CBR Method for Decision-making Support in Operation Efficiency Ensuring of Complex Technical Systems

Vladimir Vychuzhanin[1], Alexey Vychuzhanin[1] and Nickolay Rudnichenko[1]

[1] *Odessa Polytechnic National University, Shevchenko Avenue 1, Odessa, 65001, Ukraine*

Abstract

The sequence of decision-making, using the precedent-based reasoning method of the CBR cycle, which takes into account the operations of processing and structuring data based on precedents within the framework of the functioning of the developed application software system, ensures high performance, adaptability, work with incomplete information, versatility and learning ability when supporting decision-making to ensure efficiency operation of complex technical systems. The proposed information intelligent system allows, taking into account partial and complete failures of the operability of subsystems, components, elements and mutual connections of critical use complex technical systems to increase the efficiency of their operation as a result of system equipment pre-failure technical condition identifying signs at an early development stage.

Keywords

Information intelligent system, risk assessment, forecasting, technical condition, complex technical system, CBR cycle structure, similarity degree algorithm, complex systems pre-failure maintenance, performance

1. Introduction

Currently, there is an increase in the use of decision-making technologies for the effective operation of complex technical systems (CTS) [1, 2]. CTS are hierarchical structures with non-trivial internal connections, multiple functional subsystems, components, elements and their mutual connections, which are in various states of failure. The operation of CTS is associated with uncertainties [3]. One of the main causes of man-made accidents associated with the operation of CTS used in transport, aviation, energy, etc. remains the failure of their subsystems, components, and elements [4]. In this regard, such CTS are classified as systems of critical use. As the safety requirements for expensive CTS increase, the requirements for their efficiency also increase, depending on: the ability to operate the systems in the event of partial failures; time and resource during operation. In order to identify signs of pre-failure technical condition (TC) of CTS equipment of critical use at an early development stage it is necessary to use intelligent information technologies to support decision-making [5, 6]. The core of such information technologies are conceptual stochastic models and methods for diagnosing vehicles, effective intelligent information systems (IIS) for assessing, forecasting and controlling vehicles. The use of reasoning method based on CBR (Case-Based Reasoning) precedents simplifies decision-making for various types of uncertainties in the initial CTS data and expert knowledge, as well as in cases of emergency situations [7].

2. Description of problem

In order to increase the operational efficiency of critical ship CTS, solving the necessary decision-making problem is complicated due to the need to take into account a significant number of various factors. First of all, this is a large amount of information required to make an appropriate decision. When assessing, predicting and controlling a vehicle, it is necessary to take into account the mutual influence and different degrees of subsystems failure, components,

CMIS-2024: Seventh International Workshop on Computer Modeling and Intelligent Systems, May 3, 2024, Zaporizhzhia, Ukraine

icst_nuop@ukr.net (V.Vychuzhanin); vychuzhanin.o.v@op.edu.ua (A. Vychuzhanin); nickolay.rud@gmail.com (N. Rudnichenko)

0000-0002-6302-1832 (V.Vychuzhanin); 0000-0001-8779-2503 (A. Vychuzhanin); 0000-0002-7343-8076 (N. Rudnichenko)

elements and their parameters on each other.

The operation of shipboard CTS is subject to uncertainties that are difficult or impossible to fully describe, understand or predict. The presence of uncertainties affects onto vehicle's assessment, forecasting and management, which is due to the lack of clear data on the consequences of failures in the structural and functional relationships of vehicle's subsystems, components and elements makes problem more relevant.

In accordance with the requirements of the Register of Maritime Shipping, all ships must be equipped with systems based on artificial intelligence technology [4]. To implement such technology, appropriate algorithmic and software tools are required to provide support for decision-making that is adequate to the goal. However, analysis and assessment of the technical characteristics of CTS equipment and the development of control actions in most cases are often carried out by ship operators on the basis of heuristic rules. It is impossible for personnel servicing CTS of critical applications to analyze and evaluate the TC of system equipment. This problem can be solved by using methods used in IIS systems.

The development of IIS systems for assessment, forecasting and control vehicle's subsystems, components, elements in order to ensure ship CTS survivability under adverse impacts and damaging factors is one of the promising areas in ensuring CTS operation safety. Such IIS systems can be implemented both in the form of separate stand-alone solutions and in the form of modules that complement ready-made general-purpose control and decision-making systems with the necessary functionality. They will allow us making quick decisions at the stage of eliminating the consequences of adverse impacts and damaging factors, and ensure the efficiency ship CTS operation due to the ability to evaluate, predict and control their vehicles [8, 9].

Well-known models, methods, algorithms and software packages reflect, to one degree or another, the facts of successful solution of problems of using IIS systems for assessments, forecasting and management of CTS vehicles, having advantages and disadvantages. The main complexities of modern technologies and methods which are useful for IIS systems lie in solving complex formalized problems arising from insufficient efficiency: solving problems of training, customization and adaptation to the problem area; processing fuzzy and multi-criteria source information sets; data interpretation and expert knowledge (experience) accumulation. These limitations in IIS systems are addressed by using CBR [6, 7].

Methods for finding solutions in IIS systems based on precedents are an approach based on the use of analogies with previously solved problems to find and adapt solutions to new situations. Similar methods include the steps that form a CBR cycle:

- capturing precedents from the precedent library (PL);
- indexing (organizing precedents to search for similar cases);
- search for the most suitable precedents for a new task.
- adaptation (modification of the found precedent to suit the current task);
- evaluation and implementation (checking the adapted solution for its suitability and, if necessary, implementation).

CBR advantages: adaptability; working with incomplete information; versatility; learning ability. Use cases can be represented in ways including textual description, diagrams, tables, prototypes and use cases, modeling via UML. Each of these methods can be effective depending on the context and goals of the project.

From the analysis it follows that the task of ensuring the efficient operation of ship CTS is still far from a final solution. The problem of creating IIS systems used to solve complex, difficult-to-formalize problems for assessing and predicting the TC of complex ship systems of critical use remains relevant [10, 11].

Thus, taking into account the specifics and existing problems in ensuring the efficiency of operation of ship CTS, an information approach to the procedures for creating models, methods, algorithms, and software packages for IIS systems that ensure the functioning of complex systems for TC seems promising. The development and research of an IIS system using CBR in order to increase the efficiency learning mechanisms implementation and problem specifics adaptation to the current environment for applied specific applications, as well as to increase decision makers (DM) efficiency based on the results of assessing and predicting ship complex systems TC is important and relevant.

3. Conceptualization and formalization of data and knowledge

The development and research of the IIS system, which uses the method of reasoning based on precedents when assessing failures in CTS, was carried out taking into account: partial and complete failures subsystems operability, components, elements and their mutual connections; precedent knowledge model based on the vehicle dynamics model from a serviceable state to complete failure. The vehicle dynamics model takes into account cause-and-effect relationships and the hierarchical structure of the vehicle system, consisting of: elements (E); components (C); subsystems (S); CTS.

