical detection point and probability of interruption in each of task scenarios.

Table 4.4 Results of Adversary Path Analysis Example 1

Scenario	Path No	Detection Location	Vertices Number							PI
1	Path 3	B	4	8	15	22	29	36	46	0.459
2	Path 2	B	4	9	16	23	30	37	46	0.5705
3	Path 5	E	4	10	17	24	31	38	46	0.4997
4	Path 1	B	4	12	19	26	33	40	46	0.8378
5	Path 6	E	4	13	20	27	34	41	46	0.1771
6	Path 4	B	4	14	21	28	35	42	46	0.459
Where ◯ : CDP										

The lowest value of PI is 0.1771 in scenario 5. Although, in scenario 5 has additional probability of detection in the last task, but the PI and CDP is still lower than the closest scenario 1. It is caused by the different detection location point. The detection location in scenario 1 is in the beginning of task (B) but in scenario 5 is in the end of task (E). The concept of PPS is to detect the adversary as soon as possible for providing enough time of the response force to interrupt the adversary. So in this case, the response force in scenario 1 may arrive early than scenario 5 due to the in the scenario 1 is received the alarm notification faster than scenario 5. This is an example of how the detection location point plays an important role in PPS effectiveness.

If the scenario 5 is assumed the most vulnerable path in the system. The system needs to improve. There are several option may be selected from above scenario. For example, the scenario 3 installs detection element in the first task without changing the detection location. The result is increase probability of interruption but CDP is still lowest. Other example is scenario 2 which changes the detection location to the beginning (B). It improves the quality of CDP. The best example of system improvement is in scenario 4. The PI is 0.8373. The additional delay time in the last task contributes to improve the PI and CDP value significantly.

The analysis results above is verified with other tool by comparison method. EASI tool is used for this purpose. It is obtained that the results value in analysis tool is the same as the results from EASI tool. This is a proof that this analysis tool may use in the further calculation related to PPS performance measurement. Following is the results calculated on EASI tool with the same scenario as above.

Table 4.5 EASI Analysis Results for Scenario 1 and 2.

Probability of Interruption:			0.46
Probability of Alarm Communication		Response Force Time (in Seconds)	
		Mean	Standard Deviation
0.95		300	90
		Delays (in Seconds):	
P(Detection)	Location	Mean:	Standard Deviation
0	B	10	3.0
0	B	12	3.6
0.9	B	90	27.0
0	B	10	3.0
0.9	B	90	27.0
0	B	120	36.0

Probability of Interruption:			0.57
Probability of Alarm Communication		Response Force Time (in Seconds)	
		Mean	Standard Deviation
0.95		300	90
		Delays (in Seconds):	
P(Detection)	Location	Mean:	Standard Deviation
0.9	B	10	3.0
0	B	12	3.6
0.9	B	90	27.0
0	B	10	3.0
0.9	B	90	27.0
0	B	120	36.0

Table 4.6 EASI Analysis Results for Scenario 3 and 4.

Probability of Interruption:			0.50
Probability of Alarm Communication		Response Force Time (in Seconds)	
		Mean	Standard Deviation
0.95		300	90
		Delays (in Seconds):	
P(Detection)	Location	Mean:	Standard Deviation
0.9	E	10	3.0
0	E	12	3.6
0.9	E	90	27.0
0	E	10	3.0
0.9	E	90	27.0
0	E	120	36.0

Probability of Interruption:			0.84
Probability of Alarm Communication		Response Force Time (in Seconds)	
		Mean	Standard Deviation
0.95		300	90
		Delays (in Seconds):	
P(Detection)	Location	Mean:	Standard Deviation
0.9	B	10	3.0
0	B	12	3.6
0.9	B	90	27.0
0	B	10	3.0
0.9	B	90	27.0
0	B	240	72.0

Table 4.7 EASI Analysis Results for Scenario 5 and 6.

Probability of Interruption:			0.18
Probability of Alarm Communication		Response Force Time (in Seconds)	
		Mean	Standard Deviation
0.95		300	90
		Delays (in Seconds):	
P(Detection)	Location	Mean:	Standard Deviation
0	E	30	9.0
0	E	12	3.6
0.9	E	90	27.0
0	E	10	3.0
0.9	E	90	27.0
0	E	120	36.0

Probability of Interruption:			0.46
Probability of Alarm Communication		Response Force Time (in Seconds)	
		Mean	Standard Deviation
0.95		300	90
		Delays (in Seconds):	
P(Detection)	Location	Mean:	Standard Deviation
0	B	30	9.0
0	B	12	3.6
0.9	B	90	27.0
0	B	10	3.0
0.9	B	90	27.0
0	B	120	36.0

4.2 PRACTICAL EXAMPLE 2

This second example is obtained from Garcia example scenario[8] and already mentioned in previous Chapter for describing adversary sequence diagram (ASD) method. Fig 4.5 shows the adversary sequence diagram and map facility of this example. Fig 4.6 shows the network diagram which is generated by analysis tool. It represents likelihood adversary path of the facility which is shown in ASD as well.

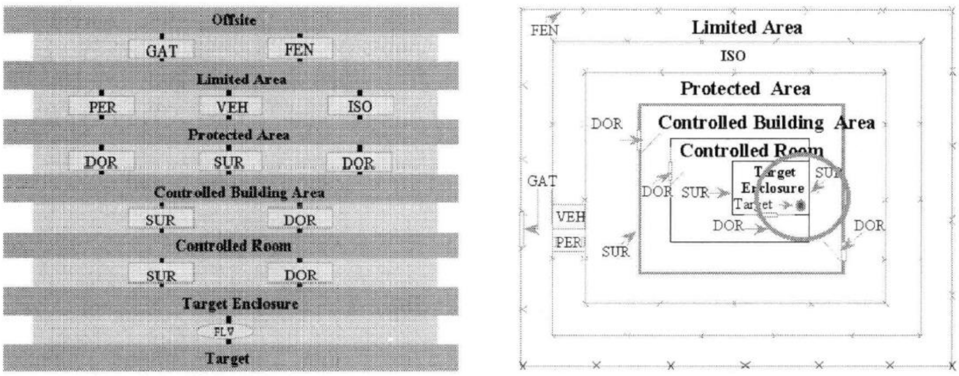


Figure 4.4 Adversary Sequence Diagram and 2D Map of Sample Facility

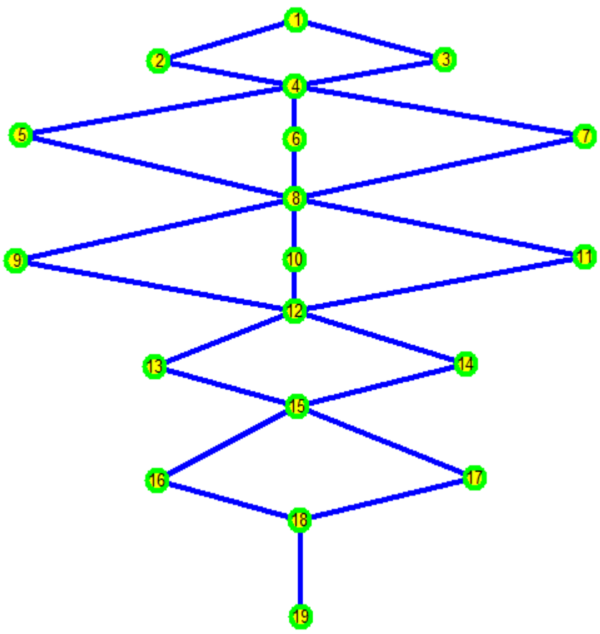


Figure 4.5 Network Diagram of Sample Facility

The scenario is the outsider travelling on foot carrying explosives and metal tools to sabotage the target. Running speed is approximately 10 feet per second. Time unit is second. The path starts at point 1 and end in point 19. Others required data and the PPS facility data are shown in Table 4.9 and Table 4.10 as follows

Table 4.8 PPS Complement Data Example 2

Parameters	Value
Response Force Time	300
Probability Of Communication	0.95
Standard Deviation Factor	0.3
Probability Of Neutralization	1

Table 4.9 Adversary Task and PPS Data for Example 2

Edge No.	**Action**	**Safeguard Elements**	**PD**	**Td**
1-2	Climb Gate			10
1-3	Climb Fence	Tilt / Vibration Fence Sensor	0.8	10
2-4	Penetrate Limited Area From Gate			5
3-4	Penetrate Limited Area From Fence			5
4-5	Penetrate Personal Entrance	2 Doors, Check Badge Reader	0.85	32
4-6	Penetrate Vehicle Entrance	Guard Officer	0.5	30
4-7	Penetrate Isolation Zone	Microwave Exterior Sensor	0.9	5
5-8	Penetrate Protected Area From Personal Entrance			5
6-8	Penetrate Protected Area From Vehicle Entrance			5
7-8	Penetrate Protected Area From Isolated Zone			5
8-9	Open External Door 1	6 Inch Metal	0.99	60
8-10	Penetrate Wall	30 cm Reinforced Concrete Wall		180
8-11	Open External Door 1,	6 Inch Metal	0.99	60
9-12	Penetrate Controlled Building Area From Door 1			5
10-12	Penetrate Controlled Building Area From Wall			5
11-12	Penetrate Controlled Building Area From Door 2			5
12-13	Penetrate Wall	30 cm Reinforced Concrete Wall		180
12-14	Open External Door	3 Inch Metal	0.99	30
13-15	Penetrate Controlled Room From Wall			5
14-15	Penetrate Controlled Room From Door			5
15-16	Penetrate Wall	30 cm Reinforced Concrete Wall		180

15-17	Open Interior Door	1.6 mm Metal	0.99	60
16-18	Penetrate Target Enclosure From Wall			5
17-18	Penetrate Target Enclosure From Interior Door			5
18-19	Penetrate And Sabotage Target			171

Actually, there are 72 likelihood of path scenario has been calculated by using this analysis tool. The calculation results is represented in Table 4.11.

Table 4.10 Results of Adversary Path Analysis Example 2

Scenario	Path No	Vertices Number												PI
1	2	1	3	4	5	8	9	12	13	15	16	18	19	0.94
2	27	1	2	4	7	8	10	12	13	15	16	18	19	0.81
3	72	1	3	4	7	8	11	12	14	15	17	18	19	0.66
4	32	1	2	4	6	8	10	12	13	15	17	18	19	0.57
5	12	1	2	4	6	8	10	12	13	15	16	18	19	0.45
Where ◯ : CDP														

The highest probability of interruption is 0.94 in path 2, and the lowest is 0.45 in path 12. Path no 12 has the lowest PI value but the CDP looks normal. This problem is due to the path has provided with adequate delay elements but it is not supported by good detection system. It shows in Table 4.10 that along sequence of path no 12 have only single detection point in the guard post (edge 4-6). This is the example of the system which has good delay elements but poor in detection.

4.3 HYPOTHETICAL FACILITY

This section presents the example of PPS performance measurement in hypothetical facility. The purpose of this section is to demonstrate complex analysis by using this analysis tool. A brief history of this hypothetical facility as follows.

The Lagassi Nuclear Research Institute (LNRI) was started in 1950 to serve as the nation's premier nuclear energy research facility. One of the facilities is the PTR

Research Reactor. This reactor is type light water moderated with highly enriched uranium (HEU)-fueled research reactor. The reactor is used for research on advanced reactor components, special fuel assemblies, and production of radionuclides for the medical industry. These nuclear material inventories and amount of radioisotopes may become a potential target for terrorist attack. PTR research reactor has already established physical protection system (PPS) but since the escalation of political condition in the country and the information from intelligence that the terrorist plans to attack the strategic facility, the government decides to improve the current PPS.

Fig 4.7 to Fig 4.11 are the map of LNRI facilities which shows in details the location and specification of safeguard elements in system, entry control points, and how to access to the vital area.

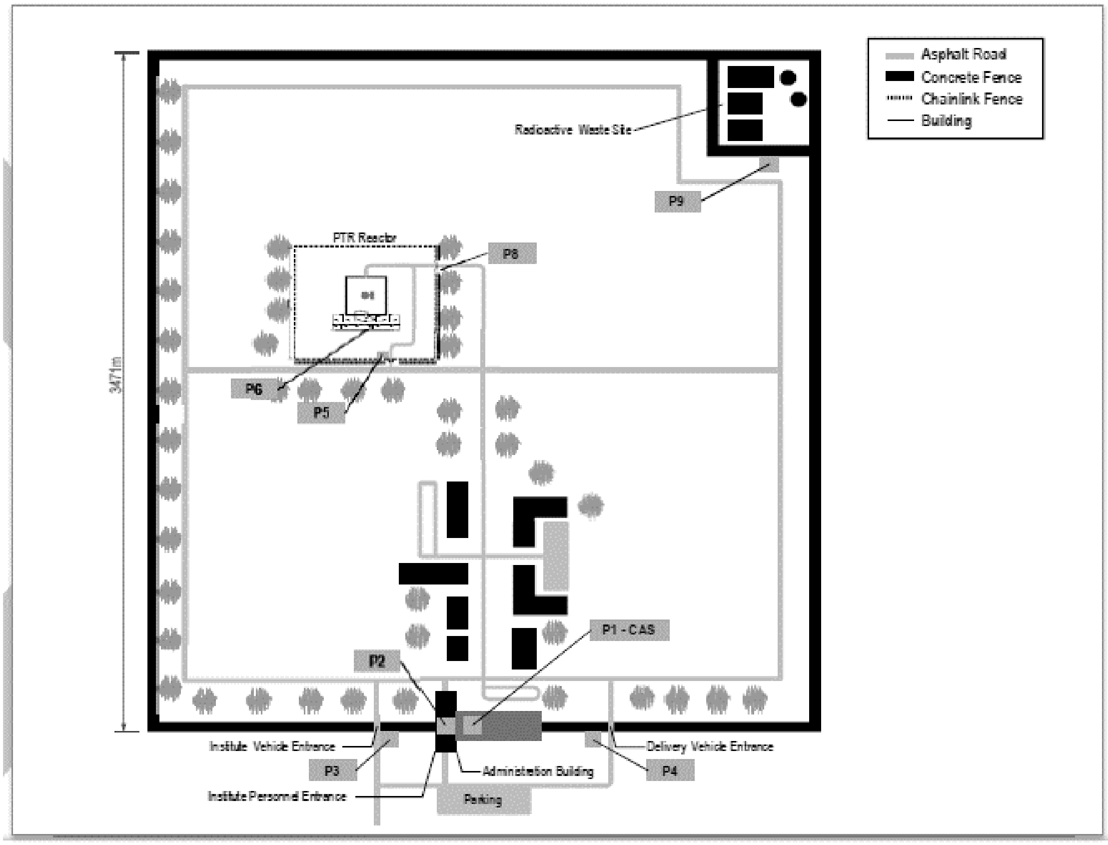


Figure 4.6 Map of LNRI facilities And Post Security.

