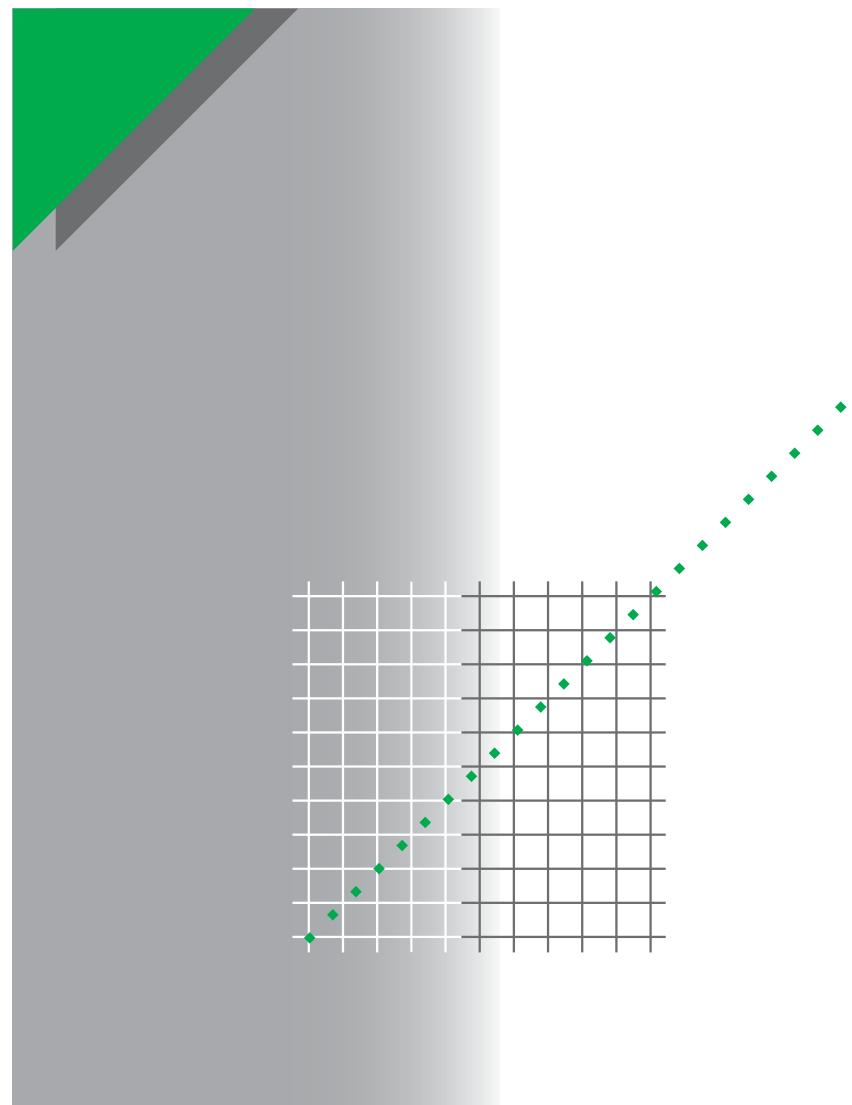


## Выпуск № 31

### Нечеткая логика



*Компания Schneider Electric приступила к выпуску «Технической коллекции Schneider Electric» на русском языке.*

**Техническая коллекция** представляет собой серию отдельных выпусков для специалистов, которые хотели бы получить более подробную техническую информацию о продукции Schneider Electric и ее применении, в дополнение к тому, что содержится в каталогах.

В **Технической коллекции** будут публиковаться материалы, которые позволяют лучше понять технические и экономические проблемы и явления, возникающие при использовании электрооборудования и средств автоматизации Schneider Electric.

**Техническая коллекция** предназначена для инженеров и специалистов, работающих в электротехнической промышленности и в проектных организациях, занимающихся разработкой, монтажом и эксплуатацией электроустановок, распределительных электрических сетей, средств и систем автоматизации.

Техническая коллекция будет также полезна студентам и преподавателям ВУЗов. В ней они найдут сведения о новых технологиях и современных тенденциях в мире Электричества и Автоматики.

В каждом выпуске **Технической коллекции** будет углубленно рассматриваться конкретная тема из области электрических сетей, релейной защиты и управления, промышленного контроля и автоматизации технологических процессов.

**Валерий Саженков,**  
Технический директор  
ЗАО «Шнейдер Электрик»,  
Кандидат технических наук

# Выпуск № 31

## Нечеткая логика



Франсуа ШЕВРИ

Пришел в компанию Telemecanique в 1987; принят в лабораторию «Advanced Automation Laboratory» Исследовательского отдела в 1993. Закончил CNAM Industrial Automation engineering, его диссертация была посвящена интегрированию алгоритмов нечеткой логики в программируемые контроллеры Schneider. Он принимал активное участие в подготовке продуктов с нечеткой логикой для семейства ПЛК Micro/Premium, и внедрению данной техники, в частности в автомобильной и пищевой промышленности.

---



Франсуа ГЕЛИ

После окончания Ecole Centrale de Paris в 1988, он поступил в отделение компании Telemecanique в Японии в 1990 и получил награду за его работы в области применения нечеткой логики в автоматизированном управлении в 1994. Он руководил отделом «Advanced Automatic Department» компании Schneider с 1995, где он занимался подготовкой библиотеки нечеткой логики для стандартных языков МЭК для ПЛК.

---

## Словарь терминов

---

**Активизация:**

См. «Степень истинности».

**Результат:**

Результат правила представляет собой запись, включающую в себя лингвистическую переменную и лингвистическое описание, расположенные после *then* в правиле. Результат может состоять из комбинации нескольких записей.

**Условие:**

См. «Утверждение».

**Объединение данных:**

Объединение данных заключается в извлечении из некоторого объема данных, одного или более элемента, которые могут быть разных типов.

Например: из переменных R, V и B, которые представляют цвет бисквита, можно получить представление о стадии готовности бисквита. Также используется термин "Объединение датчиков".

**Обратное преобразование (дефазификация):**

Обратное преобразование нечеткого множества лингвистических выходных переменных в числовые значения, выполняемое после формирования нечеткого логического вывода.

**Степень активизации:**

См. «Степень истинности».

**Степень принадлежности:**

Элемент x принадлежит нечеткому множеству A со степенью принадлежности, находящейся в диапазоне от 0 до 1. Значение степени принадлежности дает функция принадлежности  $\mu_A(x)$ .

**Степень истинности:**

Степень истинности или степень активизации правила принимает значение в диапазоне от 0 до 1 и вычисляется из значения степени принадлежности утверждения правила. От нее напрямую зависит результат данного правила. При этом правило считается активным при значении y.

**Подготовка задачи для решения методами нечеткой логики (фазификация):**

Преобразование числовых значений в степени принадлежности к нечеткому множеству с помощью оценки функции принадлежности.

**Нечеткое множество:**

В классической теории множеств, характеристическая функция определяет множество: значением функции может быть одно из двух дискретных величин: 0 (элемент не принадлежит множеству) или 1 (элемент принадлежит множеству). Нечеткое множество определяется функцией принадлежности, значением которой может быть любое действительное число в промежутке от 0 до 1.

**Предположение:**

Вычисление степеней активизации для всех правил в базе, а также всех нечетких множеств лингвистических переменных, содержащихся в результатах этих правил.

**База знаний:**

Полный набор функций принадлежности и правил системы нечеткого управления, включая результаты экспертиз, опыт операторов, инженеров и т.д.

**Лингвистический терм:**

Описание, связанное с функцией принадлежности и характеризующее лингвистическую переменную.

**Лингвистическая переменная:**

Числовая переменная, имеющая имя (давление, температура и т.д.), с которой связаны лингвистические описания.

**Функция принадлежности:**

Функция  $\mu_A(x)$  для любого входного значения x связана со степенью его принадлежности к множеству A. Значение степени принадлежности находится в диапазоне [0; 1].

**Утверждение:**

Также можно назвать предложением или условием; утверждение правила является записью, состоящей из лингвистической переменной и лингвистического описания, записанными между *if* и *then* правила. Утверждение может представлять собой комбинацию из нескольких записей, связанных операторами AND, OR, NOT.

**Предположение:**

См. «Утверждение».

**Объединение датчиков:**

См. «Объединение данных».

**Одноэлементное множество:**

Функция принадлежности  $\mu_A(x)$ , которая равна нулю для всех значений переменной x, кроме единственного значения  $x_0$ .

# Нечеткая логика

Изначально это была только теория, а в настоящее время Нечеткая логика превратилась в полноценную методику управления. Она используется совместно с другими методиками управления, удачно дополняя их.

Нечеткая логика совсем не заменяет традиционные методики управления, а напротив она используется совместно с традиционными методиками и позволяет упростить создание и расширить возможности традиционных методик. Преимущества базируются на следующем:

- формализуется и объединяется опыт операторов и разработчиков в настройке пульта регулирования;
- предлагается простой метод управления для сложных процессов;
- постоянно учитывается опыт по управлению процессами данного типа, принимая во внимание исключения разного рода и особенности системы;
- учитываются исходные данные разного рода и производится объединение разных исходных данных.

Каким образом эти методики влияют на процесс промышленного управления? Какое влияние на качество и стоимость произведенной продукции?

При помощи рассмотрения нескольких теоретических понятий, примеров автоматизации промышленных объектов и конкурентных преимуществ данная статья технической коллекции отвечает на поставленные выше вопросы, а также на возможные вопросы инженеров систем автоматизации и вопросы потенциальных пользователей.

## Содержание

	Стр.
<b>1. Предисловие</b>	<b>4</b>
1.1. Нечеткая логика сегодня	4
1.2. История нечеткой логики	4
1.3. Универсальность и использование нечеткой логики в системах управления	5
<b>2. Теория нечетких множеств</b>	<b>6</b>
2.1. Понятие частичной принадлежности к множеству	6
2.2. Функция принадлежности	6
2.3. Операторы нечеткой логики	8
2.4. Правила нечеткой логики	9
<b>3. Пример учебного приложения</b>	<b>14</b>
3.1. Предисловие	14
3.2. Представление учебного приложения	14
3.3. Лингвистические переменные и термы	15
3.4. Правила и выходы	15
<b>4. Применение</b>	<b>16</b>
4.1. Области применения баз нечетких правил	16
4.2. Разработка приложения	16
4.3. Использование приложения	17
4.4. Выбор применяемой технологии	17
4.5. Стандарты	18
<b>5. Приложения нечеткой логики</b>	<b>19</b>
5.1. Типы приложений	19
5.2. Примеры достижений в промышленности	20
<b>6. Заключение</b>	<b>24</b>
<b>Приложение</b>	<b>26</b>
<b>Библиография</b>	<b>28</b>

# 1. Предисловие

## 1.1. Нечеткая логика сегодня

В большинстве, используемых в настоящее время приложений автоматизации технологических процессов, нечеткая логика позволяет применить опыт операторов и технологов для управления процессами.

В начале девяностых годов двадцатого века появился интерес к методам управления, которые получили название «нечеткая логика управления». В то время в основном в Японии появилось большое количество электротехнических и электронных приложений в бытовой технике, в которых были применены методы управления данного типа. Стиральные машины, работающие автоматически и не требующие какой-либо регулировки, видеокамеры с технологией Steadyshot (TM) обеспечивающей автоматическую стабилизацию изображения и многие другие инновации стали привлекать внимание широкой общественности к термину «Нечеткая логика».

В автомобильной промышленности благодаря применению методов нечеткой логики появилось автоматическое переключение передач в трансмиссии, системы впрыска, шумоподавляющие системы и кондиционеры воздуха.

Количество приложений основанных на данных методах управления непрерывно увеличивается для непрерывных процессов, для приложений пакетной обработки, а также для автоматизированных систем (рассматриваются в данной статье). Нечеткая логика, благодаря использованию её в этой отрасли, получила описание и формулировку в качестве метода программирования. Она позволяет систематизировать эмпирические знания и применить их для управления процессами в случае трудностей с применением классических методов управления. Теория нечеткой логики позволяет описать наборы методов управления, которые несложно применить для реальной системы и позволяет учсть опыт операторов и технологов для динамического управления процессом.

Это позволяет описывать на Нечеткой логике отдельные части производственного процесса, такие как инициализация, задание параметров и т.д.

Эта статья Технической коллекции описывает методы применения Нечеткой логики для использования её в автоматизации производственных процессов.

## 1.2. История нечеткой логики

### Появление Нечеткой логики

Термин «нечеткое множество» ("fuzzy set") впервые появился в 1965, когда профессор Лотфи А.Заде (Lotfi A. Zadeh) из университета в Барклей (Berkeley), USA опубликовал статью озаглавленную "Fuzzy sets". В это время он сформулировал множество теоретических приемов описания алгоритмов в этой области, после чего многочисленные теоретики подхватили идею и стали разрабатывать свои описания.

