

Research Article

On-Line Interpretation and Real-Time Diagnosis of Rocket's Single Equipment

Erbao Xu¹,¹ Yan Li,¹ Lining Peng,¹ Yuxi Li,² and Mingshun Yang¹

¹School of Mechanical and Precision Instrument Engineering, Xi'an University of Technology, Xi'an 710048, China ²Xi'an Modern Control Technology Research Institute, Xi'an 710048, China

Correspondence should be addressed to Erbao Xu; baojams@126.com

Received 8 December 2020; Revised 9 February 2021; Accepted 24 February 2021; Published 13 March 2021

Academic Editor: Bekir Sahin

Copyright © 2021 Erbao Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The work state of a launch vehicle is generally interpreted automatically on software. However, the sheer number of target parameters makes it difficult to realize real-time interpretation, and abnormal interpretation result does not necessarily mean that the vehicle is in abnormal state. This paper introduces the edge computing to achieve on-line interpretation and real-time diagnosis of a single launch vehicle. Firstly, the parameters to be interpreted were subjected to thresholding, leaving only those with high interpretation value. Next, the interpretation server layer of the real-time diagnosis model was built based on the attribute and value reduction algorithm of variable precision rough set (VPRS). Moreover, the higher-grade criteria were written in criterion modeling language (CML) and used to interpret the various higher-grade interpretation data pushed by the edge layer in real time. On this basis, the outputs of the edge layer and interpretation server layer were integrated to achieve the real-time diagnosis of single vehicle faults. Finally, the proposed model was proved feasible through the application in a launch vehicle.

1. Introduction

During the "Twelfth Five-Year Plan" and "Thirteenth Five-Year Plan" period, China's launch vehicle and weapon models entered a high-density launch period, and the research model entered a critical period of development. Compared with the past, the launch task is heavy and the test cycle is significantly shortened. In the process of test and launch, the measurement data interpretation is an important part, which is directly related to the function and performance judgment and evaluation of each system of the aircraft [1, 2].

In the current telemetry and other parameter interpretations of launch vehicle, the parameter data received by the system in real time are generally saved. After the test is finished, all the data of all parameters stored in the disk are processed uniformly, and then, in the interpretation server, the automatic interpretation software is used for interpretation, and generally a fixed threshold comparison is used. The statistical results show that the time allocation of the launch vehicle to perform a total inspection test at the shooting range is 30% for test preparation and 40% for power-on test, and the remaining 30% is for postprocessing and interpretation [1].

However, CZ3, cz5, cz7, and other new rocket models adopt a high bit rate measurement system, with many fastchanging parameters and large volume of data, resulting in heavy interpretation workload, long interpretation time, and increased risk of misjudgment. Especially for attitude control and guidance profession, the workload of interpretation is larger. Therefore, the amount of parameter data to be interpreted is extremely large, and the interpretation server needs to carry out complex analysis processing such as path selection for the whole package of data before automatic interpretation [2]. The hysteresis nature of interpretation results caused by after-event interpretation makes it impossible to find and locate problems in the first place and on the spot and then to grasp the real-time status of the rocket, which has seriously affected the quality and progress of the rocket range test. After the test, the state of the rocket based on the interpretation afterwards is also meaningless due to the lack of timeliness and accuracy. Therefore, only if the time for data processing and interpretation is infinitely shortened, real-time on-line interpretation is achieved, and the speed and accuracy of condition diagnosis are improved can it meet the needs of modern rocket range testing [3].

It can be seen that the problems of the current autointerpretation system of the launch vehicle include the following: the hysteresis nature is caused by the large amount of data as well as the use of after-event interpretation; most of the criterions are threshold criterions, which are single and lack intelligence; the computing resources of the interpretation server are mostly used for threshold comparison of large amounts of normal data parameters, as well as lack of focus and special attention on key parameters or over limit data that truly reflect the fault status of the equipment. Therefore, at present, although the autointerpretation software is a great workload and long period by hand, it is inefficient and inaccurate. Domestic and foreign scholars have also conducted extensive research on this issue.

In Guo et al.'s research [4], for several types of parameters of the rocket range, the median filtering algorithm is used to achieve smooth filtering of the step parameters and pulse parameters. A bilateral multipoint threshold judgment method is designed to automatically and accurately identify the step parameters and realize automatic auxiliary interpretation function, but this method is limited to step parameters and pulse parameters, and it is not applicable to a large number of slowly varying parameters that are difficult to interpret; Shi et al. [5] believed that an open interpretation function compiling platform is proposed based on the research background of on-orbit fault diagnosis. From the perspective of the architecture design of the telemetry parameter interpretation platform and from the perspective of the architecture design of the telemetry parameter interpretation platform, the composition, implementation technical details, and interpretation rules of the platform are discussed, but the analysis of the telemetry parameter interpretation method is lacking. Dong et al. [6] proposed a novel method of automatic interpretation using linear prediction method and prediction method based on GM (1, 1) model to solve the problem of slow-varying telemetry parameters changing. To some extent, this method can realize the interpretation of slow variation parameter but lacks the mining of historical data of telemetry parameters. Based on the statistical characteristics of historical data, Hu et al. [7] put forward a more comprehensive content and a view calendar in view of the weak link of the existing rocket flight data comparison and achieved the same parameter history data comparison function and also achieved the slicing viewing function for the step type signal of severe jump; however, this research mainly focuses on the historical telemetry parameter data and lacks the consideration of massive real-time data interpretation and real-time diagnosis; Li et al. [8] believed that a new method of automatic interpretation of telemetry slow variation parameter based on the statistical characteristics of historical data is proposed, which can effectively identify abnormal parameters in the slow variation parameter. Compared with the traditional manual interpretation method, this method can effectively

