

Архитектура параллельных вычислительных систем



К.Т.Н., доцент

**Костичев Сергей
Валентинович**

**Альтернативные
вычислительные
системы**

snenv@mail.ru

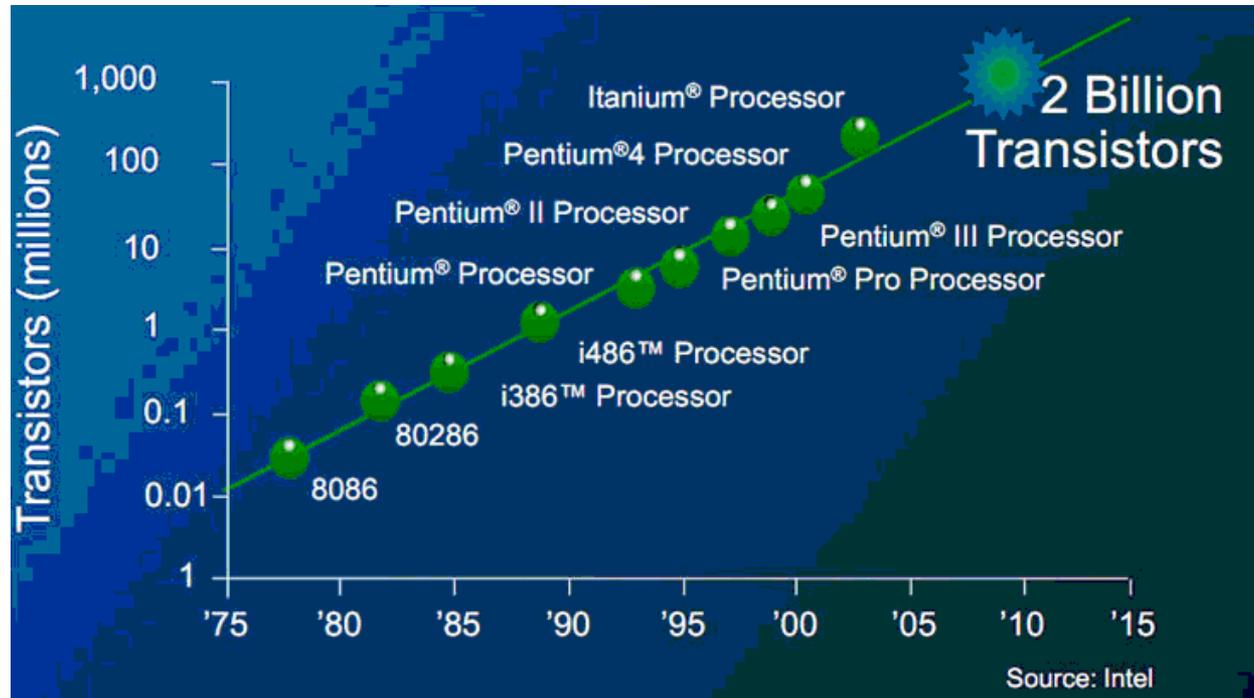
Учебные вопросы:

- 1. Оптические вычислительные системы.**
- 2. Квантовые компьютеры.**
- 3. ДНК-процессоры,**
- 4. Коммуникационные процессоры**
- 5. Процессоры баз данных**
- 6. Нейропроцессоры**
- 7. Процессоры нечеткой логики.**



Развитие ВТ (физические способы реализации логических алгоритмов)

- механические устройства (вычислительная машина Бэббиджа)
- ламповые (компьютеры 40-50-х годов Марк I и Марк II)
- транзисторные
- интегральные схемы



Производительность процессоров растет в основном только за счет параллелизации вычислений.

Достижения пределов применения полупроводниковых технологий -> кремниевой технологии начинают искать замену -> Появление ВУ работающих на совершенно ином принципе.

Возможные альтернативы замены современных компьютеров :

- Оптические компьютеры
- Квантовые компьютеры
- ДНК-процессоры,
- Нейропроцессоры
- Процессоры нечеткой логики



1. Оптические компьютеры



Использование оптических методов в ВТ ведется **по 3-м основным направлениям.**

- использование **аналоговых интерференционных оптических вычислений** для решения отдельных специальных задач
- создание** чисто **оптических** или гибридных (оптоэлектронных) **соединений** вместо обычных, менее надежных, электрических соединений
- создание компьютера**, полностью состоящего из оптических устройств обработки информации (MTI, Sandia Laboratories, Intel, IBM)

Электрон хорошо, но фотон – лучше.



Преимущества **применение оптического излучения в качестве носителя информации** по сравнению с электрическими сигналами:

- световые потоки, в отличие от электрических, **могут пересекаться друг с другом**;
- световые потоки могут быть **локализованы в поперечном направлении до нанометровых размеров** и передаваться по свободному пространству;
- **скорость распространения** светового сигнала выше скорости электрического;
- **взаимодействие световых потоков с нелинейными средами распределено по всей среде**; это дает новые степени свободы (по сравнению с электронными системами) в организации связи и создании параллельных архитектур.



