

УДК 681.513.54

ПРИМЕНЕНИЕ НЕЧЕТКОЙ ЛОГИКИ В СИСТЕМАХ РЕАЛЬНОГО ВРЕМЕНИ НА БАЗЕ ПРОГРАММНО-ТЕХНИЧЕСКОГО КОМПЛЕКСА УМИКОН

© 2022 г. Р. Беррекси^{1,*}, А. О. Толоконский^{1,**}, З. Лаидани¹, К. К. Абдулрахим¹, В. Н. Кудратов¹

¹ Национальный исследовательский ядерный университет "МИФИ", Москва, 115409, Россия

*e-mail: berreksirabie@gmail.com

**e-mail: AOTolokonskij@mephi.ru

Поступила в редакцию 27.05.2022 г.

После доработки 28.05.2022 г.

Принята к публикации 31.05.2022 г.

При усложнении технологии производства и объектов управления в современных промышленных областях управления инженерными системами, классические методы управления имеют свои известные недостатки с точки зрения стабилизации и производительности. Для преодоления этих недостатков требуется разработка и реализация более эффективных интеллектуальных алгоритмов систем управления технологическими процессами (АСУТП). Повышение эффективности систем управления технологическими процессами является одной из важных задач в современных областях управления. Чтобы достичь этих целей, необходимо внедрение новых подходов к построению систем управления реальным временем, таких как, адаптивное управление, методы управления, основанные на искусственном интеллекте, в том числе на нечеткой логике. Системы управления, основанные на нечеткой логике, чаще всего применяются в случаях недостаточного количества информации об объектах управления, когда классические ПИД-регуляторы не могут обеспечить необходимое качество управления и регулирования. Некоторые объекты управления АЭС относятся к объектам управления данного типа. В работе рассматриваются этапы разработки нечеткого алгоритма и его реализация в системе реального времени и интегрировании его в инструментальный для создания систем управления технологическими процессами (АСУТП). Алгоритм включает этапы фазификации, построения базы правил и дефазификации. Алгоритм нечеткой логики реализован в среде интерпретатора MikBASIC, являющегося системой технологического программирования базы данных реального времени MWBridge. Настоящая статья посвящена разработке нечеткого регулятора в реальном масштабе времени для управления задвижкой с целью поддержания параметра выхода системы в соответствии с заданными значениями.

Ключевые слова: АСУТП, программно-технический комплекс, нечеткая логика, фазификация, алгоритм, дефазификация

DOI: 10.56304/S2304487X22010059

ВВЕДЕНИЕ

В современных промышленных областях управление инженерными системами, используются цифровые ПИД-регуляторы, обеспечивающие требуемое качество управления при управлении линейными и определениями объектами благодаря своим преимуществам, чаще всего они выполняются с использованием системами управления технологическими процессами (АСУТП) на основе SCADA-систем. Однако линейные ПИД-регуляторы имеют свои недостатки с точки зрения стабилизации и производительности для систем, у которых точная математическая модель неизвестна или присутствует нестационарность или нелинейность в параметрах описывающих объект управления. При этих недостатках инже-

неры и исследователи по автоматизации заинтересовались новыми подходами к управлению, такими как адаптивное управление и также методы, основанные на искусственном интеллекте, в том числе по нечеткой логике [2].

1. НЕЧЕТКАЯ СИСТЕМА

Нечеткая логика или теория нечетких множеств привлекла внимание большого числа исследователей. Так как нечеткое логическое управление не требует точных математических моделей системы, а его алгоритм обычно построен из человеческих знаний, которые были запрограммированы на языке нечеткой логики (функции принадлежности, степень принадлежности,

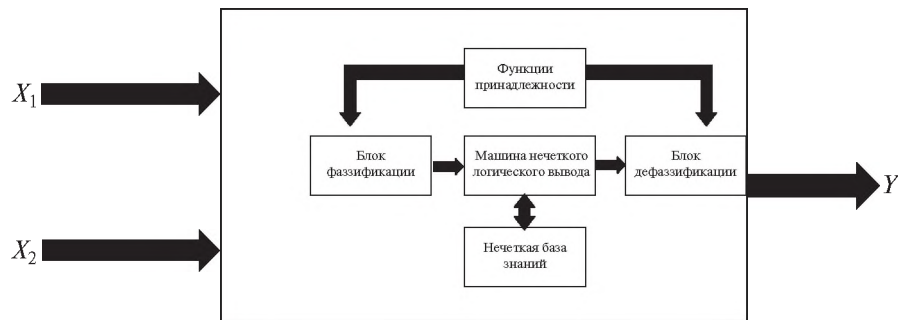


Рис. 1. Структура нечеткого регулятора.

базы правил и дефаззификация). Поэтому оно предпочитается больше, чем другие классические методы. Действительно, надежность и простота, которые ее характеризуют, являются основными причинами, которые часто заставляют нас больше искать этот метод [3–5].

Нечеткая логика — это форма логики многих значений, которые являются приблизительными, а не фиксированными и точными. По сравнению с традиционными двоичными наборами (где переменные могут принимать истинные или ложные значения), переменные нечеткой логики могут иметь истинное значение в диапазоне от 0 до 1.

На рисунке 1 показана общая структура нечеткой системы.

Система на рисунке 1 может быть описана как:

$$Y = f(x_1, x_2),$$

где f представляет собой отображение ввода-вывода нечеткой системы.

В общем, целый процесс нечеткого управления состоит из трех этапов: фаззификации (переход к нечеткости), разработки логического вывода и дефаззификации.

Фаззификация

Фаззификация состоит в преобразовании точных значений входных переменных в лингвистические значения на основе функций принадлежности. Последнее позволяет определить степень принадлежности лингвистической переменной с помощью применения некоторых положений теории нечетких множеств.

Разработка логического вывода

Этот этап состоит из двух блоков: механизма логического вывода и базы правил. На этом этапе определяются продукционные правила, указывающие взаимосвязи между входами и выходами лингвистическими переменными. Основаны на наборе логических правил в форме операторов “Если” — “То”, другими словами используются принципы: если (Условия), то (действия), которые получаются при знании конструкции систе-

мы и благодаря человеческому опыту. Другими словами совокупность таких правил описывает стратегию управления, применяемую в данной задаче.

Дефаззификация (преобразует нечеткие значения в чистые значения).

Дефаззификация представляет собой переход от нечетких значений величин к определенным физическим параметрам, являющийся выходом блока вывода, который можно использовать для принятия решения и/или управления исполнительными механизмами.

Предполагается нечеткая система с двумя предшествующими переменными и последующим параметром. Мы можем дать ему набор правил в виде:

R_n : Если x_1 является A_i , и x_2 является B_j , то Y является U_k ,

где n обозначает количество нечетких правил.

Переменные x_1 и x_2 обозначают два входа нечеткой системы, а переменная Y обозначает выход.

