

Multiple robots (robotic centers) and systems. Remote sensing and non-destructive testing
Роботизированные комплексы и системы. Технологии дистанционного зондирования неразрушающего контроля

UDC 004.896

<https://doi.org/10.32362/2500-316X-2026-14-4-36-52>

EDN FJRJYK



RESEARCH ARTICLE

Fault diagnostics in intelligent control systems of autonomous robots: An ontological approach

Yuri A. Bykovtsev[@], Valery M. Lokhin, Sergey V. Manko, Mikhail P. Romanov

MIREA – Russian Technological University, Moscow, 119454 Russia

[@] Corresponding author, e-mail: bykovtsev@mirea.ru

• Submitted: 23.12.2025 • Revised: 23.01.2026 • Accepted: 24.05.2026

Abstract

Objectives. The work set out to develop diagnostic tools and methods for autonomous robots for the rapid detection of faults based on the analysis of cause-and-effect relationships between observed manifestations and the sources of failures.

Methods. The paper presents an analytical review of the current scientific and technical literature on the topic of the research, which includes ontology construction methods for developing a diagnostic model for autonomous robots, Boolean algebra methods for generating queries to the ontological diagnostic model, modifications to conditional and unconditional diagnostic methods for finding cause-and-effect relationships between failures and malfunctions based on the use of an ontological model, and machine experimentation methods for estimating the processing time of queries to the ontological diagnostic model.

Results. The presented ontological model and methods for conditional and unconditional diagnostics is used to establish cause-and-effect relationships between observed failures and the malfunctions that cause them.

Conclusions. Based on an ontological approach and taking into account the availability of advanced tool support, including ontology description languages, ontology modeling tools, and ontology-based inference and reasoning technologies, a new paradigm is identified for the development of a promising class of autonomous robotic systems with increased fault tolerance and, consequently, adaptability. The development of an autonomous robot presupposes the presence of an onboard intelligent control system built on a hierarchical principle that comprises executive, tactical, and strategic levels. The results of the ontological model provide an effective basis for activating and utilizing the intelligent control system's rich algorithm databases at all levels of the control hierarchy to support increased adaptability.

Keywords: autonomous robots, intelligent systems, self-diagnostics, fault tolerance, ontology

For citation: Bykovtsev Yu.A., Lokhin V.M., Manko S.V., Romanov M.P. Fault diagnostics in intelligent control systems of autonomous robots: An ontological approach. *Russian Technological Journal*. 2026;14(4):36–52. <https://doi.org/10.32362/2500-316X-2026-14-4-36-52>, <https://elibrary.ru/FJRJYK>

Financial disclosure: The authors have no financial or proprietary interest in any material or method mentioned.

The authors declare no conflicts of interest.

НАУЧНАЯ СТАТЬЯ

Диагностика неисправностей в интеллектуальных системах управления автономными роботами на основе онтологического подхода

Ю.А. Быковцев[@], В.М. Лохин,
С.В. Манько, М.П. Романов

МИРЭА – Российский технологический университет, Москва, 119454 Россия

[@] Автор для переписки, e-mail: bykovtsev@mirea.ru

• Поступила: 23.12.2025 • Доработана: 23.01.2026 • Принята к опубликованию: 24.05.2026

Резюме

Цели. Целью работы является создание средств и методов диагностики автономных роботов для оперативного выявления неисправностей на основе анализа причинно-следственных связей между наблюдаемыми проявлениями и источниками возникновения отказов.

Методы. В ходе выполнения работы использовались методы аналитического обзора современной научно-технической литературы по тематике проводимых исследований; методы построения онтологий при разработке модели диагностики автономных роботов; методы булевой алгебры при формировании запросов к онтологической модели диагностики; модификации методов условной и безусловной диагностики в приложении к поиску причинно-следственных связей между отказами и неисправностями на основе использования онтологической модели; методы машинного эксперимента при оценке продолжительности обработки запросов к онтологической модели диагностики.

Результаты. Предложена онтологическая модель и методы условной и безусловной диагностики, позволяющие обеспечить установление причинно-следственных связей между наблюдаемыми отказами и вызывающими их неисправностями.

Выводы. Показано, что на основе онтологического подхода с учетом наличия развитой инструментальной поддержки, включая языки описания онтологий, средства их моделирования, а также технологии логического вывода и рассуждений на онтологиях, создается новая парадигма разработки перспективного класса автономных робототехнических комплексов, обладающих повышенной степенью отказоустойчивости и, соответственно, адаптивностью. Поскольку создание автономного робота предполагает наличие бортовой интеллектуальной системы управления, построенной по иерархическому принципу и содержащей исполнительный, тактический и стратегический уровни, повышение адаптивности может быть обеспечено за счет эффективного использования «богатых» баз алгоритмов интеллектуальной системы управления на всех уровнях иерархии управления, которые подключаются в зависимости от результатов работы онтологической модели.

Ключевые слова: автономные роботы, интеллектуальные системы, самодиагностика, отказоустойчивость, онтология

Для цитирования: Быковцев Ю.А., Лохин В.М., Манько С.В., Романов М.П. Диагностика неисправностей в интеллектуальных системах управления автономными роботами на основе онтологического подхода. *Russian Technological Journal*. 2026;14(4):36–52. <https://doi.org/10.32362/2500-316X-2026-14-4-36-52>, <https://elibrary.ru/FJRJYK>

Прозрачность финансовой деятельности: Авторы не имеют финансовой заинтересованности в представленных материалах или методах.

Авторы заявляют об отсутствии конфликта интересов.

INTRODUCTION

One of the top priorities for Russia's technological advancement in the 21st century involves the development of cutting-edge robotic systems. Intensive work on the development of remotely controlled, semi-automatic, and autonomous robots is being carried out by many teams at academic and industry-specific research institutes, specialized design bureaus, and the country's leading universities. Various types of developed robot models are beginning to find active application across a wide range of fields and purposes. However, an objective assessment of the operational experience with modern robotic complexes (RC) indicates the need for significant improvements in the level of their autonomy, adaptability, reliability, and fault tolerance when operating under extreme environmental conditions or in the presence of uncertainty factors and random disturbances.

One of the most important and pressing aspects of ensuring the fault tolerance of advanced autonomous robot prototypes involves the development of onboard self-diagnostic tools and methods for the rapid detection of emerging malfunctions to inform appropriate decisions regarding the full or partial restoration of lost functionality.

FEATURES OF THE DESIGN OF INTELLIGENT ON-BOARD CONTROL SYSTEMS FOR AUTONOMOUS ROBOTS

The autonomous operation of the RC is ensured by a combination of various factors, the most important of which is the presence of an onboard control system equipped with the necessary capabilities to plan behavior, appropriate actions, and movements based on the analysis of collected sensor data regarding changes in its own state and environmental conditions (Fig. 1). At the same time, the effectiveness of the full-scale implementation of the system's functional capabilities is largely determined by its adaptive properties in terms of adjusting its parameters or transforming its structure to maintain the required control quality in the face of environmental changes, disturbances, and a variety of other uncertainty factors.

Until recently, developers of robotics systems for specialized, industrial, and household applications had focused primarily on various aspects of enhancing the autonomy and adaptability of their designs through the comprehensive application of knowledge processing technologies [1, 2].

Active efforts in this field have led to significant progress on a number of challenging issues related

