

DOI:
**ОПРЕДЕЛЕНИЕ И ИСПОЛЬЗОВАНИЕ МНОГОЗНАЧНЫХ НЕЙРОННЫХ СЕТЕЙ
ПРИ ПРИНЯТИИ РЕШЕНИЯ ПО УПРАВЛЕНИЮ ГРУППОЙ БЕСПИЛОТНЫХ
ЛЕТАТЕЛЬНЫХ АППАРАТОВ**

Максимов Д.Ю.

Институт проблем управления им. В.А. Трапезникова РАН, Россия, г. Москва

ул. Профсоюзная д.65

jhanjaa@ipu.ru

Аннотация: Для принятия управляющего решения в группе БПЛА используется многозначная ассоциативная память. Это новое понятие, обобщающее аналогичное в нечетких нейронных сетях. В многозначной нейронной сети входные данные – элементы или подмножества решетки лингвистических переменных. Предполагается, что решетка дистрибутивна и определяет импликацию, используемую для определения выхода нейросети.

Ключевые слова: многозначные нейронные сети, нечеткие нейронные сети, ассоциативная память, беспилотные летательные аппараты.

Работа выполнена по Программе Президиума РАН №7(30) и соответствует области исследований, поддержанных грантами РФФИ 19-07-01039 и 16-37-50030

Введение

Для распознавания складывающихся в полете типовых ситуаций и получения соответствующих управляющих решений в [1] используется нечеткая логика второго уровня (т.н. нечеткозначная логика). В таком случае, складывающаяся ситуация оценивается набором лингвистических переменных вместе со степенями значимости/уверенности значений этих переменных. Здесь степени уверенности принимают, в свою очередь, значения в числовом интервале, обычно $[0,1]$. Однако, способы определения нечеткости субъективны – степень нечеткости определяется экспертами. В [2] продемонстрировано, что не обязательно нужно использовать числа: в качестве оценки опять могут выступать лингвистические значения (но уже частично-упорядоченные), которые не требуют обязательной численной оценки. Такие оценки определяются самой ситуацией и не нуждаются в экспертном мнении (в первом приближении). В этом случае, для их сравнения и получения управляющих решений в [2] используются понятия не нечеткой логики, а многозначной, в которой шкала истинностных значений является дистрибутивной решеткой общего вида. Такие шкалы истинностных значений обобщают линейно-упорядоченную шкалу (в частности, нечеткой логики) [3] и естественным образом определяют импликацию как решеточную операцию.

При таком подходе легко выбирать управляющие решения в случае простой решетки истинностных значений (лингвистических значений степеней уверенности/нечеткости). Но в случае сложной решетки вычисление импликаций становится трудоемкой задачей. В этой работе предлагается для вычисления импликаций и для быстрого получения управляющего решения использовать многозначные нейронные сети. Такие сети обобщают нечеткие нейронные сети [4] на случай использования операций многозначной логики вместо нечеткой.

Работа организована следующим образом: в разделе 1 вводится понятие многозначной нейронной сети, исследуются свойства таких сетей, приводится обобщение обучающего алгоритма нечетких нейросетей на многозначный случай. Показано, что, как и в случае нечетких нейросетей, существуют ограничения для входных переменных – не всякий паттерн может быть сохранен в многозначной ассоциативной памяти. Поэтому предлагаются разные варианты для определения выхода по входам в зависимости от требований задачи, что не рассматривалось в теории нечетких нейросетей. В разделе 2 рассматриваются способы получения управляющих решений с помощью многозначных нейронных сетей в типовой ситуации обеспечения безопасности группового полета БПЛА по взаимному расположению объектов и при внешней помехе. Входными данными являются лингвистические переменные (частично-упорядоченные), а не числа, что, тем не менее, позволяет принять адекватное управляющее решение. В частности, этот результат предоставляет необходимый вычислительный аппарат для способа принятия решения на основе обобщенной нечеткой логики, разработанного ранее [2].

1 Многочисленные нейронные сети

Нечеткие нейронные сети прямого распространения, в которых внутренние операции основаны на нечетких операциях объединения и пересечения « $\wedge - \vee$ », были предложены в [5]. Такие сети называются *нечеткой ассоциативной памятью*, и нечеткая информация в них представляется элементами отрезка $[0, 1]$. В *многочисленной ассоциативной памяти* вместо элементов отрезка $[0, 1]$ будем использовать элементы полной решетки:

Решетка – это частично-упорядоченное множество, имеющее для любых двух элементов их точную верхнюю грань или объединение \vee (sup, max) и точную нижнюю грань или пересечение \wedge (inf, min). *Точная верхняя грань* двух элементов — это наименьший элемент решетки, содержащий оба эти элемента. Двойственно определяется *точная нижняя грань*. *Полная решетка* – это решетка, в которой любые два подмножества имеют объединение и пересечение. Это означает, что в непустой полной решетке есть наибольший «1» и наименьший «0» элементы. *Образующими* решетки называются такие ее элементы, из которых все остальные получаются путем объединения и пересечения.

Если взять такую решетку в качестве шкалы истинностных значений в многочисленной логике, то наибольшему элементу будет соответствовать полная истинность (*true*), наименьшему – полная ложность (*false*), а промежуточные элементы будут соответствовать частичной истинности так же, как в нечеткой логике частичную истинность оценивают элементы отрезка $[0, 1]$.

В логиках с такой шкалой истинностных значений импликацию можно определить через умножение элементов решетки (резидуальные логики), либо внутренним образом, только из решеточных операций. Решетка, имеющая внутренние импликации, называется *брауэровой* решеткой. В такой решетке *импликация* $c = a \Rightarrow b$ определяется как наибольший c : $a \wedge b = a \wedge c$. Импликация $\neg a = a \Rightarrow 0$ называется *псевдодополнением* a . В брауэровых решетках выполняются законы дистрибутивности для объединения и пересечения. Обратное верно только для конечных решеток.

Мы будем предполагать используемые в многочисленной нейронной сети решетки полными, дистрибутивными и конечными.

