Dariusz Ruciński[1]
ORCID: 0000-0001-5458-9170

University of Siedlce
Faculty of Exact and Natural Sciences
Institute of Computer Science
ul. 3 Maja 54, 08-110 Siedlce, Poland

[1] `dariusz.rucinski@uws.edu.pl`

Selected applications of models based on quantum computing

DOI: 10.34739/si.2024.30.05

Abstract. The work discusses mathematical models using paradigms developed for quantum phenomena, including the vector-matrix Hilbert space and quantum gates for linear transformations of states to build a day-ahead market model (DAM). The article presents the most popular directions in the development of models using tools used in the description of quantum phenomena to present the methodology of conducting this type of calculation. Mathematical structures such as Hilbert space and operations on this space were used to build a quantum-inspired Artificial Neural Network. The work presents the proposition of implementation of DAM system to build a Neural Network Model with usage quantum phenomena based on Hilbert space. The concept of quantum processing based on quantum circuits was also noticed due to its large use and the development of implementing tools as a supplement to the entire area of development of building models based on quantum computing.

Keywords: Hilbert space, quantum measurement, density matrix, Bloch sphere, quantum gates, quantum circuits, quantum key distribution, Quantum-inspired ANN.

1 Introduction

Quantum computing is based on Hilbert space, which plays a fundamental role in the mathematical description of the structure of laws related to quantum mechanics and quantum computing. They are the subject of many studies, both theoretical [2-4,7-9, 16, 21] and practical solutions [10]. One of the two basic mathematical structures describing the Hilbert space is the structure related to the notion of continuity, and the other is linearity. The properties relevant to Hilbert spaces are:

- the concept of the length of a vector (in this case, the norm of a vector),
- the existence of boundaries (and thus the completeness of space),
- orthogonality of vectors (generally, i.e., taking into account the angle between the vectors).

2 Fundamentals of quantum computing

An essential feature of a space is also its completeness, i.e. assuming that a metric space is complete, i.e. if every convergent sequence (in the sense of Cauche convergence) reaches its limit. In a space having such properties it is possible to describe the equations of quantum mechanics. The description of the vector state on the Hilbert space is as follows:

$$\psi = \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \alpha|0> +\beta|1>, \tag{1}$$

where:

ψ - vector state,

α - probability modulus (expressed in a set of real or imaginary numbers), indicating that the vector state will be in the state described by the observable $| \ 0 >$,

β - probability modulus (expressed in a set of real or imaginary numbers), indicating that the vector state will be in the state described by the observable $| \ 1 >$.

To facilitate calculations it was introduced by Dirac [4] the concepts such as bracket, which consists of the vector bra $< \varphi$, the conjugated vector ket $|\varphi\rangle$, and the operations and properties between them. For example of a description of quantum states in H^{2n} dimensional space is as follow:

Base state $| \ 00 >$, which is the state of the vector in H^4 of dimensional space corresponds to the Kronecker product $H^2 \otimes H^2$ in this case $|0 > \otimes|0 >$ that is:

$$| \ 00 >= \begin{bmatrix} 1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ 0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{2}$$

The operator acting on a Hilbert space (called observable) represents a value that can measure the vector states and is represented by the Hermitian operator ($A\dagger = A$).

The quantum state is described by the direction, hence only the relations (proportions) between α and β are important not their values. The system after the measurement is located on one of the observables, and the probability of finding the state vector after the measurement on a given observable, can be for example represented by α, with probability $|\alpha|^2$. The complex numbers α and β are called probability amplitudes, and the square of their modulus is the probability of a real number.

The norm of the state vector is thus determined as follows [22]:

$$||\psi|| = \sqrt{\langle\psi|\psi >} = \sqrt{\begin{bmatrix} \alpha \ \beta \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}} = \sqrt{\alpha^2 + \beta^2}. \tag{3}$$

2.1 Linear operators

In general, the operator A on a vector space is called the mapping of a given vector belonging to that space to a vector belonging to the same vector space, which can be written [21]:

$$A : |\psi >\longrightarrow |\psi >' . \tag{4}$$

A transformation by an operator is linear, and so an operator can be called linear if it satisfies the axioms of linearity:

$$\begin{gathered} A\left(|\psi_1 > +|\,\psi_2 >\right) = A\,|\psi_1 > +A|\,\psi_2 >, \\ A(\alpha \mid \psi >) = \alpha A(|\,\psi >). \end{gathered} \tag{5}$$

From equations (4) and (5) it can be concluded that every observable is represented by a linear operator in a Hilbert space, which is the eigenstate of a Hermitian operator representing some measurable property. A measurement operation in a Hilbert space is an action on a state vector with a special linear operator, which is called observable.

