Int. J. Appl. Math. Comput. Sci., 2008, Vol. 18, No. 3, 341–359
DOI: 10.2478/v10006-008-0031-x

NATURAL QUANTUM OPERATIONAL SEMANTICS WITH PREDICATES

MAREK SAWERWAIN, ROMAN GIELERAK

Institute of Control and Computation Engineering
University of Zielona Góra, ul. Podgórna 50, 65–246 Zielona Góra, Poland
e-mail: {M.Sawerwain,R.Gielerak}@issi.uz.zgora.pl

A general definition of a quantum predicate and quantum labelled transition systems for finite quantum computation systems is presented. The notion of a quantum predicate as a positive operator-valued measure is developed. The main results of this paper are a theorem about the existence of generalised predicates for quantum programs defined as completely positive maps and a theorem about the existence of a GSOS format for quantum labelled transition systems. The first theorem is a slight generalisation of D'Hondt and Panagaden's theorem about the quantum weakest precondition in terms of discrete support positive operator-valued measures.

Keywords: quantum computation, predicate notion for quantum programs, quantum labelled transition systems.

1. Introduction

Operational semantics (Plotkin, 2004) developed in the mid-1970s is the first formal method describing the behaviour of computer programs. Nowadays, three major approaches to the semantic analysis of classical programs exist: operational, denotational and axiomatic semantics. A motivation underlying the progress in the area of operational semantics (this paper discusses only such semantics) is the interest in how the process of computation is conducted. It is especially significant in quantum computation science. Although an enormonous activity in this area has been observed in the last period, few quantum algorithms exist. Many researchers took a sceptical point of view concerning further development of quantum algorithms (Shor, 2004).

It seems to us that none of the existing quantum computational models was sufficiently well understood in order to develop systematic tools for the construction of new, interesting quantum algorithms. For example, the entanglement is not fully explained, but this phenomenon is used directly in many significant quantum algorithms: the teleportation protocol, superdense coding and the cryptographic protocol called E91 are good examples. Therefore, research on operational semantics is very welcome and important for better understanding of the very nature of quantum algorithms.

In this paper, we concentrate only on quantum operational semantics with predicates for finite quantum systems. The notion of the quantum predicate is inspired by the classical predicate notion and can be used in any discrete quantum model including the standard circuit and one-way quantum computation models (Jozsa, 2005; Raussendorf and Briegel, 2001; Raussendorf *et al.*, 2003).

There are many important reasons for considering operational semantics for quantum programs:

- Operational semantics is a simple theory suitable to express the meaning of classical programs. It is our hope that a similar theory can be formulated for quantum computation models.
- The introduction of an appropriate theoretical notion to compare two seemingly different quantum algorithms or quantum programs will be helpful, and the bisimulation notion of different quantum computational models is very welcome.
- The quantum computation model is very different from classical computation models (CCMs). The notions of superposition or entanglement do not appear in CCMs. The application of some notions of classical operational semantics theory to describe the meaning of quantum programs is helpful to prove some interesting properties, e.g., the determinism and finiteness of quantum computation.
- The quantum counterpart of the classical program synthesis algorithm can be developed with the use of predicates.

The presented paper is organised as follows: Section 2 contains basic definitions and some facts from the theory of quantum labelled transition systems (termed QLTS). We also establish a GSOS (grand or general structural operational semantics) format for finite quantum labelled transitions systems, which is based on the work (Aceto, 1994). We also recall classical definitions of predicates together with the weakest precondition and the strongest postcondition notions.

In Section 3 the notion based on a positive operator-valued measure as a general quantum predicate is introduced and some elementary properties of this object are formulated. The compatibility of our general definition with those previously used in several papers is demonstrated as well. We also present a technical theorem about the existence of the predicates for a quantum labelled transition. The second very important result of the present paper is the proof of the existence of the weakest quantum precondition for predicates by a POVM supported by discrete event spaces. A generalisation of this result to an arbitrary POVM is presented in (Gielerak and Sawerwain, 2007). The duality between the weakest precondition and the strongest postcondition is also shown in this section.

Section 4 contains some applications of quantum predicates to an analysis of some well-known quantum algorithms like the Grover algorithm and a superdense coding protocol. We also present a procedure to simulate the application of a unitary gate with a one-way quantum computation model (termed 1WQC). The proof tree of such a procedure will be presented in the last part of Section 4.

It is the main drawback of the present paper that it focuses the semantic analysis of the algorithms discussed below on the classical aspects only and thus it should be treated as a preliminary step in the direction of developing sufficiently efficient and powerful quantum computational tools for deeper understanding of the very nature of the quantum computation process.

2. Quantum labelled transition systems

A transition system is the triple $(S, A, \longrightarrow)$, where S is a set of states, A is a set of actions (also called the label) and $\longrightarrow$ is a relation called the transition relation. The relation $\longrightarrow$ satisfies

$$\rightarrow \subseteq S \times A \times S. \tag{1}$$

If $(s, \alpha, s^{'}) \in \rightarrow$, then we write $s \xrightarrow{\alpha} s^{'}$. Any configuration s such that $s \overset{\alpha}{\nrightarrow}$, which means that there is no possibility to leave this state by the allowed actions from the set A, is called terminal (or final).

Let us recall the classical notion of the 0–1 Boolean predicate. Let $f \in S$. We call the state f a predicate for a state $x \in S$ or for a sequence $x_1, x_2, x_3, \ldots, x_n \in S$, if for some Boolean function $\mathrm{Pre} : S \times S \rightarrow \{\mathrm{true}, \mathrm{false}\}$, we have

$$\mathrm{Pre}(f, x) = \mathrm{true}, \tag{2}$$

and, respectively for the sequence,

$$\forall_i \ \mathrm{Pre}(f, x_i) \ = \ \mathrm{true}. \tag{3}$$

In other words, if f is a predicate and s a state, then we write $s \models f$ instead of $\mathrm{Pre}(f, x)$, which means that the state s is true for the predicate f.

The weakest precondition (in the literature known as the weakest liberal precondition) is a well-known paradigm of the goal-directed programming methodology and semantics for programming languages. The weakest-precondition was developed by (de Bakker and de Roever, 1972; de Bakker and Meertens, 1975) and popularised by Dijkstra (1976). This notion is connected with the Hoare triple $\{f_1\}P\{f_2\}$ (Hoare, 1969), where f_1 and f_2 denote some predicates and P is a program. In other words, the Hoare triple says that, if f_1 is true in some entry state and executing P in the entry state can yield another final state, then f_2 is true in this final state.

For any action $a \in A$ and a predicate f_2, we define the predicate $wp(a, f_2)$ as

$$s \models wp(a, f_2) = \forall_{t \in S}((s, a, t) \in \longrightarrow) \Rightarrow (t \models f_2). \tag{4}$$

wp is the weakest precondition operator and the predicate $wp(a, f_2)$ is the weakest one satisfying the Hoare triple $\{wp(a, f_2)\}a\{f_2\}$. The Hoare triple can be expressed with the wp operator $\models f_1 \Rightarrow wp(a, f_2)$.

The strongest postcondition (termed 'sp') is defined by

$$t \models sp(a, f_1) = \exists_{s \in S}(s, a, t) \in \longrightarrow) \wedge s \models f_1. \tag{5}$$

From the definition of the Hoare triple we obtain that $\models sp(a, f_1) \Rightarrow f_2$ is equivalent to $\{f_1\}a\{f_2\}$.

