

МЕТОДЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

ГАЗАНОВА Н.Ш.

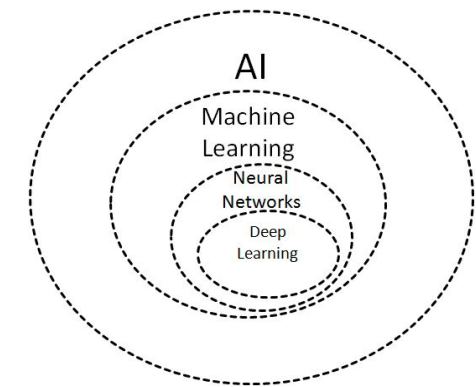
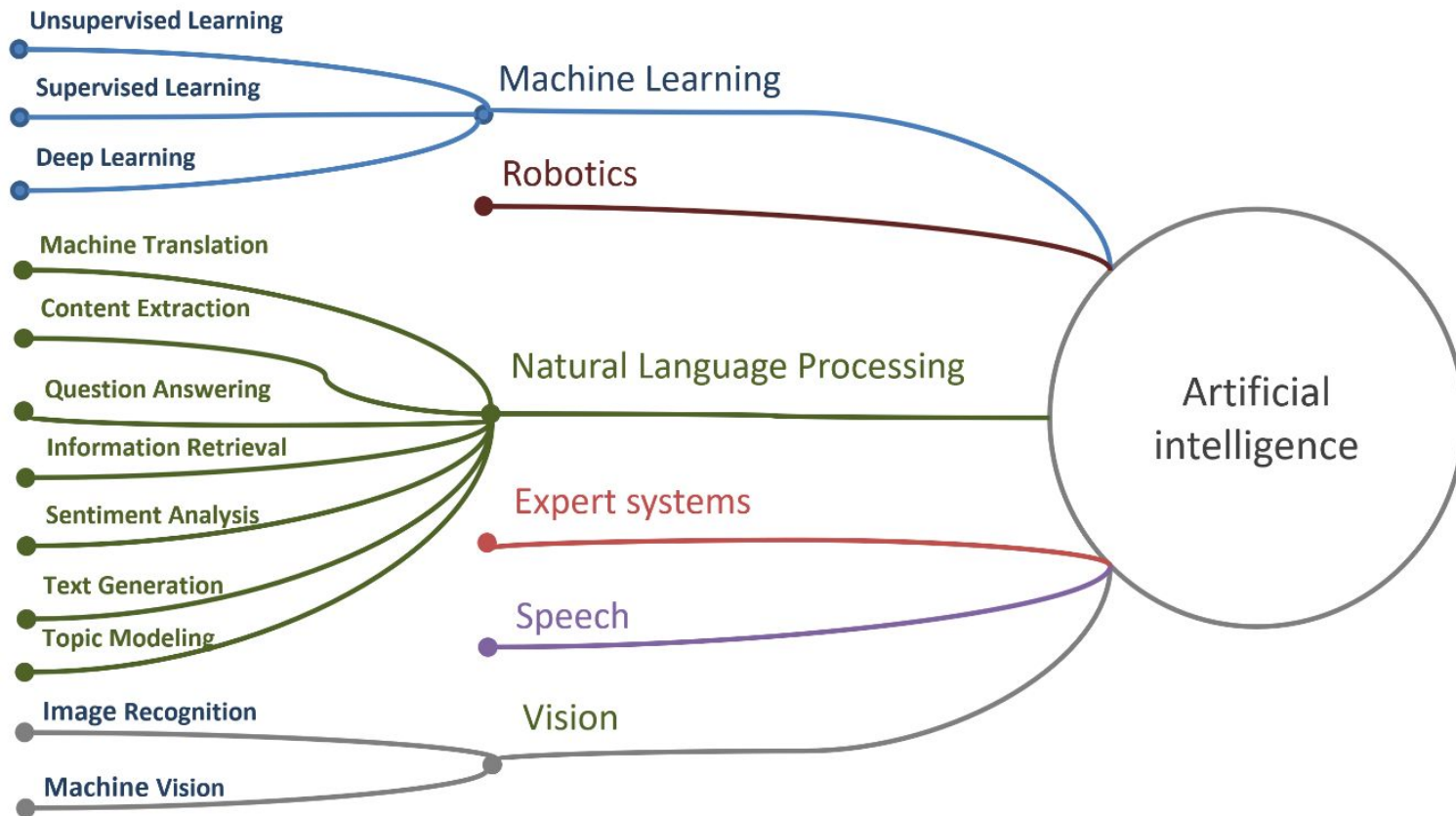




1. ВВЕДЕНИЕ В ИИ



ПОДРАЗДЕЛЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА



ФУНКЦИОНАЛЬНЫЙ ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

Моделирование человеческих рассуждений

Формальные теории

Доказательство теорем

Логическое

программирование

Представление знаний:

фреймы,

семантические сети...

Символьные

преобразования

выражений

Экспертные системы

Нечеткие логики...

не учится

Анализ данных и машинное обучение

Обучение с учителем

Пространство

признаков. Линейные
разделители.

Байесовское обучение.

Ошибки 1 и 2 рода.

Обучение без учителя

Кластерный анализ.

Снижение размерности

Нейронные сети

Обратное

распространение...

не рассуждает



2. ПРЕДСТАВЛЕНИЕ ЗНАНИЙ



МОДЕЛИ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Модель	Достоинства	Недостатки
Продукции	Наглядность, высокая модульность, легкость внесения дополнений и изменений, простота механизма логического вывода, простота интерпретации.	При накоплении большого числа (нескольких сотен) продукций они начинают противоречить друг другу, возникают трудности при добавлении правил, зависящих от уже имеющихся в базе знаний, отсутствует целостный образ знаний, неясна взаимосвязей между правилами.
Семантические сети	Наглядность, соответствует представлениям об организации долговременной памяти человека, позволяет снизить объем хранимых данных.	Представляют собой пассивные структуры, для обработки которых необходим специальный аппарат формального вывода и планирования, произвольная структура и различные типы вершин и связей усложняют процедуру обработки информации, сетевая модель не дает ясного представления о структуре предметной области.
Фреймы	Гибкость, наглядность, удобный способ включения процедурных знаний, сводимость к другим моделям, модульность.	Отсутствие универсальной процедуры управления выводом кроме механизма наследования, является идеологической концепцией.



ПРОДУКЦИОННАЯ МОДЕЛЬ

Продукция – это предложение-образец вида «Если, то», по которому осуществляется поиск в базе знаний.

СЕМАНТИЧЕСКАЯ СЕТЬ

Семантическая сеть — это ориентированный граф, вершины которого — понятия, а дуги — отношения между ними. Узлы в семантической сети обычно соответствуют объектам, концепциям, событиям или понятиям.

Любой фрагмент сети, например одна вершина, две вершины и соединяющие их дуги, называют подсетью. Логический вывод (поиск решения) на семантической сети заключается в том, чтобы найти или сконструировать подсеть, удовлетворяющую некоторым условиям.

ФРЕЙМЫ

Фрейм (англ. frame) - абстрактный образ для представления некоторого стереотипа восприятия. Каждый фрейм имеет собственное название и список слотов и их значений.

Значениями могут быть данные любого типа, а также название другого фрейма. Таким образом, фреймы образуют сеть. Кроме того, существует связь между фреймами типа АКО (a kind of), которая указывает на фрейм более высокого уровня иерархии, откуда неявно наследуются список и значения слотов. При этом возможно множественное наследование – перенос свойств от нескольких прототипов.

Любой фрейм может быть представлен следующим образом:

(ИМЯ ФРЕЙМА:

(имя 1-го слота: значение 1-го слота),

(имя 2-го слота: значение 2-го слота),

.....

(имя N-го слота: значение N-го слота))

или таблично.

ПРЕДМЕТ ЛОГИКИ

Предметом изучения логики являются следующие стороны и особенности мышления:

- 1) Логика изучает элементарные единицы мышления – понятия, а также операции, позволяющие их связать в мысли – это суждения и умозаключения.
- 2) Законы мышления, выработка которых и составляет одну из главных задач логики.
- 3) Правильность мышления в логики рассматривается не со стороны содержания мышления, а исключительно со стороны формы выражения мысли.

ОБЪЕКТ ИЗУЧЕНИЯ

Объектом изучения логики являются рассуждения (умозаключения), истинность или ложность которых определяются связями между входящими в них утверждениями.

Умозаключение — это логическая операция, в результате которой из одного или нескольких принятых утверждений (посылок) получается новое утверждение — заключение (следствие).

Все люди смертны; Сократ – человек; следовательно, Сократ смертен.

Первые два высказывания – это посылки вывода, третье – его заключение.

ДЕДУКЦИЯ

Во всех случаях, когда требуется рассмотреть какое-то явление на основании уже известного общего принципа и вывести в отношении этого явления необходимое заключение, мы умозаключаем в форме дедукции.

Если данное число делится на 6, то оно делится на 3. Данное число делится на 6.

Следовательно данное число делится на 3.

ИНДУКЦИЯ

Полная индукция – это метод доказательства, при котором вывод доказываемся для конечного числа частных случаев.

Неполная индукция – это метод доказательства, при котором в умозаключении вывод не следует логически из посылок и может содержать информацию, отсутствующую в них (запрещена в формальной логике).

Аргентина является республикой; Бразилия — республика; Венесуэла — республика; Эквадор — республика.

Аргентина, Бразилия, Венесуэла, Эквадор — латиноамериканские государства.

Следовательно — все латиноамериканские государства являются республиками.

Италия — республика; Португалия — республика; Финляндия — республика; Франция — республика.

Италия, Португалия, Финляндия, Франция — западноевропейские страны.

Следовательно — все западноевропейские страны являются республиками.

ЛОГИКА ВЫСКАЗЫВАНИЙ

Одним из основных объектов исследования математической логики является высказывание.

Высказывание (простое высказывание) – это утвердительное повествовательное предложение, являющееся истинным или ложным.

В математической логике высказывания обозначаются заглавными буквами латинского алфавита: A, B, C, \dots
ИСТИНА — 1, ЛОЖЬ — 0

ЛОГИЧЕСКИЕ СВЯЗКИ

Связка	Операция	Обозначение	Правила чтения	Пример A – Преподаватель читает лекции, B – Преподаватель ведет практику
Не	Отрицание	\bar{A}	Не A	\bar{A} – Преподаватель не читает лекции, \bar{B} – Преподаватель не ведет практику
И	Конъюнкция	$A \wedge B$	A и B	$A \wedge B$ – Преподаватель читает лекции и (преподаватель) ведет практику
Или	Дизъюнкция	$A \vee B$	A или B	$A \vee B$ – Преподаватель читает лекции или (преподаватель) ведет практику
Если..., то...	Импликация	$A \Rightarrow B$	Если A , то B	$A \Rightarrow B$ – Если преподаватель читает лекции, то он (преподаватель) ведет практику
..., тогда и только тогда, когда...	Эквиваленция	$A \Leftrightarrow B$	A тогда и только тогда, когда B	$A \Leftrightarrow B$ – Преподаватель читает лекции, тогда и только тогда, когда он (преподаватель) ведет практику

ПРИМЕР

Дано высказывание: «Если я поеду автобусом, и автобус опоздает, то я опоздаю на работу. Если я опоздаю на работу, то я не сделаю в срок важную работу. Если я не сделаю в срок важную работу, то я лишусь премии. Следовательно, если я не поеду автобусом, то я сделаю в срок важную работу и не лишусь премии».

