DEPARTMENT OF ENGINEERING MANAGEMENT

A metaheuristic approach for security budget allocation in utility networks

Jochen Janssens, Luca Talarico & Kenneth Sörensen

UNIVERSITY OF ANTWERP
Faculty of Applied Economics

City Campus
Prinsstraat 13, B.226
B-2000 Antwerp
Tel. +32 (0)3 265 40 32
Fax +32 (0)3 265 47 99
www.uantwerpen.be

FACULTY OF APPLIED ECONOMICS

DEPARTMENT OF ENGINEERING MANAGEMENT

A metaheuristic approach for security budget allocation in utility networks

Jochen Janssens, Luca Talarico & Kenneth Sörensen

RESEARCH PAPER 2014-028
DECEMBER 2014

University of Antwerp, City Campus, Prinsstraat 13, B-2000 Antwerp, Belgium
Research Administration – room B.226
phone: (32) 3 265 40 32
fax: (32) 3 265 47 99
e-mail: joeri.nys@uantwerpen.be

D/2014/1169/028

A metaheuristic approach for security budget allocation in utility networks

Jochen Janssens, Luca Talarico, Kenneth Sörensen

University of Antwerp Operations Research Group ANT/OR

Prinsstraat 13, 2000 Antwerp, Belgium

October 2014

Real-life utility networks such as smart grids, pipelines and water networks can be exposed to safety and security related risk. Preventative measures can be applied to mitigate such risks that may result in service interruptions for the users of these networks. An index of (dis)connectivity, that measures how many users will be affected by a network failure, is defined and embedded into a model aimed at increasing security in a utility network. The model proposed in this paper assumes that all edges (e.g. pipes, cables) and nodes (e.g. switching- or connection stations, substations in an electricity network) have a certain probability of failing (due to external attacks), which can be reduced by taking appropriate security strategies. It then determines the security strategies that should be taken to minimize the probability of disconnecting any node from any other node. We propose a metaheuristic approach to solve this problem. Detailed experiments on realistic instances are conducted and the relationship between solutions and problem parameters are tested by simulating different scenarios.

Keywords: network security, metaheuristics, knapsack problem.

1 Introduction

People make the assumption that utility networks such as the ones used to transport electricity, water, gas and data are always available. They take for granted that a high level of service is guaranteed at all times and that the network can handle whichever load is placed on it. In reality, failures may happen which can lower the service level or even discontinue the supply. This might result in a disconnection of certain subsections from the rest of the network.

Network interruptions can have different causes: (i) *safety*-related such as construction defects, material failures, ground movement, natural disasters, unintentional human errors (ii) *security*-related such as intentional terrorist acts, vandalism, sabotages etc.

In Rees et al. (2011), security countermeasures are used to guarantee the confidentiality, availability, and integrity of data in computer systems that might be subjected to cyber-attacks. A decision support system, based on a genetic algorithm, is proposed to select the "best" combination of countermeasures that respect the user's preferred trade-off between the cost of the selected security measures and resulting risk.

A mixed integer programming approach is proposed by Sawik (2013) to support decision makers in the selection of the optimal mix of countermeasures in IT systems to prevent or mitigate cyber-threats. The proposed model relies on the definition of a limited number of potential attack scenarios that simplifies the decision process aimed at balancing expected worst-case losses and the cost of the selected security measures. Differently from his approach, in this paper a generalization of the decision model to be applied to utility networks is proposed and an efficient metaheuristic is proposed to solve larger instances. Furthermore, every possible attack/threat scenario is also considered in the approach described in this paper.

Our approach extends the works of Reniers and Dullaert (2012) and Reniers et al. (2012) by defining a single-objective problem and proposing a quantitative method to select appropriate security strategies. A different objective function is used, which relies on the minimization of the probability of the network to be not accessible between any couple of network nodes instead of the maximization of the effectiveness of the security measures used. Moreover, in our work, since a list of security strategies is defined for each arc and each node of the network, the model incorporates not only decisions taken at the level of the network, as done in Reniers and Dullaert (2012), Reniers et al. (2012) and Sawik (2013), but it depends on the choices made at the level of single network arcs or single network nodes.