### Первые применения

В то же время некоторые исследователи начали применять методы Нечеткой логики для решения проблем, которые считались сложными. В 1975 году профессор Мамдани (Mamdani) из Лондона разработал и опубликовал методику управления двигателем паровой турбины. В 1978 году датская компания F.L. Smidt разработала систему управления для печи обжига кирпича. Это было первым применением в промышленности системы на основе нечеткой логики.

### Творческий бум

Применение нечеткой логики вызвало настоящий бум творческой активности в Японии, где исследования были не только теоретические, но и прикладной, ориентированный на применения, характер. В конце восьмидесятых термин «Нечеткая логика» получил широкое распространение, потому что эти методы управления стали применять в стиральных машинах, видеокамерах и другой бытовой технике.

Промышленные применения, такие как водоподготовка, портовые краны, системы управления в метро, а также системы вентиляции и кондиционирования воздуха начали так же широко использовать Нечеткую логику. В завершении внедрения методов управления на основе нечеткой логики в промышленности, её стали использовать в системах моделирования в финансовой сфере и диагностики в медицине. Начиная с 1990, в Германии и США стали создавать большое количество приложений на основе методов нечеткой логики.

## 1.3. Универсальность и использование нечеткой логики в системах управления

### Универсальность

Нечеткая логика базируется на следующих наблюдениях:

- Знания и умения, которые человек часто использует для разрешения какой-либо проблемы, являются не совершенными:
  - они могут быть сомнительными (человек может быть не уверен в их эффективности);
  - или не проверенными.
- Человек часто решает сложные проблемы на основе приблизительных исходных данных (точность исходных данных при этом не требуется), например, для того чтобы выбрать квартиру для проживания человек может рассматривать разные исходные данные, среди которых могут быть район, близость магазинов, расстояние до работы и стоимость аренды. При этом, однако, не требуется точность всех параметров исходной информации.
- В промышленности операторы очень часто с легкостью решают весьма сложные и комплексные проблемы без предварительной проработки возможной проблемы и моделирования системы. Подобно тому, как и для управления автомобилем не требуется предварительное моделирование поездки, не смотря на то, что автомобиль является очень сложной системой и поездка может быть не простой.
- Чем сложнее система, тем сложнее её моделирование и предсказание её поведения во время работы.

Из всего сказанного выше можно сделать следующие выводы:

- часто проще и полезнее моделировать поведение оператора системы управления, чем моделировать работу самой системы;
- вместо того, чтобы использовать точные математические вычисления и уравнения, более эффективно использовать качественные оценки ситуации и применять соответствующие меры обработки.

### Использование в системах управления

«Нечеткая логика», называемая “fuzzy control”, хорошо известна инженерам программистам систем управления, как удобное средство программирования и мониторинга приложений управления технологическими процессами. По аналогии с традиционными средствами управления технологическими процессами, системы на основе

нечеткой логики могут использоваться для описания петель регулирования и участвовать в вычислении управляющего воздействия в соответствии с одной или большим количеством точек задания для одного или большего количества измерений.

Правила нечеткой логики позволяют обеспечить:

- применение существующего опыта управления;
- использовать гибкие правила в случае невозможности точно моделировать систему при помощи традиционных средств;
- улучшение качества управления при помощи:
  - саморегулирования системы управления;
  - упреждающее изменение выходного воздействия (функция упреждения), базируясь на событиях, которые не могут быть учтены в случае применения традиционных способов управления.

### Применения ноу-хау наилучшим образом

Жизненный опыт и ноу-хау наилучшим образом могут быть применены при управлении с нечеткой логикой. Нечеткие правила позволяют производить управление в случае, когда нет возможности управления в ручном режиме или по известным правилам.

Когда существует накопленный опыт и/или ноу-хау их можно трансформировать в правила нечеткой логики и обеспечить управление наиболее простым способом. Нечеткая логика, кроме этого, позволяет получить максимальную пользу от практического опыта и обеспечить отсутствие потерь.

Этапу накапливания опыта и выявления полезных ноу-хау необходимо уделять максимальное внимание, так как упущение важных особенностей функционирование систем может привести к некорректной обработке. В связи с чем этот важный этап разработки системы должен выполняться вручную с особой тщательностью.

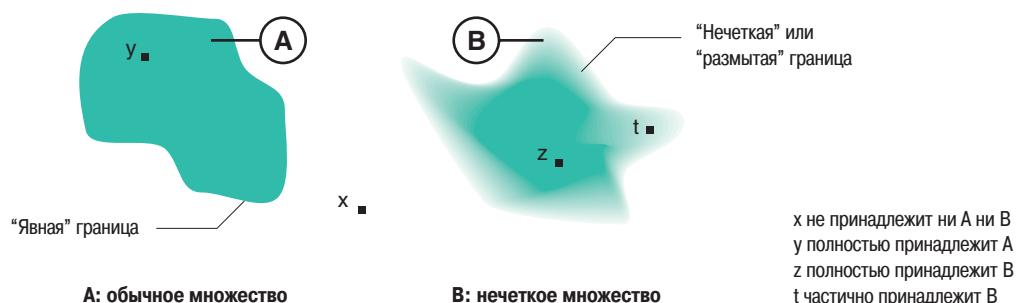
При наличии практического опыта можно описать этот опыт в виде правил нечеткой логики, когда система является сложной и/или нет готового опыта её создания или моделирования, в этом случае необходимо использовать глобальный подход к описаниям частей этой системы. Методы нечеткой логики не заменяют традиционные подходы к созданию систем управления, а наоборот, дополняют их.

## 2. Теория нечетких множеств

### 2.1. Понятие частичной принадлежности к множеству

В теории множеств элемент либо принадлежит множеству, либо нет. Понятия множеств используется во многих математических теориях. Это важное понятие, однако, не рассматривает простые ситуации, когда не все ясно и понятно. Например при выборе фруктов очень не сложно определить общий набор яблок. Однако, при этом непросто определить набор спелых яблок. Мы понимаем, что яблоки созревают постепенно ... понятие набора спелых яблок, таким образом, является не четким (расплывчатым).

Понятие нечеткого множества было принято для того, чтобы учесть представленную выше ситуацию. Теория нечетких множеств базируется на понятии частичной принадлежности к множеству: каждый элемент принадлежит к нечеткому множеству немного или частично. Очертание нечеткого множества (см. [рис. 1](#)) не имеет "Явной" границы, а представляется "нечетким" или "размытым".



**Рис. 1.** Сравнение обычного и нечеткого множеств

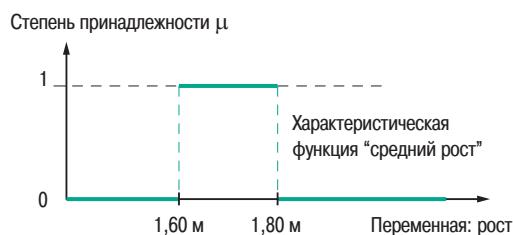
### 2.2. Функция принадлежности

Нечеткое множество определяется при помощи «функции принадлежности», которая соответствует понятию «характеристическая функция» в классической логике.

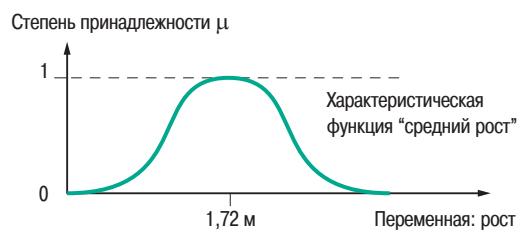
Предположим, мы хотим определить множество людей «среднего роста». В классической логике мы должны определить, что средним ростом мы считаем рост между 1,60 м и 1,80 м. Характеристическая функция в этом случае (см. [рис. 2](#)) дает «0» для всех, отличных от заданного диапазона ростов и «1» если рост соответствует

заданному. Нечеткое множество людей «среднего роста» задается при помощи «функции принадлежности», которая в этом случае может принимать значения в диапазоне [0;1].

Каждый возможный рост в этом случае будет представлен в виде «функции принадлежности» к нечеткому множеству людей «средний рост» (см. [рис. 3](#)), который принимает значения между 0 и 1.

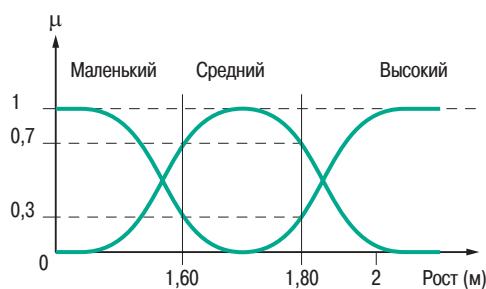


**Рис. 2.** Характеристическая функция



**Рис. 3.** Функция принадлежности

Несколько нечетких множеств можно определить через одну переменную, которая принимает нескольких значений, например множества "маленький рост", "средний рост" и "высокий рост". Каждое понятие описывает функция принадлежности (см. [рис. 4](#)).



**Рис. 4.** Функция принадлежности, переменная и лингвистическое описание

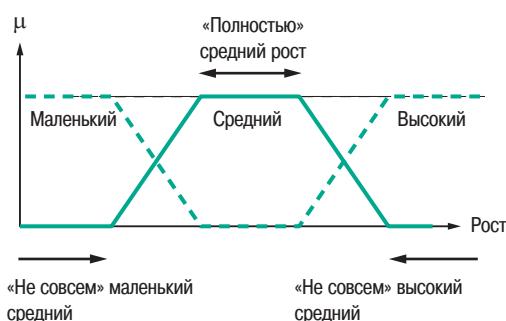
Этот пример показывает возможность применения нечеткой логики. Человека ростом 1,80 м можно отнести к группе "высокий рост" со степенью принадлежности 0,3 и к группе "средний рост" со степенью принадлежности 0,7. В классической логике переход от среднего роста к высокому мгновенен. Человек с ростом 1,80 м скорее всего принадлежит группе среднего роста, а человек ростом 1,81 м скорее всего высокий. Переменная (рост) и ее описания (средний, высокий), определенные функцией принадлежности, являются лингвистической переменной и лингвистическими описаниями соответственно.

Оба понятия лингвистическая переменная и лингвистическое описание могут быть напрямую использованы в правилах.

Функция принадлежности может принимать любую форму. Чаще для их представления используются кусочно-линейные линии (см. [рис. 5](#)).

Кусочно-линейные функции принадлежности обычно используется поскольку:

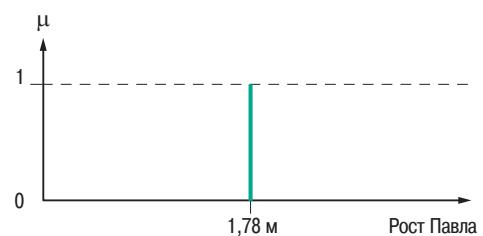
- они характеризуются простотой;
- они содержат точки, позволяющие задать области, где понятие является истинным, а где ложным, таким образом упрощает описание системы.



**Рис. 5.** Кусочно-линейная функция принадлежности

Далее в данном документе используются функции принадлежности этого типа.

Иногда функция принадлежности равна 1 для единственного значения переменной и 0 для всех других значений. В этом случае её называют функцией принадлежности одноэлементного множества. Ниже представлена (см. [рис. 6](#)) функция принадлежности одноэлементного множества переменной (рост), выраженной специфичным значением (рост Павла) данной переменной (подробная информация приведена в приложении).



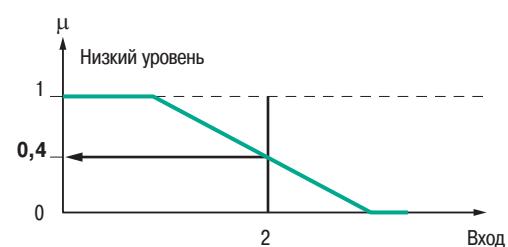
**Рис. 6.** Функция принадлежности одноэлементного множества

#### Подготовка задачи – определение степени принадлежности

Подготовка задачи для решения методами нечеткой логики (фаззификация) позволяет конвертировать реальные значения переменных в нечеткие.

Фаззификация заключается в определении степени принадлежности переменной (измерение, например) к нечеткому множеству. Например (см. [рис. 7](#)), если текущее значение «входной» переменной 2 то она принадлежит множеству "низкий уровень" со степенью принадлежности 0,4. Это и есть результат фаззификации.

Также можно сказать, что предположение "низкий уровень" является истинным при 0,4. При этом можно говорить о степени истинности предположения. Степень принадлежности и степень истинности, таким образом, являются схожими понятиями.