2.2 Density matrix

In the case of a complex system of a Hilbert space, the information about the state vector is based on the so-called partial trace (Tr). The determination of the partial trace is based on the density matrix or the density operator ρ. The density matrix is determined for a given Hilbert space and is given by:

$$|\psi >< \psi| = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \begin{bmatrix} \alpha \ \beta \end{bmatrix} = \begin{bmatrix} \alpha^2 & \alpha\beta \\ \beta\alpha & \beta^2 \end{bmatrix}. \tag{6}$$

At the same time, from the principle of superposition, it is known that: $\alpha^2 + \beta^2 = 1$. As can be seen from the relationship (6), such values (being the square of the probability modulus) are located on the main diagonal of the density matrix, so the trace value Tr is determined as the sum of the diagonal of the density matrix. The sum of the diagonal values of the density matrix is called the trace of the density matrix and is denoted by Tr.

The concept of the density matrix is also associated with the density operator ρ, when we only have a set of vector states and the probability of the system being in a given vector state is defined as:

$$\rho = \sum_i p_i\,|\psi_i >< \psi_i| = \sum_i p_i \begin{bmatrix} \alpha_i^2 & \alpha_i\beta_i \\ \beta_i\alpha_i & \beta_i^2 \end{bmatrix}, \tag{7}$$

where:

p_i - the probability of a quantum state being in a given space described by the state vector of that space.

2.3 State vector measurement

The measurement of the state vector of a quantum system is determined by the base systems at the time of measurement and then the value of the vector will take one of the values represented by these systems, also called observables or bases. Until the measurement, the state of the vector is in the superposition of states representing the observables. If the state vector is in the eigenstate of a given Hermitian operator (observables), then after measurement its value will be equal to that of the observables, this is the so-called pure state, and the measurement result is determined. In most cases, however, the state vector is in the so-called superposition of states, and at the time of measurement its state is subject to the so-called collapse, i.e. it is in one of the eigenstates of the Hermitian matrix. The probability of a state vector on a given observable (i) can be defined as:

$$p(i) = < \psi \left| M_i^\dagger M_i \right| \psi > = |\lambda_i|^2 , \tag{8}$$

where:

M_i - i-th Hermitian measurement operator,

$M_i^\dagger$ - conjugate i-th Hermitian measurement operator,

λ_i– i-th observable.

The normalized value of the state vector V on the observable (i) after the measurement is determined by the relationship [23]:

$$V_i = \frac{M_i \mid \psi >}{\sqrt{\left|\psi > M_i^\dagger M_i \right| \psi \right\rangle}}. \tag{9}$$

The use of the relationship (8) allows to determine of the probability of a state vector on a given observable (i), while the dependence (9) allows to determine of its normalized quantity.

2.4 Bloch Sphere

To illustrate the transformations by quantum states performed by gates, the Bloch sphere is used [15]. The Bloch sphere represents the quantum state of a single-qubit system. Each point on the surface of a sphere corresponds to a certain quantum state. The sphere has a radius of length 1, and the point $(0, 0, 0)$ is in its center. Selected points on the sphere represent specific quantum states.

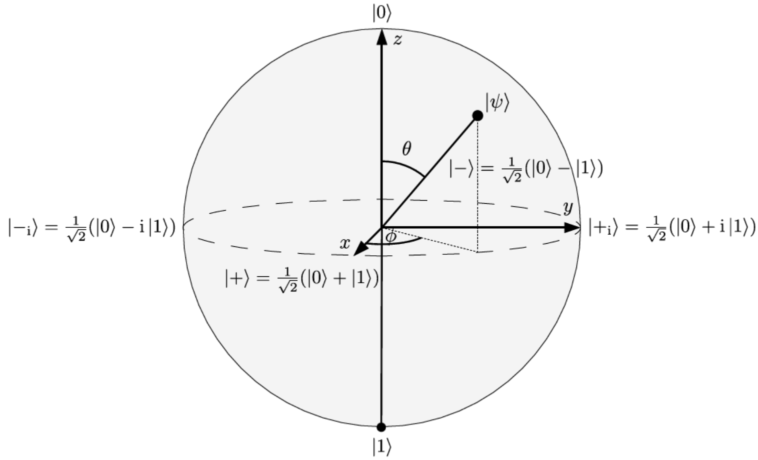

Figure 1. Bloch Sphere, z-axis represents the base quantum state of | 0 > and | 1 >, x - axis represents Hadamard state, y-axis represents the base of complex component. Source [6].