Before we introduce the definition of a quantum labelled transition system, we give some remarks about the form of the states which are processed by a given transition system.

Let M be a quantum system. Then there exists an associated Hilbert space $\mathcal{H}_M$ of states. In this paper the set of pure states will be identified with the unit sphere

$$\partial E(\mathcal{H}) = \{|\psi\rangle \in \mathcal{H}_M : \||\psi\rangle\| = 1\} \subset \mathcal{H}_M,$$

and the set of all states will be denoted by $E(\mathcal{H}_M)$.

If the system considered is composed of n identical copies of M, then the corresponding state space is formed by applying the tensor product $\otimes$. In particular, pure states correspond to points of the unit sphere of the space $\mathcal{H}_M^{\otimes^n}$.

In many realistic situations the notion of a pure state is not the appropriate one and must be generalised to the notion of a mixed state. Mixed states of a system M correspond to the convex set $E(\mathcal{H}_M)$ of all semipositive endomorphisms ρ of the space $\mathcal{H}_M$ and with the trace equal to 1, i.e., $\mathrm{Tr}\,(\rho) = 1$.

The set of all states on $\mathcal{H}_M$ will be denoted by $E(\mathcal{H}_M)$. The unpleasant feature of the convex set $E(\mathcal{H}_M)$ which causes many problems is the lack of a simplex structure. From the very definition it follows that any mixed state $\rho \in E(\mathcal{H}_M)$, called the density matrix, frequently has the property of being nonnegative and having $\mathrm{Tr}\,(\cdot)$ equal to 1, i.e., $\mathrm{Tr}\,(\rho) = 1$. If, additionally,

$$\mathrm{Tr}\,\left(\rho^2\right) = 1,$$

then it follows that ρ is a pure state, i.e., there exists $|\psi\rangle \in \mathcal{H}_M$ such that $\rho = |\psi\rangle\langle\psi|$.

The possibility of describing quantum algorithm structures in terms of the underlying generalisation of labelled transition system methods was started in (Sawerwain *et al.*, 2006). Let us recall some definitions.

Definition 1. A general quantum labelled transition system (qLTS) is the triple $\langle E_r(\mathcal{H}), \mathrm{Op}, \twoheadrightarrow\rangle$, where:

- $E_r(\mathcal{H})$ is a closed subspace of the set of all states on $\mathcal{H}$, in the sequel denoted by S,
- Op is some set of operations which are realised by completely positive maps that might include, among other things, some unitary operations, and/or some measurement operations,
- $\twoheadrightarrow$ is the labelled transition relation: $\twoheadrightarrow \subseteq E_r(\mathcal{H}) \times \mathrm{Op} \times E_r(\mathcal{H})$; in particular, we write $\rho \overset{\alpha}{\twoheadrightarrow} \rho^{'}$ if $(\rho, \alpha, \rho^{'}) \in \twoheadrightarrow$.

Depending on the particular content of the set Op, we may consider many admissible computational steps. For example, we can select two basic types of transition:

- A β-type transition, denoted by $\overset{\beta}{\twoheadrightarrow}$, represented by a unitary operator from $\mathcal{U}(\mathcal{H})$, the set of all unitary operators acting on the Hilbert space $\mathcal{H}$. The β-type transition is denoted by the following rule:

$$\frac{-}{|\psi\rangle \overset{\beta}{\twoheadrightarrow} |\psi'\rangle}.$$

- A μ-type transition, denoted by $\overset{\mu}{\twoheadrightarrow}$. This step is represented by the measurement operation from $\mathcal{O}(\mathcal{H})$ and denoted by the following rule, which measures a part or the whole of the quantum register:

$$\frac{-}{|\psi\rangle|\phi\rangle \overset{\mu}{\twoheadrightarrow} |\psi_k\rangle_\perp |\phi\rangle} \quad \text{or} \quad \frac{-}{|\psi\rangle \overset{\mu}{\twoheadrightarrow} |\psi_k\rangle_\perp}.$$

For a $\overset{\beta}{\twoheadrightarrow}$ type transition, we distinguish an adjoint Hermitian operator $\overset{\beta^\dagger}{\twoheadrightarrow}$. This operator represents a reversible operation and it is still a unitary operator.

The β-type transition has the specific property of being reversible, which generally does not appear in the classical LTS theory. Although it might be understood as completely trivial, we formulate the following proposition stressing the fact of the existence of a reverse operation in every β step in a given quantum transition system.

Proposition 1. *Let $\langle \partial E_r(\mathcal{H}_M), \mathrm{Op}, \twoheadrightarrow\rangle$ be a quantum labelled transition system and $\mathcal{U}(\mathcal{H}_M) \subseteq \mathrm{Op}$, where $\mathcal{U}(\mathcal{H}_M)$ denotes the set of all unitaries acting on $\mathcal{H}_M$ and $\partial E_r(\mathcal{H}_M)$ stands for a set of allowed pure states. Then, for any $(|\psi\rangle, U, |\psi^{'}\rangle) \in \twoheadrightarrow$ it follows that $(|\psi^{'}\rangle, U^\dagger, |\psi\rangle) \in \twoheadrightarrow$.*

Proof. To prove this proposition, we use the adjointness property in the Hilbert space $\mathcal{H}$. Let $|x\rangle, |y\rangle, |z\rangle$ be vectors in $\mathcal{H}$ and let U be a unitary operator. By definition, $\langle Ux|y\rangle = \langle x|U^\dagger y\rangle$. Therefore,

$$(|x\rangle \overset{U}{\twoheadrightarrow} |y\rangle \overset{U^\dagger}{\twoheadrightarrow} |z\rangle \wedge |z\rangle \overset{U^\dagger}{\twoheadrightarrow} |y\rangle \overset{U}{\twoheadrightarrow} |x\rangle) \Rightarrow |x\rangle = |z\rangle. \quad (6)$$

The property $U^\dagger = U^{-1}$ is also used in this proof. ■

Remark 1. The β-type transition can be formulated as follows: For any $|\psi\rangle \overset{\beta}{\twoheadrightarrow} |\psi^{'}\rangle$ there exists a transition $|\psi^{'}\rangle \overset{\beta^\dagger}{\twoheadrightarrow} |\psi\rangle$, where $\beta\beta^\dagger = \beta^\dagger\beta = \mathbb{I}$.

Remark 2. In fact it is well-known that among completely positive operations only unitary operations are reversible.

The concept of an operational trace is very useful in operational proofs.

Definition 2. Let s, t be a pair of quantum states belonging to the set of states of a given qLTS $\langle E_r(\mathcal{H}_M), \mathrm{Op}, \twoheadrightarrow\rangle$. We will say that the state t is *reachable* from s iff there exists a sequence of operators $(a_1, \dots, a_n)$ from Op and such that

$$s \overset{a_1}{\twoheadrightarrow} q_1 \overset{a_2}{\twoheadrightarrow} q_2 \overset{a_3}{\twoheadrightarrow} q_3 \overset{a_4}{\twoheadrightarrow} \dots q_{n-1} \overset{a_n}{\twoheadrightarrow} t \quad (7)$$

for some intermediate states $q_i \in E_r(\mathcal{H}_M)$. Any such sequence will be called the operational trace of the pair (s, t) and will bedenoted by $T_{op}(s, t)$.

