Article

Not peer-reviewed version

Fuzzy Tsetlin Automata and Their Application in Analog Computing

Evgeny Kagan * and Alexander Rybalov

Posted Date: 11 October 2025

doi: 10.20944/preprints202510.0856.v1

Keywords: fuzzy automata; Tsetlin automata; quantum computation; analog computing doi:10.20944/preprints202510.0856.v1

Article

Fuzzy Tsetlin Automata and Their Application in Analog Computing

Evgeny Kagan [1,*] and Alexander Rybalov [2]

[1] Ariel University, Kiryat ha-Mada, Ariel 4070000, Israel
[2] LAMBDA Lab, Tel Aviv University, Ramat Aviv, Tel Aviv 6997801, Israel
* Correspondence: evganyk@ariel.ac.il

Abstract

In the paper we suggest a system of fuzzy Tsetlin automata for analog computations. The system deals with fuzzy analogs of qubits and models operations of quantum gates used for quantum computations. These fuzzy information units are defined by matrix with fuzzy normalized rows. Operations with these units are conducted by fuzzy Tsetlin automata based on the uninorm and absorbing norm. It is demonstrated that the suggested system implements basic quantum computations that allows its application for general analog computing.

Keywords: fuzzy automata; Tsetlin automata; quantum computation; analog computing

MSC: 03D99; 68Q05; 81P68; 94D05

---

1. Introduction

It is widely accepted [1] that computation is a process of translating one sequence of symbols to another following a finite number of predefined rules. If the first sequence, called input sequence, and the second sequence, called output sequence, consist of the digits representing the numbers, then computation process is also known as calculation process.

Computation process implemented in physical devices is known as computing while the devices are called computation machines. Computation machines dealing with numbers are called arithmetic machines or calculators and are widely known as computers.

During the history of computation were developed innumerous types of computation machines which according to their basic functionality are combined into two wide classes: analog computers and digital computers.

In analog computers, input sequences represent the results of the measurements of certain observed external processes, and translation of these sequences into the outputs is conducted with respect to the internal processes which directly model the external processes [2]. In contrast, digital computers deal with input sequences without necessarily references to the observed reality and translate these sequences into the outputs following the rules which represent abstract mathematical operations [3]. Programming of analog computers is a tuning of the parameters of the modelling processes [4] and programming of digital computers is a defining the rules of translation of the inputs into the outputs [5].

Formally, analog computers operate with continuous functions such that each computer operation is defined by an operator in a certain space. Such principle of functionality of analog computers results in higher speed of calculations, but because of the noise, it leads to irreducible errors at each stage of computing from data transmission to data storing.

This weakness led to a decrease in interest in analog computing and gave rise to the development of digital computers operating with a finite number of variables with the values from discrete alphabets, usually from the binary alphabet $\{0,1\}$.

A new wave of interest to analog computation was initialed by the ideas of quantum computing [6,7] which reintroduced the considerations of the logic of quantum mechanics [8] and its use in computations. Quantum mechanical computation assumes that the input sequences are transformed using quantum mechanical operators and the output sequences are obtained by observations of the results of such transformations [9]. An information unit used in quantum mechanical computation is a qubit which is a superposition of the values 0 and 1. Implementation of such computations assumes the use of quantum mechanical devices which conduct these quantum mechanical transformations and return the results in the form of macroscopic observable values.

Direct analogy between representation of the data in quantum mechanical computation and in fuzzy sets gave a rise to the attempts to model quantum mechanical computation by the means of fuzzy logic [10].

Probably, one of the most successful models was suggested by Hannachi et al. [11,12] who defined fuzzy models of quantum gates and constructed fuzzy version of one- and two-bits Deutsch-Jozsa algorithm [13]. Later, these models were adopted for control of mobile robots [14,15] that validates possibility of their implementation in nonquantum physical devices. Along with that, the Hannachi et al. model does not include complex gates conducting rotations of the qubits which are necessary for a complete quantum computing system.

Moreover, as it was demonstrated by Kreinovich et al. [16], there is a crucial difference between quantum and fuzzy logics led by the "square root of not" appearing in quantum logic. Later, Kosheleva and Kreinovich [17] demonstrated that formulation of quantum logic by the means of fuzzy logic requires additional terms and operations. Kosheleva and Kreinovich interpreted the resulting logic as complex fuzzy logic [18,19] that, however, does not exclude the use of other rich enough fuzzy logics.

In the paper, we suggest a fuzzy system which implements quantum computations and can be used for analog computations in general. The system consists of fuzzy Tsetlin automata that utilize extended uninorm and absorbing norm. To model the qubits representing the data, in the system we suggest fuzzy analog of qubits – the f-bits – which are matrix extensions of the qubits with uninorm based normalization. We demonstrate that the suggested system implements basic quantum computations that allows its application for general analog computing.

2. Materials and Methods

The suggested system is based on fuzzy Tsetlin automata and deals with the operations which represent the operations used in quantum computations. In the section, we start with the basic notes on quantum gates and the methods of quantum computations and then recall definition of fuzzy Tsetlin automaton and its main properties.

2.1. Basic Operations of Quantum Computation

The information units and operations used in quantum computations are defined as follows. Since quantum computations are not a subject of this paper, we use commonly known vector and matrix notation; for widely accepted quantum mechanical notation used in quantum information theory see e.g. the book [20] by Nielsen and Chuang.

In quantum computations, a unit of information is called qubit and is represented by a two-elements complex vector $z = (z_1, z_2)$ such that $0 \leq |z_1|^2 + |z_2|^2 \leq 1$. The distinguished vectors $\mathbb{O} = (1,0)$ and $\mathbb{I} = (0,1)$ are associated with classical false 0 and true 1 values, and any vector z is represented by a combination

$$z = (z_1, z_2) = z_1\mathbb{I} + z_2\mathbb{O} = z_1(0,1) + z_2(1,0)$$

of the vectors $\mathbb{O}$ and $\mathbb{I}$.

Usually, row-vector $z = (z_1, z_2)$ is denoted by $\langle z|$ and is called bra-vector and column vector $z = (z_1, z_2)^T$ is denoted by $|z\rangle$ and is called ket-vector. Vectors $\mathbb{O} = |0\rangle$ and $\mathbb{I} = |1\rangle$ represent the states of electron "spin up" and "spin down", respectively, and the vectors $\left(\frac{1}{\sqrt{2}},\frac{1}{\sqrt{2}}\right)=\frac{1}{\sqrt{2}}|1\rangle+\frac{1}{\sqrt{2}}|0\rangle$ and $\left(\frac{1}{\sqrt{2}},-\frac{1}{\sqrt{2}}\right)=\frac{1}{\sqrt{2}}|1\rangle-\frac{1}{\sqrt{2}}|0\rangle$ represent the states "spin right" and "spin left", respectively.

The basic operators applied to the qubits are represented by the following matrices called quantum gates:

- Pauli operators:

$$I=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix},\quad X=\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix},\quad Y=\begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}\quad \text{and}\quad Z=\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix};$$

- Square root of not:

$$V=\frac{1}{2}\begin{pmatrix}1+i & 1-i\\ 1-i & 1+i\end{pmatrix}=\sqrt{X};$$

- Hadamard operator:

$$H=\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}=\frac{1}{\sqrt{2}}(X+Z);$$

- Phase shift operator:

$$S=\begin{pmatrix}1 & 1\\ 1 & i\end{pmatrix}=\sqrt{Z};$$

- T-gate:

$$T=\begin{pmatrix}1 & 1\\ 1 & e^{i\pi/4}\end{pmatrix}=\sqrt{S}=\sqrt[4]{Z};$$

- Two-qubit controlled NOT ($CNOT$) operator:

$$CNOT=\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix}=\begin{pmatrix}I & 0\\ 0 & X\end{pmatrix}.$$

The square root of not is an operator which acts as $V\big(V(z)\big)=X(z)=\text{not}\, z$; consequently, the phase shift operator acts as $S\big(S(z)\big)=Z(z)$ and the T-gate operators acts as $T\big(T(z)\big)=S(z)$.

Note that in contrast to the Boolean operators, the one-qubit operators are reversible, that is $XX|z\rangle=|z\rangle$, $HH|z\rangle=|z\rangle$ and so on.

Our goal is to suggest fuzzy operations which emulate these quantum gates.

2.2. Fuzzy Tsetlin Automata

The suggested system is based on fuzzy Tsetlin automata. These automata extend binary Tetlin automata [21] which are defined as follows.

Let $\mathfrak{A}$ be an automation with the set of states $\{0,1\}$, and assume that being in the state $s_i\in\{0,1\}$ the automaton receives an income $o\in\{0,1\}$ such that $o=1$ is interpreted as a payoff and $o=0$ is interpreted as a reward (no payoff).

If $o=1$, then the automaton changes its state to the other, and if $o=0$, the automaton stays in its current state. Such transitions of the automaton $\mathfrak{A}$ are defined by the matrix $\rho(o)=\left\|\rho_{ij}(o)\right\|$ such that $\rho(1)=\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$ and $\rho(0)=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$.

It is easy to demonstrate that such behavior of the Tsetlin automaton $\mathfrak{A}$ is represented by Boolean xor operator

$$s(t+1)=\text{xor}\big(s(t),o(t)\big),\qquad s(t)\in\{0,1\},\qquad o(t)\in\{0,1\},\qquad t=0,1,2,\ldots,$$

that allows direct extension of binary automaton $\mathfrak{A}$ to fuzzy automaton $\tilde{\mathfrak{A}}$ which operates with the states and incomes from the interval $[0,1]$.

To define fuzzy Tsetlin automaton $\tilde{\mathfrak{A}}$ we need the following fuzzy aggregators that extend Boolean operations.

Function $\oplus_\theta\colon[0,1]\times[0,1]\to[0,1]$ with the parameter $\theta\in[0,1]$ such that

1. $x\oplus_\theta y=y\oplus_\theta x$ (commutativity),

2. $(x \oplus_\theta y) \oplus_\theta z = x \oplus_\theta (y \oplus_\theta z)$ (associativity),
3. $x \le y$ implies $x \oplus_\theta z \le y \oplus_\theta z$ (monotonicity),
4. $\theta \oplus_\theta x = x$ for some $\theta \in [0,1]$ (identity),

and such that $\oplus_1$ is a t-norm and $\oplus_0$ is a t-conorm is called uninorm. Parameter θ is called identity or neutral element of the uninorm.