2.5 Quantum gates

An infinite number of quantum gates (operators) can be constructed, it is enough for the gate, which is a square matrix, to satisfy the conditions of the Hermitian transformation, i.e. for a given matrix the property of conjugation and transposition:

$$\mathrm{M} = \mathrm{M}\,\dagger\,. \tag{10}$$

However, in the construction of models based on quantum calculations, single-qubit gates are most often used, one of the most commonly used are the Pauli gates.

$\mathbf{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, which corresponds to the rotation about the x-axis on the Bloch sphere (also called the NOT gate),

$\mathbf{Y} = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$, which corresponds to the rotation about the y-axis on the Bloch sphere,

$\mathbf{Z} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, which corresponds to the rotation about the z-axis on the Bloch sphere,

and Hadamard's gate:

$\mathrm{H} = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

Two-qubit gates are usually obtained by the Kronekcker product of any two one-qubit gates, i.e.:

$$U_1 \otimes U_2,$$

where U - any single-qubit gate.

Very often, a two-qubit gate known as the controlled reversal is used, the so-called controlled-NOT or Controlled -X.

$$\mathrm{CNOT} = \begin{bmatrix} 1\,0\,0\,0 \\ 0\,1\,0\,0 \\ 0\,0\,0\,1 \\ 0\,0\,1\,0 \end{bmatrix}.$$

Examples of CNOT gate operations on two-qubit states:

$\mathrm{CNOT}|00>=|00>, \mathrm{CNOT}|01>=01>, \mathrm{CNOT}|10>=|11>, \mathrm{CNOT}|11>=|10>$.

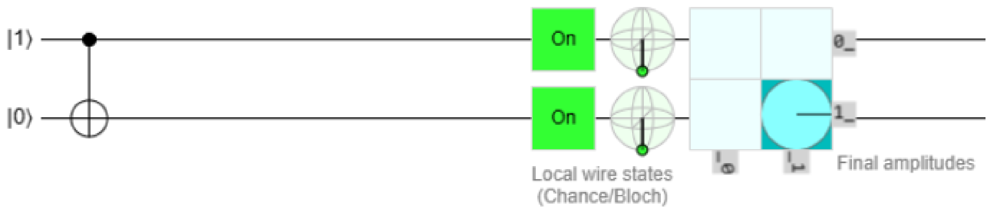

Figure 2. Example of how a CNOT gateway works on a two-qubit state | 10 > (left side) transform on |11> state (right down corner). Source: own elaboration on the site https://algassert.com/quirk.

Based on quantum gates, so-called quantum circuits are built, with the help of which any mathematical formulas implementing quantum models can be implemented. Serial and parallel connections of individual gates can be represented graphically, where each circuit line means qubit, and each gate (representing operator) is marked on the appropriate qubit or several qubits. The time in the circuit runs sequentially from left to right. An example of a quantum circuit realizing an entangled state from a separated state is shown in Fig. 3.

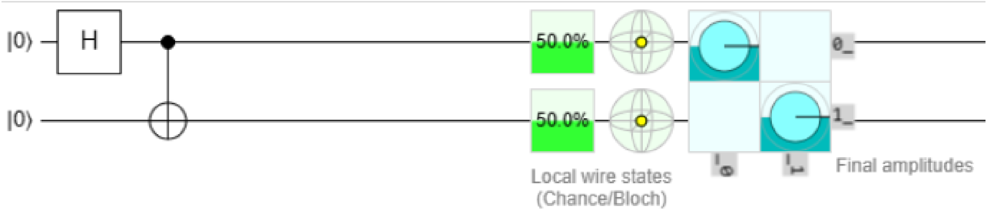

Figure 3. An example of a circuit for obtaining an entangled state. The initial state of the system is $|00\rangle$. After applying the Hadamard H gate, we will get the state $\sqrt{\frac{1}{2}}(|00\rangle + |01\rangle)$. Then, when the CNOT gateway is applied, the state changes to $\sqrt{\frac{1}{2}}(|00\rangle + |11\rangle)$ which is an entangled state (it cannot be represented as the product of two states). Source https://algassert.com/quirk.