**Рис. 7.** Подготовка задачи - фаззификация

## 2.3. Операторы нечеткой логики

Операторы используются для записи комбинаций логических понятий нечеткой логикой, чтобы вычислять степени истинности. Применяются стандартные логические операторы AND, OR и NOT.

Пример: Подходящая квартира = Разумная цена аренды AND Достаточная площадь.

### Выбор операторов

Существует несколько вариантов этих операторов (см. приложение). Чаще всего используются операторы Заде, описанные далее.

Степень истинности предположения A записывается как  $\mu(A)$ .

### Пересечение

Логическим оператором пересечения множеств является оператор AND. Степенью истинности предположения "A AND B" является минимум из степеней истинности A и B:

$$\mu(A \text{ AND } B) = \text{MIN}(\mu(A), \mu(B))$$

Например:

"Низкая Температура" истинно на 0,7.

"Низкое Давление" истинно на 0,5.

Поэтому утверждение "Низкая Температура" AND "Низкое Давление" истинно на 0,5 = MIN(0,7; 0,5).

ВАЖНО: при использовании AND в нечеткой логике результат аналогичен классической логике: 0 и 1 получается 0.

### Объединение

Логическим оператором объединения множеств является оператор OR. Степенью истинности предположения "A OR B" является максимум из степеней истинности предположений A и B:

$$\mu(A \text{ OR } B) = \text{MAX}(\mu(A), \mu(B))$$

Например:

"Низкая Температура" истинно на 0,7.

"Низкое Давление" истинно на 0,5.

Поэтому утверждение "Низкая Температура" OR "Низкое Давление" истинно на 0,7.

ВАЖНО: при использовании OR в нечеткой логике результат аналогичен классической логике: 0 OR 1 получается 1.

### Отрицание

Логический оператор отрицания обозначается NOT.

$$\mu(\text{NOT } A) = 1 - \mu(A)$$

Например:

"Низкая Температура" истинно на 0,7.

Поэтому утверждение NOT "Низкая Температура" (можно записать как "Температура НЕ низкая") истинно на 0,3.

ВАЖНО: при использовании NOT в нечеткой логике результат аналогичен классической логике NOT(0) получается 1 и NOT(1) получается 0.

### Язык лестничных схем для нечеткой логики

Язык лестничных схем – это графический язык, который обычно используется инженерами систем управления для написания логических условий в классических описаниях алгоритмов. Компания Schneider предложила использовать этот же язык для описания выражений нечеткой логики.

Ниже приведен пример описания системы управления климатом:

горячий, влажный воздух не комфортен (формируется испарина); аналогично дыхание затруднено, если воздух холодный и очень сухой. Наиболее комфортная ситуация, когда воздух горячий и сухой или холодный и влажный. Эти формулы могут быть описаны на языке лестничных схем для нечеткой логики (рис. 8):

Комфорт = (Низкая температура AND Высокая влажность) OR (Высокая температура AND Низкая влажность).

Это логическое выражение описывает ощущение человека в помещении, в котором отсутствует принудительная циркуляция воздуха.

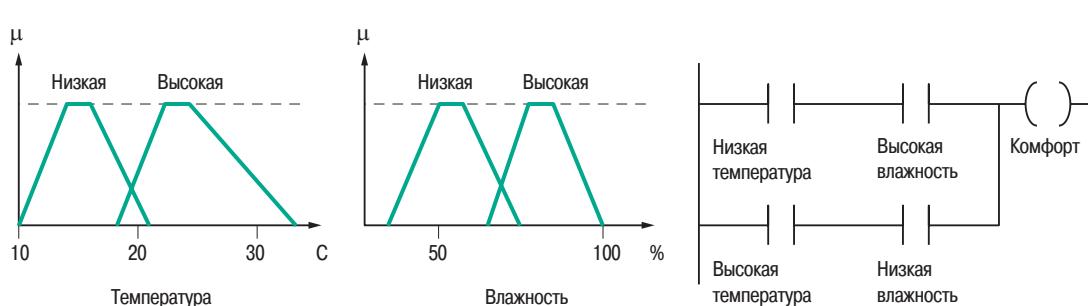


Рис. 8. Язык лестничных схем для нечеткой логики

## Классификация нечеткой логики

Классификация обычно состоит из двух частей:

- подготовка: определение рассматриваемых классов;
- во время работы: назначение элементов классам.

Понятия классов и множеств идентичны с обычной логикой.

Существует три типа методов назначения в соответствии с получаемым результатом:

- двоичный: элемент либо принадлежит, либо не принадлежит классу;
- вероятностный: элемент, вероятно, принадлежит двоичному классу, например, установка диагноза по симптомам, которые перечисляют пациент (диагностирование),
- градационный: элемент имеет степень принадлежности какому-либо классу, например, «салат» принадлежит к группе «свежая зелень».

Методы классификации, независимо от типа выдаваемого результата: двоичный, вероятностный или градуированный, могут быть реализованы через:

- эксперимент (в случае применения языка лестничных схем для нечеткой логики, описанной выше);
- примеры, используемые для учебных целей (например, классификаторы нейронных сетей);
- математических или физических знаний о проблеме (например, комфортность тепловой обстановки может быть получена из уравнений теплового баланса).

Градационный (или нечеткий) метод классификации может использоваться для управления петлями регулирования. Примером этого может служить промышленная печь для производства бисквитов, описание которой приводится далее.

## 2.4. Правила в нечеткой логике

### Нечеткая логика и искусственный интеллект

Целью базовых правил нечеткой логики является формализация и применение человеческого умозаключения. Таким образом, нечеткая логика является частью искусственного интеллекта. Базы правил нечеткой логики являются наиболее часто используемым инструментом в приложениях с нечеткой логикой. Базы правил нечеткой логики представляют собой набор правил, которые обычно используются параллельно, но в некоторых приложениях могут быть объединены.

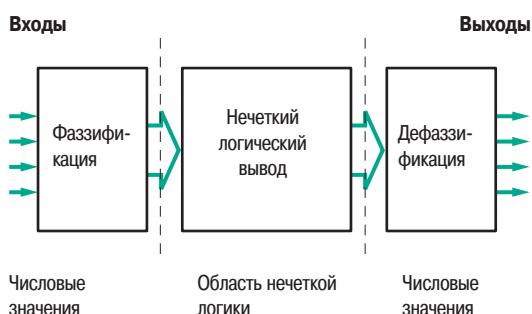
Применяются правила следующего типа:

IF “утверждение” THEN “результат.

Например: IF “высокая температура” AND “высокое давление” THEN “хорошая вентиляция” AND “открытый клапан”.

Базы правил нечеткой логики, подобно традиционным экспертным системам, основываются на базе знаний, построенной на основе человеческого опыта. В то же время существует существенные отличия в обработке и характеристиках этих знаний (см. [рис. 9](#)).

Процесс обработки нечеткой логики состоит из трех частей (см. [рис. 10](#)).



**Рис. 10.** Процесс обработки нечеткой логики

### Утверждение

Утверждение (также известное, как предположение или условие) является комбинацией предположений и операторов AND, OR, NOT.

“Высокая температура” и “Высокое давление” предположения из примера выше, соединенные оператором AND, образуют правила .

### База правил нечеткой логики

Немного правил

Последовательная обработка

Каскадирование возможно, но обычно не используется

Правила обрабатываются параллельно

Интерполяция между правилами, если они противоречат друг другу

### Обычные правила (экспертная система)

Много правил

Двоичная обработка

Правило каскадирования  $A \text{ OR } B \Rightarrow C$ ,  
 $C \Rightarrow D$ ,  
 $D \text{ AND } A \Rightarrow E$

Правила обрабатываются последовательно

Нет интерполяции, нет противоречий

**Рис. 9.** Нечеткая логика и традиционная

## Нечеткий логический вывод

Наиболее часто используется механизм нечеткого логического вывода, называемый механизмом Мамдани. Он представляет собой упрощение более общего механизма, который базируется на “нечетком выводе” и обобщенном правиле дедукции (*generalised modus ponens*). Информация о этих концепциях приведена в приложении. Далее используются только базы правил Мамдани.

## Результат

Результат нечеткого правила представляет собой комбинацию предложений объединенных операторами AND. В примере, представленном выше “хорошая вентиляция” и “открытый клапан” являются предложениями этого правила.

Инструкция “OR” не используется при формировании предложений результата, потому что она вносит неоднозначность в правило (при этом для выявления корректной обработки необходима дополнительная экспертиза). Обработка неоднозначностей не рассматривается механизмом логического вывода Мамдани, он обрабатывает только неточности. Поэтому правила нечеткой логики, основанные на механизме

Мамдани не подходят для применения в медицине, где результаты неоднозначны. Теория вероятностей, предложенная Лотфи Заде, описывает соответствующую методику для этих целей.

Отрицание также не применяется предложений результата в механизме Мамдани. Поскольку, если правило имеет результат THEN “вентиляция НЕ в среднем положении” это не означает, что можно сказать “слабая вентиляция” или “сильная вентиляция”. Это представляет собой еще один вариант неопределенности.

### **Механизм логического вывода Мамдани**

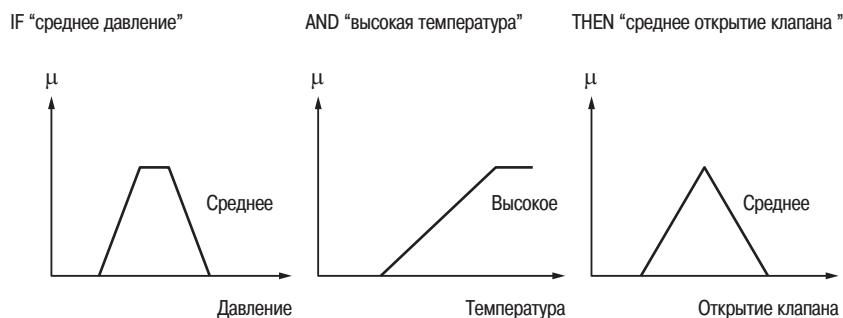
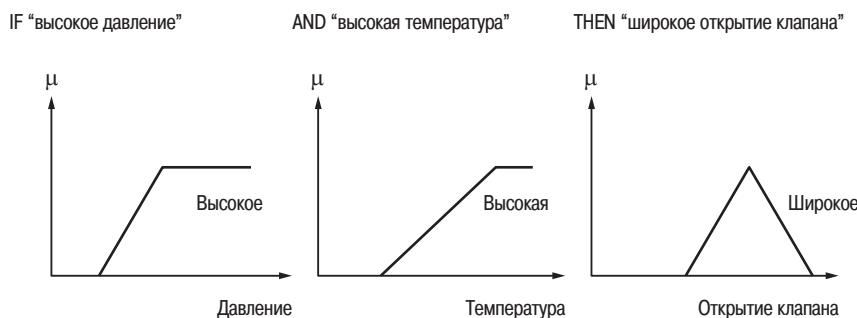
## ■ Основная идея

База нечетких правил Мамдани содержит лингвистические правила, использующие функции принадлежности для описания применяемых концепций (см. [рис. 11](#)).

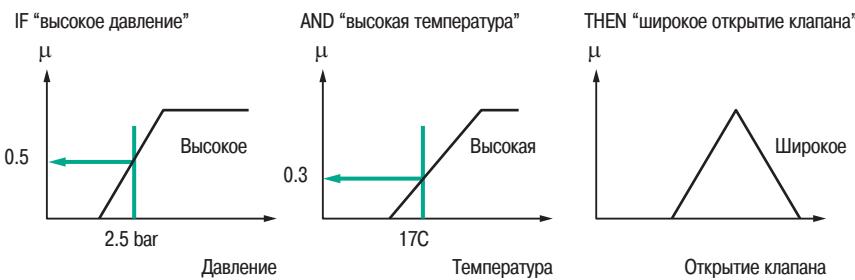
Механизм логического вывода состоит из следующих этапов:

## ■ Постановка задачи (фаззификация)

Процесс фазификации заключается в выведении функций принадлежности, используемых в утверждениях правил как показано на [рис. 12](#):



**Рис. 11.** Результат правила

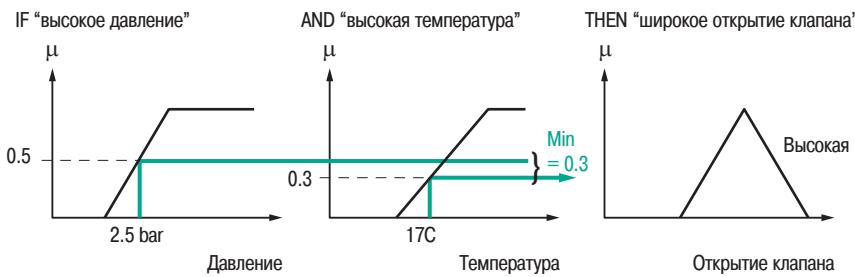


**Рис. 12.** Фазификация

#### ■ Степень активизации

Степень активизации правила вычисляется на основе оценки утверждений каждого правила, состоящих из логических комбинаций нескольких утверждений

(см. секцию 2.3.), как показано на **рис. 13**. Результатом операции “AND” является минимальное значение степеней истинности предложений.