During operation, subsystems, components, elements of the CTS in emergency scenarios, taking into account Harrington's generalized desirability function, can be in one of the following TC [12]: 0 - 0.2 - the level of risk and consequences are minimal, not affecting the operation of the CTS (RMi) ; 0.2 - 0.37 - the risk level is acceptable and the consequences are insignificant, allowing the CTS to be operated without repair (RA); 0.37 - 0.63 - the level of risk is maximum, the consequences are significant, but allow the operation of the CTS when performing repair work (RMa); 0.63 – 1.0 - the risk level is critical, the consequences are catastrophic, preventing the operation of the CTS (RC). Taking into account [13] for the hierarchical structure of the CTS, the transitions of the TC can be represented in the form of a TC matrix (Fig. 1).

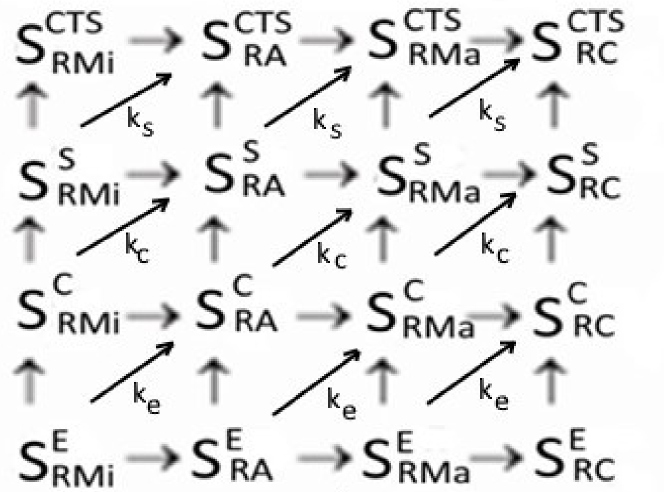

Figure 1: TC CTS matrix

k_e, k_c, k_s – coefficients of weight (significance) of an element, component, subsystem in the corresponding CTS structures.

The description of the problematic situation during the CTS operation consists of the partial or complete functionality loss consequences of subsystems, components, elements and their mutual connections

4. Method of presenting precedents

The method of presenting precedents is implemented as follows. In the proposed CBR cycle (Fig. 2), to support the exchange of knowledge, the initial task generation block receives a set of input diagnostic parameters of the complex system's vehicle and an array of ontologies representing a structured description ship's CTS domain area. As a result, the structure of a new precedent object is generated and content is extracted using the nearest neighbor method based on the results of assessing the degree of similarity (proximity) of the scenario under consideration with the vehicle and taking into account the data in the knowledge base (KB). Based on the implementation of this procedure, a solution object is formed, which can be changed for its target adaptation, taking into account all aspects of the considered scenarios of partial and complete failures CTS subsystems, components, elements using the transformation method [14]. The updated precedent is checked for logical consistency based on the use of predicative productions using the ontological constructing logical inference Hermit method [15]. The solution obtained as a result of the performed actions is exported as a separate object containing recommendations for decision-makers (DM) and metadata. After this, the precedent is saved in the precedent database, which is an integral KB part (Fig. 2).

The sequence of decision-making (Fig. 3) using the proposed CBR cycle, taking into account the operations of processing and structuring data according to precedents within the framework of the application software system operation is carried out next way

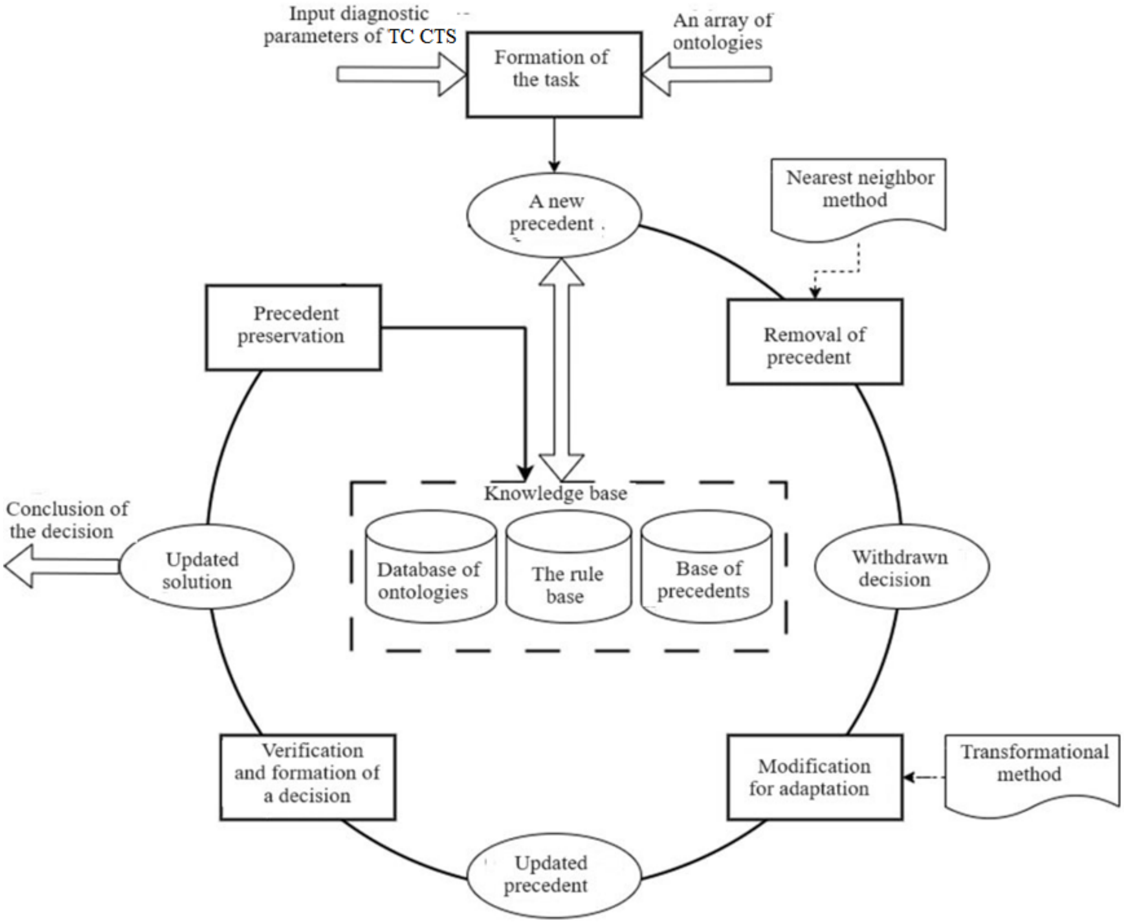

Figure 2: CBR cycle structure

When the software system is launched, the main form of the user interface is initialized, which provides the ability to import input data for generating a task. Then control parameters and options are set to configure the operation of all modules involved in the information processing cycle, in particular the data processing (DPMod), case extraction (PEMod) and adaptation (AMod) modules. Next, a request is made to transfer the generated data arrays to DPMod, in which the procedures for processing the data are carried out step by step (including checking them for consistency and breaking them into fragments), forming a collection for storing precedents, taking into account the presence of metadata (such as a short text description of the precedent purpose, its identifier, date of formation and some statistical indicators). After this, a request is made to obtain a specific use case in PEMod, in which metric estimation actions are performed based on the nearest neighbor's method. The result is sent to DPMod as a collection based on an associative array. DPMod, after checking and verifying the results, sends the generated collection to AMod for the purpose of carrying out the adaptation procedure. For this purpose, a transformation method is used based on comparison of a precedent with a rules set that take into account the correspondence logical products. As a result, the updated collection of precedents is sent to DPMod to form a list of final decisions and validate them. To display the results in the form of text records and graphical representation, the resulting solution in serialized form (in json format) is sent to the main form of the interface.

