Multi-layered Security Systems for Force Protection

Nathanael J. K. Brown
Sandia National Laboratories
PO Box 5800, Albuquerque, NM 87185-1188
Phone: (505) 844-0664
Fax: (505) 284-0976
E-mail Address: jlgearh@sandia.gov

Katherine A. Jones
Sandia National Laboratories
PO Box 5800, Albuquerque, NM 87185-1188
Phone: (505) 284-2090
Fax: (505) 284-0976
E-mail Address: kajones@sandia.gov

Linda K. Nozick and
Ningxiong Xu
Cornell University
Phone: (607) 255-6496
lkn3@cornell.edu

Alisa Bandlow, Kristin L. Adair, Jared L. Gearhart, John L. Russell
Sandia National Laboratories

82nd MORS Symposium
Working Group 28
December 31, 2014

ABSTRACT

Identifying an optimal design for a physical security system is critical to mission performance. Historically, analyses of security systems have been performed using directed graph and path analysis tools like adversary sequence diagrams/attack graphs. However, there are many dimensions in the design space of a security system, including the selection of technologies, alternative locations/configurations, different threats, cost limitations, and impacts from false alarms. The multiple dimensions of this problem make it effectively impossible to evaluate all permutations of potential system architectures. In practice, the individuals configuring the system drive the examination of a small subset of candidate architectures. This type of process is likely to lead to suboptimal decisions. There have been several historical incidents that highlight a need for a more effective security system design process to protect national security assets.

To address this need, we have developed a game theoretic model to optimize the design of security systems which explicitly includes opportunities to layer security barriers and the performance of a range of different technologies based on how they are to be deployed. The model also includes the ability to consider budget limitations and the impact of false alarms on system performance. We demonstrate this model on a realistic problem instance.

INTRODUCTION

On July 28, 2012, three trespassers cut through security fences and reached the exterior of the Highly Enriched Uranium Materials Facility (HEUMF) at the Y-12 National Security Complex in Oak Ridge, Tennessee. This site is one of four production facilities in the Nuclear Security Enterprise. According to a report by the U.S. Department of Energy Office of the Inspector General (2012), the trespassers vandalized the building and were able to get through several layers of fencing before being physically observed and interrupted by security forces (U.S. Department of Energy, Office of Inspector General, Office of Audits and Inspections, 2012). This was a defeat of what was believed to be a highly reliable, layered security system. Issues cited by the Department of Energy (DOE) inquiry into the incident included poor response to alarms, failures to maintain critical security equipment, over-reliance on compensatory measures, misunderstanding of protocols, and miscommunication. Similar incidents have occurred in the past at facilities protecting high value assets.

The configuration of layered security measures is at the center of efforts to protect a range of systems, from high-value facilities to large-scale infrastructures. The research described here is part of a Laboratory Directed Research and Development (LDRD) Project at Sandia National Laboratories (Sandia) and was motivated in part by the Y-12 incident. The incident highlighted an ongoing need for innovative approaches to security system design. Sandia has a long history of providing technology and systems solutions to design, develop, test, and implement physical security technologies and systems to protect nuclear weapons and other high value assets and facilities. Mathematical modeling and complex systems analysis expertise at the laboratory, leveraging subject matter expertise from the physical security program, supported this research path.

Historically, analyses of security systems have been performed using directed graph and path analysis tools like Adversary Sequence Diagrams. However, there are many dimensions in the design space of a security system, including the selection of technologies, alternative locations/configurations, different threats, and competing cost limitations. The multiple dimensions of this problem make it effectively impossible to evaluate all permutations of potential system architectures. The experience of the individuals configuring the system drives the careful examination of a small subset of architectures.

The goal for this research effort is the creation of a consistent and robust mathematical framework using complex systems analysis algorithms and techniques to better understand the emergent behavior, vulnerabilities, and resiliency of layered security systems, subject to budget constraints and competing security priorities.

LITERATURE REVIEW

There are two ideas in the literature that are of particular relevance to this research. The first concept is *attack graphs*. Phillips and Swiler (1998) first defined attack graphs as a convenient mechanism to integrate the path choices facing an intruder attempting to reach a target. An edge represents a change in state in the system caused by a single action, and a node represents a possible state of the attack. The weights on the edges are the probabilities of success, the cost of the action, or some benefit-to-cost value associated with the particular action. This structure is then used to perform shortest path-type computations. The approach assumes the availability of a generic attack network template and an attacker profile, which is then used to customize the attack graph to the capabilities of the attacker. There has been substantial research to expand this core idea in a number of publications, including improved methods for the

generation of attacks that are then integrated into these attack graphs as in Chen et al. (2009), Ou el al. (2006) and Sheyner et al. (2002), as well as research to integrate minimum cost hardening measures as in Wang et al. (2013) and Chen et al. (2008). In practice, the construction of these graphs is quite complicated and, in some applications, overwhelming.

The second concept is *attacker-defender* models, which is an expansion of the types of attacks represented with attack graphs. This includes their extension to *defender-attacker-defender* models and the application of those models to many different domains. For example, see Romero et al. (2012), Arroyo and Fernández (2009) and Salmeron et al. (2009) for applications to power systems. See Alderson et al. (2011), Reilly et al. (2012), Murry-Tuite and Fei (2010) and Jones et al. (2006) for applications focused on transportation infrastructure.

Our mathematical model of interest has a defender-attacker structure for which we do not attempt to distill the path network representation down *a priori* into an attack graph. We use the probability of interruption given detection as the relevant measure of interest for the quality of the path from the perspective of the intruder. It is also one of the measures used to evaluate the performance of a collection of security investments. We develop a solution strategy for this model formulation using an extension of a label correcting algorithm and a global search technique.

