Информатика
Предмет информатики

Современную информатику рассматривают как:

1) **фундаментальную естественную междисциплинарную науку**

научная методология разработки информационного обеспечения процессов управления материальными объектами и интеллектуальными процессами любой природы

2) прикладную дисциплину

Создание технологий и устройств обработки и передачи информации

3) Отрасль промышленного производства

Потребить, создать и т.д. информацию

Основные направления информатики

Теоретическая информатика - математическая дисциплина включающая такие дисциплины и направления как:

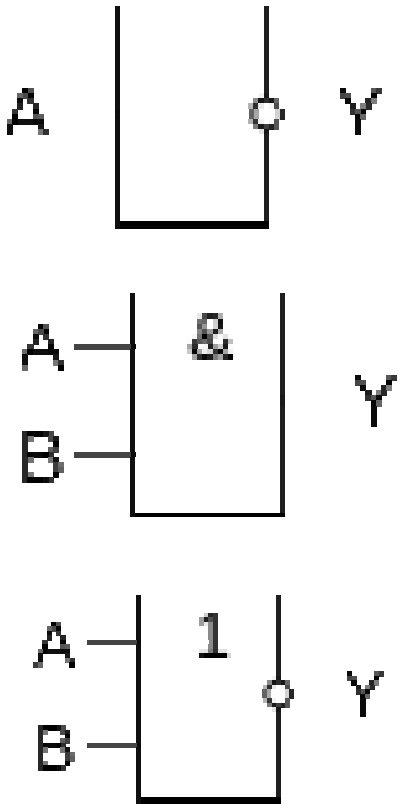
Дискретная математика, логика высказываний, логика предикатов, теория нечетких множеств, теория алгоритмов, теория параллельных вычислений, теория автоматов, теория сетей Петри, машина Тьюринга

- вычислительная математика и вычислительная геометрия, методы оптимизации
- теория кодирования, изучение свойств информации, передача информации по различным каналам связи и особенности передачи информации, сигналы
- системный анализ, моделирование, имитационное моделирование, теория массового обслуживания
- теория принятия решений, теория игр, исследование операций, математическое программирование

Дискретная математика

- Ложь, истина, бинарная логика
- Операции

И, или, не, возможность представить любую логическую формулу посредством данных операций (СКНФ, СДНФ)



логика высказываний, логика предикатов

- Выражения и атомы которые могут принимать значения истина или ложь связываются стандартными логическими операциями (и, или, не, исключающее или)
- В логике предикатов появляются кванторы существования и всеобщности.

$$\exists, \forall$$

Все люди смертны:

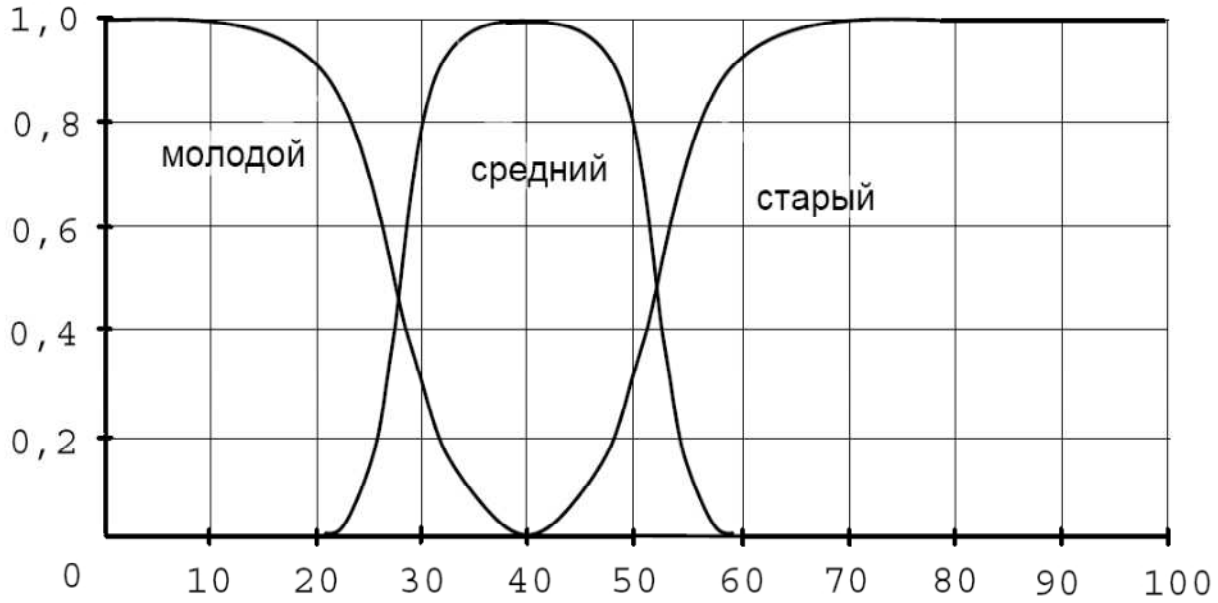
$$\forall x Человек(x) \rightarrow Смертен(x)$$

теория нечетких множеств

- Часто понятия которыми оперирует человек нечеткие:

Мягкий, быстрый, высокий, умный, красивый, медленный, темный, светлый, и так далее..

Вводится специальная функция которая показывает с какой силой какой-то элемент множества принадлежит данному понятию и соответствующие операции на нечетких множествах.



- Сложность алгоритмов и разрешимость

В рамках классической теории осуществляется классификация задач по классам сложности (P-сложные, NP-сложные, экспоненциально сложные и др.). К классу P относятся задачи, которые могут быть решены за время, полиномиально зависящее от объёма исходных данных, с помощью детерминированной вычислительной машины (например, машины Тьюринга), а к классу NP — задачи, которые могут быть решены за полиномиально выраженное время с помощью недетерминированной вычислительной машины, то есть машины, следующее состояние которой не всегда однозначно определяется предыдущими.

Машина Тьюринга

- Абстрактный исполнитель (абстрактная вычислительная машина). Была предложена Аланом Тьюрингом в 1936 году для формализации понятия алгоритма.
- Машина Тьюринга представляет собой простейшую вычислительную машину с линейной памятью, которая согласно формальным правилам преобразует входные данные с помощью последовательности элементарных действий.
-
- Элементарность действий заключается в том, что действие меняет лишь небольшой фрагмент данных в памяти (в случае машины Тьюринга — лишь одну ячейку), и число возможных действий не бесконечно. Несмотря на простоту машины Тьюринга, на ней можно вычислить всё, что можно вычислить на любой другой машине, осуществляющей вычисления с помощью последовательности элементарных действий. Это свойство называется полнотой.

Принцип работы

Конкретная машина Тьюринга задаётся перечислением элементов множества букв алфавита A, множества состояний Q и набором правил, по которым работает машина.

Они имеют вид:

$q_i a_j \rightarrow q_{i1} a_{j1} d_k$

(если головка находится в состоянии q_i, а в обозреваемой ячейке записана буква a_j, то головка переходит в состояние q_{i1}, в ячейку вместо a_j записывается a_{j1}, головка делает движение d_k, которое имеет три варианта: на ячейку влево (L), на ячейку вправо (R), остаться на месте (N)). Для каждой возможной конфигурации

$<q_i, a_j>$ имеется ровно одно правило (для недетерминированной машины Тьюринга может быть большее количество правил). Правил нет только для заключительного состояния, попав в которое, машина останавливается. Кроме того, необходимо указать конечное и начальное состояния, начальную конфигурацию на ленте и расположение головки машины.

Направления информатики

- **Кибернетика**
- **Программирование**
- **Искусственный интеллект**
- **Информационные системы**
- **Вычислительная техника.**

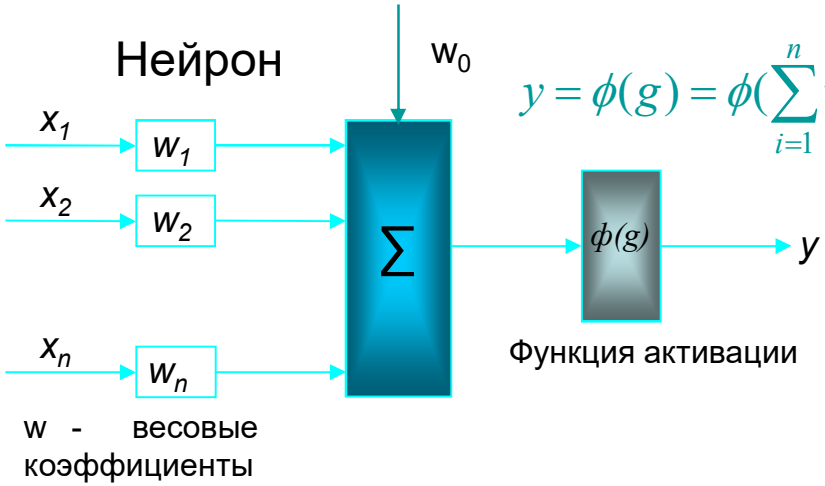
Кибернетика

Теория передачи сигналов
Теория управления
Теория автоматов
Теория принятия решений
Синергетика
Теория алгоритмов
Распознавание образов
Теория оптимального управления
Теория обучающихся систем

