KDS-95

ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ ЛОГИЧЕСКИХ РАССУЖДЕНИЙ

В.П.Семеренко

(286021,Украина,Винница,Хмельницкое шоссе,95,
Винницкий государственный технический университет)

Abstract. Formalization of inference method in propositional calculus and predicate calculus is considered and new interpretation categorical syllogism is given. Hardwarily implementation inference by means programmed systolic structure is proposed.

Введение

Решение многочисленных задач искусственного интеллекта связано с широким использованием логических представлений и конструкций.

Со времени Аристотеля логика прошла длительный путь становления и развития. Современная математическая логика достигла впечатляющих успехов и стала основой для теории проектирования вычислительной техники.

Необходимость создания компьютеров новых архитектур выдвигает перед логикой новые задачи. Важнейшей из них является эффективная аппаратная поддержка логического вывода с использованием принципов параллельных вычислений.

Традиционные методы доказательства логических законов являются очень громоздкими и трудноформализуемыми. Поэтому необходим поиск новых интерпретаций логического вывода, которые позволили бы гармонично соединить строгую теорию логики с перспективной элементной базой будущих поколений компьютеров.

Логический вывод в исчислении высказываний

Центральной проблемой искусственного интеллекта с позиций логики является формализация логического вывода, т.е. формализация логических рассуждений [1-4].

Как и в [1], под правильным рассуждением будем понимать такое рассуждение, когда между посылками $A_1, A_2, \ldots, A_n$ и заключением C этого рассуждения имеет место логическое следование:

$$\left(A_1, A_2, \ldots, A_n\right) \rightarrow C$$

т.е. из посылок $A_1, A_2, \ldots, A_n$ следует (или выводится) заключение C .

Это правило вывода удобно записать в виде следующей формулы:

$$\left\{A_1, A_2, \ldots, A_n\right\} \models C \ . \qquad (1)$$

Тогда, для доказательства корректности данного рассуждения, достаточно доказать, что формула (1) является тождественно-истинной. В этом и заключается суть определения разрешающей процедуры.

Известные методы выполнения разрешающей процедуры в исчислении высказываний либо очень громоздки (использование таблиц истинности), либо трудноформализуемы (преобразования нормальных форм).

Сформулируем новый способ построения разрешающей процедуры, лишенный указанных недостатков.

Для этого будем рассматривать исчисление высказываний не в аксиоматическом виде, а в виде теории истинностных функций[1].

Согласно этой теории для любой формулы $F(A_1, A_2, \dots, A_n)$ в исчислении высказываний имеется n-местная функция на множестве {0,1}, которая заданным значениям истинности переменных $A_1, A_2, \dots, A_n$ сопоставляет значение истинности формулы F. Эту функцию будем называть истинностной функцией $f(A_1, A_2, \dots, A_n)$ формулы $F(A_1, A_2, \dots, A_n)$.

Аналогично трем классам формул можно ввести три класса истинностных функций: тождественно-истинных (общезначимых), тождественно-ложных (опровержимых) и выполнимых (нейтральных).

Булева алгебра истинностных функций изоморфна булевой алгебре кубических функций[6]. Операциям конъюнкции, дизъюнкции и отрицания в булевой алгебре истинностных функций соответствуют операции пересечения, объединения и дополнения кубов в булевой алгебре кубических функций, а нормальным формам соответствуют кубические покрытия.

Кубическое D-покрытие (кубическое R-покрытие) некоторой истинностной функции f - это представленная в кубической форме (т.е. в алфавите {0,1, x }) минимальная дизъюнктивная нормальная форма (МДНФ) прямой функции f (инверсной функции $\bar{f}$). Каждой импликанте МДНФ соответствует один куб покрытия, прямое (инверсное) значение переменной в импликанте МДНФ соответствует единичному (нулевому) значению компоненты куба покрытия, а число переменных истинностной функции равно числу компонентов куба покрытия. Аналогичным образом кубическое K-покрытие (кубическое Q-покрытие) некоторой истинностной функции f соответствует минимальной конъюнктивной нормальной форме (МКНФ) прямой функции f (инверсной функции $\bar{f}$).

