Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 270 (1) (2011) 41–57

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Game Semantics for Quantum Data

Yannick Delbecque[a,1] [,2]

[a] *School of computer science*
McGill University
Montreal, Canada

Abstract

This paper presents a game semantics for a simply-typed λ-calculus with qbits constants and associated quantum operations. The resulting language is expressive enough to encode any quantum circuit. The language uses a notion of extended variable, similar to that seen in functional languages with pattern matching, but adapted to the needs of dealing with tensor products. The game semantics is constructed from classical game semantics using quantum interventions as questions and measurements results as answers. A soundness result for the semantics is given.

Keywords: Game semantics, quantum programing languages, quantum games.

1 Introduction

An important problem in the development of higher-order quantum programming languages is to find an appropriate structure to define a denotational semantics. Previous works on quantum λ-calculi were based on the idea that quantum data should be used linearly since it cannot be duplicated. This idea was implemented by adding a quantum tensor type operation and typing rules to forbid duplication. This approach may seem natural, but the often counterintuitive behavior of classical-quantum interactions makes difficult the construction of the appropriate syntax and typing rules. Since no denotational semantics could be found for a complete quantum λ-calculus, we still lack a soundness result for a complete language that would validates the choices made. For example, there was no denotational semantics given in the first presentations of the quantum λ-calculus developed by Selinger and Valiron [14,16]. They proposed in [15] a denotational semantics for the linear part of the quantum λ-calculus; their interpretation is in the category of completely positive maps on finite dimensional Hilbert spaces. Working with this restricted

[1] This research was supported by a grant from NSERC (Canada)

[2] Email: yannick@delbecque.org

doi:10.1016/j.entcs.2011.01.005

language allows them to avoid the problem of finding a structure which can model completely the possible interactions between quantum data and classical data in higher order quantum programming languages.

In this paper we introduce a new λ-calculus equipped with extra structure to allow it to represent manipulation of quantum data. The proposed language was build with the goal of proving soundness using a denotational semantics where quantum states and operations are represented represented as *strategies* which makes someone choose the actions according to the laws of quantum mechanics. Our proposed model is built upon ideas from game semantics augmented with quantum strategies which describe the behavior of quantum states and quantum operations.

The language we define in this paper introduces new features and ideas. Perhaps one of the most important one is the fact that we forbid abstraction over part of a tensor of unknown qbits of the form $x \otimes y$. This is motivated by the fact that abstraction should intuitively be interpreted using a correspondence between programs of type $\mathrm{qbit} \otimes \mathrm{qbit} \multimap \mathrm{qbit} \otimes \mathrm{qbit}$ with those of type $\mathrm{qbit} \multimap (\mathrm{qbit} \multimap \mathrm{qbit} \otimes \mathrm{qbit})$. This seems problematic, since this should be a correspondence between functions with two input qbits, which may be in some entangled state, with functions using only separated qbits. A consequence of this is that there is no tensor type operation in the proposed language, only types $\mathrm{qbit}^{\otimes n}$ for n qbits. We use *extended variables*, which are tensor of variables, to keep track of possible entanglements between qbit variables. Finally, the model forces us to distinguish between tensor of known and unknown qbits, leading to three different typing rules for the tensor operations.

2 Simply typed λ-calculus with quantum data

2.1 Syntax

We now introduce a λ-calculus with quantum data language (QDL). The syntax of QDL is that of a classical simply typed λ-calculus with pairing and conditionals, with extra constructs that give the language enough expressiveness to encode usual manipulations of quantum data as can be described with the low level formalism of quantum circuits.

We first need to introduce a syntax which allows one to refer to specific qbits in a tensor product. An *extended variable* is an expression of the form $x_1 \otimes \cdots \otimes x_n$, where the x_i are variables such that $x_i \neq x_j$ if $i \neq j$. Two extended variables $x_1 \otimes \cdots \otimes x_n$ and $y_1 \otimes \cdots \otimes y_m$ are *disjoint* if $x_i \neq y_j$ for all i, j. Two such extended variables can be joined to form a new extended variable $x_1 \otimes \cdots \otimes x_n \otimes y_1 \otimes \cdots \otimes y_m$. Note that when we use $x_1 \otimes \cdots \otimes x_n$ to refer to an arbitrary extended variable, the case $n = 1$ is also possible. To simplify the notation, we use $\overline{x}$ instead of $x_1 \otimes \cdots \otimes x_n$, leaving the number n implicit.

Table 1
QDL typing rules.

$$\frac{}{\Gamma, \Delta, \overline{x}\colon A \vdash \overline{x}\colon A} \qquad \frac{}{\Gamma, \Delta \vdash *\colon \top} \qquad \frac{}{\Gamma, \Delta \vdash 0\colon \mathrm{bool}} \qquad \frac{}{\Gamma, \Delta \vdash 1\colon \mathrm{bool}}$$

$$\frac{\Gamma, \Delta, \overline{x}\colon A \vdash M\colon B}{\Gamma, \Delta \vdash \lambda\overline{x}.\, M\colon A \Rightarrow B} \qquad \frac{\Gamma, \Delta_1 \vdash M\colon A \Rightarrow B \qquad \Gamma, \Delta_2 \vdash N\colon A}{\Gamma, \Delta_1, \Delta_2 \vdash MN\colon B}$$

$$\frac{\Gamma, \Delta_1 \vdash M_1\colon A_1 \qquad \Gamma, \Delta_2 \vdash M_2\colon A_2}{\Gamma, \Delta_1, \Delta_2 \vdash \langle M_1, M_2\rangle\colon A_1 \times A_2} \qquad \frac{\Gamma, \Delta \vdash M\colon A \times B}{\Gamma, \Delta \vdash \mathsf{fst}\, M\colon A} \qquad \frac{\Gamma, \Delta \vdash M\colon A \times B}{\Gamma, \Delta \vdash \mathsf{snd}\, M\colon A}$$

$$\frac{\Gamma, \Delta_1 \vdash P\colon \mathrm{bool} \qquad \Gamma, \Delta_2 \vdash M\colon A \qquad \Gamma, \Delta_2 \vdash N\colon A}{\Gamma, \Delta_1, \Delta_2 \vdash \mathsf{if}\, P\, \mathsf{then}\, M\, \mathsf{else}\, N\colon A} \qquad \frac{}{\Gamma, \Delta \vdash \rho\colon \mathrm{qbit}^{\otimes n}}$$

$$\frac{\Gamma, \Delta_1 \vdash Q\colon \mathrm{qbit}^{\otimes(n+1)} \qquad \Gamma, \Delta_2, b\colon \mathrm{bool}, \overline{x}\colon \mathrm{qbit}^{n} \vdash M\colon A}{\Gamma, \Delta_1, \Delta_2 \vdash \mathsf{let}\, b, \overline{x} = \mathsf{meas}_i\, Q\, \mathsf{in}\, M\colon A} \qquad \frac{\Gamma, \Delta \vdash M\colon \mathrm{qbit}^{\otimes n}}{\Gamma, \Delta \vdash \mathcal{U}\, M\colon \mathrm{qbit}^{\otimes n}}$$

$$\frac{\Gamma, \Delta \vdash Q\colon \mathrm{qbit}^{\otimes n}}{\Gamma, \Delta \vdash \mathsf{meas}\, Q\colon \mathrm{bool}} \qquad \frac{\Gamma, \Delta_1 \vdash M_1\colon \mathrm{qbit}^{\otimes n} \qquad \Gamma, \Delta_2 \vdash M_2\colon \mathrm{qbit}^{\otimes m}}{\Gamma, \Delta_1, \Delta_2 \vdash M_1 \otimes M_2\colon \mathrm{qbit}^{\otimes n} \otimes \mathrm{qbit}^{\otimes m}}\ \mathrm{FV}(M_i) \cap |\Delta_i| = \emptyset$$