Risk analysis for physical protection of critical infrastructure or facilities typically includes consideration of threat, vulnerability, and consequence as defined by Rinaldi (2004). The initial version of this model is primarily focused on characterizing vulnerability of the system to a specific threat. We use probability of detection, probability of interruption, delay time, and response time as defined by Garcia (2007) as elements in our model.

MODEL FORMULATION

The attacker is an *intruder* whom we assume has perfect knowledge of the security measures. Their goal is to reach a specific *target* within the network and then exit without being intercepted (interrupted) by the *protective force*. The defender is the operator of the security system and has a goal which can be represented by three objectives. The first objective is to minimize the investment cost. The second objective is to minimize the occurrence of *nuisance and false alarms* (*NAR/FAR*). The third objective is to maximize the *probability of interruption* (P_{int}), which is the probability that an intruder is detected and interrupted by the protective force while traversing the network. The last objective is accomplished by adding detection and delay elements to impede the progress of an intruder (e.g., fences, ditches, etc.) such that once detected, there is sufficient time for the protective force to intercept the intruder prior to them exiting the network.

More formally, we assume that there is a single *origin-destination pair* that is of interest. Let directional links be denoted by *(i,j).* Let z^r be a binary decision variable that indicates whether path *r* has been selected from a specific origin point to the target of interest. The goal of the intruder with a given security architecture is to minimize the probability that, if detected, they are intercepted by the protective force before leaving the system. To achieve this goal after being detected, the intruder must have a remaining travel time to exit which is less than the time needed by the protective force to respond. Explicitly, this can be represented as:

$$\min_{z^r} \sum_r g^r z^r \tag{1}$$

Such that

$$\sum_r z^r = 1 \quad (2)$$
$$z^r \in \{0,1\} \quad \forall r \quad (3)$$

where g^r is the probability that the time remaining in the path once detected exceeds the response time. The method for calculating g^r based on travel time and detection probability for each link is discussed in the Solution Methodology section below.

In order to identify the trade-off frontier between the probability of interruption given detection, system cost, and the nuisance alarm rate we construct the following multi-objective defender-attacker model given in equations (4)-(11).

$$\min_{I_{ij}^y} \sum_{ij} \sum_y c_{ij}^y I_{ij}^y \quad (4)$$

$$\min_{I_{ij}^y} \sum_{ij} \sum_y A_{ij}^y I_{ij}^y \quad (5)$$

$$\max_{I_{ij}^y} \left[\min_{z^r} \sum_r g^r z^r \right] \quad (6)$$

Such that

$$\sum_r z^r = 1 \quad (7)$$
$$z^r \in \{0,1\} \quad \forall r \quad (8)$$
$$I_{ij}^y \in \{0,1\} \quad \forall (i,j), y \quad (9)$$

where c_{ij}^y is the cost of investment *y* on arc *(i,j)*, A_{ij}^y is the nuisance alarm rate on arc *(i,j)* when investment *y* is made on arc *(i,j)*, and I_{ij}^y is a binary decision variable that takes on a value of 1 if investment *y* is made on arc *(i,j)* and is zero otherwise.

SOLUTION METHODOLOGY

We developed a genetic algorithm to solve the formulation given in equations (4)-(9). Before describing that algorithm, we focus on the solution method to identify the optimal path for the intruder, given a fixed collection of security measures. This step constitutes a computationally efficient method to solve the formulation given in (1)-(3). This step is also an important element of the genetic algorithm.

The computation of g^r can be illustrated as follows. Consider the three link path given in Figure 1. The intruder must proceed from A, to B, to C and finally to D. Assume the response time for the protective force is four minutes. The intruder could be detected on the first, second, or third link, or not at all. If they are detected on the third link (i.e., from C to D), there is insufficient time for the protective force to respond. But, if they are caught on the first or second links, there is sufficient time to respond. Hence, the coefficient *g* for this route is 0.25 + (1-0.25)*0.5=0.625, which is the probability that they are interrupted given the probabilities of the two possible interruption scenarios: they are detected on link A-B (0.25) or they are detected on link B-C (0.5) and *not* detected on link A-B (1-0.25).

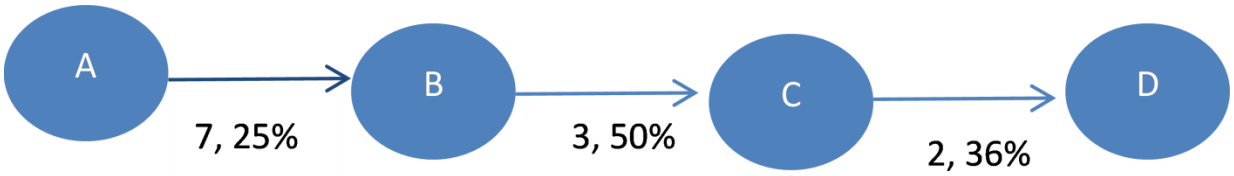

Figure 1: Example of computation g^r, with travel time and detection probability shown for each link

It is computationally inefficient to perform path enumeration to identify all the paths and then compute the values of g^r. Rather, an algorithm that does this enumeration implicitly to identify the optimal route *r* is more computationally attractive. Hence, we develop a modification of Dijkstra's algorithm, which is also a label correcting algorithm (LCA). For the remainder of this paper we simply refer to it as the LCA.