3 An example of the application based on quantum models

One of the most promising areas of application of models based on quantum computing is quantum cryptography. It allows you to create safe protocols using the properties of quantum mechanics, such as the ban on cloning or the so-called collapse when measuring the quantum state. One of the models based on these properties is the Quantum encryption key distribution BB84[1].

3.1 Quantum key distribution

Distribution of the encryption key consists of establishing a string of characters that are the basis for encrypting and decrypting messages transmitted by the network. One of the most common methods of determining and distributing an encryption key based on the classical asynchronous model is the protocol RSA [2].

3.2 Protocol BB84

The BB84 protocol is based on the properties of quantum states, combinations of two groups of singlequbit mixed states are used to determine the key:

Mixed state for the z-axis on a Bloch sphere:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{11}$$

Mixed state for the x-axis on a Bloch sphere:

$$|+\rangle = \sqrt{\frac{1}{2}}(|0\rangle + |1\rangle), |-\rangle = \sqrt{\frac{1}{2}}(|0\rangle - |1\rangle). \tag{12}$$

i.e. a total of four single-qubit states and on two groups of measurement operators.

Measurement on the z-axis:

$$\begin{aligned} Z|0\rangle &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 1, \\ Z|1\rangle &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0, \end{aligned} \tag{13}$$

and on the x-axis

[1] It is a quantum key distribution protocol invented by Charles Bennett and Gilles Brassard in 1984. It is the first quantum cryptography protocol.

[2] Algorithm Rivest-Shamira-Adleman (RSA) -one of the first and currently most popular public-key asymmetric cryptographic algorithms, designed in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman.

$$X|+\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \sqrt{\frac{1}{2}}(|0\rangle + |1\rangle) = |+\rangle,$$
$$X|-\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \sqrt{\frac{1}{2}}(|0\rangle - |1\rangle) = |-\rangle. \tag{14}$$

The use of two types of measuring gates allows to distinguish the measurement results of the first and second type of quantum states. The measurement in this case will be unambiguous for the states $|0\rangle, |1\rangle$ using operators X and for states $|+\rangle, |-\rangle$, using operator measurements Z.

The algorithm for determining the key runs according to the following scheme:

Step 1. The sender (S) sends to the receiver (R) sequences of random qubits, from the set $\{|0\rangle, |1\rangle, +\rangle, |-\rangle\}$.

Step 2. The receiver (R) chooses randomly and independently which measurement operator to measure each qubit. The choice is made from a set of fixed measurement operators.

Step 3. The receiver (R) records both the choice of operator and the measurement result for the individual qubits sent by the sender (S) and sends the results of its measurements to him in a transparent way.

Step 4. (S) provides (R) information on which measurements are matched to the states of the given qubits.

Step 5. (S) and (R) divide the qubits into two parts: one contains those measured using the measurement operator appropriate for the given qubit value, and the other group for which the measurement operators used were not valid.

Step 6. The set of quantum states measured by appropriate measurement operators is a set of qubits, which can be an encryption key, e.g. in XOR encryption for the Vernon algorithm.

If incompatibilities were detected during protocol execution (S) and (R), it means that qubits have already been measured by a third party and should not be used to create a key. In this case, they must refrain from transmitting secret information.

Table 1. Shows the measurement results obtained by (R) depending on the type of measurement operator used for a given state of the vector sent by(S).

Uploaded qubit	measurement	result
$\|0\rangle$	Z	0
$\|1\rangle$	Z	1
$\|+\rangle$	X	+
$\|-\rangle$	X	-
$\|0\rangle$	X	50%(+ or -)
$\|1\rangle$	X	50%(+ or -)
$\|+\rangle$	Z	50%(1 or 0)
$\|-\rangle$	Z	50%(1 or 0)

4 Quantum-inspired research based on neural networks

The construction of models based on artificial intelligence [5, 18] and based on artificial neural networks is the subject of many publications and research [11-14, 17-20, 22-23].