Преимущества оптических вычислений

- информация в виде светового луча передается с **большой скоростью** и на **большие расстояния** при **минимальных потерях**
- обработка такой информации осуществляется во время ее передачи и без энергозатрат
- **высокая степень параллелизации** при передаче и обработке данных (за счет одновременной работы с волнами разной длины)
- **меньшее энергопотребление,**
- **отсутствие электрических наводок и сложность в перехвате данных** (так как в окружающее пространство ничего не излучается)



Не проблема — сделать оптические аналоги базовых пп элементов

Проблема — заставить все это работать правильно и быстро.

Построение оптического компьютера - отказ от архитектуры фон Неймана и от принципа двоичного кодирования.

Эффективней будет работать с двумерными изображениями (прирост производительности 10-20 раз). Но как работать ?

Ближайшее будущее – объединение привычной пп электроники с оптическими устройствами

Первые решения с 1990 года.

Основа - явление оптической бистабильности.

Оптическая бистабильность - это одно из проявлений взаимодействия света с веществом. При этом определенной интенсивности и поляризации падающего на вещество излучения соответствуют **два возможных стационарных состояния световой волны**. Это явление аналогично обычной петле магнитного гистерезиса.



- Набор оптических логических устройств реализуется на основе пассивных нелинейных **резонаторов-интерферометров**. На практике они обладают временами переключения $\sim 10^{-8}$ - 10^{-9} с.
- Минимальный размер оптического элемента памяти определяется **минимально необходимым числом атомов**, для которого устойчиво наблюдается оптическая бистабильность. Это ~ 1000 атомов, что соответствует 1-10 нанометрам.
- Время переключения оптических элементов памяти - доли пикосекунды, это на 3 порядка ниже, чем у пп элементов.
- **Созданы отдельные компоненты** (оптические процессоры, ячейки памяти), но до полной сборки еще далеко.
- **Основная проблема** - синхронизация работы отдельных элементов оптического компьютера в единой системе.



- **Основное достоинство оптических компьютеров - создание большего количества параллельных архитектур, по сравнению с пп компьютерами.**
- Цифровой оптический процессор с числом параллельных каналов $\sim 10^5$ - 10^6 может совершать до 10^{13} - 10^{15} операций в секунду (при времени переключения в одном канале $\sim 10^{-8}$ - 10^{-9} с), что значительно выше чем в современных пп процессорах.



Сейчас есть только один коммерческий продукт – гибридный чип **EnLight256** (компания Lenslet, Израиль)



Характеристики

- до 8×10^{12} (8 тераоп) операций в секунду
- обработка до 15 потоков HD-видео в реальном времени .
- данные одновременно поступают от 256 оптических входов — лучи 256 лазеров складываются или перемножаются, когда освещают специальную матрицу (пространственно-световой модулятор 256x256);
- размеры процессора– 15x15 см.
- выходные оптические сигналы результата вычислений считываются массивом из 256 световых детекторов.

Это **первый оптический DSP** , превосходящий в 3 раза лучшие электронные DSP.

EnLight256 – это **гибридный** оптический процессор. Ядро этого процессора – оптическое, а входная и выходная информация представляется в электронном виде.

Причины:

- сегодня создать оптический компьютер очень дорого
- неизвестно, как он будет работать

Достоинство: меняется только ядро (все остальное остается электрическим) и получается огромный прирост производительности.

Из некоммерческих проектов существуют опытные образцы оптических процессоров, выполненных по технологии 65 и 32нм, но и с ними не так все просто.



2. Квантовые компьютеры



- Ричард Фейнман: «На обычных компьютерах принципиально **невозможно точно рассчитать реальную физическую систему**» (1981г). Предложил использовать для расчетов физических систем квантовые компьютеры.
- **Квантовые вычисления** — это вычислительная модель, которая отличается от модели Тьюринга и фон Неймана, и предполагается, что для некоторых задач она является более эффективной.
- Квантовая вычислительная модель основана на нескольких простых правилах преобразования входной информации. Они обеспечивают массовую параллелизацию ВП. **Можно одновременно вычислить значение функции для всех её аргументов** (это будет единственный вызов функции).



- Основа квантовой вычислительной модели - понятие **кубита**. Это аналог бита в классической теории информации. Но **кубит может одновременно принимать несколько значений**. Он находится в **суперпозиции** своих состояний. Значение кубита - линейная комбинация его **базовых состояний**, и коэффициенты при базовых состояниях являются комплексными числами.
- **Базовыми состояниями** являются значения 0 и 1 (в квантовых вычислениях обозначаются $|0\rangle$ и $|1\rangle$).
- Суперпозиция одного кубита записывается как $A|0\rangle + B|1\rangle$, где A и B — некоторые комплексные числа, $|A|^2 + |B|^2 = 1$



Кубит может принимать:

1) $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

2) $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

3) любое возможное
сочетание
векторов $|0\rangle$ и $|1\rangle$

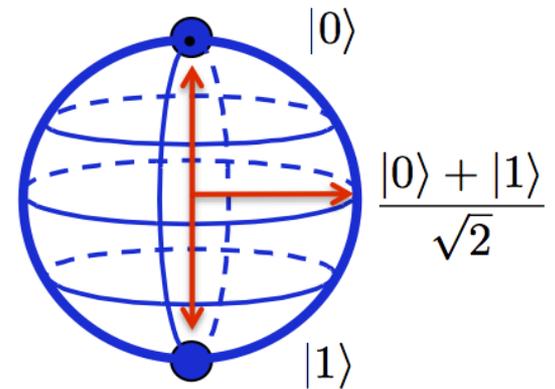


0



1

Classical Bit



Qubit

■ Для 2-х кубитов возможны 4 значения: 00, 01, 10 и 11. Два кубита представляют собой суперпозицию 4-х базовых значений: $A|00\rangle + B|01\rangle + C|10\rangle + D|11\rangle$.