Let $\{N, Z\}$ be the directed graph where *N* is the set of nodes and *Z* is the set of links. Let δ_u^- be the set of links $(i,u) \in Z$, T_{ij} be the travel time of link $(i,j) \in Z$ and D_{ij} be the detection probability on the link. We assume that the detection probabilities are independent. Also, let T be the required time to interrupt the intruder. Let a be the origin and b be the target. We define $I_i = (I_{i1}, I_{i2})$ for each $i \in N$, where I_{i1} represents the shortest travel time from node i to target b if the travel time is shorter than T, and I_{i2} represents the probability that the intruder is interrupted when they are in node i. The objective for the intruder is to minimize I_{a2}. The following algorithm is used to find the probability.

1. Let $S \leftarrow b$, $I_b = (0,0)$, $C = \phi$, and $I_i = (inf, 1)$, $i \in N \backslash \{b\}$.
2. For $u \in S$, if $u = a$, then stop and report I_{a2}. Otherwise do the following:
    a. For each $v \in \delta_u^-$, If $I_{u1} + T_{vu} < I_{v1}$ and $I_{u1} + T_{vu} < T$, then let $I_{v1} = I_{u1} + T_{vu}$ and $I_{v2} = 0$.
    b. For each $v \in \delta_u^-$, if $I_{u1} + T_{vu} \geq T$ and $D_{vu} + (1 - D_{vu}) * I_{u2} < I_{v2}$, let $I_{v1} = I_{u1} + T_{vu}$ and $I_{v2} = D_{vu} + (1 - D_{vu}) * I_{u2}$.
    c. Find v^* that minimizes I_{v1} for $v \in N \backslash C \cup S$.
    d. If $I_{v^*1} < T$, then do the following:
        i. $S \leftarrow v^*$.
        ii. Go to step 3.
    e. If $I_{v^*1} \geq T$, do the following:
        i. Find v^{**} that minimizes I_{v2} for $v \in N \backslash C \cup S$.
        ii. $S \leftarrow v^{**}$.
        iii. Go to step 3.
3. Remove u from S, $C = C \cup \{u\}$ and go to step 2.

The algorithm starts at the target node *b* and steps through the graph in reverse order (in practice the node *b* will be a dummy exit node from the system). The set *S* contains the next node to be explored by the algorithm. The set *C* is the set of nodes that have been permanently labeled with a final travel time and probability of interruption. All nodes except *b* are temporarily labeled

with infinite travel times and likelihoods of detection of 1.0. At each step of the algorithm, a node to explore (node *u)* is identified. The temporary label for each node *v* that has a link leading to node *u* is updated if necessary. If the travel time from node *v* to node *b* through node *u* is shorter than *T* and the current travel time for node *v*, the travel time is set to the shorter travel time and the probability is set to zero for node *v*. For cases where the travel time from node *v* through node *u* exceeds *T* and the probability of detection is less than the current temporary label, the travel time and probability of detection are updated. This ensures that at each step of the algorithm each node is labeled with the values that correspond to the best known path for the intruder. Once each neighbor of node *u* has been explored, the next node to explore is identified. The current node and the permanently labeled nodes cannot be selected as the next node to explore. The node with the shortest travel time is explored next, unless the travel time is longer than *T*, in which case, the node with the lowest probability is considered next. The node that was just explored is added to the set of permanently labeled nodes. The algorithm is repeated until the next mode to consider is the origin node *a*.

Figure 2 is an illustration of the algorithm where the response time is assumed to be six time units. The links have been labeled with a travel time and a probability of detection. The goal of the intruder is to get from *a* to *b*.