The presented methods of building a neural model based on the Day-Ahead Market system operating at TGE S.A. are an extension of previous work [17-18,23]. All operations performed on the Artificial Neural Network are based on mathematical structures specific to Hilbert spaces. The main idea of quantum-inspired computation is:

1. conversion of values represented in the space of real numbers into a quantum state in a Hilbert space, which is represented in the space of complex numbers,
2. performing learning operations for the neural model of the DAM system,
3. measuring quantum states that are represented in the space of real numbers as a result of their "collapse".

The obtained results were compared with a model obtained traditionally, i.e. one in which learning operations were performed only in the space of real numbers.

To implement the quantum-inspired ANN, it was assumed to replace numbers from the decimal system with a 12-element value in the binary system. Binary values can be thought of as one-bit pure states in a Hilbert space that can be quantized, i.e. converted into mixed states using a developed procedure. Individual values from the binary system are therefore converted into quantum states. Subsequently, 12 ANN were built separately for each order of magnitude of quantum states (a value of 12 was assumed to be sufficient to ensure adequate accuracy of calculations). In this way, 12 ANN-based models were built for individual quantum states. Figure 4 shows the idea of building the neural networks, it shows the network for the oldest order of magnitude, for the remaining 11 orders of magnitude, the networks are constructed in an analogous way.

4.1 Procedure algorithm

The model was implemented in the MATLAB environment by using its own m-files. The data used for the training model comes from DAM (first half of the 2019 year), where the volume of electricity delivered and sold is a 24-element input vector, and the volume-weighted average price of electricity obtained for each separate hour of the day is a 24-element output vector. The procedure of data processing was as follows:

1. Separation of the youngest order of quantum states for input and output and base weight matrix,
2. Determine the number of learned networks (Ls) for each order of magnitude,
3. Call a learning function for delimited sets,
4. Recording of the obtained QiANN model,
5. If the value (Ls) has been reached, go to step 6, otherwise go to the point. 3,
6. If not all orders of quantum states have been processed, isolate the next higher order of quantum states for input and output and base weight matrix and transition to page 3 otherwise go to step 7,

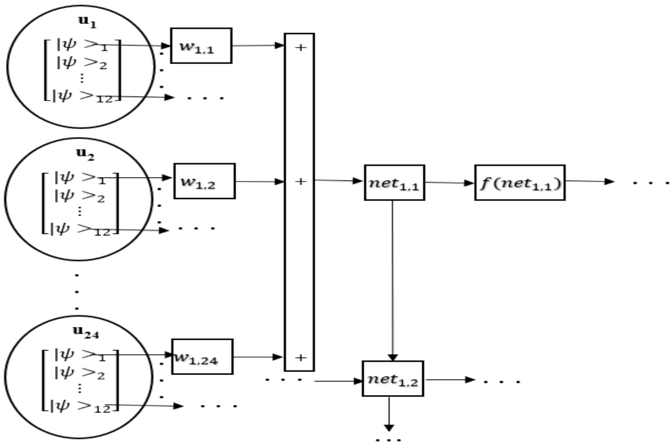

Figure 4. Example of processing mixed quantum states for the hidden layer for the first qubit for the first neuron. Symbols: $f(\text{net}_{1,1})$ - activation function, net $_{1,1}$ - the value of the weighted sum of the input signals of the first quantum state, $w_{1,n}$- weight density matrix, $|\psi>_n$ −n-th quantum mixed state.Source: Own elaboration.

7. Calculation of average quantum states for each order,
8. Measurements of average quantum states - reduction to base 0 or 1,
9. Conversion of base states to a 12-element numeric sequence analogous to binary values,
10. Convert binary values to real numbers,
11. Calculation of mean quadratic error (MSE) as the difference between actual target values and values obtained from QiANN models in point 10.