This is necessary for the subsequent initialization of the process of sending precedent data, storing it in the knowledge base and issuing an information notification to the user about transaction results.

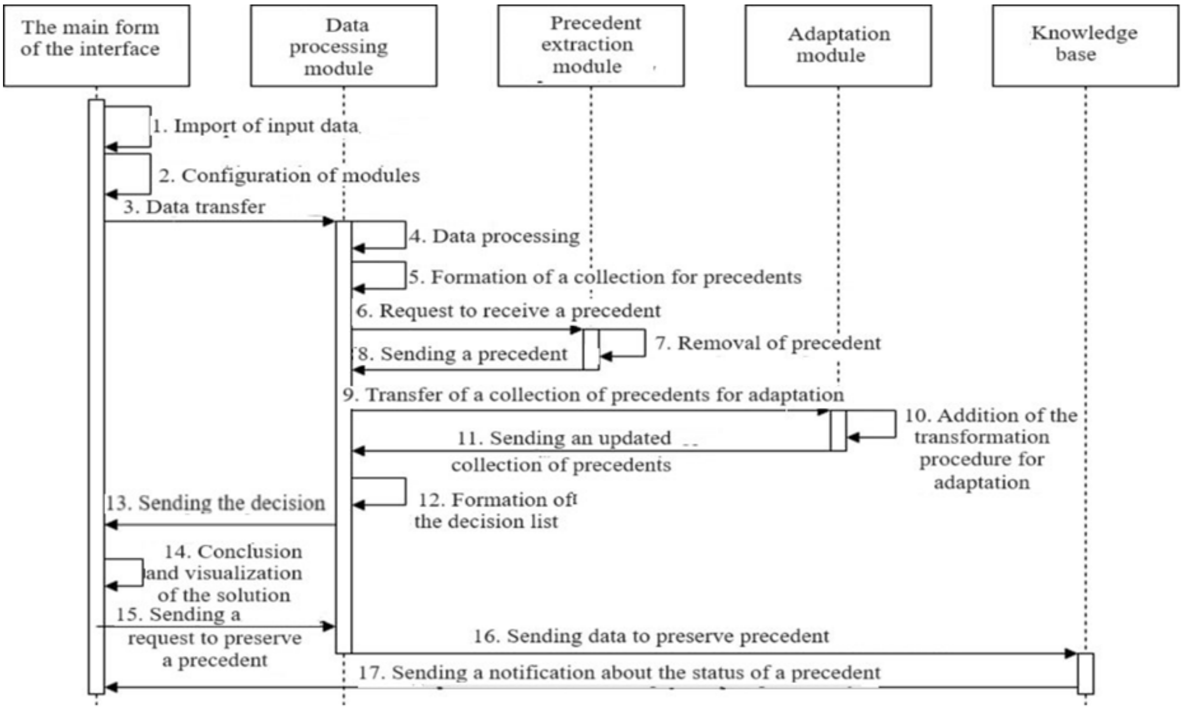

Figure 3: Decision sequence diagram

To represent precedents, a simple parametric representation is sufficient, i.e. presentation of a precedent in the parameters set form and some specific values for solution (decision maker's diagnosis and recommendations)

$$CASE = (R, P, D, W^{f}_{\vartheta_{S(C,E)_{n(m)}}}, W^{f}_{\omega_{I_{S(C)_{a(z)}}}},\ RE, SR,\ DR), \tag{1}$$

where $R, P, D,$ - parameters describing the use case;

$R\left\{R_{S(C,E)_{n(m)}}, R_{I_{S(C)_{a(z)}}}\right\}$ - set of assessments of subsystem's failures risk components, elements and their interrelations CTS and DM recommendations;

$P\left\{P_{S(C,E)_{n(m)}}, P_{I_{S(C)_{a(z)}}}\right\}$ - set of assessments of subsystems failure probabilities, components, elements and their interrelations CTS and DM recommendations;

$D\left\{D_{S(C,E)_{n(m)}}, D_{I_{S(C)_{a(z)}}}\right\}$ - - set of damage assessments from failures of subsystems, components, elements and their interrelations CTS and DM recommendations;

$W^{f}_{\upsilon_{S(C,E_{n(m)}}}$ - performance assessment (partial or complete) of subsystems, components, elements and DM recommendations;

$W^{f}_{\omega_{I_{S(C)_{a(z)}}}}$ - performance assessing (partial or complete) of intercomponent (intersystem) connections and DM recommendations:

RE – set of refined specific estimates TC parameters of subsystems, components, elements and their CTS mutual connections, decisions made ($re_1, ..re_N \in RE$);

SR – saved values of many updated estimates technical system parameters of subsystems, components, elements and their CTS mutual connections, decisions made;

DR - diagnosis and DM recommendations [13];

S, C, E – subsystem, component, element of CTS;
IS, IC – intersystem, intercomponent communication;
n, m – number, hierarchical level in CTS;
a – intercomponent number;
z – interconnection number.

$$R_{s(c,e)_{n(m)}} = \{r_{s(c,e)_{n(m)}} \mid s(c,e) = \overline{1, S(C,E)}, n_{s(c,e)} = \overline{1, N_{S(C,E)}}, m_{s(c)} = \overline{1, M_{S(C)}}\}, \quad (2)$$
$$R_{I_{s(c)a(z)}} = \{r_{i_{s(c)a(z)}} \mid i_{s(c)} = \overline{1, I_{S(C)}}, a = \overline{1, A}, z = \overline{1, Z}\},$$

where $r_{s(c,e)_{n(m)}}$ - CTS failures risk of subsystems, components, elements;

$r_{i_{s(c)}{}_{a(z)}}$ - CTS failures risk of intersystem, intercomponent connections;

$n_{s(c,e)}$ - number of the subsystem, component, element in the CTS;

$m_{s(c)}$ - intersystem hierarchical level CTS intersystem, intercomponent number;

$N_{S(C,E)}$ - CTS subsystems, components, elements number;

$M_{S(C)}$ - hierarchical levels CTS intersystem, intercomponent connections number;

A - intercomponent connections number;
Z - intersystem connections number.

$$P_{S(C,E)_{n(m)}} \cdot \lambda(t)_{S(C,E)_{n(m)}} = \frac{\alpha_{S(C,E)_{n(m)}} \cdot \exp(-\alpha_{S(C,E)_{n(m)}} \cdot T_{S(C,E)_{n(m)}})}{\exp(-\alpha_{S(C,E)_{n(m)}} \cdot T_{S(C,E)_{n(m)}})} = \alpha_{S(C,E)_{n(m)}}, \quad (3)$$
$$P_{I_{S(C)a(z)}} \cdot \lambda_{I_{S(C)a(z)}}(t) = \frac{\alpha_{I_{S(C)a(z)}} \cdot \exp(-\alpha_{I_{S(C)a(z)}} \cdot T_{I_{S(C)a(z)}})}{\exp(-\alpha_{I_{S(C)a(z)}} \cdot T_{I_{S(C)a(z)}})} = \alpha_{I_{S(C)a(z)}}$$

where λ - failure rate;
α – distribution parameter, $\alpha \approx 1/\widehat{T}_o$, $\widehat{T}_o$ – mean time to failure estimation.