improve the efficiency and accuracy of the interpretation of telemetry parameters but still does not achieve a real realtime interpretation. Guo et al. [9] proposed an intelligent fault diagnosis method for aeroengine sensors combining a deep learning algorithm (CNN) with time-frequency analysis, wherein the signal recognition problem is transformed into an image recognition problem. In this paper, a large amount of historical data is used for CNN model training, which is still after diagnosis and does not take real-time diagnosis as the research goal. Glowacz [10] described fault diagnosis method based on analysis of thermal images. It was found that RE_{EID-3-} CLASSES</sub> of the backpropagation neural network (97.91%) was slightly lower than the result of the NN (100%). The results of the experimental analysis showed usefulness of the proposed approach. This approach can find application for protection of rotating machinery and engines.

In recent years, with the universal application of the Internet of Things technology and the gradual rise of edge computing technology, it is urgent to use these new technologies to technically reconstruct the autointerpretation system of the launch vehicle. In this paper, a "double-layer interpretation" architecture of edge-interpretation server layer is proposed, which has achieved good results in actual rocket range tests. It has realized on-line automatic interpretation and real-time diagnosis, which verifies the effectiveness and practicability of the method.

The remainder of this article is arranged as follows: Section 2 describes the concept and construction process of "double-layer interpretation" and gives its network topology diagram and software framework diagram; Section 3 describes the role and the specific data flow of edge interpretation and clarifies the significance and innovation of introducing the edge interpretation layer; Section 4 describes the work undertaken by the interpretation server layer, introduces the original CML criterion syntax and the format of higher-grade criterion, and gives the process of extracting the simplest diagnostic rules by VPRS step by step. In Section 5, the effectiveness of the proposed strategy is verified by a project example. Firstly, the construction process of the edge interpretation layer is described in detail according to Section 2. Then, the specific process of highergrade interpretation is shown through the software interface in Section 3. Finally, a specific example is given to show the detailed steps of using VPRS to extract the simplest diagnostic rules for real-time interpretation. In Section 6, we get our final conclusion; that is, the "double-layer interpretation" strategy can realize on-line interpretation and realtime diagnosis of single equipment of launch vehicle. The block diagram of the research is shown in Figure 1.

2. On-Line Interpretation Architecture

The "double-layer interpretation" of the edge-interpretation server layer mainly includes two aspects: Firstly, at the edge, the source of data generation usually refers to the rocket or the general inspection workshop, and the existing sensors and other data acquisition equipment upgrade hardware to make it an edge computing node, giving it the ability to



FIGURE 1: Block diagram of the research.

perform simple calculations, small-scale data storage, and data transmission. For devices that are not convenient to upgrade, a data collection box can be deployed separately as an edge computing node. The edge computing nodes distributed in different subsystems or data sources constitute the edge-interpretation layer. Each node collects the telemetry parameters of each corresponding equipment in real time and compares the threshold directly in the node. Most of the normal data that have not exceeded the limit will be selectively transmitted out after local aggregation within the edge node according to the set time interval, and all data that exceed or are about to exceed the limit will be specially marked and immediately transmitted out. Secondly, a new higher-grade criterion is added to the traditional interpretation server. Meanwhile, a diagnosis model is established by VPRS data mining method. The relationship between parameter overrun and equipment status is mined from the previous fault data of equipment, the importance of different interpretation parameters is analyzed quantitatively, and accurate and highly simplified diagnosis rules are deduced. Finally, for the high-value parameter interpretation data from the edge interpretation layer in real time, the highergrade criterion is used for in-depth interpretation. If the interpretation result is abnormal, it is substituted into the diagnosis model for state diagnosis.

The "double-layer interpretation" network deployment topology is shown in Figure 2, and its software architecture is shown in Figure 3.

3. Edge Interpretation

Compared with the remote computing mode of cloud or dedicated server, edge computing refers to localized data collection, instant computing, real-time on-line diagnosis, timely response, and precise control at the data source close to the physical environment [11, 12]. Edge computing integrates the functions of computing, storage, and transmission into the equipment, so that data can be processed in a timely and effective manner near the source of the collection without being transferred to a cloud server, which will greatly improve data processing efficiency and value density and reduce cloud server concentration load of processing data [13, 14].

In the "double-layer interpretation" architecture, the edge-interpretation layer undertakes important functions such as raw data collection, data cleaning, data threshold interpretation, data sieving, and uploading. According to the distribution of data sources, edge nodes are allocated and deployed reasonably, and each edge node collects the parameters of the corresponding subsystem or stand-alone equipment. In the edge node, the threshold criterion corresponding to the acquisition parameters is stored in advance. At the same time, an over limit recognition area is added based on the fixed upper and lower limit of the threshold criterion. The range of the recognition area is slightly larger than the threshold range. The massive parameter of real-time data is first compared with the corresponding threshold in the edge node. For most normal data that are within the threshold and do not enter the over-limit identification area, they are temporarily stored in the internal buffer of the edge node, and then the maximum and minimum values of the data in this period are selected at a certain interval and sent to the interpretation server. For a few abnormal data that exceed the threshold range, they are immediately uploaded to the interpretation server. For data that enter the over-limit identification area but do not exceed the threshold for the time being, they are closely followed by measures to increase the collection frequency. The flow chart of edge-interpretation layer data is shown in Figure 4.