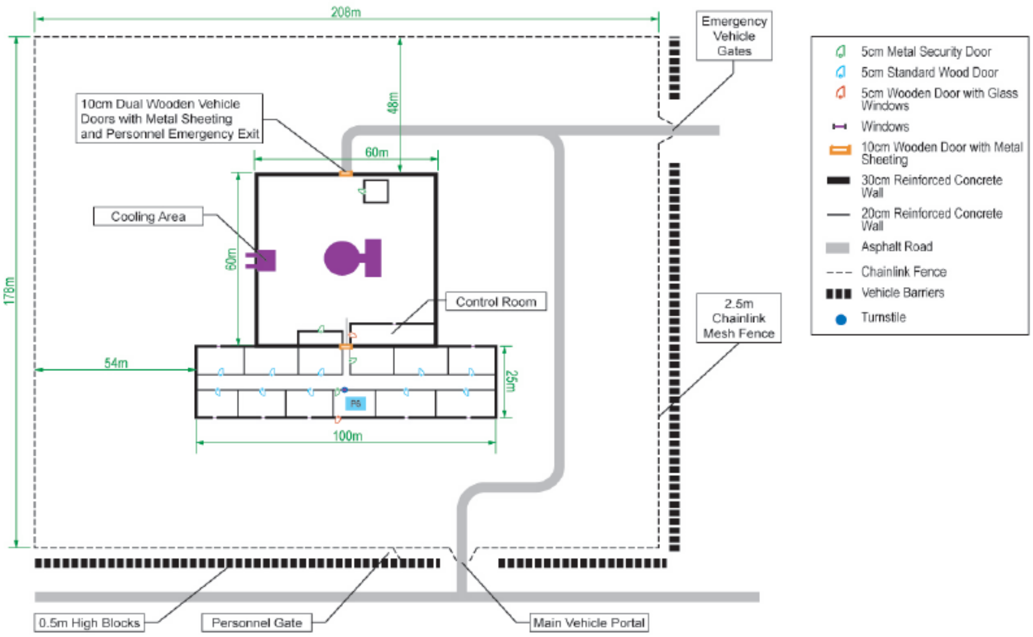


Figure 4.7 PTR Engineering Drawing 1

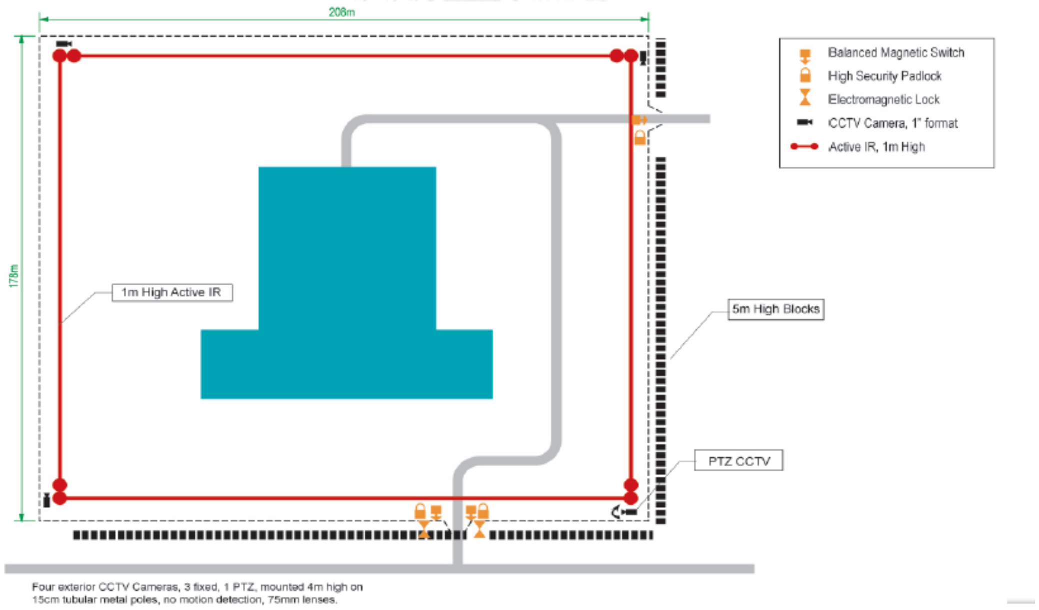


Figure 4.8 PTR Engineering Drawing 2

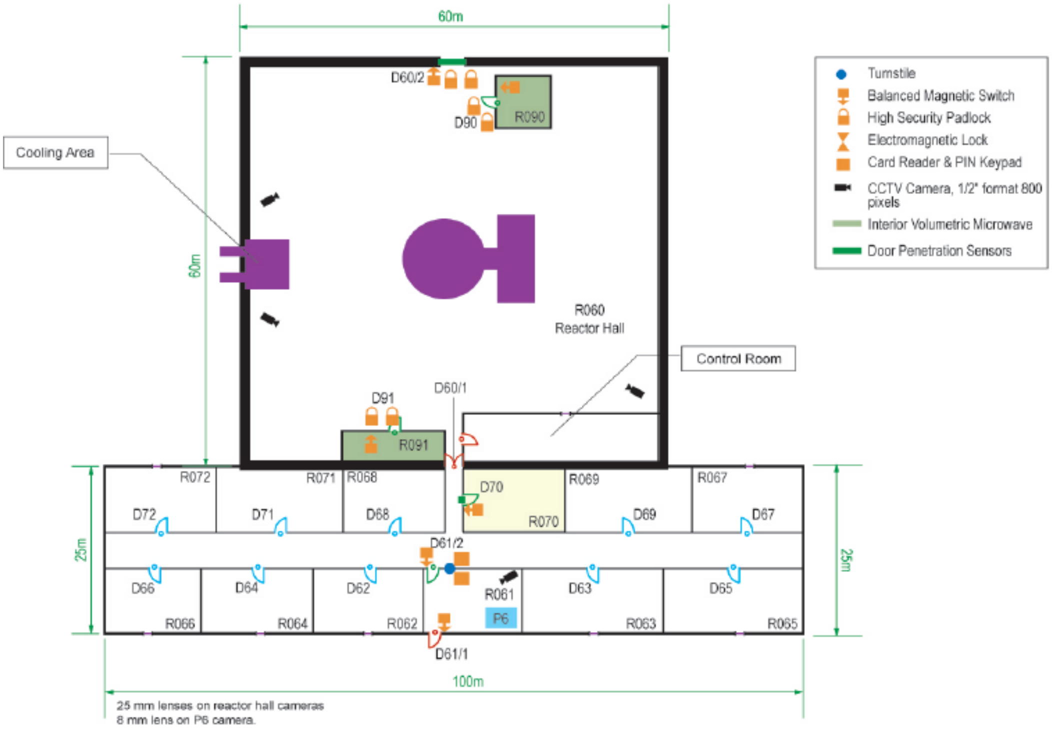


Figure 4.9 PTR Engineering Drawing 3

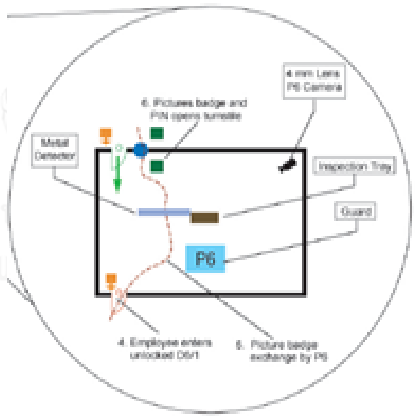


Figure 4.10 PTR Engineering Drawing 4

Following Fig 4.12 is the adversary sequence diagram (ASD) of the facility. This figure represents the picture of adversary paths into a facility and all of the protection elements by layer. The movement from one layer to another is defined as adversary action. In this ASD is set that the adversary mission begin from outside of facility and finish after reach the target. The likelihood threat is sabotage scenario with explosive material. The goal of adversary's mission is to explode the fresh fuel storage in the PTR building.

The adversary has two possible tactics scenarios for completing the mission, i.e. force attack and stealth attack scenario. In the first tactics, the adversary needs to reach the target as fast as possible. Vehicle land is one of possible equipment to use. In the ASD is shown that the possible sequence of path starts from offsite facility, penetrate institute vehicle gate, across the LNRI facility ground, penetrate emergency vehi cle gate, across PTR ground, explode the vehicle door , go inside the reactor hall, explode the door and enter the fuel storage room and sabotage the fresh fuel.

The stealth tactics may use two probable scenarios. The concept of this scenario is the adversary infiltrates to the facility with lowest detection. The first scenario i.e. the adversary enters the facility as workers. In this mission, the adversary needs to cover the action with fake personal identity, uniform, behave as usual and use the facility common way. The probable sequence of path as follows: the adversary starts from outside of facility, enters institute personnel entrance, crosses the LNRI ground, walks to PTR personnel gate, crosses the PTR ground, enter the main entrance, follows the personal identity check gate, walks at corridor reactor office, penetrate the reactor door, cross the reactor hall, then sabotage the fresh fuel storage.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

The second stealth tactics is thief scenario. The adversary requires to select the best time to penetrate and looking the most weakness points in the facility and it is necessary to bring some equipment for defeating the protection elements. The possible sequence of path is from outside facility, the adversary climbs the fences, penetrates to inside of reactor building by breaking the window, open the reactor door with hand tool, runs to the reactor hall and sabotage the target.

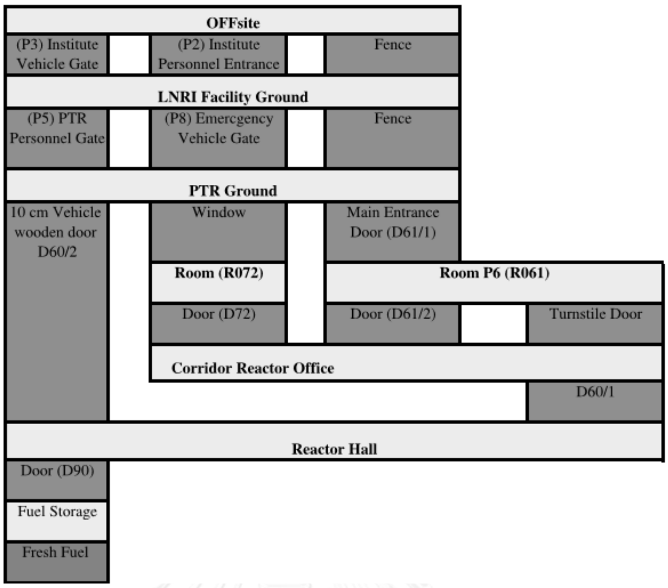


Figure 4.11 Adversary Sequence Diagram (ASD) of Hypothetical Facility

The network of adversary paths in this analysis tool as shown in Fig 4.13 is generated based on the above adversary sequence diagram. There are 24 vertices with connecting edges.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

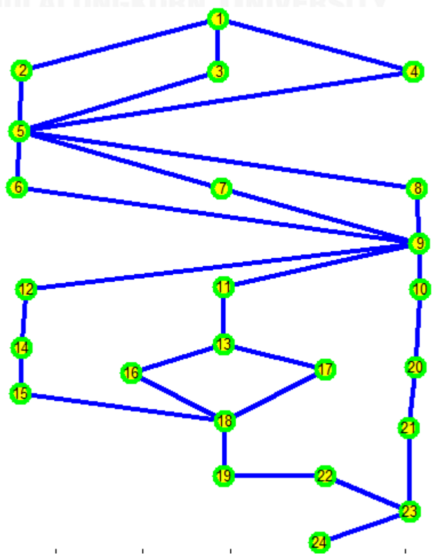


Figure 4.12 Network of Adversary Path

The list of adversary action and metric performance of safeguard elements in detect and delay the adversary is shown in the following Table 4.12.

Table 4.11 Adversary Paths and PPS Metric Performance.

1	**V2**	**ADV METH**	**PPS ELM.**	**TYPE**	**SENSOR**	**EQUIP**	**DIST**	
1	2	Climbing	Fence	2.5-m concrete panel wall		Ladder	0.5	
1	3	Land Vehicle	Institute Vehicle Gate	Guards Post P3		Truck	5	E
1	4	Walking	Institute Personal Entrance (Pedestrian Gate)		Visual Id Check		5	E
2	5	Running	Facility Ground				250	
3	5	Running	Facility Ground				800	
4	5	Land Vehicle	Facility Ground				800	
5	6	Land Vehicle	Emergency Vehicle Gate	P8		Truck	5	
5	7	Climbing	Fence	2.5 Chain link Mesh Fence	Active Infrared	Ladder	0.5	E
5	8	Walking	PTR Personal Gate	P5	High Security Padlock	Hand tool	5	B
					Balanced Magnetic Switch			B
6	9	Land Vehicle	PTR Ground			Truck	150	
7	9	Running	PTR Ground				100	
8	9	Running	PTR Ground				100	
9	12	Walking	Window R072	Tempered Glass Window		Hand tool	0.5	B
9	10	Land Vehicle	10 cm Dual Wooden Vehicle Doors	With Metal Sheeting	Balanced Magnetic Switch	Explosive 0.5 kg	5	B
9	11	Walking	Main Entrance Door	5 cm Wooden Door With Glass Window	Balanced Magnetic Switch		2	B

12	14	Walking	Room R072				5	
14	15	Walking	Door D72,	5cm Wooden Door With Glass Window		Fire Ax	2	B
15	18	Running	Corridor PTR Office				10	
11	13	Walking	Room R061	P6 guard at post	Video Motion		5	M
13	16	Walking	Turnstile Door	Guard at Post	Video Motion	Hand tool	5	M
13	17	Walking	Door D61/2	5 cm wood door pedestrian door	Balanced Magnetic Switch	Fire Ax	5	B
16	18	Running	Corridor PTR Office				10	
17	18	Running	Corridor PTR Office				10	
18	19	Walking	Door 60/1 (Enter Reactor)	10 cm Wooden Door With Metal Sheeting	Balanced Magnetic Switch	Explosive 1 Kg	1	B
19	22	Running	Reactor Hall	3 x CCTV Camera, ½ inch Format 800			10	M
10	20	Running	Reactor Hall	3 x CCTV Camera, ½ inch Format 800			5	M
22	23	Open	Door D90	5 cm Metal Security Door	High Security Padlock	Hand tool	1	B
					Balanced Magnetic Switch			
20	23	Open	Door D90	5 cm Metal Security Door	High Security Padlock	Explosive 1 Kg	1	
23	24	Target					1	

Following Table 17 is PPS performance calculation results of several paths in hypothetical facility. The highest value of PI is 0.709, and the lowest one is 0.049.

Table 4.12 Calculation Results PPS Performance Evaluation of Hypothetical Facility

Scenario	Path No	Vertices Number													PI
1	PATH 36	1	4	(5)	8	9	11	13	16	18	19	22	23	24	0.709697
2	PATH 20	(1)	2	5	6	9	11	13	16	18	19	22	23	24	0.40406
3	PATH 15	(1)	2	5	8	9	11	13	17	18	19	22	23	24	0.232475
4	PATH 19	(1)	2	5	6	9	12	14	15	18	19	22	23	24	0.080574
5	PATH 1	1	2	5	6	9	10	20	23	24	No CDP				0.049343
	Where ( ): CDP														

Data of analysis for overall paths represents the results that PPS performance of this hypothetical facility is low. Table 18 shows the results of CDP calculation in overall path. As remainder that the requirement for PPS performance is TR-TG > 0, and the CDP should located as close as possible to the end of sequence. It means that the response force has enough time to interrupt the adversary. But in this case, almost all of the path has CDP close to the beginning point. Even in path 1, the CDP is not available due to TR-TG for all of path sequence lower than 0. In the adversary point of view, he has completed the mission without interrupted.

