Verification of Linear Optical Quantum Computing using Quantum Process Calculus

Sonja Franke-Arnold
School of Physics and Astronomy
University of Glasgow, UK
sonja.franke-arnold@glasgow.ac.uk

Simon J. Gay
School of Computing Science
University of Glasgow, UK
Simon.Gay@glasgow.ac.uk

Ittoop Vergheese Puthoor *
School of Computing Science and
School of Physics and Astronomy
University of Glasgow, UK
ittoop@dcs.gla.ac.uk

We explain the use of quantum process calculus to describe and analyse linear optical quantum computing (LOQC). The main idea is to define two *processes*, one modelling a linear optical system and the other expressing a specification, and prove that they are *behaviourally equivalent*. We extend the theory of *behavioural equivalence* in the process calculus Communicating Quantum Processes (CQP) to include multiple particles (namely photons) as information carriers, described by *Fock states* or *number states*. We summarise the theory in this paper, including the crucial result that equivalence is a *congruence*, meaning that it is preserved by embedding in any context. In previous work, we have used quantum process calculus to model LOQC but without verifying models against specifications. In this paper, for the first time, we are able to carry out verification. We illustrate this approach by describing and verifying two models of an LOQC CNOT gate.

1 Introduction

Quantum information processing (QIP) is a field of research, which involves the study of storing and manipulating information in systems that are governed by the laws of quantum mechanics. This provides huge potential in quantum computation, cryptography and communication [15], and first secure cryptography systems are already commercially available [9]. Linear optical quantum computing (LOQC) is being pioneered for applications in scalable quantum computing [10]. LOQC is based on *spatial encoding* where a quantum bit is encoded as a superposition of two spatial modes or the two optical paths that can be travelled by a single photon [16]. The inherent weak interaction between photons as information carriers makes them highly suitable for communication applications.

Quantum process calculus is a class of *formal methods*, able to describe and analyse the behaviour of systems that combine quantum and classical elements. The success of formal methods in classical computer science has motivated the development of quantum process calculus called Communicating Quantum Processes (CQP) [6]. CQP provides an abstract model of the quantum system, with the assumption that a qubit is considered as a localised unit of information. CQP verifies the correctness of a system by employing the theory of *behavioural equivalence* [3] between processes. Also, the equivalence is a *congruence*, meaning that it is preserved by inclusion in any environment. The theory has been applied to the analysis of a quantum error correcting code [2].

Contributions: This paper enhances from previous work [5] significantly in two different ways. First, we provide the theory of equivalence in CQP for LOQC, which has been extended from Davidson thesis [3], in order to analyse and verify a realistic experimental system. The *congruence* property of equivalence in CQP is applied to the LOQC CNOT gate, which provides us for the first time with a more

*Supported by a Lord Kelvin / Adam Smith Scholarship from the University of Glasgow.

J. Borgström, S. Crafa (Eds.): Combined Workshop on Expressiveness in
Concurrency and Structural Operational Semantics (EXPRESS/SOS 2014)
EPTCS 160, 2014, pp. 111–129, doi:10.4204/EPTCS.160.10

physical understanding of the property of equivalence. Second, we present two models of an experimental system that demonstrates LOQC CNOT gate and prove that they are equivalent to their specification. These two models not only demonstrates the gate but uses two different measurement semantics which exhibits the flexibility of process calculus approach to work at different levels of abstraction. In our second model, we demonstrate *post-selection*, which plays an important role in LOQC, where one considers only a subset of all experimental runs that fulfil predefined criteria.

The present paper begins in Section 2 by recalling the basic concepts of quantum optics which are needed to understand LOQC. In Section 3 we review the language of CQP, illustrated with a model of the experimental system that demonstrates LOQC CNOT gate. Section 4 summarises the extension of the theory of equivalence in CQP, which is applied to LOQC. In Section 5 we describe the post-selection process and analyse a model of an experimental system demonstrating *post-selective* LOQC CNOT gate. Finally, Section 6 concludes with an indication of directions for future work.

Related Work: All the quantum process calculi which have been developed so far considered a qubit as an abstract particle that can be sent or received through channels. Feng *et al.* [4] developed qCCS, a quantum extension of the classical value-passing CCS [12] and proved that weak bisimilarity is a congruence. The result is applied to quantum teleporation, superdense coding and quantum key distribution protocols [11].

2 Background

We recall briefly the aspects of quantum theory and quantum optics relevant for this paper. For more detailed information we refer to the book by Nielsen and Chuang [15] and research papers [10, 16, 18].

A *qubit* is an information unit comprising two states ($|0\rangle$ and $|1\rangle$) which are called the *standard* basis. The *state space* $\mathbb{H}$ (or Hilbert space) of a qubit consists of all *superpositions* of the basis states: $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ where α and β are complex numbers such that $|\alpha|^2 + |\beta|^2 = 1$. A qubit is conventionally realised by an individual photon with the two basis states refering to orthogonal polarisation directions of the photon ($|0\rangle = |H\rangle$ and $|1\rangle = |V\rangle$). We refer to the qubit as a polarisation qubit where H and V denote horizontal and vertical polarisation, respectively. We introduce the notation $\alpha|H\rangle + \beta|V\rangle = \alpha|10\rangle_{HV} + \beta|01\rangle_{HV}$, where the entries in the ket states represent the number of photons (photon number n) in the state basis indicated by the subscripts. This will allow us to generalise the notation to more than one photon. Two photons in the states $\alpha_i|H\rangle + \beta_i|V\rangle$ (where i is 1,2 respectively for each photon) can then be encoded in the shorthand $\alpha_1\alpha_2|20\rangle_{HV} + \beta_1\beta_2|02\rangle_{HV} + (\alpha_1\beta_2 + \alpha_2\beta_1)|11\rangle_{HV}$, if they are indistinguishable in all other parameters. In LOQC [10], we consider qubits which are encoded in different optical paths 'a' and 'b' rather than different polarisation states. This is referred to as *dual rail logic*. Again, we denote the quantum states in the *number state* basis, giving the number of photons travelling along the different paths. The basis states in dual rail logic are then $|0\rangle \to |10\rangle_{ab}$, and similarly for $|1\rangle \to |01\rangle_{ab}$. In experiments, the conversion of a *polarisation* qubit into a *dual rail* qubit is accomplished by the combination of a polarising beam splitter (PBS) and a phase shifter (PR) [16], which works as a unitary operation PS.

Definition 1 (PS operator) *A PS is an operator that transforms a polarisation qubit $|\psi\rangle \in \mathbb{H}_q$ to a dual rail qubit $|\phi\rangle \in \mathbb{H}_s$, where $\mathbb{H}_q$ and $\mathbb{H}_s$ are the respective Hilbert spaces for the polarisation and dual rail qubits. The action of PS is then defined by $\mathsf{PS}|H\rangle \equiv \mathsf{PS}|10\rangle_{HV} = |10\rangle_{ab}$ and $\mathsf{PS}|V\rangle \equiv \mathsf{PS}|01\rangle_{HV} = |01\rangle_{ab}$*

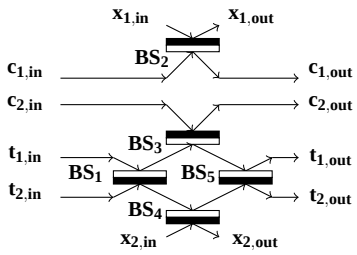

Figure 1: A schematic representation of the LOQC CNOT gate. BS_1 and BS_5 are beam splitters of reflectivity $\frac{1}{2}$ and the others are of reflectivity $\frac{1}{3}$. The dark side of the BS indicates the side from which a sign change occurs upon reflection.

Operations on number states (or *Fock states* $|n\rangle$) are described in terms of the creation and annihilation operators $\hat{a}^\dagger$ and $\hat{a}$, which when acting on a state $|n\rangle$ increase or decrease the photon number (n) by one. Therefore, each Fock state can be built up from creation operators given by $|n\rangle = \frac{(\hat{a}^\dagger)^n}{\sqrt{n!}}|0\rangle$. In LOQC, optical elements such as phase shifters and non polarising beam splitters perform *unitary transformations*, which describe the evolution of a closed quantum system. A unitary transformation in LOQC [14] can be described by its effect on each photon path's creation operator. A non polarising beam splitter (BS) is defined by the transformation matrix

$$U(BS) \quad = \quad \begin{pmatrix} \cos\theta & e^{i\phi}\sin\theta \\ e^{-i\phi}\sin\theta & -\cos\theta \end{pmatrix}$$

The reflectivity and transmissivity of a BS are given by $\eta = \cos^2\theta$ and $1-\eta = \sin^2\theta$, respectively, θ is the angle between the polarisation direction of the input photon and the crystal axis of the BS and ϕ is the relative phase between the light modes in the two output paths. Here we consider $\phi = 0$, which is the case for BSs in integrated circuits. If the state $|mn\rangle_{ab}$ is incident on a BS with m photons along path a and n photons along path b, the transformation is:

$$|mn\rangle_{ab} = \frac{(\hat{a}_a^\dagger)^m}{\sqrt{m!}}\frac{(\hat{a}_b^\dagger)^n}{\sqrt{n!}}|00\rangle_{ab} \rightarrow \frac{1}{\sqrt{m!n!}}(\hat{a}_a^\dagger\cos\theta + \hat{a}_b^\dagger\sin\theta)^m(\hat{a}_a^\dagger\sin\theta - \hat{a}_b^\dagger\cos\theta)^n|00\rangle_{ab} \tag{1}$$

The *controlled Not* (or CNOT) is a quantum gate that is a primary component in building a quantum computer. The operation of the gate is that it flips the second qubit (target qubit) if and only if the first qubit (control qubit) is 1. On qubits, we have $\mathsf{CNOT}|0x\rangle = |0x\rangle$ and $\mathsf{CNOT}|1x\rangle = |1y\rangle$ where $x,y \in \{0,1\}$ and $y = x \oplus 1$ with $\oplus$ denoting addition modulo 2. In dual rail logic, this becomes $\mathsf{CNOT}|10yx\rangle = |10yx\rangle$ and $\mathsf{CNOT}|01yx\rangle = |01xy\rangle$.

In the following we summarise the theory and operation of the LOQC CNOT gate [16, 18]. The BSs used in the LOQC CNOT gate [16, 18] have reflectivities of $\eta = \frac{1}{2}$ or $\frac{1}{3}$. The operation is specified by a control qubit, characterised by the number states c_1 and c_2, and a target qubit, characterised by t_1 and t_2, as well as two auxiliary vacuum states (absence of a qubit or photon) x_1 and x_2, written as $|c_1c_2t_1t_2\rangle|x_1x_2\rangle$. Consider the general input state

$$|\psi\rangle_{\text{in}}|00\rangle = (\alpha|1010\rangle + \beta|1001\rangle + \gamma|0110\rangle + \delta|0101\rangle)|00\rangle \tag{2}$$

The schematic representation of the LOQC CNOT gate is shown in Figure 1. Using the operators for each BS as discussed in Eq. 1 and applying it to the input state, Eq. 2 we get the output state of the CNOT

$$
\begin{array}{rcl}
T & ::= & \mathsf{Int} \mid \mathsf{Qbit} \mid \mathsf{NS} \mid \mathsf{Bit} \mid \hat{}[\tilde{T}] \mid \mathsf{Op}(1) \mid \mathsf{Op}(2) \mid \cdots \\
v & ::= & x \mid 0 \mid 1 \mid \cdots \mid \mathsf{H} \mid \cdots \\
e & ::= & v \mid \mathsf{measure}\ \tilde{e} \mid \mathsf{psmeasure}\ \tilde{e} \mid \tilde{e} *= e \mid e + e' \mid (e,e) \mid \mathsf{if}\ e\ \mathsf{then}\ e\ \mathsf{else}\ e \mid x : \mathsf{NS}, y : \mathsf{NS} *= \mathsf{PS}(z) \\
P & ::= & \mathbf{0} \mid (P|P) \mid P+P \mid e?[\tilde{x} : \tilde{T}].P \mid e![\tilde{e}].P \mid \{e\}.P \mid (\mathsf{qbit}\ x)P \mid (\mathsf{ns}\ x)P \mid (\mathsf{new}\ x : \hat{}[T])P
\end{array}
$$

Figure 2: Syntax of CQP.

