ISSN 2278-3091

Volume 9, No.4, July – August 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse39942020.pdf

https://doi.org/10.30534/ijatcse/2020/39942020

Decision Support System on Optimization of Information Protection Tools Placement

Lakhno V. A[1]., Lakhno M.V.[2]
Sauanova K. T[3]., Sagyndykova Sh. N.[4], Adilzhanova S. A.[5]

[1]National University of Life and Environmental Sciences of Ukraine, Department of Computer systems and networks, Kyiv, Ukraine, valss21@ukr.net

[2]National University of Life and Environmental Sciences of Ukraine, Department of Computer systems and networks, Kyiv, Ukraine, t121@i.ua

[3] Almaty University of Power Engineering & Telecommunications, Almaty, Kazakhstan, klartag@mail.ru

[4] Almaty University of Power Engineering & Telecommunications, Almaty, Kazakhstan, tomka2001@mail.ru

[5]Al-Farabi Kazakh National University, Almaty, Kazakhstan,asaltanat81@gmail.com

ABSTRACT

The article discusses the possibilities of modifying the genetic algorithm (GA) to solve the problem of selecting and optimizing the configuration of information protection tools (IPT) for the security circuits of information and communication systems (ICS). The scientific novelty of the work lies in the fact that the GA proposes the use of the total amount of risks from information loss, as well as the integral indicator of IPT and cost indicators for each class of IPT, as criteria for optimizing the composition of IPT. The genetic algorithm in the problem of optimizing the choice of the composition of IPT for ICS is considered as a variation of the problem associated with multi-selection. In this regard, the optimization of the IPT placement along the ICS protection circuits is considered as a modification of the combinatorial knapsack problem. The GA used in the computing core of the decision support system (DSS) differs from the standard GA. As a part of the GA modification, the chromosomes are presented in the form of matrices, the elements of which are numbers that correspond to the IPT numbers in the ICS nodes. In the process of modifying the GA, a *k*-point crossing-over was applied. A fitness function is presented as the sum of efficiency factors. At the same time, in addition to the traditional absolute indicators of the IPT efficiency, the total value of risks from information loss, as well as cost indicators for each class of IPT are taken into account. The practical value of the research lies in the implementation of DSS based on the proposed modification of GA. Computational experiments on the selection of a rational software algorithm for implementing the model are performed. It is shown that the implementation of the GA in DSS allows to accelerate the search for optimal variants of cybersecurity (CS) tools placement for ICS by more than 25 times. This advantage allows not only to quickly search through the variety of hardware-software IPT and their combinations for ICS, but also to combine the proposed algorithm with the available models and algorithms for optimizing the composition of the ICS cybersecurity circuits. Potentially, such a combination of models and algorithms will make it possible to quickly rebuild the ICS protection, adjusting its profiles in accordance with new threats and classes of cyber attacks.

Key words: decision support system, information protection tools, multi-criteria optimization, knapsack problem, genetic algorithm

1. INTRODUCTION

As the number and complexity of successfully implemented cyberattacks on various information and communication systems (ICS) [1] – [2] grows, the need for qualitatively new procedures for forming the composition of information protection (IP) and cybersecurity (CS) complexes on all ICS protection circuits increases. Note that the permanent task of creating effective circuits for the ICS cybersecurity has generated a lot of researches on the optimization of the composition of information protection tools (IPT) and CS. These researches are primarily intended to answer questions related to the solution of multi-criteria optimization problems that have such properties as: a complex configuration of the acceptable area of application of individual IPT; multi-extremeness of the considered functions; algorithmic definition of functions, etc. Moreover, in real-life problems of creating effective multi-circuit CS systems [3] – [4], it is rarely customary to evaluate solutions using a single criterion. Therefore, in such problems it is important not only to find acceptable Pareto-optimal solutions, but also to approximate the many variants obtained in order to offer the decision-maker (DM) an objective choice of IPT according to the corresponding CS circuits of ICS. The solution of the above-described problems of creating multi-circuit information protection systems in the context of an increase in the number of attempts of destructive effects on ICS requires the use not only classical optimization procedures, but also more universal methods, for example, genetic algorithms (GA), which have proved their effectiveness in solving many complex problems [5] – [6].

The GA efficiency is determined by careful tuning and control of their parameters. This somewhat complicates the use of the GA in conventional engineering calculations of the IPT effectiveness along the ICS circuits. However, the use of the GA becomes quite justified if, in addition to the traditional multi-criteria optimization task on selection of the composition of IPT for ICS, there is considered the value of risks, as well as the cost indicators of the selected IPT for specific assets (databases, knowledge bases, mail, website, etc.). The decision search procedure can be more effective if you use the potential of intelligent decision support systems (DSS), the computing core of which is actually based on the use of GA.

The above reasoning predetermined the relevance of researches aimed at improving evolutionary algorithms and models for the DSS computational core in the process of multi-criteria optimization of the composition of IPT along the CS circuits of ICS.

