Optimisation of security system design by quantitative risk assessment and genetic algorithms

Francesco Flammini*

Innovation & Competitiveness Unit
ANSALDO STS Italy, Via Argine 425, Napoli, Italy
Email: francesco.flammini@ansaldo-sts.com
*Corresponding author

A. Gaglione and Nicola Mazzocca

Department of Computer and Systems Engineering
University of Naples Federico II
Via Claudio 21, Napoli, Italy
Email: andrea.gaglione@unina.it

Concetta Pragliola

Innovation & Competitiveness Unit
ANSALDO STS Italy, Via Argine 425, Napoli, Italy
Email: concetta.pragliola@ansaldo-sts.com

Abstract. The design of physical security systems for critical infrastructures is a delicate task that requires a balance between the cost of protection mechanisms and their expected effect on risk mitigation. This paper presents an approach usable to support the design of security systems by automatically optimizing some parameters, basing on external constraints (e.g. limited available budget) and using quantitative risk assessment. Risk assessment is performed using a software tool that implements a quantitative methodology. The methodology accounts for the attributes of threats (frequency, system vulnerability, expected consequences) and protection mechanisms (cost, effectiveness, coverage, etc.). The optimization is performed by means of genetic algorithms with the objective of achieving the set of parameters that minimizes the risk while fitting external budget constraints, hence maximizing the return on investment. The paper also describes an example application of the approach to the design of physical security systems for metro railways.

Keywords: infrastructure security, decision support systems, risk assessment, return on investment, rail-based mass transit systems.

Reference to this paper should be made as follows: Flammini, F., Gaglione, A., Mazzocca N. and Pragliola C. (2011) 'Optimisation of security system design through quantitative risk assessment and genetic algorithms', *Int. J. Risk Assessment and Management*, Vol. 15, Nos. 2/3, pp.205-221.

Biographical notes: Francesco Flammini got with honors his laurea (July 2003) and doctorate (December 2006) degrees in Computer Engineering from the University Federico II of Naples. From October 2003 to January 2007, he has worked in Ansaldo STS as a Software Engineer in the RAMS unit on the verification and validation of real-time control systems. Since February 2007, he has worked in the Innovation unit on the protection of transportation infrastructures. He has been an Adjunct Professor of Software Engineering and Computer Science and currently serves as the Editor in Chief for the International Journal of Critical Computer-Based Systems.

Andrea Gaglione received his B.S. degree and M.S. degree in Computer Engineering, both summa cum laude, from the Second University of Naples in 2004 and 2006, respectively. He got a Ph.D. in Computer and Control Engineering from the University of Naples Federico II in 2009 and his research activities include Sensor Networks, Event Recognition, and Critical Infrastructure Protection.

Nicola Mazzocca is a full professor of High-Performance and Reliable Computing at the Computer and System Engineering Department of the University of Naples Federico II, Italy. He owns an MS Degree in Electronic Engineering and a Ph.D. in Computer Engineering, both from the University of Naples Federico II. His research activities include methodologies and tools for design/analysis of distributed systems; secure and real-time systems and dedicated parallel architectures.

Concetta Pragliola got her laurea and doctorate degrees in Electronic Engineering from the University Federico II of Naples in October 1985. From January 1987 to October 1992 she has worked in the Research Department of Ansaldo Transporti on Expert Systems and Simulation programs. From November 1992 to October 2001, she has worked in the Information Technology Department of Ansaldo Trasporti, being involved in PDM systems. From November 2001 to November 2006 she has worked in Elsag as an Account Manager. Since December 2006 she has worked in the Innovation unit of Ansaldo STS specializing on the design of security systems.

1. Introduction

In the design of physical security systems for critical infrastructure protection, it is very important to demonstrate to the customer both the expected return on investment and the optimality of the proposed system design. The formal evidence of this is impossible to achieve without adopting quantitative risk assessment approaches. In particular, the operators of public transportation systems are very interested not only in being compliant to the security norms but also in mitigating the effect of criminality. In fact, criminal acts can damage the infrastructures and also have a negative impact on the usage rate of the system by the citizens. Since different protection mechanisms can be employed, each of them usually capable to face several types of threats, ranging from vandalism to terrorism, it is essential to quantify their effect on risk as accurately as possible. This is useful to fine tune design parameters in order to maximize the return on investment, which is especially important when the available budget is limited (which is generally true).

Flammini et al. (2008) have described a method and a tool (i.e. Q-RA) to automatically compute the expected annual benefit of a security system starting from quantitative attributes of threats and protection mechanisms. Among the attributes of protection mechanisms the authors have introduced the concept of "coverage" (*COV*), which can be used as an indication of the size of the protection mechanism with respect to the infrastructure to be protected. The *COV* parameter is the only free parameter of the optimization problem, since it is assumed that deterrent, protective and rationalizing effectiveness cannot be easily modified for any specific protection mechanism. The genetic algorithm presented in this paper is able to detect the optimal set of *COV* parameters for the protection mechanisms specified in the Q-RA input database. The results can be evaluated to guide the design of the security system.

Genetic algorithms (GA) are a biologically inspired heuristic search method (Whitley, 1994). As their name suggests, GA mimic the evolution of living beings, according to the Darwinian theories. Solutions of a problem are thought of as individuals who reproduce according to the natural processes of selection, crossover and mutation. GA have been demonstrated to be a very general method capable to find approximate solutions for complex and non linear engineering problems where no specific algorithms exist, as in the case of the optimization problem (and its possible extensions) described in this paper.