Три кубита представляют собой суперпозицию 8 базовых значений.

Квантовый регистр из N кубитов одновременно хранит в себе 2^N комплексных чисел.



- Т.к. входные данные представляют собой суперпозицию всех возможных значений входного аргумента, функция должна принять такую суперпозицию и обработать её.

В рамках модели квантовых вычислений функция – это матрица с ограничением - **эрмитова** матрица.

Свойство эрмитовой матрицы: при умножении матрицы на свою эрмитово-сопряжённую, должна получиться единичная матрица.

Эрмитово-сопряжённая матрица получается при помощи транспонирования исходной матрицы и заменой всех её элементов на их комплексно-сопряжённые.

Доказано

- любую функцию можно специальным образом преобразовать в эрмитову матрицу.
- любую эрмитову матрицу можно представить посредством тензорного произведения небольшого набора базисных матриц, представляющих базисные логические операции



- В модели квантовых вычислений есть ещё одна операция - **измерение**, Можно измерить вектор и получить из него конкретное значение кубита. Т.е. суперпозиция преобразовывается в конкретное значение. Вероятность получения того или иного значения равна квадрату модуля комплекснозначного коэффициента.
- **Квантовый алгоритм** представляет собой умножение матрицы на вектор. В результате получается новый вектор. Числа в векторах и матрицах являются комплексными
- Имея N кубитов можно одновременно обработать 2^N комплексных чисел. В выходном векторе будут результаты обработки всех этих чисел одновременно. Это **достоинство** модели квантовых вычислений. Но получить можно только одно значение (путем **измерения**), и оно каждый раз различное в зависимости от распределения вероятностей. Это **ограничение** модели квантовых вычислений.



□ **Суть квантового алгоритма.**

- Создаётся равновероятностная суперпозиция всех возможных значений входного параметра.
- Эта суперпозиция подаётся на вход функции.
- По результатам её выполнения делается вывод о свойствах этой функции.

Мы не можем получить все результаты, но можем сделать выводы о свойствах функции.

Квантовые компьютеры - это не замена обычным. Они быстрее только в вычислениях, где есть возможность использовать все доступные суперпозиции.



Сейчас **разработан ряд квантовых алгоритмов, которые решают прикладные задачи:**

- задачи из области криптографии (компрометация различных криптографических систем и протоколов).
- типовые математические задачи на графах и матрицах, при этом такие задачи имеют очень большую область применения.
- ряд алгоритмов аппроксимации и эмуляции, которые используют аналоговую составляющую модели квантовых вычислений.



- Наиболее известным квантовым алгоритмом является **алгоритм Шора** (1997): решение задачи разложения чисел на простые множители (**задача факторизации**)
 $15 = 3 \cdot 5$, $55 = 5 \cdot 11$ – элементарно
13297 или 99487? Или 100 знаков?

Многие современные протоколы обеспечивают защищенную передачу данных, используя вычислительную сложность этой задачи для генерации секретного ключа.

Квантовые компьютеры, исполняя алгоритм Шора, должны справляться с этой задачей за считанные секунды (в теории). Тогда такие системы перестанут быть секретными и безопасными.



Проблема: как воплотить кубит в реальность?

Необходимо найти физическую систему с двумя квантовыми уровнями, которые можно использовать в качестве базовых состояний кубита — нуля и единицы.

Фейнман: использовать закрученные в разные стороны фотоны
1995г – 1-й экспериментально созданный кубит (ионы)

Другие физические реализации: ядра атомов, электроны, фотоны, дефекты в кристаллах, сверхпроводящие цепи.

- Основные соревновательные **параметры кубитов**:
 - время их жизни
 - количество кубитов, которые можно заставить работать вместе (**проблема** - кубиты не любят соседствовать друг с другом). Сегодня ученым удалось заставить работать вместе максимум 50-70 кубитов



Отличия от традиционных компьютеров

- Вместо традиционных битов – квантовые биты (кубиты)
- N-разрядный регистр хранит 2^N значений одновременно (квантовая суперпозиция)
- Возможность параллельной работы со всеми значениями одновременно (квантовый параллелизм)
- Процент ошибок вычислений выше требуемого
- Сложное строение и принцип работы



Достоинства

- Высокая производительность
- Использование квантовых регистров многократно повышает объем памяти
- Параллельная работа с данными

Недостатки

- Технология, необходимая для реализации квантовых компьютеров, еще не разработана
- Высокий процент ошибок



Выводы

- ❑ Полноценный универсальный квантовый компьютер является пока **гипотетическим устройством**, сама возможность построения которого связана с серьёзным развитием квантовой теории в области многих частиц и сложных экспериментов.
- ❑ По прогнозу исследователей из компании Cisco Systems, полноценный квантовый компьютер должен появиться к середине 2020-х годов.
- ❑ Для полноценной работы, скорее всего, придется охлаждать модули на сотни градусов по Цельсию, что значительно ограничивает область использования квантовых компьютеров.