Let $\mathcal{M}$ be a projective measurement with the spectral set $\{\lambda_m, |\psi\rangle_m\}, m = 1, \dots, N$ and let $s \in \partial E(\mathcal{H})$ be a system state. Then the action of $\mathcal{M}$ on the state s can be described by the probabilistic data $\{U_\alpha, p_\alpha\}, \alpha = 1, \dots, N$, where U_α is a unitary action determined by the equality $U_\alpha|s\rangle = |\psi_\alpha\rangle$ (which, of course, does not determine the operator U_α in a unique manner!) and where

$p_\alpha = |\langle\psi|\psi_\alpha\rangle|^2$ are the corresponding probabilities. Having such data, we can replace each μ-computational step of a given qLTS $\langle E(\mathcal{H}_M), \mathrm{Op}, \twoheadrightarrow\rangle$ by the corresponding stochastic β-unitary step connected to the probabilistic representation as above.

Definition 3. Let $\langle E(\mathcal{H}_M), \mathrm{Op}, \twoheadrightarrow\rangle$ be a qLTS. A probabilistic β-step representation of all μ-steps of a given system will be called a probabilistic version of $\langle E(\mathcal{H}_M), \mathrm{Op}, \twoheadrightarrow\rangle$.

However, if a quantum algorithm is expressed throughout a quantum circuit containing only unitary gates and measurement gates taking measurements for states which are eigenstates of an observable, then the computation process is deterministic.

Proposition 2. *Any deterministic quantum algorithm has the following operational description:*

$$\frac{}{|\psi^{n-1}\rangle \overset{\beta}{\twoheadrightarrow} |\psi^n\rangle}, \quad \frac{}{|\psi^n\rangle \overset{\beta}{\twoheadrightarrow} |\psi^{n+1}\rangle_\perp},$$

$$\frac{}{|\psi^{n+1}\rangle_\perp \overset{\mu}{\twoheadrightarrow} |\psi^{n+1}\rangle_\perp},$$

where $|\psi\rangle_\perp$ is the state which is one of the eigenstates of the observable used in the measurement process.

2.1. Operational semantics as forward semantics.

The operational meaning in qLTS is defined by the natural operational function: $\mathcal{O}_N : E(\mathcal{H}) \to \mathrm{Op}^*$ (by Op^* we denote the set of all finite sequences $(op_1, op_2, \ldots, op_n)$ of the set Op). For a class of qLTS where the states are described by pure states, the function $\mathcal{O}_N$ can be defined as

$$\mathcal{O}_N(|\psi\rangle) = \Big\{ u \in \mathrm{Op} : \exists |\psi^{'}\rangle \in E(\mathcal{H}) \wedge (|\psi\rangle, u, |\psi^{'}\rangle) \in \twoheadrightarrow \Big\}. \tag{8}$$

In terms of semantics, the operational functions $\mathcal{O}_N$ describe quantum programs as finite sequences of admissible operators. This means that the operational meaning of a given quantum program P is denoted by

$$[\![P]\!] = [u_1, u_2, u_3, \ldots, u_n], \quad u_i \in \mathrm{Op}. \tag{9}$$

However, according to Definition 3, forward semantics can be represented as a sequence of unitary operators (albeit the probabilistic one) or a sequence of sets (containing unitary operators only),

$$[\![P]\!] = [\{\mathcal{U}\}_1, \{\mathcal{U}\}_2, \{\mathcal{U}\}_3, \ldots, \{\mathcal{U}\}_n]. \tag{10}$$

2.2. Operational proof trees.

Similarly to the classical framework of operational semantics, the notion of a proof tree can be formulated for quantum labelled transition systems. Let s be a state. Then a state t is reachable from s if a trace exists for s such that

$$T_{\mathrm{op}}(s,t) = (a_1, a_2, a_3, \ldots, a_n).$$

The existence of a set of labels means that a set of states reachable from s exists,

$$s \overset{a_1}{\longrightarrow} q_1 \overset{a_2}{\longrightarrow} q_2 \overset{a_3}{\longrightarrow} q_4 \ldots q_n \overset{a_n}{\longrightarrow} t.$$

Then the states q_i and t are elements of the set $\mathrm{rea}(s)$ composed of all elements that are reachable from s.

Definition 4. Let $L = \langle E(\mathcal{H}_M), \mathrm{Op}, \twoheadrightarrow\rangle$ be a qLTS. A *proof tree* (process graph) of a given L is a directed and rooted graph. The edges of this graph are labelled by the operations $u \in \mathrm{Op}$, and the edges $E(t)$ are described by the relation $E(t) \subseteq N(t) \times \mathrm{Op} \times N(t)$. More formally, we say that a proof tree t is the following triple:

$$(N(t), R(t), E(t)),$$

where N denotes the nodes (states from $E(\mathcal{H}_M)$), R is a root node (initial state) and E is a set of edges of the proof tree t.

It is possible to define the process graph as $(s, (S, A, \twoheadrightarrow))$, where s is the initial state and the triple $(S, A, \twoheadrightarrow)$ is a qLTS. Then $(s, (S, A, \twoheadrightarrow))$ is a process graph where s represents the root of the graph, and we restrict $(S, A, \twoheadrightarrow)$ only to the part composed of states that are reachable from the root state s.

2.3. Quantum labelled transition systems as GSOS rules.

Generally, checking whether a given quantum programming language yields a finite quantum labelled transition system is very hard. In any case, we have to build a corresponding proof tree to check all computational paths. Therefore, it is interesting to develop another language specification where the finiteness of the defined quantum labelled transition system arises naturally. The GSOS format (Bloom, 1989; Bloom et al., 1989) is one example of such a language specification.

Let $S = \{x_1, x_2, \ldots, x_n\}$ denote the set of states belonging to the appropriate Hilbert space $\mathcal{H}_M$, and let the set $L = \{l_1, l_2, \ldots, l_n\}$ represent the set of actions (β and μ steps) which can be applied to states from the set S.

Definition 5. A *signature* is a collection Σ of function symbols f and $f \notin S$. The signature Σ is equipped with a function $\mathrm{ar} : \Sigma \to \mathbb{N}$. The value $\mathrm{ar}(f)$ gives the – admissible arrnes of f. The set $\mathbb{T}(\Sigma)$ of terms over Σ is described recursively by

(i) $S \in \mathbb{T}(\Sigma)$,

(ii) if $f \in \Sigma$ and $t_1, t_2, \dots, t_{\mathrm{ar}(f)} \in \mathbb{T}(\Sigma)$, then $f(t_1, t_2, \dots, t_{\mathrm{ar}(f)}) \in \mathbb{T}(\Sigma)$.

The GSOS format for a quantum labelled transition system has a form similiar to the classical version (Aceto, 1994).

Definition 6. A GSOS rule has the following form:

$$\frac{\begin{array}{lll} x_i \xrightarrow{l} (\cdot) & l \in R_i, 1 \leq i \leq n & \text{prohibited formula} \\ x_i \overset{l}{\nrightarrow} & l \in P_i, 1 \leq i \leq n & \text{negative formula} \\ x_i \xrightarrow{l_j} y_{i_j} & 1 \leq j \leq m & \text{positive formula} \end{array}}{f(x_1, x_2, \dots, x_n) \xrightarrow{a} t},$$

where f represents the operation symbol β or a μ computational step. Here t is a target and it is a term where at most states x_i and y_i appear. The set of rules will be denoted by R.