Several research papers are available dealing with automated risk assessment (Bang et al., 2004) or with the application of GAs to other classes of optimization problems, such as system reliability (Painton and Campbell, 2005). Lot of efforts have been devoted to physical protection systems design (Garcia, 2001), by also exploiting wireless sensor technology (Flammini et al., 2009). Some works have been devoted to the use of GAs for improving network security (Banković et al., 2007; Yao, 2010) and supporting sensor placement, sensor activation and network clustering for the optimization of sensor network design (Indu et al., 2009; Ferentinos and Tsiligiridis, 2007; Martins et al., 2010); however, to the best of our knowledge no prior work addresses the use of GA to support the choice of protection mechanisms in the design of physical security systems. Instead, some applications have been recently reported in the field of on-line risk evaluation (Abraham et al., 2009).

The rest of this paper is organized as follows. Section 2 presents the method used for the analysis, that is the mathematical model used for risk assessment and mitigation. Section 3 describes the Q-RA toolkit and one example application. Section 4 defines the objective of the optimization, the genetic algorithm we have developed and the results of an example application. Finally, Section 5 draws conclusions and provides some hints about future developments.

2. The Q-RA methodology

2.1 Risk assessment model

With reference to a specific threat, the quantitative risk *R* can be formally defined as follows:

$$R[€ / \mathrm{year}] = P \cdot V \cdot D \qquad (1)$$

Where:

- *P* is the frequency of occurrence of the threat, which can be measured in [events / year];
- *V* is the vulnerability of the system with respect to the threat, that is to say the probability that the threat will cause the expected consequences (damage);
- *D* is an estimate of the measure of the expected damage occurring after a successful attack, which can be expressed in currency, e.g. *Euros* [€].

The vulnerability *V* is a dimensionless parameter, since it represents the conditional probability:

$$P(success \mid threat) \qquad (2)$$

Therefore, a quantitative way to express the risk associated with a specific threat is to measure it in lost *Euros* per year: [€ / year]. The overall risk can be obtained as the sum of the risks associated with all threats, assuming all the contributions being independent (worst case):

$$R_T[\text{€} / \text{year}] = \sum_i P_i \cdot V_i \cdot D_i \qquad \forall \text{ threat } i \qquad (3)$$

Despite of the simplicity of (3), the involved parameters are not easy to obtain. The analysis involves both procedural and modeling aspects. Procedural aspects include brainstorming sessions, site surveys, design review, statistic data analysis, expert judgment, etc. Formal modeling languages which can be used to analytically compute P, V and D include Attack Trees, Bayesian Networks, Stochastic Petri Nets and possibly other formalisms which are able to take into account the uncertainty inherently associated with the risk as well as the possibility of strategic attacks (Nicol et al., 2004). In fact, sometimes the three parameters might feature an inter-dependence which should be taken into account by the model, too.

2.2 Risk mitigation model

Protection mechanisms are able to reduce the risk by having three main effects:

- Protective, aimed at the reduction of V
- Deterrent, aimed at the reduction of P
- Rationalizing, aimed at the reduction of D

In particular, the rationalizing effect allows a better management of the response force and/or an improved coordination of security personnel for counteracting the threats (e.g. by *Closed Circuit TeleVision*, CCTV).

Therefore, by quantifying the listed effects it is possible to estimate the risk mitigation, considering any combinations of threats and protection mechanisms.

A possible way to compute risk mitigation is to associate threats and protection mechanisms by means of threat categories and geographical references, namely *sites*. A site can be considered as a particular kind of critical asset (actually, an aggregate asset), sometimes defined as "risk entity". Each threat happens in at least one site and, homogeneously, each protection mechanism protects at least one site. For a railway infrastructure, a site can be an office, a bridge, a tunnel, a parking area, a platform, a control room, etc.

In the assumption that:

- Threat T belongs to category C;
- Threat T happens in (or passes through) site S;

- Protection *M* is installed in site *S*;
- Protection *M* is effective on threat category *C*;

then it can be stated that *M* protects against *T*. Threat categories should be defined according to the specific application domain and used to classify threats upon the basis of relevant common factors (e.g. damaging the furniture of a station might fall in a threat category named *vandalism*).

Based on the above definitions, it is possible to express the total mitigated risk, which the system is exposed to as follows:

$$R_{TM}[€ / \text{year}] = \sum_i R_i \cdot \prod_j MIT_{ji} \qquad \textbf{(4)}$$

Where:

- R_{TM} is the total mitigated risk;
- R_i is the initial risk associated with threat *i* (computed according to (1));
- MIT_{ji} is the mitigation effect provided by protection mechanism *j* on threat *i*, taking into account its coverage and effectiveness.

Hence, the total mitigated risk is expressed as follows:

$$R_{TM}[€ / \text{year}] = \sum_i R_i \cdot \prod_j (1 - E_{Pji} \cdot COV_j) \cdot (1 - E_{Dji} \cdot COV_j) \cdot (1 - E_{Rji} \cdot COV_j) \qquad \textbf{(5)}$$

Where:

- E_{Pji} is an estimate of the protective effect of protection mechanism *j* on threat *i*;
- E_{Dji} is an estimate of the deterrent effect of protection mechanism *j* on threat *i*;
- E_{Rji} is an estimate of the rationalizing effect of protection mechanism *j* on threat *i*;
- COV_j is a measure of the coverage of protection mechanism *j* (e.g. percentage of the physical area or perimeter of the site).