Markings for Fig. 5:

$uk_1^1 \ldots uk_{12}^1$ - input values of quantum mixed states for consecutive artificial neural networks from 1 to 12,

$wk_1^{1,1} \ldots wk_{12}^{1,1}$ - values of density matrix elements for weights of layers of consecutive artificial neural networks from 1 to 12,

$b_1^1 \ldots b_{12}^1$ - values of density matrix elements for biases of layers of hidden artificial neural networks from 1 to 12 ,

$nk_1^1 \ldots nk_{12}^1$ - values of quantum mixed-state adders (sum of input products and weights and bias) for layers of hidden artificial neural networks from 1 to 12,

$yk_1^1 \ldots yk_{12}^1$ - values of quantum-stored activation functions with arguments in the form of quantum mixed-state adders for individual neurons of the hidden layer for subsequent artificial neural networks from 1 to 12,

$\mathrm{wk}_1^{2,1} \ldots \mathrm{wk}_{12}^{2,1}$ - values of density matrix elements for output layer weights for subsequent artificial neural networks from 1 to 12,

$b_1^2 \ldots b_{12}^2$ - values of density matrix elements for the bias of the output layer for subsequent artificial neural networks from 1 to 12,

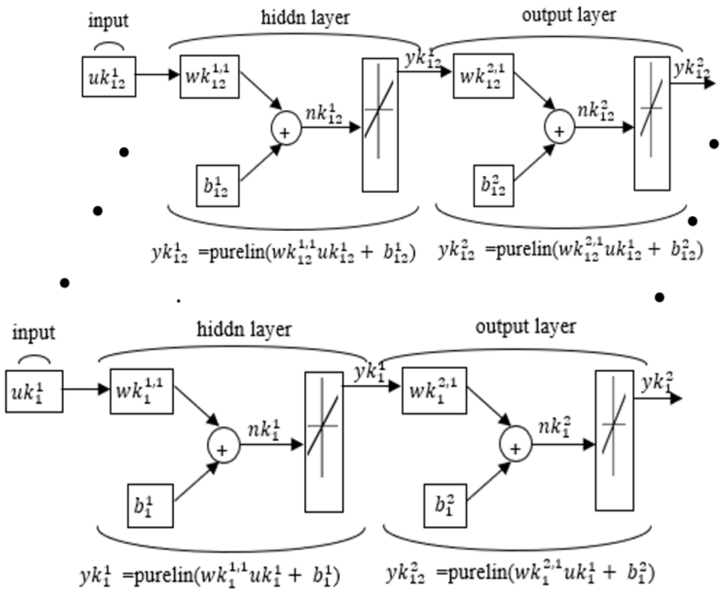

Figure 5. The idea of the architecture of neural networks. Source: own elaboration based on [1].

$nk_1^2 \ldots nk_{12}^2$ - values of quantum mixed-state adders (sum of input products and weights and bias) for output layers of subsequent artificial neural networks from 1 to 12,

$yk_1^2 \ldots yk_{12}^2$ – values of quantum-stored activation functions with arguments in the form of quantum mixed-state adders for individual neurons of the output layer of subsequent artificial neural networks from 1 to 12.

Determination of the values of quantum mixed states denoted by $\mathbf{nk}^1$ boils down to multiplying the density matrix of weights $\mathbf{wk}^1$ in the layer hidden layer by the quantum mixed states which represent input quantities $\mathbf{uk}^1$, as a result, the obtained output is a quantum mixed state $\mathbf{yk}^1$, which is also the entrance to the output layer. In a similar way, the quantum mixed state is determined for the output layer $\mathbf{yk}^2$, where $\mathbf{yk}^1$ is processed by the output layer weights $\mathbf{wk}^2$.

An important issue was the assessment of the value of the quantum adder **nk,** in terms of its value, if the **nk** value exceeded its order of magnitude, its excess was transferred to the next ANN representing the older magnitude of the quantum state (on the principle of the transfer bit). This operation was designed to transfer excess values of the quantum mixed state between the next twelve ANN’s. Due to the operations that can be performed in Hilbert space related to the vector of quantum mixed states, the purelin() activation functions were chosen for all layers of artificial neural networks were used as activation functions.

5 Results

Analyzing the obtained results, which is shown on figure 6, it can be concluded that it is possible to implement a quantum-inspired Artificial Network to teach the model of the Day-Ahead Market System at TGE S.A. Referring to the results obtained, it can be noted that QiANN tends to average the results, i.e. it is more resistant to interference. However, the average error of the MSE value is greater for it than for the Perceptron neural network and is 0.09, and for the Perceptron ANN MSE is 003. It can also be seen that quantum-inspired neural networks, whose operation is based on mathematical models appropriate for quantum computing, are limited to actions possible for these models, i.e., on a Hilbert space. The obtained QiANN model, although the result of many studies and trials, is a proposal of a certain method for implementing neural models and is the subject of further work aimed at using the potential of neural models and the use of quantum inspirations.

