

## Подбор параметров нагрузочного тестирования систем мониторинга с использованием нечеткой логики

*А. А. Темичев, Р. А. Файзрахманов*

*Пермский национальный исследовательский политехнический университет*

**Аннотация:** В статье предлагается метод подбора параметров системы нагрузочного тестирования для оценки производительности распределенной системы мониторинга, с целью перевода ее в требуемое состояние нагрузки. Предлагаемый метод основывается на использовании интеллектуальных технологий, а именно теории нечетких множеств. Новшеством предлагаемого метода по сравнению с существующими, заключается в одновременном использовании технологии обратной связи, которая позволяет получать информацию о состоянии системы, и методах нечеткой логики, позволяющих гибко реагировать на изменение состояния системы мониторинга, без участия человека, используя информацию, полученную посредством обратной связи, тем самым постоянно поддерживая требуемую нагрузку на систему мониторинга. Для демонстрации работы, предложенный метод был реализован в составе демонстрационной системы нагрузочного тестирования. Далее была произведена экспериментальная проверка реализованной системы на простейшей системе мониторинга. Результаты эксперимента, приведенные в заключительной части статьи, показали, что метод формирования нагрузки, предложенный в статье, позволяет стабильно удерживать требуемую нагрузку на сервер системы мониторинга, при различных изменениях состояния ее работы.

**Ключевые слова:** нечеткая логика, обратная связь, нагрузочное тестирование, система мониторинга

### Введение

Современное развитие промышленного и хозяйственного сектора сейчас трудно представить без использования систем учета, получивших название системы удаленного мониторинга (СМ). Высокое качество работы таких систем является их неотъемлемой составляющей, т.к. от достоверности и своевременности данных, получаемых этими системами, зависят эффективность функционирования производства, жизни людей, экономические показатели [1] и т.д., таким образом, тестирование программных и аппаратных компонентов СМ перед их внедрением, является одной из самых важных и ответственных задач на этапе их жизненного цикла.

С целью повышения качества работы таких систем выполняют оценку их производительности, для этого производятся различные виды

тестирования [2]: тестирование производительности, нагрузочное тестирование, стресс тестирование и т.п.

Одним из наиболее используемых видов тестирования высоконагруженных систем, таких как СМ, является нагрузочное тестирование, используемое, когда необходимо узнать степень нагрузки на систему, выражающуюся в таких показателях, как процент использования центрального процессора, сети и памяти серверной ЭВМ СМ.

Информация такого типа, является очень важной т.к. каждый объект, на который внедряется СМ, является уникальным и требует своей конфигурации, определяемой такими параметрами как: количество программируемых логических контроллеров-регистраторов (ПЛК), пропускная способность сети, наличие серверных и клиентских ЭВМ определенных конфигураций и т.д., а это означает, что и конфигурация СМ для таких объектов требуют индивидуального подхода.

Для нагрузочного тестирования систем используют различные программные средства, но, несмотря на проработанность таких средств, им присущи некоторые недостатки, а именно:

1. Большой объем ручной работы по конфигурированию и управлению процессом тестирования,
2. Необходимость остановки тестового сценария для изменения параметров тестового сценария,
3. Трудности использования информации, полученной в процессе предыдущих тестов, в последующих тестовых сценариях.

Одним из способов устранения вышеуказанных недостатков является использование метода на основе интеллектуальных технологий [3,4], в данном случае аппарата теории нечетких множеств, с использованием обратной связи (ОС), который позволяет обеспечить управление процессом тестирования без непосредственного участия человека на основе задания

---

лишь некоторых опорных параметров тестового сценария и использовании метрик производительности сервера СМ. Еще одним преимуществом использования аппарата нечеткой логики является то, что данные о характеристиках параметров при тестировании СМ часто носят расплывчатый характер, а процессы и зависимости имеют сложно предсказуемое и нелинейное поведение, и требуют продолжительного и трудоемкого определения, т.к. каждая тестируемая СМ имеет различное внутреннее строение процедур получения, обработки и хранения данных. Таким образом, применение нечеткой логики позволяет формализовать параметры тестирования и воздействия на СМ, повысить эффективность их подбора в процессе тестирования, и что немаловажно, рассмотреть их комплексно, а также существенно упростить сам процесс тестирования с точки зрения трудозатрат.