The values of parameters expressing coverage and effectiveness are in the range [0..1]. E_{Pji}, E_{Dji} and E_{Rji} can be obtained by historical datasets (e.g. average reduction of vandal attacks after CCTV installation in similar infrastructures), appropriate models and/or by the judgment of domain experts. The formula can be validated by attempts using sample data and boundary analysis: for instance, when both the coverage and one of the effectiveness parameters are set to 1, the risk is mitigated to 0, as expected; on the opposite, if either the coverage or all the effectiveness parameters are set to 0, the risk is not mitigated at all. Figure 1 reports an example risk evaluation based on (5) using sample data. In such evaluation it is assumed that a single protection mechanism is used and all the other data is kept constant.

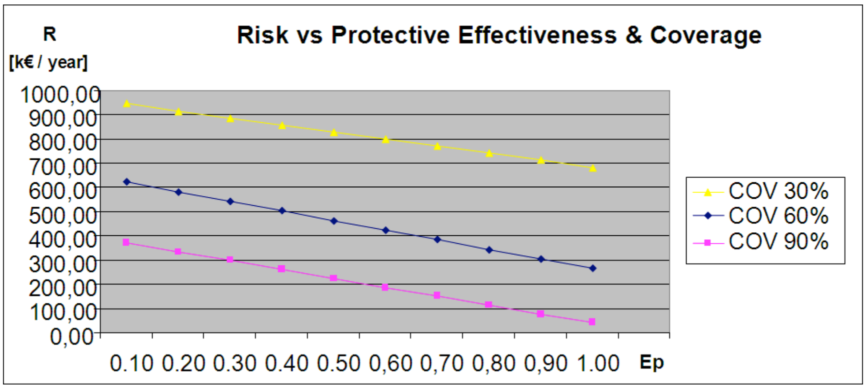


Figure 1. Risk evaluation using sample data.

The expected benefit can be defined simply as the balance between the annual investment on security (protection mechanisms) and the achieved risk mitigation:

(6)

$$EB[€/\text{year}] = \text{risk reduction} - \text{total investment in security} = (\mathrm{R}_\mathrm{T} - \sum_i R_i) - \sum_\mathrm{j} \mathrm{COST}_\mathrm{j}$$

Where:

- *EB* is the Expected Benefit, which can be positive or negative;
- COST_j is the cost of the protection mechanism *j*, obtained considering all the significant costs (acquisition, installation, management, maintenance, etc.).

Therefore, the return on investment can be obtained from the expected benefit EB considering the cost of the invested capital (which depends on the rate of interest, the years to pay-off, possible external funding, etc.).

Expressions (5) and (6) need to be computed starting from a database of attack scenarios, sites, protection mechanisms and related significant attributes. The management of such data and the computation of results are performed by an automatic tool which will be described in detail in next section.

3. The Q-RA toolkit

3.1 Description

A tool has been developed which automatically manages risk data and evaluates risk and benefit indices starting from input data. The tool has been named simply Q-RA (Quantitative Risk Analysis), to be pronounced as [kura] (sounding like the Italian for “cure”).

In particular, the inputs of the tool are:

- A list of threats, characterized by:
  - Threat identifier;
  - Short description of the attack scenario (including the adversary category, required tools, etc.);
  - Threat category (e.g. vandalism, theft, sabotage, terrorism, flooding, etc.);
  - Initial estimated *P*, *V* and *D;*
  - Site (geographical reference).
- A list of protection mechanisms, characterized by:
  - Protection mechanism identifier;
  - Short description of the protection mechanism;
  - List of threat categories on which the protection mechanism is effective;
  - Expected protective (E_{Pji}), deterrent (E_{Dji}) and rationalizing (E_{Rji}) effectiveness;
  - Estimated coverage (*COV*);
  - Site (geographical reference);
  - Annual cost (acquisition, management, maintenance, etc.).

A database is used in order to store and correlate the input data. Data referring to economic aspects is also managed (number of years to dismiss, rate of interest, etc.). The tool provides features allowing the user for inserting the inputs, updating them to modify some parameters (i.e. frequency of threats) and finally removing them.

Parameters can be chosen using average or worst case considerations. Sensitivity analysis can be performed acting on input data ranges in order to evaluate the effect of uncertainty intervals upon the computed results and possibly defining lower and upper bounds.

The tool elaborates data according to the relationships defined in the database (in particular, using the common attributes of site and threat category) and the mathematical models of (5) and (6), providing:

- The risk associated with each threat (R_i) and the overall risk (R_T);
- The total risk reduction considering all the threats;
- Annual cost of the single protection mechanism and of the whole security system;
- Expected Benefit (*EB*).

The points listed above are part of the informal functional requirements specification. Application specific requirements have also been added, like the possibility of specifying a day/night attribute for both threats (some scenarios cannot happen when the service is interrupted, e.g. a subway station is closed to the public) and protection mechanisms (some technologies, e.g. motion detection, can be activated only when the service is interrupted). Non functional requirements of the tool include user friendliness, data import / export facilities using standard formats (e.g. CSV, Comma Separated Values), platform independence and use of freeware software (possibly), user identification and rights management (still to be implemented).