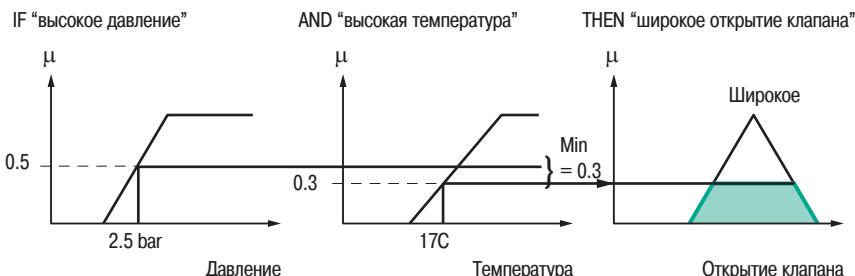


**Рис. 13.** Активизация

#### ■ Результат

Степень активизации правила используется для того, чтобы определить результат правила: данная операция называется выводом результата. Применяются несколько операторов вывода результата (см. приложение), чаще всего используется оператор выбора минимума.

Результирующее нечеткое множество строится с помощью поиска минимальных значений среди степеней активизации и функций принадлежности и сортировки «урезанных» функций принадлежности (см. **рис. 14**).

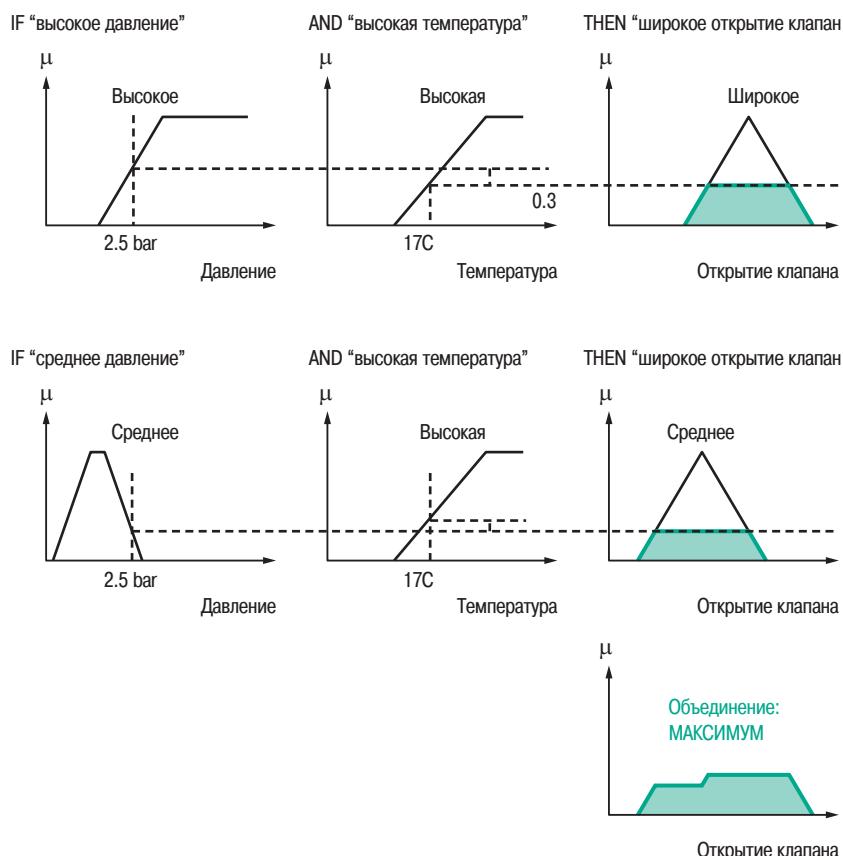


**Рис. 14.** Вывод результата

### ■ Объединение

Общее выходное нечеткое множество получается путем объединения нечетких множеств, полученных в результате применения каждого правила формирования для данного выхода. На примере ниже представлен случай, когда на формирование одного выхода влияют

два правила. Предполагается, что правила связаны логической операцией "OR", поэтому необходимо найти максимальное значение среди результирующих функций принадлежности для каждого правила (см. [рис. 15](#)).



**Рис. 15.** Объединение правил

### Дефазификация

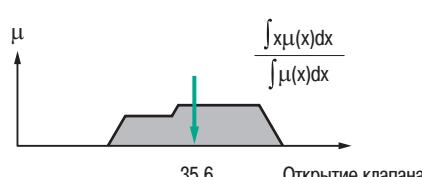
На последнем этапе нечеткого логического вывода, выходное нечеткое множество уже определено, но оно не может быть напрямую использовано для предоставления оператору точной информации или для управления исполнительным механизмом. Необходимо выполнить переход из "мира нечеткой логики" в "реальный мир": этот этап называется **дефазификация**.

Можно использовать разнообразные методы дефазификации, однако чаще всего используется метод вычисления "центра тяжести" нечеткого множества (см. [рис. 16](#)).

### "Свободные" правила и правила "возможности"

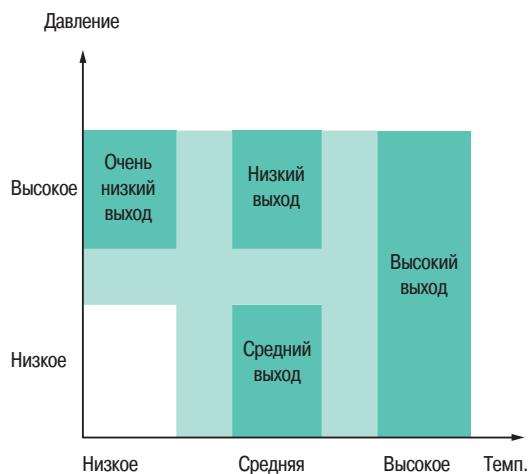
В базах нечетких правил, в общем случае, используются функции принадлежности для системных переменных и правила, которые могут быть записаны в текстовом виде. В каждом правиле используется свой вход и выход, как показано в примере ниже:

- R1: IF "Высокая температура" THEN "Высокий выход"
- R2: IF "Средняя температура" AND "Низкое давление" THEN "Средний выход"
- R3: IF "Средняя температура" AND "Высокое давление" THEN "Низкий выход"
- R4: IF "Низкая Температура" AND "Высокое давление" THEN "Очень низкий выход"



**Рис. 16.** «Центр тяжести» нечеткого множества

“Зоны действия” правил и их наложения могут быть представлены в виде графической диаграммы, показанной на [рис. 17](#).



**Рис. 17.** Результаты применения правил

Можно сделать следующие наблюдения:

- нет необходимости покрывать все области: в данном случае комбинация «Низкая температура» и «Низкое давление» не рассматривается. Это можно объяснить, например, тем, что данная комбинация физически невозможна для данного механизма, или тем, что она не представляет интереса. Такая графическая диаграмма

является лучшим способом проверки на наличие возможных ошибок или упущений;

- первое правило учитывает только параметр «Температура»: это нормально, потому что отражает существующий опыт.

Однако, для многих приложений определяют «таблицы» правил. При таком подходе, все пространство разбито на ячейки и каждой ячейке назначено правило. Преимуществом такого подхода является его методичность, однако:

- он не всегда позволяет использовать простое выражение существующего накопленного опыта (с минимальным количеством правил),
- он может применяться только для двух или трех входов, тогда как базы “свободных” правил могут быть построены на основе большого количества переменных.

### Замечания

■ Поведение базы нечетких правил зависит от входов и является статичным и нелинейным.

■ Базы нечетких правил сами по себе не являются динамическими, хотя они часто используют, как входы, переменные, отображающие динамику системы (производные, интегралы и т.д.) или времени.

■ Основным преимуществом контроллера с функцией нечеткого ПИД-регулирования, часто представляемого как учебный пример для объяснения идеи нечеткой логики, является выполнение нелинейного ПИД-регулирования, которое может быть использовано вместо традиционного ПИД-регулирования. Более того, при традиционном подходе трудно учсть весь накопленный опыт для конкретного применения.

### 3. Пример учебного приложения

#### 3.1. Предисловие

Для понимания большинства достижений в области применения нечеткой логики требуется некоторые предварительные специальные знания в прикладной области.

Для облегчения восприятия следующий пример основан на придуманном приложении и предназначен для пояснения процедуры создания базы нечетких правил.

#### 3.2. Представление учебного приложения

В примере рассматривается процесс мытья салата при производстве расфасованного салата для отделов свежих овощей в супермаркетах. Салат необходимо отрезать, вымыть и упаковать. Целью мытья является удаление земли с салата, а также различных микроорганизмов, которые могут размножаться в течение срока хранения продукта. Производителю нужно автоматизировать процесс мытья.

Процесс мытья является непрерывным. Листья салата помещаются в барабан, который перемещается внутри тоннеля, оснащенного соплами, распыляющими воду и хлорин. Вода смывает землю, а хлорин убивает микроорганизмы (см. [рис. 18](#)).

Отделом маркетинга сформулированы принципы работы, приведенные ниже в порядке их важности:

- Уважение к покупателям:
  - гарантия качества:
    - очень чистый салат (внешне);
    - без вкуса и запаха хлора;
  - гарантия безопасности:
    - допустимое количество микроорганизмов.
- Повышение прибыльности:
  - повышение объемов производства;
  - экономия воды;
  - экономия хлорина.

Операторы вручную контролируют процесс мытья, проверяя воду после мытья в конце тоннеля. Если вода чистая, то основываясь на практическом опыте можно прийти к выводу, что салат внешне чистый. Техническим решением такой задачи является установка оптического датчика, измеряющего прозрачность воды.

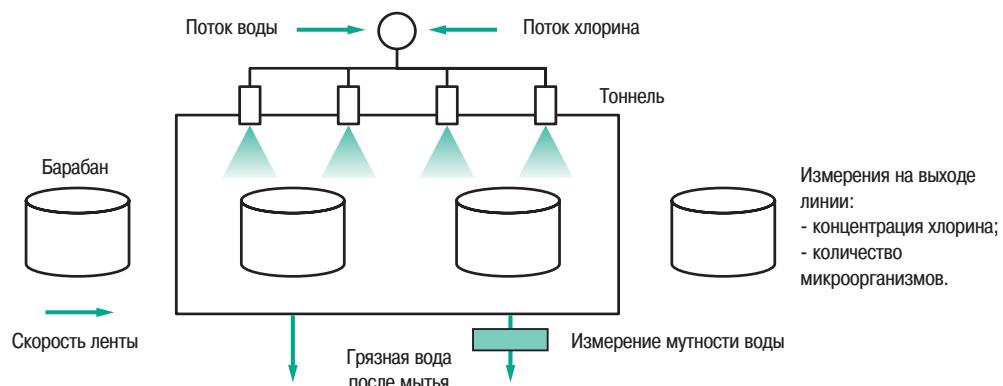
Кроме того, операторы должны подготовить отчет, в котором приводится количество микроорганизмов и концентрацию остаточного хлорина на вымытом салате на выходе линии.

Также необходимо контролировать:

- скорость движения ленты с салатом (для повышения производительности линии на выходе);
- количество распыляемого хлорина;
- количество распыляемой воды.

Накладываемые ограничения:

- механическая скорость ленты;
- поток воды, не вызывающий повреждения листьев салата.