Let $\oplus_\theta$ be a uninorm with the neutral element $\theta \in (0,1)$ and let $\ominus_\theta\colon [0,1] \to [0,1]$ be a function that satisfies the following requirements:

1. $\ominus_\theta$ is continuous,
2. $\ominus_\theta$ is strictly monotonously decreasing.

If for any $x, y \in [0,1]$ holds

$$\ominus_\theta (x \oplus_\theta y) = (\ominus_\theta x) \oplus_\theta (\ominus_\theta y).$$

then function $\ominus_\theta$ is called strong negation or negation, for briefness, with respect to the uninorm $\oplus_\theta$.

Function $\otimes_\vartheta\colon [0,1] \times [0,1] \to [0,1]$ with the parameter $\vartheta \in [0,1]$ such that

1. $x \otimes_\vartheta y = y \otimes_\vartheta x$ (commutativity),
2. $(x \otimes_\vartheta y) \otimes_\vartheta z = x \otimes_\vartheta (y \otimes_\vartheta z)$ (associativity),
3. $\vartheta \otimes_\vartheta x = \vartheta$ for any $x \in [0,1]$ (existence of zero)

is called absorbing norm. Parameter ϑ is called absorbing element of the absorbing norm.

Similar to the negation operator, given absorbing norm $\otimes_\vartheta$ it is defined the division operator $\oslash_\vartheta\colon [0,1] \times [0,1] \to [0,1]$ such that

$$(x \otimes_\vartheta y) \oslash_\vartheta y = x \otimes_\vartheta (y \oslash_\vartheta y) = x.$$

The value

$$\lambda = y \oslash_\vartheta y$$

is called unit with respect to the absorbing norm $\otimes_\vartheta$.

Uninorm was defined by Yager and Rybalov in the paper [22], negation – by Dombi in the paper [23], absorbing norm – by Rudas in the paper [24] and division – by Fodor et al. in the report [25]. For further definitions and facts see the book [26] and references therein.

Let $\oplus_\theta$ be a uninorm with neutral element $\theta \in (0,1)$. Then there exists an invertible continuous strictly monotonously increasing function $u\colon (0,1) \to (-\infty, \infty)$ with the limits $lim_{x\to 0} u(x) = -\infty$ and $lim_{x\to 1} u(x) = +\infty$, such that $u(\theta) = \theta$ and for any $x, y \in (0,1)$ holds

$$x \oplus_\theta y = u^{-1}\big(u(x) + u(y)\big).$$

The function u is called generating function of the uninorm $\oplus_\theta$.

Let $\ominus_\theta$ be a negation with respect to the uninorm $\oplus_\theta$ based on the generating function u. Then

$$\ominus_\theta x = u^{-1}\big(-u(x)\big).$$

Finally, let $\otimes_\vartheta$ be an absorbing norm with absorbing element $\vartheta \in (0,1)$. Then there exists an invertible continuous strictly monotonously increasing function $v\colon (0,1) \to (-\infty, \infty)$ with the limits $lim_{x\to 0} v(x) = -\infty$ and $lim_{x\to 1} v(x) = +\infty$, such that $v(\vartheta) = \vartheta$ and for any $x, y \in (0,1)$ holds

$$x \otimes_\vartheta y = v^{-1}\big(v(x) \times v(y)\big).$$

The function v is called generating function of the absorbing norm $\otimes_\vartheta$.

Since absorbing norm is a fuzzy extension of the Boolean *not xor* operator, fuzzy Tsetlin automaton can be defined as a direct negation of the absorbing norm.

Fuzzy Tsetlin automaton $\mathfrak{A}$ is a fuzzy automaton with the state transition function

$$s(t+1) = \ominus_\theta \big(s(t) \otimes_\vartheta o(t)\big), \qquad t = 0, 1, 2, \ldots$$

where $s(t) \in [0,1]$ is a state and $o(t) \in [0,1]$ is an input of the automaton at time t.

Following this definition, fuzzy Tsetlin automaton can be considered as a simplest form of the Tsetlin neuron used in fuzzy neural networks [26,27].

3. Results

Let us start with the definition of fuzzy information unit. We assume that such a unit is analog to the quantum information unit – the qubit – but, in contrast to the probabilities of quantum states, it represents the truth values of the statements.

Let $\gamma_\theta = \|\gamma_{ij}\|_{2\times 2}$ be a real matrix. If in the matrix γ_θ holds

$$0 \le \gamma_{11} \le 1, \quad 0 \le \gamma_{22} \le 1, \quad \text{and} \quad \gamma_{12} = \ominus_\theta \gamma_{11}, \quad \gamma_{21} = \ominus_\theta \gamma_{22}.$$

then it is considered as a fuzzy information unit and is called f-bit.

Note that in contrast to the qubit z which is a complex vector with the elements associated with the probabilities of the quantum states, the f-bit γ_θ is a real matrix. The elements of this matrix are interpreted as truth values and are normalized such that

$$\gamma_{11} \oplus_\theta \gamma_{12} = \theta \quad \text{and} \quad \gamma_{21} \oplus_\theta \gamma_{22} = \theta.$$

In other words, similar to the qubit $z = (z_1, z_2)$ where the values $|z_1|^2$ and $|z_2|^2$ are associated with the probabilities of the complementary events, in the f-bit $\gamma_\theta = \|\gamma_{ij}\|_{2\times 2}$ the values in the rows are associated with the truth values of the complementary statements.

The suggested system consists of four fuzzy Tsetlin automata $\tilde{\mathfrak{A}}$ acting with the elements of fuzzy information units.

Let $c_\theta = \|c_{ij}\|_{2\times 2}$ be a matrix of control values. We assume that matrix c_θ has a structure of f-bit, namely

$$0 \le c_{11} \le 1, \quad 0 \le c_{22} \le 1, \quad \text{and} \quad c_{12} = \ominus_\theta c_{11}, \quad c_{21} = \ominus_\theta c_{22}.$$

Let x_θ be an input f-bit and let c_θ be a control f-bit. Then, controlled information processing is defined as follows:

$$y_{11} = \ominus_\theta (x_{11} \otimes_\vartheta c_{11}), \quad y_{12} = x_{12} \otimes_\vartheta c_{12},$$

$$y_{21} = x_{21} \otimes_\vartheta c_{21}, \quad y_{22} = \ominus_\theta (x_{22} \otimes_\vartheta c_{22}),$$

In other words, the automata processing the elements of main diagonals are fuzzy Tsetlin automata and the automata processing the element of antidiagonals are negated fuzzy Tsetlin automata.

In brief, this operation is written as

$$y = \tilde{\mathfrak{T}}(x_\theta, c_\theta).$$

Lemma 1. *If* $\theta = \vartheta = \frac{1}{2}$, *then* y *is an f-bit.*

Proof. We need to show that

$$0 \le y_{ij} \le 1, \quad i, j = 1, 2,$$

and that

$$y_{12} = \ominus_\theta y_{11} \text{ and } y_{21} = \ominus_\theta y_{22}.$$

The first statement follows directly from definitions of the operations $\ominus_\theta$ and $\otimes_\vartheta$ for any θ and ϑ.

Consider the second statement. Since $\theta = \vartheta = \frac{1}{2}$, we have

$$y_{12} = x_{12} \otimes_\vartheta c_{12} = (\ominus_\theta x_{11}) \otimes_\vartheta (\ominus_\theta c_{11}) = x_{11} \otimes_\vartheta c_{11} = \ominus_\theta y_{11},$$

and

$$y_{21} = x_{21} \otimes_{\vartheta} c_{21} = (\ominus_{\theta} x_{22}) \otimes_{\vartheta} (\ominus_{\theta} c_{22}) = x_{22} \otimes_{\vartheta} c_{22} = \ominus_{\theta} y_{22}.$$

Note that if $\theta = \vartheta \neq \frac{1}{2}$, then the above transformations do not hold, and the statement of the theorem is not true. □

Let $\theta = \vartheta = \frac{1}{2}$. We define the following control matrices.

- Fuzzy Pauli control matrices:

$$\tilde{I}_{\theta} = \begin{pmatrix} \ominus_{\theta} \lambda & \lambda \\ \lambda & \ominus_{\theta} \lambda \end{pmatrix}, \quad \tilde{X}_{\theta} = \begin{pmatrix} \lambda & \ominus_{\theta} \lambda \\ \ominus_{\theta} \lambda & \lambda \end{pmatrix},$$

$$\tilde{Y}_{\theta} = \begin{pmatrix} \lambda & \ominus_{\theta} \lambda \\ \lambda & \ominus_{\theta} \lambda \end{pmatrix} \quad \text{and} \quad \tilde{Z}_{\theta} = \begin{pmatrix} \ominus_{\theta} \lambda & \lambda \\ \ominus_{\theta} \lambda & \lambda \end{pmatrix};$$

- Fuzzy root of not:

$$\tilde{V}_{\theta} = \begin{pmatrix} \nu & \ominus_{\theta} \nu \\ \ominus_{\theta} \nu & \nu \end{pmatrix}, \quad \text{s.t.} \quad \nu \oplus_{\theta} \nu = \lambda;$$

- Fuzzy Hadamard control matrix:

$$\tilde{H}_{\theta} = \begin{pmatrix} (\ominus_{\theta} \lambda) \oplus_{\theta} (\ominus_{\theta} \lambda) & \lambda \oplus_{\theta} \lambda \\ \lambda \oplus_{\theta} \lambda & (\ominus_{\theta} \lambda) \oplus_{\theta} (\ominus_{\theta} \lambda) \end{pmatrix};$$

- Fuzzy phase shift:

$$\tilde{S}_{\theta} = \begin{pmatrix} \ominus_{\theta} \nu & \nu \\ \ominus_{\theta} \nu & \nu \end{pmatrix}, \quad \text{s.t.} \quad \nu \oplus_{\theta} \nu = \lambda;$$

- Fuzzy T-gate:

$$\tilde{T}_{\theta} = \begin{pmatrix} \ominus_{\theta} \tau & \tau \\ \ominus_{\theta} \tau & \tau \end{pmatrix}, \quad \text{s.t.} \quad \tau \oplus_{\theta} \tau = \nu$$

- Fuzzy two-f-bit controlled NOT ($CNOT$):

$$\widetilde{CNOT}_{\theta} = \begin{pmatrix} \ominus_{\theta} \lambda & \lambda & \theta & \theta \\ \lambda & \ominus_{\theta} \lambda & \theta & \theta \\ \theta & \theta & \lambda & \ominus_{\theta} \lambda \\ \theta & \theta & \ominus_{\theta} \lambda 1 & \lambda \end{pmatrix} = \begin{pmatrix} \tilde{I}_{\theta} & \theta \\ \theta & \tilde{X}_{\theta} \end{pmatrix}.$$

Since $0 \leq \lambda \leq 1$, each presented matrix is an f-bit.