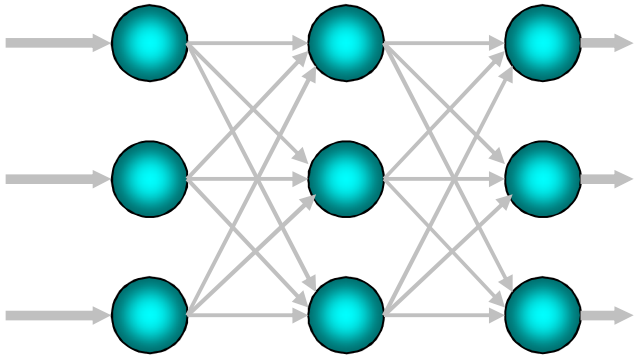
Искусственный интеллект
Кибернетика второго порядка
Метакибернетика
Компьютерное зрение
Системы управления
Эмерджентность
Обучающиеся организации
Новая кибернетика
Теория общения

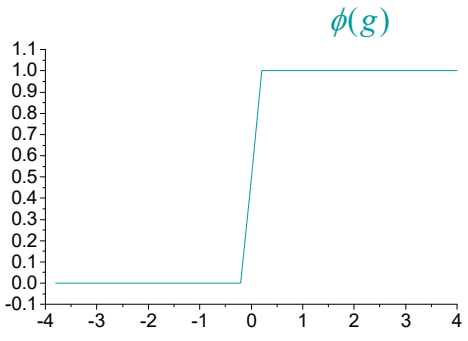
Биоинженерия
Биологическая кибернетика
Биоинформатика
Бионика
Медицинская кибернетика
Нейрокибернетика
Гомеостаз
Синтетическая биология
Системная биология

Нейронные сети

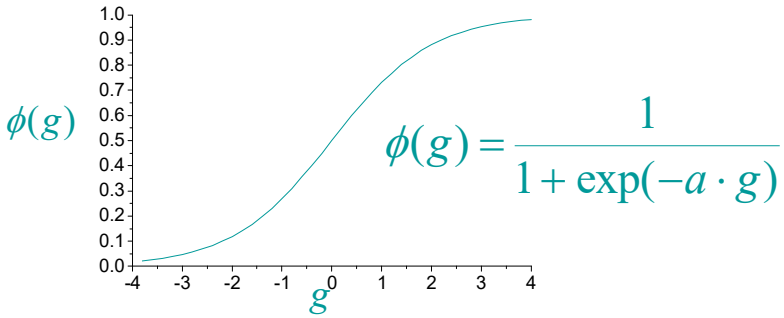


$$y = \phi(g) = \phi(\sum_{i=1}^{n} w_i x_i + w_0)$$



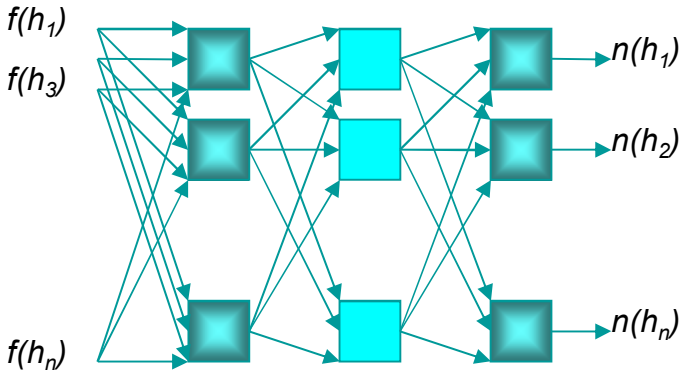


Пороговая функция активации

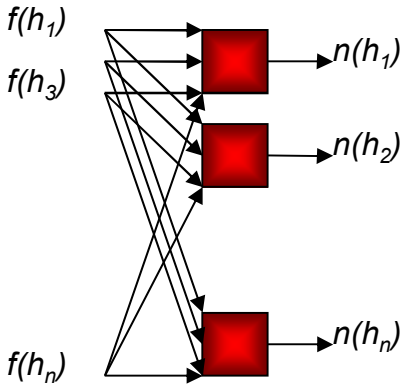


Сигмовидная функция активации

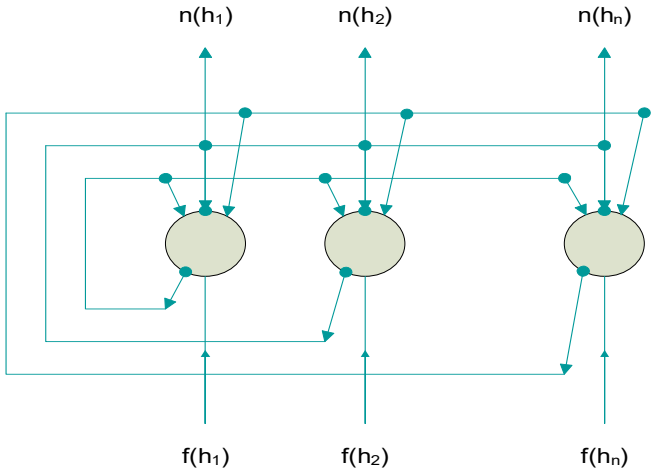
Виды нейронных сетей



Трехслойная нейронная сеть



Однослойный персептрон



Сеть Хопфилда

Число связей в многослойном персептроне

$$N = N_{in}N_1 + \ldots + N_iN_{i+1} + N_{n-1}N_{out}$$

Число связей в полно-связной нейронной сети для обработки вектора данных из 512 точек

$$N = 3 \cdot 512 \cdot 512 = 786432$$

Методы обучения нейронной сети

$$E = \frac{1}{2}\sum_{l=1}^{M}\sum_{i=1}^{N}(y_i(W) - d_{l,i})^2 \to \min$$

Метод обратного Распространения ошибки

$$w_{ij}(t+1) = w_{ij}(t) - h\frac{\partial E}{\partial w_{ij}}$$

$$\delta_j^{(n)} = \frac{\partial y_j^{(n)}}{\partial S_j^{(n)}}\sum_k \delta_j^{(n+1)} w_{jk}^{(n+1)}$$

$$\delta_j^{(n)} = (y_i^{(n)} - d_i)\frac{\partial y_j^{(n)}}{\partial S_j^{(n)}}$$

$$\Delta w_{ij} = -h\delta_j^{(n)} x_i^n$$

$$w_{ij}(t+1) = w_{ij}(t) + h\Delta w_{ij}$$

Стратегии улучшения работы алгоритма обучения

Устранение возможной блокировки сети

$$\Delta w_{ij}(t) = -h\delta_j^{(n)} x_i^n + \mu\Delta w_{ij}(t-1)$$

Устранение переобучения, путем кросс-проверки точности сети на другой выборке

Обхождение локальных минимумов с помощью увеличения нейронов скрытого слоя. Случайное изменения весовых коэффициентов

Технологии нейронных сетей

- Сети RNN
- Fast R-CNN, Faster R-CNN
- Сверточные нейронные сети
- Сети на основе слоев LSTM, GRU
- YOLO v3-5
- BERT, GPT-2,3

- Управляют беспилотником
- В 1996 году фирмой Accurate Automation Corp(**http://www.accurate-automation.com**), Chattanooga, TN по заказу NASA и Air Force был разработан экспериментальный автопилотируемый гиперзвуковой самолет-разведчик LoFLYTE (Low-Observable Flight Test Experiment — рис. 4). Самолет имел длину всего 2,5 м и вес 32 кг и был предназначен для исследования новых принципов пилотирования. LoFLYTE использовал нейронные сети, позволяющие автопилоту обучаться, копируя приемы пилотирования летчика. Поскольку самолет был предназначен для полетов со скоростью 4-5 махов, то быстрота реакции пилота-человека могла быть недостаточной для адекватного отклика на изменение режима полета. В этом случае на помощь приходили нейронные сети, которые перенимали опыт управления у летчика и за счет высокой скорости обработки информации позволяли быстро находить выход в аварийных и экстремальных ситуациях