*Рис. 18. Процесс мытья салата*

### 3.3. Лингвистические переменные и термы

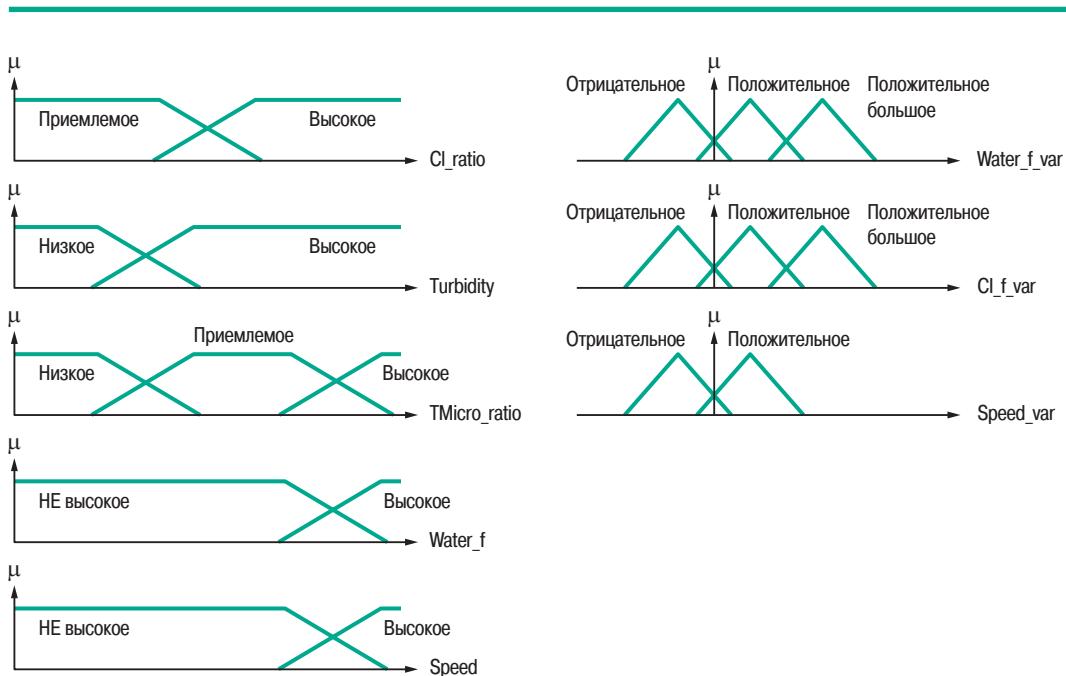
Используются следующие переменные:

- Входы:
  - количество микроорганизмов: Micro\_ratio;
  - остаточная концентрация хлорина: Cl\_ratio;
  - прозрачность воды: Turbidity;
  - скорость ленты конвейера: Speed;
  - поток воды: Water\_f.

■ Выходы:

- изменение потока воды: Water\_f\_var;
- изменение потока хлорина: Cl\_f\_var;
- изменение скорости: Speed\_var.

В результате совместного обсуждения с опытными операторами, специалистами по микробиологии и контроллерами качества салата были выведены следующие функции принадлежности (см. [рис. 19](#)):



**Рис. 19.** Кусочно-линейная функции принадлежности

### 3.4. Правила и выходы

#### Запись правил нечеткой логики

В процессе консультации с операторами были выведены следующие семь правил:

- Салат вымыт плохо:
 

IF Turbidity = Высокое AND Water\_f = НЕ Высокое THEN Water\_f\_var = Положительное большое.
- Салат вымыт плохо, но скорость движения ленты большая:
 

IF Turbidity = Высокое AND Water\_f = Высокое THEN Speed\_var = Отрицательное.
- Большое количество микроорганизмов:
 

IF Micro\_ratio = Высокое THEN Cl\_f\_var = Положительное большое.
- Выходные измерения в норме и производительность может быть увеличена:
 

IF Turbidity = Низкое AND Micro\_ratio = НЕ Высокое AND Speed = Высокое AND Cl\_ratio = Приемлемое AND Water\_f = Высокое THEN Speed\_var = Положительное

AND Cl\_f\_var = Положительное AND Water\_f\_var = Положительное.

- Салат имеет запах и привкус хлорина, и при этом очень малое количество микроорганизмов:
 

IF Cl\_ratio = Высокое AND Micro\_ratio = Низкое THEN Cl\_f\_var = Отрицательное.

■ Выходные измерения в норме и производительность максимальная: экономия воды.

- IF Speed = Высокое AND Cl\_ratio = Приемлемое AND Turbidity = Низкая THEN Water\_f\_var = Отрицательное.

- Очень малое количество микроорганизмов:
 

IF Micro\_ratio = Низкое THEN Cl\_f\_var = Отрицательное.

#### Дефазификация

Обратное преобразование (дефазификация) выполняется по методу определения «центра тяжести» выходного нечеткого множества.

## 4. Применение

### 4.1. Области применения баз нечетких правил

Базы нечетких правил могут использоваться для решения задач приложения, если соблюдаются следующие условия:

- сохраняется возможность воздействия на процесс (управляемость процесса);
- существует накопленный опыт и проверенные знания;
- переменные (входы и выходы) можно измерять и наблюдать (измеримость);

- качественная экспертная оценка (если она математическая, то необходимо выбрать традиционное автоматическое управление);
- последовательная экспертная оценка (если она дискретная, то больше подходит использование экспертных систем).

### 4.2. Разработка приложения

#### Выбор операторов

В большинстве приложений используются базы нечетких правил Мамдани. Данный механизм подходит, если в описании отсутствуют неопределенности.

Чаще используются трапециевидные функции принадлежности, поскольку они просты в применении и упрощают построение базы знаний. Выходные функции принадлежности часто являются функциями принадлежности одноэлементного множества, кроме случаев, когда правила связаны друг с другом. Выходные функции принадлежности треугольной формы фактически подразумевают неопределенность значения, выдаваемого на выход и не оказывают существенного влияния на интерполяцию между правилами.

В заключении, дефазификация обычно выполняется по методу вычисления «центра тяжести» выходного множества (рассматриваются все активные правила): использование «среднего максимума» для проблем принятия решений позволяет принимать решение, когда правила «конфликтуют» между собой и среднего решения не получается.

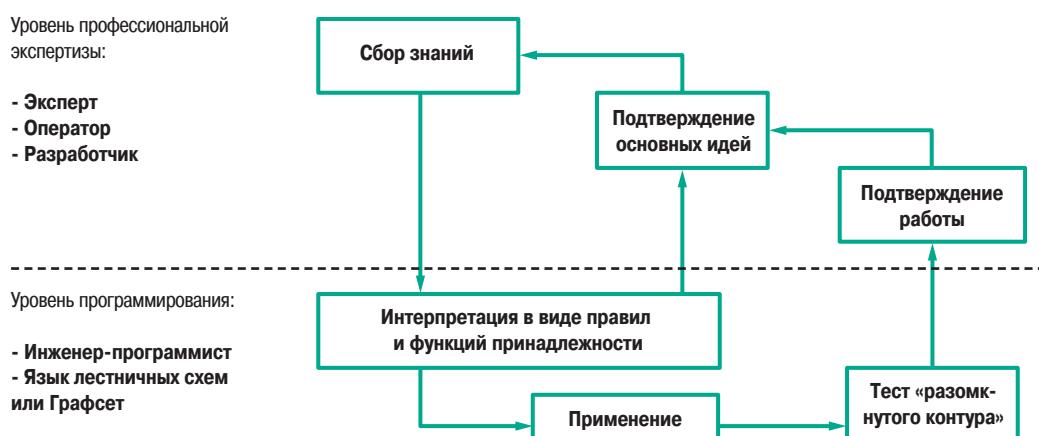
#### Методология

Разработка базы нечетких правил является интерактивным процессом. Большой частью это сбор знаний и опыта. Одним из преимуществ нечеткой логики является возможность построения базы правил, одобренной специалистами до проверки ее работы в реальных условиях (см. [рис. 20](#)).

#### Построение базы знаний

Этот процесс состоит из трех этапов:

- составление списка переменных, которые должны быть учтены: они станут лингвистическими переменными в базе нечетких правил;
- составление перечня качественных оценок, которые применяются к рассмотрению и определяют, когда они являются истинными, а когда ложными: эти оценки станут лингвистическими терминами базы правил;
- составление формулировок правил: они должны описывать поведение в каждом конкретном случае.



**Рис. 20.** Методология разработки

Однако необходимо формулировать как можно меньшее количество функций принадлежности и правил, с целью ограничения числа параметров, которые необходимо будет впоследствии настраивать и обеспечить четкость базы правил. Обращаем ваше внимание, что проще добавлять правила, чтобы принять во внимание новые ситуации, чем удалять их.

#### Обоснование базы знаний

Этот процесс состоит из нескольких этапов:

- представление и обсуждение базы правил с экспертами, которые помогут собрать накопленные знания в данной области. Целью обсуждения является выявление упущеных моментов и обеспечение ясности правил;
- симуляция «открытого контура»: эксперты сравнивают поведение базы правил с ожидаемым поведением;
- если процесс можно симулировать, то выполняется и симуляция «замкнутого контура».

#### Настройка

Базы правил, написанные с соблюдением вышеизложенных принципов часто сразу дают удовлетворительный результат. Однако может потребоваться модификация или настройка базы правил. При поиске возможных причин отклонений от нормы необходимо учитывать следующие принципы:

- если поведение контроллера замкнутого контура противоположно ожидаемому, то вероятнее всего некоторые правила некорректно написаны;
- если поведение необходимо оптимизировать производительность, обычно достаточно соответствующим образом настроить функции принадлежности;
- если система неустойчива и временами работает, а временами нет, то вероятнее всего не все случаи были рассмотрены и необходимо добавить новые правила.

### 4.3. Использование приложения

#### Функции оператора

Степень участия оператора в управлении системой на базе нечеткой логики значительно варьируется в зависимости от приложения.

Необходимо учитывать следующие особенности:

- полностью автономная система: конечный пользователь незнаком с нечеткой логикой и не осведомлен о ее использовании;
- блок управления с нечеткой логикой выполнен в виде «черного ящика» и может быть отсоединен или переведен в режим ручного управления оператором;
- оператор может модифицировать (настроить) функции принадлежности в зависимости от ситуации или может изменять производительность (например);
- оператор может читать правила (например, степень активизации): он хорошо в них разбирается и может интерпретировать действия базы правил. Например, он может контролировать базу правил в исключительных ситуациях;
- оператор является главным разработчиком базы: он позволяет записать его собственное ноу-хау и подтверждает получившееся в результате поведение системы.

#### Изменение производительности

В процессе эксплуатации необходимо, чтобы была возможность настраивать базу правил с целью изменения производительности системы или изменения типа выпускаемого продукта. Изменения могут быть следующими:

- изменяется задание (рабочая температура и т.д.), например в связи с изменением выпускаемого продукта. При этом необходимо изменить уставки или правила входных функций принадлежности;
  - изменяются размеры системы; необходимо изменить функции принадлежности;
  - изменяется тип системы (например, перемещение базы правил с одного механизма на другой); необходимо изменить правила и функции принадлежности.
- Чаще всего встречаются изменения первого типа. Они могут быть выполнены квалифицированными операторами.

### 4.4. Выбор применяемой технологии

Большинство современных приложений работают на стандартных аппаратных платформах (микроконтроллеры, ПЛК, микропроцессоры, микрокомпьютеры, и т.д.).

Разработано разнообразное ПО, предназначенное для построения баз нечетких правил для микроконтроллеров, ПЛК и микрокомпьютеров и позволяющее быстро создавать базы правил.

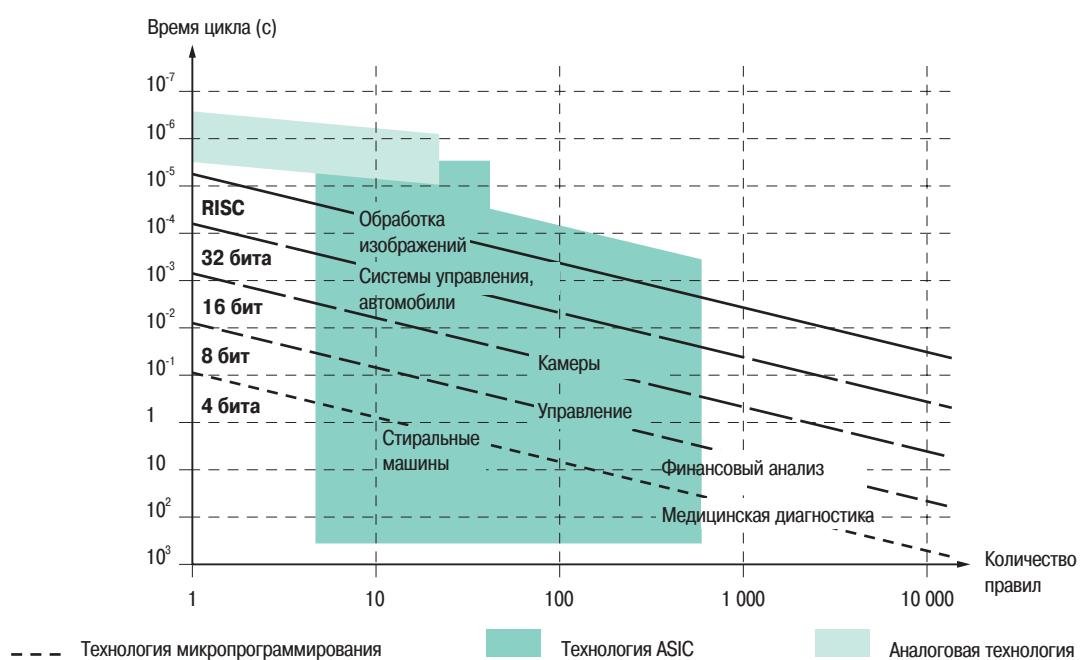
Нечеткий логический вывод может быть напрямую запрограммирован (на языке ассемблера, С и т.д.). Недостатками такого решения являются длительная фаза создания первого образца и наличие навыков программирования и управления алгоритмами нечеткой логики.

