



**ARTICLE**

# Foundations of Photonic Quantum Computation. Part 1. Introduction to Quantum Operators

Martin Bombardelli<sup>1,2,3</sup>, Gérard Fleury<sup>1</sup>, Philippe Lacomme<sup>1,\*</sup> and Bogdan Vulpescu<sup>2</sup>

<sup>1</sup>UMR 6158 LIMOS, Université Clermont Auvergne, 1 Rue de la Chébarde, Aubière, France

<sup>2</sup>UMR 6533 LPCA, Université Clermont Auvergne, 4 Av. Blaise Pascal, Aubière, France

<sup>3</sup>CGI 10 Rue Eric de Cromières, Clermont-Ferrand, France

\*Corresponding Author: Philippe Lacomme. Email: [philippe.lacomme@isima.fr](mailto:philippe.lacomme@isima.fr)

Received: 31 October 2025; Accepted: 23 April 2026; Published: 26 June 2026

**ABSTRACT:** This work aims to present the fundamental concepts necessary for performing computations on photonic quantum computers by detailing the gates specific to this architecture and introducing the relationships between standard Pauli gates and their counterparts in photonic systems. Physical considerations related to the optical components are addressed. Theoretical aspects concerning quantum operators are presented, and applied sections demonstrating implementations using the Perceval library developed by Quandela are given. This paper provides a comprehensive, state-of-the-art description of photonic quantum gates in a formal way. The main contribution of this paper is to allow researchers with prior knowledge of the gate model of quantum computing to have a comprehensive understanding of how some photonic quantum computers work. Three optical components are used in this document: beam splitters, phase shifters and permutation components. Most of the more common quantum gates are built from the physical description of these components and the associated physical laws. Extensive focus is placed on controlled gates, which have more complicated constructions. Two different constructions of the CNOT gate are given to show how versatile photonic quantum computers can be. The main drawback of this type of controlled gate is discussed, namely the probabilistic nature of its implementation.

**KEYWORDS:** Introduction; photonic; quantum computing; encoding; gate model; physics

## 1 Introduction

### 1.1 Photonic Quantum Computing

Simulating quantum physics on a computer is widely believed to be an algorithmically difficult task. This was already the case as early as 1980, which led Manin [1,2] to briefly mention the idea of using quantum systems to build quantum automata. The same year, Benioff [3] introduced quantum Turing machines. One of the problems that arise when one wants to simulate quantum physics accurately is the amount of memory needed to fully represent the quantum state of the system [4]. This difficulty in simulating quantum systems led Richard Feynman to imagine a “computer” made of and for quantum systems in his foundational article “Simulating Physics with Computers” [5]. The concept was then extended to what is known as gate model quantum computing (QC) by David Deutsch [6]. The first advantage of this quantum model of computation was found by leveraging the query model on Deutsch’s problem. QC allows us to check if a given function  $f: \{0,1\} \rightarrow \{0,1\}$  is constant ( $f(0) = f(1)$ ) or if it is balanced ( $f(0) \neq f(1)$ ) in one query whereas its

classical counterpart needs two measurements. An extended version was then proposed in [7], achieving an even greater improvement over the classical counterpart.

Building on these early successes, researchers came up with other quantum algorithms which had a lower algorithmic complexity than their classical counterparts. The canonical and most famous examples are Grover's [8] search algorithm and Shor's factoring algorithm [9].

Photonic QC is one of many physical realizations of a quantum computer. The first proposal came from [10], which introduced "dual-rail encoding," making it possible to encode qubits into Fock states [11]. This led Knill, Laflamme, and Milburn to develop an entire protocol for physically realizing a gate-model quantum computer using optical components, known as the KLM protocol [12]. This protocol was further developed in the following years in [13,14]. This model was refined in [15] into a more complex model of photonic QC, called one-way QC or measurement-based QC, which lies outside the scope of this work.

Photons have proven to be good candidates for the main applications of quantum information. First, they have been used to simulate Hamiltonian dynamics with good precision and high success probability on a 4-qubit photonic silicon chip [16]. Photonic quantum information has applications to cybersecurity with many examples of quantum key distribution being achieved using photons [17–19]. In the field of optimization, although no advantage has been found yet, several promising papers aim to reduce the qubit count required to use photonic quantum computers to solve hard optimization problems [20–22].

Another reason to consider photons a powerful quantum resource is that Scott Aaronson and Anton Arkhipov proposed a protocol for demonstrating quantum supremacy, called boson sampling, which can be implemented using photons and linear optical components [23]. In 2020, ref. [24] reported a claim of quantum computational advantage using this protocol.

## 1.2 Understanding Modes and Defining Qubits from Modes

To create qubits in a photonic quantum computer, the first method proposed was path encoding, introduced by [10]. The locations of the photons—referred to as spatial modes—define the state of the qubit. Spatial modes can, for instance, correspond to optical fibers. For convenience, throughout this chapter, we use the term *mode* to refer specifically to spatial mode.

To describe a system composed of photons and modes. We use the notation  $|n_0, n_1, \dots, n_{m-1}\rangle_{01\dots(m-1)}$  where  $0, 1, \dots, (m-1)$  denote the spatial modes occupied by  $n_0, n_1, \dots, n_{m-1}$  photons. This type of states is referred to as Fock state.

To define a Qubit, let us now consider a specific case where a single photon is distributed over two modes. The definition of the two basic states of a qubit in the dual rail encoding relies on the location of the said photon.

$$|0\rangle = |1, 0\rangle_{01}$$

$$|1\rangle = |0, 1\rangle_{01}$$

When the photon is in mode 0, this corresponds to the Fock state  $|1, 0\rangle_{01}$  sometimes written simply as  $|1, 0\rangle$  (although this notation should be avoided to prevent confusion between qubit states and Fock states).

### Remark

When possible and in cases where ambiguity may arise, we use the notation  $|n_0 n_1 \dots n_{m-1}\rangle_f$  to indicate explicitly that the set of modes corresponds to a Fock state.

It is also common to omit the commas between each component of the state, so that  $|n_0, n_1, \dots, n_{m-1}\rangle_f$  is often written as  $|n_0 n_1 \dots n_{m-1}\rangle_f$ .

The Fock state  $|1, 0\rangle_{01}$  is encoding the qubit state  $|0\rangle$  and a representation is provided in Fig. 1.

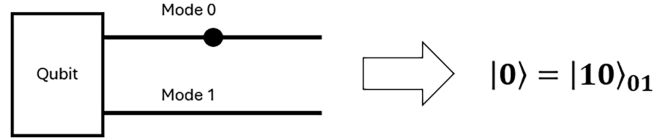


Figure 1: Qubit state  $|0\rangle = |10\rangle_{01}$ .

Similarly, the Fock state  $|01\rangle_{01}$  is encoding the qubit state  $|1\rangle$  and a representation is provided in Fig. 2.

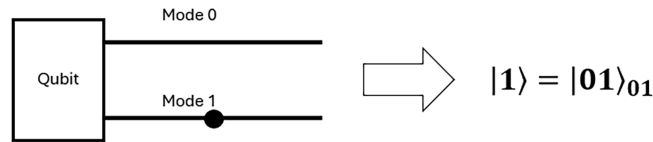


Figure 2: Qubit state  $|1\rangle = |01\rangle_{01}$ .

This encoding is called the dual mode encoding. But there exist many other possibilities to encode qubits, qudits or even continuous variable quantum information! [25].

### 1.3 Creation and Annihilation Operators for the Modes

The creation and annihilation operators associated with a mode  $a$  are denoted  $\hat{a}_a^\dagger$  and  $\hat{a}_a$ , respectively. Their actions are as follows:

- The creation operator  $\hat{a}_0^\dagger$  allows the transition from  $|0\rangle_0$  to  $|1\rangle_0$  in spatial mode 0. For example,  $\hat{a}_0^\dagger$  transforms the vacuum state  $|00\rangle_{01}$  into  $|10\rangle_{01}$ .
- The annihilation operator  $\hat{a}_0$  allows the transition from  $|1\rangle_0$  to  $|0\rangle_0$ . For example,  $\hat{a}_0$  transforms  $|10\rangle_{01}$  into the vacuum state  $|00\rangle_{01}$ .

These operators correspond to the emission or annihilation of a photon in a given mode. They play a crucial role when describing the operation of complex optical components.

## 2 The Generic Beam Splitter Component

### 2.1 Physical Description of the Action of the Beam Splitter

A beam splitter is an optical component used in photonic circuits. Let  $E_a$  and  $E_b$  denote the amplitudes of the two input electric fields, and  $E_c$  and  $E_d$  the amplitudes of the two output fields, as illustrated in Fig. 3. Let  $r_{ac}$  and  $r_{bd}$  be the reflection coefficients,  $t_{ad}$  and  $t_{bc}$  the transmission coefficients. We assume that the magnitudes are restricted to  $|r_{ac}|, |r_{bd}|, |t_{ad}|, |t_{bc}| \in [0, 1]$ .

The output field amplitudes can be expressed in terms of the input amplitudes through the so-called continuity relations:

$$E_c = r_{ac}E_a + t_{bc}E_b \tag{1}$$

$$E_d = t_{ad}E_a + r_{bd}E_b \tag{2}$$

where

$r_{ac}$ : is the contribution of  $E_a$  found at the output  $E_c$

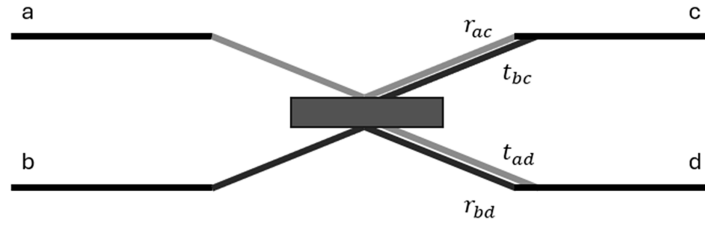
$r_{bd}$ : is the contribution of  $E_b$  found at the output  $E_d$

$t_{ad}$ : is the contribution of  $E_a$  found at the output  $E_d$

$t_{bc}$ : is the contribution of  $E_b$  found at the output  $E_c$

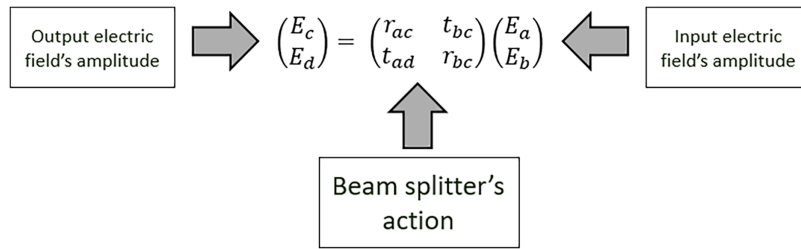
The conservation of energy (proportional to the squared modulus of  $E_x$ ) leads to:

$$|E_a|^2 + |E_b|^2 = |E_c|^2 + |E_d|^2 \quad (3)$$



**Figure 3:** Representation of a beam splitter with two input modes and two output modes.

Using (1) and (2), the action of the beam splitter can be described in matrix form based on the field amplitudes of each mode, as shown in Fig. 4.



**Figure 4:** Action of a beam splitter.

Let us note:

$$U = \begin{bmatrix} r_{ac} & t_{bc} \\ t_{ad} & r_{bd} \end{bmatrix}$$

If the entire input field is injected into mode  $a$ , the input amplitudes satisfy

$$|E_a|^2 = 1, |E_b|^2 = 0,$$

which is consistent with the normalization condition  $|E_a|^2 + |E_b|^2 = 1$ , then we have:

$$\begin{pmatrix} r_{ac} & t_{bc} \\ t_{ad} & r_{bd} \end{pmatrix} \cdot \begin{pmatrix} E_a \\ 0 \end{pmatrix} = \begin{pmatrix} E_c \\ E_d \end{pmatrix}$$

It follows that:

$$r_{ac} \cdot E_a = E_c \quad (4)$$

$$t_{ad} \cdot E_a = E_d \tag{5}$$

Staying in the case of one single input (here in  $a$ ), we have  $|E_b|^2 = 0$ , applying to (3) gives:

$$|E_a|^2 = |E_c|^2 + |E_d|^2 \tag{6}$$

because (4) and (5):

$$|E_a|^2 = |r_{ac}E_a|^2 + |t_{ad}E_a|^2$$

So

$$|E_a|^2 = (|r_{ac}|^2 + |t_{ad}|^2) \cdot |E_a|^2$$

Because we consider that we have an input in  $a$ , we have  $E_a \neq 0$  and we can divide both terms of the equation by  $|E_a|^2$

$$1 = |r_{ac}|^2 + |t_{ad}|^2 \tag{7}$$

By the applying the same process this time with  $E_a = 0$  and  $E_b \neq 0$ :

$$\begin{pmatrix} r_{ac} & t_{bc} \\ t_{ad} & r_{bd} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ E_b \end{pmatrix} = \begin{pmatrix} E_c \\ E_d \end{pmatrix}$$

One would obtain:

$$1 = |r_{bd}|^2 + |t_{bc}|^2 \tag{8}$$

Given (1) and (2), it is possible to calculate:

$$|E_c|^2 = |r_{ac}E_a + t_{bc}E_b|^2 = |r_{ac}E_a|^2 + |t_{bc}E_b|^2 + r_{ac}E_a (t_{bc}E_b)^* + t_{bc}E_b (r_{ac}E_a)^*$$

Similarly

$$|E_d|^2 = |t_{ad}E_a|^2 + |r_{bd}E_b|^2 + t_{ad}E_a \cdot r_{bd}^*E_b^* + r_{bd}E_b \cdot t_{ad}^*E_a^*$$

