Avances en Ciencias e Ingenierías
ISSN: 1390-5384
ISSN: 2528-7788
ISSN-L: 1390-5384
avances@usfq.edu.ec
Universidad San Francisco de Quito
Ecuador

Bañuelos Aquino, José David; Orizaga Trejo, José Antonio
Study of syllogism in the paradigm of quantum logic
Avances en Ciencias e Ingenierías, vol. 17, núm. 1, 2025, Enero, pp. 1-16
Universidad San Francisco de Quito
Ecuador

Disponible en: https://www.redalyc.org/articulo.oa?id=726182578010

Cómo citar el artículo
Número completo
Más información del artículo
Página de la revista en redalyc.org

redalyc.org
Sistema de Información Científica Redalyc
Red de revistas científicas de Acceso Abierto diamante
Infraestructura abierta no comercial propiedad de la academia

aci avances en ciencias e ingenierías

Artículo/Article
Vol. 17, nro. 1
e3709

Study of syllogism in the paradigm of quantum logic

José David Bañuelos Aquino[1]*, José Antonio Orizaga Trejo[1]

[1]Universidad de Guadalajara, CUCEA, Zapopan, México, ORCID

*Autor para correspondencia/Corresponding autho david.banuelos@q-team.mx

Estudio del silogismo en el paradigma de la lógica cuántica

Abstract

The present document aims to lay the foundations for an inferential logical reasoning system in the paradigm of quantum computing by using hypothetical syllogisms in its different figures. Using the implication gate in the form Jauch, a quantum algorithm was developed that performs the operation analogous to the hypothetical syllogism within the paradigm of quantum logic. Depending on the order of its premises, it can be treated as an inductive, deductive, or abductive syllogism. The results of the execution of this algorithm were obtained using the quantum circuit simulator Q-team.

Keywords: Quantum information theory, quantum computation, quantum implication operation, quantum syllogism, Q-Team.

Resumen

El presente documento tiene como objetivo sentar las bases para un sistema de razonamiento lógico inferencial en el paradigma de la computación cuántica mediante el uso de silogismos hipotéticos en sus diferentes figuras. Utilizando la compuerta de implicación en la forma Jauch, se desarrolló un algoritmo cuántico que realiza la operación análoga al silogismo hipotético dentro del paradigma de la lógica cuántica. Dependiendo del orden de sus premisas puede ser tratado como un silogismo inductivo, deductivo o abductivo según sea el caso. Los resultados de la ejecución de este algoritmo se obtuvieron utilizando el simulador de circuitos cuánticos Q-Team.

Palabras clave: Teoría de la información cuántica, computación cuántica, operación de implicación cuántica, silogismo cuántico, Q-Team.

OPEN ACCESS

Licencia Creative Commons Atribución-NoComercial 4.0

Editado por / Edited by: Dennis Cazar

Recibido / Received: 28/11/ 2024

Aceptado / Accepted: 09/12/2024

Publicado en línea / Published online: 14/05/2025

INTRODUCTION

With the rise of technologies in quantum computing, as well as artificial intelligence, we find an area of opportunity within the development of intersectional systems between these two fields. However, although it is possible to note certain parallels between the ways of developing algorithms using logical operations, we find that there is no one-to-one analogy between the different operations that exist in both paradigms. Therefore, it is necessary to define the bases that allow us to develop quantum algorithms from traditional definitions that facilitate the creation of intelligent systems.

 DOI: https://doi.org/10.18272/aci.v17i1.3709

In this paper we show the development of an inferential reasoning algorithm based on syllogisms, using a definition of prepositional logic approached from the paradigm of quantum logic, which was tested using the quantum algorithms simulator Q-Team.

REASONING

What is reasoning?

Reasoning is the mental process by which the relationship between two premises, ideas, concepts, facts, or data is established to generate information and create a conclusion; the conclusion of this reasoning can be true or false, and depending on the criterion applied to it, it is valid or not. Reasoning allows you to analyze and form your own criteria, expanding the knowledge acquired without resorting to experience. Likewise, it represents the middle point between instinct and thought.