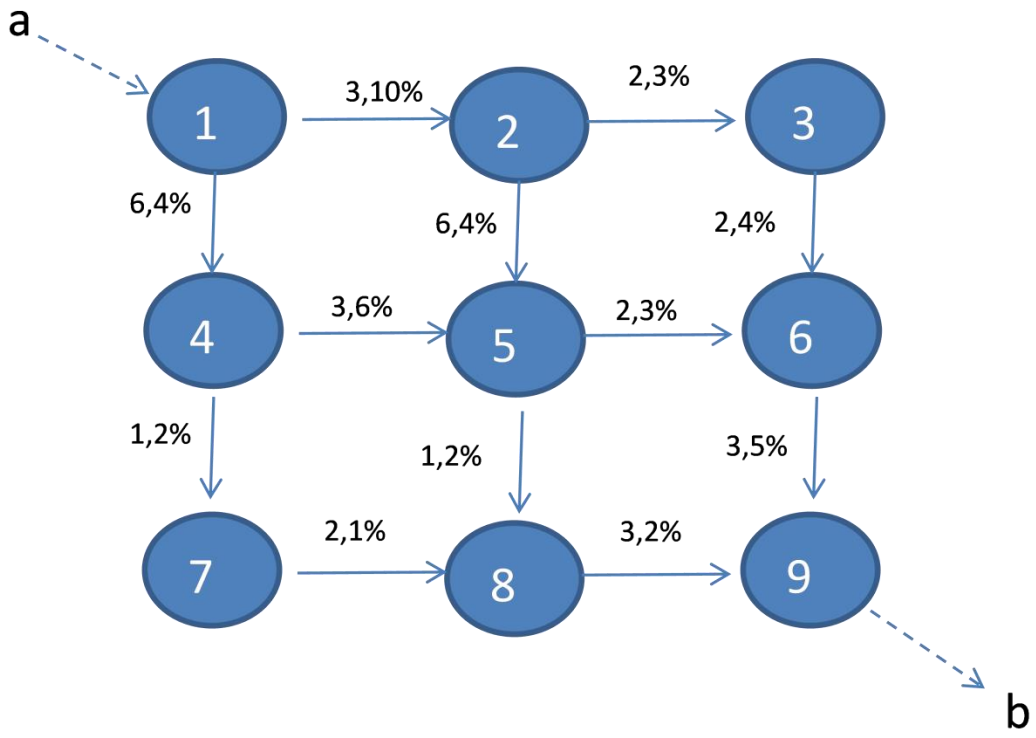

Figure 2: Illustrative security network

We execute the algorithm from *b* to *a* (i.e., reverse direction) because our interest is only in computing the relevant interruption probability once a partial path exceeds the response time of the protective force. Eight iterations are required in this example. A node is labeled with a belief of how much time a partial path from that node to the target will require and an estimate of the probability they will be detected if intercepted using that partial path. The first three iterations are as follows:

1. Node 9 has two neighbors: nodes 6 and 8. Each of these neighbors is given a new temporary label where the travel time is 3 for each, and the probability of interruption is zero, since the force response time of 6 is not exceeded for either partial path.
2. Since nodes 6 and 8 are equally attractive at this point, we arbitrarily choose node 6 as the next node to expand from. Node 6 has two neighbors: nodes 3 and 5. Each of these neighbors is given a new temporary label where the travel time is 5 for each, and the probability of interruption is zero, since the force response time of 6 is not exceeded for either partial path to node 9.
3. Node 8 is then evaluated for its two neighbors: nodes 5 and 7. The temporary label previously given to node 5 (5,0%) is replaced by (4,0%) because it is better to use a path from node 5 to the target via node 8. The new path requires only 4 units of time rather than 5 units of time given that neither of these partial paths has yet to accumulate any probability of interruption if detected. Node 7 is labeled with a temporary label of (5,0%).

After eight iterations, Figure 3 shows the final labeling of all nodes. Labels that have been crossed out were updated during the procedure (and we suppress the initial labels of $(inf, 1)$ for all nodes that are not the target for clarity). In labels where the travel time is six or greater, that value has been labeled as "F" indicating that the force response time has been met starting with that node and moving to the target. Node 2 is permanently labeled, and the question remains as to whether that partial path when extended to the origin is better than the one which has already been identified via node 4 because 3% < 2% * (1-4%) + 4%. However, since (1-10%) * 3% + 10% > 2% * (1-4%) + 4%, the old label is not replaced. At this point we can permanently label node 1, and the algorithm terminates. The best path for the intruder is (1,4), (4,7), (7,8) and (8,9) with a probability that, after detection, the remainder of the path travel time exceeds 6 is 5.92%.

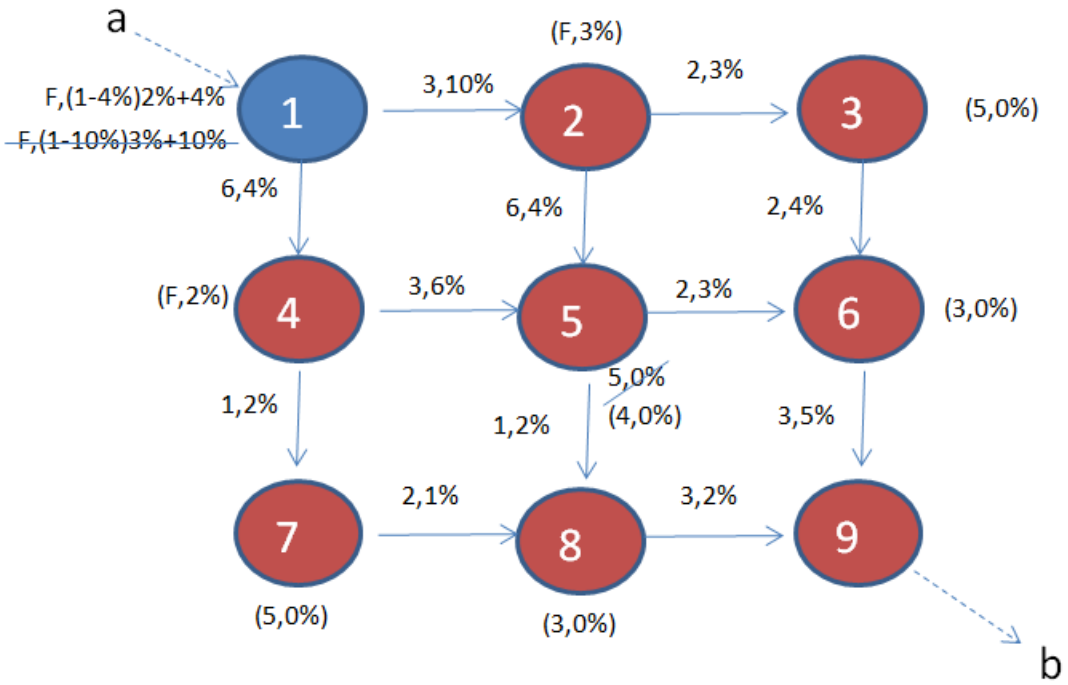

Figure 3: The eighth iteration

Given the computational complexity of the investment problem, we develop a genetic algorithm (GA) to search the solution space. This algorithm combines a GA with local search methods to create a set of Pareto optimal points similar to the techniques referenced in Konak et al. (2006) and Deb (2012). The solution procedure is summarized in the following list and in Figure 4.