Quantifying damage from failure $n(m,e)$ subsystem, component, element to determine the risk of failure:

$$D_{S(C,E)_{n(m)}} = \{d_{s(c,e)_{n(m)}} \mid s(c,e) = \overline{1, S(C,E)}, n = \overline{1, N}, m = \overline{1, M}\}, \quad (4)$$

where $d_{s(c)_{n(m)}}$ - damage from failure of a component subsystem, CTS element.

Quantifying damage from failure $a(z)$ intersystem (intercomponent) communication to determine the risk of failure:

$$D_{I_{S(C)a(z)}} = \{d_{I_{s(c)a(z)}} \mid i_{s(c)} = \overline{1, I_{S(C)}}, a = \overline{1, A}, z = \overline{1, Z}\}, \quad (5)$$

where $d_{i_{s(c)}{}_{a(z)}}$ - damage from failure of intersystem (intercomponent) communication.

Performance of subsystems, components, elements at different degrees of its loss:

$$W^f_{\upsilon_{S(C,E)_{n(m)}}} = \{W_f^{<n_{s(c)}, m_{s(c)}>} \mid f = \overline{0,1}; n_{s(c,e)} = \overline{1, N_{S(C,E)}}; m_{s(c)} = \overline{1, M_{S(C)}}\} \quad (6)$$

Performance of intercomponent (intersystem) communication at different degrees of its loss:

$$W^{f}_{\omega_{I_{S(C)_{a(z)}}}} = \{W_f^{<a,z>} \mid f = \overline{0,1}; a = \overline{1,A}; z = \overline{1,Z};\} \quad (7)$$

There are known methods for extracting precedents and their modifications. The most common methods are: determination of the nearest neighbor (NN – Nearest Neighbor) [16]; extracting precedents based on decision trees [17]; extracting precedents based on knowledge [16, 17]; extracting precedents taking into account their applicability [18]. The listed methods use different metrics.

Among the main metrics, the method of determining the nearest neighbor was used, which makes it possible to easily calculate the degree of similarity of the current problem situation and precedents with BP. The nearest neighbor method uses a simple coordinate-by-coordinate comparison of the current situation with a precedent, where each parameter for describing precedents is considered as one of the coordinates. The DCT distance between the point corresponding to the current situation and the point corresponding to the precedent is determined. The effectiveness of the nearest neighbor method largely depends on the choice of metric. If the case C and the current problem situation T are given in an n-dimensional property space, then the degree of similarity or proximity S(C,T) of the case C and the current situation T can be determined using one of the metrics that determine the distance between two points x_i^C and x_i^T, in particular, the Euclidean distance:

$$D_{CT} = \sqrt{\sum_{i=1}^{n} (x_i^C - x_i^T)^2} \quad (8)$$

To determine the similarity degree (SIM) value, the maximum distance is calculated Dmax in the selected metric using the boundaries of parameter ranges to describe use cases. After this, the value of the degree of similarity is determined using the boundary of the parameter ranges to describe the initial and final precedents, i = 1, ..., n. The degree of similarity value can be calculated as follows:

$$SIM = 1 - D_{CT} / D_{max} \quad (9)$$

5. IIS system with CBR implementation

The implementation of the IIS system with CBR (Fig. 4) connects the vehicle diagnostic results with a database (DB), KB, expert system containing calculated, experimental, and data obtained by experts during CTS operation.

IIS contains: interface module; KB with a library of precedents and a DB; request formalization module; module of recommendations for ensuring the effectiveness of the CTS; libraries of structural diagrams of ship CTS; module for formalizing expert assessments; knowledge formalization module. The implementation of the developed strategy in the IIS system is ensured by targeted actions in accordance with decision support to search for failures of subsystems, components, elements and their mutual connections based on vehicle established assessments.

The KB model is production-based; in terms of implementing its software functionality, it is object-oriented. The developed knowledge base is represented by rules obtained on the basis of data mining (multi-level hierarchical structure of the knowledge base tree), expert assessments, results of applying diagnostic models for the CTS TC, and functioning in accordance with the developed decision-making sequence (Fig. 3) and taking into account the CTS TC matrix (Fig. 1). The entire list of data and expert assessments are received from the database upon request to the KB. As a result, at the output of the knowledge base, estimates of the TC of subsystems, components, elements and their mutual connections are formed. TC assessments go to the module of recommendations for ensuring the effective operation of the CTS and then to the decision maker for managing the TC of a complex system. The database contains: CTS DB; DB of failure risk criteria; DB TC CTS; DB of degradation processes; DB of measures to reduce the risk of CTS failures.

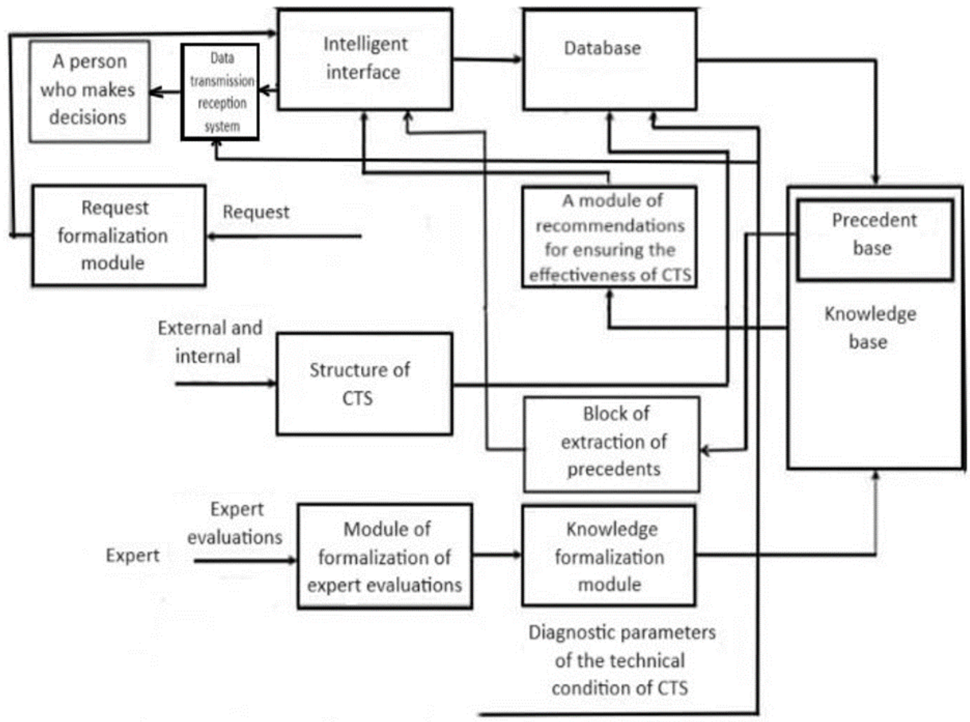

Figure 4: Block diagram of the IIS system with CBR assessment, forecasting and control of the CTS vehicle

The library of precedents consists of rule base (RB) of incidents, RB of emergency situations. Diagnosis data of a problem situation (complete or partial failures of equipment and their mutual connections between CTS and operability), obtained separately from the IIS, enters the IIS with CBR assessment, forecasting of the CTS TC. Taking into account the diagnostic data, the knowledge base and RB receive an established assessment of the technical system of a complex system, and recommendations are formed for decision-making by the D. Based on the assessment results, the TC of a complex system is predicted. The output may include a list of completed actions, additional comments, and links to other use cases.