In a previous model of Janssens et al. (2014), service interruptions, as a consequence of *security*-related accidents, have been treated considering only two points in the network i.e. the point from which the service or the product is send to the customer and the customer or the point to which the product, service is delivered through the network. This simplified model enables network providers and managers to reduce the probability of a network breakdown by applying security strategies to arcs, that might be potential target of terrorists. This paper extends this basic model to suit more realistic cases.

In reality, the arcs are not the only network elements that are at risk. The probability for an attack on substations in an electricity network, switching stations in a communication network, etc. should also be considered. These strategic points in the network might be of even higher importance to the network than the arcs, and the survival of them might be needed for a reliable service to its customers. It is clear that when a node is unavailable, it is equivalent to all of the arcs that connect to this node being unavailable.

The application of security strategies is more often than not subject to a security budget that service providers have to their disposal.

The goal of the service provider is to enable the service to all customers at all times. Previously, only connections between two points in the network, origin and destination of a service, have been considered. In a more realistic case, however, the connectivity between all nodes should be analysed. For example, in a communication network, it is clear that every customer should be able to reach every other customer. In this paper, we extend the model of Janssens et al. (2014) to guide the service providers in their decision process of which security strategies to apply to maximise the probability of connectivity between all customers. This decision process is subjected to a security budget limitation.

The paper is organized as follows. In Section 2, the problem of selecting the best strategies to increase the security of the whole network is described and modelled as an optimization problem. In Section 3, we present a metaheuristic to solve the network security problem. Section 4 shows some results. Section 5 presents the findings and concludes the paper.

2 Problem description

The utility network can be represented by using a graph $\mathcal{G} = \{\mathcal{N}, \mathcal{A}\}$, where N represents the set of nodes and $\mathcal{A}$ the set of arcs, connecting the nodes. All arcs $i \in \mathcal{A}$ and nodes $k \in \mathcal{N}$ have a probability of being attacked, denoted as p_i^a and p_k^n, where the index a refers to arcs and index n is used for the nodes. Successful attacks will be called "events" for the remainder of this work. We assume that when an attack happens, it is always successful and destroys the node or arcs it happens on.

A set of security strategies $\mathcal{S}_i^a$ and $\mathcal{S}_k^n$, is defined for each arc $i \in \mathcal{A}$ and each node $k \in \mathcal{N}$ in their respective ways. For each security strategy j in $\mathcal{S}_i^a$ (or $\mathcal{S}_k^n$) of arc i (node k) there is a cost c_{ij}^a (c_{kj}^n), and a value p_{ij}^a (p_{kj}^n), which is the probability of an event on arc i (node k) when this security strategy is applied. A security strategy can be a combination of single security measures (see e.g. Table 1). A combination of security measures can have a different effectiveness than the sum of the impact of the individual security measures due to some interaction effects. In some cases, combinations of single security measures might not be available due to their incompatibility (e.g. in Table 1, Infra-red remote sensors and Thermal infra-red remote sensors are not compatible and hence this combination is not in the list of security strategies). The probability p_i^a of arc i and p_k^n of node k after applying a security strategy j will take the probability p_{ij}^a or p_{kj}^n that is associated to that security strategy.

The default security strategy 0 for arc i, that has a cost $c_{i0}^a = 0$, is a base case that indicates that no security measure is applied. Its related probability p_{i0}^a represents the

chance of an event on arc i in case no security strategy is selected. This also applies to the security strategies and probabilities for the nodes.

