

УДК 681.518.3  
DOI: 10.15827/0236-235X.136.597-607

Дата подачи статьи: 08.11.21, после доработки: 09.11.21  
2021. Т. 34. № 4. С. 597–607

## **Метод синтеза нечетких регуляторов на основе кластеризации**

*В.В. Игнатьев*<sup>1</sup>, *к.т.н., ведущий научный сотрудник, vvignatev@sfnedu.ru*  
*В.В. Соловьев*<sup>1</sup>, *ст. преподаватель*

<sup>1</sup> Южный федеральный университет, г. Таганрог, 347922, Россия

Целью авторов исследования является разработка метода синтеза нечетких регуляторов по экспериментальным данным на основе кластеризации как самого простого способа определения количества функций принадлежности и создания базы правил. Для достижения поставленной цели предлагается использовать экспериментальные данные о входных и выходных сигналах системы управления техническим объектом с классическим регулятором. На основе этих данных разработан метод кластеризации, позволяющий определять терм-множества входных и выходных лингвистических переменных нечеткого регулятора, реализующего алгоритм нечеткого вывода Мамдани, и составлять базу правил.

Кластеризация выполняется путем оценки границ интервалов варьирования экспериментальных данных, равномерного разделения на кластеры в зависимости от требуемой мощности терм-множеств лингвистических переменных и определения принадлежности данных к кластерам. Поскольку экспериментальные данные связаны, то есть для каждого момента времени сохраняются данные как о входных, так и о выходных сигналах классического регулятора и определяется их принадлежность к кластерам, разработка базы правил нечеткого регулятора не вызывает затруднений.

Разработанное авторами в среде MatLab ПО позволяет как снимать экспериментальные данные, так и синтезировать нечеткий регулятор и проверять его работоспособность. Модель системы управления создана в среде Simulink, метод кластеризации и определения параметров лингвистических переменных реализован в виде программы в m-файле, а нечеткий регулятор в пакете расширения Fuzzy Logic Toolbox.

В качестве примера рассмотрен процесс заполнения соединенных контейнеров, математической моделью которого является передаточная функция второго порядка с запаздыванием. Для выбора оптимальной структуры нечеткого регулятора проведено исследование на основе экспериментальных данных. Полученные в данной работе результаты сравнивались с классическим ПД-регулятором, модель которого реализована в среде Simulink. Результаты исследования будут полезны разработчикам нечетких моделей управления.

**Ключевые слова:** нечеткий регулятор, ПД-регулятор, кластеризация данных, вес правил, редукция базы правил, автоматический синтез базы правил.

На сегодняшний день синтез систем управления техническими объектами с нечеткими регуляторами привлекает внимание исследователей всего мира, так как позволяет получать лучшие показатели качества управления (длительность переходного процесса, перерегулирование) по сравнению с системами с классическими регуляторами. В данной работе под классическими регуляторами авторы понимают ПИД-регуляторы и их различные модификации. Исследованиям, связанным со структурами нечетких регуляторов и синтезом баз нечетких правил, посвящено немало публикаций.

В работе [1] рассматривается нечеткий ПИД-регулятор, у которого каждая из составляющих описывается независимой базой нечетких правил. С помощью генетического

алгоритма выполнялась настройка коэффициентов регулятора и нелинейных преобразующих функций. В результате исследования установлено, что с применением интеллектуальных технологий можно достичь значительного улучшения показателей качества управления.

По аналогии с эталонной моделью, описывающей динамику объекта управления, некоторые исследователи предлагают модель желаемого поведения, в которой, кроме входных и выходных сигналов, используются и их производные [2]. Метод адаптации параметров регулятора позволяет изменить количество функций принадлежности или степень их перекрытия.

Задаче синтеза нечеткого регулятора при наличии шумов в канале измерения посвя-

щена работа [3]. Авторы рассматривают последовательную процедуру оценки состояния и синтеза управления. Параметрическую неопределенность предложено формализовать, используя теорию нечетких интервалов и модель Такаги–Сугено, которая описывает динамику объекта управления.

Использование нечеткой модели Такаги–Сугено для описания динамики системы обсуждается и в [4]. Рассматривается процедура синтеза нечеткого регулятора для нелинейных объектов с шумами и запаздыванием по состоянию. Авторами теоретически обоснована методика анализа устойчивости таких систем и сформулированы условия устойчивости.

В качестве объекта управления в исследовании [5] рассмотрен квадрокоптер с линеаризованной математической моделью. По результатам моделирования системы управления с ПИД-регуляторами были сохранены значения ошибок для линейных и угловых параметров, позволившие синтезировать нечеткие регуляторы полета квадрокоптера с лучшими показателями качества управления.

Работа [6] посвящена нестационарным нелинейным системам с нечеткими ПИД-регуляторами. Предложен метод построения областей гарантированной устойчивости, позволяющий определить области, превышающие по площади те, которые получают классическими методами. В основу метода положен алгоритм построения сплайновых функций Ляпунова.

Авторы [7] показывают применение методов интеллектуального управления с нечеткой логикой для формализации неопределенности процесса сушки древесины. Терм-множества лингвистических переменных составлены на основе треугольных и трапециевидных функций принадлежности, равномерно распределенных на универсальных множествах.

В работе [8] как объект управления рассматривается ветроэнергетическая установка. Разработанный авторами алгоритм самоорганизации последовательно формирует базу правил нечеткого регулятора. Результаты моделирования системы с нечетким регулятором демонстрируют меньшую длительность переходного процесса при разгоне и торможении по сравнению с ПИ-регулятором.

В качестве объекта управления в [9] предложен паровой котел, для которого синтези-

ровался нечеткий регулятор. Авторы сняли экспериментальные данные с котла и, применяя кластеризацию, получили терм-множества нечетких переменных с треугольными функциями принадлежности. Отмечено, что при синтезе нечетких регуляторов по экспериментальным данным существует проблема появления нечетких правил с одинаковым антецедентом, но разным консеквентом.

