Hybridization of fuzzy time series and fuzzy ontologies in the diagnosis of complex technical systems

N G Yarushkina¹, V S Moshkin¹, I A Andreev¹ and G I Ishmuratova¹

¹Ulyanovsk State Technical University, Severny Venetz street, 32, Ulyanovsk, Russia, 432027

e-mail: PostForVadim@yandex.ru

Abstract. The article provides a formal description of fuzzy ontologies and features of the representation of elements of fuzzy axioms in FuzzyOWL notation. An ontological model for assessing the state of helicopter units has been developed. According to the proposed approach, the summarizing of the state of a complex technical system is carried out by means of an inference based on a fuzzy ontology. As part of this work, experiments were conducted to search for anomalous situations and search for possible faulty helicopter units using the developed approach to the integration of fuzzy time series and fuzzy ontology. The proposed approach of hybridization of fuzzy time series and fuzzy ontologies made it possible to reliably recognize anomalous situations with a certain degree of truth, and to find possible faulty aggregates corresponding to each anomalous situation.

1. Introduction
The uncertainty of data and information incompleteness is an inalienable part of any complex technical system, in which the functioning quality of processes depends on a person. In the analysis, modeling, and design of such systems, a large distribution was obtained by expert systems that use experience and knowledge of the expert.

Expert assessments represent the qualitative aspect of the system element being evaluated and are presented in linguistic form.

Currently, the inference methodology of expert assessments based on the subject ontologies that play the role of a knowledge base in decision support systems (DSS) is used in various subject areas, including in the field of situational control in the energy sector [1], designing complex diagnostic systems [2], etc. Also, ontologies have been used as a knowledge base of intelligent risk prevention systems in the context of heterogeneous information for the complex technical systems critical infrastructure design phase [3].

Despite the application breadth, the classical languages of ontology and semantic networks, which are usually used to summarize and characterize the features of a subject domain, cannot be used to solve uncertainties and inaccuracies in the knowledge inherent in most real world applications in this area.

Fuzzy set theory, as well as fuzzy logic, is formalism suitable for processing incomplete knowledge, therefore ontologies based on such logic are adequate means of formalization.
One of the most effective solutions for representing a knowledge base in the context of accounting for fuzziness and uncertainty in human reasoning and evaluations in the DSS is a representation in the form of fuzzy ontologies. For example, fuzzy ontologies are used in such systems as disease diagnosis systems [4-6], fuzzy search engines [7, 8], knowledge systems based on group decision making about the importance of data [9], etc. In most cases, such systems operate with facts objects or terms that are described in natural language and contain the features of the considered domain [10, 11].

Fuzzy time series (FTS) is a way to obtain expert assessments that satisfy the conditions for completeness, consistency, and adequacy [12].

One of the main areas of application for FTS is process diagnostics. Diagnosis is the process by which a search for problems in the system occurs: defects, anomalies, faults, or lack thereof. When solving problems of diagnostics of complex technical systems, the state of which is determined by the data set in the form of FTS, it is advisable to apply methods for comparing the dynamics of processes with the expected or required dynamics.

Therefore an urgent task requiring a systemic solution is the interpretation of the results of the analysis in the form of expert assessments. To summarize the results obtained in the analysis of FTS, a system of rules is usually applied, which are stored in the knowledge base of the expert system. The knowledge base for solving this problem is ontologies and similar graph forms of knowledge representation and storage, which allow to take into account the semantic features of the object of the specified subject area, and not only their inference [13, 14].

Interpretation of the extracted comparisons in the form of expert assessments, the values of which are presented in the form of semantic units that correspond to certain classes of fuzzy ontology, taking into account the deviations between the current and the required FTS, can be obtained by solving the problem of integrating FTS and fuzzy ontology. Thus, the purpose of this work is the development of algorithms and models for the integration of fuzzy ontologies and FTS in the tasks of diagnosing complex technical systems.