Depending on the type of mechanisms employed during this process, the reasoning can be divided into:

Logical reasoning:

In logical reasoning, the conclusion is implicit in the premises that compose it. Likewise, depending on the form or structure of the premises, different forms of logical reasoning can be found.

- **Deductive:** The conclusion is obtained from the premises. Example: "All human beings are mortal. Socrates is a human being. Therefore, Socrates is mortal."
- **Inductive:** The conclusion is obtained from the premises, but not necessarily. Example: "The sun has risen every day of history. Therefore, it is likely to come out tomorrow."
- **Abductive:** The most plausible explanation for a phenomenon is sought. Example: "There are footprints of mud on the ground. Therefore, it is likely that it rained."

Non-logical reasoning:

In non-logical reasoning, the conclusion is not necessarily implicit in the premises, so it is necessary to resort to context and/or experience to obtain a conclusion. [1, 2]

The syllogism

It is an inferential reasoning method that uses two premises to generate a conclusion, as can be seen in the following example:

Major premise:	All humans are mortal
Minor premise:	Frank is human
Conclusion:	Frank is mortal

These premises are composed of three terms (major, middle, and minor) whose order and presence within the premises allow us to reach a conclusion that consists of a third premise in which the middle term will not appear. Below, the distribution of the terms in the previous example is shown.

Major term	All humans are mortal
Middle term	Frank is human
Minor term	Frank is mortal

Both premises and terms are subject to rules that are described below:

Rules of premises:

1. The major premise contains a major term and a middle term.
2. The minor premise contains a minor term and a middle term.
3. The conclusion contains the minor term and the major term in that order.

Rules of terms:

The syllogism consists of 3 terms: Major, Middle, and Minor

1. No term can be more extensive in conclusion than in premises.
2. The middle term should not appear in the conclusion.
3. The middle term should appear in the major and minor premise.

The relationship between the different terms within the premises is established using judgments based on prepositions with which it is possible to evaluate the validity of the conclusions generated. These mechanisms are described by the operators of the prepositional logic that will be addressed in detail in the next section. These mechanisms must adhere to the following rules.

Rules of prepositions:

Two negative premises do not give a conclusion.

1. Two affirmative premises do not give a negative conclusion.
2. Two particular premises do not give a conclusion.
3. Conclusion always is next to the weaker party.

Through these we build the different figures of the syllogism (deductive, inductive, and abductive), and we find four different forms as can be seen in the following tables: [1, 2]

TABLE 1. Deductive Syllogism.

Figure 1	Figure 2	Figure 3	Figure 4
$m \to T$	$T \to m$	$m \to T$	$T \to m$
$t \to m$	$t \to m$	$m \to t$	$m \to t$
$t \to T$	$t \to T$	$t \to T$	$t \to T$

TABLE 2. Inductive Syllogism.

Figure 1	Figure 2	Figure 3	Figure 4
$m \to t$	$m \to T$	$T \to m$	$t \to m$
$m \to T$	$m \to t$	$m \to t$	$m \to T$
$t \to T$	$t \to T$	$t \to T$	$t \to T$

TABLE 3. Abductive Syllogism.

Figure 1	Figure 2	Figure 3	Figure 4
$T \to m$	$m \to T$	$t \to m$	$m \to t$
$t \to m$	$t \to m$	$T \to m$	$T \to m$
$t \to T$	$t \to T$	$t \to T$	$t \to T$

Prepositional logic

Prepositional logic is a formal system that allows us to perform operations on statements or sets of data using logical operators that allow us to obtain the degree of veracity of these, as shown in the following table:

TABLE 4. Basic operations of prepositional logic.

