Algorithmization of processes for detecting vulnerabilities of UMV interfaces based on probabilistic automata

A V Skatkov, A A Bryukhovetsky and D V Moiseev
Sevastopol state university, 33 Universitetskaya str., Sevastopol, Russia, 299053
E-mail: dmitriymoiseev@mail.ru

Abstract: The algorithmic approach based on the methods of adaptive intelligent technology for monitoring the state of objects of computer systems is considered. The approach is focused on the detection of changes in the state of controlled unmanned vehicle resources: communication channel, processor, memory. An adaptive model using the Bayesian classifier for assessing changes in the state of unmanned vehicle resources is presented. The model is based on a probabilistic automaton with adaptive self-tuning. The approach proposed in the article is focused on solving problems of detecting moments in time of changes in the state of controlled unmanned vehicle objects, which are resources: communication channel, processor, memory.

1. Introduction

The approach proposed in the article is focused on solving problems of detecting moments in time of changes in the state of controlled unmanned vehicle (UMV) objects, which are resources: communication channel, processor, memory. However, the speed and reliability of the assessment of the situation can be crucial. Implementation of such problems in real time is not always possible using an analytical approach, since these problems are characterized by inconsistency, non-linearity, undifferentiability, multi-extremality, complex topology of the domain of acceptable values, high computational complexity of optimized functions, high dimension of the search space, etc. In the context of a lack of a priori information, most of the problems of data analysis are related to the study of stochastic systems [1-4]. One of the most effective tools for modeling complex stochastic systems is the methodology of probabilistic-automaton modeling [5]. The productivity of this methodology is determined by the following features:

- availability of tools that provide an adequate description of complex stochastic systems and their functioning processes;
- ability to build unified models for a wide class of systems;
- use of decision support systems when it is necessary to accurately evaluate the choice of various alternatives based on probabilistic automata.

This paper is devoted to the application of the model of probabilistic automata for decision support in evaluating the information states of smart city transport infrastructure objects. Such objects, for example, include intelligent control systems for unmanned air and ground vehicles, systems that provide machine interaction using the Internet of things technology, and others. The heterogeneity of applications and wireless communications in the smart city infrastructure significantly complicates the security of facilities [6]. Methods of preventing attacks for the safe operation of vehicles must be...
dynamic and respond to possible threats. A proactive approach to threats should be a key requirement that must be met. However, since it is impossible to predict all possible threats to the UMV, it is important that users have as few violations as possible as a result of the attack. In [7-11], methods for detecting and classifying the main types of attacks on UMV that affect the performance of vehicles are considered. The main types of attacks include: distributed denial-of-service (DDoS) attacks, Black-hole attacks, man in the middle (MITM) attacks, Sybil-pseudo-spoofing attacks, impersonation-based attacks, Falsified-information attacks, and others.

Various probabilistic automaton models are currently known. The reason for the variety of automatic models is due to the breadth of their application. Probabilistic automata are used in such areas as: logical control, mathematical linguistics, theory of formal languages, modeling of human behavior, when describing models of enterprise information security, etc. The criterion of applicability of the automaton approach is best expressed through the concept of «complex behavior» [11]. An object can be said to have complex behavior if it can perform one of several output reactions as a response to some input action. It is significant that the reaction may depend not only on the input effect, but also on the background.

2. Problem statement
The aim of this work is to develop an adaptive model using a Bayesian classifier for estimating the states of UMV resources. The model is based on a probabilistic automaton with adaptive self-tuning. On the basis of the proposed model, the problem of assessing the state of resources is solved in order to increase the reliability of the results of classification of information situations. Denote

\[ R = \{R_1, \ldots, R_j, \ldots, R_r\} \]

set of the controlled UMV resources. Let's define the controlled characteristics of resources that we will use to evaluate their state (the values of the characteristics are normalized, defined in the range \([0;1]\)):

- \(D_j\) – loading of the \(j\)-th resource,
- \(V_j\) – rate of change \(D_j\), where \(V= (D(t_i) – D(t_i-1))/\Delta t\).