Theorem 1. *Operation $\mathfrak{T}$ with the defined above control matrices implements the basic gates of quantum computation.*

Proof. Denote the input qubit by

$$|z\rangle = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix},$$

and the input f-bit by

$$x_{\theta} = \begin{pmatrix} x_1 & \ominus_{\theta} x_1 \\ \ominus_{\theta} x_2 & x_2 \end{pmatrix}.$$

We have (for detailed calculations see Appendix A):

- Pauli operators:

$$I|z\rangle = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}, \quad X|z\rangle = \begin{pmatrix} z_2 \\ z_1 \end{pmatrix}, \quad Y|z\rangle = \begin{pmatrix} -iz_2 \\ iz_1 \end{pmatrix}, \quad Z|z\rangle = \begin{pmatrix} z_1 \\ -z_2 \end{pmatrix};$$

- Fuzzy Pauli operators:

$$\mathfrak{T}(x_{\theta}, \tilde{I}_{\theta}) = \begin{pmatrix} x_1 & \ominus_{\theta} x_1 \\ \ominus_{\theta} x_2 & x_2 \end{pmatrix}, \quad \mathfrak{T}(x_{\theta}, \tilde{X}_{\theta}) = \begin{pmatrix} \ominus_{\theta} x_1 & x_1 \\ x_2 & \ominus_{\theta} x_2 \end{pmatrix},$$

$$\mathfrak{T}(x_{\theta}, \tilde{Y}_{\theta}) = \begin{pmatrix} \ominus_{\theta} x_1 & x_1 \\ \ominus_{\theta} x_2 & x_2 \end{pmatrix}, \quad \mathfrak{T}(x_{\theta}, \tilde{Z}_{\theta}) = \begin{pmatrix} x_1 & \ominus_{\theta} x_1 \\ x_2 & \ominus_{\theta} x_2 \end{pmatrix};$$

- Square root of not:

$$V|z\rangle = \frac{1}{2}\binom{z_1+z_2+i(z_1-z_2)}{z_1+z_2+i(z_2-z_1)};$$

- Fuzzy root of not:

$$\widetilde{\mathfrak{T}}\left(x_\theta, \tilde{V}_\theta\right) = \begin{pmatrix} \ominus_\theta\,(x_1 \otimes_\vartheta \nu) & x_1 \otimes_\vartheta \nu \\ x_2 \otimes_\vartheta \nu & \ominus_\theta\,(x_2 \otimes_\vartheta \nu) \end{pmatrix}, \quad \nu \oplus_\theta \nu = \lambda;$$

- Hadamard operator:

$$H|z\rangle = \frac{1}{\sqrt{2}}\binom{z_1+z_2}{z_1-z_2};$$

- Fuzzy Hadamard operator:

$$\widetilde{\mathfrak{T}}\left(x_\theta, \widetilde{H}_\theta\right) = \begin{pmatrix} x_1 \otimes_\vartheta (\lambda \oplus_\theta \lambda) & (\ominus_\theta\, x_1) \otimes_\vartheta (\lambda \oplus_\theta \lambda) \\ (\ominus_\theta\, x_2) \otimes_\vartheta (\lambda \oplus_\theta \lambda) & x_1 \otimes_\vartheta (\lambda \oplus_\theta \lambda) \end{pmatrix};$$

- Phase shift operator:

$$S|z\rangle = \binom{z_1+z_2}{z_1+iz_2};$$

- Fuzzy phase shift operator:

$$\widetilde{\mathfrak{T}}\left(x_\theta, \tilde{S}_\theta\right) = \begin{pmatrix} x_1 \otimes_\vartheta \nu & (\ominus_\theta\, x_1) \otimes_\vartheta \nu \\ x_2 \otimes_\vartheta \nu & (\ominus_\theta\, x_2) \otimes_\vartheta \nu \end{pmatrix}, \quad \nu \oplus_\theta \nu = \lambda;$$

- T-gate:

$$T|z\rangle = \binom{z_1+z_2}{z_1+e^{i\pi/4}z_2};$$

- Fuzzy T-gate:

$$\widetilde{\mathfrak{T}}\left(x_\theta, \tilde{T}_\theta\right) = \begin{pmatrix} x_1 \otimes_\vartheta \tau & \ominus_\theta\,(x_1 \otimes_\vartheta \tau) \\ x_2 \otimes_\vartheta \tau & \ominus_\theta\,(x_2 \otimes_\vartheta \tau) \end{pmatrix}, \quad \tau \oplus_\theta \tau = \nu.$$

Finally, let

$$|w\rangle = \binom{w_1}{w_2},$$

be additional qubit and

$$y_\theta = \begin{pmatrix} y_1 & \ominus_\theta\, y_1 \\ \ominus_\theta\, y_2 & y_2 \end{pmatrix}.$$

be additional f-bit.

The $CNOT$ gate and two-f-bit extension of the $\widetilde{\mathfrak{T}}$ operation with fuzzy $CNOT$ control matrix act as follows:

- Two-qubit controlled NOT ($CNOT$) operator:

$$CNOT|zw\rangle = \begin{pmatrix} z_1 \\ z_2 \\ w_2 \\ w_1 \end{pmatrix};$$

- Fuzzy two-f-bit controlled NOT ($CNOT$) operator:

$$\widetilde{\mathfrak{T}}\left(x_\theta, y_\theta, \widetilde{CNOT}_\theta\right) = \begin{pmatrix} x_1 & \ominus_\theta\, x_1 & \theta & \theta \\ \ominus_\theta\, x_2 & x_2 & \theta & \theta \\ \theta & \theta & \ominus_\theta\, y_1 & y_1 \\ \theta & \theta & y_2 & \ominus_\theta\, y_2 \end{pmatrix}.$$

Representation of the $CNOT$ gate by its fuzzy analog ends the proof. □

First, let us demonstrate that the operation $\widetilde{\mathfrak{T}}$ with the introduced fuzzy Pauli control matrices meets the main property of quantum gates.

Lemma 2. *Operation* $\widetilde{\mathfrak{T}}$ *with fuzzy Pauli control matrices* $\tilde{I}_\theta$, $\tilde{X}_\theta$, $\tilde{Y}_\theta$ *and* $\tilde{Z}_\theta$ *is reversible such that*

$$\widetilde{\mathfrak{T}}^2(x_\theta, c_\theta) = x_\theta.$$

Proof. Let

$$x_\theta = \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix} \quad \text{and} \quad c_\theta = \begin{pmatrix} c_1 & \ominus_\theta c_1 \\ \ominus_\theta c_2 & c_2 \end{pmatrix}$$

be an f-bit and a control matrix, respectively.

Then

$$\widetilde{\mathfrak{T}}^2(x_\theta, c_\theta) = \widetilde{\mathfrak{T}}\left(\widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} c_1 & \ominus_\theta c_1 \\ \ominus_\theta c_2 & c_2 \end{pmatrix}\right), \begin{pmatrix} c_1 & \ominus_\theta c_1 \\ \ominus_\theta c_2 & c_2 \end{pmatrix}\right) =$$

$$= \widetilde{\mathfrak{T}}\left(\begin{pmatrix} \ominus_\theta (x_1 \otimes_\vartheta c_1) & x_1 \otimes_\vartheta c_1 \\ x_2 \otimes_\vartheta c_2 & \ominus_\theta (x_2 \otimes_\vartheta c_2) \end{pmatrix}, \begin{pmatrix} c_1 & \ominus_\theta c_1 \\ \ominus_\theta c_2 & c_2 \end{pmatrix}\right) =$$

$$= \begin{pmatrix} x_1 \otimes_\vartheta c_1 \otimes_\vartheta c_1 & \ominus_\theta (x_1 \otimes_\vartheta c_1 \otimes_\vartheta c_1) \\ \ominus_\theta (x_2 \otimes_\vartheta c_2 \otimes_\vartheta c_2) & x_2 \otimes_\vartheta c_2 \otimes_\vartheta c_2 \end{pmatrix}.$$

For the fuzzy Pauli control matrices $\tilde{I}_\theta$, $\tilde{X}_\theta$, $\tilde{Y}_\theta$ and $\tilde{Z}_\theta$, we have either $c_i = \lambda$ or $c_i = \ominus_\theta \lambda$, $i = 1, 2$. Then

$$c_i \otimes_\vartheta c_i = \lambda, \quad i = 1, 2.$$

Hence,

$$\widetilde{\mathfrak{T}}^2(x_\theta, c_\theta) = \begin{pmatrix} x_1 \otimes_\vartheta c_1 \otimes_\vartheta c_1 & \ominus_\theta (x_1 \otimes_\vartheta c_1 \otimes_\vartheta c_1) \\ \ominus_\theta (x_2 \otimes_\vartheta c_2 \otimes_\vartheta c_2) & x_2 \otimes_\vartheta c_2 \otimes_\vartheta c_2 \end{pmatrix} =$$

$$= \begin{pmatrix} x_1 \otimes_\vartheta \lambda & \ominus_\theta (x_1 \otimes_\vartheta \lambda) \\ \ominus_\theta (x_2 \otimes_\vartheta \lambda) & x_2 \otimes_\vartheta \lambda \end{pmatrix} = \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix} = x_\theta,$$

and operation $\widetilde{\mathfrak{T}}$ is reversible. □

Corollary 1. *$\widetilde{CNOT}_\theta$ is reversible.*

Proof. Reversibility of $\widetilde{CNOT}_\theta$ follows directly from the reversibility of $\tilde{I}_\theta$ and $\tilde{X}_\theta$. □

Now, let us demonstrate the relations between the introduced control matrices.