Table 1: Examples of security strategies for pipelines

Strategy	security measures	cost	probability
0	-	0	0.6
1	Fences	100	0.5
2	Infra-red remote sensors	150	0.45
3	Thermal infra-red remote sensors	200	0.4
4	Fences & Infra-red remote sensors	230	0.32
5	Fences & Thermal infra-red remote sensors	290	0.25

In the simplified model in Janssens et al. (2014) with a given origin node o and a destination node d in the network, we can say that these two nodes are disconnected if all possible paths between o and d are not available. This would make it impossible for a service or good from node o to reach node d (e.g. it would be impossible to make a phone call from node o to node d, if all connections to node d were unavailable). This check for connectivity can be extended to be applied to any couple of nodes in the network. The service would be considered disrupted if there exists at least one couple of nodes in the network that cannot be connected. This gives rise to a natural index of connectivity. If we would consider a combination of events on the network, either on arcs or nodes, this index can be calculated as the percentage of couples of nodes that can reach one another. If we consider a network with n nodes, where $n = |\mathcal{N}|$, and a combination of events, that cuts the network into pieces, the index of connectivity, Con, would be calculated as follows:

$$Con = \frac{\sum_{o=1}^{n} \sum_{d=1}^{n} r_{od}}{n \cdot (n-1)} \tag{1}$$

where r_{od} is 1 if there is an available path from node o to node d, 0 otherwise.

To calculate the probability of a combination of events to happen, and by extension the risk for any combination to disconnect service anywhere in the network, we make use of probability theory. Probability theory is used extensively in reliability theory and in reliability studies of systems, a field of research that has received a lot of attention in the past years. For an overview, we refer to Bazovsky (2004); Ministery of Defence (UK) (2011); Romeu (2004).

In order to mathematically describe the decision problem associated to the selection of the best security strategy mix to increase network security, first we will have to define the probability associated with disconnection of service in network $\mathcal{G}$ between any couple of nodes. For this purpose we define a set $\mathcal{C}$ which contains all critical combinations of events on the network. A critical combination of events is defined as a combination of events that disrupts the service for at least one couple of nodes in the network.

The reader should be aware of the complexity of calculating the set $\mathcal{C}$. The current approach used is to calculate all possible combinations of events happening, which is a set of $2^{|\mathcal{A}|+|\mathcal{N}|}$ combinations. For this set we can then check if the combinations are critical. It might be worth investigating in the future if a faster approach can be found to calculate the set $\mathcal{C}$, without explicitly calculating and enumerating all possible combinations.

Each element l of set $\mathcal{C}$ is made by combinations of arcs and nodes for which an event happens, contained in the set $\mathcal{A}_l^E$ and $\mathcal{N}_l^E$ respectively, and combinations of arcs and nodes for which no events happen contained in sets $\mathcal{A}_l^N$ and $\mathcal{N}_l^N$ respectively. It should be noted that $\mathcal{A}_l^E \cup \mathcal{A}_l^N \cup \mathcal{N}_l^E \cup \mathcal{N}_l^N = \mathcal{A} \cup \mathcal{N},\ \forall l \in \mathcal{C}$. In other words, an element l in set $\mathcal{C}$ contains a critical scenario. If the arcs or nodes in $\mathcal{A}_l^E$ and $\mathcal{N}_l^E$ are out of service, a network breakdown between at least one couple of nodes is generated. The cardinality of set $\mathcal{C}$ depends on the topology of the network $\mathcal{G}$.

Let B represent the available security budget and x_{ij}^a a binary variable, that takes values 1 when the security strategy j on arc i is applied, and 0 otherwise, and x_{kj}^n a binary variable, that takes values 1 when the security strategy j on node k is applied, and 0 otherwise. Set $\mathcal{S}_i^a$ includes all the security strategies j for arc i with $j = 0$ being the situation in which no security measures for arc i are applied. The same applies for set $\mathcal{S}_k^n$ for all nodes $k \in \mathcal{N}$.

$$\min \sum_{l \in \mathcal{C}} R_l \tag{2}$$

s.t.