1.1 Двухслойная max-min многочисленная ассоциативная память

Как и в случае нечетких нейросетей, в max-min многочисленной ассоциативной памяти не рассматривается передаточная функция, поскольку операторы « $\vee - \wedge$ » ограничивают выходы к диапазону входов, а \wedge также является обрезающим оператором [4]. Предположим, входной сигнал $\mathbf{x} \in L^n$, а выходной $\mathbf{y} \in L^m$, где L – используемая решетка. В таком случае отношение «вход – выход» в двухслойной многочисленной ассоциативной памяти может быть записано как $\mathbf{y} = \mathbf{x} \circ \mathbf{W}$, где \circ обозначает « $\vee - \wedge$ » операцию композиции, а $\mathbf{W} = (w_{ij})_{n \times m} \in \mu_{n \times m}$ – $n \times m$ матрица весов связей с элементами из L (Рис. 1).

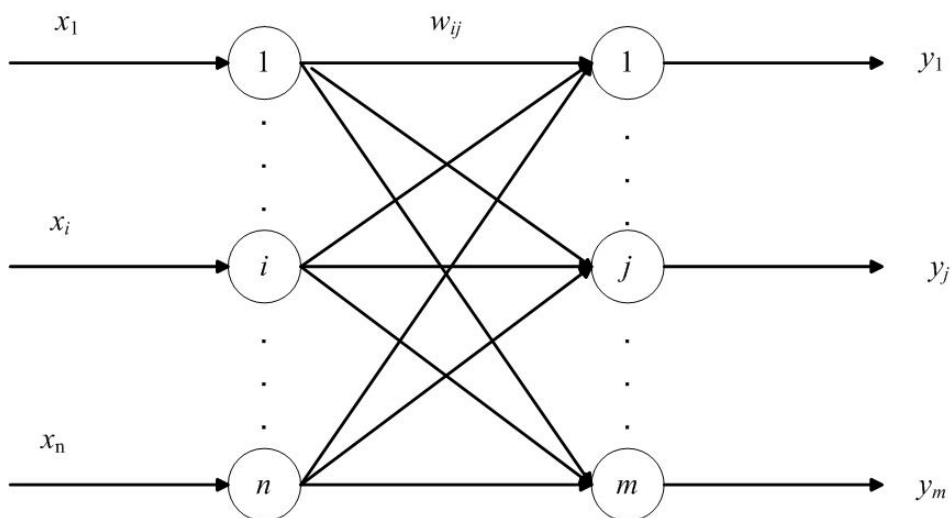


Рис. 1. Топология двухслойной ассоциативной памяти

Можно предложить несколько простейших вариантов композиции:

$$(1) \quad y_j = \vee_i \{x_i \wedge w_{ij}\};$$

$$(2) \quad y_j = \bigvee_i \{x_i \Rightarrow w_{ij}\};$$

$$(3) \quad y_j = \bigwedge_i \{x_i \Rightarrow w_{ij}\};$$

$$(4) \quad y_j = \bigwedge_i \{x_i \vee w_{ij}\}.$$

В теории простейшей нечеткой ассоциативной памяти рассматривается только вариант (1). В более сложных моделях используется комбинация \vee и t -нормы [4, 6] и комбинация еще с импликацией [6]. Но, в силу того что у нас используется в качестве множества переменных не линейный отрезок, а общая решетка без умножения элементов, нам эта теория не потребуется. Мы рассмотрим сначала вариант (1) $y_j = \bigvee_i \{x_i \wedge w_{ij}\}$, а потом (3).

Как и в [4], обозначим: $M = \{1 \dots m\}$, $N = \{1 \dots n\}$ – множества индексов, $(X, Y) = \{\mathbf{x}_k, \mathbf{y}_k \mid k \text{ in } P\}$, $P = \{1 \dots p\}$ – семейство пар многозначных паттернов с $\mathbf{x}_k = (x_1^k, \dots, x_n^k)$, $\mathbf{y}_k = (y_1^k, \dots, y_m^k)$ и

$$G_{ij}(X, Y) = \{k \in P \mid x_i^k > y_j^k\}, \quad E_{ij}(X, Y) = \{k \in P \mid x_i^k = y_j^k\}, \quad GE_{ij}(X, Y) = G_{ij}(X, Y) \cup E_{ij}(X, Y),$$

$$L_{ij}(X, Y) = \{k \in P \mid x_i^k < y_j^k\}, \quad LE_{ij}(X, Y) = L_{ij}(X, Y) \cup E_{ij}(X, Y),$$

$$NC_{ij}(X, Y) = \{k \in P \mid x_i^k \notin GE_{ij}(X, Y) \cup LE_{ij}(X, Y)\}.$$

В отличие от [4], определим матрицу $\mathbf{W} = \mathbf{W}_0 = (w_{ij}^0)_{n \times m}$ так:

$$(5) \quad w_{ij}^0 = \bigwedge_{k \in P} \{x_i^k \Rightarrow y_j^k\}.$$

Для $i \in N$ и $j \in M$ определим множества $S_{ij}^G(\mathbf{W}_0, Y)$ и M^w следующим образом:

$$(6) \quad S_{ij}^G(\mathbf{W}_0, Y) = \{k \in GE_{ij}(X, Y) \mid y_j^k \leq w_{ij}^0\},$$

$$(7) \quad M^w = \{\mathbf{W} \in \mu_{n \times m} \mid \forall k \in P, \mathbf{x}_k \circ \mathbf{W} = \mathbf{y}_k\}.$$

Теорема 1 (для правила (1); [4] в случае нечетких нейросетей). Для заданного семейства пар многозначных паттернов $\{\mathbf{x}_k, \mathbf{y}_k \mid k \text{ in } P\}$ и $\mathbf{W}_0 = (w_{ij}^0)_{n \times m}$ выполняются следующие утверждения:

(а) $\forall k \in P, \mathbf{x}_k \circ \mathbf{W}_0 \subset \mathbf{y}_k$, и, если матрица \mathbf{W} удовлетворяет условиям $\forall k \in P, \mathbf{x}_k \circ \mathbf{W} \subset \mathbf{y}_k$, то $\mathbf{W} \subset \mathbf{W}_0$;

(б) Если $M^w \neq \emptyset$, то $\mathbf{W}_0 \subset M^w$ и $\bigvee \mathbf{W} = (w_{ij})_{n \times m} \in M^w$, $\mathbf{W} \subset \mathbf{W}_0$, т.е. $\forall i \in N, \forall j \in M, w_{ij} \leq w_{ij}^0$;

(с) Множество $M^w \neq \emptyset$, если $\forall j \in M, \bigcup_{i \in N} S_{ij}^G(\mathbf{W}_0, Y) = P$.