Например, МКНФ истинностной функции f (инверсной функции $\bar{f}$):

$$f(a,b,c,d)=(a\vee\bar{d})\wedge(\bar{b}\vee\bar{c})\wedge(c\vee d)$$

$$\bar{f}(a,b,c,d)=(\bar{a}\vee c\vee\bar{d})\wedge(b\vee\bar{c}\vee d)\wedge(\bar{a}\vee b\vee\bar{c})$$

соответствуют следующие покрытия:

$$K=\begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{bmatrix} 1 & x & x & 0 \\ x & 0 & 0 & x \\ x & x & 1 & 1 \end{bmatrix} \end{array}, \qquad Q=\begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{bmatrix} 0 & x & 1 & 0 \\ x & 1 & 0 & 1 \\ 0 & 1 & 0 & x \end{bmatrix} \end{array}.$$

Если функция f содержит операции импликации и эквивалентности, тогда их необходимо вначале выразить через операции дизъюнкции и конъюнкции, а затем получить соответствующие покрытия.

Согласно известной процедуры Эрбрана [4], для доказательства общезначимости формулы (1) или функции f достаточно показать противоречивость данной формулы или функции.

При кубической интерпретации разрешающей процедуры доказательство общезначимости функции может быть сведено к доказательству того, что R-покрытие или K-покрытие функции f пусты. Сам процесс доказательства состоит в том, что по заданному D-покрытию (Q-покрытию) по строго формализованному алгоритму ведется поиск кубов R-покрытия (K-покрытия). В случае нахождения хотя бы одного такого куба рассматриваемая функция будет выполнимой, а при отсутствии кубов R -покрытия (K -покрытия) - функция будет общезначимой.

Аналогично, для доказательства опровержимости истинностной функции достаточно показать, что ее D -покрытие или Q -покрытия пусты.

Для выполнения разрешающей процедуры по такому способу необходимо иметь в качестве исходных данных кубическое покрытие всей функции, что в ряде случаев затруднительно. Более удобен декомпозиционный алгоритм выполнения разрешающей процедуры, который состоит из следующих этапов.

1. Каждую посылку A_i формулы (1) представить в виде истинностной подфункции f_i и записать ее D_i -покрытие ($i=1\div n$) в формате $A_1\ A_2 \ldots\ A_n\ C$.

2. Заключение C формулы (1) представить в виде истинностной подфункции f_c и записать ее R_c -покрытие в формате $A_1\ A_2 \ldots\ A_n\ C$.

3. Для полученных покрытий

$$D_1, D_2, \ldots , D_n, R_c \ . \tag{2}$$

определить результат их пересечения:

$$D_1 \cap D_2 \cap \ldots \cap D_n \cap R_c \ . \tag{3}$$

Смысл операции пересечения кубов (¿ -операции) такой же, как и в [6].

Если результат (3) пуст, тогда исходная формула (1) общезначима, а если результат (3) не пуст - формула (1) выполнима.

Пусть покрытия $D_1, D_2, \ldots , D_n, R_c$ содержат соответственно $k_1, k_2, \ldots , k_n, k_{n+1}$ кубов. Тогда для подтверждения общезначимости формулы (1) всегда потребуется N_k итераций:

$$N_k = k_1 \times k_2 \times \ldots \times k_n \times k_{n+1} \ .$$

Здесь под итерацией понимается взаимное пересечение ($n+1$) кубов, взятых по одному из каждого из покрытий (2).

Формула (1) будет выполнимой, если будет найден непустой результат пересечения ($n+1$) указанных кубов хотя бы на одной итерации. Поэтому, для подтверждения выполнимости формулы (1) может потребоваться от одной до N_k итераций.