$$\sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{S}_i^a} c_{ij}^a \cdot x_{ij}^a + \sum_{k \in \mathcal{N}} \sum_{j \in \mathcal{S}_k^n} c_{kj}^n \cdot x_{kj}^n \leq B \tag{3}$$

$$p_i^a = \sum_{j \in \mathcal{S}_i^a} p_{ij}^a \cdot x_{ij}^a \qquad \forall i \in \mathcal{A} \tag{4}$$

$$p_k^n = \sum_{j \in \mathcal{S}_k^n} p_{kj}^n \cdot x_{kj}^n \qquad \forall k \in \mathcal{N} \tag{5}$$

$$R_l = \prod_{i \in \mathcal{A}_l^E} p_i^a \cdot \prod_{i \in \mathcal{A}_l^N} (1 - p_i^a) \cdot \prod_{k \in \mathcal{N}_l^E} p_k^n \cdot \prod_{k \in \mathcal{N}_l^N} (1 - p_k^n) \qquad \forall l \in \mathcal{C} \tag{6}$$

$$\sum_{j \in \mathcal{S}_i^a} x_{ij}^a = 1 \qquad \forall i \in \mathcal{A} \tag{7}$$

$$\sum_{j \in \mathcal{S}_k^n} x_{kj}^n = 1 \qquad \forall k \in \mathcal{N} \tag{8}$$

$$x_{ij}^a \in \{0, 1\} \qquad \forall i \in \mathcal{A}, \forall j \in \mathcal{S}_i^a \tag{9}$$

$$x_{kj}^n \in \{0, 1\} \qquad \forall k \in \mathcal{N}, \forall j \in \mathcal{S}_k^n \tag{10}$$

The objective function (2) minimizes the total probability for service interruption in the network. The total network probability is given by the sum of probabilities associated to single combinations of events happening. Constraint (3) ensures that the total cost associated to the selected security strategies does not overcome the predefined security budget B. Equations (4) and (5) are used to define the probabilities p_i^a, associated to an attack happening on arc i, and p_k^n, associated to an attack on node k. Equation (6) gives us the probability that a combination of events, which disconnects the network between at least one couple of nodes, might happen. Equations (7) and (8) force the decision process to select at maximum one security strategy to protect arc i or node k. It should be noticed that $x_{i0}^a = 1$ and $x_{k0}^n = 1$ means that for arc i and node k no security measures have been applied. To conclude, constraints (9)-(10) enforce the domain of the decision variables. Applying a partial security strategy is not allowed.

3 Solution approach

Exact approaches to solve the problem, described in Eq.(2)-(10), are viable methods for small instances only, due to the exponential explosion of the number of critical combinations to be considered. In fact, in the worst case, the number of critical combinations that have to be analysed and updated trough the solution process is equal to $2^{|\mathcal{A}|+|\mathcal{N}|}$. Moreover, the maximum amount of possible ways to combine security strategies is equal to $\prod_{i\in\mathcal{A}} |\mathcal{S}_i^a| \cdot \prod_{k\in\mathcal{N}} |\mathcal{S}_k^n|$. In case the number of security strategies is equal for all nodes and arcs and simplified as $|\mathcal{S}|$ this value can be rewritten as $|\mathcal{S}|^{|\mathcal{A}|+|\mathcal{N}|}$.

The decision problem of selecting the best mix of security strategies given a budget limitation belongs to the more general category of knapsack problems. The knapsack problem represents a well-known class of combinational optimization problems (see Wilbaut et al. (2008) for more details). The problem, presented in Section 2, has a non-linear objective function and therefore belongs to the class of non-linear knapsack problems, also known as non-linear resource allocation problems. The latter is known to be even more complex to solve than linear Knapsack problems (Bretthauer and Shetty, 2002).

As the problem instances become larger, an exact algorithm will require an exponential amount of time. Therefore, the optimality is sacrificed for near optimal solutions that can be calculated in a very short amount of time and a metaheuristic approach is preferred. In Algorithm 1, an iterated local search algorithm (ILS) is proposed (Lourenço et al., 2010) that combines a greedy random adaptive search procedure (GRASP) with a variable neighbourhood descent (VND) to improve the current solution. Two perturbation heuristics are used to escape from local optima. In addition, a tabu list is used during the whole execution of the heuristic to avoid an exploration of solutions that have been analysed in previous iterations.

