The path analysis algorithm for a 2D map configuration of Physical Protection System

Sung Soon Jang
Physical Protection Devision, 1418 Yuseong-daero, Yuseong-gu, Daejeon 34101, Korea Institute of Nuclear Nonproliferation and Control

1. Introduction

A physical protection system (PPS) integrates people, procedures, and equipment for the protection of assets against theft, sabotage or other malevolent attacks. Even when a strong PPS is provided, without regular assessments, a PPS might waste valuable resources on unnecessary protection or, worse yet, fail to protect the asset.

In evaluating the effectiveness of a PPS, there are two main perspectives. The first addresses a path analysis of potential outside attacks and the second deals with neutralization. The concern in this abstract is with the path analysis.

Due to the complexity of protection systems, a path analysis usually requires computer modeling techniques. A path analysis determines the ordered series of a potential adversary's actions or the adversary path. The analysis evaluates the probability that a response force will interrupt this adversary before his/her task is completed. The Estimation of Adversary Sequence Interruption (EASI) calculates the probability of interruption for a pre-determined adversary path. EASI was developed in 1960. For a multi-path analysis, the Systematic Analysis of Vulnerability to Intrusion (SAVI) was developed in 1980. The Analytic System and Software for Evaluating Safeguards and Security (ASSESS) is an enhanced version of SAVI with additional insider analysis and neutralization modules. Also, computer-based combat simulators have been developed and are used for the assessment of physical protection system.

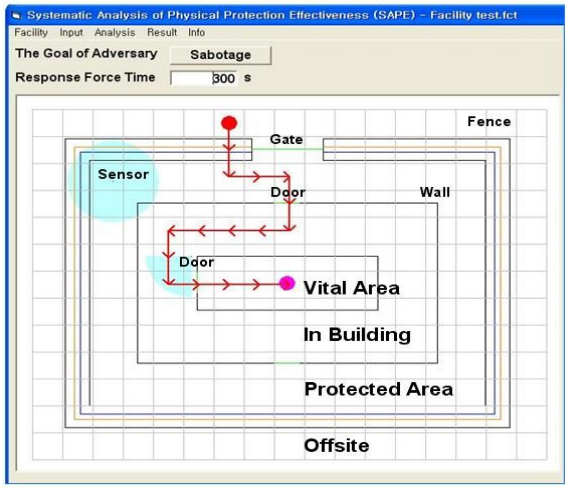

Figure 1: An adversary path on the 2D grid representation of physical protection system

This abstract addresses a path analysis for the 2D-square grid representation of physical protection system for a better representation, and proves the correctness proof of algorithm.

2. Timely Detection of a PPS

Weather a path is successful or not is measured by the Probability of Interruption (PI). The probability of interruption means the probability that security system detects adversary in time so to response/interrupt the adversary before his completion of the task.

The Timely Detection Model focuses on the measure PI as the measure of effectiveness of a path. The figure below depicts the adversary timeline at the top, indicating the Task Time it takes the adversary to complete his activities on the path, and also the sensing opportunities along the path which may cause the adversary to be detected. Below the adversary timeline there is a comparison between the PPS Response Time and the Adversary Task Time Remaining on the path after first sensing at each possible sensing opportunity.

If PRT (PPS Response Time) < Adversary Task Time Remaining After First Sensing then the corresponding sensing opportunity is considered timely; if this is not the case, then the opportunity is not timely.

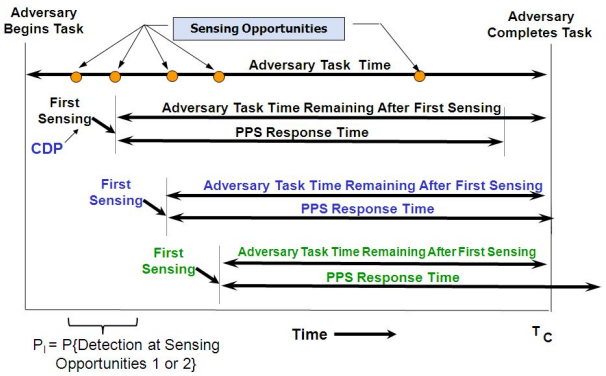

Figure 2: Relationship between the Adversary Timeline and the Response Timeline

The PI is equivalent to the probability that the adversary is detected at a timely sensing opportunity. For the example in Figure 2, the first two sensing opportunities are timely, so PI = P(Detection at Sensing Opportunity 1 OR Sensing Opportunity 2). The Critical Detection Point or CDP is the last sensing

opportunity on the path that is timely, in this case Sensing Opportunity 2.