Примером задачи нагрузочного тестирования, которая может решаться с использованием предложенного метода, является выяснение, с какой интенсивностью СМ может принимать и обрабатывать входные данные с множества ПЛК, при заданных параметрах длины сообщения и количества ПЛК на каналах связи, и при определенной загрузке центрального процессора серверной ЭВМ СМ.

### **Синтез блока управления нагрузкой**

Структурная схема взаимодействия всех компонентов при тестировании СМ, с учетом внутренней структуры блока управления нагрузкой с использованием нечеткой логики и технологии обратной связи представлена на рис. 1, где укрупненные блоки *US*, *TSS*, *MSS* соответственно обозначают границы работы оператора тестирования, системы тестирования и тестируемой СМ. Стрелки, идущие от блока *User*, обозначают входные данные, задаваемые оператором тестирования. Стрелка идущая от блока *MS* к блоку *FB\_CPU* обозначает ОС, идущую от СМ к системе тестирования.

---

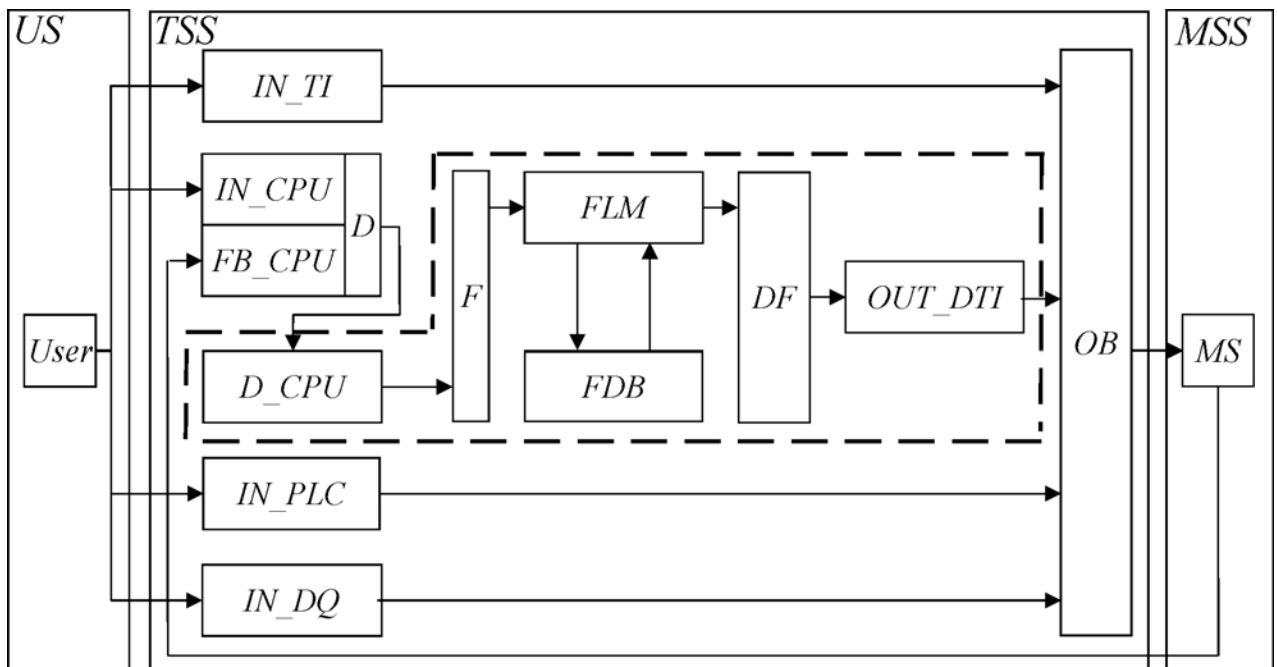


Рис. 1. – Структурная схема взаимодействия компонентов тестирования СМ

Жирной пунктирной линией обозначен блок нечеткого вывода, включающий в себя, следующие компоненты:

1.  $OUT\_DTI$  – выходная лингвистическая переменная, отражающая коэффициент приращения значения интенсивности передачи данных на каналах связи к СМ (пакеты/с). Расчет выходной интенсивности ( $OUT\_TI$ ) вычисляется по формуле:

$$OUT\_TI_i = Round(OUT\_TI_{i-1} \cdot OUT\_DTI_i),$$

где  $OUT\_DTI = [0-10]$   $i = \overline{0, n}$ .