Algorithm 1 Metaheuristic structure

Step 0: Initialization
Read instance & Initialize Heuristic parameters
let x be the current solution and $f(x)$ its cost
let x^* be the best solution found so far and $f(x^*)$ its cost
$x^* \leftarrow \{\emptyset\}$, $f(x^*) \leftarrow \infty$
Step 1: Construction phase
$x \leftarrow$ GRASP heuristic() (see Algorithm 2)
while (stopping criterion not reached) **do**
  Step 2: Intensification phase
  $k \leftarrow 0$
  while $(k < k_{Max})$ **do**
    $x' \leftarrow N_k(x)$
    if $(f(x') < f(x))$ **then**
      $x \leftarrow x'$
    else
      $k \leftarrow k + 1$
    end if
  end while
  if $(f(x) < f(x^*))$ **then**
    $x^* \leftarrow x$,$f(x^*) \leftarrow f(x)$
  end if
  update number of iterations without improvement
  Step 3: Diversification phase
  if (max number of iterations without improvement not reached) **then**
    $x \leftarrow$ Perturbation(x)
  else
    $x \leftarrow$ GRASP heuristic()
  end if
end while
return x^*

After the initialization step, an initial solution is generated by using a GRASP constructive heuristic in Step 1 of Algorithm 1. This solution is constructed step by step by adding promising security strategies for arcs and nodes. This selection is repeated until the security budget does not allow any further security strategies as shown in Algorithm 2. To facilitate the process of finding the arcs and nodes that have the highest potential to reduce the probability of service failure between nodes in the network, we define a method that ranks the set of critical combinations, $\mathcal{C}$, based on the index of connectivity Con defined in section 2. Let Con_l represent the connectivity measure for element l in $\mathcal{C}$, and R_l the probability for that combination to happen. We can then define ρ_i^a and ρ_k^n as the rank of arc i and node k as follows:

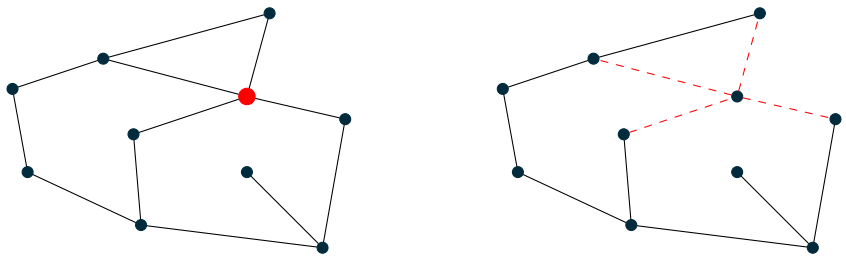
$$\rho_i^a = \sum_l (1 - Con_l) \cdot R_l \cdot f_{il}^a \qquad \forall l \in \mathcal{C} \tag{11}$$

$$\rho_k^n = \sum_l (1 - Con_l) \cdot R_l \cdot f_{kl}^n \qquad \forall l \in \mathcal{C} \tag{12}$$

where f_{il}^a (f_{kl}^n) is 1 if the arc i (node k) is present in set $\mathcal{A}_l^E$ ($\mathcal{N}_l^E$), 0 otherwise.

To calculate the connectivity for each critical combination, we apply an algorithm where, starting from a start node, all neighbours of that node are assigned to the same group. If no more neighbours are found, we move to the next node that has not been assigned to a group yet, until no more nodes are left. To calculate the amount of disconnections between nodes, the cardinality of each group is multiplied with that of every other group, and then summed. We divide the amount of disconnections with the total number of theoretical connections to get a percentage value, then we subtract that from 1 to get the percentage of connectivity.

