

М.О. Балабекова*, М.С. Шәмші

к.т.н, доцент, ЮКУ М. Ауэзова, Шымкент, Казахстан
магистрант, ЮКУ им. М. Ауэзова, Шымкент, Казахстан

*Автор для корреспонденции: dadyiba@mail.ru

ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА ДИАГНОСТИКИ СОСТОЯНИЯ БОРТОВОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ

Аннотация

В статье рассматривается задача диагностики состояния бортовой вычислительной системы в условиях изменяющихся нагрузок и неопределенности входных данных с целью повышения достоверности принимаемых решений. Для её решения предложена интеллектуальная система, основанная на использовании продукционных правил и элементов нечеткой логики. Выделены диагностические параметры, включающие коэффициент загрузки, время обработки информации, погрешность и наличие ошибок, на основе которых сформирована база знаний и реализован механизм логического вывода для классификации режимов функционирования. Новизна подхода заключается в сочетании продукционных правил и нечеткой интерпретации параметров, что позволяет учитывать размытые границы между состояниями системы. Разработана программная реализация, обеспечивающая ввод данных, выполнение анализа и визуализацию результатов. Результаты моделирования показали корректность работы алгоритма, устойчивость диагностических решений и снижение вероятности ложных переходов между состояниями.

Ключевые слова: бортовая вычислительная система, диагностика состояния, интеллектуальная система, база знаний, продукционные правила, нечеткая логика, логический вывод, программная реализация, анализ параметров, устойчивость системы.

Введение

Современные бортовые вычислительные системы (БВС) являются ключевыми элементами управления сложными техническими объектами и функционируют в условиях изменяющихся нагрузок, ограниченных вычислительных ресурсов и неопределенности входных данных [1]. В этих условиях возрастает значимость задач своевременной и достоверной диагностики состояния системы. Традиционные методы диагностики, основанные на фиксированных пороговых значениях параметров, обладают ограниченной гибкостью и не позволяют в полной мере учитывать переходные режимы функционирования. В результате это может приводить к снижению точности оценки состояния системы, особенно при наличии неполной или искаженной информации. В качестве альтернативы применяются интеллектуальные методы, основанные на использовании базы знаний и механизмов логического вывода. Экспертные системы позволяют формализовать знания специалистов в виде продукционных правил и обеспечивают интерпретируемость принимаемых решений, что имеет принципиальное значение для диагностических задач [2, 3]. Дальнейшее развитие данного направления связано с применением нечеткой логики, обеспечивающей возможность обработки неопределенных и лингвистически описываемых данных [4 - 6]. Использование функций принадлежности позволяет отказаться от жестких границ между состояниями и перейти к более гибкой оценке поведения системы, что особенно важно для объектов с непрерывно изменяющимися параметрами. Современные интеллектуальные системы диагностики все чаще строятся на основе сочетания продукционных правил и методов нечеткого вывода [7, 8]. Такой подход позволяет повысить устойчивость принимаемых решений при варьировании входных параметров и учитывать неопределенность исходных данных. Вместе с тем, несмотря на наличие значительного числа исследований в данной

области, задачи построения компактных и интерпретируемых моделей, ориентированных на анализ функционирования бортовых вычислительных систем, остаются недостаточно проработанными. В частности, требуется разработка подходов, сочетающих простоту реализации, наглядность и возможность расширения базы знаний.

На основании вышеизложенного в статье рассматривается задача разработки интеллектуальной системы диагностики состояния бортовой вычислительной системы на основе продукционных правил и элементов нечеткой логики с последующей оценкой её эффективности.

Материалы и методы

Теоретический анализ. Бортовая вычислительная система рассматривается как динамическая система, функционирование которой определяется совокупностью входных данных, внутреннего состояния и управляющих воздействий. Обобщённая модель может быть представлена в виде зависимости:

$$Y(t) = F(X(t), Z(t), \theta), \quad (1)$$

где $X(t)$ – совокупность входных данных, поступающих от датчиков;

$Z(t)$ – текущее внутреннее состояние системы;

θ -параметры, характеризующие особенности ее функционирования;

$Y(t)$ – формируемые управляющие воздействия.