The hierarchical structure of the key program logic of fragments of data processing modules according to precedents and solutions is shown in Fig. 5. To form a level of abstraction and ensure polymorphism when implementing the behavior of objects that implement different stages of data processing, three interfaces are implemented.

IData - forming a path to the location input data set, initializing data structures and collections, converting data to a single normalized form with checking for missing rows, setting up a structure. IOntology - storage of structures of attributes, classes and relationships, formation of an assembly of the ontology structure with its verification and validation. IPrecedent - operating with the properties situations scenarios, problems and solutions, as well as the formation of processes for assembling a precedent, saving and serializing it, checking for logical consistency. Each has a different implementation of the logger() method to provide processes for logging the intermediate results of computational operations over time. The DataLoader class implements the IData interface, performing the task of loading data into the system and the operations of forming precedents collections, checking data integrity, performing the necessary transformations, filtering and aggregation, as well as setting the structure and issuing notifications based on the results of completed actions in the status bar.

The OntologyMaker class implements the IOntology interface, operating a private collection dynamic array ontology for aggregating individual ontology elements and is intended to form the logical basis of the system structure when constructing each individual use case, providing visualization of the ontology in a graph-oriented form.

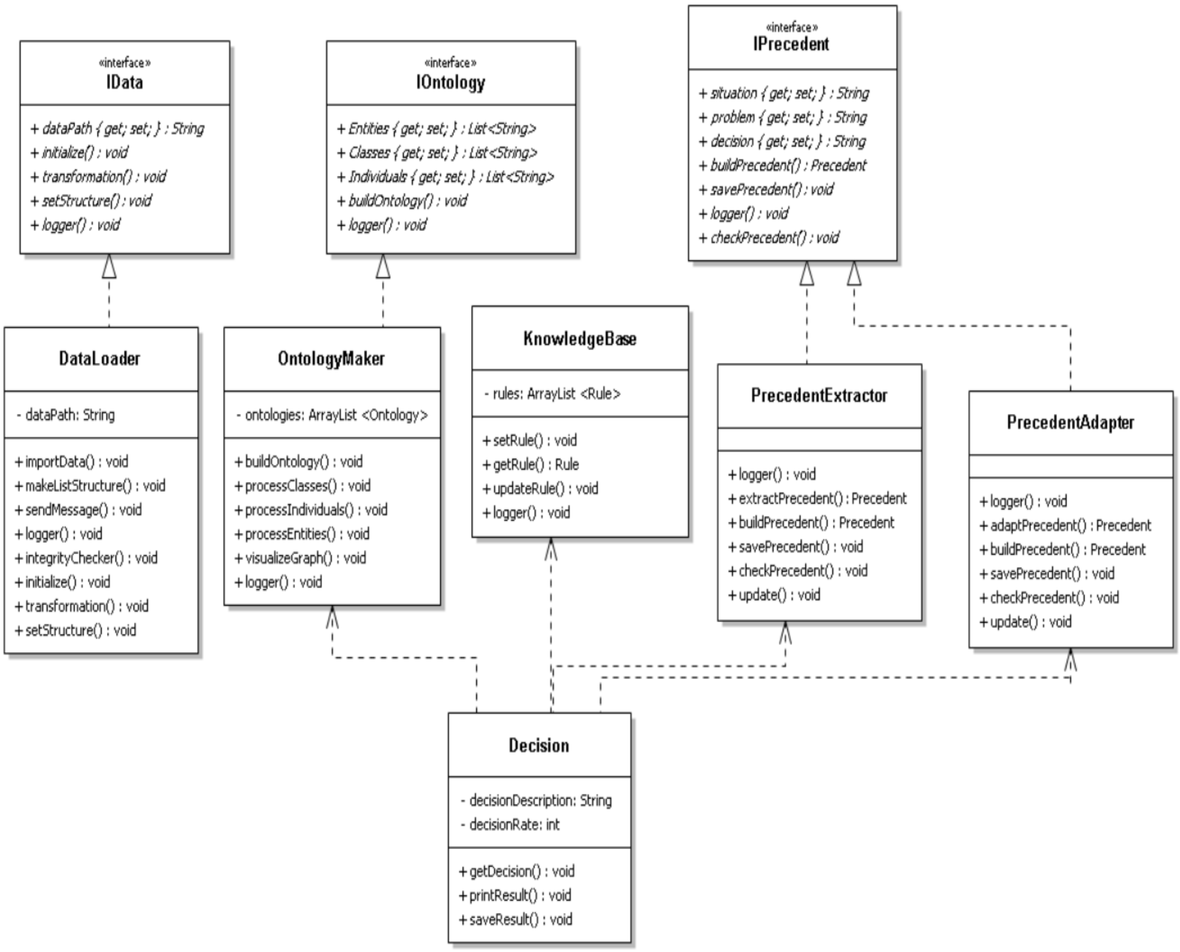

Figure 5: Key program logic hierarchical structure of data processing modules fragments by precedents and solutions

The PrecedentExtractor and PrecedentAdapter classes implement the IPrecedent interface, overriding methods for managing precedent data for use in the extraction and adaptation procedures performed in the extractPrecedent() and adaptPrecedent() methods, respectively, the result is a Precedent object. The KnowledgeBase class is separately implemented to provide CRUD operations for managing rules. Based on the use of these class instances a Decision object is formed, the state of which is described using the private properties decisionDescription and decisionRate, and the behavior is expressed using public methods for generating a decision object, outputting it, and saving the results.

For the software implementation of the IIS system with a CBR cycle, the Visual Studio development environment, the .NET 4.7 framework, the technology for creating WinForms graphical interfaces, as well as functional libraries for supporting Hermit ontologies and working with JSON were used [19]. The interface of the main form of the software system with a tab for managing the process of creating precedents within the framework of the proposed CBR cycle for assessing, forecasting and managing a vehicle, using the example of a ship's CTS [20] is shown in Fig. 6.

A main menu is provided for navigation between the processes of connecting data sources and knowledge base (File item), data management and processing (Data processing item), selection of computational operations and their implementation (Operations item), configuration of system modules and settings for its operation (Settings item), issuing reference information on the operation of the system (Help item). The interface provides four tabs for managing use cases, ontologies, rules and solutions. The precedents tab contains two panels.