The main purpose of the edge-interpretation layer is to transfer the simple but heavy work of parameter threshold comparison from server to multiple edge computing nodes for distributed parallel interpretation. According to the interpretation results, more valuable or abnormal data is screened into the interpretation server. In this way, not only can the amount of data in interpretation server be greatly reduced, but also the efficiency of interpretation can be effectively improved, and real-time on-line interpretation of data can be realized.

4. Interpretation Server

At present, most of the autointerpretation systems use threshold criterion or fixed value upper and lower limit criterion [15]. When the judged parameter data exceeds the threshold, it is regarded as a parameter anomaly. This kind of criterion is mainly applied to the interpretation of a large number of slow variation parameters such as voltage, current, temperature, and pressure. The advantage of the threshold criterion is that the calculation is simple and the reading speed is fast, but it can only simply reflect whether the parameter is within the set threshold range. However, if the parameter exceeds the threshold value, it does not mean that the single equipment is faulty. The fuzzy mapping relationship between the two usually needs to be defined by an expert system or even manually, which often causes the



FIGURE 2: Network topology diagram of double-layer interpretation.



FIGURE 3: Architecture diagram of double-layer interpretation.

problem of "judgment is not allowed," so that the interpretation work only stays in the judgment of whether the parameter is abnormal, and loses the function of diagnosing the fault state of the equipment.

In view of this, at the interpretation server layer, a new criterion description language is first defined to describe more complex and intelligent higher-grade criteria. The parameters are selected by the edge layer threshold criterion; according to their characteristics, write corresponding higher-grade criteria, perform higher-grade interpretation and obtain the interpretation results, and store them in the HBase distributed database. Then, make full use of the historical parameter data of the single equipment's faults and establish a diagnostic model.



FIGURE 4: Flow chart of data interpretation.

There are many methods to establish the diagnostic model [10]. Considering the need to extract the most simplified diagnostic rules, the algorithm is required to have the ability of parameter reduction and value reduction. So, we adopt variable precision rough set (VPRS) theory and use its attribute dependency and importance calculation methods to quantitatively analyze the parameter interpretation results and the equipment status association relationship and degree of correlation and dig out the key parameters that affect the working status of the equipment. With the help of attribute reduction and value reduction algorithms of VPRS, diagnostic rules are extracted and finally accumulated into a diagnostic rule base. Therefore, the fault status of the device can be inferred intuitively based on the parameter interpretation results, breaking the barrier between autointerpretation and fault diagnosis and realizing the on-line interpretation and real-time status diagnosis of the single equipment.

4.1. CML Criterion Description Language. At present, most of the criteria are described by natural language. How to accurately convert the criteria to computer language is the first key technology to realize autointerpretation. In order to solve the above problems, a computer advanced language CML (criterion modeling language) containing grammar and semantics is designed before the interpretation platform to implement attitude control, guidance, power, electrical overall, and measurement system or professional criterion description, mathematical modeling, and autointerpretation, while automatically generating data interpretation analysis report. As shown in Table 1, CML includes identifier, expression, and criterion syntax.

4.1.1. Identifier. The identifier is used to identify the valid character sequence of the variables and functions. It is stipulated that the identifier can only consist of three characters: letters, numbers, and underscores, and the first character must be a letter or an underscore. In CML, there are two main types of identifiers:

- Keywords. 32 keywords are specified, such as diff, mp, CP, and envelope. They have specific uses and meanings and cannot be used as variable names.
- (2) System predefined identifier. Some examples are the names of library functions provided by the system, such as sin and cos.

4.1.2. Expression. Expressions are the basic unit of a sentence. Expressions are used in time periods, processing formulas, criterion formulas, and errors. Commonly used expressions are shown in Table 1. TABLE 1: Table of expressions in CML.

Expression name	Expression meaning
	Expression 1 operator expression 2, operators include + - * /arithmetic operator and &&; not logical
Operator expression	operators
Function call expression	Function name (expression 1, expression 2,, expression n)," functions include system built-in
Function can expression	mathematical functions and user-defined functions
Take parameter value	P (parameter code [point index]) P (Ujd)-P (Ujd[-1]) calculates the difference between the current point
expression	value and the previous point value of parameter Ujd
Take parameter time	T (parameter code [point index]) T(Ujd)–T(Ujd [1]) calculating the difference between the current time
expression	and the next time of parameter Ujd.
Instruction action time	Parameter code switch symbol the sequence of occurrence " \uparrow " indicates ON, " \downarrow " indicates OFF, Tqf \uparrow 3
expression	indicates the time when the third switching state of parameter Tqf is ON
Parameter summation	Sum (enter expression, number expression) sum (P(Uz), 3) represents the value of the parameter Uz that is
expression	the sum of the first 3 points from the current point
Difference	Diff (input expression, compensation amount) diff (frameCount, 65536) calculates whether the difference
Difference expression	between the adjacent values of frameCount is 1

4.1.3. Criterion Grammar. The basic syntax format of the criterion is shown in Figure 5, including the parameters to be judged, start time, end time, criterion formula, positive error, and negative error. In automatic interpretation, the calculation result of the criterion formula is compared with the value of the parameter to be judged and whether the difference between the two is within the range of positive and negative errors. Among them, if the start time is empty, it means interpretation starts from the first point, and if the end time is empty, it means interpretation ends at the last point.