From the adversary's perspective, their best path would have the lowest Probability of Interruption over all paths through the facility. Such a path achieving the lowest is called the most vulnerable path or MVP. To determine the MVP, the adversary will start at the end of the path, minimizing delay over elements and strategies at these elements, until a CDP is located; then probability of detection is minimized starting at the CDP and moving toward the start of the path. The PI along a path is evaluated as follows.

$$P_i = 1-(1-P(D_1))(1-P(D_2))...(1-P(D_k))$$

where $P(D_i)$ is the probability of detection of ith detector and k is the last detector before CDP.

3. The Algorithm to find the most vulnerable path

Conceptually, the approach for finding the MVP follows this same sequence:

1. Start at the target, identify the set of grids that are close enough that the quickest delay to the target is less than or equal to the PPS Response Time; then
2. Start with the boundary of this set; identify the minimum probability of detection path from that boundary to the starting point for the path.

Notice that this algorithm generates the path from the target node back to the starting node. Both of these steps are addressed by the A* algorithm presented below that keeps track of the timeline and switches from minimizing delay to minimizing detection through the use of a vector-valued evaluation function and comparison operator.

Let me write the algorithm in pseudo code. Let us suppose of the arithmetic operation of cost c as follows.

$$c' < c \quad if \ \ psum < psum'$$

$$if \ \ psum = psum' \ then \ tsum < tsum$$

$$c + c' = (psum + psum', tsum + tsum')$$

Then, the pseudo code is as follows.

```
For all t in target: // from all target
  cost = (0,0)
  push_heap([t], cost)
while heap is not empty:
  (p, c) = pop_heap()
  x = last_position_path(p)
  if reached_offsite(x) :                  // found the MVP!
    return (p)                             // exit
  for all y neighbor of x:
    if y is not closed:
      d = (-log(1-detect_prob(x,y)), delay_time(x,y))
      c' = c + d
      push_heap(add_path(p, y), c')
      close(y)
```

4. Correctness of the algorithm

The correctness means that the above algorithm finds the most vulnerable path, if any. It was proved that A* algorithm finds the least-cost path. The above pseudo code is equal to the A* algorithm when the heuristic value is zero.

Thus, it is enough to show that the cost function is monotonic increase as a path expansion. This condition is written as follows.

cost(a path to x) ≤ *cost*(a path to y through x), where y is the neighbor of x.

cost(a path to y through x) = *cost*(a path to x)
+ (-*log*(1-*detect_prob*(x,y)), *delay_time*(x,y))

The *detect_prob*(x,y) is the detection probability during the moving from x to y, the real value from 0 to 1. Therefore, -*log*(1- *detect_prob* (x,y) is a positive value and increases as *detect_prob* (x,y) increases. The *delay_time*(x,y) is the delay time during the moving from x to y and, thus, a positive real value. Because positive values are added to the cost, the cost function is a monotonic increase as a path expansion as follows.

cost(a path to y through x) = *cost*(a path to x)
+ (-*log*(1- *detect_prob* (x,y)), *delay_time*(x,y))
≥ *cost*(a path to x)

5. Conclusion

In conclusion, I suggest a path analysis algorithm for the 2D-square grid representation of PPS, and prove its correctness. The algorithm finds the most vulnerable path for the given physical protection system, and would be used in the computer simulator for the effectiveness evaluation of physical protection system.

ACKNOWLEDGEMENT

This work was sponsored by funding from the Korea Foundation of Nuclear Safety, Republic of Korea.

REFERENCES

[1] Mary Lynn Garcia, *Vulnerability Assessment of Physical Protection Syste*ms, Butterworth-Heinemann (2005).
[2] *SAVI: Systematic Analysis of Vulnerability to Intrusion*, v1, SAND89-0926, Sandia National Laboratories (1989).
[3] R. A. Al-Ayat et. al., *ASSESS update : Current status and future developments*, UCRL-JC-104360, Lawrence Livermore National Laboratory (1990).
[4] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach 2nd edition*, Prentice Hall (2002).
[5]. Sung Soon Jang, et. al, "*Development a Vulnerability Assessment Code for a Physical Protection System: SAPE* ", Nuclear Engineering and Technology, v41, n5, p747 (2009).