2. Fuzzy time series and fuzzy ontology model
The models and algorithms for analyzing and forecasting the FTS are described in detail in [15, 16]. At present, the basic notation of the fuzzy ontology representation is the FuzzyOWL standard [17-20]. Formally FuzzyOWL-ontology is:

\[ I = (If, Cf, Pf, Af, Df, Qf, Lf, Modf), \]

where \( If \) is an Individual that simply represents an individual of the vocabulary; \( Cf \) is a Concept that represents a fuzzy concept of the vocabulary:

\[ C_f = \{C_f^A, C_f^C\}, \]

where \( C_f^A \) are Abstract Concepts, \( C_f^C \) - Concrete Concepts; \( Pf \) is Property that represents a fuzzy role:

\[ P_f = \{P_f^A, P_f^C\}, \]

where \( P_f^A \) are Object Properties, \( P_f^C \) are Datatype Properties; \( Df \) is Axiom that represents the axioms:

\[ D_f = \{A_f^{ABox}, A_f^{TBox}, A_f^{RBox}\}, \]

where \( A_f^{ABox} \) is the ABox that contains role assertions between individuals and membership assertions, \( A_f^{TBox} \) is the TBox that contains assertions about concepts such as subsumption and equivalence, \( A_f^{RBox} \) is the RBox that contains assertions about roles and role hierarchies. Some of the axioms are subclasses of FuzzyAxiom, which indicates that the axiom is not either true or false, but that it is true to some extent.

\( Of \) is Degree that represents a degree which can be added to an instance of FuzzyAxiom:

\[ Of = \{LD_f, MD_f, ND_f, Var_f\}, \]

where \( LD_f \) are Linguistic Degrees, \( MD_f \) are Modifier Degrees, \( ND_f \) are Numeric Degrees, \( Var_f \) are Variables.
Lf is Fuzzy Logic represents different families of fuzzy operators that can be used to give different semantics to the logic.

\[ L_f = \{ L_{f}^{\text{Luk}}, L_{f}^{\text{Zad}}, L_{f}^{\text{Goed}}, L_{f}^{\text{Prod}} \}, \]

where \( L_{f}^{\text{Luk}} \) is the fuzzy operators logic of Lukasiewicz, \( L_{f}^{\text{Zad}} \) is the fuzzy operators logic of Zadeh, \( L_{f}^{\text{Goed}} \) is the fuzzy operators logic of Goedel, \( L_{f}^{\text{Prod}} \) is the fuzzy operators of produc logic.

Modf is Fuzzy Modifier that represents a fuzzy modifier, which can be used to modify the membership function of a fuzzy concept or a fuzzy role. Current subclasses are Linear Fuzzy Modifier and Triangular Fuzzy Modifier.

Table 1 shows the elements of fuzzy axioms FuzzyOWL, as well as their possible representation.

<table>
<thead>
<tr>
<th>№</th>
<th>Element</th>
<th>Possible values</th>
<th>Representation in FuzzyOWL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LDf – Linguistic Degrees</td>
<td>«high», «above average», «low»</td>
<td>&lt;AnnotationAssertion&gt; &lt;AnnotationProperty IRI=&quot;#fuzzyLabel&quot;/&gt; &lt;IRI&gt;#HighLoad&lt;/IRI&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&lt;Literal datatypeIRI=&quot;&amp;rdf;PlainLiteral&quot;&gt;fuzzyOwl2fuzzyType=&quot;datatype&quot;; Datatype type=&quot;rightshoulder&quot; a=&quot;15.0&quot;; b=&quot;30.0&quot;;/fuzzyOwl2&lt;/Literal&gt;</td>
</tr>
<tr>
<td>2</td>
<td>MDf – Modifier Degrees</td>
<td>«very», «not very»</td>
<td>type=&quot;modified&quot; modifier=&quot;very&quot;</td>
</tr>
<tr>
<td>3</td>
<td>NDf – Numeric Degrees</td>
<td>0≤ND≤1</td>
<td>Degree Value=0.6</td>
</tr>
<tr>
<td>4</td>
<td>Varf – Variables</td>
<td>a, b, c, k1, k2</td>
<td>b=&quot;30.0&quot;;</td>
</tr>
<tr>
<td>5</td>
<td>Lf – Fuzzy Logic</td>
<td>Zadeh, Lukasiewicz</td>
<td>hasSemantics=&quot;Zadeh&quot;</td>
</tr>
<tr>
<td>6</td>
<td>Modf – Fuzzy Modifier</td>
<td>Linear, Triangular</td>
<td>&lt;Datatype type=&quot;triangular&quot; a=&quot;32.0&quot; b=&quot;41.0&quot; c=&quot;50.0&quot; /&gt;</td>
</tr>
</tbody>
</table>