4.2. Higher-Grade Criterion

4.2.1. Key Points Criterion. The key points criterion is used to interpret the value of the parameter at the specified time. In the interpretation algorithm part, several sets of keys (time, value) pairs are given. In the autointerpretation, the criterion will obtain the real data values of the parameters to be judged at several key points and make a difference with the theoretical value of the set key points to determine whether the difference is within the range of positive and negative errors.

4.2.2. Change Points Criterion. The operator CP is used to describe and interpret when the value of the parameter to be judged changes. In the interpretation algorithm part, several pairs of change points (value, time) are given. In the automatic interpretation, the criterion obtains the time when the real-time value of the parameter to be judged is equal to the value of the change point and makes a difference with the set theoretical time of the corresponding change point to determine whether the difference is within the range of positive and negative errors.

4.2.3. Consistency Criterion. The operator MP is used for description. When the interpretation is performed, the first parameter is used as the reference parameter, and the subsequent parameter values are different from the first parameter value point by point to determine whether the

Usf1 {	Tct↑1, 7	$[ct \downarrow 2]$	{Isf1 * 10 + U	f2} 0.1	0.1
Î	Î	↑	Î	↑	Î
Parameters to be judged	Start time	End time	Criterion Formula	Forward error	Negative error

FIGURE 5: Diagram of criteria grammar.

difference is within the allowed error range. Consistency criterion is usually used to judge those parameters that theoretically require consistent change trends. At the same time, in the diagnostic model, the interpretation results can be combined and processed to reduce the amount of model parameters and speed up the training speed.

4.2.4. Envelope Criterion. Take the data of multiple tests of the same parameter and take out the minimum and maximum values point by point and save them. All the minimum values form the lower envelope curve of the parameter, and the maximum value forms the upper envelope curve. By calculating whether the real-time point is over envelope or the degree of over envelope, abnormal situations such as outliers, data noise points, and data disturbance are identified.

4.2.5. Frame Dropping Analysis Criterion. It is specifically used for the frame count parameter to determine whether the difference between each frame count and the previous frame count is 1 and to confirm whether there is a frame loss situation. Generally, the default is full-time interpretation. For example, diff ("frameCount," 1, 65536) means that each data point is directly different from the previous data point. If the difference is less than 0, 65536 is added for calculation. If the theoretical value is 1, it means that no frame is dropped. If the difference is greater than 1, it means that there is a frame loss.

4.3. Variable Precision Rough Set. Variable precision rough set (VPRS), as a mathematical method to solve the nonlinear correspondence problem, can mine potential knowledge and rules from massive data without prior knowledge [16–18].

The core idea is to obtain the decision or classification rules for uncertain problems through knowledge reduction, while the classification ability is unchanged [19]. VPRS introduces the threshold parameter β , which indicates that the classification error rate can exist within a certain range. The general value range of β is $0.5 < \beta \le 1$ [20]; when $\beta = 1$, VPRS is the classic rough set.

VPRS can be expressed as a quadruple $S = \langle U, A, V, f \rangle$, where $U = \{y_1, y_2, \ldots, y_n\}$ is the universe, that is, by the sample object $y_i(i = 1, 2, \ldots, n)$ consisting of a finite set; $A = C \cup D$, $C \cap D = \emptyset$, $C = \{a_1, a_2, \ldots, a_p\}$ is a finite set of conditional attributes, D is a set of decision attributes, V_a is the range of attribute a, f is the information function, f: $U \times A \longrightarrow V$ is a single mapping, that is, $\forall a \in A, y \in U, f(y, a) \in$ V_a , and f(y, a) is the information value of each attribute of each object in U [21].

By using the parameter interpretation results of a large number of single machine faults stored in HBase, all parameters that are abnormal when the fault is retrieved are retrieved to form the condition attribute set *C*, and the fault state of single machine is the decision attribute *D*.

4.3.1. Fault Dependency Calculation. The dependency of decision attribute *D* and condition attribute *C* in VPRS is defined as follows:

$$K = \gamma(C, D, \beta) = \frac{|pos(C, D, \beta)|}{|U|}, \tag{1}$$

where $pos(C, D, \beta)$ is the positive region of β .

Based on the attribute dependency calculation method of VPRS, the dependency degree between single machine fault and each interpretation parameter is analyzed quantitatively.

4.3.2. Key Parameters. In VPRS, the impact of the conditional attribute set C on the classification after removing the attribute r is

$$K(C, \{r\}, \beta) = |\gamma(C, D, \beta) - \gamma(C - \{r\}, D, \beta)|.$$
(2)

The impact of a single attribute *r* on classification is expressed as $\gamma(\{r\}, D, \beta)$; then the importance of the attribute *r* can be defined as

$$sig(C, \{r\}, \beta) = K(C, \{r\}, \beta) + \gamma(\{r\}, D, \beta).$$
(3)

The larger the value is, the more important the attribute is. It means that the abnormal parameter has a greater impact on the single machine state. With this method, the importance ranking of the interpretation parameters can be carried out, and the key parameters affecting the single machine fault state can be found [22, 23].

