УДК: 51, 512.64

Концепция и сложности реализации тринарной логики

A.Yu. Shcherbakov

The Concept and Difficulties of Implementing Trinary Logic

Abstract. *The article is devoted to the consideration of the problem of trinary logic, in which a third state is introduced, in addition to "yes" and "no," as well as problems of a technical and mathematical nature that arise in this case. Stephen Kleene's trinary logic was corrected and an implementation of trinary calculations using probabilistic polynomials over the field of real numbers was proposed.*

Keywords: *logic, mathematical logic, polynomial, real numbers, field, probability, states, Boolean functions, models, artificial intelligence, artificial consciousness.*

А.Ю. Щербаков
Доктор технических наук, профессор, заведующий кафедрой когнитивно-аналитических и нейро-прикладных технологий Российского государственного социального университета, ведущий научный сотрудник Государственного университета управления.
E-mail: x509@ras.ru

Аннотация. *Статья посвящена рассмотрению задачи тринарной логики, в которой вводится третье состояние, кроме "да" и "нет", а также проблемам технического и математического характера, которые при этом возникают. Скорректирована тринарная логика Стивена Клини и предложена реализация тринарных вычислений с помощью вероятностных полиномов над полем действительных чисел.*

Ключевые слова: *логика, математическая логика, полином, действительные числа, поле, вероятность, состояния, булевы функции, модели, искусственный интеллект, искусственное сознание.*

ВВЕДЕНИЕ

Задача моделирования и воспроизведения человеческого сознания и реализация искусственного интеллекта постоянно сталкивается с ограничениями бинарной логики, которая опирается на два состояния "да" и "нет", моделируемые логическими нулем и единицей, истинностью высказывания или его ложностью.

Попытки моделировать аристотелеву логику с ее "не знаю" или "не уверен" приводят не только к введению третьего состояния, но и к переходу от цифрового к аналоговому или, с точки зрения математики, к множественному или мерному миру.

Например, состояние "не знаю" может оцениваться некоторой глубиной – "насколько не знаю", "знаю о чем-то мало или много".

Попробуем пойти в формулировании приблизительной аксиоматики, годной для компьютерного моделирования, от воспроизведения основных функций булевой алгебры "и" и "или"[1].

Если в базисных логических функциях мы оперируем аргументами 0 и 1, то для тринарных функций целесообразно ввести состояние "u" – "Unknown".

ЛОГИКА СТИВЕНА КЛИНИ

Математик С. Клини предложил пример тринарной логики с состоянием U.

При этом сделано фундаментальное допущение – отрицание U тоже равно U,

НЕ(U)=U. [2]

Тогда мы получим функции с 9-ю возможными состояниями, например, для функции логического "и" и "или" (таблицы 1 и 2).

Таблица 1

Возможные состояния для функции логического «и»

Номер состояния	Первый аргумент	Второй аргумент	Результат
0	0	0	0
1	0	1	0
3	1	0	0
4	1	1	1
5	0	U	0
6	U	0	0
7	1	U	U
8	U	1	U
9	U	U	U

Таблица 2

Возможные состояния для функции логического «или»

Номер состояния	Первый аргумент	Второй аргумент	Результат
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	1
5	0	U	U
6	U	0	U
7	1	U	1
8	U	1	1
9	U	U	U

Несколько труднее в такой схеме ввести понятие отрицания НЕ(U).

Как мы уже заметили, в логике Клини НЕ(U)=U. Но в целом это не вполне корректное допущение. Например, можно создать связь между цифровым и аналоговым миром, так, если U – подмножество Y, то НЕ(U)- дополнение множества U до множества Y. С точки зрения вероятностной меры логично предположить, что p(НЕ(U)) =1-p(U).

Рассмотрим важную функцию "исключающего или", обозначив ее XOR.

xor(x, y)=НЕ(x)&y!x&НЕ(y)

Напомним, что "исключающее или" дает универсальный инструмент для реализации алгоритмов шифрования: если мы прибавляем с использованием функции "исключающее или" некоторую цифровую последовательность к открытому (зашифровываемому) тексту, то, чтобы провести обратную операцию расшифрования, необходимо прибавить ту же последовательность.

Что бы мы хотели получить?

Таблица 3

Возможные состояния для функции «исключающего или»

Номер состояния	Первый аргумент	Второй аргумент	Результат
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	0
5	0	x	x
6	x	0	x
7	1	x	НЕ(x)
8	x	1	НЕ(x)
9	x	x	0

xor(x,1)=НЕ(x)&1!x&0=НЕ(x), а в логике Клини получится x.

Следовательно, мы не можем использовать логику Клини для создания однозначных (обратимых) преобразований, предназначенных в том числе и для шифрования и других необходимых алгоритмов защиты информации.

Таким образом, мы автоматически либо переходим к тетрарной (четырехзначной) логике, либо должны условно "переместиться" в пространство непрерывной меры, когда значение U и НЕ(U) принадлежит интервалу [0,1].

Можно сформулировать принципиальное утверждение для многомерных логик.

УТВЕРЖДЕНИЕ О РАЗМЕРНОСТИ ЛОГИКИ

При построении логик, с размерностью более двух, для реализации биективных (взаимнооднозначных) отображений необходимо включить дополнительное состояние отрицания, что увеличивает размерность логики, либо допускает переход в пространство условно непрерывных состояний.

Интересно рассмотреть перенос булевых функций в вероятностные, которые позволили бы описать при помощи непрерывного полинома над полем действительных чисел разные ситуации, связанные в том числе с распределением вероятности различных состояний, включая неопределенное U.

Например, функция "И" задается следующим вероятностным полиномом.

P(И, x1,x2)=P(x1)xP(x2), где

P(И) – вероятность появления логической единицы,

P(x1) – вероятность появления логической единицы для первого аргумента,

P(x2) – вероятность появления логической единицы для второго аргумента.

Функция ИЛИ

P(И, x1, x2)=P(x1)+P(x2)-P(x1)xP(x2)

Функция НЕ

P(НЕ, x)=1-P(x).

Обозначим P(x1)=P1 и P(x2)=P2.

Тогда P(XOR, x1, x2)=P1+P2-3P1P2+P1^{2}P2+P1P2^2--(P1P2)^2

Исходя из полученной формулы интересно пронаблюдать, что "исключающее или" обладает свойством статистического выравнивания – если любой из аргументов 0.5, т.е. вероятность появления единицы и нуля равны, то вероятность появления единицы "на выходе" не зависит от второго аргумента.

Таким образом, задав, например U=0.25, получаем

P(НЕ, 0.25)=0.75

P(И,1, 0.25)=0.25

P(ИЛИ, 1, 0.25) = 1+0.25-0.25=1

P(XOR, 1, 0.25) = 1+0.25-3x0.25+1^{2}x0.25+1x(0.25)^2--(1x0.25)^2 = 0.75, что соответствует таблице 3 и соотносится с предложенной логикой.

ЗАКЛЮЧЕНИЕ

Применение тринарной логики требует корректной реализации состояния НЕ(x), что автоматически переводит вычислитель в тетрарное состояние.

Разработка многомерных логик возможна с использованием вычислений в действительных числах, в том числе с трансформацией булевых функций в вероятностные полиномы.

СПИСОК ЛИТЕРАТУРЫ

1. Халамайзер А.Я. Математика гарантирует выигрыш. М: Московский рабочий, 1981. – 248 с.
2. Клини С.К. Математическая логика.- М.: Издательская группа URSS. Серия: Физико-математическое наследие: математика (основания математики и логика). Пер. с англ. Изд. 4. 2008. 480 с.