$$\frac{\Gamma, \Delta_1, \overline{x_1}\colon \mathrm{qbit}^{\otimes n} \vdash M_1\colon \mathrm{qbit}^{\otimes n} \qquad \Gamma_2, \Delta_2, \overline{x_2}\colon \mathrm{qbit}^{\otimes m} \vdash M_2\colon \mathrm{qbit}^{\otimes m}}{\Gamma, \Delta_1, \Delta_2, \overline{x_1} \otimes \overline{x_2}\colon \mathrm{qbit}^{\otimes n} \otimes \mathrm{qbit}^{\otimes m} \vdash M_1 \otimes M_2\colon \mathrm{qbit}^{\otimes n} \otimes \mathrm{qbit}^{\otimes m}}\ \mathrm{FV}(M_i) \setminus |\Delta_i| = \{\overline{x}_i\}$$

$$\frac{\Gamma, \Delta_1, \overline{x}\colon \mathrm{qbit}^{\otimes n} \vdash M_1\colon \mathrm{qbit}^{\otimes n} \qquad \Gamma, \Delta_2 \vdash M_2\colon \mathrm{qbit}^{\otimes m}}{\Gamma, \Delta_1, \Delta_2, \overline{x}\colon \mathrm{qbit}^{\otimes n} \vdash M_1 \otimes M_2\colon \mathrm{qbit}^{\otimes n} \otimes \mathrm{qbit}^{\otimes m}}\ \begin{array}{l} \mathrm{FV}(M_1) \setminus |\Delta_1| = \{\overline{x}_1\} \\ \mathrm{FV}(M_2) \cap |\Delta_2| = \emptyset \end{array}$$

The terms of QDL are defined recursively as follows:

$$\begin{aligned} M, N, P \ :=\ & \overline{x} \mid * \mid 0 \mid 1 \mid \rho \mid \langle M, N\rangle \mid \mathsf{fst}\, M \mid \mathsf{snd}\, M \mid \\ & MN \mid \lambda\overline{x}.\, M \mid \mathsf{if}\, M\, \mathsf{then}\, N\, \mathsf{else}\, P \mid \\ & \mathsf{let}\, b, \overline{x} = \mathsf{meas}_i\, M\, \mathsf{in}\, N \mid \mathsf{meas}\, Q \mid \mathcal{U}\, M, \end{aligned}$$

where $b, \overline{x}, \overline{y}$ are extended variables as defined above, $i > 0$ is a natural number, ρ can be any density matrix and $\mathcal{U}$ is a superoperator corresponding to a unitary transformation U. Most of the syntax consist of standard λ-calculus operations. The term $\mathcal{U}\, M$ is the operation that correspond to applying a unitary transformation to the state described by the term M. The measurement operation syntax $\mathsf{let}\, b, \overline{x} = \mathsf{meas}_i\, M\, \mathsf{in}\, N$ means that the qbit i of the term M is measured and thereafter the measurement result is accessible in N as b and the resulting state is accessible as $\overline{x}$. Note that the variable b and $\overline{x}$ are bound in N. To measure a single qbit, we use instead the simpler syntax $\mathsf{meas}\, Q$. The set of free variables in M is denoted $\mathrm{FV}(M)$.

The types of QDL are the following:

$$A, B \ :=\ \mathrm{bool} \mid \top \mid \mathrm{qbit}^{\otimes n} \mid A \times B \mid A \Rightarrow B.$$

where $n > 0$. The type bool is the type of boolean constants, $A \times B$ and $A \Rightarrow B$ are respectively the types of pairs and functions. The type $\mathrm{qbit}^{\otimes n}$ is the type of quantum states on n qbits. The notation $\mathrm{qbit}^{\otimes n}$ stands implicitly for the product $\mathrm{qbit} \otimes \cdots \otimes \mathrm{qbit}$; we use the notation $\mathrm{qbit}^{\otimes n} \otimes \mathrm{qbit}^{\otimes m}$ to denote $\mathrm{qbit}^{\otimes(n+m)}$, although there is no $\otimes$ type operation.

The typing rules of QDL are given in table 1. We assume that contexts Γ contain

Table 2
QDL probabilistic reduction.

$$\frac{}{V \Downarrow V} \qquad \frac{M \Downarrow^{p} \lambda\overline{x}.\, M' \qquad N \Downarrow^{q} V}{MN \Downarrow^{pq} M[V/\overline{x}]} \qquad \frac{M_1 \Downarrow^{p} V_1 \qquad M_2 \Downarrow^{q} V_2}{\langle M_1, M_2\rangle \Downarrow^{pq} \langle V_1, V_2\rangle}$$

$$\frac{M \Downarrow^{p} \langle V_1, V_2\rangle}{\mathsf{fst}\, M \Downarrow^{p} V_1} \qquad \frac{M \Downarrow^{p} \langle V_1, V_2\rangle}{\mathsf{snd}\, M \Downarrow^{p} V_2}$$

$$\frac{P \Downarrow^{p} 0 \qquad M \Downarrow^{q} V}{\mathsf{if}\, P\, \mathsf{then}\, M\, \mathsf{else}\, N \Downarrow^{pq} V} \qquad \frac{P \Downarrow^{p} 1 \qquad N \Downarrow^{q} V}{\mathsf{if}\, P\, \mathsf{then}\, M\, \mathsf{else}\, N \Downarrow^{pq} V}$$

$$\frac{Q \Downarrow^{q} \rho \qquad M\left[b/m, \overline{x}/\frac{1}{p_m}[m]\rho[m]\right] \Downarrow^{r} V}{\mathsf{let}\, b, \overline{x} = \mathsf{meas}_i\, Q\, \mathsf{in}\, M \Downarrow^{p_m q r} V}\; p_m = \mathrm{tr}\left([m]^i\rho\right),\; m = 0, 1$$

$$\frac{Q \Downarrow^{q} \rho}{\mathsf{meas}\, Q \Downarrow^{p_m} m}\; p_m = \mathrm{tr}\left([m]^i\rho\right),\; m = 0, 1$$

$$\frac{M_1 \Downarrow^{p} V_1 \qquad M_2 \Downarrow^{q} V_2}{M_1 \otimes M_2 \Downarrow^{pq} V_1 \otimes V_2} \qquad \frac{M \Downarrow^{p} \rho}{\mathcal{U}\, M \Downarrow^{p} \mathcal{U}(\rho)}$$

no qbit variables and contexts Δ_k contain only qbits variables. This convention will be used throughout this paper. Rules involving classical operations correspond are direct adaptation of the standard typing rules of a typed λ-calculus. The rules for quantum constants, quantum measurements and unitary operations are straightforward. The three tensor rules allow one to take two terms of type $\mathrm{qbit}^{\otimes n}$ and $\mathrm{qbit}^{\otimes m}$ and create a term of type $\mathrm{qbit}^{\otimes(n+m)}$. The distinction between the three cases is due to the fact that known or unknown qbits must be dealt with differently. If $\Gamma, \Delta \vdash M \colon \mathrm{qbit}^{\otimes n}$, M is a known qbit when it has no dependency on some quantum state variable in Δ, i.e. if $\mathrm{FV}(M) \cap |\Delta| = \emptyset$. If instead $\mathrm{FV}(M) \cap |\Delta|$ contains only an extended variable $\overline{x}$, then the quantum state represented by M depends on the value of the quantum variable $\overline{x}$ and is thus unknown. The typing rules do not allow an unknown quantum state to depend upon more than one other quantum state. Note also that the term $x \otimes x$ is not well-typed because it is no a valid extended variable. We also have to ban duplicating terms like $\lambda x.\, x \otimes x$, so we require that qbit variables are used linearly.