4.3.3. Parameter Reduction. The attribute reduction of the variable precision rough set is to reduce the condition attribute. If the dependency of a single attribute r, $\gamma(\{r\}, D, \beta)$ is equal to $\gamma(C, D, \beta)$, the condition attribute is considered as a redundant attribute.

Through this method, unnecessary parameters can be removed from many interpretation parameters, and then diagnosis rules can be extracted from the simplified parameters, which can better serve the real-time state diagnosis of a single machine.

4.3.4. Value Reduction. After attribute reduction, the decision table forms a decision rule for each sample. However, these rules are not the most streamlined and can be simplified by the following methods: For each rule in the decision rule set, if any attribute in the rule is removed and the rule does not conflict with other rules in the set, this attribute is deleted from the rule.

After value reduction, all diagnostic rules do not contain redundant condition attributes, which means that the most concise diagnostic rules are obtained from the diagnostic model.

5. Example Verification

In order to verify the effectiveness of the proposed "double-layer interpretation" architecture, taking the Long March series XX launch vehicle in Figure 6 as an example, the front end of the instrument cabin is selected as the single equipment for the research. This single machine is mainly responsible for testing the temperature and overload of the instrument cabin and judging the status of the instrument cabin. The main parameters collected by this single machine include the following: front cabin thermocouple verification (c_1) , front cabin inner wall temperature (c_2) , front cabin noise (c_3) , front cabin end handle root inner wall temperature (c_4) , front cabin end handle root cavity temperature (c_5) , temperature of the inner wall near the front handle of the front cabin (c_6) , front cabin vibration frequency (c_7) , high accuracy overload of the front cabin (c_8) , pitching attitude angle (c_9) , rolling attitude angle (c_{10}) , front end cold end compensation (c_{11}), tip handle head pressure (c_{12}), and tip handle tail pressure (c_{13}) .

5.1. Implementation of Edge-Interpretation Layer. According to the requirements of the edge-interpretation layer in Section 2, upgrade the function of the data receiver on the arrow telemetry subsystem and add data storage, threshold comparison calculation, and data transmission hardware modules. When conducting sum-check of simulation flight at the range, the bus pushes the data of 13 monitoring parameters to the edge nodes at a speed of 10,000-bit rate. After the threshold interpretation, the local cache and screening are performed according to the flow of Figure 4 and then uploaded to the interpretation server at an average speed of 100 bits. Therefore, the total amount of data transmitted is reduced to 1%.

From the HBase database, search the fault history data of the single head end handle single machine, and select the eight sets of abnormal data shown in Table 2.



FIGURE 6: Long March XX Rocket.

TABLE 2: Original data of c10~c13 parameters.

Sample Interpretation parameter									Time					
Sample	c_1	c_2	<i>c</i> ₃	c_4	c_5	c ₆	<i>C</i> ₇	c_8	C9	c_{10}	c_{11}	c_{12}	c_{13}	Time
u_1	0.1	0.2	9.6	1.3	2.3	0.8	1.7	9.4	0.1	0.6	1.5	107	98	2
<i>u</i> ₂	0.3	8.1	10	0.1	1.1	0.1	0.2	10	0.1	0.2	1.8	159	140	3
u_3	0.1	0.3	0.4	0.2	0.1	0.3	9.7	0.1	0.6	0.1	0.8	164	170	5
u_4	0.5	0.2	0.8	9.9	0.1	0.2	9.4	1	0.8	1	0.7	145	130	8
u_5	9.1	1.1	1	0.4	0.6	0.2	0.7	0.2	0.9	8.6	0.6	185	145	10
u_6	0.1	9.4	6.6	0.3	8.8	0.7	9.4	9.4	0.1	4.5	2.8	196	163	15
<i>u</i> ₇	0.3	1.2	9.3	0.7	0.4	0.2	1.1	0.5	8	0.3	1.8	145	128	18
u_8	0.5	0.7	9.5	0.2	0.2	0.1	0.3	10	9.7	0.7	1.7	98	91	20

5.2. Implementation of Higher-Grade Criterion in Interpretation Server Layer. HBase column database and Qt cross-platform programming language are used to develop automatic interpretation software, which mainly implements functions such as data reception, criterion management, real-time automatic interpretation, and interpretation result sending. According to the specific interpretation requirements of the parameters, the syntax of the APIL language in Section 4.1, and the higher-grade criteria in Section 4.2, write the highergrade criteria of the parameters. For the syntax of highergrade criteria, the input .1 and .y syntax files are compiled into .h and .cpp files that can be directly called by QT and C++ programs by using flex lexical compiler and bison syntax compiler.

The input of the .l and .y files and the output of the .h and .cpp files in this example verification are shown in Figure 7, and some high-level criteria for some parameters are shown in Figure 8. In the autointerpretation software, the real-time input interpretation data of the edge layer is read, the higher-grade criteria designed in Figure 8 are selected, and the

specific interpretation process is performed, as shown in Figure 9.

By viewing the final interpretation report, the results of threshold and higher-grade interpretation of parameters c_{11} , c_{12} , and c_{13} are normal. Samples u_5 , u_6 , and u_7 are input into the existing diagnostic rule base, and it is determined that the stand-alone status is normal in these three cases. Therefore, these parameters and samples are not taken into consideration for subsequent real-time status diagnosis in the diagnostic model, and new diagnostic rules are extracted from the remaining abnormal parameters and fault samples.