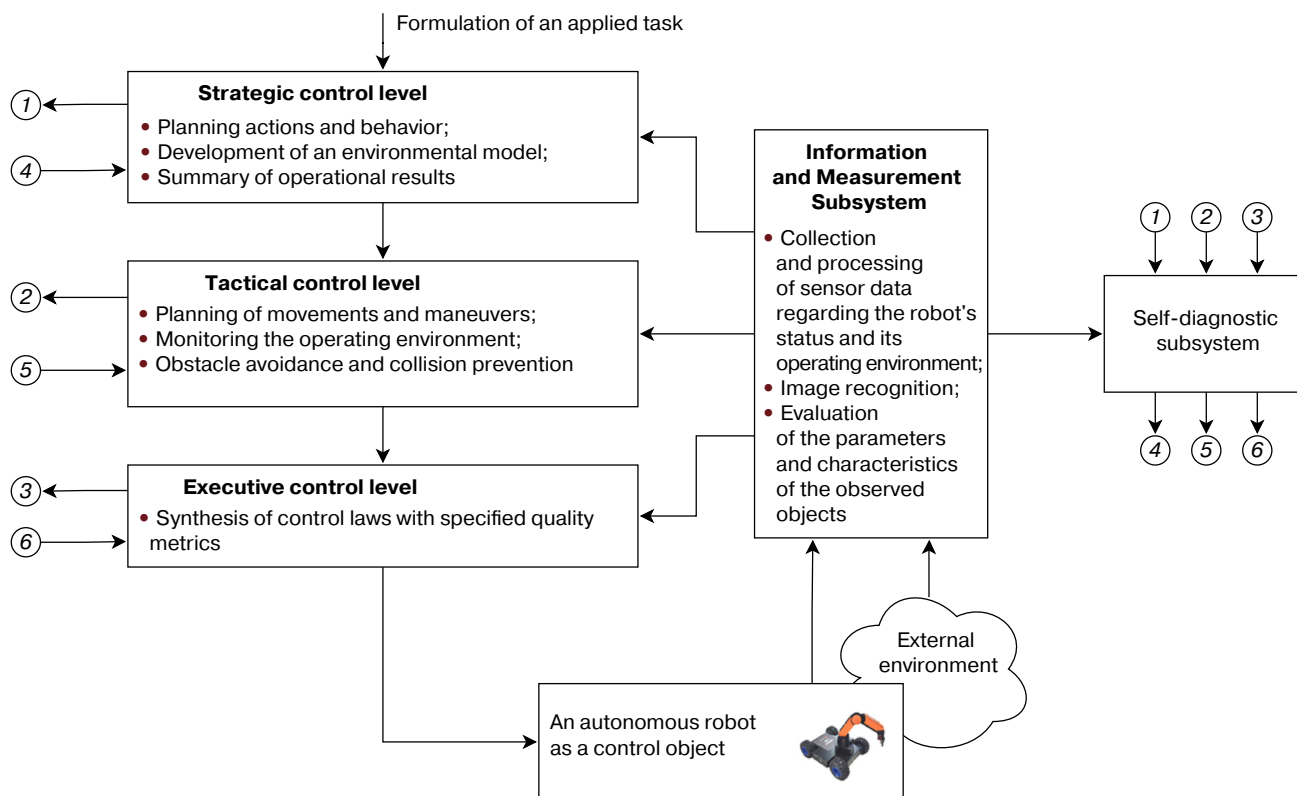


Fig. 1. Generalized structure of an autonomous robot's intelligent control system. Numbers 1, 2, and 3 denote data transmitted from the strategic, tactical, and executive levels to the diagnostic subsystem. Numbers 4, 5, and 6 denote signals from the diagnostic subsystem to the control system levels

to the intelligent control of robots operating autonomously under conditions of uncertainty. Such achievements include the development of methods for neural network recognition, intelligent planning, local navigation and mapping, deep data analysis, and self-learning.

One of the key features of the modern approach to developing intelligent control systems for autonomous robots involves the identification of subsystems at the information-measurement, behavioral, trajectory, and actuator levels, the solution of specific tasks at each of which is ensured by knowledge processing mechanisms for the selective activation of necessary algorithms depending on the nature of current situations [3, 4]. However, the range of adaptive capabilities of the control system at all levels of its hierarchy is determined by the variety of algorithms used, which are contained in the corresponding libraries, as well as the possibility of switching between them and/or effecting situational adjustment of parameters.

At the same time, it is quite clear that the reliability—and, ultimately, the effectiveness—of autonomous robots is determined not only by the sum of their intellectual and adaptive capabilities, but also by their resilience to failures inevitably caused by extreme operating conditions and harsh environmental factors.

The specific nature of the challenge of ensuring a high level of fault tolerance and survivability in advanced autonomous robotic systems stems from the need to address a set of interrelated tasks: the rapid detection and localization of failures, fault diagnosis, and the assessment of the severity of expected consequences, along with the implementation of active measures to mitigate or compensate for them.

The main challenges of the comprehensive diagnostic phase for an autonomous robot stem from two key factors:

- the wide range of possible causes that can lead to the failure of a particular functional subsystem;
- the variety of ways in which the same malfunction can manifest itself, leading to simultaneous failures in multiple subsystems.

In turn, the success of the diagnostic phase will largely determine the effectiveness of attempts to partially or fully restore the robot's operational capability through the reconfiguration of the hardware and software of the intelligent onboard control system, including changes to operating modes, procedures, and/or objectives.

Thus, the present work focuses on issues related to the development of self-diagnostic tools for autonomous robots based on an ontological model of causal relationships between observed failures and the malfunctions that cause them.

MODERN METHODS OF TECHNICAL DIAGNOSTICS

The results of research actively conducted in the field of autonomous robot diagnostics in the USA, Western European countries, and a number of other nations since the late 20th century, which can be seen as part of a broader effort to develop fault-tolerant control systems [5], are beginning to find practical application in the first experimental designs. A typical example is the autonomous unmanned RAUVER submersible vessel (Ocean Systems Laboratory, Heriot-Watt University Edinburgh, United Kingdom), which is equipped with self-diagnostic capabilities. Following the completion of appropriate diagnostic processes, on-board planners adjust the initial mission objectives as necessary [6].

The experience in carrying out search-and-rescue and other special operations using modern robotic systems (whether autonomous, semi-autonomous, or remotely controlled) clearly demonstrates that the effectiveness of their practical application is significantly limited by an insufficient level of fault tolerance [7, 8]. However, the malfunctions, failures, and breakdowns of individual functional subsystems of robot devices are highly various in nature due to being caused by random external and internal disturbances, degradation of components, elements, and materials, as well as other factors. An analysis of available statistics allows us to classify observed such failures as being caused by malfunctions in actuators, sensor subsystems, control, power, and communication systems, as well as by a number of other circumstances, including operator errors and various external influences (Fig. 2).

Actual mean time between failure (MTBF) values vary widely, ranging from a few hours to several dozen hours (depending on the type of robot, the specific tasks assigned to it, and the conditions under which they are performed) [7–10]. As an example, Table 1 presents statistical data from the U.S. Center for Robot-Assisted Search and Rescue on the assessment of the probabilities of major failures for two types of special-purpose robots over a three-year period of operation.¹

The generally accepted conceptual interpretation of fault-tolerant control systems assumes that their operational reliability must be ensured through a comprehensive set of functional capabilities designed to address the following key tasks, diagnosing emerging faults and either eliminating or compensating for them [10–12]:

¹ Carlson J. Analysis of How Mobile Robots Fail in the Field (2004). USF Tampa Graduate Theses and Dissertations. <https://digitalcommons.usf.edu/etd/980>. Accessed April 30, 2026.

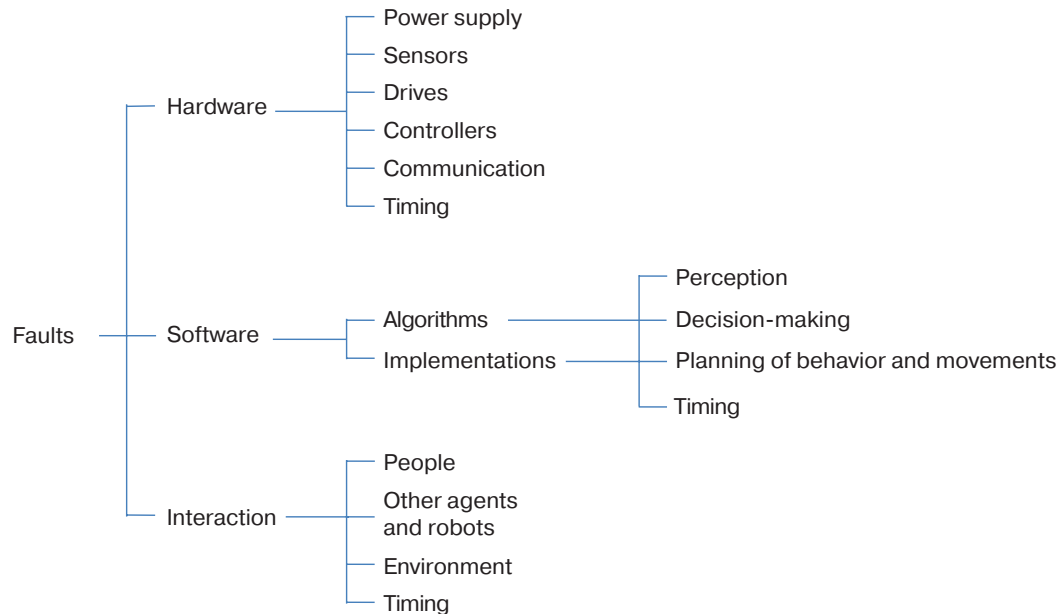


Fig. 2. Classification of faults in robotic systems

Table 1. Probability estimates for major failures in special-purpose mobile robots

Manufacturer	Actuators	Control system	Power supply system	Communication system	Sensor subsystem
Inuktun (Canada)	0.45	0.32	0.00	0.02	0.12
Irobot (USA)	0.22	0.30	0.15	0.22	0.11

- failure detection (i.e., determining that the system is behaving abnormally);
- fault localization (with the precision to identify the location and type of the failure that occurred);
- fault identification (i.e., determining the causes of a failure based on an analysis of its observable symptoms);
- assessing the severity of the identified faults and deciding whether to fully or partially restore functionality.

