

**Deep machine learning  
and pattern/face recognition  
based on quantum neural networks  
and quantum genetic algorithm**

**S.V. Ulyanov**

**A.G. Reshetnikov**

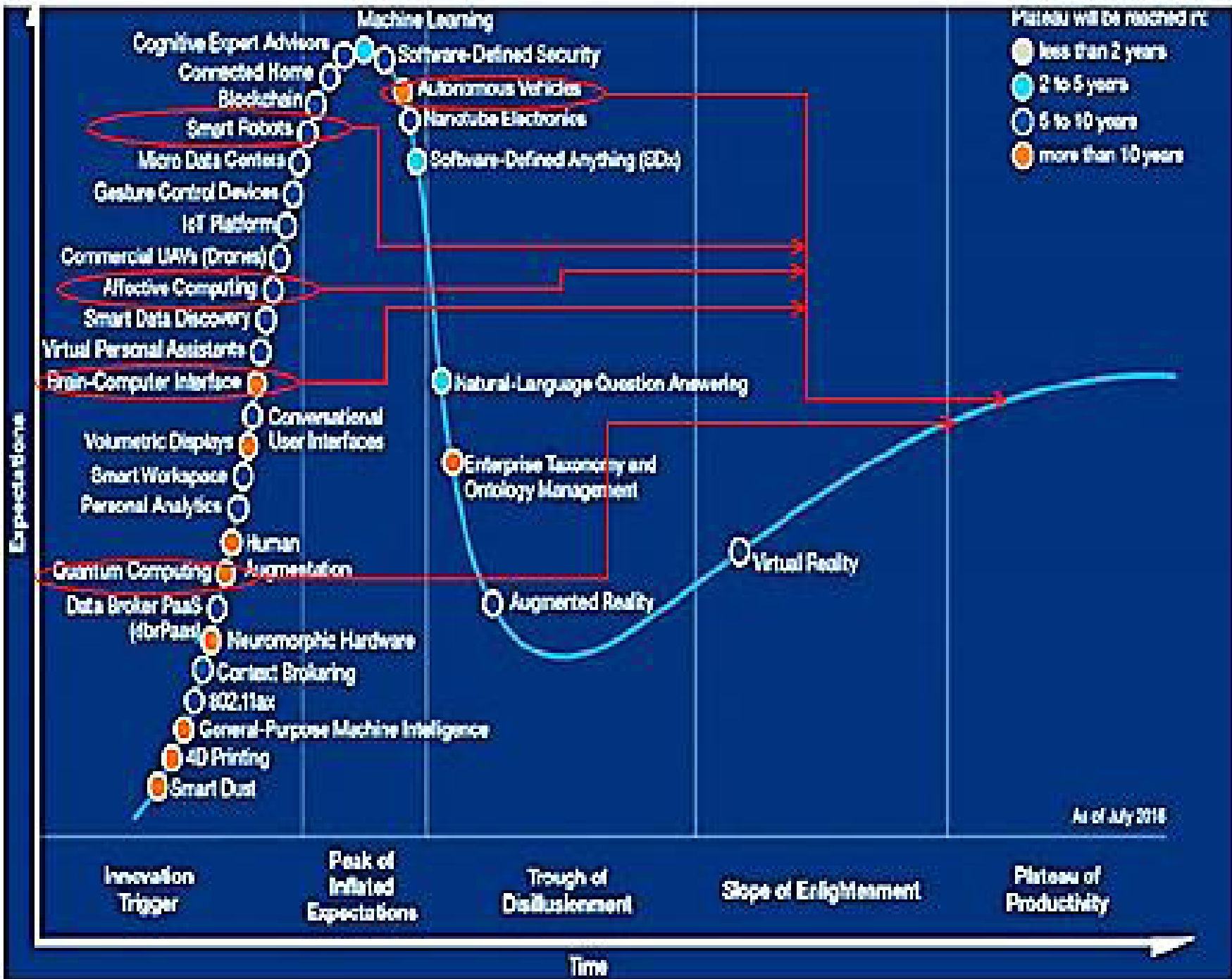
**N.V. Ryabov**

**ulyanovsv@mail.ru**

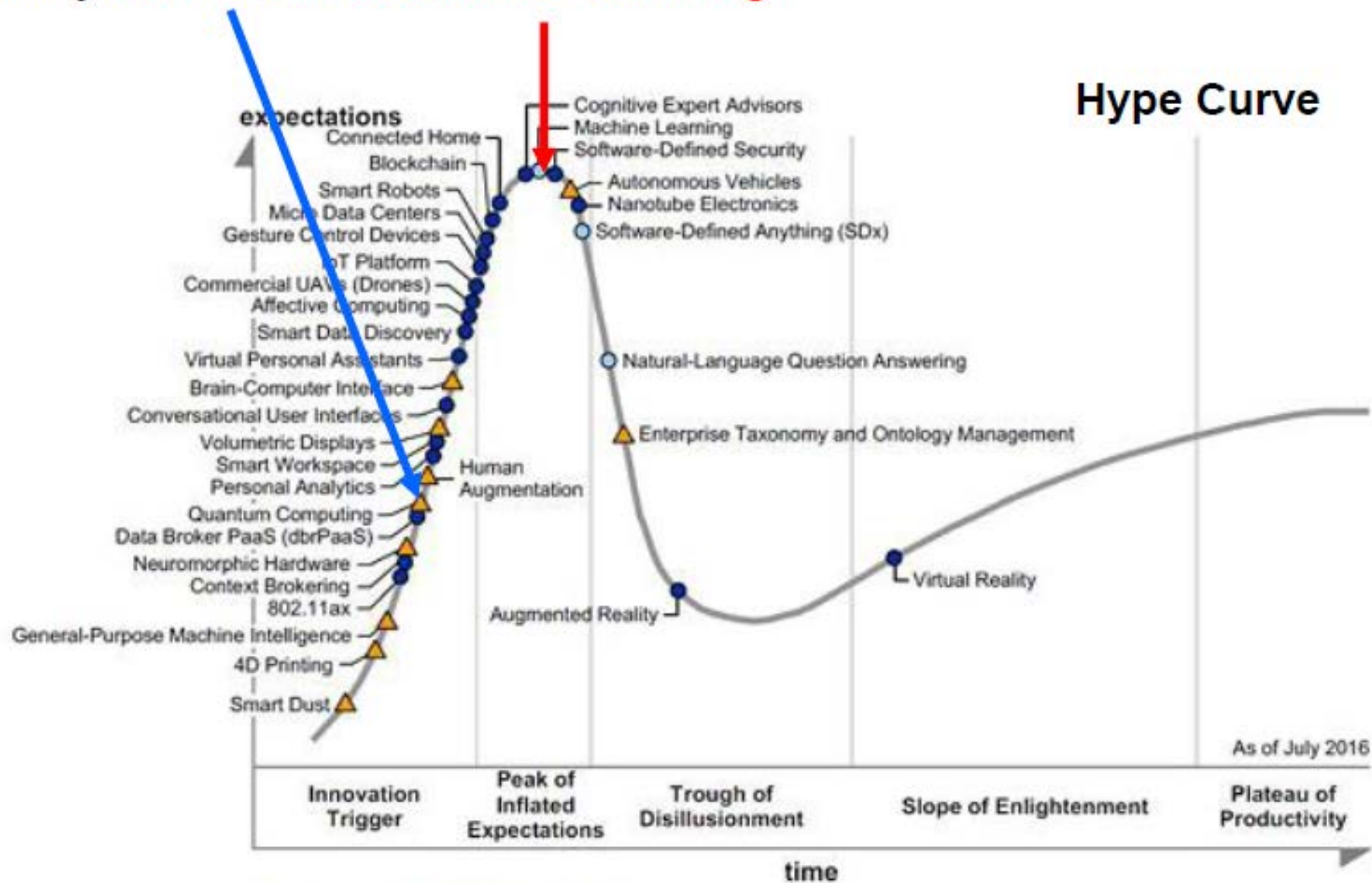
**The 8th International Conference  
"Distributed Computing and Grid-technologies  
in Science and Education"  
JINR, Dubna, September 13, 2018**

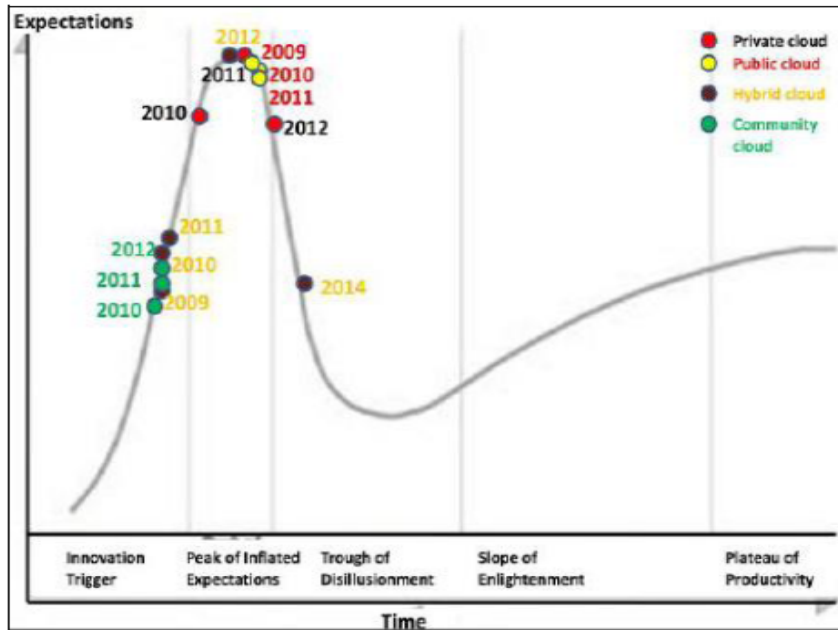
# Quantum Deep Learning

**R&D Development:  
Application Activities**

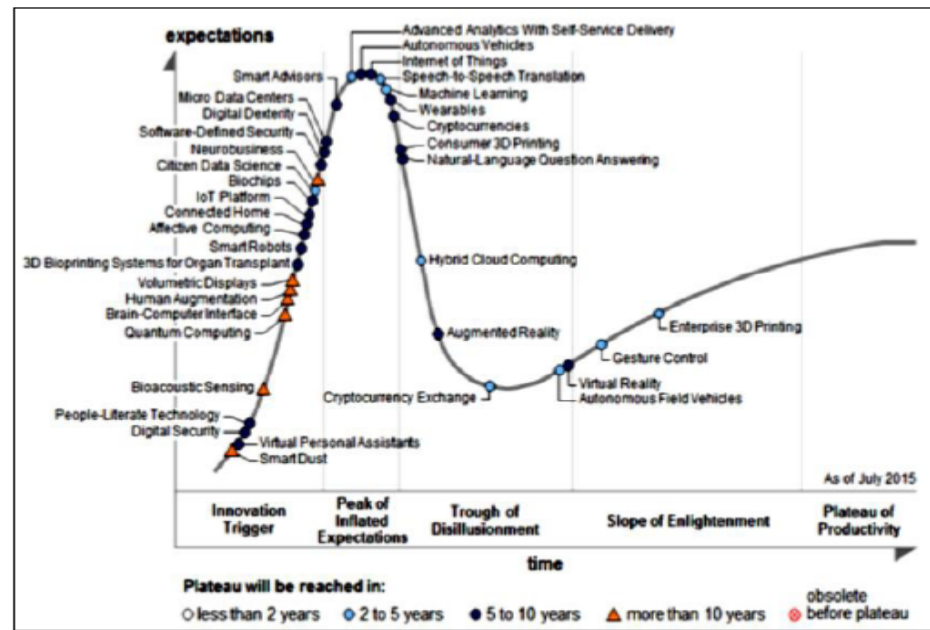


# Why Quantum Machine Learning?

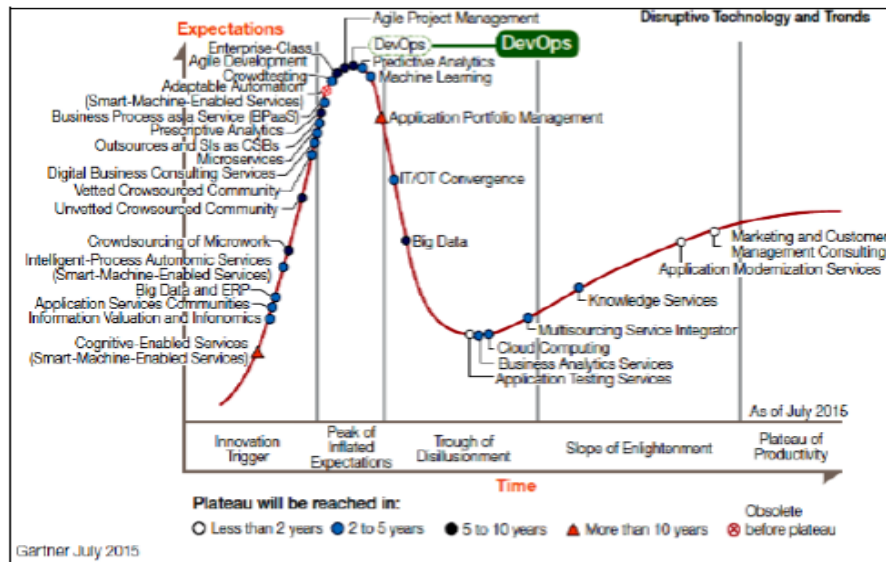




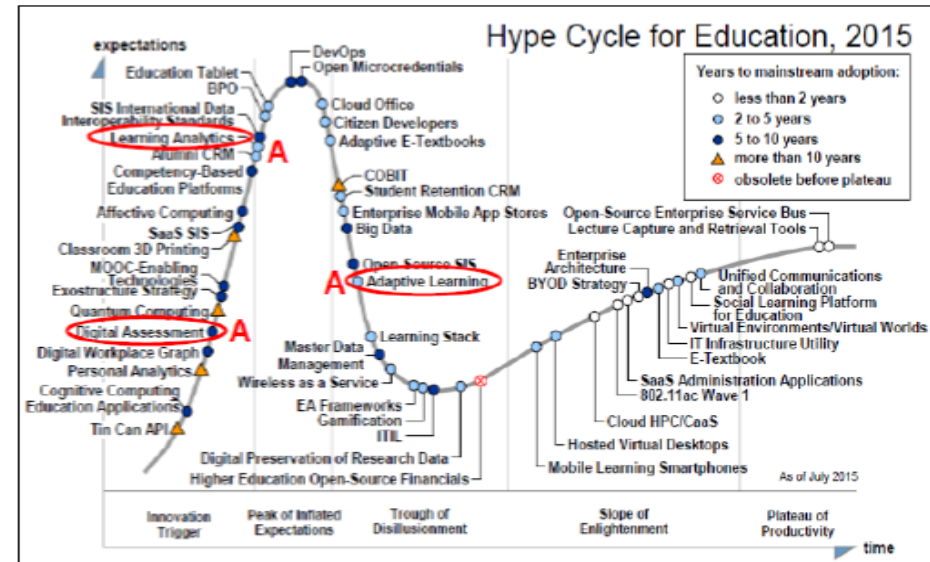
(a)



(b)



(c)



(d)

Рис. 3. Развитие ИТ в области облачных вычислений (а), робототехники (б), автоматизации (в) и образовательных процессов (г)

# Background of Quantum Computational Intelligence

## Quantum Soft Computing

Quantum Fuzzy System

Quantum Neural Network

Quantum Genetic Algorithm

Applied Intelligent Toolkit

## Quantum Deep Learning Models with Applications

Big & Mining  
Experimental Data

Physical Models Design  
from Experimental Data

Quantum Intelligent Control

Quantum Search

Experiment Design

NICA megaproject

Applications

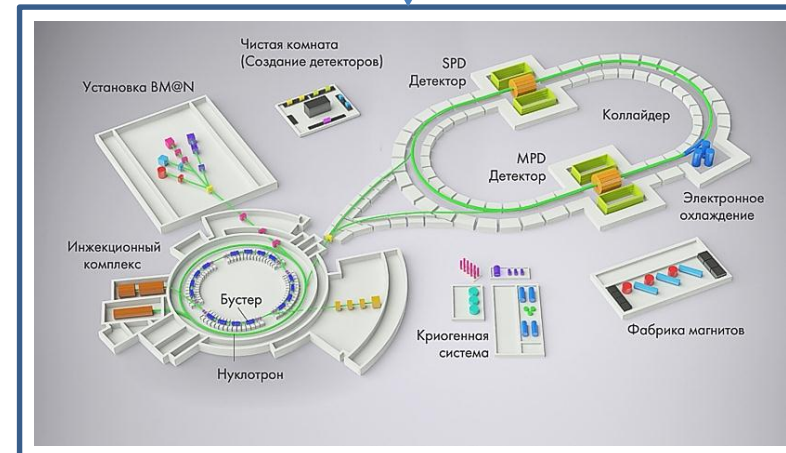
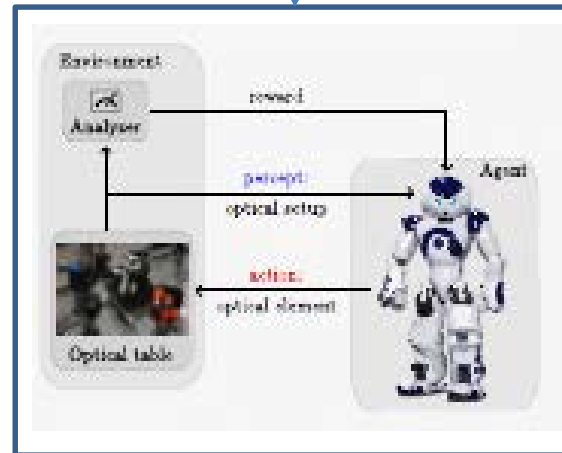
Grover Search Algorithm Simulator v0.3  
Type "Grover.exe h" for more help

Order? 1000  
K? 1000000  
Number of iterations? 100000000  
Superposition: 0.000 sec  
Entanglement: 14.652 sec  
Interference: 81.478 sec  
Total time: 96.130 sec  
Answer: YES

1000 qubit Grover's algorithm simulation (2<sup>1000</sup> elements in DB)

100 000 000 Iterations

In less than 2 minutes



# Artificial Neural Network

N	Several important ideas in the field of ANN:
1	The concept of a processing element (neuron);
2	The transformation performed by this element (in general, input summation and
3	nonlinear mapping of the result into an output value);
4	The interconnection structure between neurons;
5	The network dynamics; The leaning rule which governs the modification of interconnection strengths

N	Shortcomings of ANN:
1	The absence of rules for determining optimal architectures;
2	Limited memory capacity;
3	Slower learning for huge data sets;
4	Catastrophic forgetting due to the pattern interference etc.

## Abilities:

- Approximation of functions with any accuracy;
- learning;
- adaptation;
- knowledgebase design

## Нечеткие системы как универсальные аппроксиматоры

Методология нечеткого моделирования основана на важнейших теоремах (необходимые и достаточные условия), согласно которым нечеткие системы обладают свойствами *универсальных аппроксиматоров* (*universal approximators*).

*Теорема о необходимых условиях:* Для любой действительной непрерывной функции  $f$  на компактном множестве  $U \subseteq R^n$  и произвольной  $\varepsilon$  существует нечеткая логическая система  $F$  (с нечеткой импликацией в виде нечеткой конъюнкции (умножения), с синглетон-фаззификатором, дефаззификатором «по центру тяжести» и Гауссовскими функциями принадлежности) такая, что

$$|F(x) - f(x)| < \varepsilon.$$

Эти теоремы были доказаны Ulyanov, Wang L.-X. и Kosko B.

*Теорема о достаточных условиях:* Нечеткая логическая система может аппроксимировать любую действительную непрерывную функцию.

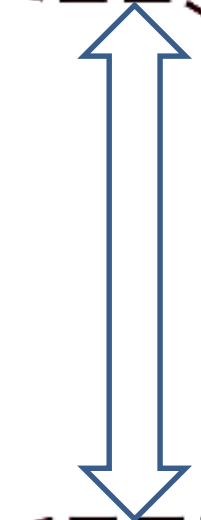
Эта теорема была доказана Buckley J.J.

Эти две теоремы объясняют, почему нечеткие системы так привлекательны в инженерных приложениях теории управления: нечеткие контроллеры могут рассматриваться как универсальные аппроксиматоры систем с неизвестной динамикой и структурой.



Универсальный аппроксиматор

Нечеткий регулятор



Нечеткая Нейронная сеть

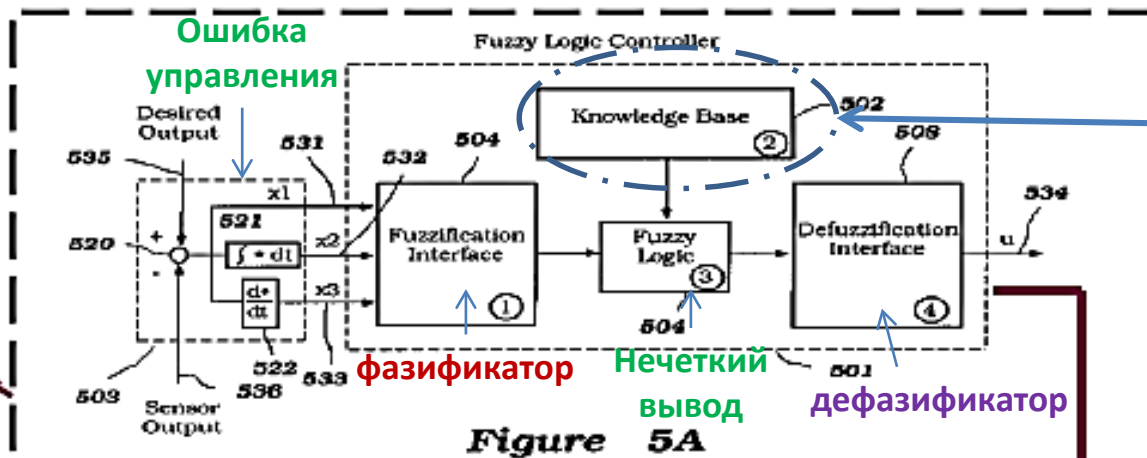


Figure 5A

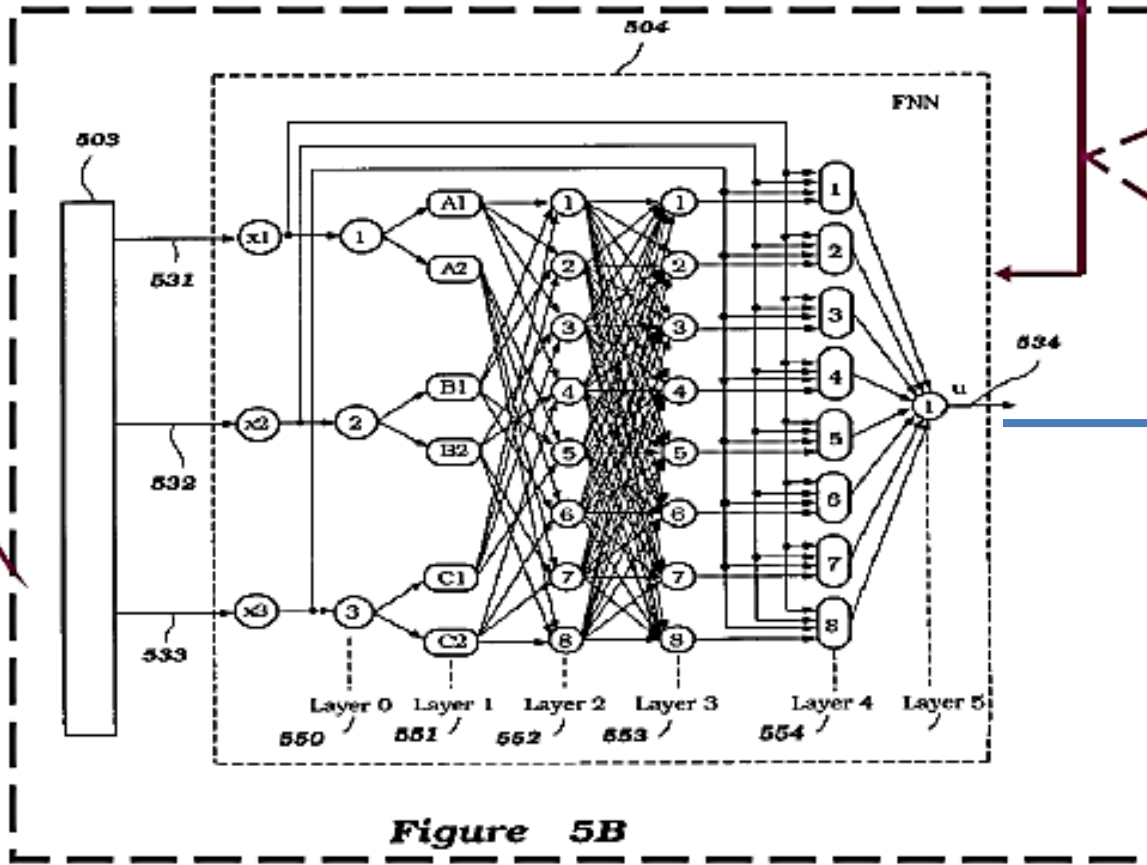


Figure 5B

Эквивалентные структуры

База знаний

$\langle C, F, P^0, \mu, \Omega, \Gamma(p^\Gamma), \Delta(p^\Delta), \Psi \rangle$



Type I (Fig.14a: Consequence - Constant ):

$R^i$  : IF  $x_1$  is  $A_{i_1}$  and  $x_2$  is  $A_{i_2}$

THEN  $y = f_i$ ,  $i = 1, 2, \dots, n$ ;

$$y^* = \left[ \frac{\sum_{i=1}^n \mu_i f_i}{\sum_{i=1}^n \mu_i} \right] = \sum_{i=1}^n \hat{\mu}_i f_i$$

Type II (Fig.14b: Consequence - First-order linear equation ):

$R^i$  : IF  $x_1$  is  $A_{i_1}$  and  $x_2$  is  $A_{i_2}$

THEN  $y = f_i(x_1, x_2)$ ,  $i = 1, 2, \dots, n$ ;

$$y^* = \left[ \frac{\sum_{i=1}^n \mu_i f_i(x_1, x_2)}{\sum_{i=1}^n \mu_i} \right] = \sum_{i=1}^n \hat{\mu}_i f_i(x_1, x_2),$$

$f_i(x_1, x_2) = a_{i0} + a_{i1}x_1 + a_{i2}x_2$ ;

$a_{ij}$  ( $j = 0, 1, 2$ ) - constant

Type III (Fig.14c: Consequence - Fuzzy variable ).