2. REVIEW AND ANALYSIS OF PREVIOUS STUDIES

Genetic algorithms used in solving multicriteria optimization problems are variants of evolutionary search methods [7]. A large number of works have been devoted to researches in this area over the past few years. So, for example, a model was described in [8], in accordance with which a population of IPT elements (individuals) is created, where in the optimization problem each individual corresponds to one of the possible solutions. In order to find the best solution, the authors used their own objective function. The paper does not indicate how and where specifically the proposed solutions were used in practice.

In [9] – [10], GAs were studied, which can be attributed to two groups. In the first group, GAs with binary coding were studied [10] – [11]. In the second group, respectively, GAs with valid coding [12] – [13]. In [12], it was shown that in the first group it is possible to achieve higher efficiency of searching for extreme values on the set of feasible solutions.

In [12] – [14] it was shown that the constant mutation of objects is used in most GA implementations. Variables in this case are more flexible, which makes it possible to find initial solutions already at sufficiently early stages of the GA operation. The software implementation of the algorithm is not presented in the works.

It was shown in [15] – [16] that, taking into account the great dependence of the successful operation of the algorithm on the mutation caused by the features of the task of forming CS circuits of ICS, a variable mutation looks preferable from the point of view of searching for a global optimum. This can be explained by the fact that at early stages of the GA, a large element of chance will act.

In [17] – [18], the features of using a modified GA in similar multi-criteria optimization problems were considered. The difference between a GA with a relative fitness function and a standard GA lies in the fact that during the operation of the algorithm, the fitness function used here was not the sum of the IPT efficiencies, which actually made up the chromosome, but the sum of the ratio of the efficiencies to the limiting characteristics of the IPT, or so-called - efficiency coefficient. Such a modification of the GA is essentially a disjunction of the standard GA and the greedy algorithm.

In [19] – [20], the possibilities of reducing the number of tunable GA parameters were investigated. Unlike the standard one, the solutions proposed by the authors did not contain a crossover operator. In fact, the solution was obtained on the basis of statistical information about the search space. Thus, accumulating and using this information, these algorithms can independently adapt to the task.

The task of optimizing the choice of the composition of IPT for ICS can be considered as a variation of the tasks associated with multi-selection [21] – [22]. In these works, the optimization of the placement of the CS circuit components is considered as a certain modification of the combinatorial knapsack problem. This approach is distinguished by a fairly simple formalization of the formulation and interpretation of the solution. However, the authors did not provide a complete solution and comparison of solution algorithms, for example, genetic, simple enumeration, dynamic programming, etc.

However, we should note that the problem of a multiplicative knapsack does not reflect all the possibilities associated with replacing objects from one class of IPT by the objects from other classes. However, replaceable objects perform equivalent functions. Therefore, instead of the total value of objects, it is necessary to introduce a function that displays many goals. This is done in order to reflect the interchangeability or equivalence of objects in the "knapsack" [24] – [25].

It was shown in [17] – [20] that standard and modified GAs are efficient enough to solve most complex optimization problems [23] and are promising for further study and improvement.

All of the above mentioned predetermined the relevance of the research aimed at the development of a GA for the DSS computing core in the problems of optimizing the selection of IPT and CS for ICS of various objects of informatization.

3. THE PURPOSE AND OBJECTIVES OF THE WORK

The purpose of the research is the development of a genetic algorithm for the computing core of a decision support system in the process of selecting and optimizing the composition of information protection tools.

In order to achieve the goal of the research, it is necessary to solve the following tasks:

To develop clarifications for the GA taking into account the total amount of risk and the cost of the CS system;

To develop and implement DSS based on the computing core of the GA for the selection and optimization of the IPT composition;

To conduct computational experiments in order to verify the performance of the modified GA and the proposed DSS.

4. METHODS AND MODELS

The ICS infrastructure from the point of view of ensuring CS and IP is presented on Figure 1.

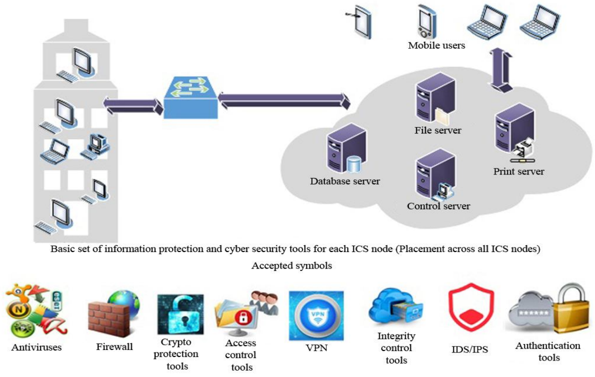


Figure 1: The ICS infrastructure from the point of view of ensuring cybersecurity and information protection

By default, standard IPT [23] are installed in the ICS nodes: antivirus; firewall; tools for: 1) intrusion detection 2) cryptographic IP; 3) access control; 4) integrity control; 5) authentication, etc.