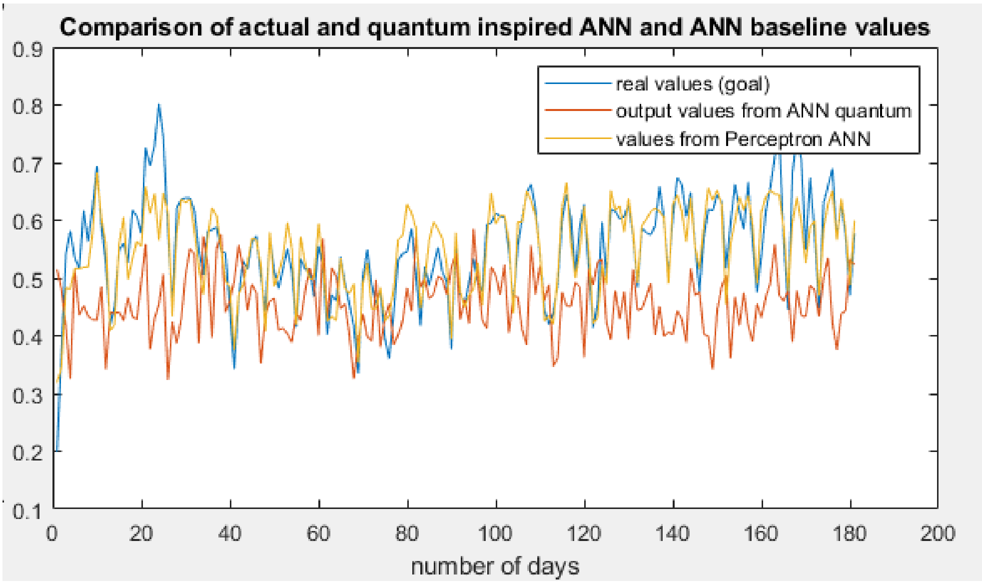

Figure 6. Comparison of actual normalized price values with the Perceptron.ANN model and the quantum-inspired model. Symbols: x-axis (number of days) - consecutive days of the examined period (in this case 181 days) y-axis (average value of the normalized price). Blue – actual normalized price values, yellow – Perceptron ANN output, red – quantum-inspired ANN output. Development of own sources in the MATLAB environment [1].

6 Conclusion

The use of a mathematical apparatus based on models developed to describe quantum phenomena is the subject of extensive and dynamically conducted research. It should be emphasized, however, that they are in the experimental phase, and their practical use is still the subject of work that will take place in the future. The designed and implemented network based on edge processing tends to average the results. An interesting development direction for the construction of the Day-Ahead Market model may be the use of methods based on quantum circuits, especially since, as indicated in this paper, interesting tools facilitating its implementation have been developed. An interesting approach that will be developed in the future is the possibility of building hybrid networks combining the advantages of classical neural networks with quantum-inspired networks.