It is worth noticing that, when a node is subjected to an attack, it is equivalent to an attack on all the incident arcs. In other words if a node is not available due to an attack, the arcs entering in and leaving from that node need to be considered out of service. In the calculations of the index of connectivity an attack on a node is treated as attacks on the arcs that enter or leave from the node. An example is shown in Figure 1



(a) Attack on a node (b) Equivalent attack on arcs

Figure 1: Substitution of an attack on a node (1(a)) with attacks on arcs (1(b)) for the computation of the index of connectivity

After the generation of an initial solution by the GRASP heuristic, a local search heuristic is applied during the Intensification step (Step 2 of Algorithm 1). The local search makes use of a VND heuristic to improve the current solution. The VND heuristic explores the neighbourhoods of three different local search operators. The first one (N_1) tries to replace one security strategy used inside an arc (or node) with another not used security strategy within the same arc (or node). The second move (N_2) tries to replace one security strategy used for an arc (or node) with another security strategies associated to a different arc (or node). The third move (N_3) removes the security strategies from two arcs (or nodes) and they are then replaced with different security strategies from the same two nodes or arcs (this can be seen as a budget redistribution between the arcs or nodes). A move is executed until no further improvement is found. The whole VND heuristic is executed until a local optimum is found and no further improvement can be obtained.

Algorithm 2 GRASP Heuristic

```
Initialize parameters
let x be an empty solution
let B be the remaining budget
let T be the list with arcs and nodes sorted on rank
let H be the list of considered security strategies
let s_ij be security strategy j for arc or node t_i, and c_ij its cost
let s*_ij be the selected security strategy for arc or node t_i
let s* be the selected security strategy
let b be the best reduction so far, b ← 0
let r_ij be the reduction of security strategy s_ij
while (B > 0 && ! all strategies checked) do
  if (Best improvement strategy) then
    T ← top ranking arcs and nodes
    for all t_i in T do
      let S be the set of security strategies for arc or node t_i
      for all s_ij in S do
        if (s_ij not in tabu list && c_ij < B && r_ij > b) then
          b = r_ij
          s*_ij ← s_ij
        end if
      end for
      H add s*_ij
    end for
    for all s_ij in H do
      if (r_ij > b) then
        b = r_ij
        s* ← s_ij
      end if
    end for
    x add s_ij, update B
  else
    t_i ← random arc or node without security strategy selected
    select random security strategy s_ij from t_i
    x add s_ij, update B
  end if
end while
return x
```

Finally, in step 3, which is a diversification stage, a perturbation is applied to escape this local optimum, and the algorithm continues with a local search on this perturbed solution. The arcs and security strategies that are removed from the solution in the perturbation step, are added to the tabu list for a specified amount of iterations. If, after a fixed number of perturbations, the algorithm cannot find a better solution, the algorithm is restarted from a new solution generated by the GRASP constructive heuristic.

4 Computational results

In this section, the computational experiments are shown. The solution approach, described before, has been tested on some realistic instances.

In preliminary tests, the metaheuristic has been tuned in order to find its best parameter settings by performing a full factorial experiment on realistic instances. Next the algorithm is executed with these optimal parameter settings on a larger set of instances to analyse the behaviour of the algorithm. The obtained results are compared with the best solutions found by an exact approach that is run for a maximum amount of time of 5 hours. The exact approach generates solutions for the problem by using a full enumeration method with the budget constraint as a cutting plane. In Table 2, the percentage difference between the best solutions obtained by the metaheuristic over 25 runs and the exact approach is reported in column *Best Gap*, while column *Average Gap* represents the percentage gap between the average solution over 25 runs and the best solution found by the exact approach. It can be observed that the metaheuristic finds, in most cases, solutions that are better than the exact approach, that is run for a limited time. Moreover, these solutions of high quality are also found in a very limited running time, that is only a small fraction of the time allowed to the exact approach.