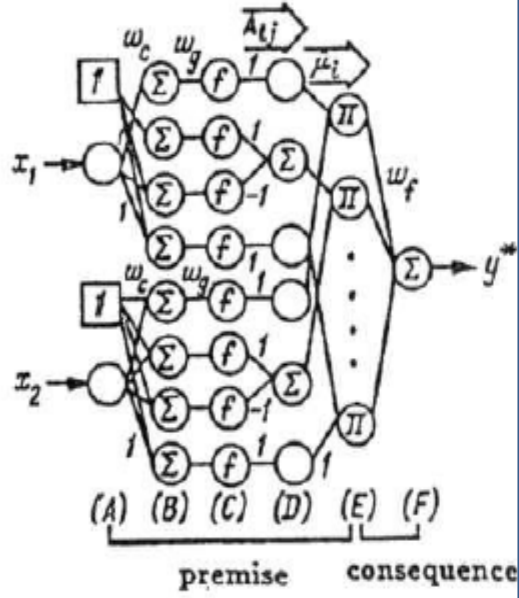
$R_k^i : (IF\ x_1\ is\ A_{i_1} \ and\ x_2\ is\ A_{i_2}\ THEN\ y\ is\ B_k) \ is\ \tau_{R_k^i},$

$i = 1, 2, \dots, n; k = 1, 2;$

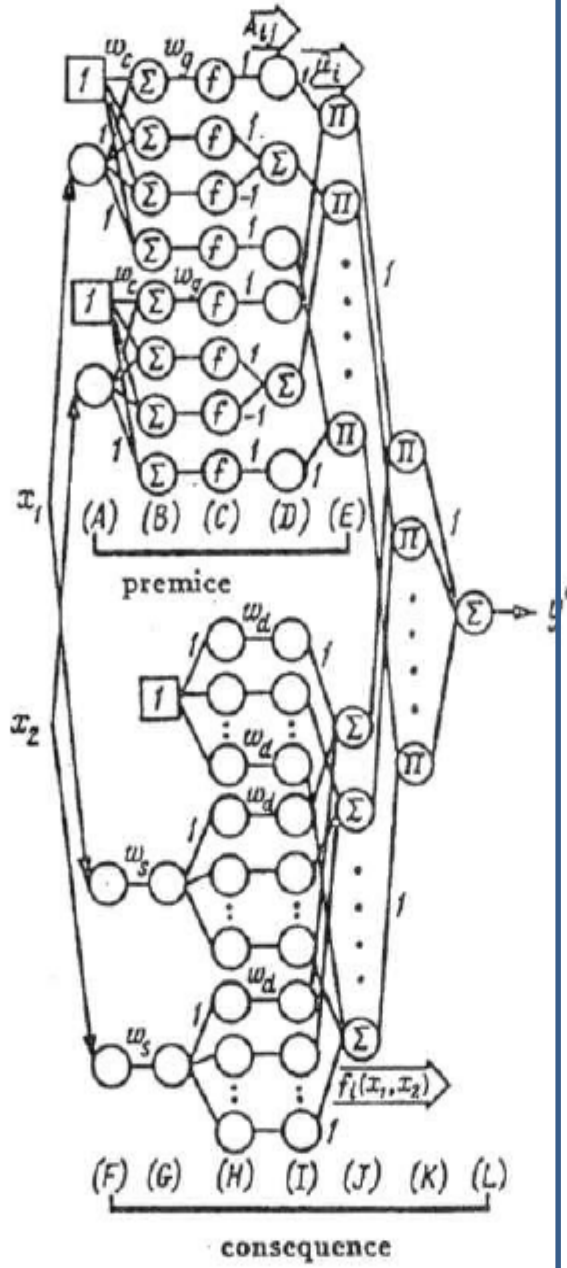
$$\mu_k^* = \sum_{i=1}^n \hat{\mu}_i \tau_{R_k^i}; \quad y^* = \left[ \frac{\sum_{k=1}^2 \mu_k^* B_k^{-1}(\mu_k^*)}{\sum_{k=1}^2 \mu_k^*} \right]$$

$$= \sum_{k=1}^2 \tilde{\mu}_k^* B_k^{-1}(\mu_k^*)$$

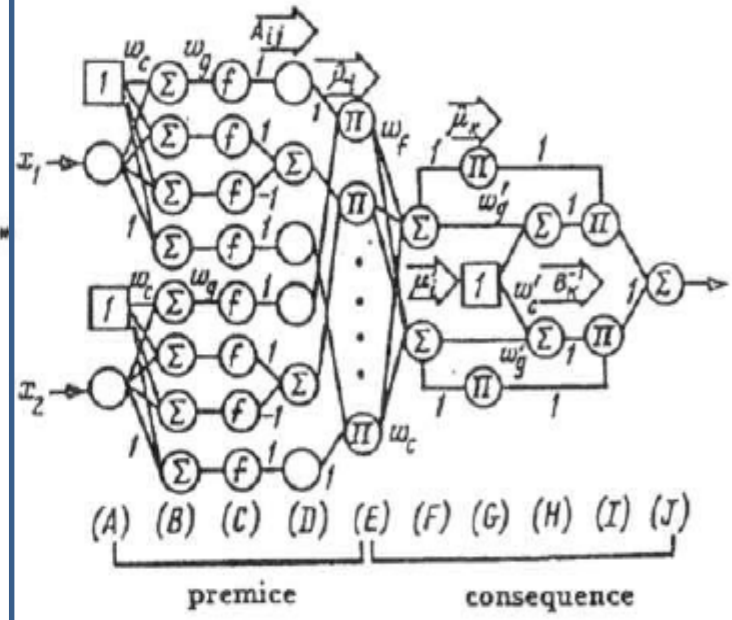
here  $\tau_{R_k^i} \in [0, 1]$  is linguistic truth value of the fuzzy rule  $R_k^i$ ,  $\mu_k^*$ , i.e., the truth value of the consequence, and  $B_k^{-1}(\mu_k^*)$  is the universe function of the membership function in the consequence  $B_k(y)$ , layers E through J are the consecutive part;  $\mu_k^*$  and  $B_k^{-1}(\mu_k^*)$  are the outputs of the units in layers F and H, respectively; the normalized truth value in the consequence  $\tilde{\mu}_k^*$  is calculated in layer G, and the inferred value is obtained as the sum of the product of  $\tilde{\mu}_k^*$  and  $B_k^{-1}(\mu_k^*)$  in layer I through J.



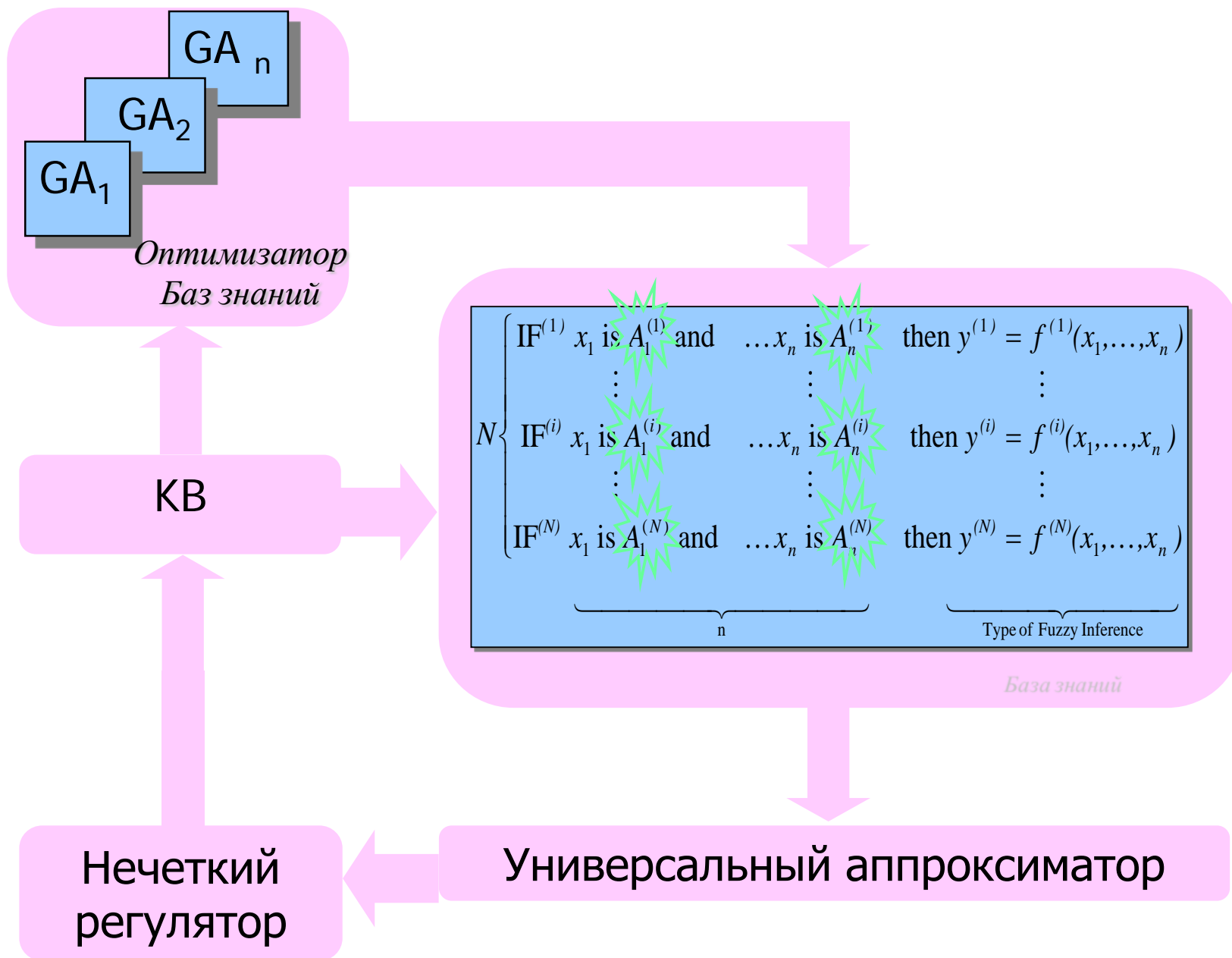
(a) Model I



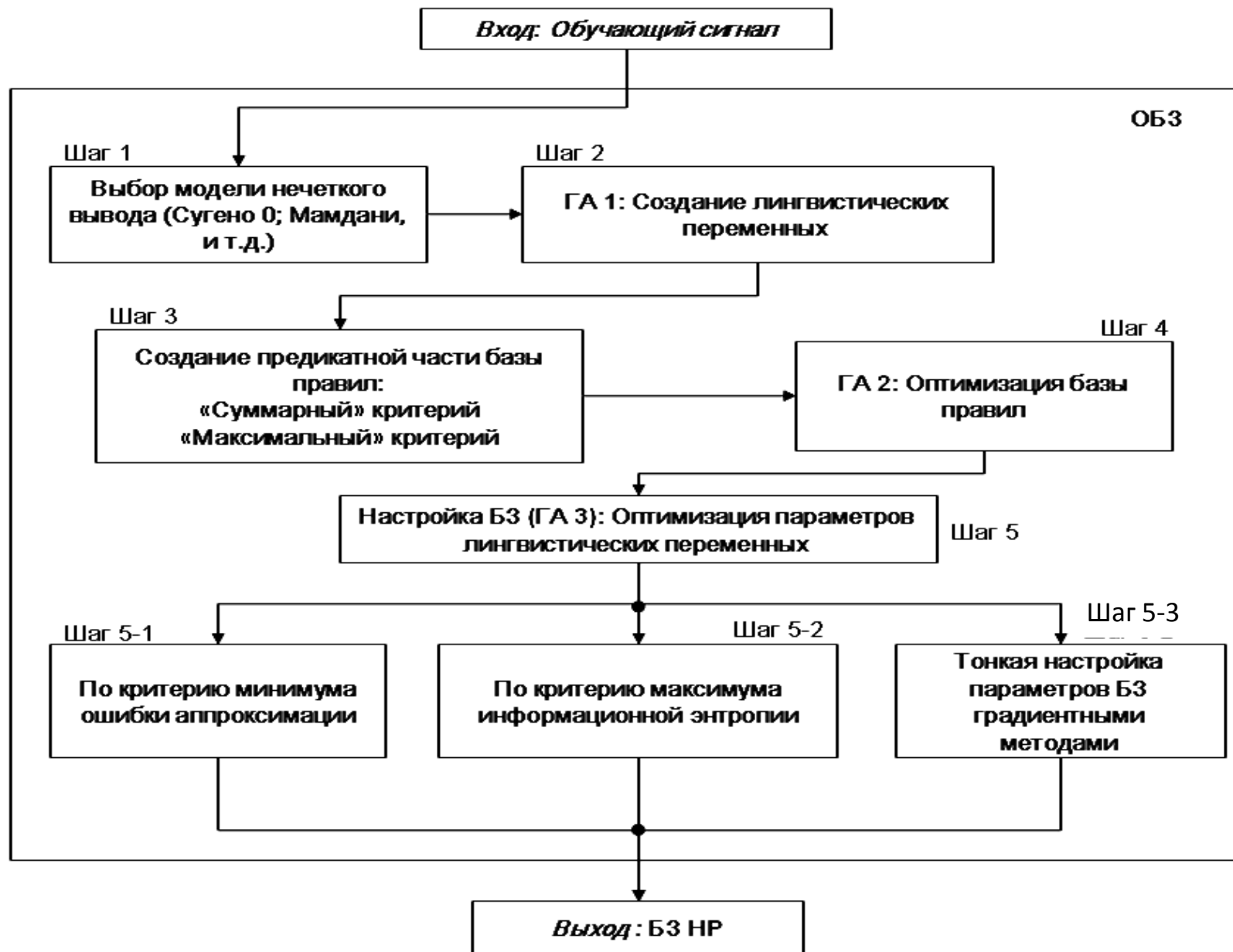
(b) Model II



(c) Model III









## Традиционная структура продукционных правил Если А То В

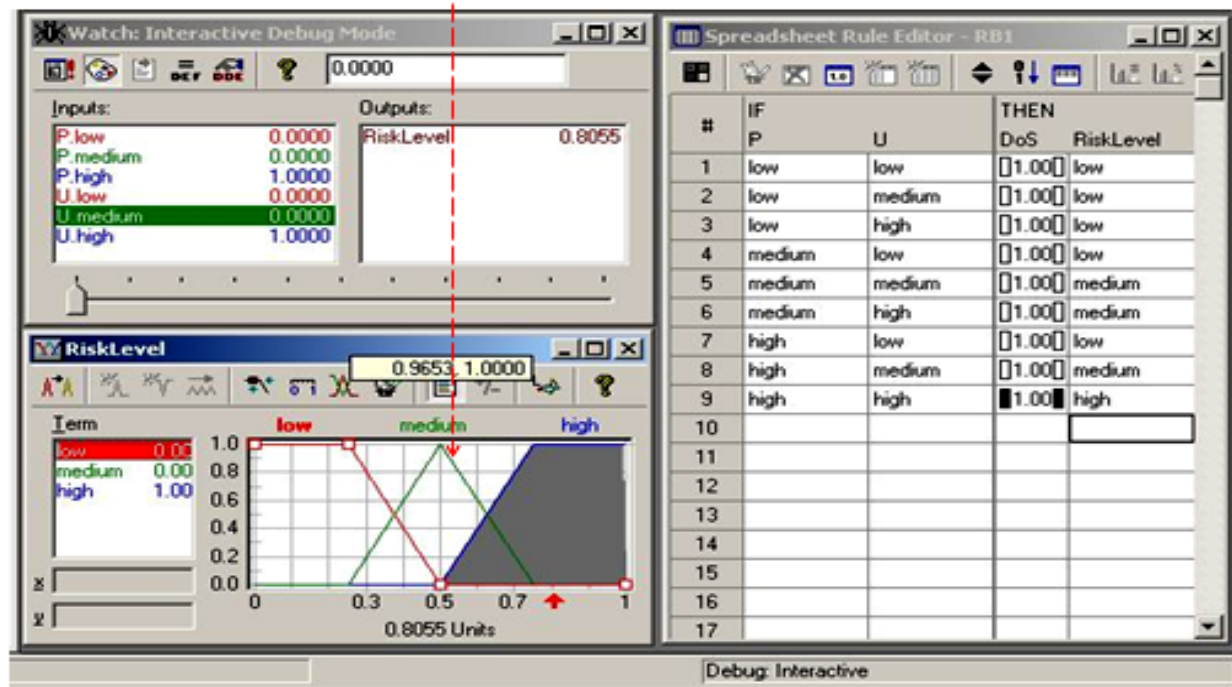
Если ошибка  $e$  *малая* величина, скорость ошибки  $\dot{e}$  *средняя* величина и накопление ошибки  $\int edt$  *средняя* величина,

То параметры регулятора  $k_p$  *большая* величина,  $k_d$  *малая* величина,  $k_I$  *средняя* величина.

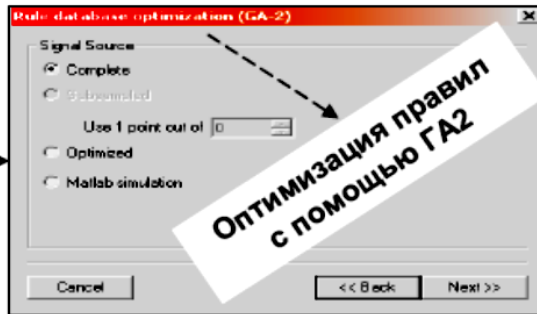
## Структура с информационным риском $\Delta\rho$

Если ошибка  $e$  *малая* величина, скорость ошибки  $\dot{e}$  *средняя* величина, накопление ошибки  $\int edt$  *средняя* величина, и степень приращения риска  $\Delta\rho$  *средняя* величина,

То параметры регулятора  $k_p$  *средняя* величина,  $k_d$  *большая* величина,  $k_I$  *малая* величина

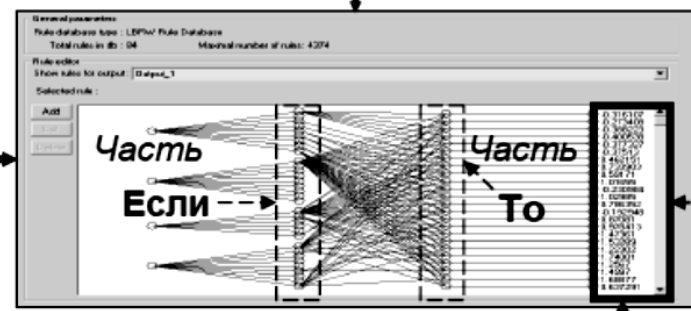
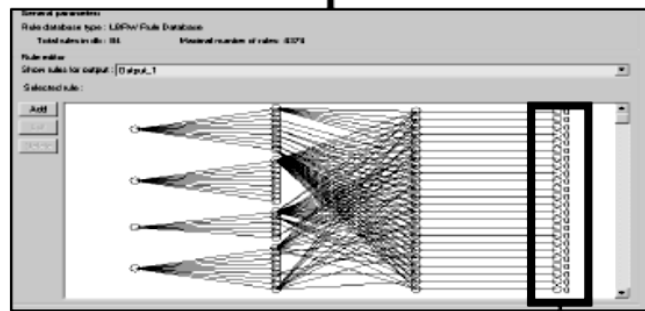


**Обучающий сигнал**



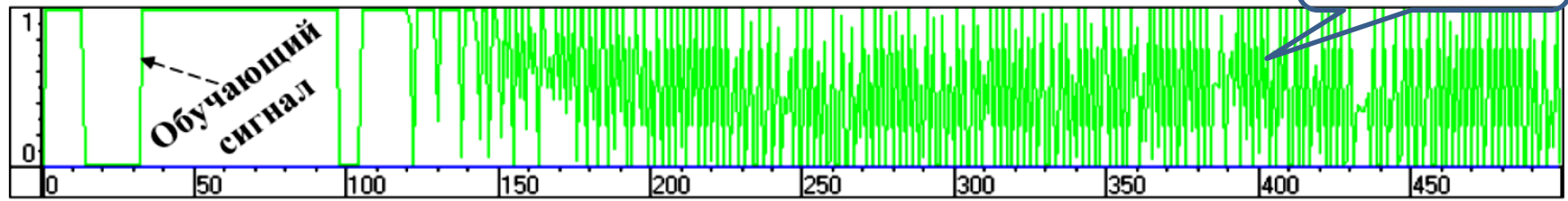
Оптимизация количества правил осуществляется при достижении заданной ошибки аппроксимации с помощью ГА2

**Аппроксимация обучающего сигнала**

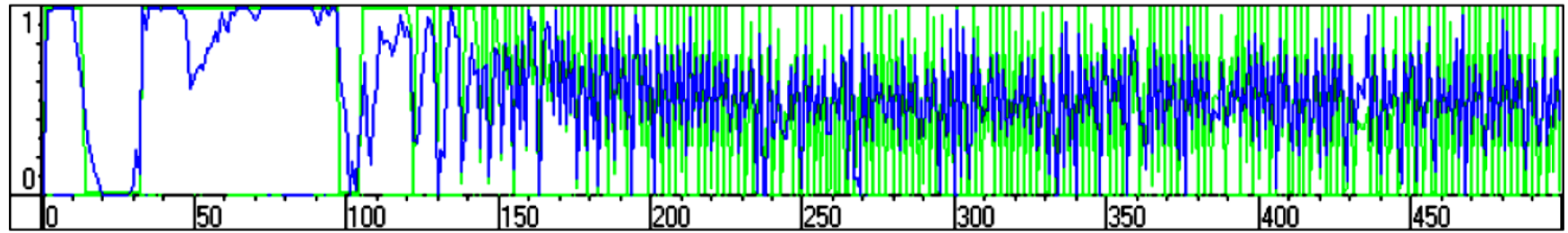


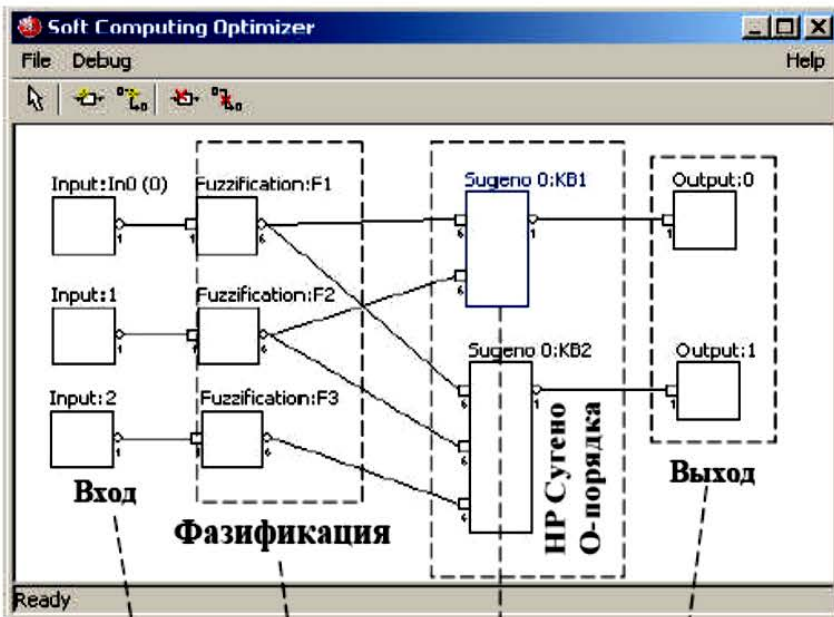
Параметры выходного сигнала

Redundant information



**Оптимальная аппроксимация обучающего сигнала**





Sugeno 0:KB1

Block Name : KB1 **Б31** Rename Close

Statistics

Rule database type : Complete database

Total rules in db : 25 Maximal number of rules:25

Rule editor

Show rules for output 1

Selected rule : If F1.1=F1.F1\_1 & F2.1=F2.F2\_3 Then out1=0.4

**Продукционное правило**

Add	N	F1.1	F2.1	out1
	0	F1.F1_1	F2.F2_1	0.2
Edit	1	F1.F1_1	F2.F2_2	0.11
	2	F1.F1_1	F2.F2_3	0.4
Delete	3	F1.F1_1	F2.F2_4	0.3
	4	F1.F1_1	F2.F2_5	0.4

Graphical Table

Sugeno 0:KB1

Block Name : KB1 Rename Close

Statistics

Rule database type : Complete database

Total rules in db : 25 Maximal number of rules:25

Rule editor

Show rules for output 1

Selected rule : If F1.1=F1.F1\_1 & F2.1=F2.F2\_3 Then out1=0.4

**Продукционное правило**

Graphical Table

Формирование продукционных правил  
в виде: «Если А, То В»

If - part

F1.1=F1.F1\_1  
F2.1=F2.F2\_3

Then - part

out1=0.4

F1.F1\_1 0.4 set

Cancel OK

**Soft Computing Optimizer**

File Action View

General Variables Rule database

Model

File name: LVsampleGA2MLinEdEX.sco

Inference model: Sugeno order 0 fuzzy inference model

Inference mode: Product inference

Number of input variables: 2

Number of output variables: 3

Learning data

File name: TSLVsample.mat

Number of elements: 101

Log Simulation preview

Output\_2  
Output\_3  
Variables will be optimized together  
Matlab simulation  
Testing matlab fitness function  
-0.00799161 -0.0286861 -17.8154 Degree: 3  
GA-2 optimization completed

**Процедура вызова МатЛаб: Инициализация и команды оценок**

**Rule database optimization (GA-2)**

Matlab Parameters

Initialize session command:  
cdscript\vdp\_initlink

Fitness calculation command:  
SCO\_Fitness=fitsco

Close session command:

Test only

Cancel << Back Next >>

Create variables

Create rule database

Optimize rules

Refine KB

Back Propagation

GA Test

**modelDoubleVDPFC**

File Edit View Simulation Format Tools Help

Normal

Derivative

SCoptimizer

SCoptimi

**Симулинка модель**

**ControlError**

0.6  
0.4  
0.2  
0  
-0.2  
-0.4

0 10 20 30 40 50

Time offset: 0

**Ошибка аппроксимации**

**Controller gains**

10  
8  
6  
4  
2  
0

0 10 20 30 40 50

Time offset: 0

**Сигналы управления**

Start Release Soft Computing Optimizer 4 matlab 9.27

(51) Int.Cl. <sup>5</sup>	識別記号	F I		
G 0 5 B 13/02		G 0 5 B 13/02		M
				N
				Z
	13/04		13/04	
G 0 6 F 15/18	5 5 0	G 0 6 F 15/18	5 5 0 C	
審査請求 未請求 請求項の数6 FD (全10頁)				

(21) 出願番号 特願平9-87426

(22) 出願日 平成9年(1997)3月21日

(71) 出願人 557047900

セルゲイ ヴィ. ウリヤノフ

東京都調布市小島町1-1-1 公務員宿舎 R E 504

(71) 出願人 000010076

ヤマハ発動機株式会社

静岡県磐田市新貝2500番地

(72) 発明者 セルゲイ ヴィ. ウリヤノフ

東京都調布市小島町1-1-1 公務員宿舎 R E 504

(74) 代理人 弁理士 八木田 茂 (外1名)