Adding both equations together:

$$\begin{aligned} |E_c|^2 + |E_d|^2 &= (|r_{ac}|^2 + |t_{ad}|^2) |E_a|^2 + (|r_{bd}|^2 + |t_{bc}|^2) |E_b|^2 + r_{ac}E_a \cdot (t_{bc}E_b)^* + t_{bc}E_b \cdot (r_{ac}E_a)^* \\ &\quad + t_{ad}E_a \cdot r_{bd}^*E_b^* + r_{bd}E_b \cdot t_{ad}^*E_a^* = \underbrace{(|r_{ac}|^2 + |t_{ad}|^2)}_{=1(\text{see } ((7)))} |E_a|^2 + \underbrace{(|r_{bd}|^2 + |t_{bc}|^2)}_{=1} |E_b|^2 \\ &\quad + (t_{ad}r_{bd}^* + r_{ac}t_{bc}^*) E_aE_b^* + (r_{bd}t_{ad}^* + t_{bc}r_{ac}^*) E_a^*E_b \end{aligned}$$

So

$$|E_c|^2 + |E_d|^2 = |E_a|^2 + |E_b|^2 + (t_{ad}r_{bd}^* + r_{ac}t_{bc}^*) E_aE_b^* + (r_{bd}t_{ad}^* + t_{bc}r_{ac}^*) E_a^*E_b$$

### Reminder

According to (3), we should have:

$$|E_a|^2 + |E_b|^2 = |E_c|^2 + |E_d|^2$$

By substituting  $|E_c|^2 + |E_d|^2$  using the above relation:

$$|E_a|^2 + |E_b|^2 = |E_a|^2 + |E_b|^2 + (t_{ad}r_{bd}^* + r_{ac}t_{bc}^*)E_aE_b^* + (r_{bd}t_{ad}^* + t_{bc}r_{ac}^*)E_a^*E_b$$

We get:

$$(t_{ad}r_{bd}^* + r_{ac}t_{bc}^*)E_aE_b^* + (r_{bd}t_{ad}^* + t_{bc}r_{ac}^*)E_a^*E_b = 0$$

Since this expression must hold for any choice of  $E_a$  and  $E_b$ , it implies the two following relations:

$$(t_{ad}r_{bd}^* + r_{ac}t_{bc}^*) = 0 \quad (9)$$

$$(r_{bd}t_{ad}^* + t_{bc}r_{ac}^*) = 0 \quad (10)$$

**Theorem 1:**  $U$  is unitary.

**Proof:** Consider the matrix:

$$U^\dagger U = \begin{bmatrix} r_{ac}^* & t_{ad}^* \\ t_{bc}^* & r_{bd}^* \end{bmatrix} \cdot \begin{bmatrix} r_{ac} & t_{bc} \\ t_{ad} & r_{bd} \end{bmatrix}$$

$$U^\dagger U = \begin{bmatrix} r_{ac}r_{ac}^* + t_{ad}t_{ad}^* & r_{ac}^*t_{bc} + t_{ad}^*r_{bd} \\ t_{bc}^*r_{ac} + r_{bd}^*t_{ad} & t_{bc}^*t_{bc} + r_{bd}r_{bd}^* \end{bmatrix}$$

$$U^\dagger U = \begin{bmatrix} |r_{ac}|^2 + |t_{ad}|^2 & r_{ac}^*t_{bc} + t_{ad}^*r_{bd} \\ t_{bc}^*r_{ac} + r_{bd}^*t_{ad} & |t_{bc}|^2 + |r_{bd}|^2 \end{bmatrix}$$

Using (7)–(10), we directly get:

$$U^\dagger \cdot U = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Therefore, the beam splitter's operation is unitary.  $\square$

## 2.2 Parameterization of the Beam Splitter Matrix $U$

The gate  $U$  representing the action of a beam splitter can then be denoted as  $BS$ :

$$BS = \begin{pmatrix} r_{ac} & t_{bc} \\ t_{ad} & r_{bd} \end{pmatrix}$$

There are several conventions for the beam splitter, some of which belong to  $U(2)$  but not to  $SU(2)$ . The goal of this section is to understand how those different conventions appear and why different forms exist.

**Theorem 2:**  $\phi_{ad} - \phi_{bd} - \phi_{ac} + \phi_{bc} = \pi$  with  $\phi_{ij}$  is the phase of the exponential form of  $t_{ij}$ .

**Proof:** As a starting point, let us take the relation demonstrated in the preceding section:

$$t_{bc}^* \cdot r_{ac} + r_{bd}^* \cdot t_{ad} = 0$$

As all the coefficients are complex numbers, each one can be represented in its exponential form so we can be written:

$$|t_{bc}| e^{-i.\phi_{bc}} . |r_{ac}| . e^{i.\phi_{ac}} + |r_{bd}| . e^{-i.\phi_{bd}} . |t_{ad}| . e^{i.\phi_{ad}} = 0$$

$$|t_{bc}| . |r_{ac}| . e^{i.(\phi_{ac}-\phi_{bc})} + |r_{bd}| . |t_{bd}| . e^{i.(\phi_{ad}-\phi_{bd})} = 0$$

So

$$|t_{bc}| . |r_{ac}| . e^{i.(\phi_{ac}-\phi_{bc})} = - |r_{bd}| . |t_{ad}| . e^{i.(\phi_{ad}-\phi_{bd})}$$

$$|t_{bc}| . |r_{ac}| = - |r_{bd}| . |t_{ad}| . e^{i.(\phi_{ad}-\phi_{bd}-\phi_{ac}+\phi_{bc})}$$

$$\frac{|r_{ac}|}{|t_{ad}|} = - \frac{|r_{bd}|}{|t_{bc}|} . e^{i.(\phi_{ad}-\phi_{bd}-\phi_{ac}+\phi_{bc})}$$

It is now possible to analyze:

$$\frac{|r_{ac}|}{|t_{ad}|} = - \frac{|r_{bd}|}{|t_{bc}|} . e^{i.(\phi_{ad}-\phi_{bd}-\phi_{ac}+\phi_{bc})}$$

All magnitudes are positive real numbers, so  $e^{i.(\phi_{ad}-\phi_{bd}-\phi_{ac}+\phi_{bc})}$  can only take the value  $-1$ .

From this we find the first important relation:

$$\phi_{ad} - \phi_{bd} - \phi_{ac} + \phi_{bc} = \pi \tag{11}$$

□

**Theorem 3:**  $r^2 + t^2 = 1$  with  $t := |t_{ad}| = |t_{bc}|$  and  $r := |r_{ac}| = |r_{bd}|$ .

**Proof:** To find the second important relation, let us start with:

$$\frac{|r_{ac}|}{|t_{ad}|} = \frac{|r_{bd}|}{|t_{bc}|}$$

Squaring both sides of the equation gives:

$$\frac{|r_{ac}|^2}{|t_{ad}|^2} = \frac{|r_{bd}|^2}{|t_{bc}|^2}$$

Using (7) and (8):

$$\frac{1 - |t_{ad}|^2}{|t_{ad}|^2} = \frac{1 - |t_{bc}|^2}{|t_{bc}|^2}$$

i.e.,

$$\frac{1}{|t_{ad}|^2} - 1 = \frac{1}{|t_{bc}|^2} - 1$$

It follows that

$$|t_{ad}| = |t_{bc}| \text{ and we note } \mathbf{t} := |t_{ad}| = |t_{bc}|$$

The equation  $\frac{|r_{ac}|}{|t_{ad}|} = \frac{|r_{bd}|}{|t_{bc}|}$  can be rewritten  $\frac{|r_{ac}|}{t} = \frac{|r_{bd}|}{t}$  which implies that:

$$|r_{ac}| = |r_{bd}| \text{ and we note } r := |r_{ac}| = |r_{bd}|$$

Recalling (8)

$$|r_{ac}|^2 + |t_{ad}|^2 = 1$$

And using the new notation leads to the second important relation:

$$r^2 + t^2 = 1 \tag{12}$$

□

### Remark on the Physical Meaning of the Coefficients $r$ and $t$

The coefficients  $r^2 = R$  and  $t^2 = T$  correspond to:

$$\text{The reflectance } R = \frac{P_{\text{reflected}}}{P_{\text{incoming}}}$$

$$\text{The transmittance } T = \frac{P_{\text{transmitted}}}{P_{\text{incoming}}}$$

We can also define:

$$\text{The absorbance } A = \frac{P_{\text{absorbed}}}{P_{\text{incoming}}} \text{ where all the } P \text{ values represent powers.}$$

Energy conservation requires  $R + T + A = 1$  and throughout this document, we consider lossless beam splitters (i.e.,  $A = 0$ )

By taking the complex form of its coefficients, the BS operator can be parameterized as:

$$BS = \begin{pmatrix} |r_{ac}| \cdot e^{i \cdot \phi_{ac}} & |t_{bc}| \cdot e^{i \cdot \phi_{bc}} \\ |t_{ad}| \cdot e^{i \cdot \phi_{ad}} & |r_{bd}| \cdot e^{i \cdot \phi_{bd}} \end{pmatrix}$$

And by definition of  $r$  and  $t$ :

$$BS = \begin{pmatrix} r \cdot e^{i \cdot \phi_{ac}} & t \cdot e^{i \cdot \phi_{bc}} \\ t \cdot e^{i \cdot \phi_{ad}} & r \cdot e^{i \cdot \phi_{bd}} \end{pmatrix}$$

Thus, the BS operator is completely defined by the six parameters:

$$r, t, \phi_{ac}, \phi_{bc}, \phi_{ad}, \phi_{bd}$$

And by applying the two important relations (11) and (12):

$$r^2 + t^2 = 1$$

$$\phi_{ad} - \phi_{bd} - \phi_{ac} + \phi_{bc} = \pi$$

We would like to reduce the number of parameters of  $BS$ .

To do this, we start by using (11):

$$\begin{cases} r^2 + t^2 = 1 \\ r = |r_{ac}| \geq 0 \text{ and } t > 0 \end{cases} \Rightarrow \exists \theta \in \left[0, +\frac{\pi}{2}\right], r = \sin(\theta) \text{ and } t = \cos(\theta) \text{ with } \theta = \arctan\left(\frac{r}{t}\right)$$

One can now write:

$$BS = \begin{pmatrix} r.e^{i.\phi_{ac}} & t.e^{i.\phi_{bc}} \\ t.e^{i.\phi_{ad}} & r.e^{i.\phi_{bd}} \end{pmatrix} = \begin{pmatrix} \sin(\theta).e^{i.\phi_{ac}} & \cos(\theta).e^{i.\phi_{bc}} \\ \cos(\theta).e^{i.\phi_{ad}} & \sin(\theta).e^{i.\phi_{bd}} \end{pmatrix}$$

In their paper, ref. [26] propose the change of variables:

$$\phi_t = \frac{1}{2}(\phi_{ad} - \phi_{bc})$$

$$\phi_r = \frac{1}{2}(\phi_{ac} - \phi_{bd} - \pi)$$

$$\phi_0 = \frac{1}{2}(\phi_{ad} + \phi_{bc})$$

To obtain

$$\phi_{ac} = \phi_0 + \phi_r$$

$$\phi_{ad} = \phi_0 + \phi_t$$

$$\phi_{bc} = \phi_0 - \phi_t$$

$$\phi_{bd} = \phi_0 - \phi_r - \pi$$

### Verification

The verifications for  $\phi_{ad}$  and  $\phi_{bc}$  are straightforward; let us detail  $\phi_{ac}$  and  $\phi_{bd}$

For  $\phi_{ac}$ :

$$\phi_0 + \phi_r = \frac{1}{2}(\phi_{ad} + \phi_{bc}) + \frac{1}{2}(\phi_{ac} - \phi_{bd} - \pi)$$

$$= \frac{1}{2}(\phi_{ad} + \phi_{bc} + \phi_{ac} - \phi_{bd} - \pi)$$

$$\phi_0 + \phi_r = \frac{1}{2} \left( \underbrace{\phi_{ad} - \phi_{ac} + \phi_{bc} - \phi_{bd} - \pi}_{\pi(11)} + \phi_{ac} + \phi_{ac} \right)$$

$$\phi_0 + \phi_r = \phi_{ac}$$

And for  $\phi_{bd}$ :

$$\phi_0 - \phi_r - \pi = \frac{1}{2}(\phi_{ad} + \phi_{bc}) - \frac{1}{2}(\phi_{ac} - \phi_{bd} - \pi) - \pi$$

$$\phi_0 - \phi_r - \pi = \frac{1}{2} \left( \underbrace{\phi_{ad} - \phi_{ac} + \phi_{bc} - \phi_{bd}}_{\pi(11)} + \phi_{bd} + \phi_{bd} + \pi \right) - \pi = \frac{(2\phi_{bd} + 2\pi)}{2} - \pi$$

$$\phi_0 - \phi_r - \pi = \phi_{bd}$$

We can therefore rewrite by replacing each of the expressions:

$$BS = \begin{pmatrix} \sin(\theta).e^{i.\phi_{ac}} & \cos(\theta).e^{i.\phi_{bc}} \\ \cos(\theta).e^{i.\phi_{ad}} & \sin(\theta).e^{i.\phi_{bd}} \end{pmatrix} = \begin{pmatrix} \sin(\theta).e^{i.(\phi_0+\phi_r)} & \cos(\theta).e^{i.(\phi_0-\phi_t)} \\ \cos(\theta).e^{i.(\phi_0+\phi_t)} & \sin(\theta).e^{i.(\phi_0-\phi_r-\pi)} \end{pmatrix}$$

Thus, a common expression for the noiseless beam splitter becomes:

$$BS_1 \equiv BS = e^{i.\phi_0} \begin{pmatrix} \sin(\theta).e^{i.\phi_r} & \cos(\theta).e^{-i.\phi_t} \\ \cos(\theta).e^{i.\phi_t} & -\sin(\theta).e^{-i.\phi_r} \end{pmatrix}$$

We obtained an expression of BS using only 4 parameters:

$\theta, \phi_0, \phi_r$  and  $\phi_t$

### 2.3 Important Remark on Different Conventions

The definitions of  $\phi_r, \phi_t$  and  $\phi_0$  are not universal; for example, by changing the definition of  $\phi_r$ :

$$\phi_r = \frac{1}{2}(\phi_{ac} - \phi_{bd} + \pi)$$

We would obtain:

$$BS_2(\theta, \phi_r, \phi_t, \phi_0) = e^{i.\phi_0} \begin{pmatrix} -\sin\left(\frac{\theta}{2}\right).e^{i.\phi_r} & \cos\left(\frac{\theta}{2}\right).e^{-i.\phi_t} \\ \cos\left(\frac{\theta}{2}\right).e^{i.\phi_t} & \sin\left(\frac{\theta}{2}\right).e^{-i.\phi_r} \end{pmatrix}$$

Or alternatively, by defining:

$$r^2 + t^2 = 1 \Rightarrow \exists \theta \in [0, \pi], r = \cos\left(\frac{\theta}{2}\right) \text{ and } t = \sin\left(\frac{\theta}{2}\right) \text{ with } \theta = \arctan\left(\frac{t}{r}\right)$$

One could also have obtained:

$$BS_3(\theta, \phi_r, \phi_t, \phi_0) = e^{i.\phi_0} \begin{pmatrix} \cos\left(\frac{\theta}{2}\right).e^{i.\phi_r} & \sin\left(\frac{\theta}{2}\right).e^{-i.\phi_t} \\ \sin\left(\frac{\theta}{2}\right).e^{i.\phi_t} & -\cos\left(\frac{\theta}{2}\right).e^{-i.\phi_r} \end{pmatrix}$$

In by using  $BS_3$  and considering  $\phi_r = \phi_t = \phi_0 = 0$  and  $\theta = \frac{\pi}{2}$ , we obtain the expression of the Hadamard gate.

$$BS_3\left(\frac{\pi}{2}, 0, 0, 0\right) = \begin{pmatrix} \cos\left(\frac{\pi}{4}\right) & \sin\left(\frac{\pi}{4}\right) \\ \sin\left(\frac{\pi}{4}\right) & -\cos\left(\frac{\pi}{4}\right) \end{pmatrix} = \begin{pmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{pmatrix} = H$$

### 2.4 Quandela Conventions

Note that the conventions presented above are not exhaustive. Perceval, the library proposed by [27], offers by default three conventions.