Operator	Notation	General Application	Truth Table	
Negation	¬	A = Socrates is mortal. ¬A = Socrates is not mortal.	**A**	**¬A**
			V	F
			F	V
Conjunction	Λ	A = Socrates is mortal. B= Zeus is immortal. AΛB = Sócrates is mortal, and Zeus is immortal.	**AB**	**AΛB**
			FF	F
			FV	F
			VF	F
			VV	V

 DOI: https://doi.org/10.18272/aci.v17i1.3709

aci avances en ciencias e ingenierías

Operator	Notation	General Application	Truth Table	
Disjunction	V	A = Socrates is mortal. B= Zeus is immortal. AVB = Sócrates is mortal, or Zeus is immortal.	**AB**	**AVB**
			FF	F
			FV	V
			VF	V
			VV	V
Implication	→	A= Today is Monday. B=Tomorrow is Tuesday. *A→B= If today is Monday, then tomorrow is Tuesday.*	**AB**	**A→B**
			FF	V
			FV	V
			VF	F
			VV	V
Biconditional	↔	A= Today is Monday. B= Tomorrow is Tuesday. *A↔B= Today is Monday if and only if tomorrow is Tuesday.*	**AB**	**A↔B**
			FF	V
			FV	F
			VF	F
			VV	V

Likewise, with these operators we can build more complex definitions, which can be used to evaluate the type of reasoning we are working with, provided that the rules defined for it (prepositions, premises, and terms) are met. [1, 2]

The implication

As was described in the previous section, the implication is a prepositional operator that allows us to establish the cause-effect relationship (if, then) of two premises. This in turn can be defined in different ways within prepositional logic. However, any definition can be reduced in the following fundamental ways:

Jauch: $A \to B = \neg A \vee B$ (1)
Sasaki Hook: $A \to B = \neg A \vee (A \wedge B)$ (2)
Dishkant: $A \to B = B \vee (\neg B \wedge \neg A)$ (3)
Kalmbach: $A \to B = (\neg A \wedge B) \vee (\neg A \wedge \neg B) \vee (A \wedge (\neg A \vee B))$ (4)
Non-Tollens: $A \to B = (B \wedge \neg A) \vee (B \wedge A) \vee (\neg B \wedge (B \vee \neg A))$ (5)
Relevance: $A \to B = (A \wedge B) \vee (\neg A \wedge B) \vee (\neg A \wedge \neg B)$ (6)

With the conjunction, negation, and disjunction operators, we can develop any operation using these definitions. For the present work we will focus on the implication using the Jauch form. [2, 3, 4, 5]

Combinational logic circuits

Combinational logic circuits are a set of logical gates that execute the logical operators corresponding to negation, union, and disjunction. These allow complex operations to be performed using the values zero and one, true and false, or another binary system.

 DOI: https://doi.org/10.18272/aci.v17i1.3709

TABLE 5. Basic operations of combinational logic circuits.

Operator	Notation	Truth Table	
Negation		**A**	**¬A**
		1	0
		0	1
Conjunction		**AB**	**A∧B**
		00	0
		01	0
		10	0
		11	1
Disjunction		**AB**	**AVB**
		00	0
		01	1
		10	1
		11	1

These have technical applications within electronics and computing and are currently the basis of the operations on microprocessors.

With this we can define the implication in its Jauch form as the following combinational circuit:

FIGURE 1. Combinational circuit of the implication in its Jauch form.

Quantum logic

Quantum logic is a formal system composed of operators that allow us to perform experiments using the principles of quantum mechanics used in quantum information theory that are presented below:

- **Quantum superposition.** This describes how a physical system such as a particle is in all its possible states simultaneously if the system is totally isolated and there is no observer to measure it. Depending on the existence or absence of an observer, the states can have the following properties:
    - **Coherence:** A quantum system is coherent when there is no observer to measure the system. If the system is coherent, it will be in all its states simultaneously.

- **Decoherence:** A quantum system is coherent or loses coherence when an observer measures it, and it acquires only one of its possible states.

- **Quantum entanglement.** This describes how two particles that are generated in the same event maintain the same state. If one changes, the other will change simultaneously, acquiring the same state as the other without the need of any physical connection and regardless of the distance between them. [6, 7, 8, 9, 10]

With these principles, a new computational system was proposed that could surpass the capabilities of classical computing by taking advantage of these properties.