Предыдущая переменная x_1 связана с пятью нечеткими множествами A_i , а каждое нечеткое множество A_i ($i = 1: 5$) связано с функцией принадлежности $\mu_{A_i}(x_1): R \rightarrow [0, 1]$, которая дает степень принадлежности переменной x_1 относительно нечеткого множества A_i . μ_{A_i} .

Та же самая структура рассматривается для второй входной переменной x_2 , состоящей из пяти нечетких множеств B_j ($j = 1: 5$), характеризующих их функцией принадлежности $\mu_{B_j}(x_2): R \rightarrow [0, 1]$ и для последующей переменной y , которая связана с нечеткими множествами U_k и их функцией принадлежности.

На рис. 2, рис. 3 C_{A_i} и $C_{A_{i+1}}$ обозначают центры двух соседних функций принадлежности.

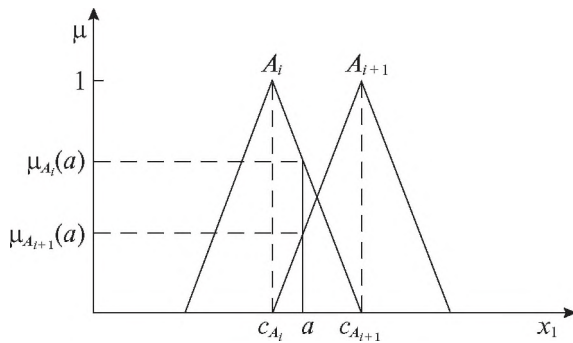


Рис. 2. Две функции принадлежности для входной переменной x_1 .

Степень принадлежности чистых переменных a в связи с нечеткими множествами A_i и A_{i+1} рассчитываются следующим образом:

$$\mu_{A_i}(a) = \frac{C_{A_{i+1}} - a}{C_{A_{i+1}} - C_{A_i}}, \quad (1)$$

$$\mu_{A_{i+1}}(a) = \frac{a - C_{A_i}}{C_{A_{i+1}} - C_{A_i}}. \quad (2)$$

Наконец, нечеткая выходная переменная преобразуется в числовую переменную решения с помощью процесса дефаззификации. Наиболее распространенным и полезным методом дефаззификации является “метод центра тяжести”. Этот метод очень популярен и широко используется для расчетов. Чтобы получить собственное значение выхода, мы используем следующее уравнение для дефаззификации:

$$Y = \frac{\sum_{i=1}^N y_i \mu_i}{\sum_{i=1}^N \mu_i}, \quad (3)$$

где Y — результат операции дефаззификации, N — количество правил, y_i — центр нечеткого набора выходных данных, подразумеваемый в i -е правило, μ_i — степени определенности рассчитываются как: $\mu_i = \mu_{A_i} \mu_{B_i}$.

2. АЛГОРИТМ НЕЧЕТКОГО РЕГУЛЯТОРА В СИСТЕМЕ РЕАЛЬНОГО ВРЕМЕНИ

База данных реального времени в САУ, как правило, представляет общее пространство памяти ядра процесса (среда программирования или физическое устройство, которое выполняет одну или несколько задач за ограниченное время), где агрегируют переменные (параметры), отслеживаемые технологическим процессом [8].

Системы управления, основанные на базе данных реального времени, имеют общее назначение, могут выполнять несколько задач одновременно, наиболее распространенными задачами являются запись данных, чтение и запись, расчет переменных, прогнозирование трендов, адресация процессов, активация индикаторов и др.

Помимо примечательной особенности этих систем, это высокая емкость хранения, свобода масштабирования оборудования и самой системы, широкая поддержка работы с различными протоколами и другими системами обработки в реальном времени, интерфейсы множественного доступа, возможность ручного управления, параметров, процессов или контролируемых устройств.

Программно-технический комплекс имеет графическую среду, в которой представлено множество инструментов для обработки данных в режиме реального времени, MWBridge также является инструментом для конфигурирования, а программирование модулей ПТК MIKKON, его дизайн позволяет работать с данными вручную и в среде программирования [9].

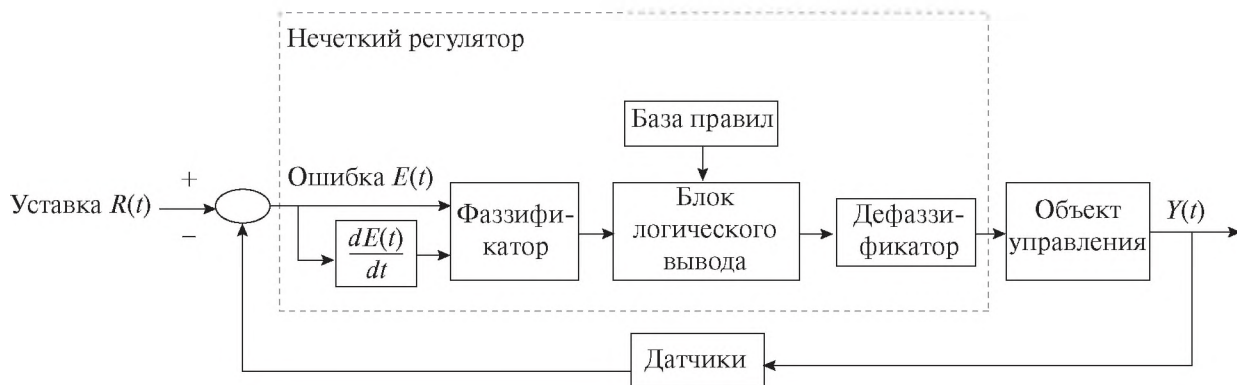


Рис. 3. Структура нечеткого регулятора.

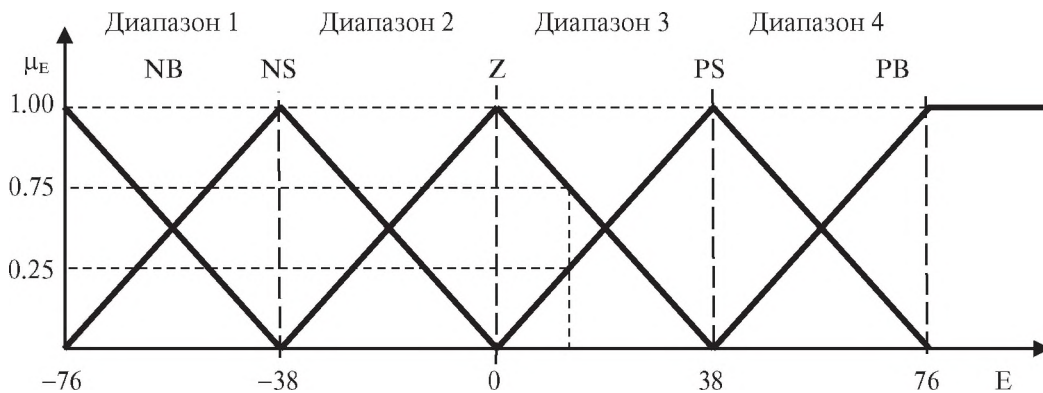


Рис. 4. Функции принадлежности входного переменного “Ошибка управления”.