gate as:

$$
\begin{array}{c}
|\psi\rangle_{\mathrm{in}}|00\rangle \rightarrow \frac{1}{3}\{(\alpha|1010\rangle + \beta|1001\rangle + \gamma|0101\rangle + \delta|0110\rangle)|00\rangle + \sqrt{2}(\alpha+\beta)|0100\rangle|10\rangle \\
+\sqrt{2}(\alpha-\beta)|0000\rangle|11\rangle + (\alpha+\beta)|1100\rangle|00\rangle + (\alpha-\beta)|1000\rangle|01\rangle + \sqrt{2}\alpha|0010\rangle|10\rangle \\
+\sqrt{2}\beta|0001\rangle|10\rangle - \sqrt{2}(\gamma+\delta)|0200\rangle|00\rangle - (\gamma-\delta)|0100\rangle|01\rangle + \sqrt{2}\gamma|0020\rangle|00\rangle \\
+(\gamma-\delta)|0010\rangle|01\rangle + (\gamma+\delta)|0011\rangle|00\rangle + (\gamma-\delta)|0001\rangle|01\rangle + \sqrt{2}\delta|0002\rangle|00\rangle\}
\end{array}
\tag{3}
$$

LOQC embeds qubits into the larger dual-rail space, to enable a particular physical realisation of unitary operators to be used. However, this introduces the possibility that the result of the final measurement may be outside the embedding and hence not interpretable as a computational result. *Post-selection* compensates for this possibility by discarding the undesirable measurement results at the expense of introducing a non-zero probability that the overall computation fails. From these states we *post-select* only those where one photon is found in the target and one in the control state, by discarding all terms apart from the first four terms in the first line of Eq. 3, giving

$$
|\phi\rangle_{ps} = \alpha|1010\rangle + \beta|1001\rangle + \gamma|0101\rangle + \delta|0110\rangle \tag{4}
$$

Successful *post-selection* occurs only with a probability of one-ninth and the relationship between Eq. 2 and Eq. 4 is a controlled-NOT transformation.

3 Communicating Quantum Processes (CQP)

CQP [6] is a quantum process calculus, which was established for formally defining the structure and behaviour of systems that comprise both quantum and classical communication and computation. The language is based on the π-calculus [13] with primitives for quantum information. The general idea is that a system is considered to be made up of independent components or *processes*. The *processes* can communicate by sending and receiving data along *channels* and these data are qubits, number states or classical values. A distinctive feature of CQP is its static type system [7], the purpose of which is to classify classical and quantum data and also to enforce the no-cloning property of quantum information. We now present CQP including the extensions required for LOQC.

3.1 Syntax of CQP

The syntax of CQP is defined by the grammar as shown in Figure 2. We use the notation $\tilde{e} = e_1, \ldots, e_n$, and write $|\tilde{e}|$ for the length of a tuple. The syntax consists of types T, values v, expressions e (including quantum measurements and the conditional application of unitary operators $\tilde{e} *= e$), and processes P. Values v consist of variables (x,y,z etc), literal values of data types (0,1,..), unitary operators such as the

$$
\begin{array}{lcl}
v & ::= & \ldots \mid q \mid s \mid c \\
E & ::= & [] \mid \mathsf{measure}\ E, \tilde{e} \mid \mathsf{measure}\ v, E, \tilde{e} \mid \ldots \mid \mathsf{measure}\ \tilde{v}, E \mid E + e \mid v + E \mid \mathsf{if}\ E\ \mathsf{then}\ e\ \mathsf{else}\ e \\
F & ::= & []?[\tilde{x}].P \mid []![\tilde{e}].P \mid v![[].\tilde{e}].P \mid v![v,[],\tilde{e}].P \mid \cdots \mid v![\tilde{v},[]].P \mid \{[]\}.P
\end{array}
$$

Figure 3: Internal syntax of CQP.

Hadamard operator H. Expressions e consist of values, measurements $\mathsf{measure}\ e_1,\ldots,e_n$, applications $e_1,\ldots,e_n *= e$ of unitary operators and applications $x:\mathsf{NS}, y:\mathsf{NS} *= \mathsf{PS}(z)$ of PS operator, expressions involving data operators such as $e+e'$ and a pair of values (e,e). We have a new addition to the expression called *post-selective* measurement $\mathsf{psmeasure}\ e_1,\ldots,e_n$. Processes include the nil process $\mathbf{0}$, parallel composition $P|P$, inputs $e?[\widetilde{x}:\widetilde{T}].P$, outputs $e![\widetilde{e}].P$, actions $\{e\}.P$ (typically a unitary operation or measurement), typed channel restriction $(\mathsf{new}\ x:\widehat{}[\widetilde{T}])P$, qubit declaration $(\mathsf{qbit}\ x)P$ and number state declaration $(\mathsf{ns}\ x)P$.

In order to define the operational semantics we provide the *internal syntax* in Figure 3. We assume a countably infinite set of qubit names, ranging over $q,r,\ldots$, a countably infinite set of number state names $s,t,\ldots$ and similarly channel names. Values are supplemented with either qubit names q or number state names s, which are generated at run-time and substituted for the variables used in qbit and ns declarations respectively. Evaluation contexts for expressions ($E[]$) and processes ($F[]$) are used to define the operational semantics [20]. Later in the paper, we also use parameterised process definitions.

3.2 Linear Optical Elements in CQP

First, we define a process *PolSe* which provides the input to the LOQC CNOT gate by converting a polarisation qubit to a dual rail qubit.

$$PolSe(a:\widehat{}[\mathsf{Qbit}], c:\widehat{}[\mathsf{NS}], d:\widehat{}[\mathsf{NS}]) \quad = \quad a?[q_0:\mathsf{Qbit}].\{s_0:\mathsf{NS}, s_1:\mathsf{NS} *= \mathsf{PS}(q_0)\}.c![s_0].d![s_1].\mathbf{0}$$

PolSe is parameterized by three channels, a,c and d. The right hand side of the definition specifies the behaviour of the process *PolSe*. The polarisation qubit (say q_0) is received as input through channel a (whose type is $\widehat{}[\mathsf{Qbit}]$) indicated as $a?[q_0:\mathsf{Qbit}]$. The term $\{s_0:\mathsf{NS}, s_1:\mathsf{NS} *= \mathsf{PS}(q_0)\}$ specifies that the PS operation is applied to qubit q_0 thereby generating s_0 and s_1 of type number states (NS). PS corresponds to the transformation produced by the combination of PBS and PR, introduced by Definition 1. The last two terms ($c![s_0]$ and $d![s_1]$) indicate that the respective values of the number states are sent through the respective output channels. The term $\mathbf{0}$ simply indicates termination.

Next, we define a non polarising beam splitter in CQP as *BS*, which is a primary component in the LOQC CNOT gate.

$$BS(e:\widehat{}[\mathsf{NS}], f:\widehat{}[\mathsf{NS}], h:\widehat{}[\mathsf{NS}], i:\widehat{}[\mathsf{NS}], \eta) \quad = \quad e?[s_2:\mathsf{NS}].f?[s_3:\mathsf{NS}].\{s_2, s_3 *= \mathsf{B}_\eta\}.h![s_2].i![s_3].\mathbf{0}$$

where η is the reflectivity. In a similar way, process *BS* receives inputs s_2 and s_3 from e and f. Then performs the unitary operation represented by $\{s_2, s_3 *= \mathsf{B}_\eta\}$ on the number states as defined by Eq. 1. Here B_η is the unitary operation represented by the matrix $U(BS)$ for $\phi = 0$. The number states are then output on h and i.

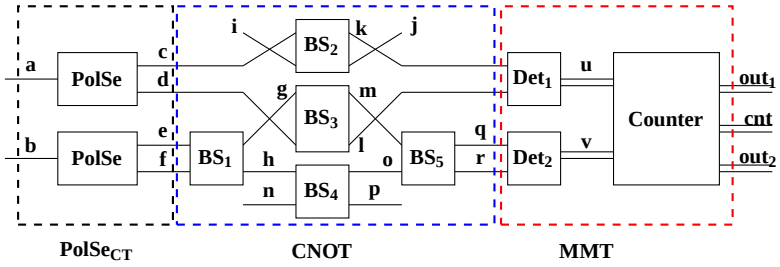

Figure 4: Model of LOQC CNOT gate: The dashed lines enclose the subsystems which are defined in the text.

In this paper, we present two types of measurements. We define *Det* and *PDet* which represent the detectors that performs measurement and *PDet* performs *post-selective* measurement.

$$\begin{aligned} Det(l{:}\widehat{}[\mathsf{NS}], m{:}\widehat{}[\mathsf{NS}], u{:}\widehat{}[\mathsf{Int},\mathsf{Int}]) &= l?[s_0:\mathsf{NS}].m?[s_1:\mathsf{NS}].u![\mathsf{measure}\ s_0,s_1].\mathbf{0} \\ PDet(l{:}\widehat{}[\mathsf{NS}], m{:}\widehat{}[\mathsf{NS}], u{:}\widehat{}[\mathsf{Bit}]) &= l?[s_0:\mathsf{NS}].m?[s_1:\mathsf{NS}].u![\mathsf{psmeasure}\ s_0,s_1].\mathbf{0} \end{aligned}$$

Here, the detectors measure a pair of number states. The expression $\mathsf{measure}\ s_0,s_1$ probabilistically evaluates to a pair of positive integers which is the number of photons detected in the respective channels and $\mathsf{psmeasure}\ s_0,s_1$ produces a zero or one which is a result of *post-selection*. The different measurement semantics enables us to work at different levels of abstraction by showing the flexibility of the process calculus approach and is discussed in detail in later sections of the paper.

3.3 The LOQC CNOT Gate in CQP : First Model

The structure of the first model of the experimental system that demonstrates LOQC CNOT gate is shown in Figure 4. The system receives two polarisation qubits (control and target) as inputs through the channels a and b. The qubits are then converted to number states by the process $PolSe_{CT}$, and these are provided as the input to the CNOT gate represented by process *CNOT*. The output of *CNOT* is then measured by the process *MMT*. The whole model is then defined as a parallel composition of $PolSe_{CT} \mid CNOT \mid MMT$. The CQP definition of the model is

$$Model_1(\widetilde{X}) = (\mathsf{new}\ \widetilde{Y})(PolSe_{CT}(\widetilde{U}) \mid CNOT(\widetilde{V}) \mid MMT(\widetilde{W}))$$

where each process is parameterised by their respective list of the channels ($\widetilde{X},\widetilde{U},\widetilde{V}$ and $\widetilde{W}$) on which it interacts with other processes. $\widetilde{X}$ contains channels a,b,out_1,cnt and out_2. $\widetilde{U}$ contains a,b,c,d,e,f and $\widetilde{W}$ contains k,l,q,r,out_1,cnt,out_2. The scope of the list of channels ($\widetilde{Y}$) is restricted, indicated by new in the definition. $\widetilde{Y}$ comprises of the channels $c,d,e,f,g,h,m,l,k,o,q,r,u$ and v. We have omitted the types from our definitions, for brevity. Also, the definitions include a list of channels rather than individual channel names. The CQP definition for $PolSe_{CT}$ is $PolSe_{CT}(\widetilde{U}) = PolSe(a,c,d) \mid PolSe(b,e,f)$. Recall from Section 3.2 that *PolSe* represents the combination of a PBS and PR.