Решение: Введем обозначения:

«я поеду автобусом» – A , «автобус опоздает» – B , «я опоздаю на работу» – C , «я лишусь премии» – D , «я сделаю в срок важную работу» – E .

Данное высказывание является примером **умозаключения**. Формальный признак этого - наличие слова «следовательно», что свидетельствует о некотором выводе на основании определенных исходных данных. Тогда основание для вывода называется посылкой, а результат вывода – заключением. Для записи высказывания такой структуры имеется специальная форма в виде дроби, в которой вверху записывается посылка, а внизу – заключение.

Формальная запись высказывания имеет вид:

$$\frac{(A \& B \rightarrow C), (C \rightarrow \neg E), (\neg E \rightarrow D)}{(\neg A \rightarrow E \& \neg D)}$$

ЗАДАНИЕ

Представить в виде логической формулы:

- *Солнце светит тогда и только тогда, когда на небе нет туч.*
- *Подготовка специалистов высокой квалификации возможна лишь на базе всемерного развития вузовской науки, усиления связи вузовской, академической и отраслевой науки, обеспечения единства научной и учебной работы*
- *Хлеба уцелеют в различных погодных условиях тогда и только тогда, когда будут выполнены все мелиоративные работы, а если хлеба не уцелеют, то фермеры обанкротятся и оставят фермы*
- *Обвиняемый может быть либо исполнителем, либо организатором совершенного преступления. Обвиняемый является организатором преступления. Следовательно, он не является исполнителем преступления*
- *Чтобы завтра пойти на занятия, я должен встать рано. Если я сегодня пойду в кино, то лягу спать поздно. Если я лягу спать поздно, то встану поздно. Следовательно, либо я не пойду в кино, либо не пойду на занятия.*

ЛОГИКА ПРЕДИКАТОВ 1 –ГО ПОРЯДКА

$P(x)$ – одноместный предикат
 $x \in M$

$I_p = \{x \mid x \in M, P(x)=1\}$ – область истинности

$P(x,y)$ – двухместный предикат (определен на множестве $M=M_1 \times M_2$ и принимает значения из множества $\{1,0\}$)

$P(x_1, x_2, \dots, x_n)$ – n-местный (n-арный) предикат

ОПЕРАЦИИ НАД ПРЕДИКАТАМИ

1. Конъюнкция предикатов $P(x)$ и $Q(x)$

$P(x) \& Q(x)$

Область истинности $I_{P \& Q} = I_P \cap I_Q$

2. Дизъюнкция $P(x)$ и $Q(x)$

$P(x) \vee Q(x)$

Область истинности $I_{P \vee Q} = I_P \cup I_Q$

3. Отрицание предиката $P(x)$

$\neg P(x)$

Область истинности $I_{\neg P} = M \setminus I_P$

4. Импликация предикатов $P(x)$ и $Q(x)$

$P(x) \rightarrow Q(x)$

Область истинности $I_{P \rightarrow Q} = I_{\neg P} \cup I_Q = M \setminus I_P \cup I_Q$

КВАНТОРЫ

Квантор всеобщности

\forall - для любого, для каждого, для всех.

\exists - существует, найдётся.

$P(x)$ – x называют свободной переменной

$\forall_x P(x)$ – x называю связанной квантором \forall

$P(x) = \langle\langle x \text{-четное число} \rangle\rangle$, определенные на множестве натуральных чисел \mathbb{N}

$\exists_x P(x)$ – «существует натуральное число x , что x -четно»

$\forall_x P(x)$ – «любое натуральное число - четно»

ОПРЕДЕЛЕНИЯ

Множество, на котором заданы предметные переменные и предметные постоянные, называют *областью определения предиката* или *универсумом*.

Множество значений предметных переменных, на котором предикат принимает значение «истины», называют *множеством истинности предиката*.

Суждение, в котором утверждается или отрицается наличие каких-либо свойств или отношений у **части** предметных переменных, называют *частным суждением*.

Предметную переменную называют *связанной*, если она находится под действием квантора. Если вхождение предметной переменной свободно от квантора, ее называют *свободной*.

Между элементами области определения предиката могут быть заданы функциональные отношения. На это указывает функциональный символ: $f(x_1, x_2, \dots, x_n)$.

ОБЩЕЗНАЧИМОСТЬ И ВЫПОЛНИМОСТЬ

Формула логики предикатов называется **выполнимой** в области M , если существуют значения переменных, входящих в эту формулу и отнесенных к области M , при которых формула принимает истинное значение.

Формула выполнима, если существует область, на которой выполнима эта формула.

Формула логики предикатов называется **тождественно истинной** в области M , если для всех значений переменных из области M формула принимает истинное значение.

Формула, тождественно истинная в любой области, называется **общезначимой (логическим законом)**.

Пример 1. $\forall_x (P(x) \vee \neg P(x))$ - это пример логического закона.

Пример 2. Определить выполнимость формулы $\exists_x (P(x))$

Пусть M - это множество натуральных чисел, $P(x) = \langle x \text{ - простое число} \rangle$. Тогда $\exists_x (P(x))$

- это истинное высказывание, т.е. формула $\exists_x (P(x))$ выполнима.

ИСЧИСЛЕНИЕ ВЫСКАЗЫВАНИЙ И ИСЧИСЛЕНИЕ ПРЕДИКАТОВ

В математике термином «исчисление» обозначаются разные области знаний, а также формальные теории (множества формул, полученных из аксиом с помощью правил вывода).

Правила вывода новых высказываний, основанные на *известных высказываниях*, имеющих значение «истина», представляют *исчисление высказываний* (ИВ). В этом случае высказывания, из которых делают вывод новых высказываний, называют *посылками*, а получаемое новое высказывание – *заключением*.

В исчислении предикатов изучаются те же логические законы, что и в алгебре предикатов, но изучение их ведется аксиоматически.

АКСИОМЫ ИВ

Среди множества тождественно истинных формул можно выделить подмножества, которые представляют аксиомы исчисления высказываний. В зависимости от числа логических связок могут быть найдены различные полные системы аксиом:

Логические связки \rightarrow и \neg :

$$(A \rightarrow (B \rightarrow A)),$$

$$(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C)),$$

$$(\neg A \rightarrow \neg B) \rightarrow ((\neg A \rightarrow B) \rightarrow A).$$

Логические связки \neg , $\&$, \vee , \rightarrow :

$$A1. A \rightarrow (B \rightarrow A),$$

$$A2. (A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C)),$$

$$A3. (A \& B) \rightarrow A,$$

$$A4. (A \& B) \rightarrow B,$$

$$A5. A \rightarrow (B \rightarrow (A \& B)),$$

$$A6. A \rightarrow (A \vee B),$$

$$A7. B \rightarrow (A \vee B),$$

$$A8. (A \rightarrow C) \rightarrow ((B \rightarrow C) \rightarrow ((A \vee B) \rightarrow C)),$$

$$A9. (A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A),$$

$$A10. (A \rightarrow B) \rightarrow ((A \& C) \rightarrow (B \& C)),$$

$$A11. (A \rightarrow B) \rightarrow ((A \vee C) \rightarrow (B \vee C)),$$

$$A12. \neg \neg A \rightarrow A$$

Существует два способа доказательства того, что приведенные ППФ являются аксиомами:

- через таблицы истинности – каждая сложная формула, соответствующая аксиоме, будет принимать только значение «истина»,
- через эквивалентные преобразования – всегда будет получаться значение истины.

АКСИОМЫ ИП

Схемы аксиом для импликации, конъюнкции, дизъюнкции и отрицания остаются теми же. Добавляется новая группа.

Аксиома 1. $\forall_x A \rightarrow A (t/x)$

Аксиома 2. $A (t/x) \rightarrow \exists_x A$

(t/x) – t вместо x

ПРИМЕНЕНИЕ КВАНТОРОВ К ЕСТЕСТВЕННОМУ ЯЗЫКУ

$\forall x A(x)$	<p>Для любого x (имеет место) $A(x)$ $A(x)$ при произвольном x Для всех x (верно) $A(x)$ $A(x)$, каково бы ни было x Для каждого x (верно) $A(x)$ Всегда имеет место $A(x)$ Каждый обладает свойством A Свойство A присуще всем Всё удовлетворяет A Любой объект является A Всякая вещь обладает свойством A</p>	$\forall x \neg A(x)$	<p>Для всякого x неверно $A(x)$ $A(x)$ всегда ложно ничто не обладает свойством A всё суть не A</p>
$\exists x A(x)$	<p>Для некоторых x (имеет место) $A(x)$ Для подходящего x (верно) $A(x)$ Существует x, для которого (такой, что) $A(x)$ Имеется x, для которого (такой, что) $A(x)$ Найдется x, для которого (такой, что) $A(x)$ У некоторых вещей есть признак A Хотя бы для одного x (верно) $A(x)$ Кто-нибудь относится к (есть) A По крайней мере, один объект есть A</p>	$\neg \exists x A(x)$	<p>Не существует x такого, что $A(x)$ Нет никакого x, такого, что $A(x)$ Нет x, такого, что $A(x)$ $A(x)$ не выполняется ни для какого x Ничто не обладает свойством A Никто не есть A Неверно, что для некоторых x $A(x)$ Ни для какого x не верно $A(x)$</p>
$\neg \forall x A(x)$	<p>Не для каждого x (имеет место) $A(x)$ Не при всяком x (верно) $A(x)$ $A(x)$ оказывается истинным не для всех x Не всё обладает свойством A Не все суть A A не всегда верно</p>	$\exists x \neg A(x)$	<p>Для некоторого x неверно $A(x)$ Что-то не обладает свойством A Кто-то есть не A</p>

ПРИМЕР

Судья, являющийся родственником потерпевшего, не может участвовать в рассмотрении дела.

Пусть предметная переменная x определена на множестве индивидов.

Введем предикаты:

$P(x)$ - быть судьёй (x – судья),

$D(x)$ - быть родственником потерпевшего (x – родственник потерпевшего),

$C(x)$ - участвовать в рассмотрении дела (x – участвует в рассмотрении дела).

Тогда формула имеет вид:

$$\forall x (P(x) \& D(x) \rightarrow \neg C(x))$$

ПРИМЕРЫ

$P_1(x) := \langle x \text{ – студент} \rangle$,

$P_2(x, y) := \langle x \text{ обучается в университете } y \rangle$,

$P_3(x) := \langle x \text{ имеет зачетную книжку} \rangle$,

M – множество индивидов,

Y – множество учебных заведений,

$x \in M, y \in Y$ – предметные переменные.

Тогда:

- $\exists_x \exists_y (P_1(x) \& P_2(x, y) \& \neg P_3(x))$ - «существуют студенты (x) некоторых университетов (y), которые не имеют зачетной книжки» - частное суждение,
- $\forall_x (P_1(x) \& P_3(x))$ - «все студенты имеют зачетные книжки» - общее суждение,
- $\forall_x \forall_y (P_1(x) \& P_2(x, y) \& P_3(x))$ - «все студенты всех университетов имеют зачетные книжки» - общее суждение.

ПРОБЛЕМЫ В ИП

Для обоснования исчисления предикатов, как для любой аксиоматической теории, необходимо рассмотреть проблемы разрешимости и непротиворечивости.