Для анализа состояния системы выделены ключевые диагностические параметры [1]:

1) коэффициент загрузки вычислительной системы U ; 2) время обработки информации T_{proc} ; 3) величина вычислительной погрешности Δ ; 4) наличие ошибок и сбоев в работе отдельных модулей.

Выбранные показатели позволяют судить как о быстродействии системы, так и о её устойчивости к внешним и внутренним воздействиям. На основе указанных параметров сформирована база знаний в виде продукционных правил [2, 3], описывающих различные режимы функционирования системы (таблица 1).

Таблица 1 - База продукционных правил

№	Условие (ЕСЛИ)	Заключение (ТО)	Состояние
1	$U < 0.7$ И $\Delta < 0.05$ И $T_{proc} < 100$ мс	система работает корректно	Нормальное
2	$0.7 \leq U \leq 0.85$ И $T_{proc} \leq 120$ мс	допустимая загрузка	Рабочее
3	$U > 0.85$ И $T_{proc} > 120$ мс	перегрузка системы	Перегруженное
4	$U > 0.9$ И $T_{proc} > 150$ мс	критическая перегрузка	Критическое
5	$\Delta > 0.1$	высокая погрешность	Нестабильное
6	$\Delta > 0.05$ И $\Delta \leq 0.1$	умеренная погрешность	Допустимое
7	$T_{proc} > 150$ мс	задержка обработки	Замедленное
8	ошибки передачи данных	нарушение обмена	Сбой
9	отказ вычислительного модуля	система не функционирует	Отказ
10	$U > 0.8$ И ошибки передачи данных	перегрузка с ошибками	Критическое
11	$U < 0.6$ И $T_{proc} > 130$ мс	неэффективное использование ресурсов	Неэффективное
12	$\Delta > 0.08$ И ошибки	нестабильная работа	Нестабильное

При анализе функционирования бортовой вычислительной системы значения параметров изменяются в широких пределах и не всегда могут быть однозначно отнесены к «нормальным» или «критическим», вследствие чего границы между состояниями оказываются размытыми. Для описания подобных ситуаций параметры интерпретируются в виде лингвистических переменных («низкий», «средний», «высокий»), что позволяет более гибко оценивать состояние системы (таблица 2). То есть применение элементов нечеткой логики обеспечивает учёт неопределённости входных данных [4 – 6].

Таблица 2 - Лингвистические переменные диагностических параметров

Параметр	Обозначение	Лингвистические значения
Коэффициент загрузки	U	низкий, средний, высокий
Время обработки	T_{proc}	малое, допустимое, большое
Погрешность	Δ	малая, средняя, высокая
Ошибки	-	отсутствуют, присутствуют

Алгоритм интеллектуального анализа состояния бортовой вычислительной системы ориентирован на последовательную обработку диагностических параметров с последующим формированием заключения о текущем режиме её функционирования. Представленная продукционная база знаний на основе нечетких значений позволяет установить соответствие между значениями параметров и состоянием системы.

Экспериментальная часть. Для проверки предложенного подхода была разработана модель интеллектуальной системы диагностики и осуществлена ее программная реализация. Программная модель реализована по модульному принципу и включает взаимосвязанные функциональные компоненты (рисунок 1).



Рисунок 1 - Структура программной модели диагностируемой системы

Ввод исходных данных предусматривает задание диагностических параметров - коэффициента загрузки, времени обработки, погрешности и наличия ошибок - с контролем корректности вводимых значений. Предварительная обработка направлена на приведение данных к требуемому виду и, при необходимости, вычисление дополнительных характеристик. Основу системы составляет блок логического вывода, в котором на основе продукционных правил формируется диагностическое заключение. Отображение результатов осуществляется с использованием графического интерфейса, обеспечивающего ввод параметров, запуск анализа и визуализацию полученных данных, включая функции принадлежности.

Реализация программного обеспечения выполнена на языке Python с использованием библиотеки Tkinter для создания графического интерфейса, а также Matplotlib для визуализации функций принадлежности. Программа реализована в виде оконного приложения с графическим интерфейсом пользователя (рисунок 2), включающего вкладки для ввода данных, выполнения анализа и отображения результатов.