Table 4.13 CDP and PI Calculation Results from MATLAB

P	TR-TG													PI
1.	-12.4	-74.9	-87.2	-96.2	-96.2	-175	-176	-206						0.049
2.	125.1	-74.9	-87.2	-96.2	-96.2	-175	-176	-206						0.433
3.	3.424	-59.1	-71.2	-96.2	-96.2	-175	-176	-206						0.105
. . .														…
34.	229.3	29.25	29.25	-7.88	-7.88	-13.6	-52.4	-93	-93	-115	-116	-206		0.71
35.	166	-34	-34	-71.1	-76.3	-77.5	-90	-93	-93	-115	-116	-206		0.531
36.	229.3	29.25	29.25	-7.88	-7.88	-13.6	-52.4	-93	-93	-115	-116	-206		0.71

This problem can be solved by upgrade the system with more delay elements with high probability of detection. Table 19 and 20 show the data of upgrade system by adding delay time 120 s and the probability of detection 0.9. The upgrade elements is located in edge 6-9 when penetrate the PTR ground. Now, the CDP in path 1 is available in vertices no 5. PI value also increases from previous 0.049 to 0.30. And the highest PI increases to 0.85 in path 36.

Table 4.14 CDP and PI Calculation Results from MATLAB

P	**TR-TG**												**PI**
1.	109.3	46.8	34.5	-25.5	-143	-201	-236	-236	-221	0	0	0	0.309
2.	246.8	46.8	34.5	-25.5	-143	-201	-236	-236	-221	0	0	0	0.609
3.	109.1	46.63	34.5	-25.5	-143	-201	-236	-236	-221	0	0	0	0.458
34.	324.3	124.3	124.3	52.13	-7.88	-13.6	-52.4	-93	-93	-115	-116	-206	0.85
35.	261	61	61	-11.1	-76.3	-77.5	-90	-93	-93	-115	-116	-206	0.732
36.	324.3	124.3	124.3	52.13	-7.88	-13.6	-52.4	-93	-93	-115	-116	-206	0.85

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Table 4.15 CDP and PI Calculation Results of Upgrade System

Scenario	**Path No**	**Vertices Number**													**PI**
1	PATH 36	1	4	5	8	9	11	13	16	18	19	22	23	24	0.850361
2	PATH 20	1	2	5	6	9	11	13	16	18	19	22	23	24	0.657746
3	PATH 15	1	2	5	8	9	11	13	17	18	19	22	23	24	0.547037
4	PATH 19	1	2	5	6	9	12	14	15	18	19	22	23	24	0.372197
5	PATH 1	1	2	5	6	9	10	20	23	24					0.309467

Where ◯ : CDP

4.4 LIMITATION OF THE SOFTWARE AND FUTURE WORKS

There are many limitation of this PPS analysis tool due to this research is the first version. Some of them is defined as follows.

1. This analysis tool is capable to calculate only for sabotage scenario.

2. This analysis tool cannot calculate probability of neutralization as one of parameter in PPS measurements.
3. Program codes behind this analysis tool have some repetition calculation and unnecessary commands.
4. It is difficult to reanalysis the results and re generate the upgrade network system.
5. The display of calculation results is not available in GUI.

The future works of this analysis tool is concern as follows.

1. It needs to optimize the current code by rearranged the commands and revised some repetition code.
2. It needs to create some interface for display the results.
3. It is necessary to validate the safeguard elements database.
4. It is recommended to add more PPS performance measurement functions such as thief scenario.
5. In the future, response force will define as function of distance and location of arrival in the system. Response force may be varied depend on this function in the system.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

CHAPTER V
CONCLUSION

5.1 CONCLUSION

This research contributes significantly in the research of physical protection in nuclear facility. Indonesia which has four nuclear facility and up 700 industries with related to radioactive material requires this analysis tool for evaluating PPS effectiveness. The flexibility of this tool for further improvement and the availability of the network method to develop in wide application are one of the reasons to continue this research in the future.

This computer-based analytical tool has been developed for measuring the effectiveness of the system. This analysis tool has capability

1. To generate network model as adversary path.
2. To analysis scenario both in single path or multiple paths
3. To calculate the probability of detection, probability of interruption, critical detection point as PPS performance measurement.
4. To determine and rank the most vulnerable path in facility

The results of calculation analysis in this analysis tool are already verified by comparison method with EASI tool. It makes a proof that this analysis tool is proper to use in PPS analysis. Since this tool is the first research and has many limitation. It is necessary to improve further and optimized the code for obtaining better performance analysis.

CHULALONGKORN UNIVERSITY

REFERENCES

1. Park, C.-K., et al., *A PSA-based vital area identification methodology development.* Reliability Engineering & System Safety, 2003. **82**(2): p. 133-140.
2. George Bunn, C.B.e.a., *Research Reactor Vulnerability to Sabotage by Terrorist.* Science and Global Security, 2003. **11**: p. 85-107.
3. IAEA, *Nuclear Security Recomendation on Physical Protection of Nuclear Material and Nuclear Facilities (IAEA INFCIRC225 rev 5).* 2011.
4. IAEA, *The International Legal Framework for Nuclear Security*. 2011.
5. BAPETEN, *Laporan Keselamatan Nuklir 2014, Annual Indonesian Nuclear Safety Report 2014.* 2014.
6. http://www.bapeten.go.id/peta/, *Indonesian Nuclear Technology Application Utilization Map*. 2014.
7. https://www.inmm.org/Physical_Protection.htm. 2015 [cited 2015 5 Nopember].
8. Garcia, M.L., *Design and evaluation of physical protection systems*. 2007: Butterworth-Heinemann.
9. Vintr, Z., M. Vintr, and J. Malach. *Evaluation of physical protection system effectiveness*. in *Security Technology (ICCST), 2012 IEEE International Carnahan Conference on*. 2012. IEEE.
10. Jang, S.S., et al., *DEVELOPMENT OF A VULNERABILITY ASSESSMENT CODE FOR A PHYSICAL PROTECTION SYSTEM.* Nuclear Engineering and Technology, 2009. **41**(5): p. 747-752.
11. Hester, P.T., *Facility protection optimization under uncertainty*. 2007, Vanderbilt University.
12. Newman, M.E., *The structure and function of complex networks.* SIAM review, 2003. **45**(2): p. 167-256.
13. Byrne, R., J. Feddema, and C. Abdallah, *Algebraic Connectivity and Graph Robustness.* 2009.
14. Ahuja, R.K., T.L. Magnanti, and J.B. Orlin, *Network flows*. 1988, DTIC Document.

APPENDIX

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
function PPS_ETOOL
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                              PPS_eTOOL
%
%
%
%                  PHYSICAL PROTECTION SYSTEM eVALUATION TOOL
%
%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%======================================================================
% Create Figure
 h.f = figure('Visible','off','Position',[30,5,1300,700]);
%======================================================================
% Create axes
 h.ax = axes('Units','pixels','Position',[250,195,560,480]);
%======================================================================
% LOAD DATABASE FOR PPS FUNCTIONS
 DBASE=load('DBASE4.mat');
%======================================================================
% Create Data Structure in Figure
myhandles = guihandles(h.f);
%======================================================================
% Variable Field For Data Structure
myhandles.DB = DBASE.PD;        % Database
myhandles.ADE = DBASE.ADTDBASE; % Database
myhandles.advmeth ='';          % Variable For  Input Adversary
Method
myhandles.sensor1='';           % Data from input PD sensor 1
myhandles.sensor2='';
myhandles.sensor3='';
myhandles.sensor4='';
myhandles.sensor5='';
myhandles.sensor6='';
myhandles.sensor7='';
myhandles.pos ='';              % Graph Position X and Y
myhandles.A='';                 % Matrix adjacency of all graph
myhandles.A1='';                % MPTD
myhandles.A2='';                % MTM
myhandles.Asp='';               % Sparse Matrix Adjacency
myhandles.edge_list_prime='';   % Edges list in graph in type string
myhandles.ELPD = zeros(3);      % Edge list prime from input PD
myhandles.val='';               % Value from input global edge list
2input PD
myhandles.Pd_i ='';             % Primary Probability of Detection
before Balances
myhandles.PM ='';               % Matrix of Global PATH
myhandles.nPath = '';           % Number of Path
myhandles.PATHS = '';           % Global List PATHs
myhandles.listofArcs ='';       % Edges list in graph in type numeric
myhandles.PD ='';               % Probability of Detection in arc i
(final)
myhandles.Ladj='';
myhandles.ADT ='';
myhandles.ADTtype='';
myhandles.equipments ='';
myhandles.ELTD='';
```

```
myhandles.linind ='';
myhandles.TD = '';
myhandles.PTD='';
myhandles.dl='';
myhandles.PE='';
myhandles.AB='';
myhandles.Aadj='';
myhandles.xyi='';
myhandles.travelt='';
myhandles.npdet='';
myhandles.SafeG='';
myhandles.ELD='';
myhandles.A5='';
myhandles.ResDISTANCE='';
myhandles.vrate='';
myhandles.ResTimeTravel='';
myhandles.A4 ='';
myhandles.ELV='';
myhandles.A3 ='';
myhandles.ELPTD='';
myhandles.ELSDT='';
myhandles.A7='';
myhandles.ELDL ='';
myhandles.PI1=[];
myhandles.PI=[];
myhandles.PFD=[];
%=====================================================================
% Save The Data Structure
guidata(h.f,myhandles);
%=====================================================================
% Create Panel

h.panelA = uipanel('Title','CREATE
GRAPH','BackgroundColor',[1,0.4,0.6],...
          'FontSize',8,'Units','Pixels','Position',[5,185,240,500]);

h.panelB = uipanel('Title','INPUT PPS FUNCTIONS','FontSize',8,...

'Units','Pixels','visible','on','BackgroundColor',[1,0.4,0.6],...
          'Position',[830,185,230,500]);

h.panelC =
uipanel('Title','RESULTS','BackgroundColor',[1,0.4,0.6],...
          'FontSize',8,'Units','Pixels','Position',[250,15,570,150]);

h.panelD = uipanel('Title','INPUT PPS FUNCTIONS','FontSize',8,...

'Units','Pixels','visible','on','BackgroundColor',[1,0.4,0.6],...
           'Position',[1070,185,230,500]);

h.panelE = uipanel('Title','PATH
ANALYSIS','BackgroundColor',[1,0.4,0.6],...

'FontSize',8,'Units','Pixels','Position',[830,15,470,150]);

h.panelF = uipanel('Title','PATH ANALYSIS','FontSize',8,...
```

```
'Units','Pixels','visible','oFF','BackgroundColor',[1,0.4,0.6],...
           'Position',[1040,15,230,670]);

h.panelG = uipanel('Title','','FontSize',8,...

'Units','Pixels','visible','on','BackgroundColor',[1,0.4,0.6],...
          'Position',[5,15,230,150]);

%=====================================================================
% Create Components for " Create Graph "

h.rbh1 = uicontrol('Parent',h.panelA,'Style','radiobutton',...
        'String','FIXED GRAPH (X,Y)
','BackgroundColor',[1,0.4,0.6],...
        'Max',1,'Min',0,'Value',0,'fontsize',10,'Position',[10 430
150 30]);

h.rbh2 = uicontrol('Parent',h.panelA,'Style','radiobutton',...
        'String','INTERACTIVE
GRAPH','BackgroundColor',[1,0.4,0.6],...
        'Max',1,'Min',0,'Value',0,'fontsize',10,'Position',[10 400
190 30]);

h.select = uicontrol('Parent',h.panelA,'BackgroundColor','y',...
         'Style','pushbutton', 'String', 'SELECT','fontsize',10,...
         'callback',(@SelectGraph),'Position',[10,360,110,35]);

h.txt(1) = uicontrol('Parent',h.panelA,'Style','text','String',...
          'X ', 'BackgroundColor',[1,0.4,0.6],'fontsize',14,...
          'Position',[20,320,25,25]);

h.txt(2) = uicontrol('Parent',h.panelA,'Style','text','String',...
          'Y  ','Callback',(@addrow_colom),...
          'BackgroundColor',[1,0.4,0.6],'fontsize',14,...
          'Position',[85,320,25,25]);

h.edt(1) =
uicontrol('Parent',h.panelA,'Style','edit','BackgroundColor',...
            'white','fontsize',12,'Position',[10 285 45 35]);

h.edt(2) = uicontrol('Parent',h.panelA,'Style','edit',...
           'BackgroundColor','white',...
           'fontsize',12,'Position',[70 285 45 35]);

h.bt(1) = uicontrol('Parent',h.panelA,'BackgroundColor','y',...
           'Style','pushbutton', 'String', 'GET
DATA','fontsize',10,...
           'callback',(@CreateGraph),'Position',[150,285,70,35]);

h.bt(6) =
uicontrol('Parent',h.panelA,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@ExportGraphData),...

'String','EXPORT','fontsize',10,'Position',[150,100,70,35]);
```

```
h.bt(4) =
uicontrol('Parent',h.panelA,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@DelEdges),...
          'String','DELETE','fontsize',10,'Position',[150,50,70,35]);

h.btadd(1) =
uicontrol('Parent',h.panelA,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@ADDEdges),...
          'String','ADD','fontsize',10,'Position',[150,10,70,35]);

h.bt(5) =
uicontrol('Parent',h.panelA,'BackgroundColor','y','Style',...

'pushbutton','Callback',(@UpdateGraphData),'visible','off',...
          'String','UPDATE','fontsize',10,'Position',[150 90 70 35]);


h.inpedt(1) = uicontrol('Parent',h.panelA,'Style','edit',...
           'BackgroundColor','white','fontsize',12,'Position',[10 10
45 35]);

h.inpedt(2) = uicontrol('Parent',h.panelA,'Style','edit',...
           'BackgroundColor','white',...
           'fontsize',12,'Position',[70 10 45 35]);

h.txted(1) = uicontrol('Parent',h.panelA,'Style','text','String',...
            'X ', 'BackgroundColor',[1,0.4,0.6],'fontsize',14,...
            'Position',[20,50,25,25]);

h.txted(2) = uicontrol('Parent',h.panelA,'Style','text','String',...
            'Y  ','Callback',(@addrow_colom),...
            'BackgroundColor',[1,0.4,0.6],'fontsize',14,...
            'Position',[85,50,25,25]);


h.adjlist = uicontrol('Parent',h.panelA,'style','list',...
            'unit','pix','position',[10 75 110 150],...
            'min',0,'max',2,'fontsize',12,'string',{});

h.bt(7)   =
uicontrol('Parent',h.panelA,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@graph_gui),...
          'String','OPEN
GUI','fontsize',10,'Position',[10,240,110,35]);

h.bt(8)   =
uicontrol('Parent',h.panelA,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@importadj),...
          'String','GET
DATA','fontsize',10,'Position',[150,240,70,35]);

h.plotBT =
uicontrol('Parent',h.panelA,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@PLOTGraph),...
          'String','PLOT','fontsize',10,'Position',[150,360,70,35]);
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
h.txtinput = uicontrol('Parent',h.panelA,'Style','text','String',...
             'DATA INPUT',
'BackgroundColor',[1,0.4,0.6],'fontsize',10,...
             'Position',[130,190,100,25]);

h.chmanual =
uicontrol('Parent',h.panelA,'Style','checkbox','String','MANUAL',...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
            'Position',[150 150 80 20],'callback',{@manual});

 h.chetool =
uicontrol('Parent',h.panelA,'Style','checkbox','String','ETOOL',...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
            'Position',[150 170 80 20],'callback',{@etool});


set(h.edt(1),'enable','off');
set(h.edt(2),'enable','off');set(h.bt(1),'enable','off')
set(h.bt(7),'enable','off');set(h.bt(8),'enable','off');

%=====================================================================
======

h.dialog =
figure('Visible','off','units','pixels','menubar','none','position',[
600 200 250 150]);

h.reset = uicontrol('Parent',h.dialog,'BackgroundColor',[0 0.9
0.1],'Style',...
          'pushbutton','Callback',(@resetbt),...
          'String','OK','fontsize',10,'Position',[100 40 50 35]);

h.txtdial =
uicontrol('Parent',h.dialog,'Style','text','String','SORRY..,DATA IS
NOT AVAILABLE ',...