Проблема разрешимости исчисления предикатов есть проблема поиска эффективной процедуры в доказательстве. Исчисление предикатов – пример неразрешимой формальной системы, т.к. нет единого эффективного алгоритма в доказательстве любой формулы. Наличие кванторов, ограничивающих области определения, наличие сколемовских функций не позволяет использовать таблицы истинности.

Проблема непротиворечивости исчисления предикатов заключена в доказательстве невыводимости формулы и ее отрицания. Исчисление предикатов непротиворечиво, т. к. каждая доказуемая формула является тождественно истинной формулой. Тогда ее отрицание является тождественно ложной формулой и при доказательстве в исчислении предикатов ведет к противоречию.

ЗАКОНЫ АЛГЕБРЫ ПРЕДИКАТОВ

Закон	Равносильные формулы
Коммутативности	$\forall_x \forall_y (F(x, y)) \equiv \forall_y \forall_x (F(x, y))$ $\exists_x \exists_y (F(x, y)) \equiv \exists_y \exists_x (F(x, y))$
Дистрибутивности	$\forall_x (F_1(x)) \& \forall_x (F_2(x)) \equiv \forall_x (F_1(x) \& F_2(x))$ – для формул с кванторами всеобщности по одной переменной x $\exists_x (F_1(x)) \vee \exists_x (F_2(x)) \equiv \exists_x (F_1(x) \vee F_2(x))$ – для формул с кванторами существования по одной переменной x
Идемпотентности	$\forall_x (F(x)) \vee \forall_x (F(x)) \equiv \forall_x (F(x))$ $\forall_x (F(x)) \& \forall_x (F(x)) \equiv \forall_x (F(x))$ $\exists_x (F(x)) \vee \exists_x (F(x)) \equiv \exists_x (F(x))$ $\exists_x (F(x)) \& \exists_x (F(x)) \equiv \exists_x (F(x))$
Исключения третьего	$\forall_x (F(x)) \vee \neg \forall_x (F(x)) = \text{и}$ $\exists_x (F(x)) \vee \neg \exists_x (F(x)) = \text{и}$
Противоречия	$\forall_x (F(x)) \& \neg \forall_x (F(x)) = \text{л}$ $\exists_x (F(x)) \& \neg \exists_x (F(x)) = \text{л}$
Отрицание кванторов	$\forall_x (\neg F(x)) \equiv \neg \exists_x (F(x))$ $\forall_x (F(x)) \equiv \neg \exists_x (\neg F(x))$ $\exists_x (\neg F(x)) \equiv \neg \forall_x (F(x))$ $\exists_x (F(x)) \equiv \neg \forall_x (\neg F(x))$
Дополнения	$\neg(\neg \forall_x (F(x))) \equiv \forall_x (F(x))$ $\neg(\neg \exists_x (F(x))) \equiv \exists_x (F(x))$

ПРАВИЛА СТАНДАРТИЗАЦИИ ПРЕДЛОЖЕНИЙ

Правило 1. Исключение знаков импликации.

Знак импликации можно исключить подстановкой в исходном утверждении вместо « $A \rightarrow B$ » формулы $\overline{A} \vee B$

Правило 2. Уменьшение области действия знаков отрицания.

Для этого применяются следующие равносильности:

Формула	Результат
$\overline{A \wedge B}$	$\overline{A} \vee \overline{B}$
$\overline{A \vee B}$	$\overline{A} \wedge \overline{B}$
$\overline{\overline{A}}$	A
$\overline{\forall x A}$	$\exists x \overline{A}$
$\overline{\exists x A}$	$\forall x \overline{A}$

Правило 3. Стандартизация переменных.

В области действия каждого квантора переменная, связываемая им, является **немой** переменной. Поэтому везде в области действия кванторов ее можно заменить произвольной переменной, входящей в область действия этих кванторов. Это не приведет к изменению значения истинности правильно построенной формулы (ППФ). Другими словами, стандартизация переменных в ППФ означает переименование немых переменных с тем, чтобы каждый квантор имел свою собственную немую переменную. Так, вместо $\forall x \{P(x) \rightarrow \exists x Q(x)\}$ следует записать $\forall x \{P(x) \rightarrow \exists y Q(y)\}$.

ПРАВИЛА СТАНДАРТИЗАЦИИ ПРЕДЛОЖЕНИЙ

Правило 4. Исключение кванторов существования (\exists).

Рассмотрим ППФ следующего вида:

$$\forall_y \exists_x P(x,y)$$

которую можно интерпретировать, например, так: «для всех y существует такой x (возможно зависимый от y), что x больше y ». Заметим, что поскольку квантор существования ($\exists x$) находится внутри области действия квантора всеобщности (" y "), допускается, что x , который «существует» может зависеть от значения y . Пусть эта зависимость определяется явно с помощью некоторой функции $q(x)$, отображающей каждое значение y в x , который «существует». Такая функция называется **функцией Сколема**.

Если вместо x , который «существует», подставить функцию Сколема, то можно исключить квантор существования, т.е. вместо $\forall_y \exists_x P(x,y)$ можно записать $\forall_y P(q(y),y)$.

В связи с этой процедурой отметим еще следующие особенности применения функции Сколема:

1. Для обозначения функции Сколема не должны использоваться функциональные буквы, которые уже имеются в формуле.
2. Если квантор существования находится в области двух и более кванторов всеобщности, то функция Сколема будет зависеть соответственно от двух и более аргументов.
3. Если исключаемый квантор существования не принадлежит области действия ни одного квантора всеобщности, то функция Сколема не содержит аргумента, т.е. является константой. Так, формула $\exists_x F(x)$ при исключении квантора существования преобразуется в формулу $F(a)$, где a - константа, про которую известно, что она «существует».

ПРАВИЛА СТАНДАРТИЗАЦИИ ПРЕДЛОЖЕНИЙ

Правило 5. Приведение к предваренной нормальной форме.

На этом этапе уже не осталось кванторов существования, а каждый квантор всеобщности имеет свою переменную. Теперь можно перенести все кванторы всеобщности в начало ППФ, расположенную за ними. Полученная в результате этого ППФ называется **предваренной нормальной формой** ППФ.

ППФ в предваренной нормальной форме состоит из цепочек кванторов всеобщности с различными своими переменными. Эта цепочка называется префиксом, а расположенная за ней формула, не содержащая кванторов, называется **матрицей**.

Правило 6. Приведение матрицы КНФ.

Любую матрицу, полученную после применения правила 5, можно представить в виде конъюнкции конечного множества дизъюнкций предикатов и (или) их отрицаний. Говорят, что такая матрица имеет **конъюнктивную нормальную форму** (КНФ).

Примеры матриц в КНФ:

$$\{P(x) \vee Q(x, y)\} \wedge \{P(w) \vee \bar{R}(y)\} \wedge Q(x, y)$$

$$P(x) \vee Q(x, y)$$

$$P(x) \wedge Q(x, y)$$

$$\bar{R}(y)$$

ПРАВИЛА СТАНДАРТИЗАЦИИ ПРЕДЛОЖЕНИЙ

Правило 7. Исключение кванторов всеобщности.

Так как все переменные ППФ должны быть связанными, то оставшиеся на этом этапе переменные относятся к кванторам всеобщности. Поскольку порядок расположения кванторов всеобщности несущественен (см. формулу 5 выше), то можно эти кванторы явным образом не указывать, условившись, что все переменные в матрице относятся к кванторам всеобщности. После этого остается лишь матрица в КНФ.

Правило 8. Исключение связки конъюнкции.

Теперь можно исключить знак конъюнкции, заменяя $A \wedge B$ двумя ППФ A, B . Результатом замены будет конечное множество ППФ, каждое из которых представляет собой дизъюнкцию атомных формул и (или) их отрицаний. **Атомную** формулу или ее отрицание называют **литералом** (или литерой), а ППФ, состоящую лишь из дизъюнкций, литералом - предложением (или дизъюнктом).

Примечание. Если вместо переменных в литералы подставляются выражения, не содержащие переменных, получается, так называемый, константный частный случай этого литерала. Например, $Q(a, f(q(b)))$ есть константный частный случай ППФ $Q(x, y)$.

ПРИМЕР

Существует диспетчерская группа управления технологическим оборудованием, которая представляет собой трехуровневую систему, включающую:

- диспетчеров производственного объединения (3^й уровень);
- заводских диспетчеров (2^й уровень);
- цеховых диспетчеров (1^й уровень).

Заводским диспетчерам подконтрольны лишь те, кто управляет цехом, и не подконтрольны диспетчеры объединения.

В результате неправильного действия одного из диспетчеров возникла неисправность оборудования, которую поручили проанализировать заводским диспетчерам. Работа цеховых диспетчеров - потенциальных виновников этой неисправности — анализировалась и проверялась лицами; действительно виновными в произошедшем. Установлено также, что диспетчера объединения не имеют к этому всему никакого отношения.

Требуется формальными методами доказать, что действительным виновником аварии является один из контролеров, т.е. заводских диспетчеров.

Для формализации задачи введем следующие обозначения:

- одноместный предикат $P(x)$ (x - диспетчер любого уровня, в обязанности которого входит задача управления технологическим оборудованием);
- одноместный предикат $Q(x)$ (x - диспетчер объединения);
- одноместный предикат $R(x)$ (x - диспетчер завода)
- одноместный предикат $C(x)$ (x - виновник аварии);
- двуместный предикат $V(x, y)$ (y проверял x).

ПРИМЕР

Тогда исходные предпосылки и соответствующие формулы могут быть записаны в следующем виде:

1) Для любого диспетчера x из того, что он является диспетчером любого уровня ($P(x)$), в обязанности которого входит управление технологическим оборудованием, и не являющегося диспетчером объединения ($\bar{Q}(x)$), следует, что существует y , проверяющий x и являющийся диспетчером завода.

$$\forall x(P(x) \wedge \bar{Q}(x) \rightarrow \exists y(V(x,y) \wedge R(y))) \quad (1)$$

2) Существует такой диспетчер x , что он является виновником аварии ($C(x)$) и в обязанности которого входит управление технологическим оборудованием ($P(x)$) и для любого проверяющего диспетчера y , если он проверяет x , то он является виновником аварии ($C(y)$).

$$\exists x\{C(x) \wedge P(x) \wedge [\forall y(V(x,y) \rightarrow C(y))]\} \quad (2)$$

3) Для любого диспетчера x , если он является виновником аварии ($C(x)$), то он не является диспетчером объединения ($\bar{Q}(x)$)

$$\forall x((C(x) \rightarrow \bar{Q}(x))) \quad (3),$$

Вывод (теорема) может быть записан в виде: существует такой диспетчер x , который является виновником аварии ($C(x)$) и диспетчером завода ($R(x)$)

$$\exists x(C(x) \wedge R(x)) \quad (4)$$

ПРИМЕР

Приведем формулы (1), (2) и (3), представляющие предпосылки, к предваренной нормальной форме и получим множество дизъюнктов (предложений).

1. исключение законов импликации

$$\forall x \left[\overline{P(x) \wedge \overline{Q(x)} \vee \exists y (V(x, y) \wedge R(y))} \right]; \quad (1a)$$

$$\exists x (C(x) \wedge P(x) \wedge \left[\forall y (\overline{V(x, y)} \vee C(y)) \right]); \quad (2a)$$

$$\forall x (\overline{C(x)} \vee \overline{Q(x)}). \quad (3a)$$

2. Уменьшение области действия законов отрицания

$$\forall x \left[\overline{P(x)} \vee Q(x) \vee \exists y (V(x, y) \wedge R(y)) \right]; \quad (1b)$$

$$\exists x (C(x) \wedge P(x) \wedge \left[\forall y (\overline{V(x, y)} \vee C(y)) \right]); \quad (2a)$$

$$\forall x (\overline{C(x)} \vee \overline{Q(x)}). \quad (3a)$$

3. Стандартизация переменных.