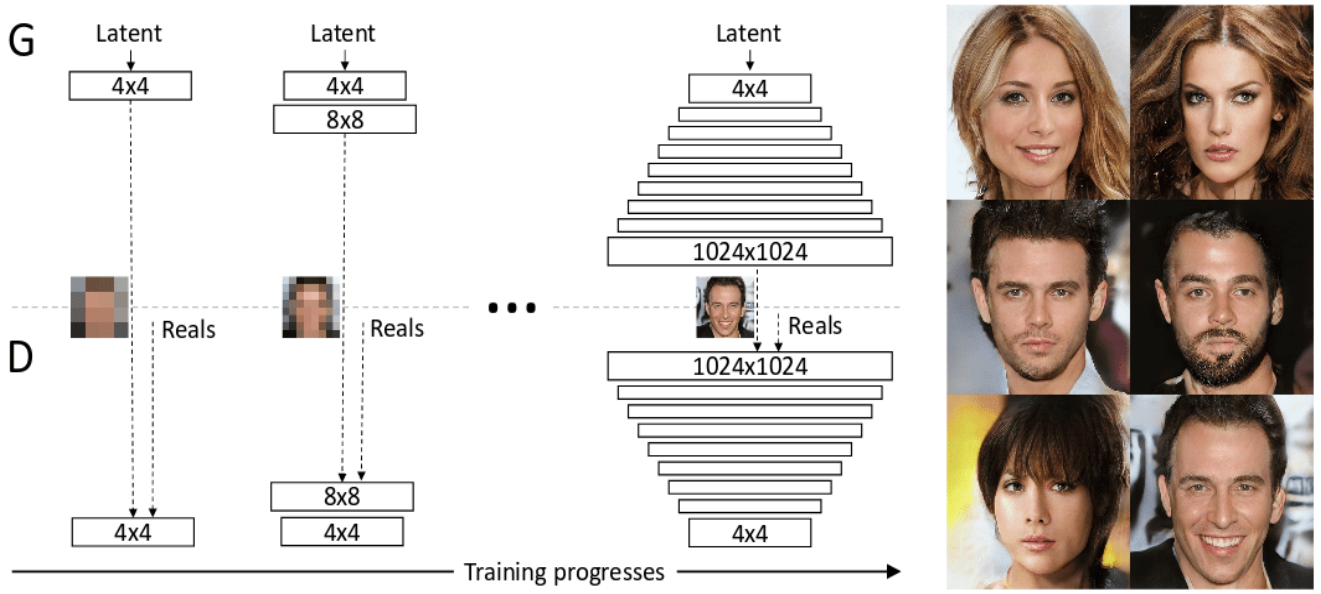
Библиотеки для создания и обучения нейронных сетей. Для построения графов вычислений для распараллеливания на процессорах и GPU.

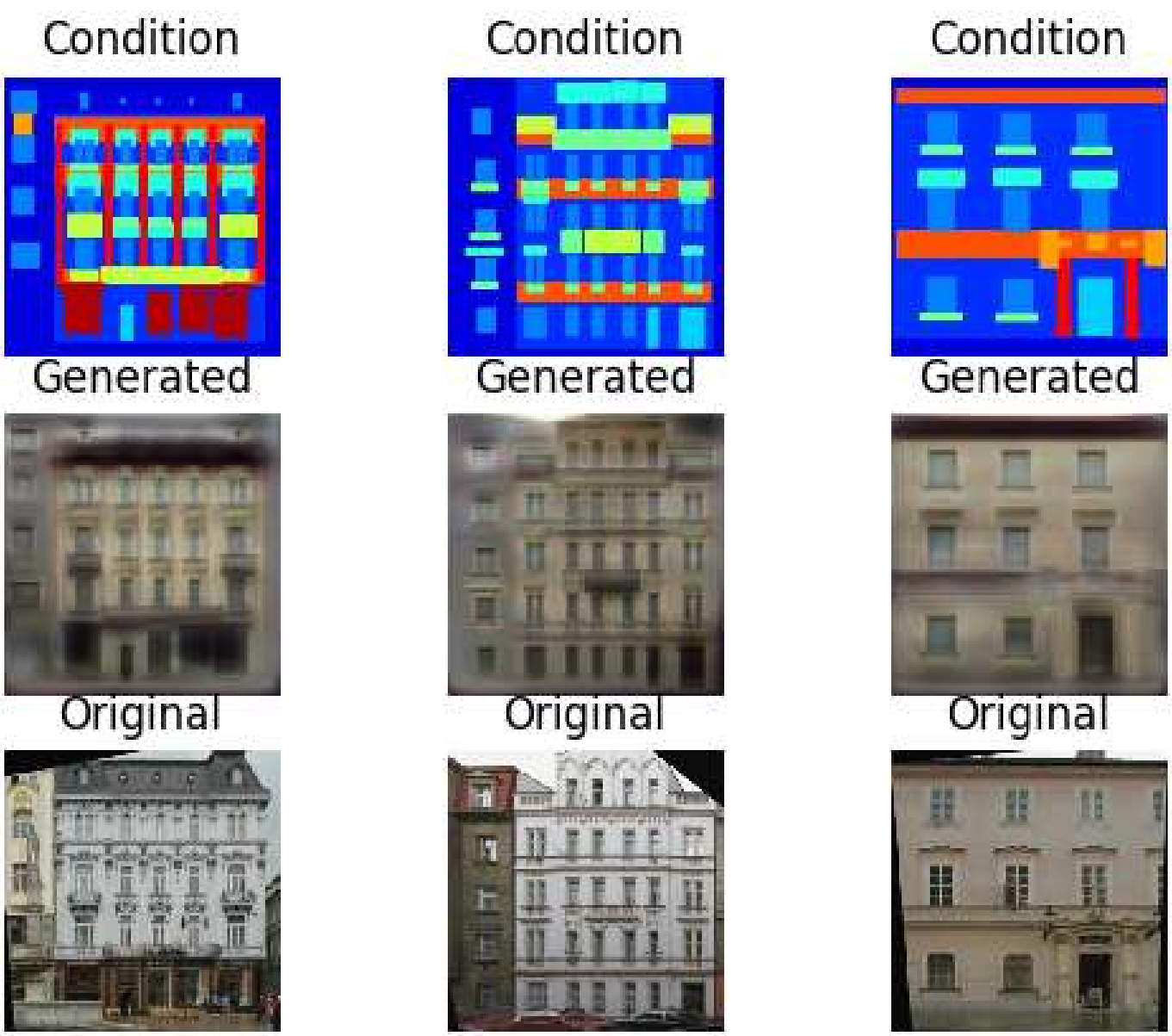
- Keras
- Caffe
- PyTorch
- Theano
- TensorFlow

Основные направления

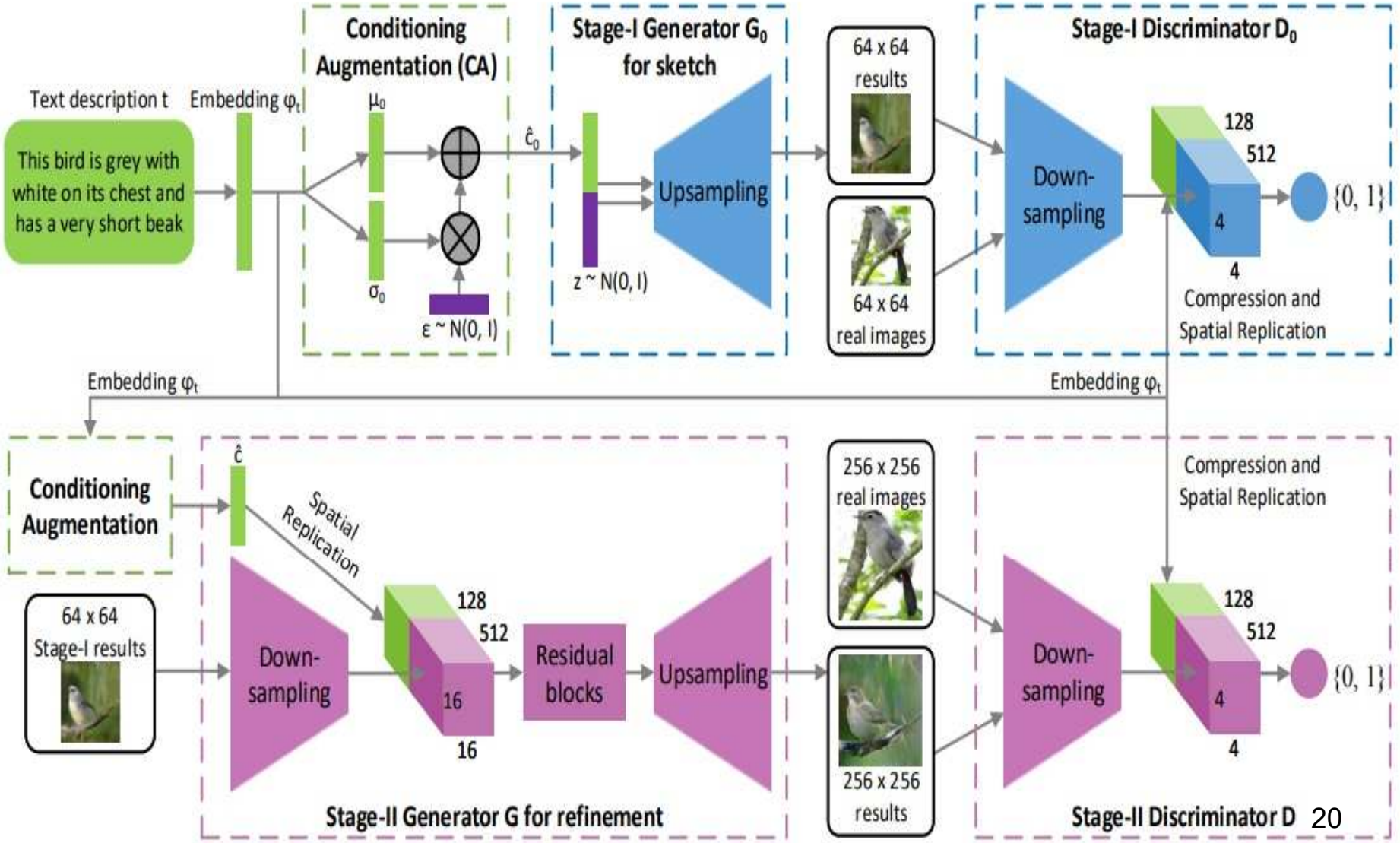
- Генерация и повышение разрешения изображений (вариационные аутоэнкодеры, генеративные состязательные сети (GAN).
- Обучение с подкреплением (Reinforcement learning) DQN (deep q network, q - learning), A123C (advantage asynchronous actor critic), PPO (proximal policy optimization)
- Классификация изображений.
- Сегментация и детектирование изображений. (R-CNN, Fast R CNN, Faster R CNN, Yolo v1-5)
- NLP (Nature language processing)