In the context of developing tools and methods for the automatic diagnosis of faults and the restoration of operational capability in autonomous robots, it is also necessary to distinguish between the following critically important types of failures:

- total and partial, the former of which result in the loss of an object's functionality, while the latter result in its limitation;
- defects that can be repaired (until the object regains its original functionality) and defects that can be compensated for (enabling lost functions to be restored to a limited extent).

In turn, the severity of the faults identified during the diagnostic process is assessed on a four-category scale:

- low level—at which the robot, by adjusting certain operational parameters, can continue to carry out its assigned mission at the same level of autonomy;
- average level—at which the robot can continue to perform a shortened version of the specified target mission while maintaining the same level of operational autonomy;
- significant level—at which the robot cannot continue to perform its assigned mission while maintaining the same level of operational autonomy;
- fatal level—at which the robot cannot continue to perform its assigned mission.

As a review of the literature [10–21] shows, three fundamentally different approaches serve as the conceptual foundation for the development of methods for fault detection and diagnosis; these approaches are associated, respectively, with the use of analytical models [11, 19], data analysis [13, 17], and knowledge processing [6, 16, 18]. The corresponding classification of diagnostic methods is shown in Fig. 3.

Model-based diagnostic methods involve identifying faults based on deviations between the actual observed behavior of a system and its expected behavior. The main drawback of this approach stems from the difficulty of

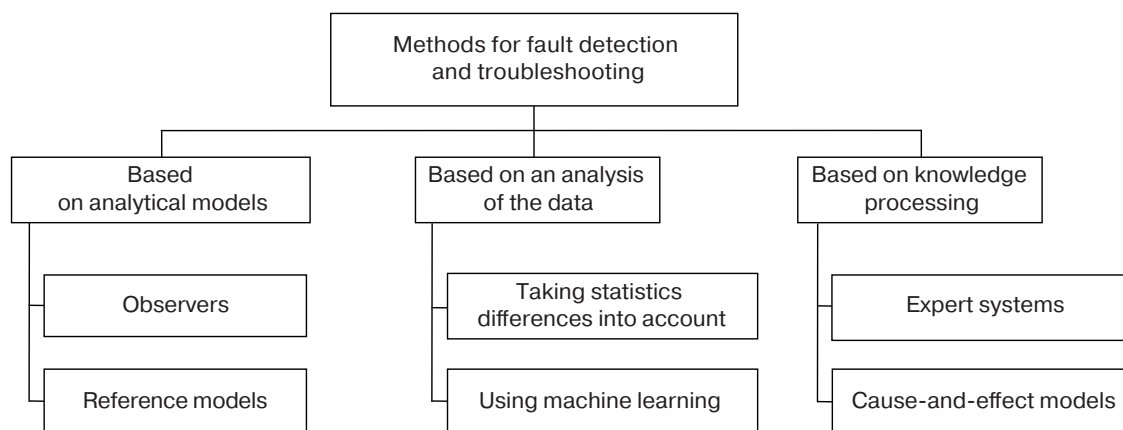


Fig. 3. Classification of methods for fault detection and diagnosis

creating adequate mathematical models of the diagnostic objects [15]. Nevertheless, model-based diagnostic methods are widely used to detect faults that arise during the operation of RC [11, 14, 18, 19, 22–27]. Within this class, methods using a reference model and an observer, which are implemented using various modifications of the Kalman filter, have become the most widespread due to their effectiveness.

A fundamentally different approach based on the analysis of data streams and the rapid detection of anomalies in the changes of monitored signals is actively used for the real-time diagnosis of robots, as well as other types of complex dynamic systems, without requiring the prior creation of mathematical models of system behavior [20, 21]. In this case, deviations of current data from statistically expected values are considered an indicator of a malfunction. To extract the necessary features that enable the detection of faults during the processing of the incoming data stream, various methods of signal analysis are applied in the time, frequency, or time-frequency domains, as well as those based on machine learning [20, 28].

For example, analyzing signals in the time domain yields a range of statistically significant characteristics, including the mean and root-mean-square values, the area under the curve, the slope and shape factor, the variance, entropy, and so on.

In turn, the analysis of signals in the frequency domain, performed using Fourier transforms, also employs relevant statistical characteristics [29, 30], the main ones being the center frequency, root mean square frequency, root of the variance frequency, and a number of others [31].

Time-frequency analysis, performed using the wavelet transform or the short-time Fourier transform, provides a detailed picture of the signal's characteristics in both the time and frequency domains [32, 33].

A set of features against which current data can be compared to detect failures or diagnose malfunctions

can be generated through machine learning using neural networks or Bayesian networks, cluster and regression analysis, or a number of other alternative mechanisms, using examples from a pre-labeled sample or operational observations from a “sliding window” [20, 34].

The main drawback of data-driven diagnostic methods is the challenge of processing large volumes of information in real time.

Knowledge-based intelligent diagnostics assumes that fault detection can be performed either at the level of expert representations, which encapsulate the experience of highly qualified specialists, or using causal models that describe the diagnostic object in terms of its structure² and the characteristics of the applied tasks being performed, as well as the manifestations of possible failures and their probable causes [6, 16, 18, 35–44].

The main drawback of intelligent diagnostic methods lies in their ability to detect and identify only a limited set of faults from among the types predefined in the knowledge base.

Various forms of knowledge representation are used to construct intelligent diagnostic models, including first-order predicates, production rules, semantic networks, frame-based structures, ontologies, etc. [16, 35, 36, 41, 45–48]. The chosen form of knowledge representation determines the set of tools and methods required for their logical processing during intelligent diagnostics. As presented in Table 2, existing approaches to fault diagnosis are often complementary in terms of their capabilities [16, 20].

² GOST R 27.302-2009. National Standard of the Russian Federation. *Reliability in Engineering: Fault Tree Analysis* (approved and put into effect by Order No. 1249-st of the Federal Agency for Technical Regulation and Metrology dated December 15, 2009). Moscow: Standartinform; 2011, 22 p. (In Russ.).

Table 2. Key features of modern approaches to fault diagnosis

Approaches	Benefits	Problems of practical application
Based on models	• Real-time processing; • The ability to detect unknown faults	• Modeling of complex systems; • Modeling of context
Based on data analysis	• Does not require a model; • The ability to detect unknown faults	• Real-time processing
Based on knowledge	• Real-time processing; • Low computational load	• Detecting of the unknown faults

For example, model-based diagnostic methods are used to detect faults in actuators, drives, internal sensors, and other system components, whose behavior can be described in terms of analytical models.

In turn, data-driven diagnostic methods are widely used in the processing of external sensor readings to detect and locate failures caused by both internal system malfunctions and environmental factors.

An effective approach to diagnosing faults in external sensors within multisensor systems involves the use of various data fusion methods, including those based on the Kalman or Dempster–Shafer filters.

The most significant advantage of a knowledge-based approach to fault detection lies in its ability to integrate data-driven and model-based diagnostic approaches within a hybrid monitoring system to identify causal relationships between observed failure symptoms and specific malfunctions or emergency situations.

Diagnosis and troubleshooting of malfunctions related to various types of software and hardware failures within the onboard computing system are carried out using established and proven methods in accordance with the relevant plan.

A summary of the results of the review leads to a number of key conclusions:

- The comprehensive use of modern technical diagnostic methods ensures the prompt detection of malfunctions that occur during the operation of an autonomous robot.
- In order to reliably identify the types of malfunctions, the vast majority of which negatively impact the performance of multiple robot subsystems simultaneously, specialized diagnostic models are required for establishing causal relationships between observed failure symptoms and their root causes.
- The use of an ontological approach enables the creation of a self-diagnostic model for an autonomous robot that captures the full set of cause-and-effect

relationships underlying its potential failures and specific malfunctions, taking into account the hierarchical structure of intelligent onboard control systems.

Depending on the severity of the detected faults, system recovery can be achieved in several different ways, including adjusting control algorithm parameters, reconfiguring hardware and software, dynamically modifying task execution plans, or adjusting operational objectives [10].

The development of tools and methods for self-diagnosis and recovery of intelligent onboard control systems opens up real prospects for creating a new generation of robotic systems characterized by a high degree of fault tolerance, autonomy, and operational reliability [20, 23].

ONTOLOGICAL MODEL FOR INTELLIGENT DIAGNOSTICS OF AN AUTONOMOUS ROBOT

The most promising approach to creating fault-tolerant control systems for autonomous robots with built-in onboard self-diagnostic capabilities involves the use of ontological knowledge representations [45, 46, 48]. Here, an ontology is understood as a formal description of a conceptual model of a domain within a specific context [49, 50].

It is important to note that the main advantage of an ontological representation of knowledge lies in its applicability to solving various functional and practical problems, both at the level of an individual robot and at the level of a robotic group.

Formally, an ontology is defined by the following tuple:

$$O = \langle X, R, F \rangle,$$

where X is a finite set of concepts in the domain; R is a finite set of relations between concepts; F is a finite set of interpretation functions defined on concepts and/or relations.