Эту операцию выполнять нет необходимости.

4. Исключение кванторов существования

$$\forall x \left[\overline{P(x)} \vee Q(x) \vee (V(x, f(x)) \wedge R(f(x))) \right]; \quad (1b)$$

$$C(a) \wedge P(a) \wedge \left[\forall y (\overline{V(a, y)} \vee C(y)) \right]; \quad (2b)$$

$$\forall x (\overline{C(x)} \vee \overline{Q(x)}). \quad (3a)$$

ПРИМЕР

5. Приведение к предваренной нормальной форме.

В последних формулах не осталось кванторов существования, а каждый квантор всеобщности имеет свою переменную. Теперь можно перенести все кванторы всеобщности на начало формулы и считать областью действия каждого квантора всю часть формулы, расположенную за ней. В результате мы получим предваренную нормальную форму.

$$\forall x \{ \bar{P}(x) \vee Q(x) \vee V(x, f(x)) \} \vee \{ \bar{P}(x) \vee Q(x) \vee R(f(x)) \}; \quad (1г)$$

$$C(a) \wedge P(a) \wedge \{ \forall y (\bar{V}(a, y) \vee C(y)) \}; \quad (2б)$$

$$\forall x (\bar{C}(x) \vee \bar{Q}(x)). \quad (3а)$$

6. Исклечение кванторов всеобщности и представление в виде множества дизъюнктов (предложений)

1) $\bar{P}(x) \vee Q(x) \vee V(x, f(x))$;

2) $\bar{P}(x) \vee Q(x) \vee R(f(x))$;

3) $C(a)$;

4) $P(a)$;

5) $\bar{V}(a, y) \vee C(y)$;

6) $\bar{C}(x) \vee \bar{Q}(x)$.

Отрицание вывода (теоремы) имеет вид (из формулы (4))

7) $\bar{C}(x) \vee \bar{R}(x)$.

ЗАДАНИЕ

1. $\exists_x (Q(x) \wedge \forall_y (D(y) \rightarrow T(x, y)))$
2. $\forall_x (Q(x) \rightarrow \forall_y (C(y) \rightarrow \neg T(x, y)))$
 - Т. $\forall_y (D(x) \rightarrow \neg C(x))$

ПОИСК НАИБОЛЕЕ ОБЩЕГО УНИФИКАТОРА(НОУ)

В задаче доказательства теорем методом резолюции при поиске резольвенты двух предположений необходимо выполнить следующие процедуры:

1. переменные одного предложения переименовываются так, чтобы они отличались от переменных другого предложения;
2. находится подстановка, при которой какой-либо литерал одного предложения становится дополнительным к какому-либо литералу другого предложения и производится эта подстановка в оба предложения;
3. литералы, дополнительные друг к другу, вычеркиваются;
4. если имеются одинаковые литералы, то все они, кроме одного в каком-либо предложении, вычеркиваются;
5. дизъюнкция литералов, оставшихся в обоих предложениях и есть резольвента.

ОПРЕДЕЛЕНИЕ 1

Фактор какого-либо предложения, называется следствием этого предложения, полученного так:

- а) находится подстановка, при котором какие-либо литералы одинаковы;
- б) после выполнения этой подстановки все одинаковые литералы кроме одного, вычеркиваются;
- в) дизъюнкция оставшихся литералов и есть фактор.

Процесс нахождения фактора называется **факторизацией**. Отметим, что факторизация – это реализация процедуры 4 из названных процедур резольвирования предложений.

Нетрудно видеть, что процедура поиска подстановки является важнейшей в методе резолюции.

В общем случае любую подстановку используемую при применении принципа резольвенции можно представить в виде множества упорядоченных пар

$$\Theta = \{t_1|x_1, t_2|x_2, \dots, t_n|x_n\},$$

где пара $t_i|x_i$ означает, что всюду, где производится данная подстановка, переменная x_i заменяется термом t_i . Такая пара называется подстановочной компонентой.

Например, применяя к литералу $P(x, f(y), b)$ подстановки

$$\Theta_1 = \{z|x, w|y\}$$

$$\Theta_2 = \{a|y\}$$

$$\Theta_3 = \{g(z) |x, a|y\}$$

Получим соответствующие им частные случаи исходного литерала

$$P_{\Theta_1} = P(z, f(w), b)$$

$$P_{\Theta_2} = P(x, f(a), b)$$

$$P_{\Theta_3} = P(g(z), f(a), b)$$

ОПРЕДЕЛЕНИЕ 2

Композицией двух подстановок α и β называется подстановка $\alpha\beta$, которая получается при применении подстановки β к термам подстановки α с последующим добавлением любых пар из β , содержащих переменные, не входящие в число переменных из α .

Например, если $\alpha = \{f(x)|z\}$ и $\beta = \{a|x, b|y, d|z\}$, то композиция

$$\alpha\beta = \{f(a)|z, b|y\}$$

Если подстановка α применяется к каждому элементу множества $\{L_i\}$ литералов, то множество соответствующих ей частных случаев обозначается через $\{L_i\}$.

ОПРЕДЕЛЕНИЕ 3

Множество $\{L_i\}$ литералов называется, **унифицируемым**, если существует такая подстановка q , что

$$L_{1\theta} = L_{2\theta} = L_{3\theta} = \dots$$

В этом случае подстановку q называют унификатором для $\{L_i\}$, поскольку ее применение сжимает множество до одного элемента. Например, подстановка $q = \{a/x, b/y\}$ унифицирует множество

$$\{P(x, f(y), b), P(x, f(b), b)\} \text{ и дает } \{P(a, f(b), b)\}.$$

Для упрощения решения задачи доказательства теоремы надо стремиться к тому, чтобы найти такую подстановку (унификатор), которая при ее применении позволяет сделать несколько (чем больше тем лучше) дизъюнктов идентичными. Поэтому процедуры унификации системы дизъюнктов является основной в формальных преобразованиях, выполняемых при нахождении резольвент.

ОПРЕДЕЛЕНИЕ 4

Унификатор λ для множества литералов $\{Li\}$ называется **простейшим** (или **наиболее общим**), если каким бы ни был унификатор для $\{Li\}$, найдется такая подстановка s , что $\{Li\}\lambda s = \{Li\}$.

Отметим, что «общий» частный случай, соответствующий наиболее общему унификатору (НОУ), единственен с точностью до алфавитных вариантов.

В этом определении сказано "каким бы ни был унификатор q ", т.е. имеется в виду некоторое множество унификаторов. Это означает, что реализация процедуры унификации связана с проверкой большого числа подстановок. Поэтому ручная унификация для более или менее сложных примеров связана с большими затратами времени.

Существует множество алгоритмов поиска НОУ.

В основе машинного алгоритма определения НОУ лежит идея устранения рассогласования пар дизъюнктов.

ОПРЕДЕЛЕНИЕ 5

Два дизъюнкта называются **рассогласованными**, если в одинаковых позициях у них стоят различные символы.

Например дизъюнкты $P(x)$ и $P(a)$ рассогласованны т.к. в первом из них имеется переменная x , а во втором на этом месте находится константа a . Такое рассогласование обозначается как $\{x,a\}$.

Пример.

Дано множество дизъюнктов
 $\{P(x, f(y)), P(x, a), P(x, g(z))\}$

В этих дизъюнктах первые четыре символа ('P' '(' 'x' ',') совпадают. Первый слева символ, не встречающихся в других дизъюнктах, находится в пятой слева позиции первого дизъюнкта. Это - функциональный символ $f(y)$. В двух других дизъюнктах символы, находятся в пятой позиции, - это символы a и $g(z)$. Следовательно, множество рассогласований имеет вид

$\{f(y), a, g(z)\}$.

Из этого примера следует, что алгоритм унификации должен обеспечить просмотр символов множества дизъюнктов на каждой позиции, фиксируя рассогласования между ними и осуществляя операцию подстановки с целью ликвидации этого рассогласования.

АЛГОРИТМ

Введем следующие обозначения:

S -некоторое множество дизъюнктов

S_k - некоторый пример множества S (получаемый после некоторой подстановки на k -ом шаге работы алгоритма)

b -унификатор

ε -символ чисто служебного характера

v -переменная

t -пропорциональная переменная, отличная от v и подставляемая вместо нее (подстановка t_k/v_k)

D -множество рассогласований для S

Алгоритм поиска НОУ:

1. Положить $k=0$, $S_k=S$, $b_k=\varepsilon$ (пустая подстановка). Перейти к шагу 2.
2. Если S_k единственно (т. е. S_k - одноэлементное множество), то остановка (алгоритм реализован), b_k = НОУ для S . Если S_k не единственно, то найти множество рассогласований для S_k . Перейти к следующему шагу.
3. Если в множестве рассогласований D_k существует элементы v_k и t_k такие, что v_k - это переменная, не встречается в t_k , то перейти к шагу 4. В противном случае остановка: множество не унифицируемо.
4. Пусть $b_{(k+1)}$ равно композиции постановок b_k и t_k/v_k , а $S_{(k+1)}=S_k\{t_k/v_k\}$
5. Положить $k=k+1$. Перейти к шагу 2.

В результате работы этого алгоритма шаг за шагом анализируется каждая позиция множества S дизъюнктов, начиная с крайне левого символа первого дизъюнкта и находится множество рассогласований D_0, D_1, D_2, \dots

ПРИМЕР 1

Найти НОУ для $S = \{Q(x,y), Q(a,c)\}$. Алгоритм поиска:

1. $d_0 = e$, $d_0 = S$. Поскольку S не единственно, то d_0 не является НОУ для S .
2. Определяем множество Рассогласований на данном шаге $D_0 = \{x, a\}$ - рассогласование на 3-ей позиции.
3. В D_0 имеется переменная $v_0 = x$, которая не встречается в терме $t_0 = a$.
4. Пусть $d_1 = d_0 \{t_0 | v_0\} = e \{a|x\} = \{a|x\}$ $S_1 = S_0 \{t_0 | v_0\} = \{Q(x,y), Q(a,c)\} \{a|x\} = \{Q(a,y), Q(a,c)\}$
5. S_1 не единственно. Находим множество рассогласований D_1 для S_1 :
 $D_1 = \{y, c\}$ - на 5-й позиции
6. из D_1 система $v_1 = y$, $t_1 = c$
7. Пусть $d_2 = d_1 \{t_1 | v_1\}$; $S_2 = S_1 \{t_1 | v_1\}$ $d_2 = \{a|x\} \{c|y\} = \{a|x, c|y\}$ $S_2 = \{Q(a, y), Q(a, c)\} \{c|y\} = \{Q(a, c), Q(a, c)\} = (Q(a, c))$
8. S_2 - единственно. НОУ для S есть $d_2 = \{a|x, c|y\}$

ЗАДАНИЕ

1. $\forall_x (T(x) \wedge \neg Q(x) \rightarrow \exists_y (P(y) \wedge C(x,y)))$

2. $\exists_x (D(x) \wedge T(x) \wedge \forall_y (C(x,y) \rightarrow D(y)))$

3. $\forall_x (D(x) \rightarrow \neg QP(x))$

T. $\exists_x (P(x) \wedge D(x))$

МЕТОД РЕЗОЛЮЦИИ В ИВ

Назовем два дизъюнкта, содержащие **одинаковые** ПП, но с противоположными знаками, *контрарными парами (атомами)*.