Each BS is represented by a process *BS* and is annotated to show the correspondence with Figure 4. BS_2 and BS_3 have their inputs crossed over, corresponding to their orientation [16]. Vacuum states y and z are created by $(\mathsf{ns}\ y,z)$ and communicated to BS_2 and BS_4 respectively through the channels i and n. These *BSs* combine to form *CNOT* defined as:

$$\begin{aligned} CNOT(\widetilde{V}) = (\mathsf{new}\ g,h,m,o,i,j,n,p)(\mathsf{ns}\ y,z)(&BS_1(e,f,g,h,\tfrac{1}{2}) \mid i![y].\mathbf{0} \mid BS_2(i,c,k,j,\tfrac{1}{3}) \mid j?[y].\mathbf{0} \mid \\ &BS_3(g,d,m,l,\tfrac{1}{3}) \mid n![z].\mathbf{0} \mid BS_4(h,n,o,p,\tfrac{1}{3}) \mid p?[z].\mathbf{0} \mid BS_5(m,o,q,r,\tfrac{1}{2})) \end{aligned}$$

Here $\widetilde{V}$ contains the channels c,d,e,f,k,l,q and r. The outputs of *CNOT* are sent through the channels k,l,q and r, to the process *MMT*. The unused *BS* outputs j and p are absorbed by $j?[y]$ and $p?[z]$.

$$MMT(\widetilde{W}) = (\mathsf{new}\ u,v)(Det_1(k,l,u) \mid Det_2(q,r,v) \mid Counter(u,v,out_1,cnt,out_2,b))$$

MMT performs the measurement. Detectors Det_1, Det_2 are annotated to match Figure 4 and measure the number states associated with the control and target qubits. The output of a detector are two classical values which represents the measurement outcome, that is the number of photons detected. The outcomes of the detector processes are given as inputs to the process *Counter*.

$$\begin{aligned} Counter(u,v,out_1,cnt,out_2,b:\mathsf{Bit}) = u?[c_0:\mathsf{Int},c_1:\mathsf{Int}]\,.\,v?[t_0:\mathsf{Int},t_1:\mathsf{Int}]\,.\\ out_1![\textbf{if } (c_0+c_1=1) \textbf{ then } c_1 \textbf{ else } 0]\,.\,out_2![\textbf{if } (t_0+t_1=1) \textbf{ then } t_1 \textbf{ else } 0]\,.\\ cnt![\textbf{if } (c_0+c_1=1) \textit{ and } (t_0+t_1=1) \textbf{ then } b=1 \textbf{ else } b=0]\,.\,\mathbf{0} \end{aligned}$$

Counter represents the coincidence measurement in optical experiments. Coincidence is observed by detecting two photons, one at channel u and the other at v. It also provides the correct output of the CNOT gate in terms of classical bits through the channels out_1 and out_2. The coincidence count (b) is recorded as 1 at the output of the channel cnt. The unsuccessful outcomes of the CNOT gate are recorded as 0 at the three output channels. This is determined by the **if**...**else** conditions in the definition. When we consider the correctness of the system, we will prove that $Model_1$ is equivalent to the following $Specification_1$ process. We use the same process $PolSe_{CT}$ as the input for $Specification_1$.

$$Specification_1(a,b,out_1,cnt,out_2) = (\mathsf{new}\ c,d,e,f,g)(PolSe_{CT}(\widetilde{U}) \mid OP(\widetilde{C}) \mid Output(\widetilde{D}))$$

There may be other ways of expressing the specification, for example without converting the polarisation qubit into the spatial encoding, but we do not investigate them in the present paper. Here, $\widetilde{C}$ is a list of channels containing c,d,e,f,g,h,i,j,k and $\widetilde{D}$ consists $g,h,i,j,k,out_1,cnt,out_2$. *OP* performs the CNOT operation with a certain probability and is defined by

$$\begin{aligned} OP(c,d,e,f,g,h,i,j,k) = (\mathsf{qbit}\ :q_2)\,.\,c?[s_0]\,.\,d?[s_1]\,.\,e?[s_2]\,.\,f?[s_3]\,.\,\{s_2,s_3 *= \mathsf{H}\}\,.\,\{q_2 *= \mathsf{U}_{\frac{1}{9}}\}\,.\\ \{(s_0,s_1),(s_2,s_3) *= \mathsf{CZ}\}\,.\,\{s_2,s_3 *= \mathsf{H}\}\,.\,h![s_0]\,.\,i![s_1]\,.\,j![s_2]\,.\,k![s_3]\,.\,g![\mathsf{measure}\ q_2]\,.\,\mathbf{0} \end{aligned}$$

OP possesses a qubit q_2 (initialised to $|0\rangle$). A random bit is generated with certain probability ($\frac{8}{9}$ for bit 0) by measuring q_2 after the unitary operation with $U_{\frac{1}{9}}$. This is followed by a series of unitary operations namely Hadamard operation (H) which is applied twice on a pair of number states (s_2,s_3) and a controlled Z (CZ) where s_0,s_1 acts as the control pair and s_2,s_3 is the target pair. The combination of a H, CZ and another H constitutes a CNOT, which is an abstract version of the number state computation. The theory of these operators for number states are not discussed in this paper but are provided in [14]. The data are then communicated to the process *Output*:

$$\begin{aligned} Output(g,h,i,j,k,out_1,cnt,out_2) = g?[x:\mathsf{bit}]\,.\,h?[s_0]\,.\,i?[s_1]\,.\,j?[s_2]\,.\,k?[s_3]\,.\,cnt![x]\,.\\ out_1![\textbf{if } (x=1) \textbf{ then } \mathsf{measure}\ s_1 \textbf{ else } 0]\,.\,out_2![\textbf{if } (x=1) \textbf{ then } \mathsf{measure}\ s_3 \textbf{ else } 0]\,.\,\mathbf{0} \end{aligned}$$

This gives the correct output in the form of classical bits of the CNOT operation when x equals one, which is artificially making the specification work with a certain probability ($\frac{1}{9}$). When x equals zero, the specification does not work and we get zero at all the output channels.

$$(\tilde{x}:\tilde{T};\sigma;\omega;u+v) \longrightarrow_v (\tilde{x}:\tilde{T};\sigma;\omega;w) \text{if } u \text{ and } v \text{ are integer literals and } w=u+v \quad \text{(R-PLUS)}$$

$$(\tilde{x}:\tilde{T};[\tilde{x}\mapsto \sum_{\tilde{s}\geq 0}\alpha_{\tilde{s}}|\beta_{\tilde{s}}\rangle|\tilde{s}\rangle];\omega;\mathsf{measure}\ s_a,s_b) \longrightarrow_v \oplus_{k,l\geq 0} g_{kl}(\tilde{x}:\tilde{T};[\tilde{x}\mapsto \sum_{\tilde{s'}\geq 0}\frac{\alpha_{\tilde{s'}}}{\sqrt{g_{kl}}}|\beta_{\tilde{s'}}\rangle|\widetilde{s'}\rangle];\omega;\lambda yz\bullet(y,z);k,l) \text{ where } g_{kl}=\sum_{\tilde{i}}|\alpha_{\tilde{s'}}|^2,$$

(R-MEASURE-NS-2)

$$\tilde{s}=s_0,\ldots,s_{n-1},\tilde{s'}=s_0,\ldots,s_{a-1},k,\ldots,l,s_{b+1},\ldots,s_{n-1},\tilde{i}=s_0,\ldots,s_{n-1}\setminus(s_a,s_b) \text{and } (a,b)\in\{0,\ldots,n-1\} \text{ and } a\neq b$$

$$(\tilde{x}:\tilde{T};[\tilde{x}\mapsto \sum_{\tilde{s}\geq 0}\alpha_{\tilde{s}}|\beta_{\tilde{s}}\rangle|\tilde{s}\rangle];\omega;\mathsf{psmeasure}\ s_a,s_b) \longrightarrow_v \oplus_{k,l\in\{0,1\},k\neq l} h_{kl}(\tilde{x}:\tilde{T};[\tilde{x}\mapsto \sum_{\tilde{s'}\geq 0}\frac{\alpha_{\tilde{s'}}}{\sqrt{h_{kl}}}|\beta_{\tilde{s'}}\rangle|\widetilde{s'}\rangle];\omega;\lambda z\bullet z;l) \text{ where } h_{kl}=\sqrt{g_{op}}\frac{1}{\sum_{\tilde{j}}|\alpha_{\tilde{s'}}|^2}$$

(R-PS-MEASURE)

$$\text{and } g_{op}=\sum_{\tilde{i}}|\alpha_{\tilde{s'}}|^2,\ o,p\geq 0,\ \tilde{s}=s_0,\ldots,s_{n-1},\tilde{s'}=s_0,\ldots,s_{a-1},o,\ldots,p,s_{b+1},\ldots,s_{n-1},$$

$$\tilde{i}=s_0,\ldots,s_{n-1}\setminus(s_a,s_b)\ \tilde{s'}=s_0,\ldots,s_{a-1},k,\ldots,l,s_{b+1},\ldots,s_{n-1},$$

$$\text{and } \tilde{j}=s_0,\ldots,s_{a-1},k,\ldots,l,s_{b+1},\ldots,s_{n-1} \text{ and } (a,b)\in\{0,\ldots,n-1\} \text{ and } a\neq b$$

$$(q_0,\ldots,q_{n-1}=\alpha_0|\phi_0\rangle+\cdots+\alpha_{2^n-1}|\phi_{2^n-1}\rangle;\omega;\mathsf{measure}\ q_0,\ldots,q_{r-1}) \longrightarrow_v \oplus_{0\leq m<2^r} g_m\ (q_0,\ldots,q_{n-1}=\frac{\alpha_{l_m}}{\sqrt{g_m}}|\phi_{l_m}\rangle+\cdots+\frac{\alpha_{u_m}}{\sqrt{g_m}}|\phi_{u_m}\rangle;\omega;\lambda x\bullet x;m)$$

(R-MEASURE-QBIT)

$$\text{where } l_m=2^{n-r}m, u_m=2^{n-r}(m+1)-1, g_m=|\alpha_{l_m}|^2+\cdots+|\alpha_{u_m}|^2$$

$$(\tilde{q}:\mathsf{Qbit},\tilde{s}:\mathsf{NS};[\tilde{q},\tilde{s}\mapsto|\psi\rangle];\omega;s_0,\ldots,s_{2r-1}*=U) \longrightarrow_v (\tilde{q}:\mathsf{Qbit},\tilde{s}:\mathsf{NS};[\tilde{q},s_0,\ldots,s_{n-1}\mapsto(I_{|\tilde{q}|}\otimes U\otimes I_{(n-2r)})|\psi\rangle];\omega;\mathsf{unit})$$

(R-TRANS-NS)

$$(\tilde{x}:\tilde{T};\sigma;\omega;\text{if true then } e \text{ else } e') \longrightarrow_v (\tilde{x}:\tilde{T};\sigma;\omega;e) \text{ or } (\tilde{x}:\tilde{T};\sigma;\omega;\text{if false then } e' \text{ else } e) \longrightarrow_v (\tilde{x}:\tilde{T};\sigma;\omega;e') \quad \text{(R-IFTHEN)}$$

$$\frac{\forall i\in I.(\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\psi_i\rangle];\omega;e\{\widetilde{u_i}/\widetilde{y}\}) \longrightarrow_v \oplus_{j\in J_i} g_{ij}\ (\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\psi_{ij}\rangle];\omega;\lambda\widetilde{z}\bullet e'\{\widetilde{u_i}/\widetilde{y}\};\widetilde{v_{ij}})}{\oplus_{i\in I} h_i\ (\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\psi_i\rangle];\omega;\lambda\widetilde{y}\bullet E[e];\widetilde{u_i}) \longrightarrow_e \oplus_{\substack{i\in I\\ j\in J_i}} h_i g_{ij}\ (\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\psi_{ij}\rangle];\omega;\lambda\widetilde{y}\widetilde{z}\bullet E[e'];\widetilde{u_i},\widetilde{v_{ij}})} \quad \text{(R-CONTEXT)}$$

Figure 5: Transition rules for values and expressions.

3.4 Semantics of CQP

In the previous section, we have described informally the behaviour of the processes which represent the linear optical elements of the CNOT gate model. In this section we will explain the formal semantics of CQP although without giving all of the definitions. Full definitions are in the Appendix. The execution of a system is not completely described by the process term (which is the case for classical process calculus) but also depends on the quantum state. Hence the operational semantics are defined using *configurations*, which represent both the quantum state and the process term.

Definition 2 (Configuration) *A configuration is defined as a tuple of the form* $(\tilde{x}:\tilde{T};\sigma;\omega;P)$ *where* $\tilde{x}$ *is a list of names (qubits* $\tilde{q}$*, number states* $\tilde{s}$ *or both) associated with their types* $\tilde{T}$*,* σ *is a mapping from names (*$\tilde{x}$*) to the quantum state and* ω *is a list of names associated with the process P*

We operate with configurations such as $(q_1:\mathsf{Qbit},s_0:\mathsf{NS},s_1:\mathsf{NS};[q_1,s_0,s_1\mapsto(|0\rangle|10\rangle+|1\rangle|01\rangle)];q_1;c![q_1].P)$

This configuration means that the global quantum state consists of a qubit, q_1, number states s_0 and s_1, in the specified state; that the process term under consideration has access to qubit q_1 but not to the number states; and that the process itself is $c![q_1].P$. The semantics of CQP consists of labelled transitions between configurations which is essentially required for the equivalence of processes. We now present the complete *labelled transition rules* of CQP that are extended from the previous work in order to verify LOQC, which is the focus of this paper.