3. ДНК-процессоры



Биокомпьютинг - гибрид информационных, молекулярных технологий и биохимии.

Наибольшее распространение - в качестве основного элемента (процессора) используются **молекулы** дезоксирибонуклеиновой кислоты (**ДНК**).

Также в качестве био-процессора могут использоваться белковые молекулы и биологические мембраны.

ДНК-процессор характеризуется **структурой** и **набором команд**.

В этом случае:

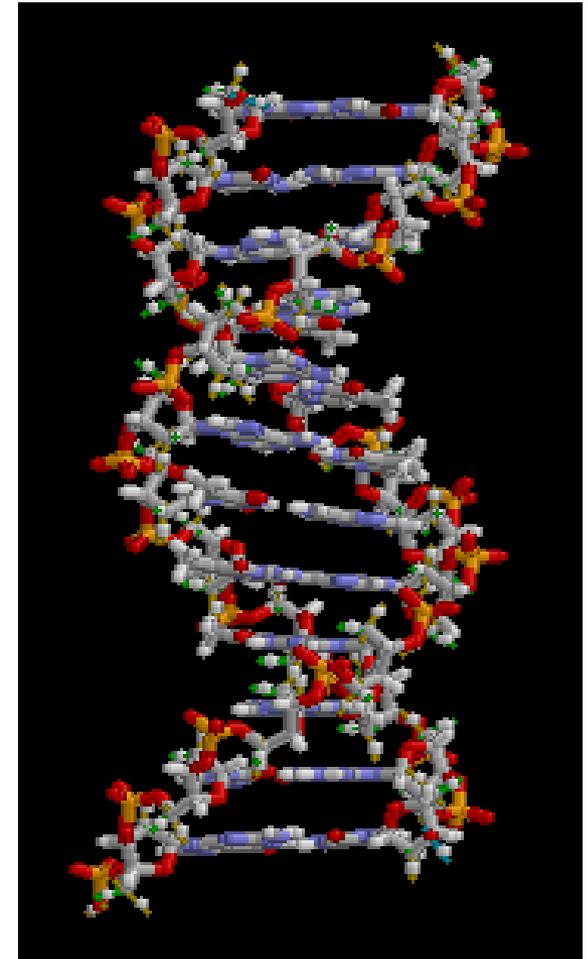
- структура процессора – это структура молекулы ДНК.
- набор команд – это перечень биохимических операций с молекулами.

Принцип устройства **компьютерной ДНК-памяти** основан на последовательном соединении четырех нуклеотидов: аденин, гуанин, тимин, цитозин



Основные этапы ДНК-процессоров

- Теоретическое обоснование возможности – середина 50-х годов (Р.Фейман)
- Проработка теории в деталях – 70-е (Ч.Бенетт), 80-е (М.Конрад)
- Первый компьютер на базе ДНК – 1994 г. (Леонард Адлеман)
- И.Шапиро (Вейцмановский институт естественных наук) – первые модели биокомпьютеров – 1999, 2001 г.
- Фирма Olympus Optical объявляет о первой коммерческой версии ДНК-компьютера – 2002 г.
- Начало работы IBM в области создания ДНК-компьютера – 2009 г.



Возможные области применения ДНК-компьютера

- Мощные ВС высокой степени параллелизма
- Нанофабрика лекарств, фармакология
- Биотехнологии, медицина
- ВС со сверхмалым потреблением энергии



Достоинства ДНК-компьютера

- более простая технология изготовления (не требует для своей реализации жестких условий, как при производстве полупроводников);
- использование тернарного кода (информация кодируется тройками нуклеотидов); это позволяет за меньшее количество шагов перебрать большее число вариантов при анализе сложных систем;
- потенциально высокая производительность (до 10^{14} оп/с) за счет одновременного вступления в реакцию триллионов молекул ДНК;
- возможность хранения данных с плотностью, в триллионы раз превышающей показатели оптических дисков;
- низкое энергопотребление.



Недостатки ДНК-компьютера

- сложность со считыванием результатов – современные способы определения кодирующей последовательности несовершенны, сложны, трудоемки и дороги;
- низкая точность вычислений, связанная с возникновением мутаций, прилипанием молекул к стенкам сосудов и т.д.;
- невозможность длительного хранения результатов вычислений в связи с распадом ДНК в течение времени.



4. Коммуникационные процессоры



- ❑ Это нечто среднее между жесткими специализированными ИС и гибкими процессорами общего назначения.
- ❑ Особенности:
 - построены с учетом сетевых задач,
 - оптимизированы для сетевой работы
 - позволяют освободить вычислительный процессор от нагрузки, связанной с передачей сообщений между процессорными узлами.
- ❑ Коммуникационный процессор
 - имеет собственную память
 - оснащен высокоскоростными внешними каналами для соединения с другими процессорными узлами.
- ❑ Коммуникационные процессоры программируются, как и ПК-процессоры.