2.  $D\_CPU$  – входная лингвистическая переменная отражающая расхождение требуемого показателя нагрузки на ЦП сервера СМ, от получаемого по обратной связи и вычисляется по формуле:

$$D\_CPU = IN\_CPU - FD\_CPU,$$

где  $D\_CPU = [0-100]$ .

3.  $F$  – блок фаззификации.
4.  $FLM$  – блок нечеткого вывода.
5.  $FDB$  – база нечетких правил.

6.  $DF$  – блок дефаззификации.

К элементам не входящим в блок нечеткого управления нагрузкой, но влияющим на формирование выходной последовательности данных относятся:

1.  $IN\_PLC$  – параметр, определяющий количество имитируемых ПЛК на каналах связи с СМ.

2.  $IN\_DQ$  – параметр, определяющий длину сообщения, в строках.

3.  $IN\_TI$  – параметр, определяющий начальную интенсивность передачи пакетов.

4.  $FB\_CPU$  – параметр, определяющий процент загрузки ЦП сервера СМ, получаемого по обратной связи, с сервера СМ.

5.  $IN\_CPU$  – параметр, определяющий целевое значение загрузки ЦП сервера СМ.

6.  $D$  – блок, вычисляющий расхождения между параметрами  $IN\_CPU$  и  $FB\_CPU$ .

7.  $OB$  – блок, формирующий и отправляющий в СМ сформированные пакеты данных, с заданными параметрами количества ПЛК, длины сообщения и интенсивностью.

Входная лингвистическая переменная  $D\_CPU$  включает в себя семь термов: большое отрицательное ( $bn$ ), средне отрицательное ( $mn$ ), мало отрицательное ( $ln$ ), близко к нулю ( $z$ ), мало положительное ( $lp$ ), средне положительное ( $mp$ ), большое положительное ( $bp$ ).

Выходная лингвистическая переменная  $OUT\_DTI$  включает себя семь термов: близкий к нулю ( $z$ ), очень малый ( $vl$ ), малый ( $l$ ), средне малый ( $ml$ ), средний ( $m$ ), средне большое ( $mb$ ), большое ( $b$ ).

После определения количества термов каждой лингвистической переменной и распределения функций принадлежности были синтезированы правила вывода вида  $R_i : \text{ЕСЛИ } D\_CPU = A_{il}, \text{ ТО } OUT\_DTI = B_i$ , где  $A_{ij}$  –

---

множества функций принадлежности  $i$ -му терму  $j$ -й переменной;  $B_i$  – множество функций принадлежности  $i$ -му терму выходной переменной.

В данном случае сформированная база знаний нечеткого блока управления нагрузкой состоит из семи правил:

1.  $R_1$ : ЕСЛИ  $D\_CPU = bn$ , ТО  $OUT\_DTI = z$
2.  $R_2$ : ЕСЛИ  $D\_CPU = mn$ , ТО  $OUT\_DTI = vl$
3.  $R_3$ : ЕСЛИ  $D\_CPU = ln$ , ТО  $OUT\_DTI = lpi$
4.  $R_4$ : ЕСЛИ  $D\_CPU = z$ , ТО  $OUT\_DTI = ml$
5.  $R_5$ : ЕСЛИ  $D\_CPU = lp$ , ТО  $OUT\_DTI = m$
6.  $R_6$ : ЕСЛИ  $D\_CPU = mp$ , ТО  $OUT\_DTI = mb$
7.  $R_7$ : ЕСЛИ  $D\_CPU = lp$ , ТО  $OUT\_DTI = b$

Для построения функций принадлежности входных и выходных лингвистических переменных, используемых в базах правил, были выбраны трапециевидные и треугольные функции принадлежности, широко используемые в различных исследованиях [5,6,7,8]. Треугольная функция принадлежности в общем случае аналитически задается с помощью следующего выражения:

$$f_{\Delta}(x, a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq c \leq x \\ 0, & c \leq x \end{cases},$$

где  $a, b, c$  — некоторые числовые параметры, принимающие произвольные действительные значения и упорядоченные отношением  $a \leq b \leq c$ .