Naturally, for a specific ICS, the list can be supplemented due to insufficiency or reduced due to redundancy.

The NIST recommendations describe in detail the architecture, the main vulnerabilities, and the features of ensuring CS and IP in ICS. We note, however, that today there is no universal approach that can provide an unambiguous solution in the process of searching for the optimal solution of IPT and CS placement through the ICS nodes, taking into account all the features of a particular ICS, analyzing the mechanisms for ensuring CS and the effectiveness of IPT in relation to the available range of threats. As a result, the question of creating such an approach arises. In this case, the resulting solution should differ in the following features:

1. Possibilities for designing different variants for the information and cybersecurity circuits, based on the structure of a specific ICS.
2. Opportunities for the selection of IPT based on the needs to counter specific threats of different classes.
3. The ability to adaptively (evolutionarily) change the algorithm for selecting and optimizing sets of IPT and CS, based on the evolution of attack mechanisms. And this, in turn, makes it impossible to use only exact methods for selecting IPT for ICS nodes.

Based on the foregoing, we consider the possibility of using GA to solve the problem of multi-selection in the process of selecting the optimal configuration (hereinafter referred to as a set) of IPT (for example, antivirus, firewalls, intrusion detection tools, etc.) for ICS.

Formalization of the problem in terms of GA.

We believe that the chromosome is a set of protective measures (for example, the rules for observing the information security policy at the object of protection), including IPT. The set is encoded as a binary number [8-9]. If the binary digit of the number is equal to one (**1**), then the corresponding IPT or information protection measure with the corresponding number is included in the set. Then the range of code changes can be represented as follows:

$$G = \left(d_0 d_1 ... d_{NC}\right)_2 = \left(0...2NC\right)_{10}, \tag{1}$$

where $NC -$ the number of available IPT and protective measures that are potentially considered for inclusion in the optimal set; $d_i -$ inclusion of IPT and/or protective measures in the set.

In terms of GA, a population will consist of instances with different chromosomes. The population size is limited by the maximum number of instances in it. Each population instance can be described as follows:

$$Ch = \{G, C, R\}, \tag{2}$$

where $G -$ genetic code of an instance in a population;

$C -$ cost of IPT and/or corresponding protective measures;

$R -$ the total risk of information loss (or its confidentiality, integrity), taking into account the selected IPT and/or corresponding protective measures (hereinafter - IPT).

In the process of modifying the algorithm in order to determine the risk, the following assumption is used. The absolute value of losses in monetary terms for a particular ICS depends on a chain of elements – threats, vulnerabilities, IPT [2–3]. Therefore, the number of risks is the number of combinations of threats and assets:

$$R = TH \cdot M, \tag{3}$$

where $TH -$ the number of threats, $M -$ the number of assets.

Formula (3) does not take into account the combination of several threats, as well as the internal influence between IPT. Therefore, a more appropriate way to determine the risks for ICS is a method that is based on the preparation of attack profiles [3–4]. With this method, attack profiles are considered as sequences of attacks that consist of a combination of different threats [1–4]. Then the number of risks can be described by the following dependence:

$$R = \left(2^{TH}\right)^{TA}, \tag{4}$$

where $TA -$ the number of attacks.

Consequently, the value of the risk for a given attacks profile can be considered as the amount of total damage from successful attacks.

If there is no chain reaction during the attack, then the total risk value can be represented as the mathematical expectation of damage for each ICS asset:

$$r = \sum P_{i,j} \cdot D_{i,j}, \quad i = \overline{1, TH},\ j = \overline{1, M}, \tag{5}$$

where $P_{i,j} -$ the probability of an ICS information security incident caused by a threat (i) to the asset (j);

$D_{i,j} -$ the amount of damage associated with the incident (accepted in cash).

A chromosome (Ch) can be represented as a matrix. Then, the rows of the matrix will represent the placement points, respectively, the columns are the classes of tools that include specific IPT (for example, the antivirus software class includes all variants of the antivirus programs under consideration – Avast, Avira, AVG, Bitdefender, etc.). The matrix element g_{ij} shows the number of information protection tools from the class j located on the node i. If $g_{ij} = 0$, then we believe that no tool is used from the class j on the node i, for example, antivirus software is not used on the firewall.

The scheme of the formation of the GA chromosome (Ch) is presented in table 1.

Table 1: The chromosome formation scheme

Countermeasure Classes Network nodes	N_1	N_2	…	N_{NC}
K_1	g_{11}	g_{12}	…	g_{1NC}
K_2	g_{21}	g_{22}	…	g_{2NC}
…	…	…	…	…
K_{KC}	g_{KC1}	g_{KC2}	…	$g_{KC,NC}$

In this format for representing the chromosome and in the context of the problem being solved, we assume that K_{KC} and N_{NC} are, respectively, the number of ICS nodes and IPT on the node.

To calculate the risk (R), we use the following model.