The main difference between the classical and quantum versions of GSOS is the definition of the substitution function. The substitution function, or more precisely a closed Σ-substitution, is a function σ from a finite set of variables to finite closed terms over the signature Σ. In other words, for each term P, $P\sigma$ denotes the result of the substitution $\sigma(x)$ for each variable x occurring in the term P. The quantum substitution function has the following forms:

1. $\sigma_\beta(\cdot)$ – unitary steps on the variables,
2. $\sigma_\mu(\cdot)$ – measurement steps on the variables,
3. $\sigma^e_\beta(\cdot)$ – unitary steps on the entangled variables,
4. $\sigma^e_\mu(\cdot)$ – measurement steps on the entangled variables.

Definition 7. Let $G_q = (\Sigma_{G_q}, R_{G_q})$ be a quantum GSOS system. Then the operator dependence $\overset{G}{\prec}$ for the process graph is given by a directed graph for G_q with:

1. the set of nodes from Σ_{G_q},
2. the set of edges E given by (f, g) iff a rule $\rho \in R_{G_q}$ exists for the operation f and the target term g.

Generally, we write this fact as $f \overset{G}{\prec} g$.

Lemma 1. *For any quantum GSOS system $G_q = (\Sigma_{G_q}, R_{G_q})$ and for $T \equiv f(T_1, T_2, \dots, T_l) \in \mathbb{T}(\Sigma_{G_q})$, we have*

$$\mathrm{rea}(T) \subseteq \{g(Y_1, Y_2, \dots, Y_n) \,|\, f \overset{G}{\prec} g \wedge,$$
$$\forall_{i \in \{1,\dots,n\}} \exists_{j \in \{1,\dots,l\}} : Y_i \in \mathrm{rea}(T_j)\} \cup \bigcup_{i=1}^{l} \mathrm{rea}(T_i).$$

Proof. In this sketch of the proof, we firstly assume that $Q \in \mathrm{rea}(T)$. Then the definition of $\mathrm{rea}(\cdot)$ allows us to assume that $T \twoheadrightarrow Q$. The theorem can be proved by structural induction on the length of the derivation relation $T \twoheadrightarrow Q$. We wish to show that

$$Q \subseteq \{g(Y_1, Y_2, \dots, Y_n) \,|\, f \overset{G}{\prec} g \wedge,$$
$$\forall_{i \in \{1,\dots,n\}} \exists_{j \in \{1,\dots,l\}} : Y_i \in \mathrm{rea}(T_j)\} \cup \bigcup_{i=1}^{l} \mathrm{rea}(T_i).$$

First, we prove the basic case, where $T \equiv Q$. This case is trivial because the relation $T \twoheadrightarrow Q$ uses the operator $\overset{G}{\prec}$ which, by the definition of $\mathrm{rea}(\cdot)$, is reflexive for all closed terms $Y \in \mathbb{T}(\Sigma_{G_q})$. Therefore, $Y \in \mathrm{rea}(Y)$ by the definition of $\mathrm{rea}(\cdot)$.

The inductive step is used for $T \twoheadrightarrow Y \twoheadrightarrow Q$ for some $Y \in \mathbb{T}(\Sigma_{G_q})$. From the relation $T \twoheadrightarrow Y$ it is known that a rule $\rho_q \in R_{G_q}$ exists with f as the operation symbol in the form $T \equiv f(x_1, x_2, \dots, x_l)\sigma$ and the target $Y \equiv (x_i, y_j)\sigma$, where σ is a quantum substitution function. To compute the value of σ, we use the set of hypotheses for the rule ρ_Q.

If the target has the form x_i or y_{i_j}, then we compute $\sigma(x_i)$ or $\sigma(y_{i_j})$. The quantum substitution functions work on a Hilbert state space. For two given states there always exists at least one unitary transformation between them. This fact means that we directly obtain $Y \in \mathrm{rea}(T_i)$ for some i.

If the target has the form $g(g_1, g_2, \dots, g_n)$ for some $g \in \Sigma_{G_q}$, then we have to use the structural induction for $g(g_1, g_2, \dots, g_n)\sigma$. We obtain

$$\forall_{p \in \{1,\dots,n\}} \exists_{j \in \{1,\dots,l\}} : \sigma(z_p) \in \mathrm{rea}(T_j).$$

The inductive hypothesis applied to the relation $g(g_1, g_2, \dots, g_n) \twoheadrightarrow Q$ is omitted. ■

Direct application of Lemma 1 gives the following result.

Theorem 1. *Let $G_q = (\Sigma_{G_q}, R_{G_q})$ be a simple quantum GSOS system. Then for all closed terms $T \in \mathbb{T}(\Sigma_{G_q})$ the process graph of T is a finite graph.*

Proof. To prove this theorem, we show that $\mathrm{rea}(T)$ is finite for all closed terms $T \in \mathbb{T}(\Sigma_{G_q})$, and this fact can be proved by structural induction.

Firstly, we assume that $T \equiv f(T_1, T_2, \dots, T_l)$. Then the set $\mathrm{rea}(T_i)$ is finite for each $i \in \{1, \dots, l\}$. This assumption means the finiteness of each $\mathrm{rea}(T_i)$. Then we can show that $\mathrm{rea}(T)$ is finite, which can be easily shown using Lemma 1 because the set $\mathrm{rea}(T)$ contains the following subset:

$$\{g(Y_1, Y_2, \ldots, Y_n) \,|\, f \overset{G}{\prec} g \wedge,$$
$$\forall_{i \in 1,\ldots,n} \exists_{j \in \{1,\ldots,l\}} : Y_i \in \mathrm{rea}(T_j)\} \cup \bigcup_{i=1}^{l} \mathrm{rea}(T_i).$$

■

3. General quantum predicates as positive operator-valued measures

The notion of 0–1 valued predicates in the context of quantum systems was introduced by G. Birkhoff and J. von Neumann already in 1936 (Birkhoff and Neumann, 1936). A fundamental discovery made there was that the calculus of the introduced quantum predicates forms a structure slightly different from the orthomodular complete lattice known in the context of classical 0–1 predicates. The quantum predicate calculus forms a structure known as the quantum logic lattice. Gleason (1957) proved that the basic representation of the quantum lattice is that given by the algebra of orthogonal projectors acting in a Hilbert space and, moreover, such a representation is unique up to a unitary isomorphism provided that the dimension of the underlying space is greater than 2.

Since this discovery, a lot of work has been done in this area bearing the name of quantum logic theory. However, the application of quantum logic to quantum programming theory (QPT) is not widely discussed yet. In this section we present a very general notion of predicates applicable to the QPT area of research for finite quantum computation systems.