В качестве примера выполним разрешающую процедуру для следующей формулы:

$$\{a \to (b \cup c),\ b \to d,\ c \to d\} \mid= d \ . \tag{4}$$

1. Запишем покрытия D_1 , D_2 и D_3 для соответствующих трех посылок формулы (4) в формате $(a\ b\ c\ d)$:

$$D_1 = \begin{array}{c} \begin{matrix} a & b & c & d \end{matrix} \\ \begin{bmatrix} 0 & x & x & x \\ x & 1 & x & x \\ x & x & 1 & x \end{bmatrix} \end{array}, \quad D_2 = \begin{array}{c} \begin{matrix} a & b & c & d \end{matrix} \\ \begin{bmatrix} x & 0 & x & x \\ x & x & x & 1 \end{bmatrix} \end{array}, \quad D_3 = \begin{array}{c} \begin{matrix} a & b & c & d \end{matrix} \\ \begin{bmatrix} x & x & 0 & x \\ x & x & x & 1 \end{bmatrix} \end{array} .$$

2. Запишем покрытие R 4c 0 для заключения формулы (4):

$$R_c = \begin{array}{c} \begin{matrix} a & b & c & d \end{matrix} \\ \begin{bmatrix} x & x & x & 0 \end{bmatrix} \end{array} .$$

3. Выполним взаимное пересечение полученных покрытий:

$$D_1 \cap D_2 \cap D_3 \cap R_c \ .$$

На одной из двенадцати итераций получается непустой результат пересечения кубов указанных покрытий:

$$\begin{array}{cccc} 0 & x & x & x \\ x & 0 & x & x \\ x & x & 0 & x \\ x & x & x & 0 \\ \hline 0 & 0 & 0 & 0 \end{array} .$$

Следовательно, формула (4) выполнима.

Очевидно, что число итераций, необходимых для определения выполнимости формулы (1) связано со стратегией выбора кубов из покрытий(2).Для минимизации разрешающей процедуры можно использовать различные варианты упорядочения кубов в покрытиях.

Если посылка A_i формулы (1) представляет собой сложное выражение, состоящее из h дизъюнктов, тогда из них можно сформировать h отдельных D_i^j -покрытий,а общее D_i -покрытие будет в этом случае равно:

$$D_i = D_i^1 \cap D_i^2 \cap \ldots \cap D_i^h \,, \qquad j = 1 \div h \,.$$

Аналогично, если заключение C формулы (1) состоит из h конъюнктов, тогда из них можно сформировать h отдельных R_c^j -покрытий, а общее R_c -покрытие будет равно:

$$R_c = R_c^1 \cap R_c^2 \cap \ldots \cap R_c^h \,, \qquad j = 1 \div h \,.$$

Если посылки A_i и заключение C представляют собой выражения, содержащих операции импликации и эквивалентности, тогда необходимо их выразить через операции дизъюнкции и конъюнкции.

Логический вывод в исчислении предикатов

Известные методы формализации логического следования в исчислении предикатов (аксиоматический, секвенциальный, натуральный вывод) представляют собой длительный и трудоемкий процесс, поскольку каждый шаг доказательства формулы (1) требует определенной интуиции, навыков, и, в конечном итоге, все сводится к полному перебору возможных вариантов.

Более эффективное решение проблемы логического вывода может быть найдено, если ее также свести к проблеме нахождения методов распознавания общезначимости или выполнимости логических формул.

Поскольку для исчисления предикатов не существует общего способа установления общезначимости, поэтому также ограничимся рассмотрением только одноместного исчисления предикатов [3,4].

Как и в исчислении высказываний, будем различать три класса формул исчисления предикатов: тождественно-истинных (истинных при любых интерпретациях), тождественно-ложных (ложных при любых интерпретациях) и выполнимых (истинных только при некоторых интерпретациях).

Рассмотрим вначале возможные подходы к формализации доказательства для конечной предметной области M , содержащей предметные переменные $a_1, a_2, \ldots , a_n$:

$$M = \{a_1, a_2, \ldots , a_n\} \,. \qquad (5)$$

В этом случае все кванторы переходят в конечные формулы исчисления высказываний:

$$\forall x F(x) = F(a_1) \wedge F(a_2) \wedge \ldots \wedge F(a_n) \ , \qquad (6)$$

$$\exists x F(x) = F(a_1) \cup F(a_2) \cup \ldots \cup F(a_n) \ . \qquad (7)$$

По аналогии с исчислением высказываний можно определить также R -, K - и Q - покрытия предикатных формул. Каждое из четырех видов покрытий однозначно определяет вид других покрытий, поэтому, в дальнейшем будем рассматривать только D - и R - покрытия.

