Formal Verification of Optical Quantum Flip Gate

Mohamed Yousri Mahmoud[1], Vincent Aravantinos[2], and Sofiène Tahar[1]

[1] Electrical and Computer Engineering Dept., Concordia University,
1455 De Maisonneuve Blvd. W., Montreal, Canada
{mo_solim,tahar}@ece.concordia.ca
http://hvg.ece.concordia.ca
[2] Software and Systems Engineering, Fortiss GmbH,
Gürickestraße 25, 80805, Munich, Germany
aravantinos@fortiss.org
http://www.fortiss.org/en

Abstract. Quantum computers are promising to efficiently solve hard computational problems, especially NP problems. In this paper, we propose to tackle the formal verification of quantum circuits using theorem proving. In particular, we focus on the verification of quantum computing based on coherent light, which is typically light produced by laser sources. We formally verify the behavior of the quantum flip gate in HOL Light: we prove that it can flip a zero-quantum-bit to a one-quantum-bit and vice versa. To this aim, we model two optical devices: the beam splitter and the phase conjugating mirror and prove relevant properties about them. Then by cascading the two elements and utilizing these properties, the complete model of the flip gate is formally verified. This requires the formalization of some fundamental mathematics like exponentiation of linear transformations.

Keywords: Quantum optics, Quantum flip gate, Beam splitter, Phase conjugating mirror, Theorem proving, HOL Light.

1 Introduction

Classical computers (i.e., Turing machines) inefficiently solve hard computational problems, e.g., NP and NP-complete problems. In 1980, Feynman proposed a new machine model which uses quantum mechanics: the quantum computer [4]. This model showed that it can solve some hard problems in polynomial time: a well known example is Shor's algorithm for integer factorization [11]. This result has great consequences on computational theory in general, and security of systems in particular: quantum cryptography became a hot area of research where powerful and secure systems are developed. In addition, limitations are arising in the everlasting quest for more powerful classical computers: power dissipation problems, density limitations, and all their workarounds like multi-core systems. This all shows how important quantum computers could be in the future.

G. Klein and R. Gamboa (Eds.): ITP 2014, LNAI 8558, pp. 358–373, 2014.

The quantum computer model proposed by Feynman consists of a new notion of a bit, called quantum bit (abbreviated as *qbit*), and a set of universal quantum gates, e.g., the flip gate (the quantum counterpart of the classical NOT gate) and the Hadamard gate [17]. A quantum circuit is made of a collection of these gates and qbits. Different means and technologies can be used to implement this model, such as: superconducting circuits [1], ion traps [6], quantum dots [12] and optical circuits [8]. Optical circuits and ion traps are today the most promising ones since they can realize the highest number of bits in laboratory, till now [9]. In this work, we focus on optical circuits which serve as the basis of several implementations of quantum computers, e.g., [19] and [10]. A major task for each of these implementations is to make sure that it satisfies the proposed specifications in the original mathematical model. This verification process is of course very different from its counterpart for classical computers.

For quantum mechanics, and more specifically quantum optics, the available verification methods are lab-simulation and paper-and-pencil, the latter is assisted by numerical methods or computer algebra systems ("CAS"). In lab-simulation, the systems are simulated *physically* in an optical laboratory, i.e., a physical system is set up, whose basic components have properties similar to the ones of the intended system. It is then assumed that this simulation system will behave in a way similar to the actual system to be verified. Note that using computers for the simulation of quantum systems is so complex that it cannot be efficient enough to verify a complete system [4]. In the paper-and-pencil approach, the whole verification process is done by modeling the system and proving– using existing physics knowledge– that the system satisfies its specifications. However, this process is handled by a human and is thus very error-prone, particularly when the system is very large and especially when considering the complex mathematics that one has to deal with in quantum mechanics. Thus, computer methods are used to help the human and decrease the risk of errors: numerical methods (typically Matlab [20]) and Computer Algebra Systems ("CAS", typically Mathematica [3]). Both are used to help the simplification and generation of intermediate mathematical steps. However, these tools are not sufficient: they cannot fully substitute for the paper-and-pencil approach since they cannot mathematically express the whole model of the system. Moreover, they are also error-prone because of the numerical approximations and heuristics used in their computations. This is particularly true for complex computations involved in quantum mechanics. Therefore, we propose to use the theorem proving for the verification of quantum optical computers.

As a first step towards our ultimate goal, in this paper, we focus on the formalization of quantum computers implemented by coherent light (typically laser light). In particular, we formally verify the behavior of one of the universal quantum gates in this implementation, the *flip gate*. To this end, we have to consider the formalization of both physical and mathematical aspects. Mathematically, we implement the quantum operator exponentiation which is similar to exponentiation, but in infinite-dimension linear spaces. We then use this as well as some preliminary work presented in [13] and [14] to develop the theory of

coherent light. Coherent light is at the essential basis of two important optical elements: the beam splitter and the phase conjugate mirror, from which the flip gate can finally be built. This development demonstrates the theoretical feasibility of our approach: starting from the formalization of some abstract theory, we progressively build a model for concrete implementation of a practical quantum gate and verify that it has the expected behavior. *This work was completely implemented in HOL Light, the sources are available at [15].*

The rest of the paper is organized as follows: Section 2 gives preliminaries about quantum optics and quantum computers, and recalls the formalization of some of the foundational notions. Section 3 presents the formal development of the exponentiation of quantum operators. Section 4 describes the coherent light formalization and Section 5 deals with the flip gate verification and the formalization of the required devices. Finally, we conclude the paper in Section 6.

2 Preliminaries

In this section, we briefly introduce some notions of quantum computers and quantum optics, in particular optical coherent light. We then give more details about quantum operators that are useful in quantum optics, specifically when implementing a flip gate. We finally give the basic formal mathematical definitions that are used in our formalization.