Алгоритм вывода по методу резолюции (при имеющемся заключении):

Шаг 1: принять отрицание заключения, т.е. $\neg\Phi$,

Шаг 2: привести все формулы посылок и отрицания заключения в КНФ,

Шаг 3: сформировать множество K дизъюнктов всех посылок и отрицания заключения: $K = \{D_1, D_2, \dots, D_k\}$,

Шаг 4: выполнить анализ пар множества K по правилу: если существует пара элементов, содержащих контрарные атомы, то соединить эту пару логической связкой дизъюнкции и сформировать новый дизъюнкт - *резольвенту*, исключив из нее контрарные атомы,

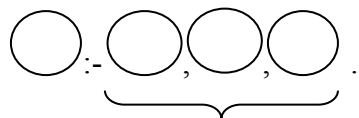
Шаг 5: если в результате соединения дизъюнктов будет получена пустая резольвента – аналог ложности формулы (обозначается \bullet), то конец, т.к. доказательство подтвердило истинность заключения. Иначе включить резольвенту в множество дизъюнктов K и перейти к шагу 4. При этом по закону идемпотентности любой дизъюнкт и любую резольвенту можно использовать неоднократно, т.е. из множества K *не следует* удалять использованные в соединении дизъюнкты.

ПРИМЕР

$$\frac{\forall x(P_3(x) \& \neg P_1(x) \rightarrow \exists y(P_5(y) \& P_4(x,y))), \exists x(P_2(x) \& P_3(x) \& \forall y(P_4(x,y) \rightarrow P_2(y))), \forall x(P_2(x) \rightarrow \neg P_1(x))}{\exists x(P_5(x) \& P_2(x))}$$

ЛОГИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

Предложения Пролога состоят из головы и тела



Тело – список целей, разделенных запятыми. Запятая соответствует конъюнкции.

Факты – это предложения с пустым телом. Запрос имеет только тело. Отношения могут определяться фактами, перечисляющие n -объектов, для которых это отношение верно или оно может определяться правилом.

Аргументами отношений могут являться конкретные объекты (атомы) или абстрактные объекты (переменные). По ходу вычислений вместо переменной может быть поставлен конкретный объект. Этот процесс называют конкретизацией или унификацией переменной.

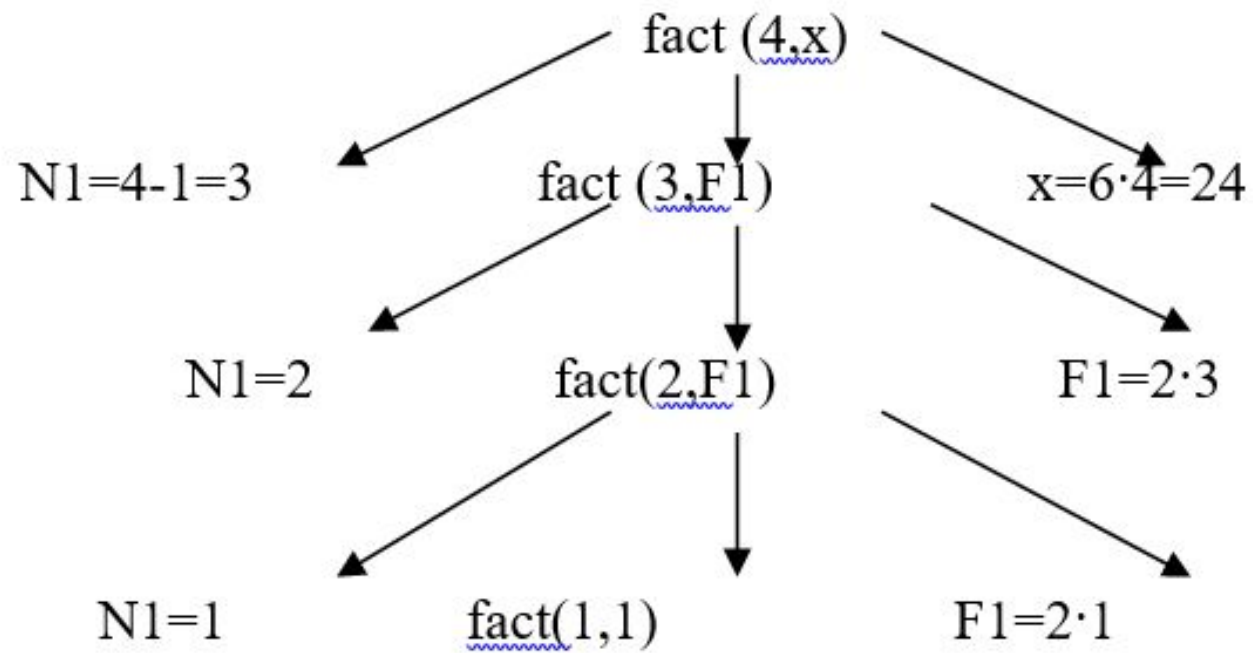
Множество предложений об одном и том же отношении называется **процедурой**. Такие предложения необходимо записывать рядом. Каждая процедура допускает процедурную и логическую интерпретацию.

Пример:

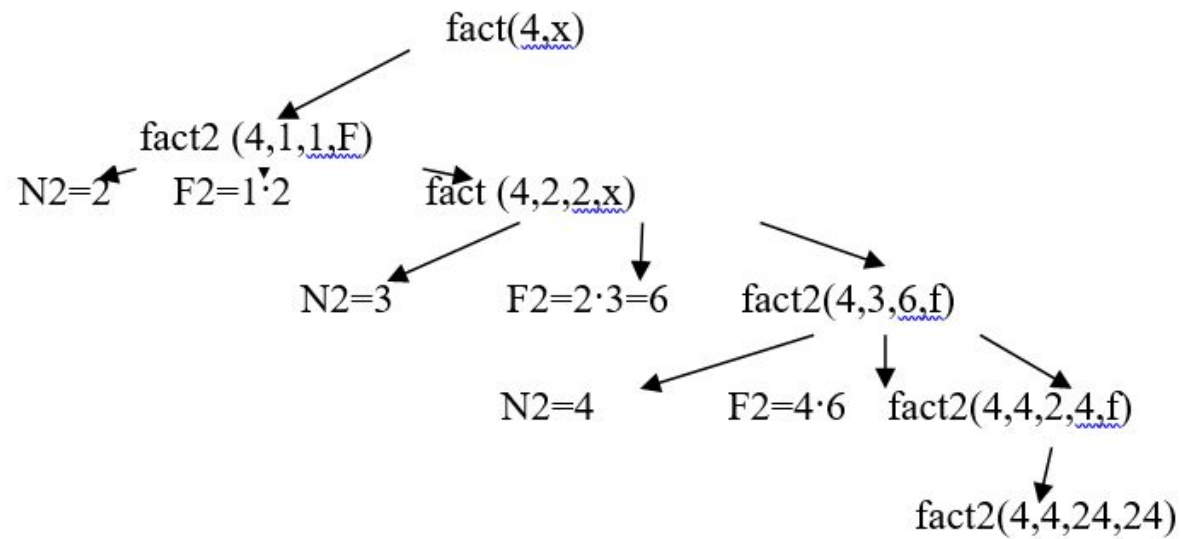
A: - A1, A2, A3

Логическая интерпретация: A истина, когда истина A1, A2, A3.

НИСХОДЯЩАЯ РЕКУРСИЯ



ВОСХОДЯЩАЯ РЕКУРСИЯ





4. НЕЧЕТКАЯ ЛОГИКА



НЕЧЕТКАЯ ЛОГИКА

- Нечеткая логика основана на использовании оборотов естественного языка - «далеко», «близко», «холодно», «горячо».
- Диапазон ее применения - от бытовых приборов до управления сложными промышленными процессами.
- Многие задачи управления просто не могут быть решены классическими методами из-за очень большой сложности математических моделей.

НЕЧЕТКАЯ ЛОГИКА

- *Нечетким множеством* на множестве X назовем пару (x, μ_A) ,
- где $\mu_A(x)$ – функция, каждое значение которой $\mu_A(x) \in [0, 1]$ -степень принадлежности точки $x \in X$ множеству. X – универсальное множество.
- Функция μ_A – называется *функцией принадлежности множества*.

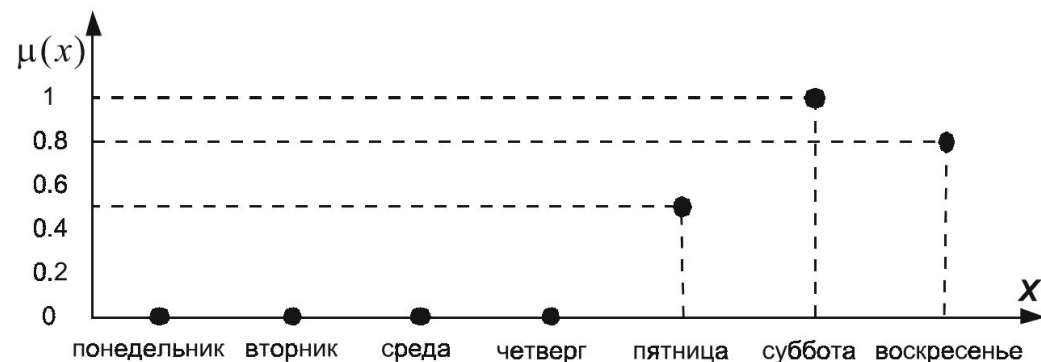


ПРИМЕР

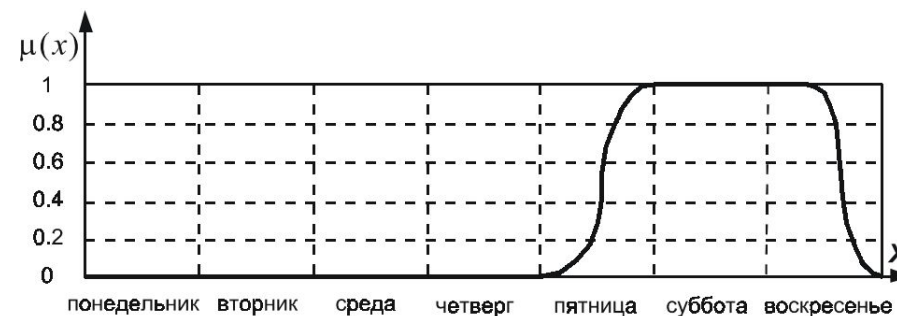
Построить нечеткое множество, которое содержательно описывало бы выходные дни недели.

ГРАФИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ

нечеткого множества A , описывающего выходные дни недели, в форме значений функции принадлежности этого нечеткого множества



нечеткого множества A , описывающего выходные дни недели, в форме кривой его функции принадлежности



ОСНОВНЫЕ ХАРАКТЕРИСТИКИ НЕЧЕТКИХ МНОЖЕСТВ

- **Множество a - уровня** - это обобщение понятия носителя нечеткого множества. Это обычное множество A_a , удовлетворяющее условию $A_a = \{x \in X \mid \mu_A(x) \geq a\}$, где a - некоторое действительное число из интервала $[0,1]$
- Величина $h_A = \sup\{\mu_A(x)\}$, где супремум берется по всем значениям функции принадлежности для $x \in X$, называется **высотой нечеткого множества**.
- **Нечеткое множество называется нормальным**, если максимальное значение его функции принадлежности равно 1 и при этом существуют элементы множества, функция принадлежности которых равна 1.
- **Нечеткое множество называется субнормальным**, максимальное значение его функции принадлежности меньше 1.
- **Ядром нечеткого множества A** называют такое обычное множество A_1 , элементы которого удовлетворяют условию $A_1 = \{x \in X \mid \mu_A(x) = 1\}$.