Lemma 3.

$$\tilde{I}_\theta \oplus_\theta \tilde{X}_\theta = \begin{pmatrix} \theta & \theta \\ \theta & \theta \end{pmatrix}, \quad \tilde{Y}_\theta \oplus_\theta \tilde{Z}_\theta = \begin{pmatrix} \theta & \theta \\ \theta & \theta \end{pmatrix},$$

$$\tilde{I}_\theta \oplus_\theta \tilde{I}_\theta = \tilde{H}_\theta, \quad \tilde{V}_\theta \oplus_\theta \tilde{V}_\theta = \tilde{X}_\theta, \quad \tilde{T}_\theta \oplus_\theta \tilde{T}_\theta = \tilde{S}_\theta, \quad \tilde{S}_\theta \oplus_\theta \tilde{S}_\theta = \tilde{Z}_\theta.$$

Proof. The statement is obtained by direct calculations (see Appendix B). □

Note that $\widetilde{\mathfrak{T}}$ operation with fuzzy Hadamard control matrix $\tilde{H}_\theta$ and the control matrices $\tilde{V}_\theta$, $\tilde{S}_\theta$ and $\tilde{T}_\theta$ is not reversible. Along with that, for these matrices we can define the operations which, together with corresponding operators, provide the reversibility of the operation $\widetilde{\mathfrak{T}}$.

Let $x \in (0, 1)$. Denote by

$$\oplus_\theta^n x = \underbrace{x \oplus_\theta \dots \oplus_\theta x}_{n \text{ times}} \quad \text{and} \quad \otimes_\vartheta^n x = \underbrace{x \otimes_\vartheta \dots \otimes_\vartheta x}_{n \text{ times}},$$

the n times application of the uninorm $\oplus_\theta$ and absorbing norm $\otimes_\theta$. It means,

$$\oplus_\theta^1 x = x \quad \text{and} \quad \otimes_\vartheta^1 x = x.$$

For any $n = 1, 2, \dots$ we have,

$$\otimes_{\vartheta}^{n} \lambda = \lambda,$$

and since $\nu \oplus_{\theta} \nu = \lambda$ and $\tau \oplus_{\theta} \tau = \nu$, we have for even n

$$\oplus_{\theta}^{n} \nu = \oplus_{\theta}^{n/2} \lambda \quad \text{and} \quad \oplus_{\theta}^{n} \tau = \oplus_{\theta}^{n/2} \nu,$$

and for odd n

$$\oplus_{\theta}^{n} \nu = \nu \oplus_{\theta} \left(\oplus_{\theta}^{(n-1)/2} \lambda\right) \quad \text{and} \quad \oplus_{\theta}^{n} \tau = \tau \oplus_{\theta} \left(\oplus_{\theta}^{(n-1)/2} \nu\right).$$

We introduce the following matrices

$$\Sigma_n(x \mid \theta) = \begin{pmatrix} \oplus_{\theta}^{n} x & \oplus_{\theta}^{n} x \\ \oplus_{\theta}^{n} x & \oplus_{\theta}^{n} x \end{pmatrix} \quad \text{and} \quad \Sigma_n^{-1}(x \mid \theta, \vartheta) = \begin{pmatrix} \lambda \oslash_{\vartheta} (\oplus_{\theta}^{n} x) & \lambda \oslash_{\vartheta} (\oplus_{\theta}^{n} x) \\ \lambda \oslash_{\vartheta} (\oplus_{\theta}^{n} x) & \lambda \oslash_{\vartheta} (\oplus_{\theta}^{n} x) \end{pmatrix},$$

$$\Pi_n(x \mid \vartheta) = \begin{pmatrix} \otimes_{\vartheta}^{n} x & \otimes_{\vartheta}^{n} x \\ \otimes_{\vartheta}^{n} x & \otimes_{\vartheta}^{n} x \end{pmatrix} \quad \text{and} \quad \Pi_n^{-1}(x \mid \vartheta) = \begin{pmatrix} \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{n} x) & \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{n} x) \\ \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{n} x) & \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{n} x) \end{pmatrix}.$$

Since we assume that $\theta = \theta = \frac{1}{2}$, below we will omit the notation of θ and θ and will write $\Sigma_n(x)$, $\Sigma_n^{-1}(x)$, $\Pi_n(x)$ and $\Pi_n^{-1}(x)$.

To define the required operators, we will need the matrices

$$\Sigma_4^{-1}(\lambda) = \begin{pmatrix} \lambda \oslash_{\vartheta} (\oplus_{\theta}^{4} \lambda) & \lambda \oslash_{\vartheta} (\oplus_{\theta}^{4} \lambda) \\ \lambda \oslash_{\vartheta} (\oplus_{\theta}^{4} \lambda) & \lambda \oslash_{\vartheta} (\oplus_{\theta}^{4} \lambda) \end{pmatrix} =$$

$$= \begin{pmatrix} \lambda \oslash_{\vartheta} \left(\underbrace{\lambda \oplus_{\theta} \dots \oplus_{\theta} \lambda}_{4\ times}\right) & \lambda \oslash_{\vartheta} \left(\underbrace{\lambda \oplus_{\theta} \dots \oplus_{\theta} \lambda}_{4\ times}\right) \\ \lambda \oslash_{\vartheta} \left(\underbrace{\lambda \oplus_{\theta} \dots \oplus_{\theta} \lambda}_{4\ times}\right) & \lambda \oslash_{\vartheta} \left(\underbrace{\lambda \oplus_{\theta} \dots \oplus_{\theta} \lambda}_{4\ times}\right) \end{pmatrix},$$

$$\Pi_2^{-1}(\nu) = \begin{pmatrix} \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{2} \nu) & \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{2} \nu) \\ \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{2} \nu) & \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{2} \nu) \end{pmatrix} = \begin{pmatrix} \lambda \oslash_{\vartheta} (\nu \otimes_{\vartheta} \nu) & \lambda \oslash_{\vartheta} (\nu \otimes_{\vartheta} \nu) \\ \lambda \oslash_{\vartheta} (\nu \otimes_{\vartheta} \nu) & \lambda \oslash_{\vartheta} (\nu \otimes_{\vartheta} \nu) \end{pmatrix}.$$

and

$$\Pi_4^{-1}(\tau) = \begin{pmatrix} \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{4} \tau) & \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{4} \tau) \\ \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{4} \tau) & \lambda \oslash_{\vartheta} (\otimes_{\vartheta}^{4} \tau) \end{pmatrix} =$$

$$= \begin{pmatrix} \lambda \oslash_{\vartheta} \left(\underbrace{\tau \otimes_{\vartheta} \dots \otimes_{\vartheta} \tau}_{4\ times}\right) & \lambda \oslash_{\vartheta} \left(\underbrace{\tau \otimes_{\vartheta} \dots \otimes_{\vartheta} \tau}_{4\ times}\right) \\ \lambda \oslash_{\vartheta} \left(\underbrace{\tau \otimes_{\vartheta} \dots \otimes_{\vartheta} \tau}_{4\ times}\right) & \lambda \oslash_{\vartheta} \left(\underbrace{\tau \otimes_{\vartheta} \dots \otimes_{\vartheta} \tau}_{4\ times}\right) \end{pmatrix}.$$

Lemma 4. *For fuzzy Hadamard operator holds*

$$\tilde{\mathfrak{T}}\left(\tilde{\mathfrak{T}}^2\left(x_{\theta}, \tilde{H}_{\theta}\right), \Sigma_2^{-1}(\lambda)\right) = x_{\theta}.$$

Proof. The statement is obtained by direct calculations (see Appendix C). □

Similar statement holds for fuzzy square root of not $\tilde{V}_{\theta}$, fuzzy phase shift $\tilde{S}_{\theta}$ and fuzzy T-gate $\tilde{T}_{\theta}$.

Lemma 5.

$$\tilde{\mathfrak{T}}\left(\tilde{\mathfrak{T}}^2\left(x_{\theta}, \tilde{V}_{\theta}\right), \Pi_2^{-1}(\nu)\right) = x_{\theta},$$

$$\tilde{\mathfrak{T}}\left(\tilde{\mathfrak{T}}^2\left(x_{\theta}, \tilde{S}_{\theta}\right), \Pi_2^{-1}(\nu)\right) = x_{\theta},$$

$$\tilde{\mathfrak{T}}\left(\tilde{\mathfrak{T}}^4\left(x_{\theta}, \tilde{T}_{\theta}\right), \Pi_4^{-1}(\tau)\right) = x_{\theta}.$$

Proof. The statement is obtained by direct calculations (see Appendix D). □

Finally, let us present an example of the generating function $u \equiv v$ with $\theta = \vartheta = \frac{1}{2}$ and corresponding values λ, ν and τ.

Example 1. Let $x \in (0,1)$ and $\xi \in (-\infty, \infty)$ and assume that [26]

$$u(x) = v(x) = \ln\frac{x^\alpha}{1-x^\alpha}, \qquad u^{-1}(\xi) = v^{-1}(\xi) = \left(\frac{e^\xi}{1+e^\xi}\right)^{1/\alpha},$$

where $\alpha = 1/\log_2\frac{1}{\theta}$, $\theta \in (0,1)$.

For $\theta = \frac{1}{2}$ we have $\alpha = 1/\log_2\frac{1}{1/2} = 1$. So

$$u(x) = v(x) = \ln\frac{x}{1-x}, \qquad u^{-1}(\xi) = v^{-1}(\xi) = \frac{e^\xi}{1+e^\xi}.$$

Then

$$\lambda \approx 0.7311, \qquad \nu \approx 0.6225 \qquad \text{and} \qquad \tau \approx 0.5622.$$

In fact,

$$\lambda = x \oslash_\vartheta x = u^{-1}\big(u(x)/u(x)\big) = u^{-1}(1) = \frac{e}{1+e} \approx 0.7311.$$

Then

$$u^{-1}\big(u(\nu) + u(\nu)\big) = u^{-1}\big(2u(\nu)\big) = \frac{\exp\left(2\ln\frac{\nu}{1-\nu}\right)}{1+\exp\left(2\ln\frac{\nu}{1-\nu}\right)} = \frac{\nu^2}{2(\nu-1)\nu+1} = \lambda,$$

which gives $\nu \approx 0.6225$ and similarly

$$u^{-1}\big(u(\tau) + u(\tau)\big) = u^{-1}\big(2u(\tau)\big) = \frac{\exp\left(2\ln\frac{\tau}{1-\tau}\right)}{1+\exp\left(2\ln\frac{\tau}{1-\tau}\right)} = \frac{\tau^2}{2(\tau-1)\tau+1} = \nu$$

gives $\tau \approx 0.5622$. □

Summarizing, operation $\tilde{\mathfrak{T}}$ with the defined fuzzy control matrices implements the corresponding quantum gates. Operation $\tilde{\mathfrak{T}}$ is reversible for the fuzzy Pauli operators $\tilde{I}_\theta$, $\tilde{X}_\theta$, $\tilde{Y}_\theta$ and $\tilde{Z}_\theta$ and for fuzzy controlled not operator $\widetilde{CNOT}_\theta$, and for fuzzy Hadamard operator $\widetilde{H}_\theta$ operation it is reversed using the matrix $\Sigma_2^{-1}(\lambda)$. For square root of not $\tilde{V}_\theta$ and fuzzy phase shift operation $\tilde{\mathfrak{T}}$ is reversed using the matrix $\Pi_2^{-1}(\nu)$ and for the fuzzy T-gate $\tilde{T}_\theta$ it is reversed using the matrix $\Pi_4^{-1}(\tau)$. Finally, example 1 demonstrates that the introduced operations and values are computable.