Expression Transition Rules : For the evaluation of expressions we also introduce *expression configurations* $(\tilde{x}:\tilde{T};\sigma;\omega;e)$, which are similar to configurations, but include an expression in place of the process. The semantics of expressions is defined by the reduction relations $\longrightarrow_v$ (on values) and

$$(\widetilde{p},\widetilde{q}:\mathsf{Q\tilde{b}it},\widetilde{r},\widetilde{s}:\mathsf{N\tilde{S}},[\widetilde{pqrs}\mapsto|\psi\rangle];\widetilde{p},\widetilde{q},\widetilde{r},\widetilde{s};c![\widetilde{v},\widetilde{q},\widetilde{s}].P)\xrightarrow{c![\widetilde{v},\widetilde{q},\widetilde{s}]}_p(\widetilde{p},\widetilde{q}:\mathsf{Q\tilde{b}it},\widetilde{r},\widetilde{s}:\mathsf{N\tilde{S}},[\widetilde{pqrs}\mapsto|\psi\rangle];\widetilde{p},\widetilde{r};P) \quad \text{(P-OUT)}$$

$$(\widetilde{q}:\mathsf{Q\tilde{b}it},\widetilde{s}:\mathsf{N\tilde{S}},[\widetilde{qs}\mapsto|\psi\rangle];\omega;c?[\widetilde{y},\widetilde{x}].P)\xrightarrow{c?[\widetilde{v},\widetilde{p},\widetilde{r}]}_p(\widetilde{q}:\mathsf{Q\tilde{b}it},\widetilde{s}:\mathsf{N\tilde{S}},[\widetilde{qs}\mapsto|\psi\rangle];\omega,\widetilde{p},\widetilde{r};P\{\widetilde{v},\widetilde{r}/\widetilde{y},\widetilde{p}/\widetilde{x}\}) \quad \text{(P-IN)}$$

$$\frac{(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;P)\xrightarrow{\alpha}_p(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega';P')}{(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;P\parallel Q)\xrightarrow{\alpha}_p(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega';P'\parallel Q)} \quad \text{(P-PAR)}$$

$$\frac{(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;P)\xrightarrow{\alpha}_p(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega';P')}{(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;P+Q)\xrightarrow{\alpha}_p(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega';P')} \quad \text{(P-SUM)}$$

$$\frac{(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;P)\xrightarrow{\alpha}_p(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;P')}{(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;(\mathsf{new}\ c)P)\xrightarrow{\alpha}_p(\widetilde{x}:\tilde{T},[\widetilde{x}\mapsto|\psi\rangle];\omega;(\mathsf{new}\ c)P')}\ \text{if } \alpha\notin\{c?[\cdot],c![\cdot]\} \quad \text{(P-RES)}$$

$$(\tilde{x},\tilde{y}:\mathsf{Q\tilde{b}it},q:\mathsf{Qbit},\tilde{z}:\mathsf{N\tilde{S}};[\tilde{x},q,\tilde{y},\tilde{z}\mapsto|\phi\rangle];\omega;\{s,t*=\mathsf{PS}(q)\}.P)\xrightarrow{\tau}(\tilde{x},\tilde{y}:\mathsf{Q\tilde{b}it},\tilde{z}:\mathsf{N\tilde{S}},s:\mathsf{NS},t:\mathsf{NS};[\tilde{x},\tilde{y},\tilde{z},s,t\mapsto|\psi\rangle];\omega';P) \quad \text{(P-PS)}$$

where $|\phi\rangle=|\alpha\rangle|0\rangle|\beta\rangle|\gamma\rangle+|\alpha'\rangle|1\rangle|\beta'\rangle|\gamma'\rangle$, $|\psi\rangle=|\alpha\rangle|\beta\rangle|\gamma\rangle|10\rangle+|\alpha'\rangle|\beta'\rangle|\gamma'\rangle|01\rangle q\in\omega$ and $s,t\notin\omega$, $q\notin\omega'$ and $s,t\in\omega'$

Figure 6: Transition rules for pure process configurations.

$\longrightarrow_e$ (on expressions), given in Figure 5. Rule R-PLUS deal with the evaluation of terms that result in values. Rules R-MEASURE-NS-2,R-PS-MEASURE and R-MEASURE-QBIT are measurement rules which produces a mixed configuration. First two rules measure a pair of number states and the last rule measures qubit. R-MEASURE-NS-2 produces a mixed configuration over the possible measurement outcomes k and l. The measurement outcomes are classical values which are the number of photons detected. R-PS-MEASURE is a *post-selective* measurement rule which produces a mixed configuration over the possible measurement outcome l. Rule R-TRANS-NS deals with unitary transformations which result in literal unit. The important aspect of R-TRANS-NS and the measurement rules is the effect they have on the quantum state.

The rule R-CONTEXT has two primary purposes; it is used for the evaluation of expressions in an expression context E and it is also used of the evaluation of expressions in mixed configurations. The evaluation of a mixed expression configuration $\oplus_{i\in I}\ h_i\ (\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\psi_i\rangle];\omega;\lambda\widetilde{y}\bullet E[e];\widetilde{u_i})$ is determined by the evaluation of each component. For a given component, the pure expression configuration is obtained by substitution of the respective values; $(\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\psi_i\rangle];\omega;E[e]\{\widetilde{u_i}/\widetilde{y}\})$. For this configuration we isolate the context and consider the evaluation of $e\{\widetilde{u_i}/\widetilde{y}\})$. The resulting configuration may be a mixed expression configuration with new variables $\widetilde{z}$ introduced; specifically we end up with a term $\lambda\widetilde{z}\bullet e'\{\widetilde{u_i}/\widetilde{y}\};\widetilde{v_{ij}}$ where, due to the use of the substitution, e' is constant across each i. The results for each i are combined to give the final term $\lambda\widetilde{yz}\bullet E[e'];\widetilde{u_i},\widetilde{v_{ij}}$ incorporating variables $\widetilde{z}$ and $\widetilde{y}$.

Pure Configuration Transition Rules : The rules for pure process configurations are given in Figure 6. This defines the input and output transitions for pure configurations. It is used in the hypothesis of L-OUT-QBIT, L-OUT-NS and L-COM to determine the actions of the individual components in a mixed configurations. The rules namely choice (P-SUM), parallel (P-PAR) and restriction (P-RES) are required to define input and output actions for arbitrary process constructions. These rules are applicable for both qubits and number states and P-PS is for the conversion of polarisation qubit to the number states.

Mixed Configuration Transition Rules : The transition relation on mixed configurations is defined by the rules in Figure 7. The rule L-PROB is a probabilistic transition in which p_i is the probability of the transition. The rules L-IN, L-OUT-QBIT and L-OUT-NS represent the input and output actions respectively, which are the visible interactions with the environment. $P\{\widetilde{v_i}/\widetilde{z}\}$ indicates that P with a list of values v_i is substituted for the list of variables $\widetilde{z}$. When the two processes of input and output actions

$$\boxplus_j p_j(\oplus_i g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;P_i)) \overset{p_i}{\rightsquigarrow} \oplus_i g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;P_I) \qquad \text{(L-PROB)}$$

$$\oplus_i g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet c?[\widetilde{q},\widetilde{s}].P;\widetilde{v_i}) \overset{c?[\widetilde{p},\widetilde{r}]}{\longrightarrow} \oplus_i g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega,\widetilde{r},\widetilde{p};\lambda\widetilde{z}\bullet P\{\widetilde{p}/\widetilde{q},\widetilde{r}/\widetilde{s}\};\widetilde{v_i}) \qquad \text{(L-IN)}$$

$$\frac{\forall i\in I.((\widetilde{p},\widetilde{q}):\widetilde{\mathsf{Qbit}},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{pqs}\mapsto|\alpha_i\rangle|\beta\rangle];\widetilde{p},\widetilde{s};P\{\widetilde{v_i}/\widetilde{x}\}) \overset{c![\widetilde{u_i},\widetilde{r}]}{\longrightarrow}_p ((\widetilde{p},\widetilde{q}):\widetilde{\mathsf{Qbit}},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{pqs}\mapsto|\alpha_i\rangle|\beta\rangle];\widetilde{p'},\widetilde{s};P'\{\widetilde{v_i}/\widetilde{x}\})}{\oplus_{i\in I} g_i\ ((\widetilde{p},\widetilde{q}):\widetilde{\mathsf{Qbit}},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{pqs}\mapsto|\alpha_i\rangle|\beta\rangle];\widetilde{p},\widetilde{s};\lambda\widetilde{x}\bullet P;\widetilde{v_i}) \overset{c![U,\widetilde{r}]}{\longrightarrow} \boxplus_{j\in J} p_j(\oplus_{i\in I_j} \frac{g_i}{p_j}\ ((\widetilde{p'},\widetilde{q}):\widetilde{\mathsf{Qbit}},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{p'rqs}\mapsto\Pi|\alpha_i\rangle|\beta\rangle];\widetilde{p'},\widetilde{s};\lambda\widetilde{x}\bullet P';\widetilde{v_i}))}$$

$$\text{where } U=\{\widetilde{u_i}\mid i\in I\}=\{\widetilde{w_j}\mid j\in J\} \text{ and } \forall j\in J, I_j=\{i\mid\widetilde{u_i}=\widetilde{w_j}\}, p_j=\sum_{i\in I_j} g_i$$

$$\text{and } \widetilde{r}\subseteq\widetilde{p},\widetilde{p'}=\widetilde{p}\setminus\widetilde{r},\Pi \text{ corresponds to the permutation } \pi:\widetilde{pqs}\mapsto\widetilde{p'rqs}\,.$$

(L-QBIT)

$$\frac{\forall i,j\in I.(\widetilde{p}:\widetilde{\mathsf{Qbit}},(\widetilde{t},\widetilde{s}):\widetilde{\mathsf{NS}};[\widetilde{pts}\mapsto|\alpha\rangle|\beta_{ij}\rangle];\widetilde{p},\widetilde{s};P\{\widetilde{v_{ij}}/\widetilde{x},\widetilde{w_{ij}}/\widetilde{y}\}) \overset{c![\widetilde{u_{ij}},\widetilde{r}]}{\longrightarrow}_p (\widetilde{p}:\widetilde{\mathsf{Qbit}},(\widetilde{t},\widetilde{s}):\widetilde{\mathsf{NS}};[\widetilde{pts}\mapsto|\alpha\rangle|\beta_{ij}\rangle];\widetilde{p},\widetilde{s'};P'\{\widetilde{v_{ij}}/\widetilde{x},\widetilde{w_{ij}}/\widetilde{y}\})}{\oplus_{i,j\in I} g_{ij}\ (\widetilde{p}:\widetilde{\mathsf{Qbit}},(\widetilde{t},\widetilde{s}):\widetilde{\mathsf{NS}};[\widetilde{pts}\mapsto|\alpha\rangle|\beta_{ij}\rangle];\widetilde{p},\widetilde{s};\lambda\widetilde{xy}\bullet P;\widetilde{v_{ij}},\widetilde{w_{ij}}) \overset{c![U,\widetilde{r}]}{\longrightarrow} \boxplus_{k\in J} p_k(\oplus_{i,j\in I_k} \frac{g_{ij}}{p_k}\ (\widetilde{p}:\widetilde{\mathsf{Qbit}},(\widetilde{t},\widetilde{s'}:\widetilde{\mathsf{NS}});[\widetilde{pts'r}\mapsto\Pi|\alpha\rangle|\beta_{ij}\rangle];\widetilde{p},\widetilde{s'};\lambda\widetilde{xy}\bullet P';\widetilde{v_{ij}},\widetilde{w_{ij}}))}$$

$$\text{where } U=\{\widetilde{u_{ij}}\mid i,j\in I\}=\{\widetilde{e_k}\mid k\in J\}, \text{ and } \forall k\in J, I_k=\{i,j\mid\widetilde{u_{ij}}=\widetilde{e_k}\}, p_k=\sum_{i,j\in I_k} g_{ij}$$

$$\text{and } \widetilde{r}\subseteq\widetilde{s},\widetilde{s'}=\widetilde{s}\setminus\widetilde{r},\Pi \text{ corresponds to the permutation } \pi:\widetilde{pts}\mapsto\widetilde{ptrs'}\,.$$