УЛЬЯНОВ

Дата

1997.03.21

(54) 【発明の名称】 非線形の物理的な制御対象の最適制御のための自己組織化方法及び装置

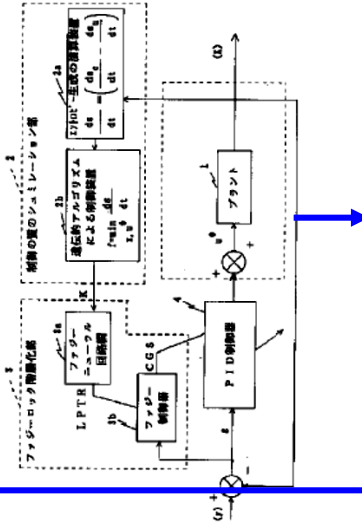
(57) 【要約】

【課題】 制御器設計の難しい非線形系の制御対象を制御できる新規な方法及び装置を提供する。

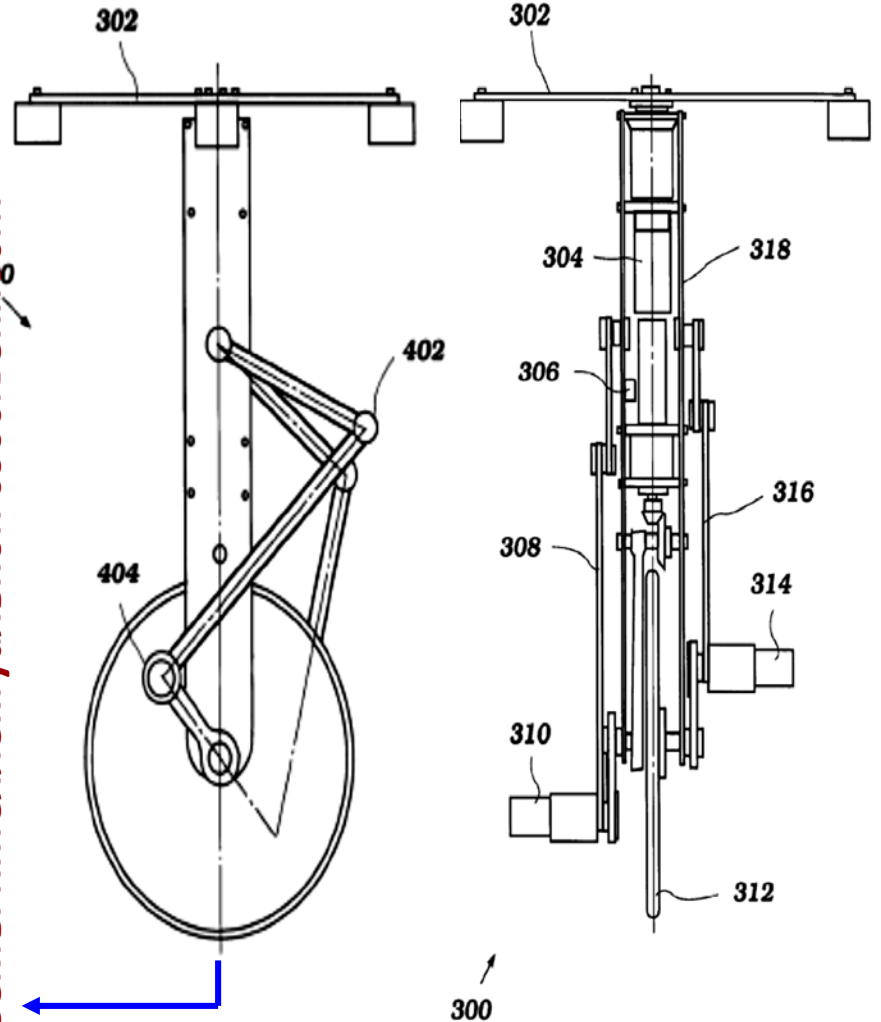
【解決手段】 インテリジェント制御システムと複雑な制御対象の動的挙動とにおける、生成エントロピーの最小化及びA I ロバスト制御システムの自己組織化を行うことを特徴とする。

【効果】 これにより非線形のエネルギー散逸モデルとして描かれる複雑で非線形な制御対象を最適制御することができるようになる。

Принцип минимума производства энтропии в системе и объекте управления



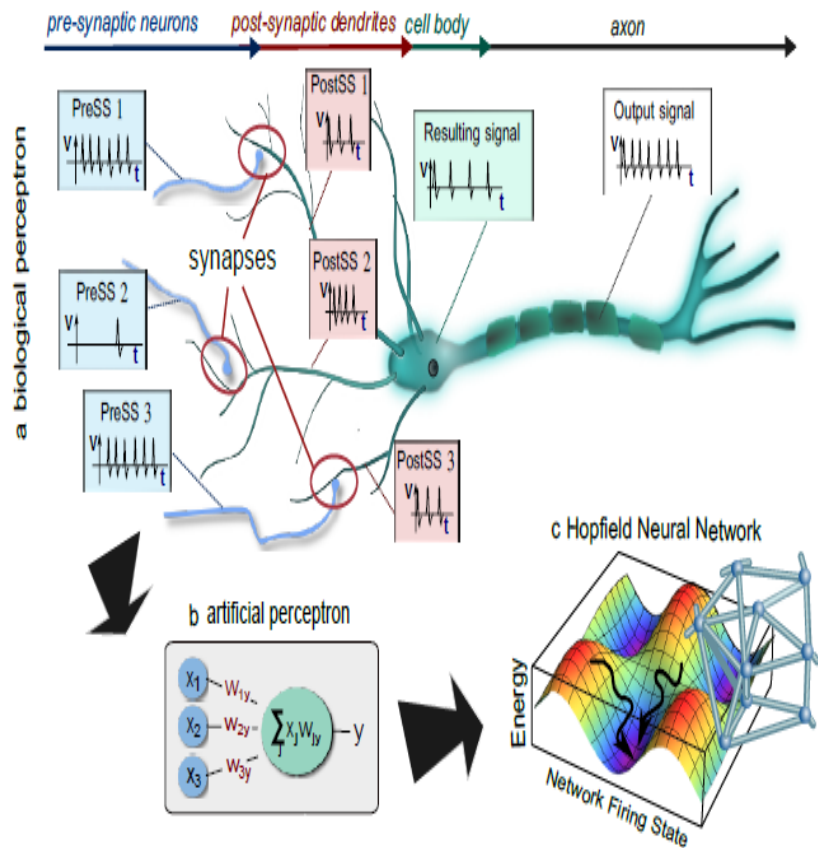
# Аппаратная реализация принципа самоорганизации на основе термодинамического принципа минимума производства энтропии



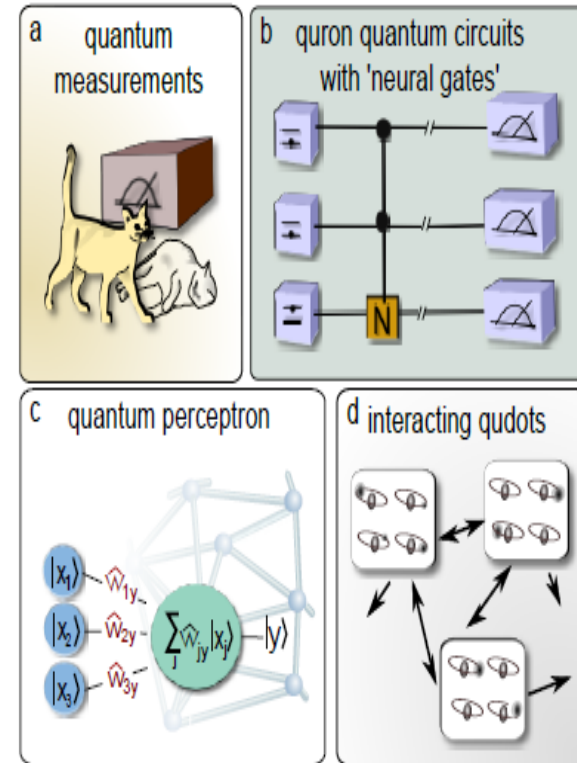
Объекты интеллектуальной собственности



# Quantum Neural Network



**Fig. 1** Illustration of the perceptron mechanism in neural computing. (a) In this illustration of a biological perceptron, three 'input neurons' (blue) feed their pre-synaptic signals into the 'output neuron' (green). The pre-synaptic signals are transformed into post-synaptic signals by synapses (red) and travel through the dendrites to the cell body where they add up to a resulting signal. The output signal that propagates along the axon and feeds into the next layer of neural cells nonlinearly depends on the strength of the resulting signal. (b) Artificial perceptrons are based on binary neurons of the states active/resting, represented by the values  $-1, 1$ . The input neurons are denoted by  $x_1, x_2, x_3 \in \{-1, 1\}$ , the synaptic weights are chosen to be  $w_{1y}, w_{2y}, w_{3y} \in [-1, 1]$  and  $y$ 's output is  $-1$  or  $1$ , depending on the resulting signal  $\sum_j x_j w_{jy}$ . (c) This simplified perceptron mechanism leads to rich dynamics in fully connected neural networks (here illustrated by a graph), where the global firing state of a neural network converges to local attractors, a feature that gives rise to associative memory, pattern classification etc.



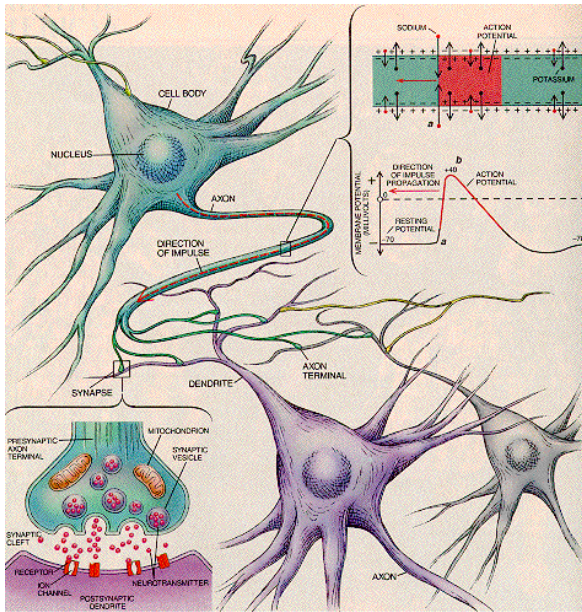
**Fig. 5** Different approaches to develop a Quantum Neural Network. (a) Several authors notice the analogy between the nonlinear stepfunction in the activation of a neuron and a measurement process, here symbolised by Schrödinger's famous cat. (b) Many contributions try to build quantum circuits with gates or features inspired by neural computing. (c) A challenge in finding a QNN model is to translate the core mechanism of a perceptron into a corresponding quantum version without losing the rich dynamics of neural networks. (d) An interesting branch of proposals understands interacting quantum dots consisting of four atoms sharing two electrons as a QNN.

$$|quron\rangle = \text{rest}|0\rangle + \text{active}|1\rangle$$

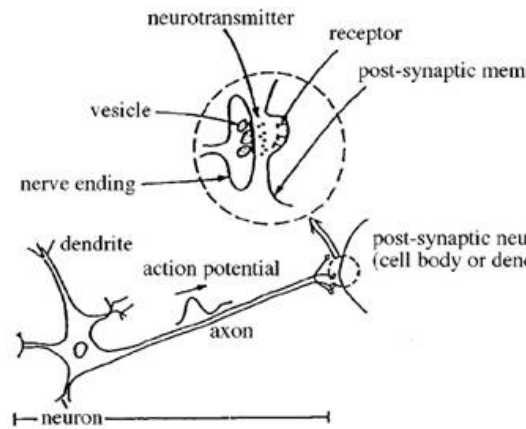
$$|\text{rest}\rangle^2 + |\text{active}\rangle^2 = 1$$



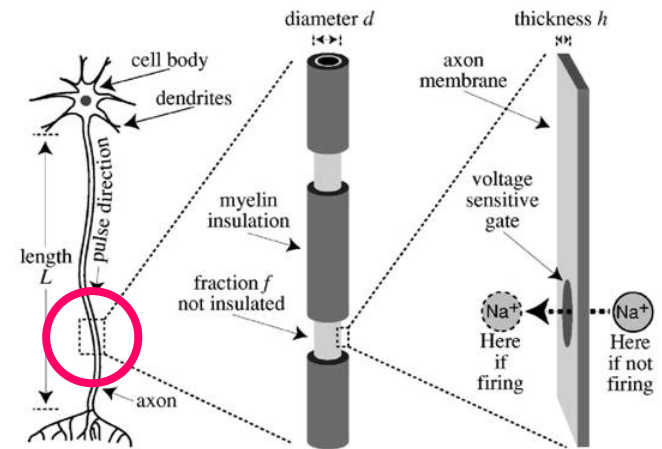
# Quantum Model Representation of Neural Network



*Natural Neural Network*



*Model representation of Natural Neuron*

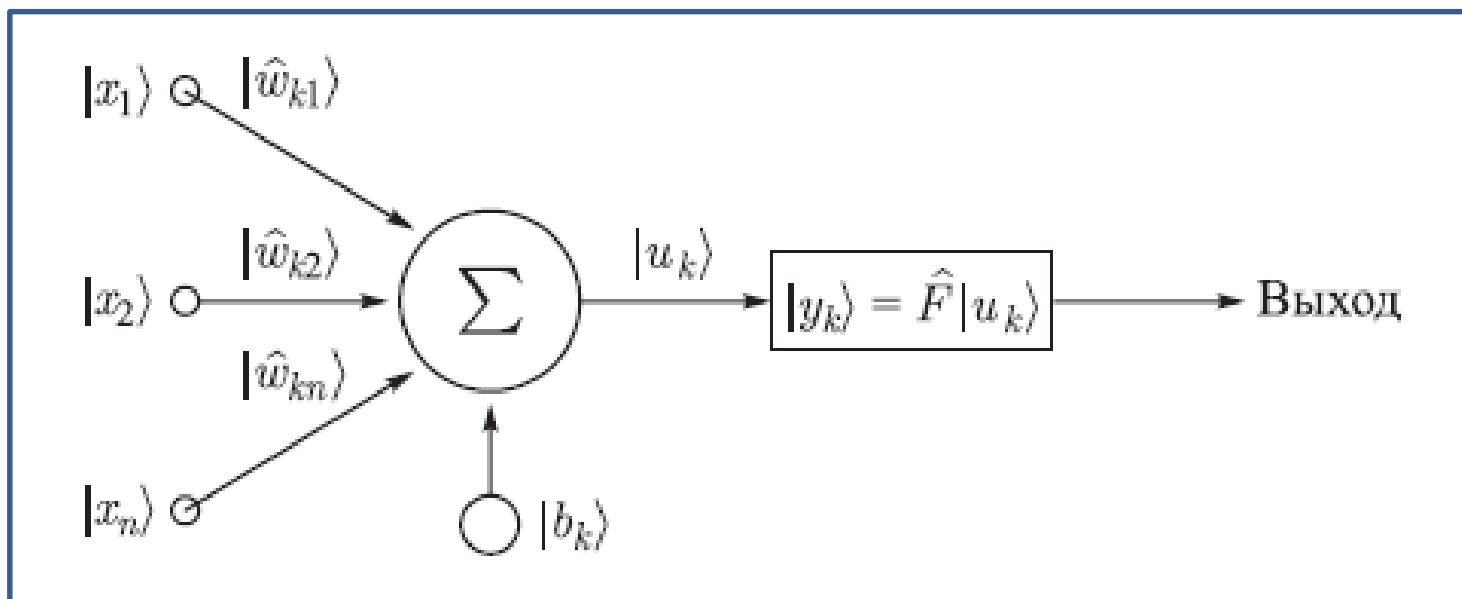


*Quantum level representation:*

*Quantum Nature of Communication Channel*

# Quantum computation and ANN: Quantum Neural Network (QNN)

N	The main concept of quantum computation:
1	<b>Wave function description of quantum computing evolution</b>
2	<b>Superposition of classical states (coherence)</b>
3	<b>Entanglement</b>
4	<b>Interference</b>
5	<b>Measurement (decoherence)</b>
6	<b>Unitary transformations (reversible computation)</b>



Блок-схема квантового нейрона

*Quantum computation and ANN:  
Quantum Neural Network (QNN)*

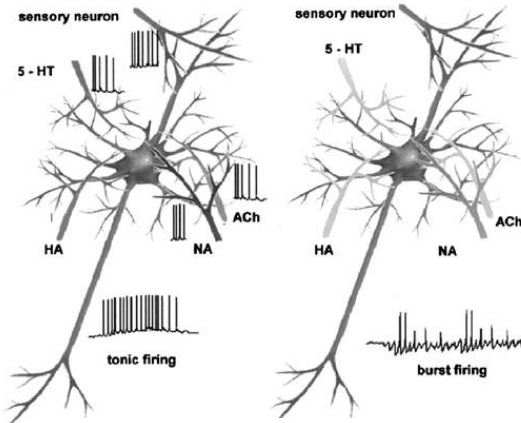
**Corresponding concepts of classical ANN and QNN**

<i>Classical neural networks</i>		<i>Quantum neural networks</i>	
<b>Neuronal state</b>	$x_j \in \{0, 1\}$	<b>Qubits</b>	$ x\rangle = a 0\rangle + b 1\rangle$
<b>Connection</b>	$\{w_{ij}\}_{ij=1}^{p-1}$	<b>Entanglement</b>	$ x_0 x_1 \dots x_{p-1}\rangle$
<b>Learning rule</b>	$\sum_{s=1}^p x_i^s x_j^s$	<b>Superposition of entanglement states</b>	$\sum_{s=1}^p a_s  x_0^s \dots x_{p-1}^s\rangle$
<b>Winner search</b>	$n = \max_i \arg(f_i)$	<b>Interference as unitary transformation</b>	$U : \Psi \rightarrow \Psi'$
<b>Output result</b>	$N$	<b>Decoherence (measurement)</b>	$\sum_s a_s  x^s\rangle \rightarrow  x^k\rangle$

# Physical Interpretation of Neuro-biological Experimental Mechanisms

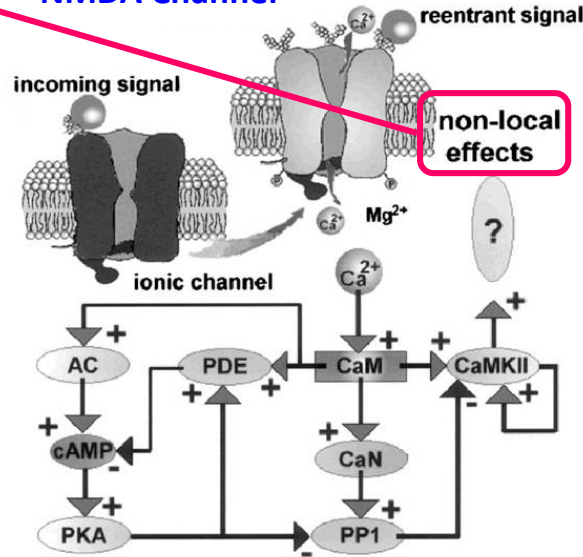
Entanglement

NMDA Channel



incoming signal

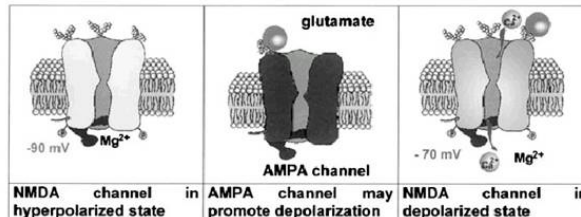
non-local effects



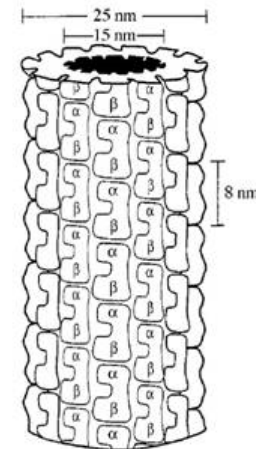
The Microtubule

Superposition

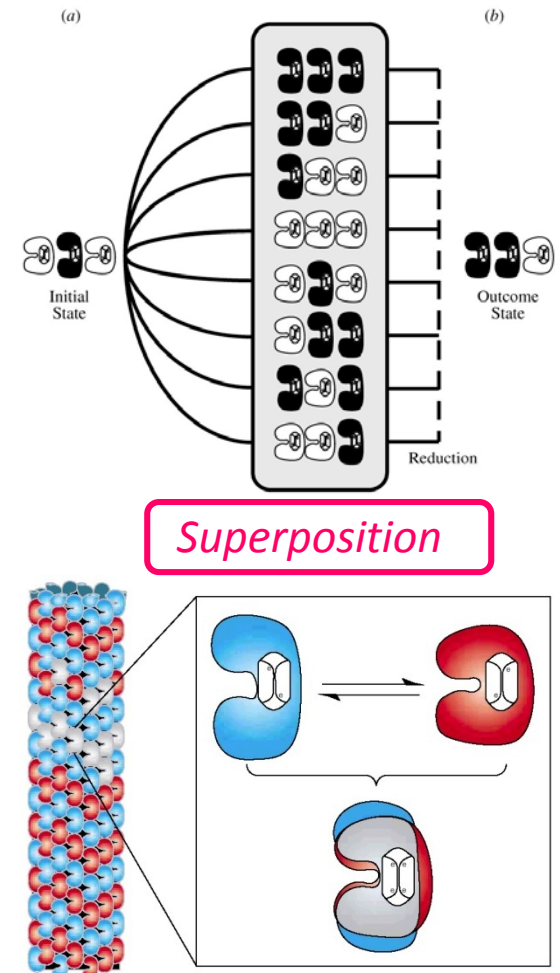
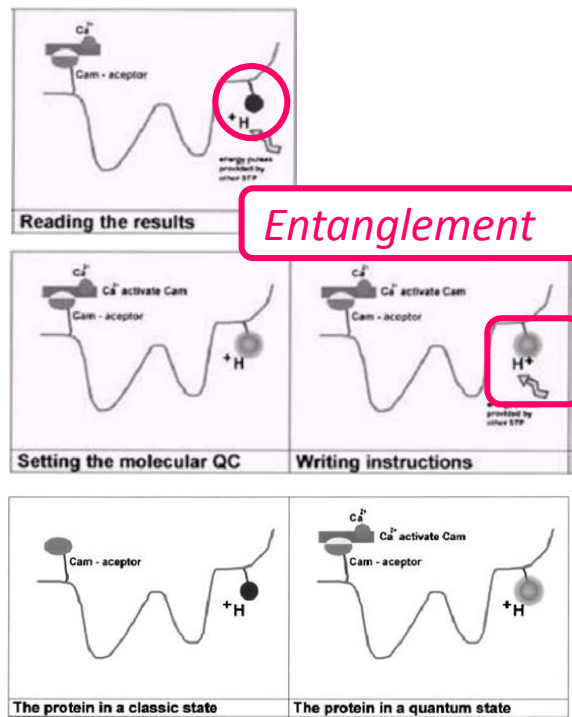
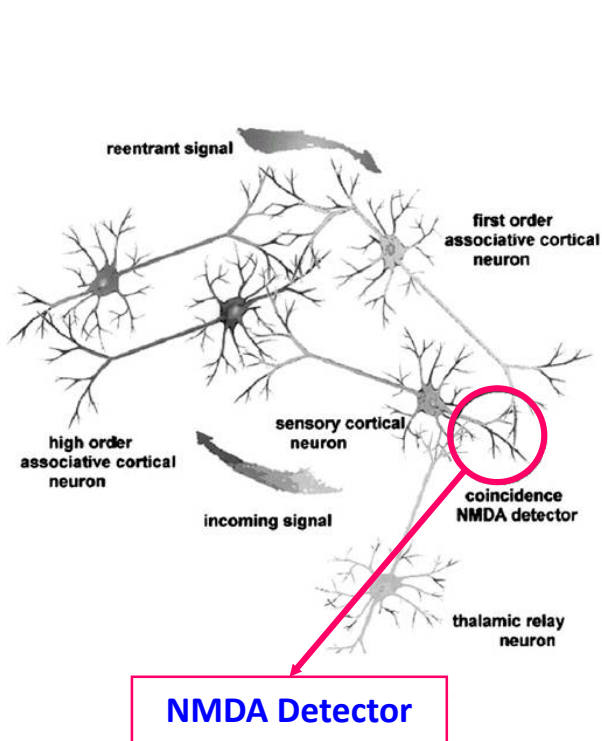
Different Coding of Communications between Neurons



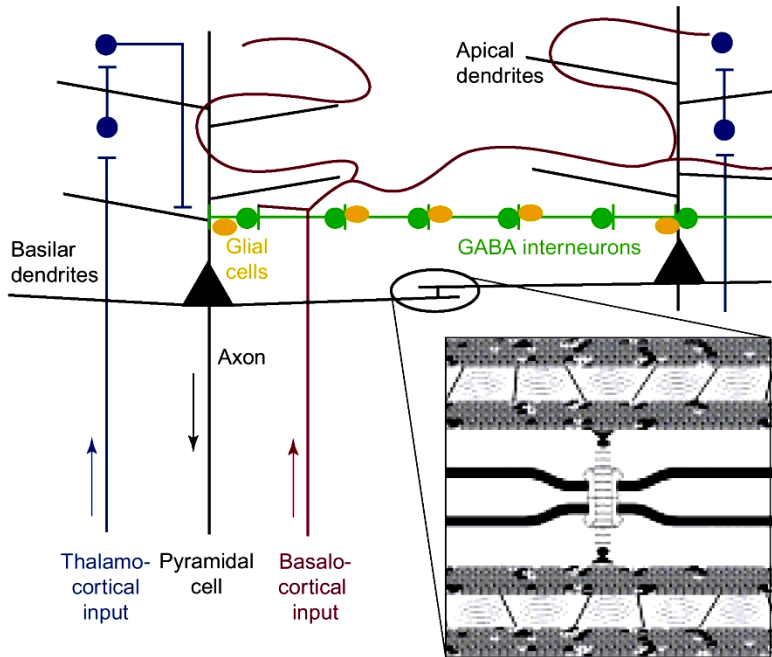
Detector System of Neuron



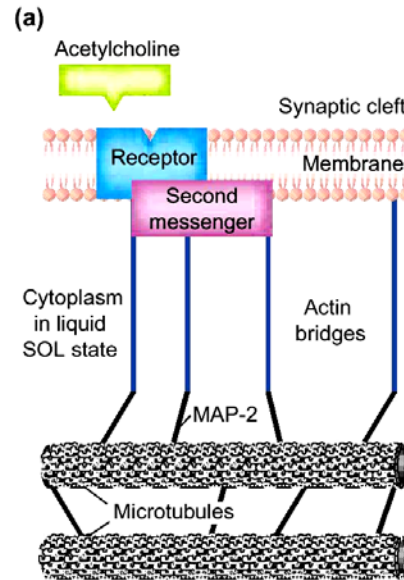
# Physical Interpretation of Macro Entanglement & Micro Superposition