We select the appropriate IPT for each carrier by the genetic code.

Let introduce the utility function in the GA – U. This function is necessary to evaluate the effectiveness of the selected IPT in the set. Note that the selected IPT must match the profile of the attack. After all, it is completely clear that, for example, free anti-virus software with limited functionality is completely useless to combat DoD/DDoS attacks, and instructions for observing the security policy for ICS alone will not protect against an insider.

Then the utility function (U) can be written as follows:

$$U(Ch) = R_0 - Ch.R, \tag{6}$$

where $Ch -$ IPT set instance;

$R_0 -$ the value of the risks associated with the loss of information, if there is not applied the corresponding set of IPT;

$Ch.R -$ the value of the risks taking into account the use of the corresponding set of IPT (instance) $Ch.G.$

However, the achieved effectiveness in protecting the ICS from attacks, respectively, requires additional costs for IPT. We take into account the impact of costs on IPT using the following relation:

$$U(Ch) = {(R_0 - Ch.R)}\big/{Ch.C}, \tag{7}$$

where $Ch.C -$ IPT set cost.

Relation (7) shows how it is possible to reduce (or increase) the risk of information loss for each invested cost unit.

Next, we consider how the resulting expressions can be applied in GA. In the syntax of high-level programming languages, the functions of crossing-over, mutation, and selection will look like this.

The crossover function is the creation of new carriers. In the process of DSS software implementation on the basis of GA, two types of crossing-over were considered. The possibilities of using single-point and n-point crossing-over were analyzed. The choice of these two types is due to the following considerations. The standard approach based on single-point crossing-over is suitable for most tasks in which the search for a solution using GA is advisable. We note, however, that for the problem of multi-selection of IPT for ICS nodes, a standard GA will be very inaccurate. This is due to the fact that the chromosome will not be a single indivisible structure. In the formulation of this problem, the chromosome can be interpreted as a system that needs a decomposition procedure. Decomposition will allow the chromosome to be divided into sections, and each section will have its own class of ICS nodes.

Let create a new instance for each pair that will inherit the features of the parents (PA).

$$\begin{aligned}
&\mathit{func}\ K(PA) := \mathit{foreach}\ Ch(X)\ \mathit{from}\ PA\\
&\mathit{and}\\
&\mathit{foreach}\ Ch(Y)\ \mathit{from}\ PA\\
&\mathit{where}\ Ch(X) != \\
&Ch(Y)\ \mathit{do}\\
&R.\mathit{add}\left(\{G : \mathit{xor}\left(Ch(Xi).G, Ch(Xj).G\right), C :, R :\}\right)\\
&\mathit{return}\ R.\mathit{add}(PA)
\end{aligned} \tag{8}$$

Next, we consider the mutation function, i.e. variation of the genetic code. Two types of mutation were considered. This is due to the following assumptions: 1) a constant mutation is used in most GA software implementations; 2) the variables of our task require more flexibility, and for our task the dependence of the GA success on the mutation is more than on crossing-over; 3) assumption 2 is due to the fact that there are objective features of solving problems associated with the formation of the CS circuits of ICS. This led to large chromosome sizes, as well as the presence of limitations.

Therefore, a variable mutation, which is characterized by elements of randomness at the early stages of the algorithm, will be more preferable from the point of view of searching for the optimal composition of the knapsack.

In the process of computational experiments, two types of mutations were considered. The first type is a constant mutation. In this case, each position in the chromosome with a probability of 1% will be inverted. The second type is a variable mutation. In this case, the probability of a mutation will depend on the current needs of the GA. The mutation coefficient will vary in the range of 1-6%.

In the GA under consideration with a relative fitness function, the fitness function did not use the sum of the IPT efficiencies, which actually made up the chromosome, but the sum of the efficiency ratios or integral IPT indicators that are a part of the class.

To do this, we randomly invert two binary digits in the chromosome:

$$\begin{aligned}
&\mathit{func}\ M(PA) := \mathit{foreach}\ Ch(X)\ \mathit{from}\ PA\ \mathit{do}\ Ch(X).G = \\
&= \mathit{xor}\left(Ch(X).G, 1 << \mathit{rand}(0, NC)\right).
\end{aligned} \tag{9}$$

Then the selection function, i.e. selection of the best carriers can be written as:

$$\mathit{func}\, S(PA) := \mathit{return}\, PA.\mathit{sort}(\ ).\mathit{slice}(1, K). \tag{10}$$

Note that in order to reduce recording and reduce the population, we leave only the K carriers that give the greatest result with respect to the utility function (U).