(L-OUT-NS)

$$\frac{\forall i\in I.(\widetilde{x}:\widetilde{T};\sigma_i;\omega,\widetilde{r};P\{\widetilde{v_i}/\widetilde{z}\}) \overset{c![\widetilde{u_i},\widetilde{r}]}{\longrightarrow}_p (\widetilde{x}:\widetilde{T};\sigma_i;\omega;P'\{\widetilde{v_i}/\widetilde{z}\}) \quad \forall i\in I.(\widetilde{x}:\widetilde{T};\sigma_i;\omega;Q\{\widetilde{v_i}/\widetilde{z}\}) \overset{c?[\widetilde{u_i},\widetilde{r}]}{\longrightarrow}_p (\widetilde{x}:\widetilde{T};\sigma_i;\omega,\widetilde{r};Q'\{\widetilde{v_i}/\widetilde{z}\})}{\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega,\widetilde{r};\lambda\widetilde{z}\bullet P\parallel Q;\widetilde{v_i}) \longrightarrow \oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega,\widetilde{r};\lambda\widetilde{z}\bullet P'\parallel Q';\widetilde{v_i})} \qquad \text{(L-COM)}$$

$$\frac{\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet P;\widetilde{v_i}) \overset{\alpha}{\longrightarrow} \oplus_{\substack{i\in I\\ j\in J_i}} g_i h_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega';\lambda\widetilde{zy}\bullet P';\widetilde{v_i},\widetilde{w_{ij}})}{\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet P\parallel Q;\widetilde{v_i}) \overset{\alpha}{\longrightarrow} \oplus_{\substack{i\in I\\ j\in J_i}} g_i h_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega';\lambda\widetilde{zy}\bullet P'\parallel Q;\widetilde{v_i},\widetilde{w_{ij}})} \qquad \text{(L-PAR)}$$

$$\frac{\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet P;\widetilde{v_i}) \overset{\alpha}{\longrightarrow} \oplus_{\substack{i\in I\\ j\in J_i}} g_i h_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega';\lambda\widetilde{zy}\bullet P';\widetilde{v_i},\widetilde{w_{ij}})}{\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet P+Q;\widetilde{v_i}) \overset{\alpha}{\longrightarrow} \oplus_{\substack{i\in I\\ j\in J_i}} g_i h_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega';\lambda\widetilde{zy}\bullet P';\widetilde{v_i},\widetilde{w_{ij}})} \qquad \text{(L-SUM)}$$

$$\frac{\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet P;\widetilde{v_i}) \overset{\alpha}{\longrightarrow} \oplus_{\substack{i\in I\\ j\in J_i}} g_i h_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega';\lambda\widetilde{zy}\bullet P';\widetilde{v_i},\widetilde{w_{ij}})}{\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet(\mathsf{new}\ c)P;\widetilde{v_i}) \overset{\alpha}{\longrightarrow} \oplus_{\substack{i\in I\\ j\in J_i}} g_i h_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega';\lambda\widetilde{zy}\bullet(\mathsf{new}\ c)P';\widetilde{v_i},\widetilde{w_{ij}})} \qquad \text{(L-RES)}$$

$$\text{if } \alpha\notin\{c?[\cdot],c![\cdot]\}$$

$$\oplus_{i\in I} g_i\ (\widetilde{q}:\widetilde{\mathsf{Qbit}},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{qs}\mapsto|\beta_i\rangle|\gamma_i\rangle];\omega;\lambda\widetilde{z}\bullet(\mathsf{qbit}\ :y)P;\widetilde{v_i}) \longrightarrow \oplus_{i\in I} g_i\ (\widetilde{q}:\widetilde{\mathsf{Qbit}},q:\mathsf{Qbit},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{q},q,\widetilde{s}\mapsto|\beta_i\rangle|\phi_j\rangle|\gamma_i\rangle];\omega,q;\lambda\widetilde{z}\bullet P\{q/y\};\widetilde{v_i})$$

where q is fresh (L-QBIT)

$$\oplus_{i\in I} g_i\ (\widetilde{q}:\widetilde{\mathsf{Qbit}},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{qs}\mapsto|\beta_i\rangle|\gamma_i\rangle];\omega;\lambda\widetilde{z}\bullet(\mathsf{ns}\ :y)P;\widetilde{X}) \overset{\tau}{\longrightarrow} \oplus_{i\in I} g_i\ (\widetilde{q}:\widetilde{\mathsf{Qbit}},r:\mathsf{NS},\widetilde{s}:\widetilde{\mathsf{NS}};[\widetilde{q},r,\widetilde{s}\mapsto|\beta_i\rangle|\psi_j\rangle|\gamma_i\rangle];\omega,r;\lambda\widetilde{m}\bullet P\{r/y\};\widetilde{v_i})$$

where r is fresh (L-NS)

$$\oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet\{u\}.P_i;\widetilde{v_i}) \overset{\tau}{\longrightarrow} \oplus_{i\in I} g_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{z}\bullet P;\widetilde{v_i}) \qquad \text{(L-ACT)}$$

$$\begin{aligned}&\oplus_{i\in I} g_i\ (\tilde{p},\tilde{q}:\tilde{\mathsf{Qbit}},q_c:\mathsf{Qbit},\tilde{r}:\tilde{\mathsf{NS}};[\tilde{p},q_c,\tilde{q},\tilde{r}\mapsto|\phi\rangle];\omega;\lambda\widetilde{z}\bullet\{s_a,s_b*{=}\,\mathsf{PS}(q_c)\};.P,\widetilde{v_i}) \qquad \text{(L-PS)}\\ &\overset{\tau}{\longrightarrow}\oplus_{i\in I} g_i\ (\tilde{p},\tilde{q}:\tilde{\mathsf{Qbit}},\tilde{r}:\tilde{\mathsf{NS}},s_a:\mathsf{NS},s_b:\mathsf{NS};[\tilde{p},\tilde{q},\tilde{r},s_a,s_b\mapsto|\psi\rangle];\omega';\lambda\widetilde{z}\bullet P;,\widetilde{v_i})\\ &\frac{\oplus_{i\in I} h_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{y}\bullet e;\widetilde{v_i}) \longrightarrow_e \oplus_{\substack{i\in I\\ j\in J_i}} h_i g_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega;\lambda\widetilde{yz}\bullet e';\widetilde{v_i},\widetilde{w_{ij}})}{\oplus_{i\in I} h_i\ (\widetilde{x}:\widetilde{T};\sigma_i;\omega;\lambda\widetilde{y}\bullet F[e];\widetilde{v_i}) \overset{\tau}{\longrightarrow} \oplus_{\substack{i\in I\\ j\in J_i}} h_i g_{ij}\ (\widetilde{x}:\widetilde{T};\sigma_{ij};\omega;\lambda\widetilde{yz}\bullet F[e'];\widetilde{v_i},\widetilde{w_{ij}})} \qquad \text{(L-EXPR)}\end{aligned}$$

Figure 7: Transition rules for mixed process configurations.

are put in parallel then each has a partner for its potential interaction, and the input and output can synchronise, resulting in a τ transition which is given by the rule L-COM. The rule L-ACT just removes actions. This is a reduction of the action expression to v which would involve effects like measurement or transformation of the quantum state. Rules L-QBIT and L-NS are for introducing additional Qbit and NS variables respectively. ns declarations represents vacuum states. Since the values associated with the an input action are determined by the environment, this action is identical across all components in a mixed configuration. L-PAR, L-SUM and L-RES can then be used to define inputs on other process constructions in a mixed configuration.

The rule L-OUT-QBIT and L-OUT-NS is the point at which mixed configurations are combined with probabilistic branching. Branching must occur when and only when there is information to distinguish the components. This information is represented by the classical values that are outputs, which may vary between the components. Some values may be the same, thereby requiring the relevant components to remain in a mixed configuration after the output. The purpose of L-OUT-QBIT and L-OUT-NS is to distribute the components according to the different values, and to assign an action label that represents the combined action of *all* components. For example in transition L-OUT-QBIT, each component has a pure transition $\xrightarrow{c![\widetilde{u}_i,\widetilde{r}]}_p$ representing the channel and qubit names that are common to all components, and the values $\widetilde{u}_i$ that are specific to that component. The combined action label $\xrightarrow{c![U,\widetilde{r}]}$ consists of these common elements and the set U of all the value tuples.

Example 1 $(q,s,t:\widetilde{T};[q,s,t\mapsto\alpha_{10}|0\rangle|10\rangle+\alpha_{01}|1\rangle|01\rangle+\alpha_{20}|0\rangle|20\rangle];q,s,t;c![\mathsf{measure}\ s,t].P)$
$\xrightarrow{\tau}\oplus_{i,j\in\{0,1,2\}}\ |\alpha_{ij}|^2(q,s,t:\widetilde{T};[q,s,t\mapsto|\beta\rangle|ij\rangle];q,s,t;\lambda yz\bullet c![y,z].P;i,j).$

This transition represents the effect of a measurement of a pair of number states (s, t), within a process which is going to output the result of the measurement. The configuration on the left is a *pure configuration*, as described before. On the right we have a *mixed configuration* in which the $\oplus$ ranges over the possible outcomes of the measurement and the $|\alpha_{ij}|^2$ are the weights of the components in the mixture. The quantum state $[q,s,t\mapsto|\beta\rangle|ij\rangle]$ corresponds to the measurement outcome. The expression $\lambda yz\bullet c![y,z].P$ represents the fact that the components of the mixed configuration have the same process structure and differ only in the values corresponding to measurement outcomes. The final terms in the configuration, i and j, shows how the abstracted variables y and z should be instantiated in each component. Thus the λyz represents a term into which expressions may be substituted, which is the reason for the λ notation. The next transition (R-PS-MEASURE) represents *post-selective* measurement which filters out the measurement values that satisfies a predefined criteria.

Example 2 $(q,s,t:\widetilde{T};[q,s,t\mapsto\alpha_{10}|0\rangle|10\rangle+\alpha_{01}|1\rangle|01\rangle+\alpha_{20}|0\rangle|20\rangle];q,s,t;c![\mathsf{psmeasure}\ s,t].P)$
$\xrightarrow{\tau}\oplus_{i,j\in\{0,1\},i\neq j}\ |\beta_{ij}|^2(q,s,t:\widetilde{T};[q,s,t\mapsto|\delta\rangle|ij\rangle];q,s,t;\lambda y\bullet c![y].P;j).$

Unlike Example 1, here i and j can have values either 0 or 1 and $i\neq j$. This is the criterion for post-selection and the weights of the components in the mixture are now $|\beta_{ij}|^2$ (where $|\beta_{ij}|^2=\frac{|\alpha_{ij}|^2}{\sum_{ij\in\{0,1\}}|\alpha_{ij}|^2}$). Also, here we measure two number states s and t, which results in one classical value. Example 3 shows the effect of the output from the final configuration of Example 2.

Example 3 $\oplus_{i,j\in\{0,1\},i\neq j}\ |\beta_{ij}|^2(\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\delta\rangle|ij\rangle];\widetilde{x};\lambda y\bullet c![y].P;i)\xrightarrow{c![j]}\boxplus_{ij\in\{0,1\},i\neq j}|\beta_{ij}|^2$
$(\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\delta\rangle|ij\rangle];\widetilde{x};\lambda y\bullet P;j)\overset{|\beta_{01}|^2}{\rightsquigarrow}(\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|1\rangle|01\rangle];\widetilde{x};\lambda y\bullet P;1)$

Here $\widetilde{x}$ is a list of names consisting q, s and t. The output transition produces the intermediate configuration, which is a probability distribution over pure configurations (in contrast to a mixed configuration;

note the change from $\oplus$ to $\boxplus$). Because it comes from a mixed configuration, the output transition contains a *set* of possible values. From this intermediate configuration there are two possible probabilistic transitions, of which one is shown ($\overset{|\beta_{01}|^2}{\rightsquigarrow}$).