2.1 Quantum Systems

A quantum system is fully described with a so-called *quantum state*, generally noted $|\psi\rangle$. Mathematically, a quantum state is a square integrable complex-valued function whose square integration is equal to one. Square integrable complex-valued functions form an inner product space whose product $\langle f|g\rangle$ is the integration of the multiplication of f by the conjugate of g.

For every system there is a (finite or infinite) set of quantum states $|\psi_1\rangle$, $|\psi_2\rangle$, ..., called *basis states*, which have the property that every state of the system can be expressed as a linear combination of them, i.e., for every state $|\psi\rangle$ of the system, there are complex numbers c_1, c_2, ... such that:

$$|\psi\rangle = \sum_{i=1,2,\ldots} |c_i| * |\psi_i\rangle \tag{1}$$

where $\sum |c_i|^2 = 1$.

An example of such a system is the basic component of the quantum computer: the quantum bit (or *qbit*). Similar to classical bits, a quantum bit is a quantum system with two basis states $|0\rangle$ and $|1\rangle$. However, contrary to its classical counterpart, the state of a qbit is not only $|0\rangle$ or $|1\rangle$, but can be a mix thereof. Indeed, such a state can be expressed as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where $|\alpha|^2 + |\beta|^2 = 1$ (according to Equation (1)).

Another example of a quantum system is light: in quantum optics, light is considered as a stream of particles called photons, in contrast to the classical theory

that considers light as an electromagnetic wave. As a quantum system, light has an infinite countable set of basis states $|0\rangle$, $|1\rangle$, ..., called *Fock states*. Light in a fock state $|n\rangle$ contains n photons. Light is said to be *coherent* if the number of photons in the light stream (at any time instant) is probabilistically Poisson distributed, i.e., the probability of having n photons is: $P(N = n) = \frac{|\alpha|^n \ e^{-|\alpha|}}{n!}$ for some complex number α. The modulus of $|\alpha|$ represents the expected number of observed photons. The coherent light is then in the quantum state $|\alpha\rangle$ which can be decomposed according to Equation (1) as follows:

$$|\alpha\rangle = e^{-\frac{|\alpha|^2}{2}} \sum_{n=0} \frac{\alpha^n}{\sqrt{n!}} |n\rangle \tag{2}$$

The essential idea of using quantum optics, and more specifically coherent light, to implement quantum computers is to realize the states $|0\rangle$ and $|1\rangle$ by the states $|0\rangle$ and $|\alpha\rangle$ of light, respectively.

2.2 Quantum Operators

Similar to classical physics, the state of a system can evolve over time. Actually, in the case of quantum physics, it can also evolve just by being observed. In any case, the evolution of a state must be a function mapping the state to another one. Since states are functions themselves, such a function is actually an operator. These operators are even restricted to be linear transformations over the state space.

In order to compute with qbits, one needs operators applied to them. As for classical circuits, this is achieved through *gates*. The quantum computer model is made of nine such gates, which we will not detail here since our focus in this paper is only one: *the quantum flip gate.* The flip gate (or Pauli-X gate) is equivalent to the classical NOT gate: applying it to $|0\rangle$ yields $|1\rangle$ and vice versa. However, due to its quantum nature, it is capable of much more: for any α, β, $\alpha|0\rangle + \beta|1\rangle$ is turned into $\alpha|1\rangle + \beta|0\rangle$.

In the case of optics, there are two basic quantum operators: the *creator* and *annihilator* operators. The creator operator is defined by:

$$\hat{a}^\dagger|n\rangle = \sqrt{n+1}|n+1\rangle \tag{3}$$

and the annihilator by:

$$\hat{a}|n\rangle = \sqrt{n}|n-1\rangle \tag{4}$$

As their names suggest, the annihilator $\hat{a}$ decreases the number of photons by one (i.e., destroys a photon) and the creator $\hat{a}^\dagger$ increases it by one. Note that the resulting quantum state is not exactly the demoted one, since it is scalar-multiplied by $\sqrt{n+1}$ and $\sqrt{n}$, respectively. However, scalar multiplication actually does not change a quantum state behavior. Thereby, the resulting state still has $n-1$ photons.

Solving Equation (3) as a recurrence relation, we obtain a general representation of any fock state $|n\rangle$:

$$|n\rangle = \frac{(\hat{a}^\dagger)^n \ |0\rangle}{\sqrt{n!}} \tag{5}$$

where $|0\rangle$ is called *vacuum* state since it does not contain any photon. Note here that the power notation used in $(\hat{a}^\dagger)^n$ means the application of the creation operator n times (recall that quantum operators are functions).

According to Equations (2) and (5), we can re-express coherent states in terms of the vacuum state and creation operator:

$$|\alpha\rangle = e^{-\frac{|\alpha|^2}{2}} \left(\sum_{n=0} \frac{(\alpha\hat{a}^\dagger)^n}{n!} \right) |0\rangle \tag{6}$$

Note that, for a linear operator $a^\dagger$, $(\alpha\hat{a}^\dagger)^n = \alpha^n(\hat{a}^\dagger)^n$.

This allows us to introduce the *displacement* operator $D(\alpha)$, which is essential for the implementation of the flip gate:

$$|\alpha\rangle = D(\alpha)|0\rangle \tag{7}$$

Here, $D(\alpha) = e^{\alpha\hat{a}^\dagger} \ e^{-\alpha^*\hat{a}} \ e^{[\alpha\hat{a}^\dagger, -\alpha^*\hat{a}]}$, where $*$ denotes the scalar multiplication with quantum operators, $**$ denotes the multiplication between quantum operators, and $[op_1, op_2] = op_1 ** op_2 - op_2 ** op_1$. The proof of Equation (7) can be found in the literature, e.g., in [16]. Note the use of exponentiation *over operators*, which is defined as follows:

$$e^{\hat{O}} = \sum_{i=0} \frac{\hat{O}^i}{i!} \tag{8}$$

Though defined similarly to the classical exponential, its properties are very different.