Dirac notation

Also called Bra-Ket notation because of the name of its elements, it denotes the quantum states of a system.

- **Bra:** represents a dual state which is obtained with the conjugate of the ket.

$$<\Psi| \tag{7}$$

- **Ket:** represents the state of the tax system which corresponds to a column vector. Its notation is as follows.

$$|\Psi> \tag{8}$$

The operators are denoted by a representative letter with some emphasis:

$$\hat{U} \tag{9}$$

Therefore, an operation on a base state is described by the operator:

$$\hat{U}\,|\Psi> \tag{10}$$

The projection of a state onto its own dual is defined as follows, where the result will always be equal to 1:

$$<\Psi|\Psi>=1 \tag{11}$$

On the other hand, if the projection of a state is made on the dual of a different one, the result will always be 0:

$$<\Omega|\Psi>=0 \tag{12}$$

The space of states

Associated with any isolated physical system, there is a complex vector space with a defined internal product (Hilbert) or space of states of the system which are defined by its state vector. This vector is formalized in the space of the system. We find two types of states with which we define the physical system:

 DOI: https://doi.org/10.18272/aci.v17i1.3709

- **Basis state:** The states in which we can find the system. These have a vector representation.
- **General state:** The amplitude of probability to find the system in a base state.

Using Dirac notation we can describe fundamental states as follows:

$$|\Psi\rangle = \alpha|\uparrow\rangle + \beta|\downarrow\rangle \tag{13}$$

Where:

Ψ= System status.
α and β= Complex numbers.
$\uparrow$ and $\downarrow$= Basis states.

The above example shows the notation for a two-state system. The notation of the basis states can be defined as appropriate depending on their number.

The formalized vector in our Hilbert space is defined as follows:

$$|\Psi\rangle = \begin{pmatrix}\alpha \\ \beta\end{pmatrix} \tag{14}$$

The qubit

The qubit is the fundamental unit of quantum information. It corresponds analogously to the bit used in classical computing. However, while the bit corresponds to electrical impulses that are interpreted in a binary way as 1 or 0, the qubit corresponds to a physical system of two basis states and has a representation similar to that of classical computing. It is denoted as:

$$|1\rangle = \begin{pmatrix}1 \\ 0\end{pmatrix};\ |0\rangle = \begin{pmatrix}0 \\ 1\end{pmatrix}$$
$$\hat{H}|\Psi\rangle = (1/\sqrt{2})\ |1\rangle + (1/\sqrt{2})\ |0\rangle \tag{15}$$

As qubits are entangled, their vector representation will grow to a ratio where n is the number of qubits. They are written by obtaining the tensor product of the qubits involved.

It is graphically represented as a sphere-shaped Hilbert space known as the Bloch sphere, and its state will determine the value of the qubit.

 DOI: https://doi.org/10.18272/aci.v17i1.3709

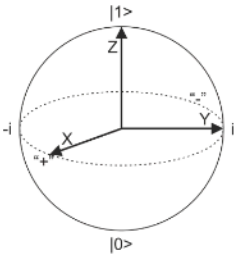


FIGURE 2. Axis on the Bloch sphere.

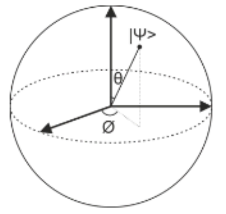


FIGURE 3. State in a Bloch sphere.

Quantum gates

Quantum gates allow modifying the state of one or more qubits in a way analogous to logic gates. The following table shows the fundamental gates:

TABLE 6. Fundamental quantum gates.