Let $\mathcal{H}$ be a Hilbert space and let $\mathcal{B}(\mathcal{H})$ stand for the C^*-algebra of bounded linear operators acting in $\mathcal{H}$. A map

$$\mathbb{F} : \beta(\mathbb{R}) \to \mathcal{B}(\mathcal{H}), \tag{11}$$

where $\beta(\mathbb{R})$ stands for the Borel σ-algebra of reals $\mathbb{R}$ will be called a positive operator-valued measure (POVM) iff

$$\text{(POVM1)} \quad \forall_{\psi \in \mathcal{H}} \quad A \in \beta(\mathbb{R}) \to \langle \psi | \mathbb{F}(A) \psi \rangle \geq 0, \tag{12}$$

i.e., the map $A \to \mathbb{F}(A)$ takes values in the space of semidefinite, nonnegative bounded operators acting on $\mathcal{H}$,

$$\text{(POVM2)} \quad \forall_{\psi \in \mathcal{H}} \quad \begin{array}{l} A \in \beta(\mathbb{R}) \to \langle \psi | \mathbb{F}(A) \psi \rangle \\ \text{is } \sigma\text{-additive on } \beta(\mathbb{R}). \end{array} \tag{13}$$

>From (POVM1) it follows that $\mathbb{F}(A)$, which will be also denoted as

$$\int \chi_A(x) \cdot \mathrm{d}F(x),$$

where $\chi_A(x)$ is the characteristic function of the set A, for any $A \in \beta(\mathbb{R})$, is a Hermitian nonnegative operator acting on $\mathcal{H}$. It is a standard assumption (although there are some important situations where this is not true) that the measure $\mathrm{d}F$ is atomic, i.e., that its support $\mathrm{supp}(\mathrm{d}F)$ is a discrete subset of $\mathbb{R}$: $\mathrm{supp}(\mathrm{d}F) = \{x_1, \ldots, x_n\}$, and then we will write

$$\mathbb{F}(\{x_i\}) = F_i$$

and also

$$\int_{\mathbb{R}} \mathrm{d}F(x) = F_1 + F_2 + \ldots + F_n.$$

Definition 8. The space of generalized predicates $\mathrm{gPre}(\mathcal{H})$ is formed by all $\mathrm{POVM}(\mathcal{H})$ obeying additionally the condition

$$\mathbb{F}(\mathbb{R}) = \int_{\mathbb{R}} \mathrm{d}F(x) \leq \mathbb{I}. \tag{14}$$

A natural partial order relation $\preceq$ is defined in $\mathrm{gPre}(\mathcal{H})$. We will write $\mathbb{F}_1 \preceq \mathbb{F}_2$ iff

$$\forall_{A \in \beta(\mathbb{R})} \forall_{\psi \in \mathcal{H}} \quad \langle \psi | F_1(A) \psi \rangle \leq \langle \psi | F_2(A) \psi \rangle. \tag{15}$$

Proposition 3. *The space* $\mathrm{gPre}(\mathcal{H})$ *is a complete partially ordered space (cpos).*

Proof. Let $(\mathbb{F}_n)_{n=1,2,\ldots}$ be an ordered chain of POVMs defined on $\mathcal{H}$ and obeying (14). We define the least upper bound of $(\mathbb{F}_n)_n$, denoted by $\mathbb{F}^*$, as follows:

$$\forall_{\psi, A} \langle \psi | \mathbb{F}^*(A) \psi \rangle = \limsup_n \langle \psi | \mathbb{F}_n(A) \psi \rangle.$$

>From the polarisation identities it follows that for any $A \in \beta(\mathbb{R})$ the operator $\mathbb{F}^*(A)$ is well defined and is nonnegative and bounded, where (14) is also taken into account.

The proof of σ-additivity is based on a version of the dominated convergence theorem which is used in the following way: Let $A = \bigcup_{n=1}^{\infty} A_n$, $A_n \in \beta(\mathbb{R})$ and $A_n \cap A_{n'} = \emptyset$ for $n \neq n'$. Then, for any n and $|\psi\rangle \in \mathcal{H}$,

$$\langle \psi | \mathbb{F}_n(A_k) \psi \rangle = \sum_{k=1}^{\infty} \langle \psi | \mathbb{F}_n(A_k) \psi \rangle$$

from (POVM2). The dominated convergence theorem for positive series gives

$$\begin{aligned} \limsup_n \langle \psi | \mathbb{F}_n(A) \psi \rangle & \\ &= \langle \psi | \mathbb{F}^*(A) \psi \rangle \\ &= \sum_{k=1}^{\infty} \limsup_n \langle \psi | \mathbb{F}_n(A_k) \psi \rangle \\ &= \sum_{k=1}^{\infty} \langle \psi | \mathbb{F}^*(A_k) \psi \rangle. \quad ■ \end{aligned} \tag{16}$$

Remark 3. In most applications, $\mathrm{POVM}(\mathbb{F})$ describing a specific measurement process connected to observable F has a discrete structure. However, this property is not stable under taking the least upper bound and that is why we have to consider the space of the POVM as above.

The important notion of the relative quantum phase seems to be an excellent example of an observable for which the measurement process is based on the POVM with continuous support. When we recall the role played by the relative phase notion in the Shor algorithm or even the fact that very important quantum calculation schemes (known as geometrical and topological calculations considered to be very promising for future quantum computer implementation) do exist in which the notion of the relative quantum phase plays a major role, we can be sure that the introduced notion of the predicate will find important applications in the semantic analysis of these sorts of quantum calculations.

Let the quantum system considered be in a state ρ and let $\mathrm{POVM}(\mathbb{F})$ correspond to an observable F. Then a statistical interpretation of $\mathbb{F}$ is that for any $A \in \beta(\mathbb{R})$ the quantity

$$\langle F(A)\rangle_\rho = \mathrm{Tr}\,(\rho F(A)) = \mathrm{Tr}\left(\int \chi_A(x)\,\mathrm{d}\mathbb{F}(x)\,\rho\right)$$

assigns a probability to the event that measuring F we obtain a value belonging to the set A.

Remark 4. Although standard observables (for infinite-dimensional systems) associated with self-adjoint operators whose continuous spectra are nonempty do exist, the main difference between the classical projective measurement and the measurement connected with $\mathrm{POVM}(\mathbb{F})$ is that in the process of measuring $\mathbb{F}$ no collapsing process takes place.

Another important difference is that in the case of discrete and finite POVM $\mathbb{F}$ with

$$\int_{\mathbb{R}} \mathrm{d}\mathbb{F}(x) = F_1 + F_2 + \ldots + F_n,$$

the supporting operators do not obey the orthogonality relation $F_iF_j = 0$ for $i \neq j$ in general, which is in contrast to the standard projective-type measurement. More information about the measurement can be found, e.g., in (Sewell, 2005; Peres, 1995).

3.1. Predicates as positive operators.

In most applications, corresponding $\mathrm{POVM}(\mathbb{F})$ for measuring the observable $\mathbb{F}$ is discrete, i.e., the support of $\mathbb{F}$ is a finite set and then

$$\int_{\mathbb{R}} \mathrm{d}\mathbb{F}(x) = F_1 + F_2 + \ldots + F_n, \tag{17}$$

where $F_i \geq 0$ for $i \geq 1, \ldots, n$ and, moreover, $\sum_{i=1}^n F_i \leq \mathbb{I}$. If this sum equals the unit operator $\mathbb{I}$, then the measurement corresponding to F is called complete and in this case we have a natural probabilistic interpretation: If the system is in the state ρ then the probability that, while measuring the quantity F, the result of the measurement will be connected to a particular F_i from the decomposition (17) is given by

$$p_i = \mathrm{Tr}\,(\rho F_i)\,. \tag{18}$$

Denote by $F_N(\mathrm{POVM}(\mathcal{H}_M))$ the set of POVMs which have supports consisting of at most N atoms. Then the partial order $\preceq$ as in the previous subsection can be introduced into $F_N(\mathrm{POVM}(\mathcal{H}_M))$ and the resulting pos is cpos for any fixed N.