Метод разработки нечетких моделей на основе экспериментальных данных с реальных объектов обсуждается в [10]. Авторы решают проблему дублирующих правил путем назначения степени их важности. При этом исходная база нечетких правил создается за счет всевозможных сочетаний функций принадлежности входных и выходных переменных, а затем корректируется с применением эвристики.

Авторы настоящей работы в [11, 12] уже рассматривали этапы синтеза нечетких моделей по экспериментальным данным, снятым с объектов управления с применением алгоритмов Мамдани и Сугено. Были предложены различные методы кластеризации данных для определения нечетких переменных и синтезированы нечеткие регуляторы, демонстрирующие лучшие показатели качества управления по сравнению с классическими регуляторами.

По результатам обзора публикаций можно сделать некоторые выводы относительно методов синтеза нечетких регуляторов по экспериментальным данным:

- обработке экспериментальных данных и синтезу нечетких моделей управления уделяется недостаточно внимания;
- исследователи отмечают возможность возникновения неопределенности при синтезе базы правил и в основном прибегают к назначению весовых коэффициентов дублирующим правилам, фактически не исключая неопределенность;
- во многих публикациях рассматриваются различные структуры нечетких регуляторов, но игнорируются вопросы унификации при их синтезе;
- процесс синтеза нечетких моделей по экспериментальным данным достаточно трудоемкий, однако решению задачи автоматизации вычислений авторы не уделяют должного внимания.

Целью настоящего исследования является разработка метода синтеза нечетких регуляторов по экспериментальным данным с при-

менением кластеризации. Отличительными особенностями предлагаемого метода являются возможность автоматического определения параметров функций принадлежности нечетких переменных, синтез базы правил и ее редукция без привлечения экспертов. Данная цель достигается путем использования разработанного ПО, которое существенно снижает трудоемкость вычислений и позволяет получить fis-структуру нечеткого регулятора.

**Метод синтеза нечетких регуляторов**

При разработке метода авторы базируются на результатах, полученных в исследованиях [11, 12], и развивают их в части устранения неопределенности в базе правил и автоматизации вычислений.

Реализация метода синтеза включает следующие этапы.

**Этап 1.** Снятие входных и выходных данных с системы управления с классическим регулятором. Это могут быть данные с реального объекта управления в рабочем режиме его функционирования либо модельные данные, генерируемые по результатам моделирования. Для синтеза нечетких регуляторов предлагается структура системы, представленная на рисунке 1.

На схеме использованы следующие обозначения:  $g$  – задающий сигнал;  $e$  – сигнал ошибки;  $u$  – сигнал управления;  $y$  – выходной сигнал;  $K$  – ключ;  $P = \langle N, B, D \rangle$ , где  $N$  – входные и выходные нечеткие переменные,  $B$  – база правил,  $D$  – параметры алгоритма нечеткого вывода. Согласно рисунку 1, первоначально ключ находится в положении 1 с подключенным классическим регулятором, входные и выходные данные которого поступают на модуль кластеризации. В данном модуле определяются параметры кластеров для каждого из измеренных сигналов согласно поступившим данным. В модуле создания базы правил формируется база правил согласно параметрам кластеров и экспериментальным данным с классического регулятора. В модуле редукции базы правил осуществляется поиск правил с одинаковым antecedентом и разным консеквентом и выполняется их объединение. В модуле синтеза нечеткого регулятора окончательно определяются параметры нечетких переменных, базы правил и алгоритма нечеткого вывода, которые загружаются в нечеткий регулятор. Затем ключ переводится в положение 2 и система управления продолжает функционирование под управлением нечеткого регулятора. Такой же процесс может осуществляться для систем, у которых классический регулятор имеет не-

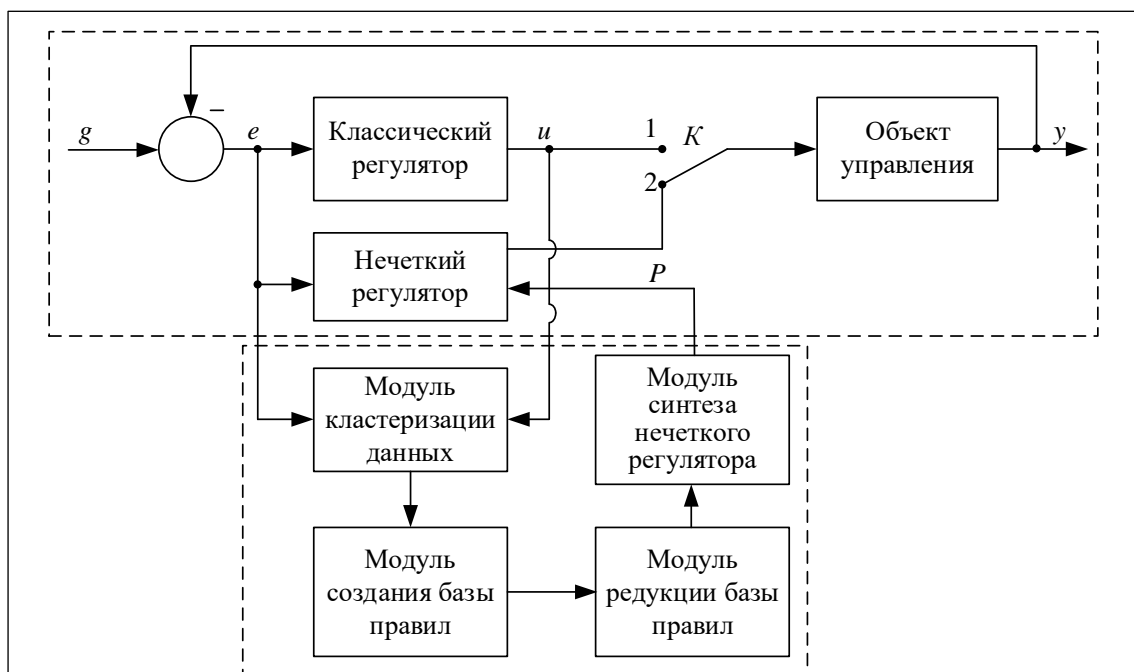


Рис. 1. Структура системы для синтеза нечетких регуляторов

Fig. 1. The structure of the system for the synthesis of fuzzy regulators

сколько входов. Важно обеспечить фиксированный шаг времени считывания данных, так как только в этом случае процедура кластеризации будет корректной.