The importance of the displacement operator is that it can be physically realized by a quantum optical device called a *beam splitter* [18]. Therefore it is an essential ingredient in the implementation of quantum computers using coherent light, as we will see in Section 5.

2.3 Quantum State Space Formalization

After presenting the essential quantum physics notions, we now briefly review the formalization of inner product spaces which was presented in [13].

First, since a quantum state is a complex-valued function, we defined a HOL type for that: $\texttt{cfun} = \texttt{A} \rightarrow \texttt{complex}$, where $\texttt{c}fun$ stands for *c*omplex *fun*ction. A is a type variable, allowing our formalization to be used to model both finite-dimension systems like quantum computers, and infinite-dimension systems like quantum light.

Additions and scalar multiplications are defined easily as the corresponding point-wise operations, which allows us to characterize the notion of linear subspace as follows:

Definition 1

```
is_cfun_subspace (spc : cfun → bool) ⇔
  ∀x y. x IN spc ∧ y IN spc ⇒
  x + y IN spc ∧ (∀a. a%x IN spc) ∧ cfun_zero IN spc
```

where `cfun_zero` is the constantly null function, and `%` denotes the scalar multiplication. The notion of inner space is then defined as follows:

Definition 2

```
is_inner_space ((s, inprod) : (qs → bool) × (cfun → cfun → complex)) ⇔
    is_cfun_subspace s ∧
    ∀x. x ∈ s ⇒
      real (inprod x x) ∧ 0 ≤ real_of_complex (inprod x x) ∧
      (inprod x x = Cx(0) ⇔ x = qs_zero) ∧
      ∀y. y ∈ s ⇒
      cnj (inprod y x) = inprod x y ∧
      (∀a. inprod x (a%y) = a * (inprod x y)) ∧
      ∀z. z ∈ s  ⇒
      inprod (x + y) z = inprod x z + inprod y z
```

where `real x` states that the complex value `x` has no imaginary part, and `real_of_complex` is a function converting a complex number into a real one (if it is real).

Once these bases are set, we can define the notion of operator over an inner space. This is achieved by first defining the type `cop = cfun → cfun`. A linear operator is then characterized as follows:

Definition 3

```
 is_linear_cop (op : cop) ⇔
   ∀x y. op (x + y) = op x + op y  ∧ ∀a. op (a % x) = a % (op x)
```

In addition, quantum operators must satisfy the property of being self-adjoint:

Definition 4

```
is_self_adjoint (s, inprod) op ⇔
   is_inner_space (s, inprod) ⇒
     is_linear_cop op ∧
     ∀x y. inprod x (op y) = inprod (op x) y
```

As seen in the previous section, exponentiation of operators requires their infinite summation. We first define infinite summation over functions:

Definition 5

```
cfun_sums innerspc f l s ⇔
   cfun_lim innerspc (λn. cfun_sum (s INTER (0..n)) f) l sequentially
```

which formalizes the fact that $\lim_{n\to\infty} \sum_{i=0}^{n} f_i = l$: `INTER` is the sets intersection operator, `cfun_lim` is the notion of limit defined for quantum states, `cfun_sum` is finite summation over quantum states, and `sequentially` means that the

summation index will be increased sequentially, i.e., 1,2,3,.. More details about implementing infinite summation and related notions are presented in [14].

In practice it is more convenient to actually retrieve the limit in a functional way. To do so we use the Hilbert choice operator `@` as follows:

Definition 6
`cfun_infsum innerspc s f` = `@l. cfun_sums innerspc f l s`

This is useful only at the condition that the sum is convergent, which we express by the following predicate:

Definition 7
`cfun_summable innerspc s f` = $\exists$`l. cfun_sums innerspc f l s`

In conjunction with infinite summation, *bounded* operators are of particular importance. Indeed, the application of a bounded operator commutes with infinite summation: i.e., for a bounded operator `cop`:

$$\texttt{cop}\ (\texttt{cfun\_infsum f s}) = \texttt{cun\_infsum}\ (\lambda\texttt{n. cop}\ (\texttt{fn}))\ \texttt{s}.$$

Bounded operators are defined as follows:

Definition 8
`is_bounded (s, inprod) h` $\Leftrightarrow$ `is_inner_space (s, inprod)`
$\Rightarrow$ `is_closed_by s h` $\wedge$ $\exists$`B. 0` $<$ `B` $\wedge$
($\forall$`x. x IN s` $\Rightarrow$ `cfun_norm inprod (h x)`)) $\leq$ `B` $*$ `cfun_norm inprod x`)))

where `is_closed_by s h` $\Leftrightarrow$ $\forall$`x.x IN s` $\Rightarrow$ `h x IN s`, and `cfun_norm inprod x` = $\sqrt{\texttt{real\_of\_complex (inprod x x)}}$. A linear operator `h` is bounded if for all `x` the norm of `h x` is lower or equal to the norm of `x` up to multiplication by a scalar `B`. Note that `B` does not depend on `x`.

3 Quantum Operator Exponentiation

Quantum operator exponentiation is essential for the formalization of the displacement operator. In order to tackle the exponentiation, we have first to consider the infinite summation over quantum operators, which is done simply by using the pointwise infinite summation over complex functions:

Definition 9
`cop_sums (s, inprod) f l set` $\Leftrightarrow$ $\forall$`x. x IN s` $\Rightarrow$
`cfun_sums (s, inprod) (`λ`n.(f n) x) (l x) set`

This definition is an easy adaptation of the `cfun` case: the only differences are the types of `f`, `l`, and `set`, and the fact that the pointwise definition is restricted to the values that belong to the inner space. This latter point is very important since this summation might not exist for some operators, if defined over the complete extension of `cfun`: for instance, many sequences of square-integrable functions do not have a limit that remains square-integrable.