Example 2.1 Quantum teleportation can be implemented in the quantum data λ-calculus. Consider the following QDL teleportation term:

```
teleport:
λx. let $b_x, y \otimes z$ = meas$_1$cnot$^{12}$ $((\mathcal{H}\, x) \otimes [\beta_{00}])$ in
    let $b_y, z'$ = meas$_1$ $y \otimes z$ in
       if $b_x$ then
          if $b_y$ then $\mathcal{U}_{00}\, z'$ else $\mathcal{U}_{01}\, z'$
       else
          if $b_y$ then $\mathcal{U}_{10}\, z'$ else $\mathcal{U}_{11}\, z'$
```

where the unitary superoperators $\mathcal{U}_{b_x b_y}$ are the usual correction unitary operations of the teleportation protocol and $[\beta_{00}]$ is the Bell state $(|00\rangle + |11\rangle)/\sqrt{2}$.

Using the type inference rules, we can derive that $\vdash \mathsf{teleport} \colon \mathrm{qbit} \Rightarrow \mathrm{qbit}$.

Any quantum circuit can be implemented as a QDL term in a similar manner.

The input qbits are represented as a qbit variable $\overline{x}$ which is tensored with ancilla qbits if necessary. The unitary transformation can then be applied to the resulting term. Finally, measurements operations are used to extract the result.

The operational semantics of the λ-calculus with quantum data is given as a big-step probabilistic reduction relation $M \Downarrow^p V$ between terms and values. *Values* are the terms defined recursively by

$$V, W \;:=\; 0 \mid 1 \mid * \mid \rho \mid \lambda\overline{x}.\, M \mid \langle V, W\rangle \mid V \otimes W.$$

The reduction relation is defined by the rules given in table 2. The operational semantics of classical operations is defined using standard reduction rules. The quantum operations reduction rules makes reduction of quantum terms follow the rules of quantum mechanics.

Example 2.2 The term $\mathsf{teleport}\ \rho$ reduces with probability 1 to ρ.

3 Denotational semantics

3.1 Probabilistic game semantics

The game semantics presented in this paper is constructed using the definitions of probabilistic games semantics introduced by Danos and Harmer [3]. We give here an overview of the basic definitions and facts of probabilistic game semantics.

Definition 3.1 An *arena* A is a triple $(M_A, \lambda_A, \vdash_A)$ where M_A is a set of *moves*, the function $\lambda_A \colon M_A \to \{\mathrm{O}, \mathrm{P}\} \times \{\mathrm{Q}, \mathrm{A}\} \times \{\mathrm{I}, \mathrm{N}\}$ is a labeling which assigns moves to the two players *Opponent* and *Player*, and tells us which moves are *Questions* and which are *Answers*, and whether they are *Initial* or *Noninitial* moves, and finally $\vdash_A \subseteq M_A \times M_A$ is a relation, called the *enabling relation*, such that

(A1) if $a \vdash_A b$, then $\lambda_A^{\mathrm{OP}}(a) \neq \lambda_A^{\mathrm{OP}}(b), \lambda_A^{\mathrm{QA}}(a) \neq \lambda_A^{\mathrm{QA}}(b)$,

(A2) if $\lambda_A^{\mathrm{IN}}(a) = \mathrm{I}$, then $\lambda_A(a) = \mathrm{OQI}$,

(A3) if $a \vdash b$ and $\lambda_A^{\mathrm{QA}}(b) = \mathrm{A}$ then $\lambda_A^{\mathrm{QA}}(a) = \mathrm{Q}$,

where the functions λ_A^{OP}, λ_A^{QA} and λ_A^{in} are λ_A composed with the projections on the sets $\{\mathrm{O}, \mathrm{P}\}$, $\{\mathrm{Q}, \mathrm{A}\}$ and $\{\mathrm{I}, \mathrm{N}\}$.

We use the convention that M_A^X, where X is some list of superscripts taken from the set of move labels $\{\mathrm{O}, \mathrm{P}, \mathrm{Q}, \mathrm{A}, \mathrm{I}, \mathrm{N}\}$ denote the set of moves labeled with these labels. Moves in an arena are thus of various types, and the constraints on the enabling relation $\vdash_A$ limits the possible interactions in the arena by limiting which moves can be made at a certain point given the past interactions. The condition (A1) forces that only Player moves to enable Opponent moves and *vice versa*, (A2) asks for all initial moves to be questions by Opponent and finally (A3) says that answers can only be enabled by questions.

A *play* in A is a sequence of moves $s \in M_A^*$. This does not take into account the enabling relation; we define a *justified play* to be a play where each occurrence of a non-initial move b has a pointer to a previous occurrence of a move a with $a \vdash_A b$. We finally need to enforce alternation of the two players. A *legal play* is a justified play where Opponent and Player alternate with; we denote the set of legal plays in A by $\mathcal{L}_A$. Note that because all initial moves are Opponent moves, Opponent is always making the first move. The sets of odd and even length legal plays are respectively denoted by $\mathcal{L}_A^{\mathrm{odd}}$ and $\mathcal{L}_A^{\mathrm{even}}$.

Example 3.2 The **bool** arena is defined with $M_{\mathbf{bool}} = \{?, 0, 1\}$ $\lambda_{\mathbf{bool}}(?) = (\mathrm{O}, \mathrm{Q}, \mathrm{I})$ and $\lambda_{\mathbf{bool}}(0) = \lambda_{\mathbf{bool}}(1) = (\mathrm{P}, \mathrm{A}, \mathrm{N})$ and with the enabling relation $? \vdash_{\mathbf{bool}} 0, 1$.

Example 3.3 The *empty arena* I is the arena with no moves at all. The only legal play in I is the empty play ε.

Suppose $sa \in \mathcal{L}_A$. Starting from a and following the justification pointers will always lead to an occurrence of an initial move b, which we call the *hereditary justifier* of a in sa. We can see that every legal play will be partitioned in subplays, each one consisting of all occurrences of moves hereditarily justified by a given initial move. These subplays are called *threads*. The *current thread* of a legal play sa ending with an opponent move, denoted by $\lceil sa \rceil$, is the thread of sa where a occurs. If sa ends with a Player move, the current thread is then defined by $\lceil s \rceil a$. We want the current thread to be a legal play, so it is necessary to impose an extra condition on legal plays: a legal play s is *well-threaded* if for every subplay ta ending with a Player move, the justifier of a is in $\lceil t \rceil$. In a well-threaded play, player always plays in the last thread where Opponent played.

Given arenas A, B, the *product* $A \odot B$ and *arrow* $A \multimap B$ operations are defined respectively as follows:

- $M_{A \odot B} = M_A + M_B$ (disjoint union)
- $\lambda_{A \odot B} = [\lambda_A, \lambda_B]$ (copairing)
- $m \vdash_{A \odot B} n$ iff $m \vdash_A n$ or $m \vdash_B n$.

- $M_{A \multimap B} = M_A + M_B$
- $\lambda_{A \multimap B} = \left[\langle \overline{\lambda}_A^{\mathrm{OP}}, \lambda_A^{\mathrm{QA}}, \overline{\lambda}_A^{\mathrm{IN}} \rangle, \lambda_B\right]$
- $m \vdash_{A \multimap B} n$ iff $m \vdash_A n$ or $m \vdash_B n$ or $\lambda_B^{\mathrm{IN}}(n) = \lambda_A^{\mathrm{IN}}(m) = \mathrm{I}$.

where $\overline{\lambda}_A^{\mathrm{OP}}$ inverts the roles of the two players and $\overline{\lambda}_A^{\mathrm{IN}}$ makes all moves of A non-initial. The product arena $A \odot B$ is intuitively understood as the arena where at each of Opponent's turn she can choose to play a move in either A or B, and where Player must answer in the last component where Opponent played. In the arena $A \multimap B$, after Opponent makes an initial move in B, at each of his turns Player can choose to play either one of his moves in B or an Opponent move in A.

Given a legal play s in an arena A, let $\mathrm{next}_A(s) = \{a \in M_A | sa \in \mathcal{L}_A\}$ be the set of all moves that can be legally made after the play s.