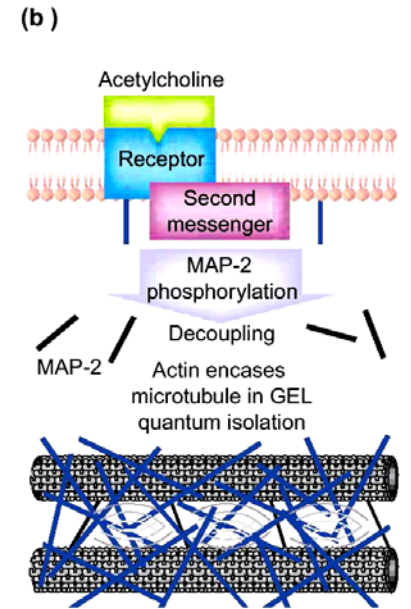
# Physical Interpretation of Neuro-biological Measurement



**Quantum Tunneling**  
40Hz (25micro sec.)switching

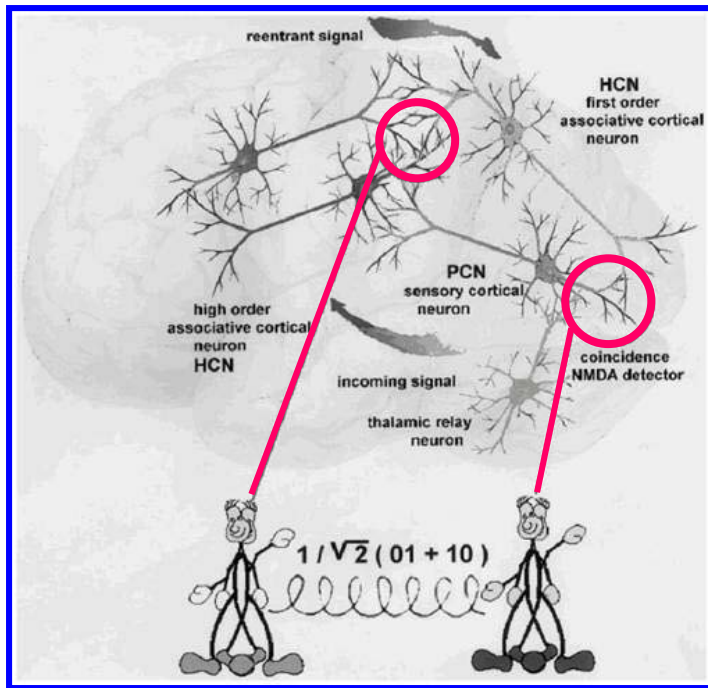


**Classical representation**



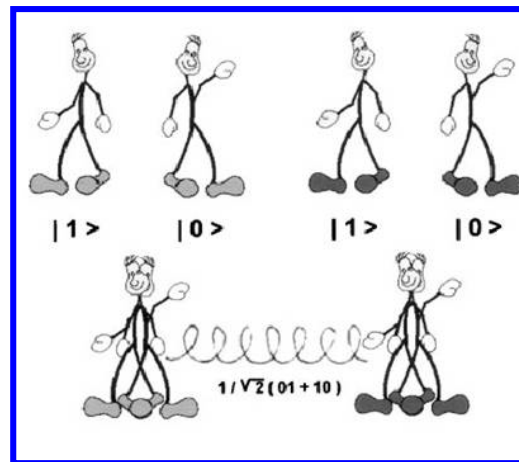
**Quantum representation**

# Macro level description of Entanglement in neural network

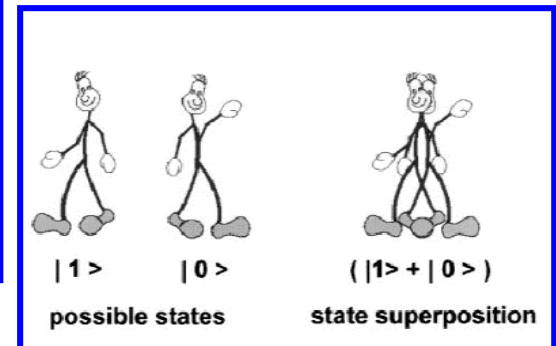


Background for Quantum Artificial Neural Network Structure Design

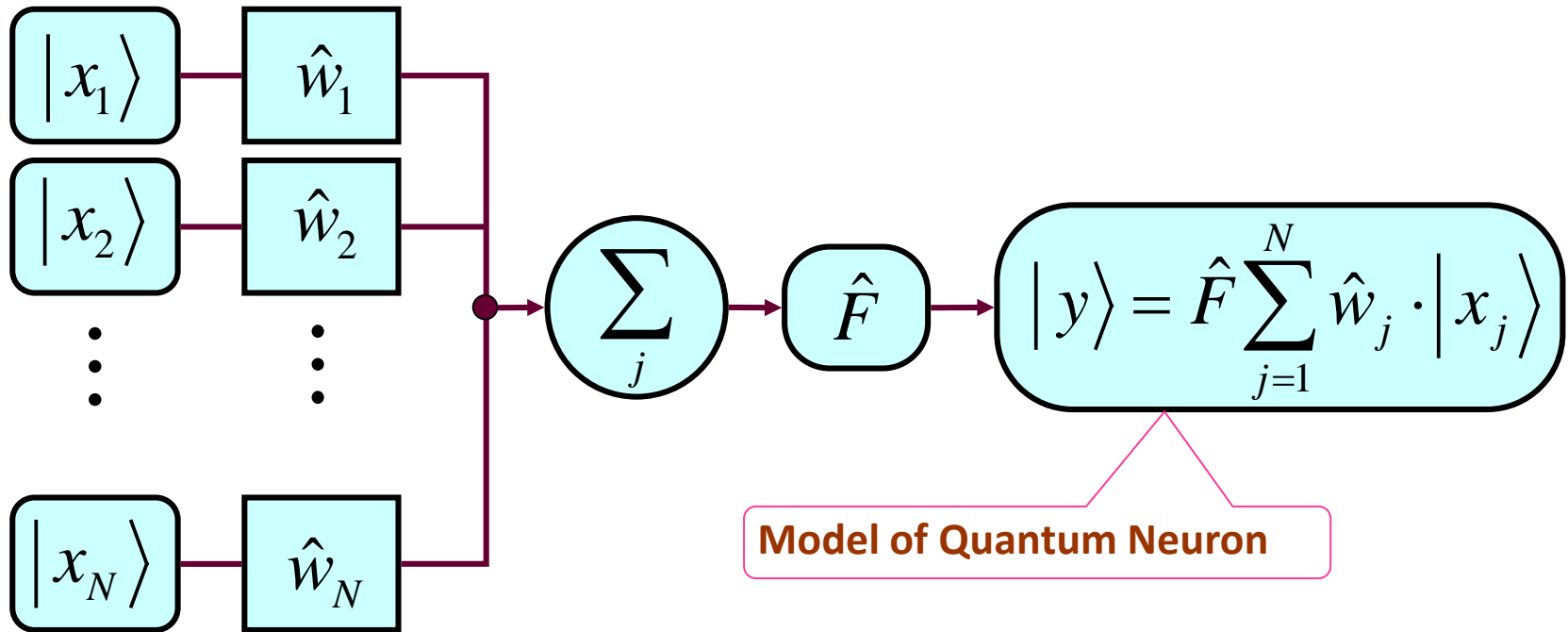
Entanglement



Superposition



## Model of Quantum Neuron (QN)



**Mathematical model of QN is  $|y\rangle = \hat{F} \sum_{j=1}^N \hat{w}_j |x_j\rangle$ , where  $\hat{w}_j$  is  $2 \times 2$  matrices acting on the basis  $\{|0\rangle, |1\rangle\}$ ,  $\hat{F}$  is an operator that can be implemented by the network of quantum gates.**



# Example: *Learning process of quantum neuron*

Let us  $\hat{F} = \hat{I}$  is the identity operator:  $|y\rangle = \hat{I} \sum_{j=1}^N \hat{w}_j |x_j\rangle = \sum_{j=1}^N \hat{w}_j |x_j\rangle$ .

The quantum learning rule is provided in analogy with classical case as following:

$\hat{w}_j(t+1) = \hat{w}_j(t) + \eta(|d\rangle - |y(t)\rangle)\langle x_j|$ , where  $|d\rangle$  is the desired output.

This learning rule drives the quantum neuron into desired state  $|d\rangle$  used for teaching.

Taking the module-square difference of the real and desired outputs, we yield as following:

$$\begin{aligned} \left\| |d\rangle - |y(t+1)\rangle \right\|^2 &= \left\| |d\rangle - \sum_{j=1}^N \hat{w}_j(t+1) |x_j\rangle \right\|^2 \\ &= \left\| |d\rangle - \sum_{j=1}^N \left( \underbrace{\hat{w}_j(t) |x_j\rangle}_{=|y_j(t)\rangle} + \eta [ |d\rangle - |y(t)\rangle ] \underbrace{\langle x_j | x_j \rangle}_{=1} \right) \right\|^2 \\ &= \left\| |d\rangle - |y(t)\rangle - \sum_{j=1}^N \eta [ |d\rangle - |y(t)\rangle ] \right\|^2 \\ &= (1 - n\eta)^2 \left\| |d\rangle - |y(t)\rangle \right\|^2 = (1 - n\eta)^2 \left\| |\varepsilon\rangle \right\|^2 \end{aligned}$$

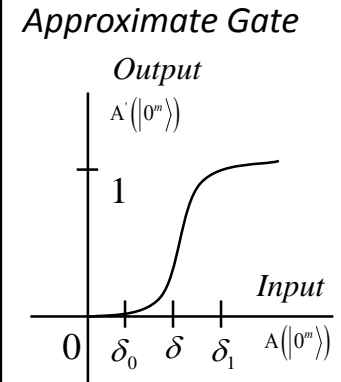
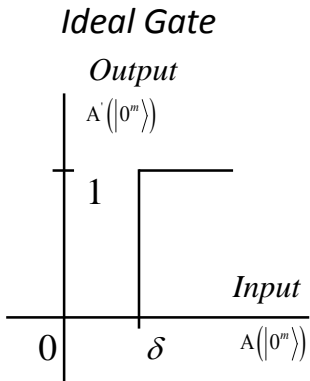
**Conclusion:** For small  $\eta \left( 0 < \eta < \frac{1}{n} \right)$  and normalized input states  $\langle x_j | x_j \rangle = 1$

the result of iteration converges in the desired state.

The whole network can be composed from the primitive elements using standard rules of ANN architectures.

# Quantum Neuron

	Wave function	Gate	Graphical Form
1	$\psi(\omega_0) = A \cdot \sin(\omega_0)$	1D NOT	
2	$\psi(\omega_0) = A \cdot \sin\left(\frac{\omega_0 + \pi}{2}\right)$	1-input bipolar TRUE	
3	$\psi(\omega_0, \omega_1) = A \cdot \sin\left(\frac{n\omega_0}{a} \omega_0\right) \times \sin\left(\frac{n\omega_1}{a} \omega_1\right)$	2-input XOR	
	$\psi = A \cdot \sin(\omega_0) \times \sin\left[\frac{\omega_1 + \pi}{2}\right]$		



# Computational power of Quantum Neuron

N	Function Type
1	<p style="text-align: center;"><b><u>NOT-function:</u></b></p> <p><b>The output is</b> <math> y\rangle = \hat{F} \sum_{j=1}^N \hat{w}_j  x_j\rangle</math>, <math>\hat{w} = \begin{pmatrix} 1 &amp; 0 \\ 0 &amp; 1 \end{pmatrix} = \hat{I}_{2 \times 2}</math>, <math>\hat{F} = \begin{pmatrix} 0 &amp; 1 \\ 1 &amp; 0 \end{pmatrix} \equiv NOT</math></p> <p><b>For</b> <math> x\rangle =  0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}</math> <b>we can get</b> <math> y\rangle = \begin{pmatrix} 0 &amp; 1 \\ 1 &amp; 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} =  1\rangle</math>.</p> <p><b>For</b> <math> x\rangle =  1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}</math> <math> y\rangle = w x\rangle =  0\rangle</math></p> <p><b>Thus QN has the same power as NOT gate.</b></p>
2	<p style="text-align: center;"><b><u>The Walsh-Hadamard transformation:</u></b></p> <p><b>The output is</b> <math> y\rangle = \hat{F} \sum_{j=1}^N \hat{w}_j  x_j\rangle</math>, <math>\hat{w} = \begin{pmatrix} 1 &amp; 0 \\ 0 &amp; 1 \end{pmatrix} = \hat{I}_{2 \times 2}</math>, <math>\hat{F} = \hat{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 &amp; 1 \\ 1 &amp; -1 \end{pmatrix}</math>.</p> <p><b>For</b> <math> x\rangle =  0\rangle</math> <b>we obtain</b> <math> y\rangle = \frac{1}{\sqrt{2}} ( 0\rangle +  1\rangle)</math>.</p> <p><b>For</b> <math> x\rangle =  1\rangle</math> <b>we obtain</b> <math> y\rangle = \frac{1}{\sqrt{2}} ( 0\rangle -  1\rangle)</math>.</p> <p><b>Thus Walsh-Hadamard transformation can be implemented by a QN with <math>2^n</math> inputs and one output choosing</b> <math>\hat{w} = \frac{1}{\sqrt{2}} \hat{I}</math> <b>and</b> <math>\hat{F} = \hat{H}</math>, <b>and QN can implemented the superposition of classical states as</b> <math>\frac{1}{\sqrt{2^n}} \sum_{j=1}^{2^n-1}  x_j\rangle</math>.</p>

# Computational power of Quantum Neuron

N	Function Type																				
3	<p style="text-align: center;"><b><u>XOR-function:</u></b></p> <p><b>QN have two inputs</b> <math>( x_1\rangle,  x_2\rangle)</math> <b>and the output is</b></p> <p><b>Choosing</b> <math>\hat{w}_1 = \hat{w}_2 = \hat{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 &amp; 1 \\ 1 &amp; -1 \end{pmatrix}</math> <b>and</b> <math>\hat{F} = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 &amp; 1 \\ 1 &amp; -1 \end{pmatrix} \begin{pmatrix} \text{sign}(\ast) &amp; 0 \\ 0 &amp; \text{sign}(\ast) \end{pmatrix}</math>,</p> <p><b>Where</b> <math>\text{sign}(\ast)</math> <b>is sign function, we can produce the value of the XOR function as following:</b></p> <div style="text-align: right; margin-right: 100px;"> <div style="border: 1px solid black; padding: 2px; display: inline-block;">XOR function</div>  <math>\Downarrow</math> </div> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th><math> x_1\rangle</math></th> <th><math> x_2\rangle</math></th> <th><math>\hat{w}_1 x_1\rangle + \hat{w}_2 x_2\rangle</math></th> <th><math> y\rangle</math></th> </tr> </thead> <tbody> <tr> <td><math> 0\rangle</math></td> <td><math> 0\rangle</math></td> <td><math>(\sqrt{2}, \sqrt{2})^T</math></td> <td><math> 0\rangle</math></td> </tr> <tr> <td><math> 0\rangle</math></td> <td><math> 1\rangle</math></td> <td><math>(\sqrt{2}, 0)^T</math></td> <td><math> 1\rangle</math></td> </tr> <tr> <td><math> 1\rangle</math></td> <td><math> 0\rangle</math></td> <td><math>(\sqrt{2}, 0)^T</math></td> <td><math> 1\rangle</math></td> </tr> <tr> <td><math> 1\rangle</math></td> <td><math> 1\rangle</math></td> <td><math>(\sqrt{2}, -\sqrt{2})^T</math></td> <td><math> 0\rangle</math></td> </tr> </tbody> </table>	$ x_1\rangle$	$ x_2\rangle$	$\hat{w}_1 x_1\rangle + \hat{w}_2 x_2\rangle$	$ y\rangle$	$ 0\rangle$	$ 0\rangle$	$(\sqrt{2}, \sqrt{2})^T$	$ 0\rangle$	$ 0\rangle$	$ 1\rangle$	$(\sqrt{2}, 0)^T$	$ 1\rangle$	$ 1\rangle$	$ 0\rangle$	$(\sqrt{2}, 0)^T$	$ 1\rangle$	$ 1\rangle$	$ 1\rangle$	$(\sqrt{2}, -\sqrt{2})^T$	$ 0\rangle$
	$ x_1\rangle$	$ x_2\rangle$	$\hat{w}_1 x_1\rangle + \hat{w}_2 x_2\rangle$	$ y\rangle$																	
$ 0\rangle$	$ 0\rangle$	$(\sqrt{2}, \sqrt{2})^T$	$ 0\rangle$																		
$ 0\rangle$	$ 1\rangle$	$(\sqrt{2}, 0)^T$	$ 1\rangle$																		
$ 1\rangle$	$ 0\rangle$	$(\sqrt{2}, 0)^T$	$ 1\rangle$																		
$ 1\rangle$	$ 1\rangle$	$(\sqrt{2}, -\sqrt{2})^T$	$ 0\rangle$																		

## Example:

### *Nonlinear mapping – QN with $N$ input and one output*

---

**The output**  $|y\rangle$  **is**  $|y\rangle = \hat{F} \sum_{j=1}^N \hat{w}_j |x_j\rangle = \hat{F} \sum_{j=1}^N \begin{pmatrix} w_{1j} & w_{3j} \\ w_{2j} & w_{4j} \end{pmatrix} \begin{pmatrix} a_j \\ b_j \end{pmatrix}$

$$= \hat{F} \sum_{j=1}^N \begin{pmatrix} w_{1j} a_j + w_{3j} b_j \\ w_{2j} a_j + w_{4j} b_j \end{pmatrix}$$
$$= \hat{F} \begin{pmatrix} \sum_{j=1}^n w_{1j} a_j + \sum_{j=1}^n w_{3j} b_j \\ \sum_{j=1}^n w_{2j} a_j + \sum_{j=1}^n w_{4j} b_j \end{pmatrix}$$

**For this case the output vector  $\hat{Y}$  of classical ANN can be written as**

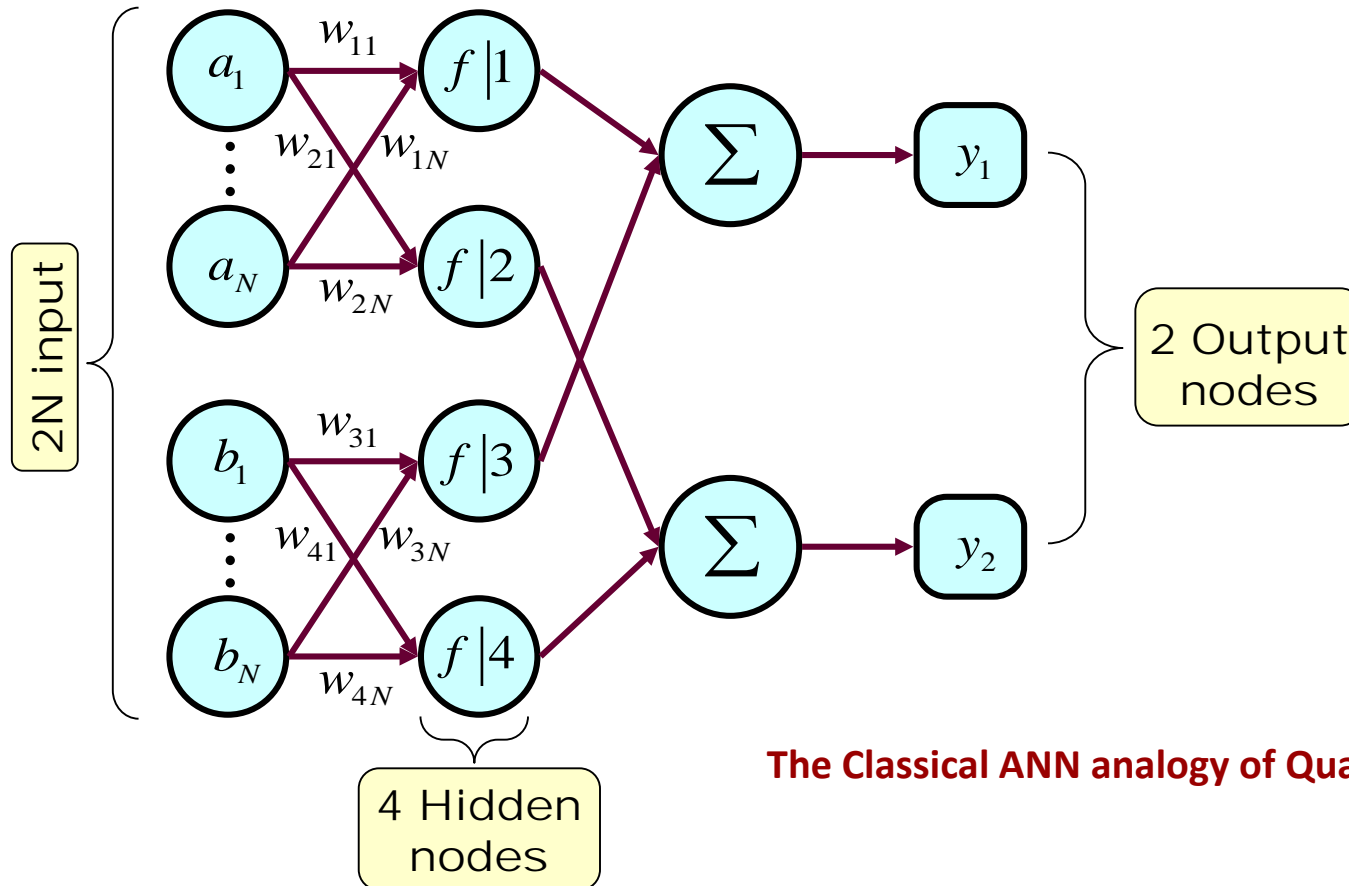
$$\hat{Y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} f \left( \sum_{j=1}^n w_{1j} a_j \right) + f \left( \sum_{j=1}^n w_{3j} b_j \right) \\ f \left( \sum_{j=1}^n w_{2j} a_j \right) + f \left( \sum_{j=1}^n w_{4j} b_j \right) \end{pmatrix}$$

**If in QN the operator  $\hat{F} = \hat{I}$  and the function  $f(x) = x$ , we can obtain  $\hat{Y} = |y\rangle$ .**

# Example: *Nonlinear mapping – QN with $N$ input and one output*

## Conclusion:

The QN has the same computational power as the classical ANN with  $2N$  input nodes, four (4) hidden nodes and two (2) output nodes.



**The Classical ANN analogy of Quantum Neuron**

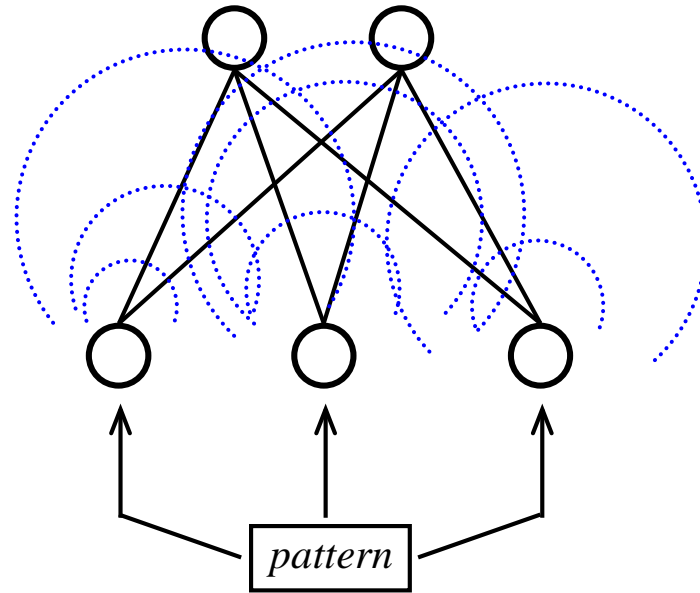
# Quantum Neural Network (General)

*CLASSICAL*

Output neurons  
(level n)

Connections

Input neurons  
(level 1)



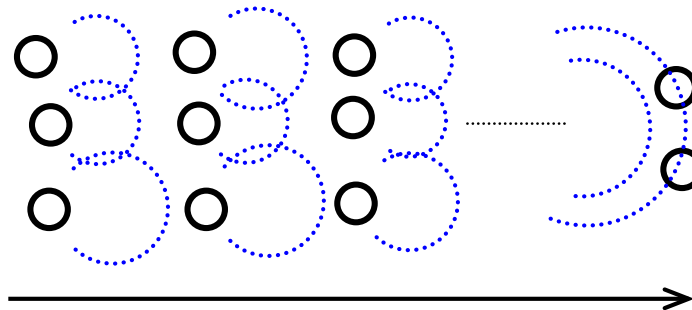
*QUANTUM*

Detector screen  
(pattern interference)

Superposition and phase shifting

Slits  
(unitary evolution/  
entanglement)