Before applying selection, we also preliminarily calculate $Ch(X).C$ and $Ch(X).R$ for the population. In accordance with [5.7], it is accepted that the initial population at least includes two instances. Then each era in the GA [6–7] will consist of a sequential application of the basic functions discussed above. Accordingly, we get:

$$\mathit{func}E(\ ) := ((PA = K(PA), M(PA)), (P = S(PA))). \tag{11}$$

As a result of the DSS operation, the optimal set of IPT for ICS node is determined based on the integral indicator of each IPT from the corresponding class and cost of this IPT. As the quality index or the degree of attainability of the desired goals for a specific IPT there is accepted an integral indicator (InI) of IPT [23]. InI can also be interpreted as a generalized indicator of the quality of the most important characteristics of a particular IPT. At the same time, we believe that the InI is calculated as the degree of proximity of the IPT parameters to ideal characteristics in the space of selected particular indicators [23]. To calculate the InI of IPT, the following dependence was used:

$$IND_j = \sum_{i=1}^{k} \beta_i \cdot a_{ij}, \qquad (12)$$

where $\beta_i -$ weight of the criterion used to evaluate the *i*-th IPT (for example, for firewalls you can use the following criteria: firewalls test to protect against internal attacks; firewalls test to protect against external attacks; personal IDS/IPS test to protect against attacks on vulnerable applications; documentation available and etc.)

$a_{ij} -$ degree of achievement of a given level of node protection for the *j*-th attack class;

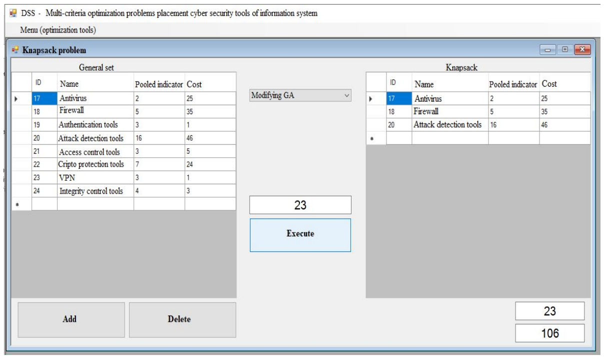
$k -$ number of IPT classes for a particular type of ICS node.

Based on the analysis and calculation of InI for classes of IPT, you can get an idea of the maximum value of InI based on weight coefficients for IPT classes without using an expert method of their formation

Decision Support System Software Implementation

The software implementation of the DSS computational core, the models described above, as well as the algorithm that implements the task of searching for the optimal strategy for the formation of IPT sets for CS circuits of ICS, was performed in C# programming language in the Microsoft Visual Studio 2019 environment. The concept of DSS is based on the existing ICS architecture, a combination of classes and IPT, as well as applying the hierarchy analysis method (T. Saati method) at the first stage, the GA solves the problem of forming the optimal variant of IPT placement in each of the ICS key nodes, see Figure 1. The main DSS window is shown on Figure 2.

After entering all the data related to the general set of objects to ensure CS and IP (i.e., chromosomes) that could potentially be used on the ICS node to ensure the required level of cyber security and adjustment of the calculation parameters, the GA starts to work directly. For this, GA forms a population that consists of 25 chromosomes. Next, the fitness function (efficiency) of each chromosome is calculated. GA uses *k*-point crossing-over. In fact, *k* is the number of classes of IPT placement points for ICS.


DSS - Multi-criteria optimization problems placement cyber security tools of information system

Menu (optimization tools)

Knapsack problem

General set

ID	Name	Pooled indicator	Cost
17	Antivirus	2	25
18	Firewall	5	35
19	Authentication tools	3	1
20	Attack detection tools	16	46
21	Access control tools	3	5
22	Cripto protection tools	7	24
23	VPN	3	1
24	Integrity control tools	4	3

Modifying GA

23

Knapsack

ID	Name	Pooled indicator	Cost
17	Antivirus	2	25
18	Firewall	5	35
20	Attack detection tools	16	46

23

106

Figure 2: General view of the DSS interface

The final set for the knapsack, i.e. the analyzed ICS node, is shown in the table on the right side of the form on Figure 2. That is, the problem of forming a multi-knapsack is actually being solved. DSS is developed using an object-oriented approach.

The GA used in the DSS computational core differs from the standard GA by the following features: chromosomes are presented in the form of matrices, the elements of which are numbers that correspond to the numbers of IPT in the ICS nodes; *k*-point crossing-over is applied. A variable mutation was used, i.e. the probability of a mutation can adaptively change during the operation of the GA, depending on needs. A fitness function is presented as the sum of efficiency coefficients. At the same time, in addition to the traditional absolute indicators of the IPT effectiveness (which are integrated in the integral indicator), the total value of risks from information loss, as well as cost indicators for each class of IPT are taken into account.

Computational Experimentat

In order to verify the adequacy of the algorithm and the DSS by multi-criteria optimization of the IPT placement on the ICS nodes, corresponding computational experiments were carried out, see Figures 3–6.

Computational experiments were conducted for randomly generated sets of IPT. The efficiency of a modified GA, the branch and bound method, and the greedy algorithm were compared.