5. Процессоры баз данных



- Это программно-аппаратные комплексы, предназначенные для выполнения всех или некоторых функций систем управления базами данных (СУБД).

- Процессоры баз данных
 - выполняют функции управления и распространения,
 - обеспечивают дистанционный доступ к информации через шлюзы,
 - обеспечивают репликацию обновленных данных с помощью различных механизмов тиражирования.

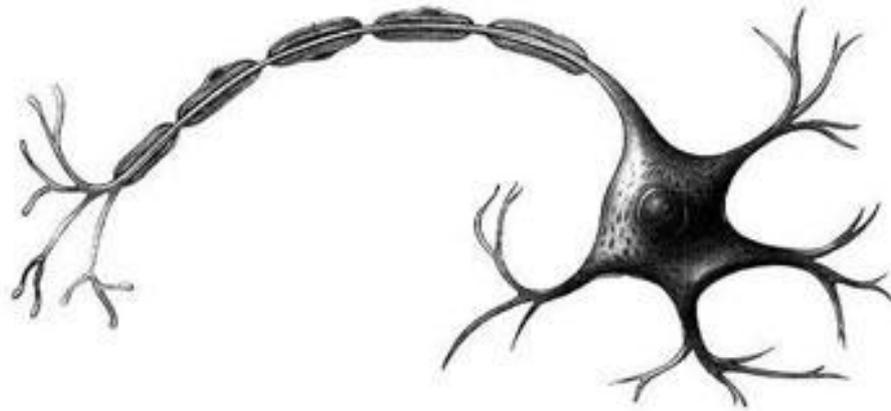
- Наметился переход от тривиальной архитектуры клиент – сервер к трехуровневой архитектуре с распределенными базами данных:
 - клиент,
 - сервер с СУБД,
 - серверы собственно с данными.



6. Нейронные процессоры



Нейрокомпьютинг - это научное направление, занимающееся разработкой вычислительных систем 6-го поколения - нейрокомпьютеров, которые состоят из большого числа параллельно работающих простых вычислительных элементов (**нейронов**).
Основа – принципы обработки информации, заложенные в искусственных нейронных сетях (ИНС).

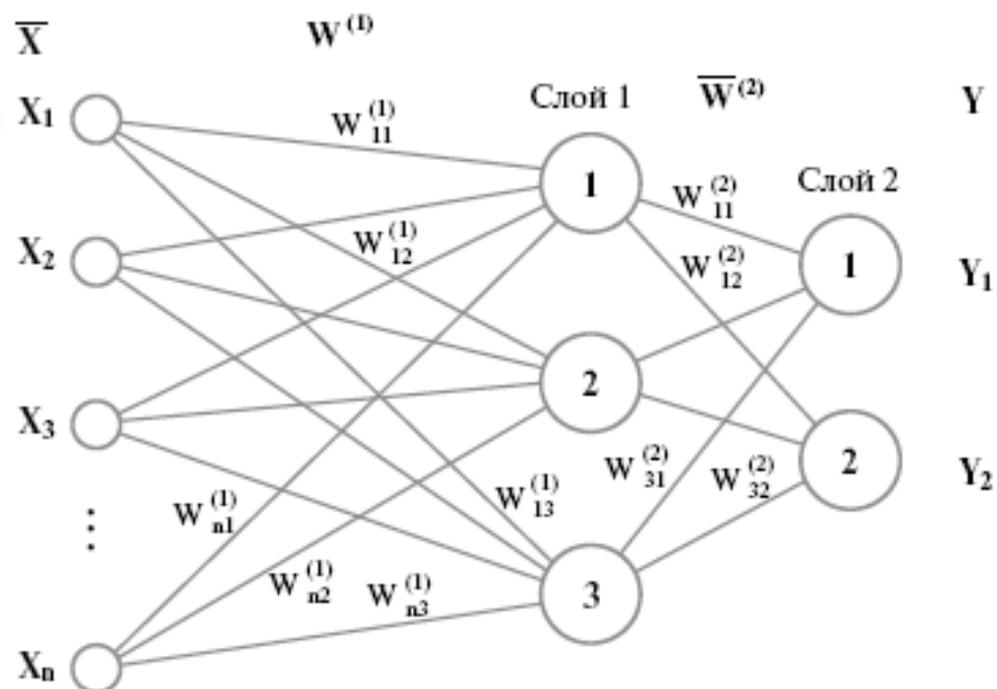


Элементы связаны между собой, образуя нейронную сеть.

Нейронная сеть - совокупность нейронов, которые определенным образом соединены друг с другом каналами обмена информации - «синаптическими связями».

Нейроны выполняют **единообразные** вычислительные действия и **не требуют внешнего управления**.

Большое число параллельно работающих нейронов обеспечивают высокое быстродействие.



Основным элементом проектирования сети является ее **обучение**. При обучении и переобучении НС ее весовые коэффициенты изменяются. При функционировании НС они остаются постоянными, формируя долговременную память.