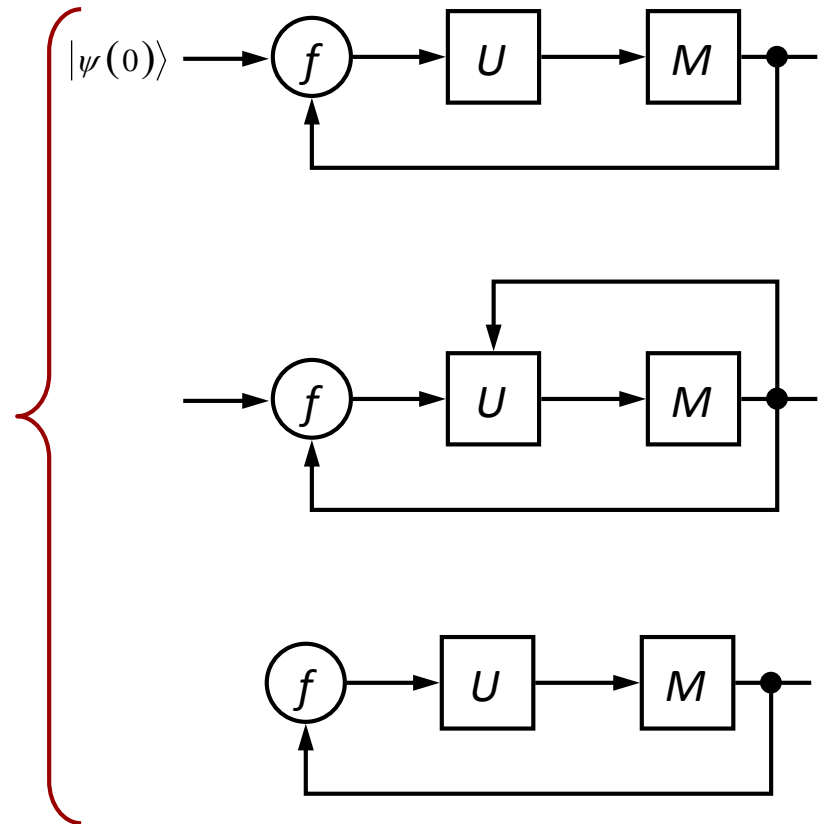
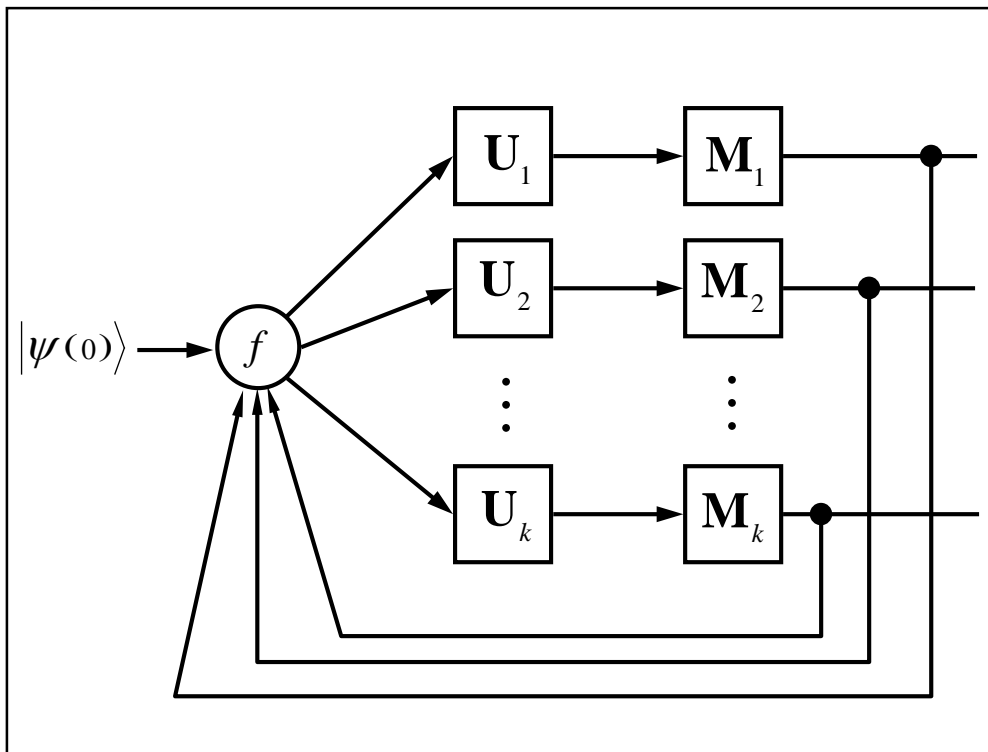
*input*



*output*

# Quantum Learning & Adaptation

## Quantum Recurrent Neural Network





# QNN as efficient unitary approximator

$$\hat{U}_j(x_j; f) |0_j\rangle = \sqrt{1 - f(x_j)} |0_j\rangle + \sqrt{f(x_j)} |1_j\rangle.$$

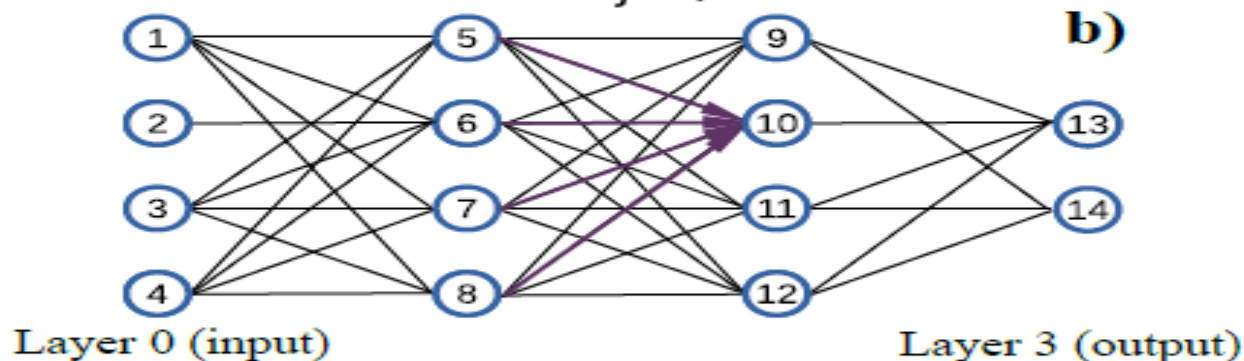
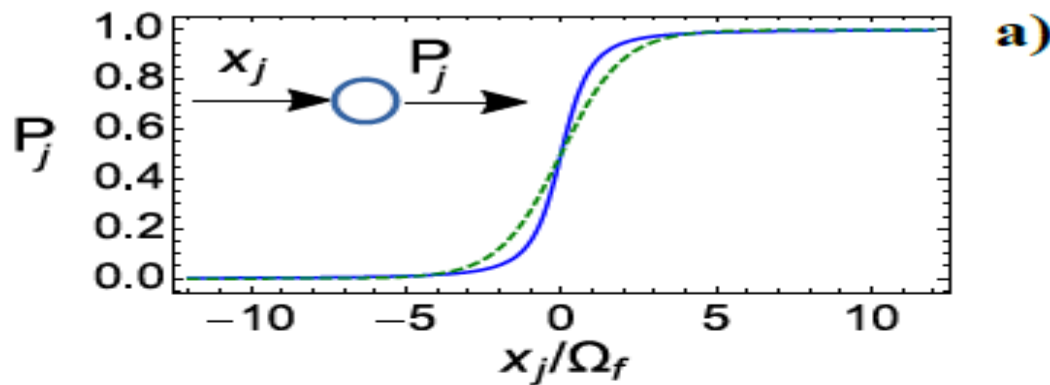
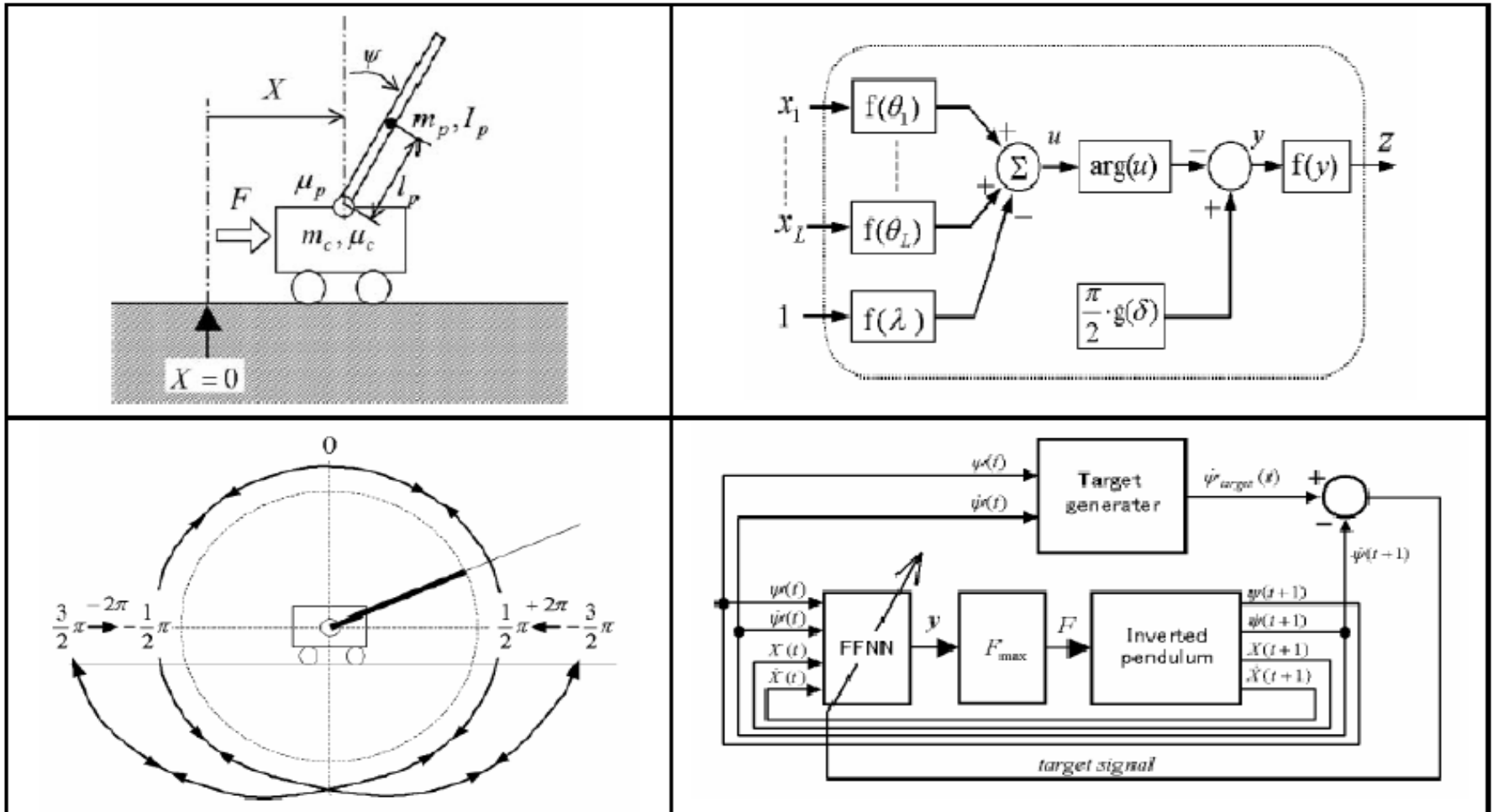


FIG. 1. (a) Quantum perceptron as a qubit that excites coherently according to (1) with a probability  $P_j = \frac{1}{2}(1 + \langle \hat{\sigma}_j^z \rangle) = f(x_j)$  that grows nonlinearly with the activation potential  $x_j$ . (b) When this perceptron is integrated in a feed-forward neural network, the potential depends on neurons in earlier layers, e.g.  $x_{10} = \sum_{k=5}^8 w_{10,k} \hat{\sigma}_k^z + \theta_{10}$ .

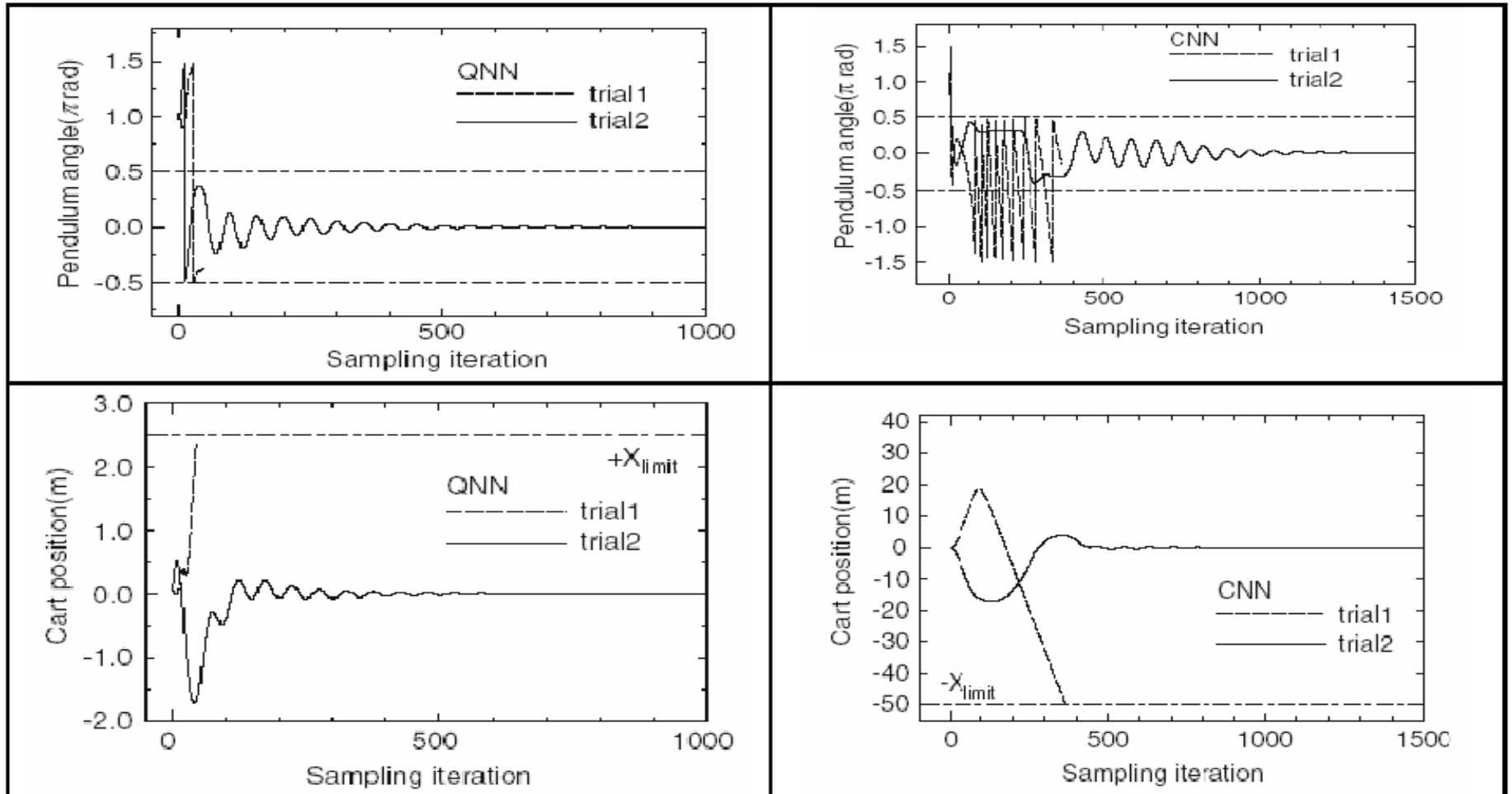
# QNN Application

- Intelligent control of inverted pendulum
- Visual–Motor Coordination using a Quantum Clustering based Neural Control Scheme
- Pattern recognition

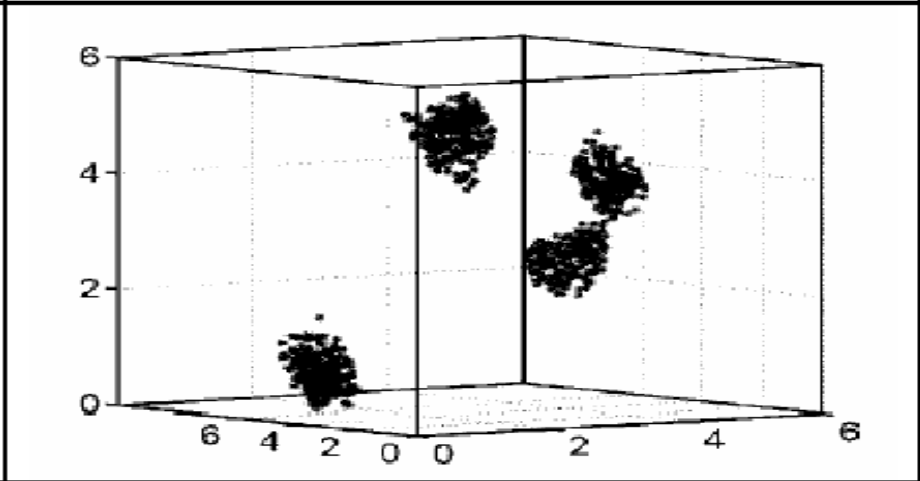
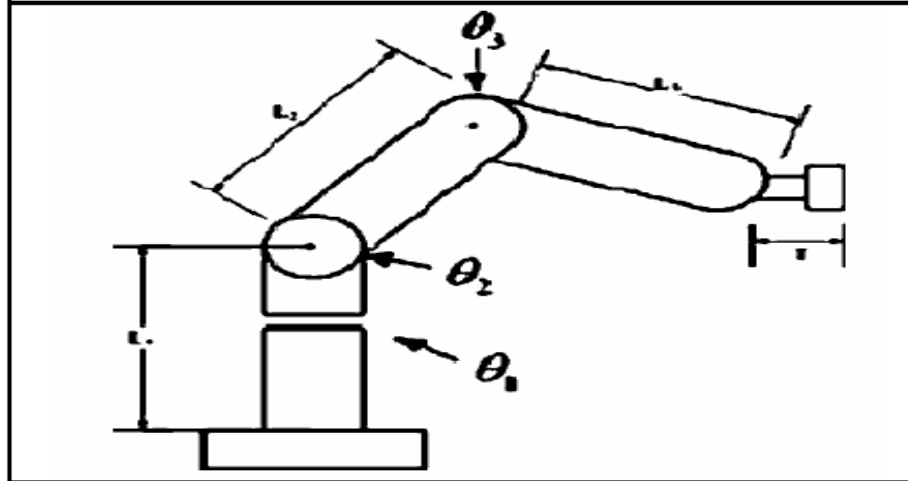
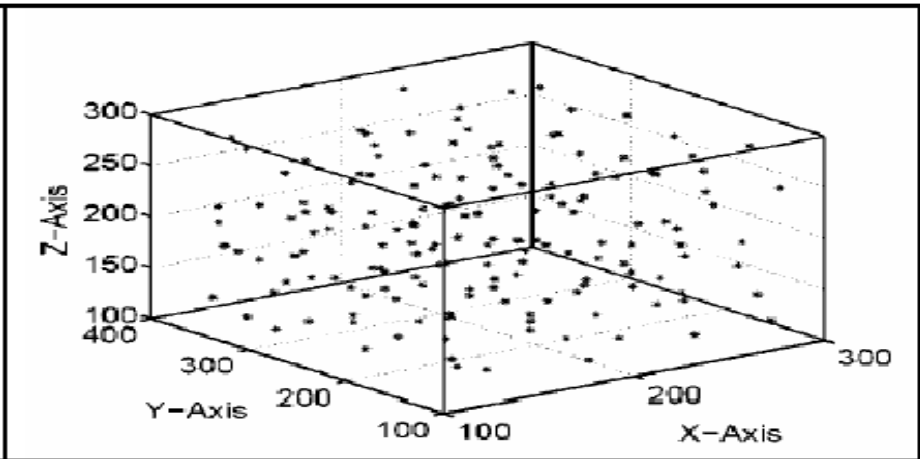
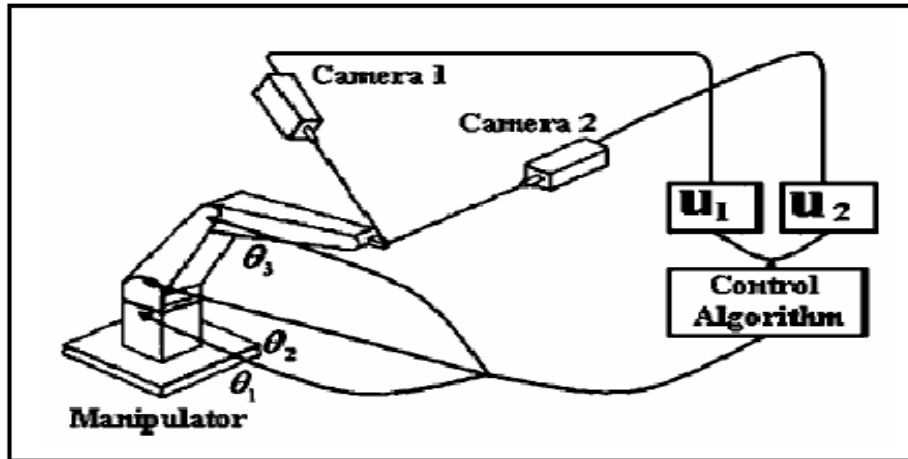
# Example: Inverted pendulum



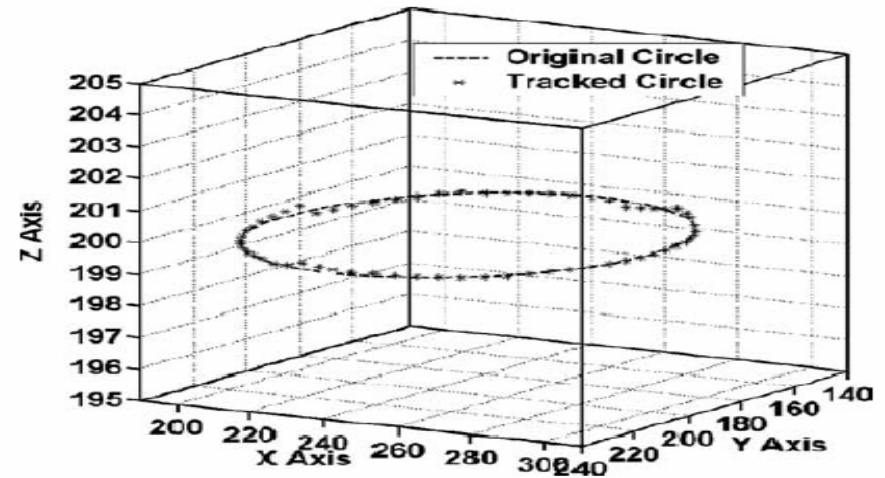
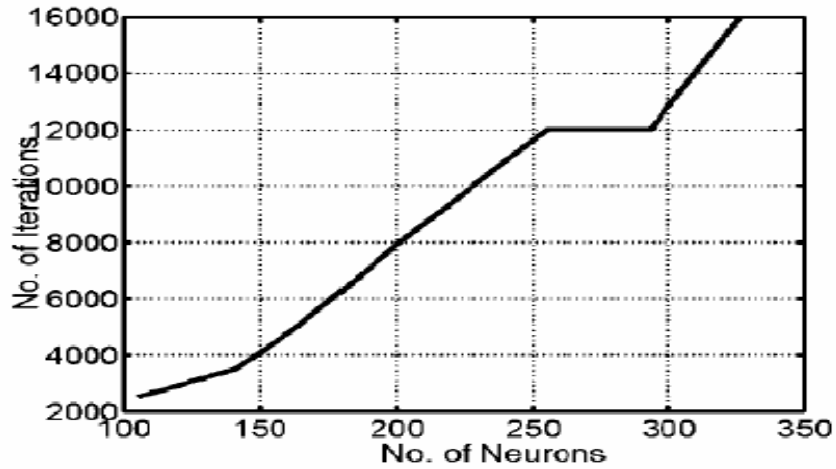
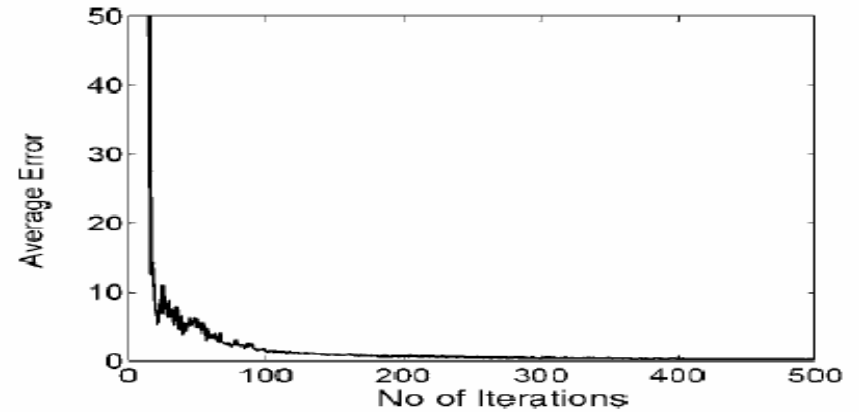
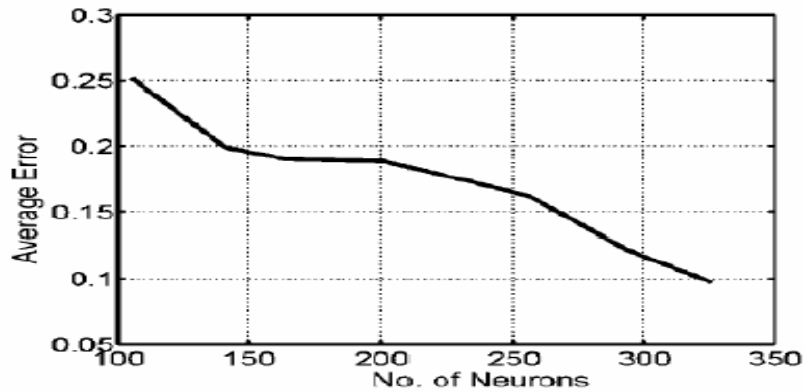
# Comparison QNN with CNN Control



# Quantum Clustering



# Quantum Clustering



# **Проблема распознавания лиц на основе новой технологии интеллектуальных квантовых алгоритмов**

- 1. Увеличение четкости изображения при маскирующих признаках;**
- 2. Повышение надежности распознавания и выделения образа лица при динамических изменениях позиции и помехах;**
- 3. Разработка интеллектуального квантового контроллера сжатия и распознавания изображений**
- 4. Решение проблемы на основе новых квантовых алгоритмов распознавания изображений**
- 5. Три вида последовательного применения квантовых операторов: суперпозиция, квантовая корреляция и интерференция.**
- 6. Перечисленные операторы отсутствуют в классических алгоритмах и реализуются программным способом**
- 7. Программный продукт и интеллектуальный контроллер являются объектами интеллектуальной собственности (ОИС)**
- 8. ОИС защищаются патентами**

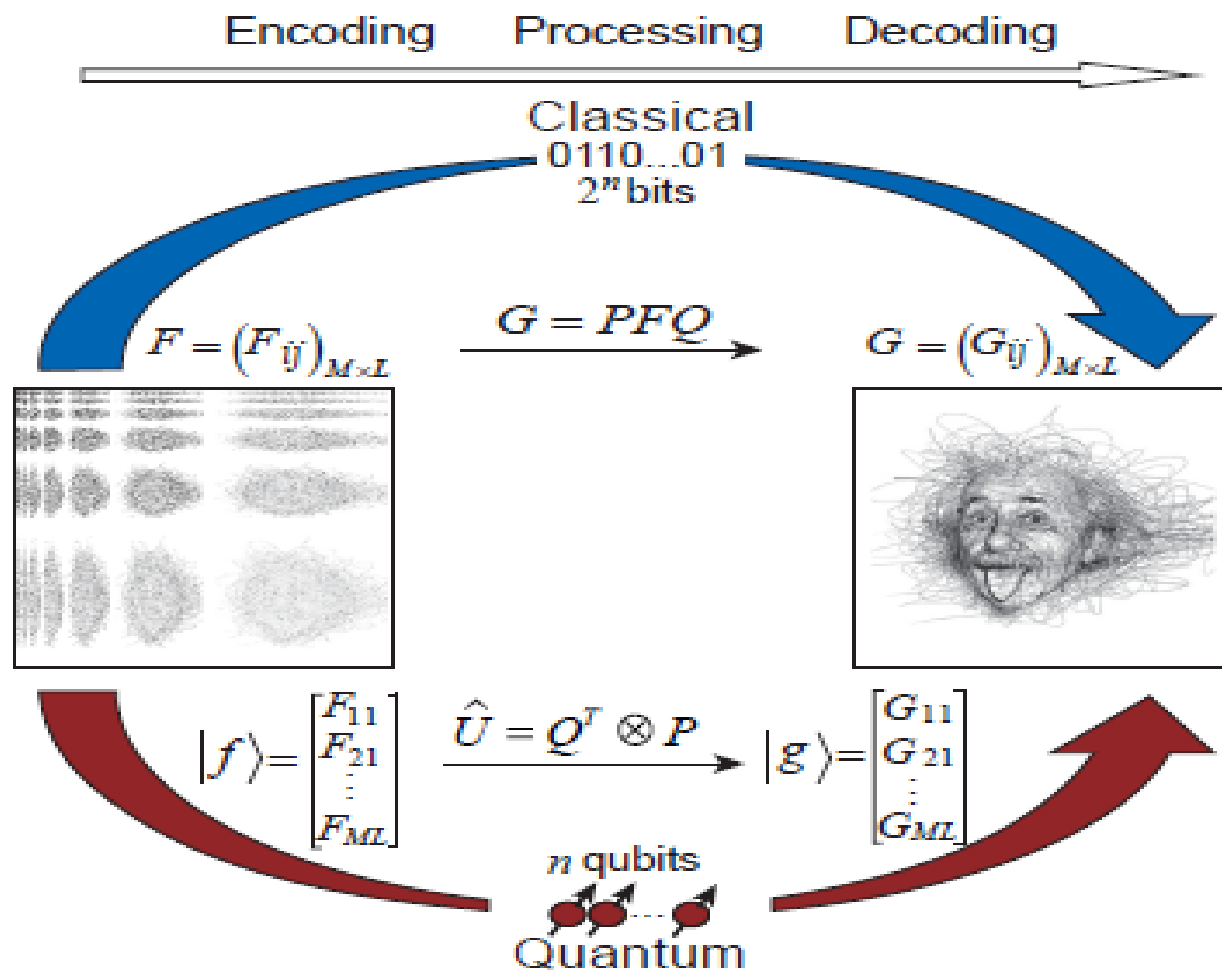


FIG. 1. Comparison of image processing by classical and quantum computers.  $F$  and  $G$  are the input and output images, respectively. On the classical computer, an  $M \times L$  image can be represented as a matrix and encoded with at least  $2^n$  bits [ $n = \lceil \log_2(ML) \rceil$ ]. The classical image transformation is conducted by matrix computation. In contrast, the same image can be represented as a quantum state and encoded in  $n$  qubits. The quantum image transformation is performed by unitary evolution  $\hat{U}$  under a suitable Hamiltonian.