'fontsize',11,'BackgroundColor',[1,0.4,0.6],'Position',[15,90,220,35]
);

%=====================================================================
%Components for input PPS parameters


h.bt(2) =
uicontrol('Parent',h.panelD,'BackgroundColor','y','Style',...
           'pushbutton','Callback',(@InputPDi),...
           'String','INPUT','fontsize',10,'Position',[25,12,70,35]);
```

```
h.bt(13) =
uicontrol('Parent',h.panelD,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@UpdateCallback),...
          'String','SAVE','fontsize',10,'Position',[120,12,70,35]);

h.txt(3) =
uicontrol('Parent',h.panelB,'Style','text','String','EDGE:',...
           'fontsize',10,'BackgroundColor',[1,0.4,0.6],...
           'Position',[2,455,50,25]);

h.edt3 = uicontrol('Parent',h.panelB,'Style','edit','fontsize',16,...
            'BackgroundColor','white','Position',[50,450,70,35]);

h.txtPAC =
uicontrol('Parent',h.panelB,'Style','text','String','PAC:',...
           'fontsize',10,'BackgroundColor',[1,0.4,0.6],...
           'Position',[125,455,50,25]);

h.edtPAC =
uicontrol('Parent',h.panelB,'Style','edit','fontsize',16,...
            'BackgroundColor','white','Position',[170,450,50,35]);


h.txt(6) = uicontrol('Parent',h.panelB,'Style','text','String','1.
METHOD            ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[5,320,110,25]
);

h.txt(7) =
uicontrol('Parent',h.panelB,'Style','text','String','D',...

'fontsize',11,'BackgroundColor',[1,0.4,0.6],'Position',[80,410,120,25
]);


h.pop(1) =
uicontrol('Parent',h.panelB,'Style','popupmenu','callback',{@InputAdv
Meth},...
                'String',{'Walking','Slow
Walk','Running','Crawling','Rolling','Jumping',...

'Tunneling','Trenching','Bridging','Cutting','Climbing','Land
Vehicle'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[20,275,200,50]);

h.edtm = uicontrol('Parent',h.panelB,'Style','edit',...
                'BackgroundColor','white',...
                'fontsize',16,'Position',[150,405,70,35]);


h.txt(7) = uicontrol('Parent',h.panelD,'Style','text','String','6.
EXTERIOR SENSORS ',...
```

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

```
'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[2,450,150,25]
,'visible','on');

h.cb(1) =
uicontrol('Parent',h.panelD,'Style','checkbox','BackgroundColor',[1,0
.4,0.6],...
            'fontsize',10,'Max',2,'Min',0,'Value',0,'Position',[18
425 50 20],'visible','on');

h.txt(8) = uicontrol('Parent',h.panelD,'Style','text','String','7.
INTERIOR SENSORS ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[2,390,150,25]
,'visible','on');

h.cb(2) =
uicontrol('Parent',h.panelD,'Style','checkbox','BackgroundColor',[1,0
.4,0.6],...
      'fontsize',10,'Max',2,'Min',0,'Value',0,'Position',[18 365 20
20],'visible','on');


h.txt(9) = uicontrol('Parent',h.panelD,'Style','text','String','8.
POSITION SENSORS ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[2,330,150,25]
,'visible','on');

h.cb(3) =
uicontrol('Parent',h.panelD,'Style','checkbox','BackgroundColor',[1,0
.4,0.6],...
      'fontsize',10, 'Max',2,'Min',0,'Value',0,'Position',[18 305 20
20],'visible','on');


h.txt(10) = uicontrol('Parent',h.panelD,'Style','text','String','9.
FENCE SENSORS     ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[2,270,150,25]
,'visible','on');

h.cb(4) =
uicontrol('Parent',h.panelD,'Style','checkbox','fontsize',10,'Backgro
undColor',[1,0.4,0.6],...
                'Max',2,'Min',0,'Value',0,'Position',[18 245 20
20],'visible','on');

h.txt(101) = uicontrol('Parent',h.panelD,'Style','text','String','10.
BARRIER SENSORS ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[2,210,150,25]
,'visible','on');
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
h.cb(5) =
uicontrol('Parent',h.panelD,'Style','checkbox','fontsize',10,'Backgro
undColor',[1,0.4,0.6],...
             'Max',2,'Min',0,'Value',0,'Position',[18 185 20
20],'visible','on');

h.pop(2) =
uicontrol('Parent',h.panelD,'Style','popupmenu','callback',{@InputSen
sor1},...
                'String',{'Electric Field','Micro Wave','Active
Infrared','Passive Infrared',...
                'Fence Motion','Taut Wire','Seismic','Seismic
Magnetic','Ported Coaxial',...
                'Fiber Optic Cable','Video Motion','Multiple non
Complementary','Multiple Complementary'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,400,180,50],'visible','on');

h.pop(3) =
uicontrol('Parent',h.panelD,'Style','popupmenu','callback',{@InputSen
sor2},...
                'String',{'Sonic','Capacitance','Micro Wave','Active
Infrared','Passive Infrared',...
                'Video Motion','Ultrasonic','Multiple non
Complementary','Multiple Complementary'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,340,180,50],'visible','on');

h.pop(4) =
uicontrol('Parent',h.panelD,'Style','popupmenu','callback',{@InputSen
sor3},...
                'String',{'Position Switch','Balanced Magnetic
Switch'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,280,180,50],'visible','on');

h.pop(5) =
uicontrol('Parent',h.panelD,'Style','popupmenu','callback',{@InputSen
sor4},...
                'String',{'Electric Field','Fence Motion','Taut
Wire','Vibration','Strain',...
                'Multiple Sensors'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,220,180,50],'visible','on');

h.pop(6) =
uicontrol('Parent',h.panelD,'Style','popupmenu','callback',{@InputSen
sor5},...
                'String',{'Glass Breakage','Vibration','Conducting
Tape','Grid Mesh',...
                'Multiple Sensors'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,160,180,50],'visible','on');

h.txt(11) = uicontrol('Parent',h.panelD,'Style','text','String','11.
ENTRY CONTROL     ',...
```

CHULALONGKORN UNIVERSITY

```
'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[2,150,150,25]
,'visible','on');


h.cb(6) =
uicontrol('Parent',h.panelD,'Style','checkbox','fontsize',10,'Backgro
undColor',[1,0.4,0.6],...
               'Max',2,'Min',0,'Value',0,'Position',[18 130 20
20],'visible','oN');

h.pop(7) =
uicontrol('Parent',h.panelD,'Style','popupmenu','callback',{@InputSen
sor6},...
               'String',{'Visual ID Check (ID)','Metal
Detector','Explosive Detector',...
               'SNM Detector For Personel',...
               'SNM Detector For Vehicles','Guard at Post'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,100,180,50],'visible','on');

%Components For Timely DETECTION

h.fencetxt = uicontrol('Parent',h.panelB,'Style','text','String','2.
FENCES 2.5 m ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[5,270,110,25]
);

h.chde(1) =
uicontrol('Parent',h.panelB,'Style','checkbox','Callback',(@Chfence),
...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[20 250 180 20]);

h.fence  =
uicontrol('Parent',h.panelB,'Style','popupmenu','callback',{@fences},
...
                'String',{'Chain-Link Mesh 1','Chain-Link Mesh
2','Chain-Link Mesh 3',...
                'Welded Wire Fabric Fence','Concrete Panel Wall'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,195,180,80]);

h.gatestxt = uicontrol('Parent',h.panelB,'Style','text','String','3.
GATES AND TURNSTILES',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[5,215,175,25]
);

h.chde(2) =
uicontrol('Parent',h.panelB,'Style','checkbox','callback',{@chgates},
...
```

```
'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[20 200 180 20]);

h.gates  =
uicontrol('Parent',h.panelB,'Style','popupmenu','callback',{@gates},.
..
                'String',{'Vehicle Gate','Pedestrian Gate','Steel
Turnstile',...
                'High Security Padlock'},...
                'fontsize',10,'BackgroundColor','w',...
                'Position',[40,145,180,80]);

h.wallstxt = uicontrol('Parent',h.panelB,'Style','text','String','4.
WALLS ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[5,160,67,25])
;

h.chde(3) =
uicontrol('Parent',h.panelB,'Style','checkbox','callback',{@chwalls},
...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[20 145 180 20]);

h.walls  =
uicontrol('Parent',h.panelB,'Style','popupmenu','callback',{@walls},.
..
                'String',{'Concrete 10x210','Concrete
15x210','Concrete 20x210','Concrete 30x210',...
                'Concrete 46x350','Concrete 60x350','Wood Studs and
Sheetrock Wall',...
                'Tempered Glass
Window'},'fontsize',10,'BackgroundColor','w',...
                'Position',[40,90,180,80]);

h.doorstxt = uicontrol('Parent',h.panelB,'Style','text','String','5.
DOORS ',...

'fontsize',10,'BackgroundColor',[1,0.4,0.6],'Position',[5,105,67,25])
;

h.chde(4) =
uicontrol('Parent',h.panelB,'Style','checkbox','callback',{@chdoors},
...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[20 90 180 20]);

h.doors  =
uicontrol('Parent',h.panelB,'Style','popupmenu','callback',{@doors},.
..
```

```
                'String',{'Standard industrial pedestrian door
1','Standard pedestrian industrial door 2',...
                'Standard industrial pedestrian door 3','30-cm-Wood
Pedestrian door  ',...
                '10-cm-Wood Pedestrian door ','5-cm-Wood Pedestrian
door ',...
                 '5-cm-Wood Pedestrian door Wt Glass panel','0.75-cm
Steel Plate Door',...
                 'Class V or VI Vault door','High Security
Padlock',...
                 'Standard 10-cm Wooden Vehicle Door','Sheet metal
roll-up Vehicle Door',...
                 'Magazine Door','Heavy Vehicle Door','30-cm Wood
Vehicle Door',...
                 '60-cm Steel and Congrete Rolling Vehicle Door','30-
cm Steel and Congrete Rolling Vehicle Door',...
                 'Standard Industrial Vehicle Door Hollow
Steel'},'fontsize',10,'BackgroundColor','w',...
                'Position',[40,30,180,80]);

h.txt(4) =
uicontrol('Parent',h.panelB,'Style','text','String','PD',...
           'fontsize',11,'BackgroundColor',[1,0.4,0.6],...
           'Position',[95,350,80,35]);


h.txttime =
uicontrol('Parent',h.panelB,'Style','text','String','Time',...
           'fontsize',11,'BackgroundColor',[1,0.4,0.6],...
           'Position',[5,350,50,35]);
h.txtsdtime =
uicontrol('Parent',h.panelB,'Style','text','String','SD-F',...
           'fontsize',11,'BackgroundColor',[1,0.4,0.6],...
           'Position',[5,400,50,35]);

h.tgjk = uicontrol('Parent',h.panelC,'Style','edit','fontsize',14,...
            'BackgroundColor','white','Position',[230,65,70,35]);

h.time = uicontrol('Parent',h.panelB,'Style','edit','fontsize',14,...
            'BackgroundColor','white','Position',[50,360,70,35]);

h.sdtime =
uicontrol('Parent',h.panelB,'Style','edit','fontsize',14,...
            'BackgroundColor','white','Position',[50,405,70,35]);

h.chTg =
uicontrol('Parent',h.panelC,'Style','checkbox','String','RESPONSE
EARLIER',...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[230 110 160 20]);
h.txtRFT =
uicontrol('Parent',h.panelC,'Style','text','String','RFT',...
           'fontsize',12,'BackgroundColor',[1,0.4,0.6],...
           'Position',[300,60,50,35]);
```

```
h.detectm =
uicontrol('Parent',h.panelB,'Style','edit','fontsize',14,...
            'BackgroundColor','white','Position',[150,360,70,35]);


%Components for Input PATH
h.numOfPath =
uicontrol('Parent',h.panelC,'Style','text','String','N.PATHS:',...
           'fontsize',10,'BackgroundColor',[1,0.4,0.6],...
           'Position',[5,25,60,25]);



h.numpath =
uicontrol('Parent',h.panelC,'Style','edit','String','',...
           'fontsize',13,'BackgroundColor','white',...
           'Position',[75,25,45,35]);

h.plotpath =
uicontrol('Parent',h.panelC,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@PlotPaths),...
          'String','PLOT','fontsize',10,'Position',[135,25,70,35]);

h.txt(3) = uicontrol('Parent',h.panelC,'Style','text','String','START
-   TARGET ',...
           'fontsize',10,'BackgroundColor',[1,0.4,0.6],...
           'Position',[5,60,125,25]);

h.tnode =
uicontrol('Parent',h.panelC,'Style','edit','fontsize',12,...
            'BackgroundColor','white','Position',[75 95 45 35]);

h.snode =
uicontrol('Parent',h.panelC,'Style','edit','fontsize',12,...
            'BackgroundColor','white','Position',[10 95 45 35]);

h.createpath =
uicontrol('Parent',h.panelC,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@CreatePATHS),...
          'String','INPUT', 'busyaction','cancel',...
                 'interrupt','off','fontsize',10,'Position',[135 95
70 35]);

%====================================================================
===
%Components for PPS Analysis

h.txtDET =
uicontrol('Parent',h.panelD,'Style','text','String','LOCATION OF
DETECTION',...
           'fontsize',10,'BackgroundColor',[1,0.4,0.6],...
           'Position',[15,75,180,35]);
```

```
h.Lef(1) =
uicontrol('Parent',h.panelD,'Style','checkbox','String','B',...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[15 65 30 20],'callback',(@detlocb));

h.Lef(2) =
uicontrol('Parent',h.panelD,'Style','checkbox','String','M',...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[90 65 30 20],'callback',(@detlocm));

h.Lef(3) =
uicontrol('Parent',h.panelD,'Style','checkbox','String','E',...

'fontsize',10,'Max',1,'Min',0,'Value',0,'BackgroundColor',[1,0.4,0.6]
,...
           'Position',[160 65 30 20],'callback',(@detloce));

h.PD = uicontrol('Parent',h.panelC,'BackgroundColor','y',...
            'Style','pushbutton', 'String',
'P.DETECTION','fontsize',10,...
            'callback',(@calcPD),'Position',[410,100,150,35]);

h.PTD = uicontrol('Parent',h.panelC,'BackgroundColor','y',...
            'Style','pushbutton', 'String', 'P.TIMELY
DETECTION','fontsize',10,...
            'callback',(@calcPTD),'Position',[410,50,150,35]);



h.PE = uicontrol('Parent',h.panelC,'BackgroundColor','y',...
            'Style','pushbutton', 'String', 'P.EFECTIVENESS',...