The class of predicates on which our discussion will be focused will be that corresponding to the space

$$\mathrm{DPre}(\mathcal{H}_M) = \bigcup_{N\in\mathbb{N}} F_N(\mathrm{POVM}(\mathcal{H}_M)).$$

Definition 9. For a predicate $\mathbb{F} \in \mathrm{DPre}(\mathcal{H}_M)$ the satisfability of $\mathbb{F}$ is defined as the map

$$\begin{aligned}\mathrm{sat}_{\mathbb{F}} : \beta(\Sigma) \times E(\mathcal{H}) \ni (A,\rho) &\to \mathrm{sat}_{\mathbb{F}}(A,\rho)\\ &= \mathrm{Tr}\,(F(A)\rho)\,.\end{aligned}$$

In particular, for a fixed $\rho \in E(\mathcal{H}_M)$, the number $\mathrm{Tr}\,(\rho\mathbb{F})$ represents a degree of satisfying the predicate $\mathbb{F}$ by a system being actually in the state ρ. In particular, if $\mathrm{Tr}\,(\rho\mathbb{F}) = 0$, we say that the predicate $\mathbb{F}$ is not fulfilled by the state ρ. The following lemma (see also (Raynal, 2006)) seems to be useful in order to explain what happens in this case.

Lemma 2. *For any positive operators A and B, $\mathrm{Tr}(AB) = 0$ if and only if the bases of these operators are orthogonal:*

$$\mathrm{Tr}(AB) = 0 \;\Leftrightarrow\; \{|\psi_i^A\rangle\} \perp \{|\psi_i^B\rangle\}. \tag{19}$$

Proof. >From the Hermitian property of A and B and the spectral theorem it follows that there exist orthonormal systems of vectors $\{|\psi_i^A\rangle\} \subset \mathcal{H}_M$, $\{|\psi_j^B\rangle\} \subset \mathcal{H}_M$ being the corresponding eigenvectors for A and B in which the operators A and B are represented by

$$A = \sum \lambda_i^A |\psi_i^A\rangle, \quad B = \sum \lambda_j^B |\psi_j^B\rangle$$

with $\lambda_i^A > 0, \lambda_j^B > 0$. Both systems $\{|\psi_i^A\rangle\}, \{|\psi_j^B\rangle\}$ could be completed to orthonormal bases of $\mathcal{H}_M$, and the point is that these complementary vectors must belong to the kernels of A and B. Recall that the bases of A and B are given by

$$b(A) = \mathscr{L}_H\{|\psi_i^A\rangle\}$$

and

$$b(B) = \mathscr{L}_H\{|\psi_j^B\rangle\},$$

respectively, where $\mathscr{L}_H$ means the operation of taking the linear hull.

Using these bases, the trace of AB can be computed:

$$\begin{aligned}&\mathrm{Tr}_{\mathcal{H}_M}(AB)\\&\quad= \mathrm{Tr}_{\mathcal{H}_M}\left(\sum_i \lambda_i^A |\psi_i^A\rangle\langle\psi_i^A| \sum_j \lambda_j^B |\psi_j^B\rangle\langle\psi_j^B|\right)\\&\quad= \sum_{ij} \lambda_i^A \lambda_j^B |\langle\psi_i^A|\psi_j^B\rangle|^2. \qquad (20)\end{aligned}$$

>From this identity it is easy to conclude that $\mathrm{Tr}_{\mathcal{H}_M}(AB) = 0$ iff $b(A) \perp b(B)$. ■

The application of Lemma 2 allows us to derive the notion of unambiguous predicates.

Definition 10. Two predicates $\mathbb{F}_1, \mathbb{F}_2 \in \mathrm{DPre}(\mathcal{H}_M)$ are *unambiguous* iff

$$\forall_{A\in\beta(R)}\ \mathrm{Tr}(\mathbb{F}_1(A)\mathbb{F}_2(A)) = 0,$$

which will be denoted by $\mathbb{F}_1 \perp \mathbb{F}_2$.

This means that the predicates F_1 and F_2 give us information according to the orthogonal subspaces which are pointed by one of them.

Theorem 2. *For any state $\rho \in E(\mathcal{H}_M)$ there exists at least one predicate $\mathbb{F} \in \mathrm{DPre}(\mathcal{H}_M)$ such that*

$$\mathrm{sat}_{\mathbb{F}}(A, \rho) > 0$$

for some $A \in \beta(\Sigma)$.

Proof. Let $\mathcal{H}_A$ be a a nontrivial closed subspace of $\mathcal{H}_M$. Then we can decompose $\mathcal{H}_M = \mathcal{H}_A \oplus \mathcal{H}_A^\perp$. Let E_A be the orthogonal projector projecting $\mathcal{H}_M$ onto $\mathcal{H}_A$, i.e., $E_A : \mathcal{H}_M \to \mathcal{H}_A$.

Taking a normalised vector $|e\rangle \in \mathcal{H}_M$ and applying E_A, we can write

$$|e\rangle = |e_A\rangle + |e_A^\perp\rangle,$$

where

$$|e_A\rangle = E_A|e\rangle, \quad |e_A^\perp\rangle = (1 - E_A)|e\rangle.$$

It is clear that $|e_A\rangle \in \mathcal{H}_A$ and $|e_A^\perp\rangle \in \mathcal{H}_A^\perp$. Assume that $|e_A\rangle \neq |0\rangle$.

Let us choose a state $\rho \in E(\mathcal{H}_M)$ such that the basis of ρ_A is equal to $\mathcal{H}_A$. Let us assume that our measurement is of a projective type and consists in measuring the 0–1 quantum predicate composed from the unique projector $P_a = |e_A\rangle\langle e_A|$. Provided the system M is in the state ρ_A, actually the result of measuring the projective predicate P_a is equal to 1 with probability

$$\mathrm{Tr}(\rho P_a) = \mathrm{Tr}(\rho P_a^2) = \mathrm{Tr}(P_a \rho P_a) > 0,$$

and thus

$$\mathrm{sat}_{\{P_a\}}(\rho_A) = \mathrm{Tr}(P_a \rho P_a) > 0$$

from the assumption that $|e\rangle \notin \mathcal{H}_A^\perp$. ■

3.2. Weakest precondition using the Kraus representation.

The Kraus representation (Kraus, 1983) will play a crucial role in our generalised quantum weakest precondition result. Therefore, we recall the following theorem known as the Kraus theorem (or the operator-sum representation).

Theorem 3. *Let the dimension $\dim \mathcal{H}_n$ of the space be equal to $n < \infty$. Then, for any completely positive map $\mathcal{E}$ on $L(\mathcal{H}_n)$ there exists a family of linear endomorphims $(F_i)_{i=1,\ldots,n^2}$ such that for any $\rho \in E(\mathcal{H}_M)$ we can write*

$$\mathcal{E}(\rho) = \sum_i F_i \rho F_i^\dagger \qquad (21)$$

and, moreover, if $\mathcal{E}$ is trace-preserving, then also

$$\sum_i F_i^\dagger F_i = \mathbb{I}.$$

The proof of this theorem can be found in many papers, e.g., a detailed exposition of the Kraus theorem can be found in the paper (Choi, 1975).