**Этап 2.** Определение количества кластеров для каждого из сигналов в ряду экспериментальных данных. Известно, что при разработке нечетких моделей с привлечением экспертных знаний обычно стараются использовать не более  $7 \pm 2$  функций принадлежности для каждой нечеткой переменной [13]. При создании нечетких моделей без привлечения экспертов количество функций принадлежности лингвистических переменных определяется вычислительными ресурсами, а также разницей между измерениями в ряду экспериментальных данных. Если рассматривать равномерное распределение кластеров в диапазоне изменения экспериментального сигнала, то можно сформулировать граничные условия для определения их количества.

Допустим, минимальный размер кластера равен величине  $\delta$ , которая определяет максимальное расстояние между соседними измерениями в ряду данных. Если выбрать кластер меньшего размера, то после определения принадлежности экспериментальных данных кластерам появятся кластеры, в которые не попало ни одного измерения. Величину  $\delta$  можно определить в цикле, перебирая данные каждого измеренного сигнала. В связи с этим максимальное количество кластеров сигнала может быть найдено по формуле

$$N_{i\max} = \left\lceil \frac{s_{i\max} - s_{i\min}}{\delta_i} \right\rceil, \tag{1}$$

где  $N_{i\max}$  – максимальное количество кластеров  $i$ -го сигнала;  $s_{i\max}$ ,  $s_{i\min}$  – максимальное и минимальное значения сигнала;  $\delta_i$  – максимальное отклонение между двумя соседними измерениями;  $\lceil \rceil$  – округление до целого в большую сторону.

На основе выражения (1) можно вычислить размер кластера:

$$pc_i = \frac{s_{i\max} - s_{i\min}}{N_{i\max}}, \tag{2}$$

где  $pc_i$  – размер кластера  $i$ -го сигнала.

Для устранения неопределенности необходимо, чтобы количество кластеров данных выходного сигнала  $N_{out}$  удовлетворяло условию

$$N_{out} \leq \prod_{i=1}^m N_{iin}, \tag{3}$$

где  $N_{iin}$  – количество кластеров  $i$ -го входного сигнала нечеткой модели;  $m$  – количество входных сигналов.

При выполнении условия (3) все функции принадлежности нечеткой переменной выходного сигнала будут учтены в базе правил. Если условие (3) не выполняется, необходимо уменьшить количество кластеров данных выходного сигнала и пересчитать их размеры по формуле (2). Минимальное количество кластеров для сигналов равно 2, поэтому количество кластеров для каждой из переменных выбирается из интервала  $[2, N_{i\max}]$ .

**Этап 3.** Создание нечетких переменных. По результатам второго этапа для каждой переменной определены количество кластеров  $N$ , интервал изменения  $[s_{\min}, s_{\max}]$  и размер  $pc$ . Границы и центры кластеров можно определить по формулам:

$$\begin{aligned} c_i &= [s_{i\min}, s_{i\min} + 0.5pc], \quad ck_i = c_{i1}, \\ c_j &= [\max(c_{j-1}), \max(c_{j-1}) + pc], \\ ck_j &= 0.5(c_{j1} + c_{j2}), \quad j = 2, 3, \dots, k-1, \\ c_k &= [\max(c_{k-1}), \max(c_{k-1}) + 0.5pc], \\ ck_k &= c_{k2}, \end{aligned} \tag{4}$$

где  $k$  – количество кластеров переменной;  $c_i$  – интервалы кластеров;  $ck_i$  – центры кластеров [14].

В параметрах  $c_{ij}$  выражения (4) во втором индексе 1 означает левую границу интервала ( $c_{i1}$ ), 2 – правую ( $c_{i2}$ ). Следует отметить, что в (4) под центром кластера понимается не его геометрический центр, а точка, к которой присоединяются соседние функции принадлежности. Выражения (4) для рядов экспериментальных данных позволяют поставить в соответствие номера кластеров, в которые они попадают.

Для каждого экспериментального сигнала вводится лингвистическая переменная  $L$ , которая характеризуется набором

$$L = \langle N, T, X \rangle, \tag{5}$$

где  $N$  – имя лингвистической переменной;  $T = \{“j”\}$ ,  $j = 1, 2, \dots, k$  – терм-множество лингвистической переменной;  $X = [s_{\min}, s_{\max}]$  – область определения элементов из  $T$  (базовое множество).

В качестве элементов терм-множества  $T$  будем использовать функции принадлежности z-типа для граничных термов и треугольные для всех остальных.

Следует отметить, что в работах [9, 10, 14] использовались только треугольные функции принадлежности. Однако по результатам ис-

следований установлено, что после синтеза нечеткого регулятора и апробации его с объектом управления могут возникать коллизии из-за выхода сигналов за пределы универсальных множеств и, как следствие, некорректного нечеткого вывода. Кроме того, коэффициенты аналитических выражений функций принадлежности зависят от параметров соседних кластеров, что позволяет создавать нечеткие переменные, как показано на рисунке 2.

Как видно из этого рисунка, количество функций принадлежности равно количеству кластеров. Для учета неопределенности каждая функция принадлежности превышает размеры своего кластера и перекрывается соседними.

**Этап 4.** Создание базы нечетких правил. База правил нечеткого регулятора формируется автоматически после определения принадлежности точек экспериментальных данных кластерам, как представлено в таблице, где  $W$  – количество точек экспериментальных данных, попавших в кластер (данный параметр необходим на следующем этапе для анализа и коррекции базы правил).