Трапециевидная функция принадлежности в общем случае может быть задана аналитически следующим выражением:

$$f_{\Delta}(x, a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{c-x}{c-b}, & c \leq x \leq d \\ 0, & d \leq x \end{cases}$$

где  $a$  и  $d$  – параметры, характеризующие нижнее основание трапеции;  $b$  и  $c$  – параметры, характеризующие верхнее основание трапеции и упорядоченные отношением  $a \leq b \leq c \leq d$ .

Полученные функции принадлежности лингвистических переменных приведены на рис. 2, 3.

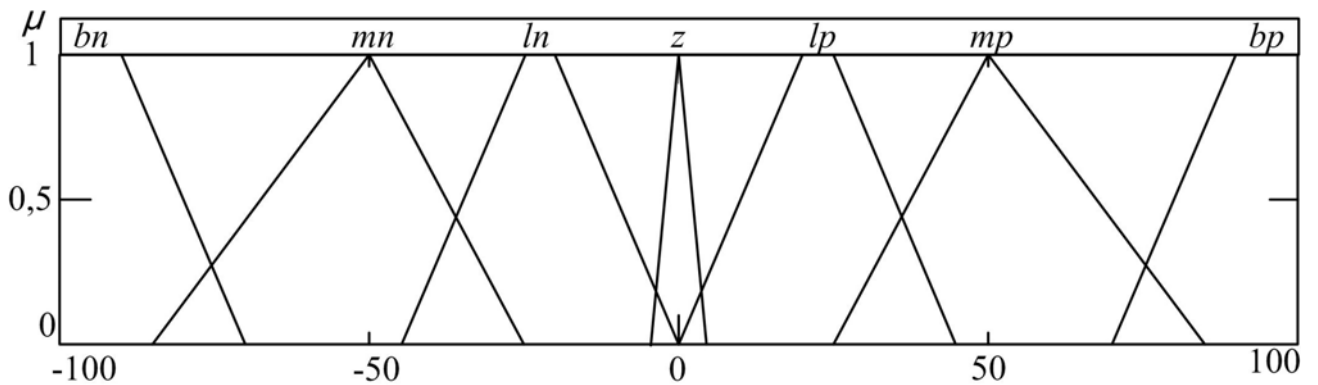


Рис. 2. – Функции принадлежности термов лингвистической переменной

$D\_CPU$

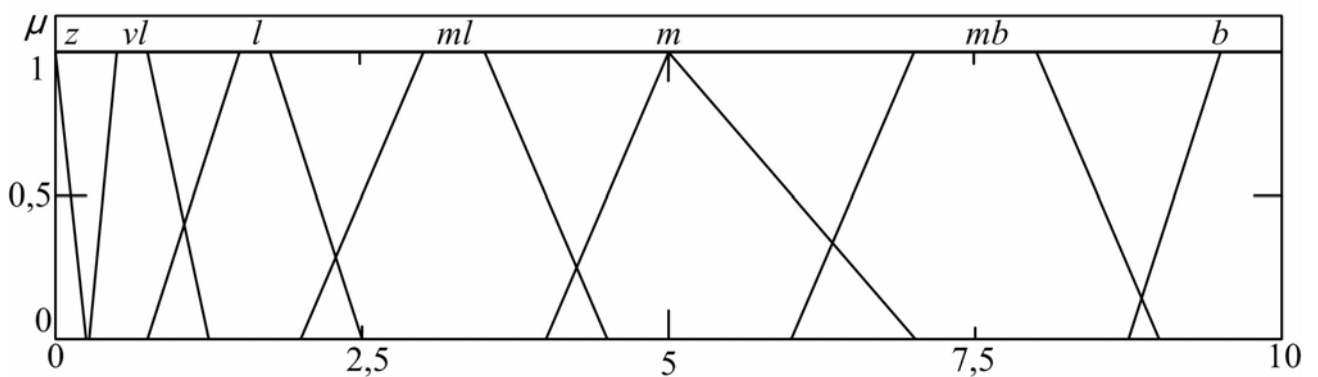


Рис. 3. – Функции принадлежности термов лингвистической переменной

$OUT\_DTI$

Для получения четкого значения лингвистической переменной  $OUT\_DTI$  используется алгоритм нечеткого вывода Мамдани с центроидным методом приведения к четкости [9].