<table>
<tr><th>Operator</th><th>Notation</th><th>General Application</th><th colspan="2">Truth Table</th></tr>
<tr><td rowspan="3">Pauli-X</td><td rowspan="3">X</td><td rowspan="3">$\hat{X} \doteq \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} b \\ a \end{pmatrix}$</td><td>|Q></td><td>X|Q></td></tr>
<tr><td>|0></td><td>|1></td></tr>
<tr><td>|1></td><td>|0></td></tr>
<tr><td rowspan="3">Pauli-Y</td><td rowspan="3">Y</td><td rowspan="3">$\hat{Y} \doteq \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} -ib \\ ia \end{pmatrix}$</td><td>|Q></td><td>Y|Q></td></tr>
<tr><td>|0></td><td>I|1></td></tr>
<tr><td>|1></td><td>I|0></td></tr>
<tr><td rowspan="3">Pauli-Z</td><td rowspan="3">Z</td><td rowspan="3">$\hat{Z} \doteq \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} a \\ -b \end{pmatrix}$</td><td>|Q></td><td>Z|Q></td></tr>
<tr><td>|0></td><td>|0></td></tr>
<tr><td>|1></td><td>|-1></td></tr>
<tr><td rowspan="3">Phase change</td><td rowspan="3">P</td><td rowspan="3">$\hat{P} \doteq \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} a \\ e^{i\theta} b \end{pmatrix}$</td><td>|Q></td><td>P|Q></td></tr>
<tr><td>|0></td><td>|0></td></tr>
<tr><td>|1></td><td>|$e^{i\theta}$></td></tr>
<tr><td rowspan="3">Hadamard</td><td rowspan="3">H</td><td rowspan="3">$\hat{H} \doteq \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} a+b \\ a-b \end{pmatrix}$</td><td>|Q></td><td>H|Q></td></tr>
<tr><td>|0></td><td>(|0>+|1>)/√2</td></tr>
<tr><td>|1></td><td>(|0>−|1>)/√2</td></tr>
<tr><td rowspan="5">Controlled gate</td><td rowspan="5"></td><td rowspan="5">$\widehat{CU} \doteq \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} a \\ b \\ d \\ c \end{pmatrix}$</td><td>|Q0Q1></td><td>CX| Q0Q1></td></tr>
<tr><td>|00></td><td>|00></td></tr>
<tr><td>|01></td><td>|01></td></tr>
<tr><td>|10></td><td>|11></td></tr>
<tr><td>|11></td><td>|10></td></tr>
</table>

DOI: https://doi.org/10.18272/aci.v17i1.3709

Quantum algorithms

Quantum algorithms are analogous to a classical logic circuit; a quantum algorithm is a circuit that is composed of a set of quantum gates that operate on one or more qubits in one or more scenarios. [6]

Qubit in a quantum algorithm

These are represented as horizontal lines in which we can add different quantum gates that will allow us to modify the state of the qubit. Depending on whether we use a controlled quantum gate or not, we can find the next kinds of qubits:

- Control qubits: depending on whether the qubit or qubits are in the basis state |1>, the target qubits will perform the operation designated by the quantum gate being controlled.

- Target qubits: These are used to represent the output of gate operations, and it is in these that the operation will be performed if and only if the control qubit(s) are in the basis state |1> that require control. These are usually initialized in |0>. [6]

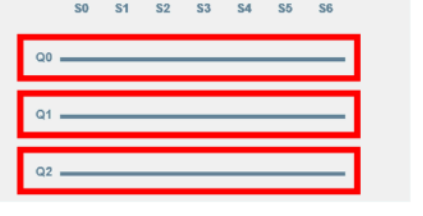


FIGURE 4. Qubits in a quantum algorithm.

Scenarios

Scenarios are represented as vertical lines that intersect the qubits and correspond to the instants of time in which we can add a gate on a qubit. [6]

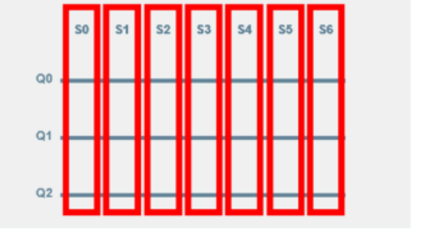


FIGURE 5. Scenarios in a quantum algorithm.

 DOI: https://doi.org/10.18272/aci.v17i1.3709

Terminals

Terminals indicate what will be done in the last scenario of each qubit. There are two cases. [6]

Measurement: indicates that the measurement will be made in this qubit.