Some implementation details are reported in the following. The software design has been performed using an object-oriented approach based on the Unified Modeling Language (UML) and the Java programming language. In order to guarantee the persistence of objects (threats, protection mechanisms and sites), a relational database (based on MySQL) has been designed starting from Entity Relationship (E-R) diagrams. The GUI (Graphical User Interface) of the tool is web-based, exploiting JSP (Java Server Page) and Apache Tomcat technologies.

As an example, the conceptual class diagram related to the specific domain is reported in Figure 2, where the attributes and interrelationships of the entities described in the previous section are graphically shown.

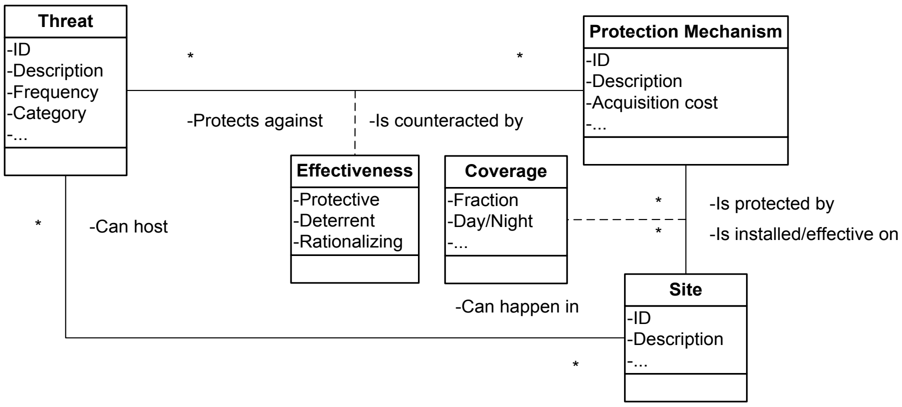


Figure 2. Conceptual class diagram.

3.2. Example application

Let us consider a case-study of a railway or subway station. The following threats against the infrastructure should be considered:

- Damage to property and graffitism (vandalism);
- Theft and aggressions to personnel and passengers (micro-criminality)
- Tampering and forced service interruption (sabotage)
- Bombing or spread of NBCR[1] contaminators (terrorism)

[1] Nuclear Biological Chemical Radiological.

Let us consider the example scenarios reported in Table 1 and the protection mechanisms listed in Table 2, both referring to a specific station. It is assumed that the values are obtained by analyzing historical data of successful and unsuccessful attacks before and after adopting specific countermeasures (such data is usually available for comparable installations). The expected damage relates to the single attack and it is computed by predicting the:

- Expense needed to restore the assets
- Consequences of service interruptions or degradations
- Decreased usage of the transportation system by the passengers due to the feeling of insecurity
- Human injuries or loss of lives

The estimated annual cost of the protection mechanisms also accounts for maintenance and supervision, while acquisition and installation costs are accounted separately. Please note that the effect of protection mechanisms may vary according to threat category. Furthermore, all the specified values should not be considered as the results of specific analyses, but only as realistic pseudo-data; using real data would require an extensive justification, possibly via a model-based evaluation, which is not in the scope of this paper.

Figure 3 reports a screenshot of the GUI representing the input mask for the attributes of protection mechanisms, while Figure 4 reports the results of the example application computed by the tool. In the assumptions of the example, the positive expected benefit resulting from the adoption of the protection mechanisms and being 36722 €/year clearly justifies the investment.

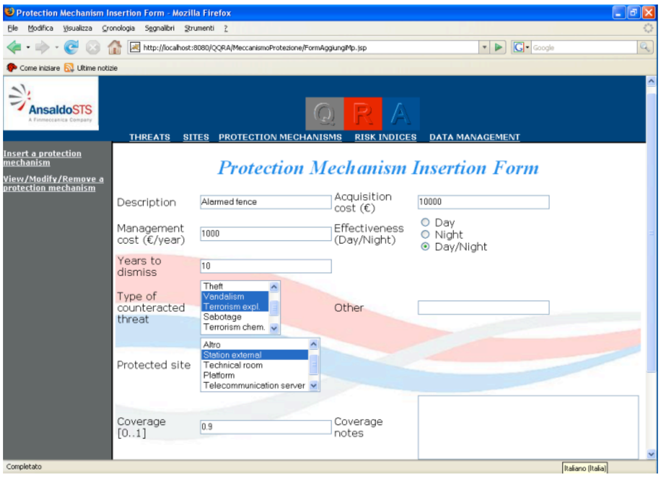
Table 1. Attack scenarios considered in the example application.

Threat ID	**Threat Description**	**Threat Category**	**Site**	**Est. P [# / Year]**	**Est. V_{INIT}**	**Exp. D [K€]**
1	Graffitism	Vandalism	Station Ext.	60	0.9	0.5
2	Theft of PCs	Theft	Tech. Room	4	0.8	14
3	Glass Break	Vandalism	Station Ext.	12	1	0.5
4	Bombing	Terrorism Expl.	Platform	0.01	1	900
5	Hacking	Sabotage	TLC Server	2	0.8	10
6	Gas Attack	Terrorism Chem.	Platform	0.01	1	160
7	Furniture Damage	Vandalism	Hall Platform	70 50	1 1	0.1 0.1
8	Infrastruct. Damage	Physical Sabotage	Platform	4	0.9	5

Table 2. Protection mechanisms considered in the example application.