Definition 3.4 A *probabilistic strategy* for Player is a function $\sigma \colon \mathcal{L}_A^{\mathrm{even}} \to [0, 1]$ such that

$$\sigma(\epsilon) = 1 \quad \text{and} \quad \sigma(s) \geq \sum_{b \in \mathrm{next}(sa)} \sigma(sab)$$

The set of *traces* of a strategy σ in A is the set of even length legal plays which are assigned a non-zero probability by σ: it is denoted $\mathcal{T}_\sigma$. A strategy σ is *deterministic* if $\sigma(s) = 1$ for all $s \in \mathcal{T}_\sigma$.

It is possible to describe a probabilistic strategy σ in conditional form. The probability $\sigma(b \mid sa) = \frac{\sigma(sab)}{\sigma(s)}$ is the probability of Player choosing to play b after the play sa.

Composition of strategies is the way interactions between parts of a program are encoded in game semantics. Given two strategies $\sigma \colon A \multimap B$ and $\tau \colon B \multimap C$, we define a new strategy $\sigma;\tau \colon A \multimap C$ obtained by letting σ and τ "interact" on B. Before giving the definition of composition, it is necessary to formalise this notion of interaction. The set of interactions for A, B, C is

$$\mathcal{I}_{A,B,C} = \{u \in (M_A + M_B + M_C)^* \mid u|_{AB} \in \mathcal{L}_{A \multimap B}, u|_{BC} \in \mathcal{L}_{B \multimap C}, u|_{AC} \in \mathcal{L}_{A \multimap C}\}$$

where $u|_{AB}$ is the sub sequence of u obtained by deleting the moves of C, and similarly for $u|_{BC}$. The case of $u|_{AC}$ is a bit different because deleting from u the moves of B and their associated pointers might leave the moves of A or C that are justified by B-moves without justifiers. In this case, we define the justifiers of $u|_{AC}$ to be as follows: a move a in C justified by a move b in B will be justified by the first move of either A or C we get to by following back the justification pointers from a in u. The set of *witnesses* $\mathrm{wit}(s)$ of $s \in \mathcal{L}_{A \multimap C}$ in an interaction $\mathcal{I}_{A,B,C}$ is the set of interactions $u \in \mathcal{I}_{A,B,C}$ such that $u|_{AC} = s$. The composition of two strategies $\sigma \colon A \multimap B$ and $\tau \colon B \multimap C$ can now be defined as follows:

$$[\sigma;\tau](s) = \sum_{u \in \mathrm{wit}(s)} \sigma(u|_{AB})\tau(u|_{BC}).$$

The *identity strategy* (or so-called "copycat strategy") $\mathrm{id}_A \colon A \multimap A$ is neutral with respect to composition. It is defined as the strategy which makes Player copy Opponent moves between corresponding components. Formally, this is defined as the deterministic strategy with trace

$$\mathcal{T}\left(1_A(s)\right) = \left\{s \in \mathcal{L}_{A_l \multimap A_r} \mid \forall s' \sqsubseteq^{\text{even}} s.\ s'|_{A_r} = s'|_{A_r}\right\}.$$

Using all the structure defined so far it is possible to define a category of arenas and probabilistic strategies. Taking arenas as objects, a morphism $A \to B$ is a strategy in $A \multimap B$. Composition of strategy is the needed composition, with the identity strategies as identity morphisms. It is associative, and it is shown in [3] that probabilistic strategies are closed under composition. This category is also symmetric monoidal. The operation $\odot$ is a tensor product, which acts on morphisms as follows. Given $\sigma \colon A \to C$ and $\tau \colon B \to D$ and $s \in \mathcal{L}^{\text{even}}_{(A \odot B) \multimap (A' \odot B')}$, we set $[\sigma \odot \tau](s) = \sigma(s|_{A \multimap C})\tau(s|_{C \multimap D})$. All coherence isomorphisms are easily defined using variants of the copycat strategy.

Threads have an important role in game semantics as a way to characterize the strategies that encodes programs with side-effects, like stores. This is achieved by forcing Player to use only the limited information available in the current thread instead of using all the information that can be extracted from the whole previous plays, including move made in other threads.

A strategy σ is *well-threaded* if $\mathcal{T}_\sigma$ consists only of well-threaded plays. Note that this condition forces Player to answer in the last thread where Opponent played. Given two well-threaded plays $sab \in \mathcal{L}_A^{even}$ and $ta \in \mathcal{L}_A^{odd}$ with $\lceil sa \rceil = \lceil ta \rceil$, we define $\mathrm{match}(sab, ta)$ to be the unique legal play tab with b justified as in $\lceil sa \rceil$. A well-threaded strategy σ is said to be *thread independent* if $sab \in \mathcal{T}_\sigma$, $t \in \mathcal{T}_\sigma$, $a \in \mathrm{next}(t)$ and $\lceil sa \rceil = \lceil ta \rceil$ implies that

$$\frac{\sigma(sab)}{\sigma(s)} = \frac{\sigma((\mathrm{match}(sab, ta))}{\sigma(t)}.$$

The meaning of this condition is that if Player plays according to σ, Player chooses his answers with probabilities that only depend on the current thread, i.e. $\sigma(b \mid sa) = \sigma(b \mid ta)$.

The diagonal strategy $\Delta_A \colon A \to A \odot A$ is defined as the deterministic strategy with trace set $\left\{ s \in \mathcal{L}^{even}_{A \multimap A_l \odot A_r} \mid \forall s' \sqsubseteq^{\text{even}} s. s'|_{A_l} \in \mathrm{id}_{A_l} \wedge s'|_{A_r} \in \mathrm{id}_{A_r} \right\}$. This is similar to the definition of the identity strategy: Δ instructs Player to use copying strategies between A and its two copies A_l and A_r. Possible conflicts in A are resolved by separating in different threads moves made according to the left or the right copy plays. There is also a unique strategy $\diamond_A \multimap I$, namely the trivial strategy with trace $\{\varepsilon\}$.

The *pairing* of two thread independent strategies $\sigma \colon A \multimap B$ and $\tau \colon A \multimap C$ is defined by $\langle \sigma, \tau \rangle = \Delta_A; \sigma \odot \tau$. Thus when Player plays using the pair strategy $\langle \sigma, \tau \rangle$, he plays using σ after an initial move in B, and using τ after an initial move in C.

For each arena A, $(A, \Delta_A, \diamond_A)$ is a *comonoid*. It is shown in in [8] that a strategy $\sigma : A \multimap B$ is thread independent if and only if σ is a comonoid homomorphism. Using a known fact in category theory[9], this implies that the restriction of the category of arena and probabilistic strategies to thread independent strategies is a Cartesian closed category. Note that projections strategies like $\pi_A \colon A \odot B \multimap A$ are defined as copying strategies which makes Player copies Opponent's moves between the two A component arenas.

3.2 Quantum arenas

To model the quantum part of QDL, we have to define an arena where quantum data can be represented as a strategy. This arena is defined in a similar way as the **bool** arena. A play begin with Opponent asking Player about the measurement result of a quantum measurement performed on the current quantum state. Player's answers are the possible measurements results and each answer can be chosen with a probability consistent with quantum mechanics. The type of quantum measurement which can be used by Opponent is the general description of quantum measurements called *intervention operators* introduced by Peres [11]. The measurement process is conceived of as a unitary interaction of a measurement apparatus with the quantum system to be measured, followed by a projective measurement on the combined system. Let $\mathrm{D}(H)$ be the set of density matrices on H and $\mathrm{SD}(H)$ be set of Hermitian positive operators of trace less than one. A *quantum intervention* on a Hilbert space H is a collection of superoperators $\mathcal{E} = \{\mathcal{E}_m \colon \mathrm{SD}(H) \to \mathrm{SD}(H_m)\}$ indexed by measurement results m, such that we have $\sum_m \mathrm{tr}\,(\mathcal{E}_m(\rho)) = 1$ for any state ρ. If the system is initially in state ρ, performing the quantum intervention yields result m with probability $p_m = \mathrm{tr}\,(\mathcal{E}_m(\rho))$ and leaves the system in state $\mathcal{E}_m(\rho)/p_m$. Note that the space H_{B_m} may depend on the measurement outcome.