Нейрон – элементарный процессор, характеризующийся

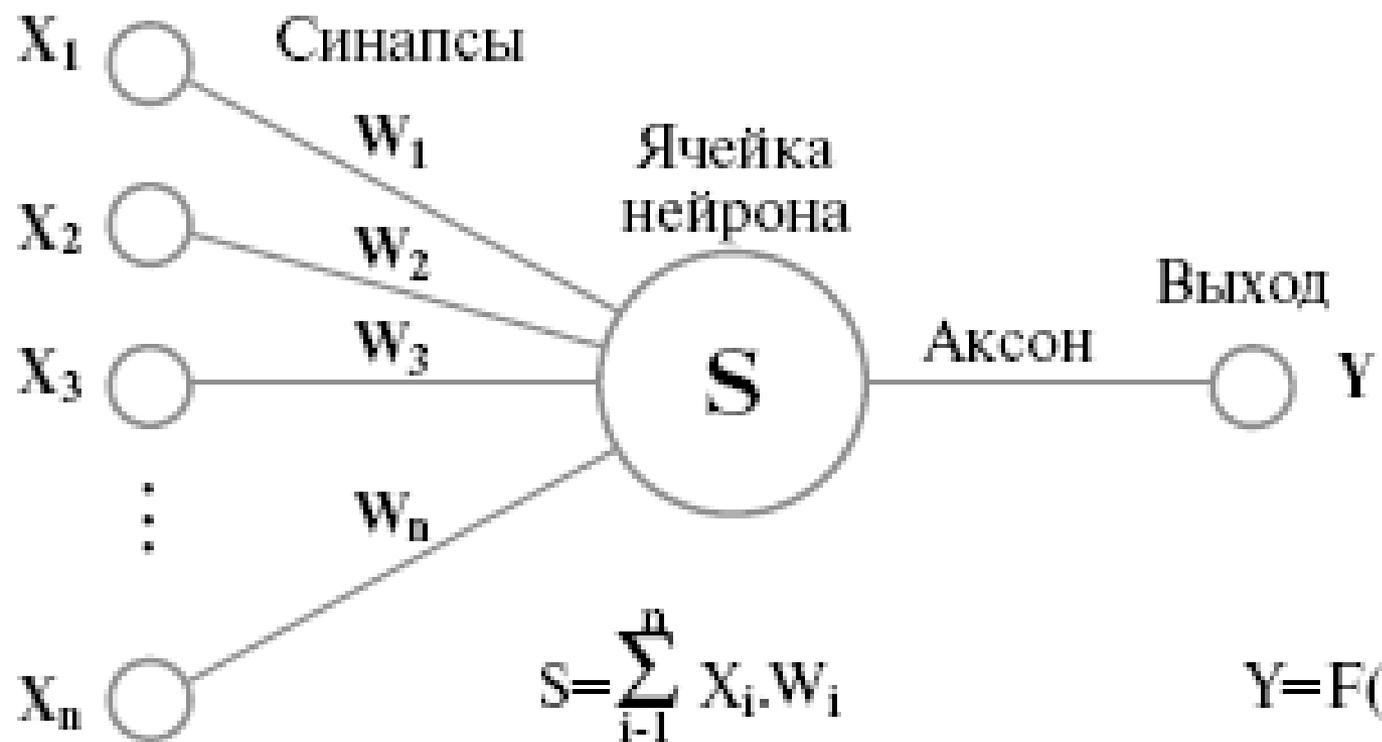
- входным состоянием,
- выходным состоянием,
- передаточной функцией (функция активации),
- локальной памятью

Состояния нейронов изменяются в процессе функционирования и составляют **кратковременную** память нейросети.

Каждый нейрон вычисляет взвешенную сумму пришедших к нему по синапсам сигналов и производит над ней нелинейное преобразование.



Входы



Классы задач :

- решения так называемых плохо формализуемых задач (распознавание образов)
- обработка сигналов, изображений и т.д.
- общематематические задачи, решаемые в нейросетевом логическом базисе

7. Процессоры с нечеткой логикой



Терема FAT (Fuzzy Approximation Theorem) (Коско) –

“любая математическая система может быть

аппроксимирована системой, основанной на нечеткой логике ”



- ❑ **Нечеткая логика** – это логическая или управляющая система n -значной логической системы, которая использует степени состояния («степени правды») входов и формирует выходы, зависящие от состояний входов и скорости изменения этих состояний.
- ❑ Это не обычная «истинная или ложная» (1 или 0), булева (двоичная) логика, Она обеспечивает основы для приблизительного рассуждения с использованием неточных решений и позволяет использовать лингвистические переменные.
- ❑ Нечеткая логика решает те же задачи, что и двоичная логика, но часто она удобнее, производительнее и дешевле.

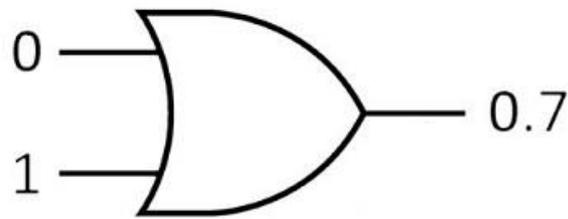


Достоинство нечеткой логики - простота содержательного представления.