2.1. Этапы реализации Алгоритм нечеткой логики:

1. Определение входных и выходных переменных.
2. Определение функции принадлежности для каждого входа и выхода.
3. Установление базы правил.
4. Дефазификация нечетких значений.

При проектировании системы управления на основе нечеткой логики, качество управления в таких системах зависит не только от величины сигнала ошибки, но и от скорости его изменения, т.е. производной сигнала ошибки.

На вход нечеткого регулятора подавалась уставка регулирования и еще один вход регулятора поступает от обратной связи. На основании значений сигналов рассогласования и от скорости его изменения, т.е. производной сигнала ошибки определяется степень открытой задвижки на выходе системы нечеткого регулятора. Нечеткий регулятор принимает меры в зависимости от определенной ошибки. Эта ошибка и ее производная применяются к регулятору, который затем предпринимает необходимые действия и определяет положение задвижки. Положение задвижки определяется правилами, написанными в базе правил регулятора нечеткой логики.

3. ОПРЕДЕЛЕНИЕ ВХОДНЫХ И ВЫХОДНЫХ ПЕРЕМЕННЫХ

$E(t)$ “Ошибка управления” и “Скорость изменения ошибки” $dE(t)/dt$ представляют собой входные переменные нечеткого регулятора, а φ “положение регулирующей задвижки” это выходная переменная нечеткого регулятора, лингвистическими переменными (ЛП) “Ошибка управления”, “Скорость изменения ошибки”, “положение регулирующей задвижки” являются параметры $E(t)$, $dE(t)/dt$ и φ соответственно.

Для лингвистической переменной $E(t)$ и $dE(t)/dt$ предполагаем следующее терм-множество $T(E(t), dE(t)/dt) = \{NB - \text{“большая отрицательная”}, NS - \text{“отрицательная”}, Z - \text{“ноль”}, PS - \text{“положительная”}, PB - \text{“большая положительная”}\}$. Причем значения $E(t)$ и $dE(t)/dt$ входят в диапазон $[-76, +76]$.

Было использовано в качестве лингвистических переменных φ следующее терм-множество $T(\varphi) = \{NB - \text{“большой угол влево”}, NS - \text{“небольшой угол влево”}, Z - \text{“нуль”}, PS - \text{“небольшой угол вправо”}, PL - \text{“большой угол вправо”}\}$. Причем положение регулирующей задвижки φ входит в диапазон $[-100\%, 100\%]$.

4. ФУНКЦИИ ПРИНАДЛЕЖНОСТИ ДЛЯ КАЖДОГО ВХОДА И ВЫХОДА

Каждое лингвистическое значение E и dE основано на симметричной треугольной функции принадлежности, которая показана ниже.

Нечеткая входная переменная “Ошибка управления” на графике, состоит из четырех областей, как показано на рис. 4.

Пять функций принадлежности f_1 [1], f_1 [2], f_1 [3], f_1 [4] и f_1 [5] используются, чтобы изображать различные диапазоны входной нечеткой переменной “Ошибка управления”.

Пять функций принадлежности используются f_2 [1], f_2 [2], f_2 [3], f_2 [4] и f_2 [5] для отображения различных диапазонов нечеткой входной переменной “Скорость изменения ошибки” на графике, состоящем из четырех областей, как показано на рисунке 5.

На рисунке 6 показаны пять функций принадлежности для нечеткой выходной переменной “положение регулирующей задвижки”.

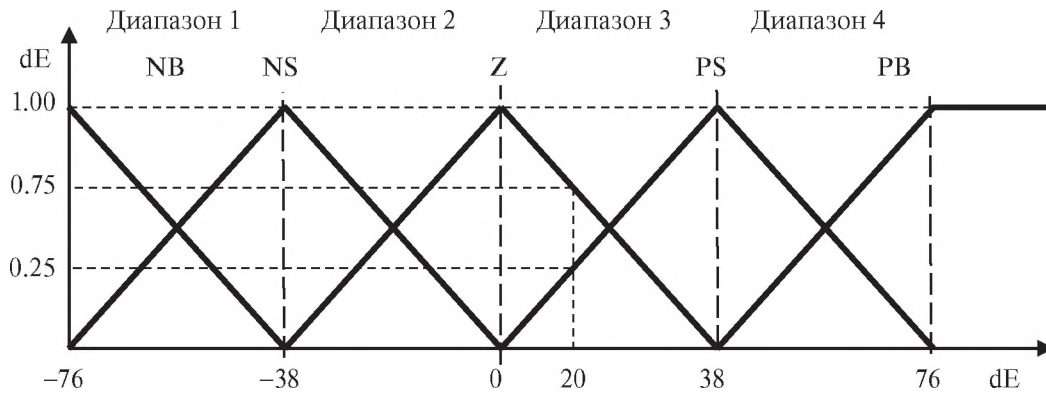


Рис. 5. Функции принадлежности входного переменного “Скорость изменения ошибки”.

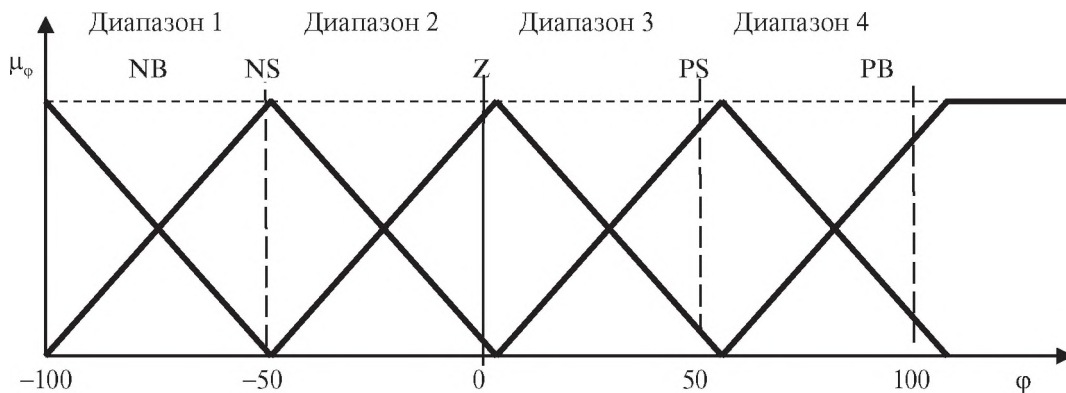


Рис. 6. Функции принадлежности входного переменного “Угол вентилля горячей воды”.