These characteristics are vector quantities with components of elements of sets \(R_j\).

Denote \(S_t\) – state of the \(i\)-th resource at time \(t\). The main task is adaptive estimation of probability values \(P(S_t)\) states of UMV resources formed for the normal behavior of the resource on the time interval \(T_N\) nd the values obtained as a result of external influence on the interval \(T_{2N}\). The use of this model will allow us to obtain reliable estimates of the hypotheses of the appearance of states \(S_t\).

3. Method for detecting changes in the state of UMV resources.
It is proposed to use an automatic probabilistic model for dynamic estimation of UMV resource states. For this purpose, we define a probabilistic automaton as a system

\[ \Sigma = (S, X, Y, \Phi, \Psi, S_I) \], (1) \]

where
- \(S_I \in S\) – initial state,
- \(S\) – set of state vector values,
- \(X\) – set of input vector values,
- \(Y\) – set of output vector values,
- \(\Phi\) – transition function,
- \(\Psi\) – output function.

We will record changes in the state of \(S_t\) – th resource on the \(T_N\) time interval at time points \(\{t_0, t_1, \ldots, t_j, \ldots, t_n\}\). In accordance with the given scheme (1), the automaton functions at discrete time points, which are clock cycles \(t_0, t_1, t_2\ldots\ldots\). Each clock cycle is associated with an input signal about the state of the resource – \(X\), output signal – \(Y\) and an internal state signal – \(S\). We will consider the
probabilistic Moore automaton model. In this case, the elements of the transition and output matrices represent the corresponding estimates of the probabilities of transitions between states. Let's introduce rules for the specified transitions:

\[ P(s_j(t+1)) = \Phi(s_j(t),x(t)) \] – probability of transition to a new state,

\[ P(y_j(t+1)) = \Psi(s_j(t+1)) \] – probability of output signal appearance,

where \( j=1,m \), \( m \) – number of states of the automaton, \( t = 0,1,2\ldots \).

At the initial time \( t \) the machine is in the state of \( S_I \). No output signal is generated at this time. For the initial state, the probability distribution of transition to internal states is set. The initial distribution of States is shown in figure 1.

![Figure 1](image1.jpg)

**Figure 1.** Initial state distribution of the probabilistic automaton.

In the first clock cycle when the signal \( x_j \) is received automaton with \( P_{Ij} \) probability goes into a state of \( S_j \) and the output signal \( Y_j \) is generated. The initial probability distribution of States is formed on the basis of a priori information that can be obtained, for example, during the normal functioning of the UMV in the absence of external disturbances.

The transition table sets estimates of the probability of transition to the state \( S_k(t+1) \) depending on the state \( s_j(t) \) if a signal is received \( x_i(t) \). Let's denote this probability – \( P_{ijk} \). In each row of the matrix, the transition probabilities form a complete group: \( \sum_{k=1}^{m} P(S_{i,j,k}) = 1 \). For a probabilistic automaton, you need as many transition tables as there are input signals \( x_i(t) \). An example of a transition table (figure 2) for an input signal is shown below \( x_i(t) \).

![Figure 2](image2.jpg)

**Figure 2.** The table of transitions of a probabilistic automaton.

The table of outputs of the Moore automaton (figure 3) is simplified in comparison with the Mile automaton, since the output signal \( Y \) depends only on the internal state \( S \) and does not depend on the input signal \( X \). Denote \( P_{ij} \) – probability of an output state \( Y_j(t+1) \) provided that the machine was in the state \( S_i(t+1) \).