Для приложений с точно определенным временем ответа или для того, чтобы снизить себестоимость производства продукции, использование микросхем нечеткой логики является предпочтительным. Использование таких электронных схем возрастает, поскольку:

- операции, которые применяются для обработки нечеткого логического вывода, являются элементарными и выполняются над целыми числами;
- некоторые операции могут выполняться параллельно;
- вычисления выполняются последовательными шагами, таким образом может быть реализована простая архитектура типа «трубопровод».

В частности, большое количество компонентов ASIC разработаны для специфичных рынков (автомобили, электронных бытовых устройств и т.д.).

В настоящее время они часто встраиваются в микроконтроллеры, даже недорогие, где они используются для ускорения обработки нечеткой логики. На рисунке 21 представлен как пример, график зависимости требований приложений, выраженный в количестве правил (сложность приложения) и времени цикла (скорость) насколько позволяет технология. Предположим, что правила имеют одно утверждение и один результат. Необходимое технико-экономическое обоснование заключается в поиске компромисса между гибкостью, которую обеспечивает решение на основе программирования, экономичностью и производительностью готовых аппаратных решений.



**Рис. 21.** Производительность компонентов и области применения

## 4.5. Стандарты

### Компоненты

Отсутствие стандартов является одной из причин, задерживающих развитие микросхем нечеткой логики. Эти компоненты не совместимы друг с другом, поскольку каждый из них является результатом разработки различных производителей.

### Программное обеспечение

В зависимости от ПО, недостаточная мобильность также тормозит распространение использования нечеткой логики в промышленности.

В настоящее время рабочая группа, к которой компания Schneider принимает активное участие, разрабатывает совместный стандарт языка «Нечеткая логика» для программируемых контроллеров (первая официальная редакция стандарта МЭК 61131-7 вышла в 1997). Другие инициативы в области стандартизации нечеткой логики должны также опираться на данный стандарт.

## 5. Приложения нечеткой логики

### 5.1. Типы приложений

#### Выполняемые функции

В таблице ниже представлены функции, выполняемые в промышленности средствами систем с нечеткой логикой (X обозначает возможное применение, XX обозначает, что техника хорошо подходит для решения такого типа задач). Использование баз правил предпочтительнее использовать в тех случаях, когда необходимо выполнять интерполяцию или действия, тогда как методы классификации более подходят для задач оценки и диагностирования. Однако в приложениях иногда требуется реализация нескольких функций и одновременная поддержка градуированности информации.

	Базы правил	Алгоритмы классификации
Регулирование, управление	XX	
Автоматическая настройка параметров	XX	
Помощь в принятии решений	XX	X
Диагностирование	X	XX
Контроль качества		XX

#### Нечеткая логика и другие технологии

Нечеткая логика является развитием и обобщением булевой логики. Она позволяет представить градуированность в виде понятий, которые ранее были либо истинными, либо ложными.

Вероятности, кроме возникновения событий двоичного характера (истинно либо ложно) позволяют управлять неопределенностью этих событий.

На границе между этими подходами находится теория вероятностей (изобретена Лотфи Заде), которая учитывает и градуированность, и неопределенность (см. рис.22).

Для применений автоматизированного управления / регулирования, базы нечетких правил часто сравнивают с нейронными сетями и традиционным автоматизированным управлением. Три указанных выше подхода требуют соответственно: наличия практического опыта, данных для проведения анализа и динамической модели процесса.

Данные подходы можно сравнивать только тогда, когда возможно использование всех трех подходов, что часто бывает в теоретических исследованиях, но редко на практике. Если реализация всех подходов возможна, то приоритетное значение приобретают практические аспекты.



Рис. 22. Сравнение нечеткой логики с другими технологиями управления.

В частности, нечеткая логика имеет преимущество в простоте ее освоения операторами.

#### Гибридизация технологий

Нечеткая логика часто используется совместно с другими технологиями. Такие комбинации являются предпочтительными, когда используются сильные стороны каждого подхода.

■ «Обучающиеся» правила нечеткой логики или нечеткие нейронные сети

Базы нечетких правил могут быть модифицированы с помощью методов обучения.

Первые методы, известные как «самоорганизующийся контроллер», получили развитие в 1974 и были направлены на эвристическое изменение сущности правил нечеткой логики, связанных с «таблицами правил». Фактический опыт накапливается в процессе обучения, но функции принадлежности остаются прежними.

Второй подход состоит в изменении типичных параметров функций принадлежности. В отличии от первого метода, правила и структуры не изменяются. С помощью правил оптимизации изменяются параметры функций принадлежности, например, используются методы градации или общие методы оптимизации, такие как генетические алгоритмы или моделируемая нормализация.

Данный подход часто называют «нейронной нечеткой логикой», в частности когда используется плавный переход - градиент. Использование градиента для оптимизации параметров подобно «обратному распространению», используемому в нейронных сетях, также называемому «многоуровневыми персепtronами (распознающими элементами)» и применяемому для оптимизации масс между уровнями в нейронных сетях.

Третий подход (который может быть квалифицирован как структурная оптимизация базы правил) направлен на одновременное определение правил и функций принадлежности посредством обучения. Тогда обычно имеет место процесс обучения без ссылки на практический опыт. В результате получаются правила, которые теоретически могут быть использованы в процессе накопления практического опыта.

#### ■ Использование нечеткой логики совместно с автоматическим управлением

Иногда база нечетких правил является частью контроллера. Использование нечеткой логики для симулирования пропорциональных термов позволяет описать все типы нелинейностей. Специфические случаи неустойчивой работы, такие как перегрузки, техническое обслуживание или частичная неисправность можно легко интегрировать.

База нечетких правил может использоваться вне контура управления, для наблюдения за контроллером. В этом случае, для того, чтобы настроить параметры контроллера в соответствии с условиями работы системы управления, она заменяет оператора.

## 5.2. Примеры достижений в промышленности

Сегодня нечеткая логика распространена в качестве одного из методов, которые обычно используются для управления производственными процессами. Хотя ПИД-контроллеры удовлетворяют требованиям большинства приложений, нечеткая логика используется все чаще из-за ее преимуществ в адаптации, особенно для контроля качества продукции и затрат. Из-за конкуренции интеграторы и конечные пользователи обычно подробно не рассказывают о преимуществах, которыми обладает нечеткая логика для некоторых приложений. Эти приложения основываются на всестороннем сборе секретов производства и применения новых технических решений. Поэтому особенно важной становится конфиденциальность. Этим объясняется и причина, по которой невозможно детально описать все примеры, приведенные ниже.

### Очистные сооружения

Самые современные очистные сооружения используют биологические процессы (культивирование бактерий в вентилируемых баках) для очистки сточных вод перед сбросом их в окружающую среду. Органические вещества, содержащиеся в сточных водах, используются бактериями для создания их клеточных компонентов. Бактерии выделяют углекислый газ ( $\text{CO}_2$ ) и азот ( $\text{N}_2$ ). В баки подается воздух. Энергия, расходуемая на вентиляцию часто составляет более половины общей энергии, потребляемой заводом. Для того, чтобы обеспечить оптимальное развитие бактерий, необходимо тщательно контролировать концентрацию  $\text{NH}_4$  и  $\text{O}_2$  в вентилируемых баках, более того, чтобы снизить затраты на электроэнергию, необходимо поддерживать минимально допустимый биологическим процессом поток воздуха.

Дополнительно к данным требованиям необходимо учитывать некоторые специфические рабочие моменты, например, очень высокий восходящий поток, который является экстремальным обстоятельством, при котором происходит существенное изменение параметров, влияющее на емкость баков сточных вод.

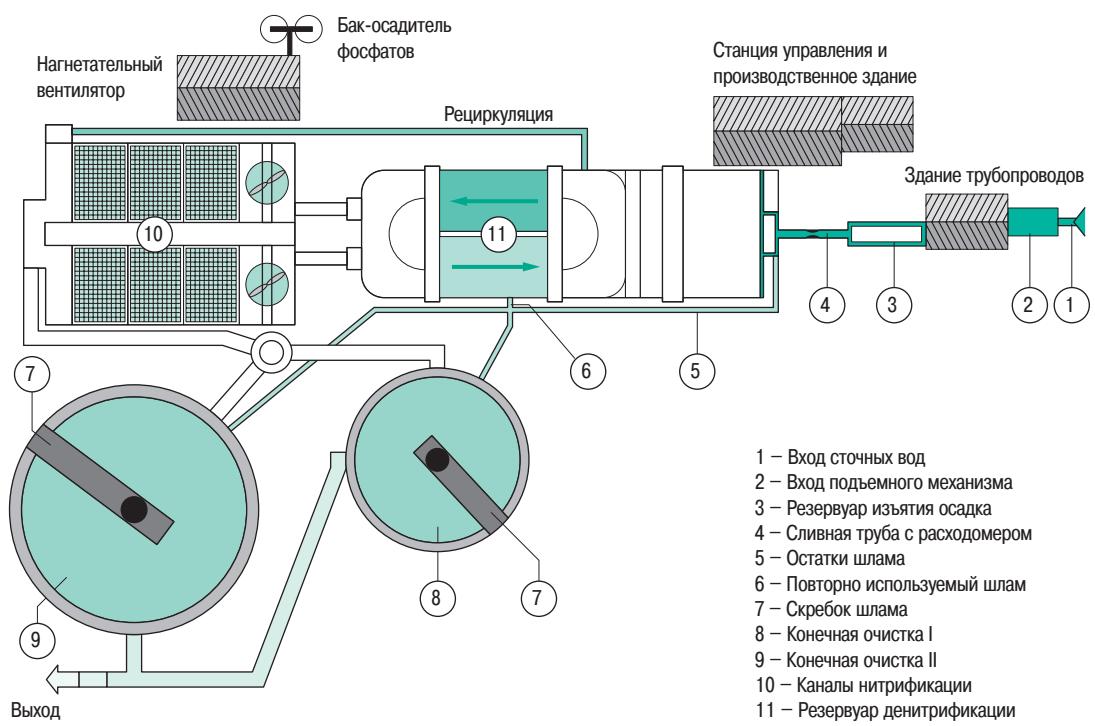
Хотя возможно построить частичные математические модели производственных процессов, но полные модели отсутствуют и общая стратегия управления часто разрабатывается эвристически.

В настоящее время нечеткая логика часто используется на очистных сооружениях. Схема производственного процесса, представленная на [рис. 23](#), применялась в Германии и находится в работе с 1994. Обработка нечеткой логики выполняется программируемыми контроллерами Schneider Modicon посредством их стандартных функциональных модулей нечеткого управления.

Разработчики подчеркивают преимущества использования нечеткой логики для управления: исключения, то есть ситуации, когда емкость баков частично снижается, обрабатываются просто и без нарушения последовательности работ.

Ниже представлен метод, описывающий данные исключительные ситуации в контуре управления:

В контуре управления описывается пропорциональный терм, который должен адаптировать исключительные ситуации: этот терм сначала записывается посредством нечеткой логики, затем данный блок нечеткой логики встраивается в контур управления.



**Рис. 23.** Схема очистного сооружения

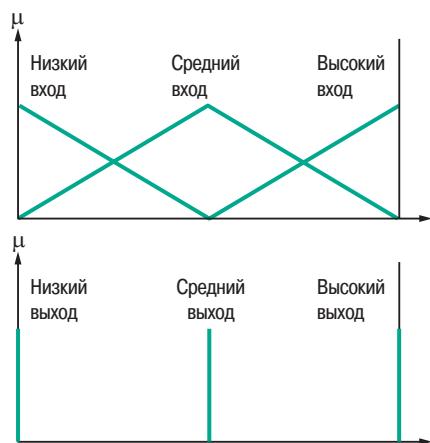
После того, как функции принадлежности должны образом настроены, необходимо описать два правила, достаточных для пропорционального контроллера:

IF Низкий вход THEN Низкий выход  
IF Высокий вход THEN Высокий выход

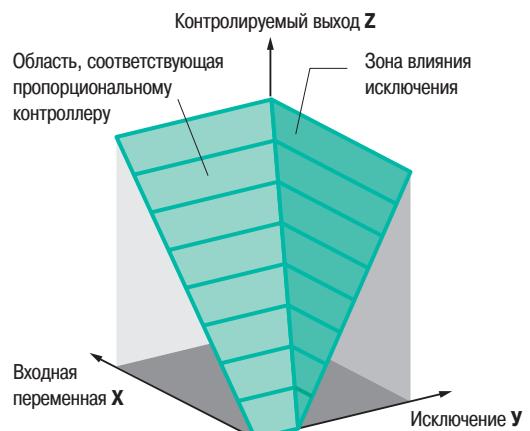
Третье правило добавляется по запросу операторов:  
IF Средний вход THEN Средний выход (см. [рис. 24](#)).