PROT. ID	**COUNTERMEASURE DESCRIPTION**	**ACQ. COST [K€]**	**MANAG. COST [K€ / YEAR]**	**SITE**	**COV**	**THREAT CATEGORIES**	**E_P**	**E_D**	**E_R**
1	ALARMED FENCE	10	1	STATION EXT. STATION INT. (NIGHT)	0.9	VANDALISM THEFT P. SABOTAGE	0.9 0.9 0.9	0.3 0.3 0.3	0.2 0.2 0.2
2	VOLUMETRIC DETECTOR	5	1	TECH. ROOM	1	THEFT	0.8	0.6	0.2
3	VIDEO-SURVEILLANCE (INTERNAL)	150	20	HALL, PLATFORM	0.95	VANDALISM THEFT SABOTAGE TERRORISM EXPL. TERRORISM CHEM.	0.4 0.6 0.6 0.4 0.4	0.6 0.6 0.6 0.3 0.3	0.3 0.3 0.8 0.6 0.6
4	CHEM. DETECTOR	50	2	PLATFORM	0.9	TERRORISM CHEM.	0.6	0.2	0.4
5	INTRUSION DETECTION SYSTEM	1	0.5	TLC SERVER	1	L. SABOTAGE	0.9	0	0
6	EXPLOSIVE DETECTOR	50	2	STATION INT. (*)	1	SABOTAGE TERRORISM EXPL.	0.8 0.8	0.4 0.1	0.1 0.1

(*): detectors are physically installed near turnstiles, but the protection is effective on the whole station internal.


Protection Mechanism Insertion Form - Mozilla Firefox
File Modifica Visualizza Cronologia Segnalibri Strumenti ?
http://localhost:8080/QQRA/MeccanismoProtezione/FormAggiungiMp.jsp
Google
Come iniziare Ultime notizie
AnsaldoSTS
A Finmeccanica Company
Q R A
THREATS SITES PROTECTION MECHANISMS RISK INDICES DATA MANAGEMENT
Insert a protection mechanism
View/Modify/Remove a protection mechanism
Completato
Italiano (Italia)

Protection Mechanism Insertion Form

Description: Alarmed fence

Acquisition cost (€): 10000

Management cost (€/year): 1000

Effectiveness (Day/Night): ○ Day ○ Night ● Day/Night

Years to dismiss: 10

Type of counteracted threat: Theft, Vandalism, Terrorism expl., Sabotage, Terrorism chem.

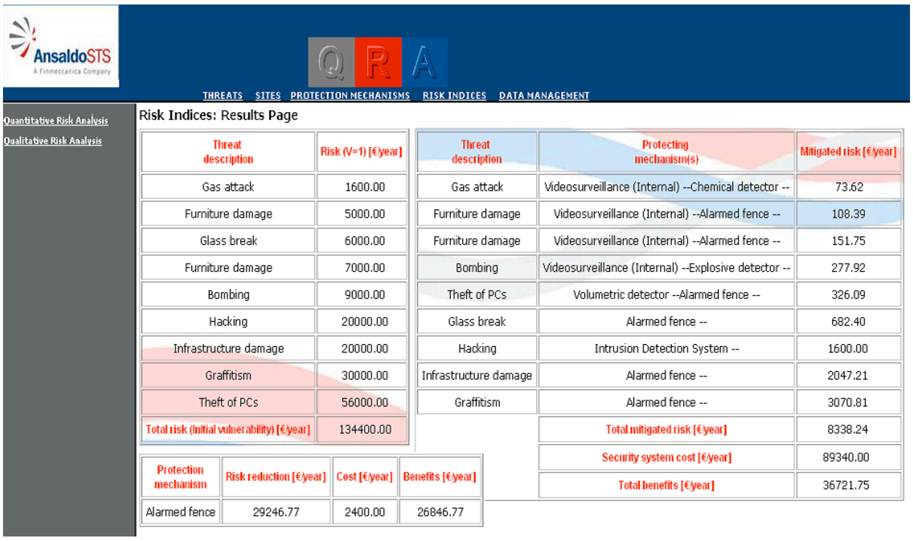
Other:

Protected site: Altro, Station external, Technical room, Platform, Telecommunication server

Coverage [0..1]: 0.9

Coverage notes:

Figure 3. The Q-RA input data mask for protection mechanisms.



Risk Indices: Results Page

Threat description	Risk (V=1) [€/year]
Gas attack	1600.00
Furniture damage	5000.00
Glass break	6000.00
Furniture damage	7000.00
Bombing	9000.00
Hacking	20000.00
Infrastructure damage	20000.00
Graffitism	30000.00
Theft of PCs	56000.00
Total risk (Initial vulnerability) [€/year]	134400.00

Threat description	Protecting mechanism(s)	Mitigated risk [€/year]
Gas attack	Videosurveillance (Internal) --Chemical detector --	73.62
Furniture damage	Videosurveillance (Internal) --Alarmed fence --	108.39
Furniture damage	Videosurveillance (Internal) --Alarmed fence --	151.75
Bombing	Videosurveillance (Internal) --Explosive detector --	277.92
Theft of PCs	Volumetric detector --Alarmed fence --	326.09
Glass break	Alarmed fence --	682.40
Hacking	Intrusion Detection System --	1600.00
Infrastructure damage	Alarmed fence --	2047.21
Graffitism	Alarmed fence --	3070.81
	Total mitigated risk [€/year]	8338.24
	Security system cost [€/year]	89340.00
	Total benefits [€/year]	36721.75

Protection mechanism	Risk reduction [€/year]	Cost [€/year]	Benefits [€/year]
Alarmed fence	29246.77	2400.00	26846.77

Figure 4. Q-RA output data presentation for the example application.