The left panel contains a component for a structure hierarchical representation of the considered ship CTS and their subsystems, components and elements in the form of a tree of nodes. A field is provided to display a brief set of metadata for the precedent selected in the tree. The form contains buttons for opening a window for a detailed description of parameters, constructing a descriptive summary cross-table for all values of ship CTS and their equipment, as well as buttons for initializing the processes of opening forms for extracting, adapting and validating (checking) compiled precedents. To launch the procedure for saving data based on the use case and serializing objects, use the Save button.

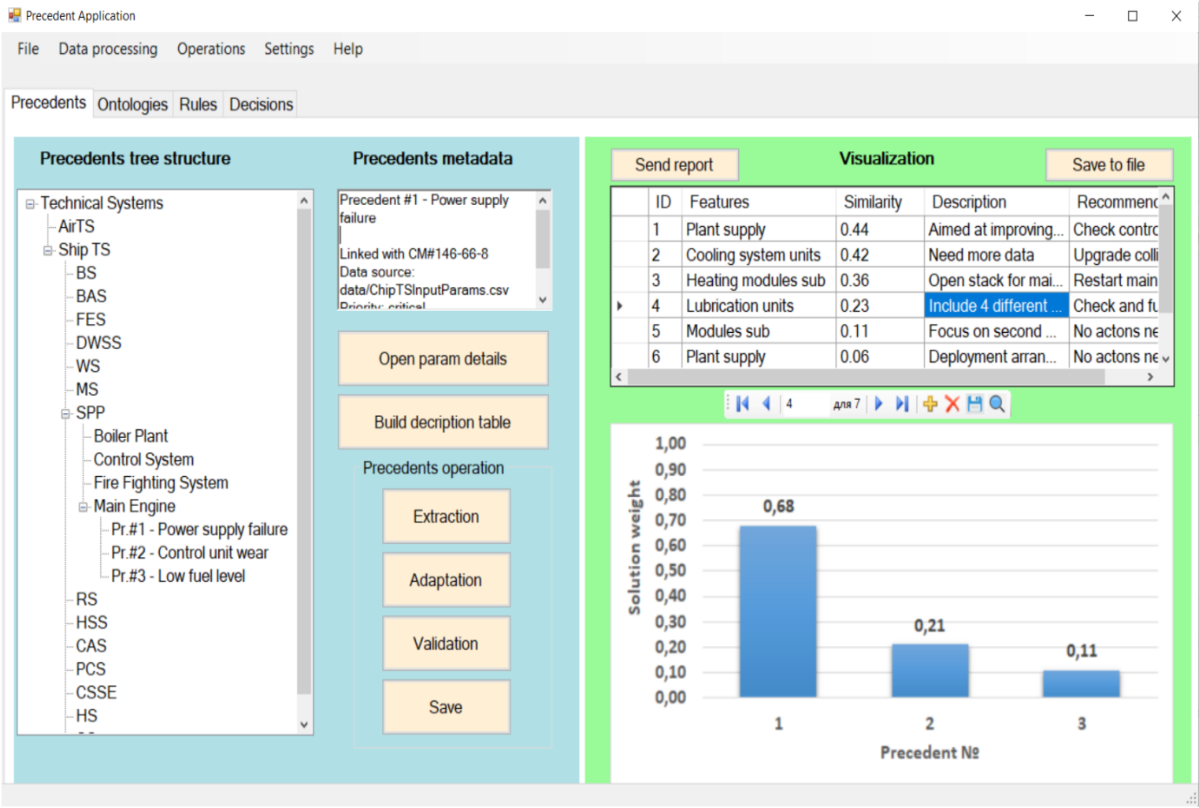

Figure 6: Software system main form interface with a tab for managing creating precedents process

The right panel of the form is intended to display a table of generated precedents with an assessment of their degree of similarity, description and a short set of standard recommendations. For ease of management, a component for quick table navigation with support for CRUD operations and search is provided. Below is a visualization component of the highest priority options for adapting precedents to a specific operating scenario for the CTS in question after all analytical procedures have been completed. The buttons for sending a report and saving to a file provide the ability to locally save visualization results in pdf and csv formats, respectively.

6. Experimental system research

The ship power study subsystems risk of failures assessment results formed on the basis of the compiled precedents, are shown in Fig. 7.

The results of functional elements and functional connections failure risk prediction results, for example, for the ship power plant main engine subsystem, can be seen in the block of the failure risk prediction view interface when navigating to the predictions web page (Fig. 8).

The disadvantage of the method of precedents with a CBR cycle is an increase in the time of searching for the nearest precedents. Therefore, a comparative analysis of the search time for the nearest precedent was conducted depending on precedent database size based on data caching when the data structure was initialized in collection form, which is based on an associative array.

The graph reflecting the time of determining the TC of a complex system from the number of precedents is shown in Fig. 9. Time spent on finding the nearest precedent with 10,000 precedents in the BZ was about 370 ms. The first closest precedent out of 5000 precedents was obtained in about 50 ms.