Как только описан пропорциональный терм, исключения введены в алгоритм в виде правил, другие правила формулируются в зависимости от комбинаций других входных переменных.

Простой пример такой возможности представлен на [рис. 25](#).



**Рис. 24.** Симулирование пропорционального терма контроллера



**Рис. 25.** Внесение исключения в терм

В таблице на [рис. 26](#) перечислены правила для рециркуляции. Пропорциональный терм создается на основе входной переменной "Содержание NO<sub>x</sub>". Две

входных переменных "Содержание O<sub>2</sub> при нитрировании" и "Содержание O<sub>2</sub> при денитрировании" определяют исключительную ситуацию в первом правиле.

IF Содержание O <sub>2</sub> при нитрировании	AND Содержание O <sub>2</sub> при денитрировании	AND Содержание NO <sub>x</sub>	THEN Количество рециркуляций
Не низкое	Больше 0		Низкое
		Низкое	Низкое
		Нормальное	Нормальное
		Высокое	Высокое

[Рис. 26.](#) Таблица правил функции рециркуляции

Ниже приведено описание другой обработки с помощью нечеткой логики: часть шлама, осаждаемого в нижнем резервуаре возвращается в контур рециркуляции. В таблице на [рис. 27](#) перечислены правила рециркуляции шлама. Первое правило описывает исключения, вызванные очень высоким восходящим потоком. При таких условиях высокая степень рециркуляции может привести к перегрузке установки.

Обнаружение исключительных условий производится по сильной замутненности, поскольку из-за чрезвычайно высокого потока откладывается минимальный осадок шлама.

К сведению, другие функции установки также используют нечеткую логику:

- выпуск воздуха,
- управление извлечением шлама.

IF Замутненность сбрасываемых вод	AND Количество осадка рециркулируемого шлама	AND Уровень	THEN Количество рециркуляций шлама
Высокое		Низкое	Низкое
	Нормальное	Низкое	Низкое
	Высокое	Низкое	Нормальное
	Низкое	Нормальное	Высокое
	Нормальное	Нормальное	Нормальное
	Высокое	Нормальное	Высокое
	Низкое	Высокое	Нормальное
	Нормальное	Высокое	Высокое

[Рис. 27.](#) Таблица правил функции рециркуляции шлама

### Пищевое производство

Автоматизация промышленных печей производственных линий, которые используются для приготовления бисквитов, представляет интерес для производителей бисквитов во Франции и Германии. Для таких систем управления традиционный подход не всегда приемлем из-за нелинейностей, разнообразия и разнородности параметров. Моделирование процесса приготовления является сложным и неполным. Однако, опытные операторы превосходно управляют процессом приготовления, благодаря своему большому практическому опыту.

В примере ниже представлено описание производственной линии приготовления бисквитов.

Группа французских компаний обратилась к компании Schneider, которая совместно с институтом ENSIA (Высший Институт сельского хозяйства и пищевой промышленности Франции) выработали решение для автоматизации данного производственного процесса.

При приготовлении бисквитов основными их характеристиками, которые можно измерить, являются цвет, влажность и размеры. На значения данных характеристик влияют изменения количества ингредиентов теста, условия окружающей среды, время нахождения бисквитов в печи и т.д.

Эти влияния должны быть компенсированы настройками печи и скоростью движения ленты конвейера. Контроль качества продукции для данного производственного процесса можно разбить на несколько функциональных этапов:

- создание условий и объединение данных;
- оценка субъективных количеств (связано с качеством);
- обнаружение отклонений качества;
- принятие решения;
- субъективная оценка.

Нечеткая логика позволяет учитывать качественные переменные и проводить «профессиональную» экспертизу. Базы нечетких правил используются совместно с другими технологиями (см. [рис. 28](#)).

Функции	Применяемая технология
Объединение датчиков	
Субъективная оценка	Нечеткая классификация
Обнаружение отклонений качества	Язык лестничных схем нечеткой логики
Принятие решения	Базы нечетких правил

**Рис. 28.** Функции и применяемые технологии

#### ■ Субъективная оценка

Большинство понятий определения качества опирается на несколько переменных. Одним из факторов оценки качества продукции является цвет (по трем измерениям), поэтому функции принадлежности описываются по нескольким переменным. Алгоритмы классификации, основываясь на входных переменных, выполняют градуированную оценку таких качественных переменных (например: верх бисквита хорошо пропечен, перепечено и т.д.).

#### ■ Обнаружение отклонений качества

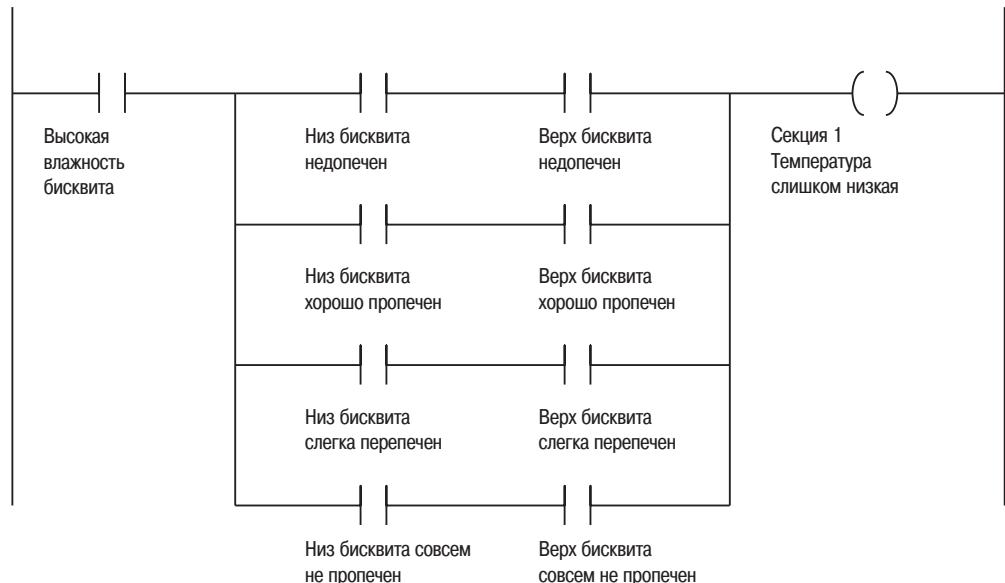
Язык лестничных схем нечеткой логики используется для обнаружения отклонений качества выпекаемых бисквитов (см. [рис. 29](#)). Печь имеет 3 секции.

Общая оценка работы является удовлетворительной.

### Другие примеры

#### ■ Системы автоматизации

Контроллеры GPC (Global Predictive Controller – контроллеры общего управления с прогнозированием) являются очень эффективными, однако требуется произвести настройку 4 параметров: N1, N2, Nu, I (2 диапазона управления, диапазон прогнозирования, весовой коэффициент). Такая настройка очень сложна и обычно требует наличия практического опыта. Дочерняя компания NUM компании Schneider в настоящее время занимается разработкой новых цифровых технологий управления и используют для будущих изделий технологии GPC.



**Рис. 29.** Применение языка лестничных схем нечеткой логики для обнаружения отклонений качества

Компания Schneider разработала для блоков NUM метод автоматической настройки параметров с помощью баз нечетких правил для таких контроллеров. Около двадцати правил применяются для быстрой и достоверной настройки параметров. Кроме того присутствие специалиста по наблюдению и управлению более не требуется, что редко случается в цифровых автоматизированных системах.

#### ■ Автомобильная промышленность

Автомобильные компании Renault и Peugeot (PSA) анонсировали выпуск автоматической коробки передач, в которой применяется нечеткая логика для адаптации к стилю вождения человека, находящегося за рулем.

#### ■ Цементные заводы

Первое промышленное применение нечеткой логики, которое затем было скопировано другими

производителями, было реализовано датской компанией F.L. Smidth Automation и предназначено для управления сушильной печью при производстве цемента. В данном процессе должно быть учтено большое количество переменных и, в частности, климатические воздействие на сушильную печь высотой несколько десятков метров.

#### ■ Электронная и электрическая бытовая техника общедоступного применения

Большое количество приложений доступно широкой публике, особенно в Японии. Например, компактные портативные цифровые видеокамеры, обладающие повышенной чувствительностью к перемещениям. В данных устройствах нечеткая логика управляет системой стабилизации изображения.

## 6. Заключение

- Классифицируемая как техника с искусственным интеллектом, нечеткая логика используется для моделирования и замещения опыта управления процессом и опыта разработчиков и операторов.
- Как средство повышения качества и производительности, нечеткая логика предлагает преимущества повышения конкурентоспособности производственным компаниям в поисках технико-экономической оптимизации.
- В данном документе определены области, в которых данный подход может быть использован наиболее эффективно.
- Благодаря удобному пользовательскому интерфейсу среды программирования ПЛК, нечеткая логика доступна всем инженерам, занимающимся автоматическим управлением, которые хотят повысить уровень своих навыков и знаний, а также эксплуатационные

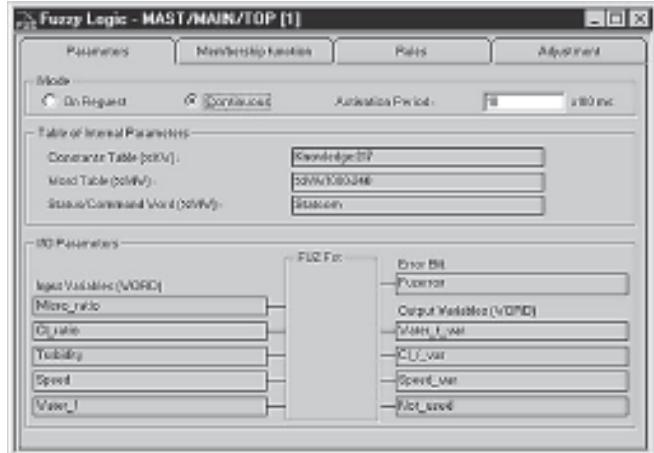
характеристики своих разработок. Такой инструментарий имеется в наличии в среде программирования для некоторых контроллеров (см. [рис. 30](#)).

■ Конкуренция с другими традиционными средствами управления (например, ПИД-контроллеры) вызывает развитие в большинстве областей применения.

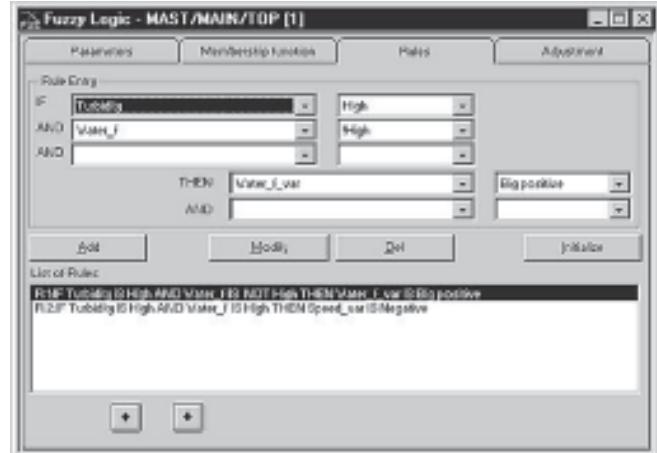
■ Нечеткая логика имеет свои специальные области, в которых превосходно работает: это области, где важен практический опыт, нюансы при принятии решений и учет нелинейных и субъективных параметров, не говоря уже о противоречащих друг другу факторах принятия решения. Контакт со специалистами компании Schneider поможет пользователям и разработчикам найти подходящее решение для ответа на их закономерных вопрос:

“Какие несомненные преимущества может внести нечеткая логика в мое приложение?”