These properties of operation $\tilde{\mathfrak{T}}$ with the defined fuzzy control matrices allows its direct application in quantum algorithms. Below, we illustrate the use of this operation by formulating fuzzy version of the one qubit Deutsch-Jozsa algorithm [13].

Example 2. In this example, we formulate the fuzzy version of the one qubit Deutsch-Jozsa algorithm [13], which recognizes whether a Boolean function $f: \{0,1\} \to \{0,1\}$ is constant (its outputs do not depend on inputs) or balanced (a half of its outputs are zeros and a half are units). The example is inspired by the implementation of the model suggested by Hannachi, Hatakeyama and Hirota [12].

Recall that since we consider Boolean function f of one argument, it can be one of four functions $f_k: \{0,1\} \to \{0,1\}$, $k = 1, \ldots, 4$, such that the functions $f_1(x) = 0$ and $f_2(x) = 1$ are constant and the functions $f_3(x) = x$ and $f_4(x) = \text{not}\, x$ are balanced.

To solve this problem, classical algorithm requires two calls of the function f. At first, the algorithm sets $x = 0$. Assume that the result is $f(0) = 0$. It means that the function f is either f_1 (constant) or f_3 (balanced). Now the algorithm sets $x = 1$. Then the result is either $f(1) = 0$, which means that the function f is f_1 (constant), or $f(1) = 1$, which means that the function f is f_3

(balanced). Certainly, the algorithm can start with the value $x = 1$ and then set the value $x = 0$; in this case it will choose among the functions f_2 and f_4.

In contrast, the Deutsch-Jozsa algorithm solves this problem in one call of the operator associated with the function f. Detailed description of the Deutsch-Jozsa algorithm is given in Appendix E.

Following the scheme of the Deutsch-Jozsa algorithm, the suggested algorithm obtains two f-bits x_θ and y_θ as an input, applies to them fuzzy Hadamard operator $\widetilde{\mathfrak{T}}(\cdot, \widetilde{H}_\theta)$, then applies a two-f-bit oracle operator $\widetilde{\mathfrak{T}}(\cdot, \widetilde{U}_{\theta f})$. The obtained f-bits are processes by the operator $\widetilde{\mathfrak{T}}(\widetilde{\mathfrak{T}}(\cdot, \widetilde{H}_\theta), \Sigma_2^{-1}(\lambda)\ )$, which is an inverse of the operator $\widetilde{\mathfrak{T}}(\cdot, \widetilde{H}_\theta)$, and the resulting f-bits x'_θ and y'_θ are compared. If $x'_\theta = y'_\theta$, then the function f is constant and if $x'_\theta \neq y'_\theta$, then the function f is balanced.

Similar to the Deutsch-Jozsa algorithm, we assume that the functions f_1, f_2, f_3 and f_4 are respectively associated with the two-f-bit operators with the control matrices:

$$\widetilde{U}_{\theta 1} = \begin{pmatrix} \tilde{I}_\theta & \theta \\ \theta & \tilde{I}_\theta \end{pmatrix}, \quad \widetilde{U}_{\theta 2} = \begin{pmatrix} \tilde{X}_\theta & \theta \\ \theta & \tilde{X}_\theta \end{pmatrix}, \quad \widetilde{U}_{\theta 3} = \begin{pmatrix} \tilde{I}_\theta & \theta \\ \theta & \tilde{X}_\theta \end{pmatrix} \quad \text{and} \quad \widetilde{U}_{\theta 4} = \begin{pmatrix} \tilde{X}_\theta & \theta \\ \theta & \tilde{I}_\theta \end{pmatrix}.$$

respectively.

Let

$$x_\theta = \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix} \quad \text{and} \quad y_\theta = \begin{pmatrix} y_1 & \ominus_\theta y_1 \\ \ominus_\theta y_2 & y_2 \end{pmatrix}.$$

The two-f-bit state $\tilde{\psi}_0$ is defined as a fuzzy superposition of the f-bits x_θ and y_θ:

$$\tilde{\psi}_0 = \begin{pmatrix} x_\theta & \ominus_\theta y_\theta \\ \ominus_\theta y_\theta & x_\theta \end{pmatrix}.$$

To the state $\tilde{\psi}_0$ the two-f-qubit Hadamard operator is applied:

$$\tilde{\psi}_1 = \widetilde{\mathfrak{T}}(\tilde{\psi}_0, \widetilde{H}_\theta) = \begin{pmatrix} x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) & y_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \\ y_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) & x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \end{pmatrix}.$$

The two-f-qubit state $\tilde{\psi}_1$ is processed by the operator $\widetilde{\mathfrak{T}}(\cdot, \widetilde{U}_{\theta f})$

$$\tilde{\psi}_2 = \widetilde{\mathfrak{T}}(\tilde{\psi}_1, \widetilde{U}_{\theta f}) =$$

$$= \begin{pmatrix} \ominus_\theta (x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \tilde{u}_1) & y_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \theta \\ y_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \theta & \ominus_\theta (x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \tilde{u}_2) \end{pmatrix} =$$

$$= \begin{pmatrix} \ominus_\theta (x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \tilde{u}_1) & \theta \\ \theta & \ominus_\theta (x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \tilde{u}_2) \end{pmatrix},$$

where $\tilde{u}_1$ and $\tilde{u}_2$ stand for the matrices $\tilde{I}_\theta$ and $\tilde{X}_\theta$ with respect to the control matrices $\widetilde{U}_{\theta 1}$, $\widetilde{U}_{\theta 2}$, $\widetilde{U}_{\theta 3}$ and $\widetilde{U}_{\theta 4}$.

The state $\tilde{\psi}_2$ is processed by the reverse fuzzy Hadamard operator

$$\tilde{\psi}_3 = \widetilde{\mathfrak{T}}(\widetilde{\mathfrak{T}}(\tilde{\psi}_2, \widetilde{H}_\theta), \Sigma_2^{-1}(\lambda)\ ) =$$

$$= \begin{pmatrix} (x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \tilde{u}_1) \oslash_\vartheta (\lambda \oplus_\theta \lambda) & \theta \\ \theta & (x_\theta \otimes_\vartheta (\lambda \oplus_\theta \lambda) \otimes_\vartheta \tilde{u}_2) \oslash_\vartheta (\lambda \oplus_\theta \lambda) \end{pmatrix} =$$

$$= \begin{pmatrix} x_\theta \otimes_\vartheta \tilde{u}_1 & \theta \\ \theta & x_\theta \otimes_\vartheta \tilde{u}_2 \end{pmatrix},$$

Finally, if $x_\theta \otimes_\vartheta \tilde{u}_1 = x_\theta \otimes_\vartheta \tilde{u}_2$, then the function f is constant and if $x_\theta \otimes_\vartheta \tilde{u}_1 \neq x_\theta \otimes_\vartheta \tilde{u}_2$, then the function f is balanced.

Hence, similarly to the Deutsch-Jozsa algorithm, the problem of recognition of the constant and balanced function is solved by a single call of the operator associated with the function f.

It is clear that similar to the Deutsch-Jozsa algorithm, the suggested fuzzy algorithm is useless and does not solve any practical problem. Along with that, as the Deutsch-Jozsa algorithm demonstrates that effectiveness of quantum computations, the suggested fuzzy algorithm demonstrates that using the suggested operations the problem can be solved as effectively as by quantum algorithm. □

4. Discussion

In the paper we suggested a fuzzy system for implementing analog or hybrid computations. The units of information in the system are defined as -f-bits which are the matrices of definite form. Operations in the system are defined using fuzzy Tsetlin automata and are implemented by the operator with definite control matrices.

The considered control matrices are such that the resulting operator implements operations used in quantum computations. As a result, the system implements quantum computations by means of fuzzy logic that is illustrated by fuzzy implementation of the Deutsch-Jozsa algorithm.

The suggested operator with the presented control matrices is certainly not a unique possible implementation of the system. It was chosen to coincide with the gates used in quantum computations and to demonstrate an effectiveness of the suggested system and the use of fuzzy Tsetlin automata for analog computations.

Further research will concentrate on simulations of the suggested scheme and its implementation using standard microelectronic devices.

Author Contributions: Conceptualization, E.K. and A.R.; methodology, A.R.; software, E.K.; validation, E.K. and A.R.; formal analysis, E.K.; investigation, E.K.; writing—original draft preparation, E.K.; writing—review and editing, A.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments: The authors thank Prof. E. Bormashenko from Ariel University for fruitful discussions which inspired the work on this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Calculations for the proof of Theorem 1.