Let us now consider, in each case, all phases set to zero.

**Quandela Convention  $H$ :**

$$BS_H(\theta \in [0; \pi], 0, 0, 0) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & -\cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

In particular,  $BS_H\left(\frac{\pi}{2}, 0, 0, 0\right) = H$  defines the Hadamard gate. This convention can be obtained with the same definitions of  $\phi_r, \phi_t$  and  $\phi_0$  as in the  $BS_1$  convention but by choosing:

$$\theta \in [0, \pi], r = \cos\left(\frac{\theta}{2}\right) \text{ et } t = \sin\left(\frac{\theta}{2}\right)$$

**Quandela Convention  $R_x$ :**

$$BS_{R_x}(\theta \in [0; \pi], 0, 0, 0) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & i \sin\left(\frac{\theta}{2}\right) \\ i \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

The operator  $BS_{R_x}$  is obtained by again choosing  $\theta \in [0, \pi], r = \cos\left(\frac{\theta}{2}\right)$  and  $t = \sin\left(\frac{\theta}{2}\right)$ , in the previous calculations, along with the selections of  $\phi_{t,r,0}$  given in [Table 1](#).

**Table 1:** Choice of the new  $\phi_{t,r,0}$ .

New Parameters $\phi_{t,r,0}$	Previous Parameters $\phi_{t,r,0}$ Introduced Page 8
$\phi_t = \frac{1}{2}(\phi_{ad} - \phi_{bc})$	$\phi_t = \frac{1}{2}(\phi_{ad} - \phi_{bc})$
$\phi_r = \frac{1}{2}(\phi_{ac} - \phi_{bd})$	$\phi_r = \frac{1}{2}(\phi_{ac} - \phi_{bd} - \pi)$
$\phi_0 = \frac{1}{2}(\phi_{ad} + \phi_{bc} - \pi)$	$\phi_0 = \frac{1}{2}(\phi_{ad} + \phi_{bc})$

**New parameters for  $\phi_{t,r,0}$  to define  $BS_{R_x}$**

$$\phi_t = \frac{1}{2}(\phi_{ad} - \phi_{bc}),$$

$$\phi_r = \frac{1}{2}(\phi_{ac} - \phi_{bd}),$$

$$\phi_0 = \frac{1}{2}(\phi_{ad} + \phi_{bc} - \pi)$$

which can be rewritten

$$\phi_{ac} = \phi_0 + \phi_r$$

$$\phi_{bd} = \phi_0 - \phi_r$$

$$\phi_{ad} = \phi_0 + \phi_t + \frac{\pi}{2}$$

$$\phi_{bc} = \phi_0 - \phi_t + \frac{\pi}{2}$$

**Table 1** provides the correspondence between the old definitions of  $\phi_{t,r,0}$  and the new definitions.

**Quandela Convention  $R_y$ :**

$$BS_{Ry}(\theta \in [0; \pi], 0, 0, 0) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

This corresponds exactly to the  $RY$  gate according to IBM's notation. This expression arises through similar transformations.

### 2.5 Symmetric Beam Splitter

A beam splitter is said to be symmetric if it is assigned  $r = \frac{1}{\sqrt{2}}$  and  $t = \frac{1}{\sqrt{2}}$ . This corresponds to the photon having a “one in two chance of being reflected” and a “one in two chance of being transmitted”.

It means choosing  $\theta = \frac{\pi}{4}$  in most conventions

In the first convention,

$$BS_1(\theta, \phi_r, \phi_t, \phi_0) = e^{i \cdot \phi_0} \begin{pmatrix} \sin(\theta) \cdot e^{i \cdot \phi_r} & \cos(\theta) \cdot e^{-i \cdot \phi_t} \\ \cos(\theta) \cdot e^{i \cdot \phi_t} & -\sin(\theta) \cdot e^{-i \cdot \phi_r} \end{pmatrix}$$

From a mathematical point of view, we choose  $\phi_r = \phi_t = \phi_0 = 0$  with  $\theta = \frac{\pi}{4}$  and we obtain, for example:

$$BS_1\left(\frac{\pi}{4}, 0, 0, 0\right) = 1 \begin{pmatrix} \frac{1}{\sqrt{2}} \cdot 1 & \frac{1}{\sqrt{2}} \cdot 1 \\ \frac{1}{\sqrt{2}} \cdot 1 & -\frac{1}{\sqrt{2}} \cdot 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = H$$

The values of the symmetric beam splitter in the Quandela convention (for  $\theta = \frac{\pi}{2}$  and all phases equal to 0) are:

$$BS_H\left(\frac{\pi}{2}, 0, 0, 0, 0\right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

which represents the Hadamard gate

$$BS_{Rx}\left(\frac{\pi}{2}, 0, 0, 0, 0\right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix}$$

$$BS_{Ry}\left(\frac{\pi}{2}, 0, 0, 0, 0\right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$$

All these conventions are equivalent in the sense that one can be transformed into another.

Now, let us clarify a subtlety that was previously overlooked. The Perceval implementation is as follows:

$$BS_{Rx}(\theta, \Phi_{tl}, \Phi_{bl}, \Phi_{tr}, \Phi_{br}) = \begin{pmatrix} e^{i \cdot (\Phi_{tl} + \Phi_{tr})} \cos\left(\frac{\theta}{2}\right) & i \cdot e^{i \cdot (\Phi_{bl} + \Phi_{tr})} \sin\left(\frac{\theta}{2}\right) \\ i \cdot e^{i \cdot (\Phi_{br} + \Phi_{tl})} \sin\left(\frac{\theta}{2}\right) & e^{i \cdot (\Phi_{bl} + \Phi_{br})} \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

$$BS_H(\theta, \Phi_{tl}, \Phi_{bl}, \Phi_{tr}, \Phi_{br}) = \begin{pmatrix} e^{i \cdot (\Phi_{tl} + \Phi_{tr})} \cos\left(\frac{\theta}{2}\right) & e^{i \cdot (\Phi_{bl} + \Phi_{tr})} \sin\left(\frac{\theta}{2}\right) \\ e^{i \cdot (\Phi_{br} + \Phi_{tl})} \sin\left(\frac{\theta}{2}\right) & -e^{i \cdot (\Phi_{bl} + \Phi_{br})} \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

$$BS_{Ry}(\theta, \Phi_{tl}, \Phi_{bl}, \Phi_{tr}, \Phi_{br}) = \begin{pmatrix} e^{i \cdot (\Phi_{tl} + \Phi_{tr})} \cos\left(\frac{\theta}{2}\right) & -e^{i \cdot (\Phi_{bl} + \Phi_{tr})} \sin\left(\frac{\theta}{2}\right) \\ e^{i \cdot (\Phi_{br} + \Phi_{tl})} \sin\left(\frac{\theta}{2}\right) & e^{i \cdot (\Phi_{bl} + \Phi_{br})} \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

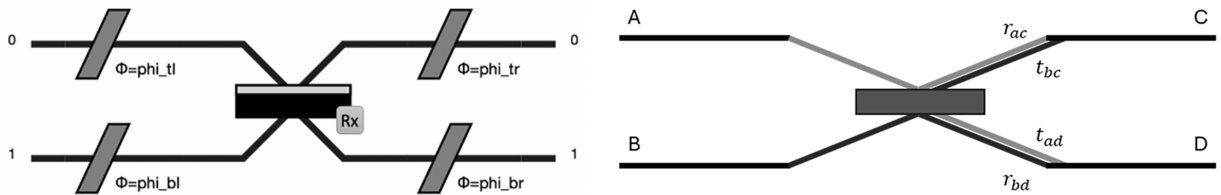
With the special cases:

$$BS_H(\theta, 0, 0, 0, 0) = BS_H(\theta) = \frac{1}{\sqrt{2}} \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & -\cos\left(\frac{\theta}{2}\right) \end{pmatrix} \tag{13}$$

$$BS_{Rx}(\theta, 0, 0, 0, 0) = BS_{Rx}(\theta) = \frac{1}{\sqrt{2}} \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & i \cdot \sin\left(\frac{\theta}{2}\right) \\ i \cdot \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \tag{14}$$

$$BS_{Ry}(\theta, 0, 0, 0, 0) = BS_{Ry}(\theta) = \frac{1}{\sqrt{2}} \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \tag{15}$$

It should be noted that the library allows passing 4 phases as parameters, which differs from the 3 phases described previously. Fig. 5 shows on the left the placement of these four phases the library uses, to be compared with the three parameters of the generic beam splitter on the right we described.



**Figure 5:** Differences between the phase parameters of the beam splitter provided by the Perceval library [27] and those considered previously.

The phases considered are those associated with each complex coefficient describing the action of the beam splitter, whereas Perceval adds a phase to each mode (inputs and outputs) that can be controlled using

phase shifters (described in the next section). This allows them to obtain a more tunable beam splitter. By default, in the Quandela library, the parameters are  $\theta = \frac{\pi}{2}$  and  $\phi = 0$  for all  $\phi$ , so that:

$$BS_H(\theta) = \frac{1}{\sqrt{2}} \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & -\cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

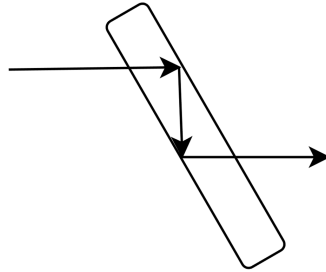
$$BS_H() = BS_H\left(\frac{\pi}{2}, 0, 0, 0, 0\right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

The notation  $BS_H()$  corresponds to calling the procedure  $BS_H()$  without specifying the parameters.

### 3 Photonic Circuit Representation and Two Other Optical Components

#### 3.1 The Phase Shifter Component

Apart from the beam splitter, there is a second essential optical component that enables transformations on Fock states. One can also use phase shifters: a transparent plate acting on a single input mode and allowing the phase of the output mode to be changed if a photon is present. Physically, a phase shifter is a device that “slows down” the photon, creating a phase shift relative to the rest of the system. There are several ways to implement it. One way is simply to place a transparent plate in the photon’s path, as stressed in the figure below (Fig. 6).



**Figure 6:** Representation of a type of physical phase shifter.

The phase shifter acts on a single input mode and produces a single output mode. In other words, its action can be described in terms of Fock states. Recall that the basis states of qubits are encoded on two modes by

$$|0\rangle = |10\rangle_{01}$$

And

$$|1\rangle = |01\rangle_{01}$$

Let  $PS_p(\phi)$  denote the operator representing the application of a phase shifter with phase  $\phi$  to mode  $p$ . Its action on Fock states can be described as follows:

$$PS_p(\phi) |0\rangle_p \mapsto |0\rangle_p \text{ and } PS_p(\phi) |1\rangle_p \mapsto e^{i\phi} |1\rangle_p$$

Its action on Fock states can be described as follows:

$$PS_1(\phi) |0\rangle = PS_1(|10\rangle_{01}) = |10\rangle_{01} = |0\rangle$$

This time, applying it to the qubit |1⟩:

$$PS_1(\phi) |1\rangle = PS_1(\phi) |01\rangle_{01} = e^{i\phi} |01\rangle_{01} = e^{i\phi} |1\rangle$$

We thus have the action on qubits:

$$PS_0(\phi) = \begin{pmatrix} e^{i\phi} & 0 \\ 0 & 1 \end{pmatrix} \text{ and } PS_1(\phi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$$

**Reminders**

According to the definition of  $R_z$ :

$$R_z(\phi) = e^{-\frac{iZ\theta}{2}} = \begin{pmatrix} e^{-\frac{i\phi}{2}} & 0 \\ 0 & e^{\frac{i\phi}{2}} \end{pmatrix}$$

This corresponds (up to a global phase) to the  $P$  and  $R_1$  classical quantum gates:

$$P(\phi) = R_1(\phi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$$

The action of  $PS_1(\phi)$  is exactly that of the gate  $P(\phi) = e^{\frac{i\phi}{2}} R_z(\phi)$ , and in matrix form:

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix} = e^{\frac{i\phi}{2}} \begin{pmatrix} e^{-\frac{i\phi}{2}} & 0 \\ 0 & e^{\frac{i\phi}{2}} \end{pmatrix}$$

We also immediately obtain:

$$PS_1(\phi) = R_1(\phi) = P(\phi)$$

From a Fock state perspective, we have the following:

$$PS_1(\phi) |0\rangle = PS_1(\phi) |10\rangle_{01} \quad \underbrace{=} \quad |10\rangle_{01} = |0\rangle$$

*There is no photon  
on mode 1 and  
PS<sub>1</sub> acts on mode 1*

$$PS_1(\phi) |1\rangle = PS_1(\phi) |01\rangle_{01} \quad \underbrace{=} \quad e^{i\theta} |01\rangle_{01} = e^{i\theta} |1\rangle$$

*There is a photon  
on mode 1 and  
PS<sub>1</sub> acts on mode 1*

**Theorem 4:** Applying a phase shift to each mode is equivalent to applying a global phase shift to the qubit. Let  $|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$  be a qubit in an arbitrary state, then:

$$e^{i\cdot\phi} \cdot |\psi\rangle = PS_0(\phi) \cdot PS_1(\phi) \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

**Proof:**

$$|\psi_0\rangle = PS_0(\phi) \cdot PS_1(\phi) \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

$$|\psi_2\rangle = PS_0(\phi) \cdot \begin{pmatrix} \alpha \\ e^{i\cdot\phi}\beta \end{pmatrix}$$

$$|\psi_3\rangle = \begin{pmatrix} e^{i\cdot\phi}\alpha \\ e^{i\cdot\phi}\beta \end{pmatrix} = e^{i\cdot\phi} \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

□

### 3.2 Definition of the Permutation Component

The permutation “component” is an optical element whose action is very simple: it exchanges the number of photons between the modes given as parameters. It can be realized by redirecting the optical modes through the use of mirrors for instance.