![Figure 3](image3.jpg)

**Figure 3.** Table of outputs of the probabilistic Moore automaton.
With a fairly general statement of the problem, we are talking about monitoring the results of observations on the state of UMV resources. We assume that the state of the resource Rj at a given time t depends on the values of the characteristics Dj, Vj. Let these states be at a time interval denoted as Stj – set of possible object Rj states. State values are defined and normalized in the range [0;1]. Let the States be defined on the intervals [Ik;Ik+1], where Ik – threshold for setting the scope for determining the state of a resource, k=0,1…,j. Figure 1 shows an example of the difference of Stj states of the Rj –th resource at intervals: St0 ϵ[0;I0], St1 ϵ(I0;I1],….., Stj ϵ(Ij-1;1] :

![Figure 1](image)

Without loss of generality, we will further consider two possible states of the resource – S0, S1. Let's assume that the area with the number "0" indicates the normal state of the resource, and the area with the number "1" indicates the critical state.

Let's consider the process of functioning of the automaton on the example of a binary tree (figure 5)

![Figure 5](image)

Denote the level of the tree l=1,2,…,n, and k =1,2,…,2n – number of the vertex at the tree level. We will use the resource states at the current time as input signals. On the arcs, we will display the probabilities of transitions between internal states under the influence of the input signal. Denote P(S0), P(S1) – estimates of state transition probabilities S0, S1 respectively, P(S0)+ P(S1)=1. Initially, the machine is in its initial state – S1.

At each clock cycle ti ( i=1,2,…,n) the automaton enters a new state, which is located on the tree at
the next level. Let the current coordinate of a vertex in the tree be denoted by \((l, k(l))\). Then the automaton under the influence of the input signal from level \(l\) will move to level \((l+1)\) either in the left branch – state \(S_0\), or in the right one – state \(S_1\). Accordingly, the new position at level:

\[
\text{number_left}_{k(l+1)} = 2k(l)-1, \quad \text{number_right}_{k(l+1)} = 2k(l).
\]

Thus it is possible to save the sequence of state transitions of the automaton. This sequence represents a path in the time of \(n\) clock cycles, where \(n\) – is the length of the path in the tree. The path can be represented by a binary word \(d(n)\) of length \(n\) bits, in which each bit is \(d_i \in \{0,1\}\), where «0» denotes the state \(S_0\), «1» – the \(S_1\) state. If the path length is even, combinations with the same number of zero and single bits occur in a binary word. This may in some cases create additional uncertainties when making decisions in the process of classifying information states of resources. Therefore, we recommend using an odd number \(n\).

Thus, the state of a binary word describes a single path on the tree in \(n\) clock cycles. With an equally likely choice of States at the level \(l\), the probability of selecting any state (vertex) is \(1/2^l\), and the probability of a path of length \(n\) is defined as \(\prod_{l=1}^{n} (1/2^l)\). Since transitions are generally not equally likely, the probability estimate \(P(S_{n,k(n)})\) of transition to the \(k\)-th vertex at the level \(n\) (denote as \(k(n)\)) will be equal to

\[
P(S_{n,k(n)}) = \prod_{l=1}^{n} P(S_{l,k(l)})
\]

In the operating mode, the UMV is affected by external influences that lead to changes in the values of a priori probabilities. In order to compensate for the influence of external factors, it is proposed to use a method based on a probabilistic adaptive classifier based on a posteriori information obtained in the process of monitoring the state of UMV resources. The main task is to obtain estimates of the values of conditional distributions that determine the probability that an observation belongs to each of the possible classes. One of the known methods is based on the application of the Bayes theorem. In the context of the problem of detecting changes in the state of resources, the formula gets the following interpretation:

\[
P(S_i \mid N_{il}) = \frac{P(S_i)P(N_{il} \mid S_i)}{\sum_{i=1}^{n} P(S_i)P(N_{il} \mid S_i)}
\]

where \(S_i\), \(i = \{0,1\}\) – a priori information about the hypothesis of state probability estimates;

\(N_{il}\) – a posteriori information about the appearance of the \(N\) number \(S_i\) states at the \(l\)-th level;

\(P(S_i)\) – estimation of the probability of occurrence of the \(S_i\) state;

\(P(S_i \mid N_{il})\) – estimation of the conditional probability of state observation \(S_i\) at occurrence \(N_{il}\).