ОСНОВНЫЕ ХАРАКТЕРИСТИКИ НЕЧЕТКИХ МНОЖЕСТВ

- **Нечеткое множество называется унимодальным** (строго унимодальным), если его функция принадлежности $\mu_A(x)$ является унимодальной (строго унимодальной). Функция принадлежности $\mu_A(x)$ называется унимодальной (строго унимодальной) на интервале $[a,b] \subset \mathbb{R}$, если непрерывна на $[a,b]$, а также существует непустой $[c,d] \subset [a,b]$, такой что $a \leq c \leq d \leq b$ и выполняются условия:
 - функция $\mu_A(x)$ строго монотонно возрастает на интервале $[a,c]$ при $a < c$
 - функция $\mu_A(x)$ строго монотонно убывает на интервале $[d,b]$ при $d < b$
 - функция $\mu_A(x)$ принимает свое максимальное значение на интервале $[c,d]$, т.е. любая точка $x_m \in [c,d]$ является точкой максимума функции принадлежности относительно интервала $[a,b]$:

$$x_m = \arg \max \{ \mu(x) \}, x \in [a,b]. \quad (1)$$

Любая точка $x_m \in A$ нечеткого множества A , удовлетворяющая условию (1), называется **модальным значением** или **модой** нечеткого множества A .

ТИПЫ ФУНКЦИЙ ПРИНАДЛЕЖНОСТИ



Z - функция

П - функция

Л - функция

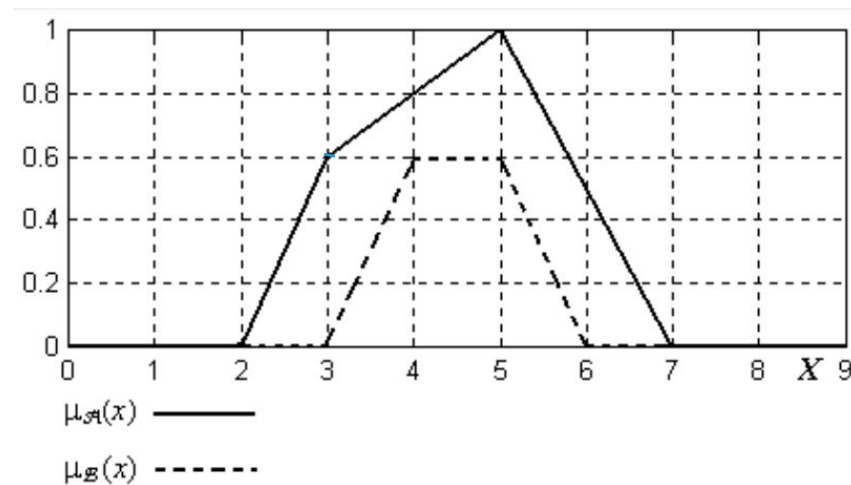
S - функция

ОСНОВНЫЕ ОПЕРАЦИИ С НЕЧЕТКИМИ МНОЖЕСТВАМИ

1. Включение (Доминирование)

$$\mu_A(x) \leq \mu_B(x)$$

$$A \subseteq B$$

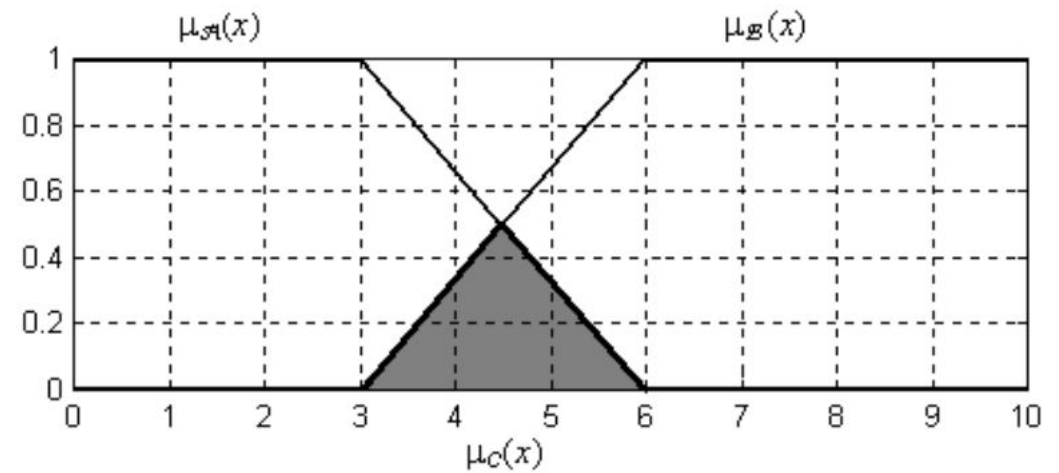


ОСНОВНЫЕ ОПЕРАЦИИ С НЕЧЕТКИМИ МНОЖЕСТВАМИ

2. Пересечение

$$\mu_C(x) = \min \{ \mu_A(x), \mu_B(x) \}$$

$$C = A \cap B$$

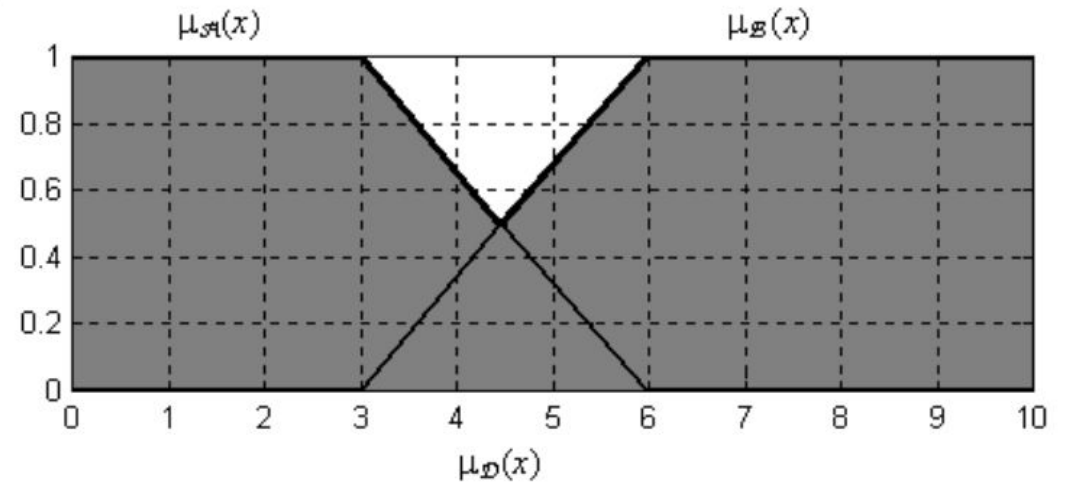


ОСНОВНЫЕ ОПЕРАЦИИ С НЕЧЕТКИМИ МНОЖЕСТВАМИ

3. Объединение

$$\mu_D(x) = \max\{\mu_A(x), \mu_B(x)\}$$

$$D = A \cup B$$

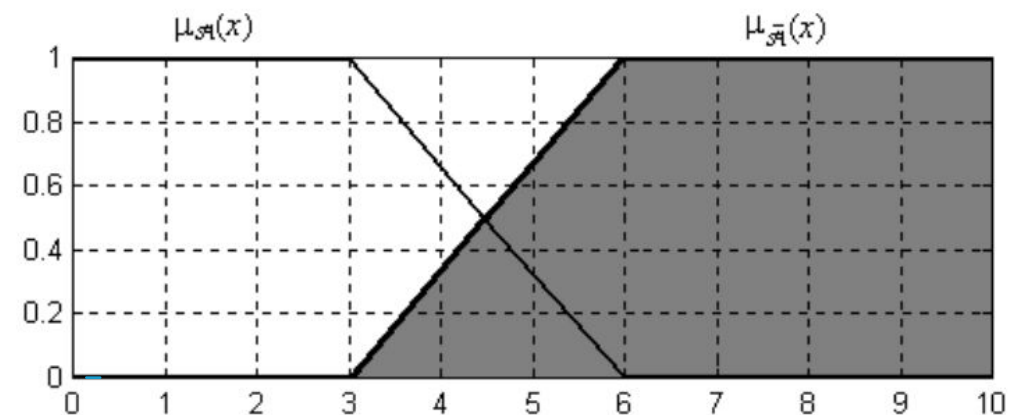


ОСНОВНЫЕ ОПЕРАЦИИ С НЕЧЕТКИМИ МНОЖЕСТВАМИ

4. Дополнение

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x)$$

\bar{A}

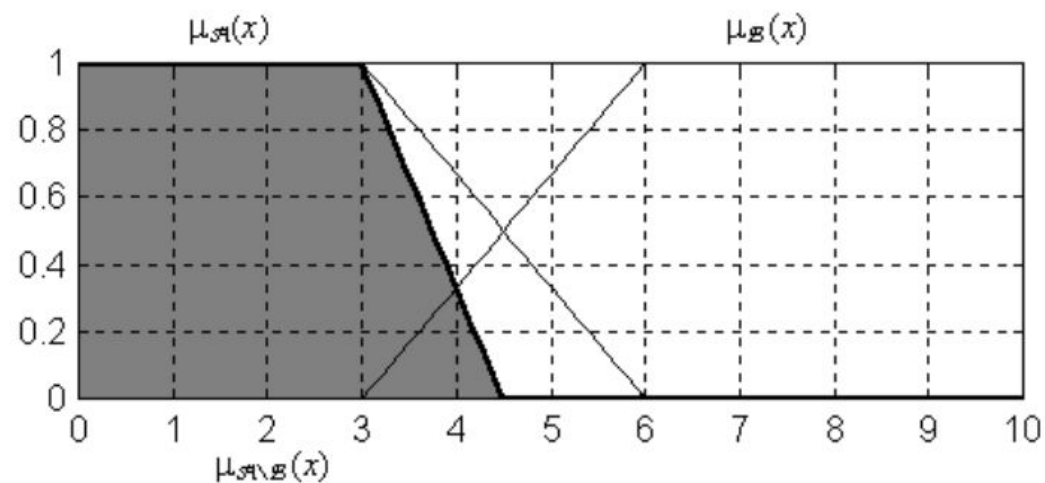


ОСНОВНЫЕ ОПЕРАЦИИ С НЕЧЕТКИМИ МНОЖЕСТВАМИ

5. Разность

$$\mu_{A \setminus B}(x) = \min \{ \mu_A(x), 1 - \mu_B(x) \}$$

$$A \setminus B = A \cap \neg B$$



НЕЧЕТКАЯ ПЕРЕМЕННАЯ. ЛИНГВИСТИЧЕСКАЯ ПЕРЕМЕННАЯ

- **Нечеткая переменная** определяется кортежем параметров $\langle \alpha, X, A \rangle$, где α - название нечеткой переменной, X – область определения нечеткой переменной, $A = \{x, \mu_A(x)\}$, $x \in X$ - заданное на X нечеткое множество, описывающее возможные значения нечеткой переменной.
- **Лингвистическая переменная** является обобщением понятия нечеткой переменной и определяется кортежем параметров $\langle \beta, T, X, G, M \rangle$

β - название лингвистической переменной;

T – базовое *терм-множество* лингвистической переменной, состоящее из множества ее значений (*термов*), каждое из которых представляет собой название отдельной нечеткой переменной α ;

X – область определения нечетких переменных, названия которых составляют терм-множество лингвистической переменной;

G – синтаксическая процедура, позволяющая оперировать элементами терм-множества и генерировать новые термы;

M – семантическая процедура, позволяющая преобразовывать значения лингвистических переменных, полученных процедурой G , в нечеткие переменные, путем формирования соответствующих нечетких множеств.