'fontsize',10,'callback',(@calcPE),'Position',[410,5,150,35]);

%####################################################################
%Components for Equipments Tool

h.gm =
figure('Visible','off','units','pixels','menubar','none','position',[
325 13 520 630]);

h.chb(1) = uicontrol('Parent',h.gm,'Style','checkbox',...
            'String','No Equipment',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18 30
220 20]);

h.chb(2) = uicontrol('Parent',h.gm,'Style','checkbox',...
            'String','Ladder',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18 60
220 20]);

h.chb(3) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Tarpaulin',...
```

```
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18 90
220 20]);

h.chb(4) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Pliers',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
120 220 20]);

h.chb(5) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Manual Bolt
Cutter',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
150 220 20]);

h.chb(6) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Circular
Saw',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
180 220 20]);

h.chb(7) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Manual Bolt
Cutter Large Opening',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
210 220 20]);

h.chb(8) =
uicontrol('Parent',h.gm,'Style','checkbox','string','Circular Saw
Large Opening',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
240 220 20]);

h.chb(9) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Gloves',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
270 220 20]);

h.chb(10) =
uicontrol('Parent',h.gm,'Style','checkbox','String','SledgeHammer',..
.
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
300 220 20]);

h.chb(11) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Truck',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
330 220 20]);

h.chb(12) =
uicontrol('Parent',h.gm,'Style','checkbox','String','HandTools',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
360 220 20]);

h.chb(13) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 16
Kg',...
```

```
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
390 220 20]);

h.chb(14) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 10
Kg',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
420 220 20]);

h.chb(15) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 5
Kg',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
450 220 20]);

h.chb(16) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 4
Kg',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
480 220 20]);

h.chb(17) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 3
Kg',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
510 220 20]);

h.chb(18) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 2
Kg',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
540 220 20]);

h.chb(19) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 1
Kg',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
570 220 20]);

h.chb(20) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive 0.5
Kg',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[18
600 220 20]);

h.chb(21) = uicontrol('Parent',h.gm,'Style','checkbox','String','Hack
Saw',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
600 220 20]);

h.chb(22) = uicontrol('Parent',h.gm,'Style','checkbox','String','1.8
m Pry Bar',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
570 220 20]);
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
h.chb(23) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Power Saw',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
540 220 20]);

h.chb(24) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Sledgehammer,
Hand boltcutter',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
510 220 20]);

h.chb(25) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Sledgehammer,
Cutting torch',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
480 220 20]);

h.chb(26) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Sledgehammer,
Circular saw',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
450 220 20]);

h.chb(27) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Rotohammer,
Sledgehammer',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
420 220 20]);

h.chb(28) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive12',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
390 220 20]);

h.chb(29) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Explosive30',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
360 220 20]);

h.chb(30) =
uicontrol('Parent',h.gm,'Style','checkbox','String','Hammer',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
330 220 20]);


h.chb(31) =
uicontrol('Parent',h.gm,'Style','checkbox','String','FireAx',...
            'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
300 220 20]);

h.chb(32) =
uicontrol('Parent',h.gm,'Style','checkbox','String','CircSawFireAxPry
Bar',...
           'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
270 220 20]);
```

```
h.chb(33) =
uicontrol('Parent',h.gm,'Style','checkbox','String','SHammerPryBar',.
..
           'fontsize',10,'Max',2,'Min',1,'Value',1,'Position',[260
240 220 20]);


h.equiptxt = uicontrol('Parent',h.panelB,'Style','text','String','GET
EQUIPMENTS ',...

'fontsize',11,'BackgroundColor',[1,0.4,0.6],'Position',[5,50,150,25])
;

h.chreset = uicontrol('Parent',h.panelB,'style','check',...
                 'unit','pix',...
                 'position',[10,15,120,35],...
                 'string','SHOW LISTS',...
                 'fontsize',12);

set(h.chreset,'callback',{@ch_call,h})

h.resbt = uicontrol('Parent',h.panelB,'BackgroundColor',[0 0.9
0.1],'Style',...
       'pushbutton','Callback',(@DelayElement),...
      'String','GET DATA','fontsize',10,'Position',[150 15 72 35]);


h.btsave =
uicontrol('Parent',h.panelG,'BackgroundColor','y','Style',...
          'pushbutton','Callback',(@import_ppsdata),...
          'String','RECALL
DATA','fontsize',10,'Position',[25,12,100,35]);

h.fm = uimenu(h.f,...
              'label','Save State',...
              'callback',{@fm_call},...
              'enable','off');
h.fm(2) = uimenu(h.f,...
                 'label','Load State',...
                 'callback',{@fm_call},...
                 'enable','on');

%set global properties
    set(h.f,'Visible','on');

%=====================================================================
======
% Function 1 for Select Type of Graph
%=====================================================================
======
function SelectGraph(hObject,eventdata)
myhandles = guidata(gcbo);

cla(h.ax,'reset');
% Clear axes screen
```

```
L = get(h.adjlist,{'string'});
if ~isempty(L{1})
L{1} = [];                      % Delete the selected strings.
set(h.adjlist,'string',L{1});   % Set the new string.
end
% Clear All Graph Edges List

R(1)= get(h.rbh1,'Value');       % Option 1 = Grid Graph
R(2)= get(h.rbh2,'Value');       % Option 2 = Interactive Graph


if R(1)==1 && R(2)==0
set(h.adjlist,'string','');
set(h.edt(1),'enable','on');
set(h.edt(2),'enable','on');set(h.bt(1),'enable','on');
set(h.bt(7),'enable','off');set(h.bt(8),'enable','off');

elseif R(1)==0 && R(2)==1
set(h.adjlist,'string','')
set(h.edt(1),'enable','off');
set(h.edt(2),'enable','off'); set(h.bt(1),'enable','off');
set(h.bt(7),'enable','on');   set(h.bt(8),'enable','on');

elseif R(1)== 1 && R(2)== 1
set(h.adjlist,'string','Please Choose one');

end

guidata(gcbo,myhandles)
end

%======================================================================
% Function 2 for Create Interactive Graph
%======================================================================
function graph_gui(hObject,eventdata)

     interactive_graph_gui   % Independent GUI Code

end

%======================================================================
% Function 3 for Import Matrix Adjacency From Workspace
%======================================================================
function importadj(hObject,eventdata)
myhandles = guidata(gcbo);

v = evalin('base','adj');            % Import matrix with name file :
adj
v2 = evalin('base','xy');            % Import Coordinate location of G
adj
Aadj =full(v);

myhandles.Aadj=Aadj;                 % Save the data structure in Var
Aadj
myhandles.xyi=v2;
```

```
guidata(gcbo,myhandles);
end

%=====================================================================
% Function 4 for Import ReCall PPS Data in Matrix Form from Workspace
%=====================================================================
function import_ppsdata(hObject,eventdata,handles)
myhandles = guidata(gcbo);
% Usefull for reAnalysis previous data
MoD = evalin('base','MoD');
MoVr = evalin('base','MoVr');
MoTT = evalin('base','MoTT');
MPD = evalin('base','MPD');
MoTD = evalin('base','MoTD');
MoSD =evalin('base','MoSD');
MoDL =evalin('base','MoDL');

myhandles.A5 = MoD;
myhandles.A4 = MoVr;
myhandles.A3 = MoTT;
myhandles.A1 = MPD;
myhandles.A2 = MoTD;
myhandles.A7 = MoSD;
myhandles.A6 = MoDL;

guidata(gcbo,myhandles);
end

%=====================================================================
% Function 5 for Create Grid Graph
%=====================================================================
function CreateGraph(hObject,eventdata)
myhandles = guidata(gcbo);

    R = str2double(get(h.edt(1),'string'));
    C = str2double(get(h.edt(2),'string'));

%Function for create graph
    [X,Y] = gen_mesh(R,C);          % create mesh with
        A = gen_adjacency(X,Y,1);   % gen_adjacency is independent
algorithm
      Asp = sparse(A);

     assignin('base','adj',Asp);
  %edge_list_prime = plot_adjacency(A,X,Y);

%=====================================================================
   %  listofArcs = num2str(edge_list_prime);
   %  set(h.adjlist,'str',listofArcs)
%=====================================================================
pos=[X,Y];

 assignin('base','xy',pos);


myhandles.pos = pos;
```

```
myhandles.A = A;
myhandles.Asp = Asp;
%myhandles.edge_list_prime =edge_list_prime; %global list for all
arcs
%myhandles.listofArcs =listofArcs;

%#####################################################################
######
%display the variable in ( h.Result )

%set(h.Result,'string',A)
%assignin('base','A',A)
%assignin('base','xy',[X Y])
guidata(gcbo,myhandles);

end

%=====================================================================
% Function 6 for Plot the Graph
%=====================================================================
function PLOTGraph(hObject,eventdata)
myhandles = guidata(gcbo);
At = myhandles.A ;
 pos1 = myhandles.pos ;
Aadj = myhandles.Aadj;
 pos2 = myhandles.xyi;

assignin('base','At',At)
cla(h.ax,'reset');

R(1)= get(h.rbh1,'Value');
R(2)= get(h.rbh2,'Value');


 if R(1)==1 && R(2)==0
       Af=At;
       pos = pos1;

elseif R(1)==0 && R(2)==1
         Af=Aadj;
       pos = pos2;
 end

 X = pos(:,1);
 Y = pos (:,2);

 Asp = sparse(Af);
 edge_list_prime = plot_adjacency(Af,X,Y);
 listofArcs = num2str(edge_list_prime);
 set(h.adjlist,'str',listofArcs);

myhandles.pos = pos;
myhandles.A = Af;
myhandles.A1= Af;
myhandles.A2= Af;
myhandles.A3= Af;
```

```
myhandles.A4= Af;
myhandles.A5= Af;
myhandles.A6 =Af;
myhandles.A7 =Af;
myhandles.Asp = Asp;
myhandles.edge_list_prime =edge_list_prime; %global list for all arcs
myhandles.listofArcs =listofArcs;

guidata(gcbo,myhandles);
end

%=====================================================================
======
% Function 7 for Edit Graph (Add edges in Graph)
%=====================================================================
======
function ADDEdges(hObject,eventdata)
asp= myhandles.Asp;
pos= myhandles.pos;


x1 = str2num(get(h.inpedt(1),'string'));
x2 = str2num(get(h.inpedt(2),'string'));

asp(x1,x2)= 1;

X = pos(:,1);
Y = pos (:,2);

cla(h.ax,'reset');
edge_list_prime = plot_adjacency(asp,X,Y);
listofArcs = num2str(edge_list_prime);
set(h.adjlist,'str',listofArcs);

myhandles.pos=pos;
myhandles.Asp= asp ;
assignin('base','Asp',asp);
guidata(gcbo,myhandles);
end



%=====================================================================
======
%  Function 8 for Edit Graph (Delete edges in Graph)
%=====================================================================
======

function DelEdges(hObject,eventdata)
asp= myhandles.Asp;
pos= myhandles.pos;

val =cell2mat(get(h.adjlist,{'value'}));
b =cell2mat(get(h.adjlist,{'string'}));
c= str2num(b(val,:));
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
x1 = c(:,1);
x2 = c(:,2);

asp(x1,x2)=0; asp(x2,x1)=0;    %adjacency matrix changed
X = pos(:,1);
Y = pos (:,2);

cla(h.ax,'reset');
edge_list_prime = plot_adjacency(asp,X,Y);

listofArcs = num2str(edge_list_prime);
set(h.adjlist,'str',listofArcs);

myhandles.pos=pos;
myhandles.Asp= asp ;
assignin('base','Asp',asp);
guidata(gcbo,myhandles);
end

function UpdateGraphData(hObject,eventdata)
myhandles = guidata(gcbo);

cla(h.ax,'reset');
pos=myhandles.pos;
asp=myhandles.Asp;

X = pos(:,1);
 Y = pos (:,2);

edge_list_prime = update_plot_adjacency(asp,X,Y);

%=====================================================================
======
     listofArcs = num2str(edge_list_prime);
     set(h.adjlist,'str',listofArcs);
%=====================================================================
======




guidata(gcbo,myhandles);
end

%=====================================================================
======
function ExportGraphData (hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);

val =cell2mat(get(h.adjlist,{'value'}));
b =cell2mat(get(h.adjlist,{'string'}));
set(h.edt3,'string',b(val,:));
```

```
set(h.chb(1:33),'value',1);
guidata(gcbo,myhandles);
end

%==========================================================================
======
function manual(varargin)
%==========================================================================
======

switch get(h.chmanual,'value')
    case 0     % The text is visible, make it invisible.
        set(h.detectm,'enable','off');
        set(h.time,'enable','off');

    case 1    % The text is invisible, make it visible.
        set(h.detectm,'enable','on');  set(h.chetool,'value',0);
        set(h.time,'enable','on');
        set(h.walls,'enable','off');
        set(h.doors,'enable','off');
        set(h.fence,'enable','off');
        set(h.gates,'enable','off');
        set(h.cb(1:6),'enable','off')
        set(h.pop(2:7),'enable','off')
        set(h.chde(1:4),'enable','off');


end

end

%==========================================================================
======
function etool(varargin)
%==========================================================================
======

switch get(h.chetool,'value')
    case 0     % The text is visible, make it invisible.
        set(h.detectm,'enable','on');
        set(h.time,'enable','on');

    case 1    % The text is invisible, make it visible.
        set(h.detectm,'enable','off');     set(h.chmanual,'value',0);
        set(h.time,'enable','off');
        set(h.walls,'enable','on');
        set(h.doors,'enable','on');
        set(h.fence,'enable','on');
        set(h.gates,'enable','on');
        set(h.cb(1:6),'enable','on')
        set(h.pop(2:7),'enable','on')
        set(h.chde(1:4),'enable','on');
end
end

%==========================================================================
======
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
function detlocb(varargin)
%=====================================================================
======

switch get(h.Lef(1) ,'value')
    case 0     % The text is visible, make it invisible.
        set(h.Lef(2:3) ,'enable','on');

    case 1    % The text is invisible, make it visible.
        set(h.Lef(2:3) ,'enable','off');
set(h.Lef(2:3),'value',0);

end
end

%=====================================================================
======
function detlocm(varargin)
%=====================================================================
======

switch get(h.Lef(2) ,'value')
    case 0     % The text is visible, make it invisible.
        set(h.Lef(1) ,'enable','on');
        set(h.Lef(3) ,'enable','on');

    case 1    % The text is invisible, make it visible.
        set(h.Lef(1) ,'enable','off');     set(h.Lef(1),'value',0);
        set(h.Lef(3) ,'enable','off');     set(h.Lef(3),'value',0);
end
end

function detloce(varargin)
switch get(h.Lef(3) ,'value')
    case 0     % The text is visible, make it invisible.
        set(h.Lef(1:2) ,'enable','on');

    case 1    % The text is invisible, make it visible.
        set(h.Lef(1:2) ,'enable','off');
set(h.Lef(1:2),'value',0);

end
end

%=====================================================================
======
function Chfence(varargin)
%=====================================================================
======

switch get(h.chde(1),'value')
    case 0     % The text is visible, make it invisible.
        set(h.fence,'enable','on');
        set(h.gates,'enable','on');    %set(h.chgates,'value','Min');
        set(h.walls,'enable','on');   %set(h.chwalls,'value','Min');
        set(h.doors,'enable','on'); %set(h.chdoors,'value','Min');
```