Согласно [1], если формулы (6) и (7) исчисления высказываний будут тождественными (тождественно-истинными или тождественно-ложными), тогда соответствующие им предикатные формулы будут n-тождественными. А если какая-либо формула является n -тождественной для любого числа n , тогда она является тождественной в конечном. В [1] также доказано, что тождественная в конечном формула одноместного исчисления предикатов всегда будет выводимой.

Рассмотрим кубическое представление предикатных формул.

Формуле (6) соответствует D-покрытие вида:

$$D = \begin{array}{c} \begin{matrix} F(a_1) & F(a_2) & \ldots & F(a_n) \end{matrix} \\ \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix} \end{array}$$

Формуле (7) соответствует D -покрытие вида:

$$D = \begin{array}{c} \begin{matrix} F(a_1) & F(a_2) & \ldots & F(a_n) \end{matrix} \\ \begin{bmatrix} 1 & x & \ldots & x \\ x & 1 & \ldots & x \\ \ldots & \ldots & \ldots & \ldots \\ x & x & \ldots & 1 \end{bmatrix} \end{array} .$$

По аналогии с исчислением высказываний, можно определить и другие кубические покрытия предикатных формул. Не умаляя общности, ограничимся двухэлементной предметной областью:

$$M = \{a_1, a_2\} \ . \qquad (8)$$

Для получения кубического представления произвольной предикатной формулы одноместного исчисления предикатов на конечной предметной области необходимо вначале иметь покрытия типовых фрагментов таких формул. В [1] показано, что любая формула одноместного исчисления предикатов может быть разложена в примарные формулы, т.е. формулы, которые либо являются элементарными, либо имеют один из следующих двух видов:

$$\forall x \left(F_1(x) \cup F_2(x) \cup \ldots \cup F_n(x)\right) \ , \qquad (9)$$

$$\exists x \left(F_1(x) \wedge F_2(x) \wedge \ldots \wedge F_n(x)\right) \ , \qquad (10)$$

где $F_i(x)$ - либо элементарные формулы исчисления высказываний либо их отрицания ($i = 1 \div n$).

Для предметной области (8) кубические покрытия формулы (9) при n =2 имеют вид:

$$D=\begin{array}{cccc} F_1(a_1) & F_1(a_2) & F_2(a_1) & F_2(a_2) \end{array} \atop \begin{bmatrix} 1 & 1 & x & x \\ x & x & 1 & 1 \end{bmatrix} \quad , \quad R=\begin{array}{cccc} F_1(a_1) & F_1(a_2) & F_2(a_1) & F_2(a_2) \end{array} \atop \begin{bmatrix} 0 & x & 0 & x \\ 0 & x & x & 0 \\ x & 0 & 0 & x \\ x & 0 & x & 0 \end{bmatrix} \quad .$$

Для предметной области (8) кубические покрытия формулы (10) имеют вид:

$$D=\begin{array}{cccc} F_1(a_1) & F_1(a_2) & F_2(a_1) & F_2(a_2) \end{array} \atop \begin{bmatrix} 1 & x & 1 & x \\ 1 & x & x & 1 \\ x & 1 & 1 & x \end{bmatrix} \quad , \quad R=\begin{array}{cccc} F_1(a_1) & F_1(a_2) & F_2(a_1) & F_2(a_2) \end{array} \atop \begin{bmatrix} 0 & 0 & x & x \\ x & x & 0 & 0 \\ x & 1 & x & 1 \end{bmatrix} \quad .$$

Формулы вида:

$$\forall x\, (F_1(x) \wedge F_2(x) \wedge \ldots \wedge F_n(x)) ,$$
$$\exists x\, (F_1(x) \cup F_2(x) \cup \ldots \cup F_n(x))$$

представляются соответственно в виде:

$$\forall x\ F_1(x) \wedge \forall x F_2(x) \wedge \ldots \wedge \forall x F_n(x) ,$$
$$\exists x\ F_1(x) \cup \exists x F_2(x) \cup \ldots \cup \exists x F_n(x) .$$

Если в формуле присутствуют логические операции импликации и эквивалентности,тогда они должны быть заменены операциями дизъюнкции и конъюнкции.