БАЗЫ ПРАВИЛ

Количество активных правил равно m^n , где m – максимальное количество функций принадлежности для входных переменных и n – количество входов. Для нашей схемы $m = 5$ и $n = 2$, поэтому общее количество активных правил равно 25. Общее количество правил равно произведению числа функций, сопровождаемых входными переменными в их рабочем диапазоне. Две входные переменные, описанные здесь, состояли из пяти функций принадлежности. Таким образом, требовалось $5 \times 5 = 25$ правил. Совокупность всех правил удобно представить в виде таблицы (табл. 1).

Можно представить Базы лингвистических правил следующим образом:

R_1 : Если $E(t) = NB$ и $dE(t)/dt = NB$, то $Y = NB$.

R_5 : Если $E(t) = NB$ и $dE(t)/dt = PB$, то $Y = Z$.

R_8 : Если $E(t) = NS$ и $dE(t)/dt = Z$, то $Y = NS$.

R_{10} : Если $E(t) = NS$ и $dE(t)/dt = PB$, то $Y = PS$.

R_{16} : Если $E(t) = PS$ и $dE(t)/dt = NB$, то $Y = NS$.

R_{20} : Если $E(t) = PS$ и $dE(t)/dt = PB$, то $Y = PB$.

Алгоритм, разработанный для этой системы, состоит из двух нечетких входных переменных. Пять треугольных функций принадлежности для каждого входа, и один выход – регулирующая задвижка. Регулирующая задвижка для выходного переменного также состоит из пяти функций принадлежности. Значение каждой переменной может лежать в любом из четырех диапазонов.



Рис. 7. Блок-схема операции фаззификации.

Таблица 1. Базы лингвистических правил для нечеткого логического регулятора

$dE(t)/dt$	$E(t)$ “Ошибка управления”				
	NB f_1 [1]	NS f_1 [2]	Z f_1 [2]	PS f_1 [4]	PB f_1 [5]
NB f_2 [1]	NB	NB	NS	NS	Z
NS f_2 [2]	NB	NS	NS	Z	PS
Z f_2 [3]	NS	NS	Z	PS	PS
PS f_2 [4]	NS	Z	PS	PS	PB
PB f_2 [5]	Z	PS	PS	PB	PB

f_1 и f_2 – лингвистические значения нечеткой переменной “Ошибка управления”, f_3 и f_4 для “Скорость изменения ошибки” [6, 7].

Для обсуждения разработки нечеткого регулятора в реальном масштабе времени, используем конкретные значения входных нечетких переменных “Ошибка управления” и “Скорость изменения ошибки”.

Мы выбираем два случайных значения входных переменных для нечеткого регулятора, чтобы продемонстрировать, как фаззификация выполняется в регулировании на основе нечеткой логики, используемой в этой системе. Пусть значение сигнала “Ошибка управления” $E(t) = 40$, и ее “Скорость изменения ошибки” $dE(t)/dt = 10$.

Прежде всего, необходимо определить, в каких диапазонах находятся входные переменные, далее рассчитать степени принадлежности для каждого входа.

В данном случае $E(t)$ находится в **диапазоне 4** и пересекается с нечеткими переменными “положительная” и “большая положительная”, f_0 и f_1 представляют собой степени принадлежности первой входной переменной $E(t)$ к нечетким переменным “положительная” и “большая положительная” соответственно. Таким же образом, для входного значения “Скорость изменения ошибки” $dE(t)/dt$, находящегося в **диапазоне 3**, соответствующее пересечение нечетких переменных “ноль” и “положительная”, f_2 и f_3 – степени принадлежности переменной $dE(t)/dt$ к нечетким переменным “ноль” и “положительная” соответственно.

Рассмотрим, как вычисляются степени принадлежности.

$$\begin{aligned} f_0 &= \frac{C_{A_{i+1}} - a}{C_{A_{i+1}} - C_{A_i}}; & f_1 &= 1 - f_0; \\ f_2 &= \frac{C_{A_{i+1}} - a}{C_{A_{i+1}} - C_{A_i}}; & f_3 &= 1 - f_2. \end{aligned} \quad (4)$$

После того, как мы подставим в формулу численные значения, получим:

$$f_0 = \frac{76 - 40}{38} = 0.947,$$

$$f_1 = 1 - 0.947 = 0.053,$$

$$f_2 = \frac{38 - 10}{38} = 0.736,$$

$$f_3 = 1 - 0.736 = 0.264.$$

Механизм логического вывода состоит из четырех операторов AND, это не логические AND, а выбор минимального значения для вывода. Этот последний принимает четыре входных данных от фаззификатора f_0 , f_1 , f_2 и f_3 , и применяет композицию min-max для получения выходных значений μ_i . Метод логического вывода min-max использует оператор min между двумя входными данными (продукционные правила), являющийся результатом условий правила, и правила в конечном итоге объединяются с помощью оператора OR и интерпретируются как операция max для каждого возможного значения выходной переменной. Следовательно, будут образованы четыре выхода μ_1 , μ_2 , μ_3 и μ_4 .

$$\mu_1 = f(0) \wedge f(2) = f(0) \text{ AND } f(2) = 0.947 \text{ AND } 0.736 = 0.736$$

$$\mu_2 = f(0) \wedge f(3) = f(0) \text{ AND } f(3) = 0.947 \text{ AND } 0.264 = 0.264$$

$$\mu_3 = f(1) \wedge f(2) = f(1) \text{ AND } f(2) = 0.053 \text{ AND } 0.736 = 0.053$$

$$\mu_4 = f(1) \wedge f(3) = f(1) \text{ AND } f(3) = 0.053 \text{ AND } 0.264 = 0.053$$

Знак $\wedge$ между значениями функции принадлежности используется для процесса Min-AND. В этом процессе мы получаем минимум значений функции AND. Эта интерпретация используется в процессе Мамдани-мин.

Затем значение в базе правил, по адресу нечетких входов, соответствует одному из центров выходной функции принадлежности. Конечный результат системы или выходной переменной после операции дефаззификации для управления исполнительным механизмом, определяющий степень задвижки – это средневзвешенное значение всех выходов правила в соответствии с методом центра тяжести (C.O.G), и вычисленное как

Выход для регулирующей задвижки

$$Y = \frac{\sum_{i=1}^N y_i \mu_i}{\sum_{i=1}^N \mu_i}, \text{ где } i = \text{от } 1 \text{ до } 4. \quad (5)$$

Определение значений y_i , соответствующих μ_i

Основываясь на значениях входных переменных $E(t)$ и $dE(t)/dt$, и после определения нахождения их диапазонов, определяются продукционные правила, указывающие взаимосвязи между входами и выходами лингвистическими переменными.