ЭТАПЫ НЕЧЕТКОГО ВЫВОДА

- 1 Формирование базы правил
- 2 Фаззификация входных переменных
- 3 Агрегирование подусловий
- 4 Активизация подзаключений
- 5 Аккумуляирование заключений

ФОРМИРОВАНИЕ БАЗЫ ПРАВИЛ

ПРАВИЛО_1: ЕСЛИ «Условие_1» то «Заключение_1» (F1)

ПРАВИЛО_2: ЕСЛИ «Условие_2» то «Заключение_2» (F2)

...

ПРАВИЛО_n: ЕСЛИ «Условие_n» то «Заключение_n» (Fn)

или

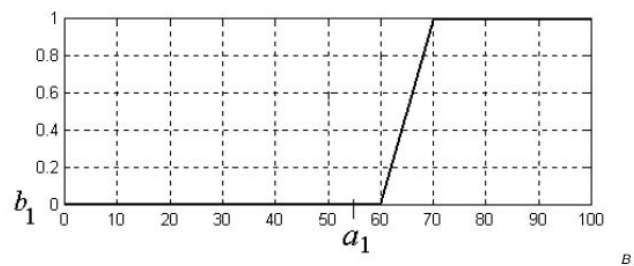
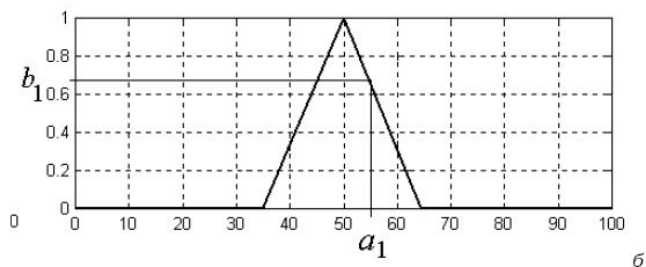
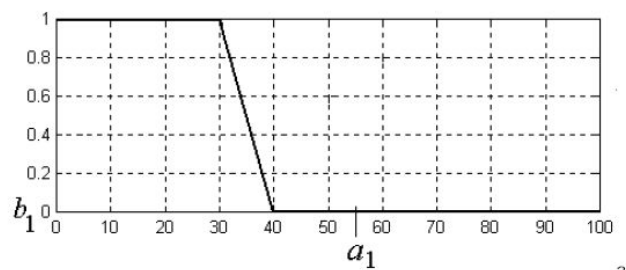
RULE_1: IF «Condition_1» THEN «Conclusion_1» (F1)

RULE_2: IF «Condition_2» THEN «Conclusion_2» (F2)

...

RULE_n: IF «Condition_n» THEN «Conclusion_n» (Fn)

ФАЗЗИФИКАЦИЯ



Пример

Фаззификация высказываний:

- Скорость автомобиля малая
- Скорость автомобиля средняя
- Скорость автомобиля высокая

Текущая скорость автомобиля $a_1=55$

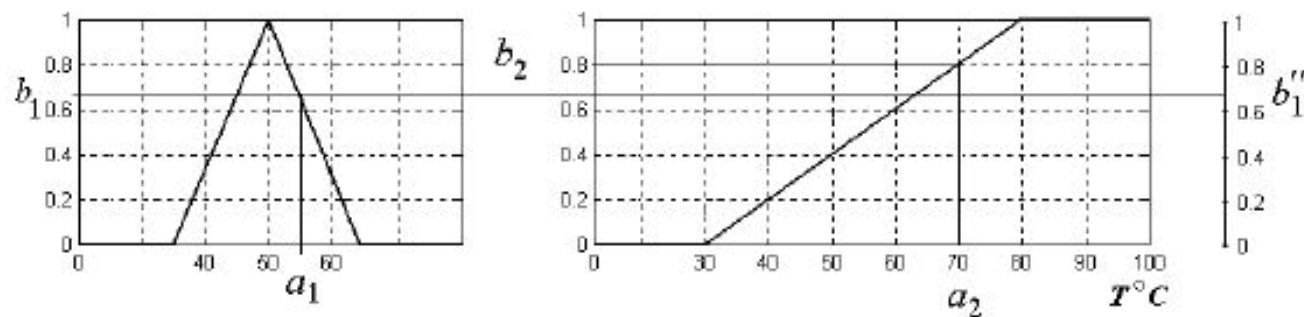
АГРЕГИРОВАНИЕ

Пример

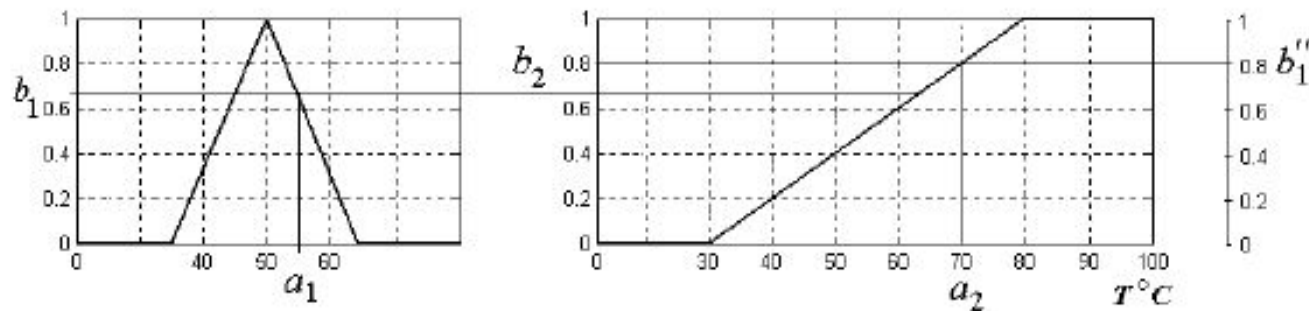
Агрегирование двух высказываний:

- «скорость автомобиля средняя» И «кофе горячий»
- «скорость автомобиля средняя» ИЛИ «кофе горячий»

Текущая скорость автомобиля $a_1=55$, а температура кофе $a_2=70$



а



б

АКТИВИЗАЦИЯ

ПРАВИЛО_1: ЕСЛИ Условие_1 ТО Подзаключение_1.1 И/ИЛИ Подзаключение_1.2 (F1)