Доказательство: (а) По определению \mathbf{W}_0 и импликации имеем:

$$\bigvee_{i \in N} \{w_{ij}^0 \wedge x_i^k\} = \bigvee_{i \in N} \{(\bigwedge_{k \in P} \{x_i^k \Rightarrow y_j^k\}) \wedge x_i^k\} \leq \bigvee_{i \in N} \{(x_i^k \Rightarrow y_j^k) \wedge x_i^k\} = \bigvee_{i \in N} \{y_j^k \wedge x_i^k\} \leq y_j^k. \quad \text{Таким образом}$$

$\forall k \in P, \mathbf{x}_k \circ \mathbf{W}_0 \subset \mathbf{y}_k$. Также, если \mathbf{W} удовлетворяет указанным условиям, то $\bigvee_{i \in N} \{w_{ij} \wedge x_i^k\} \leq y_j^k$ и $\forall i \in N, w_{ij} \wedge x_i^k \leq y_j^k$. Откуда $\forall k \in P, w_{ij} \leq x_i^k \Rightarrow y_j^k$. Таким образом, $w_{ij} \leq \bigwedge_{k \in P} \{x_i^k \Rightarrow y_j^k\}$. Следовательно, $\forall i \in N, j \in M, w_{ij} \leq w_{ij}^0$, так что $\mathbf{W} \subset \mathbf{W}_0$ и (а) доказано.

(б) Предположим, $\mathbf{W} \in M^w$. Тогда, $\forall k \in P, j \in M, \bigvee_{i \in N} \{x_i^k \wedge w_{ij}\} = y_j^k$. Отсюда $\forall i \in N, x_i^k \wedge w_{ij} \leq y_j^k$ и $\forall k \in P, w_{ij} \leq x_i^k \Rightarrow y_j^k$. Аналогично (а) $\mathbf{W} \subset \mathbf{W}_0$. Также, $\forall k \in P, j \in M, \forall i \in N, w_{ij}^0 \wedge x_i^k \leq y_j^k$. Тогда выполняется неравенство $y_j^k \geq \bigvee_{i \in N} \{x_i^k \wedge w_{ij}^0\} \geq \bigvee_{i \in N} \{x_i^k \wedge w_{ij}\} = y_j^k$ и, следовательно, $\bigvee_{i \in N} \{x_i^k \wedge w_{ij}^0\} = y_j^k$. Таким образом, $\mathbf{W}_0 \subset M^w$ и (б) доказано.

(с) Пусть $\forall j \in M, \bigcup_{i \in N} S_{ij}^G(\mathbf{W}_0, Y) = P$. Тогда $\forall j \in M, k \in P$ существует такой $i_0 \in N$, что $k \in S_{i_0 j}^G(\mathbf{W}_0, Y)$. Таким образом, $w_{i_0 j}^0 \geq y_j^k, k \in GE_{i_0 j}(X, Y)$ и $x_{i_0}^k : k \in GE_{i_0 j}(X, Y)$. Следовательно, $x_{i_0}^k \wedge w_{i_0 j}^0 \in \{x^k \mid x^k \in L, k \in GE_{i_0 j}(X, Y)\}$. Отсюда

$$(8) \quad \bigvee_{i \in N} \{x_i^k \wedge w_{ij}^0\} \in \{y^l \mid y^l \in L, l \in GE_{ij}(X, Y)\}.$$

По определением импликации и \mathbf{W}_0 имеем из (5) $\forall i \in N, j \in M, k \in P, x_i^k \wedge w_{ij}^0 \in \{y^l \mid y^l \in L, l \in LE_{ij}(X, Y)\}$. Отсюда

$$(9) \quad \bigvee_{i \in N} \{x_i^k \wedge w_{ij}^0\} \in \{y^l \mid y^l \in L, l \in LE_{ij}(X, Y)\}.$$

Комбинируя (8) и (9), получаем, что $\mathbf{W}_0 \subset M^w$ и, следовательно, $M^w \neq \emptyset$. Таким образом, (с) также доказано.

Смысл утверждения (с) этой теоремы в том, что не всякая пара паттернов \mathbf{x} - \mathbf{y} может быть с уверенностью сохранена в такой ассоциативной памяти (так же, как и в нечетком случае) – должно хватать достаточно больших пересечений импликаций в паттернах (5) для того, чтобы $\bigcup S = P$ (Теорема 1 (с), (6)). Кроме того, все \mathbf{x} не могут концентрироваться в нижней зоне (Рис. 2), иначе их объединение может не дать \mathbf{y} (в этом случае от (5) остается 1 и, следовательно, в (1) остаются только \mathbf{x}).