In turn, the ontological model, which allows us to describe the relationships between ontologies at different levels, is also defined as a tuple:

$$Z = \langle O, P, M \rangle,$$

where O is a top-level ontology (meta-ontology) that contains general concepts and relationships independent of the domain; P is a set of subject-area ontologies and task ontologies (taking user preferences into account); M is the inference engine model of the ontological system (for example, to modify search relevance criteria or repository formation criteria).

The use of ontologies enables a wide range of functional capabilities, the main ones being:

- formalized knowledge representation, including the organization of general and domain-specific terminology, the description of domain areas for use by various intelligent agents, and the indexing, filtering, and classification of formalized knowledge;
- acquiring and organizing knowledge;
- semantic integration of information resources (creation of a unified online knowledge space);
- enabling users to find the information they need (indexing, filtering, and classifying formalized knowledge).

The creation and processing of ontological knowledge representation models is supported by a well-developed set of tools, including ontology description languages (SHOE³, XOL⁴, OIL⁵, RDFS⁶, DAML⁷+OIL, and OWL⁸) [51], ontology modeling tools (*Apollo*⁹, *WebOnto*¹⁰, *WebODE*¹¹, *OntoEdit*¹², and *Protégé*¹³) [52], as well as technological procedures for logical inference and reasoning on

ontologies (*Racer*¹⁴, *Pellet*¹⁵, *FaCT++*¹⁶, *FOWL*¹⁷, *KAON2*¹⁸, *Hoolet*¹⁹, *Jess*²⁰) [50, 53–58].

The effectiveness of ontological knowledge representation models using in fault detection and identification tasks is demonstrated by examples of real-world technical diagnostic systems for complex electromechanical equipment, CNC machine tools, forklifts, autonomous robots, and other types of dynamic objects [36, 48, 50, 59, 60].

Ontological models are also becoming the foundational framework for the emergence of a new paradigm in the development of the next generation of autonomous mobile robots [45, 46, 61–63]. A striking illustration of this trend is the 2015 publication by the IEEE Robotics and Automation Society Ontologies for Robotics and Automation Working Group (IEEE Robotics and Automation Society Ontologies for Robotics and Automation Working Group, IEEE ORA WG) of the first-ever special standard, IEEE 1872-2015²¹, which defines a set of ontologies related to robotics and automation (R&A) [64]. From the perspective of current trends, this concept aligns with the idea of integrating intelligent control and diagnostic systems for autonomous robots.

The development of any ontology, including an ontological model for the diagnostics of an autonomous robot, is an iterative process that involves the following main stages [50], whose sequence is given in Fig. 4 in the form of a flow diagram:

- identifying the entities involved in the model (decomposition);
- creation of individual entities (identification);
- creation of classes that correspond to groups of entities and assigning entities to classes (classification);
- determining ways to express information about the characteristics and relationships of entities (properties description);
- assigning values to properties, creating relationships (values, relationships).

This approach allows the developer to focus on identifying the core concepts and objects to be incorporated into the model, and only then to define the general properties or characteristics according to which these objects can be grouped into classes to organize

³ Simple HTML ontology extensions is the ontology description language based on the HTML data markup language (hypertext markup language).

⁴ XML-based ontology exchange language is the ontology description language designed for exchanging ontology descriptions (XML—eXtensible Markup Language).

⁵ Ontology interchange language.

⁶ RDF Schema is the ontology description language, a semantic extension of the RDF language (resource description framework).

⁷ Digital asset modeling language is the open-source smart contract language.

⁸ Web ontology language is the language for describing ontologies for the semantic web.

⁹ <https://www.apollographql.com/>. Accessed April 30, 2026.

¹⁰ <http://kmi.open.ac.uk/projects/webonto/>. Accessed April 30, 2026.

¹¹ <https://oeg.fi.upm.es/index.php/en/old-technologies/60-webode/index.html>. Accessed April 30, 2026.

¹² <https://www.softpedia.com/get/Science-CAD/OntoStudio.shtml>. Accessed April 30, 2026.

¹³ <https://protege.stanford.edu/>. Accessed April 30, 2026.

¹⁴ <https://www.chai.uni-hamburg.de/~moeller/racer/>. Accessed April 30, 2026.

¹⁵ <http://pellet.owldl.com/>. Accessed April 30, 2026.

¹⁶ <http://owl.cs.manchester.ac.uk/tools/fact/>. Accessed April 30, 2026.

¹⁷ <https://fowl.sourceforge.net/>. Accessed April 30, 2026.

¹⁸ <https://km.aifb.kit.edu/projects/owltools/>. Accessed April 30, 2026.

¹⁹ <http://owl.man.ac.uk/hoolet/>. Accessed April 30, 2026.

²⁰ <https://www.ida.liu.se/~hener77/JessTab/>. Accessed April 30, 2026.

²¹ <https://standards.ieee.org/ieee/1872/5354/>. Accessed April 30, 2026.

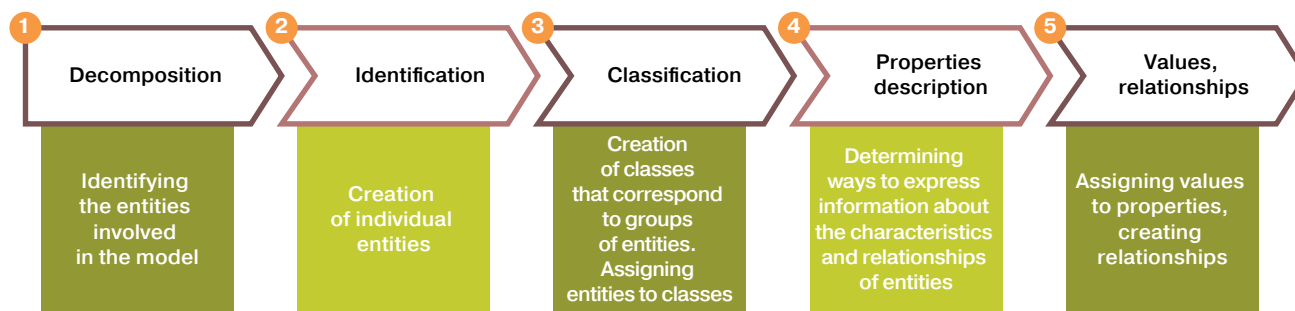


Fig. 4. Stages in the development of a diagnostic model ontology

causal, logical, quantitative, and other relationships between individual classes. At the same time, the set of basic concepts will depend directly not only on the subject area or the specific object under study, but also on the task that the ontological model is designed to solve.

One example is the development of an ontological model for diagnosing the mobile platform of an autonomous RC, which includes the following main components:

- actuators that integrate a motor, feedback sensors, and a dedicated drive control system (CS) controller;
- a control system for the platform's executive level, comprising a set of actuators, a machine vision system (MVS), and a global navigation satellite system (GNSS), as well as a central computer that generates control signals (to the actuator control system);
- power supply system with a controller;
- communication module for the tactical level of the RC control system.

In this case, the primary entities of the ontological model being created (Fig. 5) should not be the individual components of the mobile platform, but rather their potential failures and malfunctions..

For example, in a drive control system, situations such as a complete motor shutdown, a loss of power to the drive, or CS sensor failures can be classified under the separate category of "*Drive Control System Failure*."

By applying similar reasoning to each specific component or subsystem of the mobile platform, it is possible to identify the following set of classes:

- executive-level failure;
- executive level sensor error;
- drive control system failure;
- drive control unit error;
- no power supply;
- no orders from the tactical level;
- a mechanical fault;

- failure caused by interaction with the medium;
- overload.

For the identified classes of potential malfunctions in specific platform subsystems that could lead to errors in its movement, a more general class, "*Error Causes*," is additionally introduced. At the same time, the errors themselves, represented by the subclasses "*Movement Blocked*" and "*Incorrect Movement*," are grouped into the "*Errors*" class. The relationships between faults and their manifestations at the level of observable failures are defined using the pointer properties "*entails*" and "*is caused by*," which will determine the corresponding relationships between the "*Errors*" and "*Error Causes*" classes.

In this context, as properties-values characterizing the possible states of the defined entities (true or false (false)) within the ontological model, two parameters are considered: "*Error State*" and "*Suspected Error*." The former is defined for objects in both the "*Errors*" class and the "*Error Causes*" class, while the latter applies only to objects in the latter class.

It is clear that the ontological model (Fig. 5) can be expanded to account for a broader range of malfunctions and failures, which occur not only at the level of the mobile platform, but also within the RC system as a whole.

ALGORITHMS FOR UNCONDITIONAL AND CONDITIONAL FAULT DETECTION

Fault diagnosis and analysis of their potential consequences based on observed failures using an ontological model requires the formulation of queries.

In accordance with the unconditional diagnosis algorithm, when the property-value or property-pointer of an individual ontology object changes, the logical inference mechanism processes each query from its knowledge base independently of the results of processing previous queries.