**Пример базы нечетких правил**  
**An example of a fuzzy rules database**

№	Входные переменные				Выходная переменная	Количество измененных значений
	$T_{inp1}$	$T_{inp2}$	...	$T_{inpM}$	$T_{out}$	
1	2	4	...	6	2	18
2	3	1	...	11	8	35
3	6	2	...	4	1	4
...	...	...	...	...	...	...

**Этап 5.** Анализ базы правил. При анализе базы правил необходимо определить ее полноту и непротиворечивость.

Полная база ( $R_F$ ) должна содержать  $N_{inp1} \cdot N_{inp2} \cdot \dots \cdot N_{inpM}$  нечетких правил. Если количество правил меньше  $R_F$ , база правил неполная. Если количество правил больше  $R_F$ , база правил избыточная. Следует отметить, что при создании нечетких регуляторов по экспериментальным данным нельзя гарантировать получение полной базы правил, так как не все сочетания элементов терм-множеств входных лингвистических переменных могут попасть в нее.

Также база правил может быть противоречивой, если в ней присутствуют правила с одинаковым антецедентом и разным консеквентом. Проверку на полноту и непротиворечивость можно выполнить аналитически.

**Этап 6.** Коррекция базы правил. В случае наличия противоречивых правил необходимо откорректировать базу правил. Можно либо свернуть противоречивые правила и синтезировать одно правило с обобщающим элементом терм-множества в консеквенте, либо исключить из противоречивых правил те, которые имеют меньшее количество попаданий экспериментальных точек  $W$  (см. таблицу).

Свертка правил была выполнена по методу из работы [14]. Выбираются центры кластеров консеквентов  $ck$  избыточных правил и ищется среднее арифметическое по формуле

$$s = \frac{\sum_{j=1}^p ck_j}{p}, \tag{6}$$

где  $p$  – количество избыточных правил с одинаковым антецедентом.

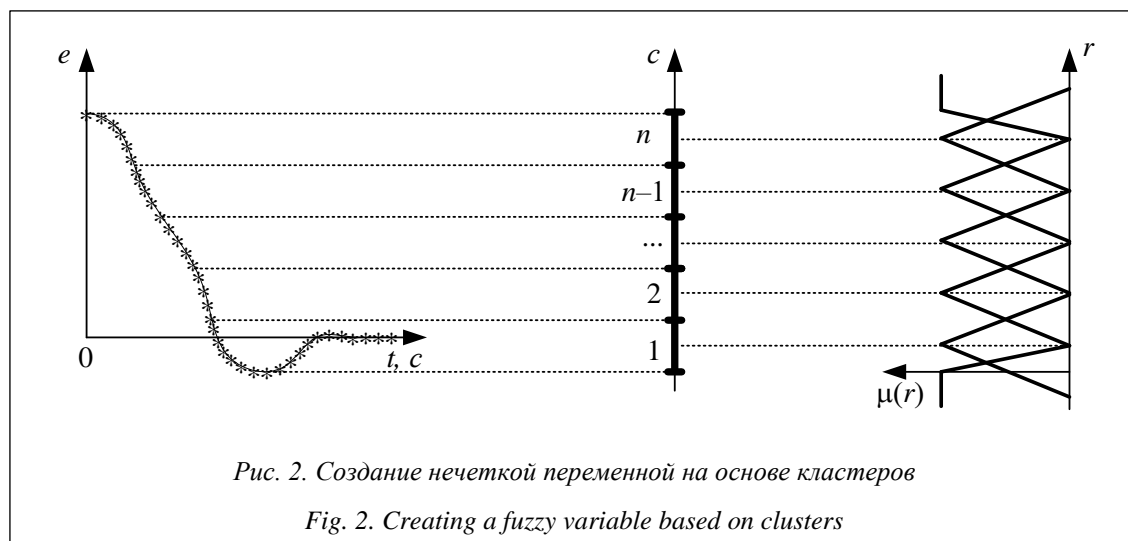


Рис. 2. Создание нечеткой переменной на основе кластеров

Fig. 2. Creating a fuzzy variable based on clusters

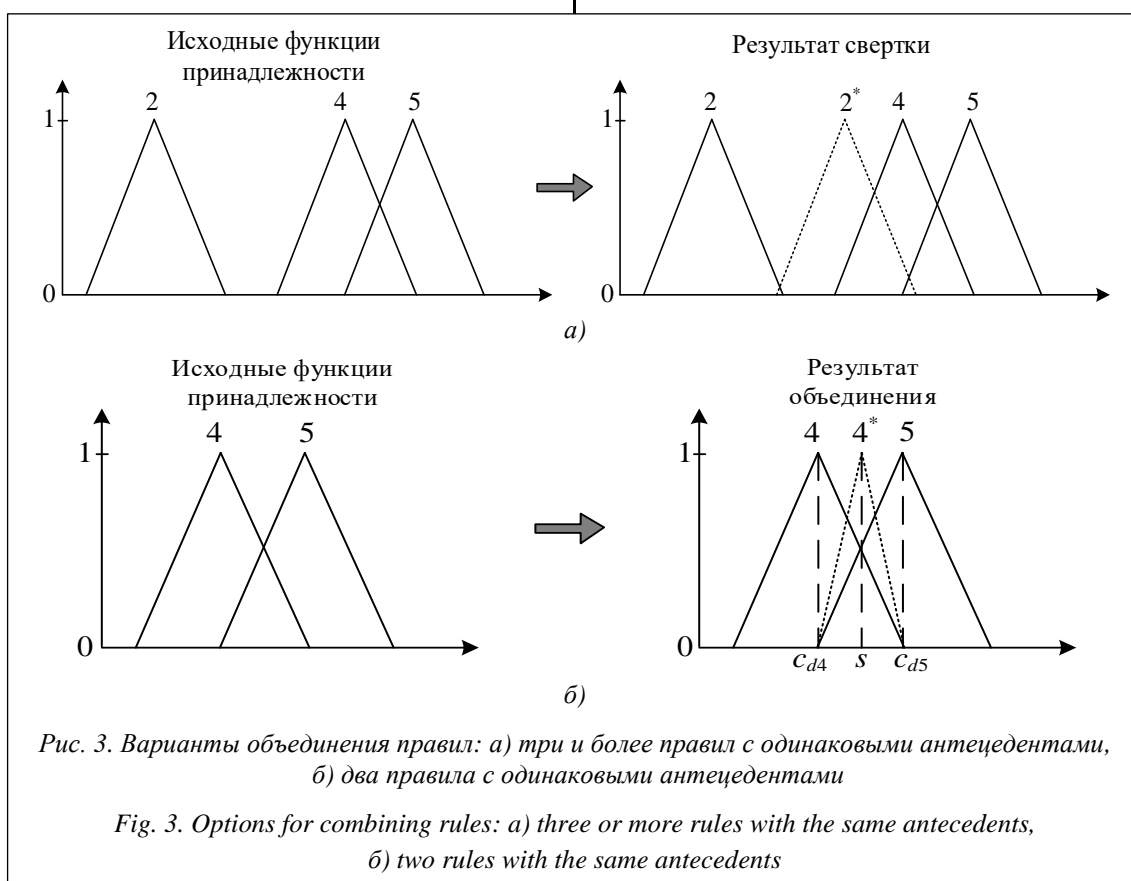
Затем определяется принадлежность  $s$  одному из кластеров из  $T_{out}$ , его номер назначается консеквенту в одном из избыточных правил, а остальные удаляются. Следует отметить, что если избыточных правил два и их консеквенты соответствуют расположенным рядом термам, то использование выражения (6) приводит к неопределенности. Для данного случая нужно ввести в терм-множество выходной лингвистической переменной дополнительный терм с параметрами:

$$a_j = c_{d(j-1)}, b_j = s, d_j = c_{d(j+1)},$$

где  $j - 1, j + 1$  – центры кластеров, образующих функции принадлежности консеквентов избыточных правил (рис. 3).

выступают входные и выходная лингвистические переменные с заданными универсальными и терм-множествами, база правил в виде массива, столбцы которого определяют термы входных переменных, выходной переменной и вес правила. Создается пустая *fis*-структура [15], в которую последовательно заносятся данные о входных и выходной переменных командой *addvar*, о термах – командой *addmf*, о базе правил – командой *addrule*.

**Этап 8.** Верификация и анализ результата. Сформированная *fis*-структура сохраняется в файл и затем экспортируется в блок *Fuzzy Controller* в среде *Simulink* для верификации и анализа. Сначала сравниваются вы-



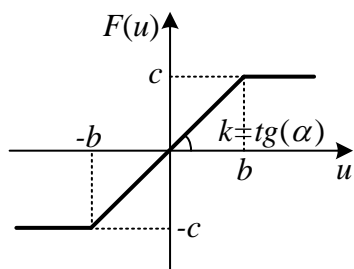
Выполняя данную процедуру для всех наборов избыточных правил, получаем редуцированную базу правил.

**Этап 7.** Синтез нечеткого регулятора. К началу данного этапа уже определены терм-множества входных и выходных лингвистических переменных, получена и редуцирована база правил. Авторы разработали программный модуль в среде *MatLab 2018b* для создания нечеткого регулятора. В качестве входных параметров программного модуля

ходные сигналы классического и нечеткого контроллеров, а затем выходные сигналы с объекта управления с обоими регуляторами.

### Результаты моделирования

В качестве объекта управления была выбрана математическая модель из работы [2], которая описывается нелинейным звеном типа «насыщение»



$$F(u) = \begin{cases} -c, & u < -b, \\ ku, & |u| \leq b, \\ c, & u > b, \end{cases}$$

и последовательно включенной передаточной функцией  $G(s) = \frac{1}{s^2 + 0,05s + 1}$ .

В качестве классического регулятора использовался ПИД-регулятор с параметрами  $K_p = 14,18$ ,  $K_d = 6,81$  и ограничением на амплитуду сигнала производной ошибки в интервале  $[-2, 2]$ . На вход системы подавалось единичное ступенчатое воздействие. В нелинейном звене установлена величина насыщения  $|c| = 10$ .

По результатам моделирования классического регулятора были записаны входные и выходные сигналы в количестве 500 точек с шагом дискретизации 0,01 секунды. В соответствии с этапами 1–7 синтезирован нечеткий регулятор, содержащий входные пере-

менные с 11 термами и выходную переменную с 7 термами, как показано на рисунке 4.

Результаты сравнения классического и нечеткого регуляторов отражены на рисунке 5.

Таким образом, сгенерирована база, содержащая 62 нечетких правила, которая была редуцирована до 46 правил из-за правил с одинаковым antecedентом.

### Обсуждение результатов

По результатам моделирования синтезирован нечеткий регулятор для объекта управления в виде нелинейного звена и передаточной функции второго порядка, у которого входная лингвистическая переменная «ошибка» содержала 11 термов на универсальном множестве  $[-0,16, 1]$ , лингвистическая переменная «производная ошибки» – 11 термов на универсальном множестве  $[-0,5, 0,5]$ , лингвистическая переменная «сигнал управления» – 7 термов на универсальном множестве  $[-1,53, 532,81]$ . Размер универсальных множеств определялся автоматически по диапазонам изменения экспериментальных данных. Количество термов для каждой переменной задавалось вручную. Как видно из результатов моделирования, база нечетких правил неполная, так как должна содержать 121 правило. Это можно устранить путем снятия экспериментальных