## КВАНТОВОЕ РАСПОЗНАВАНИЕ ЛИЦ И КВАНТОВАЯ ВИЗУАЛЬНАЯ КРИПТОГРАФИЯ: МОДЕЛИ И АЛГОРИТМЫ

Ульянов Сергей Викторович<sup>1</sup>, Петров Сергей Павлович<sup>2</sup>

Также, в зависимости от дальнейших преобразований, может возникнуть необходимость создания суперпозиции пикселей входного изображения. Суперпозиция создается в несколько шагов:

– Кодирование цветов пикселей (представленных в виде вещественных чисел) в комплексные амплитуды квантовых состояний:

$$\delta: \mathcal{R}^3 \rightarrow \mathbb{C}_1^2, \quad (x_1, x_2, x_3) \mapsto (r_1 e^{i\phi_1}, r_2 e^{i\phi_2}),$$

где  $x_1, x_2, x_3$  – компоненты цветовой модели *RGB* (*red, green, blue*),

$$r_1 := \sqrt{1 - x_3^2}, \quad r_2 := x_3, \quad \phi_1 := \arcsin(2x_1 - 1), \quad \phi_2 := \arcsin(2x_2 - 1).$$

Пусть  $z_1 = r_1 e^{i\phi_1}, z_2 = r_2 e^{i\phi_2}$ , тогда имеем цвет пикселя в виде:

$$|q_i\rangle = z_1 |0\rangle + z_2 |1\rangle.$$

Обратное преобразование выполняется по следующей схеме:

$$\gamma: \mathbb{C}_1^2 \rightarrow \mathcal{R}^3, \quad (z_1, z_2) \mapsto \left( \frac{1 + \sin \phi_1}{2}, \frac{1 + \sin \phi_2}{2}, |z_2| \right),$$

где  $\phi_1 := \arg(z_1), \phi_2 := \arg(z_2)$ .

– Кодирование координат пикселей осуществляется следующим образом:

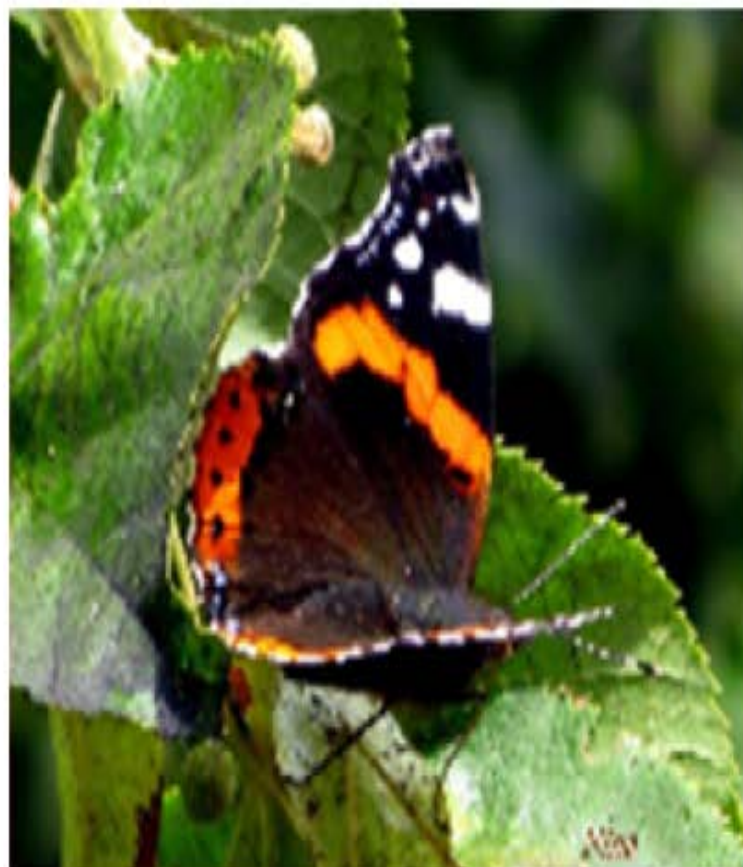
$$|k\rangle = |x\rangle |y\rangle = |x_{n-1} x_{n-2} \dots x_0\rangle |y_{n-1} y_{n-2} \dots y_0\rangle, \quad x_i, y_i \in \{0, 1\},$$

где состояния  $|x\rangle$  и  $|y\rangle$  кодируют координаты пикселей (номера столбца и строки пикселя соответственно).

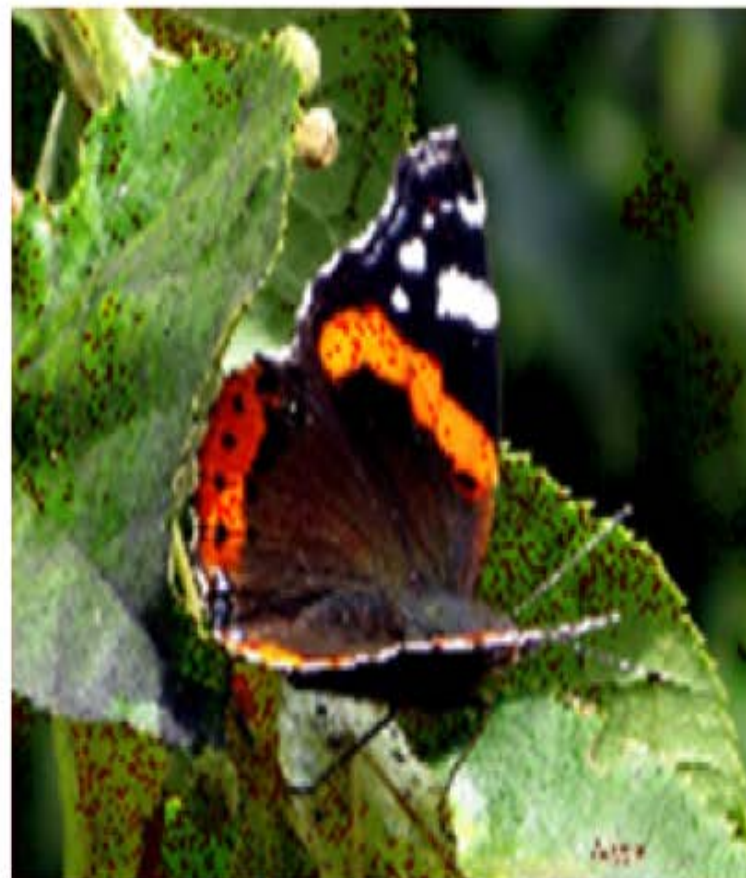
В итоге получаем суперпозицию квантовых состояний пикселей входного изображения в виде:

$$|I\rangle = \frac{1}{2^n} \sum_{k=0}^{2^{2n}-1} |q_k\rangle \otimes |k\rangle.$$

Source 800\*1000 image



Decoded quantum 800\*1000 image

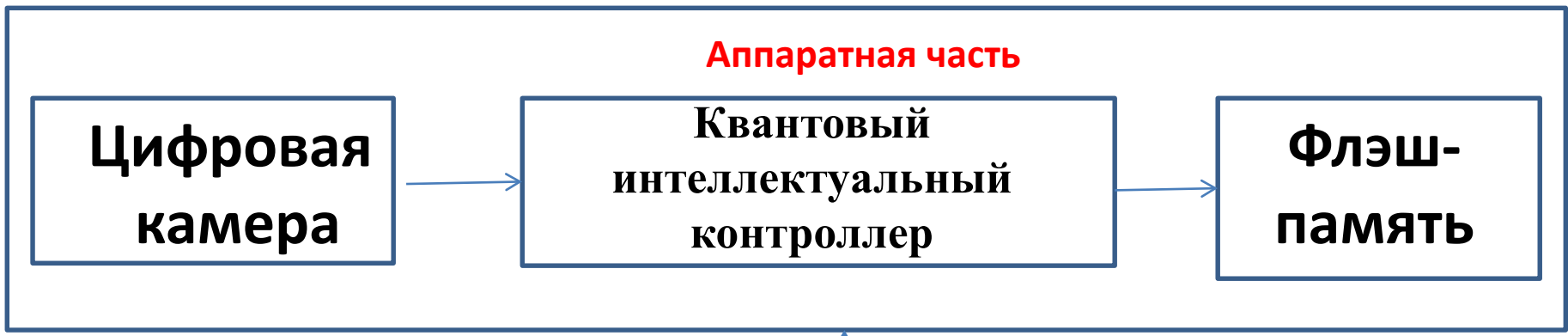


*Рис. 3. Декодирование исходного изображения из квантовой суперпозиции.*

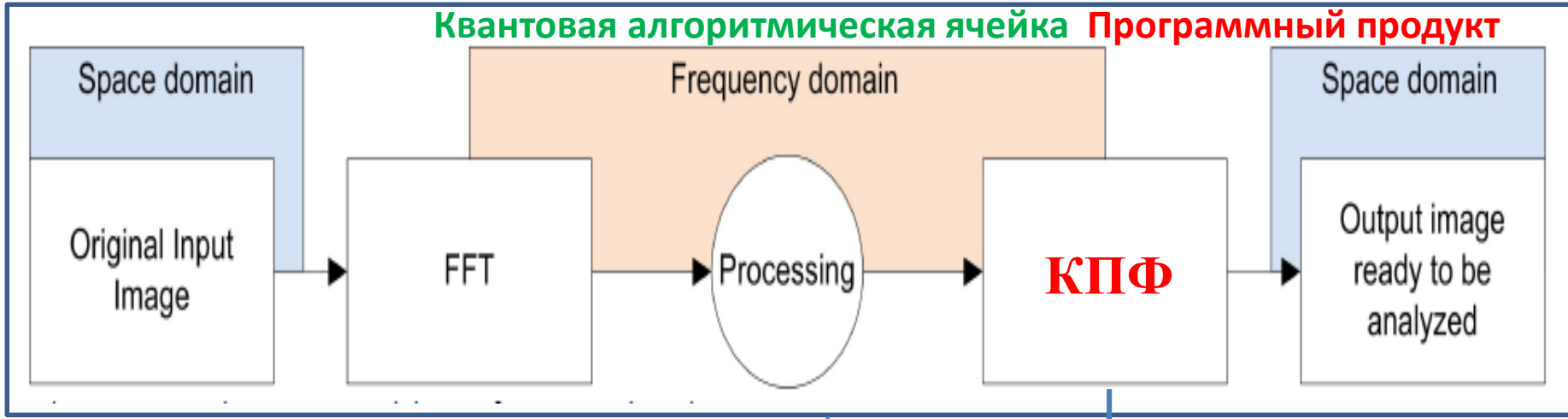
*Количество неправильно декодированных пикселей: 64973 из 800000 (~8.1%)*

# Квантовый интеллектуальный контроллер

## Аппаратная часть



## Квантовая алгоритмическая ячейка Программный продукт



## Квантовый алгоритм распознавания лиц

## Алгоритмическое обеспечение



*Рис. 8. Слева – карта краев, полученная квантовым алгоритмом, справа – результат работы алгоритма по обнаружения лиц*

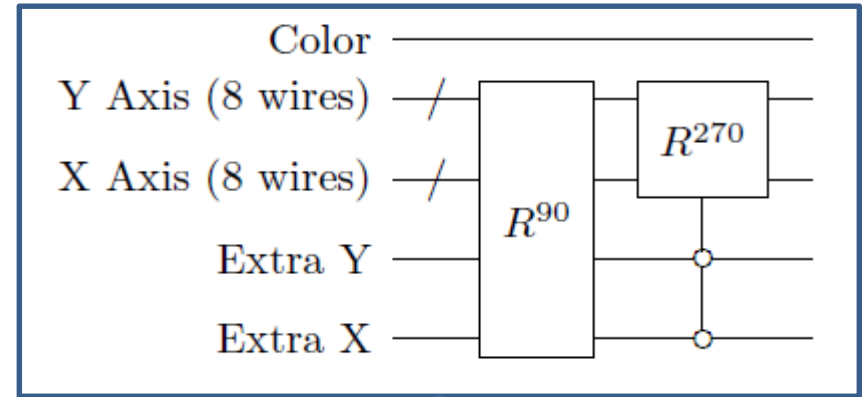


*Рис. 9. Карты краев полученные квантовым алгоритмом выделения границ*

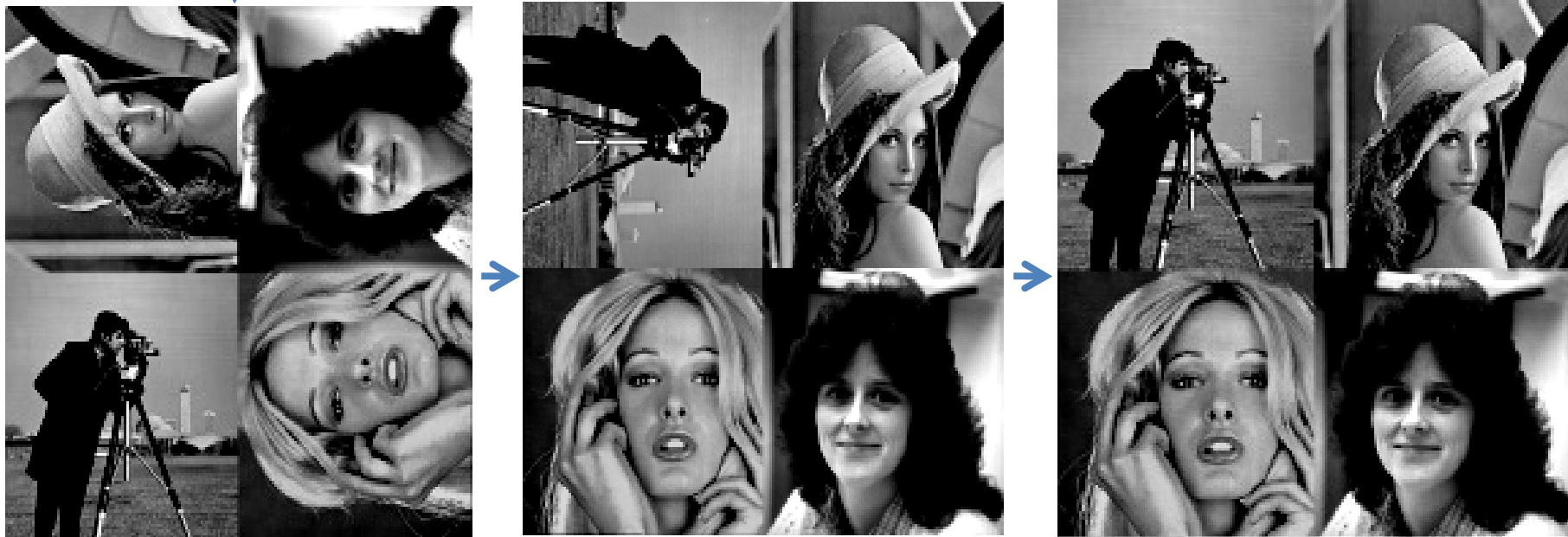
# Применение квантовой суперпозиции

Table 3: Running time (seconds) of geometric transformations

No. of qubits	Flip	Swap	Rotation	Others
14	1.10	1.95	2.46	6.32
16	1.51	5.56	6.80	15.40
18	5.11	21.47	25.56	56.60
20	20.57	93.61	116.30	243.51
22	90.11	428.52	517.09	1116.9



Квантовая ячейка



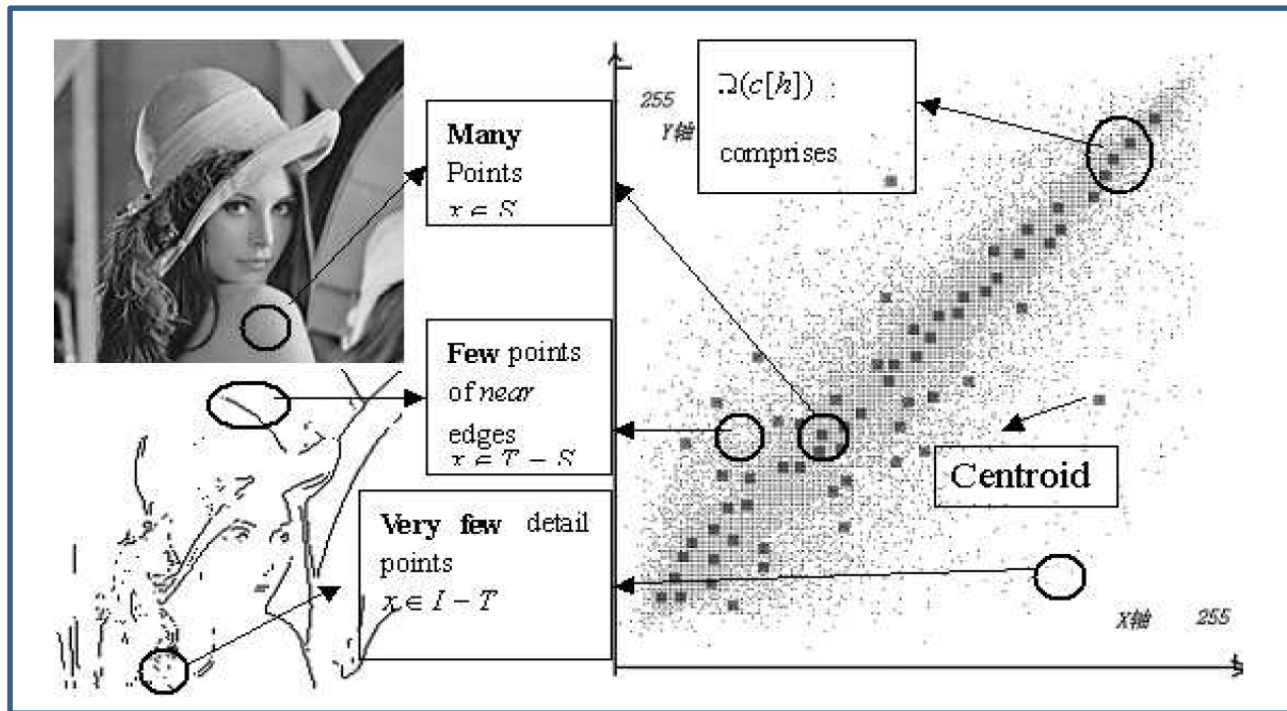
Геометрические преобразования изображения в группе лиц



*Рис. 10. Результат работы алгоритма по обнаружения лиц*

# **Quantum algorithm of vision image compression**

The illustration of the properties of image that is applied to design the hybrid algorithm:



Quantum algorithm application

Notation 1  $x$ : input vector

Notation 2  $\delta_0 = \min\{d(c[i], c[j]) \mid i \neq j, 0 \leq i, j < N\}$

Notation 3  $I$ : whole encoding space

Notation 4  $S = \{x \mid d(x, c[i_0]) < \frac{\delta_0}{2}\}$  and  $T = \{x \mid d(x, c[i_0]) < \hat{\delta}\}$ , where  $\hat{\delta} \geq \frac{\delta_0}{2}$ .

Notation 5  $\Omega(c[i]) = \{c[j] \mid d(c[i], c[j]) < 2\hat{\delta}, c[j] \in C\}$

Notation 6  $Inf_{\Omega} = \min\{|\Omega[c[i]]| \mid 0 \leq i < N\}$

(Image Lena with size  $256 \times 256$  is divided into  $2 \times 1$  image blocks to form vectors [1].)

1. Almost feature vectors  $x$  concentrate on themselves centroids generally. And the solution is unique for  $x \in S$
2. The set  $T-S$  comprises the points of near edges of image and it includes few points in practice. And the solution is included in a small set  $\Omega(c[h])$  for  $x \in T-S$ .
3. The set  $I-T$  comprises special points such as very detail points or points at edges maybe.
4. Statistically,  $|S| \gg |T-S| \gg |I-T|$

The above four statistical properties of image for VQ are applied to accelerate the quantum algorithm:

1. Sub-procedure 1 acts on the set  $S$  with time complexity  $\left[\frac{4}{3}\sqrt{N}\right]$  approximately.
2. Sub-procedure 2 acts on the set  $T-S$  with time complexity less than  $\sqrt{N}$  approximately.
3. Classical full search algorithm acts on  $I-T$ .

The phenomenon should be noticed, that these properties are not powerful for classical methods to solve the open problem of the fast encoding of VQ, by contrast, it is powerful for quantum methods.



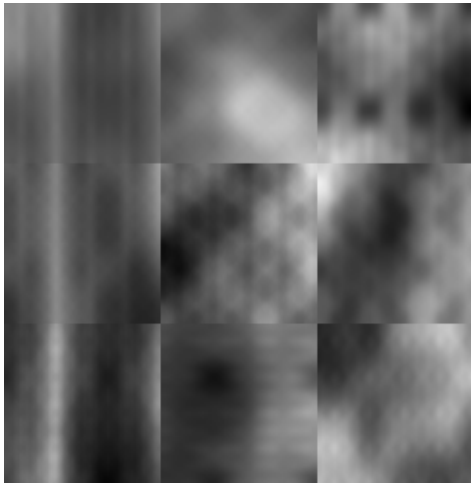
## Efficient description

### QPEG

- Read image by blocks
- Fourier transform
- RG address and fill  $\{\Gamma^{(a)}\}$
- Set compression level:  $\chi$  ←
- Find optimal
- gzip (lossless, entropic compression)
- (define discretize  $\Gamma$ 's to improve gzip)
- diagonal organize the frequencies and use 1d RG
- work with differences to a prefixed table



Max  $\chi = 81$



$\chi = 1$   
PSNR=17



$\chi = 4$   
PSNR=25



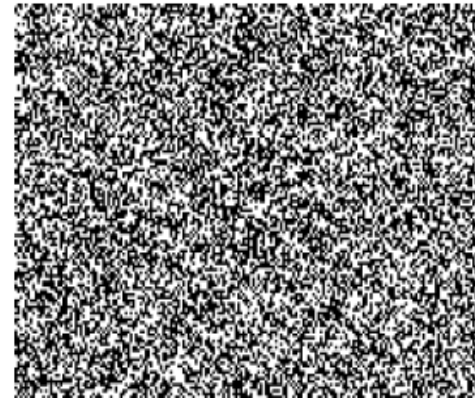
$\chi = 8$   
PSNR=31

# Visual Quantum Cryptography

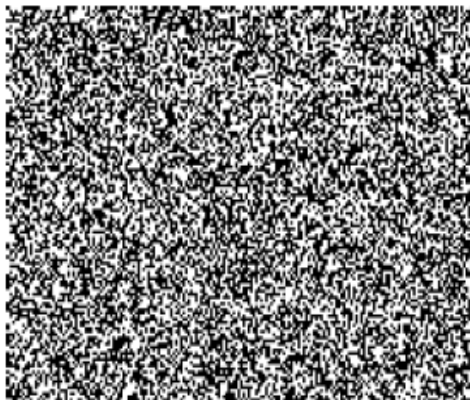
Original binary image



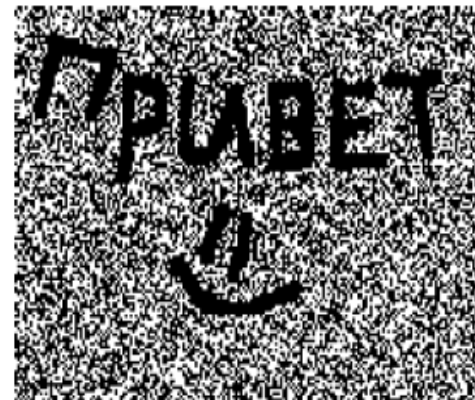
Shadow image S1



Shadow image S2



Superimposed image



*Рис. 7. Результат работы квантового криптографического алгоритма: надпись исходного изображения проявляется только при наложении двух «теневого» изображений*

# Коммерческая привлекательность разработки

## Область применения

- Интеллектуальная робототехника с цифровыми видеокамерами
- Системы дистанционного наблюдения и контроля в условиях помех