Example 4 $\oplus_{i,j\geq 0}\, g_{\,ij}(\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\beta\rangle|ij\rangle];\widetilde{x};\lambda yz\bullet(c![y]\,.\,P\,|\,c?[y]\,.\,Q);i,j)\xrightarrow{\tau}$
$\oplus_{i,j\geq 0}\, g_{\,ij}(\widetilde{x}:\widetilde{T};[\widetilde{x}\mapsto|\beta\rangle|ij\rangle];\widetilde{x};\lambda yz\bullet(P\,|\,Q);i,j)$

Measurement outcomes may be communicated between processes without creating a probability distribution. In these cases an observer must still consider the system to be in a mixed configuration as the outcomes are communicated internally and not to the environment.

Example 5 $(q:\mathsf{Qbit},r:\mathsf{Qbit},p:\mathsf{NS},t:\mathsf{NS};[q,r,p,t\mapsto\alpha|00\rangle|10\rangle+\beta|11\rangle|01\rangle];q,r,p,t;\{u:\mathsf{NS},v:\mathsf{NS}*=\mathsf{PS}(q)\}\,.\,P)\xrightarrow{\tau}(r:\mathsf{Qbit},\tilde{s'}:\tilde{\mathsf{NS}};[r,\tilde{s'}\mapsto\alpha|0\rangle|1010\rangle+\beta|1\rangle|0101\rangle];r,\tilde{s'};P)$

Example 5 represents the transition P-PS, which is the conversion of a polarisation qubit (q) to the number states (u and v). $\tilde{s'}$ indicates that it is a list of names comprising p,t,u and v of type NS.

3.5 Execution of $Model_1$

Let $t=(\emptyset;\emptyset;\emptyset;Model_1)$ be the initial configuration. The semantics of CQP is non-deterministic and hence the transitions can proceed in different order. In the first few steps, the process $PolSe_{CT}$ receives qubits q_1 and q_2 from the environment, constructing a global quantum state $|\phi\rangle_q=\alpha|00\rangle+\beta|01\rangle+\gamma|10\rangle+\delta|11\rangle$. We get the configuration, $(q_1:\mathsf{Qbit},q_2:\mathsf{Qbit},q_1q_2=|\phi\rangle_q;q_1,q_2;(PolSe_{CT}{}'\,|\,CNOT\,|\,MMT))$. After some τ transitions corresponding to $PolSe_{CT}$ operations, the qubits are converted to the respective number states s_0,s_1,s_2 and s_3 by PS operator giving the quantum state $|\phi\rangle_s=\alpha|1010\rangle+\beta|1001\rangle+\gamma|0110\rangle+\delta|0101\rangle$. The configuration is now $(\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi\rangle_s;s_0,s_1,s_2,s_3;(PolSe_{CT}{}''\,|\,CNOT\,|\,MMT))$. After another set of τ transitions corresponding to the $CNOT$ process, we get the state $|\phi\rangle_{out}$ which is given by Eq. 3. The configuration now becomes $(\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi\rangle_{out};s_0,s_1,s_2,s_3;(CNOT'\,|\,MMT))$. After the measurement by both detectors, the outcomes are communicated to the *Counter*. This happens internally and hence, we get the mixed configuration:

$$\oplus_{\substack{ij\geq 0\\ kl\geq 0}}\, g_{ij}h_{ijkl}\;(\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi_{ijkl}\rangle;s_0,s_1,s_2,s_3;\lambda\widetilde{y}\bullet Counter';i,j,k,l)$$

Here $\widetilde{y}$ is a list of measurement outcomes (c_0,c_1,t_0 and t_1). The output transitions produces the configuration below, which is a mixed state given by $\oplus_{i,j,k,l,m\in\{0,1\}}\, g_{ijm}h_{ijklm}\;(\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi_{ijkl}\rangle;\widetilde{s};\lambda\widetilde{z}\bullet\mathbf{0};i,j,k,l,m)$ where $\widetilde{z}$ is c_1,t_1,b. The mixture contains both the successful and unsuccessful outcomes of $Model_1$.

4 Behavioural Equivalence of CQP Processes

We now extend the theory of equivalence in CQP to apply it for LOQC. The process calculus approach to verification is to define a process *Model* which models the system of interest, another process *Specification* which expresses the specification that *Model* should satisfy, and then prove that *Model* and *Specification* are equivalent. We begin with the definition of *probabilistic branching bisimilarity*, which is a congruence for CQP.

4.1 Probabilistic Branching Bisimilarity

There are several types of probabilistic bisimilarity for classical probabilistic process calculi, including *probabilistic branching bisimilarity* [19]. The equivalence for CQP defined by Davidson [3], which turns out to be a congruence, is a form of probabilistic branching bisimilarity, adapted to the situation in which probabilistic behaviour comes from quantum measurement. A key point is that when considering matching of input or output transitions involving qubits, it is the reduced density matrices of the transmitted qubits that are required to be equal. We will now define probabilistic branching bisimilarity in full. The definitions in the remainder of this section are an extension from Davidson's thesis [3].

Notation: Let $\xrightarrow{\tau}{}^{+}$ denote zero or one τ transitions; let $\Longrightarrow$ denote zero or more τ transitions; and let $\overset{\alpha}{\Longrightarrow}$ be equivalent to $\Longrightarrow\xrightarrow{\alpha}\Longrightarrow$. We write $\widetilde{q}$ for a list of qubit names, and similarly for other lists.

Definition 3 (Density Matrix of Configurations) *Let* $\sigma_{ij} = [\widetilde{x} \mapsto |\psi_{ij}\rangle]$ *and* $\widetilde{y} \subseteq \widetilde{x}$ *and* $t_{ij} = (\widetilde{x} : \widetilde{T}; \sigma_{ij}; \omega; \lambda\widetilde{w} \bullet P; \widetilde{v_{ij}})$ *and* $t = \oplus_{ij}\, g_{ij}\, t_{ij}$. *Then*

1. $\rho(\sigma_{ij}) = |\psi_{ij}\rangle\langle\psi_{ij}|$
2. $\rho^{\widetilde{y}}(\sigma_{ij}) = \mathrm{tr}_{\widetilde{x}\setminus\widetilde{y}}(|\psi_{ij}\rangle\langle\psi_{ij}|)$
3. $\rho(t_{ij}) = \rho(\sigma_{ij})$
4. $\rho^{\widetilde{y}}(t_{ij}) = \rho^{\widetilde{y}}(\sigma_{ij})$
5. $\rho(t) = \sum_{ij} g_{ij}\rho(t_{ij})$
6. $\rho^{\widetilde{y}}(t) = \sum_{ij} g_{ij}\rho^{\widetilde{y}}(t_{ij})$

We also introduce the notation ρ_E to denote the reduced density matrix of the *environment* qubits or number states. Formally, if $t = (\widetilde{x} : \widetilde{T}; [\widetilde{x} \mapsto |\psi\rangle]; \widetilde{y}; P)$ then $\rho_E(t) = \rho^{\widetilde{r}}(t)$ where $\widetilde{r} = \widetilde{x} \setminus \widetilde{y}$. The definition of ρ_E is extended to mixed configurations in the same manner as ρ. The probabilistic function $\mu : \mathscr{S} \times \mathscr{S} \to [0,1]$ is defined in the style of [19]. It allows non-deterministic transitions to be treated as transitions with probability 1, which is necessary when calculating the total probability of reaching a terminal state. $\mu(t,u) = \delta$ if $t \overset{\delta}{\rightsquigarrow} u$; $\mu(t,u) = 1$ if $t = u$ and $t \in \mathscr{T}_n$; $\mu(t,u) = 0$ otherwise.

Definition 4 (Probabilistic Branching Bisimulation) *An equivalence relation* $\mathscr{R}$ *on configurations is a* probabilistic branching bisimulation *on configurations if whenever* $(t,u) \in \mathscr{R}$ *the following conditions are satisfied.*

I. *If* $t \in \mathscr{T}_n$ *and* $t \xrightarrow{\tau} t'$ *then* $\exists u', u''$ *such that* $u \Longrightarrow u' \xrightarrow{\tau}{}^{+} u''$ *with* $(t,u') \in \mathscr{R}$ *and* $(t',u'') \in \mathscr{R}$.

II. *If* $t \xrightarrow{c![V,\widetilde{X}_1]} t'$ *where* $t' = \boxplus_{j\in\{1\ldots m\}} p_j t'_j$ *and* $V = \{\widetilde{v}_1, \ldots, \widetilde{v}_m\}$ *and* $\widetilde{X}_1$ *is either* $\widetilde{q}_1$ *or* $\widetilde{s}_1$ *then* $\exists u', u''$ *such that* $u \Longrightarrow u' \xrightarrow{c![V,\widetilde{X}_2]} u''$ *with*
   a) $(t,u') \in \mathscr{R}$,
   b) $u'' = \boxplus_{j\in\{1\ldots m\}} p_j u''_j$,
   c) *for each* $j \in \{1,\ldots,m\}$, $\rho_E(t'_j) = \rho_E(u''_j)$.
   d) *for each* $j \in \{1,\ldots,m\}$, $(t'_j, u''_j) \in \mathscr{R}$.

III. *If* $t \xrightarrow{c?[\widetilde{v}]} t'$ *then* $\exists u', u''$ *such that* $u \Longrightarrow u' \xrightarrow{c?[\widetilde{v}]} u''$ *with* $(t,u') \in \mathscr{R}$ *and* $(t',u'') \in \mathscr{R}$.

IV. *If* $s \in \mathscr{T}_p$ *then* $\mu(t,D) = \mu(u,D)$ *for all classes* $D \in \mathscr{T}/\mathscr{R}$.

This relation follows the standard definition of branching bisimulation [8] with additional conditions for probabilistic configurations and matching quantum information. In condition II we require that the distinct set of values V must match and although the names ($\widetilde{X}_1$ and $\widetilde{X}_2$) need not be identical which is either the qubit names ($\widetilde{q}_1$ and $\widetilde{q}_2$) or number state names ($\widetilde{s}_1$ and $\widetilde{s}_2$), their respective reduced density matrices ($\rho^{\widetilde{X}_1}(t)$ and $\rho^{\widetilde{X}_2}(u')$) must. Condition IV provides the matching on probabilistic configurations

following the approach of [19]. It is necessary to include the latter condition to ensure that the probabilities are paired with their respective configurations. This leads to the following definitions. The essential definitions are presented in this paper and the others are provided in the appendix.

Definition 5 (Probabilistic Branching Bisimilarity) *Configurations t and u are* probabilistic branching bisimilar, *denoted $t \leftrightarrows u$, if there exists a probabilistic branching bisimulation $\mathscr{R}$ such that $(t,u) \in \mathscr{R}$.*

Definition 6 (Probabilistic Branching Bisimilarity of Processes) *Processes P and Q are* probabilistic branching bisimilar, *denoted $P \leftrightarrows Q$, if and only if for all σ, $(\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \leftrightarrows (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; Q)$.*

Definition 7 (Full probabilistic branching bisimilarity) *Processes P and Q are* full probabilistic branching bisimilar, *denoted $P \leftrightarrows^c Q$, if for all substitutions κ and all quantum states σ, $(\widetilde{x} : \widetilde{T}; \sigma; \widetilde{q}, \widetilde{s}; P\kappa) \leftrightarrows (\widetilde{x} : \widetilde{T}; \sigma; \widetilde{q}, \widetilde{s}; Q\kappa)$.*

In order to state the *congruence* theorem, we need an assumption that processes are typable. Its essential idea is to associate each qubit or number state with a unique owning component of the process. In particular this means that when we consider a process P in a context, $C[P]$, the context cannot manipulate quantum state that is owned by P. The full type system is a straightforward extension of the system from CQP, taking account of number states.

Theorem 1 (Full probabilistic branching bisimilarity is a congruence) *If $P \leftrightarrows^c Q$ then for any context $C[]$, if $C[P]$ and $C[Q]$ are typable then $C[P] \leftrightarrows^c C[Q]$.*

4.2 Correctness of $Model_1$

We now sketch the proof that $Model_1 \leftrightarrows^c Specification_1$, which by Theorem 1 implies that the LOQC CNOT gate works in any context.

Proposition 1 $Model_1 \leftrightarrows^c Specification_1$.