В данном случае:

$dE(t)/dt$	$E(t)$ “Ошибка управления”		μ_i	y_i
	PS, $f(0)$	PB, $f(1)$	$\mu_1 = 0.736$	50
Z, $f(1)$	PS	PS	$\mu_2 = 0.264$	50
PS, $f(2)$	PS	PB	$\mu_3 = 0.053$	50
			$\mu_4 = 0.053$	$(50 + (50 \cdot \mu_4))/2$

$$Y = \frac{y_1 \cdot \mu_1 + y_2 \cdot \mu_2 + y_3 \cdot \mu_3 + y_4 \cdot \mu_4}{\mu_1 + \mu_2 + \mu_3 + \mu_4}$$

После того, как мы подставим в формулу численные значения, получим

$$Y = \frac{0.736 \cdot 50 + 0.264 \cdot 50 + 0.053 \cdot 50 + (50 + (50 + ((50 \cdot 0.053)/2)) \cdot 0.053}{0.736 + 0.264 + 0.053 + 0.053}$$

$$Y = 50.1$$

Таким образом, значение задвижки, поддерживающее желаемой уставкой на выходе системы, φ “положение регулирующей задвижки” равно сумме предыдущего положения задвижки и выхода дефазификации Y .

5. РАЗРАБОТКА НЕЧЕТКОГО РЕГУЛЯТОРА В СИСТЕМЕ РЕАЛЬНОГО ВРЕМЕНИ

На рисунке 8 представлена блок-схема алгоритма нечеткого регулятора, реализованная в системе реального времени.

Способ реализации нечеткого алгоритма в системе реального времени заключается в описании всех функций принадлежности для каждого входа и выхода (этап фаззификация) в виде систем математических уравнений, определяет их диапазон нахождения. Далее рассчитать степень истинности каждой функции принадлежности, затем формулируются базы лингвистических правил. Наконец, преобразуется нечеткий выходной переменной в числовую переменную с помощью процесса дефаззификации. В расчетной процедуре интерпретатора MikBASIC реализуется программа, которая содержит систему разностных уравнений, описывающих систему управления. Выполнение программы осуществляется при помощи базы данных реального времени.

Таблица 2. Сравнение смоделированного и рассчитанного результата

Результат	Степень клапана
MATLAB моделирования	50.1
Расчетные значения	50.06

Структурная схема ПТК, которая была использована для выполнения нашей задачи, представлена на рисунке 9.

MWBridge включает в себя интерпретатор языка программирования MikBasik. Полученные уравнения необходимо задать и написать в переменных базы данных реального времени MWBridge (интерфейс интерпретатора MikBAS-IC), входящего в состав ПТК “УМИКОН”.

На рисунках 10 и 11 показаны интерфейс интерпретатора Mik BASIC, где реализация уравнений, необходимых для алгоритма управления температурой с использованием нечеткой логики. На рисунке 12 представлена база данных реального времени программы MWBridge, в которой записаны расчетные параметры нечеткого регулятора, а также входные и выходные переменные для управления задвижкой.

6. СРАВНЕНИЕ МЕЖДУ РАССЧИТАННЫМ И СМОДЕЛИРОВАННЫМ РЕЗУЛЬТАТОМ

Для подтверждения расчетных результатов, полученных в системе базы данных реального времени MWBridge, было проведено сравнение с результатом моделирования в среде MATLAB. Результат моделирования при одном значении входных переменных “Ошибка управления” $E(t) = 40$, и ее “Скорость изменения ошибки” $dE(t)/dt = 10$, для которых система была разработана, показан на рисунке 13.

Расчетные значения выходных переменных в соответствии с заданными значениями входных переменных для этой конструкции системы были такими же, как и результаты моделирования, показанные в таблице 2. Это показывает надежность системы управления с использованием нечеткой логики в системах реального времени.

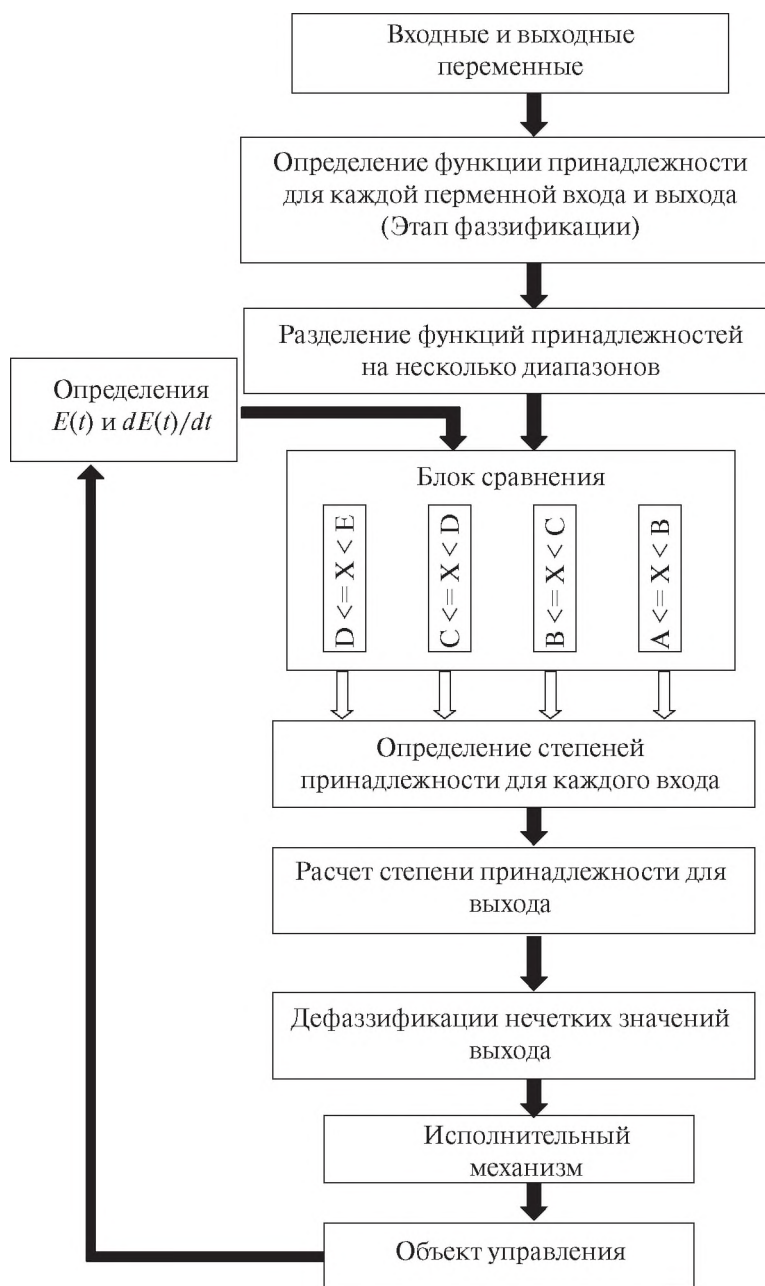


Рис. 8. Блок-схема нечеткого регулятора.