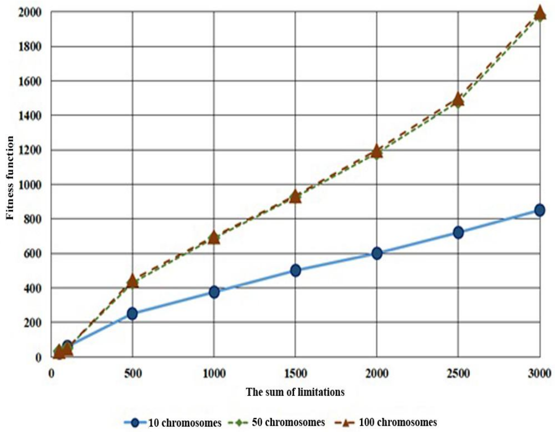


Figure 3: Comparison of algorithm efficiency for a different number of chromosomes in a population

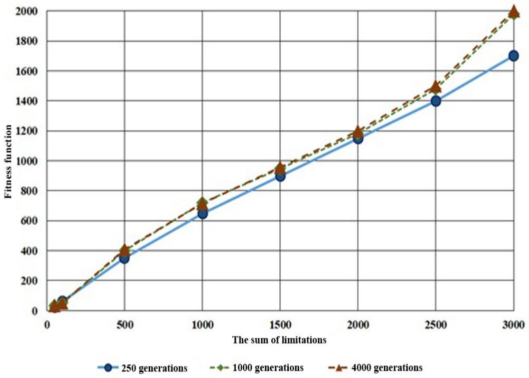


Figure 4: Comparison of algorithm efficiency for a different number of generations

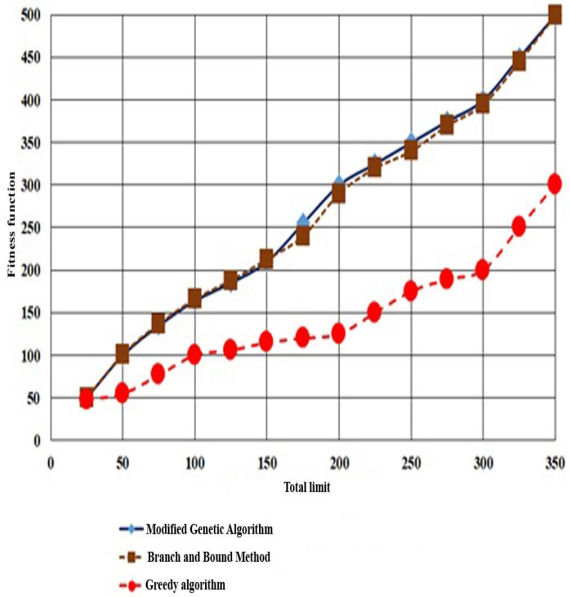


Figure 5: The results of computational experiments comparing the efficiency of algorithms used in DSS

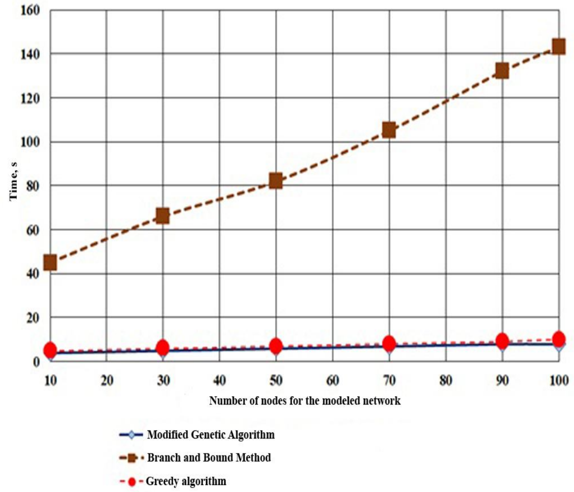


Figure 6: The results of computational experiments comparing the running time of algorithms

5. DISCUSSION OF THE RESULT OF A COMPUTATIONAL EXPERIMENT

The graphs of Figure 3 show the results of GA studies in the search for the optimal number of chromosomes in a population to solve the problem of searching for the optimal sets of information protection tools for ICS.

As it is shown by computational experiments, the optimal result cannot be achieved if the number of chromosomes is not large (less than 20). However, with an increase in their number to more than 22–25, the efficiency of the algorithm did not increase. Based on a series of more than 500 computational experiments, it was found that for the final version of the algorithm and its software implementation in DSS it is enough to take 25 chromosomes in a population.

A series of computational experiments were also carried out in the search for the optimal number of generations for the considered GA, see Figure 4. The verification showed that the GA efficiency does not increase after overcoming the 450–500 generation threshold. This circumstance gives a reason to limit the number of generations in the GA for our DSS to 500 generations.

During the computational experiments, it was found that the GA is quite high in efficiency, as well as in speed, see Figures 5, 6.