Mao et al. (2016b) (128 × 128) Gulrajani et al. (2017) (128 × 128) Our (256 × 256)




Condition
Condition
Condition
Generated
Generated
Generated
Original
Original
Original

StackGAN – генерация изображений птиц по текстовому описанию



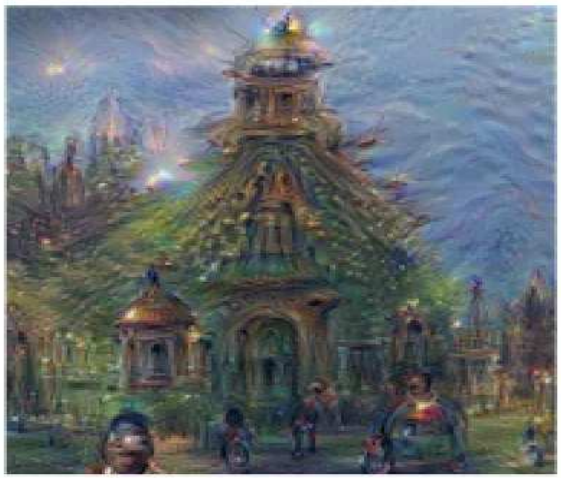
Нейронные сети рисуют картины с помощью смешения стилей

Horizon

Trees

Leaves

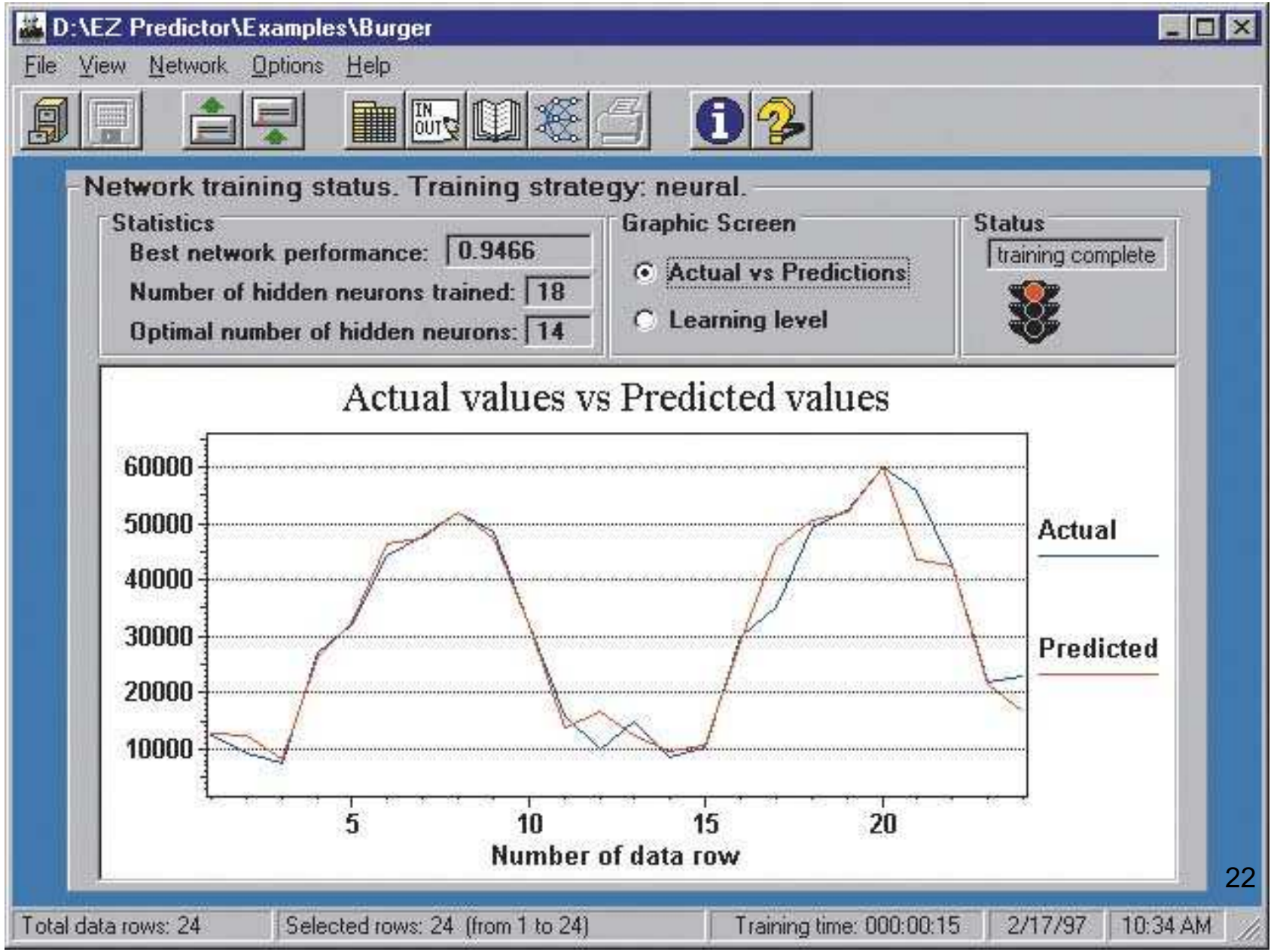
Towers & Pagodas



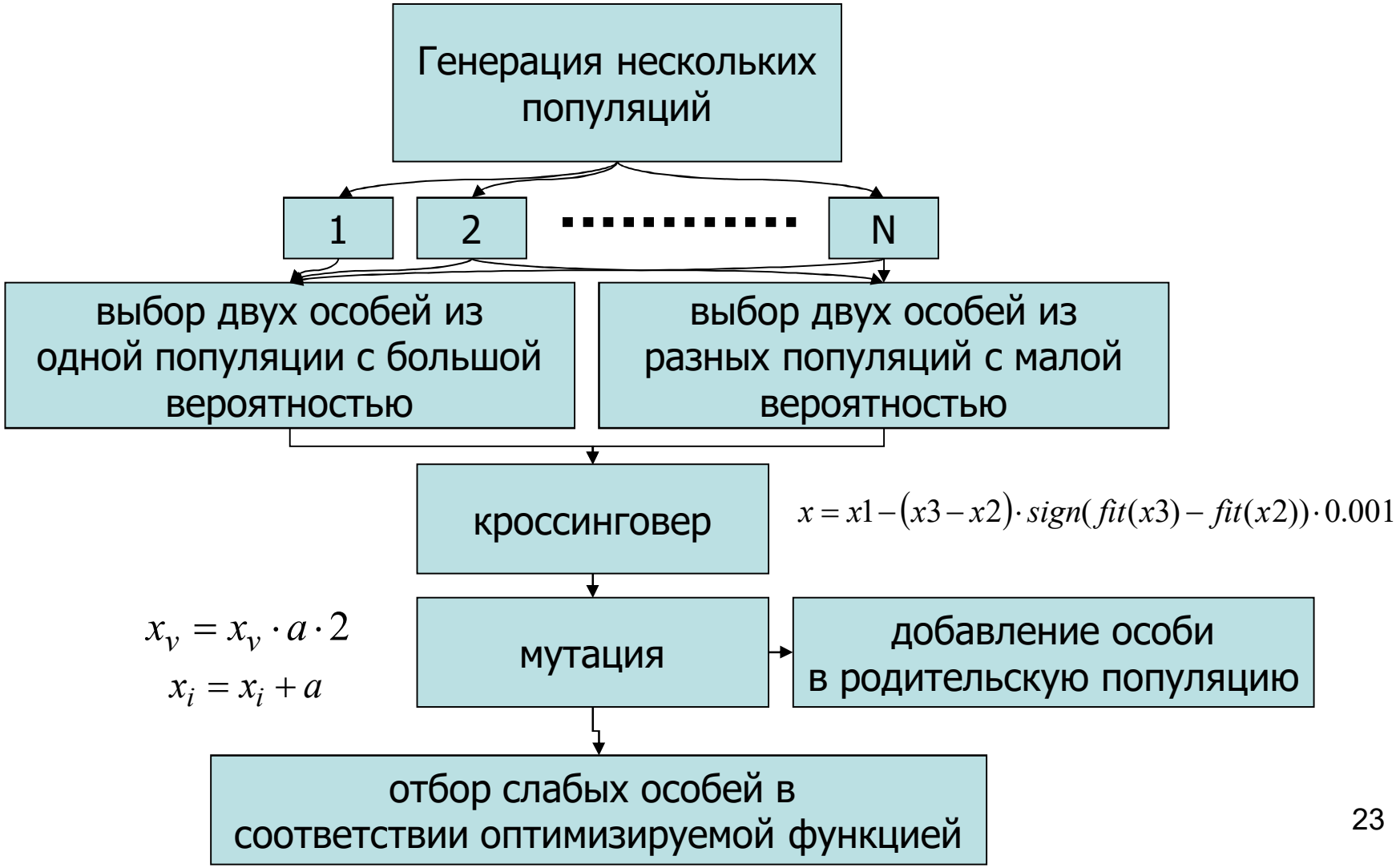
Buildings

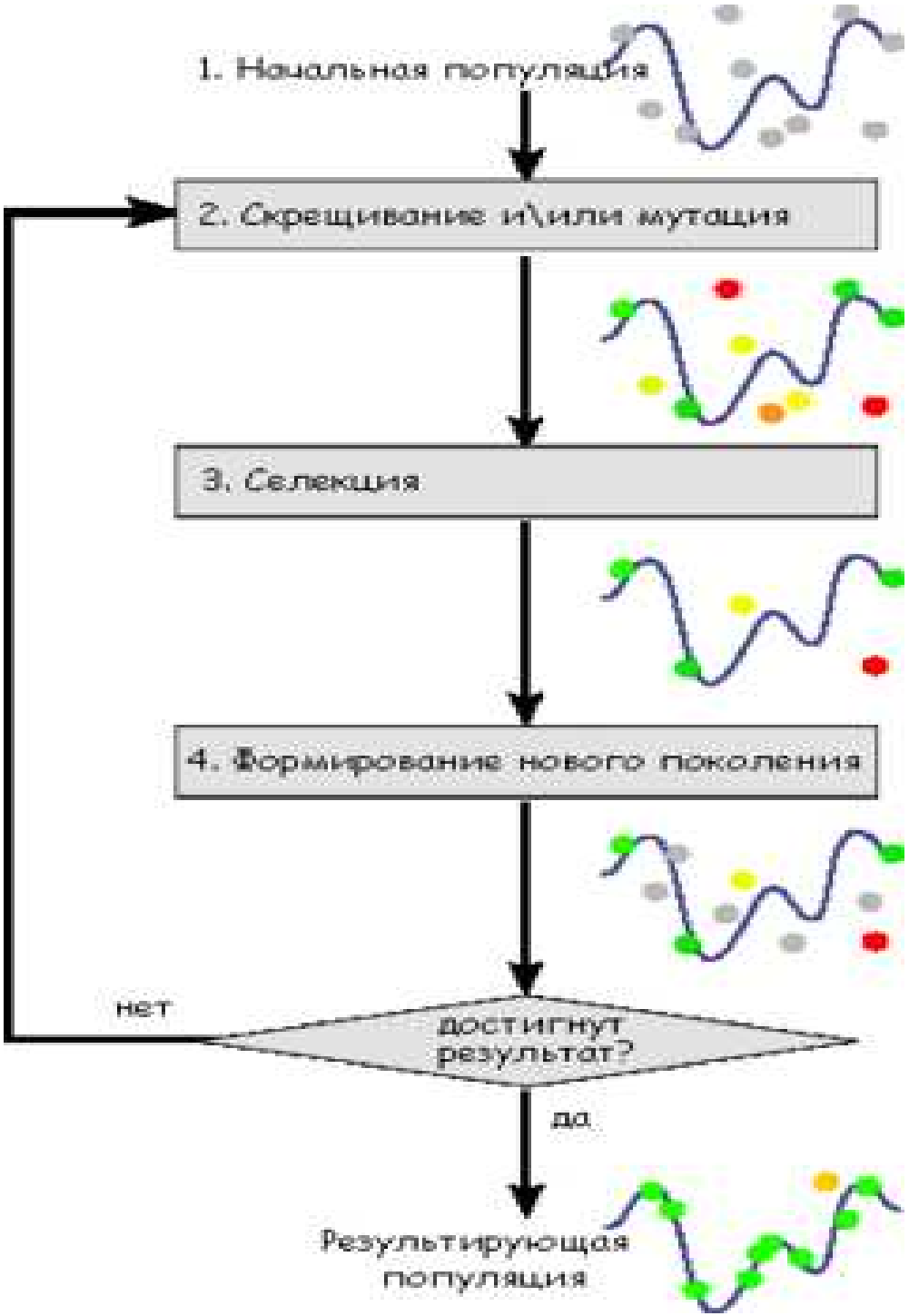
Birds & Insects

Предсказание, прогнозирование

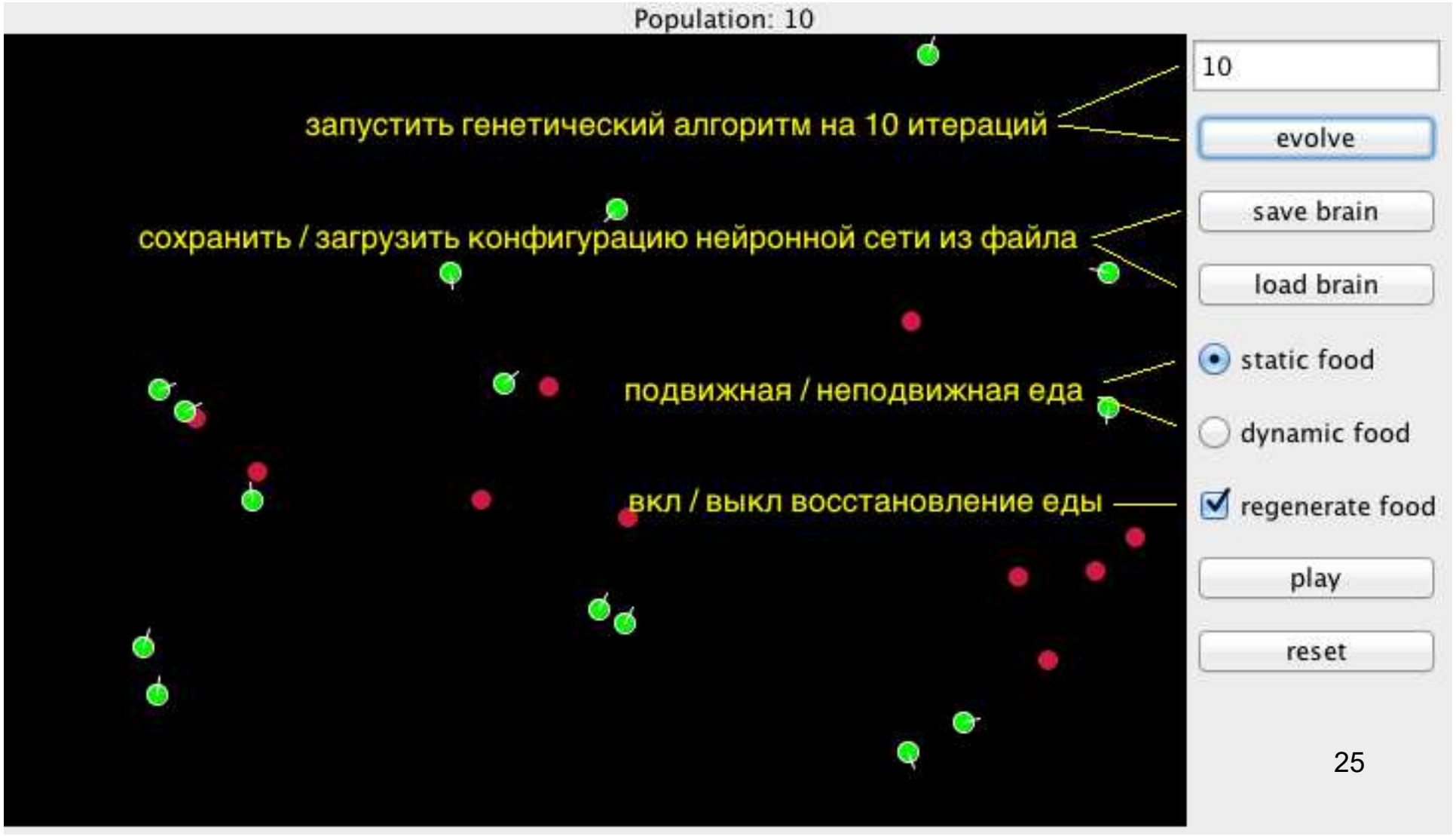


Оптимизация с использованием генетического алгоритма