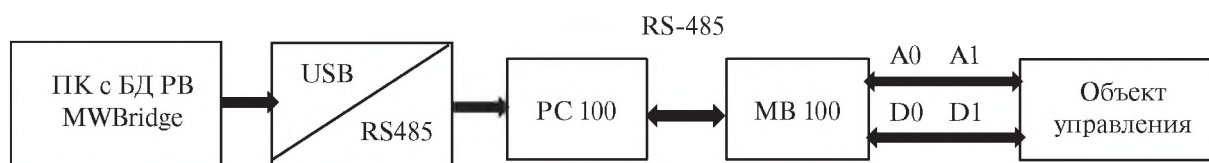


Рис. 9. Структурная схема ПТК.

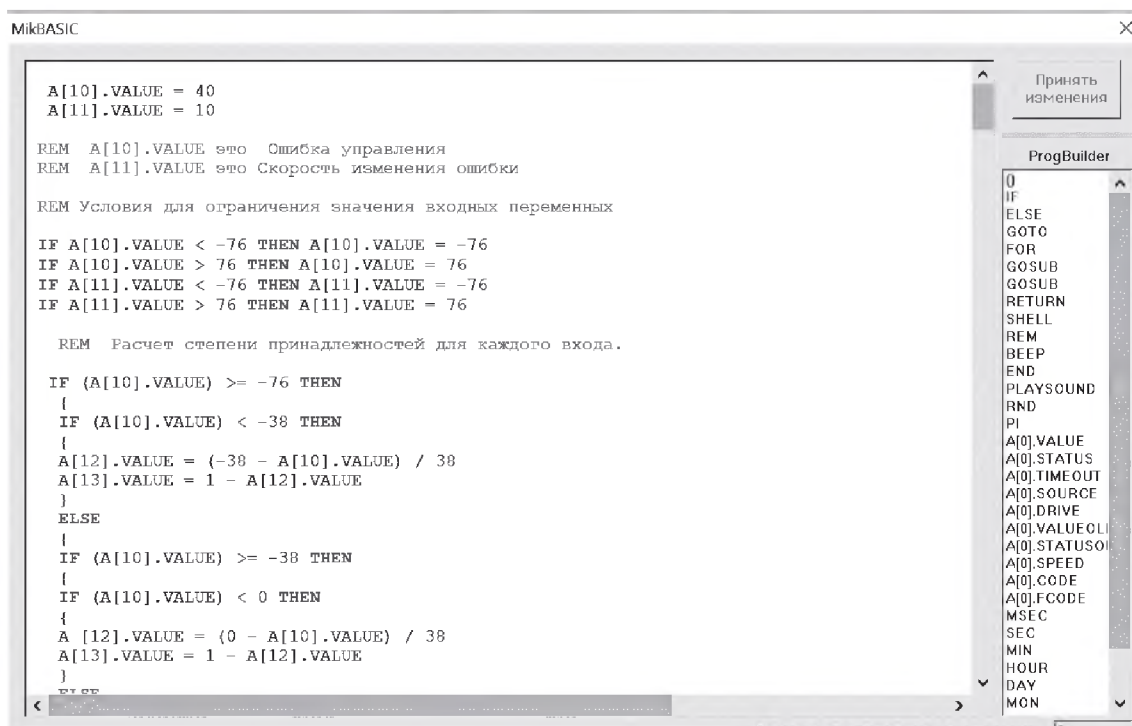


Рис. 10. Реализация уравнений расчета степеней принадлежности для каждого входа в интерфейсе MikBASIC.

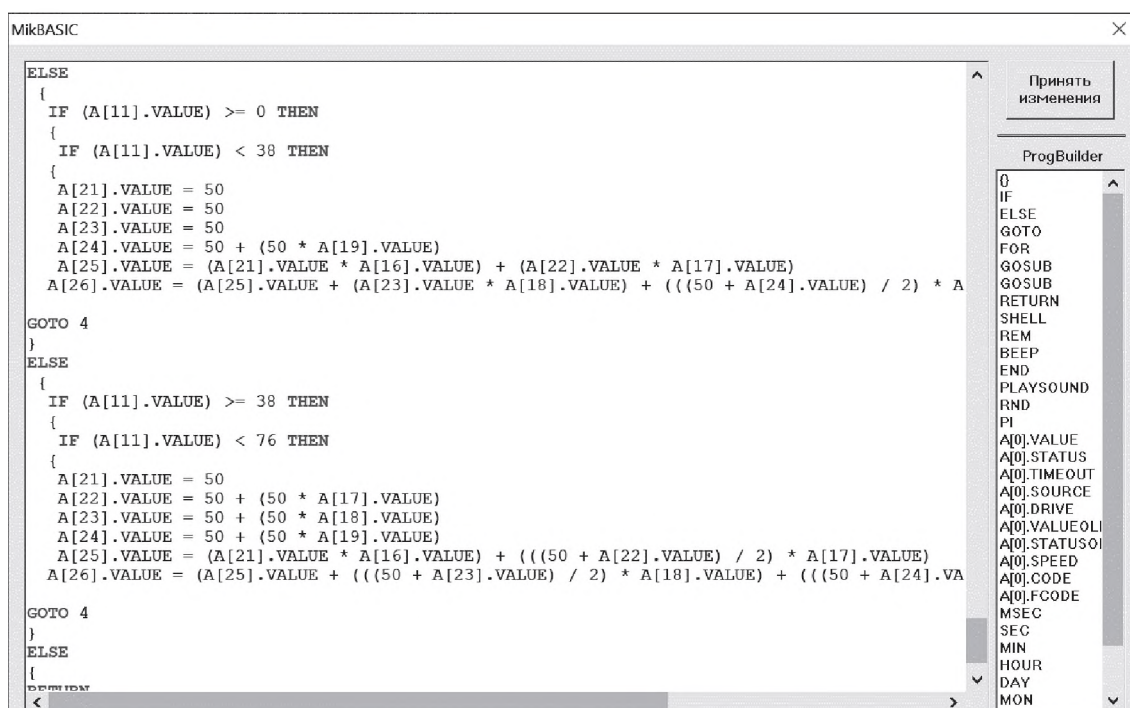


Рис. 11. Реализация уравнений операции дефаззификации (расчет значения выхода для регулирующего клапана) в интерфейсе MikBASIC.