1. **Select security investments to apply**: The decision regarding which investments to apply to which links drives this optimization. There are three objectives that must be simultaneously considered: minimization of security investment cost, minimization of NAR/FAR, and maximization of the intruder's probability of interruption. The initial collection of solutions will be a completely random selection, which has a size on the order of a few thousand. Each new collection of solutions is generated by the GA crossover/mutate process described in step 3.
2. **Determine the worst probability of interruption per solution:** This step is accomplished by selecting the "best" path from the perspective of the intruder, which is determined by the LCA described above.
3. **Crossover and Mutate:** After the first iteration, genetic crossover is used to create each new generation of potential security investments based on preserving desirable objective characteristics, but also to allow random mutation to provide solution diversity. The gene used for this algorithm is a vector of binary decision variables. Each entry in the vector is an ID which corresponds to a security investment of a specific type at a specific location within the network. Note that multiple investment types can appear at the same location (e.g., a fence and a magnetic sensor). By utilizing the biased parent selection criteria as described in Brown et al. (2013) and a region crossover procedure, each new pair of individuals is created via the genetic crossover of two parents which are relatively close to the Pareto frontier based on their fitness score. The same variable rate mutation strategy employed in Brown et al. (2013) is used to counter the tendency for crossover to produce homogenous populations as described in Sait (1999).
4. **Exit Strategy:** The algorithm is run a fixed number of iterations.

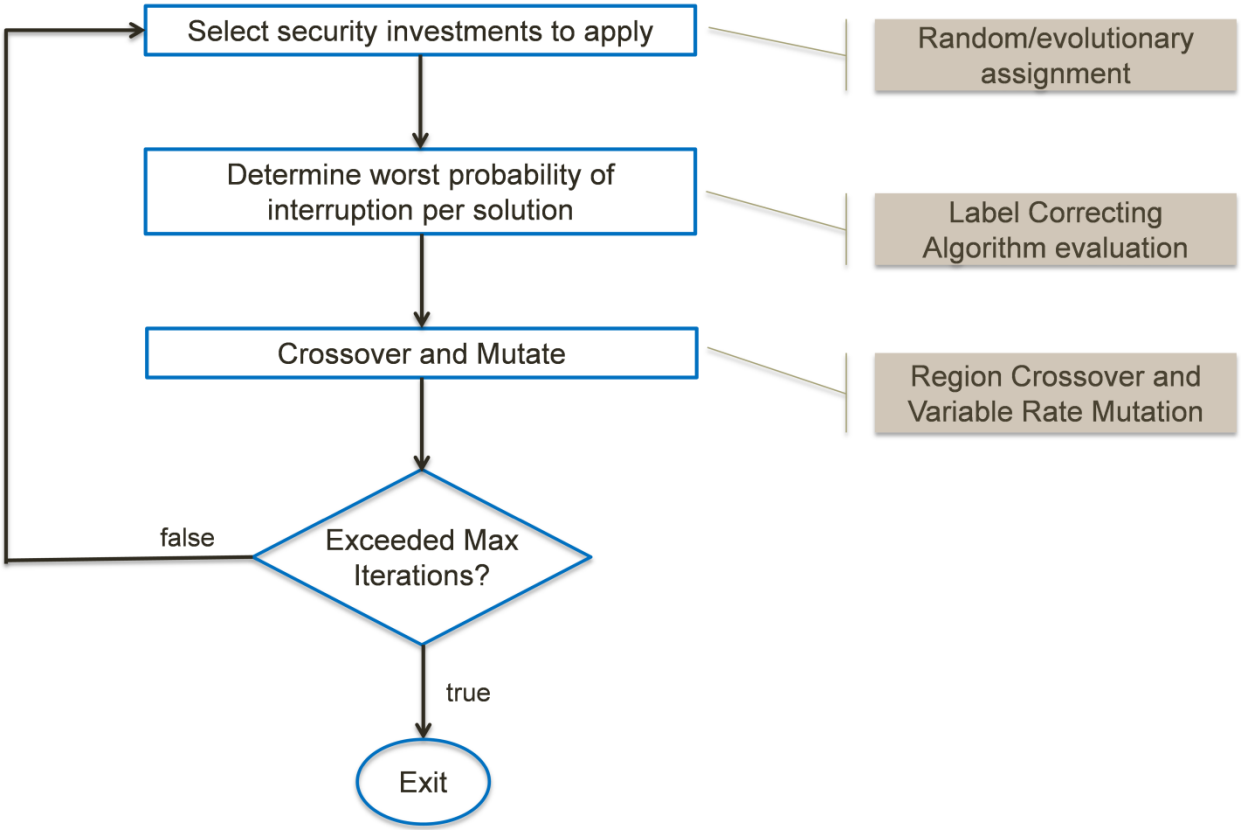

Figure 4: Defender investment optimization

The region crossover strategy referenced in Step 3 above is based on the idea that it is important to find synergies between investments on the same link and on neighboring links. This strategy makes use of the concept of a two-dimensional encoding strategy as described in Cohoon and Pairs (1987). The procedure is as follows:

1. From one to four random two-dimensional regions within the network are selected, each ranging in size from 5% to 25% of the total network area, but not exceeding 85% of the total network area in aggregate.
2. Investments from each parent strategy are collected based on their presence within the random regions.
3. Each parent has a primary child which receives all of the parent's investments outside of the region but none of the investments within the region.
4. The investments within the region are then assigned such that each child receives those from its non-primary parent.