For example, for  $PERM_{10}$ :

$$|n_0 n_1\rangle_{01} \mapsto |n_1 n_0\rangle_{01}$$

Another example involving more modes:

$$PERM_{2301}(|n_0 n_1 n_2 n_3\rangle_{0123}) = |n_2 n_3 n_0 n_1\rangle_{0123}$$

2301 corresponds to the permutation

$$\begin{array}{cccc} 0 & 1 & 2 & 3 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 3 & 0 & 1 \end{array}$$

One can observe that  $PERM_{01} = Id$  as it corresponds to the permutation:

$$\begin{array}{cc} 0 & 1 \\ \downarrow & \downarrow \\ 0 & 1 \end{array}$$

### 3.3 Graphical Representations of the Beam Splitter and the Phase Shifter

Photonic systems can be described using various graphical representations of their optical components. To recognize them, here are a few examples. In the following diagrams, each line represents an optical mode. Let us begin with the beam splitter, shown in Fig. 7 and let us continue with the phase shifter in Fig. 8.

We can also give a representation of the  $PERM$  component truly representing how the photons move from one mode to another. Like shown in Fig. 9.

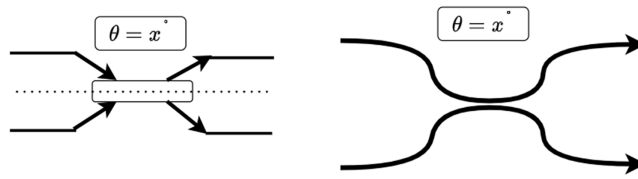


Figure 7: Two representations of the beam splitter.

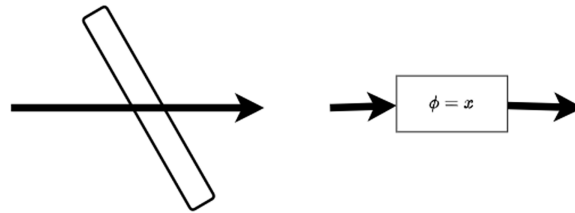


Figure 8: Two representations of the phase shifter.

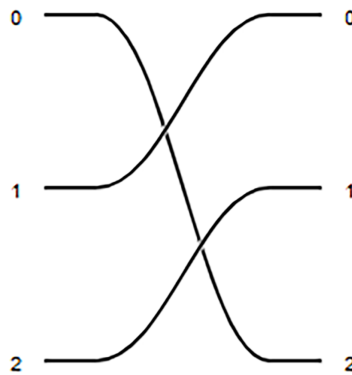


Figure 9: A representation of  $PERM_{201}$ .

### 3.4 Graphical Representations of a Photonic Circuit

These optical components can then be used in an optical circuit. The optical circuit will be composed of different lines representing different modes.

Then the components are placed onto the circuit. Like in gate model of quantum computation, the components are applied in order from left to right. To keep this in mind, we can imagine photons coming from the left of the circuit and moving along the lines until it reaches the end of the circuit.

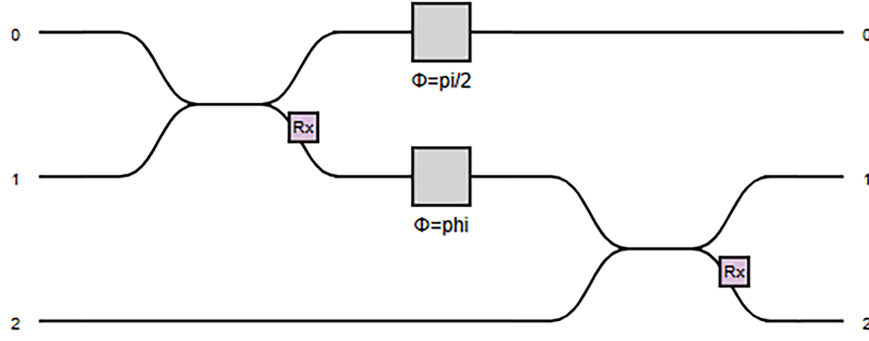
An example of circuit with two Phase Shifters and two Beam Splitters  $BS_{Rx}$  is given in Fig. 10.

Note that the circuit does not directly produce the input Fock state. In some cases, we place it on the left side of the circuit.

## 4 Modelling Single-Qubit Gates

### 4.1 From a Unitary on Fock States to a Unitary on Qubits

Before starting to describe how to rebuild the quantum gates on qubits, it is important to understand the link between a unitary acting on Fock states and a unitary acting on qubits. To understand this, we will need to use the creation and annihilation operators.



**Figure 10:** Representation of a quantum photonic circuit made with the Perceval library.

Let  $U$  be a unitary matrix associated with an optical component or an optical circuit composed of two input modes 0 and 1 and two output modes  $out_0$  and  $out_1$ . The effect of  $U$  is described by the following relation:

$$\begin{pmatrix} \hat{a}_{out_0} \\ \hat{a}_{out_1} \end{pmatrix} = U \begin{pmatrix} \hat{a}_0 \\ \hat{a}_1 \end{pmatrix}$$

where  $\hat{a}_m$  is the annihilation operator on mode  $m$ .

To better understand what happens, we multiply both sides on the left by  $U^\dagger$ .

$$U^\dagger \begin{pmatrix} \hat{a}_{out_0} \\ \hat{a}_{out_1} \end{pmatrix} = \begin{pmatrix} \hat{a}_0 \\ \hat{a}_1 \end{pmatrix}$$

We then take the conjugate transpose of the equation:

$$\left( U^\dagger \begin{pmatrix} \hat{a}_{out_0} \\ \hat{a}_{out_1} \end{pmatrix} \right)^\dagger = \begin{pmatrix} \hat{a}_0 \\ \hat{a}_1 \end{pmatrix}^\dagger$$

$$\Leftrightarrow (\hat{a}_{out_0}^\dagger \quad \hat{a}_{out_1}^\dagger) (U^\dagger)^\dagger = (\hat{a}_0^\dagger \quad \hat{a}_1^\dagger)$$

Applying the transposition  $()^T$  gives:

$$U^T \begin{pmatrix} \hat{a}_{out_0}^\dagger \\ \hat{a}_{out_1}^\dagger \end{pmatrix} = \begin{pmatrix} \hat{a}_0^\dagger \\ \hat{a}_1^\dagger \end{pmatrix}$$

$$\text{with } U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}$$

$$\Leftrightarrow \begin{cases} \hat{a}_0^\dagger = U_{11}\hat{a}_{out_0}^\dagger + U_{21}\hat{a}_{out_1}^\dagger \\ \hat{a}_1^\dagger = U_{12}\hat{a}_{out_0}^\dagger + U_{22}\hat{a}_{out_1}^\dagger \end{cases}$$

This if we rename the output mode with the same name as the corresponding input mode, we can rewrite the previous relations:

$$\Leftrightarrow \begin{cases} \hat{a}_0^\dagger \xrightarrow{U} U_{11}\hat{a}_0^\dagger + U_{21}\hat{a}_1^\dagger \\ \hat{a}_1^\dagger \xrightarrow{U} U_{12}\hat{a}_0^\dagger + U_{22}\hat{a}_1^\dagger \end{cases}$$

We now have an expression to describe how an operator  $\hat{a}_{0/1}^\dagger$  is transformed by the circuit/component represented by  $U$ .

Let us consider an example with for instance:

$$U = BS_H\left(\frac{\pi}{2}\right) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \left( BS_H\left(\frac{\pi}{2}\right) \right)^T$$

Replacing  $U$  by  $BS_H\left(\frac{\pi}{2}\right)$  in the previous equation:

$$\Leftrightarrow \begin{cases} \hat{a}_0^\dagger = \frac{1}{\sqrt{2}} (\hat{a}_{out_0}^\dagger + \hat{a}_{out_1}^\dagger) \\ \hat{a}_1^\dagger = \frac{1}{\sqrt{2}} (\hat{a}_{out_0}^\dagger - \hat{a}_{out_1}^\dagger) \end{cases}$$

We can rewrite the system as:

$$\Leftrightarrow \begin{cases} \hat{a}_0^\dagger \xrightarrow{BS_H\left(\frac{\pi}{2}\right)} \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger + \hat{a}_1^\dagger) \\ \hat{a}_1^\dagger \xrightarrow{BS_H\left(\frac{\pi}{2}\right)} \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger - \hat{a}_1^\dagger) \end{cases}$$

Then we use theses relations to describe the effect of the component on a Fock state starting from:

$$|11\rangle_{01} = \hat{a}_0^\dagger \hat{a}_1^\dagger |00\rangle_{01}$$

We use the relations from the previous system to get the result after the component:

$$\begin{aligned} & \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger + \hat{a}_1^\dagger) \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger - \hat{a}_1^\dagger) |00\rangle_{01} = \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger \hat{a}_0^\dagger + \hat{a}_1^\dagger \hat{a}_1^\dagger) |00\rangle_{01} \\ & = \frac{1}{\sqrt{2}} |00\rangle_{01} + \frac{1}{\sqrt{2}} |11\rangle_{01} \end{aligned}$$

Starting with two photons on different modes, applying a beamsplitter to both modes results in getting either two photons in the first output mode or two photons in the second output mode. Never in between. This is known as the Hong Ou Mandel effect [28].

### 4.2 Realization of the X Gate

Our goal is to construct an X gate using the permutation component (Fig. 11). This representation should not be confused with that of a Beam Splitter. By swapping the modes and maintaining the convention that “mode 0 is the top mode in the diagram” we apply the transformation

$$PERM_{10}: |n_0 n_1\rangle_{01} \mapsto |n_1 n_0\rangle_{01}$$

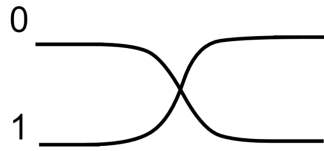
For an arbitrary qubit state  $(\alpha |0\rangle + \beta |1\rangle)$ , this corresponds to the action of the X–gate:

$$PERM_{10} (\alpha |0\rangle + \beta |1\rangle) = PERM_{10} (\alpha |10\rangle_{01} + \beta |01\rangle_{01})$$

$$= \alpha PERM_{10} (|10\rangle_{01}) + \beta PERM_{10} (|01\rangle_{01})$$

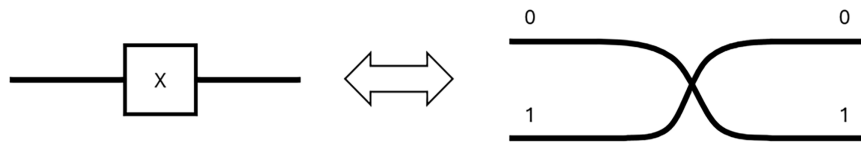
$$= \alpha |01\rangle_{01} + \beta |10\rangle_{01}$$

$$= \alpha |1\rangle + \beta |0\rangle$$



**Figure 11:** Permutation of the two modes of a Fock state realizing the X gate.

We indeed recover the effect of the Pauli-X gate (Fig. 12).



**Figure 12:** Equivalent of the X gate on qubits using PERM on the Fock state.

### 4.3 Realization of the SWAP Gate

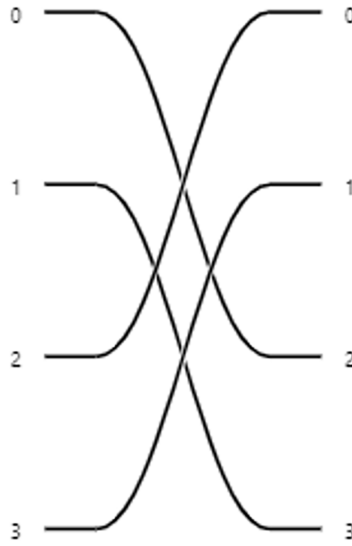
We now focus on the standard SWAP gate applied to qubits. Compared to the X gate described earlier, we must redefine the permutation over all four modes, rather than just two.

To define a gate that swaps qubit 1 with qubit 2, i.e., performs the following operation:

$$SWAP |10\rangle = |01\rangle$$

One can use the following permutations:

$$\begin{matrix} 0 & 1 & 2 & 3 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 3 & 0 & 1 \end{matrix}$$



**Figure 13:** Mode permutations to define a permutation gate.

A visualization is given in Fig. 13.