FIGURE 6. Measurement in a quantum algorithm.

- **Terminal:** indicates that this qubit will not have continuity in the circuit.

FIGURE 7. Terminal in a quantum algorithm.

Quantum simulators

Quantum simulators are represented as vertical lines that intersect the qubits and correspond to the instants of time in which we can add a gate on a qubit. [1]

Q-Team

Q-Team is a quantum circuit simulator developed as part of the master's thesis of the author of this project at the University of Guadalajara. It has a graphical interface that allows the circuits to be developed without the need to write a single line of code. Likewise, a programming framework can be used for users who feel more comfortable in a development environment. It also has three methods of operation that allow obtaining results from the generated circuits through the manual entry of the inputs, by a list of inputs, or using a module that obtains all possible input states. [6]

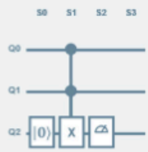
Quantum algorithms from classical definitions

Although in both quantum logic and prepositional logic we find that operators are used to modify the state or the information that we have at the beginning, there is no one-to-one analogy between these operators with the exception of negation and the Pauli X gate. However, quantum circuits can be built to perform analogous operations to those found in prepositional logic, as long as the following rules are fulfilled.

- The circuit must have the same number of inputs as the number of inputs in the logic circuit, not counting the objective qubits.
- The measurement of results must be made on the same number of qubits as the outputs of the combinational logic circuit.
- The qubits to be observed must be identified.
- Target qubits will always be initialized in the |0> basis state.
- There should be a one-to-one correspondence between the base states |0> and 1>, and binary states 0 and 1, respectively, both in the inputs and outputs.

DOI: https://doi.org/10.18272/aci.v17i1.3709

TABLE 7. Basic operations of combinational logic circuits.

<table>
<tr><th>Operator</th><th>Notation</th><th colspan="2">Truth Table</th><th>Analogous quantum circuit</th><th colspan="2">Truth Table</th></tr>
<tr><td rowspan="3">Negation</td><td rowspan="3"></td><td>A</td><td>¬A</td><td rowspan="3"></td><td>|Q></td><td>X|Q></td></tr>
<tr><td>1</td><td>0</td><td>|0></td><td>|1></td></tr>
<tr><td>0</td><td>1</td><td>|1></td><td>|0></td></tr>
<tr><td rowspan="5">Conjunction</td><td rowspan="5"></td><td>AB</td><td>A∧B</td><td rowspan="5"></td><td>|Q0Q1Q2></td><td>∧|Q0Q1Q2></td></tr>
<tr><td>00</td><td>0</td><td>|000></td><td>|000></td></tr>
<tr><td>01</td><td>0</td><td>|010></td><td>|010></td></tr>
<tr><td>10</td><td>0</td><td>|100></td><td>|100></td></tr>
<tr><td>11</td><td>1</td><td>|110></td><td>|111></td></tr>
<tr><td rowspan="5">Disjunction</td><td rowspan="5"></td><td>AB</td><td>AVB</td><td rowspan="5"></td><td>|Q0Q1Q2></td><td>V|Q0Q1Q2></td></tr>
<tr><td>00</td><td>0</td><td>|000></td><td>|000></td></tr>
<tr><td>01</td><td>1</td><td>|010></td><td>|011></td></tr>
<tr><td>10</td><td>1</td><td>|100></td><td>|101></td></tr>
<tr><td>11</td><td>1</td><td>|110></td><td>|111></td></tr>
</table>

We can observe that for cases where there are two inputs and an output, there is a one-to-one relationship between inputs A and B with the inputs of the qubits of Q1 and Q2. Likewise there is a one-to-one relationship between the output of the operator with the output measurement of the qubit Q3.

In cases where there is only one input and one output, the qubit α serves as an input, and it is also on this that the output is measured.