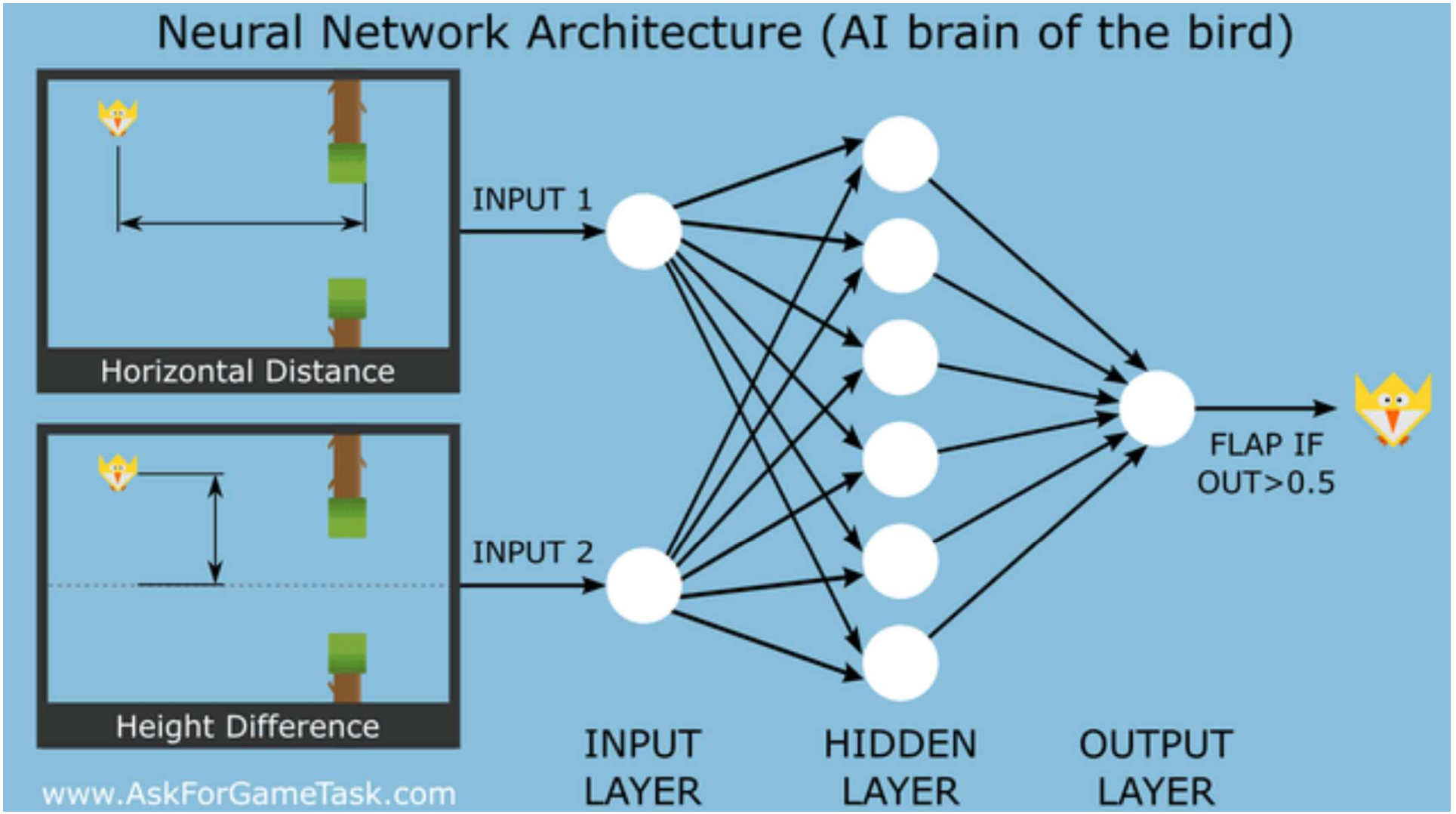



1. Начальная популяция
2. Скрещивание и\или мутация
3. Селекция
4. Формирование нового поколения
нет
достигнут результат?
да
Результирующая популяция

Пример обучения нейронной сети агента в искусственной виртуальной среде.
Используется генетический алгоритм отбора для настройки параметров агента.




In1 In2 Out
GEN 11
GEN 12
Fitness: -518.33 -82.33
Score: 0 0
Fitness: 365.89 -79.93
Score: 0 0
Fitness: 358.71 -76.67
Score: 0 0
Fitness: 420.94 -88.66
Score: 0 0
Fitness: 540.91 -98.37
Score: 0 0
Fitness: 452.45 -81.88
Score: 0 0
Fitness: 402.81 -110.42
Score: 0 0
Fitness: 406.78 -243.86
Score: 0 0
Fitness: 477.84 -76.94
Score: 0 0