$$y_i = \frac{\int_{Min}^{Max} x \cdot \mu_i(x) dx}{\int_{Min}^{Max} \mu_i(x) dx}$$

### Эксперимент

С целью проверки эффективности разработанного метода было проведено экспериментальное испытание на основе простейшей специально разработанной для нагрузочного теста СМ, структура которой показана на рис. 4.

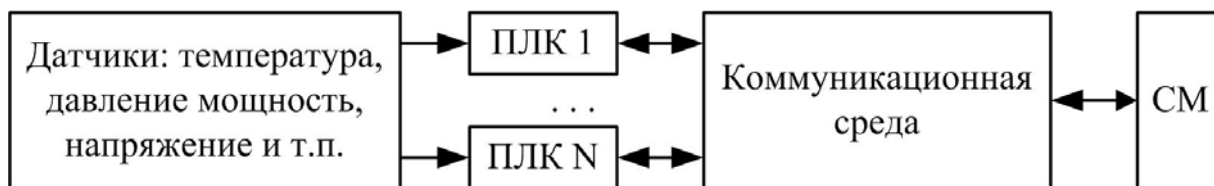


Рис. 4. – Структура типовой системы мониторинга

СМ позволяет принимать данные с нескольких сотен портов, функционирующих как каналы приема данных с ПЛК, по схеме один порт – один ПЛК. Принятые данные сохраняются на НЖМД. Формат пересылаемых данных представляет собой массив строк данных вида: «ЗНАЧЕНИЕ; ДАТА ВРЕМЯ», после получения данных СМ посылает сообщение о подтверждении приема пакета данных от ПЛК. Подобным образом работают многие реальные СМ, например, система мониторинга *OpenJEVis* компании *Envidatec GmbH (Germany, Hamburg)* [10].

В качестве системы тестирования СМ, выступает программная система тестирования (СТ), разработанная на основе описанного метода. СТ позволяет формировать нагрузку по каналам связи, имитируя данные поступающие с виртуальных ПЛК в тестируемую СМ. Настройка СТ заключается в выставлении пользователем требуемой нагрузки на



центральный процессор серверной ЭВМ СМ, длины сообщения в строках, а также количества виртуальных ПЛК. На рис. 5 изображена блок - схема взаимодействия СТ и СМ,

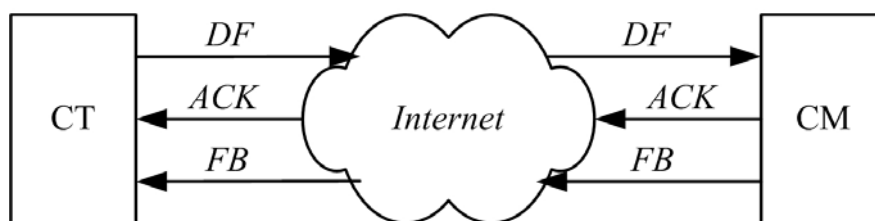


Рис. 5. – Блок - схема взаимодействия СТ и СМ

где:

1. *DF* – поток данных формируемый СТ и направляемый в СМ.
2. *ACK* – сообщение подтверждения получения пакета данных СМ
3. *FB* – данные о производительности, поступающие с серверной ЭВМ

СМ.

В качестве аппаратного обеспечения на котором базировались СТ и СМ, были использованы две равные конфигурации, представляющие собой два персональных ПК аналогичной конфигурации, с процессором *Intel Core 2 Duo 6400*, 2Гб ОЗУ, сетевой картой с пропускной способностью 1Гбит/с и 32 битной операционной системой *Microsoft Windows 7*.

### Результаты эксперимента

В ходе эксперимента было произведено три тестовых испытания, в каждом из которых произведено по три тестовых прогона СТ с различными параметрами для создания требуемой нагрузки на ЦП сервера СМ. Результаты тестирования и параметры СТ и СМ приведены в таблице №1.