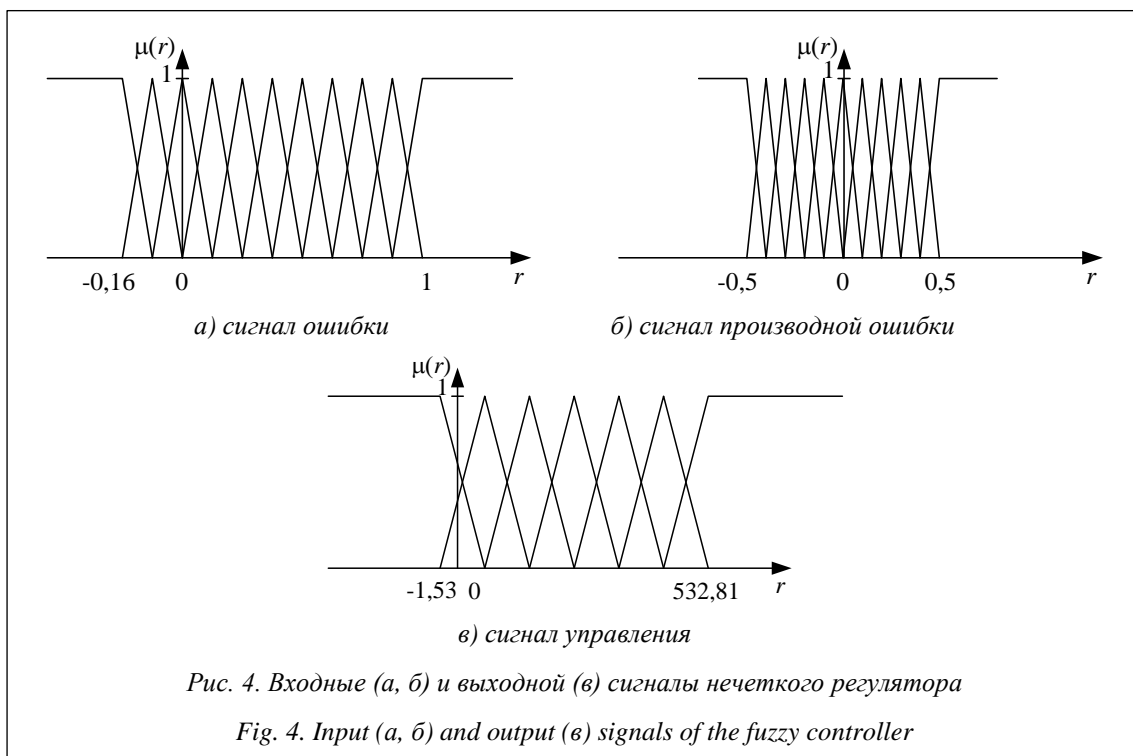


Рис. 4. Входные (а, б) и выходной (в) сигналы нечеткого регулятора

Fig. 4. Input (a, б) and output (в) signals of the fuzzy controller

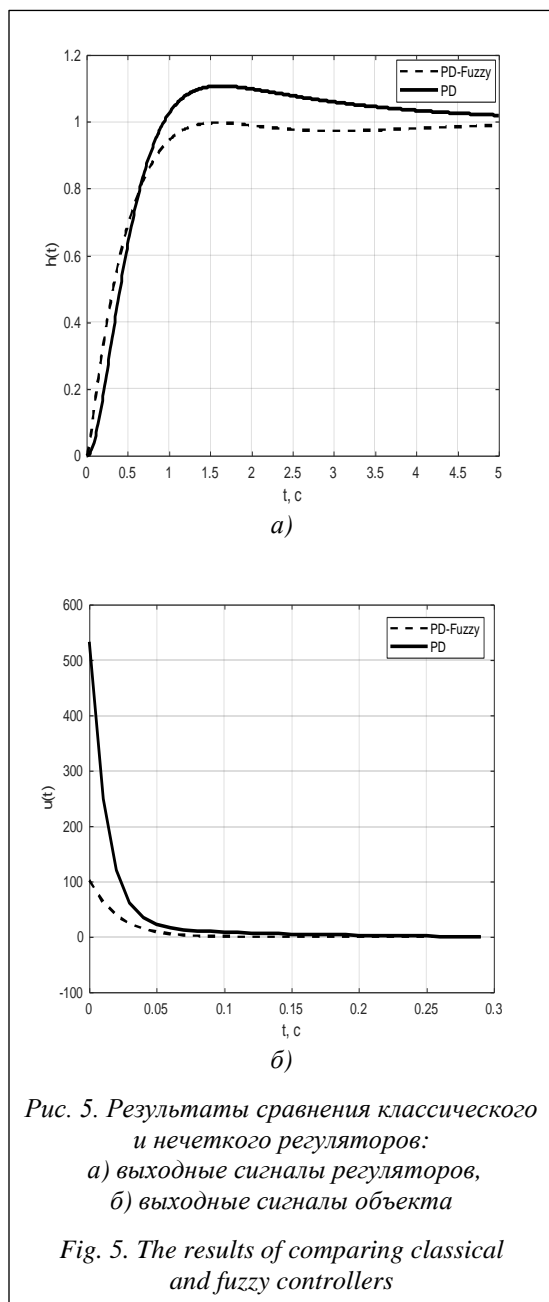


Рис. 5. Результаты сравнения классического и нечеткого регуляторов:  
 а) выходные сигналы регуляторов,  
 б) выходные сигналы объекта

Fig. 5. The results of comparing classical and fuzzy controllers

данных при разных входных сигналах, чтобы ряды экспериментальных данных содержали больше разных значений.

Результаты сравнения показывают, что система управления с нечетким регулятором обеспечивает меньшую длительность переходного процесса и отсутствие перерегулирования (рис. 5а), а также меньшие энергетические затраты на управление (рис. 5б).

## Заключение

Отличительной особенностью предлагаемого метода является возможность автоматического определения параметров нечеткого регулятора по экспериментальным данным с редуцированной базой правил.

При применении метода определяется максимальное количество кластеров для заданных рядов данных с учетом ограничений на их количество для выходного сигнала регулятора, что положительно сказывается на его работоспособности, так как задействуются все элементы терм-множеств лингвистических переменных в базе нечетких правил. Параметры функций принадлежности вычисляются таким образом, чтобы обеспечить перекрытие между соседними подмножествами на 50 %, что обеспечивает равномерное покрытие экспериментальных данных и устраняет неопределенность в алгоритме нечеткого вывода Мамдани. Рассмотрены два способа редукции базы правил за счет их объединения или удаления. В результате исследований авторы пришли к следующему выводу: если количество точек экспериментальных данных в кластерах, соответствующих функциям принадлежности konsekventa у дублирующих правил, отличается в десять и более раз, удаление избыточных правил предпочтительнее их объединения, так как улучшаются показатели качества управления. В противном случае целесообразно использовать объединение правил, например, способом, предложенным в данной статье, однако вопрос уточнения параметров объединенной функции принадлежности все еще остается открытым.