```
    case 1    % The text is invisible, make it visible.
        set(h.fence,'enable','on');
        set(h.gates,'enable','off');    set(h.chde(2),'value',0);
        set(h.walls,'enable','off');    set(h.chde(3),'value',0);
        set(h.doors,'enable','off');    set(h.chde(4),'value',0);
end

end

%=====================================================================
======
function chgates(varargin)
%=====================================================================
======

switch get(h.chde(2),'value')
    case 0     % The text is visible, make it invisible.
        set(h.fence,'enable','on');
        set(h.gates,'enable','on');     %set(h.chgates,'value','Min');
        set(h.walls,'enable','on');    %set(h.chwalls,'value','Min');
        set(h.doors,'enable','on'); %set(h.chdoors,'value','Min');

    case 1    % The text is invisible, make it visible.
        set(h.fence,'enable','off');    set(h.chde(1),'value',0);
        set(h.gates,'enable','on');
        set(h.walls,'enable','off');    set(h.chde(3),'value',0);
        set(h.doors,'enable','off');    set(h.chde(4),'value',0);
end


end

%=====================================================================
======
function chwalls(varargin)
%=====================================================================
======

switch get(h.chde(3),'value')
    case 0     % The text is visible, make it invisible.
        set(h.fence,'enable','on');
        set(h.gates,'enable','on');     %set(h.chgates,'value','Min');
        set(h.walls,'enable','on');    %set(h.chwalls,'value','Min');
        set(h.doors,'enable','on'); %set(h.chdoors,'value','Min');

    case 1    % The text is invisible, make it visible.
        set(h.fence,'enable','off');    set(h.chde(1),'value',0);
        set(h.gates,'enable','off');    set(h.chde(2),'value',0);
        set(h.walls,'enable','on');
        set(h.doors,'enable','off');    set(h.chde(4),'value',0);
end


end
```

```
%======================================================================
======
function chdoors(varargin)
%======================================================================
======

switch get(h.chde(4),'value')
    case 0     % The text is visible, make it invisible.
        set(h.fence,'enable','on');
        set(h.gates,'enable','on');    %set(h.chgates,'value','Min');
        set(h.walls,'enable','on');   %set(h.chwalls,'value','Min');
        set(h.doors,'enable','on'); %set(h.chdoors,'value','Min');

    case 1    % The text is invisible, make it visible.
        set(h.fence,'enable','off');     set(h.chde(1),'value',0);
        set(h.gates,'enable','off');     set(h.chde(2),'value',0);
        set(h.walls,'enable','off');     set(h.chde(3),'value',0);
        set(h.doors,'enable','on');
end


end


%======================================================================
======
function InputAdvMeth(hObject,eventdata)
%======================================================================
======

myhandles = guidata(gcbo);

    contents = cellstr(get(hObject,'String'));
    ch= contents{get(hObject,'Value')};

     if (strcmp(ch,'Walking'))
            npdet=1;vrate=4;
     elseif(strcmp(ch,'Slow Walk'))
            npdet=2;vrate=4;
     elseif(strcmp(ch,'Running'))
            npdet=3;vrate=4;
     elseif(strcmp(ch,'Crawling'))
            npdet=4;vrate=4;
     elseif(strcmp(ch,'Rolling'))
            npdet=5;vrate=4;
     elseif(strcmp(ch,'Jumping'))
            npdet=6;vrate=4;
     elseif(strcmp(ch,'Tunneling'))
            npdet=7;vrate=4;
     elseif(strcmp(ch,'Trenching'))
            npdet=8;vrate=4;
     elseif(strcmp(ch,'Bridging'))
            npdet=9;vrate=4;
     elseif(strcmp(ch,'Cutting'))
```

```
            npdet=10;vrate=4;
     elseif(strcmp(ch,'Climbing'))
            npdet=11;vrate=4;
     elseif(strcmp(ch,'Land Vehicle'))
            npdet=16;vrate = 16.7; %v= m/s
     end


%*********************************************************************
****


myhandles.advmeth = npdet;
myhandles.vrate = vrate;

guidata(gcbo,myhandles)

end

%=====================================================================
======
function fences(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);

contents = cellstr(get(hObject,'String'));
chn = contents{get(hObject,'Value')};

    if (strcmp(chn, 'Chain-Link Mesh 1'))
    a=1; b=1;
    set(h.chb(10:33),'enable','off');

    elseif(strcmp(chn,'Chain-Link Mesh 2'))
    a=1; b=2;
    set(h.chb(1:33),'enable','off');
    set(h.chb(2),'enable','on');
    set(h.chb(4:6),'enable','on');


    elseif(strcmp(chn,'Chain-Link Mesh 3'))
    a=1;   b=3;
  %  set(h.gm,'Visible','on');
    set(h.chb(1:33),'enable','off');
    set(h.chb(1:2),'enable','on');
    set(h.chb(4:5),'enable','on');


    elseif(strcmp(chn,'Welded Wire Fabric Fence'))
    a=1;   b=4;
  %  set(h.gm,'Visible','on');
    set(h.chb(1:33),'enable','off');
    set(h.chb(2),'enable','on');
    set(h.chb(5:6),'enable','on');
```

```
    elseif(strcmp(chn,'Concrete Panel Wall'))
    a=1;   b=5;
  %  set(h.gm,'Visible','on');
    set(h.chb(1:33),'enable','off');
    set(h.chb(1:2),'enable','on');
    set(h.chb(10),'enable','on');

    end


myhandles.ADT = a;
myhandles.ADTtype=b;
guidata(gcbo,myhandles)

end

%=====================================================================
======
function gates(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);

    contents = cellstr(get(hObject,'String'));
    gates = contents{get(hObject,'Value')};

        if (strcmp(gates, 'Vehicle Gate'))
            a=2;  b=1;
            set(h.chb(1:33),'enable','off');
            set(h.chb(2),'enable','on');
            set(h.chb(4),'enable','on');
            set(h.chb(11),'enable','on');


        elseif(strcmp(gates,'Pedestrian Gate'))
            a=2;  b=2;
            set(h.chb(1:33),'enable','off');
            set(h.chb(5),'enable','on');
            set(h.chb(10),'enable','on');
            set(h.chb(21),'enable','on');
            set(h.chb(22),'enable','on');

        elseif(strcmp(gates,'Steel Turnstile'))
            a=2;  b=3;
            set(h.chb(1:33),'enable','off');
            set(h.chb(6),'enable','on');
            set(h.chb(12),'enable','on');



        elseif(strcmp(gates,'High Security Padlock'))
            a=2;  b=4;
            set(h.chb(1:33),'enable','off');
```

```
            set(h.chb(12),'enable','on');
            set(h.chb(19),'enable','on');
            set(h.chb(23),'enable','on');


        end

myhandles.ADT = a;
myhandles.ADTtype=b;
%assignin('base','a',a)
guidata(gcbo,myhandles)

end

%======================================================================
======
function walls(hObject,eventdata)
%======================================================================
======

myhandles = guidata(gcbo);

    contents = cellstr(get(hObject,'String'));
   walls = contents{get(hObject,'Value')};

        if (strcmp(walls, 'Concrete 10x210'))
            a=3;  b=1;
            set(h.chb(1:33),'enable','off');
            set(h.chb(24:27),'enable','on');
            set(h.chb(19),'enable','on');
            set(h.chb(17),'enable','on');
            set(h.chb(14:15),'enable','on');


        elseif (strcmp(walls, 'Concrete 15x210'))
            a=3;  b=2;
            set(h.chb(1:33),'enable','off');
            set(h.chb(17),'enable','on');
            set(h.chb(19),'enable','on');
            set(h.chb(15),'enable','on');




        elseif(strcmp(walls,'Concrete 20x210'))
            a=3;  b=3;
            set(h.chb(1:33),'enable','off');
            set(h.chb(15),'enable','on');
            set(h.chb(17:18),'enable','on');
            set(h.chb(27:28),'enable','on');

        elseif(strcmp(walls,'Concrete 30x210'))
            a=3;  b=4;
            set(h.chb(1:33),'enable','off');
            set(h.chb(15),'enable','on');
            set(h.chb(18),'enable','on');
```

```
            set(h.chb(27),'enable','on');


        elseif(strcmp(walls,'Concrete 46x350'))
            a=3;  b=5;
            set(h.chb(1:33),'enable','off');
            set(h.chb(13),'enable','on');


        elseif(strcmp(walls,'Concrete 60x350'))
            a=3;  b=6;
            set(h.chb(1:33),'enable','off');
            set(h.chb(29),'enable','on');

        elseif(strcmp(walls,'Wood Studs and Sheetrock Wall'))
            a=3;  b=7;
            set(h.chb(1:33),'enable','off');
            set(h.chb(1),'enable','on');
            set(h.chb(10),'enable','on');
            set(h.chb(6),'enable','on');

         elseif(strcmp(walls,'Tempered Glass Window'))
            a=3;  b=8;
            set(h.chb(1:33),'enable','off');
            set(h.chb(1),'enable','on');
            set(h.chb(12),'enable','on');
        end

myhandles.ADT = a;
myhandles.ADTtype=b;
%assignin('base','a',a)
guidata(gcbo,myhandles)

end

%=====================================================================
======
function doors(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);

   contents = cellstr(get(hObject,'String'));
   doors = contents{get(hObject,'Value')};


        if (strcmp(doors, 'Standard industrial pedestrian door 1'))
            a=4;  b=1;
            set(h.chb(1:33),'enable','off');
            set(h.chb(19),'enable','on');
            set(h.chb(25),'enable','on');
            set(h.chb(20),'enable','on');

        elseif(strcmp(doors,'Standard pedestrian industrial door 2'))
```

```
            a=4;  b=2;

            set(h.chb(1:33),'enable','off');
            set(h.chb(30:31),'enable','on');

        elseif(strcmp(doors,'Standard industrial pedestrian door 3'))
            a=4;  b=3;
            set(h.chb(1:33),'enable','off');
            set(h.chb(5),'enable','on');


        elseif(strcmp(doors,'30-cm-Wood Pedestrian door'))
            a=4;  b=4;
             set(h.chb(1:33),'enable','off');
            set(h.chb(32),'enable','on');
                  set(h.chb(19),'enable','on');
                  set(h.chb(17),'enable','on');

        elseif(strcmp(doors,'10-cm-Wood Pedestrian door '))
            a=4;  b=5;
            set(h.chb(1:33),'enable','off');
            set(h.chb(32),'enable','on');
                  set(h.chb(19),'enable','on');



        elseif(strcmp(doors,'5-cm-Wood Pedestrian door '))
            a=4;  b=6;
            set(h.chb(1:33),'enable','off');
            set(h.chb(31),'enable','on');
                  set(h.chb(19),'enable','on');
                  set(h.chb(6),'enable','on');


        elseif(strcmp(doors,'5-cm-Wood Pedestrian door Wt Glass 
panel'))
            a=4;  b=7;

            set(h.chb(1:33),'enable','off');
            set(h.chb(31),'enable','on');
                  set(h.chb(19),'enable','on');
                  set(h.chb(6),'enable','on');


        elseif(strcmp(doors,'0.75-cm Steel Plate Door'))
            a=4;  b=8;
            set(h.chb(1:33),'enable','off');
            set(h.chb(25),'enable','on');
                  set(h.chb(33),'enable','on');

        elseif(strcmp(doors,'Class V or VI Vault door'))
            a=4;  b=9;
            set(h.chb(1:33),'enable','off');
            set(h.chb(33),'enable','on');
                  set(h.chb(18),'enable','on');
```

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

```
elseif(strcmp(doors,'High Security Padlock'))
    a=4;  b=10;
    set(h.chb(1:33),'enable','off');
    set(h.chb(12),'enable','on');
          set(h.chb(19),'enable','on');
          set(h.chb(23),'enable','on');

elseif(strcmp(doors,'Standard 10-cm Wooden Vehicle Door'))
    a=4;  b=12;
    set(h.chb(1:33),'enable','off');
    set(h.chb(20),'enable','on');
          set(h.chb(25),'enable','on');
          set(h.chb(11),'enable','on');

elseif(strcmp(doors,'Sheet metal roll-up Vehicle Door'))
    a=4;  b=13;
    set(h.chb(1:33),'enable','off');
    set(h.chb(31),'enable','on');
          set(h.chb(19),'enable','on');

elseif(strcmp(doors,'Magazine Door'))
    a=4;  b=14;
 set(h.chb(1:33),'enable','off');
    set(h.chb(20),'enable','on');
          set(h.chb(11),'enable','on');
          set(h.chb(25),'enable','on');

elseif(strcmp(doors,'Heavy Vehicle Door'))
    a=4;  b=15;
    set(h.chb(1:33),'enable','off');
    set(h.chb(31),'enable','on');
          set(h.chb(19),'enable','on');
          set(h.chb(11),'enable','on');

elseif(strcmp(doors,'30-cm Wood Vehicle Door'))
    a=4;  b=16;
    set(h.chb(1:33),'enable','off');
    set(h.chb(20),'enable','on');
          set(h.chb(25),'enable','on');
          set(h.chb(6),'enable','on');


 elseif(strcmp(doors,'60-cm Steel and Congrete Rolling
Vehicle Door'))
    a=4;  b=17;

    set(h.chb(1:33),'enable','off');
    set(h.chb(25),'enable','on');
```

```
        elseif(strcmp(doors,'30-cm Steel and Congrete Rolling Vehicle
Door'))
            a=4;  b=18;
            set(h.chb(1:33),'enable','off');
            set(h.chb(32),'enable','on');
                  set(h.chb(19),'enable','on');
                  set(h.chb(17),'enable','on');

        elseif(strcmp(doors,'Standard Industrial Vehicle Door Hollow
Steel'))
            a=4;  b=11;
            set(h.chb(1:33),'enable','off');
            set(h.chb(25),'enable','on');
                  set(h.chb(16),'enable','on');



        end

myhandles.ADT = a;
myhandles.ADTtype=b;
%assignin('base','a',a)
guidata(gcbo,myhandles)

end

%=====================================================================
======
function DelayElement (hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
a= myhandles.ADT ;
b= myhandles.ADTtype;
Ade = myhandles.ADE;
assignin('base','ade',Ade)

del=cell2mat(get(h.chb(1:33),'value'));
       set(h.chb(11),'enable','off');
  if del(1)==2
       npdet = 12;
       se(1)= Ade(a,b).NoEquipment;

  end

  if del(2)==2
       npdet = 13;
       se(2)= Ade(a,b).Ladder;

  end

  if del(3)==2
       npdet = 13;
       se(3)= Ade(a,b).Tarpaulin;

  end
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
if del(4)==2
      npdet = 13;
      se(4)= Ade(a,b).Pliers;

end

if del(5)==2
      npdet = 13;
      se(5)= Ade(a,b).ManualBoltCutter;

end

if del(6)==2
    npdet = 13;
    se(6)= Ade(a,b).CircSaw;

end

if del(7)==2
     npdet = 13;
     se(7)= Ade(a,b).MBClargeOpen;

end

if del(8)==2
      npdet = 14;
      se(8)= Ade(a,b).CircSawLargeOpen;

end

if del(9)==2
    npdet = 13;
    se(9)= Ade(a,b).Gloves;

end

if del(10)==2
      npdet = 13;
      se(10)= Ade(a,b).SledgeHammer;

end

if del(11)==2
   npdet = 16;
   se(11)= Ade(a,b).Truck;

end

if del(12)==2
      npdet = 13;
      se(12)= Ade(a,b).Handtools;

end
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
if del(13)==2
     npdet = 15;
     se(13)= Ade(a,b).Explosive16;

end

if del(14)==2
      npdet = 15;
      se(14)= Ade(a,b).Explosive10;

end

if del(15)== 2
    npdet = 15;
    se(15)= Ade(a,b).Explosive5;

end

if del(16)==2
     npdet = 15;
     se(16)= Ade(a,b).Explosive4;

end

if del(17)==2
    npdet = 15;
    se(17)= Ade(a,b).Explosive3;

end

if del(18)==2
    npdet = 15;
    se(18)= Ade(a,b).Explosive2;

end

if del(19)==2
     npdet = 15;
     se(19)= Ade(a,b).Explosive1;

end

if del(20)==2
      npdet = 15;
      se(20)= Ade(a,b).Explosive05;

end

if del(21)==2
     npdet = 14;
     se(21)= Ade(a,b).Hacksaw;

end
```

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

```
if del(22)==2
    npdet = 14;
    se(22)= Ade(a,b).PryBar18;

end

if del(23)==2
     npdet = 14;
     se(23)= Ade(a,b).PowerSaw;

end

if del(24)==2
     npdet = 14;
     se(24)= Ade(a,b).SHammerHandboltCutter;

end

if del(25)==2
    npdet = 14;
    se(25)= Ade(a,b).SHammerCuttingtorch;

end

if del(26)==2
   npdet = 14;
   se(26)= Ade(a,b).SHammerCircSaw;

end

if del(27)==2
  npdet = 14;
  se(27)= Ade(a,b).RotoHammerSledgeH;
end

if del(28)==2
  npdet = 15;
  se(28)= Ade(a,b).Explosive12;
end

if del(29)==2
  npdet = 15;
  se(29)= Ade(a,b).Explosive30;
end

if del(30)==2
  npdet = 12;
  se(30)= Ade(a,b).Hammer;
end

 if del(31)==2
  npdet = 12;
  se(31)= Ade(a,b).FireAx;
 end
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
   if del(32)==2
    npdet = 13;
    se(32)= Ade(a,b).CircSawFireAxPryBar;
   end

   if del(33)==2
    npdet = 12;
    se(33)= Ade(a,b). SHammerPryBar;
   end

  SafeGuardElements = sum(se);

myhandles.SafeG = SafeGuardElements;
myhandles.npdet =npdet;
assignin('base','np',npdet);
assignin('base','SGE', SafeGuardElements);
guidata(gcbo,myhandles)
end