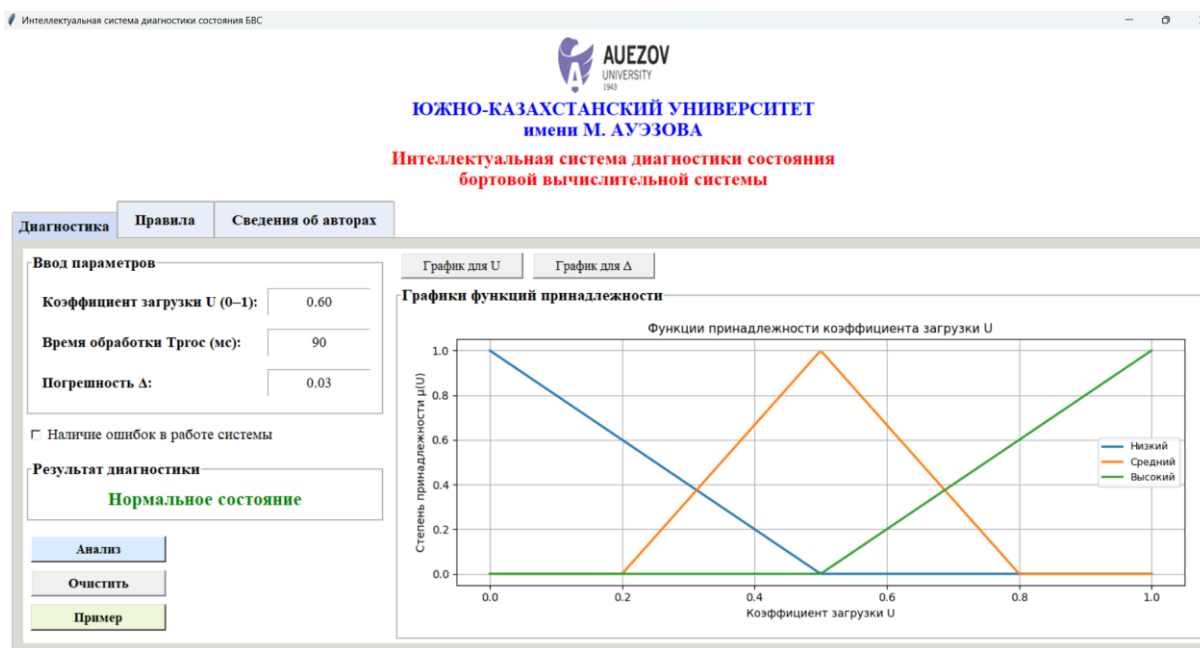


Рисунок 2 - Интерфейс программной реализации интеллектуальной системы диагностики состояния бортовой вычислительной системы

Основная вкладка «Диагностика» предназначена для задания параметров системы (коэффициента загрузки, времени обработки, погрешности и наличия ошибок) и запуска процедуры анализа. Управление осуществляется с помощью кнопок, обеспечивающих выполнение диагностики, сброс данных, загрузку тестовых значений и построение графиков функций принадлежности. Результаты отображаются в текстовой форме с указанием текущего состояния системы (нормальное, рабочее, перегруженное или критическое) и сопровождаются визуальной индикацией. Дополнительно реализована графическая интерпретация параметров, позволяющая оценить их принадлежность к лингвистическим значениям. Вкладка «Правила» содержит базу знаний, а вкладка «Сведения об авторах» - информацию о разработчиках системы.

В рамках построения эксперимента использованы параметры, с заданными диапазонами согласно типичным характеристикам вычислительных систем:

- U : от 0.1 до 1.0;
- T_{proc} : от 50 до 200 мс;
- Δ : от 0.01 до 0.1;
- ошибки: 0 - отсутствуют, 1 - присутствуют.

Для повышения наглядности и воспроизводимости эксперимента были сформированы конкретные наборы входных данных, представленные в таблице 3.

В ходе эксперимента рассматривались основные режимы функционирования системы: нормальный, рабочий, перегруженный, нестабильный и критический, отличающиеся значениями коэффициента загрузки, времени обработки, погрешности и наличием ошибок.