Let H be an Hilbert space. The arena $[H]$ is the arena where questions are quantum interventions of the form

$$\mathcal{E}_? = \left\{\mathcal{E}^?_m \colon \mathrm{SD}(H) \to \mathrm{SD}(H_m)\right\}.$$

The possible answers to $\mathcal{E}_?$ are the possible measurements results m. A play in this arena is a sequence of moves $\mathcal{E}_{?[1]}m_1 \cdots \mathcal{E}_{?[n]}m_n$ where the quantum interventions $\mathcal{E}_{?[k]}$ may all be different.

A quantum state ρ is modeled by a probabilistic strategy $[\rho]$ in $[H]$. The strategy $[\rho]$ is defined by the weights $[\rho]\left(\mathcal{E}_{?[1]}m_1 \ldots \mathcal{E}_{?[n]}m_n\right) = \mathrm{tr}\left(\mathcal{E}^{?[1]}_{m_1} \ldots \mathcal{E}^{?[n]}_{m_n}(\rho)\right)$. Superoperators are composed as usual, but we use a convenient convention: if the domain of $\mathcal{E}$ does not match the codomain of $\mathcal{F}$ we put $\mathcal{E}\mathcal{F} = 0$. This convention is consistent with the quantum mechanical interpretation of superoperators: an impossible operation is assigned probability zero. Note that the strategy $[\rho]$ is *thread independent*: the answer to the last question always depend of the previous questions which in general have modified the initial state $[\rho]$.

Using these strategies, we can now represent any trace-preserving superoperator $\mathcal{F}$ taking states in H_A to states in H_B as a strategy. This strategy is denoted $[\mathcal{F}]$; it makes Player answers questions about the output state by measuring the input state in the way described by the following typical play:

$$\begin{array}{ccc}
[H_A] & \overset{[\mathcal{E}]}{\multimap} & [H_B] \\
 & & \mathcal{E}_? \\
\mathcal{E}_?\mathcal{F} & & \\
m & & \\
 & & m
\end{array}$$

The quantum intervention $\mathcal{E}_?\mathcal{F}$ is the quantum intervention $\left\{\mathcal{E}^?_m\mathcal{F}\right\}$ obtained by composing each intervention $\mathcal{E}_m$ with $\mathcal{F}$. All the quantum operations of QDL are interpreted using variants of this basic scheme. In particular, the unitary strategy is a special case of the above with $\mathcal{F}$ being the superoperator $\mathcal{U}$ associated to a unitary operation U. The way a new quantum intervention is created from the initial one $\mathcal{E}_?$ motivate the use of quantum intervention: implementing a similar scheme with other quantum measurement formalisms like projective measurements would not allow to represent quantum operations as general as trace-preserving superoperators.

There is a similarity between consistent histories approach to quantum mechanics [5,10,7] and the scheme used to define the strategy $[\rho]$. There is a clear connection in "spirit" in the sense that both are based on sequences of measurement results. In this perspective, the above idea used to represent a quantum operation $\mathcal{F}$ is new and could be a structured way to think about quantum operations in that context.

3.3 Definition of the denotational semantics

We now use the quantum arena defined in the last section to define a denotational semantics for QDL. First, the types are interpreted as follows:

$$\llbracket \text{bool} \rrbracket = \textbf{bool} \qquad \llbracket \top \rrbracket = \top \qquad \left[\!\left[\text{qbit}^{\otimes n}\right]\!\right] = \textbf{qbit}^{\otimes n}$$
$$\llbracket A \multimap B \rrbracket = \llbracket A \rrbracket \multimap \llbracket B \rrbracket \qquad \llbracket A \odot B \rrbracket = \llbracket A \rrbracket \odot \llbracket B \rrbracket$$

The arena $\textbf{qbit}^{\otimes n}$ is the arena $\left[\mathbb{C}^{2n}\right]$ corresponding to the state space of n qbits. The other arenas are operations are taken directly from classical game semantics. The arena $\top$ has one possible even-length play: $?*$, and there is thus only one possible strategy aside from the empty one. We denote this strategy $*$. The type operations $\times$ and $\Rightarrow$ correspond respectively to the arena operations $\odot$ and $\multimap$. Given a context $\Gamma = x_1\colon A_1, \ldots, x_n\colon A_n$, we set $\llbracket \Gamma \rrbracket$ to be $\llbracket A_1 \rrbracket \odot \cdots \odot \llbracket A_n \rrbracket$.

We now turn to the definition of the interpretation $\llbracket M \rrbracket$ of a term $\Gamma \vdash M\colon A$. The definition is by induction on the derivation of $\Gamma \vdash M\colon A$.

In the base case we must deal with variable and constant terms. For variables, the interpretation of $\Gamma, \overline{x}\colon A \vdash \overline{x}\colon A$ is defined using the projection strategies $\pi_A\colon \llbracket \Gamma \rrbracket \odot \llbracket A \rrbracket \to \llbracket A \rrbracket$. The denotations of the constants 0, 1, and $*$ are the standard constant strategies. A quantum state constant $\rho\colon \text{qbit}^{\otimes n}$ is interpreted as the quantum strategy $[\rho]$ in $\textbf{qbit}^{\otimes n}$.

We describe the inductive cases involving quantum operations or new ideas. The other cases are interpreted using the standard ideas of classical game semantics.

The definition of $\llbracket \Gamma, \Delta_1, \Delta_2 \vdash \mathsf{if}\, P\, \mathsf{then}\, M\, \mathsf{else}\, N\colon A \rrbracket$ differs from the usual definition for conditionals used in game semantics because of the linearity constraint. Assume that

$$\llbracket P \rrbracket : \llbracket \Gamma \rrbracket \odot \llbracket \Delta_1 \rrbracket \multimap \textbf{bool} \text{ and } \llbracket M \rrbracket, \llbracket N \rrbracket : \llbracket \Gamma \rrbracket \odot \llbracket \Delta_2 \rrbracket \multimap \llbracket A \rrbracket$$

are already defined. Using the symmetry strategy associated to $\odot$ and the duplicating strategy Δ, we can define a strategy

$$r\colon (\llbracket \Gamma \rrbracket \odot \llbracket \Delta_1 \rrbracket \odot \llbracket \Delta_2 \rrbracket) \multimap (\llbracket \Gamma \rrbracket \odot \llbracket \Delta_1 \rrbracket) \odot (\llbracket \Gamma \rrbracket \odot \llbracket \Delta_2 \rrbracket)$$

which reorganize the input arena. With this strategy, we can define $\llbracket \mathsf{if}\, P\, \mathsf{then}\, M\, \mathsf{else}\, N \rrbracket$ to be the composition $r; \llbracket P \rrbracket \odot \mathrm{id}; \mathsf{cond}(\llbracket M \rrbracket, \llbracket N \rrbracket)$, where

$$\mathsf{cond}(\llbracket M \rrbracket, \llbracket N \rrbracket)\colon \textbf{bool} \odot (\llbracket \Gamma \rrbracket \odot \llbracket \Delta_2 \rrbracket) \multimap \llbracket A \rrbracket$$

is defined using a conditional strategy operation defined in general by the following idea. Given any two arenas A and B and two strategies $\sigma, \tau\colon A \to B$, the strategy

$\mathsf{cond}(\sigma,\tau)\colon (\mathbf{bool} \odot A) \multimap B$ is the strategy that makes Player answer an initial move in B by asking for a Boolean b in the **bool** component and then makes Player play in the components A and B using the strategy σ if $b = 1$ and τ if $b = 0$.