Figure 5 gives a simple example of the region crossover strategy. In this scenario, there are only two types of investments that can be made at a location: a fence (F) or security camera (SC). The top half of the figure illustrates the investments at their physical locations for each of the parent solutions, P1 and P2. The shaded areas indicate the regions to be swapped between the parent solutions. C1 and C2 are the child investment strategies resulting from the crossover procedure. C1 is the primary child of P1 and receives all of P1's investments *outside* the region combined with all of P2's investments *within* the region. Similarly, C2 is the primary child of P2

and receives all of P2's investments outside the region and all of P1's investments within the region.

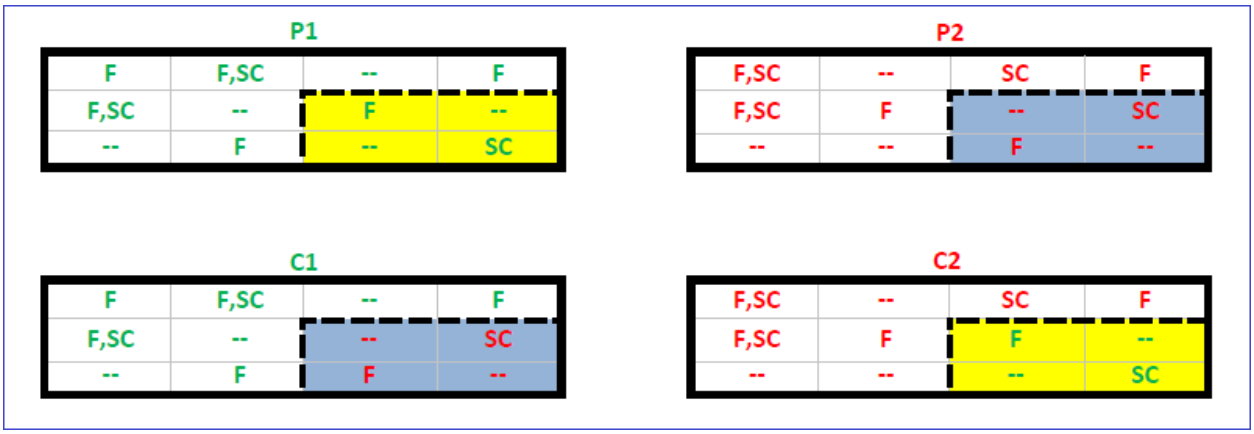

Figure 5: Region crossover

A key part of this procedure is determining which solutions to retain after each crossover iteration based on their "fitness" to the overall goal. An effective strategy is to construct a Pareto frontier with the initial population, which is updated during each crossover iteration, and to use the distance from the frontier as the fitness value for each solution. A common distance metric is the Euclidean distance between the candidate solution and the closest solution on the frontier. For a three-dimensional objective, this calculation requires three subtraction operations, three multiplication operations, two additions, and a square root. Since this calculation can become somewhat expensive as the sizes of both the frontier and candidate solutions grow, the algorithm is modified to instead use the taxicab (a.k.a. Manhattan) distance, as described by Krause (1987), which requires only three subtraction operations per solution pair. By applying an appropriate normalization factor to each objective after each crossover iteration, it is possible to calculate the relative fitness values for each collection of solutions such that they mirror the results produced when using Euclidean distance. The objective normalization factors are determined by calculating the average solution, which is composed of the average objective values across the frontier, and by setting the objective weights such that each objective contributes equally to the average solution.

In addition to homogeneity, another side effect of the crossover procedure is that there tends to be a large number of investments in each child solution. Since many of the investments do not improve the probability of interruption, P_{int}, it is useful to "clean" the solutions on the frontier by executing an iterative local search which examines each investment and removes it if P_{int} is not impacted. This procedure has the additional benefit of reducing the investment cost and (typically) reducing NAR/FAR as well. To increase the diversity of the solutions on the frontier, an additional procedure is performed on each solution which randomly decimates the remaining investments to produce solutions that have a lower, but still non-zero, P_{int}. The final output of the overall optimization is a Pareto frontier which holds the final collection of solutions and avoids arbitrary weighting of the objectives to determine an aggregate value. This technique allows a decision maker to select an investment strategy which satisfies their needs based on which objective(s) are most important to them.

ILLUSTRATIVE EXAMPLE

The following example uses notional data to demonstrate the type of analysis that can be performed using the tool. In order to allow analysts to interact with the model, a user interface was developed in Java and Python. The software allows users to create analysis scenarios, populate the input data, solve the model, and view the results. The notional example was created and analyzed through that interface.

This example assumes that an entirely new system is to be designed, rather than improving upon an existing design. There is a target (some high value asset) that the intruder wants to acquire and escape with. There are two buildings on site, but the target is not in either of the two buildings. For the purposes of this analysis, the buildings are treated as barriers that the intruder cannot pass through and the system owner can't place security measures in or on. In this case, there is a single intruder, and the force response time is 45 seconds.

The security investments under consideration for this example include radar (R), fence (F), buried cable (BC), magnetic (Mag) sensor, microwave (Mic) sensor, and security camera (SC). The fence, buried cable, microwave sensor, and magnetic sensor investments can be applied on a per link basis. If radar is installed, it will cover a four link by four link (200 ft by 200 ft) area. Security cameras cover a two link by two link (100 ft by 100 ft) area directed away from the investment node location, to the north, northeast, east, or southeast of the node.

Table 1 shows the cost, NAR/FAR, probability of detection, and delay time expected for each of the investments. Only the fence is considered to be a barrier that can increase delay time on affected links. The other investments are sensors and only impact the probability of detection on the link. For the purpose of this analysis, we allow any combination of investment types to appear at the same location.