Она позволяет

- упростить проблему,
- представить проблему в более доступном виде
- повысить производительность системы.

Схема разницы между нечеткой и булевой логикой

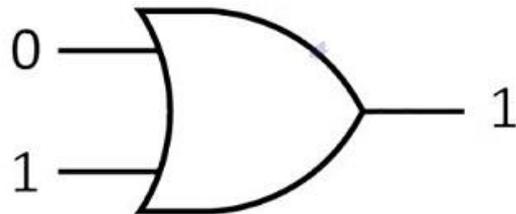


Нечеткая логика



Четкая логика: да или нет (1, 0).

Нечеткая логика: конечно, да; возможно, нет; не могу сказать; возможно да и т.д.



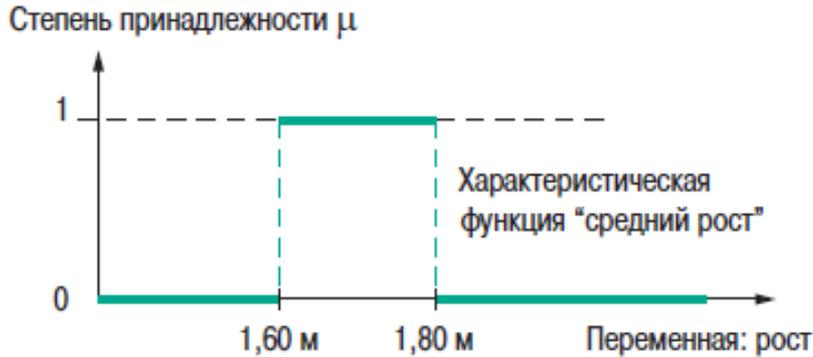
Булева логика



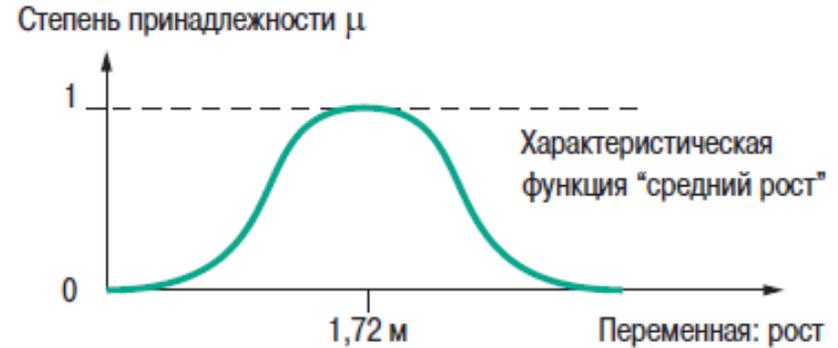
Нечеткая логика была разработана в 1965 году профессором Лотфи Заде в Калифорнийском университете в Беркли.

Функция принадлежности

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

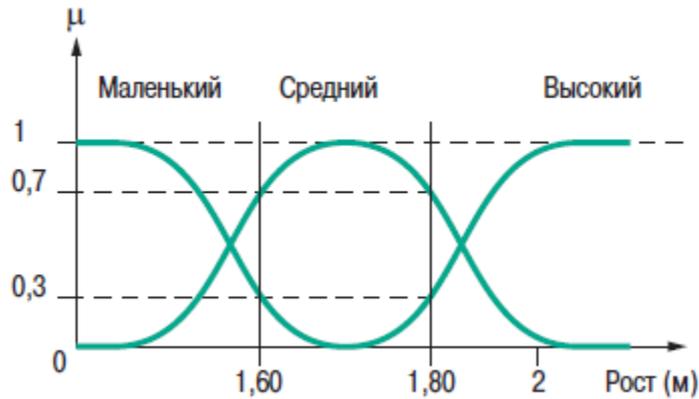


Характеристическая функция



Функция принадлежности

Несколько нечетких множеств можно определить **через одну переменную, которая принимает нескольких значений**, например множества “маленький рост”, “средний рост” и “высокий рост”. Каждое понятие описывает функция принадлежности.



Такая переменная называется **лингвистической переменной**.

Человека ростом 1,80 м можно отнести к группе “высокий рост” со степенью принадлежности 0,3 и к группе “средний рост” со степенью принадлежности 0,7.

В классической логике переход от среднего роста к высокому мгновенен.

Принцип решения задачи с помощью нечеткой логики



- **Фазифаер** (Fuzzifier или оператор размытия) - численные данные (показания измерительных приборов, результаты анкетирования) фаззируются (переводятся в нечеткий формат); Входной сигнал может принимать множество значений.

- **Контроллер**. Он состоит из базы знаний и механизма вывода. База знаний хранит функции принадлежности и нечеткие правила, полученные путем знания работы системы в среде. Входной сигнал обрабатывается по определенным правилам;

- **Дефазифаер** (оператор восстановления чёткости) - данные дефаззируются и в виде привычной информации подаются на выход. Выполняется обратный процесс фазифаера.