Let $\mathbb{F} \in \mathrm{DPre}(\mathcal{H}, \Sigma)$ be a POVM and such that

$$\forall_{A\in\beta(\Sigma)} \int_A \mathrm{d}F(x) = \sum_{\alpha\in A} F_\alpha,$$

and let $\mathcal{E}$ be a CP map on $\mathcal{H}_M$. We want to construct a $\mathbb{G} \in \mathrm{DPre}(\mathcal{H}_M, \Sigma)$ obeying (22) and being largest in the sense of the $\preceq$ ordering, i.e.,

$$\forall_{A\in\beta(\Sigma)} \mathrm{sat}_{\mathbb{G}}(A, \rho) \leq \mathrm{sat}_{\mathbb{F}}(A, \mathcal{E}(\rho)). \qquad (22)$$

With the help of Theorem 3 we have

$$\begin{aligned}&\mathrm{Tr}\,(F(A)\mathcal{E}(\rho))\\&\quad= \mathrm{Tr}\left(F(A)\left(\sum_i F_i \rho F_i^\dagger\right)\right)\\&\quad= \sum_i \mathrm{Tr}\left(F_i^\dagger F(A) F_i \rho\right)\\&\quad= \mathrm{Tr}\,((\mathcal{E}^\star F)(A)\rho)\,.\end{aligned}$$

Thus we conclude that

$$\mathrm{WP}(\mathbb{F}, \mathcal{E}) = \mathcal{E}^\star \mathbb{F}, \qquad (23)$$

where $\mathcal{E}^\star$ corresponds to the $\mathcal{E}$ quantum channel.

Theorem 4. *Let $\dim \mathcal{H}_n < \infty$. Then for any discrete $\mathbb{F} \in \mathrm{DPre}(\mathcal{H}_M)$ and any quantum program $\mathcal{E}$ (= any completely positive endormorphism of $L(\mathcal{H}_M)$) there exists a unique $\mathbb{G} \in \mathrm{DPre}(\mathcal{H}_M)$ obeying the weakest precondtion postulate for a pair $(\mathbb{F}, \mathcal{E})$. Moreover, the predicate $\mathbb{G}$ can be calculated by the action of the quantum channel $\mathcal{E}^\star$ on $\mathrm{DPre}(\mathcal{H}_M)$.*

Remark 5. It the case where the support of $\mathbb{F}$ consists of only one atom, the corresponding quantum predicate $\mathcal{E}^{\star}\mathbb{F}$ is the same as that constructed in (D'Hondt and Panangaden, 2006). A generalisation of this result to the infinite dimension setting and a continuously supported POVM is presented elsewhere (Gielerak and Sawerwain, 2007).

3.3. Postcondition and its duality with the weakest precondition.

Let $\mathbb{M} \in \mathrm{DPre}(\mathcal{H}, \Sigma)$ be a predicate defined as a POVM. Then the satisfability can be written as

$$\begin{aligned}\forall_{A\in\beta(\Sigma)}\mathrm{sat}_{\mathbb{M}}(A,\rho) &= \mathrm{Tr}\left(M(A)\rho\right)\\ &= \mathrm{Tr}\left(\left(\sum_i F_i M(A) F_i^{\dagger}\right)\rho\right)\\ &= \mathrm{Tr}\left(\left(\sum_i F_i^{\dagger}\rho F_i\right)M(A)\right)\\ &= \mathrm{Tr}\left(\mathcal{E}(\rho)M(A)\right)\\ &= \forall_{A\in\beta(\Sigma)}\mathrm{sat}_{\mathbb{M}}(A,\mathcal{E}(\rho)),\end{aligned}$$

where $\mathcal{E}(\cdot)$ represents the transformation of the entry state ρ. In other words, $\mathcal{E}$ is a completely positive quantum program executed on the state ρ, and the operator $M\mathcal{E}(\cdot)$ is called the strongest postcondition.

This means that the satisfabilities for the entry and the final state with appropriate predicates for completely positive quantum programs are always positive,

$$\forall_{A\in\beta(\Sigma)} \quad \mathrm{sat}_{\mathbb{M}}(A,\mathcal{E}(\rho)) > 0 \Rightarrow \mathrm{sat}_{\mathbb{M}}(A,\rho) > 0. \tag{24}$$

This duality can be written as the following inference rule:

$$\frac{\mathcal{E}(\rho) \models \mathbb{M}}{\rho \models \mathrm{WP}(\mathbb{M},\mathcal{E})}. \tag{25}$$

3.4. Projective predicates for a quantum labelled transition system.

The notion of quantum labelled transition systems can be equipped with the predicate notion. However, in order to introduce this notion in a way similar to the classical situation, we must introduce a partial order for the quantum state. This problem was extensively and fruitfully discussed in (Coecke and Martin, 2002). This approach to the notion quantum predicate calculus should be considered as a special case of POVM based predicates presented in Section 3.

For a proper definition of the partial order for density matrices and for operators which transform these matrices, we can use the following well-known proposition (for another example of this proof, see (D'Hondt and Panangaden, 2006)):

Proposition 4. *For a Hermitian operator O and a density operator ρ we have that $\mathrm{Tr}(O\rho) \in \langle 0, 1\rangle$ iff O is positive and the eigenvalues of O are bounded by one.*

Proof. For any state vector $|\psi\rangle \in \mathcal{H}$ it is known that $\mathrm{Tr}(O|\psi\rangle\langle\psi|) = \langle\psi|O|\psi\rangle$. If we assume that $\mathrm{Tr}(O\rho) \in \langle 0, 1\rangle$ for any density operator ρ, where $\rho = |\psi\rangle\langle\psi|$, then we obtain $0 \le \langle\psi|O|\psi\rangle$ because the operator O is positive. On the other hand, it is known that $O\psi = \lambda\psi$, and therefore $\mathrm{Tr}(O|\psi\rangle\langle\psi|) = \langle\psi|O|\psi\rangle = \lambda\langle\psi|\psi\rangle$ and $\lambda \le 1$. ■

We define the partial order for the matrices as follows: Let D_n be the set of density matrices on the n-dimensional Hilbert space denoted by $\mathcal{H}^n$:

$$D_n = \{\rho \in \mathbb{C}^n \mid \rho \ge 0 \text{ and } \mathrm{Tr}(\rho) = 1\}.$$

The partial order of complex matrices (known as the Löwner partial order on the matrices (Löwner, 1934)) is defined as

$$A, B \in \mathbb{C}^{n\times n}, A \preceq B, \text{iff } B - A \text{ is positive.}$$

Unfortunately, there exist many examples for basic states for which the partial order on matrices fails, which confirms the fact that the introduced order is only a partial order on the space ρ. For example, consider the following density matrix:

$$L = \left(\begin{array}{c|c} A & B \\ \hline C & D\end{array}\right),$$

where $B \ne 0$ and $C \ne 0$. It is trivial that the zero element in the matrix partial order precedes other matrices,

$$\left(\begin{array}{c|c} 0 & 0 \\ \hline 0 & 0\end{array}\right) \preceq \left(\begin{array}{c|c} A & 0 \\ \hline 0 & 0\end{array}\right) \preceq \left(\begin{array}{c|c} A & 0 \\ \hline 0 & D\end{array}\right).$$

It is easy to find matrices which do not fulfil the partial order, e.g.,

$$\left(\begin{array}{c|c} A & 0 \\ \hline 0 & 0\end{array}\right) \not\preceq \left(\begin{array}{c|c} A & B \\ \hline C & D\end{array}\right)$$

or

$$\left(\begin{array}{c|c} A & 0 \\ \hline 0 & D\end{array}\right) \not\preceq \left(\begin{array}{c|c} A & B \\ \hline C & D\end{array}\right).$$

In fact, it is rather well known that no linear order on the spaces $\mathbb{C}^n$ exists for any $d \ge 1$.