The first quantum operation we deal with is the measurement case. Suppose that

$$[\![Q]\!] : [\![\Gamma]\!] \odot [\![\Delta_1]\!] \multimap \mathbf{qbit}^{\otimes(n+1)} \text{ and } [\![M]\!] : [\![\Gamma]\!] \odot [\![\Delta_2]\!] \odot \mathbf{bool} \odot \mathbf{qbit}^{\otimes n}$$

are already defined. We can define $[\![\mathsf{let}\, b, \overline{x} = \mathsf{meas}_i\, Q\, \mathsf{in}\, M]\!]$ as the composition $r; [\![Q]\!]\,; \mathrm{meas}_i; [\![M]\!]$ where meas_i is the strategy described as follows. Let $\mathcal{C}$ be the quantum intervention corresponding to a projective measurement in the canonical basis and $\mathcal{I}$ be the identity quantum intervention. If the first move is a question in the $\mathbf{qbit}^{\otimes n}$ arena, Player use the left scheme and if the first move is in the **bool** arena, then Player use the right scheme.

$$\begin{array}{cccc} \mathbf{qbit}^{\otimes(n+1)} & \xrightarrow{\mathrm{meas}_i} \mathbf{bool} & \odot & \mathbf{qbit}^{\otimes n} \\ & & & \mathcal{E}_? \\ \mathcal{E}_? \otimes \mathcal{C} & & & \\ (m,b) & & & \\ & & & m \\ & ? & & \\ & b & & \end{array} \qquad \begin{array}{cccc} \mathbf{qbit}^{\otimes(n+1)} & \xrightarrow{\mathrm{meas}_i} \mathbf{bool} & \odot & \mathbf{qbit}^{\otimes n} \\ & ? & & \\ \mathcal{I} \otimes \mathcal{C} & & & \\ b & & & \\ & b & & \\ & & & \mathcal{E}_? \\ \mathcal{E}_? \otimes \mathcal{I}^i & & & \\ m & & & \\ & & & m \end{array}$$

where $\mathcal{E} \otimes \mathcal{F}$ stands for the quantum intervention $\{\mathcal{E}_{m_1} \otimes \mathcal{F}_{m_2}\}_{(m_1,m_2)}$. It is important to point out that in the right scheme, Player must question Opponent two times. Since the first intervention $\mathcal{I} \otimes \mathcal{C}$ alter the state, Opponent's answer to the second question $\mathcal{E}_? \otimes \mathcal{I}^i$ depends on the first answer given. This is the only instance in the semantics described in this paper where more than one thread is necessary the $\mathrm{qbit}^{\otimes n}$ arena. Because of the side effects of measurements, we are forced to use thread dependent strategies to describe quantum states. This is the point where we are forced to assume that qbit types are linear, since thread dependent strategies cannot be duplicated using the usual Δ duplicating strategy. In contrast, previous work on quantum λ-calculi justified the need of the linearity hypothesis by no-cloning theorem.

There are three tensor cases to deal with. In the first case, we tensor two known qbits. Suppose that the strategies

$$[\![\Gamma, \Delta_1, \overline{x}_1 \colon \mathrm{qbit}^{\otimes n} \vdash M_1 \colon \mathrm{qbit}^{\otimes n}]\!] \text{ and } [\![\Gamma, \Delta_2, \overline{x}_2 \colon \mathrm{qbit}^{\otimes m} \vdash M_2 \colon \mathrm{qbit}^{\otimes m}]\!]$$

are already defined, where $\mathrm{FV}(M_i) \setminus |\Delta_i| = \emptyset$, $i = 1, 2$. The strategy $[\![M_1 \otimes M_2]\!]$ is defined as the composition $r; [\![M_1]\!] \otimes [\![M_2]\!]$, where the strategy $[\![M_1]\!] \otimes [\![M_2]\!]$ is defined by the following scheme:

$$\begin{array}{ccccc}
(\llbracket\Gamma\rrbracket \odot \llbracket\Delta_1\rrbracket) & \odot & (\llbracket\Gamma\rrbracket \odot \llbracket\Delta_2\rrbracket) & \overset{\llbracket M_1\rrbracket\otimes\llbracket M_2\rrbracket}{\multimap} & \mathbf{qbit}^{\otimes n} \otimes \mathbf{qbit}^{\otimes m} \\
 & & & & \mathcal{E}_? \\
a_1 & & & & \\
\vdots & & & & \\
a_n & & & & \\
 & & b_1 & & \\
 & & \vdots & & \\
 & & b_k & & \\
 & & & & m
\end{array}$$

where the probability that Player answers m to $\mathcal{E}_?$ after the interactions $s = a_1 \dots a_n$ and $t = b_1 \dots b_k$ is $\mathrm{tr}\,(\mathcal{E}_m\, \rho_s \otimes \rho_t)$. Note that while we take the tensor product of the two output quantum arenas, we must take the classical game product of the classical input arenas.

In the second case, we tensor two qbits each constructed from unknown qbits. This case is similar to the first one: suppose that

$$\left[\!\left[\Gamma, \Delta_1 \vdash M_1 \colon \mathrm{qbit}^{\otimes n}\right]\!\right] \text{ and } \left[\!\left[\Gamma, \Delta_2 \vdash M_2 \colon \mathrm{qbit}^{\otimes m}\right]\!\right]$$

are already defined and that $\mathrm{FV}(M_i) \cap |\Delta_i| = \{\overline{x}_i\}$. The strategy $\llbracket M_1 \otimes M_2\rrbracket$ is defined to be the composition $r \odot \mathrm{id}; \llbracket M_1\rrbracket \otimes \llbracket M_2\rrbracket$, but this time the strategy $\llbracket M_1\rrbracket \otimes \llbracket M_2\rrbracket$ must be defined using the scheme that follows :

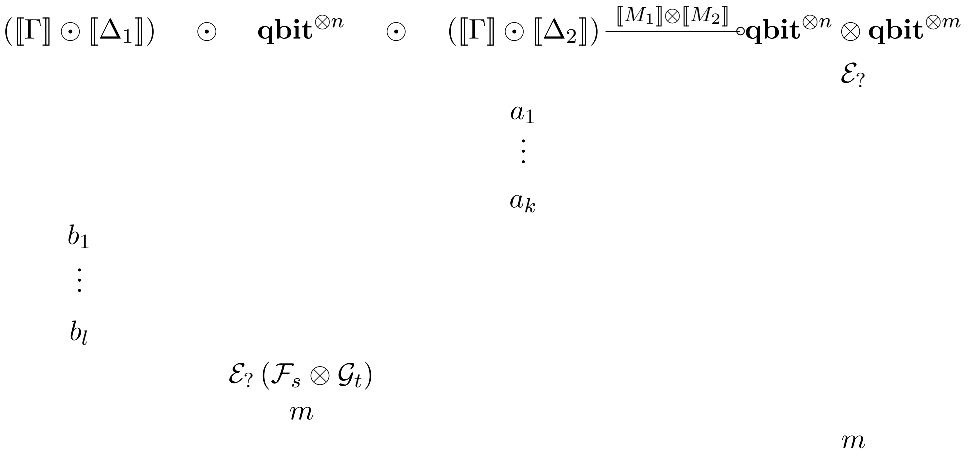
$$\begin{array}{ccccccc}
(\llbracket\Gamma\rrbracket \odot \llbracket\Delta_1\rrbracket) & \odot & (\llbracket\Gamma\rrbracket \odot \llbracket\Delta_2\rrbracket) & \odot & \mathbf{qbit}^{\otimes n} \otimes \mathbf{qbit}^{\otimes m} & \overset{\llbracket M_1\rrbracket\otimes\llbracket M_2\rrbracket}{\multimap} & \mathbf{qbit}^{\otimes n} \otimes \mathbf{qbit}^{\otimes m} \\
 & & & & & & \mathcal{E}_? \\
a_1 & & & & & & \\
\vdots & & & & & & \\
a_n & & & & & & \\
 & & b_1 & & & & \\
 & & \vdots & & & & \\
 & & b_m & & & & \\
 & & & & \mathcal{E}_?\,(\mathcal{F}_s \otimes \mathcal{G}_t) & & \\
 & & & & m & & \\
 & & & & & & m
\end{array}$$

where $\mathcal{F}_s$ and $\mathcal{G}_t$ are the two trace-preserving superoperators used by Player respectively in $\llbracket M_1\rrbracket$ and $\llbracket M_2\rrbracket$.