%=====================================================================
======
function InputSensor1(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
DBase = myhandles.DB;
a=  myhandles.advmeth;
a1= myhandles.npdet;

 contents = cellstr(get(hObject,'String'));
    sensor = contents{get(hObject,'Value')};

     if (strcmp(sensor,'Electric Field'))
           b(1:2)=[DBase(a).EF ,DBase(a1).EF];
     elseif(strcmp(sensor,'Micro Wave'))
            b(1:2)=[DBase(a).MW,DBase(a1).MW];
     elseif(strcmp(sensor,'Active Infrared'))
            b(1:2)=[DBase(a).AIR,DBase(a1).AIR];
     elseif(strcmp(sensor,'Passive Infrared'))
            b(1:2)=[DBase(a).PIR,DBase(a1).PIR];
     elseif(strcmp(sensor,'Fence Motion'))
            b(1:2)=[DBase(a).FM,DBase(a1).FM];
     elseif(strcmp(sensor,'Taut Wire'))
            b(1:2)=[DBase(a).TW,DBase(a1).TW];
     elseif(strcmp(sensor,'Seismic'))
            b(1:2)=[DBase(a).SE,DBase(a1).SE];
     elseif(strcmp(sensor,'Seismic Magnetic'))
           b(1:2)=[DBase(a).SEorMG,DBase(a1).SEorMG];
     elseif(strcmp(sensor,'Ported Coaxial'))
            b(1:2)=[DBase(a).PCo,DBase(a1).PCo];
     elseif(strcmp(sensor,'Fiber Optic Cable'))
            b(1:2)=[DBase(a).FOC,DBase(a1).FOC];
     elseif(strcmp(sensor,'Video Motion'))
            b(1:2)=[DBase(a).VM,DBase(a1).VM];
```

```
     elseif(strcmp(sensor,'Multiple non Complementary'))

b(1:2)=[DBase(a).ExteriorMultipleNC,DBase(a1).ExteriorMultipleNC];
     elseif(strcmp(sensor,'Multiple Complementary'))

b(1:2)=[DBase(a).ExteriorMultipleC,DBase(a1).ExteriorMultipleC];
     end

   if  b(1:2) == 100
       set(h.bt(13),'enable','off')
       set(h.dialog,'visible','on')
   end


   assignin('base','S',b)
myhandles.sensor1 = b;
%myhandles.PDSensor1 = Ss1;
guidata(gcbo,myhandles)

end

%=====================================================================
======
function InputSensor2(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
DBase = myhandles.DB;
a= myhandles.advmeth;
a1= myhandles.npdet;

 contents = cellstr(get(hObject,'String'));
    sensor = contents{get(hObject,'Value')};

     if (strcmp(sensor,'Sonic'))
            b(1:2)=[DBase(a).Sonic,DBase(a1).Sonic];
     elseif(strcmp(sensor,'Micro Wave'))
            b(1:2)=[DBase(a).MW,DBase(a1).MW];
     elseif(strcmp(sensor,'Active Infrared'))
            b(1:2)=[DBase(a).AIR,DBase(a1).AIR];
     elseif(strcmp(sensor,'Passive Infrared'))
            b(1:2)=[DBase(a).PIR,DBase(a1).PIR];
     elseif(strcmp(sensor,'Video Motion'))
            b(1:2)=[DBase(a).VM,DBase(a1).VM];
     elseif(strcmp(sensor,'Capacitance'))
           b(1:2)=[DBase(a).Capacitance,DBase(a1).Capacitance];
     elseif(strcmp(sensor,'Ultrasonic'))
           b(1:2)=[DBase(a).Ultrasonic,DBase(a1).Ultrasonic];
     elseif(strcmp(sensor,'Multiple non Complementary'))

b(1:2)=[DBase(a).InteriorMultipleNC,DBase(a1).InteriorMultipleNC];
     elseif(strcmp(sensor,'Multiple Complementary'))
            b(1:2)=[DBase(a). InteriorMultipleC,DBase(a1).
InteriorMultipleC];
     end
```

```
      if  b(1:2) == 100
        set(h.bt(13),'enable','off')
        set(h.dialog,'visible','on')
      end

myhandles.sensor2 = b;

guidata(gcbo,myhandles)


end

%=====================================================================
======
function InputSensor3(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
DBase = myhandles.DB;
a= myhandles.advmeth;
a1= myhandles.npdet;

 contents = cellstr(get(hObject,'String'));
    sensor = contents{get(hObject,'Value')};

     if (strcmp(sensor,'Position Switch'))
            b(1:2)=[DBase(a).PosSwitch,DBase(a1).PosSwitch];

     elseif(strcmp(sensor,'Balanced Magnetic Switch'))
            b(1:2)=[DBase(a).BalancedMagSwitch,DBase(a1).PosSwitch];

     end

      if  b(1:2) == 100
        set(h.bt(13),'enable','off')
        set(h.dialog,'visible','on')
      end
myhandles.sensor3 = b;
guidata(gcbo,myhandles)


end

%=====================================================================
======
function InputSensor4(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
DBase = myhandles.DB;
a= myhandles.advmeth;
```

```
a1= myhandles.npdet;

 contents = cellstr(get(hObject,'String'));
    sensor = contents{get(hObject,'Value')};

     if (strcmp(sensor,'Electric Field'))
            b(1:2)=[DBase(a).EF,DBase(a1).EF];
     elseif(strcmp(sensor,'Vibration'))
           b(1:2)=[DBase(a).FenceVibration,DBase(a1).FenceVibration];
     elseif(strcmp(sensor,'Fence Motion'))
            b(1:2)=[DBase(a).FM,DBase(a1).FM];
     elseif(strcmp(sensor,'Taut Wire'))
            b(1:2)=[DBase(a).TW,DBase(a1).TW];
     elseif(strcmp(sensor,'Strain'))
            b(1:2)=[DBase(a).Strain,DBase(a1).Strain];
     elseif(strcmp(sensor,'Multiple Sensors'))

b(1:2)=[DBase(a).FenceMultiSensor,DBase(a1).FenceMultiSensor];

     end

      if  b(1:2) == 100
        set(h.bt(13),'enable','off')
        set(h.dialog,'visible','on')
      end

myhandles.sensor4 = b;

guidata(gcbo,myhandles)


end

%=====================================================================
======
function InputSensor5(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
DBase = myhandles.DB;
a= myhandles.advmeth;
a1= myhandles.npdet;


 contents = cellstr(get(hObject,'String'));
    sensor = contents{get(hObject,'Value')};

     if (strcmp(sensor,'Glass Breakage'))
            b(1:2)=[DBase(a).GlassBreakage,DBase(a1).GlassBreakage];

     elseif(strcmp(sensor,'Vibration'))

b(1:2)=[DBase(a).BarrierVibration,DBase(a1).BarrierVibration];
     elseif(strcmp(sensor,'Conducting Tape'))
```

```
b(1:2)=[DBase(a).ConductingTape,DBase(a1).ConductingTape];
     elseif(strcmp(sensor,'Grid Mesh'))
            b(1:2)=[DBase(a).GridMesh,DBase(a1).GridMesh];
     elseif(strcmp(sensor,'Multiple Sensors'))

b(1:2)=[DBase(a).BarrierMultiSensor,DBase(a1).BarrierMultiSensor];
     end

      if  b(1:2) == 100
        set(h.bt(13),'enable','off')
        set(h.dialog,'visible','on')
      end

myhandles.sensor5 = b;

guidata(gcbo,myhandles)

end


%=====================================================================
======
function InputSensor6(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
DBase = myhandles.DB;
a= myhandles.advmeth;
a1= myhandles.npdet;

 contents = cellstr(get(hObject,'String'));
    sensor = contents{get(hObject,'Value')};

     if (strcmp(sensor,'Visual ID Check (ID)'))
            b(1:2)=[DBase(a).VisualCheckID,DBase(a1).VisualCheckID]

     elseif(strcmp(sensor,'Metal Detector'))
            b(1:2)=[DBase(a).MetalDetector,DBase(a1).MetalDetector];
     elseif(strcmp(sensor,'Explosive Detector'))

b(1:2)=[DBase(a).ExplosiveDetector,DBase(a1).ExplosiveDetector];
     elseif(strcmp(sensor,'SNM Detector For Personel'))

b(1:2)=[DBase(a).SNMDetForPerson,DBase(a1).SNMDetForPerson];
     elseif(strcmp(sensor,'SNM Detector For Vehicles'))

b(1:2)=[DBase(a).SNMDetForVehicle,DBase(a1).SNMDetForVehicle];
     elseif(strcmp(sensor,'Guard at Post'))
            b(1:2)=[DBase(a).GuardAtPost,DBase(a1).GuardAtPost];

     end

     if  b(1:2) == 100
        set(h.bt(13),'enable','off')
```

```
        set(h.dialog,'visible','on')
     end

myhandles.sensor6 = b;
guidata(gcbo,myhandles)


end

%=====================================================================
======
function InputPDi(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
vrate = myhandles.vrate;
%Ade = myhandles.ADE;
%a= myhandles.ADT;
%b= myhandles.ADTtype;
S1=myhandles.sensor1;
S2=myhandles.sensor2;S3=myhandles.sensor3;
S4=myhandles.sensor4;S5=myhandles.sensor5;
S6=myhandles.sensor6;
sfe = myhandles.SafeG;
assignin('base','sfe',sfe)
%---------------------------------------------------------------------
------
 p = str2num(get(h.edt3,'string'));  %get node list for input the
data
%---------------------------------------------------------------------
------

%INPUT 1. DISTANCE of PATH

dist = str2num(get(h.edtm,'string'));
eld(1,1:2) = p(1:2);
eld(1,3)=dist;
assignin('base','eld',eld);



%INPUT 2. ADV MOVE RATE
elvrate (1,1:2) = p(1:2);
elvrate(1,3)= vrate;
assignin('base','elvrate',elvrate);
ti = dist/vrate;

%INPUT 4. Time Delay
inst1=cell2mat(get(h.chde(1:4),'value'));

if inst1(1)==1
sens=sfe;
elseif inst1(2)==1
sens=sfe;
elseif inst1(3)==1
sens=sfe;
```

```
elseif inst1(4)==1
sens=sfe;
else
sens=0;
end

eltd (1,1:2) = p(1:2);
eltd(1,3)= sens;
assignin('base','eltd',eltd);

%INPUT 4. PD from INSTALLED SENSOR
%--------------------------------------------------------------------
------
y= cell2mat(get(h.cb(1:6),'value'));
%--------------------------------------------------------------------
------

if y(1)==2
s(1)=S1(1);
elseif y(1) == 0
    s(1)= 0;
end

if y(2)==2
    s(2)=S2(1);
    elseif y(2) == 0
    s(2)= 0;
end

if y(3)==2
    s(3)=S3(1);
    elseif y(3) == 0
    s(3)= 0;
end

if y(4)==2
    s(4)=S4(1);
    elseif y(4) == 0
    s(4)= 0;
end

if y(5)==2
    s(5)=S5(1);
    elseif y(5) == 0
    s(5)= 0;
end

if y(6)==2
    s(6)=S6(1);
    elseif y(6) == 0
    s(6)= 0;
end

%--------------------------------------------------------------------
------
x = length(s);
```

```
Pdprime = zeros(1,x);
z=zeros(1,x);
Pdo = 0;

for     i=1:x;
      %Pdprime(i) = (s(i));
     % z(i) = 1-Pdprime(i);
     z(i) = 1-s(i);
   PDiPrime = 1-(1-Pdo)*prod(z); % still need to count with previous
arcs

end
%-----------------------------------------------------------------------
------
%DETECTION LOCATION

loc=cell2mat(get(h.Lef(1:3),'value'));

if loc(1) == 1
 DL= 66;  % 'B';

elseif loc(2) == 1
 DL= 77; %'M';

elseif loc(3) == 1
 DL= 69; %'E';

elseif loc==0
 DL= 69 %'E';
end

eldl(1,1:2) = p(1:2);
eldl(1,3)= DL;

assignin('base','eldl',eldl)

%INPUT MANUAL
y=get(h.chmanual,'value');
x=str2num(get(h.detectm,'str'));
t=str2num(get(h.time,'str'));
sdf = str2num(get(h.sdtime,'str'));

if y==1
    PDi = x;
    timetravel = t;
    SD=sdf*t;
elseif y==0
    PDi = PDiPrime;
    timetravel(1) = ti
    timetravel(2) =sens
    SD= sdf*(sum(timetravel(1:2)));
end

%INPUT PD
elpd(1,1:2) = p(1:2);
elpd(1,3)= PDi;
```

```
%INPUT 3. TIME TRAVEL
eltt (1,1:2) = p(1:2);
eltt(1,3)= timetravel(1);

elsdt (1,1:2) = p(1:2);
elsdt(1,3)= SD;

assignin('base','eltt',eltt);
assignin('base','elpd',elpd);
assignin('base','elsdt',elsdt);
myhandles.ELD=eld;
myhandles.ELV=elvrate;
myhandles.ELTT = eltt;
myhandles.ELPD = elpd;
myhandles.ELTD = eltd;
myhandles.ELSDT = elsdt;
myhandles.ELDL = eldl;

guidata(gcbo,myhandles);

end

%=====================================================================
======
function resetbt(varargin)
%=====================================================================
======

% Callback for pushbutton.
 set(h.bt(13),'enable','on')
  set(h.dialog,'visible','off')
end

function UpdateCallback(hObject, eventdata, handles)
 eld = myhandles.ELD;
 MoD= myhandles.A5;
 MoTT= myhandles.A3;
 MoVr= myhandles.A4;
 MPD = myhandles.A1;
 MoTD = myhandles.A2;
  MoDL = myhandles.A6;
MoSD=myhandles.A7;

 elpd = myhandles.ELPD;
 elvrate= myhandles.ELV;
 eltt=myhandles.ELTT;
 eltd=myhandles.ELTD;
 elsdt=myhandles.ELSDT;
eldl=myhandles.ELDL;

  PAC= str2num(get(h.edtPAC,'string'));