It was established that the time spent on solving the problem using GA is approximately 16–25 times less in comparison with the indices of the branch and bound method. The greedy algorithm is significantly inferior to both the GA and the branch and bound method from the point of view of being adaptable to solving a multi-criteria optimization problem, taking into account the imposed limitations and the number of variables.

Thus, the analysis shows that the developed models and the algorithm are reliable, and the results of computational experiments are repeatedly confirmed by a series of practical implementations.

Certain disadvantages of the study, at the present stage of its implementation, include the fact that not all possible algorithms for solving the problem were analyzed. In particular, nowadays, DSS does not allow solving the problem on the basis of a modified Whitley model (D. Whitley) [24] or an ant algorithm [25] – [26].

Currently, work is underway to add these algorithms to the list of available in DSS [27] – [28]. A wide range of DSS variants will make it more functional for solving the problem under consideration.

6. CONCLUSIONS

Therefore, the following results are stated in the article:

1. The possibility of modifying the genetic algorithm (GA) to solve the problem associated with the selection and optimization of configuration variants of information protection tools (IPT) for the security circuits of information and communication systems is considered. The scientific novelty of the work lies in the fact that the GA proposes the use of the total value of risks from information loss, integrated indicators of IPT, as well as cost indicators for each class of IPT as criteria for optimizing the composition of IPT. The genetic algorithm in the problem of optimizing the choice of the composition of IPT for ICS is considered as a variation of the problem associated with multi-selection. In this regard, the optimization of the IPT placement along the ICS protection circuits is considered as a modification of the combinatorial knapsack problem. The practical value of the research lies in the implementation of a decision support system based on the proposed modification of the GA.

2. Computational experiments were carried out to select a rational program algorithm for implementing the model. As a rational variant, it is proposed to use the GA modification. It is shown that the implementation of GA in DSS allows to accelerate the search for optimal placement of CS tools for ICS by more than 25 times. This advantage allows not only to quickly search through various variants of IPT hardware-software and their combinations for IC, but also to combine the model presented in the article with the available models and algorithms for optimizing the composition of the ICS cybersecurity circuits. Potentially, such a combination of models and algorithms will make it possible to quickly rebuild the ICS protection.

ACKNOWLEDGMENT

This work was carried out as a part of grant funding for the project AP05132723 “Development of adaptive expert systems in the field of cybersecurity of critically important informatization objects” (Republic of Kazakhstan). The authors of the article are also grateful to the staff of the Department of Computer Systems and Networks of the National University of Life and Environmental Sciences of Ukraine for help with the DSS software implementation and data processing from computational experiments.