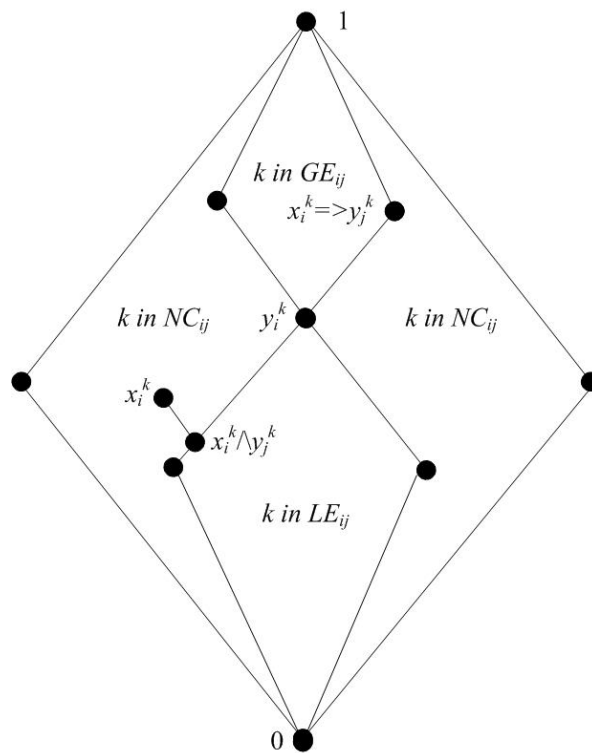


Рис. 2. Импликация и зоны сравнимости в решетке (схематически)

Если, однако, все входные \mathbf{x} находятся в основном в нижней зоне, то необходимо рассмотреть иной вариант композиции – например, вариант композиции (3) $y_j = \bigwedge_i \{x_i \Rightarrow w_{ij}\}$.

В этом случае определим матрицу $\mathbf{W} = \mathbf{W}_0 = (w_{ij}^0)_{n \times m}$ так:

$$(5') \quad w_{ij}^0 = \begin{cases} \bigvee_{k \in P} \{x_i^k \wedge y_j^k\}, & L_{ij}(X, Y) \cup NC_{ij}(X, Y) \neq \emptyset; \\ 0, & L_{ij}(X, Y) \cup NC_{ij}(X, Y) = \emptyset. \end{cases}$$

Для $i \in N$ и $j \in M$ определим множество $S_{ij}^G(\mathbf{W}_0, Y)$ следующим образом (M^w остается прежним (7)):

$$(6') \quad S_{ij}^G(\mathbf{W}_0, Y) = \{k \in LE_{ij}(X, Y) \cup NC_{ij}(X, Y) \mid y_j^k \geq x_i^k \Rightarrow w_{ij}^0\}.$$

Теорема 2. (для правила (3)) Для заданного семейства пар многозначных паттернов $\{\mathbf{x}_k, \mathbf{y}_k \mid k \in P\}$ и $\mathbf{W}_0 = (w_{ij}^0)_{n \times m}$ выполняются следующие утверждения:

(a) $\forall k \in P, \mathbf{x}_k \circ \mathbf{W}_0 \supset \mathbf{y}_k$, и, если матрица \mathbf{W} удовлетворяет условиям $\forall k \in P, \mathbf{x}_k \circ \mathbf{W} \supset \mathbf{y}_k$, то $\mathbf{W} \supset \mathbf{W}_0$;

(b) Если $M^w \neq \emptyset$, то $\mathbf{W}_0 \subset M^w$ и $\forall \mathbf{W} = (w_{ij})_{n \times m} \in M^w, \mathbf{W} \supset \mathbf{W}_0$, т.е. $\forall i \in N, \forall j \in M, w_{ij} \geq w_{ij}^0$;

(c) Множество $M^w \neq \emptyset$, если $\forall j \in M, \bigcup_{i \in N} S_{ij}^G(\mathbf{W}_0, Y) = P$.

Доказательство: (a) Из определений (3) и (5') получаем:
 $\bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}^0\} = \bigwedge_{i \in N} \{x_i^k \Rightarrow \bigvee_{k \in P} (x_i^k \wedge y_j^k)\} \geq \bigwedge_{i \in N} \{x_i^k \Rightarrow (x_i^k \wedge y_j^k)\} = \bigwedge_{i \in N} \{x_i^k \Rightarrow y_j^k\} \geq y_j^k$. Таким образом $\forall k \in P, \mathbf{x}_k \circ \mathbf{W}_0 \supset \mathbf{y}_k$. Также, если \mathbf{W} удовлетворяет указанным условиям, то $\bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}\} \geq y_j^k$ и $\forall i \in N, x_i^k \Rightarrow w_{ij} \geq y_j^k$. Откуда $\forall k \in P, w_{ij} \geq x_i^k \wedge y_j^k$. Таким образом, $w_{ij} \geq \bigvee_{k \in P} \{x_i^k \wedge y_j^k\}$. Следовательно, $\forall i \in N, j \in M, w_{ij} \geq w_{ij}^0$, так что $\mathbf{W} \supset \mathbf{W}_0$ и (a) доказано.

(b) Предположим, $\mathbf{W} \in M^w$. Тогда, $\forall k \in P, j \in M, \bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}\} = y_j^k$. Отсюда $\forall i \in N, x_i^k \Rightarrow w_{ij} \geq y_j^k$ и $\forall k \in P, w_{ij} \geq x_i^k \wedge y_j^k$. Аналогично (a) $\mathbf{W} \supset \mathbf{W}_0$. Также, $\forall k \in P, j \in M, \forall i \in N, x_i^k \Rightarrow w_{ij}^0 \geq y_j^k$. Тогда выполняется неравенство $y_j^k \leq \bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}^0\} \leq \bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}\} = y_j^k$ и, следовательно, $\bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}^0\} = y_j^k$. Таким образом, $\mathbf{W}_0 \subset M^w$ и (b) доказано.