Proof: First we prove that $Model_1 \leftrightarrows Specification_1$, by defining an equivalence relation $\mathscr{R}$ that contains the pair $((\widetilde{x} : \widetilde{T}; \sigma; \emptyset; Model_1), (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; Specification_1))$ for all σ and is closed under their transitions. $\mathscr{R}$ is defined by taking its equivalence classes to be the $F_i(\sigma)$ defined below, for all states σ, which group configurations according to the sequences of observable transitions leading to them.

$$
\begin{aligned}
F_1(\sigma, q_1) &= \{ f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]}{\Longrightarrow} f \text{ and } P \in E \} \\
F_2(\sigma, q_1, q_2) &= \{ f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]}{\Longrightarrow} \stackrel{b?[q_2]}{\Longrightarrow} f \text{ and } P \in E \} \\
F_3(\sigma, q_2) &= \{ f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]}{\Longrightarrow} \stackrel{b?[q_2]}{\Longrightarrow} \stackrel{out_1![c_1]}{\Longrightarrow} f \text{ and } P \in E \} \\
F_4(\sigma) &= \{ f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]}{\Longrightarrow} \stackrel{b?[q_2]}{\Longrightarrow} \stackrel{out_1![c_1]}{\Longrightarrow} \stackrel{out_2![c_2]}{\Longrightarrow} f \text{ and } P \in E \} \\
F_5(\sigma) &= \{ f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]}{\Longrightarrow} \stackrel{b?[q_2]}{\Longrightarrow} \stackrel{out_1![c_1]}{\Longrightarrow} \stackrel{out_2![c_2]}{\Longrightarrow} \stackrel{cnt![y]}{\Longrightarrow} f \text{ and } P \in E \}
\end{aligned}
$$

Here E is $\{Model_1, Specification_1\}$ and we now prove that $\mathscr{R}$ is a probabilistic branching bisimulation. It suffices to consider transitions between F_i classes, as transitions within classes must be τ and are matched by τ. If $f, g \in F_1(\sigma)$ and $f \stackrel{a?[q_1]}{\longrightarrow} f'$ then $f' \in F_2(\sigma)$ and we find g', g'' such that $g \Longrightarrow g' \stackrel{a?[q_1]}{\longrightarrow} g''$ with $g' \in F_1(\sigma)$ and $g'' \in F_2(\sigma)$, so $(f, g') \in \mathscr{R}$ and $(f', g'') \in \mathscr{R}$ as required. Transitions from $F_2(\sigma)$,$F_3(\sigma)$ and $F_4(\sigma)$ are matched similarly. There are no transitions from $F_5(\sigma)$. There is no need for a probability calculation (case IV of Definition 4) because the probabilistic configurations do not arise as the measurement results are communicated internally. Finally, because $Model_1$ and $Specification_1$ have no free variables, their equivalence is trivially preserved by substitutions. □

5 LOQC CNOT Gate: A Second Model

The first model includes an explicit implementation of the *post-selection* procedure, meaning that the specification process has to include the success probability of $\frac{1}{9}$. We now consider a more abstract model, by introducing a new measurement operator which includes *post-selection* and restricts attention to the successful outcomes. This is achieved by replacing the process *MMT* of our first model by the process *PSM* which performs *post-selective* measurement and enables a simpler specification to be used. The CQP definition of $Model_2$ is given as $Model_2(\widetilde{A}) = (\mathsf{new}\ \widetilde{B})(PolSe_{CT}(\widetilde{C}) \,|\, CNOT(\widetilde{D}) \,|\, PSM(\widetilde{E}))$. Processes $PolSe_{CT}$ and *CNOT* are defined in the previous model. The process *PSM* is defined as $PSM(\widetilde{E}) = PDet_1(\widetilde{F}) \,|\, PDet_2(\widetilde{G})$. We prove that $Model_2$ is equivalent to $Specification_2$:

$$\begin{aligned}
OPCNOT(\widetilde{C}) &= c?[s_0].d?[s_1].e?[s_2].f?[s_3].\{s_2,s_3 *= \mathsf{H}\}.\\
&\{(s_0,s_1),(s_2,s_3) *= \mathsf{CZ}\}.\{s_2,s_3 *= \mathsf{H}\}.h![s_0].i![s_1].j![s_2].k![s_3].\mathbf{0}\\
Output(\widetilde{D}) &= h?[s_0].i?[s_1].j?[s_2].k?[s_3].out_1![\mathsf{measure}\ s_1].out_2![\mathsf{measure}\ s_3].\mathbf{0}\\
Specification_2(\widetilde{A}) &= (\mathsf{new}\ \widetilde{E})(PolSe_{CT}(\widetilde{B}) \,|\, OPCNOT(\widetilde{C}) \,|\, Output(\widetilde{D}))
\end{aligned}$$

The analysis of $Model_2$ and the proof of its correctness are provided in the Appendix.

6 Conclusion and Future Work

The main contribution of this paper is the extension of theory of equivalence of CQP to verify linear optical quantum computing. This is the first work in using quantum process calculus to verify a physical realisation of quantum computing. We have defined the linear optical elements in CQP, and have described and analysed two models of the linear optical experimental system that demonstrates a CNOT gate. Using our second model, we have also described and verified post-selection in CQP.

These two models use different measurement semantics in order to work at different levels of abstraction. This shows that the process calculus is flexible enough to support a range of descriptions, from detailed hardware implementations up to more abstract specifications. The importance of process calculus is that it provides a systematic methodology for verification of quantum systems. The essential property that the equivalence is a congruence guarantees that equivalent processes remain equivalent in any context, and supports equational reasoning. The fact that CQP can also express classical behaviour means that we have a uniform framework in which to analyze classical and quantum computation and communication.

Shor's algorithm operating on four qubits using the basic linear optical elements has been demonstrated [17]. In this paper, we present the modelling of these elements with a future aim to formally analyse quantum algorithms in CQP using LOQC. This provides a platform to learn about quantum complexity in LOQC using CQP and also to verify it. The long-term goal is to develop software for automated analysis of CQP models, following the established work in classical process calculus and recent work on automated equivalence checking of concurrent quantum programs [1].

References

[1] E. Ardeshir-Larijani, S. J. Gay & R. Nagarajan (2014): *Verification of Concurrent Quantum Protocols by Equivalence Checking*. In: *Proceedings of the 20th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, 8413, LNCS, pp. 500–514, doi:10.1007/978-3-642-54862-8_42.

[2] T. A. S. Davidson, S. J. Gay, R. Nagarajan & I. V. Puthoor (2011): *Analysis of a Quantum Error Correcting Code using Quantum Process Calculus*. In: *Proceedings of the International Workshop on Quantum Physics and Logic (QPL)*, 95, EPTCS, pp. 67–80, doi:10.4204/EPTCS.95.7.

[3] Timothy A. S. Davidson (2011): *Formal Verification Techniques using Quantum Process Calculus*. Ph.D. thesis, University of Warwick.

[4] Yuan Feng, Runyao Duan & Mingsheng Ying (2011): *Bisimulation for quantum processes*. In: *Proceedings of the 38th Annual ACM Symposium on Principles of Programming Languages*, ACM, pp. 523–534, doi:10.1145/1926385.1926446.

[5] S. Franke-Arnold, S. J. Gay & I. V. Puthoor (2013): *Quantum process calculus for linear optical computing*. In: *Proceedings of the 5th Conference on Reversible Computation (RC)*, 7948, LNCS, pp. 234–246, doi:10.1007/978-3-642-38986-3_19.

[6] Simon J. Gay & Rajagopal Nagarajan (2005): *Communicating Quantum Processes*. In: *Proceedings of the 32nd Annual ACM Symposium on Principles of Programming Languages*, ACM, pp. 145–157, doi:10.1145/1040305.1040318.

[7] Simon J. Gay & Rajagopal Nagarajan (2006): *Types and Typechecking for Communicating Quantum Processes*. *Mathematical Structures in Computer Science* 16(3), pp. 375–406, doi:10.1017/S0960129506005263.

[8] Rob J. van Glabbeek & W. Peter Weijland (1996): *Branching time and abstraction in bisimulation semantics*. *Journal of the ACM* 43(3), pp. 555–600, doi:10.1145/233551.233556.

[9] IDQ: *ID Quantique*. Available at http://www.idquantique.com/company/presentation.html.

[10] E. Knill, R. Laflamme & G. J. Milburn (2001): *A scheme for efficient quantum computation with linear optics*. *Nature* 409, p. 46, doi:10.1038/35051009.

[11] T. Kubota, Y. Kakutani, G. Kato, Y. Kawano & H. Sakurada (2012): *Application of a process calculus to security proofs of quantum protocols*. In: *Proceedings of WORLDCOMP/FCS2012*.

[12] Robin Milner (1989): *Communication and Concurrency*. Prentice-Hall.

[13] Robin Milner (1999): *Communicating and Mobile Systems: the Pi-Calculus*. Cambridge University Press.

[14] C. R. Myers & R. Laflamme (2005): *Linear Optics Quantum Computation: an Overview*. *arXiv: quant-ph/0512104v1*.

[15] M. A. Nielsen & I. L. Chuang (2000): *Quantum Computation and Quantum Information*. Cambridge University Press.

[16] J. L. O'Brien, G. J. Pryde, A. G. White, T. C. Ralph & D. Branning (2003): *Demonstration of an all-optical quantum controlled-NOT gate*. *Nature* 426, p. 264, doi:10.1038/nature02054.

[17] A. Politi, J. C. F. Matthews & J. L. O'Brien (2009): *Shor's Quantum Factoring Algorithm on a Photonic Chip*. *Science* 325, p. 1221, doi:10.1126/science.1173731.

[18] T. C. Ralph, N. K. Lanford, T. B. Bell & A. G. White (2002): *Linear optical controlled-NOT gate in the coincidence basis*. *Physical Review Letters A* 65, pp. 062324–1, doi:10.1103/PhysRevA.65.062324.

[19] Nikola Trčka & Sonja Georgievska (2008): *Branching Bisimulation Congruence for Probabilistic Systems*. *Electronic Notes in Theoretical Computer Science* 220(3), pp. 129 – 143, doi:10.1016/j.entcs.2008.11.023.

[20] Andrew K. Wright & Matthias Felleisen (1994): *A syntactic approach to type soundness*. *Information and Computation* 115(1), pp. 38–94, doi:10.1006/inco.1994.1093.

7 Appendix

7.1 Definitions and Lemmas for Equivalence

Definition 8 (Context) *A* context *C is a process with a non-degenerate occurrence of* **0** *replaced by a hole,* $[\cdot]$. *Formally,*

$$C ::= [] \mid (C \parallel P) \mid \alpha.C + P \mid \alpha.C \mid (\mathsf{new}\ \hat{x}[T])C$$

for $\alpha \in \{e?[\tilde{x}:\tilde{T}], e![\tilde{e}], \{e\}, (\mathsf{qbit}\ x), (\mathsf{ns}\ r)\}$.

Definition 9 (Congruence) *An equivalence relation $\mathscr{R}$ on processes is a* congruence *if $(C[P], C[Q]) \in \mathscr{R}$ whenever $(P,Q) \in \mathscr{R}$ and C is a context.*

Definition 10 (Non-input, non-qubit or non-number state context) *A* non-input, non-qubit or non-number state context *is a context in which the hole does not appear under an input or qubit and number state declaration.*

Definition 11 (Non-input, non-qubit or non-number state congruence) *An equivalence relation $\mathscr{R}$ on processes is a* non-input, non-qubit or non-number state congruence *if $(C[P], C[Q]) \in \mathscr{R}$ whenever $(P,Q) \in \mathscr{R}$ and C is a non-input, non-qubit or non-number state context.*

The first lemma provides a general form for representing mixed configurations related by internal transitions. Due to space constraints the proofs of all lemmas and theorems are not provided in this paper.

Lemma 1 (General form of internal transitions) *If* $t = \oplus_{\substack{ab\in I_{kl}\\ kl\in J}} g_{abkl}\ (\widetilde{x}:\widetilde{T}; \sigma_{abkl}; \widetilde{q}, \widetilde{s}; \lambda\widetilde{y}\widetilde{z} \bullet P; \widetilde{w}_{abkl})$ *and* $t \Longrightarrow t'$ *then there exist sets* I'_{kl} *such that* $t' = \oplus_{\substack{ab\in I'_{kl}\\ kl\in J}} g'_{abkl}\ (\widetilde{x}:\widetilde{T}; \sigma'_{abkl}; \widetilde{q}', \widetilde{s}'; \lambda\widetilde{y}'\widetilde{z}' \bullet P'; \widetilde{w}'_{abkl})$.