Таблица №1

Параметры и результаты тестовых испытаний

| Входные параметры        | Тестовые испытания |     |     |
|--------------------------|--------------------|-----|-----|
|                          | 1                  | 2   | 3   |
| <i>IN_DQ</i> , строк     | 100                | 100 | 100 |
| <i>IN_TI</i> , пакетов/с | 1                  | 1   | 1   |

---

|   |       |       |       |
|---|-------|-------|-------|
| <i>IN CPU</i> , %                                 | 10    | 30    | 50    |
| Результирующие параметры при <i>IN PLC = 250</i>  |       |       |       |
| <i>FB CPU</i> , %)                                | 9-11  | 30-31 | 49-52 |
| <i>OUT TI</i> , пакетов/с                         | 6,6   | 21,3  | 34    |
| Результирующие параметры при <i>IN PLC = 500</i>  |       |       |       |
| <i>FB CPU</i> , %)                                | 9-13  | 29-33 | 48-51 |
| <i>OUT TI</i> , пакетов/с                         | 3     | 8,7   | 14,1  |
| Результирующие параметры при <i>IN PLC = 1000</i> |       |       |       |
| <i>FB CPU</i> , %                                 | 10-12 | 29-32 | 49-52 |
| <i>OUT TI</i> , пакетов/с                         | 1     | 2,9   | 5     |

Из результатов эксперимента можно видеть, что формирование нагрузки с использованием метода основанного на использовании нечеткой логики и обратной связи, позволило привести тестируемую демонстрационную (оцениваемую) СМ к целевым значениям загруженности, а именно нагрузки на центральный процессор.

Наблюдаемый, некоторый, разбег в загрузке ЦП сервера СМ, получаемый по обратной связи (параметр *FB\_CPU*), объясняется работой операционной системы: работой ядра, служебных программ и т.п., таким образом, расхождение от целевого значения нагрузки на ЦП в 1-5% можно считать допустимым.

Преимуществами предлагаемого метода является существенное уменьшение количества ручного труда и человек-часов, перед использованием стандартных средств, где подбор параметров для создания целевой нагрузки может быть очень продолжительным по времени, вследствие выполнения оператором многократных запусков СТ, и модификации параметров создания нагрузки, будь то целевая выборка данных, интенсивность их передачи и т.п. до получения оптимального результата.

Практическая значимость данного метода заключается в возможности оценки потенциала развития распределённой СМ [11], характеристик хранилища данных и внесения в программное обеспечение необходимых

доработок на ранних этапах жизненного цикла – до ввода в эксплуатацию или без использования на реальном оборудовании.

Дальнейшим развитием предложенного метода видится внедрение новых целевых параметров, таких как: нагрузка на сеть, систему памяти, систему хранения данных, а с точки зрения параметров, обеспечивающих нагрузку, включение автоматических способов генерирования содержимого и объема тестовой выборки, в частности внесение в тестовую выборку различных типов ошибок, позволяющих дополнительно нагрузить подсистему обработки входных данных тестируемой СМ.

### Литература

1. Темичев А.А., Файзрахманов Р.А. Проблемы современных распределенных систем мониторинга // Современные тенденции в образовании и науке. Тамбов: Бизнес-Наука-Общество, 2013. С. 132 – 133.
2. Гленфорд М., Том Б., Кори С. Искусство тестирования программ. 3 изд. М.: Диалектика, 2012. 272 с.
3. Файзрахманов Р.А., Долгова Е.В., Рахманов А.А. Задача адаптивного управления насосным оборудованием на основе интеллектуальных технологий // Электротехника. 2013. № 11. С. 17–19.
4. Усков А.А., Круглов В.В. Интеллектуальные системы управления на основе методов нечеткой логики. Смоленск: Смоленская городская типография, 2003. 177 с.
5. Файзрахманов Р.А., Полевщиков И.С. Оценка качества выполнения упражнений на компьютерном тренажере перегрузочной машины с использованием нечетких множеств // Инженерный вестник Дона. 2012. №4-1. URL: [ivdon.ru/magazine/archive/n4t1y2012/1265](http://ivdon.ru/magazine/archive/n4t1y2012/1265).
6. Файзрахманов Р.А., Полевщиков И.С. Методика автоматической оценки качества выполнения упражнений на компьютерном тренажере оператора производственно-технологической системы с использованием

нечетких множеств // Инженерный вестник Дона. 2014. №4. URL: ivdon.ru/ru/magazine/archive/n4y2014/ 2706

7. Azarkasb S.O. An Efficient Intrusion Detection System Based on Fuzzy Genetic approaches // Life Science Journal. 2013. №10. URL: lifesciencesite.com/ljsj/life1008s/002\_19226life1008s\_6\_21.pdf.