Таблица 3 - Исходные данные для эксперимента

№	U	T_{proc} (мс)	Δ	Ошибки	Результат системы
1	0.30	70	0.02	0	Нормальное
2	0.45	90	0.03	0	Нормальное
3	0.60	110	0.04	0	Рабочее
4	0.75	130	0.05	0	Перегруженное
5	0.85	150	0.06	0	Перегруженное
6	0.55	120	0.08	0	Нестабильное
7	0.70	140	0.09	1	Критическое
8	0.90	180	0.10	1	Критическое

Анализ результатов моделирования показал, что разработанная система адекватно реагирует на изменение входных параметров и формирует диагностическое заключение в соответствии с заданной базой знаний. При увеличении коэффициента загрузки и времени обработки происходит переход системы от нормального режима к рабочему и далее к перегруженному состоянию. Повышение погрешности сопровождается идентификацией нестабильного режима, тогда как появление ошибок приводит к распознаванию критического состояния. Во всех исследованных сценариях работа алгоритма соответствовала ожидаемой логике функционирования. Установлено, что наибольшее влияние на итог диагностики оказывают коэффициент загрузки и время обработки информации, в то время как погрешность в большей степени отражает устойчивость функционирования системы.

Результаты и их обсуждение

Результаты моделирования показали, что разработанная интеллектуальная система обеспечивает корректное определение состояния бортовой вычислительной системы при различных сочетаниях диагностических параметров. В ходе эксперимента анализировались режимы, отличающиеся коэффициентом загрузки, временем обработки информации, величиной вычислительной погрешности и наличием ошибок передачи данных. Исходные данные и соответствующие результаты диагностики приведены в таблице 3.

Анализ экспериментальных данных показал, что при низких значениях коэффициента загрузки и времени обработки система функционирует в устойчивом режиме. Для экспериментов №1 и №2 диагностический модуль определил состояние как нормальное, поскольку коэффициент загрузки не превышал 0.45, а время обработки оставалось менее 100 мс. При этом вычислительная погрешность находилась в пределах допустимых значений, а ошибки передачи данных отсутствовали. Подобный режим соответствует стабильной работе вычислительной системы при допустимой нагрузке [7].

При увеличении коэффициента загрузки и времени обработки наблюдалось изменение режима функционирования системы. В эксперименте №3 состояние было определено как рабочее, что соответствует допустимому уровню загрузки вычислительных ресурсов. Дальнейший рост параметров приводил к переходу системы в перегруженное состояние. Так, в экспериментах №4 и №5 при коэффициенте загрузки 0.75–0.85 и времени обработки 130–150 мс диагностическая система фиксировала признаки перегрузки. В этих режимах возрастает вероятность задержек обработки информации и снижается эффективность распределения вычислительных ресурсов [7, 8].

Сравнение предложенного подхода с традиционным пороговым методом показало различие в характере формирования диагностических решений (таблица 4). В пороговых системах изменение состояния происходит скачкообразно при достижении фиксированных значений параметров. Использование элементов нечеткой логики позволяет реализовать более плавный переход между режимами функционирования. Это особенно заметно при анализе

коэффициента загрузки системы, представленного на рисунке 3. В области граничных значений изменение диагностического состояния происходит последовательно, без резких переходов между режимами.

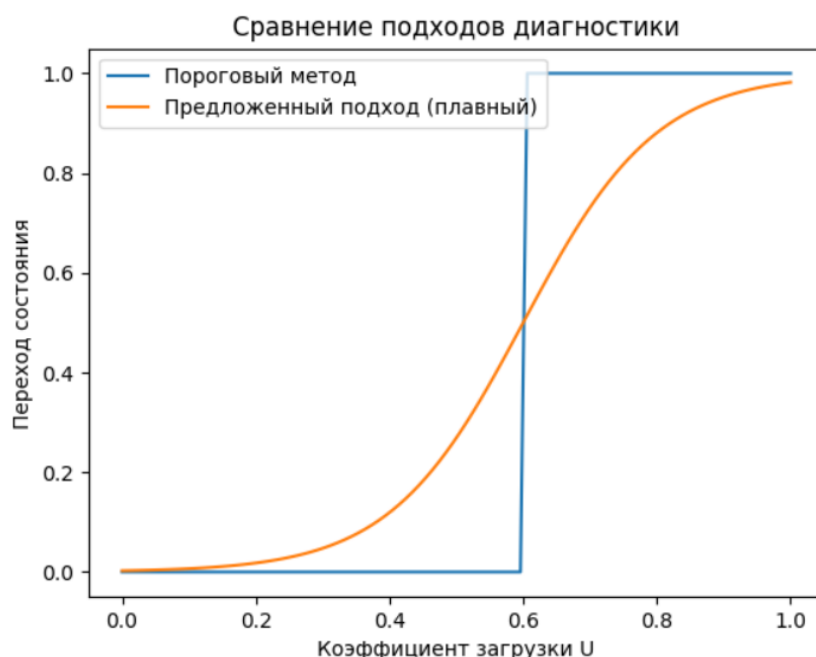


Рисунок 3 - Сравнение порогового и предложенного подходов по коэффициенту загрузки системы