4. Optimization approach

4.1 Definition of the optimization problem

The objective of the optimization problem is to maximize the expected benefit (*EB*) of the security system. *EB* is automatically computed by the Q-RA by means of expression (6).

The constraint of the optimization problem is to fulfil an externally set annual budget limit; that is, to verify that the sum of costs associated with the set of protection mechanisms is less than a specified amount (problem input).

We assume here that the only free parameters of the problems are the coverage levels of the protection mechanisms. This is not a restrictive assumption, since the method works with any number of parameters and its utility is even higher when the number of variables increases. Of course, a *COV* value set to 0 means that the related protection mechanism is excluded from the solution; in that case, it should not be selected.

Given *M* protection mechanisms and using a granularity of 1% in the variation of the coverage, the search space (*SS*) of the possible solutions to the problem has the following cardinality:

$$C_{SS} = 100^{M}$$

It could be shown that if *M* = 10 and the *EB* evaluation of a single solution by Q-RA requires just one second, then several geological eras would be needed to exhaustively explore the search space to find the optimal solution.

For non linear optimization problems featuring such a huge search space, Genetic Algorithms (GA) represent one of the viable methods to try finding solutions which at least approach the best one.

Figure 5 provides an at-a-glance representation of input and outputs of the optimization problem.

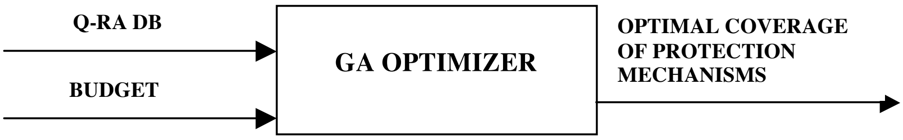


Figure 5. Input and output of the GA-based optimizer.

4.2 Description of the Genetic Algorithm

In a classical GA all the optimization parameters have to be coded into a binary system (Holland 1975, 1992). Typically, the Gray code is used where a small distance in the problem correspond to a small distance in the representation space as well. The number of bits used is usually problem dependent. All these bit strings are then combined to form a chromosome leading to a genotypic representation of the optimization parameters. In the following of this section we will explain how the general algorithm can be applied to our optimization problem.

The general structure of a canonical genetic algorithm with its pseudo-code is reported in Figure 6 (Alotto et al., 1998). The initialization procedure (line 1) creates a random population of individuals by associating a coverage percentage to any protection mechanisms obtaining a vector of M elements, the so-called "chromosomes". The population size N (usually a few hundred individuals) can be considered as fixed, since the convergence of the algorithm has little dependence on such a parameter. The process is repeated until the number of the individuals equals the specified population size.

The fitness function, applied to each individual of the population, is then calculated (line 5) by using the following formula:

$$F_i = \frac{EB_i}{\frac{1}{N}\sum_{j=1}^{N} EB_j}$$

Where:

- F_i is the computed fitness of individual i
- EB_j is the expected benefit associated with solution j

The suitability of each individual is also checked with respect to the problem specific constraint. If the cost of an individual exceeds the budget limitation that indicates an unfeasible solution, hence (instead of discarding the individual) a penalty term is added to its fitness value as follows:

$$F_i^{'} = F_i \cdot k \cdot \frac{BUDGET}{COST_i}$$

Where:

- $F_i^{'}$ is the new computed fitness of the individual *i*
- $COST_i$ is the cost of the individual *i*
- k is a constant in [0,1], which can be conveniently tuned

A selection process (line 6) is then applied in such a way that configurations with better fitness values are chosen for breeding with a probability higher than the other ones (the so called "roulette wheel sampling" approach has been adopted, see Goldberg, 1989).

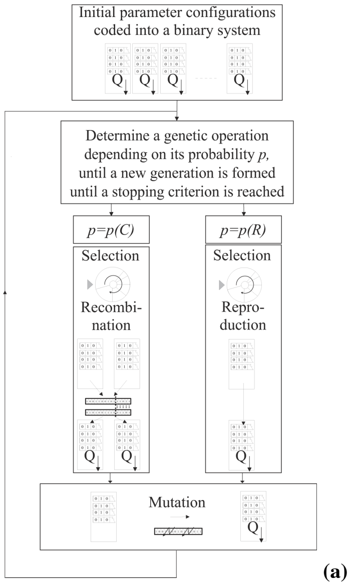
In the classical binary implementation, *recombination* and *reproduction* are performed in parallel (line 7) with a certain probability (*p(C)* and *p(R)*, respectively). Recombination is usually performed by some kind of crossover: in particular, we used the "one point crossover" (see Figure 7), according to which a point within the chromosomes of two parent configurations is selected randomly and all the bits following this position until the end of the bit-strings are mutually exchanged to produce two new configurations. Instead, reproduction allows to propagate one parent configuration to a descendant without any change. Finally, the new individuals undergo a bitwise mutation (line 9) according to which a few bits within the chromosomes are inverted with a very low probability *p(M)*. This helps avoiding some problems inherently associated with the genetic process (e.g. predominant individuals leading to premature convergence and local optima).