Similarly to `cfun_infsum` and `cfun_summable`, we then define `cop_infsum` and `cop_summable`:

Definition 10
cop_infsum innerspc s f = @l. cop_sums innerspc f l s
cop_summable innerspc s f = ∃l. cop_sums innerspc f l s

Finally, we can use cop_infsum to define quantum operator exponentiation according to Equation (8):

Definition 11
cop_exp innerspc (op : cfun → cfun) ⇔
cop_infsum innerspc (from 0) (λn. $\frac{1}{!n}$ % (op pow n))

where from 0 denotes the set ℕ. We prove many properties about the exponentiation but we will present in detail the proof of only one of them, and will only mention the end result for others. We start by proving that cop_exp (cop_zero) = I, which is the scalar counterpart of $e^0 = 1$. To do so, we first need to provide the property using the predicate definition, i.e., cop_sums, as follows:

Theorem 1
∀is. is_inner_space is ⇒
cop_sums innerspc (λn. $\frac{1}{!n}$ % (cop_zero pow n)) I (from 0)

where cop_zero = λx : cfun. cfun_zero is the operator constantly equal to cfun_zero. In addition, we recall that I is the identity operator (to ease the understanding, one can remark that it corresponds to the identity matrix in a finite dimension vector space). We then use the uniqueness of cop_infsum to re-express the property in terms of cop_exp. The unicity theorem is as follows:

Theorem 2
∀s inprod f set l x.
x IN s ∧ cop_sums (s, inprod) f l set ⇒
(cop_infsum (s, inprod) set f) x = l x

It states that *if the summation has a limit*, then this limit is unique. Therefore it is also equal to cop_infsum (s,inprod) set f on the considered inner space, since the definition of cop_infsum is precisely to be any of these limits. Note that we cannot ensure that cop_infsum (s,inprod) set f and l are equal since we do not know how they affect elements outside s. This is not a restriction, on the contrary: it ensures that our theory indeed has a non-trivial model. If this was not the case, the inner space of square-integrable functions could not be used with our formalization.

In the end, we obtain the following theorem stating indeed the intended property:

Theorem 3
∀s inprod x.
x IN s ∧ is_inner_space(s, inprod) ⇒
cop_exp (s, inprod) cop_zero x = x

Another important property is the commutativity of exponentiation with the scalar multiplication of its argument:

Theorem 4

$\forall$s inprod a x.
x IN s $\wedge$ is_inner_space(s, inprod) $\Rightarrow$
(cop_exp(s, inprod) (λy.a%y)) x = cexp a%x

The scalar counterpart of this theorem is the $e^{(a.1)} = e^a.1$: indeed the identity plays here the role of the unity. Note that this result shows the compatibility of our definitions with the ones defined in HOL Light for infinite dimension linear spaces.

Like for the scalar exponentiation, cop_exp is not a linear function over operators. However, a property which has no counterpart for scalars is the linearity of cop_exp op (which is an operator). This property is essential to the development of the flip gate: indeed, it allows to generalize the effect of the gate on basis states $|0\rangle$ and $|1\rangle$ to any mixed state $c_1\,|0\rangle + c_2|1\rangle$. It also helps a lot in the intermediate steps of many proofs, by allowing to move in and out scalar values multiplied by states, i.e., cop_exp op(a % x) = a%(cop_exp op x). The linearity of cop_exp op is however true only on the concerned inner space. Therefore, we need a definition which is relaxed w.r.t. Definition 3:

Definition 12

is_set_linear_cop s (op : cop) $\Leftrightarrow$
$\forall$x y. x IN s $\wedge$ y IN s $\Rightarrow$ op (x + y) = op x + op y $\wedge$
$\forall$a. op(a % x) = a % (op x)

The linearity of cop_exp op can then be proved, as long as op is itself a linear operator:

Theorem 5

$\forall$s inprod op.
cop_summable innerspc (from 0) (λn.$\frac{1}{!n}$ % (op pow n)$\wedge$is_linear_cop op$\Rightarrow$
is_set_linear_cop s (cop_exp (s, inprod) op)

This concludes our formalization of operators exponentiation.

4 Coherent Light Formalization

In this section, the formal definition of the coherent state of light is presented, which we then re-express in terms of the displacement operator (according to the presentation of Section 2.2). This is carried out in three steps: 1) quantum light formalization, 2) formalization of fock states (which are the basis of quantum optics states space), and 3) coherent states formalization.

4.1 Single Mode

The basic building block of formalizing light in quantum theory is the formal development of electromagnetic fields [2]: Quantum physics studies a light stream as an electromagnetic field. Such a field can be reduced to the superposition of several single-mode (i.e., single resonance frequency) fields. The formal definition of a single-mode filed is as follows:

Definition 13

$$\begin{aligned}&\texttt{is\_sm}\ ((\texttt{sp},\texttt{cs},\texttt{H}),\omega,\texttt{vac}) \Leftrightarrow\\ &\quad \texttt{is\_qsys}\ (\texttt{sp},\texttt{cs},\texttt{H}) \wedge 0<\omega \wedge \exists \texttt{q}\ \texttt{p}.\ \texttt{cs}=[\texttt{q};\texttt{p}]\\ &\quad \wedge\ \forall \texttt{t}.\texttt{is\_observable sp}\ (\texttt{p t}) \wedge \texttt{is\_observable sp}(\texttt{q t})\\ &\quad \wedge\ \texttt{H t} = \tfrac{\omega^2}{2}\%((\texttt{q t})\ \texttt{pow}\ 2) + \tfrac{1}{2}\%((\texttt{p t})\ \texttt{pow}\ 2)\\ &\quad \wedge \texttt{is\_qst sp vac} \wedge \texttt{is\_eigen\_pair}\ (\texttt{H t})\ (\texttt{vac}, \tfrac{\texttt{planck}*\omega}{2})\end{aligned}$$

A single-mode field is characterized by five elements: `sp` is the quantum states space of the field; `cs` lists the elementary observables of the mode, `p` and `q` are the *canonical coordinates* of the field, out of which we build the creator and annihilator operators; `H` is expressing the amount of energy inside the field; ω is the resonance frequency; and `vac` refers to the vacuum state. More details about `is_sm` can be found in [13].