Чтобы доказать выводимость какой-либо предикатной формулы, достаточно доказать невыводимость ее обратной формулы, т.е. показать, что ее R -покрытие или Q -покрытие пусты. Аналогично, для доказательства невыводимости какой-либо предикатной формулы достаточно показать, что ее D -покрытие или K -покрытие пусты.

Таким образом, разрешающая процедура в исчислении предикатов также сводится к операциям с кубическими покрытиями, следовательно, могут быть использованы ранее рассмотренные методы логического вывода исчисления высказываний.

Для произвольной формулы исчисления предикатов метод интерпретации не всегда применим в связи с существованием бесконечных предметных областей. Однако, в ряде случаев можно использовать интерпретацию на Эрбрановой области [3].

Любую формулу исчисления предикатов можно также преобразовать в эквивалентную ей предваренную нормальную форму (ПНФ) вида:

$$Q_1(x_1)\, Q_2(x_2) \ldots Q_n(x_n)\, F_n ,$$

где Q_i - кванторы, ($i=1 \div n$).

F_n - формула, не содержащая кванторов.

Формула F_n допускает любые преобразования исчисления высказываний, в том числе получение ДНФ или КНФ. Поэтому, разрешающая процедура для формул одноместного исчисления предикатов, заданных в ПНФ, сводится к определению класса формулы F_n (или соответствующей кубической функции f),аналогично, как и в исчислении высказываний. При этом необязательно обращаться в процессе доказательства к конечной предметной области предикатов. ПНФ позволяет также выполнить разрешающую процедуру еще в ряде случаев многоместного исчисления предикатов[1],[3].

Формализация силлогистики

В последние годы возрос интерес к естественным рассуждениям, которые впервые были формализованы Аристотелем [2].

С позиций современной формальной логики аристотелевскую силлогистику можно кратко охарактеризовать как теорию четырех видов рассуждений:

1)Всякий S есть P (ASP),

2)Всякий S не есть P (ESP),

3)Некоторый S есть P (ISP),

4)Некоторый S не есть P (OSP),

где силлогистические переменные S и P стоят вместо общих и непустых терминов конкретных рассуждений.

Простой категорический силлогизм состоит из трех категорических суждений: E , F и G , из которых E и F являются посылками, а G - заключением: $E \rightarrow (F \rightarrow G)$.

Его также можно записать в виде:

$$\{E,F\} \models G \qquad (11)$$

Существует 19 правильных разновидностей (модусов) категорического силлогизма.

Рассмотрим интерпретацию силлогистики на основе исчисления предикатов. Поскольку силлогистические высказывания носят экзистенциальный характер, поэтому не все известные способы их представления на языке исчисления предикатов являются полностью адекватными.

Особенности силлогистики наиболее точно учтены в интерпретации В.А.Смирнова [5],согласно которой четыре фундаментальных силлогистических высказывания имеют вид:

ASP интерпретируется как $\exists x S(x) \wedge \forall x (S(x) \rightarrow P(x))$,

ESP интерпретируется как $\forall x (S(x) \rightarrow \overline{P(x)})$,

ISP интерпретируется как $\exists x (S(x) \wedge P(x))$,

OSP интерпретируется как $\exists x S(x) \rightarrow \exists x (S(x) \wedge \overline{P(x)})$.

Используя покрытия предикатных формул для предметной области (8), можно предложить следующую кубическую интерпретацию силлогистических высказываний:

для ASP:

$$D^A = \begin{array}{c} \begin{matrix} S(a_1) & S(a_2) & P(a_1) & P(a_2) \end{matrix} \\ \begin{bmatrix} 1 & x & 1 & 1 \\ x & 1 & 1 & 1 \end{bmatrix} \end{array} , \quad R^A = \begin{array}{c} \begin{matrix} S(a_1) & S(a_2) & P(a_1) & P_2(a_2) \end{matrix} \\ \begin{bmatrix} x & x & 0 & x \\ x & x & x & 0 \\ 0 & 0 & x & x \end{bmatrix} \end{array} ;$$