(c) Пусть $\forall j \in M, \bigcup_{i \in N} S_{ij}^G(\mathbf{W}_0, Y) = P$. Тогда $\forall j \in M, k \in P$ существует такой $i_0 \in N$, что $k \in S_{i_0 j}$. Таким образом, $x_{i_0}^k \Rightarrow w_{i_0 j}^0 \leq y_j^k, k \in LE_{i_0 j}(X, Y) \cup NC_{i_0 j}(X, Y)$ и $x_{i_0}^k : k \in LE_{i_0 j}(X, Y) \cup NC_{i_0 j}(X, Y)$. Следовательно, $x_{i_0}^k \Rightarrow w_{i_0 j}^0 \in \{y^l \mid y^l \in L, l \in LE_{i_0 j}(X, Y)\}$. Отсюда

$$(10) \quad \bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}^0\} \in \{y^l \mid y^l \in L, l \in LE_{ij}(X, Y)\}.$$

По определению \mathbf{W}_0 имеем $\forall i \in N, j \in M, k \in P, x_i^k \Rightarrow w_{ij}^0 \in \{y^l \mid y^l \in L, l \in GE_{ij}(X, Y)\}$. Отсюда

$$(11) \quad \bigwedge_{i \in N} \{x_i^k \Rightarrow w_{ij}^0\} \in \{y^l \mid y^l \in L, l \in GE_{ij}(X, Y)\}.$$

Комбинируя (10) и (11), получаем, что $\mathbf{W}_0 \subset M^w$ и, следовательно, $M^w \neq \emptyset$. Таким образом, (c) также доказано.

1.2 Обучающие алгоритмы многозначной ассоциативной памяти

В разделе 1.1 показано, что ситуация, когда входные переменные сосредоточены в основном в нижней зоне, в некотором смысле обратна ситуации, когда они находятся в основном вверху: в Теоремах 1 и 2 менялись местами минимумы и максимумы и обращались включения. Поэтому и динамические алгоритмы обучения также взаимно-обратны: один начинает работу с максимального элемента решетки в качестве начального значения весов, а другой с минимального.

Вообще, все $(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_p, \mathbf{y}_p)$ могут быть сохранены в такой ассоциативной памяти, если существует матрица \mathbf{W} , такая что $\mathbf{X} \circ \mathbf{W} = \mathbf{Y}$. В соответствии с Теоремами 1, 2, такие матрицы существуют, если выполняются условия (c) этих теорем. Поэтому, для получения решений достаточно вычислить выражения (5), (5') и проверить соответствующие условия (c). Но такой алгоритм не демонстрирует адаптивность и саморегулируемость многозначной ассоциативной памяти. Поэтому обобщим динамический δ -обучающий алгоритм нечеткой ассоциативной памяти, введенный в [7-9], на многозначный случай.

Для правила композиции (1) и условия (c) Теоремы 1 получаем

Алгоритм 1 итерации w_{ij} для $i \in N, j \in M$:

Шаг 1. Инициализация: для $\forall i \in N, j \in M$, положим $w_{ij}(0) = 1, t = 0$;

Шаг 2. Пусть $\mathbf{W}(t) = (w_{ij}(t))_{n \times m}$;

Шаг 3. Вычисляем получающийся выход: $\mathbf{X} \circ \mathbf{W}(t) = \mathbf{Y}(t)$. Т.е.

$$\forall k \in P, \forall j \in M, y_j^k(t) = \bigvee_{i=1}^n \{x_i^k \wedge w_{ij}(t)\}.$$

Шаг 4. Подбор весов. Обозначим в этом алгоритме через $\{y_j^k\}$ множество образующих элемента решетки y_j^k и через $\{w\}_{ij}$ матрицу таких множеств. Матрицы w_{ij} и $\{w\}_{ij}$ взаимно-однозначно соответствуют друг другу. Разность множеств будем обозначать знаком минус. Тогда

$$(12) \quad \{w\}_{ij}(t+1) = \begin{cases} \{w\}_{ij}(t), & k : w_{ij}(t) \wedge x_i^k \leq y_j^k, \\ \{w\}_{ij}(t) - (\{x_i^k \Rightarrow y_j^k(t)\} - \{x_i^k \Rightarrow y_j^k\}), & \text{иначе.} \end{cases}$$

Шаг 5. Для $\forall i \in N, j \in M$, проверяем $w_{ij}(t+1) = w_{ij}(t)$? Если это верно, то останавливаемся, иначе $t = t+1$ и переходим к шагу 2.

Для правила композиции (3) и условия (с) Теоремы 2 получаем все то же самое, за исключением того, что стартуем с $w_{ij}(0) = 0, t = 0$ и на шаге 4 получаем следующее выражение:

$$(13) \quad \{w\}_{ij}(t+1) = \begin{cases} \{w\}_{ij}(t) \cup [(\{x_i^k\} \cap \{y_j^k(t)\}) \cup (\{x_i^k\} \cap \{y_j^k\})], & k : x_i^k \Rightarrow w_{ij}(t) < y_j^k, \\ \{w\}_{ij}(t), & \text{иначе.} \end{cases}$$

Теорема 3. (для правила композиции (1)) Пусть последовательность матриц $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ получена Алгоритмом 1. Тогда

- (а) $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ – невозрастающая последовательность матриц;
- (б) $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ сходится;
- (с) $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ сходится к \mathbf{W}_0 , где w_{ij}^0 определена в (5).

Доказательство: (а) Для $i \in N, j \in M, k \in P$ из (12) получаем, $y_j^k(t) \geq x_i^k \wedge w_{ij}(t) > y_j^k$. Т.е. $\{w\}_{ij}(t+1) \leq \{w\}_{ij}(t)$. Следовательно, для $i \in N, j \in M : \mathbf{W}(t+1) \subset \mathbf{W}(t)$. Таким образом, $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ – невозрастающая последовательность матриц.

(б) Поскольку $\forall t = 1, 2, \dots w_{ij}(t)$ ограничены снизу наименьшим элементом решетки 0, последовательность матриц $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ сходится.