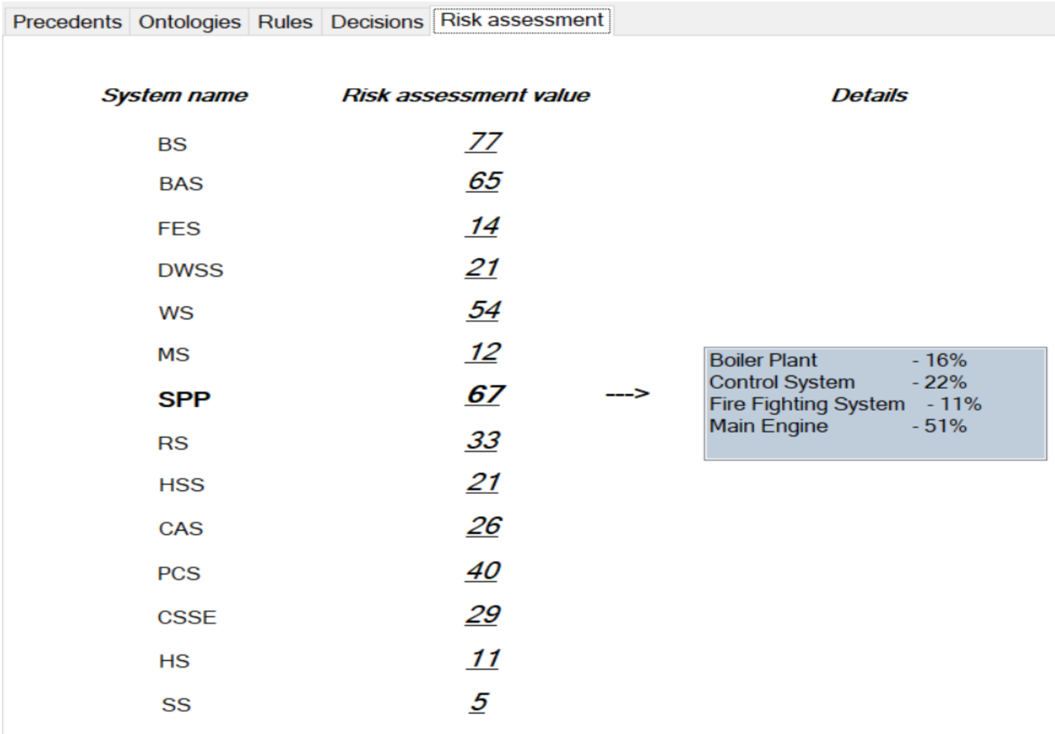

Figure 7: Interface of the risk assessment form for the analyzed subsystems

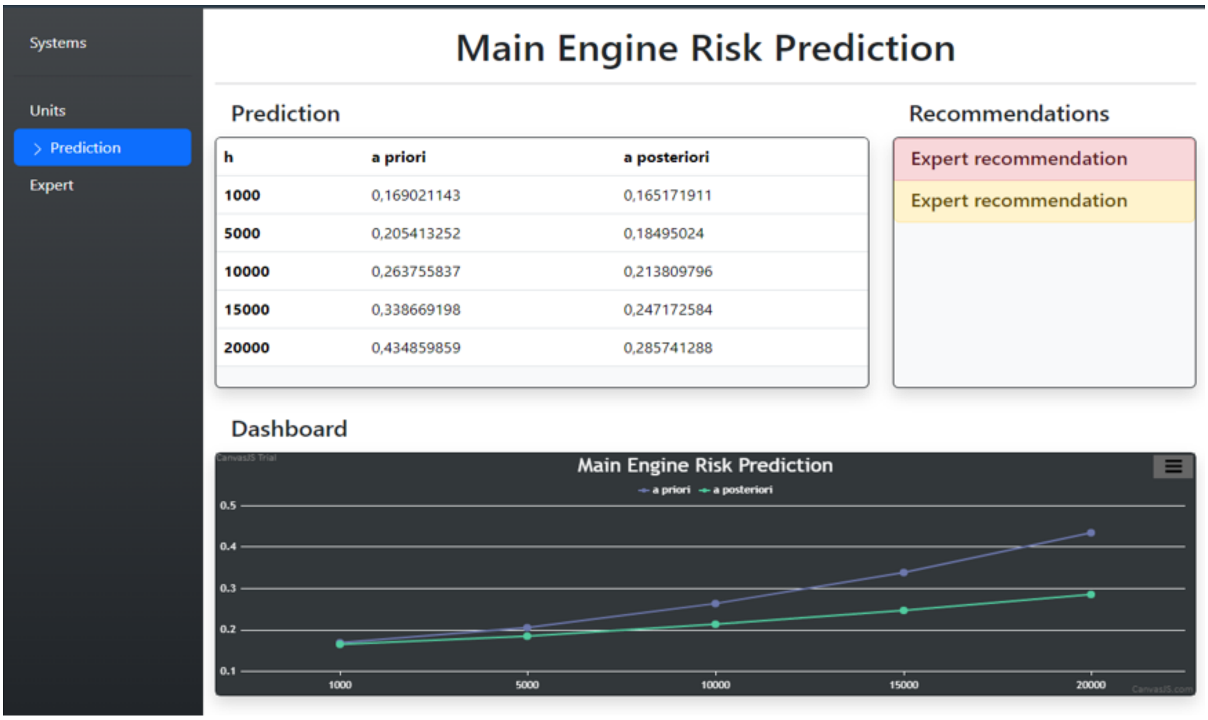

Figure 8: Main engine subsystem failure risk prediction interface unit

With an increase in the precedents number in the precedents library determining CTS TC time increases, but it does not significantly affect the overall time spent on evaluating the TC of the researched SPP subsystems.

Despite such a shortcoming, studies have shown the possibility of applying the method of reasoning based on CBR precedents, and it is expedient to use it for decision-making in real operating conditions. The developed IIS has high performance.

In order to estimate the time spent on the KB formation within the implementation framework of the proposed method by IIS means a comparison of the execution time of computational processes at the start of the system in three modes (Fig. 10): single-threaded (1 flow), two-threaded (2 flow) and four-threaded (4 flow) was carried out.

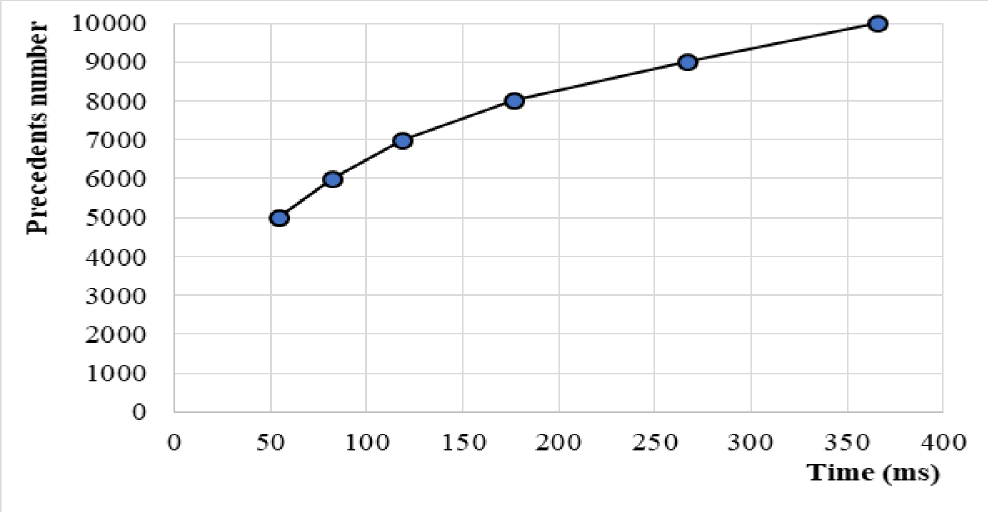

Figure 9: The time to determine the TC of a complex system is based on the number of precedents

It should be noted the general exponential nature of the time dependence of computational processes estimations execution forecasting SPP TC on the KB precedents number.