The third tensor rule is for cases where known and unknown states are tensored. In this case we have to use a conditional preparation strategy defined using a combination of schemes used in the first two cases. Assume that

$$[\![\Gamma, \Delta_1, \overline{x} \colon \mathrm{qbit}^{\otimes n} \vdash M_1 \colon \mathrm{qbit}^{\otimes n}]\!] \text{ and } [\![\Gamma, \Delta_2 \vdash M_2 \colon \mathrm{qbit}^{\otimes m}]\!]$$

are already defined and that $\mathrm{FV}(M_1) \setminus |\Delta_1| = \{\overline{x}\}$ and $\mathrm{FV}(M_2) \cap |\Delta_2| = \emptyset$. The strategy $[\![M_1 \otimes M_2]\!]$ is defined as the composition $r; [\![M_1]\!] \otimes [\![M_1]\!]$ where this time the tensor strategy $[\![M_1]\!] \otimes [\![M_2]\!]$ is defined with the scheme


$([\![\Gamma]\!] \odot [\![\Delta_1]\!])$
$\odot$
$\mathbf{qbit}^{\otimes n}$
$\odot$
$([\![\Gamma]\!] \odot [\![\Delta_2]\!]) \xrightarrow{[\![M_1]\!] \otimes [\![M_2]\!]} \mathbf{qbit}^{\otimes n} \otimes \mathbf{qbit}^{\otimes m}$
$\mathcal{E}_?$
a_1
$\vdots$
a_k
b_1
$\vdots$
b_l
$\mathcal{E}_? \, (\mathcal{F}_s \otimes \mathcal{G}_t)$
m
m

Player determines how to answer the initial question $\mathcal{E}_?$ by first playing in the $[\![\Gamma]\!] \odot [\![\Delta_2]\!]$ arena to determine which state ρ_s, $s = a_1 \ldots a_k$, to prepare; we assume this state is prepared by a superoperator $\mathcal{F}_s$. After this, Player will start an interaction in $[\![\Gamma]\!]$ in order to learn how the state represented by the term M_1 is build from its input. In this case, we assume that this construction corresponds to a superoperator $\mathcal{G}_t$, where $t = b_1 \ldots b_l$ is the interaction in the $[\![\Gamma]\!]$ part. The initial question is then transformed into the question $(\mathcal{F}_s \otimes \mathcal{G}_t)\, \mathcal{E}_?$ in the input arena $\mathbf{qbit}^{\otimes n}$, and the answer is copied back to the output arena.

4 Soundness

We now turn to the problem of proving a soundness result for the denotational semantics defined in the last section. First, we need a substitution lemma.

Lemma 4.1 *For any* λ*-calculus with quantum data terms* $\Gamma, \Delta_1, \overline{x} \colon A \vdash M \colon B$ *and* $\Gamma, \Delta_2 \vdash N \colon A$ *with* $\overline{x} \in \mathrm{FV}(M)$*, we have that*

$$\Gamma, \Delta_1, \Delta_2 \vdash M\,[N/\overline{x}] : B \text{ and } [\![M[N/\overline{x}]]\!] = r; \mathrm{id} \odot [\![N]\!]\,; [\![M]\!]$$

Proof. This is proven by structural induction on the construction of M. □

The following proposition states that when a term M reduce to some value V with probability p, the corresponding strategies $[\![M]\!]$ and $[\![V]\!]$ makes Player play in the same way with probability p.

Proposition 4.2 *If $M \Downarrow^p V$, then for all well-opened $sab \in \mathcal{T}(\llbracket V \rrbracket)$ we have that*

$$\llbracket M \rrbracket\,(b \mid sa) = p\,\llbracket V \rrbracket\,(b \mid sa).$$

Proof. By structural induction on the derivation of $M \Downarrow^p V$. Most of the proof follow the usual argument for the classical case. We skip these to focus on the cases involving quantum operations.

For measurement operations, consider first the single qbit case. Suppose that $\llbracket M \rrbracket$ behaves as $\llbracket \rho \rrbracket$ with probability p. Assume that $\mathsf{meas}\, M$ reduces to 0 with probability $p\,\mathrm{tr}(|0\rangle\langle 0|\,\rho)$. The strategy $\llbracket \mathsf{meas}\, M \rrbracket$ is the composition $\llbracket M \rrbracket\,; \mathrm{meas}$ and, by induction hypothesis, any interaction using this strategy will behave as an interaction using the strategy $[\rho]; \mathrm{meas}$. By definition of $[\rho]$, this strategy behaves as the constant strategy 0 in **bool** with probability $\mathrm{tr}\,(|0\rangle\langle 0|\,\rho)$, and thus $\llbracket \mathsf{meas}\, M \rrbracket$ behaves as $\llbracket 0 \rrbracket$ with probability $p\,\mathrm{tr}\,(|0\rangle\langle 0|\,\rho)$.

The general measurement case is similar.

To deal with the tensor operation reduction rule, suppose that the proposition holds when $M_1 \Downarrow^p V_1$ and $M_2 \Downarrow^q V_2$ and assume that $M_1 \otimes M_2 \Downarrow^{pq} V_1 \otimes V_2$. Since the definition of $\llbracket M_1 \otimes M_2 \rrbracket$ is in three cases, these must be considered separately. In the first case, M_1 and M_1 are both terms with no free variables of type qbit appearing in the type context. By definition $\llbracket M_1 \otimes M_2 \rrbracket = r \odot \mathrm{id}; \llbracket M_1 \rrbracket \otimes \llbracket M_2 \rrbracket$ and by the induction hypothesis this will behaves as $\llbracket M_1 \otimes M_2 \rrbracket = r \odot \mathrm{id}; \llbracket V_1 \rrbracket \otimes \llbracket V_2 \rrbracket$ with probability pq. The other two cases are similar, except that the definition of $\llbracket M_1 \rrbracket \otimes \llbracket M_2 \rrbracket$ is different in each case. □

The next result is adequacy, the converse of the previous one. As for classical λ-calculus, we use a computability predicate to prove adequacy for QDL. The main difference between the following definition and the usual definition of computability is the use of extended variables. Note that neither the presence of extended variables or linearity have any significant impact on this definition.

Definition 4.3 A QDL term M is computable if

(i) M is closed with $M\colon A$ and $A =$ bool, $\top$ or qbit, and if for all $sab \in \mathcal{T}\,(\,b \mid sa)$ we have that $\llbracket M \rrbracket\,(\,b \mid sa\,) = p\,\llbracket V \rrbracket\,(\,b \mid sa\,)$, then $M \Downarrow^p V$,

(ii) $\overline{x_1}\colon A_1, \ldots, \overline{x_n}\colon A_n \vdash M\colon A$ and for all computable closed terms $\Gamma \vdash N_1\colon A_1, \ldots, \Gamma \vdash N_n\colon A_n$ we have that $M[N_1/\overline{x_1}, \ldots, N_n/\overline{x_n}]$ is computable,

(iii) M is closed with $\vdash M\colon A \Rightarrow B$ and for all closed N with $\vdash N\colon A$ the term MN is computable.

Lemma 4.4 *All QDL terms are computable.*

Proof. By induction on the construction of M. The part of the proof involving classical constructs is follows the usual pattern as in classical game semantics, so we focus here on the quantum operations. Using the definition of computability, we can assume that the building components of M are computable closed terms.