3. Subject Area
Consider the use of the integration approach of FTS and fuzzy ontologies in solving the problem of diagnosing the state of a helicopter. Diagnostics of a helicopter consists in checking its units in order to establish their exploitation and the possibility of using the helicopter.

The result of the diagnosis will be assessment values of physical quantities key indicators. The main goal is to assess the danger of values. To solve this problem, it is necessary to construct models of the behavior of the selected nodes and make conclusions about the health of the nodes by using the models. Models are built at expert base of assessment about the conduct of a particular component.

Table 2 show the parameters of the membership functions used for construct the FTS (Table 2).

Thus 5 fuzzy labels are defined for each physical quantity. The task of analyzing technical time series is reduced to the task of searching for anomalous situations in TS of main gearbox and engine propulsion system physical quantities indicators [21, 22]. The analysis is a sequence of the following steps:

1. Formation of FTS on the basis of the received information on the values of key physical quantities after the end of helicopter flight.
2. Search known abnormal situations in the resulting FTS.
3. Determination of the correct operation of the nodes. Work is incorrect if at least one abnormal situation.

The fuzzy ontology was developed for experiments. The developed FuzzyOWL ontology has a hierarchical structure and includes 55 classes, eight object properties, 40 data types.
Table 2. Parameters of the membership function.

<table>
<thead>
<tr>
<th>Physical parameter</th>
<th>Range boundaries</th>
<th>Very little</th>
<th>Little</th>
<th>Good</th>
<th>Big</th>
<th>Very big</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exhaust gases</td>
<td>0-1000 °C</td>
<td>a&lt;100</td>
<td>a=100</td>
<td>a=350</td>
<td>a=600</td>
<td>a=720</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b=200</td>
<td>b=275</td>
<td>b=560</td>
<td>b=700</td>
<td>b=800</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c=200.5</td>
<td>c=350.5</td>
<td>c=600.5</td>
<td>c=720.5</td>
<td>c&gt;1000</td>
</tr>
<tr>
<td>Engine oil</td>
<td>0-150 °C</td>
<td>a&lt;0</td>
<td>a=10</td>
<td>a=20</td>
<td>a=80</td>
<td>a=120</td>
</tr>
<tr>
<td>temperature, °C</td>
<td></td>
<td>b=5</td>
<td>b=15</td>
<td>b=30</td>
<td>b=100</td>
<td>b=135</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c=10.5</td>
<td>c=20.5</td>
<td>c=60.5</td>
<td>c=120.5</td>
<td>c&gt;150</td>
</tr>
<tr>
<td></td>
<td>0-20 kgf/cm²</td>
<td>a&lt;0</td>
<td>a=2.0</td>
<td>a=5.00</td>
<td>a=10</td>
<td>a=15.2</td>
</tr>
<tr>
<td>Engine oil pressure</td>
<td></td>
<td>b=1</td>
<td>b=3.5</td>
<td>b=8</td>
<td>b=12</td>
<td>b=17.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c=2.05</td>
<td>c=5.05</td>
<td>c=10.5</td>
<td>c=15.5</td>
<td>c&gt;20</td>
</tr>
<tr>
<td>Main gearbox oil</td>
<td>0-100 °C</td>
<td>a&lt;0</td>
<td>a=10</td>
<td>a=20</td>
<td>a=50</td>
<td>a=80</td>
</tr>
<tr>
<td>temperature, °C</td>
<td></td>
<td>b=5</td>
<td>b=15</td>
<td>b=35</td>
<td>b=70</td>
<td>b=90</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c=10.5</td>
<td>c=20.5</td>
<td>c=50.5</td>
<td>c=80.5</td>
<td>c&gt;100</td>
</tr>
<tr>
<td></td>
<td>0-8 kgf/cm²</td>
<td>a&lt;0</td>
<td>a=2.0</td>
<td>a=3.45</td>
<td>a=4.50</td>
<td>a=7.5</td>
</tr>
<tr>
<td>Main gearbox oil</td>
<td></td>
<td>b=1</td>
<td>b=2.5</td>
<td>b=4</td>
<td>b=5</td>
<td>b=7.8</td>
</tr>
<tr>
<td>pressure, kgf/cm²</td>
<td></td>
<td>c=2.05</td>
<td>c=3.5</td>
<td>c=4.55</td>
<td>c=7.55</td>
<td>c&gt;8</td>
</tr>
</tbody>
</table>