	Exact Approach		**Metaheuristic**				
Instance	Best Solution	Time(s.)	Best Solution	Average Solution	Average Time (s.)	Best Gap	Average Gap
NS-n7-c5-C3-a30-x0	0.09587	18000	0.09715	0.09810	8.748	1.336%	2.329%
NS-n7-c5-C3-a30-x1	0.07922	18000	0.07531	0.07609	8.266	-4.941%	-3.960%
NS-n7-c5-C3-a30-x10	0.09040	18000	0.09021	0.09091	8.834	-0.207%	0.560%
NS-n7-c5-C3-a30-x11	0.08331	18000	0.08375	0.08479	5.538	0.527%	1.782%
NS-n7-c5-C3-a30-x12	0.05130	18000	0.05103	0.05161	4.688	-0.533%	0.598%
NS-n7-c5-C3-a30-x13	0.09077	18000	0.08955	0.08955	5.684	-1.350%	-1.350%
NS-n7-c5-C3-a30-x14	0.08065	18000	0.08088	0.08181	6.001	0.285%	1.445%
NS-n7-c5-C3-a30-x2	0.06996	18000	0.06836	0.06836	6.135	-2.283%	-2.283%
NS-n7-c5-C3-a30-x3	0.08838	18000	0.08544	0.08623	7.915	-3.320%	-2.434%
NS-n7-c5-C3-a30-x4	0.10045	18000	0.09707	0.09757	10.657	-3.365%	-2.871%
NS-n7-c5-C3-a30-x5	0.08526	18000	0.08359	0.08396	6.043	-1.967%	-1.531%
NS-n7-c5-C3-a30-x6	0.08572	18000	0.08627	0.08706	5.547	0.643%	1.569%
NS-n7-c5-C3-a30-x7	0.09421	18000	0.09164	0.09261	5.884	-2.726%	-1.698%
NS-n7-c5-C3-a30-x8	0.08398	18000	0.08285	0.08322	5.180	-1.349%	-0.906%
NS-n7-c5-C3-a30-x9	0.07961	18000	0.07802	0.07875	7.604	-1.994%	-1.078%

Table 2: Exact approach in comparison with the metaheuristic

In Figure 2, the current and best found objective values for one instance are plotted over time. The spikes in the current solution show the points in which a perturbation is executed. After a perturbation, the VND heuristic lowers the objective value in small steps until we reach a local optimum and another perturbation is executed. The objective function has a very steep descent at the start of the algorithm and gradually stabilizes. This shows that the algorithm converges towards very good results in a very short time. As CPU time gets larger, the marginal improvement of the best solution found so far becomes smaller and smaller. This shows that infinitely increasing the CPU time will not necessarily produce significantly better results.

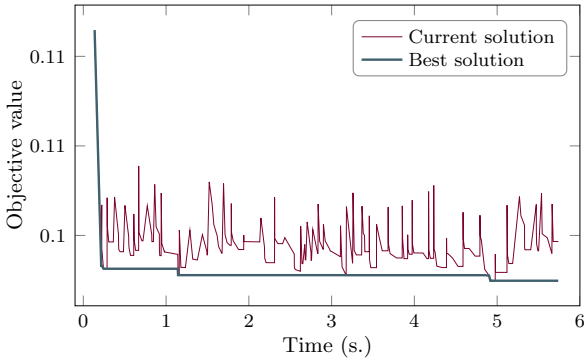


Figure 2: Plot of the objective value over time.

5 Conclusion & Discussion

In this paper, we described a model for the selection of appropriate security strategies given a limited budget, to increase the security of infrastructure such as pipelines transportation systems, telecommunication networks, smart grids, etc. Redundancies in the network might be used by service providers or network owners in order to keep the service available in case of problems affecting a single arc (node) of the network. However, multiple attacks or attacks directed to critical arcs or nodes might disrupt the complete infrastructure and service. An exact evaluation of the probability for the whole network to be down might be a difficult task, especially when several loops are present inside the network. Loops might be added to increase the networks reliability.