Investment	10-Year Cost (Thousands)	NAR/FAR	Probability of Detection on Affected Links	Delay (Seconds) on Affected Links
Radar (R)	500	2	0.75	-
Fence (F)	3	-	-	30
Buried Cable (BC)	200	4	0.9	-
Magnetic (Mag) Sensor	20	2	0.8	-
Microwave (Mic) Sensor	30	2	0.9	-
Security Camera (SC)	90	2	0.8	-

Table 1: Investment alternatives under consideration (notional data)

The model identifies a Pareto frontier of solutions. This allows the system owner to determine which solution balances system performance, NAR/FAR, and cost in a way that best meets their needs. The solutions for this example are presented in Table 2.

Solution	10-Year Cost (Thousands)	Probability of Interruption	NAR/FAR
A	$1,256	1	210
B	$864	0.9998	144
C	$516	0.9990	72
D	$234	0.9951	24
E	$144	0.9900	8
F	$138	0.9000	8
G	$122	0.8000	8
H	$114	0.9600	16
I	$102	0.8000	16

Figure 2: Solutions on the Pareto Frontier

The system owner has the ability to view the detailed solution represented by each of the points on the Pareto frontier and to see where each investment was placed for that solution (shown in Figure 7 as a dark blue link label, with fences drawn in light blue). The solution grid also shows the entry and exit path with the lowest probability of interruption, highlighted in red. Figure 7 shows the highest cost/highest P_{int} solution on the frontier.

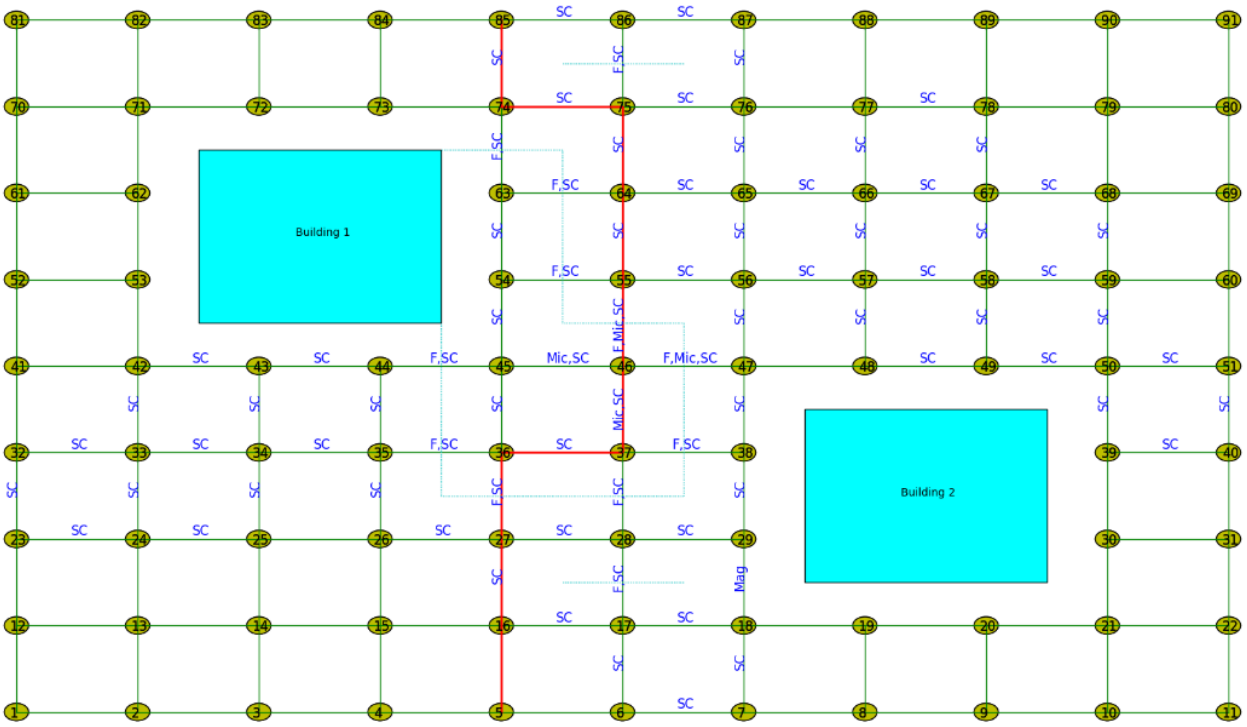

Figure 7: Highest cost and highest Probability of Interruption (P_{int})

Once these solutions have been provided, the system owner must decide what level of risk they are willing to accept, how much money they are willing to spend, and how much they are concerned with NAR/FAR versus cost and P_{int}. For example, solutions B and E in Table 2 both provide a high level of protection, at 0.9998 and 0.9900 probability of interruption respectively. However, solution B is six times more expensive than solution E and has a significantly higher NAR/FAR. Policy and budget will affect the system owner's decision. If they are required to have a P_{int} of at least 0.995, they would choose solution B. If the P_{int} provided by solution E is sufficient, they would likely choose that solution in order to save money and to avoid the increased nuisance and false alarms anticipated with solution B.

CONCLUSIONS

There have been several historical incidents that highlight a need for more effective security system design and operation to protect national security assets. There are many dimensions in the design space of a security system, including technology selection, alternative configurations, diverse threats, and budget considerations, making it effectively impossible to evaluate all permutations of potential system architectures.

We have developed a game theoretic model to optimize the design of security systems which explicitly includes opportunities to layer security barriers and the performance of a range of different technologies based on how they are to be deployed. The model also includes the ability to consider budget limitations and the impact of false alarms on system performance. We demonstrated this model on a realistic but notional problem instance.