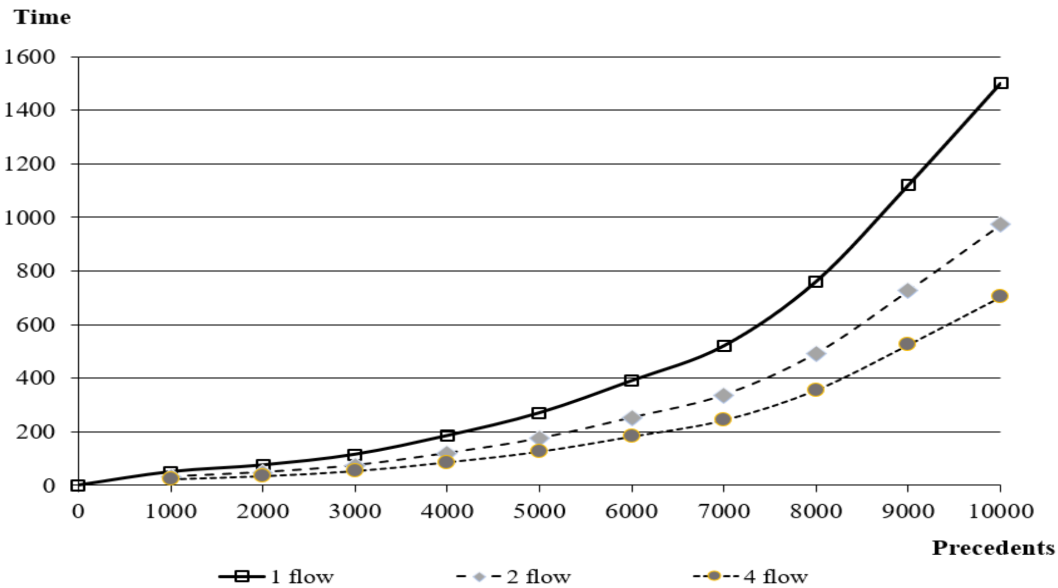

Figure 10: Execution time of computational processes depending on the number of generated precedents

Within distributed mode it becomes possible to reduce the time spent by up to 28% when using two mutually isolated data streams, and up to 42% in the case of dividing the computing load into four separate streams.

7. Conclusion

An IIS with CBR is proposed, designed for effective assessment and prediction of the TC of complex systems for critical applications by ensuring IIS speed. The effective functioning of IIS with CBR is based on the use of case-based reasoning. IIS with CBR consists of: interface module; KB with a library of precedents and a DB; request formalization module; module of recommendations for ensuring CTS effectiveness; CTS block diagram libraries; modules for formalizing expert assessments and formalizing knowledge. IIS experimental studies for assessing and predicting the technical characteristics of complex systems showed that the time required to find the nearest precedent with 10,000 precedents in the knowledge base was about 370 ms. The decision-making sequence, using the proposed IIS system with a CBR cycle, taking into account the operations of processing and structuring data according to precedents within the framework of the functioning of the developed application software system, has high performance, ensures work with incomplete information, versatility and learning ability when supporting decision-making. According to the distributed computing mode, it becomes possible to reduce time costs by up to 28% when using two data streams isolated from each other, and up to 42% when the computing load is divided into four separate data streams.

When operating the proposed IIS, partial and complete failures of the operability of subsystems, components, elements and their mutual connections in CTS are taken into account. Increased operational efficiency of the CTS is ensured by CTS speed evaluation and forecasting, as well as by the actions of decision makers aimed at making decisions on pre-failure maintenance complex systems equipment at early development failure stages.

References

[1] Benin D.M. Support systems accept solutions. Triada, 2022.

[2] E.A. Gavrilyuk, Information Decision-Support System on the Basis of the Method of Diagnostics and Control of the Technical State of Industrial Equipment Engineering, Computer Science Dynamics of Systems, Mechanisms and Machines (Dynamics) (2018). DOI: 10.1109/Dynamics.2018.8601472

[3] V. Vychuzhanin, N. Rudnichenko, Metody informatsionnykh tekhnologiy v diagnostike sostoyaniya slozhnykh tekhnicheskikh sistem, monografiya, 2019.

[4] P. Zhang, Z. GaoCao, L. Dong, Marine Systems and Equipment Prognostics and Health Management. Systematic Review from Health Condition Monitoring to Maintenance Strategy. Machines 10 (2022) 72. DOI:10.3390/machines10020072

[5] R. Garcia, M. Parades-Valverde, Exploring Intelligent Decision Support Systems, Current State and New Trends, Springer International Publishing AG, (2018).

[6] V. Vychuzhanin, N. Rudnichenko, Z. Sagova, M. Smieszek, V. V. Cherniavskyi, A. I. Golovan, M. V. Volodarets, Analysis and structuring diagnostic large volume data of technical condition of complex equipment in transport. IOP Conference Series: Materials Science and Engineering, Volume 776, 24th Slovak-Polish International Scientific Conference on Machine Modelling and Simulations - MMS 2019, 3-6 September 2019, Liptovský Ján, Slovakia, 2019 pp.1-11. DOI:10.1088/1757-899X/776/1/012049.

[7] C. Shaoli, Yi. Jianjun, J. Hui, Z. Xiaomin, Ontology and CBR based automated decision-making method for the disassembly of mechanical products. Advanced Engineering Informatics 30 3 (2016) 564-584.

[8] M. Siepermann, Risk Cost Accounting and Bottom Price Calculation – A Risk Management Information System, in International Conference on the Economics of GrIIS, Clouds, Systems, and Services, September, Springer, Cham, 2017, pp. 43-55.

[9] R. Majid, S. A. Mir, Advances in Statistical Forecasting Methods: An Overview. Economic Affairs 63 4 (2018) 815-831.

[10] F. Lan, Y. Jiang, Performance Prediction Method of Prognostics and Health Management of Marine Diesel Engine, in Proceedings of the 2020 3rd International Conference on Applied Mathematics, Modeling and Simulation, Shanghai, China, 20–21 September 2020.

[11] V. Vychuzhanin, N. Rudnichenko Complex Technical System Condition Diagnostics and Prediction Computerization. CMIS-2020 Computer Modeling and Intelligent Systems. Ceur-ws.org. 2608.2020., 2020, pp.1-15.

[12] E.C. Harrington, The desirability Function. Industrial Quality Control, 21 10 (1965) 494–498.

[13] A. F. Berman, Method of synthesis and analysis of fault trees based on the concepts of mechanism and kinetics of events. Problems of risk analysis, 15 3 (2018) 62-77.

[14] De. Sumana, B. Chakraborty, A Novel CBR-Decision Tree Based Intelligent Car Fault Diagnosis System (CFDS). International Journal of Recent Technology and Engineering 8 (2019) 2180-2193. DOI:10.35940/ijrte.B2390.078219.

[15] D. Glukhikh, I. Shchinnikov, I. Glukhikh, Using hybrid-CBR for intelligence monitoring and decision-making systems on SMART grid. Intelligent Decision Technologies. 16 (2022) 1-8. DOI:10.3233/IDT-210239.
[16] G. Luger, Artificial intelligence: structures and strategies for complex problem solving, (2009).
[17] I.D. Watson, F. Marir, Case-based reasoning: A review. The Knowledge Engineering Review, 9, 4 (1994) 355-381.
[18] B. Smyth, M. Keane, Retrieving adaptable cases. The role of adaptation knowledge in case retrieval. Springer Berlin 837 (1994) 209-220.
[19] Y. Shunkun, B. Chong, Li. Xing, T. Lin, T. Dongxiao, Optimized fault diagnosis based on FMEA-style CBR and BN for embedded software system. The International Journal of Advanced Manufacturing Technology. 94 (2018). DOI:10.1007/s00170-017-0110-y.
[20] V. Vychuzhanin, N. Shibaeva, A. Vychuzhanin, N. Rudnichenko, Intellectualization Method and Model of Complex Technical System's Failures Risk Estimation and Prediction. Ceur-ws.org, 3392, CMIS-2023. pp.130 – 140, https://ceur-ws.org/Vol-3392.