The following 3 lemmas prove that the state of qubits and number states that are not owned by a particular process is unaffected by any transitions of that process.

Lemma 2 (External state independence for $\longrightarrow_v$) *If* $\Gamma;\ \widetilde{s} \vdash e : T$ *and* $t \longrightarrow_v t'$ *where* $t = (\widetilde{s}:\widetilde{\mathsf{NS}}, \widetilde{q}:\widetilde{\mathsf{Qbit}}, \widetilde{r}:\widetilde{\mathsf{Qbit}}; [\widetilde{s}\widetilde{q}\widetilde{r} \mapsto |\psi\rangle]; \widetilde{q}, \widetilde{s}; e)$ *then* $\rho^{\widetilde{q}\widetilde{r}}(t) = \rho^{\widetilde{q}\widetilde{r}}(t')$

Lemma 3 (External state independence for $\longrightarrow_e$) *If* $\Gamma;\ \widetilde{s} \vdash e : T$ *and* $t \longrightarrow_e t'$ *where* $t = \oplus_{kl\in I}\ g_{\ kl}(\widetilde{s}:\widetilde{\mathsf{NS}}, \widetilde{q}:\widetilde{\mathsf{Qbit}}, \widetilde{r}:\widetilde{\mathsf{Qbit}}; [\widetilde{s}\widetilde{q}\widetilde{r} \mapsto |\psi_{kl}\rangle]; \widetilde{q}, \widetilde{s}; \lambda\widetilde{y} \bullet e; \widetilde{w}_{kl})$ *then* $\rho^{\widetilde{q}\widetilde{r}}(t) = \rho^{\widetilde{q}\widetilde{r}}(t')$

Lemma 4 (External state independence for $\xrightarrow{\tau}$) *If* $\Gamma;\ \widetilde{s} \vdash P$ *and* $t \xrightarrow{\tau} t'$ *where* $t = \oplus_{kl\in I}\ g_{\ kl}(\widetilde{s}:\widetilde{\mathsf{NS}}, \widetilde{q}:\widetilde{\mathsf{Qbit}}, \widetilde{r}:\widetilde{\mathsf{Qbit}}; [\widetilde{s}\widetilde{q}\widetilde{r} \mapsto |\psi_{kl}\rangle]; \widetilde{q}, \widetilde{s}; \lambda\widetilde{y} \bullet P; \widetilde{w}_{kl})$ *then* $\rho^{\widetilde{q}\widetilde{r}}(t) = \rho^{\widetilde{q}\widetilde{r}}(t')$

The next lemma proves that the action of a context on the quantum state is independent of the quantum subsystem owned by a process.

Lemma 5 (Independence of context transitions) *Assume that* $\Gamma;\ \widetilde{s}_R \vdash R$. *Let t and u be configurations where*

$$t = \oplus_{kl\in I}\ g_{\ kl}(\widetilde{x}:\widetilde{T}; [\widetilde{q}_P\widetilde{q}_R\widetilde{q}_E\widetilde{s}_P\widetilde{s}_R\widetilde{s}_E \mapsto |\psi_{kl}\rangle]; \widetilde{q}_P, \widetilde{q}_R, \widetilde{s}_P, \widetilde{s}_R; \lambda\widetilde{y} \bullet R; \widetilde{w}_R)$$

$$u = \oplus_{mn\in J}\ h_{\ mn}(\widetilde{x}:\widetilde{T}; [\widetilde{q}_Q\widetilde{q}_R\widetilde{q}_E\widetilde{s}_Q\widetilde{s}_R\widetilde{s}_E \mapsto |\phi_{mn}\rangle]; \widetilde{q}_Q, \widetilde{q}_R, \widetilde{s}_Q, \widetilde{s}_R; \lambda\widetilde{y} \bullet R; \widetilde{w}_R)$$

If $\rho^{\widetilde{q}_P\widetilde{q}_E\widetilde{s}_P\widetilde{s}_E}(t) = \rho^{\widetilde{q}_Q\widetilde{q}_E\widetilde{s}_Q\widetilde{s}_E}(u)$ *and* $t \xrightarrow{\tau} t'$ *where* $t = \oplus_{\substack{kl\in I'_{ab}\\ ab\in K}}\ g'_{klab}(\widetilde{x}:\widetilde{T}; [\widetilde{q}_P\widetilde{q}'_R\widetilde{q}_E\widetilde{s}_P\widetilde{s}'_R\widetilde{s}_E \mapsto |\psi_{klab}\rangle]; \omega_P, \omega'_R; \lambda\widetilde{y}' \bullet R'; \widetilde{w}_{R_{ab}})$ *then there exists* $u = \oplus_{\substack{mn\in J'_{ab}\\ ab\in K}}\ h'_{mnab}(\widetilde{x}:\widetilde{T}; [\widetilde{q}_Q\widetilde{q}'_R\widetilde{q}_E\widetilde{s}_Q\widetilde{s}'_R\widetilde{s}_E \mapsto |\phi_{mnab}\rangle]; \omega_Q, \omega'_R; \lambda\widetilde{y}' \bullet R'; \widetilde{w}_{R_{ab}})$ *such that* $u \xrightarrow{\tau} u'$ *and* $\rho^{\widetilde{q}_P\widetilde{q}_E\widetilde{s}_P\widetilde{s}_E}(t') = \rho^{\widetilde{q}_Q\widetilde{q}_E\widetilde{s}_Q\widetilde{s}_E}(u')$

The next two lemmas prove some simple results which are used in the proof of Theorem 2.

Lemma 6 *Let* $t = \oplus_{kl\in I}\ g_{\ kl}t_{kl}$ *and* $t' = \oplus_{kl\in I}\ g_{\ kl}t'_{kl}$ *then* $t \xrightarrow{\alpha} t'$ *if and only if* $\forall_{kl\in I}(t_{kl} \xrightarrow{\alpha} t'_{kl})$ *for* $\alpha \in \{.?[\cdot], \tau\}$

Lemma 7 *Let* $t_{mn} = \oplus_{kl\in I_{mn}}\ g_{\ klmn}(\widetilde{x}:\widetilde{T};\sigma_{klmn};\omega;\lambda\widetilde{y}\bullet P;\widetilde{w}_{klmn})$ *and* $t_{klmn} = (\widetilde{x}:\widetilde{T};\sigma_{klmn};\omega;P\{\widetilde{w}_{klmn}/\widetilde{y}\})$ *then* $\forall_{mn\in J,kl\in I_{mn}}.(t_{klmn} \xrightarrow{c?[\widetilde{u}_{mn},\widetilde{q},\widetilde{s}]}_p t'_{klmn})$ *if and only if* $\forall_{mn\in J}.(t_{mn} \xrightarrow{c?[\widetilde{u}_{mn},\widetilde{q},\widetilde{s}]}_p t'_{mn})$

We are now in a position to prove that bisimilarity is preserved by parallel composition. To prove this, we define an equivalence relation that contains the pair $((\widetilde{x}:\widetilde{T};\sigma;\emptyset;P\,|\,R),(\widetilde{x}:\widetilde{T};\sigma;\emptyset;Q\,|\,R))$ and that is closed under transitions from these configurations.

Theorem 2 (Parallel preservation for configurations) *Assume that* $\Gamma\vdash P$, $\Gamma\vdash Q$, $\Gamma\vdash P\,|\,R$, *and* $\Gamma\vdash Q\,|\,R$. *If* $(\widetilde{x}:\widetilde{T};\sigma;\emptyset;P)\leftrightarrows(\widetilde{x}:\widetilde{T};\sigma;\emptyset;Q)$ *then* $(\widetilde{x}:\widetilde{T};\sigma;\emptyset;P\,|\,R)\leftrightarrows(\widetilde{x}:\widetilde{T};\sigma;\emptyset;Q\,|\,R)$.

Using this result, we prove that the bisimilarity of processes is preserved by parallel composition.

Theorem 3 (Parallel Preservation) *If* $P\leftrightarrows Q$ *then for any process* R *such that* $\Gamma\vdash P\,|\,R$ *and* $\Gamma\vdash Q\,|\,R$ *then* $P\,|\,R\leftrightarrows Q\,|\,R$.

We now consider preservation with respect to other process constructions and can be shown that probabilistic branching bisimilarity is preserved by all process constructs except input and qubit or number state declarations.

Lemma 8 *Probabilistic branching bisimilarity is preserved by output prefix, action prefix, channel restriction and non-deterministic choice.*

Theorem 4 (Probabilistic branching bisimilarity is a non-input congruence) *If* $P\leftrightarrows Q$ *and for any non-input, non-qubit or non-number state context* C *if* $\Gamma\vdash C[P]$ *and* $\Gamma\vdash C[Q]$ *then* $C[P]\leftrightarrows C[Q]$.

7.2 Execution of $Model_2$:

Let $t = (\emptyset;\emptyset;\emptyset;Model_2)$ be the initial configuration. Like in previous case after receiving input qubits, we get the configuration as, $(q_1:\mathsf{Qbit},q_2:\mathsf{Qbit},q_1q_2=|\phi\rangle_q;q_1,q_2;(PolSe_{CT}{}'\,|\,CNOT\,|\,PSM))$. As before the qubits are converted to the number states after some τ operations and the configuration is now,

$$(\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi\rangle_s;s_0,s_1,s_2,s_3;(PolSe_{CT}{}''\,|\,CNOT\,|\,PSM))$$

After another set of τ transitions corresponding to the *CNOT* process, we get the state $|\phi\rangle_{out}$ which is given by Eq. 3. The configuration now becomes $(\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi\rangle_{out};s_0,s_1,s_2,s_3;(CNOT'\,|\,PSM))$. The execution of $Model_2$ is similar to that of $Model_1$ and differs only in the measurement. Here the detectors perform a *post-selective* measurement giving rise to the following mixed configuration:

$$\oplus_{\substack{ij\in\{0,1\},i\neq j\\ kl\in\{0,1\},k\neq l}}\ g_{ij}h_{ijkl}\ (\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi_{ijkl}\rangle;s_0,s_1,s_2,s_3;\lambda\widetilde{y}\bullet PSM';j,l)$$

The *post-selective* measurement outcomes $(\widetilde{y})$ are then given as output to the environment resulting in a probabilistic configuration given as $\boxplus_{ij\in\{0,1\},kl\in\{0,1\}}g_{ij}h_{ijkl}(\widetilde{s}:\widetilde{\mathsf{NS}};\widetilde{s}=|\phi_{ijkl}\rangle;s_0,s_1,s_2,s_3;\lambda\widetilde{y}\bullet\mathbf{0};j,l)$.

Another significant difference between the models is in the communication of the measurement outcomes. In $Model_1$, the outcomes were communicated internally and hence did not give a probabilistic configuration, which is not the case for $Model_2$.

7.3 Correctness of $Model_2$

Proposition 2 *$Model_2 \leftrightarrows^c Specification_2$.*

Proof: We have similar equivalence classes as in the previous case:

$$\begin{aligned}
F_1(\sigma, q_1) &= \{f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]}{\Longrightarrow} f \text{ and } P \in E\} \\
F_2(\sigma, q_1, q_2) &= \{f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]b?[q_2]}{\Longrightarrow\Longrightarrow} f \text{ and } P \in E\} \\
F_3(\sigma, q_2) &= \{f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]b?[q_2]out_1![c_1]}{\Longrightarrow\Longrightarrow\Longrightarrow} f \text{ and } P \in E\} \\
F_4(\sigma) &= \{f \mid (\widetilde{x} : \widetilde{T}; \sigma; \emptyset; P) \stackrel{a?[q_1]b?[q_2]out_1![c_1]out_2![c_2]}{\Longrightarrow\Longrightarrow\Longrightarrow\Longrightarrow} f \text{ and } P \in E\}
\end{aligned}$$

Here E is $\{Model_2, Specification_2\}$ and the proof is similar to the previous case. In $Model_2$, we will always get a correct output since we do not consider any error and the probability of getting one of the outputs is $\frac{1}{4}$. Similar to the previous proof, here we have no transitions from $F_4(\sigma)$. □