для ESP:

$$D^E = \begin{array}{c} \begin{matrix} S(a_1) & S(a_2) & P(a_1) & P(a_2) \end{matrix} \\ \begin{bmatrix} 0 & 0 & x & x \\ x & x & 0 & 0 \end{bmatrix} \end{array} , \quad R^E = \begin{array}{c} \begin{matrix} S(a_1) & S(a_2) & P(a_1) & P(a_2) \end{matrix} \\ \begin{bmatrix} 1 & x & 1 & x \\ x & 1 & 1 & x \\ 1 & x & x & 1 \\ x & 1 & x & 1 \end{bmatrix} \end{array} ;$$

для ISP:

$$D^I = \begin{array}{cccc} S(a_1) & S(a_2) & P(a_1) & P(a_2) \\ \end{array} \begin{bmatrix} 1 & x & 1 & x \\ x & 1 & 1 & x \\ 1 & x & x & 1 \\ x & 1 & x & 1 \end{bmatrix} \quad , \quad R^I = \begin{array}{cccc} S(a_1) & S(a_2) & P(a_1) & P(a_2) \\ \end{array} \begin{bmatrix} 0 & 0 & x & x \\ x & x & 0 & 0 \end{bmatrix} \quad ;$$

для OSP:

$$D^O = \begin{array}{cccc} S(a_1) & S(a_2) & P(a_1) & P_2(a_2) \\ \end{array} \begin{bmatrix} x & x & 0 & x \\ x & x & x & 0 \\ 0 & 0 & x & x \end{bmatrix} \quad , \quad R^O = \begin{array}{cccc} S(a_1) & S(a_2) & P(a_1) & P(a_2) \\ \end{array} \begin{bmatrix} 1 & x & 1 & 1 \\ x & 1 & 1 & 1 \end{bmatrix} \quad .$$

При таком представлении силлогистические высказывания превращаются в общезначимые формулы одноместного исчисления предикатов, следовательно, можно использовать ранее рассмотренные способы логического вывода. В качестве примера докажем выводимость первого модуса третьей фигуры силлогистики, традиционная запись которого имеет вид:

M a P
M a S
------ . (12)
S i P

В соответствии с изложенным ранее алгоритмом построения разрешающей процедуры, необходимо доказать, что

$$D_1^A \cap D_2^A \cap R_C^I = \varnothing \quad . \tag{13}$$

где D_1^A, D_2^A, R_C^I - покрытия суждений соответственно MaP, MaS и SiP.

Поскольку в (12) используется три термина, поэтому будем использовать кубические покрытия для области (8) в следующем формате:

$$D_1^A = \begin{array}{cccccc} S(a_1) & S(a_2) & M(a_1) & M(a_2) & P(a_1) & P(a_2) \\ \end{array} \begin{bmatrix} x & x & 1 & x & 1 & 1 \\ x & x & x & 1 & 1 & 1 \end{bmatrix} \quad ,$$

$$D_2^A = \begin{array}{cccccc} S(a_1) & S(a_2) & M(a_1) & M(a_2) & P(a_1) & P(a_2) \\ \end{array} \begin{bmatrix} 1 & 1 & 1 & x & x & x \\ 1 & 1 & x & 1 & x & x \end{bmatrix} \quad ,$$

$$R_C^I = \begin{array}{cccccc} S(a_1) & S(a_2) & M(a_1) & M(a_2) & P(a_1) & P(a_2) \\ \end{array} \begin{bmatrix} 0 & 0 & x & x & x & x \\ x & x & x & x & 0 & 0 \end{bmatrix} \quad .$$

Непосредственное выполнение взаимного пересечения покрытий D_1^A, D_2^A, R_C^I показывает справедливость выражения (13), и, следовательно, выводимость рассматриваемого модуса.

Аппаратная реализация логического вывода

Аппаратная реализация логического вывода требует создания обрабатывающих устройств нетрадиционной архитектуры [7].

Основным блоком устройства для выполнения рассмотренного ранее алгоритма разрешающей процедуры должен быть блок для выполнения операций над кубическими покрытиями. Поскольку обработку отдельных компонент куба можно выполнять одновременно и независимо друг от друга, поэтому процесс пересечения кубов покрытий можно распараллелить на основе систолического (параллельно-конвейерного) подхода.