Количество термов лингвистических переменных задавалось вручную, но эксперименты показывают, что его увеличение не всегда положительно сказывается на качестве управления, так как в таких случаях нечеткая модель полностью аппроксимирует классический регулятор и, как следствие, его недостатки.

Дальнейшая работа авторов направлена на оптимизацию количества термов лингвистических переменных и различных структурных реализаций нечетких регуляторов.

## Литература

1. Бураков М.В., Коновалов А.С., Яковец О.Б. Эволюционный синтез нечетких регуляторов // Информационно-управляющие системы. 2015. Т. 6. С. 28–33. DOI: 10.15217/issn1684-8853.2015.6.28.



2. Соловьев К.А., Муравьева Е.А., Султанов Р.Г. Адаптация нечеткого регулятора // Нефтегазовое дело. 2014. Т. 12. № 3. С. 123–128.
3. Kostoglotov A.A., Lazarenko S.V., Pugachev I.V. Method of synthesis of multi-mode control under the expected uncertainty using the analysis of the phase-space decomposition on the basis of the generalized power maximum condition. AIP Conference Proceedings, 2019, vol. 2188, art. 030005. DOI: 10.1063/1.5138398.
4. Jinsen Z., Wu-Hua Ch., Xiaomei L., Jing W. Fuzzy controller synthesis for nonlinear neutral state-delayed systems with impulsive effects. Information Sciences, 2020, vol. 555, pp. 293–313. DOI: 10.1016/j.ins.2020.11.001.
5. Alsafadi L.A., Chulin N.A., Mironova I.V. Synthesis of fuzzy controller based on simple PID controller. Procedia Computer Science, 2019, vol. 150, pp. 28–38. DOI: 10.1016/j.procs.2019.02.008.
6. Berdnikov V., Lokhin V. Synthesis of guaranteed stability regions of a nonstationary nonlinear system with a fuzzy controller. Civil Engineering J., 2019, vol. 5, no. 1, pp. 107–116. DOI: 10.28991/cej-2019-03091229.
7. Pobedinsky V.V., Gorokhovskiy A.G., Hertz E.F., Shishkina E.E., Pobedinsky A.A. Synthesis of a fuzzy model in MatLab for automatic control of the drying process. Proc. Int. Sci. and Pract. Conf. AgroSMART, 2018, vol. 151, pp. 243–248. DOI: 10.2991/agrosmart-18.2018.47.
8. Uskov A., Shchokin V., Mykhailenko O., Kryvenko O. The fuzzy logic controllers synthesis method in the vector control system of the wind turbine doubly-fed induction generator. E3S Web of Conferences, 2020, vol. 166, art. 04006. DOI: 10.1051/e3sconf/202016604006.
9. Muravyova E.A., Radakina D.S. Using the fuzzy controller to control process parameters. Int. Sci. J. "INDUSTRY 4.0", 2018, vol. 3, no. 5, pp. 236–239.
10. Dutu L.C., Mauris G., Bolon Ph. A fast and accurate rule-base generation method for Mamdani fuzzy systems. IEEE Transactions on Fuzzy Systems, 2018, vol. 26, no. 2, pp. 715–733. DOI: 10.1109/TFUZZ.2017.2688349.
11. Ignatyev V., Kovalev A., Spiridonov O., Kureychik V., Soloviev V., Ignatyeva A. A method of optimizing the rule base in the Sugeno fuzzy inference system using fuzzy cluster analysis. IREE, 2020, vol. 15, no. 4, pp. 316–327. DOI: 10.15866/iree.v15i4.16545.
12. Ignatyev V.V., Soloviev V.V., Beloglazov D.A., Boldyreff A.S. Development of a method for automatic generation and optimization of fuzzy controller parameters using genetic algorithm. In: Artificial Intelligence in Intelligent Systems, Silhavy R. (ed.), vol. 229, pp. 404–416. DOI: 10.1007/978-3-030-77445-5\_38.
13. Volosencu C. Properties of fuzzy systems. WSEAS Transactions on Systems, 2009, vol. 8, pp. 210–228.
14. Ignatyev V., Soloviev V., Beloglazov D., Kureychik V., Ignatyeva A., Vorotova A. System for automatic adjustment of intelligent controller parameters. Proc. CIT&DS, 2019, vol. 1084, pp. 226–242. DOI: 10.1007/978-3-030-29750-3\_18.
15. Ojha V., Abraham A., Snasel V. Heuristic design of fuzzy inference systems: A review of three decades of research. Engineering Applications of Artificial Intelligence, 2019, vol. 85, pp. 845–864. DOI: 10.1016/j.engappai.2019.08.010.

### A synthesis method for fuzzy controllers based on clustering

V.V. Ignatev<sup>1</sup>, Ph.D. (Engineering), Leading Researcher, vvignatev@sfedu.ru

V.V. Solovev<sup>1</sup>, Senior Lecturer

<sup>1</sup> Southern Federal University, Taganrog, 347922, Russian Federation

**Abstract.** The goal of this work is to develop a method for synthesizing fuzzy controllers from experimental data based on clustering, since this is the simplest way to determine the number of membership functions and create a rule base. To achieve this goal, it is proposed to use experimental data on the input and output signals of the control system for a technical object with a classical controller. The developed

method for data clustering is based on the experimental data and makes it possible to determine the term-sets of input and output linguistic variables of a fuzzy controller that implements the Mamdani fuzzy inference algorithm and to compose a rule base.