Проект: [C:\Users\Rabie PC\Desktop\A17-166]. Лицензия: [Single licension]. Пользователь: []

УСО Сеть Алго Алго ВУ ТЭП Мнемосхемы Документы Просмотр трендов

Пользовательские профили Настройки Основные Статус Сети УСО Тренды WEB Такт БД: 102[приём: 0][передача: 0][память: 245444 [13:30:39]]

Номер	Название	Описание	Посл.Значение	Статус	Время обновл.	Источник	Таймаут
A10	E(t)	Ошибка управл...	40	0	13:30:38.901		60000
A11	dE(t)/dt	Скорость измен...	10	0	13:30:38.901		60000
A12	f0/E(t)	1-Степень прин...	0.947368	0	13:30:38.901		60000
A13	f1/E(t)	2-Степень прин...	0.0526316	0	13:30:38.901		60000
A14	f2/dE(t)/dt	1-Степень прин...	0.736842	0	13:30:38.901		60000
A15	f3/dE(t)/dt	2-Степень прин...	0.263158	0	13:30:38.901		60000
A16	μ_1	Степень принад...	0.736842	0	13:30:38.901		60000
A17	μ_2	Степень принад...	0.263158	0	13:30:38.901		60000
A18	μ_3	Степень принад...	0.0526316	0	13:30:38.901		60000
A19	μ_4	Степень принад...	0.0526316	0	13:30:38.901		60000
A20	μ_j	$\mu_1 + \mu_2 + \mu_3 + \mu_4$	1.10526	0	13:30:38.901		60000
A21	S1		50	0	13:30:38.901		60000
A22	S2		50	0	13:30:38.901		60000
A23	S3		50	0	13:30:38.901		60000
A24	S4		52.6316	0	13:30:38.901		60000
A25			50	0	13:30:38.901		60000
A26	Y	Значения дефазы	50.0627	0	13:30:38.901		60000
A27	Вых-OLD	Пред.дущий в...	10	0	13:30:38.901		60000
A28	Вых	Значение клапа...	60.0627	0	13:30:38.901		60000
A29			###	###	###		60000

Группы: ☐ Тренды

Настройка: << < > >> A000 D000

Рис. 12. Базы данных реального времени программы MWBRIDGE.

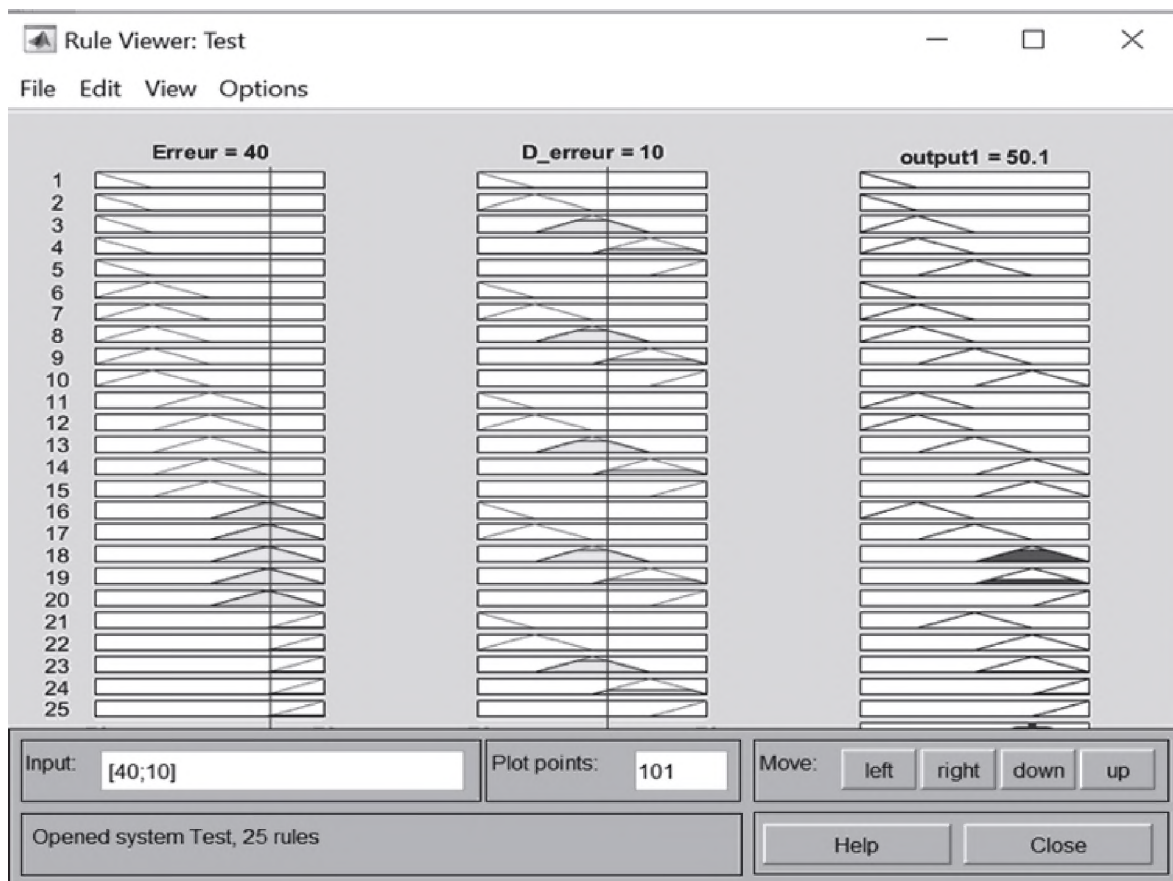


Рис. 13. Результаты моделирования с помощью средства MATLAB.

7. ЗАКЛЮЧЕНИЕ

В данной работе излагается вопрос реализации алгоритма нечеткого регулятора в системе реального времени. Введение алгоритма управления на базе нечеткой логики в системы реального времени позволяет более качественно управлять процессом регулирования, в отличие от традиционного ПИД-регулирования, особенно в случаях недостаточного количества информации об объектах управления или при управлении сложными объектами. Преимущества — это не анализ логики, а решение проблем, которые трудно анализировать и определить их математическую систему, и которые легко решаются применяя лингвистические переменные. Одной из целей этого исследования является разработка системы управления с использованием нечеткой логики в масштабе реального времени. Этот нечеткий регулятор доступен для использования в других промышленных приложениях. Далее, простота и гибкость программы позволяют этой нечеткой системе развиваться и применяться.

При сравнении моделированных и расчетных результатов мы наблюдаем, что расчетная модель и результаты моделирования совпадают. Вследствие этого, разработанную систему можно расширить на любое количество входов и выходов.