On an arbitrary 2-qubit state:

$$\alpha |00\rangle + \beta |01\rangle + \gamma |10\rangle + \mu |11\rangle$$

We apply  $PERM_{2301}$ :

$$\begin{aligned} & PERM_{2301} (\alpha |00\rangle + \beta |01\rangle + \gamma |10\rangle + \mu |11\rangle) \\ &= \alpha PERM_{2301} |00\rangle + \beta PERM_{2301} |01\rangle + \gamma PERM_{2301} |10\rangle + \mu PERM_{2301} |11\rangle \end{aligned}$$

On Fock states

$$\begin{aligned} &= \alpha PERM_{2301} |0101\rangle_{0123} + \beta PERM_{2301} |0110\rangle_{0123} + \gamma PERM_{2301} |1001\rangle_{0123} + \mu PERM_{2301} |1010\rangle_{0123} \\ &= \alpha |0101\rangle_{0123} + \beta |1001\rangle_{0123} + \gamma |0110\rangle_{0123} + \mu |1010\rangle_{0123} \\ &= \alpha |00\rangle + \beta |01\rangle + \gamma |01\rangle + \mu |11\rangle \end{aligned}$$

This is exactly the expected action of *SWAP* on qubits.

#### 4.4 Realization of the HGate

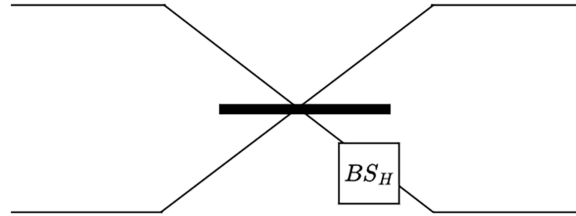
Let us consider the expression obtained for the Beam Splitter introduced in Section 2 (see (13)) :

$$BS_H = \frac{1}{\sqrt{2}} \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & -\cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

If we choose  $\theta = \frac{\pi}{2}$  and consider  $\phi = 0$  for all the phases involved, then

$$BS_H \left( \theta = \frac{\pi}{2} \right) = H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

We immediately obtain the  $H$  gate and a representation is introduced in Fig. 14.



**Figure 14:** Realization of the H-gate with a beam splitter.

But this  $BS_H$  matrix is not directly acting on qubits; here is how to analyze how it acts on qubits:

The matrix of  $BS_H$  acts on Fock states, we use the formalism of the Section 4.1 which gave:

$$\Leftrightarrow \begin{cases} \hat{a}_0^\dagger \xrightarrow{BS_H(\frac{\pi}{2})} \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger + \hat{a}_1^\dagger) \\ \hat{a}_1^\dagger \xrightarrow{BS_H(\frac{\pi}{2})} \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger - \hat{a}_1^\dagger) \end{cases}$$

Let us describe the effect on the basis states of a qubit:

$$\begin{aligned} |0\rangle &= |10\rangle_{01} = \hat{a}_0^\dagger |00\rangle_{01} \xrightarrow{BS_H(\frac{\pi}{2})} \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger + \hat{a}_1^\dagger) |00\rangle_{01} = \frac{1}{\sqrt{2}} (|01\rangle_{01} + |10\rangle_{01}) = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \\ |1\rangle &= |01\rangle_{01} = \hat{a}_1^\dagger |00\rangle_{01} \xrightarrow{BS_H(\frac{\pi}{2})} \frac{1}{\sqrt{2}} (\hat{a}_0^\dagger - \hat{a}_1^\dagger) |00\rangle_{01} = \frac{1}{\sqrt{2}} (|01\rangle_{01} - |10\rangle_{01}) = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \end{aligned}$$

This is exactly what we would expect of H applied onto qubits.

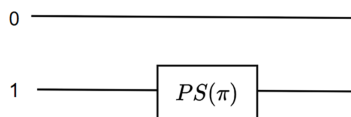
#### 4.5 Realization of Z Gate

To implement Z, one must apply a phase shift of  $\pi$  on mode 1 which is initially in the state  $|01\rangle_{01} = |1\rangle$  leading to:

$$PS_1(\pi) |0\rangle = PS_1(\pi) \cdot (|10\rangle_{01}) = |10\rangle_{01} = |0\rangle$$

$$PS_1(\pi) |1\rangle = PS_1(\pi) \cdot (|01\rangle_{01}) = -|01\rangle_{01} = -|1\rangle$$

This describes the action of the Z gate on all the basis states of a qubit therefore describing completely the action of Z. A representation of the circuit is given in Fig. 15.



**Figure 15:** Realization of the Z-gate with a phase shifter.

#### 4.6 Realization of the Y Gate

To implement the Y gate, we use will use 3 optical components:

- A Beam Splitter
- 2 Phase shifters

##### Reminder

$$Y|0\rangle = Y \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ i \end{pmatrix} = i \cdot |1\rangle$$

$$Y|1\rangle = Y \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -i \\ 0 \end{pmatrix} = -i \cdot |0\rangle$$

Using this reminder, let us prove the following statement:

##### Theorem 5:

$$Y = PS_1 \left( \frac{\pi}{2} \right) \cdot PS_0 \left( -\frac{\pi}{2} \right) \cdot PERM_{10}$$

**Proof:** Let us consider  $|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$  an arbitrary qubit state, we will show that the action of the circuit and the Y gate are the same.

We have:

$$\begin{aligned} & PS_1 \left( \frac{\pi}{2} \right) \cdot PS_0 \left( -\frac{\pi}{2} \right) \cdot PERM_{10} (\alpha|0\rangle + \beta|1\rangle) \\ &= PS_1 \left( \frac{\pi}{2} \right) \cdot PS_0 \left( -\frac{\pi}{2} \right) \cdot PERM_{10} (\alpha|10\rangle_{01} + \beta|01\rangle_{01}) \\ &= PS_1 \left( \frac{\pi}{2} \right) \cdot PS_0 \left( -\frac{\pi}{2} \right) \cdot (\alpha|01\rangle_{01} + \beta|10\rangle_{01}) \\ &= PS_1 \left( \frac{\pi}{2} \right) \cdot PS_0 \left( -\frac{\pi}{2} \right) \cdot (\alpha|1\rangle + \beta|0\rangle) \\ &= PS_1 \left( \frac{\pi}{2} \right) \cdot PS_0 \left( -\frac{\pi}{2} \right) \begin{pmatrix} \beta \\ \alpha \end{pmatrix} \\ &= PS_1 \left( \frac{\pi}{2} \right) \cdot \begin{pmatrix} \beta e^{-i \cdot t \cdot \frac{\pi}{2}} \\ \alpha \end{pmatrix} \\ &= \begin{pmatrix} -i\beta \\ i\alpha \end{pmatrix} \end{aligned}$$

And by consequence

$$PS_1 \left( \frac{\pi}{2} \right) \cdot PS_0 \left( -\frac{\pi}{2} \right) \cdot PERM_{10} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} -i\beta \\ i\alpha \end{pmatrix}$$

And as:

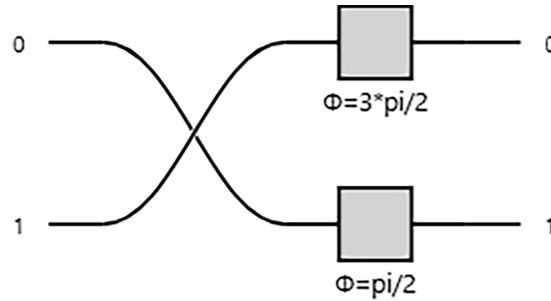
$$Y \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} -i\beta \\ i\alpha \end{pmatrix}$$

So we conclude that

$$PS_1 \left( \frac{\pi}{2} \right) . PS_0 \left( -\frac{\pi}{2} \right) . PERM_{10} = Y$$

□

This expression corresponds to the photonic circuit introduced in Fig. 16.



**Figure 16:** Photonic circuit for the Y gate on a qubit.

#### 4.7 Realization of the $RX(\theta)$ Gate

Recall that according to (14), the definition of  $RX(\theta)$  is:

$$RX(\theta) = e^{-\frac{iX\theta}{2}} = \begin{pmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}$$

The Perceval computing library natively provides a gate named  $BS_{Rx}$  as the default convention for the Beam Splitter. However, the definition of the gate provided by the library is

$$BS_{Rx} = \begin{pmatrix} \cos \left( \frac{\theta}{2} \right) & i \sin \left( \frac{\theta}{2} \right) \\ i \sin \left( \frac{\theta}{2} \right) & \cos \left( \frac{\theta}{2} \right) \end{pmatrix}$$

and does not directly correspond to the usual  $RX$  gate.

**Theorem 6:**

$$RX(\theta) = PS_0(\pi) . BS_{Rx}(\theta) . PS_0(\pi)$$

**Proof:** Once again, let us consider  $|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$  an arbitrary qubit state, We will show that the action of the circuit is the same as that of the  $RX$  gate.

We have:

$$RX(\theta) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

$$= \begin{pmatrix} \alpha \cos \frac{\theta}{2} - i \beta \sin \frac{\theta}{2} \\ -i \alpha \sin \frac{\theta}{2} + \beta \cos \frac{\theta}{2} \end{pmatrix}$$

and we have:

$$PS_0(\pi) \cdot BS_{Rx}(\theta) \cdot PS_0(\pi) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = PS_0(\pi) \cdot BS_{Rx}(\theta) \cdot \begin{pmatrix} -\alpha \\ \beta \end{pmatrix}$$

Because

$$BS_{Rx}(\theta) = \begin{pmatrix} \cos \frac{\theta}{2} & i \sin \frac{\theta}{2} \\ i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}$$

We have:

$$PS_0(\pi) \cdot BS_{Rx}(\theta) \cdot PS_0(\pi) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = PS_0(\pi) \cdot \begin{pmatrix} -\alpha \cos \frac{\theta}{2} + i \beta \sin \frac{\theta}{2} \\ -i \alpha \sin \frac{\theta}{2} + \beta \cos \frac{\theta}{2} \end{pmatrix}$$

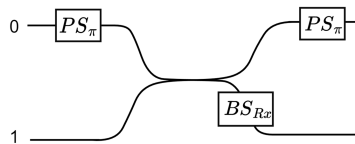
$$PS_0(\pi) \cdot BS_{Rx}(\theta) \cdot PS_0(\pi) \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha \cos \frac{\theta}{2} - i \beta \sin \frac{\theta}{2} \\ -i \alpha \sin \frac{\theta}{2} + \beta \cos \frac{\theta}{2} \end{pmatrix}$$

Therefore:

$$PS_0(\pi) \cdot BS_{Rx}(\theta) \cdot PS_0(\pi) = RX(\theta)$$

□

The photonic circuit representation is shown in Fig. 17.



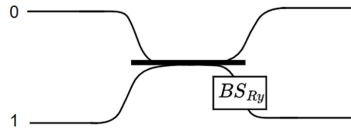
**Figure 17:** Quantum circuit for the  $RX(\theta)$  gate on qubits.

#### 4.8 Realization of the $R_Y(\theta)$ Gate

Let us note that:

$$R_Y(\theta) = \begin{pmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}$$

The definition of the component  $BS_{Ry}$  provided by the library [15] is exactly that of the standard  $R_Y$  gate and the photonic representation of the gate is shown in Fig. 18.



**Figure 18:** Representation of the photonic circuit associated with  $R_Y$ .

For completion, we can use the formalism of Section 4.1 on  $BS_{Ry}$ , thus get

$$\Leftrightarrow \begin{cases} \hat{a}_0^\dagger \xrightarrow{BS_{Ry}} \cos \frac{\theta}{2} \hat{a}_0^\dagger + \sin \frac{\theta}{2} \hat{a}_1^\dagger \\ \hat{a}_1^\dagger \xrightarrow{R_Y(\theta)} -\sin \frac{\theta}{2} \hat{a}_0^\dagger + \cos \frac{\theta}{2} \hat{a}_1^\dagger \end{cases}$$

Starting from the two basis states of a qubit, we get:

$$|0\rangle = |10\rangle_{01} = \hat{a}_0^\dagger |00\rangle_{01} \xrightarrow{BS_{Ry}} \left( \cos \frac{\theta}{2} \hat{a}_0^\dagger + \sin \frac{\theta}{2} \hat{a}_1^\dagger \right) |00\rangle_{01} = \cos \frac{\theta}{2} |10\rangle_{01} + \sin \frac{\theta}{2} |01\rangle_{01} = \cos \frac{\theta}{2} |0\rangle + \sin \frac{\theta}{2} |1\rangle$$

$$|1\rangle = |01\rangle_{01} = \hat{a}_1^\dagger |00\rangle_{01} \xrightarrow{BS_{Ry}} \left( -\sin \frac{\theta}{2} \hat{a}_0^\dagger + \cos \frac{\theta}{2} \hat{a}_1^\dagger \right) |00\rangle_{01} = -\sin \frac{\theta}{2} |10\rangle_{01} + \cos \frac{\theta}{2} |01\rangle_{01} = -\sin \frac{\theta}{2} |0\rangle + \cos \frac{\theta}{2} |1\rangle$$

This is exactly how the  $Ry$  gate acts on qubits.