- Pauli operators:

$$I|z\rangle = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}, \quad X|z\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} z_2 \\ z_1 \end{pmatrix},$$

$$Y|z\rangle = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} -iz_2 \\ iz_1 \end{pmatrix}, \quad Z|z\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} z_1 \\ -z_2 \end{pmatrix};$$

- Fuzzy Pauli operators:

$$\widetilde{\mathfrak{T}}(x_\theta, \tilde{I}_\theta) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} \ominus_\theta \lambda & \lambda \\ \lambda & \ominus_\theta \lambda \end{pmatrix}\right) =$$

$$= \begin{pmatrix} \ominus_\theta \left(x_1 \otimes_\vartheta (\ominus_\theta \lambda)\right) & (\ominus_\theta x_1) \otimes_\vartheta \lambda \\ (\ominus_\theta x_2) \otimes_\vartheta \lambda & \ominus_\theta \left(x_2 \otimes_\vartheta (\ominus_\theta \lambda)\right) \end{pmatrix} = \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix},$$

$$\widetilde{\mathfrak{T}}(x_\theta, \tilde{X}_\theta) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} \lambda & \ominus_\theta \lambda \\ \ominus_\theta \lambda & \lambda \end{pmatrix}\right) =$$

$$= \begin{pmatrix} \ominus_\theta (x_1 \otimes_\vartheta \lambda) & (\ominus_\theta x_1) \otimes_\vartheta (\ominus_\theta \lambda) \\ (\ominus_\theta x_2) \otimes_\vartheta (\ominus_\theta \lambda) & \ominus_\theta (x_2 \otimes_\vartheta \lambda) \end{pmatrix} = \begin{pmatrix} \ominus_\theta x_1 & x_1 \\ x_2 & \ominus_\theta x_2 \end{pmatrix},$$

$$\widetilde{\mathfrak{T}}(x_\theta, \tilde{Y}_\theta) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} \lambda & \ominus_\theta \lambda \\ \lambda & \ominus_\theta \lambda \end{pmatrix}\right) = \begin{pmatrix} \ominus_\theta (x_1 \otimes_\vartheta \lambda) & (\ominus_\theta x_1) \otimes_\vartheta (\ominus_\theta \lambda) \\ (\ominus_\theta x_2) \otimes_\vartheta \lambda & \ominus_\theta \left(x_2 \otimes_\vartheta (\ominus_\theta \lambda)\right) \end{pmatrix} =$$

$$\begin{pmatrix} \ominus_\theta x_1 & x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix},$$

$$\widetilde{\mathfrak{T}}\left(x_\theta, \tilde{Z}_\theta\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} \ominus_\theta \lambda & \lambda \\ \ominus_\theta \lambda & \lambda \end{pmatrix}\right) =$$
$$\begin{pmatrix} \ominus_\theta \left(x_1 \otimes_\vartheta (\ominus_\theta \lambda)\right) & (\ominus_\theta x_1) \otimes_\vartheta \lambda \\ (\ominus_\theta x_2) \otimes_\vartheta (\ominus_\theta \lambda) & \ominus_\theta (x_2 \otimes_\vartheta \lambda) \end{pmatrix} = \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ x_2 & \ominus_\theta x_2 \end{pmatrix};$$

- Square root of not:

$$V|z\rangle = \frac{1}{2}\begin{pmatrix} 1+i & 1-i \\ 1-i & 1+i \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} z_1+z_2+i(z_1-z_2) \\ z_1+z_2+i(z_2-z_1) \end{pmatrix};$$

- Fuzzy root of not ($\nu \oplus_\theta \nu = \lambda$):

$$\widetilde{\mathfrak{T}}\left(x_\theta, \tilde{V}_\theta\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} \nu & \ominus_\theta \nu \\ \ominus_\theta \nu & \nu \end{pmatrix}\right) =$$
$$= \begin{pmatrix} \ominus_\theta (x_1 \otimes_\vartheta \nu) & (\ominus_\theta x_1) \otimes_\vartheta (\ominus_\theta \nu) \\ (\ominus_\theta x_2) \otimes_\vartheta (\ominus_\theta \nu) & \ominus_\theta (x_2 \otimes_\vartheta \nu) \end{pmatrix} = \begin{pmatrix} \ominus_\theta (x_1 \otimes_\vartheta \nu) & x_1 \otimes_\vartheta \nu \\ x_2 \otimes_\vartheta \nu & \ominus_\theta (x_2 \otimes_\vartheta \nu) \end{pmatrix};$$

- Hadamard operator:

$$H|z\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} z_1+z_2 \\ z_1-z_2 \end{pmatrix};$$

- Fuzzy Hadamard operator:

$$\widetilde{\mathfrak{T}}\left(x_\theta, \widetilde{H}_\theta\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} (\ominus_\theta \lambda) \oplus_\theta (\ominus_\theta \lambda) & \lambda \oplus_\theta \lambda \\ \lambda \oplus_\theta \lambda & (\ominus_\theta \lambda) \oplus_\theta (\ominus_\theta \lambda) \end{pmatrix}\right) =$$
$$= \begin{pmatrix} (\ominus_\theta x_1) \otimes_\vartheta (\lambda \oplus_\theta \lambda) & (\ominus_\theta x_1) \otimes_\vartheta (\lambda \oplus_\theta \lambda) \\ (\ominus_\theta x_2) \otimes_\vartheta (\lambda \oplus_\theta \lambda) & \ominus_\theta \left(x_2 \otimes_\vartheta \left((\ominus_\theta \lambda) \oplus_\theta (\ominus_\theta \lambda)\right)\right) \end{pmatrix} =$$
$$\begin{pmatrix} x_1 \otimes_\vartheta (\lambda \oplus_\theta \lambda) & (\ominus_\theta x_1) \otimes_\vartheta (\lambda \oplus_\theta \lambda) \\ (\ominus_\theta x_2) \otimes_\vartheta (\lambda \oplus_\theta \lambda) & x_1 \otimes_\vartheta (\lambda \oplus_\theta \lambda) \end{pmatrix};$$

- Phase shift operator:

$$S|z\rangle = \begin{pmatrix} 1 & 1 \\ 1 & i \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} z_1+z_2 \\ z_1+iz_2 \end{pmatrix};$$

- Fuzzy phase shift operator ($\nu \oplus_\theta \nu = \lambda$):

$$\widetilde{\mathfrak{T}}\left(x_\theta, \tilde{S}_\theta\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} \ominus_\theta \nu & \nu \\ \ominus_\theta \nu & \nu \end{pmatrix}\right) =$$
$$= \begin{pmatrix} \ominus_\theta \left(x_1 \otimes_\vartheta (\ominus_\theta \nu)\right) & (\ominus_\theta x_1) \otimes_\vartheta \nu \\ (\ominus_\theta x_2) \otimes_\vartheta (\ominus_\theta \nu) & \ominus_\theta (x_2 \otimes_\vartheta \nu) \end{pmatrix} = \begin{pmatrix} x_1 \otimes_\vartheta \nu & (\ominus_\theta x_1) \otimes_\vartheta \nu \\ x_2 \otimes_\vartheta \nu & (\ominus_\theta x_2) \otimes_\vartheta \nu \end{pmatrix};$$

- T-gate:

$$T|z\rangle = \begin{pmatrix} 1 & 1 \\ 1 & e^{i\pi/4} \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} z_1+z_2 \\ z_1+e^{i\pi/4}z_2 \end{pmatrix};$$

- Fuzzy T-gate ($\tau \oplus_\theta \tau = \nu$):

$$\widetilde{\mathfrak{T}}\left(x_\theta, \tilde{T}_\theta\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}, \begin{pmatrix} \ominus_\theta \tau & \tau \\ \ominus_\theta \tau & \tau \end{pmatrix}\right) =$$
$$= \begin{pmatrix} \ominus_\theta \left(x_1 \otimes_\vartheta (\ominus_\theta \tau)\right) & (\ominus_\theta x_1) \otimes_\vartheta \tau \\ (\ominus_\theta x_2) \otimes_\vartheta (\ominus_\theta \tau) & \ominus_\theta (x_2 \otimes_\vartheta \tau) \end{pmatrix} = \begin{pmatrix} x_1 \otimes_\vartheta \tau & \ominus_\theta (x_1 \otimes_\vartheta \tau) \\ x_2 \otimes_\vartheta \tau & \ominus_\theta (x_2 \otimes_\vartheta \tau) \end{pmatrix}.$$

- Two-qubit controlled NOT ($CNOT$) operator:

$$CNOT|zw\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} z_1 \\ z_2 \\ w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \\ w_2 \\ w_1 \end{pmatrix};$$

- Fuzzy two-f-bit controlled NOT ($CNOT$) operator:

$$\widetilde{\mathfrak{T}}\left(x_\theta, y_\theta, \widetilde{CNOT}_\theta\right) =$$

$$= \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 & \ominus_\theta x_1 & \theta & \theta \\ \ominus_\theta x_2 & x_2 & \theta & \theta \\ \theta & \theta & \theta & \theta \\ \theta & \theta & \theta & \theta \end{pmatrix}, \begin{pmatrix} \theta & \theta & \theta & \theta \\ \theta & \theta & \theta & \theta \\ \theta & \theta & y_1 & \ominus_\theta y_1 \\ \theta & \theta & \ominus_\theta y_2 & y_2 \end{pmatrix}, \begin{pmatrix} \ominus_\theta \lambda & \lambda & \theta & \theta \\ \lambda & \ominus_\theta \lambda & \theta & \theta \\ \theta & \theta & \lambda & \ominus_\theta \lambda \\ \theta & \theta & \ominus_\theta \lambda 1 & \lambda \end{pmatrix}\right) =$$

$$= \begin{pmatrix} \ominus_\theta \left(x_1 \otimes_\vartheta (\ominus_\theta \lambda)\right) & (\ominus_\theta x_1) \otimes_\vartheta \lambda & \theta & \theta \\ (\ominus_\theta x_2) \otimes_\vartheta \lambda & \ominus_\theta \left(x_2 \otimes_\vartheta (\ominus_\theta \lambda)\right) & \theta & \theta \\ \theta & \theta & \ominus_\theta (y_1 \otimes_\vartheta \lambda) & (\ominus_\theta y_1) \otimes_\vartheta (\ominus_\theta \lambda) \\ \theta & \theta & (\ominus_\theta y_2) \otimes_\vartheta (\ominus_\theta \lambda) & \ominus_\theta (y_2 \otimes_\vartheta \lambda) \end{pmatrix}$$

$$=$$

$$= \begin{pmatrix} x_1 & \ominus_\theta x_1 & \theta & \theta \\ \ominus_\theta x_2 & x_2 & \theta & \theta \\ \theta & \theta & \ominus_\theta y_1 & y_1 \\ \theta & \theta & y_2 & \ominus_\theta y_2 \end{pmatrix}.$$

Appendix B

Calculations for the proof of Lemma 3.