5.3. *Real-Time Diagnosis Based on VPRS.* The diagnosis model is constructed and the diagnosis rules are extracted according to the following steps.

Step 1. Build the original decision table

Construct the universe set $U = \{u_1, u_2, u_3, u_4, u_8\}$, 5 sets of interpretation sample data as the original decision table, and

D:\YiBu\DataIP\src\DataIP.AutoIPAlgorithm\Parse	🖉 D:\YiBu\DatalP\src\DatalP.AutoIPAlgorithm\Parser\flex_bison\yyparser.y
	+1+2+3+4+5+6+7+8
100 "lm" { return token::LM; }	154 multi_param_compare_statement ';' { \$\$ = \$1; }
101 "nl" { return token::NL; }	155 key_point_statement ';' { \$\$ = \$1; }
<pre>102 "bit" { return token::BIT; }</pre>	156 change_point_statement '; { \$\$ = \$1; }
103 "A" { return token::A; }	157 turn_point_statement ';' { \$\$ = \$1; }
104 "V" { return token::V; }	158 statistics_statement ';' { \$\$ = \$1; }
105 "TT" { return token::TT; }	159 envelope_statement ';' { \$\$ = \$1; }
<pre>106 "cp" { yylval->n = 1; return token::CP;</pre>	160 integrality_statement ';' { \$\$ = \$1; }
107 "cp_xor" { yylval->n = 2; return token::	161 ;
108 "cp_all" { yylval->n = 3; return token::	162
<pre>109 "tp" { return token::TP; }</pre>	163 = envelope_statement
<pre>110 "ext" { return token::EXT; }</pre>	164 : expression ':' '{' timespan_expression '}' ENVELOPE '(' param_code ','
<pre>111 "mic" { return token::MIC; }</pre>	165 expression ':' '{' timespan_expression '}' ENVELOPE '(' param_code ','
<pre>112 "mic_cmd" { return token::MIC_CMD; }</pre>	166 expression ':' '{' timespan_expression '}' ENVELOPE '(' param_code ')'
<pre>113 "mpc" { return token::MPC; }</pre>	167 expression ': '{' timespan_expression '}' ENVELOPE '(' param_code ')'
<pre>114 "envelope" { return token::ENVELOPE; }</pre>	168
115	169 Hintegrality_statement
116 "sqr" { yylval->n = B_sqr; return token:	170 : expression : { timespan_expression } { INIEGRALITY (expression 2) }
D:\YiBu\DataIP\src\DataIP.AutoIPAIgorithm\Pars	🖉 D:\YiBu\DatalP\src\DatalP.AutolPAlgorithm\Parser\flex_bison\yyparser.cpp
190 INTTIME = 300.	699 case 38:
191 CHAPTER = 301,	700
192 LESS = 302,	701 ⊟ /* Line 678 of lalr1.cc */
193 EQUAL = 303,	702 ⊟ #line 164 "yyparser.yy"
194 MIN = 304,	<pre>703 { (yyval.a) = generator.makeEnvelopeIPStatement((yysemantic_stack_[(14) - (1</pre>
195 MIC = 305,	704 break;
196 MIC_CMD = 306,	705
197 MPC = 307,	706 case 39:
198 ENVELOPE = 308,	707
199 TABLENAME = 309,	708 ⊟/* Line 678 of lairl.cc */
200 PLOI = 310,	/09 H #11ne 165 "yyparser.yy"
201 DATA = 311,	<pre>/10 { (yyval.a) = generator.maketnvelopelPStatement((yysemantic_stack_[(13) - (1 744 beach.maketnvelopelPStatement())</pre>
202 ERRUK = 312,	711 Dreak;
1000 INTEGRALITY = 313,	1.2

FIGURE 7: Input and output files of flex and bison.



FIGURE 8: Higher-grade criterion of parameters.

10 interpretation parameters $C = \{c_1, c_2, ..., c_{10}\}$, as the set of conditional attributes in the decision table. According to the interpretation results of the parameters and the actual situation of the test, determine the type of single machine failure $D = \{d_1, d_2, ..., d_4\}$ as the decision attribute set of the decision table. Among them, d_1 represents the front cabin overload, d_2 represents the front cabin temperature abnormality, d_3 represents the front cabin vibration abnormality, and d_4 represents contact friction between the end handle and the inner wall.

Step 2. Data standardization

The condition attributes in the original decision table are standardized by Z-core normalization method, as shown in the following equation:

$$x_j = \frac{x_j - E(x_j)}{D(x_j)}, j = 1, 2, \dots m.$$
 (4)

In the previous equation, $E(x_j)$ represents the mean value of the characteristic variable x_j , and $D(x_j)$ is the standard deviation corresponding to the characteristic variable x_j .



FIGURE 9: Perform process of autointerpretation.

Step 3. Data discretization

For the condition attributes and decision attributes in Table 2, discretization is needed after standardization according to Step 2. Take the interval value of [0, 5] as 0 and the interval value of [5, 10] as 1, and divide the range of continuous attributes into several subintervals to obtain the discrete decision table S', as shown in Table 3.