The algorithm for determining the values of estimates of conditional distributions of the appearance of States belonging to each of the possible classes contains the following sequence of actions:

1. The initial probability distribution is set as \(P(l)\), values \(n\), \(m\).
2. Transition probabilities are set as \(P_{ij}\).
3. A state vector \(S_i\) is played and the sequence \(S_{l,k(l)}\) is built, where \(l=1,2,...,n\), \(k=1,2,...,2n\), a single path is being built with a length \(n\).
4. Point 3 is repeated \(m\) times, \(m\) paths are constructed.
5. The number of \(N\) appearances of \(S_i\) states at level \(n\) is calculated.
6. Estimates of the probability of occurrence of states \(P_i = N_{in} / m\) are found.
7. In accordance with formula (2), estimates of conditional probabilities of observations belonging to each of the possible classes are determined.

The expert (decision-maker) determines the scenario for conducting experiments in the interactive mode. The experimental plan includes: setting a priori probabilities \(P(l)\), \(P_{ij}\), path length – \(n\), sample size – \(m\). The main goal is to simulate the process of detecting changes in the state of UMV resources; based on a posteriori information, to determine estimates of the probability of occurrence of states \(S_0\), \(S_1\), belonging to two opposite classes – normal and critical at different levels of the binary tree; provide
decision support when assessing the impact of the stochastic environment and making external impact; using nonparametric statistics criteria, get estimates of the influence of the stochastic environment; the obtained P(Si|Nil) values, etc. The obtained statistical results of experiments are used to select the values of the model parameters taking into account the specific conditions of different resource states.

4. Conclusion
The article considers an algorithmic approach to detecting vulnerabilities in UMV interfaces based on probabilistic automata. The proposed method is focused on detecting changes in the state of controlled UMV resources: communication channel, processor, memory. The method is based on the use of an adaptive model of a probabilistic automaton and a Bayesian classifier. A posteriori information about the state of resources during the operation of the UMV is used to compensate for the effects of the external stochastic environment. The proposed adaptive approach will increase the reliability and efficiency of decision support processes when solving problems of ensuring the security of objects of the critical information infrastructure "Smart city".

Acknowledgments
This work was partially supported by the Russian Foundation for basic research (grants № 19-29-06015 and 19-29-06023).

References
[1] Shiryaev A N 2014 Probabilistic and statistical methods in decision theory (Moscow: MCNMO)
[2] Skatkov A V 2012 Information Technologies for Critical Infrastructures (Sevastopol: SNTU)
[3] Skatkov A V, Bryukhovetskiy A A and Moiseev D V 2017 Intelligent monitoring system for solving large-scale scientific problems in cloud computing environments Information and control systems 2 19-25
[4] Skatkov A, Brykhovetskiy A and Moiseev D 2018 Detecting changes simulation of the technological objects’ information states MATEC Web of Conferences 224 02072 https://doi.org/10.1051/matecconf/201822402072
[5] Pospelov D A 1970 Probabilistic automata (Moscow: Energiya)
[6] Zegzhda P D, Poltavtseva M A and Lavrov D S 2017 Systematization of cyberphysical systems and assessment of their security? Problems of information security Computer system 2 127-138
[7] Al-kahtani M S 2012 Survey on security attacks in vehicular ad hoc networks (vanets) International Conference on Signal Processing and Communication Systems 12 1-9
[8] Pan L, Zheng X and Chen H 2017 Cyber security attacks to modern vehicular systems Journal of Information Security and Applications 36 90-100
[9] Markovitz M and Wool A 2017 Field classification, modeling and anomaly detection in unknown can bus networks Vehicular Communications 9 43-52
[10] Nilsson D K, Larson U E and Picasso F A 2019 First simulation of attacks in the automotive network communications protocol flexray Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems 1 84-91
[11] Shalyto A A 2008 Automata-based programming paradigm Scientific and technical Bulletin of St. Petersburg state University ITMO. Automata-based programming 53 3-23