Table 3 contains objects properties of the used in the work (OP - oil pressure, EGT - exhaust gas temperature, OT - oil temperature, PP - power plant).

Table 3. Property of objects.

<table>
<thead>
<tr>
<th>Property</th>
<th>Domain</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>has OP main gearbox</td>
<td>main gearbox</td>
<td>OP main gearbox</td>
</tr>
<tr>
<td>has OP left engine</td>
<td>PP gearbox</td>
<td>OP PP gearbox</td>
</tr>
<tr>
<td>has OP right engine</td>
<td>PP gearbox</td>
<td>OP PP gearbox</td>
</tr>
<tr>
<td>has EGT left engine x</td>
<td>PP gearbox</td>
<td>EGT PP gearbox</td>
</tr>
<tr>
<td>has EGT right engine</td>
<td>PP gearbox</td>
<td>EGT PP gearbox</td>
</tr>
<tr>
<td>has OT main gearbox</td>
<td>main gearbox</td>
<td>OT main gearbox</td>
</tr>
<tr>
<td>has OT left engine</td>
<td>PP gearbox</td>
<td>OT PP gearbox</td>
</tr>
<tr>
<td>has OT right engine</td>
<td>PP gearbox</td>
<td>OT PP gearbox</td>
</tr>
</tbody>
</table>

Property declaration example for «hasOPMainGearbox»

```xml
<SubObjectPropertyOf>
  <ObjectProperty IRI="# hasOPMainGearbox"/>
  <ObjectProperty IRI="owl:topObjectProperty"/>
</SubObjectPropertyOf>

<ObjectPropertyDomain>
  <ObjectProperty IRI="# hasOPMainGearbox"/>
  <Class IRI="#MainGearbox"/>
</ObjectPropertyDomain>

<ObjectPropertyRange>
  <ObjectProperty IRI="# hasOPMainGearbox"/>
  <Class IRI="#OTMainGearbox"/>
</ObjectPropertyRange>
```

In addition, 40 data types were allocated: 5 fuzzy labels for 8 variants of relationships. The data type parameters correspond to the parameters of the membership function. The type of membership function in all data types was chosen triangular. Example of declaring a data type in FuzzyOWL notation:

```xml
<AnnotationAssertion>
  <AnnotationProperty IRI="#fuzzyLabel"/>
  <IRI>#BigOPMainGearbox</IRI>
  <Literal datatypeIRI="&rdf;PlainLiteral">
    <fuzzyOwl2 fuzzyType="datatype">
```

V International Conference on "Information Technology and Nanotechnology" (ITNT-2019) 255
As an object of experiments, time series for the diagnostics of helicopter units and the fuzzy ontology of the helicopter units design were investigated. In the course of these experiments, the fuzzy time series and fuzzy ontologies integration algorithms were used.

4. FTS and Fuzzy ontology integration system
A software system was developed to solve the problems of forming the inference of the recommendation based on the integration of fuzzy time series and fuzzy ontologies. The software system is written in C# on the .NET 4.5 platform. The system development was carried out in the Microsoft Visual Studio 2015 environment. SQLite was used as the DBMS. The exchange protocol is a function call to the SQLite library. This method simplifies the program and shortens the response time. To store the database (definitions, tables, indexes, and the data itself), a single standard file is used on the computer on which the program runs.

The expert develops a fuzzy ontology of the domain with the help of the ontology editor Protégé. To check the adequacy and consistency of the ontology, the built-in Reasoner HermiT or FACT++ is used. The scheme of the used software package is presented in Figure 1.