8. Beiranvand A., Khodabakhshi M., Yarahmadi M., Jalili M. Making a Mathematical Programming in Fuzzy Systems with Genetic Algorithm // Life Science Journal. 2013. №10. URL: lifesciencesite.com/ljsj/life1008s/008\_19232life1008s\_50\_57.pdf.

9. Прикладные нечеткие системы / Под ред. Тэрано Т., Асаи К., Сугэно М. М.: Мир, 1993. 368 с.

10. Темичев А.А., Кычкин А.В. Программный симулятор ПЛК VIDA350 системы энергоменеджмента Му-JEVis. Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. 2011. № 5. С. 210-220.

11. Петроченков А.Б., Кычкин А.В., Темичев А.А. Интегрированная диагностическая поддержка распределённой системы мониторинга энергоданных. Автоматизация и современные технологии. 2014. № 9. С. 10-16.

### References

1. Temichev A.A., Fayzrakhmanov R.A. Problemy sovremennykh raspredelennykh sistem monitoringa [Problems of modern distributed monitoring systems]. Sovremennye tendentsii v obrazovanii i nauke. Tambov: Biznes-Nauka-Obshchestvo, 2013. pp. 132 – 133.

2. Glenford M., Tom B., Kori S. Iskusstvo testirovaniya program. 3 izd. [The Art of Software Testing. 3rd Edition]. Moscow: Dialektika, 2012. 272 p.

3. Fayzrakhmanov R.A., Dolgova E.V., Rakhmanov A.A. Elektrotehnika. 2013. № 11. pp. 17–19.

---



4. Uskov A.A., Kruglov V.V. Intellektual'nye sistemy upravleniya na osnove metodov nechetkoy logiki. [Intelligent control systems based on fuzzy logic methods.]. Smolensk: Smolenskaya gorodskaya tipografiya, 2003. 177 p.

5. Fayzrakhmanov R.A., Polevshchikov I.S. Inzhenernyj vestnik Dona (Rus), 2012. №4-1. URL: [ivdon.ru/magazine/archive/n4t1y2012/1265](http://ivdon.ru/magazine/archive/n4t1y2012/1265).

6. Fayzrakhmanov R.A., Polevshchikov I.S. Inzhenernyj vestnik Dona (Rus), 2014. №4. URL: [ivdon.ru/ru/magazine/archive/n4y2014/2706](http://ivdon.ru/ru/magazine/archive/n4y2014/2706)

7. Azarkasb S.O. Life Science Journal, 2013, №10. URL: [lifesciencesite.com/ljsj/life1008s/002\\_19226life1008s\\_6\\_21.pdf](http://lifesciencesite.com/ljsj/life1008s/002_19226life1008s_6_21.pdf).

8. Beiranvand A., Khodabakhshi M., Yarahmadi M., Jalili M. Life Science Journal, 2013, №10. URL: [lifesciencesite.com/ljsj/life1008s/008\\_19232life1008s\\_50\\_57.pdf](http://lifesciencesite.com/ljsj/life1008s/008_19232life1008s_50_57.pdf).

9. Prikladnye nechetkie sistemy [Applied Fuzzy Systems]. Pod red. T.Terano, K.Asai, M.Sugeno. Moscow: Mir, 1993. 368 p.

10. Temichev A.A., Kychkin A.V. Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotehnika, informatsionnye tekhnologii, sistemy upravleniya. 2011. № 5. pp. 210-220.

11. Petrochenkov A.B., Kychkin A.V., Temichev A.A. Automation and modern technologies. 2014. № 9. pp. 10 – 16.