Fitness: 88.54 -79.24
Score: 0 0
The best unit was born in generation 9:
Fitness = 572.87 / Score = 0
RESTART
PLAY MORE GAMES
PAUSE
www.askforgametask.com
ASK FOR GAME TASK © 2017


Neural Network Architecture (AI brain of the bird)
Horizontal Distance
Height Difference
INPUT 1
INPUT 2
FLAP IF
OUT>0.5
INPUT
LAYER
HIDDEN
LAYER
OUTPUT
LAYER
www.AskForGameTask.com

Эволюционно настраивается животное, которое учится двигаться.
Модель движения задается полиномиальными уравнениями.
С помощью эволюционного алгоритма ищутся параметры агента.

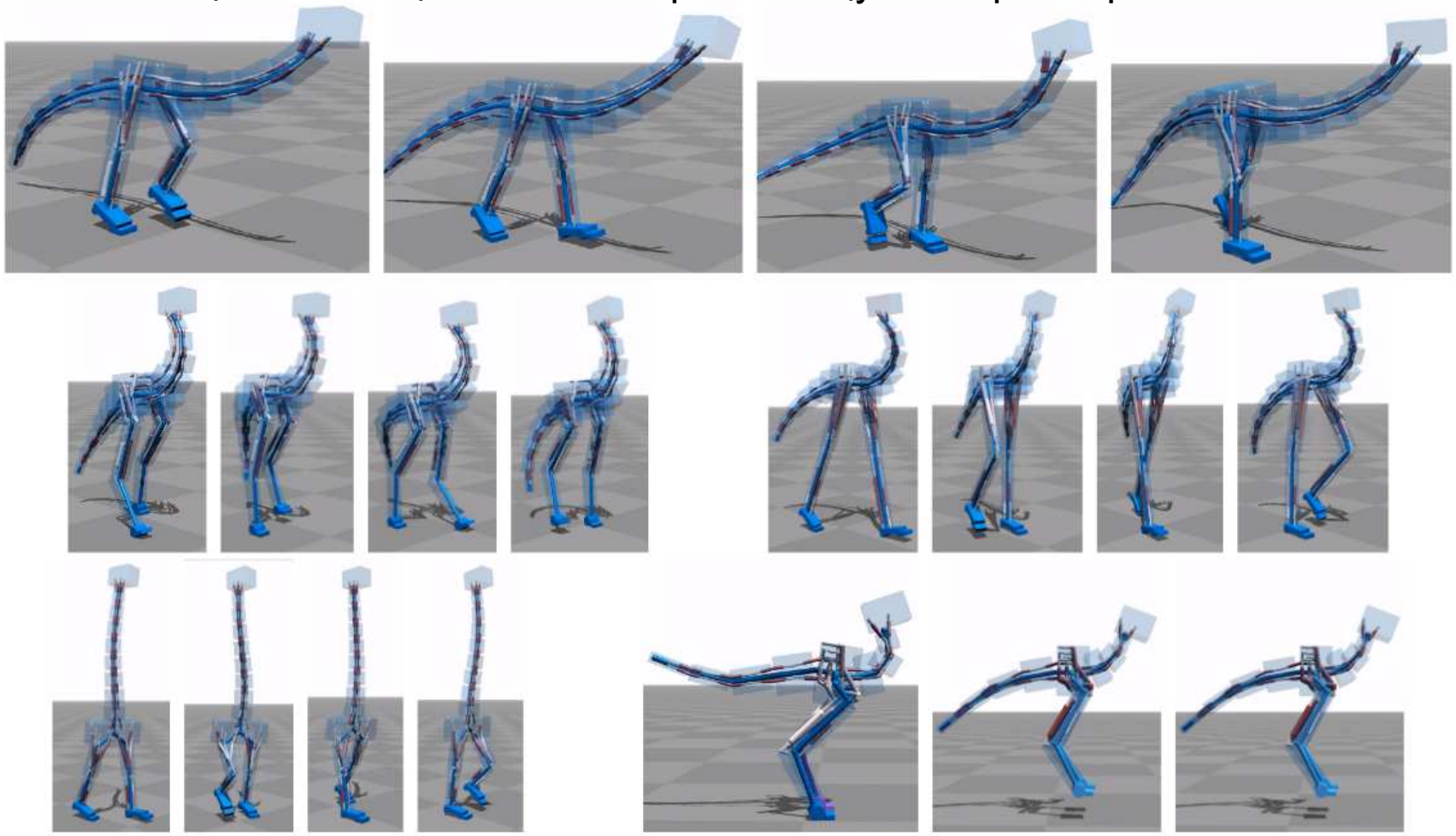
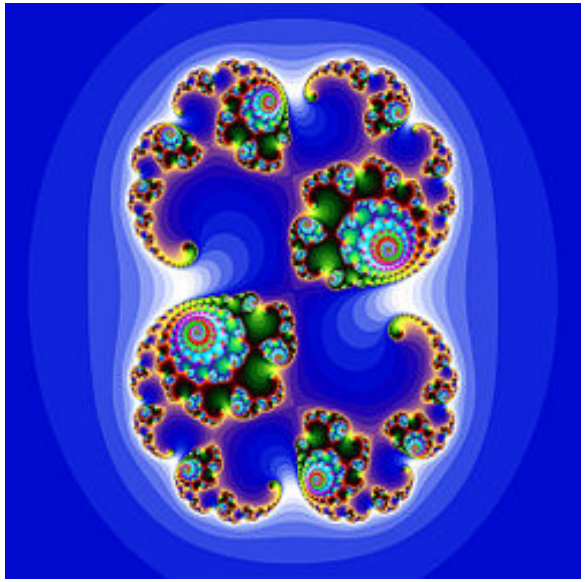
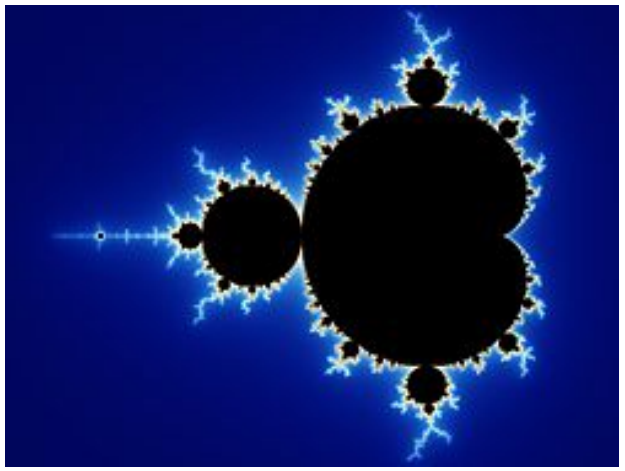