The process continues executing the same steps on the new generations until some stopping criterion is met (line 3). There are two possible stopping criteria for the algorithm:

1. the quadratic norm of the population with respect to the mean value of the population is less than a prescribed ε;

2. the maximum number of possible generations has been reached (that is, the algorithm has not converged, yet an usable solution could have been found).

Usually, the convergence of the algorithm is rather quick regardless of the specific values assigned to each parameter. The best individual of the last population can then be selected according to an appropriate criterion (e.g. minimum cost).


Initial parameter configurations coded into a binary system
Q
Q
Q
Q
Determine a genetic operation depending on its probability *p*, until a new generation is formed until a stopping criterion is reached
p=p(C)
p=p(R)
Selection
Recombi-nation
Selection
Repro-duction
Q
Q
Q
Mutation
Q
(a)

```
generation = 0
initialize population (N)
while not(stopping criteria)
  generation = generation + 1
  compute fitness for each
   individual
  selection
  recombination [p(C)] OR
  reproduction [p(R)]
  mutation [p(M)]
end while
```

(b)

Figure 6. The basic structure of the Genetic Algorithm (a) and its pseudo-code (b).

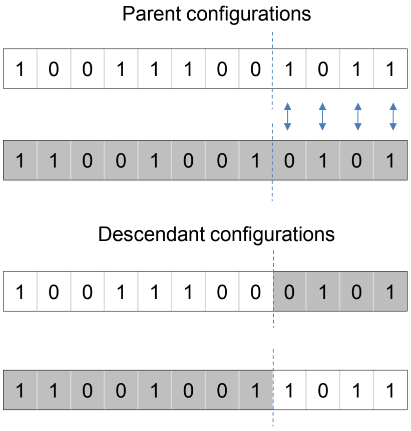


Figure 7. One point crossover.

4.3 Example experimental results

Let us consider the example scenarios reported in Table 1 and the protection mechanisms listed in Table 2, both referring to a specific station.

After a certain amount of evaluations, the strategy parameters have been assigned the following values:

- $N = 100$
- $p(C) = 0.7$
- $p(R) = 0.3$

- *p(M)* = 0.1
- ε = 0.01
- max number of generations = 500

By assuming an available budget of 25 K€, the optimal coverage set (percentage values) of protection mechanisms found by the algorithm is [98, 0, 21, 58, 61, 37]. The related total cost of the optimal set of protection mechanisms is 22891 €, while the expected benefit is 31867 €. The null value of the second coverage parameter of the solution means that the related protection mechanism should not be considered in the design of the security system, while the 98% of the first means that a complete coverage is the best solution (the result should be interpreted as an approximation of the "ideal" solution). The other values which are not 0 nor 100 demonstrate the non linearity of the optimization problem: since the "effectiveness" relationships between protection mechanisms and threats are "many-to-many" (and not straightforward), once a certain level of coverage is reached for a protection mechanism, its cost-effectiveness can be lower than the competing protection mechanisms. In other words, the cost-effectiveness of protection mechanisms is dynamic and it changes together with the values of the other coverage parameters. In other words, the input parameters are "interacting".

Different executions of the algorithm have shown a convergence to the result reported above, which has been obtained after 40 generations and 500 ms (average values). In some cases, the algorithm has converged to sub-optimal coverage sets: a typical issue with genetic algorithms which can be solved by adjusting strategy parameters (such as *p(M)*).

We have assumed that the cost of a protection mechanism has a fixed contribution (i.e. licensing, integration, etc.) and variable part which is proportional to its coverage level. That is realistic in many situations, but please note that any more complex cost function could be adopted.

Finally, we point out that in the result the total cost of the protection mechanisms exploits the most part of the available budget, but not the whole. That is a normal result and not a sign of a suboptimal solution, since the convergence criterion does not involve any "budget exploitation" check and, furthermore, we have designed the algorithm to choose the less costly individual of the last population, that is the one for which the convergence criteria are met.

5. Conclusions and future developments

In this paper we have presented an approach to support the design of physical security systems by automatically optimizing the coverage of protection mechanisms. The possibility of optimization of the return on investment (with a budget constraint) is enabled by the availability of a quantitative risk management tool. Due to the nature of the problem and to the very large search space of possible solutions, the optimization technique has been chosen to be based on genetic algorithms.

The optimizer inherits the limitations of the analytical model used for risk management, e.g. neglecting possible interdependencies between risk parameters. However, since it works independently from the specific method used to compute the total annual benefit (which is simply invocated as an external function), new models can be used in the analysis without any modification to the genetic algorithm.

Furthermore, since the optimization approach is viable regardless of problem size, it is possible to extend the set of free parameters by including effectiveness related ones. In fact, while it is reasonable that effectiveness parameters are fixed for a given protection mechanism, it is always possible to combine more devices of different nature to improve detection reliability (and thus the so called “protective effectiveness”) or reduce the false alarm rate, the latter having a negative impact on the objective function.

The optimization approach is based on a quantitative model based methodology for risk management. A good modeling of the system in terms of identified threats, threat categories, assets, protection mechanisms together with their parameters is the necessary condition for a good risk assessment and an acceptable solution of the optimization.