Пример

Разработаем простую систему нечеткого управления для управления работой стиральной машины, так чтобы нечеткая система контролировала время стирки.

Входные параметры:

степень загрязнения (качество грязи)
тип грязи.

Шаг №1 - Определим лингвистические переменные:

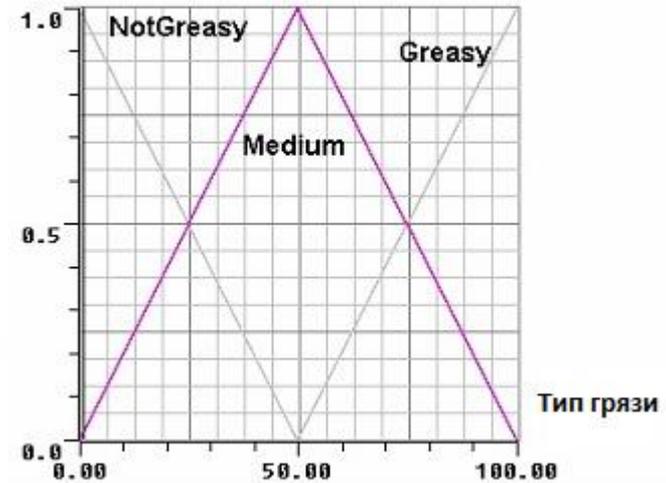
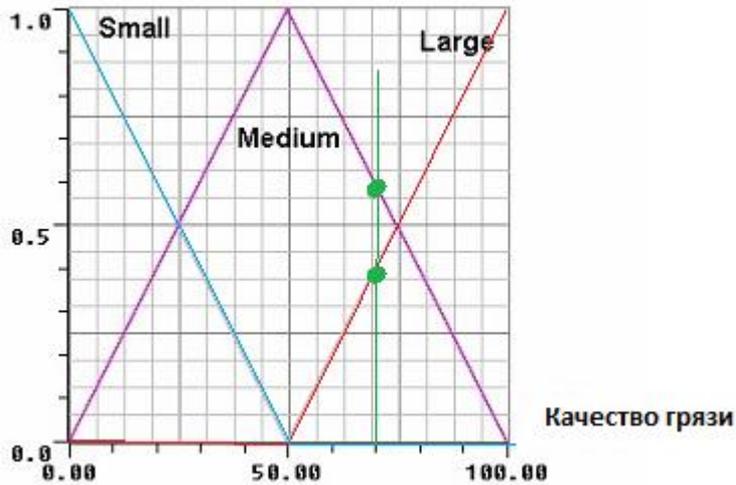
Тип грязи: {Greasy, Medium, Not Greasy} (жирное, среднее, не жирное)

Качество грязи: {Large, Medium, Small} (высокое, среднее, незначительное)

Время стирки: {Short, Very Short, Long, Medium, Very Long}



Шаг №2: Построим функции принадлежности (фазификация):



Как видно из графика, качество грязи уровня 70 со степенью принадлежности 0,35 относится к значению «Large» и со степенью принадлежности 0,65 к значению «Medium»

Шаг №3: Определим лингвистические правила для базы знаний (механизм вывода):

- IF качество грязи Small И Тип грязи Greasy, THEN Время стирки Long.
- IF качество грязи Medium И Тип грязи Greasy, THEN Время стирки Long.
- IF качество грязи Large и тип грязи Greasy, THEN Время стирки Very Long.
- IF качество грязи Small И Тип грязи Medium, THEN Время стирки Medium.
- IF качество грязи Medium И Тип грязи Medium, THEN Время стирки Medium.
- IF качество грязи Large и тип грязи Medium, THEN Время стирки Medium.
- IF качество грязи Small и тип грязи Non-Greasy, THEN Время стирки Very Short.
- IF качество грязи Medium И Тип грязи Non-Greasy, THEN Время стирки Medium.
- IF качество грязи Large и тип грязи Non-Greasy, THEN Время стирки Very Long.

Результат: время стирки



Шаг №4: Дефазификация

Выполняется обратное преобразование из нечётких значений выходных параметров – к чётким цифрам.

Математические алгоритмы этих преобразований разнообразны и зависят от конкретной задачи



Области применения

- Автоматизация;
- Искусственный интеллект;
- Нейронные сети;
- Распознавание образов, речи, изображений и музыки;
- Прогнозирование и предсказания

Разработанные на ее основе специализированные аппаратные решения (**fuzzy-вычислители**) позволяют получить реальные преимущества в быстродействии.

Если каскадировать fuzzy-вычислители, то получится один из вариантов **нейропроцессора** или нейронной сети - термин «neuro-fuzzy logic».



Нечёткую логику следует применять, когда необходимо работать с **большим количеством входных данных** и делать на их основе новые заключения и выводы.

Системы с чёткой логикой больше подходят для указания специфики и алгоритма для поиска решений.



Микроконтроллеры, поддерживающие нечеткую логику

- 68HC11, 68HC12 фирмы Motorola,
- MCS-96 фирмы Intel
- HEPE98
- W.A.R.P.(Weight Associative Rule Processor)
- фирмы SGS-THOMSON Microelectronics



Вопросы?