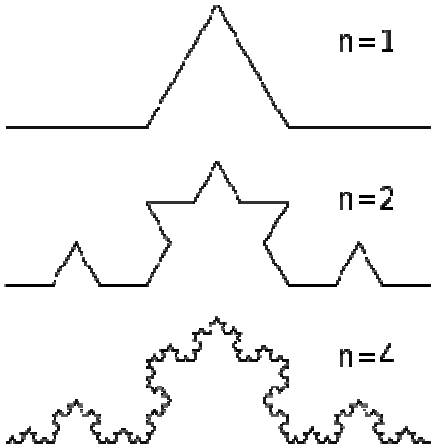
Figure 9: *Example synthesized walking and hopping gaits*

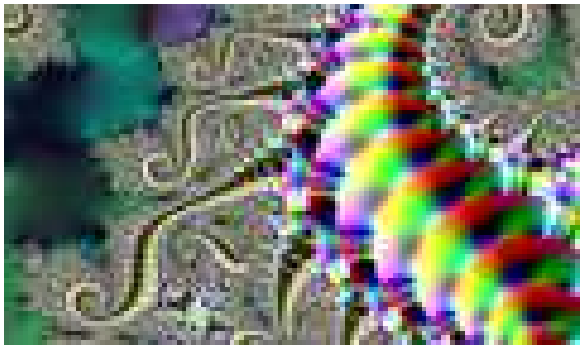
- Аппроксимация изображения полигонами

- Фракталы









Искусственный интеллект

Экспертные системы

Заменяют человека эксперта в некоторой предметной области, состоят из машины вывода и базы фактов, а также системы объяснений выводов

Базы знаний

Модели представления знаний

Семантические сети, Фреймы, Продукционные модели

Биологический интеллект (ГА, НС)

Методы познания

- Дедукция

Все люди смертны, Сократ человек, Сократ смертен

- Индукция

Очередная машина желтого цвета оказалась такси, значит такси в городе желтого цвета

- Абдукция

Все люди смертны, Сократ смертен, значит Сократ человек

- Анализ

Разложение на детали

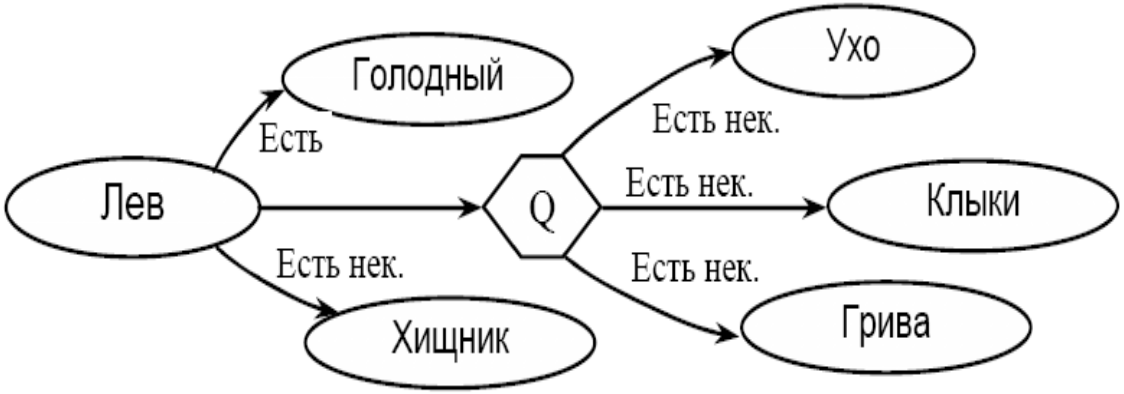
- Синтез

Объединение в общее

- Перебор вариантов
- Эвристика

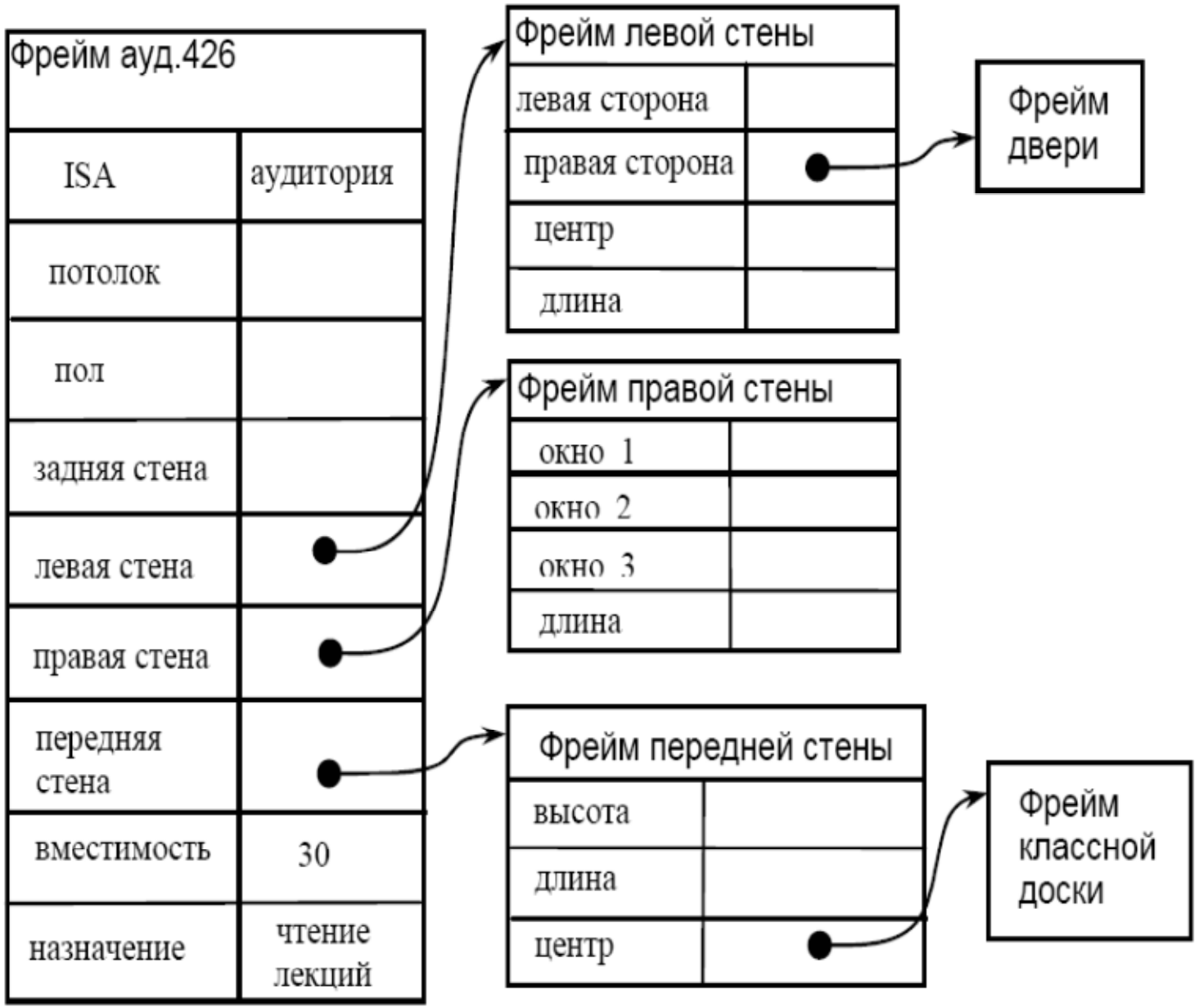
Способ или алгоритм который с высокой вероятностью приведет к результату близкому к требуемому