$$\tilde{I}_\theta \oplus_\theta \tilde{X}_\theta = \begin{pmatrix} \ominus_\theta \lambda & \lambda \\ \lambda & \ominus_\theta \lambda \end{pmatrix} \oplus_\theta \begin{pmatrix} \lambda & \ominus_\theta \lambda \\ \ominus_\theta \lambda & \lambda \end{pmatrix} = \begin{pmatrix} (\ominus_\theta \lambda) \oplus_\theta \lambda & \lambda \oplus_\theta (\ominus_\theta \lambda) \\ \lambda \oplus_\theta (\ominus_\theta \lambda) & (\ominus_\theta \lambda) \oplus_\theta \lambda \end{pmatrix} = \begin{pmatrix} \theta & \theta \\ \theta & \theta \end{pmatrix},$$

$$\tilde{Y}_\theta \oplus_\theta \tilde{Z}_\theta = \begin{pmatrix} \lambda & \ominus_\theta \lambda \\ \lambda & \ominus_\theta \lambda \end{pmatrix} \oplus_\theta \begin{pmatrix} \ominus_\theta \lambda & \lambda \\ \ominus_\theta \lambda & \lambda \end{pmatrix} = \begin{pmatrix} \lambda \oplus_\theta (\ominus_\theta \lambda) & (\ominus_\theta \lambda) \oplus_\theta \lambda \\ \lambda \oplus_\theta (\ominus_\theta \lambda) & (\ominus_\theta \lambda) \oplus_\theta \lambda \end{pmatrix} = \begin{pmatrix} \theta & \theta \\ \theta & \theta \end{pmatrix},$$

$$\tilde{I}_\theta \oplus_\theta \tilde{I}_\theta = \begin{pmatrix} \ominus_\theta \lambda & \lambda \\ \lambda & \ominus_\theta \lambda \end{pmatrix} \oplus_\theta \begin{pmatrix} \ominus_\theta \lambda & \lambda \\ \lambda & \ominus_\theta \lambda \end{pmatrix} = \begin{pmatrix} (\ominus_\theta \lambda) \oplus_\theta (\ominus_\theta \lambda) & \lambda \oplus_\theta \lambda \\ \lambda \oplus_\theta \lambda & (\ominus_\theta \lambda) \oplus_\theta (\ominus_\theta \lambda) \end{pmatrix} = \widetilde{H}_\theta,$$

$$\tilde{V}_\theta \oplus_\theta \tilde{V}_\theta = \begin{pmatrix} \nu & \ominus_\theta \nu \\ \ominus_\theta \nu & \nu \end{pmatrix} \oplus_\theta \begin{pmatrix} \nu & \ominus_\theta \nu \\ \ominus_\theta \nu & \nu \end{pmatrix} = \begin{pmatrix} \nu \oplus_\theta \nu & (\ominus_\theta \nu) \oplus_\theta (\ominus_\theta \nu) \\ (\ominus_\theta \nu) \oplus_\theta (\ominus_\theta \nu) & \nu \oplus_\theta \nu \end{pmatrix} =$$

$$= \begin{pmatrix} \lambda & \ominus_\theta \lambda \\ \ominus_\theta \lambda & \lambda \end{pmatrix} = \tilde{X}_\theta,$$

$$\tilde{T}_\theta \oplus_\theta \tilde{T}_\theta = \begin{pmatrix} \ominus_\theta \tau & \tau \\ \ominus_\theta \tau & \tau \end{pmatrix} \oplus_\theta \begin{pmatrix} \ominus_\theta \tau & \tau \\ \ominus_\theta \tau & \tau \end{pmatrix} = \begin{pmatrix} (\ominus_\theta \tau) \oplus_\theta (\ominus_\theta \tau) & \tau \oplus_\theta \tau \\ (\ominus_\theta \tau) \oplus_\theta (\ominus_\theta \tau) & \tau \oplus_\theta \tau \end{pmatrix} = \begin{pmatrix} \ominus_\theta \nu & \nu \\ \ominus_\theta \nu & \nu \end{pmatrix} = \tilde{S}_\theta,$$

$$\tilde{S}_\theta \oplus_\theta \tilde{S}_\theta = \begin{pmatrix} \ominus_\theta \nu & \nu \\ \ominus_\theta \nu & \nu \end{pmatrix} \oplus_\theta \begin{pmatrix} \ominus_\theta \nu & \nu \\ \ominus_\theta \nu & \nu \end{pmatrix} = \begin{pmatrix} (\ominus_\theta \nu) \oplus_\theta (\ominus_\theta \nu) & \nu \oplus_\theta \nu \\ (\ominus_\theta \nu) \oplus_\theta (\ominus_\theta \nu) & \nu \oplus_\theta \nu \end{pmatrix} = \begin{pmatrix} \ominus_\theta \lambda & \lambda \\ \ominus_\theta \lambda & \lambda \end{pmatrix} = \tilde{Z}_\theta,$$

Appendix C

Calculations for the proof of Lemma 4. Let

$$x_\theta = \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}$$

be an f-bit. Then

$$\widetilde{\mathfrak{T}}^2\left(x_\theta, \widetilde{H}_\theta\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 \otimes_\vartheta (\lambda \oplus_\theta \lambda) & \ominus_\theta \left(x_1 \otimes_\vartheta (\lambda \oplus_\theta \lambda)\right) \\ \ominus_\theta \left(x_2 \otimes_\vartheta (\lambda \oplus_\theta \lambda)\right) & x_2 \otimes_\vartheta (\lambda \oplus_\theta \lambda) \end{pmatrix}, \widetilde{H}_\theta\right) =$$

$$= \begin{pmatrix} x_1 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \ldots \oplus_\theta \lambda}_{4\ times}\right) & \ominus_\theta \left(x_1 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \ldots \oplus_\theta \lambda}_{4\ times}\right)\right) \\ \ominus_\theta \left(x_2 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \ldots \oplus_\theta \lambda}_{4\ times}\right)\right) & x_2 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \ldots \oplus_\theta \lambda}_{4\ times}\right) \end{pmatrix},$$

and

doi:10.20944/preprints202510.0856.v1

$$\widetilde{\mathfrak{T}}\left(\widetilde{\mathfrak{T}}^2\left(x_\theta, \widetilde{H}_\theta\right), \Sigma_2^{-1}(\lambda)\right) =$$

$$= \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right) & \ominus_\theta \left(x_1 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right)\right) \\ \ominus_\theta \left(x_2 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right)\right) & x_2 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right) \end{pmatrix}, \Sigma_2^{-1}(\lambda)\right) =$$

$$= \begin{pmatrix} x_1 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\lambda \otimes_\vartheta \dots \otimes_\vartheta \lambda}_{4\ times}\right) & \ominus_\theta \left(x_1 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\lambda \otimes_\vartheta \dots \otimes_\vartheta \lambda}_{4\ times}\right)\right) \\ \ominus_\theta \left(x_2 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\lambda \otimes_\vartheta \dots \otimes_\vartheta \lambda}_{4\ times}\right)\right) & x_2 \otimes_\vartheta \left(\underbrace{\lambda \oplus_\theta \dots \oplus_\theta \lambda}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\lambda \otimes_\vartheta \dots \otimes_\vartheta \lambda}_{4\ times}\right) \end{pmatrix} =$$

$$= \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix} = x_\theta.$$

Appendix D

Calculations for the proof of Lemma 4. Let

$$x_\theta = \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix}$$

be an f-bit. Then

$$\widetilde{\mathfrak{T}}^2\left(x_\theta, \tilde{V}_\theta\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} \ominus_\theta (x_1 \otimes_\vartheta \nu) & x_1 \otimes_\vartheta \nu \\ x_2 \otimes_\vartheta \nu & \ominus_\theta (x_2 \otimes_\vartheta \nu) \end{pmatrix}, \tilde{V}_\theta\right) =$$
$$\begin{pmatrix} x_1 \otimes_\vartheta \nu \otimes_\vartheta \nu & \ominus_\theta (x_1 \otimes_\vartheta \nu \otimes_\vartheta \nu) \\ \ominus_\theta (x_2 \otimes_\vartheta \nu \otimes_\vartheta \nu) & x_2 \otimes_\vartheta \nu \otimes_\vartheta \nu \end{pmatrix},$$

$$\widetilde{\mathfrak{T}}\left(\widetilde{\mathfrak{T}}^2\left(x_\theta, \tilde{V}_\theta\right), \Pi_2^{-1}(\nu)\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 \otimes_\vartheta \nu \otimes_\vartheta \nu & \ominus_\theta (x_1 \otimes_\vartheta \nu \otimes_\vartheta \nu) \\ \ominus_\theta (x_2 \otimes_\vartheta \nu \otimes_\vartheta \nu) & x_2 \otimes_\vartheta \nu \otimes_\vartheta \nu \end{pmatrix}, \Pi_2^{-1}(\nu)\right) =$$

$$= \begin{pmatrix} (x_1 \otimes_\vartheta \nu \otimes_\vartheta \nu) \oslash_\vartheta (\nu \otimes_\vartheta \nu) & \ominus_\theta \left((x_1 \otimes_\vartheta \nu \otimes_\vartheta \nu) \oslash_\vartheta (\nu \otimes_\vartheta \nu)\right) \\ \ominus_\theta \left((x_2 \otimes_\vartheta \nu \otimes_\vartheta \nu) \oslash_\vartheta (\nu \otimes_\vartheta \nu)\right) & \left((x_2 \otimes_\vartheta \nu \otimes_\vartheta \nu) \oslash_\vartheta (\nu \otimes_\vartheta \nu)\right) \end{pmatrix} =$$

$$= \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix} = x_\theta.$$

and

$$\widetilde{\mathfrak{T}}\left(\widetilde{\mathfrak{T}}^4\left(x_\theta, \tilde{T}_\theta\right), \Pi_4^{-1}(\tau)\right) = \widetilde{\mathfrak{T}}\left(\begin{pmatrix} x_1 \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau & \ominus_\theta (x_1 \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau) \\ x_2 \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau & \ominus_\theta (x_2 \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau \otimes_\vartheta \tau) \end{pmatrix}, \Pi_4^{-1}(\tau)\right) =$$

$$= \begin{pmatrix} \left(x_1 \otimes_\vartheta \underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right) & \ominus_\theta \left(\left(x_1 \otimes_\vartheta \underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right)\right) \\ \ominus_\theta \left(\left(x_2 \otimes_\vartheta \underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right)\right) & \left(x_2 \otimes_\vartheta \underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right) \oslash_\vartheta \left(\underbrace{\tau \otimes_\vartheta \dots \otimes_\vartheta \tau}_{4\ times}\right) \end{pmatrix}$$

$$= \begin{pmatrix} x_1 & \ominus_\theta x_1 \\ \ominus_\theta x_2 & x_2 \end{pmatrix} = x_\theta.$$

Appendix E

Before formulation of the Deutsch-Jozsa algorithm [13], recall that the function is called constant if for any argument it returns a certain predefined value, and the function is called balanced if a half of its outputs are zeros and a half are units.