As explained in Section 2.2, a single-mode field in a fock state (or photon number state) $|n\rangle$ is a light stream containing exactly n photons. These states are crucial because they form the basis of the single-mode quantum states space, and they are widely used in the development of quantum cryptography systems. According to Equation 5, we can formally define a fock state as follows:

Definition 14

$$\begin{aligned}&\texttt{let}\ (((\texttt{s},\texttt{inprod}),\texttt{cs},\texttt{H}),\omega,\texttt{vac}) = \texttt{sm in}\\ &\quad \texttt{fock sm 0} = \texttt{vac} \wedge \texttt{fock sm}\ (\texttt{SUC n}) =\\ &\qquad \texttt{get\_qst inprod}\ (\texttt{creat\_of\_sm sm}\ (\texttt{fock sm n})))\end{aligned}$$

where `get_qst` returns the normalized version of a vector, i.e., the vector divided by its norm. This is to ensure that the norm of the resulting quantum state is equal to one. Using this definition and the infinite summation, a coherent state can be defined as follows:

Definition 15

$$\begin{aligned}&\texttt{coherent sm}\ \alpha =\\ &\quad \texttt{let sm} = ((\texttt{s},\texttt{inprod}),\texttt{cs},\texttt{H}),\omega,\texttt{vac in}\\ &\qquad \exp(-\tfrac{|\alpha|^2)}{2})\%\ \texttt{cfun\_infsum}\ (\texttt{s},\texttt{inprod})\ (\texttt{from 0})\ (\lambda \texttt{n}.\tfrac{\alpha^{\texttt{n}}}{\sqrt{\texttt{n!}}}\%(\texttt{fock sm n}))\end{aligned}$$

where α is the state parameter (recall that the number of photons in a coherent stream is Poisson distributed with expectation $|\alpha|^2$). Note that Definition 15 corresponds to Equation (2).

As usual, we will often need to be able to tell when the sum in the above definition is convergent. We define therefore the predicate `coherent_summable`:

Definition 16

$$\begin{aligned}&\texttt{coherent\_summable sm}\ \alpha \Leftrightarrow\\ &\quad \texttt{let}\ (((\texttt{s},\texttt{inprod}),\texttt{cs},\texttt{H}),\omega,\texttt{vac}) = \texttt{sm in}\\ &\qquad \texttt{cfun\_summable}\ (\texttt{s},\texttt{inprod})\ (\texttt{from 0})\ (\lambda \texttt{n}.\tfrac{\alpha^{\texttt{n}}}{\sqrt{\texttt{!n}}}\%(\texttt{fock sm n}))\end{aligned}$$

We refer the reader to [14] for more details about the formalization of fock and coherent states.

The implementation of quantum coherent computer is based on the idea of expressing coherent beams in terms of the displacement operator, since it can

be easily realized using an optical beam splitter. Let us first give the formal definition of the displacement operator:

Definition 17

$$\begin{array}{l}\texttt{disp sm } \alpha = \\ \texttt{let qspc} = (\texttt{qspc\_of\_sm sm}) \texttt{ in} \\ \quad \underline{(\texttt{cop\_exp qspc } (\alpha \ \% \ \texttt{creat\_of\_sm sm})}_1 \ ** \\ \qquad \underline{\texttt{cop\_exp qspc } (-(\texttt{cnj v}) \ \% \ \texttt{a\_of\_sm sm})}_2 \ ** \\ \qquad\quad \underline{\texttt{cop\_exp qspc } ((\texttt{v} \ \% \ \texttt{creat\_of\_sm sm}) \texttt{ com } ((\texttt{cnj v}) \ \% \ \texttt{a\_of\_sm sm}))}_3\end{array}$$

where `op1 com op2 = op1 ** op2 - op2 ** op1` (called the *commutator* of `op1` and `op2`), and `creat_of_sm` and `a_of_sm` are functions that return the creator and annihilator operators, respectively.

To express a coherent state in terms of the displacement operator, we study the effect of this operator on the vacuum state: the underlined operator 3 in Definition 17 will collapse to a scalar value because `creat_of_sm sm com (a_of_sm sm) = I`; and since the two other operators are linear, we can get this scalar outside. The next step is to study the effect of the underlined operator 2 on the vacuum state. The following theorem shows that it actually acts like the identity:

Theorem 6

$$\begin{array}{l}\forall \texttt{s inprod cs H } \omega \texttt{ vac}. \\ \quad \texttt{let sm} = ((\texttt{s}, \texttt{inprod}), \texttt{cs}, \texttt{H}), \omega, \texttt{vac in} \\ \quad \texttt{is\_sm sm} \wedge \ \texttt{exp\_summable } (\texttt{qspc\_of\_sm sm}) \ (\alpha \ \% \ \texttt{a\_of\_sm sm}) \\ \qquad \Rightarrow \texttt{cop\_exp } (\texttt{qspc\_of\_sm sm}) \ (\alpha \ \% \ \texttt{a\_of\_sm sm}) \ \texttt{vac} = \texttt{vac}\end{array}$$

where `qspc_of_sm` returns the corresponding quantum states space of a given field. Thus the resulting state is again `vac`. It only remains to establish the effect of the underlined operator 1:

Theorem 7

$$\begin{array}{l}\forall \texttt{s inprod cs H } \omega \texttt{ vac } \alpha. \\ \quad \texttt{let sm} = ((\texttt{s}, \texttt{inprod}), \texttt{cs}, \texttt{H}), \omega, \texttt{vac in} \\ \quad \texttt{is\_sm sm} \wedge (\forall \texttt{m}. \ \texttt{creat\_of\_sm sm } (\texttt{fock sm m}) \neq \texttt{cfun\_zero}) \\ \quad \wedge \ \texttt{exp\_summable } (\texttt{qspc\_of\_sm sm}) \ (\alpha \ \texttt{creat\_of\_sm sm}) \\ \quad \wedge \ \texttt{cfun\_summable } (\texttt{s}, \texttt{inprod}) \ (\texttt{from 0})(\lambda \texttt{n}. \frac{\alpha \texttt{ pow n}}{\sqrt{\texttt{!n}}} \ \% \ \texttt{fock sm n}) \\ \qquad \Rightarrow \texttt{cop\_exp } (\texttt{qspc\_of\_sm sm}) \ (\alpha \ \% \ \texttt{creat\_of\_sm sm}) \ \texttt{vac} = \\ \qquad\quad \texttt{cfun\_infsum } (\texttt{s}, \texttt{inprod}) \ (\texttt{from 0})(\lambda \texttt{n}. \frac{\texttt{a pow n}}{\sqrt{\texttt{!n}}} \ \% \ \texttt{fock sm n})\end{array}$$

which corresponds almost to the definition of coherent light (see Definition 15): it differs only by multiplication with a scalar value. One then just needs to combine these results in order to obtain the final expression of coherent light in terms of the displacement operator:

Theorem 8

$$\begin{array}{l}\forall \texttt{s inprod s H } \omega \texttt{ vac } \alpha. \\ \quad \texttt{let sm} = ((\texttt{s}, \texttt{inprod}), \texttt{cs}, \texttt{H}), \omega, \texttt{vac in}\end{array}$$

$$\begin{aligned}
&\texttt{is_sm sm} \wedge \texttt{exp_summable (qspc_of_sm sm) (cnj}(-\alpha)\ \%\texttt{a_of_sm sm)}\\
&\wedge\ (\forall \texttt{n.creat_of_sm sm (fock sm n)} \neq \texttt{cfun_zero}))\\
&\wedge\ \texttt{cfun_summable (s, inprod) (from 0)}(\lambda \texttt{n.}\tfrac{\alpha\ \texttt{pow n}}{\sqrt{!\texttt{n}}}\ \%\ \texttt{fock sm n})\\
&\texttt{is_sm sm} \wedge\ \texttt{exp_summable (qspc_of_sm sm) (}\alpha\ \texttt{creat_of_sm sm)}\\
&\quad \Rightarrow \texttt{coherent sm}\ \alpha = (\texttt{disp sm}\ \alpha)\ \texttt{vac}
\end{aligned}$$

In the next section, we will see how this expression of coherent states helps in the development of the quantum flip gate.

5 Quantum Flip Gate Verification

In this section we detail the implementation of the optical flip gate [19], and explain the idea behind it. Recall that $|vac\rangle$ and $|\alpha\rangle$ are meant to implement the qbits $|0\rangle$ and $|1\rangle$, respectively. The specification of a flip gate is that it should turn $c_1\ |vac\rangle + c_2\ |\alpha\rangle$ into $c_1\ |\alpha\rangle + c_2\ |vac\rangle$, for all $c_1, c_2 \in \mathbb{C}$. The intended implementation of the gate is represented in Figure 1. First a beam splitter

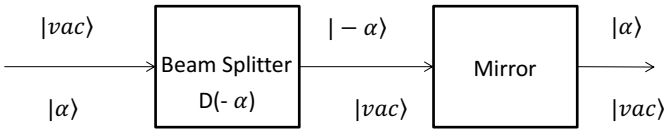

Fig. 1. Flip gate optical implementation

realizes a $-\alpha$ displacement operator. Then a phase conjugating mirror generates a beam identical to the input beam but with a reverse phase, which yields an output of $|-\alpha\rangle$ for an input of $|\alpha\rangle$.

We start by demonstrating the effect of the proposed optical flip gate on each optical qbit separately. Then, we generalize the result to any mixed qbit by using the linearity of the gate.

We start by formalizing the phase conjugating mirror as follows:

Definition 18

$$\begin{aligned}
&\texttt{mirror sm} =\\
&\quad \texttt{let sm} = ((\texttt{s}, \texttt{inprod}), \texttt{cs}, \texttt{H}), \omega, \texttt{vac in}\\
&\quad \texttt{cop_exp (s, inprod) (i}\pi\ \%\ \texttt{n_of_sm sm)}
\end{aligned}$$

We will see later that applying such quantum operator to a coherent beam result in the same beam in the reverse direction (i.e., the input beam is $|\alpha\rangle$ and the output is $|-\alpha\rangle$. This is exactly what a phase conjugating mirror does. Note that we use again quantum operator exponentiation.