Finally, it should be noticed that due to the many variables involved in the decision process, a complete security design automation is far from being obtainable; nevertheless, the results of the toolkit described in this paper serve as a useful indication, which is one step towards such a challenging goal.

References

1. Abraham, A., Grosan, C. and Snasel, V. (2009) ‘Programming Risk Assessment Models for Online Security Evaluation Systems’, in *UKSim 2009, 11th International Conference on Computer Modelling and Simulation*, pp. 41-46.
2. Alotto, P.G. et al. (1998), ‘Stochastic Algorithms in Electromagnetic Optimization’, *IEEE Transactions on Magnetics*, vol. 34, No. 5, pp 3674-3684..
3. Bang, Y.H. et al. (2004), ‘The Design and Development for Risk Analysis Automatic Tool’, in Lecture Notes in Computer Science Vol. 3043/2004, *Computational Science and Its Applications – Proc. ICCSA 2004*. Springer..
4. Banković, Z., Stepanović, D., Bojanić, S and Nieto-Taladriz, O. (2007), ‘Improving network security using genetic algorithm approach’, *Computers & Electrical Engineering*, Vol. 33, No. 5-6, pp. 438-451..
5. Broder, J.F. (2006), ‘Risk Analysis and the Security Survey’. Butterworth-Heinemann.
6. Chambers, L.D. (2000), ‘The Practical Handbook of Genetic Algorithms: Applications, Second Edition’. CRC Press.
7. Flammini, F., Gaglione, A., Mazzocca, N. and Pragliola, C. (2008), ‘Quantitative Security Risk Assessment and Management for Railway Transportation Infrastructures’, in *Proc. 3rd International Workshop on Critical Information Infrastructures Security, CRITIS’08*, Frascati (Rome), Italy, Oct. 13-15, 2008 . Springer, pp. 213-223.
8. Flammini, F. et al. (2009), ‘Wireless Sensor Data Fusion for Critical Infrastructure Security’, in *Advances in Intelligent and Soft Computing, Vol. 53, Proc. International Workshop on Computational Intelligence in Security for Information Systems, CISIS’08*. Springer, pp. 92-99.
9. Garcia, M.L. (2001), ‘The Design and Evaluation of Physical Protection Systems’. Butterworth-Heinemann.
10. Garcia, M.L. (2005), ‘Vulnerability Assessment of Physical Protection Systems’. Butterworth-Heinemann.
11. Goldberg, D.E. (1989), ‘Genetic Algorithms in Search, Optimization and Machine Learning’. Reading, MA: Addison-Wesley.
12. Hansen, J.V., Lowry, P.B., Meservy, R.D. and McDonald, D.M. (2007), ‘Genetic programming for prevention of cyberterrorism through dynamic and evolving intrusion detection’, *Decision Support Systems*, Vol. 43, Issue 4, Aug 2007, pp. 1362-1374.
13. Holland, J.H. (1975), ‘Adaption in Natural and Artificial Systems’. Ann Arbor: University of Michigan Press.
14. Holland, J.H. (1992), ‘Genetic algorithms’, *Scientific American*.

15. Lim, Y.T., Cheng, P.C., Clark, J.A. and Rohatgi, P. (2008), 'Policy evolution with Genetic Programming: A comparison of three approaches', in *IEEE World Congress on Computational Intelligence*. 1-6 Jun 2008, pp. 1792 – 1800.
16. Indu, S., Chaudhury, S. Mittal, N.R. and Bhattacharyya, A. (2009), 'Optimal sensor placement for surveillance of large spaces', in *Third ACM/IEEE International Conference on Distributed Smart Cameras, 2009. ICDSC 2009*. Como, IT, Aug 30 – Sep 2, 2009, pp. 1-8.
17. Ferentinos, K.P. and Tsiligiridis, T.A. (2007), 'Adaptive design optimization of wireless sensor networks using genetic algorithms', *Computer Networks*, Vol. 51, Issue 4, Mar 2007, pp. 1031-1051.
18. Martins, F.V.C. et al. (2010), 'An Evolutionary Dynamic Approach for Designing Wireless Sensor Networks for Real Time Monitoring', in *2010 IEEE/ACM 14th International Symposium on Ditributed Simulation and Real Time Applications (DS-RT)*. Fairfax, VA, Oct. 17-20, 2010, pp. 161-168.
19. Nicol, D.M., Sanders, W.H. and Trivedi, K.S. (2004) 'Model-based evaluation: from dependability to security', *Dependable and Secure Computing, IEEE Transactions on*, Vol. 1, Iss. 1, pp. 48-65.
20. Painton, L. and Campbell, J. (2005), 'Genetic algorithms in optimization of system reliability', *IEEE Transactions on Reliability*, Vol. 44, Issue 2, pp. 172-178.
21. U.S. Department of Transportation (2004), 'Transit Security Design Considerations', *Federal Transit Administration, Final Report*.
22. Whitley, D. (1994), 'A Genetic Algorithm Tutorial'. *Statistics and Computing*, Vol. 4, No. 2, pp. 65-85.
23. Wilson, J.M. et al. (2008), 'Securing America's Passenger-Rail Systems'. RAND Corporation.
24. Yao, X.H. (2010), 'A network intrusion detection approach combined with genetic algorithm and back propagation neural network', in *2010 International Conference on E-Health Networking, Digital Ecosystems and Technologies (EDT)*. Shenzen, CN, Apr. 17-18 2010, pp. 402-405.