Consider four Boolean functions $f_k\colon\{0,1\}\to\{0,1\}$, $k=1,\dots,4$, of one argument. Then, the functions $f_1(x)=0$ and $f_2(x)=1$ are constant and the functions $f_3(x)=x$ and $f_4(x)=\text{not}\,x$ are balanced.

Assume that there is an unknown function $f\colon\{0,1\}\to\{0,1\}$ and it is required to recognize whether the function f is constant or balanced.

The Deutsch-Jozsa algorithm acts as follows. As an input, the algorithm obtains two qubits: qubit $\mathbb{O}=(1,0)$ associated with the Boolean value 0 and qubit $\mathbb{I}=(0,1)$ associated with the Boolean value 1. These vectors are superposed into the two-qubit state

$$\psi_0=\mathbb{O}\otimes\mathbb{I}=(0,1,0,0),$$

where $\otimes$ stands for the Kronecker product.

To the state ψ_0 the two-qubit operator $H\otimes H$ is applied

$$\psi_1=\psi_0(H\otimes H)=\tfrac{1}{2}(0,1,0,0)\begin{pmatrix}H & H\\ H & -H\end{pmatrix}=\tfrac{1}{2}(1,-1,1,-1).$$

Now the state ψ_1 is processed by the oracle operator U_f which is a two-qubit operator representing the function f. Then

$$\psi_2=\psi_1U_f=\tfrac{1}{2}(1,-1,1,-1)U_f.$$

Operator U_f represents the functions f_1, f_2, f_3 and f_4. These functions are associated with the two-qubit operators:

$$U_1=\begin{pmatrix}I & 0\\ 0 & I\end{pmatrix},\quad U_2=\begin{pmatrix}X & 0\\ 0 & X\end{pmatrix},\quad U_3=\begin{pmatrix}I & 0\\ 0 & X\end{pmatrix}\quad\text{and}\quad U_4=\begin{pmatrix}X & 0\\ 0 & I\end{pmatrix}.$$

respectively.

To the state ψ_2 is processed by the qubit operator $H\otimes I$ which gives

$$\psi_3=\psi_2(H\otimes I)=\psi_2\tfrac{1}{\sqrt{2}}\begin{pmatrix}I & I\\ I & -I\end{pmatrix}.$$

Finally, in the resulting two-qubit state ψ_3, if the first qubit $\psi_{31}\neq(0,0)$, then f is constant and if the first qubit $\psi_{31}=(0,0)$, then f is balanced.

Let us consider the operator U_f defined by the operators U_1, U_2, U_3 and U_4:

$U_f=U_1$:

$$\psi_2=\tfrac{1}{2}(1,-1,1,-1)\begin{pmatrix}I & 0\\ 0 & I\end{pmatrix}=\tfrac{1}{2}(1,-1,1,-1),$$

$$\psi_3=\tfrac{1}{2}(1,-1,1,-1)\tfrac{1}{\sqrt{2}}\begin{pmatrix}I & I\\ I & -I\end{pmatrix}=\tfrac{1}{\sqrt{2}}(1,-1,0,0).$$

$U_f=U_2$:

$$\psi_2=\tfrac{1}{2}(1,-1,1,-1)\tfrac{1}{\sqrt{2}}\begin{pmatrix}X & 0\\ 0 & X\end{pmatrix}=\tfrac{1}{2}(-1,1,-1,1),$$

$$\psi_3=\tfrac{1}{2}(-1,1,-1,1)\tfrac{1}{\sqrt{2}}\begin{pmatrix}I & I\\ I & -I\end{pmatrix}=\tfrac{1}{\sqrt{2}}(-1,1,0,0).$$

$U_f=U_3$:

$$\psi_2=\tfrac{1}{2}(1,-1,1,-1)\tfrac{1}{\sqrt{2}}\begin{pmatrix}I & 0\\ 0 & X\end{pmatrix}=\tfrac{1}{2}(1,-1,-1,1),$$

$$\psi_3=\tfrac{1}{2}(1,-1,1,-1)\tfrac{1}{\sqrt{2}}\begin{pmatrix}I & I\\ I & -I\end{pmatrix}=-\tfrac{1}{\sqrt{2}}(0,0,1,-1).$$

$U_f=U_4$:

$$\psi_2=\tfrac{1}{2}(1,-1,1,-1)\tfrac{1}{\sqrt{2}}\begin{pmatrix}X & 0\\ 0 & I\end{pmatrix}=\tfrac{1}{2}(-1,1,1,-1),$$

$$\psi_3=\tfrac{1}{2}(1,-1,1,-1)\tfrac{1}{\sqrt{2}}\begin{pmatrix}I & I\\ I & -I\end{pmatrix}=\tfrac{1}{\sqrt{2}}(0,0,-1,1).$$

It is seen that for $U_f = U_1$ the first qubit of ψ_3 is $\psi_{31} = \frac{1}{\sqrt{2}}(1,-1)$ and for $U_f = U_2$ it is $\psi_{31} = \frac{1}{\sqrt{2}}(1,-1)$; hence if $f = f_1$ or $f = f_2$ it is constant. In contrast, for $U_f = U_3$ the first qubit of ψ_3 is $\psi_{31} = (0,0)$ and for $U_f = U_4$ it is $\psi_{31} = (0,0)$; hence if $f = f_3$ or $f = f_3$ it is balanced.

References

1. Rosenbloom, P. Computing and computation. *The Computer Journal* 2012, *55*(*7*), 820-824.
2. MacLennan, B. Analog computation. In *Computational Complexity: Theory, Techniques and Applications*; Meyers, R. Ed.; Springer Science + Business Media: New York, NY, USA, 2012, pp. 161-184.
3. Minsky, M. *Finite and Infinite Machines*. Prentice Hall: Englewood Cliffs, NJ, USA, 1967.
4. Rummer, D. *Introduction to Analog Computer Programming*. Holt, Rinehart and Winston, New York, NY, USA, 1969.
5. McCracken, D. *Digital Computer Programming*. John Wiley & Sons, New York, NY, USA, 1957.
6. Feynman, R. Simulating physics with computers. *Int. Journal of Theoretical Physics*, 1982, *21*(*6/7*), 467-488.
7. Feynman, R. Quantum mechanical computers. *Optic News*, 1985, *11*, 11-20. (Reprint: *Foundations of Physics*, 1986, *16*(*6*), 507-531).
8. Birkhoff, G.; Neumann J. von. The logic of quantum mechanics. *Annals of Mathematics*, 1936, *37*(*4*), 823-843.
9. Lloyd, S. Quantum information processing. In *Computational Complexity: Theory, Techniques and Applications*; Meyers, R. Ed.; Springer Science + Business Media: Ney York, NY, USA, 2012, pp. 2496-2534.
10. Pykacz, J. Fuzzy set ideas in quantum logics. *Int. Journal of Theoretical Physics*, 1992, *31*(*9*), 281-290.
11. Hannachi, S.; Dong, F., Hirota, K. Emulating quantum interference and quantum associative memory using fuzzy qubits. In *Proceedings of IEEE Int. Conference on Computational Cybernetics*, Gammarth, Tunisia, 19-21 October 2007.
12. Hannachi, S.; Hatakeyama, Y., Hirota, K. Emulating qubits with fuzzy logic. *J. Advanced Computational Intelligence and Intelligent Informatics*, 2007, *11*(*2*), 242-249.
13. Deutsch, D., Jozsa, R. Rapid solution of problems by quantum computation. Proc. Royal Society: Mathematical and Physical Sciences, 1992, 439 (1907), 553-558.
14. Rybalov, A.; Kagan, E.; Manor, Y.; Ben-Gal, I. Fuzzy model of control for quantum-controlled mobile robots. In *Proceedings of IEEE 26-th Convention of Electrical and Electronics Engineers in Israel*, Eilat, Israel, 17-20 November 2010.
15. Kagan, E.; Ben-Gal, I. Navigation of quantum-controlled mobile robots. In *Recent Advances in Mobile Robotics*; Topalov, A. Ed.; Intech: Rijeka, Croatia, 2011, 311-326.
16. Kreinovich, V.; Kohout, L.; Kim, E. Square root of "not": a major difference between fuzzy and quantum logics. In *Proceedings of Annual Meeting of the North American Fuzzy Information Processing Society NAFIPS 2008*, New York, NY, USA, 19-22 May 2008.
17. Kosheleva, O.; Kreinovich, V. A Natural formalization of changing-one's-mind leads to square root of "not" and to complex-valued fuzzy logic. In *Explainable AI and Other Applications of Fuzzy Techniques*. *NAFIPS 2021. Lecture Notes in Networks and Systems*; Rayz, J.; Raskin, V.; Dick, S., Kreinovich, V. Eds., *258*, Springer: Cham, Switzerland, 2021, 190-195.
18. Ramot, D.; Milo, R.; Friedman, G.; Kandel, A. Complex fuzzy sets. *IEEE Transactions on Fuzzy Systems*, 2002, *10*(*2*), 171-186.
19. Ramot, D.; Friedman, M.; Langholz, G.; Kandel, A. Complex fuzzy logic. *IEEE Transactions on Fuzzy Systems*, 2003, *11*(*4*), 450-461.
20. Nielsen, M. A., Chuang, I. L. *Quantum Computation and Quantum Information*. Cambridge University Press: Cambridge, UK, 2000.
21. Tsetlin, M. *Automaton Theory and Modeling of Biological Systems*. Academic Press: New York, 1973.
22. Yager, R.; Rybalov, A. Uninorm aggregation operators. *Fuzzy Sets and Systems*, 1996, *80*, 111-120.
23. Dombi, J. Basic concepts for the theory of evaluation: the aggregative operator. *European J. of Operations Research*, 1982, *10*, 282–293.
24. Rudas, I. New approach to information aggregation. *Zbornik Radova*, 2000, *2*, 163-176.

25. Fodor, J., Rudas, I., Bede, B. Uninorms and absorbing norms with applications to image processing. In *Proceedings of the 4th Serbian Hungarian Joint Symposium on Intelligent Systems*, Subotica, Serbia, 29–30 September 2006, 59–72.
26. Kagan, E., Rybalov, A., Yager, R. *Multi-Valued Logic for Decision-Making Under Uncertainty*. Springer Nature / Birkhäuser: Cham, Switzerland, 2025.
27. Kagan, E., Rybalov, A., Yager, R. Sum of certainties with the product of reasons: neural network with fuzzy aggregators. *Int. J. of Uncertainty, Fuzziness and Knowledge Based Systems*, 2022, *30*(*1*), 1–18.