СПИСОК ЛИТЕРАТУРЫ

1. Тарасов В.Б., Святкина М.Н. Интеллектуальные SCADA-системы: истоки и перспективы // Наука и образование. Электрон. журн. МГТУ им. Н.Э. Баумана. 2011. № 10.
2. Mauro Vitor de Oliveira, José Carlos Soares de Almeida. Application of artificial intelligence techniques in modeling and control of a nuclear power plant pressurizer system // Progress in Nuclear Energy, 2013. V. 63. P. 71–85.
3. Piegat A. Fuzzy Modelling and Control. Heidelberg; New York: Physica-Verlag, 2001. 728 p.
4. Thae Thae Ei Aung, Myo Maung Maung and Hla Myo Tun. Design and Development of Fuzzy Logic Controller for Liquid Flow Control // International Journal of Electronics and Computer Science Engineering. ISSN: 2277-1956/V3-N3-208-218.
5. Zaidan Qawaqzeh M., Khalil Ibrahim Y. Valve Control System Using Fuzzy Logic // British Journal of Mathematics & Computer Science; BJMCS, 2016. V. 14 (4). P. 1–7.
6. Saleem Khan M., Khaled Benkrid. Design of Liquids Mixing Control System using Fuzzy Time Control Discrete Event Model for Industrial Applications/ /World Academy of Science, Engineering and Technology, 2010. № 48. С. 545–553.
7. Крюков В.В., Тугов В.В. Управление редуцированием газа магистральных газопроводов высокого давления с применением аппарата нечеткой логики // Известия высших учебных заведений. Поволжский регион. Технические науки, 2019. Т. 49. № 1. С. 55–65.
8. Власов В.А. и др. Внедрение опыта создания АСУ ТП объектов атомной отрасли на базе ПТК УМИКОН в учебный процесс // Известия вузов. Ядерная энергетика, 2014. № 1. С. 149–155.
9. Толоконский А.О., Володин В.С. Реализация принципа разнообразия программно-технических комплексов АСУ ТП объектов атомной отрасли в учебном процессе // Подготовка кадров. Ядерная энергетика, 2018. № 1. С. 154–164.
10. Хиженяков Ю.Н. Алгоритмы нечеткого, нейронного и нейро-нечеткого управления в системах реального времени: учеб. пособие. Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2013. 156 с.

Vestnik Natsional'nogo Issledovatel'skogo Yadernogo Universiteta "MIFI", 2022, vol. 11, no. 1, pp. 68–79

Application of Fuzzy Logic in Real-Time Systems Based on the Umicon Software and Hardware Complex

R. Berreksi^{a,#}, A. O. Tolokonskiy^{a,##}, Z. Laidani^a, K. K. Abdulrahim^a, and V. N. Kudratov^a

^a National Research Nuclear University MEPHI (Moscow Engineering Physics Institute), Moscow, 115409 Russia

[#]e-mail: berreksirabie@gmail.com

^{##}e-mail: AOTolokonskiy@mephi.ru

Received May 27, 2022; revised May 28, 2022; accepted May 31, 2022

Abstract—In application to complicated production technology and control objects in modern industrial areas of engineering systems management, classical control methods have their well-known disadvantages in stabilization and performance. To overcome these disadvantages, the development and implementation of more efficient intelligent algorithms for process control systems (APCS) is required. In addition, increasing the efficiency of process control systems is one of the important tasks in modern areas of management. To achieve these goals, it is necessary to introduce new approaches to building real-time control systems, such as adaptive control and control methods based on artificial intelligence including fuzzy logic. Control systems

based on fuzzy logic are most often used when information about control objects is insufficient and classical PID controllers cannot provide the required quality of control and regulation. Some NPP control objects belong to this type of control objects. Stages of developing a fuzzy algorithm, its implementation in a real-time system, and integrating it into a toolkit for designing process control systems (APCS) are considered. The algorithm includes the stages of fuzzification, rule base, and defuzzification. The fuzzy logic algorithm is implemented in the MikBASIC interpreter environment, which is the MWBridge real-time database technological programming system. Furthermore, the aim of this work is to develop a real-time fuzzy controller to control a gate valve in order to maintain the system output parameters in accordance with the specified values.

Keywords: automatic process control system (APCS), software and hardware complex, fuzzy logic, fuzzification, algorithm, defuzzification

DOI: 10.56304/S2304487X22010059

REFERENCE

1. Tarasov V.B., Syvatkina M.N. Intellekturnye SCADA-sistemy: istoki i perspektivy [Intellektualnye SCADA-sistemy: istoki i perspektivy]. *Nauka i obrazovaniye. Elektron. jurn. MGTU im. N.E. Bauman*, 2011. № 10 (in Russian).
2. Mauro Vitor de Oliveira, José Carlos Soares de Almeida. Application of artificial intelligence techniques in modeling and control of a nuclear power plant pressurizer system, *Progress in Nuclear Energy*, 2013, vol. 63, pp. 71–85.
3. Piegat A. *Fuzzy Modelling and Control*. Heidelberg; New York: Physica-Verlag, 2001, 728 p.
4. Thae Thae Ei Aung, Myo Maung Maung and Hla Myo Tun. Design and Development of Fuzzy Logic Controller for Liquid Flow Control, *International Journal of Electronics and Computer Science Engineering* ISSN: 2277-1956/V3-N3-208-218.
5. Zaidan Qawaqzeh M., Khalil Ibrahim Y. Valve Control System Using Fuzzy Logic. *British Journal of Mathematics & Computer Science; BJMCS*, 2016. vol. 14 (4), pp. 1–7.
6. Saleem Khan M., Khaled Benkrid. Design of Liquids Mixing Control System using Fuzzy Time Control Discrete Event Model for Industrial Applications. *World Academy of Science, Engineering and Technology*, 2010, no. 48, pp. 545–553.
7. Kryukov V.V., Tugov V.V. Upravleniye redutsirovaniyem gaza magistral'nykh gazoprovodov vysokogo davleniya s primeneniym apparata nechetkoy logiki. [Control of gas reduction in main high-pressure gas pipelines using a fuzzy logic apparatus] *Izvestiya vysshikh uchebnykh zavedeniy. Povolzhskiy region. Tekhnicheskkiye nauki*, 2019, vol. 49, no. 1, pp. 55–65.
8. Vlasov V.A., Korolyov S.A., Lebedev V.O., Tolokonsky A.O. Vnedrenie opyta sozdaniya ASU TP ob'ektov atomnoy otrasli na baze PTK UMIKON v uchebnyy process [Implementation of the experience in designing APCS systems for nuclear facilities based on UMIKON package in the educational process]. *Izvestiya vuzov. Yadernaya Energetika*, 2014, no. 1, pp. 149–155 (in Russian).
9. Tolokonsky A.O., Volodin V.S. Realizatsiya principa raznoobraziya programmno-tekhnicheskikh kompleksov ASU TP ob'ektov atomnoy otrasli v uchebnom processe [Implementation of the principle of diversity of software and hardware complexes of automated process control systems for nuclear facilities in the educational process]. *Podgotovka kadrov. Yadernaya energetika*, 2018, no. 1, pp. 154–164 (in Russian).
10. Khizhnyakov Y.N. *Algoritmy nechetkogo, nejronnogo i nejro-nechetkogo upravleniya v sistemah real'nogo vremeni: ucheb. posobie*. [Fuzzy, neural and neuro-fuzzy control algorithms in real-time systems: textbook]. Perm, Publishing House of Perm. Nat. research polytechnic university, 2013, 156 p.