(с) Если нет $k \in G(X, Y) \cup NC(X, Y)$ (Рис. 2), то $\forall t : w_{ij}(t) \wedge x_i^k \leq y_j^k$. Поэтому, по (12) $w_{ij}(t) = w_{ij}(0) = w_{ij}^0 = 1$. Если такие k есть, то

$$\{w(1)\}_{ij}(1) = 1 - (\{x_i^k \Rightarrow \bigvee_{i \in N} (x_i^k \wedge w_{ij}(0))\} - \{x_i^k \Rightarrow y_j^k\}) = \{x_i^k \Rightarrow y_j^k\} \geq \{w^0\}_{ij}.$$

$$\{w(2)\}_{ij}(2) = \{x_i^k \Rightarrow y_j^k\} - (\{x_i^k \Rightarrow \bigvee_{i \in N} (x_i^k \wedge (x_i^k \Rightarrow y_j^k))\} - \{x_i^k \Rightarrow y_j^k\}). \text{ Отсюда получаем на основании}$$

определения операции разности множеств

$$\{w(2)\}_{ij} = \{x_i^k \Rightarrow y_j^k\} - (\{x_i^k \Rightarrow \bigvee_{i \in N} (x_i^k \wedge y_j^k)\} - \{x_i^k \Rightarrow y_j^k\}) = \{x_i^k \Rightarrow y_j^k\} = \{w(1)\}_{ij}.$$

Таким образом алгоритм сходится уже на втором шаге. Он сходится к \mathbf{W}_0 , поскольку для всех $k \notin G_{ij}(X, Y) \cup NC_{ij}(X, Y)$ в $w_{ij}^0 = \bigwedge_{k \in P} (x_i^k \Rightarrow y_j^k)$ имеем $\bigwedge_{k \in LE_{ij}(X, Y)} (x_i^k \Rightarrow y_j^k) \geq \bigwedge_{k \in G_{ij}(X, Y) \cup NC_{ij}(X, Y)} (x_i^k \Rightarrow y_j^k)$. Т.е.

$$\bigwedge_{k \in P} w(t)_{ij} = \bigwedge_{k \in G_{ij}(X, Y) \cup NC_{ij}(X, Y)} \{x_i^k \Rightarrow y_j^k\} = w_{ij}^0. \square$$

Из доказательства этой теоремы можно еще раз увидеть, что все входные паттерны не могут находиться в нижней зоне (Рис. 2). Также видно, что в (5) достаточно $k \in G_{ij}(X, Y) \cup NC(X, Y)$, т.е. все определяют \mathbf{x} , лежащие в верхней зоне.

Теорема 4. (для правила композиции (3)) Пусть последовательность матриц $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ получена Алгоритмом 1. Тогда

- (а) $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ – неубывающая последовательность матриц;
- (б) $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ сходится;
- (с) $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ сходится к \mathbf{W}_0 , где w_{ij}^0 определена в (5').

Доказательство: (а) Для $i \in N, j \in M, k \in P$ из (13) получаем $\{w\}_{ij}(t+1) \geq \{w\}_{ij}(t)$. Следовательно для $i \in N, j \in M : \mathbf{W}(t+1) \supset \mathbf{W}(t)$. Таким образом, $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ – неубывающая последовательность матриц.

(б) Поскольку $\forall t = 1, 2, \dots w_{ij}(t)$ ограничены сверху наибольшим элементом решетки 1, последовательность матриц $\{\mathbf{W}(t)|t = 1, 2, \dots\}$ сходится.

(с) Если все $k \in GE(X, Y)$ (Рис. 2), то $\forall t: x_i^k \Rightarrow w_{ij}(t) \geq y_j^k$ или не сравнимо с y_j^k . Поэтому по (13) $w_{ij}(t) = w_{ij}(0) = w_{ij}^0 = 0$. Если не все k такие, то для этих k получаем $\{w(1)\}_{ij} = [(\{x_i^k\} \cap \{\bigwedge_{l \in N} (x_l^k \Rightarrow w_{ij}(0))\}) \cup (\{x_i^k\} \cap \{y_j^k\})] = (\{x_i^k\} \cap \{y_j^k\})$. Отсюда

$$\{w(2)\}_{ij} = (\{x_i^k\} \cap \{y_j^k\}) \cup [(\{x_i^k\} \cap \{\bigwedge_{l \in N} (x_l^k \Rightarrow (\{x_i^k\} \cap \{y_j^k\}))\}) \cup (\{x_i^k\} \cap \{y_j^k\})] = (\{x_i^k\} \cap \{y_j^k\}) = \{w(1)\}_{ij},$$

поскольку $(\{x_i^k\} \cap \{\bigwedge_{l \in N} (x_l^k \Rightarrow (\{x_i^k\} \cap \{y_j^k\}))\}) \leq \{x_i^k\} \cap \{(x_i^k \Rightarrow (\{x_i^k\} \cap \{y_j^k\}))\} = \{x_i^k\} \cap \{y_j^k\}$.

Объединяя по всем k (оставшиеся k вклада не дают), получаем, что алгоритм сходится уже на втором шаге и к \mathbf{W}_0 . □

Из доказательства этой теоремы можно еще раз увидеть, что все входные паттерны не могут находиться в верхней зоне (Рис. 2).

2 Управление группой БПЛА

В этом разделе рассматривается задача обеспечения безопасности группового полета по взаимному расположению объектов, а также при внешней помехе, изученная в [2] с помощью многозначной логики без использования нейронных сетей. В этой ситуации должно выбираться направление ухода для предотвращения возможного столкновения [1]. В [1] используется ряд параметров, которые влияют на этот выбор. Для простоты рассмотрим только два таких параметра – «пеленг помехи» и «направление поворота». Решеточные оценки значений этих параметров найдем с помощью многозначных нейронных сетей. Из них получим управляющие решения по определению «направления поворота» в зависимости от «пеленга помехи».

Нейронные сети здесь будут выступать в двух видах: во-первых, многозначная ассоциативная память будет применена для хранения вариантов «направления поворота» в зависимости от «пеленга помехи»; во-вторых, как мы видели, для вычисления весовой матрицы в правиле композиции (1) и выходов в правиле композиции (3), необходимо вычислять импликации. В случае сложной решетки значений параметров это вычисление может быть непростым делом. Однако многозначная нейронная сеть сильно упрощает вычисления.