References

1. Beale M. H. [at all] (1992-2019) Neural Network Toolbox™ User's Guide, The MathWorks, Inc, str. 846.
2. Bernhardt Ch. (2020) Obliczenia kwantowe dla każdego (English: Quantum computing for everyone). WN PWN, Warszawa, stron 202.
3. Chudy. M. (2012) Wprowadzenie do informatyki kwantowej (English: Introduction to quantum computing). AOW EXIT, Warszawa, stron 84.
4. Dirac, P., (1939), A New Notation for Quantum Mechanics, Mathematical Proceedings of the Cambridge Philosophical Society, vol. 35, no. 3, pp. 416–18. https://doi.org/10.1017/s0305004100021162.
5. Faliński M. (2020) Wstęp do sztucznej inteligencji (English: Introduction to artificial intelligence). WN PWN, Warszawa, stron 332.
6. Gawron P., Cholewa M., Kara K., (2016), Rewolucja stanu – fantastyczne wprowadzenie do informatyki kwantowej (English: State Revolution - A fantastic introduction to quantum computing), Instytut Informatyki Teoretycznej i Stosowanej PAN, stron 130.
7. Giaro K., Kamiński M. (2003). Wprowadzenie do algorytmów kwantowych (English: Introduction to quantum algorithms). AOW EXIT, Warszawa, stron 165.
8. Heller M. (2016) Elementy mechaniki kwantowej dla filozofów. (English: Elements of quantum mechanics for philosophers). Copernicus Center Press, stron 191.
9. Hirvensalo M. (2004) Algorytmy kwantowe (English: Quantum algorithms). WSiP, stron 244.
10. Johnston E. R., Harrigan N., Gimeno-Segovia M. (2020) Komputer kwantowy. Programowanie, algorytmy, kod (English: Quantum computer. Programming, algorithms, code). Hellion, Warszawa, stron 278.
11. Kosiński, R. A. (2002) Sztuczne sieci neuronowe. Dynamika nieliniowa i chaos (English:. Artificial neural networks. Dynamika nieliniowa i chaos). WNT, Warszawa, stron 195.
12. Korbicz J., Obuchowicz A., Uciński D. (1994) Sztuczne sieci neuronowe. Podstawy i zastosowania (English: Artificial neural networks. Basics and applications), Problemy Współczesnej Nauki. Teoria i zastosowania, Informatyka, AOW PLJ, stron 251.
13. Mańdziuk J. (2000) Sieci neuronowe typu Hoppfielda (English: Hoppfield neural networks). Teoria i przykłady zastosowań. AOW EXIT, Warszawa, stron 262.
14. Marecki J. (2001) Metody sztucznej inteligencji (English: Artificial intelligence methods),. WSIiZ, Bielsko-Biała, stron 115.
15. Meglio A,, Combarro E., González-Castillo S., (2023), Practical Guide to Quantum Machine Learning and Quantum Optimization, Packt Publishing, p 680.
16. Osowski S. (2013) Sieci neuronowe do przetwarzania informacji (English: Neural networks for information processing). OW PW, Warszawa, stron 422.
17. Patterson J., Gibson A. (2017) Głębokie uczenie się. Praktyczne wprowadzenie (English: Deep Learning. Praktyczne wprowadzenie), Helion, Warszawa, stron 451.

18. Ruciński D. (2023) Modelowanie neuronalne cen na Towarowej Giełdzie Energii Elektrycznej wspomagane algorytmem ewolucyjnym oraz inspirowane obliczeniami kwantowymi (English: Price neural modeling on the Polish Power Exchange supported by an evolutionary algorithm and inspired by quantum calculations). Rozprawa doktorska pod kierunkiem dr hab. inż. Jerzego Tchórzewskiego, prof. UPH w Siedlcach, IBS PAN, Warszawa, str. 253.
19. Ruciński D., Kłopotek M., Tchórzewski J. (2005) Samoorganizujące się bezprzewodowe sieci czujników ad-hock (English: Self-organizing wireless sensor networks ad-hock), Studia Informatica. Systemy i Informatyka, Tom: 1(5), Wydawnictwo Uniwersytetu Podlaskiego, s. 69-80.
20. Rutkowska D., Piliński M., Rutkowski L. (1997). Sieci neuronowe, algorytmy genetyczne i systemy rozmyte (English: Neural networks, genetic algorithms and fuzzy systems), WN PWN, Warszawa - Łódź, stron 411.
21. Rutkowski L. (2020) Metody i techniki sztucznej inteligencji (English: Artificial intelligence methods and techniques). WN PWN, Warszawa, stron 435.
22. Sawerwain M., Wiśniewska J. (2015) Informatyka kwantowa. Wybrane obwody i algorytmy (English: Quantum informatics. Selected circuits and algorithms). WN PWN, Warszawa, stron 371.
23. Tchórzewski, J. (2021) Metody sztucznej inteligencji i informatyki kwantowej w ujęciu teorii sterownia i systemów (English:.Methods of artificial intelligence and quantum information in terms of control theory and systems), WN UPH, Siedlce, stron 343.
24. Tchórzewski J., Ruciński D. (2019) Ewolucyjnie wspierane i inspirowane kwantowo modelowanie neuronowe zastosowane na Polskiej Giełdzie Energii Elektrycznej.(English:.Evolutionarily supported and quantum-inspired neural modeling applied at the Polish Electricity Exchange) 2019 Progress in Applied Electrical Engineering (PAEE), IEEE Digital Library, Kościelisko, s. 1-8.