Таблица 4 - Сравнение методов диагностики

Критерий	Пороговый метод	Предложенный подход
Учет неопределенности	отсутствует	учитывается
Гибкость принятия решений	низкая	высокая
Чувствительность к изменениям	резкая(скачкообразная)	плавная
Точность диагностики	зависит от порогов	стабильная
Возможность расширения	ограниченная	высокая
Интерпретируемость	высокая	высокая

Дополнительный анализ был выполнен для оценки влияния вычислительной погрешности на результат диагностики. Установлено, что увеличение значения Δ оказывает существенное влияние на устойчивость функционирования системы. В эксперименте №6 при значении погрешности 0.08 система определила состояние как нестабильное даже при отсутствии ошибок передачи данных. Это свидетельствует о том, что вычислительная погрешность может рассматриваться как показатель качества обработки информации и устойчивости работы отдельных вычислительных модулей [7]. Зависимость диагностического решения от изменения вычислительной погрешности представлена на рисунке 4.

Наиболее неблагоприятные режимы наблюдались при совместном увеличении коэффициента загрузки, времени обработки и наличии ошибок передачи данных. В экспериментах №7 и №8 система классифицировала состояние как критическое. В данных условиях происходит одновременное ухудшение нескольких диагностических параметров, что приводит к снижению устойчивости функционирования вычислительной среды и повышению вероятности отказов отдельных модулей [8]. Полученные результаты показывают, что наличие

ошибок существенно усиливает влияние остальных параметров и ускоряет переход системы в аварийный режим.

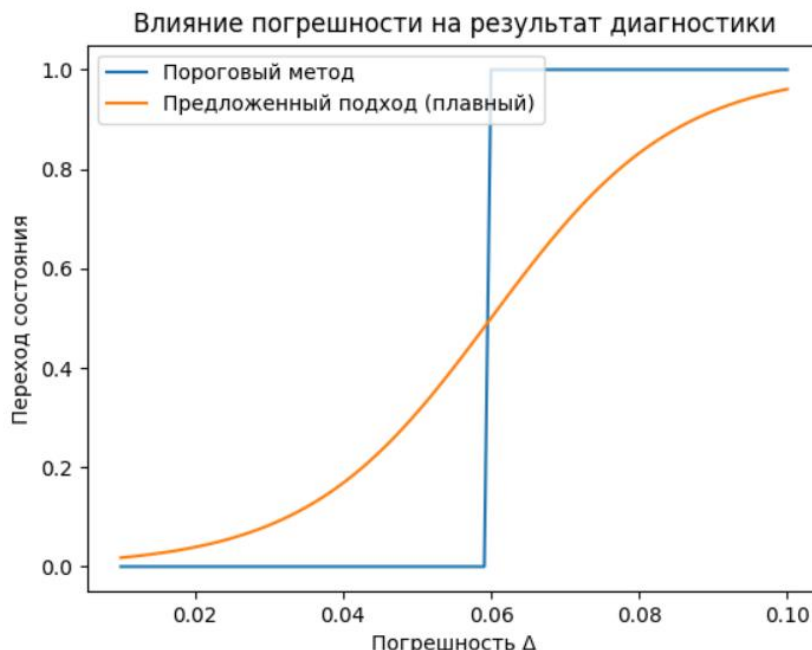


Рисунок 4 - Влияние вычислительной погрешности Δ на результат диагностики

Проведенный анализ показал, что применение продукционных правил совместно с элементами нечеткой логики позволяет учитывать неопределенность входных данных и особенности переходных режимов работы системы. В отличие от пороговых методов диагностическое решение формируется более устойчиво при изменении параметров вблизи граничных значений. Использование базы знаний обеспечивает интерпретируемость результатов диагностики и возможность дальнейшего расширения системы за счет добавления новых правил и диагностических признаков.