#### 4.9 Realization of the $R_Z(\theta)$ Gate

To rebuild the  $R_Z$  rotation gate, we start by recalling that

$$R_Z(\theta) = \begin{pmatrix} e^{-i \cdot \frac{\theta}{2}} & 0 \\ 0 & e^{i \cdot \frac{\theta}{2}} \end{pmatrix}$$

**Theorem 7:**

$$R_Z(\theta) = BS_H \cdot PS_0(\pi) \cdot BS_{Rx}(\theta) \cdot PS_0(\pi) \cdot BS_H$$

**Proof:** Let us consider  $|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ . We have:

$$RZ(\theta) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha e^{-i\frac{\theta}{2}} \\ \beta e^{i\frac{\theta}{2}} \end{pmatrix}$$

and

$$\begin{aligned} BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi).BS_H \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix} &= BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi) \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \\ &= BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi) \frac{1}{\sqrt{2}} \begin{pmatrix} \alpha + \beta \\ \alpha - \beta \end{pmatrix} \end{aligned}$$

Because

$$PS_0(\pi)|0\rangle = PS_0(\pi)|10\rangle_{01} = e^{i\pi}|10\rangle_{01} = -|0\rangle$$

$$PS_0(\pi)|1\rangle = PS_0(\pi)|01\rangle_{01} = |01\rangle_{01} = |1\rangle$$

We have

$$BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi).BS_H \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = BS_H.PS_0(\pi).BS_{Rx}(\theta) \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} -(\alpha + \beta) \\ \alpha - \beta \end{pmatrix}$$

Because

$$BS_{Rx}(\theta) = \begin{pmatrix} \cos \frac{\theta}{2} & i \sin \frac{\theta}{2} \\ i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix}$$

We have

$$BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi).BS_H \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = BS_H.PS_0(\pi) \cdot \begin{pmatrix} \cos \frac{\theta}{2} & i \sin \frac{\theta}{2} \\ i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix} \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} -(\alpha + \beta) \\ \alpha - \beta \end{pmatrix}$$

$$BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi).BS_H \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = BS_H.PS_0(\pi) \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} -(\alpha + \beta) \cos \frac{\theta}{2} + i(\alpha - \beta) \sin \frac{\theta}{2} \\ -i(\alpha + \beta) \sin \frac{\theta}{2} + (\alpha - \beta) \cos \frac{\theta}{2} \end{pmatrix}$$

and

$$BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi).BS_H \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = BS_H \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} (\alpha + \beta) \cos \frac{\theta}{2} - i(\alpha - \beta) \sin \frac{\theta}{2} \\ -i(\alpha + \beta) \sin \frac{\theta}{2} + (\alpha - \beta) \cos \frac{\theta}{2} \end{pmatrix}$$

and

$$\begin{aligned} BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi).BS_H \cdot \begin{pmatrix} \alpha \\ \beta \end{pmatrix} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} (\alpha + \beta) \cos \frac{\theta}{2} - i(\alpha - \beta) \sin \frac{\theta}{2} \\ -i(\alpha + \beta) \sin \frac{\theta}{2} + (\alpha - \beta) \cos \frac{\theta}{2} \end{pmatrix} \\ &= \frac{1}{2} \begin{pmatrix} (\alpha + \beta) \cos \frac{\theta}{2} - i(\alpha - \beta) \sin \frac{\theta}{2} - i(\alpha + \beta) \sin \frac{\theta}{2} + (\alpha - \beta) \cos \frac{\theta}{2} \\ (\alpha + \beta) \cos \frac{\theta}{2} - i(\alpha - \beta) \sin \frac{\theta}{2} + i(\alpha + \beta) \sin \frac{\theta}{2} - (\alpha - \beta) \cos \frac{\theta}{2} \end{pmatrix} \\ &= \frac{1}{2} \begin{pmatrix} 2\alpha \cos \frac{\theta}{2} - 2i\alpha \sin \frac{\theta}{2} \\ 2\beta \cos \frac{\theta}{2} + 2\beta i \sin \frac{\theta}{2} \end{pmatrix} \\ &= \begin{pmatrix} \alpha \cos \frac{\theta}{2} - i\alpha \sin \frac{\theta}{2} \\ \beta \cos \frac{\theta}{2} + \beta i \sin \frac{\theta}{2} \end{pmatrix} \end{aligned}$$

So

$$BS_H.PS_0(\pi).BS_{Rx}(\theta).PS_0(\pi).BS_H = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix} = RZ(\theta) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha e^{-i\frac{\theta}{2}} \\ \beta e^{i\frac{\theta}{2}} \end{pmatrix}$$

□

The photonic representation of the gate is shown in Fig. 19.

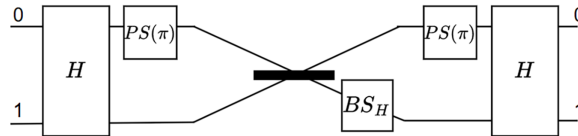
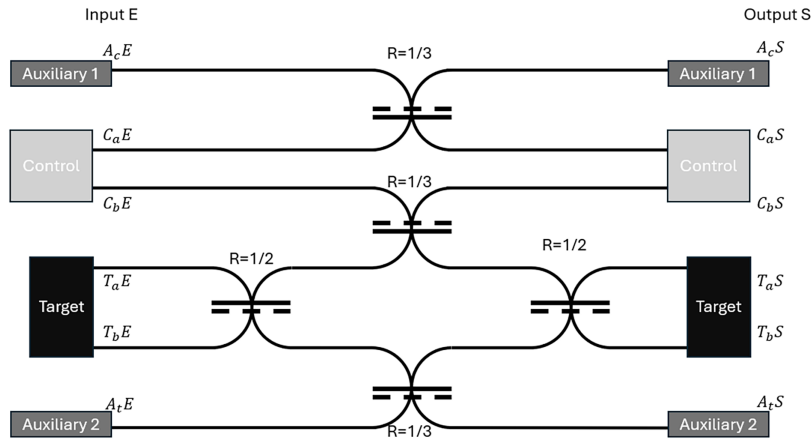


Figure 19: Representation of the photonic circuit associated to RZ.

## 5 Modelling Controlled Gates

### 5.1 Realization of CX According to [14]

Consider the definition of CX as defined by [27] and originally proposed by [14], with a description provided in Fig. 20.



**Figure 20:** Ralph CNOT circuit from [14].

This figure presents a method for implementing a CNOT gate based on 6 modes representing two qubits and two auxiliary modes. The modes  $C_a$  and  $C_b$  correspond to the control qubit in the CNOT we wish to realize. Thus  $|01\rangle_{C_a C_b}$  represents the Fock state associated with the control qubit in the state  $|1\rangle = |01\rangle_{C_a C_b}$ . Similarly, we define  $T_a$  and  $T_b$  as the modes for the “Target” qubit. Additionally, we have two auxiliary modes  $A_c$  and  $A_t$  which act on the control and target qubits, respectively. Both auxiliary modes are initialized in the states  $|0\rangle_{A_c}$  and  $|0\rangle_{A_t}$ .

To explain the CNOT operation, we need to distinguish between the input and output parts of the CNOT for a given mode. For instance, we distinguish  $C_a E$ , the first input mode for the control qubit, from  $C_a S$ , the first output mode for the control qubit. Making this distinction allows us to express the output modes as functions of the input modes.

Before diving into the explanation of the CNOT operation, let us highlight two important elements:

- Each Beam Splitter has a dashed side, which is the side that will receive a phase shift (the side that will receive the minus sign in the chosen convention).
- We consider two types of Beam Splitters: some with a reflectivity of  $R = \frac{1}{2}$  (symmetric Beam Splitters) and others with a reflectivity of  $R = \frac{1}{3}$  (asymmetric Beam Splitters).

**Notes**

It is important to recall that for a Beam Splitter with reflectivity  $\alpha$ , and whose graphical representation corresponds to the dashed side facing upward, two output states are defined:

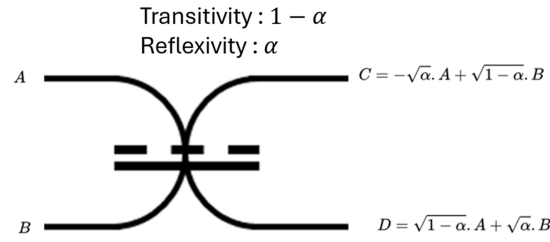
- The state  $C$  undergoes a phase shift of value  $-1$  since it is on the dashed side, and the fraction of  $A$  that is reflected by the Beam Splitter is  $-\sqrt{\alpha}$ ;
- The fraction of  $B$  that “passes through” the Beam Splitter is  $\sqrt{1 - \alpha}$ .

This leads to the output  $C = -\sqrt{\alpha}.A + \sqrt{1 - \alpha}.B$  with the phase of  $B$  unchanged since  $B$  corresponds to the side of the Beam Splitter without the dashed lines (Fig. 21).

By observing Fig. 20, we can establish the following relations:

$$A_c S = \frac{1}{\sqrt{3}}(-A_c E + \sqrt{2}C_a E)$$

$$C_a S = \frac{1}{\sqrt{3}}(C_a E + \sqrt{2}A_c E)$$



**Figure 21:** Output amplitudes of a beam splitter with reflectivity  $\alpha$ .

$$C_a S = \frac{1}{\sqrt{3}}(\sqrt{2}C_a E + A_c E)$$

$$C_b S = -\frac{1}{\sqrt{3}}C_b E + \frac{1}{\sqrt{2}} \times \left( \frac{\sqrt{2}}{\sqrt{3}}T_a E + \frac{\sqrt{2}}{\sqrt{3}}T_b E \right) = \frac{1}{\sqrt{3}}(T_a E + T_b E - C_b E)$$

$$T_a S = \frac{\sqrt{2}}{\sqrt{3}} \times \frac{1}{\sqrt{2}}C_b E + \left( \underbrace{\frac{1}{\sqrt{2}} \times \frac{1}{\sqrt{3}} \times \frac{1}{\sqrt{2}}}_{\text{first path}} + \underbrace{\frac{1}{\sqrt{2}} \times \frac{1}{\sqrt{3}} \times \frac{1}{\sqrt{2}}}_{\text{second path}} \right) T_a E$$

$$+ \left( \underbrace{\frac{1}{\sqrt{2}} \times \frac{1}{\sqrt{3}} \times \frac{1}{\sqrt{2}}}_{\text{first path}} + \underbrace{\left( -\frac{1}{\sqrt{2}} \right) \times \frac{1}{\sqrt{3}} \times \frac{1}{\sqrt{2}}}_{\text{second path}} \right) T_b E + \frac{\sqrt{2}}{\sqrt{3}} \times \frac{1}{\sqrt{2}}A_t E$$

Giving:

$$T_a S = \frac{1}{\sqrt{3}}(C_b E + T_a E + A_t E)$$

$$T_b S = \frac{1}{\sqrt{3}}(C_b E + T_b E - A_t E)$$

$$A_t S = \frac{1}{\sqrt{3}}(T_a E + T_b E - A_t E)$$

### Remark

By choosing a different reflectivity  $R$ , we would obtain different values for each relation.

Example with  $R = \frac{1}{4}$ :

$$T_a S = \frac{1}{2\sqrt{2}}C_b E + \frac{\sqrt{3}}{2}T_a E + \frac{1}{2\sqrt{2}}A_t E$$

### Note 1

The creation and annihilation operators associated with a mode 0 are denoted  $\hat{a}_0^\dagger$  and  $\hat{a}_0$ , respectively, and their actions are as follows:

- The creation operator  $\hat{a}_0^\dagger$  allows the transition from  $|0\rangle_0$  to  $|1\rangle_0$  on the spatial mode 0. For example,  $\hat{a}_0^\dagger$  transitions from  $|00\rangle_{01}$  (vacuum state) to  $|10\rangle_{01}$ .
- The annihilation operator  $\hat{a}_0$  allows the transition from  $|1\rangle_0$  to  $|0\rangle_0$ . For example,  $\hat{a}_0$  transitions from  $|10\rangle_{01}$  to  $|00\rangle_{01}$  (vacuum state).

**Note 2**

Strictly speaking, the creation operator on the output mode  $i$  should be denoted as  $a_i^\dagger$

For convenience and simplicity, we will use the shorthand notation  $(i)^\dagger$  for the creation operator on the output mode  $i$ .

For example:

$$(A_c S)^\dagger \equiv a_{A_c S}^\dagger$$

The creation operator on the output modes can be deduced from the previous relations. For example, we have:

$$\underbrace{(A_c S)^\dagger}_{\text{Creation operator of output mode } A_c S} = \frac{1}{\sqrt{3}}(- (A_c E)^\dagger + \sqrt{2} (C_a E)^\dagger)$$

*Creation operator of output mode  $A_c S$*

The final state of a two-qubit system (4 modes), to which we add the two auxiliary modes that we reorganize for clarity, can be written as:

$$|\psi_{in}\rangle = \underbrace{\alpha |00\rangle + \beta |01\rangle + \gamma |10\rangle + \delta |01\rangle}_{\text{qubits}} \underbrace{|00\rangle_{05}}_{\text{Fockstate of auxiliary modes}}$$

This state can be described in terms of Fock states and creation operators by considering only the 4 output qubits. So the state:

$$|\psi_{in}\rangle = \alpha |1010\rangle_{1234} + \beta |1001\rangle_{1234} + \gamma |0110\rangle_{1234} + \delta |1001\rangle_{1234} \otimes |1010\rangle_{1234}$$

is rewritten:

$$|\psi_{in}\rangle = (\alpha ((C_a S)^\dagger (T_a S)^\dagger |0000\rangle_{1234}) + \beta ((C_a S)^\dagger (T_b S)^\dagger |0000\rangle_{1234}) + \gamma ((C_b S)^\dagger (T_a S)^\dagger |0000\rangle_{1234}) + \delta ((C_b S)^\dagger (T_b S)^\dagger |0000\rangle_{1234})) |00\rangle_{1234}$$

Now, let us consider each term separately.