Step 4. Parameter dependency and importance calculation

According to formula (1), the dependence of fault state on each parameter is calculated; then, according to the calculation results of parameter dependence and according to formulas (2) and (3), the importance of parameters is calculated, and finally the order of parameter importance can be obtained as follows: $c_8 > c_4 > c_7 > c_2 > c_9 > c_5 > c_3 > c_6 >$ $c_{10} > c_1$. It can be seen that the key parameter among the 10 parameters is c_8 . In the subsequent manual interpretation and autointerpretation, according to the importance of the parameters, selectively increase the data upload amount of c_8 and c_4 and other parameters and speed up the sampling frequency of c_8 and c_4 and other parameters to focus on.

Step 5. Parameter reduction

According to the method in Section 4.3.3, set the threshold $\beta = 0.9$, use the VPRS knowledge reduction algorithm to reduce the 10 parameters, and the final reduced parameter set is { c_2 , c_4 , c_7 , c_8 , c_9 }.

Step 6. Value reduction

For the decision table after attribute reduction, each sample forms a diagnostic rule. The value reduction method in Section 4.3.4 is used to remove redundant parameters in each diagnostic rule, and the minimum reduction rule is obtained as shown in Table 4.

Step 7. Extract diagnostic rules

The diagnostic rules are extracted for each fault sample after attribute reduction and value reduction in Table 4, and the results are as follows:

Rule 1: IF($c_2 \in [0,5] \cap c_8 \in [5,10] \cap c_9 \in [0,5]$ or $c_8 \in [5,10] \cap c_9 \in [5,10]$) THEN d_1 (Front cabin overload)

Rule 2: IF($c2 \in [5,10] \cap c7 \in [0,5]$) THEN d2(Front cabin temperature abnormal)

Rule 3: IF($c_4 \in [0,5] \cap c_7 \in [5,10]$) THEN d_3 (Front cabin vibration abnormal)

Rule 4: IF($c_4 \in [5,10]$) THEN d_4 (Contact friction of inner wall of tip handle)

These rules are matched with the existing diagnostic rule library. If they are new rules, they are added to the rule base.

It can be seen from the range test that the real-time data of 13 parameters were initially collected at the edge, and the "double-layer interpretation" was used at the edge-interpretation layer to reduce the parameter data amount to 1/10 of the original. Then, in the interpretation server layer diagnosis model, the simplest diagnostic rules are obtained after reduction by VPRS theory. With two or three parameters, or even one parameter, the fault status of a single equipment can be determined, which greatly improves the efficiency of real-time diagnosis.

5.4. Discussion. Fault diagnosis is very important to the safety and reliability of equipment. There are many methods for equipment fault diagnosis. Some studies use machine

Abnormal cample interpretation	Interpretation parameter										Single equipment fault
Abilormal sample interpretation	\mathcal{C}_1	c_2	c ₃	c_4	c_5	c_6	c_7	c_8	C9	c_{10}	Single equipment laut
u_1	0	0	1	0	0	0	0	1	0	0	d_1
u_2	0	1	1	0	0	0	0	1	0	0	d_2
u_3	0	0	0	0	0	0	1	0	0	0	d_3
u_4	0	0	0	1	0	0	1	0	0	0	d_4
140	0	0	1	0	0	0	0	1	1	0	d,

TABLE 3: Decision table of discretization.

TABLE 4: Minimal decision table.

Fault sample	Int	erpret	Fault condition			
Tuur sample	c_2	\mathcal{C}_4	C_7	C_8	C9	Tuut condition
U_1	0	*	*	1	0	d_1
U_2	1	*	0	*	*	d_2
U_3	*	0	1	*	*	d_3
U_4	*	1	*	*	*	d_4
U_8	*	*	*	1	1	d_1

learning, artificial intelligence, and other methods, while others are based on current, acoustics, vibration, thermal analysis, and so on. In [10], an original method for feature extraction of thermal images BCAoID is proposed. The computed features were analyzed using the Nearest Neighbor classifier and the backpropagation neural network. It turns out to be the case that thermal images are a fast and noninvasive diagnostic method. It can be used for mechanical and electrical faults of the machine. This paper aims at the core problem that we have to use postinterpretation because of the large amount of interpretation data, so all the work focuses on how to reduce the amount of interpretation data and improve the speed of fault diagnosis, in order to achieve on-line interpretation and real-time diagnosis. Therefore, this paper uses VPRS method for fault diagnosis of rocket's single equipment. This method makes full use of large amounts of historical data and real-time data and uses big data mining technology to extract knowledge and intelligence. Finally, it can not only diagnose the current fault state of rocket's single equipment but also continuously simplify and optimize the diagnostic rules of equipment diagnosis. In the follow-up study, we will try to use thermal, acoustic, and other means for fault diagnosis.

6. Conclusion and Future Work

The main contributions in this paper are summarized as follows:

- (1) By deploying the edge layer before the existing server, a "double-layer interpretation" architecture that includes the edge-interpretation layer and server interpretation layer is constructed.
- (2) The specific process of edge interpretation is given. Through filtering the data according to the threshold interpretation, the amount of interpretation data is reduced and makes it possible to conduct real-time interpretation.

- (3) In the interpretation server layer, AIPL criterion modeling language and high-grade criterion are designed to interpret the abnormal parameters further. VPRS is used to mine diagnostic rules from historical fault data, establish real-time diagnostic model, and realize the real-time diagnosis.
- (4) The feasibility and effectiveness of the designed double-layer on-line interpretation and real-time diagnosis system are validated experimentally.