With the above we can define the quantum circuit analogous to the combinational of implication in its Jauch form as follows:

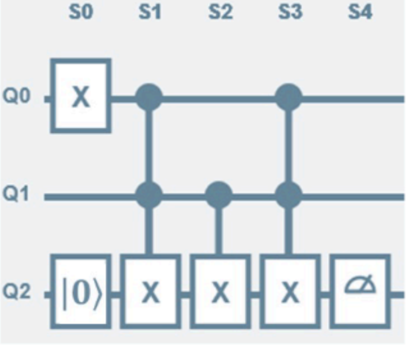


FIGURE 8. Quantum circuit of implication in its Jauch form.

 DOI: https://doi.org/10.18272/aci.v17i1.3709

TABLE 8. Quantum implication in its Jauch form truth table.

|Q0Q1Q2 >	IJauch|Q0Q1Q2 >
|000>	|101>
|010>	|111>
|100>	|000>
|110>	|011>

The syllogism in the quantum logic

As previously mentioned, through prepositional logic we can build any definition or algorithm from basic logical operations, so to develop the hypothetical syllogism the following definition will be used from implications:

$$[(P \to Q) \wedge (Q \to R) \to (P \to R)] \tag{16}$$

Where P, Q, and R are terms that can be ordered according to the way in which we are working. With this we can build a definition based on combinational logic circuits as follows:

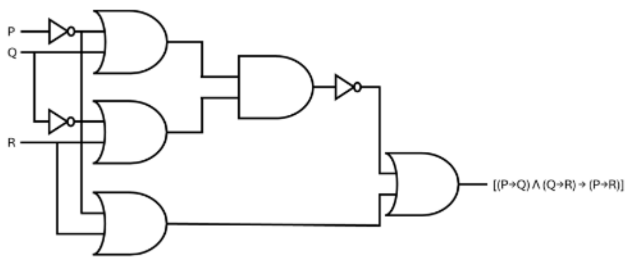


FIGURE 9. A circuit of hypothetical syllogism using implication in its Jauch form.

With the following associated truth table:

TABLE 9. Hypothetical syllogism truth table.

P	Q	R	[(P→Q) Λ (Q→R) → (P→R)]
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	1
0	1	1	1
0	1	0	1
0	0	1	1
0	0	0	1

 DOI: https://doi.org/10.18272/aci.v17i1.3709

Using the methodology mentioned above to build a quantum circuit from classical definitions results in the following quantum algorithm:

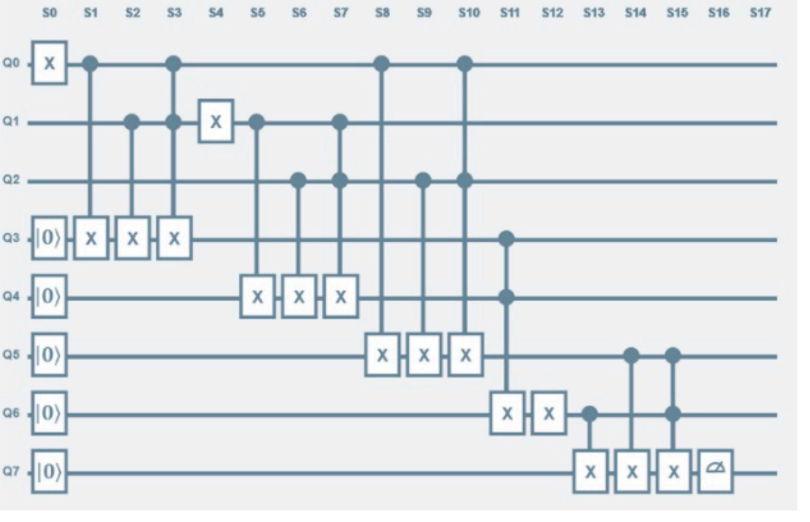


FIGURE 10. Quantum circuit of the hypothetical syllogism in its Jauch form.