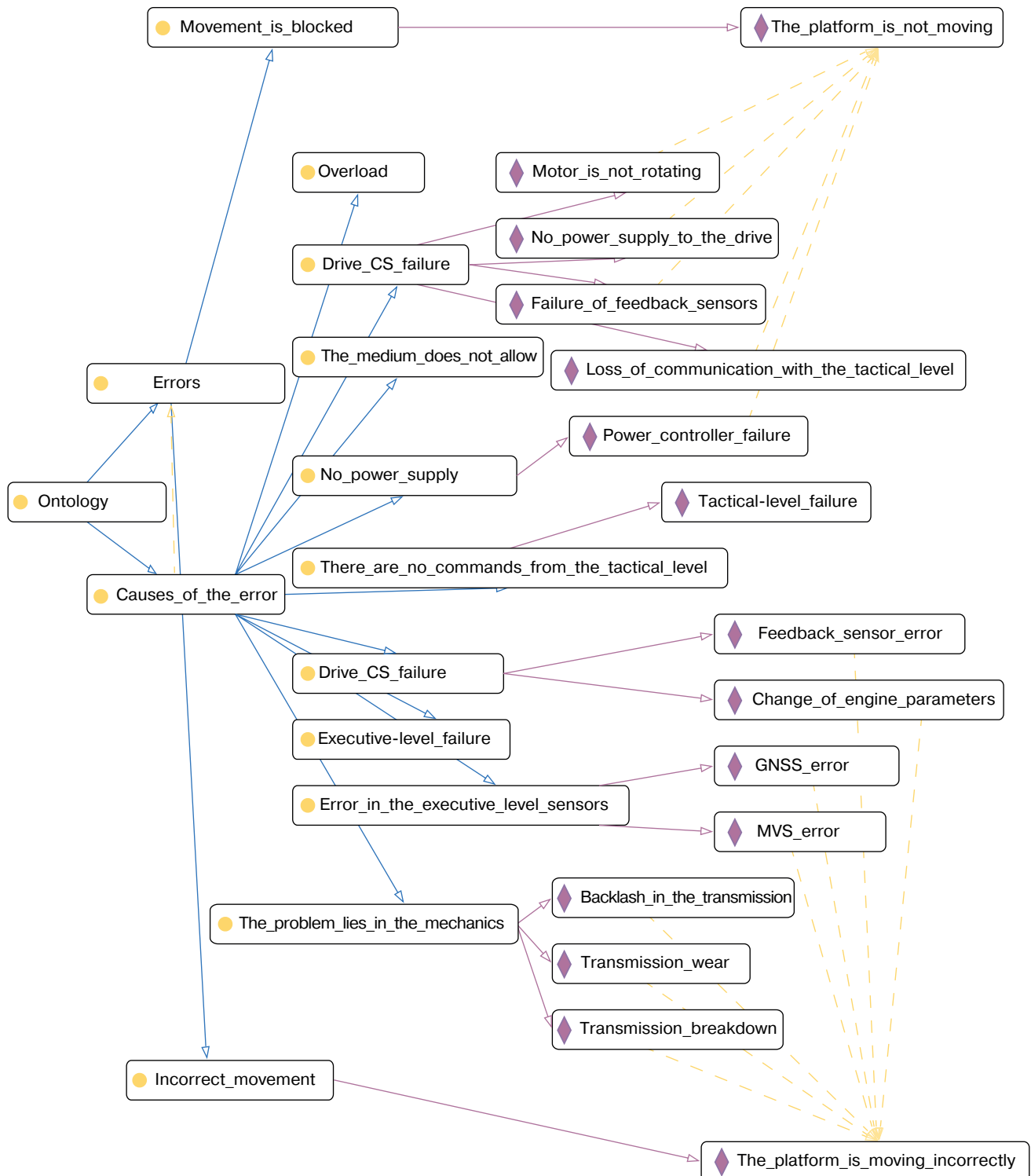


Fig. 5. Ontological model for diagnosing a mobile autonomous RC platform

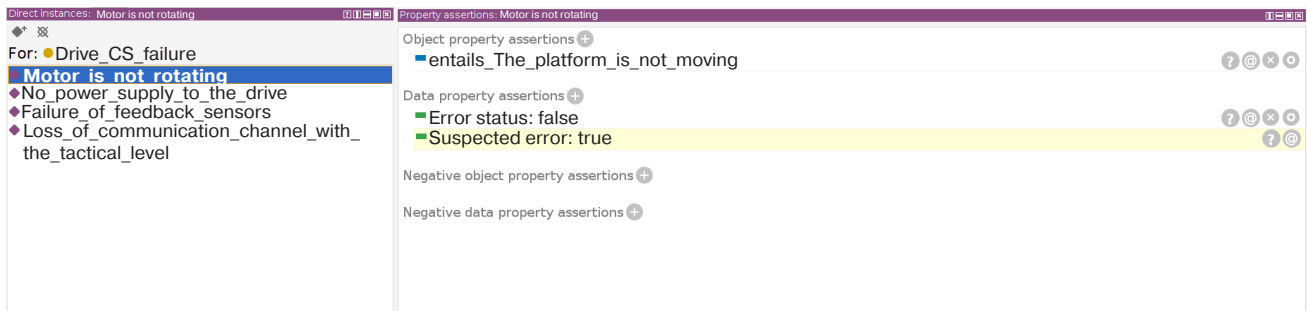


Fig. 6. Results of a simulation experiment on processing an “upstream” query regarding the potential consequences of a malfunction

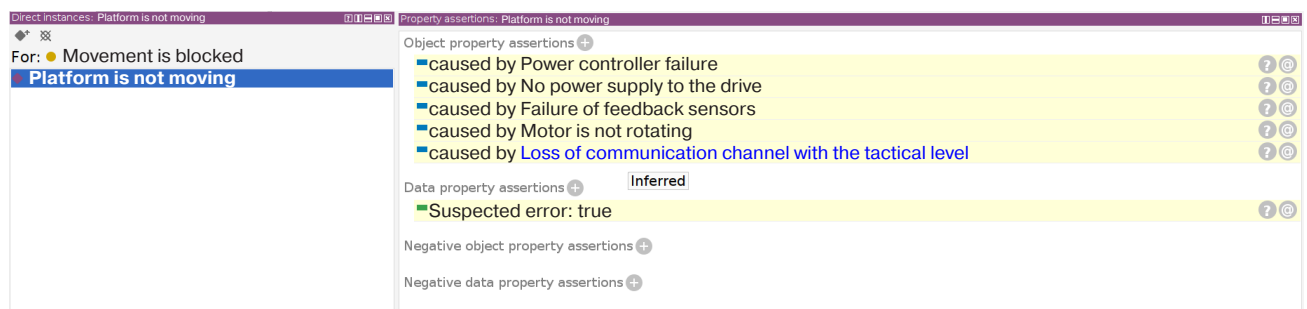


Fig. 7. Results of a simulation experiment on processing a “downstream” query regarding the possible causes of the observed failure

The general structure of queries formulated in SWRL²² (as the most feature-rich and widely used language) is as follows:

$$\begin{aligned} & \text{Class_1}(x) \wedge \text{property-value_1}(x, \text{value_1}) \wedge \\ & \wedge \text{property-pointer}(x, y) \wedge \text{Class_2}(y) \rightarrow \\ & \rightarrow \text{property-value_2}(y, \text{value_2}), \end{aligned}$$

where x and y are the objects of classes 1 and 2, respectively; property-value_1 and property-value_2 are the values (numeric, string, Boolean, etc.) of certain parameters associated with the objects of classes 1 and 2, respectively; property-pointer is the relationship linking the objects of classes 1 and 2.

This query structure enables drawing conclusions about changes in the state of an object of one class based on information about the state of objects of another class.

Specifically, an “upstream” query type, which enables tracing the manifestation of a failure at the platform level back to the occurrence of an error in a specific component, will take the form of:

$$\begin{aligned} & \text{Drive_CS_failure}(?c) \wedge \text{Error_status}(?c, \text{true}) \wedge \\ & \wedge \text{entails}(?c, ?y) \rightarrow \text{Error_status}(?y, \text{true}). \end{aligned}$$

The processed query is handled by a logical inference procedure that sequentially executes a series of relevant steps, including:

- selecting those components (c) from the “*Drive Control System Failure*” class for which the “*Error Status*” is true;
- find all objects (y) associated with the selected property-indicator “*entails*” and set the property-value “*Error Status*” for the objects y .

Thus, if for one or more objects belonging to the “*Drive Control System Failure*” class (e.g., “*Motor Not Rotating*”), the “*Error Status*” property-value takes the value “true” (true), then as a result of the request being processed by the logical inference mechanism, the property-value “*Error Status*” for the object “*Platform Not Moving*” is also assigned the value “true” (Fig. 6).