The following property is the key to verify that the mirror implements phase shifting:

Theorem 9

$\forall$ s inprod cs H ω vac θ n.
  let sm = ((s, inprod), cs, H), ω, vac in
  is_sm sm $\wedge$ exp_summable (qspc_of_sm sm) (iθ % n_of_sm sm)
  $\wedge$ creat_of_sm sm (fock sm n) $\neq$ cfun_zero
    $\Rightarrow$ $\underline{\text{cop\_exp (qspc\_of\_sm sm) (i}\theta\text{ \% n\_of\_sm sm)}_1}$ (fock sm n) =
    $\underline{\text{(cexp (i}\theta\text{) pow m) \% (fock sm n)}_2}$

The underlined expression 1 is called a *phase shifter operator*. It is a generalization of the behavior of the phase conjugating mirror, except it considers any angle θ instead of just π. Theorem 9 shows the effect of such an operator on fock states: the underlined expression 2 shows that it generates the same state but shifted by θ. By specifying $\theta = \pi$, we can then easily prove the effect of the mirror on coherent states:

Theorem 10

$\forall$ s inprod cs H ω vac α.
  let sm = (((s, inprod), cs, H), ω, vac) in
  is_sm sm $\wedge$ cfun_summable (s, inprod) (from0) (λn.$\frac{\alpha^n}{\sqrt{!n}}$ % fock sm n)
  $\wedge$ mirror_summable sm $\wedge$ is_bounded (qspc_of_sm sm) (mirror sm)
  $\wedge$ ($\forall$n.creat_of_sm sm (fock sm n) $\neq$ cfun_zero))
    $\Rightarrow$ mirror sm (coherent sm α) = coherent sm ($-\alpha$)

where mirror_summable is similar to the summable notions defined before: we define a new predicate only for simplicity. The purpose of this predicate is to ensure that the mirror operator exists.

The former theorem proves that the mirror indeed behaves as expected when applied to the qbit $|1\rangle$. We now show that it is also the case for $|0\rangle$, i.e., for the vacuum state vac:

Theorem 11

$\forall$ s inprod cs H ω vac.
  let sm = ((s, inprod), cs, H), ω, vac in
  is_sm sm $\wedge$ coherent_summable sm 0
    $\Rightarrow$ coherent sm 0 = vac

Combined with the previous theorem, this confirms that the vac state is unchanged by the mirror.

We now complete the formalization by the beam splitter, which is modeled by the displacement operator. In case that the input to the beam splitter is vac then the output will be coherent sm ($-\alpha$) according to Theorem 8. For coherent sm α input, it results in vac according to the following theorem:

Theorem 12

$\forall$ s inprod cs H ω vac α.
  let sm = ((s, inprod), cs, H), ω, vac in
  is_sm sm $\wedge$ ($\forall$b.exp_summable (s, inprod) (b%a_of_sm sm))
  $\wedge$ ($\forall$m. creat_of_sm sm (fock sm m) $\neq$ cfun_zero) $\wedge$ coherent_summable sm α

$\wedge$ exp_summable (qspc_of_sm sm) (α creat_of_sm sm)
$\wedge$ is_bounded (s, inprod) (a_of_sm sm) $\wedge$ (coherent sm $\alpha \neq$ cfun_zero)
$\wedge$ ($\forall$x op. is_linear_cop op $\wedge$ x IN s $\Rightarrow$
(cop_exp (s, inprod) ($-$op) ** cop_exp (s, inprod) (op)) x = x)
$\Rightarrow$ disp sm ($-\alpha$) (coherent sm α) = vac

The last conjunction in the premises shows an assumed property about exponentiation of quantum operators. Such property requires the proof of a the general theorem of Baker-Campbell-Hausdorff [7] [1]. A major step towards proving Theorem 12 is to evaluate the effect of `cop_exp (a_of_sm sm)` on coherent beams. The following theorem shows such effect:

Theorem 13
$\forall$s inprod cs H ω vac α.
let sm = ((s, inprod), cs, H), ω, vac in
is_sm sm $\wedge$ exp_summable (s, inprod) (cnj α%a_of_sm sm)
$\wedge$ ($\forall$m. creat_of_sm sm (fock sm m) $\neq$ cfun_zero) $\wedge$ coherent_summable sm α
$\wedge$ is_bounded (s, inprod) (a_of_sm sm) $\wedge$ (coherent sm $\alpha \neq$ cfun_zero)
$\Rightarrow$ cop_exp (qspc_of_sm sm) ((cnj α)%a_of_sm sm) (coherent sm α) =
cexp((norm α)^2)%(coherent sm α)

Now, we have all ingredients to construct the flip gate and verify its behavior. The formal definition of the flip gate is made through the cascading of the mirror and beam splitter elements. This is defined as an operators' multiplication (i.e., function composition):

Definition 19
flip_gate α sm = (mirror sm) ** (disp sm ($-\alpha$))

Based on above definition and using Theorems 10-12, we prove the correction of the gate behavior in one single theorem as follows:

Theorem 14
$\forall$s inprod cs H ω vac α.
let sm = ((s, inprod), cs, H), ω, vac in
is_sm sm $\wedge$ exp_summable ($\forall$b. (s, inprod) (b%a_of_sm sm)
$\wedge$ ($\forall$m. creat_of_sm sm (fock sm m) $\neq$ cfun_zero)
$\wedge$ ($\forall$b. coherent_summable sm b)
$\wedge$ ($\forall$c. cfun_summable (s, inprod) (from 0) (λn.($\frac{c^n}{\sqrt{!n}}$)%fock sm n))
$\wedge$ ($\forall$d. exp_summable (s, inprod) (%creat_of_sm sm (0)))
$\wedge$ is_bounded (s, inprod) (a_of_sm sm)
$\wedge$ (coherent sm $\alpha \neq$ cfun_zero)$\wedge$
$\wedge$ (cop_exp (s, inprod) ($-$op) ** cop_exp (s, inprod) (op)) x = x)
$\wedge$ mirror_summable sm $\wedge$ is_bounded (qspc_of_sm sm) (mirror sm)
$\Rightarrow$ (flip_gate α sm) (coherent sm α) = vac
$\wedge$ (flip_gate α sm) vac = coherent sm α

[1] The proof of the Baker-Campbell-Hausdorff theorem is very complex and requires a lot of prerequisites that are not available in HOL Light. The formal verification of this theorem in HOL Light is part of our future work.