**For the first term**

$$\alpha ((C_a S)^\dagger (T_a S)^\dagger |0000\rangle_f)$$

$$= \alpha \left( \left( \frac{1}{\sqrt{3}} ((C_a E)^\dagger + \sqrt{2} (A_c E)^\dagger) \right) \times \left( \frac{1}{\sqrt{3}} ((C_b E)^\dagger + (T_a E)^\dagger + (A_t E)^\dagger) \right) |0000\rangle_{1234} \right)$$

$$= \frac{\alpha}{3} \left( \left( (C_a E)^\dagger (C_b E)^\dagger + (C_a E)^\dagger (T_a E)^\dagger + (C_a E)^\dagger (A_t E)^\dagger + \sqrt{2} (A_c E)^\dagger (C_b E)^\dagger + \sqrt{2} (A_c E)^\dagger (T_a E)^\dagger + \sqrt{2} (A_c E)^\dagger (A_t E)^\dagger \right) |0000\rangle_f \right)$$

**For the second term**

$$\begin{aligned} & \beta \left( (C_a S)^\dagger (T_b S)^\dagger |0000\rangle_f \right) \\ &= \beta \left( \left( \frac{1}{\sqrt{3}} \left( (C_a E)^\dagger + \sqrt{2} (A_c E)^\dagger \right) \frac{1}{\sqrt{3}} \left( (C_b E)^\dagger + (T_b E)^\dagger - (A_t E)^\dagger \right) \right) |0000\rangle_f \right) \\ &= \frac{\beta}{3} \left( \left( (C_a E)^\dagger (C_b E)^\dagger + (C_a E)^\dagger (T_b E)^\dagger - (C_a E)^\dagger (A_t E)^\dagger + \sqrt{2} (A_c E)^\dagger (C_b E)^\dagger + \sqrt{2} (A_c E)^\dagger (T_b E)^\dagger - \sqrt{2} (A_c E)^\dagger (A_t E)^\dagger \right) |0000\rangle_f \right) \end{aligned}$$

**For the third term**

$$\begin{aligned} & \gamma \left( (C_b S)^\dagger (T_a S)^\dagger |0000\rangle_f \right) \\ &= \gamma \left( \frac{1}{\sqrt{3}} (T_a E + T_b E - C_b E) \times \frac{1}{\sqrt{3}} (C_b E + T_a E + A_t E) |0000\rangle_f \right) \\ &= \frac{\gamma}{3} \left( \left( (T_a E)^\dagger (C_b E)^\dagger + (T_a E)^\dagger (T_a E)^\dagger + (T_a E)^\dagger (A_t E)^\dagger + (T_b E)^\dagger (C_b E)^\dagger + (T_b E)^\dagger (T_a E)^\dagger + (T_b E)^\dagger (A_t E)^\dagger - (C_b E)^\dagger (C_b E)^\dagger - (C_b E)^\dagger (T_a E)^\dagger - (C_b E)^\dagger (A_t E)^\dagger \right) |0000\rangle_f \right) \end{aligned}$$

**For the last term**

$$\begin{aligned} & \delta \left( (C_b S)^\dagger (T_b S)^\dagger |0000\rangle_f \right) \\ &= \delta \left( \frac{1}{\sqrt{3}} (T_a E + T_b E - C_b E) \frac{1}{\sqrt{3}} (C_b E + T_b E - A_t E) |0000\rangle_f \right) \\ &= \frac{\delta}{3} \left( \left( (T_a E)^\dagger (C_b E)^\dagger + (T_a E)^\dagger (T_b E)^\dagger - (T_a E)^\dagger (A_t E)^\dagger + (T_b E)^\dagger (C_b E)^\dagger + (T_b E)^\dagger (T_b E)^\dagger - (T_b E)^\dagger (A_t E)^\dagger - (C_b E)^\dagger (C_b E)^\dagger - (C_b E)^\dagger (T_b E)^\dagger + (C_b E)^\dagger (A_t E)^\dagger \right) |0000\rangle_f \right) \end{aligned}$$

It is necessary to apply a post-selection on the resulting states to retain only the states having:

- Exactly one photon on the control side  $(C_a E)^\dagger$  or  $(C_b E)^\dagger$  but not  $(C_a E)^\dagger (C_b E)^\dagger$
- Exactly one photon on the target side  $(T_a E)^\dagger$  or  $(T_b E)^\dagger$  but not  $(T_a E)^\dagger (T_b E)^\dagger$
- No photons on the auxiliary modes: remove all terms involving  $(A_c E)^\dagger$  or  $(A_t E)^\dagger$

Thus, by grouping the four terms and removing the terms corresponding to the post selection conditions mentioned previously:

$$\begin{aligned}
|\psi_{out}\rangle = & \frac{\alpha}{3} \left( ((C_a E)^\dagger (C_b E)^\dagger + (C_a E)^\dagger (T_a E)^\dagger + (C_a E)^\dagger (A_\tau E)^\dagger \right. \\
& + \sqrt{2}(A_c E)^\dagger (C_b E)^\dagger + \sqrt{2}(A_c E)^\dagger (T_a E)^\dagger + \sqrt{2}(A_c E)^\dagger (A_\tau E)^\dagger \left. \right) \\
& + \frac{\beta}{3} \left( ((C_a E)^\dagger (C_b E)^\dagger + (C_a E)^\dagger (T_b E)^\dagger - (C_a E)^\dagger (A_\tau E)^\dagger + \sqrt{2}(A_c E)^\dagger (C_b E)^\dagger \right. \\
& + \sqrt{2}(A_c E)^\dagger (T_b E)^\dagger - \sqrt{2}(A_c E)^\dagger (A_\tau E)^\dagger \left. \right) |0000\rangle_f \\
& + \frac{\gamma}{3} \left( ((T_a E)^\dagger (C_b E)^\dagger + (T_a E)^\dagger (T_a E)^\dagger + (T_a E)^\dagger (A_\tau E)^\dagger + (T_b E)^\dagger (C_b E)^\dagger + (T_b E)^\dagger (T_a E)^\dagger \right. \\
& + (T_b E)^\dagger (A_\tau E)^\dagger - (C_b E)^\dagger (C_b E)^\dagger - (C_b E)^\dagger (T_a E)^\dagger - (C_b E)^\dagger (A_\tau E)^\dagger \left. \right) |0000\rangle_f \\
& + \frac{\delta}{3} \left( ((T_a E)^\dagger (C_b E)^\dagger + (T_a E)^\dagger (T_b E)^\dagger - (T_a E)^\dagger (A_\tau E)^\dagger + (T_b E)^\dagger (C_b E)^\dagger + (T_b E)^\dagger (T_b E)^\dagger \right. \\
& \left. - (T_b E)^\dagger (A_\tau E)^\dagger - (C_b E)^\dagger (C_b E)^\dagger - (C_b E)^\dagger (T_b E)^\dagger + (C_b E)^\dagger (A_\tau E)^\dagger \right) |0000\rangle_f
\end{aligned}$$

After removing the terms, we obtain:

$$\begin{aligned}
|\psi_{out}\rangle = & \frac{\alpha}{3} \left( (C_a E)^\dagger (T_a E)^\dagger |0000\rangle_f \right) + \frac{\beta}{3} \left( (C_a E)^\dagger (T_b E)^\dagger |0000\rangle_f \right) + \frac{\gamma}{3} \left( (T_b E)^\dagger (C_b E)^\dagger |0000\rangle_f \right) \\
& + \frac{\delta}{3} \left( (T_a E)^\dagger (C_b E)^\dagger |0000\rangle_f \right)
\end{aligned}$$

$$|\psi_{out}\rangle = \frac{\alpha}{3} (|1010\rangle_f) + \frac{\beta}{3} (|1001\rangle_f) + \frac{\gamma}{3} (|0101\rangle_f) + \frac{\delta}{3} (|0110\rangle_f)$$

$$|\psi_{out}\rangle = \frac{1}{3} (\alpha |00\rangle + \beta |01\rangle + \gamma |11\rangle + \delta |01\rangle)$$

We almost recover the expected action of the CNOT gate:  $(\alpha |00\rangle + \beta |01\rangle + \gamma |11\rangle + \delta |01\rangle)$  up to a factor of  $\frac{1}{3}$ .

### Example

If we started from the state  $|\psi\rangle = |10\rangle$  meaning  $\gamma = 1$  we obtain:

$$|\psi_{out}\rangle = \frac{1}{3} |11\rangle$$

We do not always find ourselves in this situation whenever we use our CNOT, and the value  $\frac{1}{3}$  is the probability amplitude associated with the success of the CNOT.

### Remark

Ralph's CNOT gate is a "simple" way to implement the CNOT gate on photonic computers using dual-rail encoding. However, the application of this gate succeeds only 1 time out of 9. This limitation is not an error rate that can be corrected with hardware improvements. It is a real fundamental limit intrinsic to the functioning of Ralph's CNOT gate.

Note that the calculation starts by defining  $\theta = 2 \times \arccos\left(\sqrt{\frac{1}{3}}\right) \simeq 1.91$  where  $1/3$  is the reflectivity (the proportion of photons reflected by the Beam Splitter). Consider the circuit in Fig. 20, where we start with

two qubits (a CNOT gate acts on two qubits), meaning 4 modes, and we need two auxiliary modes for this definition of the CNOT.

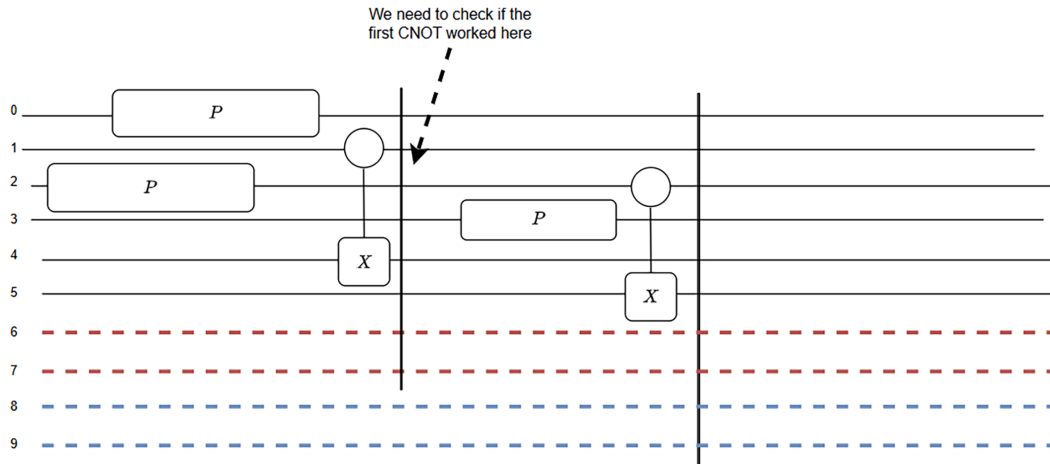
For two qubits in the state  $|00\rangle$  as input, from the perspective of Fock states, we have the state:

$$|\psi_0\rangle = |0\rangle_0 \underbrace{(|1\rangle_1|0\rangle_2)}_{\text{control qubit } |0\rangle} \underbrace{(|1\rangle_3|0\rangle_4)}_{\text{target qubit } |0\rangle} |0\rangle_5$$

In this implementation (originally proposed by [14]), the two auxiliary modes are:

- Mode 0 which appears in the first position in Fig. 20;
- Mode 5 which appears in the last position in Fig. 20.

**Conclusion for Ralph’s CNOT:** The implementation of the  $CX$  gate proposed by [14] is limited to circuits that use only a single CNOT. When a circuit contains two CNOT gates, as shown in Fig. 22 and the success conditions depend on modes that are shared with the second CNOT. While the auxiliary modes do not create a problem, the modes shared between the two CNOTs cannot be post-selected in the middle of the circuit. The CNOT gate from [14] has a success probability of  $\frac{1}{9}$ .



**Figure 22:** Incorporating two CNOT gates in a single circuit: infeasibility of implementation using the CNOT scheme by [14].

### 5.2 Implementation of the $CX_{ij}$ of [13] Using the Library [27]

There exist different ways to implement the  $CX$  gates. One other way to implement it is to use the one of [13] called heralded  $CX$ .

The heralded  $CX$  gate was proposed by Knill in 2001, and its main advantage lies in the fact that it is sufficient to measure the auxiliary qubits to determine whether the  $CX$  operation was successful. Thereby allowing to put multiple of them in the same circuit. However, its success probability is only  $\frac{2}{27}$ . To implement a photonic quantum circuit, the Perceval Library provides the  $CX$  gate from [13]. It is available in the “catalog” under the Name “heralded cnot”.

Just as for Ralph’s  $CX$ , two auxiliary modes are required, which in this case correspond to modes 4 and 5. Note that unlike the  $CX$  gate by [14], the heralded version requires both auxiliary qubits to be in state 1 at the input. Thus, the conditions for identifying a successful CNOT operation are that modes 4 and 5 must be equal to 1 at the output. There is no need to verify that we have exactly one photon on the modes representing the qubits.

An implementation using the library is given in (Appendix B).

### 5.3 Implementation the CX of Ralph and the Library [27]

The CX gate proposed by Ralph is also appearing in Perceval’s catalogue under the name “postprocessed cnot”. Therefore, you can easily build a quantum circuit using it. A code example is given in (Appendix A).

To implement this CX using the library, the only difference with the description given in Section 5.1 Realization of CX according to [14] lies in the location of two control modes. They are now located at positions 4 and 5. While in Section 5.1, the auxiliary modes were modes 0 and 5 Therefore, the conditions for identifying a successful CNOT operation are:

- modes 0 and 1 define a qubit (have exactly one photon);
- modes 2 and 3 define a qubit (have exactly one photon);
- modes 4 and 5 are equal to 0.


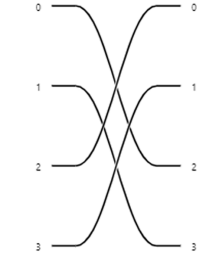
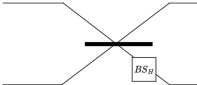
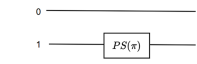

### 5.4 Representation of Gates CZ<sub>ij</sub> and CY<sub>ij</sub>

For two qubits *i* and *j*, the implementation of the gates CZ<sub>ij</sub> and CY<sub>ij</sub> can be done simply since CZ<sub>ij</sub> = H<sub>j</sub>.CX<sub>ij</sub>.H<sub>j</sub> and CY<sub>ij</sub> = S<sub>j</sub>.CX<sub>ij</sub>.S<sub>j</sub><sup>†</sup> = R<sub>1j</sub>( $\frac{\pi}{2}$ ).CX<sub>ij</sub>.R<sub>1j</sub>( $-\frac{\pi}{2}$ )

### 5.5 Summary

Table 2 provides the correspondence between quantum operators on qubits and photonic components, along with the Perceval code associated with their implementation.