Clustering is performed by evaluating the boundaries of the experimental data variation intervals, uniform division into clusters depending on the required power of term-sets of linguistic variables and determining if the data belongs to certain clusters. Since the experimental data are connected, that is, for each moment of time, data on both the input and output signals of the classical controller are stored and their belonging to clusters is determined, the development of the fuzzy controller rule base does not cause difficulties.

To simplify the research, the authors have developed software in the MatLab environment. It makes it possible both to obtain experimental data, to synthesize a fuzzy controller and check its performance. The control system model is developed in the Simulink environment, the method of clustering and determining the parameters of linguistic variables is implemented as a program in an m-file; the fuzzy controller is implemented in the Fuzzy Logic Toolbox extension package.

The high degree of integration of MatLab expansion packages made it possible to simplify the procedure of synthesizing fuzzy controllers as much as possible and to reduce it to determining the number of input and output variables and analyzing the simulation results. The paper presents the process of filling connected containers as an example, its mathematical model is a second-order transfer function with delay. To select an optimal structure of a fuzzy controller, the authors have carried out a study based on experimental data. The results obtained in this work were compared with the classical PD-controller, the model of which was implemented in the Simulink environment. The research results will be useful for developers of fuzzy control models.

**Keywords:** fuzzy controller, PID-controller, clustering, rule weight, rule base reduction, automatic synthesis rule base.

### References

1. Burakov M.V., Kononov A.S., Yakovets O.B. Evolutionary design of fuzzy controllers. *Information and Control Systems*, 2015, vol. 6, pp. 28–33. DOI: 10.15217/issn1684-8853.2015.6.28 (in Russ.).
2. Solovov K.A., Muraveva E.A., Sultanov R.G. Adaptation of fuzzy controller. *Oil and Gas Business*, 2014, vol. 12, no. 3, pp. 123–128 (in Russ.).
3. Kostoglotov A.A., Lazarenko S.V., Pugachev I.V. Method of synthesis of multi-mode control under the expected uncertainty using the analysis of the phase-space decomposition on the basis of the generalized power maximum condition. *AIP Conference Proceedings*, 2019, vol. 2188, art. 030005. DOI: 10.1063/1.5138398.
4. Jinsen Z., Wu-Hua Ch., Xiaomei L., Jing W. Fuzzy controller synthesis for nonlinear neutral state-delayed systems with impulsive effects. *Information Sciences*, 2020, vol. 555, pp. 293–313. DOI: 10.1016/j.ins.2020.11.001.
5. Alsafadi L.A., Chulin N.A., Mironova I.V. Synthesis of fuzzy controller based on simple PID controller. *Procedia Computer Science*, 2019, vol. 150, pp. 28–38. DOI: 10.1016/j.procs.2019.02.008.
6. Berdnikov V., Lokhin V. Synthesis of guaranteed stability regions of a nonstationary nonlinear system with a fuzzy controller. *Civil Engineering J.*, 2019, vol. 5, no. 1, pp. 107–116. DOI: 10.28991/cej-2019-03091229.
7. Pobedinsky V.V., Gorokhovskiy A.G., Hertz E.F., Shishkina E.E., Pobedinsky A.A. Synthesis of a fuzzy model in MatLab for automatic control of the drying process. *Proc. Int. Sci. and Pract. Conf. AgroSMART*, 2018, vol. 151, pp. 243–248. DOI: 10.2991/agrosmart-18.2018.47.
8. Uskov A., Shchokin V., Mykhailenko O., Kryvenko O. The fuzzy logic controllers synthesis method in the vector control system of the wind turbine doubly-fed induction generator. *E3S Web of Conferences*, 2020, vol. 166, art. 04006. DOI: 10.1051/e3sconf/202016604006.
9. Muravyova E.A., Radakina D.S. Using the fuzzy controller to control process parameters. *Int. Sci. J. "INDUSTRY 4.0"*, 2018, vol. 3, no. 5, pp. 236–239.
10. Dutu L.C., Mauris G., Bolon Ph. A fast and accurate rule-base generation method for Mamdani fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 2018, vol. 26, no. 2, pp. 715–733. DOI: 10.1109/TFUZZ.2017.2688349.
11. Ignatyev V., Kovalev A., Spiridonov O., Kureychik V., Soloviev V., Ignatyeva A. A method of optimizing the rule base in the Sugeno fuzzy inference system using fuzzy cluster analysis. *IREE*, 2020, vol. 15, no. 4, pp. 316–327. DOI: 10.15866/iree.v15i4.16545.

12. Ignatyev V.V., Soloviev V.V., Beloglazov D.A., Boldyreff A.S. Development of a method for automatic generation and optimization of fuzzy controller parameters using genetic algorithm. In: *Artificial Intelligence in Intelligent Systems*, Silhavy R. (ed.), vol. 229, pp. 404–416. DOI: 10.1007/978-3-030-77445-5\_38.
13. Volosencu C. Properties of fuzzy systems. *WSEAS Transactions on Systems*, 2009, vol. 8, pp. 210–228.
14. Ignatyev V., Soloviev V., Beloglazov D., Kureychik V., Ignatyeva A., Vorotova A. System for automatic adjustment of intelligent controller parameters. *Proc. CIT&DS*, 2019, vol. 1084, pp. 226–242. DOI: 10.1007/978-3-030-29750-3\_18.
15. Ojha V., Abraham A., Snasel V. Heuristic design of fuzzy inference systems: A review of three decades of research. *Engineering Applications of Artificial Intelligence*, 2019, vol. 85, pp. 845–864. DOI: 10.1016/j.engappai.2019.08.010.

#### Для цитирования

Игнатъев В.В., Соловьев В.В. Метод синтеза нечетких регуляторов на основе кластеризации // Программные продукты и системы. 2021. Т. 34. № 4. С. 597–607. DOI: 10.15827/0236-235X.136.597-607.

#### For citation

Ignatev V.V., Solovev V.V. A synthesis method for fuzzy controllers based on clustering. *Software & Systems*, 2021, vol. 34, no. 4, pp. 597–607 (in Russ.). DOI: 10.15827/0236-235X.136.597-607.