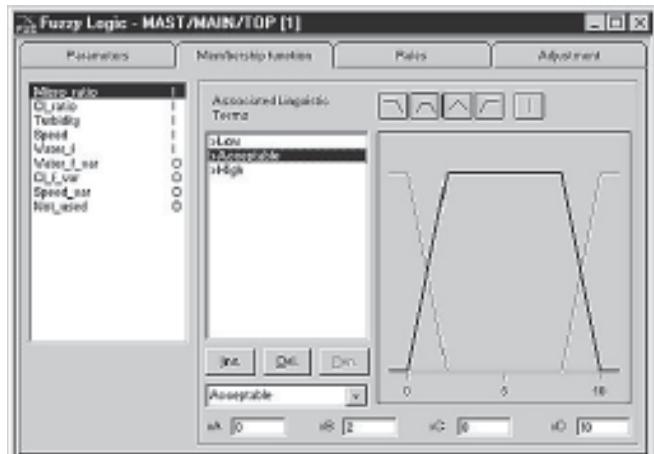
а – конфигурация модуля нечеткой логики



с – запись правил



б – определение функций принадлежности



д - симуляция - подтверждение

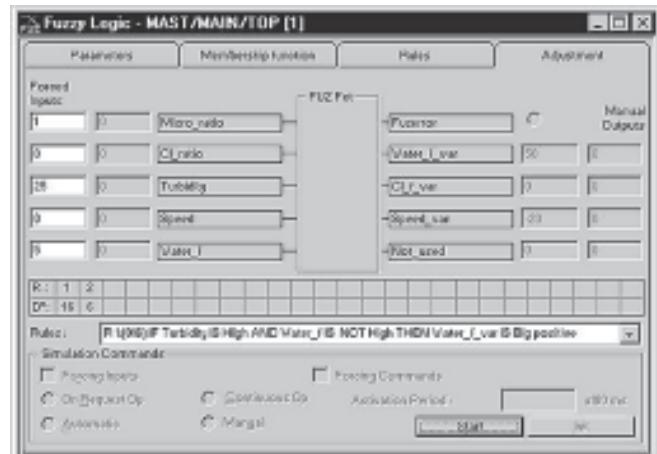


Рис. 30. ПЛК компании Schneider имеют в среде программирования на ПК дружественный пользовательский интерфейс для конфигурирования нечеткой логики

# Приложение

## Операции между нечеткими множествами

В таблице на [рис. 31](#) представлены операторы Заде.

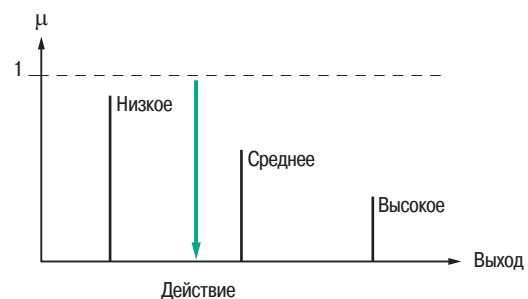
		Оператор Заде	Логическая операция
Пересечение		$\mu_{A \cap B} = \text{MIN}(\mu_A, \mu_B)$	AND
Объединение		$\mu_{A \cup B} = \text{MAX}(\mu_A, \mu_B)$	OR
Отрицание		$\mu_{\bar{A}} = 1 - \mu_A$	NOT

[Рис. 31.](#) Операции между нечеткими множествами

## Функции принадлежности одноэлементного множества

Функции принадлежности одноэлементного множества часто используются в качестве выходных функций правил нечеткой логики.

Поскольку они допускают производить интерполяцию между правилами как для функций принадлежности треугольной формы (к примеру), дальнейшие вычисления упрощаются. Нет необходимости в вычислении максимума выходной функции принадлежности (агрегация) и поиск центра тяжести также упрощается. На [рис. 32](#) представлен такой пример.



[Рис. 32.](#) Дефазификация функции принадлежности одноэлементного множества

### Нечеткий логический вывод: нечеткий вывод и обобщенное правило дедукции (Generalised Modus Ponens)

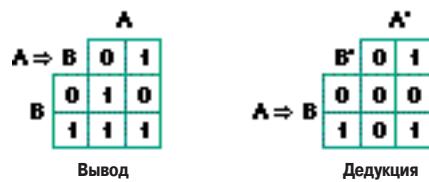
Как показано на [рис. 33](#), традиционный, направленный вперед механизм логического вывода "начиная с фронта" или "modus ponens" заключается в использовании правил, также известных как выводы, и механизма дедукции (modus ponens) для того, чтобы получить результат на основании имеющихся фактов.

Вывод " $A \Rightarrow B$ " считается истинным до тех пор, пока не становится недействительным ( $A$  истинно и  $B$  ложно): см. [рис. 34](#). Если есть информация, где вывод является истинным, а где ложным, то по правилу дедукции можно получить результат  $B'$  при анализе  $A'$ .

Аналогичный теоретический принцип можно распространить в нечеткой логике. Общая диаграмма приведена на [рис. 35](#).



*Рис. 33. Принципы логического вывода «начиная с фронта»*



*Рис. 34. Принцип вывода и правила дедукции*



*Рис. 35. Принцип нечеткого логического вывода*

Механизм обобщения вывода также известен как "нечеткий вывод". Существует несколько операторов нечеткого вывода, включая перечисленные ниже:

Оператор МАМДАНИ:  $\mu_{A \Rightarrow B} = \text{MIN}(\mu_A, \mu_B)$

Оператор ЛАРСЕНА:  $\mu_{A \Rightarrow B} = \mu_A \cdot \mu_B$

Оператор ЛУКАСЕВИЧА:  $\mu_{A \Rightarrow B} = \text{MIN}(1, 1 - \mu_A + \mu_B)$

Нечеткий вывод работает аналогично традиционному, при этом  $A$  и  $B$  представляют собой нечеткие множества.

Механизм, основанный на обобщенном правиле дедукции, подчиняется следующей формуле и используется для определения вывода нечеткого множества  $B'$ . Чаще всего применяется оператор  $T$ : Минимум (также известный как оператор Заде).

$$\mu_{B'}(y) = \text{MAX}_x(T(\mu_A(x), \mu_{A \Rightarrow B}(x, y)))$$

где  $T$ : оператор правила дедукции ( $t$  - стандартный).

Поведение оператора Лукасевича аналогично поведению традиционного вывода, когда мы ограничиваемся только булевыми переменными. Этот случай не подходит для операторов Ларсена и операторов Мамдани, которые используются в базах правил Мамдани. Такие операторы чаще всего используются поскольку:

- они имеют высокую степень устойчивости в приложениях.
  - значительно упрощаются вычисления и графическая интерпретация (см. секцию 2.4). Вычисления по входу  $x$  и выходу  $y$  производятся по формулам, приведенным ниже:
- $$\mu_{B'}(y) = \text{MAX}_x (\text{Min} (\mu_A(x), \mu_{A \Rightarrow B}(x, y)))$$
- $$= \text{Min} (\mu_B(y), \text{MAX}_x (\text{Min} (\mu_A(x), \mu_{A \Rightarrow B}(x, y))))$$

# Библиография

## Стандарты

МЭК 61131-7: Программируемые контроллеры часть 7  
Программирование нечеткого управления.

## Разнообразные работы

- Fuzzy models for pattern recognition.  
IEEE Press, 1992.  
James C. BEZDEK & Sanker K. PAL.
- Fuzzy sets and systems: Theory and applications.  
Academic Press 1980, Mathematics in Sciences and Engineering vol. 144.  
D. DUBOIS, H. PRADE.
- Evaluation subjective ; méthodes, applications et enjeux.  
Les cahiers des clubs CRIN, club CRIN logique floue.
- A.I. and expert system myths, legends and facts.  
IEEE Expert 02/90, pp 8-20, 29 réf.  
M.S. FOX.
- La logique floue et ses applications.  
Addison-Wesley, 1995.  
Bernadette BOUCHON-MEUNIER.

## Internet

- <http://www-isis.ecs.soton.ac.uk/research/nfinfo/fuzzy.html>
- <http://www.ortech-engr.com/fuzzy/reservoir.html>
- <http://www-cgi.cs.cmu.edu/afs/cs.cmu.edu/project/ai-repository/ai/areas/fuzzy/0.html>



# Schneider Electric в странах СНГ

## Беларусь

### Минск

220006, ул. Белорусская, 15, офис 9  
Тел.: (37517) 226 06 74, 227 60 34, 227 60 72

## Казахстан

### Алматы

050050, ул. Табачнозаводская, 20  
Швейцарский центр  
Тел.: (727) 244 15 05 (многоканальный)  
Факс: (727) 244 15 06, 244 15 07

### Астана

010000, ул. Бейбитшилик, 18  
Бизнес-центр «Бейбитшилик 2002», офис 402  
Тел.: (3172) 91 06 69  
Факс: (3172) 91 06 70

## Атырау

060002, ул. Абая, 2-А  
Бизнес-центр «Сугас-С», офис 407  
Тел.: (3122) 32 31 91, 32 66 70  
Факс: (3122) 32 37 54

## Россия

### Волгоград

400089, ул. Профсоюзная, 15, офис 12  
Тел.: (8442) 93 08 41

### Воронеж

394026, пр-т Труда, 65, офис 267  
Тел.: (4732) 39 06 00  
Тел./факс: (4732) 39 06 01

### Екатеринбург

620219, ул. Первомайская, 104  
Офисы 311, 313  
Тел.: (343) 217 63 37  
Факс: (343) 217 63 38

### Иркутск

664047, ул. 1-я Советская, 3 Б, офис 312  
Тел./факс: (3952) 29 00 07, 29 20 43

### Казань

420107, ул. Спартаковская, 6, этаж 7  
Тел./факс: (843) 526 55 84 / 85 / 86 / 87 / 88

### Калининград

236040, Гвардейский пр., 15  
Тел.: (4012) 53 59 53  
Факс: (4012) 57 60 79

### Краснодар

350020, ул. Коммунаров, 268 В  
Офисы 316, 314  
Тел.: (861) 210 06 38, 210 14 45  
Факс: (861) 210 06 02

### Красноярск

660021, ул. Горького, 3 А, офис 302  
Тел.: (3912) 56 80 95  
Факс: (3912) 56 80 96

### Москва

129281, ул. Енисейская, 37  
Тел.: (495) 797 40 00  
Факс: (495) 797 40 02

### Мурманск

183038, ул. Воровского, д. 5/23  
Конгресс-отель «Меридиан», офис 739  
Тел.: (8152) 28 86 90  
Факс: (8152) 28 87 30

### Нижний Новгород

603000, пер. Холодный, 10 А, этаж 8  
Тел./факс: (831) 278 97 25, 278 97 26

## Новосибирск

630005, Красный пр-т, 86, офис 501  
Тел.: (383) 358 54 21  
Тел./факс: (383) 227 62 53

## Пермь

614010, Комсомольский пр-т, 98, офис 11  
Тел./факс: (342) 290 26 11 / 13 / 15

## Ростов-на-Дону

344002, ул. Социалистическая, 74, литер A  
Тел.: (863) 200 17 22, 200 17 23  
Факс: (863) 200 17 24

## Самара

443096, ул. Коммунистическая, 27  
Тел./факс: (846) 266 41 41, 266 41 11

## Санкт-Петербург

198103, ул. Чиолковского, 9, кор. 2 А  
Тел.: (812) 320 64 64  
Факс: (812) 320 64 63

## Сочи

354008, ул. Виноградная, 20 А, офис 54  
Тел.: (8622) 96 06 01, 96 06 02  
Факс: (8622) 96 06 02

## Уфа

450098, пр-т Октября, 132/3 (бизнес-центр КПД)  
Блок-секция № 3, этаж 9  
Тел.: (347) 279 98 29  
Факс: (347) 279 98 30

## Хабаровск

680000, ул. Муравьева-Амурского, 23, этаж 4  
Тел.: (4212) 30 64 70  
Факс: (4212) 30 46 66

## Украина

Днепропетровск  
49000, ул. Глинки, 17, этаж 4  
Тел.: (380567) 90 08 88  
Факс: (380567) 90 09 99

## Донецк

83087, ул. Инженерная, 1 В  
Тел.: (38062) 385 48 45, 385 48 65  
Факс: (38062) 385 49 23

## Киев

03057, ул. Смоленская, 31-33, кор. 29  
Тел.: (38044) 538 14 70  
Факс: (38044) 538 14 71

## Львов

79015, ул. Тургенева, 72, кор. 1  
Тел./факс: (38032) 298 85 85

## Николаев

54030, ул. Никольская, 25  
Бизнес-центр «Александровский», офис 5  
Тел./факс: (380512) 58 24 67, 58 24 68

## Одесса

65079, ул. Куликово поле, 1, офис 213  
Тел./факс: (38048) 728 65 55, 728 65 35

## Симферополь

95013, ул. Севастопольская, 43/2, офис 11  
Тел.: (380652) 44 38 26  
Факс: (380652) 54 81 14

## Харьков

61070, ул. Академика Проскуры, 1  
Бизнес-центр «Telesens», офис 569  
Тел.: (38057) 719 07 79  
Факс: (38057) 719 07 49

## Центр поддержки клиентов

Тел.: 8 (800) 200 64 46 (многоканальный)  
Тел.: (495) 797 32 32, факс: (495) 797 40 04  
ru.csc@ru.schneider-electric.com  
www.schneider-electric.ru