Семантическая сеть



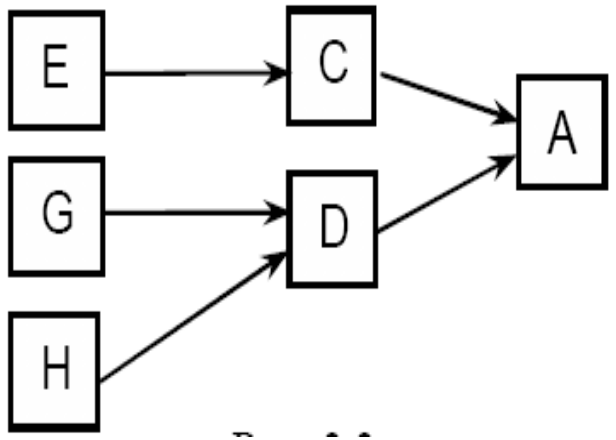
Фреймы


Конференция
Дата
Место проведения
Тема
Выступающие
IS-A
IS-A
Конференция по коммерческим вопросам
Конференция по развитию
Дата
Место проведения
Тема Торговля
Выступающие
Цель
Дата
Место проведения
Тема Развитие
Выступающие
Бюджет
IS-A
1 класс (шаблон)
Четвертая конференция по коммерческим вопросам
Дата 10.03.02 10:30
Место проведения Конференцзал
Тема
Выступающие Иванов
Цель 25 т. долл.
Экземпляр


Фрейм ауд.426
ISA
аудитория
потолок
пол
задняя стена
левая стена
правая стена
передняя стена
вместимость
30
назначение
чтение лекций
Фрейм левой стены
левая сторона
правая сторона
центр
длина
Фрейм двери
Фрейм правой стены
окно 1
окно 2
окно 3
длина
Фрейм передней стены
высота
длина
центр
Фрейм классной доски

Продукционная модель

посылка p,
правило $p \rightarrow q$,
заключение q;



$C \wedge D \rightarrow A$, $B \rightarrow A$, $E \rightarrow C$, $F \rightarrow C$, $G \wedge H \rightarrow D$, $J \wedge K \wedge L \rightarrow B$.

Программирование

- Логические языки (Prolog, Меркурий)
- Функциональные языки (Lisp, Haskell)
- Процедурные (Си, Паскаль и т.д.)
- Объектно-ориентированные (Java, C#, Ruby)
- Машинные

Пример на языке пролог

- предок(X, Z) :- родитель(X, Z).
- предок(X, Z) :- родитель(X, Y), предок(Y, Z).
- родитель(пам, боб). % Пам - родитель Боба
- родитель(том, боб).
- родитель(том, лиз).
- родитель(боб, энн).
- родитель(боб, пат).
- родитель(пат, джим).
- ?- предок(пам, X).
- X = боб;
- X = энн;
- X = пат;
- X = джим

Пример расчета факториала на haskell и Си и Паскале (рекурсия)

```
program factorial;
function fact(n: integer): longint;
begin
    if (n = 0) then
        fact := 1
    else
        fact := n * fact(n - 1);
end;

var
    n: integer;

begin
    for n := 0 to 16 do
        writeln(n, '! = ', fact(n));
end.
```

```
factorial :: Integer -> Integer
factorial 0 = 1
factorial n = n * factorial (n - 1)
```

```
#include <stdio.h>
unsigned long long factorial(unsigned long long n)
{
    if (n == 0) {
        return 1;
    } else {
        return n * factorial(n - 1);
    }
}
int main(void)
{
    int n;
    for (n = 0; n <= 16; n++) {
        printf("%i! = %lld\n", n, factorial(n));
    }
    return 0;
}
```

Пример на Лиспе

```
(defun factorial (n)
    (if (= n 0)
        1
        (* n (factorial (- n 1))) ) )

(loop for i from 0 to 16
    do (format t "~D! = ~D~%" i (factorial i)))
)
```

ООП

- Инкапсуляция
- Наследование
- Полиморфизм

Информационные системы

- Базы данных
- ГИС
- Поисковые системы

Вычислительные системы

- Компьютеры на базе транзисторных элементов
- Компьютеры на базе оптических элементов
- Биокомпьютеры
- Квантовые компьютеры

- Биокомпьютер — компьютер, который функционирует как живой организм или содержит биологические компоненты. Создание биокомпьютеров основывается на направлении молекулярных вычислений. В качестве вычислительных элементов используются белки и нуклеиновые кислоты, реагирующие друг с другом.

- Молекулярные компьютеры — вычислительные системы, использующие вычислительные возможности молекул (преимущественно, органических). Можно сказать, что молекулярные компьютеры — это молекулы, запрограммированные на нужные свойства и поведение. Молекулярные компьютеры состоят из сетевых нано-компьютеров. В работе обычной микросхемы используют отдельные молекулы в качестве элементов вычислительного тракта.

- В частности, молекулярный компьютер может представлять логические электрические цепи, составленные из отдельных молекул; транзисторы, управляемые одной молекулой, и т. п. В микросхеме памяти информация записывается с помощью положения молекул и атомов в пространстве.

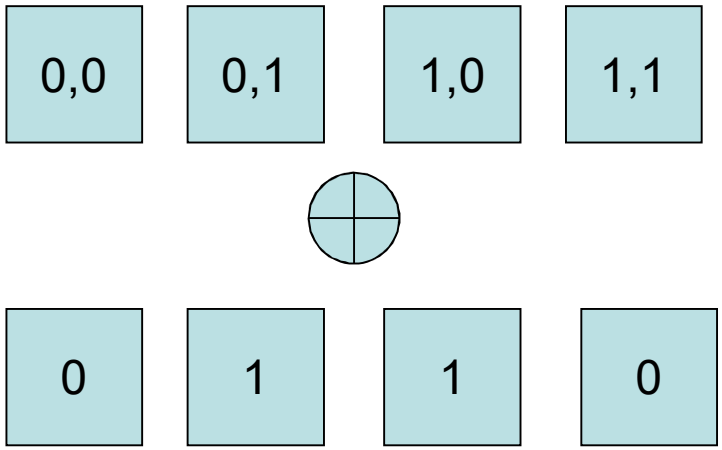
- Одним из видов молекулярных компьютеров можно назвать ДНК-компьютер, вычисления в котором соответствуют различным реакциям между фрагментами ДНК. От классических компьютеров ДНК-компьютеры отличаются тем, что химические реакции происходят сразу между множеством молекул независимо друг от друга.

Квантовый компьютер

- Кубиты, кубайты
- Вычисления проводятся сразу на всеми состояниями
- Квантовые криптографические алгоритмы

$$a_1\langle 0,0| + a_2\langle 0,1| + a_3\langle 1,0| + a_4\langle 1,1|$$

Описание квантовой системы



n- бит, 2^n – вычислений над состояниями

Оптические или **фотонные вычисления** — вычисления, которые производятся с помощью фотонов, сгенерированных лазерами или диодами. Используя фотоны, возможно достигнуть более высокой скорости передачи сигнала, чем у электронов, которые используются в современных нам компьютерах.

Большинство исследований фокусируется на замене обычных (электронных) компонентов компьютера на их оптические эквиваленты. Результатом станет новая цифровая компьютерная система для обработки двоичных данных. Такой подход дает возможность в краткосрочной перспективе разработать технологии для коммерческого применения, поскольку оптические компоненты могут быть внедрены в стандартные компьютеры, сначала создавая гибридные системы, а впоследствии и полностью фотонные. Однако опто-электронные приборы теряют 30% энергии на конвертацию электронов в фотоны и обратно. Это также замедляет передачу информации. В полностью оптическом компьютере надобность преобразования сигнала из оптического в электронный и обратно в оптический полностью исчезает