Заключение

В настоящей работе разработана интеллектуальная система диагностики состояния бортовой вычислительной системы, основанная на использовании продукционных правил и элементов нечеткой логики. Предложенный подход позволяет формализовать процесс анализа и установить связь между диагностическими параметрами и состояниями системы.

Выделение ключевых параметров (коэффициента загрузки, времени обработки, погрешности и наличия ошибок) послужило основой для построения процедуры диагностики. Представление знаний в виде продукционных правил позволило описать характерные режимы функционирования и задать соответствие между значениями параметров и диагностическими состояниями. Использование нечеткой логики обеспечивает переход от жесткого разделения состояний к более гибкой интерпретации результатов, что особенно важно при анализе систем с непрерывно изменяющимися параметрами. Реализованный алгоритм формирует диагностическое заключение при невысоких вычислительных затратах. Программная реализация подтверждает применимость разработанного решения и обеспечивает удобное взаимодействие пользователя с системой. Результаты моделирования показали согласованность принимаемых решений при изменении входных параметров. Установлено, что определяющее влияние оказывают коэффициент загрузки и время обработки, тогда как погрешность и наличие ошибок отражают переход системы в нестабильные и критические состояния. Сравнение с пороговыми методами выявило более устойчивое поведение системы,

особенно вблизи граничных значений, что позволяет повысить достоверность диагностики и снизить вероятность ложных переходов между состояниями.

Разработанное решение может быть использовано при создании интеллектуальных систем мониторинга и диагностики сложных технических объектов.

Список литературы

1. Kopetz H. Real-Time Systems: Design Principles for Distributed Embedded Applications. 2nd ed. New York: Springer, 2011, 370 p. doi: 10.1007/978-1-4419-8237-7
2. Jackson P. Introduction to Expert Systems. 3rd ed. Harlow: Addison-Wesley, 2012, 560 p.
3. Giarratano J., Riley G. Expert Systems: Principles and Programming. 4th ed. Boston: Cengage Learning, 2015, 600 p.
4. Siler W., Buckley J.J. Fuzzy Expert Systems and Fuzzy Reasoning. New York: Wiley, 2015, 425 p. doi: 10.1002/9781119028192
5. Ross T.J. Fuzzy Logic with Engineering Applications. 3rd ed. Chichester: Wiley, 2016, 600 p. doi: 10.1002/9781119994374
6. Pedrycz W., Gomide F. Fuzzy Systems Engineering: Toward Human-Centric Computing. New York: Wiley, 2017, 500 p. doi: 10.1002/9781119288411
8. Jiang W., Xie C., Zhuang M., Tang Y. Failure detection in complex systems based on fuzzy logic and data-driven models. // IEEE Access, 2019, V. 7, P. 113535–113546. doi: 10.1109/ACCESS.2019.2935202
9. Zhao R., Yan R., Chen Z., Mao K., Wang P., Gao R.X. Deep learning and its applications to machine health monitoring. // Mechanical Systems and Signal Processing, 2019, V. 115, P. 213–237. doi: 10.1016/j.ymssp.2018.05.050

References

1. Kopetz H. Real-Time Systems: Design Principles for Distributed Embedded Applications. 2nd ed. New York: Springer, 2011, 370 p. doi: 10.1007/978-1-4419-8237-7
2. Jackson P. Introduction to Expert Systems. 3rd ed. Harlow: Addison-Wesley, 2012, 560 p.
3. Giarratano J., Riley G. Expert Systems: Principles and Programming. 4th ed. Boston: Cengage Learning, 2015, 600 p.
4. Siler W., Buckley J.J. Fuzzy Expert Systems and Fuzzy Reasoning. New York: Wiley, 2015, 425 p. doi: 10.1002/9781119028192
5. Ross T.J. Fuzzy Logic with Engineering Applications. 3rd ed. Chichester: Wiley, 2016, 600 p. doi: 10.1002/9781119994374
6. Pedrycz W., Gomide F. Fuzzy Systems Engineering: Toward Human-Centric Computing. New York: Wiley, 2017, 500 p. doi: 10.1002/9781119288411
10. Jiang W., Xie C., Zhuang M., Tang Y. Failure detection in complex systems based on fuzzy logic and data-driven models. // IEEE Access, 2019, V. 7, P. 113535–113546. doi: 10.1109/ACCESS.2019.2935202
11. Zhao R., Yan R., Chen Z., Mao K., Wang P., Gao R.X. Deep learning and its applications to machine health monitoring. // Mechanical Systems and Signal Processing, 2019, V. 115, P. 213–237. doi: 10.1016/j.ymssp.2018.05.050