The most interesting case is measurement since it involve an argument specific to QDL. We begin by the one qbit measurement case. Suppose that $M = \mathsf{meas}\, N$

where N is a closed computable term of type qbit. Assume that V is a boolean value and that $[\![M]\!]\,(\, b \mid sa\,) = p\,[\![V]\!]\,(\, b \mid sa\,)$ for all well-opened $sab \in \mathcal{T}([\![V]\!])$.

When Player uses $[\![M]\!]$, a typical play is

$$\begin{array}{ccccc} I & \overset{[\![N]\!]}{\multimap} & \mathbf{qbit} & \overset{\mathsf{meas}}{\multimap} & \mathbf{bool} \\ &&&& ? \\ && \mathcal{C}_? && \\ && m && \\ &&&& m \end{array}$$

where $\mathcal{C}_?$ is the quantum intervention corresponding to a projective measurement in the canonical basis. Let p be the probability that using $[\![N]\!]$ the answer is 0 and $1-p$ the probability that the answers is 1. Although it is not possible to infer which state ρ is used to answer $\mathcal{C}_?$ using these probabilities, we know that if player was using $\rho' = p|0\rangle\langle 0| + (1-p)|1\rangle\langle 1|$ instead of ρ, we would get the same play as above. Since $\mathsf{meas}\,\rho' \Downarrow^p 0$, we get that $\mathsf{meas}\,\rho \Downarrow^p 0$ as required.

We use a similar argument to deal with the general measurement case. For unitary operations, the above problem does not occur since the strategy $[\![\mathcal{U}M]\!] = [\![M]\!]\,;[\mathcal{U}]$ provides the measurement probabilities for all quantum interventions $\mathcal{E}_?$. This allow one to find, via the Gleason theorem, a state ρ such that $[\![M]\!]$ behaves like $[\rho]$ with probability p. Using this and the induction hypothesis on M, we get the desired result. □

Adequacy is a direct corollary to the last lemma.

Theorem 4.5 *Let M be a closed term of type* bool*,* $\top$ *or* $\text{qbit}^{\otimes n}$*. If for all well-opened $sab \in \mathcal{T}\,([\![V]\!])$ we have that $[\![M]\!]\,(\, b \mid sa\,) = p\,[\![V]\!]\,(\, b \mid sa\,)$, then we have that $M \Downarrow^p V$.*

To give the final result, we need to introduce the necessary concept of contextual equivalence for QDL. A *context* $C[-]$ of type B with a hole of type A is a term $C[-]$ with a special variable "$-$" (possibly an extended variable) such that $-\colon A \vdash C[-]\colon B$. Capture-free substitution of a term N in a context $C[-]$ is denoted $C[N]$.

Definition 4.6 Two closed terms $\vdash M_1\colon A$ and $\vdash M_2\colon A$ are *contextually equivalent* if for every ground-type context $C[-]$ with a hole of type A we have that

$$C[M_1] \Downarrow^p V \iff C[M_2] \Downarrow^p V.$$

The following soundness result follows from consistency and adequacy using a standard argument.

Theorem 4.7 *(Soundness) Let M_1 and M_2 by two closed QDL terms. If $[\![M_1]\!] = [\![M_2]\!]$, then $M_1 \sim M_2$.*

5 Conclusion and future work

We introduced a new quantum λ-calculus and, using tools from game semantics, we obtained a soundness result which validated its syntax and the structure of its type system. Some important features of the language, like linearity and the different form of the tensor operation, were motivated directly using the properties of the quantum strategies used to model the language. The game semantics approach allowed us to model directly the classical and quantum constructs of the language and could be extended to languages with extra features, like recursion, using ideas from classical game semantics. Usually game semantics is used to get full-abstraction results by putting appropriate restrictions on the strategies. Here the main goal was instead to introduce a new kind of model for quantum programming languages. While the soundness result we obtained confirms the usefulness of using quantum games to model quantum types, it is a natural next step to seek a full-abstraction result for QDL. The main difficulty is that there is no known characterisation of the probabilistic strategies of the form $[\mathcal{F}]$ in $[H_A] \multimap [H_B]$ among all possible probabilistic strategies in this arena. Gleason's theorem [6] is one result in this spirit, but there is no similar result for the case of superoperators. A full abstraction result here would thus be a major advance in understanding how to characterize quantum processes. In this case the obstacle has nothing to do with the usual subtleties associated with higher-type languages.

We did not explore fully the categorical properties of quantum arenas introduced in this paper. For example, one could consider the category of the category of arenas of the form $[H]$ and probabilistic strategies that correspond to quantum operations. This category or some of its subcategories could provide new models for the categorical structures associated to quantum mechanics [1,2,13].

Finally, note that there is a way to relax the definition of quantum arena given in this paper by dropping the condition that the question in $[H]$ consists of quantum intervention on H. The resulting arena allow Opponent to use quantum interventions over any space. This possibility was useful in [4] to model a λ-calculus equipped *quantum stores* which can contain quantum states of variable size. In that language, quantum data can only used though references in the language; this makes the linearity constraint unnecessary since having multiple references to a qbit is not forbidden by the no-cloning theorem. In this case also the properties of quantum strategies were used as a guide in the construction of the language, a further demonstration of the usefulness of quantum strategies in the study of higher-order quantum programming languages.

References

[1] Abramsky, S. and B. Coecke, *A categorical semantics of quantum protocol*, in: *Proceedings of the 19th IEEE conference on Logic in Computer Science: LICS 2004* (2004), pp. 415–425.

[2] Coecke, B. and D. Pavlovic, *Quantum measurements without sums* (2006).

[3] Danos, V. and R. Harmer, *Probabilistic game semantics*, in: *ACM Transactions On Computational Logic, Special Issue for LICS'00*, Association For Computing Machinery (2002), pp. 359–382.

[4] Delbecque, Y. and P. Panangaden, *Game semantics for quantum stores*, in: A. Bauer and M. Mislove, editors, *Mathematical Foundations of Programming Semantics* (2008), pp. 119–139.

[5] Gell-Mann, M. and J. Hartle, *Classical equations for quantum systems*, Physical Review D **47** (1993), pp. 3345–3382.

[6] Gleason, A. M., *Measures on the closed subspaces of a Hilbert space*, Journal of Mathematics and Mechanics (1957), pp. 885—893.

[7] Griffiths, R., *Consistent histories and quantum reasoning*, Physical Review A **54** (1996), pp. 2759–2774.

[8] Harmer, R., "Games and Full Abstraction for Nondeterministic Languages," Ph.D. thesis, Imperial College (1999).

[9] Jacobs, B., *Semantics of weakening and contraction*, Annals of Pure and Applied Logic (1994), pp. 73–106.

[10] Omnès, R., "The Interpretation of Quantum Mechanics," Princeton Univ. Press, 1994.

[11] Peres, A., *Classical interventions in quantum systems. I. The measuring process*, Physical Review A **61** (2000).

[12] Selinger, P., *Towards a semantics for higher-order quantum computation*, Proceedings of the 2nd International Workshop On Quantum Programming Languages, Turku, Finland. (2004), pp. 127–143.

[13] Selinger, P., *Dagger compact closed categories and completely positive maps*, in: *Proceedings of the 3rd International Workshop on Quantum Programming Languages*, number 170 in Electronic Notes in Theoretical Computer Science, 2007, pp. 139–163.

[14] Selinger, P. and B. Valiron, *A lambda calculus for quantum computation with classical control*, Mathematical Structures in Computer Science **16** (2006), pp. 527–552.

[15] Selinger, P. and B. Valiron, *On a fully abstract model for a quantum linear functional language*, in: *Proceedings of the 4th International Workshop on Quantum i Programming Languages, Oxford, July 17-19*, 2006.

[16] Valiron, B., "A functional programming language for quantum computation with classical control," Master's thesis, Department of Mathematics, University of Ottawa (2004).