Наиболее эффективно обрабатывать такие данные можно с помощью программируемой систолической структуры (ПСС), состоящей из $m \times m$ вычислительных ячеек (ВЯ)[6]. При выполнении операции пересечения кубов покрытий D^a и D^b :

$$D^a = \left| \begin{matrix} d_{11}^a & \dots & d_{1m}^a \\ \dots & \dots & \dots \\ d_{m1}^a & \dots & d_{mm}^a \end{matrix} \right| , \quad D^b = \left| \begin{matrix} d_{11}^b & \dots & d_{1m}^b \\ \dots & \dots & \dots \\ d_{m1}^b & \dots & d_{mm}^b \end{matrix} \right| .$$

на m горизонтальных входов ПСС поступают соответственно m кубов D^a -покрытия, а на m вертикальных входов ПСС - m кубов D^b -покрытия (рис.1). На i-й горизонтальный вход (j-й вертикальный вход) ПСС поступают компоненты соответствующих кубов d_i^a (d_j^b):

$$d_i^a = \{ d_{i1}^a \quad \dots \quad d_{im}^a \} , \qquad d_j^b = \{ d_{j1}^b \quad \dots \quad d_{jm}^b \} .$$

с задержкой во времени на один цикл относительно поступления куба d_{i-1}^a (d_{j-1}^b) ($i = 1 \div m$, $j = 1 \div m$).

В ячейке (1,1) ПСС в течение m циклов поочередно выполняются следующие элементарные операции пересечения компонент кубов d_i^a и d_j^b :

$$s_{11}^1 = d_{11}^a \cap d_{11}^b , \quad s_{11}^2 = d_{12}^a \cap d_{12}^b , \dots , \quad s_{11}^m = d_{1m}^a \cap d_{1m}^b .$$

где s_{11}^z - результат операции пересечения в z -ом цикле,($z = 1 \div m$).

В ячейке (1,j) первой строки ПСС с задержкой во времени на (j-1) циклов относительно ячейки (1,1) поочередно выполняются следующие операции пересечения:

$$s_{1j}^j = s_{1(j-1)}^{j-1} \cap d_{j1} , \dots , \quad s_{1j}^{j+m-1} = s_{1(j-1)}^{j+m-2} \cap d_{jm} ,$$

где s_{1j}^z - результат операции пересечения в z -ом цикле,($z = 1 \div m$).

s_{1j-1}^{z-1} - результат операции пересечения в соседней,(1,j-1), ячейке в предыдущий цикл времени,($z = j - m + j - 1$).

Через ($2m - 1$) циклов на первом выходе ПСС будет получен результат пересечения первого куба d_1^a D^a -покрытия со всеми кубами D^b -покрытия. В течение последующих (m-1) циклов аналогичным образом на остальных ($m - 1$) выходах ПСС будут поочередно получены результаты пересечений второго d_2^a , … , m-го d_m^a кубов D^a -покрытия с кубами D^b -покрытия.