min-активизация

$$\mu'(y) = \min\{c_i, \mu(y)\}$$

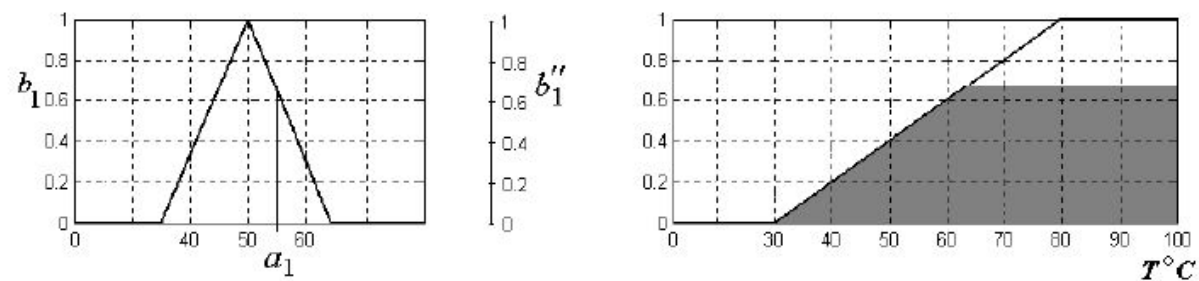
prod-активизация

$$\mu'(y) = c_i \cdot \mu(y)$$

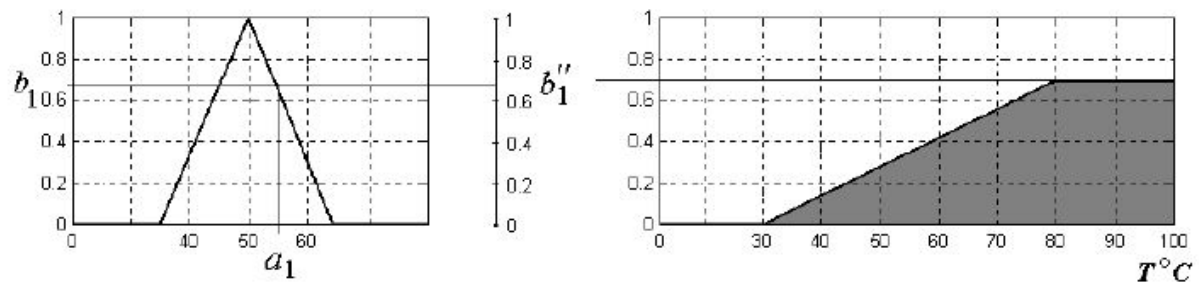
average-активизация

$$\mu'(y) = 0.5 \cdot (c_i + \mu(y))$$

АКТИВИЗАЦИЯ



а



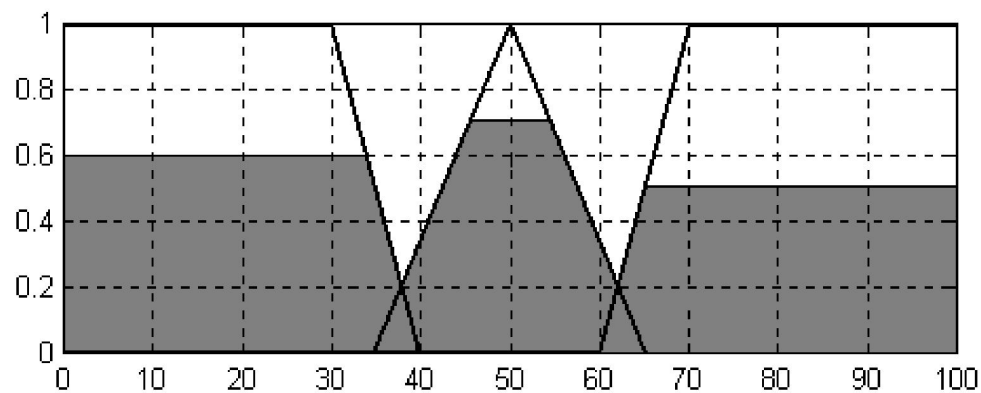
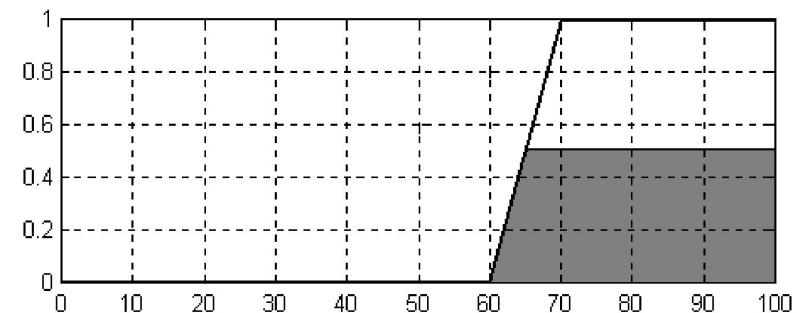
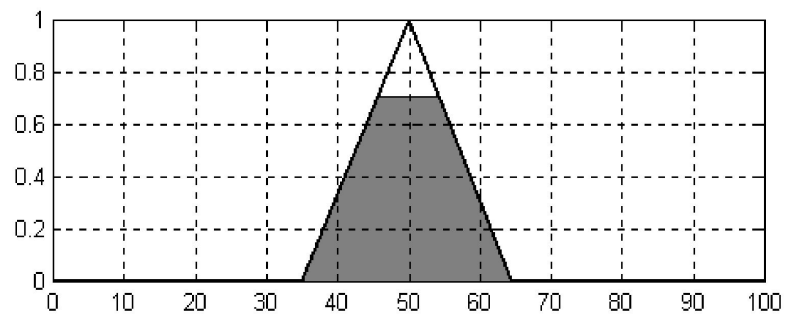
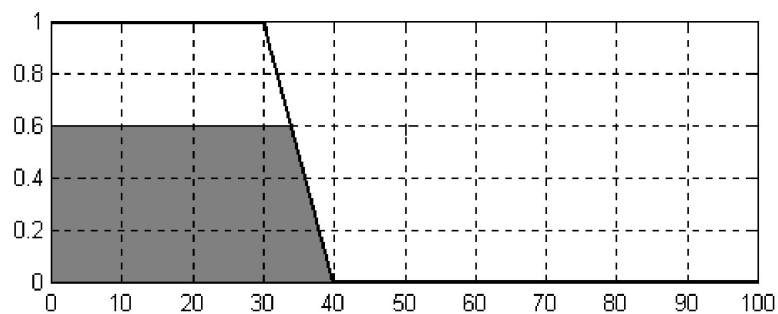
б

Пример

Активизация заключений в правиле нечеткой продукции:
- ЕСЛИ «скорость автомобиля средняя» ТО «кофе горячий»

Текущая скорость автомобиля $a_1=55$

АККУМУЛЯЦИЯ



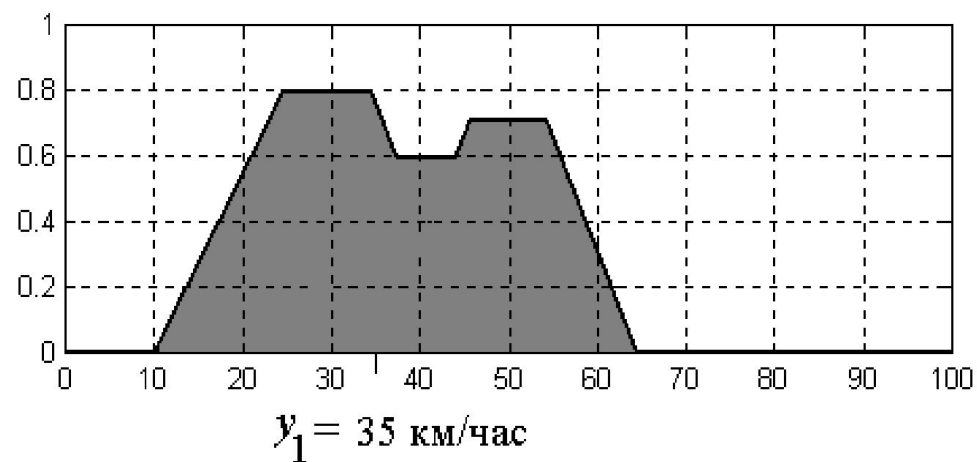
Пример

Аккумуляция заключений нечетких множеств, полученных в результате активизации входной лингвистической переменной «скорость движения автомобиля» в некоторой системе нечеткого вывода

ДЕФАЗЗИФИКАЦИЯ

Метод центра площади

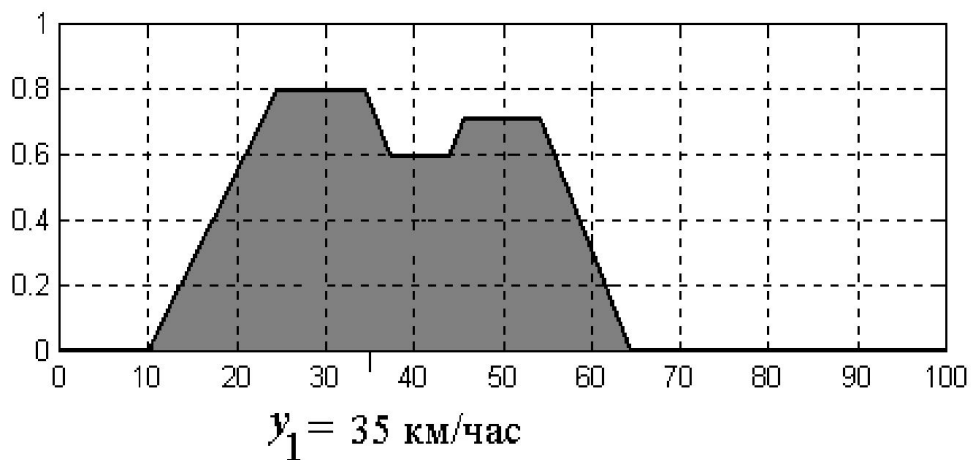
$$\int_{Min}^u \mu(x) dx = \int_u^{Max} \mu(x) dx$$



ДЕФАЗЗИФИКАЦИЯ

Метод центра площади

$$\int_{Min}^u \mu(x) dx = \int_u^{Max} \mu(x) dx$$



АЛГОРИТМ МАМДАНИ

Формирование базы правил систем нечеткого вывода.

Фаззификация входных

переменных.

Агрегирование подусловий в нечетких правилах продукций.

Для нахождения степени истинности условий каждого из правил нечетких продукций используются парные нечеткие логические операции. Те правила, степень истинности условий которых отлична от нуля, считаются активными и используются для дальнейших расчетов.

Активизация подзаключений в нечетких правилах продукций.

Осуществляется *min*-активизация. Для сокращения времени вывода учитываются только активные правила.

Аккумуляция заключений нечетких правил продукций.

Используется *max*-объединение.

Дефаззификация выходных

переменных.

Используется метод центра тяжести или метод центра площади.

АЛГОРИТМ ЦУКАМОТО

Формирование базы правил систем нечеткого вывода.

Фаззификация входных

переменных.

Агрегирование подусловий в нечетких правилах продукций.

Для нахождения степени истинности условий каждого из правил нечетких продукций используются парные нечеткие логические операции. Те правила, степень истинности условий которых отлична от нуля, считаются активными и используются для дальнейших расчетов.

Активизация подзаключений в нечетких правилах продукций.

Осуществляется min-активизация. Затем находятся нечеткие значения всех выходных лингвистических переменных в каждом из подзаключений активных правил нечетких продукций.

Аккумуляция заключений нечетких правил продукций.

Отсутствует, поскольку расчеты осуществляются с обычными числами.

Дефаззификация выходных

переменных.

Метод центра тяжести для одноточечных множеств.

АЛГОРИТМ ЛАРСЕНА

Формирование базы правил систем нечеткого вывода.

Фаззификация входных

переменных.

Агрегирование подусловий в нечетких правилах продукций.

Для нахождения степени истинности условий каждого из правил нечетких продукций используются парные нечеткие логические операции. Те правила, степень истинности условий которых отлична от нуля, считаются активными и используются для дальнейших расчетов.

Активизация подзаключений в нечетких правилах продукций.

Осуществляется *prod*-активизация. Для сокращения времени вывода учитываются только активные правила.

Аккумуляция заключений нечетких правил продукций.

Используется *max*-объединение.

Дефаззификация выходных

переменных.

Осуществляется любым методом из рассмотренных.

АЛГОРИТМ СУГЕНО

Формирование базы правил систем нечеткого вывода.

ПРАВИЛО <#>: ЕСЛИ " β_1 есть α " И " β_2 есть α " ТО " $w = \varepsilon_1 \cdot a_1 + \varepsilon_2 \cdot a_2$ "

Фаззификация входных

переменных.

Агрегирование подусловий в нечетких правилах продукций.

Для нахождения степени истинности условий всех правил, как правило используется min-конъюнкция. Те правила, степень истинности условий которых отлична от нуля, считаются активными и используются для дальнейших расчетов.

Активизация подзаключений в нечетких правилах продукций.

Осуществляется min-активизация. Затем осуществляется расчет нечетких значений выходных переменных каждого правила.

Аккумуляция заключений нечетких правил продукций.

Отсутствует.

Дефаззификация выходных переменных.

Используется метод центра тяжести для одноточечных множеств.

УПРОЩЕННЫЙ АЛГОРИТМ

Формирование базы правил систем нечеткого вывода.

ПРАВИЛО <#>: ЕСЛИ " β_1 есть α " И " β_2 есть α " ТО " $w=\varepsilon$ "

Фаззификация входных

переменных.

Агрегирование подусловий в нечетких правилах продукций.

Для нахождения степени истинности условий всех правил, как правило используется \min -конъюнкция. Те правила, степень истинности условий которых отлична от нуля, считаются активными и используются для дальнейших расчетов.

Активизация подзаключений в нечетких правилах продукций.

Осуществляется \min -активизация. Для сокращения времени вывода учитываются только активные правила.

Аккумуляция заключений нечетких правил продукций.

Используется \max -объединение.

Дефаззификация выходных переменных.

Используется метод центра тяжести для одноточечных множеств.

НЕЧЕТКИЕ ВЫСКАЗЫВАНИЯ

ПРАВИЛО_1: ЕСЛИ «Гражданин не является высокопоставленным чиновником», ТО «он подвергается таможенному досмотру» (F1=1.0)

ПРАВИЛО_2: ЕСЛИ «Гражданин является высокопоставленным чиновником», ТО «он не подвергается таможенному досмотру» (F2=0.9)

ПРАВИЛО_3: ЕСЛИ «Гражданин не подвергается таможенному досмотру», ТО «не исключается возможность провоза наркотиков» (F3=0.8)

ПРАВИЛО_4: ЕСЛИ «Количество граждан, проходящих таможенный осмотр, велико», ТО «контролер испытывает чувство усталости» (F4=0.6)

ПРАВИЛО_5: ЕСЛИ «Контролер испытывает чувство усталости», ТО «не исключается возможность провоза наркотиков» (F5=0.7)

ПРАВИЛО_6: ЕСЛИ «Гражданин подвергается таможенному досмотру» И «в отношении этого гражданина имеется агентурная информация», ТО «исключается возможность провоза наркотиков» (F6=0.95)

ПРАВИЛО_7: ЕСЛИ «Гражданин подвергается таможенному досмотру» И «контролер использует новейшие технические средства», ТО «исключается возможность провоза наркотиков» (F7=0.95)



ЛИТЕРАТУРА

1. **А. Леоненков.** Нечеткое моделирование MATLAB и fuzzyTECH
2. **Ф. Ланге.** Нечеткая логика
3. **Ф. Шеври, Ф. Гели.** Нечеткая логика



СПАСИБО ЗА ВНИМАНИЕ!