A “downstream” type of query, which enables fault diagnosis based on its manifestation at the platform failure level, is generated in the same way:

$$\begin{aligned} & \text{Movement_blocked}(?c) \wedge \text{Error_state}(?c, \text{true}) \wedge \\ & \wedge \text{caused by}(?c, ?y) \rightarrow \text{Suspected_error}(?y, \text{true}). \end{aligned}$$

The results of a simulation experiment on processing a “downstream” query regarding the possible causes of the observed failure are shown in Fig. 7.

While the absence of conditions for processing queries significantly simplifies the analysis of the results produced by the ontological diagnostic model, the increased size of both the ontology itself and the query database can lead to a critical increase in the diagnostic system’s response time when a fault occurs. In this regard, the implementation of conditional fault search appears

²² Semantic web rule language is the rule description language for the publicly accessible global Semantic web.

to be highly promising, since enabling the selection of queries to be processed from the knowledge base at the current step based on the results of the previous step, thereby reducing the overall duration of the diagnostic process.

A distinctive feature of the proposed algorithm is that it processes only those queries from the knowledge base that can lead to a change in the state of the ontological model. Here, the state of the model is understood as the set of property-value pairs of the ontology's entities, while queries are selected based on an analysis of their antecedent part.

Let $\mathbf{X}_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n})$ be the status vector of the ontological model at the current step (here, a "state" refers to the properties and values of objects); $\mathbf{X}_{i-1} = (x_{i-1,1}, x_{i-1,2}, \dots, x_{i-1,n})$ is the status vector at the previous step; $\mathbf{X} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$ is the new status associated with the occurrence of a fault; $\hat{\mathbf{X}} = (\mathbf{X}_i - \mathbf{X}_{i-1})$ is the status change vector.

In this case, the formal description of the conditional fault-finding algorithm will take the following form:

1. Accept $\mathbf{X}_i = \tilde{\mathbf{X}}$;
2. Calculate $\hat{\mathbf{X}}$;
3. If $\hat{\mathbf{X}} \neq 0$, then for each nonzero component of the vector, find a query in the knowledge base that contains that component in its antecedent. Otherwise, the algorithm terminates;
4. Accept $\mathbf{X}_{i-1} = \mathbf{X}_i$;
5. Generate a new vector $\mathbf{X}_i$ based on the results of processing the selected queries using a logical inference mechanism;
6. Go to step 2.

It is clear that the potential applicability of the diagnostic algorithms under development is largely determined by their computational complexity and average time complexity, which are primarily associated with processing queries to the ontological model, and depend not only on its structure and dimensionality, but also on the characteristics of the logical inference mechanism.

It should be noted that one of the most effective tools for constructing semantic reasoning is the Pellet logical inference engine, which, unlike many well-known alternatives (*Hermit*²³, *RacerPro*²⁴, *FaCT++*, etc.), supports working with ontologies not only at the level of classes and property-indicators, but also at the level of individual objects. This is of fundamental importance for solving diagnostic problems using unconditional and conditional search algorithms.

Estimates of the computational complexity of the Pellet logical inference mechanism as presented

in a number of studies [65, 66] follow a quadratic relationship ($O(n^2 + m^2)$) with respect to the number of classes n and relations m , and a linear relationship $O(k)$ with respect to the number of queries k . Based on these data, it can be argued that the expansion of the diagnostic model (if necessary) should be carried out by increasing the number of queries rather than by refining the structure of the ontology itself, which is oriented toward finding causal relationships between observed failures and the malfunctions causing them. At the same time, monitoring the correct operation of individual robot components, nodes, and subsystems should be carried out using local diagnostic tools and methods based on models and data analysis. A series of experiments conducted on a personal computer with a quad-core i5-7200U central processing unit operating at a clock speed of 2.50 GHz showed that the maximum processing time for a query to the presented ontological diagnostic model is 61 ms, which is a perfectly acceptable figure in the context of fault-tolerant control of autonomous robots.

CONCLUSIONS

Ensuring fault tolerance, representing one of the most critical requirements for next-generation autonomous robots, requires the development of mechanisms for rapid fault detection and the reconfiguration of the onboard control system to mitigate the consequences of such faults. Despite the wide variety of modern technical diagnostic methods, a comprehensive check of a robot's operational capability remains a highly challenging task, whose complexity stems from the need to identify the true causes of the observed malfunctions in the behavior of its functional subsystems.

The ontological model and self-diagnostic algorithms for an autonomous robot proposed in this article is designed to identify specific malfunctions based on an analysis of the cause-and-effect relationships between a set of actual symptoms and the sources of potential failures.

Unlike existing approaches based on analytical models and data analysis, ontological modeling technology can serve as a constructive basis for building a comprehensive self-diagnostic system that reflects the entire set of cause-and-effect relationships among possible failures and specific malfunctions of an autonomous robot, taking into account the hierarchical structure of the intelligent onboard control system and the specific tasks of each of its levels.

Experimental studies assessing the processing time for queries to the diagnostic ontology model have confirmed that the processing time for ten queries is approximately 60 ms, which is quite acceptable given the inertial parameters of mobile robots.

²³ <http://www.hermit-reasoner.com/>. Accessed April 30, 2026.

²⁴ https://franz.com/agraph/racer/racer_features.lhtml. Accessed April 30, 2026.

The use of an ontological model also opens up long-term prospects for improving the survivability and capability of autonomous robots operating under extreme conditions by reconfiguring the hardware and software components of the intelligent control system based on the results of self-diagnostic tests.

Authors' contributions

Yu.A. Bykovtsev—development of an example ontology model and simulation.

V.M. Lokhin—general guidance and review.

S.V. Manko—literature review and analysis.

M.P. Romanov—problem statement.

REFERENCES

1. Jang D.S., Cho D.H., Lee W.C., Ryu S.K., Jeong B., Hong M., Jung M., Kim M., Lee M., Lee S., Choi H.L. Unlocking Robotic Autonomy: A Survey on the Applications of Foundation Models. *International Journal of Control, Automation, and Systems (IJCAS)*. 2024;22(8):2341–2384. <https://doi.org/10.1007/s12555-024-0438-7>
2. Zghair N.A., Al-Araji A.S. A One Decade Survey of Autonomous Mobile Robot Systems. *International Journal of Electrical and Computer Engineering (IJECE)*. 2021;11(6):4891–4906. <http://doi.org/10.11591/ijece.v11i6.pp4891-4906>
3. Makarov I.M., Lokhin V.M. (Eds.). *Intellektual'nye sistemy avtomaticheskogo upravleniya (Intelligent Automatic Control Systems)*. Moscow: Fizmatlit; 2001, 576 p. (In Russ.). ISBN 5-9221-0162-5
4. Quan N.D., Ha H.H., Son L.D., An T.V., Hung N.T., Linh C.V. Control system architecture of an intelligent humanoid robot. *Vietnam Journal of Science and Technology*. 2022;60(6):1162–1178.
5. Zhang Y., Jiang J. Bibliographical Review on Reconfigurable Fault-Tolerant Control System. *Ann. Rev. Contr.* 2008;32(2): 229–252. <https://doi.org/10.1016/j.arcontrol.2008.03.008>
6. Hamilton K., Lane D., Taylor N., Brown K. Fault diagnosis on autonomous robotic vehicles with RECOVERY: an integrated heterogeneous-knowledge approach. In: *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation*. IEEE. 2001. V. 4. P. 3232–3237. <https://doi.org/10.1109/ROBOT.2001.933116>
7. Carlson J., Murphy R.R. How UGVs physically fail in the field. *IEEE Trans. Robotics*. 2005;21(3):423–437. <https://doi.org/10.1109/TRO.2004.838027>
8. Carlson J., Murphy R. Reliability analysis of mobile robots. In: *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*. IEEE. 2003. V. 1. P. 274–281. <https://doi.org/10.1109/ROBOT.2003.1241608>
9. Khalastchi E., Kaminka G.A., Kalech M., Lin R. Online anomaly detection in unmanned vehicles. In: *10th International Conference on Autonomous Agents and Multiagent Systems*. 2011. V. 1–3. P. 105–112.
10. Crestani D., Godary-Dejean K., Lapierre L. Enhancing Fault Tolerance of Autonomous Mobile Robots. *Robot. Autonom. Syst.* 2015;68:140–155. <https://doi.org/10.1016/j.robot.2014.12.015>
11. Hasan A., Tahavori M., Midtiby H. Model-Based Fault Diagnosis Algorithms for Robotic Systems. *IEEE Access*. 2023;11:2250–2258. <https://doi.org/10.1109/ACCESS.2022.3233672>
12. Abdilatif Alobaidy M., Alobaidy A., Abdul-Jabbar J., Saeed S. Faults Diagnosis in Robot Systems: A Review. *Al-Rafidain Engineering Journal (AREJ)*. 2020;25:164–175.
13. Golombek R., Wrede S., Hanheide M., Heckmann M. Online data-driven fault detection for robotic systems. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2011. P. 3011–3016. <https://doi.org/10.1109/IROS.2011.6095034>
14. Isermann R. Model-based fault-detection and diagnosis—Status and applications. *Ann. Rev. Contr.* 2005;29(1):71–85. <https://doi.org/10.1016/j.arcontrol.2004.12.002>
15. Chandola V., Banerjee A., Kumar V. Anomaly detection: A survey. *ACM Computing Surveys (CSUR)*. 2009;41(3):Article 15.
16. Xu S. A Survey of Knowledge-Based Intelligent Fault Diagnosis Techniques. *J. Phys.: Conf. Ser.* 2019;1187:032006. <https://doi.org/10.1088/1742-6596/1187/3/032006>
17. Khalastchi E., Kalech M., Rokach L. Sensor fault detection and diagnosis for autonomous systems. In: *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems*. 2013. P. 15–22. <https://doi.org/10.65109/RKJM1183>