М.О. Балабекова*, М.С. Шәмші

*т.ғ.к., доцент, dadyiba@mail.ru, М. Әуезов атындағы ОҚУ, Шымкент, Қазақстан
магистрант, medet.shamshi51@mail.ru, М. Әуезов атындағы ОҚУ, Шымкент, Қазақстан

БОРТТЫҚ ЕСЕПТЕУ ЖҮЙЕСІНІҢ ЖАҒДАЙЫН ДИАГНОСТИКАЛАУҒА АРНАЛҒАН ИНТЕЛЛЕКТУАЛДЫ ЖҮЙЕ

Түйін

Мақалада өзгермелі жүктемелер және кіріс деректерінің белгісіздігі жағдайында борттық есептеу жүйесінің күйін диагностикалау мәселесі қарастырылады, оның мақсаты – қабылданатын шешімдердің сенімділігін арттыру. Осы міндетті шешу үшін продукциялық ережелер мен анық емес логика элементтерін пайдалануға негізделген интеллектуалды жүйе ұсынылған. Диагностикалық параметрлер ретінде жүктеме коэффициенті, ақпаратты өңдеу уақыты, қателік шамасы және ақаулардың болуы анықталып, солар негізінде білімдер базасы құрылып, жүйенің жұмыс режимдерін жіктеуге арналған логикалық қорытынды шығару механизмі жүзеге асырылған. Ұсынылған тәсілдің жаңалығы – продукциялық ережелер мен параметрлердің анық емес интерпретациясын үйлестіруінде, бұл жүйе күйлері арасындағы нақты емес шекараларды ескеруге мүмкіндік береді. Деректерді енгізу, талдау жүргізу және нәтижелерді визуализациялауды қамтамасыз ететін бағдарламалық жүзеге асыру әзірленді. Модельдеу нәтижелері алгоритмнің дұрыс жұмыс істейтінін, диагностикалық шешімдердің тұрақтылығын және жүйе күйлері арасындағы жалған ауысулар ықтималдығының төмендегенін көрсетті.

Кілттік сөздер: борттық есептеу жүйесі, күй диагностикасы, интеллектуалды жүйе, білімдер базасы, продукциялық ережелер, анық емес логика, логикалық қорытынды, бағдарламалық жүзеге асыру, параметрлерді талдау, жүйе тұрақтылығы.

M.O. Balabekova*, M.S. Shamshi

*Cand.Tech.Sci., Associate Professor, dadyiba@mail.ru, M. Auezov SKU, Shymkent, Kazakhstan
Master's Student, medet.shamshi51@mail.ru, M. Auezov SKU, Shymkent, Kazakhstan

INTELLIGENT SYSTEM FOR DIAGNOSTICS OF THE STATE OF AN ONBOARD COMPUTING SYSTEM

Abstract

The paper addresses the problem of diagnosing the state of an onboard computing system under conditions of varying loads and uncertainty in input data, with the aim of improving the reliability of decision-making. To solve this problem, an intelligent system based on production rules and elements of fuzzy logic is proposed. Diagnostic parameters are identified, including the load factor, information processing time, error magnitude, and the presence of faults. Based on these parameters, a knowledge base is formed and an inference mechanism is implemented to classify operating modes. The novelty of the proposed approach lies in the combination of production rules with fuzzy interpretation of parameters, which allows taking into account the blurred boundaries between system states. A software implementation has been developed that provides data input, analysis, and visualization of results. The simulation results demonstrate the correctness of the algorithm, the stability of diagnostic decisions, and a reduction in the probability of false transitions between system states.

Keywords: onboard computing system, state diagnostics, intelligent system, knowledge base, production rules, fuzzy logic, inference mechanism, software implementation, parameter analysis, system stability.