%MATRIX Of DISTANCES
```

```
a=eld(1);
b=eld(1,2);
c=eld(1,3);
MoD(a,b)=c  ;
assignin('base','MoD',MoD);
%####################################################################
######
%MATRIX Of elVRATE
a1=elvrate(1);
b1=elvrate(1,2);
c1=elvrate(1,3);
MoVr(a1,b1)=c1  ;
assignin('base','MoVr',MoVr);

%MATRIX Of Time Travel
a2=eltt(1);
b2=eltt(1,2);
c2=eltt(1,3);
MoTT(a2,b2)=c2;
assignin('base','MoTT',MoTT);

%MATRIX Of Time Delay from Safeguard Elements
a4=eltd(1);
b4=eltd(1,2);
c4=eltd(1,3);
MoTD(a4,b4)=c4;
assignin('base','MoTD',MoTD);

%####################################################################
######
%MATRIX Of Prob Detection


a3=elpd(1);
b3=elpd(1,2);
c3=elpd(1,3);
MPD(a3,b3)=PAC*c3;

assignin('base','MPD',MPD);
%####################################################################
######
%MATRIX Of Time Standard Deviation
a5=elsdt(1);
b5=elsdt(1,2);
c5=elsdt(1,3);
MoSD(a5,b5)=c5;
assignin('base','MoSD',MoSD);


%MATRIX Of Delay Location
a6=eldl(1);
b6=eldl(1,2);
c6=eldl(1,3);
MoDL(a6,b6)=c6;
assignin('base','MoDL',MoDL);
```

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

```
myhandles.A5 = MoD;
myhandles.A4 = MoVr;
myhandles.A3 = MoTT;
myhandles.A1 = MPD;
myhandles.A2 = MoTD;
myhandles.A6 = MoDL;
myhandles.A7 = MoSD;

guidata(gcbo,myhandles);

end


%======================================================================
======
function CreatePATHS(hObject, eventdata)
%======================================================================
======

myhandles = guidata(gcbo);
Asp = myhandles.Asp;

%contents = cellstr(get(hObject,'String'));
 %   sensor = contents{get(hObject,'Value')};



start = str2num(get(h.snode,'string'));
target = str2num(get(h.tnode,'string'));
K = length(Asp)^2;

[ DIST, PATHS] = graphkshortestpaths( Asp, start, target, K )
%[ DIST1, PATHS1] = graphkshortestpaths( Asp, target, start, K )

PM  = zeros(length(PATHS));   %initial MATRIX
%PM1  = zeros(length(PATHS));   %initial MATRIX

for j=1:length(PATHS)

PM(j,(1: length(cell2mat(PATHS(j)))))= cell2mat(PATHS(j));
%PM1(j,(1: length(cell2mat(PATHS1(j)))))= cell2mat(PATHS1(j));

end


nPath = length(PM(:,1))%+length(PM1(:,1))


%[PM,nPath]  = path2matrix( PATHS );
set(h.numpath,'string',length(PATHS));


myhandles.PATHS = PATHS;
myhandles.PM = PM;
```

```
myhandles.nPath = nPath;

assignin('base','PM',PM);

%assignin('base','PM1',PM1);
%assignin('base','Asp',Asp);
assignin('base','PATH',PATHS);
%assignin('base','S',S);
%assignin('base','T',n2);
%assignin('base','nPath',nPath);
guidata(gcbo,myhandles)

end

%=====================================================================
======
%Callback Panel F

%=====================================================================
======
function PlotPaths(hObject, eventdata, handles)
%=====================================================================
======

myhandles = guidata(gcbo);
A= myhandles.A;   Af= myhandles.Asp;

Ladj=zeros(length(A)); L=size(A);
PATHS = myhandles.PATHS;
PM = myhandles.PM ;
nPath= myhandles.nPath;
pos= myhandles.pos;
X=pos(:,1);
Y=pos(:,2);

%for calculated linear indek  of all paths
for i=1:nPath
    d(i)=length(cell2mat(PATHS(i)));
    v = PM(i,1:(d(i)-1));              % V1
    w = PM(i,2:(d(i)));                % V2  (Edge1 = V1-V2)
    lin{i} = sub2ind(L,v,w);           % L = size matrix adj


end

NP = str2num(get(h.numpath,'string'));

plind=cell2mat(lin(NP));
Ladj(plind)=1;

figure
plot_adjacency(Af,X,Y);
plot_path_adj(Ladj,X,Y); % plot the list of PATH
```

```
myhandles.linind=lin;
myhandles.Ladj=Ladj;
%assignin('base','plind',plind);
assignin('base','PM',PM);
guidata(gcbo,myhandles)

end

%#####################################################################
######
%DELAY EQUIPMENTS

%=====================================================================
======
function ch_call(varargin)
%=====================================================================
======

% Callback for pushbutton.
%h = varargin{3};  % Get the structure.

switch get(h.gm,'visible')
   case 'off'     % The text is visible, make it invisible.
      set(h.chreset,'string','   CLOSE LISTS');
        set(h.gm,'visible','on')
 case 'on'    % The text is invisible, make it visible.
    set(h.chreset,'string','   SHOW LISTS');
   set(h.gm,'visible','off')
  otherwise  % This should never happen!
        disp('Matlab entered the twilight zone, aborting.')
        close(h.gm)
       quit
end

end




%#####################################################################
######

%#####################################################################
######
%MAIN FUNCTION TO FINAL CALCULATION
%=====================================================================
======
function calcPD(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
A =myhandles.A;
nPath = myhandles.nPath;
```

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

```
PM= myhandles.PM ;
MPd = myhandles.A1;

PATHS = myhandles.PATHS; % Global PATH

%-----------------------------------------------------------------------
------

L=size(A);

for i=1:nPath
    d(i)=length(cell2mat(PATHS(i)));
    v = PM(i,1:(d(i)-1));
    w = PM(i,2:(d(i)));
    lin{i} = sub2ind(L,v,w);
    J(i) = {MPd(cell2mat(lin(i)))};          %convert matrix to list
end

%PTDf = cell2mat(KL);
assignin('base','J',J)
assignin('base','linear',lin{i})



dl=d-1;
for j=1:(length(PATHS))
L(j,(1: length(cell2mat(J(j)))))= cell2mat(J(j)); %left
Pdj(j,(1: length(cell2mat(J(j)))))=(1-L(j,(1:
length(cell2mat(J(j)))))); % (1- PDj)
R(j,(1: length(cell2mat(J(j)))))= cumprod(Pdj (j,(1:
length(cell2mat(J(j)))))); % Sigma (1- PD)
end


assignin('base','L',L)
assignin('base','PQQ',Pdj)
assignin('base','R',R)



for i=1:nPath
for j=2:dl(i)
    PD(i,j)=L(i,j)*R(i,j);          %after balanced with the right
side
    PD(i,1)= L(i,1)
    PTD(i,j)=R(i,j);                % ( 1 - PD )
    PTD(i,1) =1-L(i,1)              % ( 1 - PD )
    PFD(i,1) = L(i,1)               % Need for PI calculations
    PFD(i,j) = L(i,j)*PTD(i,j-1)
end
end

%-----------------------------------------------------------------------
------
assignin('base','ProbTimelyDetection',PTD);
assignin('base','ProbDetection',PD);
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

```
assignin('base','PFDetection',PFD);
myhandles.PD=PD;
myhandles.dl=dl;
myhandles.PTD = PTD;
myhandles.PFD=PFD;
guidata(gcbo,myhandles);

end

%======================================================================
======
function calcPTD(hObject,eventdata)
%======================================================================
======

myhandles = guidata(gcbo);
PM = myhandles.PM; % Matrix of Global PATH
PATHS = myhandles.PATHS;
nPath = myhandles.nPath;
MoTD = myhandles.A2;
MoTT = myhandles.A3;
MoD= myhandles.A5;
MoDL = myhandles.A6;
MoSD = myhandles.A7;
PD = myhandles.PD;
PTD = myhandles.PTD;
PFD = myhandles.PFD;
A =myhandles.A;  % Matrix adjacency of all graph



 sdf = str2num(get(h.sdtime,'str'));
 RFT = str2num(get(h.tgjk,'string'));  %get response time
 Theta = get(h.chTg,'value');
 SDRFT = sdf*RFT;

assignin('base','SDRFT',SDRFT);
%######################################################################
######
 L=size(A);

for i=1:nPath
    d(i)=length(cell2mat(PATHS(i)));
    v = PM(i,1:(d(i)-1));
    w = PM(i,2:(d(i)));
    lin{i} = sub2ind(L,v,w)
    K(i) = {MoTT(cell2mat(lin(i)))};  %matrix timetravel
    J(i) = {MoTD(cell2mat(lin(i)))};  %matrix time delay
    J1(i)= {MoD(cell2mat(lin(i)))};  %matrix distances
    SD1(i)= {MoSD(cell2mat(lin(i)))}; %matrix SDev
  DLoc(i)= {MoDL(cell2mat(lin(i)))}; %matrix Delay Location

end

assignin('base','DLoc',DLoc);
```

```
dl=d-1;
dll=d-2;
dm=dll:-1:1;


for j=1:(length(PATHS))
Pt(j,(1: length(cell2mat(J(j)))))= cell2mat(J(j)); %list of time
delay
PtQ(j,(1: length(cell2mat(K(j)))))= cell2mat(K(j)); %time travel
M(j,(1: length(cell2mat(J1(j)))))= cell2mat(J1(j)); %distances
SD2(j,(1: length(cell2mat(SD1(j)))))= cell2mat(SD1(j));%Sdev
DLoc2(j,(1: length(cell2mat(DLoc(j)))))= cell2mat(DLoc(j)); % Delay
location
end

assignin('base','Dloc2',DLoc2);

assignin('base','SD2',SD2)


TMprime = Pt+PtQ; %adv time delay

%assignin('base','SD2',SD2);
assignin('base','Pt',Pt);
assignin('base','TMprime',TMprime);
assignin('base','PtQ',PtQ);

for k=1:nPath
       TMl(k,2:dl(k))=cumsum(TMprime(k,2:dl(k))); %late response need
to revise
       TMi(k,1:dl(k))=cumsum(TMprime(k,1:dl(k)));  %earlier response
       TGjk(k,1:dl(k))= RFT;       %response force time
       Dist(k)=sum(M(k,1:dl(k)));  % path distance
       SD3(k,1:dl(k))=(SD2(k,1:dl(k)).^2); %sDev^2
       cdel(k,dl(k):-1:1)=[TMprime(k,1:dl(k))]; %OK yg digunakan dlm
easi=tmi
       S4(k,dl(k):-1:1)=[SD2(k,1:dl(k)).^2];

end

assignin('base','cdel',cdel)
assignin('base','SD3',SD3)
assignin('base','S4CUM',S4)




cmdr=cumsum(cdel,2); %OK not yet reverse
CmVar=cumsum(S4,2); %sDev not yet reverse

%assignin('base','CumVar',CmVar)
assignin('base','Cmdr',cmdr)

for i=1:nPath
```

```
Comdelays(i,1:dl(i))=(cmdr(i,dl(i):-1:1)) ;   %OK
Comvar(i,1:dl(i))=(CmVar(i,dl(i):-1:1));

end
assignin('base','Cmdr1',Comdelays)
assignin('base','Cmvar1',Comvar)

RFT2=S4;
for i=1:nPath
for j=1:dl(i)
   RFT2(i,j) = (SDRFT)^2;
end
end
assignin('base','RFT2',RFT2)


for i=1:nPath
for j=1:dll(i)

if DLoc2(i,j)==66 %B
   TRUEVAR(i,j)=SD3(i,j)+Comvar(i,j+1)
   TRUEVAR(i,dl(i))=SD3(i,dl(i))

elseif DLoc2(i,j)== 77 %M
   TRUEVAR(i,j)=(0.25 * SD3(i,j)+Comvar(i,j+1))
   TRUEVAR(i,dl(i))=(0.25*SD3(i,dl(i)))

elseif DLoc2(i,j) == 69 %E
   TRUEVAR(i,j)= Comvar(i,j+1)

end
end
end

assignin('base','TRUEVAR',TRUEVAR)

for i=1:nPath
for j=1:dll(i)
if DLoc2(i,j)==66
    TRUEMEAN(i,j)=TMprime(i,j)+Comdelays(i,j+1);
    TRUEMEAN(i,dl(i))=TMprime(i,dl(i))

elseif DLoc2(i,j)== 77
    TRUEMEAN(i,j)=(0.5* TMprime(i,j)+Comdelays(i,j+1));
    TRUEMEAN(i,dl(i))=(0.5*TMprime(i,dl(i)))

elseif DLoc2(i,j) == 69
   TRUEMEAN(i,j)= Comdelays(i,j+1);
end

end
end

assignin('base','TRUEMEAN',TRUEMEAN)
%TV=TRUEVAR(i,1:dl(i))
```

```
%assignin('base','TV',TV)

for i=1:nPath
    for j=1:dl(i)
%Z1(i,j) = TRUEMEAN(i,j)-20
%Z2(i,j) = sqrt(TRUEVAR(i,j) + RFT2 (i,j))
Z(i,j)=  (TRUEMEAN(i,j)-  TGjk(i,j) ) / (sqrt(TRUEVAR(i,j) + RFT2
(i,j)))
CDP(i,j) = TRUEMEAN(i,j)-  TGjk(i,j)
NormalVal(i,j)=normcdf(Z(i,j))  %Normal Distribution
    end
end

assignin('base','CDP',CDP)
assignin('base','Z',Z)
assignin('base','NormalV',NormalVal)


for i=1:nPath
for j=1:dl(i)
    PTDf(i,j)=PFD(i,j)*NormalVal(i,j)       %

end
end

assignin('base','PTDf',PTDf);

for i=1:nPath
PI1(i,1:dl(i))=cumsum(PTDf(i,1:dl(i)))
end
assignin('base','PI1',PI1);

  PAC= str2num(get(h.edtPAC,'string'));

for i=1:nPath
PI(i,1)=(sum(PTDf(i,1:dl(i))))*PAC
end

assignin('base','PI',PI);

myhandles.PI1=PI1;
myhandles.PI=PI;

guidata(gcbo,myhandles);

end

%=====================================================================
======
function calcPE(hObject,eventdata)
%=====================================================================
======

myhandles = guidata(gcbo);
PI1=myhandles.PI1;
```

```
PD = myhandles.PD;
PTD =myhandles.PTD;
nPath = myhandles.nPath;
dl= myhandles.dl;

for i=1:nPath
for j=1:dl(i)
    PE(i,j)= PI1(i,j) ;
end
end

for i=1:nPath
     PEf(i,1) = sum(PE(i,1:dl(i)));
end


assignin('base','ProbEffectiveness',PEf);
assignin('base','ProbEffectiveness2',PE);
myhandles.PEf=PEf;
guidata(gcbo,myhandles);

end

end

%######################################################################
%#                         END OF THIS PROGRAM                        %#

%#                    created by : alim_m@batan.go.id                 %#
%#####################################################################
```

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

VITA

Mr. Alim Mardhi was born November 5, 1981 in Palembang, South Sumatera, Indonesia. He received his Bachelor of Engineering in the field of Mechanical Engineering in 2004 at Sriwijaya University Palembang. Design engineer was his first job in Manufacture Department, PT Indah Kiat tbk from 2004 to 2007. Since 2008, He has been working with BATAN, National Nuclear Energy Agency Indonesia as researcher. He has responsible for performing - reliability analysis and technology development in structural integrity of nuclear components, using a finite element method NASTRAN software as well as the experimental laboratory employing destructive and non-destructive test methods. In addition, he also highly - involved in developing various methods of remaining lifetime assessment (RLA) especially- for fatigue life behavior in nuclear installations.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University