## Преимущества новой разработки

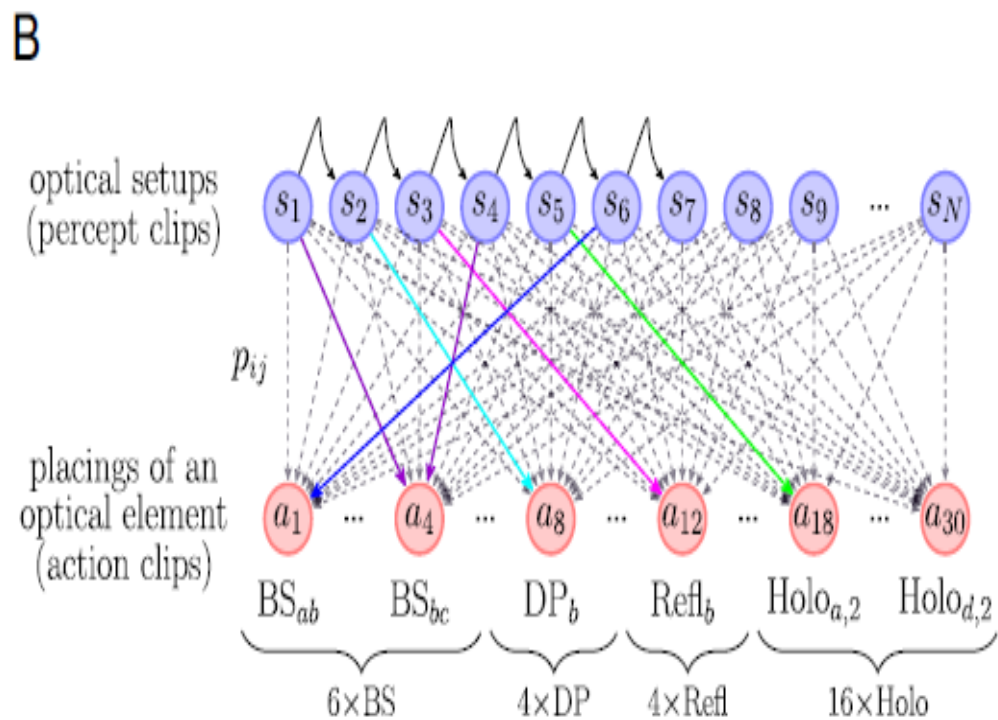
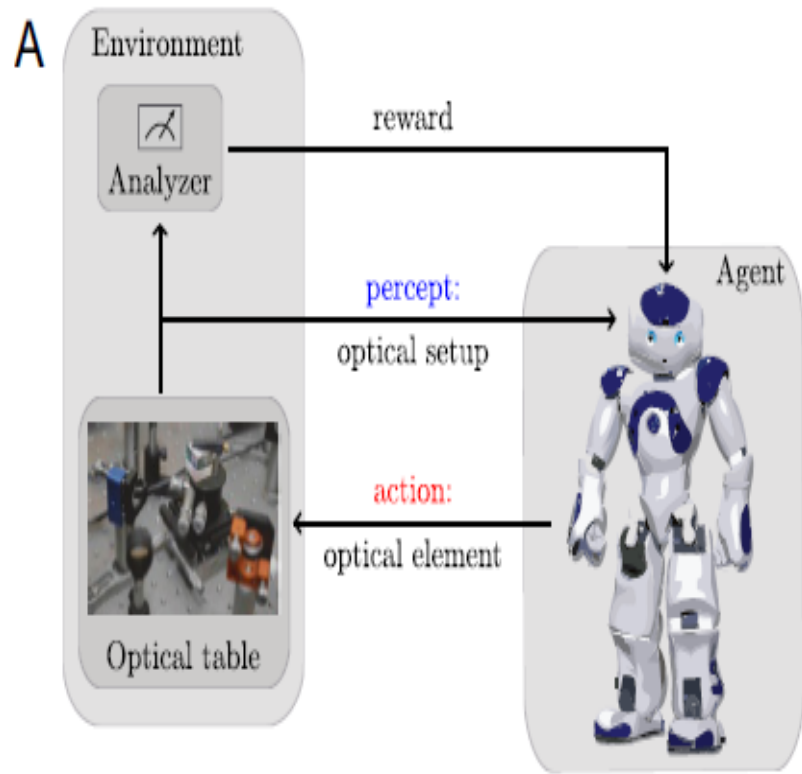
- Проектирование квантового алгоритма в виде компактной перепрограммируемой квантовой алгоритмической ячейки дает возможность разработки и реализации интеллектуального многофункционального контроллера обработки, сжатия и распознавания изображений в условиях помех.
- Конкурентно способность на рынке интеллектуальных продуктов;

## Коммерческая привлекательность разработанных продуктов

- Тиражирование разработанной структуры системы интеллектуального контроллера;
- Тиражирование разработанной аппаратной реализации интеллектуального контроллера;
- Тиражирование разработанной технологии проектирования квантового робастного интеллектуального контроллера распознавания лиц ;
- Программный продукт поддержки интеллектуальной системы управления контроллером;
- Патенты, товарный знак и лицензии на разработанные продукты.

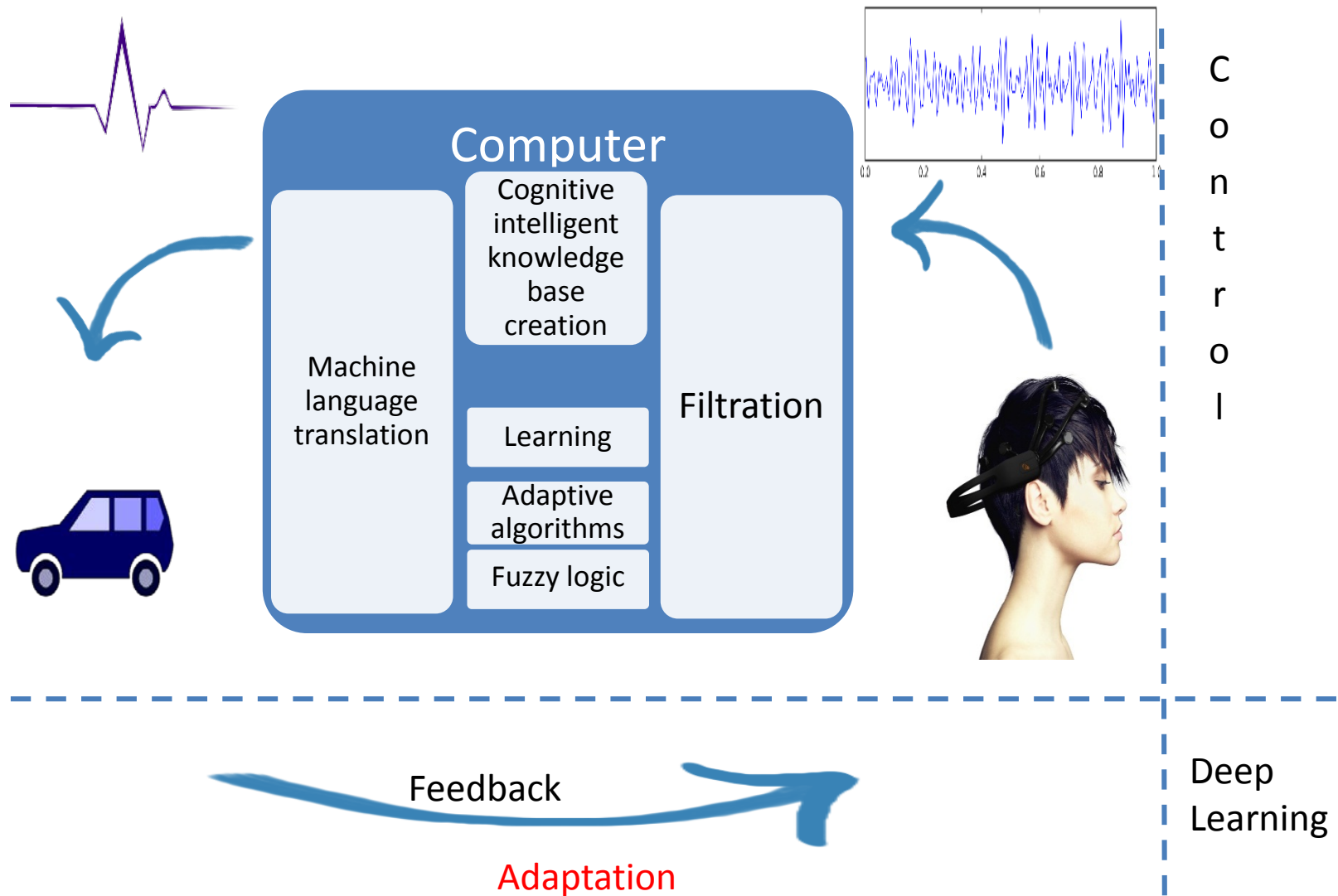
# Deep Learning Applications

1. Robotic New Experiment Design
2. Cognitive intelligent control with QNN based on EEG



**Fig. 1.** The learning agent. **(A)** An agent is always situated in an environment (9). Through sensors it perceives optical setups and with actuators it can place optical elements in an experiment. Note that, in this paper, the interaction between the agent and the environment was entirely simulated on a classical computer. (One could imagine that in the future, a real robot builds up the experiment designed by the computer.) On the side, an analyzer evaluates a proposed experiment corresponding to the current optical setup and gives rewards according to a specified task. Image of the optical table courtesy of Manuel Erhard (University of Vienna, Vienna) and is a part of the (3, 3, 3) experiment. Image of the agent reproduced from <https://openclipart.org/detail/266420/Request>. **(B)** The memory network that represents the internal structure of the PS agent. Dashed arrows indicate possible transitions from percept clips (blue circles) to action clips (red circles). Solid, colored arrows depict a scenario where a sequence of actions leads to the experiment  $\{BS_{bc}, DP_b, Refl_b, BS_{bc}, Refl_b, Holo_{a,2}\}$ . Arrows between percepts correspond to deterministic transitions from one experiment to another after placement of an optical element.

# “Brain – Cognitive regulator – Device” intelligent interface



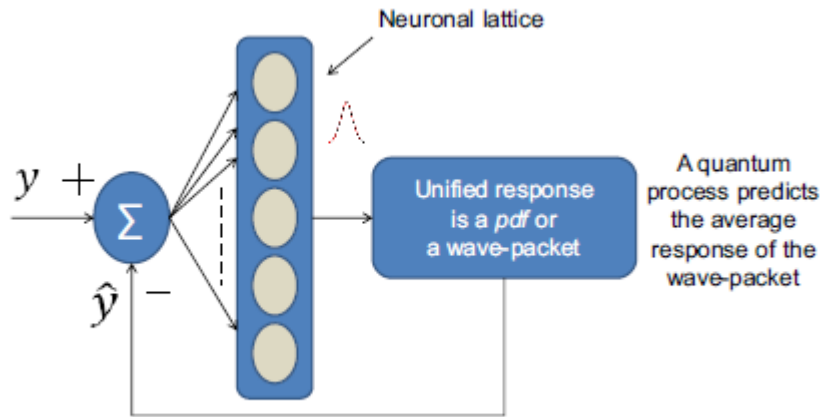


Fig. 1. Conceptual framework of RQNN model.

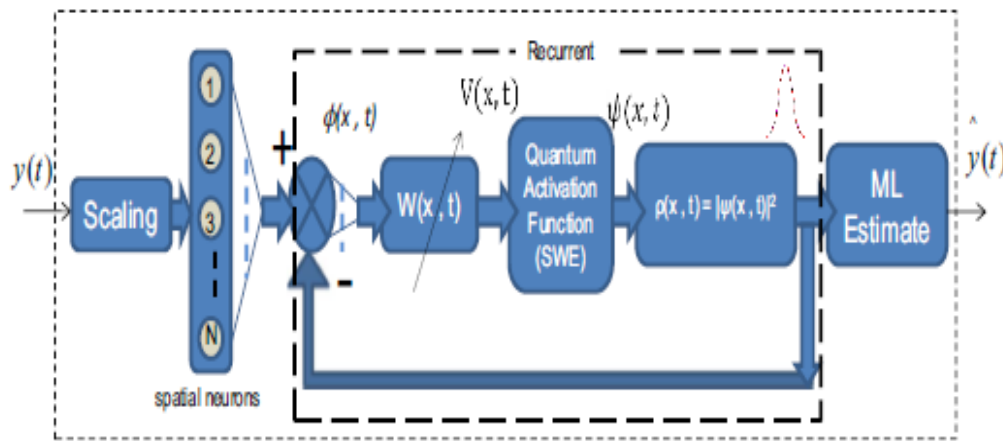


Fig. 2. Signal estimation using RQNN model.

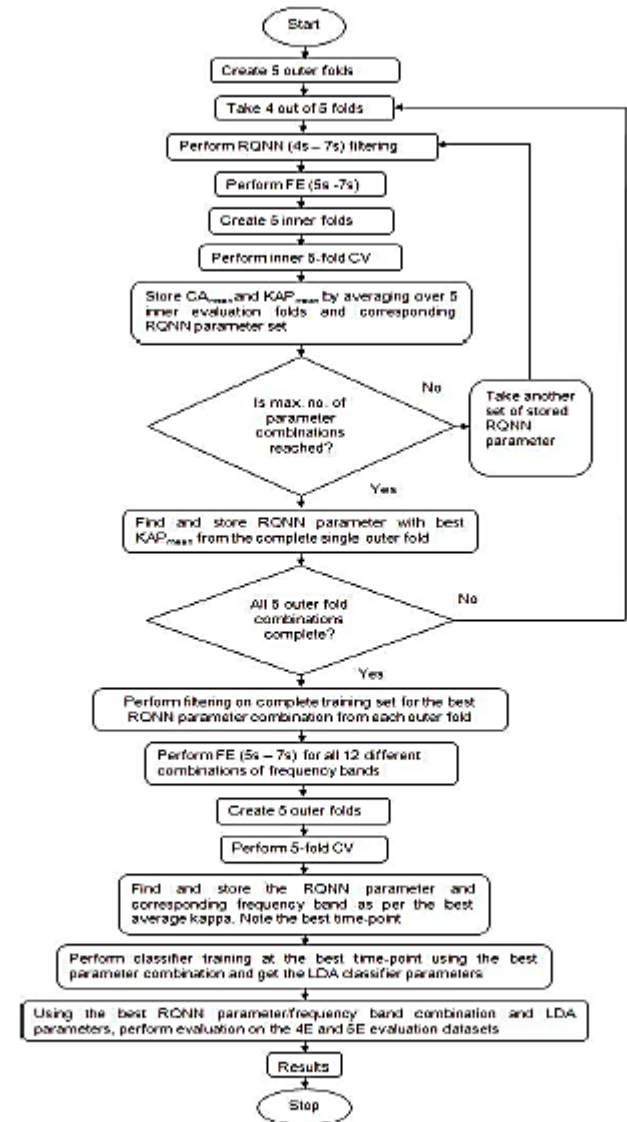


Fig. 5. Flowchart for the two-step inner-outer fivefold CV parameter selection (RQNN/frequency band).

## Quantum Neural Network-Based EEG Filtering for a Brain-Computer Interface

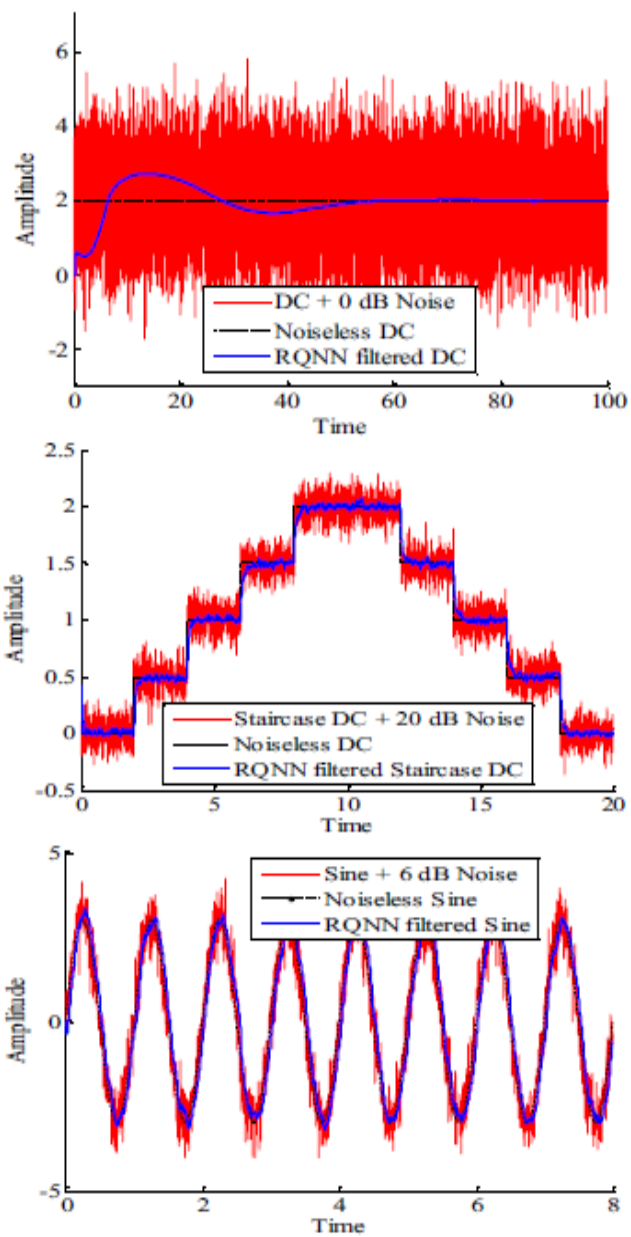


Fig. 6. DC, staircase dc, and sine signal filtering.

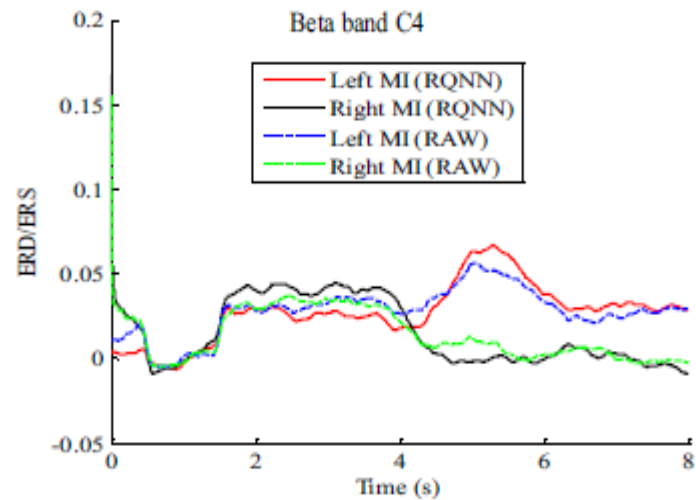


Fig. 9. ERS for RQNN-filtered and raw EEG (subject B0405E).

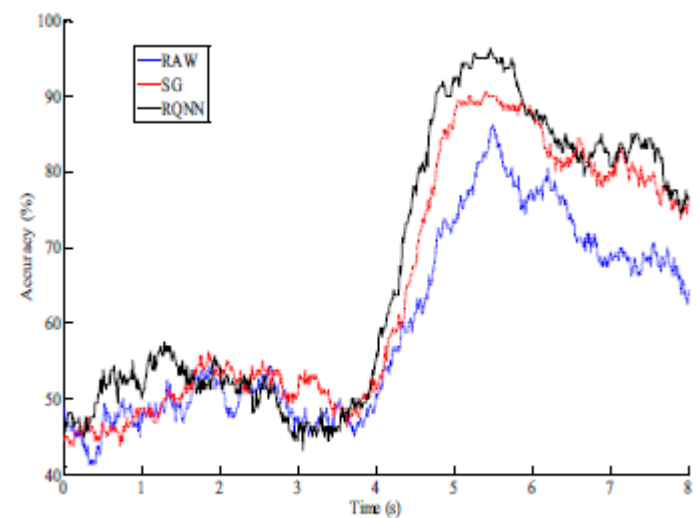


Fig. 10. Classification accuracy plot (subject B0405E).



# Quantum Genetic Algorithm

# Quantum massive parallel computing

Quantum KB optimizer

SCO

Answer

QC output

QAG design

Classical input

Problem

$$|\psi_{fin}\rangle = [(\text{Interference})(\text{Quantum oracle})](\text{Superposition})|\psi_{initial}\rangle$$