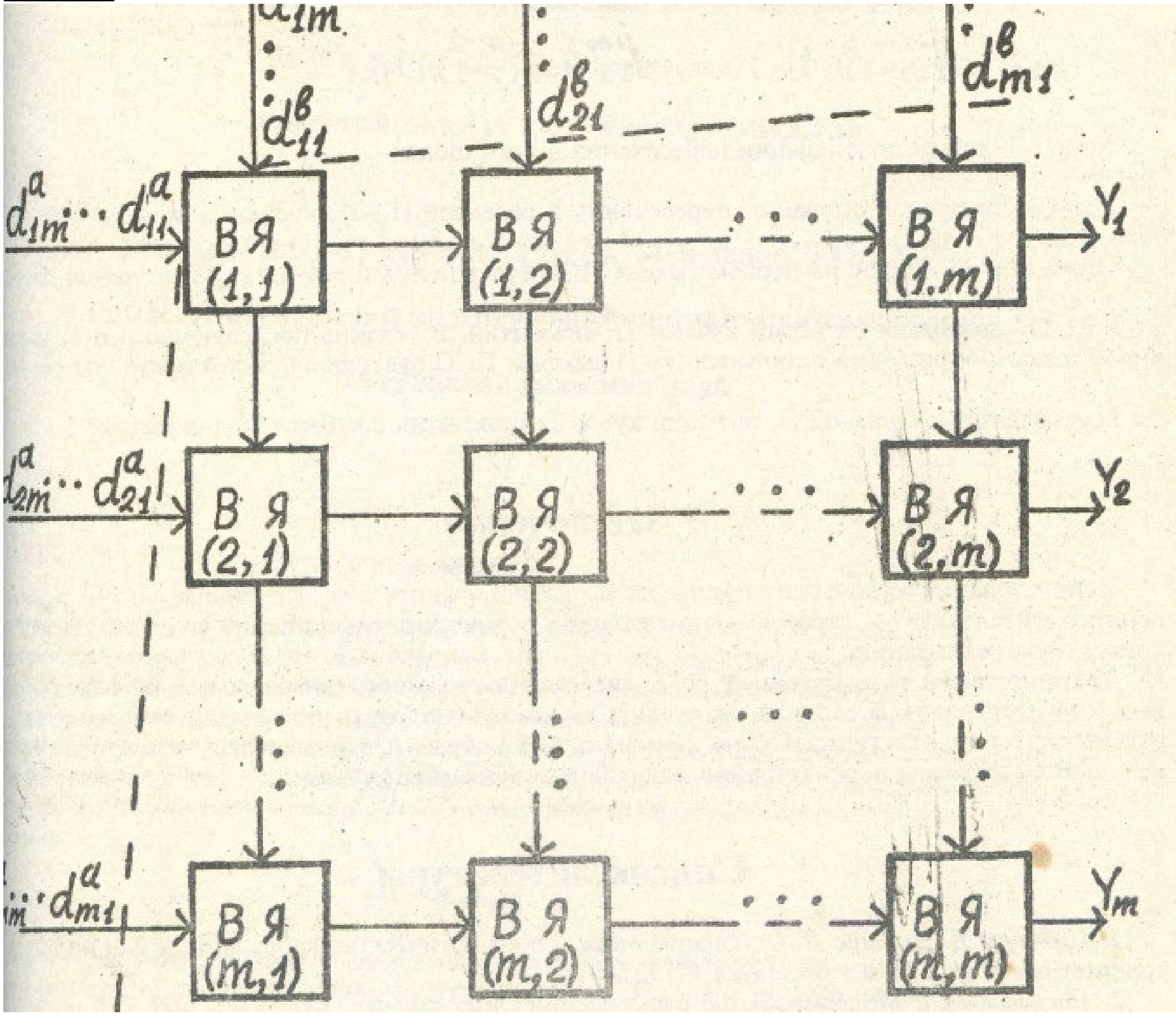

Рис. 1. Схема программируемой систолической структуры

Заключение

Использование кубических покрытий позволяет решить две важнейших задачи искусственного интеллекта - строгую формализацию логических рассуждений и их эффективную аппаратную реализацию.

Достоинствами разработанной ПСС является возможность параллельной обработки данных и универсализм. В отличии от лучших зарубежных программируемых логических интегральных схем, архитектура ПСС не зависит от вида обрабатываемых покрытий, и, поэтому, не возникает проблемы перенастройки для работы с новыми данными.

References

1. Гильберт Д., Бернайс П. Основания математики. Логические исчисления и формализация арифметики: Пер.с нем. - М.: Наука,1979.- 560 с.

2. Поспелов Д.А. Моделирование рассуждений. Опыт анализа мыслительных актов. - М.: Радио и связь, 1989. – 184 с.

3. Логический подход к искусственному интеллекту: от классической логики к логическому программированию: Пер с франц./ Тейз А., Грибомон П., Луи Ж. и др. - М.: Мир,1990. - 432 с.

4. Чень Ч., Ли З. Математическая логика и автоматическое доказательство теорем: Пер. с англ. - М.: Наука, 1983. – 360 с.

5. Смирнов В.А. Логические методы научного познания. М.: Наука, 1987. – 220 с.

6. Семеренко В.П. Систолическая реализация кубических функций // Электронное моделирование. - 1992. - 1. С. 21-25.

7. Поспелов Г.С. Искусственный интеллект - основа новой информационной технологии. - М.: Наука, 1988. – 280 с.