In addition to the VPRS method used in this paper, there are many other similar methods for status diagnosis at the interpretation server layer. Limited by the length of the article, we only adopt the VPRS method. In our next research, more methods will be tried to solve this problem, and the most suitable one will be selected finally through comparison of method results.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by Doctoral Innovation Fund of Xi'an University of Technology (no. 310-252072013) and the National Natural Science Foundation of China (Grant no. 52005404).

References

- L. P. Zhu, Y. S. Zhao, and L. M. Guo, "Real time automatic determine knowledge based research of telemeter parameter of carrier rocket," *Computer Engineering and Applications*, vol. 43, pp. 131–135, 2007.
- [2] B. L. Liu and D. Jin, "Research simulation method of satellite temperature based on telemetry data," *Journal of Astronautics*, vol. 36, pp. 763–768, 2015.
- [3] J. Guo and S. Jin, "Design and application of measurement data processing system," *Computer Measurement & Control*, vol. 27, pp. 250–252, 2019.
- [4] P. Y. Guo, S. W. Ding, Z. H. Tian, H. L. Zhang, and X. H. Zhang, "System design and method research for optical measurement images real-time interpretation in test ranges," *Journal of National University of Defense Technology*, vol. 36, pp. 168–174, 2014.

- [5] X. T. Shi, J. Y. Pang, X. Zhang, Y. Peng, and D. T. Liu, "Satellite big data analysis based on bagging extreme learning machine," *Chinese Journal of Scientific Instrument*, vol. 39, pp. 81–91, 2018.
- [6] S. H. Dong, C. W. Cui, and P. J. He, "Automatic interpretation for telemetry parameters based on prediction method," *IOP Conference Series: Earth and Environmental Science*, vol. 514, pp. 6–22, 2020.
- [7] B. Hu, Y. J. Zhang, and L. Li, "Research and design of comparative analysis platform for launch vehicle flight data," *Electronic Measurement Technology*, vol. 41, pp. 133–137, 2018.
- [8] X. Li, J. Z. Gao, J. F. Cui, and H. Q. Xie, "A novel method of automatic interpretation for low-varying telemetry parameters," *Journal of Astronautics*, vol. 39, pp. 585–592, 2018.
- [9] L. Gou, H. Li, H. Zheng, H. Li, and X. Pei, "Aeroengine control system sensor fault diagnosis based on CWT and CNN," *Mathematical Problems in Engineering*, vol. 2020, Article ID 5357146, 12 pages, 2020.
- [10] A. Glowacz, "Fault diagnosis of electric impact drills using thermal imaging," *Measurement*, vol. 171, pp. 1–11, 2021.
- [11] N. Shan, Y. Li, and X. Cui, "A multilevel optimization framework for computation offloading in mobile edge computing," *Mathematical Problems in Engineering*, vol. 2020, Article ID 4124791, 17 pages, 2020.
- [12] W. X. Yin, Y. Z. Chen, and L. Chi, "Dynamic workload allocation for edge computing," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 3, pp. 1–11, 2021.
- [13] J. L. D. Neto, S.-Y. Yu, D. F. Macedo et al., "ULOOF: a user level online offloading framework for mobile edge computing," *IEEE Transactions on Mobile Computing*, vol. 17, no. 11, pp. 2660–2674, 2018.
- [14] A. Abouaomar, S. Cherkaoui, Z. Mlika, and A. Kobbane, "Resource provisioning in edge computing for latency sensitive applications," *IEEE Internet of Things Journal*, p. 1. In press, 2021.
- [15] S. L. Huang, H. Z. Zhu, and X. L. Cheng, "Adaptive threshold setting method for layer-count in penetration fuse," Acta Armamentarii, vol. 41, pp. 1762–1771, 2020.
- [16] W. Ziarko, "Variable precision rough set model," *Journal of Computer and System Sciences*, vol. 46, no. 1, pp. 39–59, 1993.
- [17] S. Huang, "Limited and variable precision rough set model," *Journal of Information and Computational Science*, vol. 11, no. 10, pp. 3493–3501, 2014.
- [18] H. Fu, J. Xu, H. Zhang, M. Zhang, and X. Xu, "Fault diagnosis of wireless sensor network based on optimized probabilistic neural network," *Journal Européen des Systèmes Automatisés*, vol. 51, no. 4-6, pp. 295–308, 2018.
- [19] X. Chen, X. M. Wang, Y. Huang, and X. Y. Wang, "Fault diagnosis for tilt-rotor aircraft flight control system based on variable precision rough set-OMELM," *Control and Decision* of Astronautics, vol. 30, pp. 433–440, 2015.
- [20] M. Beynon and K. L. Buchanan, "An illustration of variable precision rough set theory: the gender classification of the European barn swallow (Hirundo rustica)," *Bulletin of Mathematical Biology*, vol. 65, no. 5, pp. 835–858, 2003.
- [21] X. Guan, X. Yi, and Y. He, "A new discretization algorithm for continuous interval-valued attributes," *Journal of Astronautics*, vol. 30, pp. 1164–1167, 2009.
- [22] A. M. Isa, A. N. M. Rose, and M. M. Deris, "Dominance-based soft set approach in decision-making analysis," *Advanced Data Mining and Applications*, Springer, Berlin, Germany, pp. 299–310, 2011.

[23] Z. Y. Suo, S. Y. Cheng, X. J. Yuan, and Y. M. Li, "Rule acquisition method and application of dominance decisionmaking information system," *Acta Armamentarii*, vol. 36, pp. 539–544, 2015.