Qualitative properties of function

Quantum Fourier transformation

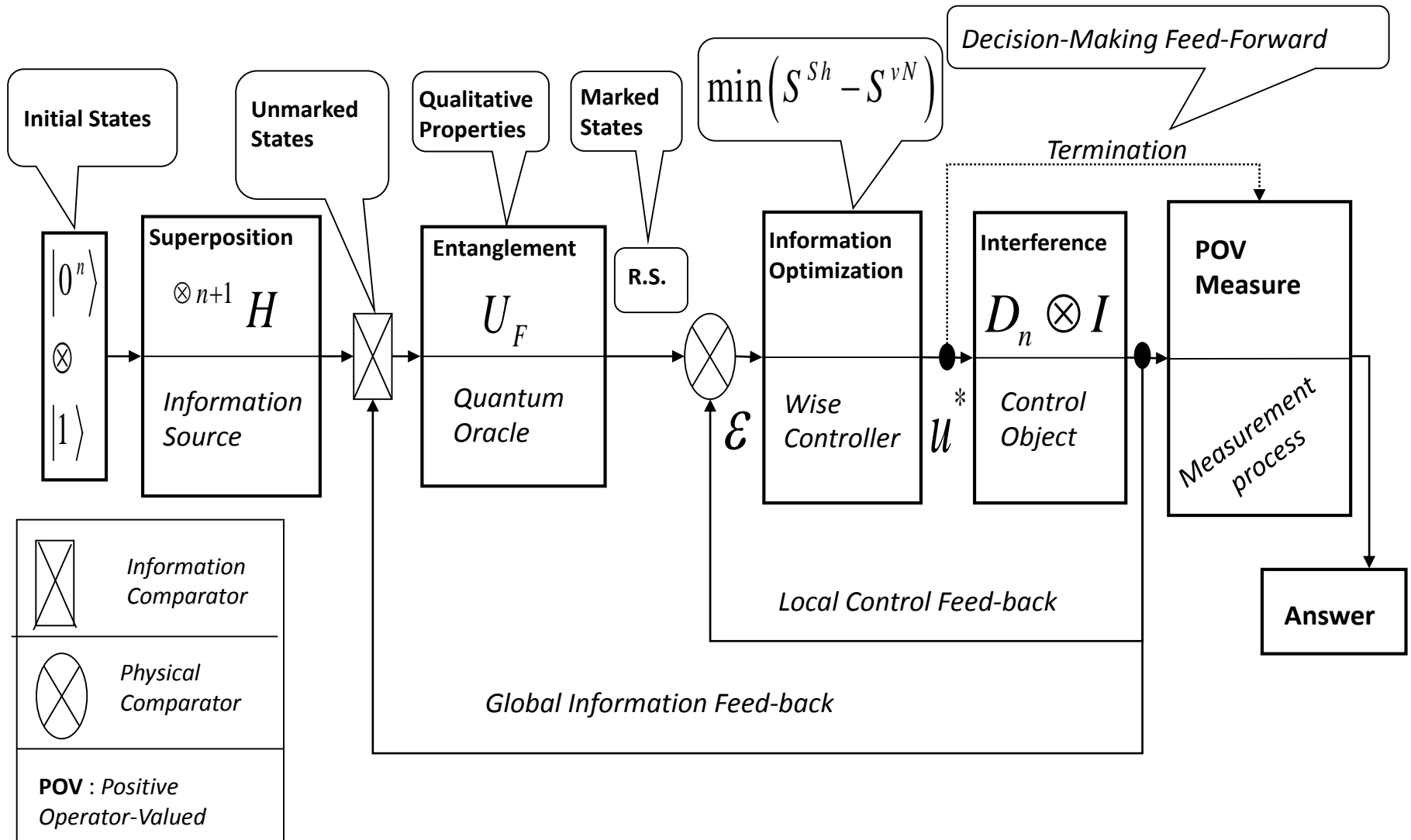
Problem oriented operator

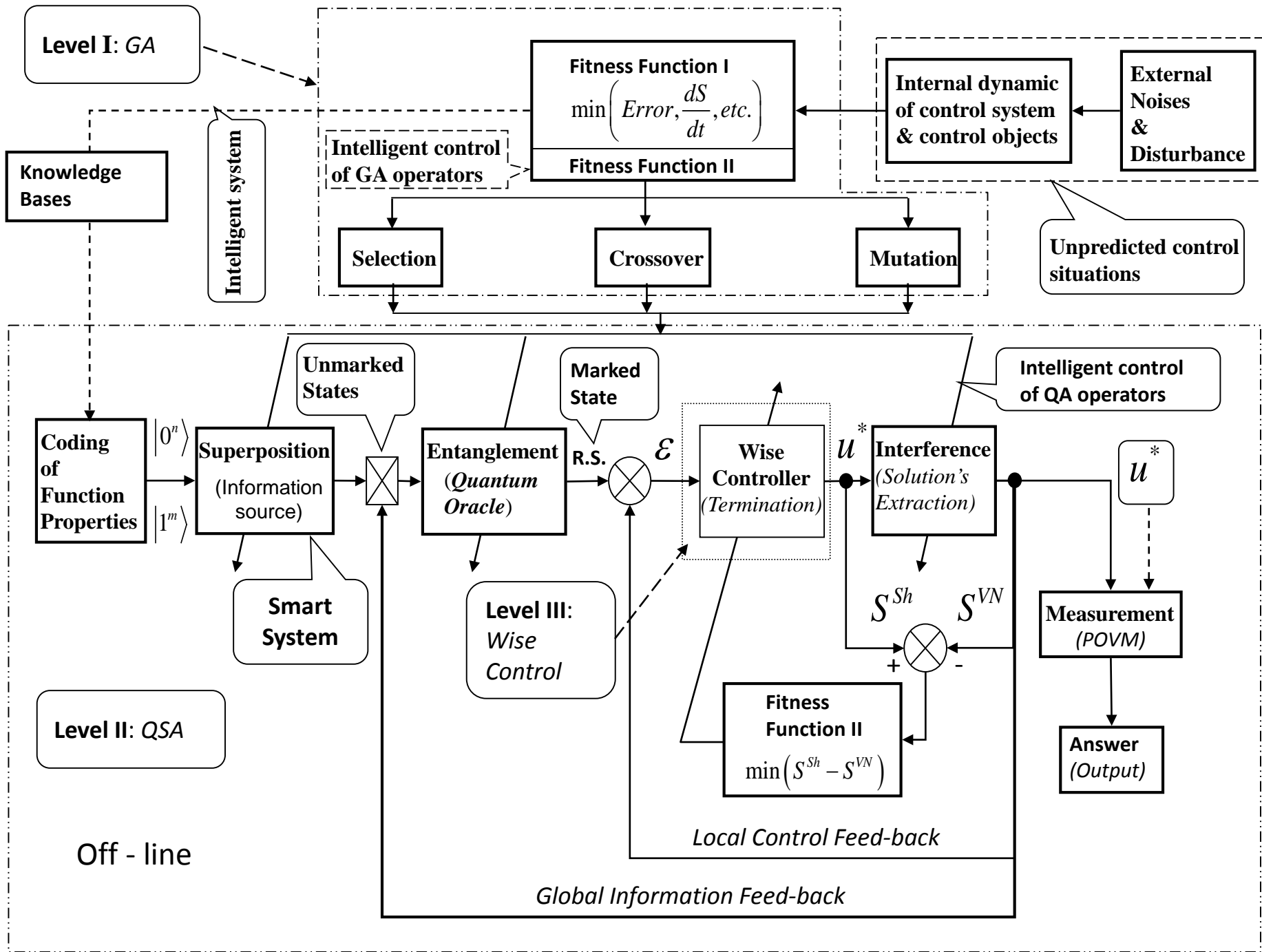
Hadamard transformation

Coding of function properties

Qualitative properties of function

Quantum oracle as black box





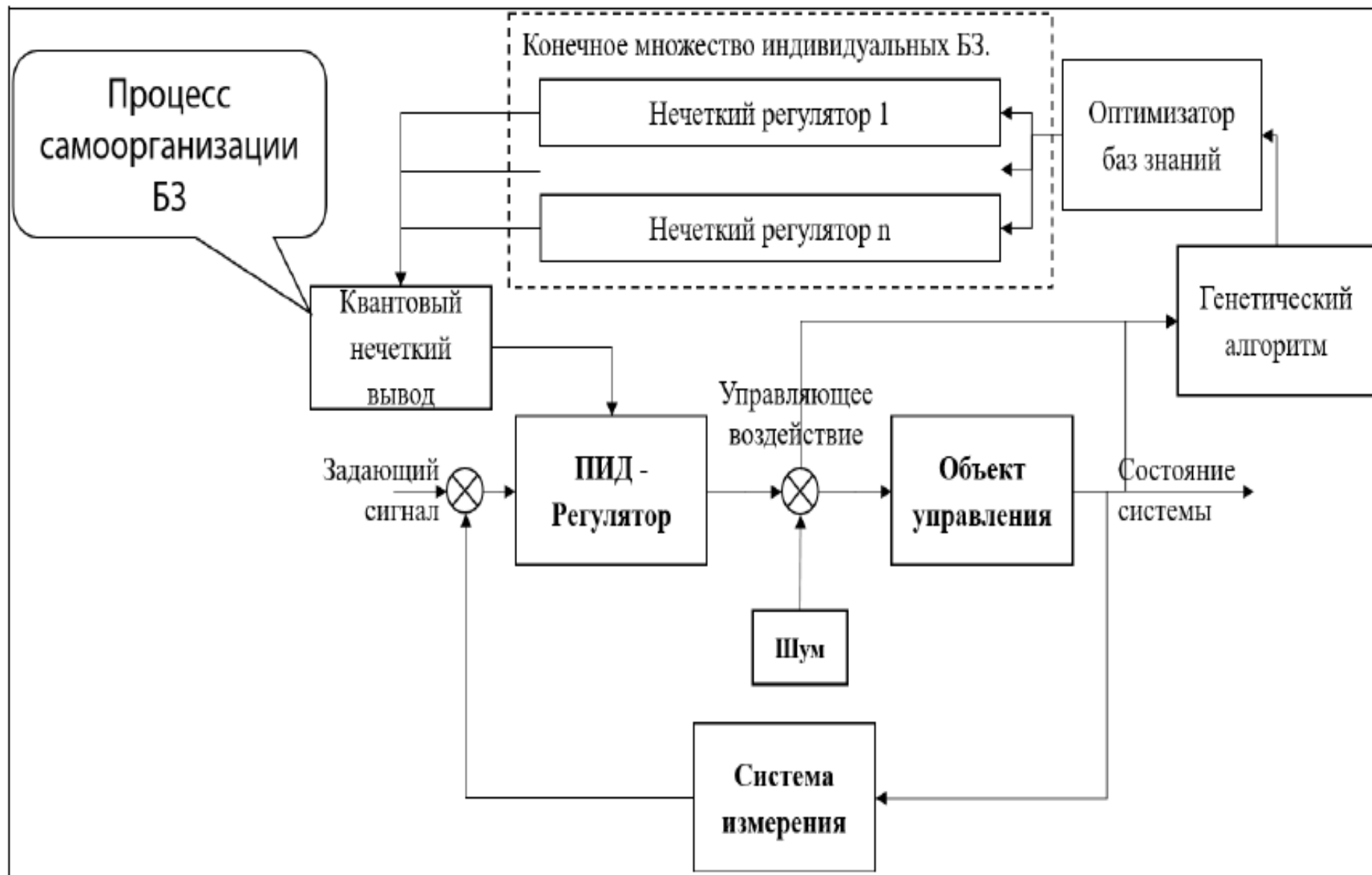


Рис. 1. Интеллектуальная система управления с квантовым нечетким выводом

Формально структуру КГПА описывается следующим множеством логических операторов:

$$QGSA = \left\{ C, Ev, P^0, L, [\Omega, \chi, \mu]_{GA\text{-операторы}}, [Sup, Ent, Int]_{QA\text{-операторы}}, \Delta \right\},$$

где  $C$  – алфавит для генетического кодирования индивидуума для конкретной задачи;  $Ev$  – функция пригодности;  $P^0$  – начальная популяция;  $L$  – размер популяции;  $\Omega$  – оператор отбора (селекции);  $\chi$  – оператор скрещивания;  $\mu$  – оператор мутации;  $Sup$  – квантовый оператор линейной суперпозиции;  $Ent$  – квантовый оператор запутывания (смешанное состояние);  $Inf$  – оператор вывода;  $\Delta$  – условие остановки, включающие такие критерии остановки, как оптимум заданной функции пригодности и минимум энтропии Шеннона/фон Неймана. Структура на рис. 2 – базовая модель

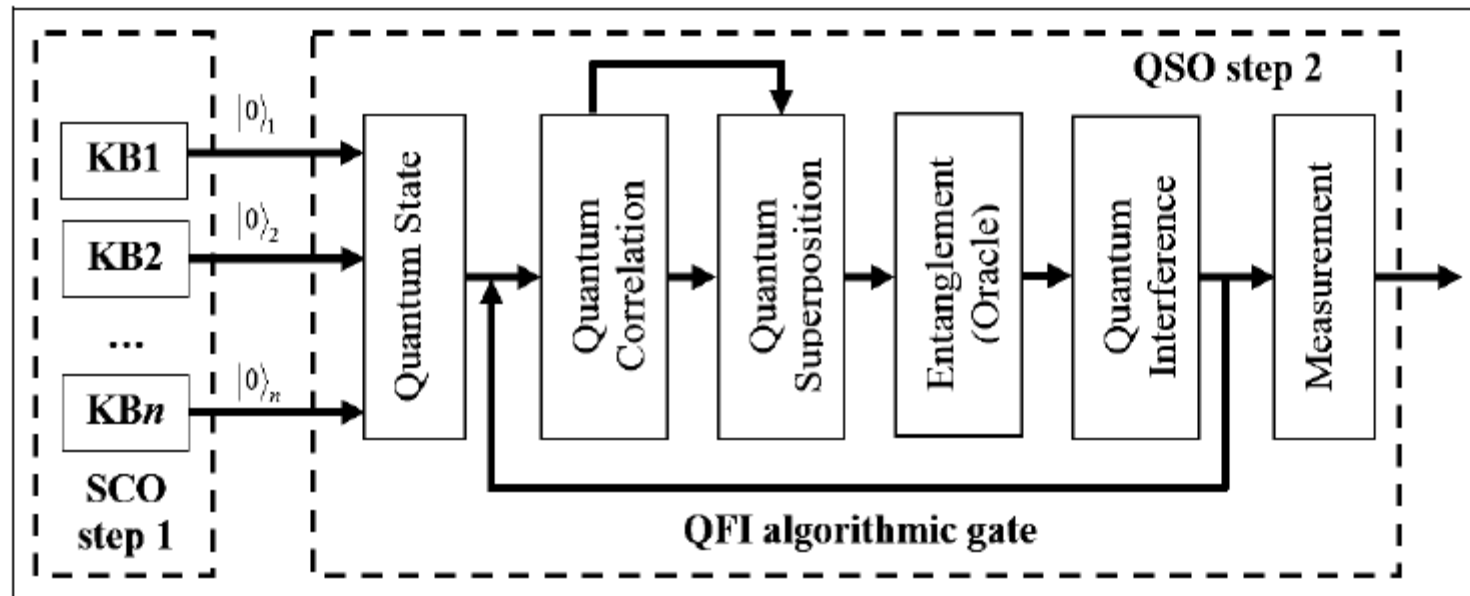
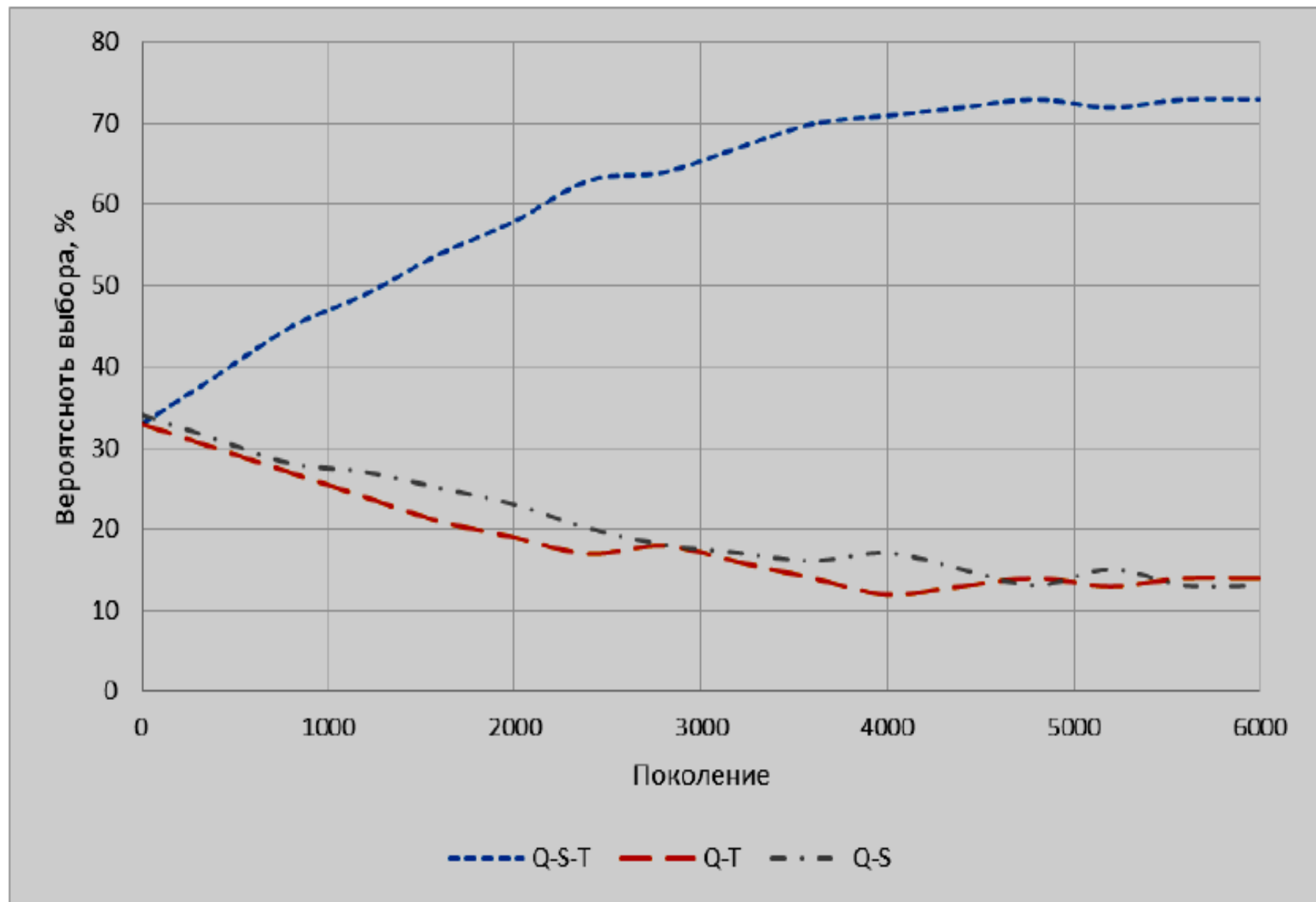


Рис. 3. Квантовая алгоритмическая ячейка (QAG) квантового нечеткого вывода



*Рис. 17. Результат выполнения квантового генетического алгоритма*

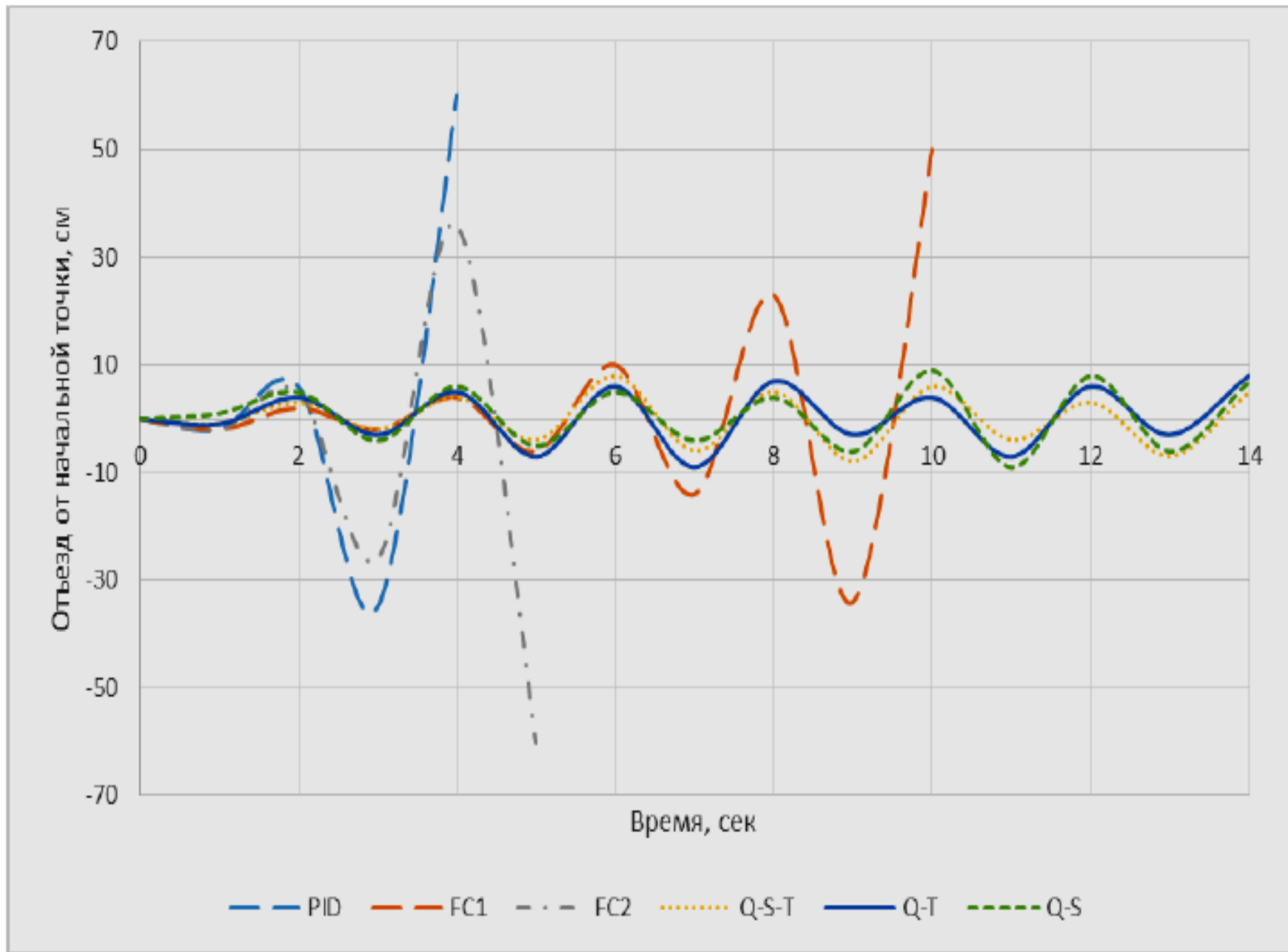


Рис. 19. Сравнение поведения различных алгоритмов



# Video Demo

1978 г.



1979 г.



1982 г.



2015 г.





Учебно-  
методические  
пособия  
Учебно-научного  
центра ОИЯИ  
Дубна

УНЦ-2013-54

С. В. Ульянов, Г. П. Решетников

**ТЕХНОЛОГИИ ИНТЕЛЛЕКТУАЛЬНЫХ  
ВЫЧИСЛЕНИЙ**

Мягкие и дробные вычисления  
в интеллектуальном управлении

*Учебно-методическое пособие*

2013



Учебно-  
методические  
пособия  
Учебно-научного  
центра ОИЯИ  
Дубна

УНЦ-2015-56

С. В. Ульянов, А. Г. Решетников, Г. П. Решетников

**ТЕХНОЛОГИИ ИНТЕЛЛЕКТУАЛЬНЫХ  
ВЫЧИСЛЕНИЙ**

Квантовые вычисления и программирование  
в самоорганизующихся интеллектуальных  
системах управления

*Учебно-методическое пособие*

2015

# Самоорганизация интеллектуального управления: Объект интеллектуальной собственности

Основа ВНИТ проектирования робастных ИМСУ



US06619441

(12) **United States Patent**  
Ulyanov

(10) Patent No.: **US 6,411,944 B1**  
(11) Date of Patent: **Jun. 25, 2002**

(54) **SELF-ORGANIZING CONTROL SYSTEM**

(57) Inventor: **Sergei V. Ulyanov, Oren 103**

(73) Assignee: **Yanaka Hasekichi Kabushiki Kaisha, Matsuda JP**

(71) Name: *Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.*

(51) Int. Cl. **H03H 1/00**

(52) Filed: **Mar. 2, 1998**

(56) Foreign Application Priority Data

Mar. 2, 1997 JP 98006

(51) Int. Cl. **G06F 0008 (2006 110)**  
(52) U.S. Cl. **70612, 7061, 70615**  
(56) Field of Search: **70612, 7061**

(58) References Cited

**U.S. PATENT DOCUMENTS**

5,793,814 A \* 1/1998 Saito et al. 7067  
5,822,289 A \* 1/1999 Saito et al. 7067  
5,722,074 A \* 1/1998 Hsu et al. 7062  
6,076,076 A \* 1/2000 Hsu et al. 7062

**OTHER PUBLICATIONS**

"Design of Expert System Based Fuzzy Logic Control", *Chi-Min Tsai and Tzu-Hong S. Li*, *Industrial Control System Laboratory, Department of Electrical Engineering, National Chung Hsing University, Taiwan, Republic of China*, 1996.

Yanaka H., et al., 1994, "Expert System, Fuzzy Logic, and Neural Network Applications in Process Control and Motion Control", *Proceeding of the IEEE*, vol. 82, No. 8, pp. 1303-1322.

Feng Q. and Yenney B., 1999, "Design and verification of control system of an inverted pendulum", *Kobunshi*, vol. 6, No. 4, pp. 295-31.

Goshima K. and Ulyanov S.V., 1993, "Mobile system with self-organizing control", *Int. J. of Computer and Systems Sciences*, vol. 3, No. 1, pp. 129-42.

Murphy R., Haganee M. and Dawson TA., 1993, "Robust controller synthesis for control of general adaptation and stability", *IEEE Transactions on Electrical Engineering*, vol. 41, No. 2, pp. 202-202.

Li M.-S., Yi S.-G., Yoo Y.-G., and Ohn Y.-G., 1995, "Fuzzy control of electrohydraulic servo-actuator position", *IEEE International Journal*, vol. 36, No. 1, pp. 78-83.

Lee YG, Kim TR, and Suh JH., 1994, "A self-organizing model self-organizing fuzzy logic controller", *Mathematics*, vol. 4, No. 1, pp. 71-98.

Lee TS and Wu J.-T., 1995, "Adaptive fuzzy-logic control system using fuzzy control", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 25-5001, No. 1, pp. 267-276.

Murphy R.M., 1993, "Fuzzy Logic Systems for Engineering Control", *Proceeding of the IEEE*, vol. 81, No. 5, pp. 745-771.

Wajsbort R., Tseloni T., Yin S., and Krupnik L., 1993, "A development of a new evolution of an autonomous control", *Proc. IEEE 35th International Conference on Intelligent Robot and Systems (ICRS '93)*, vol. 2, Canada, Dallas, pp. 505-512.

Perrow M. and Sussner J., 1987, "Threats of dynamic systems", *Electric Power*, vol. 15, No. 4, pp. 1256-1251.

*(Continued on next page)*

**Primary Examiner—George H. Davis**  
**(74) Attorney, Agent or Firm—Kashiki, Matsuda, Ohno & Ito LLP**


**ABSTRACT**

A self-organizing control system suitable for nonlinear control of a physical object is described. The control system calculates the energy production difference between a state difference (the difference of the energy made in controlled object) and a state differentiation (the difference of the energy gain) in the controlled object from a PID controller that controls the object. The energy production difference is used to generate an existing control rule by using the energy production difference as a performance function for a genetic system.

**10 Claims, 18 Drawing Sheets**



Самоорганизация



US06672181B2

(12) **United States Patent**  
Ulyanov

(10) Patent No.: **US 6,721,718 B2**  
(11) Date of Patent: **Apr. 13, 2004**

(54) **SYSTEM FOR INTELLIGENT CONTROL BASED ON SOFT COMPUTING**

(57) Inventor: **Sergei V. Ulyanov, Oren 103**

(73) Assignee: **Yanaka Hasekichi Kabushiki Kaisha, Matsuda JP**

(71) Name: *Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.*

(51) Int. Cl. **H03H 1/00**

(52) Filed: **Jul. 2, 2002**

(56) Prior Publication Data

US 20020090010 A1 May 9, 2002

**Related U.S. Application Data**

(53) Continuation of application No. 09/174,486, filed on Oct. 20, 1998, now Pat. No. 6,411,944.

(51) Int. Cl. **G06F 0008 (2006 110)**  
(52) U.S. Cl. **70612, 7061, 70615**  
(56) Field of Search: **70612, 7061, 70615**

(58) References Cited

**U.S. PATENT DOCUMENTS**

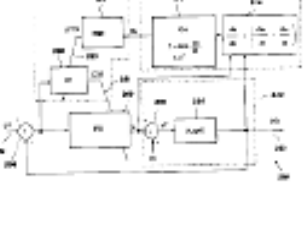
5,111,231 A \* 8/1992 Green et al. 70623  
5,142,877 A \* 9/1992 Shimizu 70619  
5,170,843 A \* 10/1992 Izumi et al. 70623  
5,224,793 A \* 1/1993 Ma 70614  
5,174,756 A \* 9/1993 Ito et al. 70612  
5,288,883 A \* 12/1993 Mizoguchi et al. 70615  
5,287,375 A \* 1/1994 Sugawara et al. 70617  
5,199,290 A \* 1/1994 Matsumoto et al. 70627  
5,224,889 A \* 1/1994 Ogawa 70617  
5,287,512 A \* 1/1994 Saito et al. 70612  
5,199,291 A \* 1/1994 Matsumoto et al. 70627  
5,199,292 A \* 1/1994 Matsumoto et al. 70627  
5,199,293 A \* 1/1994 Matsumoto et al. 70627  
5,199,294 A \* 1/1994 Matsumoto et al. 70627  
5,199,295 A \* 1/1994 Matsumoto et al. 70627  
5,199,296 A \* 1/1994 Matsumoto et al. 70627  
5,199,297 A \* 1/1994 Matsumoto et al. 70627

**Primary Examiner—George H. Davis**  
**(74) Attorney, Agent or Firm—Kashiki, Matsuda, Ohno & Ito LLP**


**ABSTRACT**

A self-organizing control system suitable for control of a continuous or variable plant is described. The self-organizing control system is configured to use a selected sensor set for controlling the plant without significant loss of control gain by factoring in a comparison to an optimal control system with an optimum sensor set. The control system calculates the information entropy provided by the selected sensor set as compared to the information entropy provided by the optimum set. The control system also calculates the difference between the entropy production rate of the plant and the entropy production rate of the controller. A genetic optimizer is used to tune a fuzzy neural network in the self-organizing control system for the genetic optimizer provides optimum control strategy in the self-organizing control system by maximizing the difference in entropy production while maintaining the same information content.

**15 Claims, 16 Drawing Sheets**



Мягкие вычисления



US06678018B1

(12) **United States Patent**  
Ulyanov

(10) Patent No.: **US 6,578,018 B1**  
(11) Date of Patent: **Jun. 10, 2003**

(54) **SYSTEM AND METHOD FOR CONTROL USING QUANTUM SOFT COMPUTING**

(57) Inventor: **Sergei V. Ulyanov, Oren 103**

(73) Assignee: **Yanaka Hasekichi Kabushiki Kaisha, Matsuda JP**

(71) Name: *Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 112 days.*

(51) Int. Cl. **G06F 0008 (2006 110)**  
(52) U.S. Cl. **70614, 70615, 70645**  
(56) Field of Search: **70615, 45, 14**

(58) References Cited

**U.S. PATENT DOCUMENTS**

5,019,242 A \* 10/1998 Matsuda et al. 70616  
5,071,578 A \* 10/1999 Kim  
6,117,766 B1 \* 11/2001 Green 70619

**OTHER PUBLICATIONS**

Michael J. A. Berry et al., *Data Mining Techniques*, 1997, Wiley, 335-346.

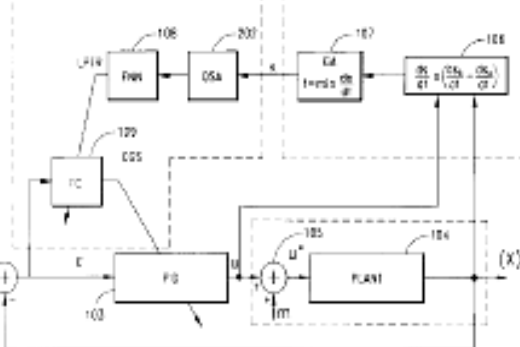
\* cited by examiner

**Primary Examiner—John F. Colburn**  
**Assistant Examiner—Joseph E. Hart**  
**(74) Attorney, Agent or Firm—Kashiki, Matsuda, Ohno & Ito LLP**

**ABSTRACT**

A methodology and an algorithm for programming a quantum logic algorithm is described. In one embodiment, an algorithm for generating a quantum gate is described. The quantum gate describes the evolution of the quantum computing algorithm and is used to implement a classical control algorithm. In one embodiment, the quantum gate is used in a quantum search algorithm to search a number of local solution spaces to find a global solution to be used in a control system to control a plant. In one embodiment, the quantum search algorithm is an iterative algorithm and an energy based basis for stopping the iteration is described. In one embodiment, the quantum search algorithm is used to improve a genetic optimizer in the control system.

**48 Claims, 48 Drawing Sheets**



Квантовые вычисления

**Thank You !**