Where: Q0=P; Q1=Q; Q2=R

From the above, we take as inputs only those where the qubits Q3, Q4, Q5, Q6, and Q7 are initialized in the base state 0, so our analogous truth table is as follows:

TABLE 10. Quantum circuit of the hypothetical syllogism in its Jauch form.

|Q0Q1Q2Q3Q4Q5Q6Q7>	[(P→Q) Λ (Q→R) → (P→R)]
|00000000>	|11011101>
|00100000>	|11111101>
|01000000>	|10010111>
|01100000>	|10111101>
|10000000>	|01001011>
|10100000>	|01101111>
|11000000>	|00010011>
|11100000>	|00111101>

Where Q7 is our observation qubit, and we see that both the classical and quantum outputs are tautological, so the circuit is analog.

 DOI: https://doi.org/10.18272/aci.v17i1.3709

CONCLUSIONS

This algorithm represents a syllogism constructed with Jauch-shaped implication operations. To understand whether the form is deductive, inductive, or abductive, we must run experiments several times, and the result with the greatest amplitude of probability will define the shape and the figure.

To understand the functioning of the syllogism in its other forms and figures, it is necessary to construct the analogous circuits that represent them, as well as those constructed from the different forms of implication, which will be dealt with in later works.

Further work could develop a decision-making algorithm to select the indicated conclusion of a syllogism based on its probability amplitude and possible context variables.

AUTHOR CONTRIBUTIONS

Mtro. José David Bañuelos Aquino & Dr. José Antonio Orizaga Trejo: conceptualization, data collection, data analysis, quantum algorithm development, manuscript writing, supervision and review and editing.

CONFLICT OF INTEREST

The authors declare that there are no conflict of interest related to this work.

 DOI: https://doi.org/10.18272/aci.v17i1.3709

REFERENCES

[1] Ivorra Castillo, C. (n.d.). *Lógica y teoría de conjuntos*. Universitat de València. https://www.uv.es/ivorra/Libros/TC.pdf

[2] Copi, I. M., & Cohen, C. (2019). *Introducción a la lógica* (14ª ed.). Limusa. https://logicaformalunah.wordpress.com/wp-content/uploads/2017/01/irving_m-_copi_carl_cohen_introduccion_a_la_log.pdf

[3] Lungarzo, C. (n.d.). *La función de la llamada lógica clásica en la lógica cuántica*. Universidad Estatal de Campinas, Campinas.

[4] Ezhov, A., Andrei, K., & Terentyeva, S. (2014). *On the quantum implication function and strategies for multi-agent models*. In PRIMA2014 *International Conference on Principles and Practice of Multi-Agent Systems*.

[5] Megill, N. D., & Pavičić, M. (2003). Quantum implication algebras. *International journal of theoretical physics*, *42*. https://doi.org/10.1023/B:IJTP.0000006007.58191.da

[6] Bañuelos Aquino, J. D., & Orizaga Trejo, J. A. (2023). *Q-Team, simulador de circuitos cuánticos*. In J. A. Orizaga Trejo, M. Cruz Herrera, J. R. Beltrán Ramírez, M. Alarcón Romero, E. A. Ramos Aceves, J. D. Bañuelos Aquino, & J. Z. Orizaga Cruz, (eds.) *Procesos de madurez en desarrollos tecnológicos en posgrado con vinculación social* (pp. 115-146). Trauco. https://www.amidibiblioteca.amidi.mx/index.php/AB/catalog/view/39/33/308

[7] Deschamps, J. (2023). *Computación cuántica: circuitos y algoritmos*. Barcelona: Marcombo, Alfaomega.

[8] Bonillo, V. (2013). *Principios fundamentales de computación cuántica*. Texto de Apoyo, Universidad de A Coruña, Departamento de Computación, Facultad de Informática. Universidad de La Coruña. https://cateinf.wordpress.com/wp-content/uploads/2016/05/principios-fundamentales-de-computacic3b3n-cuc3a1ntica.pdf

[9] Peralta, P. J. S., & Sáenz, A. L. S. (2001). Computación Cuántica. Una revolución en el tratamiento de la información. *Revista española de física, 15*(3), 1.

[10] Vázquez, J. M. (2013). *Arquitectura y tecnología de computadores*. Universidad de Sevilla.

 DOI: https://doi.org/10.18272/aci.v17i1.3709