2.1 Вычисление импликаций

Рассмотрим однослойную нейронную сеть с решеткой L в качестве слоя Рис. 3:

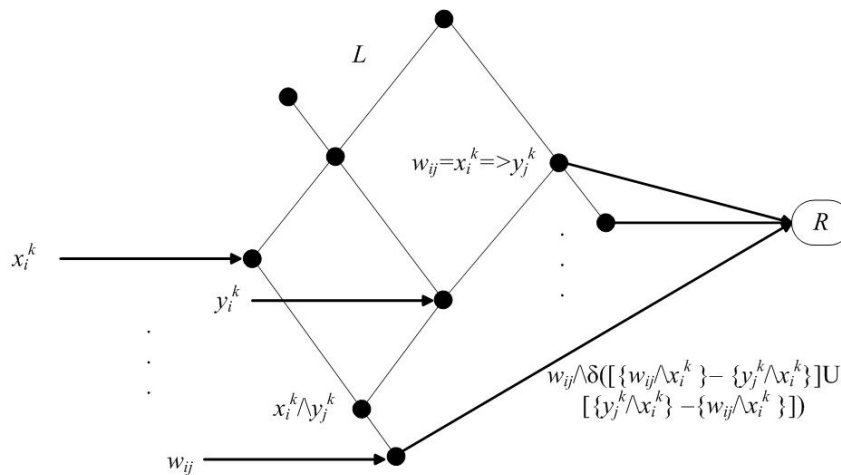


Рис. 3. Нейронная сеть для вычисления импликаций

На этом рисунке каждая вершина w_{ij} дистрибутивной решетки L является многозначным нейроном со входом w_{ij} и с выходом $w_{ij} \wedge \delta([\{w_{ij} \wedge x_i^k\} - \{y_j^k \wedge x_i^k\}] \cup [\{y_j^k \wedge x_i^k\} - \{w_{ij} \wedge x_i^k\}])$, где через $\{x\}$ обозначено множество образующих элемента решетки x , а функция δ определяется так:

$$(14) \quad \delta(\{x\}) = \begin{cases} 1, & \{x\} = \emptyset; \\ 0, & x \in L, \{x\} \neq \emptyset. \end{cases}$$

Таким образом, когда множества образующих $w_{ij} \wedge x_i^k$ и $y_j^k \wedge x_i^k$ равны, их разность дает 0 и на выход идет значение w_{ij} . В противном случае на выходе нейрона w_{ij} будет 0. В элементе сети R

вычисляется $\max(w_{ij})$. Поскольку решетка дистрибутивная, такое наибольшее значение существует. По определению оно и будет импликацией.

2.2 Управляющие решения на основе ассоциативной памяти

В полете величины управляющих воздействий не остаются постоянными из-за постоянного изменения полетной информации. Управляющие решения в этом случае можно получать, основываясь на правилах, в которых продукции получаются по схеме нечеткого логического вывода. Обычная схема здесь такова: нечеткий консеквент y получается из нечетких антецедента x и импликации $x \Rightarrow y$ по следующему правилу

$$(15) \quad y = x \circ x \Rightarrow y,$$

где операция \circ обозначает композицию нечетких переменных. При этом импликация обычно задается прецедентом, в котором x и y известны, и вычисляется по одной из множества существующих формул. Выбор определенного правила из этого множества предоставляется опыту разработчика. Связано это с тем, что в линейно-упорядоченной шкале истинностных значений нет однозначного определения отрицания и, соответственно, импликации [3]. Однако это не так в дистрибутивной решетке общего вида.

Рассмотрим часть дистрибутивной решетки с четырьмя атомами-образующими Рис.4.

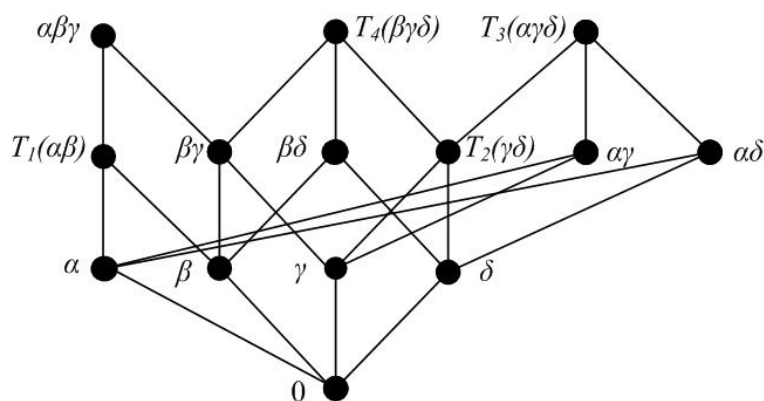


Рис. 4. Часть решетки для оценки истинностных значений ситуации обнаружения помехи

На этом рисунке для упрощения используются такие обозначения для объединений: $\alpha\beta \equiv \alpha \cup \beta$.

Элементам этой решетки можно придать следующие смыслы. Пусть лингвистическая переменная x – «пеленг помехи», – состоит из двух термов $T_1 = (\alpha, \beta, \alpha\beta, 0)$ – «помеха в горизонтальной плоскости» с оценками α – «слева», β – «справа», $\alpha\beta$ – «впереди» и $T_2 = (\gamma, \delta, \gamma\delta, 0)$ – «помеха в продольной вертикальной плоскости» с оценками γ – «сверху», δ – «снизу», $\gamma\delta$ – «впереди». Общая оценка 0 является пересечением образующих атомов и трактуется как «сзади», а объединения $\alpha\beta$ и $\gamma\delta$ трактуются как «впереди».