In a nutshell, Theorem 14 proves that a coherent beam $|\alpha\rangle$ ($|vac\rangle$) passes through a beam splitter, which in turn generates $|vac\rangle$ ($|-\alpha\rangle$), then the beam experiences a mirror which reflects it in the opposite direction to generate $|vac\rangle$ ($|\alpha\rangle$). Hence, we have the realization of the quantum flip gate. Note that given the linearity of the optical elements, this result generalizes for any mixed state $c_1 * |\alpha\rangle + c_2 * |vac\rangle$.

6 Conclusion

Quantum optics explores extremely useful phenomena and properties of light as a stream of photons. However, the analysis of quantum optical systems is complex. Traditional analysis techniques – simulation in optical laboratories, paper-and-pencil, numerical methods, and computer algebra systems – suffer from many problems: safety, cost, lack of expressiveness, human error. We believe that the proposed formalization of quantum optics can contribute to propose an alternative tackling these limitations.

Coherent light (or states) is an essential notion in quantum optics since it eases the analysis of many quantum systems. One of the most interesting applications of coherent light is quantum computers. Coherent states are proposed to model quantum bits [19], by taking $|vac\rangle$ and $|\alpha\rangle$ as $|0\rangle$ and $|1\rangle$, respectively. Many quantum gates were implemented based on this model. In this paper, we considered the quantum flip gate, which converts $\delta|0\rangle + \beta|1\rangle$ into $\beta|0\rangle + \delta|1\rangle$. We verified the behavior of this gate, which requires many formalization tasks. We started by developing the required mathematical foundations, in particular summation over quantum operators and exponentiation of quantum operators. Then we presented the formal definition of coherent beam, and expressed coherent states in terms of the displacement operator, which can be physically implemented as a beam splitter. The gate itself consists of a phase conjugating mirror along with a beam splitter. Therefore, we formalized the mirror and a displacement operator (or, equivalently, a beam splitter) and proved the required theorems to verify the gate behavior.

In the future, we plan to formalize other gates and handle other quantum computer implementations, for example the one based on squeezed states (a special case of coherent light). We also plan to extend our work to multi-mode systems, which are very useful for complicated quantum gates, in particular those that use the phenomena of entailment and teleportation [5].

References

1. Clarke, J., Wilhelm, F.K.: Superconducting quantum bits. Nature 453, 1031–1042 (2008)
2. Dirac, P.A.M.: The fundamental equations of quantum mechanics. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 109(752), 642–653 (1925)
3. Feagin, J.M.: Quantum Methods with Mathematica. Springer (2002)
4. Feynman, R.: Simulating physics with computers. International Journal of Theoretical Physics 21, 467–488 (1982)

5. Furusawa, A., van Loock, P.: Quantum Teleportation and Entanglement: A Hybrid Approach to Optical Quantum Information Processing. Wiley (2011)
6. Haeffner, H., Roos, C.F., Blatt, R.: Quantum computing with trapped ions. Physics Reports 469(4), 155–203 (2008)
7. Hall, B.: Lie Groups, Lie Algebras, and Representations: An Elementary Introduction. Graduate Texts in Mathematics. Springer (2003)
8. Jennewein, T., Barbieri, M., White, A.G.: Single-photon device requirements for operating linear optics quantum computing outside the post-selection basis. Journal of Modern Optics 58(3-4), 276–287 (2011)
9. Ladd, T.D., Jelezko, F., Laflamme, R., Nakamura, Y., Monroe, C., O'Brien, J.L.: Quantum computers. Nature 464, 45–53 (2010)
10. Li, Y., Browne, D.E., Kwek, L.C., Raussendorf, R., Wei, T.: Thermal states as universal resources for quantum computation with always-on interactions. Physical Review Letter 107, 060501 (2011)
11. Lomonaco, S.J.: Quantum Computation: A Grand Mathematical Challenge for the Twenty-first Century and the Millennium. American Mathematical Society (2002)
12. Loss, D., DiVincenzo, D.P.: Quantum computation with quantum dots. Physical Review A 57, 120–126 (1998)
13. Mahmoud, M.Y., Aravantinos, V., Tahar, S.: Formalization of infinite dimension linear spaces with application to quantum theory. In: Brat, G., Rungta, N., Venet, A. (eds.) NFM 2013. LNCS, vol. 7871, pp. 413–427. Springer, Heidelberg (2013)
14. Mahmoud, M.Y., Tahar, S.: On the quantum formalization of coherent light in HOL. In: Badger, J.M., Rozier, K.Y. (eds.) NFM 2014. LNCS, vol. 8430, pp. 128–142. Springer, Heidelberg (2014)
15. Mahmoud, M.Y., Aravantinos, V.: Formal verification of optical quantum flip gate - HOL Light script, `http://hvg.ece.concordia.ca/projects/qoptics/flipgate.html`
16. Mandel, L., Wolf, E.: Optical Coherence and Quantum Optics. Cambridge University Press (1995)
17. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge University Press (2010)
18. Paris, M.G.A.: Displacement operator by beam splitter. Physical Letters A 217(2-3), 78–80 (1996)
19. Ralph, T.C., Gilchrist, A., Milburn, G.J., Munro, W.J., Glancy, S.: Quantum computation with optical coherent states. Physical Review A 68, 042319 (2003)
20. Tan, S.M.: A computational toolbox for quantum and atomic optics. Journal of Optics B: Quantum and Semiclassical Optics 1(4), 424 (1999)