**Table 2:** Correspondence between classical gates and their realization using optical components.

Qubit Operators	Photonic Circuit
$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	
$Swap_{12} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	
$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	
$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	
$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	

(Continued)

Table 2 (continued)

Qubit Operators	Photonic Circuit
$RX(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}$	
$RY(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}$	
$RZ(\theta) = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix}$	
$CX_{ij} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	

## 6 Conclusion

This article presents the foundational concepts of photonic quantum computing from an operator- and computation-oriented perspective, providing a state-of-the-art overview of quantum gates within a universal formalism. Beam splitters and phase shifters are introduced, along with a physical description of their action on spatial modes, offering a comprehensive view of photonic quantum computers. Each basic gate is expressed in matrix form, and we demonstrate how standard quantum computation gates can be implemented using photonic components. The gates are decomposed into optical elements according to the KLM protocol and employing dual-rail encoding. Basic elements for programming with Quandela's Perceval library are included in the Appendix Section to facilitate practical implementation of these concepts. Particular emphasis is placed on different CNOT realizations, specifically the heralded CNOT and the Ralph CNOT, highlighting their differences, advantages, and limitations.

**Acknowledgement:** Authors have used ChatGPT to improve English skill.

**Funding Statement:** This work was done within the scope of a Ph.D which is funded by CGI for author Martin Bombardelli. Philippe Lacomme, Gerard Fleury and Bogdan Vulpescu are financed by a public laboratory in France (LIMOS, LIMOS and LPCA, respectively).

**Author Contributions:** The authors confirm contribution to the paper as follows: Conceptualization, Martin Bombardelli, Philippe Lacomme and Bogdan Vulpescu; methodology, Martin Bombardelli, Philippe Lacomme, Bogdan

Vulpescu and Gérard Fleury; software, Martin Bombardelli, Philippe Lacomme and Bogdan Vulpescu; validation, Martin Bombardelli, Philippe Lacomme, Bogdan Vulpescu and Gérard Fleury; formal analysis, Martin Bombardelli, Philippe Lacomme, Bogdan Vulpescu and Gérard Fleury; investigation, Martin Bombardelli, Philippe Lacomme, Bogdan Vulpescu and Gérard Fleury. All authors reviewed and approved the final version of the manuscript.

**Availability of Data and Materials:** Not applicable.

**Ethics Approval:** Not applicable.

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

## Appendix A *Ralph* Cnot

```
import perceval as pcvl
import numpy as np
def toFockState(qubitState):
    pe = {0: [1,0], 1: [0,1], 2: [0,1]}
    res = pe [qubitState [0]] + pe [qubitState [1]] + pe [qubitState
    [2]]
    return res
def strState(state):
    return str(pcvl.BasicState(state))
from perceval.components import PS, BS, PERM
import perceval as pcvl
import numpy as np
from perceval.components import BS
theta_13 = BS.r_to_theta(r=1/3)
cnot = (pcvl.Circuit(6, name = "Ralph CNOT")
.add((0, 1), BS.H(theta_13, phi_bl = np.pi,
phi_tr = np.pi/2, phi_tl = -np.pi/2))
.add((3, 4), BS.H())
.add((2, 3), BS.H(theta_13, phi_bl = np.pi,
phi_tr = np.pi/2, phi_tl = -np.pi/2))
.add((4, 5), BS.H(theta_13))
.add((3, 4), BS.H()))

# 1.initialisation
istate = [0,1,0,1,0,0]
# 2.testof the cnot
circuit = pcvl.Circuit(6)

circuit.add(0,PERM([0,1,2,4,3,5]))
```

```

circuit.add(0, PERM([0,1,3,2,4,5]))
circuit.add(0, PERM([0,2,1,3,4,5]))
circuit.add(0, PERM([1,0,2,3,4,5]))
circuit.add(0, cnot)
circuit.add(0, PERM([1,0,2,3,4,5]))
circuit.add(0, PERM([0,2,1,3,4,5]))
circuit.add(0, PERM([0,1,3,2,4,5]))
circuit.add(0, PERM([0,1,2,4,3,5]))

# 3.Displaying the circuit
pcvl.pdisplay(circuit.compute_unitary())
print(“”)
pcvl.pdisplay(circuit)

# 4.Simulation
p = pcvl.Processor(“SLOS”, circuit)
p.set_postselection(pcvl.PostSelect(“[0,1]==1 & [2,3]==1 & [4]==0 &
[5]==0”))
nsample = 50,000
from perceval.algorithm import Sampler, Analyzer
sampler = Sampler (p)
p.with_input (pcvl.BasicState ([0, 1, 0, 1, 0, 0]))
output = sampler.sample_count (1000)
pcvl.pdisplay (output[‘results’], output_format = pcvl.Format.TEXT)

```

## Appendix B *Heralded Cnot*

```

import perceval as pcvl
import numpy as np
from perceval.components import PS, BS, PERM

import perceval as pcvl
import numpy as np
from perceval.components import BS
from perceval.components import catalog
cnot = catalog [‘heralded cnot’].build_circuit()

# 1.initialization
istate = [0, 1, 0, 1, 1, 1]
# 2.testof the cnot

```

```

circuit = pcvl.Circuit(6)
circuit.add(0,cnot)
# 3.Displaying the circuit
pcvl.pdisplay(circuit.compute_unitary())
print("""
pcvl.pdisplay(circuit)
# 4 calculation
p = pcvl.Processor("SLOS", circuit)
p.set_postselection(pcvl.PostSelect("[4]==1 & [5]==1"))
nsample = 50,000
from perceval.algorithm import Sampler, Analyzer
sampler = Sampler (p)
p.with_input (pcvl.BasicState (istate)) #Corresponds to logical
qubitstate
output = sampler.sample_count (1000)
pcvl.pdisplay (output ['results'], output_format =
pcvl.Format.TEXT)
print('end')

```

### Appendix C CCNOT

```

import perceval as pcvl
import numpy as np
from perceval.components import PS, BS, PERM
import perceval as pcvl
import numpy as np
from perceval.components import BS
from perceval.components import catalog
ccnot = catalog ['toffoli'].build_circuit()

# 1.initialisation
istate = [0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0]
# 2.Cnot
circuit = pcvl.Circuit(12)
circuit.add(0,ccnot)
# 3.printof the circuit
pcvl.pdisplay(circuit.compute_unitary())
print("""

```

```

pcvl.pdisplay(circuit)
# 4 Computation
p = pcvl.Processor("SLOS", circuit)
p.set_postselection(pcvl.PostSelect("[0,1]==1 &
[2,3]==1 & [6]==0 & [7]==0 & [8]==0 & [9]==0 &
[10]==0 & [11]==0"))

nsample = 1000
from perceval.algorithm import Sampler, Analyzer
sampler = Sampler(p)
p.with_input(pcvl.BasicState(istate)) #Corresponds to logical
qubitstate
output = sampler.sample_count(nsample)
print(output ['results'])
pcvl.pdisplay(output ['results'], output_format =
pcvl.Format.TEXT)
print('end')

```

## Appendix D Perceval Implementation of $PERM_{1023}$

### Step 1. Circuit Initialization

This can be done by initializing qubits and then converting them into a **Fock state**.

For example, starting from the state  $|10\rangle$  the corresponding code would be:

```

qubit_istate = [1, 0]
istate = toFockState(qubit_istate)

```

### Step 2. Define the circuit composed of 2 qubits, i.e., 4 Fock modes

```

circuit = pcvl.Circuit(4)

```

### Step 3. Add an X gate to the circuit, applied to the first qubit

The associated code is:

```

circuit.add(0, PERM([1, 0, 2, 3]))
#or circuit.add(0, PERM([1, 0]))

```

### Step 4. Display the circuit

```

pcvl.pdisplay(circuit.compute_unitary())
print(“”)

```

```

pcvl.pdisplay(circuit)

```

### Step 5. Perform the computation

```

backend = pcvl.BackendFactory().get_backend("Naive")
backend.set_circuit(circuit)
backend.set_input_state(pcvl.BasicState(istate))

```

### Step 6. Define the possible output states

```

output_qubit_states = [

```

```
[x1, x2]
for x1 in [0, 1] for x2 in [0, 1]
]
```

### Step 7. Display the results

```
for oqstate in output_qubit_states:
    ostate = toFockState(oqstate)
    a = backend.prob_amplitude(pcvl.BasicState(ostate))
    print(strState(ostate), "-> ", strState(oqstate), a)
print('end')
```

## References

1. Manin Y. Computable and uncomputable. Moscow, Russia: Sovetskoye Radio; 1980. 128 p.
2. Anonymous. 40 years of quantum computing. *Nat Rev Phys*. 2022;4(1):1. doi:10.1038/s42254-021-00410-6.
3. Benioff P. The computer as a physical system: a microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *J Stat Phys*. 1980;22(5):563–91. doi:10.1007/BF01011339.
4. Buluta I, Nori F. Quantum simulators. *Science*. 2009;326(5949):108–11. doi:10.1126/science.1177838.
5. Feynman RP. Simulating physics with computers. In: Feynman and computation. Boca Raton, FL, USA: CRC Press; 2018. p. 133–53.
6. Deutsch D. Quantum theory, the church-turing principle and the universal quantum computer. *Proc R Soc Lond Ser A Math Phys Sci*. 1985;400(1818):97–117.
7. Deutsch D, Jozsa R. Rapid solution of problems by quantum computation. *Proc R Soc Lond Ser A Math Phys Sci*. 1992;439(1907):553–8. doi:10.1098/rspa.1992.0167.
8. Grover LK. A fast quantum mechanical algorithm for database search. In: Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing—STOC'96; 1996 May 22–24; Philadelphia, PA, USA. p. 212–9. doi:10.1145/237814.237866.
9. Shor PW. Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings 35th Annual Symposium on Foundations of Computer Science; 1994 Nov 20–22; Santa Fe, NM, USA. p. 124–34. doi:10.1109/SFCS.1994.365700.
10. Chuang IL, Yamamoto Y. Simple quantum computer. *Phys Rev A*. 1995;52(5):3489–96. doi:10.1103/physreva.52.3489.
11. Fock V. Configuration space and second quantization. In: V.A. Fock-selected works: quantum mechanics and quantum field theory. Boca Raton, FL, USA: CRC Press; 2004. p. 191–220. doi:10.1201/9780203643204.ch5b.
12. Knill E, Laflamme R, Milburn GJ. A scheme for efficient quantum computation with linear optics. *Nature*. 2001;409(6816):46–52. doi:10.1038/35051009.
13. Knill E. Quantum gates using linear optics and postselection. *Phys Rev A*. 2002;66(5):052306. doi:10.1103/physreva.66.052306.
14. Ralph TC, White AG, Munro WJ, Milburn GJ. Simple scheme for efficient linear optics quantum gates. *Phys Rev A*. 2001;65(1):012314. doi:10.1103/physreva.65.012314.
15. Raussendorf R, Briegel HJ. A one-way quantum computer. *Phys Rev Lett*. 2001;86(22):5188–91. doi:10.1103/physrevlett.86.5188.
16. Yu Y, Chi Y, Zhai C, Huang J, Gong Q, Wang J. Simulating Hamiltonian dynamics in a programmable photonic quantum processor using linear combinations of unitary operations. *arXiv:2211.06723*. 2022.
17. Riso AL, Thyagarajan K, Whiting C, Jimenez K. Constructing a photonic implementation of quantum key distribution. *arXiv:2509.04389*. 2025.

18. Zahidy M, Mikkelsen MT, Müller R, Da Lio B, Krehbiel M, Wang Y, et al. Quantum key distribution using deterministic single-photon sources over a field-installed fibre link. *npj Quantum Inf.* 2024;10(1):2. doi:10.1038/s41534-023-00800-x.
19. Zhang Y, Ding X, Li Y, Zhang L, Guo YP, Wang GQ, et al. Experimental single-photon quantum key distribution surpassing the fundamental coherent-state rate limit. *arXiv:2406.02045*. 2024.
20. Baldazzi A, Azzini S, Pavesi L. Resource-efficient variational quantum solver for the travelling salesman problem and its silicon photonics implementation. *arXiv:2511.02696*. 2025.
21. Baldazzi A, Sanna M, Borghi M, Azzini S, Pavesi L. Four-qubit variational algorithms in silicon photonics with integrated entangled photon sources. *arXiv:2501.01301*. 2025.
22. Abbassi A, Dujardin Y, Gourdin E, Lacomme P, Prodhon C. Quantum approaches to the minimum edge multiway cut problem. *arXiv:2601.00720*. 2026.
23. Aaronson S, Arkhipov A. The computational complexity of linear optics. In: *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*. San Jose, CA, USA: ACM; 2011. p. 333–42. doi:10.1145/1993636.1993682.
24. Zhong HS, Wang H, Deng YH, Chen MC, Peng LC, Luo YH, et al. Quantum computational advantage using photons. *Science*. 2020;370(6523):1460–3. doi:10.1126/science.abe8770.
25. Romero J, Milburn G. Photonic quantum computing. *arXiv:2404.03367*. 2024.
26. Campos R, Saleh B, Teich M. Quantum-mechanical lossless beam splitter: SU(2) symmetry and photon statistics. *Phys Rev A Gen Phys.* 1989;40(3):1371–84. doi:10.1103/physreva.40.1371.
27. Heurtel N, Fyrrillas A, de Gliniasty G, Le Bihan R, Malherbe S, Pailhas M, et al. Perceval: a software platform for discrete variable photonic quantum computing. *Quantum*. 2023;7:931. doi:10.22331/q-2023-02-21-931.
28. Hong C, Ou Z, Mandel L. Measurement of subpicosecond time intervals between two photons by interference. *Phys Rev Lett.* 1987;59(18):2044–6. doi:10.1103/PhysRevLett.59.2044.