Пусть лингвистическая переменная y – «направление поворота» также состоит из двух термов $T_3 = (\alpha, \gamma, \delta, \alpha\gamma, \alpha\delta, \gamma\delta, \alpha\gamma\delta, 0)$ – «в левую полусферу» и $T_4 = (\beta, \gamma, \delta, \beta\gamma, \beta\delta, \gamma\delta, \beta\gamma\delta, 0)$ – «в правую полусферу» с аналогичными комбинациями оценок как для T_1 и T_2 : например, $\alpha\gamma$ обозначает «влево и вверх», а $\alpha\gamma\delta$ – «вперед и влево». Предположим, что $x = (\langle \alpha | T_1 \rangle, \langle \gamma | T_2 \rangle)$, т.е. помеха находится слева и сверху. Надо определить направление ухода от нее $y = (\langle ? | T_3 \rangle, \langle ? | T_4 \rangle)$. Для этого следует использовать следующую композицию. Предположим, что есть прецедент $y = (\langle \delta | T_3 \rangle, \langle \beta\delta | T_4 \rangle)$ – уход вправо и вниз. Тогда для $x \Rightarrow y$ можно использовать матрицу [2]

$$(16) \quad x \Rightarrow y = \neg x = \begin{pmatrix} \gamma\delta, \beta\gamma\delta \\ \alpha\delta, \beta\delta \end{pmatrix}$$

откуда получается **правило продукции** $y = P[x, x \Rightarrow y] = \wedge(x \Rightarrow 0) = \wedge \neg x$. Здесь отрицание x берется относительно T_3 и T_4 . При таком правиле БПЛА будет всегда уходить в сторону, противоположную направлению на помеху.

Это правило продукции представляет собой правило композиции (3) в многозначной ассоциативной памяти. Такая память может использоваться для хранения вариантов управляющих

решений в зависимости от полетной ситуации. В этом случае входные и выходные паттерны таковы ($k = 16$):

$$\begin{aligned} \mathbf{x}_1 &= (\alpha, \gamma), \mathbf{y}_1 = (\delta, \beta\delta); \mathbf{x}_5 = (\beta, \gamma), \mathbf{y}_5 = (\alpha\delta, \delta); \mathbf{x}_9 = (\alpha\beta, \gamma), \mathbf{y}_9 = (\delta, \delta); \mathbf{x}_{13} = (0, \gamma), \mathbf{y}_{13} = (\alpha\delta, \beta\delta); \\ \mathbf{x}_2 &= (\alpha, \delta), \mathbf{y}_2 = (\gamma, \beta\gamma); \mathbf{x}_6 = (\beta, \delta), \mathbf{y}_6 = (\alpha\gamma, \gamma); \mathbf{x}_{10} = (\alpha\beta, \delta), \mathbf{y}_{10} = (\gamma, \gamma); \mathbf{x}_{14} = (0, \delta), \mathbf{y}_{14} = (\alpha\gamma, \beta\gamma); \\ \mathbf{x}_3 &= (\alpha, \gamma\delta), \mathbf{y}_3 = (0, \beta); \mathbf{x}_7 = (\beta, \gamma\delta), \mathbf{y}_7 = (\alpha, 0); \mathbf{x}_{11} = (\alpha\beta, \gamma\delta), \mathbf{y}_{11} = (0, 0); \mathbf{x}_{15} = (0, \gamma\delta), \mathbf{y}_{15} = (\alpha, \beta); \\ \mathbf{x}_4 &= (\alpha, 0), \mathbf{y}_4 = (\gamma\delta, \beta\gamma\delta); \mathbf{x}_8 = (\beta, 0), \mathbf{y}_8 = (\alpha\gamma\delta, \gamma\delta); \mathbf{x}_{12} = (\alpha\beta, 0), \mathbf{y}_{12} = (\gamma\delta, \gamma\delta); \mathbf{x}_{16} = (0, 0), \mathbf{y}_{16} = (\alpha\gamma\delta, \beta\gamma\delta). \end{aligned}$$

Для таких данных получаем весовую матрицу в соответствии с Алгоритмом 1 для правила композиции (3) и весовой матрицы (5’):

$$(17) \quad w_{ij} = \bigvee_{k \in P} \{x_i^k \wedge y_j^k\} = \begin{pmatrix} \bigvee_k \{0\} & \bigvee_k \{0\} \\ \bigvee_k \{0\} & \bigvee_k \{0\} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = w_{ij}(0).$$

Легко проверить, что выполняется (3): $y_j^k = \bigwedge_i \{x_i^k \Rightarrow w_{ij}\}$, если импликации ограничивать на термы T_3 и T_4 , т.е. если брать пересечение значения импликации и подрешетки соответствующего терма.

Таким образом продемонстрировано, что многозначная ассоциативная память может быть эффективным средством хранения данных о вариантах реагирования на изменение полетной ситуации, а многозначная нейронная сеть позволяет легко вычислять импликации для такой ассоциативной памяти.

Литература

1. Гайнуллин И.А., Роголев А.П. Построение нечетких баз знаний ситуационных систем интеллектуальной поддержки решения задач авиационных бортовых комплексов // Авиакосмическое приборостроение. 2007, №2. – С.57-66.
2. Maximov D. Control in a Group of Unmanned Aerial Vehicles Based on Multi-Valued Logic // Proceedings of the 12th International Conference "Management of Large-Scale System Development" (MLSD). Moscow, Russia: IEEE, 2019. – P. 1-5 <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8911092>.
3. Максимов Д.Ю. Логика Н.А. Васильева и многозначные логики // Логические исследования. 2016. Т. 22, № 1. – С. 82-107.
4. Liu P., Li H. Fuzzy neural network theory and application // Series in Machine Perception and Artificial Intelligence. Vol. 59 – London: World Scientific Publishing Co. Pte. Ltd., 2004. – 395 p.
5. Kosko B. Fuzzy associative memories // In Kandel A. (ed.), Fuzzy Expert Systems Reading – MA: Addison-Wesley, 1987.
6. Sussner P., Valle M.E. Implicative Fuzzy Associative Memories // IEEE Transactions on Fuzzy Systems. 2006. Vol. 14, No. 6. – P. 793-807.
7. Li X. Z., Ruan Da Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part I // In Fuzzy Sets and Systems, 1997. Vol. 90. – P. 11-23.
8. Li X. Z., Ruan Da Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part II // In Fuzzy Sets and Systems, 1999. Vol. 103. – P. 473-486.
9. Li X. Z., Ruan Da Novel neural algorithms based on fuzzy δ rules for solving fuzzy relation equations: Part III // In Fuzzy Sets and Systems, 2000. Vol. 109. – P. 355-362.