REFERENCES

1. A. Okutan, S.J.Yang, K. McConky, & G.Werner. **CAPTURE: Cyberattack Forecasting Using Non-Stationary Features with Time Lags**. *IEEE Conference on Communications and Network Security (CNS), 2019, pp. 205–213.* https://doi.org/10.1109/CNS.2019.8802639
2. C. Barreto & X. Koutsoukos. **Design of Load Forecast Systems Resilient Against Cyber-Attacks**. *International Conference on Decision and Game Theory for Security, 2019, pp. 1–20.*
3. Y. Chandra & P.K. Mishra. ***Design of cyber warfare testbed***. Software Engineering, 2019, pp. 249–256.
4. H. Sándor, B.Genge, Z. Szántó, L. Márton & P. Haller. **Cyber attack detection and mitigation: Software Defined Survivable Industrial Control Systems**. *International Journal of Critical Infrastructure Protection, vol.25, 2019, pp. 152–168.*
5. Z. Chiba, N. Abghour, K. Moussaid, El Omri A., & Rida M. **New Anomaly Network Intrusion Detection System in Cloud Environment Based on Optimized Back Propagation Neural Network Using Improved Genetic Algorithm**. *International Journal of Communication Networks and Information Security, vol.11(1), 2019, pp.61–84.*
6. Y. Nozaki & M. Yoshikawa. **Security evaluation of ring oscillator puf against genetic algorithm based modeling attack**. *International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2019, pp. 338–347.*
7. S. Dwivedi, M. Vardhan & S. Tripathi**. Incorporating evolutionary computation for securing wireless network against cyberthreats**. *The Journal of Supercomputing, 2020, pp.1–38.*
8. F. Zhang, H.A.D.E. Kodituwakku, J.W. Hines & J.Coble. **Multilayer Data-Driven Cyber-Attack Detection System for Industrial Control Systems Based on Network System, and Process Data**. *IEEE Transactions on Industrial Informatics, vol.15(7), 2019, pp. 4362–4369.*
9. T. Sureshkumar, B. Anand & T. Premkumar. **Efficient Non-Dominated Multi-Objective Genetic Algorithm (NDMGA) and network security policy enforcement for Policy Space Analysis (PSA)**. *Computer Communications, vol.138, 2019, pp.90–97.* https://doi.org/10.1016/j.comcom.2019.03.008
10. Q. Shang, L. Chen, D. Wang, R. Tong & P. Peng **Evolvable Hardware Design of Digital Circuits Based on Adaptive Genetic Algorithm.** *International Conference on Applications and Techniques in Cyber Security and Intelligence, 2019, pp. 791–800.*
11. Y. Yang, **Research on Hybrid Quantum Genetic Algorithm Based on Cross-Docking Delivery Vehicle Scheduling.** *The International Conference on Cyber Security Intelligence and Analytics (pp. 893–900), 2019.*
12. I. Saenko & I. Kotenko. **A role-base approach and a genetic algorithm for VLAN design in large critical infrastructures**. *Proceedings of the Genetic and Evolutionary Computation Conference Companion, 2019, pp. 1643–1650.*
13. Y. Aleksieva, H.Valchanov & V.Aleksieva. **An approach for host based botnet detection system**. *16th Conference on Electrical Machines, Drives and Power Systems (ELMA), 2019, pp. 1–4.*
14. R. Vinayakumar, M. Alazab, K.P. Soman, P. Poornachandran, A. Al-Nemrat & S. Venkatraman. **Deep learning approach for intelligent intrusion detection system**. *IEEE Access, 7, 2019, pp.41525–41550.*
15. N. Malarvizhi, P. Selvarani & P. Raj. **Adaptive fuzzy genetic algorithm for multi biometric authentication**. *Multimedia Tools and Applications,2019, pp.1–14.* https://doi.org/10.1007/s11042-019-7436-4
16. B. Alhijawi, Y. Kilani & A. Alsarhan. **Improving recommendation quality and performance of genetic-based recommender system**. *International Journal of Advanced Intelligence Paradigms, vol.15(1), pp. 77–88, 2020.*
17. U. Baroudi, M. Bin-Yahya, M. Alshammari & U.Yaqoub. **Ticket-based QoS routing optimization using genetic algorithm for WSN applications in smart grid**. *Journal of Ambient Intelligence and Humanized Computing, vol. 10(4), 2019, pp.1325–1338.*
18. T. Llansó., M. McNeil & C. Noteboom. **Multi-Criteria Selection of Capability-Based Cybersecurity Solutions**. *Proceedings of the 52nd Hawaii International Conference on System Sciences, 2019, pp. 7322–7330.*
19. T. Kong, L. Wang, D. Ma, Z. Xu, Q.Yang & K.Chen K. **A Secure Container Deployment Strategy by Genetic Algorithm to Defend against Co-Resident Attacks in Cloud Computing**. *IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2019, pp. 1825–1832.*
20. S. K. Lakshmanaprabu, S.N. Mohanty, S. Krishnamoorthy, J. Uthayakumar & K. Shankar. **Online clinical decision support system using optimal deep neural networks**. *Applied Soft Computing, vol.81, p.105487, 2019.*
21. D. Yan, F. Liu, Y. Zhang, K. Jia & Y. Zhang **Characterizing the Optimal Attack Strategy Decision in Cyber Epidemic Attacks with Limited Resources.** *International Conference on Science of Cyber Security, 2018, pp. 65–80.*
22. Y. Lee, T.J. Choi & C.W. Ahn. **Multi-objective evolutionary approach to select security solutions**. *CAAI Transactions on Intelligence Technology, vol.2(2), 2019, pp.64–67.*

23. B. Akhmetov., V. Lakhno, B. Akhmetov & Z. Alimseitova. **Development of sectoral intellectualized expert systems and decision making support systems in cybersecurity**. *Proceedings of the Computational Methods in Systems and Software, 2018, pp. 162–171.*
24. Dewri Rinku, et al. **Optimal security hardening using multi-objective optimization on attack tree models of networks**. *Proceedings of the 14th ACM conference on Computer and communications security. ACM*, 2007, pp. 204–213. https://doi.org/10.1145/1315245.1315272
25. P. Saurabh, B. Verma & S. Sharma. **Biologically inspired computer security system: the way ahead**. *International Conference on Security in Computer Networks and Distributed Systems*, 2012, pp. 474–484.
26. M.L. Priyanka, S.Rajeshwari, K.Ashwini. **An Expert model for DNA Based Encryption Technology using Cloud Computing**, *International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol. 8, No.1.3, 2019, pp.15–18,https://doi.org/10.30534/ijatcse/2019/0381.32019*
27. B. Madhuravani, N.Chandra Sekhar Reddy, K.Sai Prasad, B.Dhanalaxmi, V. Uma Maheswari. **Strong and Secure Mechanism for Data Storage in Cloud Environment,** *International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol. 8, No.1.3, 2019, pp. 29–33, https://doi.org/10.30534/ijatcse/2019/0681.32019*
28. B. Akhmetov, V.Lakhno, Y.Tkach, A.Adranova, G.Zhilkishbayeva, **Problems of Development of a Cloud-Oriented Educational Environment of the University.** *Environment, International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), Vol.9, No.2, 2020, pp. 2196–2203,https://doi.org/10.30534/ijatcse/2020/196922020*