18. Xu F., Liu X., Chen W., Zhou C., Cao B. Ontology-Based Method for Fault Diagnosis of Loaders. *Sensors*. 2018;18(3):729. <https://doi.org/10.3390/s18030729>
19. Alikhani A., Sharifi G. Application of a Model-Based Fault Detection Approach on a Spacecraft. *International Journal of Reliability, Risk and Safety: Theory and Application (IJRRS)*. 2020;3(2):19–26.
20. Khalastchi E., Kalech M. On Fault Detection and Diagnosis in Robotic Systems. *ACM Computing Surveys (CSUR)*. 2018;51(1):1–24. <https://doi.org/10.1145/3146389>
21. Quamar M., Nasir A. Review on Fault Diagnosis and Fault-Tolerant Control Scheme for Robotic Manipulators: Recent Advances in AI, Machine Learning, and Digital Twin. *arXiv preprint*. arXiv:2402.02980. 2024. <https://doi.org/10.48550/arXiv.2402.02980>
22. Brandstötter M., Hofbaur M., Steinbauer-Wagner G., Wotawa F. Model-based fault diagnosis and reconfiguration of robot drives. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2007. P. 1203–1209. <https://doi.org/10.1109/IROS.2007.4399092>
23. Pettersson O. Execution monitoring in robotics: A survey. *Robot. Auton. Syst.* 2005;53(2):73–88. <https://doi.org/10.1016/j.robot.2005.09.004>
24. Aguiar A.P. Multiple-model adaptive estimators: Open problems and future directions. In: *Proceedings of ECC'07 – European Control Conference*. IEEE. 2007. P. 5544–5545. <https://doi.org/10.23919/ECC.2007.7069028>
25. Lao L., Ellis M., Christofides P.D. Proactive fault-tolerant model predictive control. *AIChE J.* 2013;59(8):2810–2820. <https://doi.org/10.1002/aic.14074>
26. Yao J., Wang D., Cai D., Xu Y., Zhao Y. Fault-tolerant strategy and experimental study on compliance assembly of a redundant parallel six-component force sensor. *Sens. Actuators A: Phys.* 2018;272:114–124. <https://doi.org/10.1016/j.sna.2017.12.036>
27. Du Pu, Venkidasalapathy J.A., Venkateswaran S., Wilhite B., Kravaris C. Model-Based Fault Diagnosis and Fault Tolerant Control for Safety-Critical Chemical Reactors: A Case Study of an Exothermic Continuous Stirred-Tank Reactor. *Ind. Eng. Chem. Res.* 2023;62(34):13554–13571. <https://doi.org/10.1021/acs.iecr.3c01205>
28. Mogal S., Lalwani D. A brief review on fault diagnosis of rotating machineries. *Appl. Mech. Mater.* 2014;541-542:635–640. <https://doi.org/10.4028/www.scientific.net/AMM.541-542.635>
29. Li S., Xin Y., Li X., Wang J., Xu K. A review on the signal processing methods of rotating machinery fault diagnosis. In: *Proceedings IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. IEEE. 2019. P. 1559–1565. <https://doi.org/10.1109/ITAIC.2019.8785572>
30. Niu G., Dong X., Chen Y. Motor Fault Diagnostics Based on Current Signatures: A Review. *IEEE Trans. Instrum. Meas.* 2023;72:Article 3520919.
31. Niu G. Data Acquisition and Preprocessing. In: *Data-Driven Technology for Engineering Systems Health Management: Design Approach, Feature Construction, Fault Diagnosis, Prognosis, Fusion and Decisions*. Singapore: Springer; 2017. P. 49–99. https://doi.org/10.1007/978-981-10-2032-2_4
32. Islam A., Amin R., Sharan M.H., Ushno J.A., Islam K.M.T., Hossen D., Hossain M.S., Alam M.S. Designing an autonomous framework to detect and classify the fabric defects using wavelet transform and neural network. *AIP Conf. Proc.* 2023;2788:050003. <https://doi.org/10.1063/5.0148618>
33. Bai Y., Cheng W., Wen W., Liu Y. Application of time-frequency analysis in rotating machinery fault diagnosis. *Shock and Vibration*. 2023;2023:Article 9878228. <https://doi.org/10.1155/2023/9878228>
34. Surucu O., Gadsden S., Yawney J. Condition Monitoring using Machine Learning: A Review of Theory, Applications, and Recent Advances. *Expert Syst. Applicat.* 2023;221:19738. <https://doi.org/10.1016/j.eswa.2023.119738>
35. Manzoor S., Rocha Y.G., Joo S.H., Bae S.H., Kim E.J., Joo K.J., Kuc T.Y. Ontology-Based Knowledge Representation in Robotic Systems: A Survey Oriented toward Applications. *Appl. Sci.* 2021;11(10):4324. <https://doi.org/10.3390/APP11104324>
36. Diab M., Pomarlan M., Borgo S., et al. FailRecOnt – An ontology-based framework for failure interpretation and recovery in planning and execution. In: *Proceedings of the 2nd International Workshop on Ontologies for Autonomous Robotics*. 2021. P. 1–14. Available from URL: <https://hdl.handle.net/2117/357471>. Accessed July 19, 2025.
37. Yan B., Zhang T., Xie C. Fuzzy expert system for fault diagnosis of robotic assembly. In: *Proceedings of the World Congress on Intelligent Control and Automation*. IEEE. 2002. P. 445–449. <https://doi.org/10.1109/WCICA.2002.1022148>
38. De la Sen M., Minambres J.J., Garrido A.J., Almansa A., Soto J.S., Basic theoretical results for expert systems: application to the supervision of adaptation transients in planar robots. *Artif. Intell.* 2004;152(2):173–211. [https://doi.org/10.1016/S0004-3702\(03\)00136-X](https://doi.org/10.1016/S0004-3702(03)00136-X)
39. Lamine K.B., Kabanza F. History checking of temporal fuzzy logic formulas for monitoring behavior-based mobile robots. In: *Proceedings of the 12th IEEE International Conference on Tools with Artificial Intelligence*. IEEE. 2000. P. 312–319. <https://doi.org/10.1109/TAI.2000.889888>