There are at least three extensions to this model that would be useful. First, the performance of many security technologies varies based on weather conditions. Simply using average values for the detection probability, the time required to overcome each barrier, and the NAR/FAR

could be misleading. For this reason, some sensors are actually complementary rather than substitutes (as they might appear in this analysis). One method to include this in the model is to create a stochastic program for which the scenarios represent different weather and lighting conditions. With this type of approach, it may be important to maximize the minimum effectiveness of the system, where the effectiveness is measured across the different weather and lighting conditions. Second, different intruders have different capabilities. It is likely useful to include some representation of the capability of the intruders, both in skills as well as tools they might be able to carry, and therefore some explicit consideration of how they might use those items to decrease the effectiveness of the security system. Finally, this model focuses on a single group of intruders on the same path. There are likely to be scenarios for attacks that involve subsets of individuals synergistically working together to achieve their goal, and therefore intrusions that each require multiple paths.

ACKNOWLEDGMENTS

The authors would like to thank Mark Snell of Sandia National Laboratories.

ABBREVIATIONS & ACRONYMS

DoD	Department of Defense
DOE	Department of Energy
NNSA	National Nuclear Security Agency

REFERENCE LIST

Alderson, D., L., Brown, G.,G., Carlyle, M., and Wood, K., "Solving Defender-Attacker-Defender Models for Infrastructure Defense", *Proceedings of the 12th INFORMS Computing Society Conference*, p. 28-49, 2011.

Arroyo, J.M., and Fernández, F.J., "A Genetic Algorithm Approach for the Analysis of Electric Grid Interdiction and Line Switching", *15th International Conference on Intelligent System Applications to Power Systems*, December 2009.

Brown, N., J. Gearhart, D. Jones, L. Nozick, N. Romero and N. Xu, 2013, "Multi-Objective Optimization for Bridge Retrofit to Address Earthquake Hazards". *Proceedings of the 2013 Winter Simulation Conference*. Edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl.

Chen., F., Wang, L., and Jinshu., S., "An Efficient Approach to Minimum-Cost Network Hardening Using Attack Graphs", *Fourth International Conference on Information Assurance and Security*, 2008, 209-212.

Chen, F., Su, J., and Yi, Z., "A Scalable Approach to Full Attack Graph Generation", *Lecture Notes in Computer Science: Proceedings of the First International Symposium on Engineering Secure Software and Systems*, 5429, 150-163, 2009.

Cohoon, J.P. and Pairs, W.D., "Genetic Placement." *IEEE Transactions on Computer Aided Design*, 6, p. 956-964, 1987.

Deb, K. 2012. "Advances in Evolutionary Multi-objective Optimization." *SSBSE 2012*. Edited by G. Fraser. Springer-Verlag, Berlin, Heidelberg.

Garcia, M.L. 2007. *The Design and Evaluation of Physical Protection Systems.* Butterworth-Heinemann. ISBN: 978-0750683524.

Konak, A., D. W. Coit, and A. E. Smith. 2006. "Multi-objective optimization using genetic algorithms: A tutorial." *Reliability Engineering and System Safety*, available online as of January 2006.

Jones, D.A., Davis, C.E., Turquist, M.A., and Nozick, L.K., "Physical Security and Vulnerability Modeling for Infrastructure Facilities", *Proceedings of the 38th Hawaii International Conference on Systems Sciences*, 4, 1-9, 2006.

Krause, Eugene F. 1987. *Taxicab Geometry*. Dover. ISBN 0-486-25202-7.

Murray-Tuite, P., and Fei, X., "A Methodology for Assessing Transportation Network Terrorism Risk with Attacker Defender Interactions", *Computer-Aided Civil Infrastructure Engineering*, 25, p. 396-410, 2010.

Ou., X., Boyer, W., and McQueen, M., "A Scalable Approach to Attack Graph Generation", *Proceedings of the 13th Annual ACM Conference on Computer and Communication Security*, 336-345, 2006.

Philips, C.A., and Swiler, L.P., "A Graph-Based System for Network Vulnerability Analysis", *Proceedings of the 1998 New Security Paradigms Workshop, Association for Computing Machinery*, p. 71-81, 1998.

Reilly, A., Nozick, L., Xu, N., and Jones, D., "Game Theory-based Identification of Facility Use Restrictions for the Movement of Hazardous Materials Under Terrorist Threat," *Transportation Research Part E: Logistics and Transportation Review*, 48(1), p. 115-131, 2012.

Romero, N., Xu, Ningxiong, X, Dobon, I., and Jones, D., "Investment Planning for Electric Power Systems Under Terrorist Threat," *IEEE Transactions on Power Systems*, 27(1), p. 108-116, 2012.

Sait, S. M. and H. Youssef. 1999. *Iterative Computer Algorithms with Applications in Engineering: Solving Combinatorial Optimization Problems*. Los Alamitos, CA: IEEE Computer Society.

Sameron, J., Wood, K., and Baldick, "Worst-Case Interdiction of Large-Scale Electric Power Grids," IEEE Transactions in Power Systems, 24(1), p. 96-104, 2009.

Sheyner, O., Haines, J., Jha, S., Lippmann, R., and Wing, J.M., "Automated Generation and Analysis of Attack Graphs," Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy, 273-284, 2002.

U.S. Department of Energy, Office of Inspector General, Office of Audits and Inspections. 2012. *Inquiry into the Security Breach at the National Nuclear Security Administration's Y-12 National Security Complex.*

DESCRIPTORS
optimization, genetic algorithm, security systems, attacker-defender, investment, intruder, physical security