The user has the opportunity to conduct research using the developed integration system. A prerequisite for obtaining an inference is to combine a time series with annotation properties. The result of the study is the resulting list of abnormal situations and possible faulty helicopter units.

5. Experiments
Diagnostics of a helicopter consists in checking its units in order to establish their serviceability and the possibility of operating the whole helicopter. The result of the diagnosis will be an assessment of the values of key physical quantities. The main goal is to assess the danger of values. To check the adequacy of the algorithm for integrating fuzzy time series and fuzzy ontology based on FuzzyOWL, as well as the correctness of the software that implements this algorithm, a series of experiments were conducted in which possible problem situations were performed. As part of the experiment, the following actions were carried out:

1. The expert has developed a fuzzy ontology according to the FuzzyOWL standard. To build a fuzzy ontology, the Protégé [23] editor with the connected FuzzyOWL Plugin [24] was used.
2. FuzzyOWL fuzzy ontology data types contain parameters of membership functions.
3. FuzzyOWL fuzzy ontology data types contain a binding to a specific class of ontology (Table 4).

The task of the experiments is to search for possible faulty helicopter units. The analysis represents the sequence of the following steps:

1. the formation of TS on the basis of the obtained information on the values of key physical quantities after running the machine;
2. search for defective helicopter units in the received TS;
3. determination of defective helicopter units.
A helicopter unit will be considered faulty if at least one abnormal situation is detected for a physical quantity associated with a specific ontology class corresponding to the faulty unit.

The effectiveness of the diagnostic algorithm of technical systems can be evaluated when solving the problem of modeling the behavior of helicopter units. The system should correctly identify possible faulty helicopter units. To confirm the efficiency, it is necessary to analyze the data characterizing the machines, both without defects and with possible defects, and then analyze the information about the faulty units obtained by the system and received from an expert.

For the experiment, data were obtained on the run of the three machines, and data was generated that simulates certain abnormal situations. Description of the time series is given in Table 5.

The following designations are used: TVG1 - left engine exhaust temperature, TVG2 - right engine exhaust temperature, Pm1 - left engine oil pressure, Pm2 - right engine oil pressure, Tm1 - left engine oil temperature, Tm2 - right oil temperature engine, Pmp - oil pressure of the main gearbox, Tmp - oil temperature of the main gearbox.

Experiments were conducted with ten-time series. The results of experiments are shown in Table 6.
The result of the experiment is the construction of a fuzzy time series fuzzy ontology allowed us to conclude that the helicopter unit was malfunctioning when analyzing the precise values of the aggregates.

6. Conclusion
In this paper a technique for constructing fuzzy ontologies was investigated and an ontological model of the state of helicopter units was developed. In the process of integrating fuzzy time series and fuzzy ontology, the method integrating TS and ontology was implemented, and a software product was developed that ensures the implementation of this method.

Also, experiments were conducted to search for anomalous situations and search for possible faulty units using the developed approach to the integration of fuzzy time series and fuzzy ontology.

Thus, the proposed approach of hybridization of FTS and fuzzy ontologies allows one to reliably recognize anomalous situations with some degree of truth. The algorithm also finds possible faulty units corresponding to each abnormal situation.

7. References


[16] Romanov A A, Egov E N, Moshkina I A and Dyakov I F 2018 Extraction and Forecasting of the International Scientific and Practical Conference (Ulyanovsk, Russia) 50-55


[22] Danilov V A, Zheleznyak I I and Mordik V V 213 Operation and repair of the Mi-8 helicopter: [training manual for the technical staff of operational enterprises and cadets of civil aviation aviation engineering schools] (Moscow: Mechanical Engineering) p 1980

[23] Protégé: ontology editor URL: https://protege.stanford.edu


Acknowledgments
The study was supported by: The Ministry of Education and Science of the Russian Federation in the framework of the projects No. 2.1182.2017/4.6 and 2.1182.2017 and The Russian Foundation for Basic Research (Grants No. 19-07-00999 and 18-37-00450, 18-47-732007).