We assume that each arc and node presents different characteristics in term of vulnerability to external attacks due to internal and external factors such as geographical condition, length, materials used, operating conditions, etc. In addition, we assume to have for each arc or node a list of security strategies available, each one with different characteristics in terms of cost and effectiveness.

A mathematical formulation of the problem, aimed at mitigating and reducing the probability of network failures between any couple of nodes, is proposed. This decision support model relies on the definition of critical scenarios in which we suppose that arcs and nodes might be unavailable due to malicious attacks. All possible attack scenarios are evaluated in which any node or arc is considered a viable target of terrorist groups. In order to prevent such episodes, that could induce significant economic loss, security strategies can be implemented to increase the network's reliability. The goal of the problem is to select the optimal combinations of security strategies for each arc and node, within a budget defined by the service providers or network owners.

The decision model, considered in this paper, addressed multilevel decisions, since a decision made at the level of a single arc or node might affect the security of the whole network. We proposed a heuristic algorithm, which exploits the benefits offered by tabu

search combined with a GRASP and an iterated local search solution approach, to solve this combinatorial optimisation problem. Inside the metaheuristic, an approach to rank the importance of arcs or nodes to be secured is proposed. This method uses an index of disconnection between the nodes of a network, guiding the solution method to select promising strategies.

We tested our solution approach on a set of instances that mimic possible realistic scenarios. The results are compared with an exact approach which is run for a limited time.

In future research, the model can be extended even more by making a differentiation in the types of nodes (customers or suppliers) and/or considering the importance of nodes themselves. In terms of the algorithm, alternative solution approaches (e.g. evolutionary, large scale neighbourhood search, ant colony algorithms) can also be developed and compared with the one proposed in this paper.

Acknowledgements

This research is supported by the Interuniversity Attraction Poles (IAP) Programme initiated by the Belgian Science Policy Office (COMEX Project).

References

I. Bazovsky. *Reliability Theory and Practice*. Dover Civil and Mechanical Engineering Series. Dover Publications, 2004. ISBN 9780486438672.

Kurt M Bretthauer and Bala Shetty. The nonlinear knapsack problem – algorithms and applications. *European Journal of Operational Research*, 138(3):459 – 472, 2002.

J. Janssens, L. Talarico, and K. Sörensen. A decision model to increase security in a utility network. In *Proceedings of the 11th international multidisciplinary modelling & simulation multiconference*. I3M, 2014.

H.R. Lourenço, O.C. Martin, and T. Stützle. *Iterated Local Search: Framework and Applications*. Springer New York, 2010.

Ministery of Defence (UK). Chapter 6: Probabilistic R&M Parameters and redundancy calculations. In *Applied R&M Manual for Defence Systems (GR-77), Part D - Supporting Theory*. UK Ministery of Defence, Abbey Wood, Bristol, 2011.

Loren Paul Rees, Jason K. Deane, Terry R. Rakes, and Wade H. Baker. Decision support for cybersecurity risk planning. *Decision Support Systems*, 51(3):493 – 505, 2011.

G. Reniers and W. Dullaert. Tepitri: A screening method for assessing terrorist-related pipeline transport risks. *Security Journal*, 25(2):173–186, 2012.

G. LL. Reniers, K. Sörensen, and W. Dullaert. A multi-attribute systemic risk index for comparing and prioritizing chemical industrial areas. *Reliability Engineering & System Safety*, 98 (1):35–42, 2012.

Jorge Luis Romeu. Understanding series and parallel systems reliability. *Selected Topics in Assurance Related Technologies (START)*, 1(5):1 – 8, 2004.

T. Sawik. Selection of optimal countermeasure portfolio in IT security planning. *Decision Support Systems*, 55(1):156–164, 2013.

Christophe Wilbaut, Said Hanafi, and Said Salhi. A survey of effective heuristics and their application to a variety of knapsack problems. *IMA Journal of Management Mathematics*, 19 (3):227–244, 2008.