40. Lamine K.B., Kabanza F. Using temporal fuzzy logic for monitoring behavior based mobile robots. In: *Proceedings of the IASTED Conference in Robotics and Applications*. 2000. P. 116–121.
41. Gao D., Wu Ch., Zhang B., Ma X. Signed Directed Graph and Qualitative Trend Analysis Based Fault. *Chinese J. Chem. Eng.* 2010;18(2):265–276. [https://doi.org/10.1016/S1004-9541\(08\)60352-3](https://doi.org/10.1016/S1004-9541(08)60352-3)
42. Berman A.F., Pavlov N.Yu., Nikolaichuk O.A. Method for synthesis and analysis of “failure trees” based on the concepts of mechanism and events kinetics. *Problemy analiza riska = Issues of Risk Analysis*. 2018;15(3):62–77 (in Russ.). <https://www.elibrary.ru/xwpjif>
43. Byun S., Papaalias M., Márquez F.P.G., Lee D. Fault-Tree-Analysis-Based Health Monitoring for Autonomous Underwater Vehicle. *J. Mar. Sci. Eng.* 2022;10(12):1855. <https://doi.org/10.3390/jmse10121855>
44. Bai B., Xie C., Liu X., Li W., Zhong W. Application of integrated factor evaluation–analytic hierarchy process–T-S fuzzy fault tree analysis in reliability allocation of industrial robot systems. *Appl. Soft Comp.* 2022;115:108248. <https://doi.org/10.1016/j.asoc.2021.108248>
45. Manzoor S., Rocha Y.G., Joo S.H., Bae S.H., Kim E.J., Joo K.J., Kuc T.Y. Ontology-Based Knowledge Representation in Robotic Systems: A Survey Oriented toward Applications. *Appl. Sci.* 2021;11(10):4324. <https://doi.org/10.3390/app11104324>
46. Olivares-Alarcos A., Bessler D., Khamis A., et al. A review and comparison of ontology-based approaches to robot autonomy. *The Knowledge Engineering Review*. 2019;34:e29. <https://doi.org/10.1017/S0269888919000237>
47. Gribova V.V., Shalfeeva E.A. The Ontology of processes diagnosis. *Ontologiya proektirovaniya = Ontology of Designing*. 2019;4(34):449–461 (in Russ.). <https://doi.org/10.18287/2223-9537-2019-9-4-449-461>
48. Timoshenko A.A., Zuev A.V., Mursalimov E.S., Gribova V.V., Inzartsev A.V. Description and diagnosis of malfunctions in autonomous uninhabited underwater vehicles based on ontologies. *Ontologiya proektirovaniya = Ontology of Designing*. 2022;3(45):310–324 (in Russ.). <https://doi.org/10.18287/2223-9537-2022-12-3-310-324>
49. Gruber T.R. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*. 1993;5(2):199–220. <https://doi.org/10.1006/knac.1993.1008>
50. Xu Y., Zhang S., Li Y., Liu J. Ontology-based Fault Diagnosis and Maintenance Process Generation of Electromechanical System. *Int. J. Performability Eng.* 2019;15(2):454–463. <https://doi.org/10.23940/ijpe.19.02.p10.454463>
51. Taye M. Understanding Semantic Web and Ontologies: Theory and Applications. *J. Computing*. 2010;2(6):182–192.
52. Su X.M., Ilebrikke L. A Comparative Study of Ontology Languages and Tools. In: Pidduck A.B., Ozsu M.T., Mylopoulos J., Woo C.C. (Eds.). *Advanced Information Systems Engineering (CAISE)*. 2002. P. 761–765. https://doi.org/10.1007/3-540-47961-9_62
53. Haarslev V., Moller R. Racer: A Core Inference Engine for the Semantic Web. In: *Proceedings of the 2nd International Semantic Web Conference (ISWC2003)*. 2003. P. 292–297.
54. Sirin E., Parsia B., Grau B.C., et al. Pellet: A Practical OWL-DL Reasoner. *J. Web Semantics*. 2007;5(2):51–53. <https://doi.org/10.1016/j.websem.2007.03.004>
55. Tsarkov D., Horrocks I. FaCT++ Description Logic Reasoner: System Description. In: *Proceedings of the 3rd International Joint Conference on Automated Reasoning*. 2006. P. 292–297. https://doi.org/10.1007/11814771_26
56. Tsarkov D., Riazanov A., Bechhofer S., Horrocks I. Using Vampire to Reason with OWL. In: *Proceedings of the 3rd International Semantic Web Conference (ISWC2004)*. 2004. P. 471–485. https://doi.org/10.1007/978-3-540-30475-3_33
57. Abburu S. A Survey on Ontology Reasoners and Comparison. *Int. J. Computer Applications*. 2012;57(17):33–39. Available from URL: <https://www.ijcaonline.org/archives/volume57/number17/9208-3748/>. Accessed July 19, 2025.
58. Rubin D.L., Noy N.F., Musen M.A., Protégé: A Tool for Managing and using Terminology in Radiology Applications. *J. Digit. Imaging*. 2007;20(1):34–46. <https://doi.org/10.1007/s10278-007-9065-0>
59. Shen B., Zhao S.Y., Wang J.H. Ontology-Based Fault Diagnosis Knowledge Representation of CNC Machine Tool. *Appl. Mech. Mater.* 2013;427–429:1372–1375. <https://doi.org/10.4028/www.scientific.net/AMM.427-429.1372>
60. Wang D., Tang W.H., Wu Q.H. Ontology-based fault diagnosis for power transformers. In: *IEEE PES General Meeting*. IEEE. 2010. P. 1–8. <https://doi.org/10.1109/PES.2010.5589575>
61. Dogmus Z., Erdem E., Patoglu V. RehabRobo-Onto: Design, development and maintenance of a rehabilitation robotics ontology on the cloud. *Robot. Comput. Integr. Manuf.* 2015;33:100–109. <https://doi.org/10.1016/j.rcim.2014.08.010>
62. Sokolov S.M. An ontological approach to the creation of robotic complexes with an increased degree of autonomy. *Izvestiya YuFU. Tekhnicheskie nauki = Izvestiya SFedU. Engineering Sciences*. 2022;1(225):42–58 (in Russ.). <https://doi.org/10.18522/2311-3103-2022-1-42-59>
63. Lebedev S.L., Panteleyev M.G. Ontology-driven design approach to development of situation awareness subsystem of intelligent agents. *Ontologiya proektirovaniya = Ontology of Designing*. 2016;3(21):297–316 (in Russ.).
64. Bayat B., Bermejo-Alonso J., Carbonera J., et al. Requirements for building an ontology for autonomous robots. *Industrial Robot*. 2016;43(5):469–480. <https://doi.org/10.1108/IR-02-2016-0059>

65. Sirin E., Parsia B., Grau B., Kalyanpur A., Katz Y. Pellet: a practical OWL-DL reasoner. *J. Web Semantics*. 2007;5(2):51–53. <https://doi.org/10.1016/j.websem.2007.03.004>
66. Alaya N., Ben Yahia S., Lamolle M. Towards Unveiling the Ontology Key Features Altering Reasoner Performances. *arXiv preprint*. arXiv:1509.087172015. <https://doi.org/10.48550/arXiv.1509.08717>

About the Authors

Yuri A. Bykovtsev, Cand. Sci. (Eng.), Assistant Professor, Department of Management Problems, Institute of Artificial Intelligence, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow, 119454 Russia). E-mail: bykovtsev@mirea.ru. Scopus Author ID 57302607300, ResearcherID KRQ-5339-2024, RSCI SPIN-code 9961-4437, <https://orcid.org/0009-0003-6671-5674>

Valery M. Lokhin, Dr. Sci. (Eng.), Professor, Department of Management Problems, Institute of Artificial Intelligence, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow, 119454 Russia). Laureate of the State Prize of the Russian Federation in the field of science and technology. Russian Federation State Prize Laureate in the field of education. Member of the Scientific Council of the Russian Academy of Sciences on robotics and mechatronics. Honored Worker of Science of the Russian Federation. E-mail: kpu-mirea@yandex.ru. Scopus Author ID 6602931640, <https://orcid.org/0000-0001-6708-9124>

Sergey V. Manko, Dr. Sci. (Eng.), Professor, Department of Management Problems, Institute of Artificial Intelligence, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow, 119454 Russia). Laureate of the State Prize of the Russian Federation in the field of science and technology. Russian Federation State Prize Laureate in the field of education. Member of the Scientific Council of the Russian Academy of Sciences on robotics and mechatronics. E-mail: manko@mirea.ru. Scopus Author ID 55761014700, RSCI SPIN-code 2070-1592, <https://orcid.org/0000-0002-6297-8894>

Mikhail P. Romanov, Dr. Sci. (Eng.), Professor, Academic Supervisor of the Institute of Artificial Intelligence, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow 119454 Russia). E-mail: m_romanov@mirea.ru. Scopus Author ID 14046079000, RSCI SPIN-code 5823-8795, <https://orcid.org/0000-0003-3353-9945>

Об авторах

Быковцев Юрий Алексеевич, к.т.н., доцент, кафедра проблем управления, Институт искусственного интеллекта, ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78). E-mail: bykovcev@mirea.ru. Scopus Author ID 57302607300, ResearcherID KRQ-5339-2024, SPIN-код РИНЦ 9961-4437, <https://orcid.org/0009-0003-6671-5674>

Лохин Валерий Михайлович, д.т.н., профессор, кафедра проблем управления, Институт искусственного интеллекта, ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78). Лауреат государственной премии РФ в области науки и техники. Лауреат премии Правительства РФ в области образования. Член научного Совета Российской академии наук по робототехнике и мехатронике. Заслуженный деятель науки РФ. E-mail: kpu-mirea@yandex.ru. Scopus Author ID 6602931640, <https://orcid.org/0000-0001-6708-9124>

Манько Сергей Викторович, д.т.н., профессор, кафедра проблем управления, Институт искусственного интеллекта, ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78). Лауреат премии Правительства Российской Федерации в области образования. Член научного Совета Российской академии наук по робототехнике и мехатронике. E-mail: manko@mirea.ru. Scopus Author ID 55761014700, SPIN-код РИНЦ 2070-1592, <https://orcid.org/0000-0002-6297-8894>

Романов Михаил Петрович, д.т.н., профессор, научный руководитель Института искусственного интеллекта, ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78). E-mail: m_romanov@mirea.ru. Scopus Author ID 14046079000, SPIN-код РИНЦ 5823-8795, <https://orcid.org/0000-0003-3353-9945>

Translated from Russian into English by L. Bychkova

Edited for English language and spelling by Thomas A. Beavitt