Therefore, it is necessary to formulate a predicate in greater detail. The spectral order with a projection operator on a selected subspace allows us to give a more precise definition of predicates. Every self-adjoint linear operator can be decomposed by the spectral theorem:

Theorem 5. *A given self-adjoint linear operator $\rho \in D_n$ decomposes in a unique way into a linear combination of mutually orthogonal projections*

$$\rho = \sum_{\lambda\in\mathrm{spec}(\rho)} \lambda P_e^{\lambda},$$

where

$$P_e^{\lambda} P_e^{\lambda'} = \delta_{\lambda\lambda'} P_e^{\lambda} \quad , \quad \sum_{\lambda} P_e^{\lambda} = \mathbb{I}_{\mathcal{H}}.$$

The set $\mathrm{spec}(\rho) \subset \mathbb{R}$ is called the spectrum of the operator ρ and P_e^{λ} represents the projector associated with the eigenvalue λ.

An example of the spectral decomposition of a linear operator by projectors is depicted in Fig. 1. We obtain

$$\mathrm{prob}_e^{\lambda}(\rho) = \mathrm{Tr}(P_e^{\lambda}\rho). \tag{26}$$

To formalise this operation, we introduce the enumeration of the closed subspaces of the Hilbert space H^n associated with the selected quantum labelled transition system.

It is hard to decide whether A and B are comparable with respect to the relation $\preceq$ using only the knownledge about the spectra of A and B. However, there exists a nice result proven in (Coecke and Martin, 2002) that makes it possible. For this purpose, let us define the notion of labelling.

Definition 11. A *labelling* is a spectral decomposition map

$$e : \{1, \ldots, n\} \to \mathbb{L}(\mathcal{H}^n),$$

where

$$e_i \perp e_j \quad \text{for} \quad i \neq j$$

and

$$\oplus_i e_i = \mathcal{H}^n.$$

$\mathbb{L}(\mathcal{H}^n)$ means the variety of all subspaces of $\mathcal{H}$ where we assumed that $\dim \mathcal{H} < \infty$.

For given $\rho \in E(\mathcal{H})$ there exists a unique classical state and the corresponding labelling e such that $[\rho, e] = 0$ and $\mathrm{spec}(\rho_{\restriction e_i}) = x_i$. A state σ will be called the predicate for ρ iff σ is comparable with ρ and, moreover, $\sigma \preceq \rho$. In terms of the labelling e this means that $\mathrm{spec}(\sigma_{\restriction e_i}) \leq \mathrm{spec}(\rho_{\restriction e_i})$. For more details, we refer the reader to the paper (Coecke and Martin, 2002).

If we consider the predicates as projectors onto the selected subspace, then we have the following definition, where we let the eigenvalues be bounded by one. For this purpose, let us combine the partial order on the density matrices and Proposition 4.

Definition 12. A *simple predicate* f is a positive Hermitian operator with the maximum eigenvalue equal to 1.

The space of simple predicates on $\mathcal{H}$ will be denoted by $\mathrm{SPre}(\mathcal{H})$. In the set of simple predicates, a complete partial order can be formed by applying the relation $\preceq$ as above.

Proposition 5. *The partial order of predicates* $(\mathrm{SPre}(\mathcal{H}), \preceq)$ *is a complete partial order, and it has least upper bounds of increasing sequences.*

3.4.1. Satisfability for projective predicates. The satisfaction relation introduced in this part of the article can generally be written as

$$\models \subseteq E(\mathcal{H}) \times \mathrm{SPre}(\mathcal{H}). \tag{27}$$

Let ρ be a density matrix and F be a projector on a selected subspace of $\mathcal{H}$. We say that the state ρ fulfils the predicate F if and only if

$$0 < \mathrm{Tr}(F\rho) \leq 1, \tag{28}$$

and that the state ρ does not fulfil the predicate F iff

$$\mathrm{Tr}(F\rho) = 0. \tag{29}$$

In many cases this relation is too general and the introduction of some kind of threshold is necessary. For a given threshold $0 < \alpha \leq 1$, the satisfability $\models_{\alpha} E(\mathcal{H}) \times \mathrm{SPre}(\mathcal{H})$ can be rewritten as

$$\rho \models_{\alpha} F \quad \text{iff} \quad \alpha < \mathrm{Tr}(F\rho) \leq 1. \tag{30}$$

The presented interpretation of satisfability for quantum predicates allows us to derive several definitions connected with the notion of predicates. The general relation between two states can be defined as follows:

Definition 13. Let ρ and σ be quantum states. Then $\mathcal{R}$ represents a relation between ρ and σ if there exists a predicate F such that

$$F\sigma - F\rho > 0.$$

The quantum assertion for quantum states and quantum programs can be expressed in the following way:

Definition 14. Let ρ, σ be given quantum states and S represent a quantum program. Then the state ρ is called the entry state and the σ state is the final state after the execution of the program S: $\rho \overset{S}{\twoheadrightarrow} \sigma$. The form $\{f_0\}S\{f_1\}$ is a quantum assertion if and only if $f_0\rho > 0$ and $f_1\sigma > 0$.

Based on the quantum predicate notion, the invariant predicate for quantum programs can be formulated in the following form:

Proposition 6. *The predicate f is invariant for quantum instruction i in state ρ in form $\{f\}i\rho\{f\}$ if and only if $f\rho > 0$ and $fi\rho > 0$.*

Proof. It is a direct consequence of the fact that the state ρ before and after the application of instructions i contains information in the subspace pointed by the predicate f. This fact is depicted in Fig. 2. ■

The notion of invariants for an instruction can be easily extended to the notion of a general invariant for quantum programs, where the invariant is a guard for a selected subspace where the quantum program is executed. This

$$\begin{pmatrix} 0 & & & & & & 0 \\ & \ddots & & & & & \\ & & 0 & & & & \\ & & & 1 & & & \\ & & & & 0 & & \\ & & & & & \ddots & \\ 0 & & & & & & 0 \end{pmatrix} \begin{pmatrix} \langle r|e_1\rangle & & & ? & \\ & \ddots & & & \\ & & \langle r|e_i\rangle & & \\ & & & \ddots & \\ & ? & & & \langle r|e_n\rangle \end{pmatrix} = \begin{pmatrix} 0 & & & ? & & & \\ & \ddots & & \vdots & & 0 & \\ & & 0 & ? & & & \\ ? & \dots & ? & \langle r|e_i\rangle & ? & \dots & ? \\ & & & ? & 0 & & \\ & 0 & & \vdots & & \ddots & \\ & & & ? & & & 0 \end{pmatrix}.$$

Fig. 1. Decomposition of the density matrix where the projector operator allows us to obtain a selected probability.

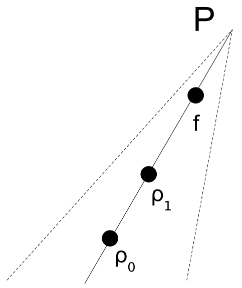

Fig. 2. In the subspace P an invariant predicate f exists and the states ρ_0, ρ_1 fulfil this predicate.

situation must be formulated by the following theorem, which is a direct counterpart of the classical invariant for general programs:

Theorem 6. *Let f be a predicate for a quantum program S that consists of k computational steps. Then f is invariant for S if and only if $fS_i\rho > 0$ for each $i = \{1, 2, \ldots, k\}$.*

Proof. Let P represent the corresponding subspace for the quantum program S. Then we assume that f is a projector on P. If after the execution of some number of instructions from S the state ρ does not fulfil f, then some instruction S_i changes the state ρ and $S_i\rho$ does not belong to the subspace P. In other words, the program S is executed in a different subspace than assumed. ■

4. Application of quantum predicate calculus to an analysis of some quantum algorithms

First, we show a simple operational description for a quantum programming language which is similar to two known quantum programming languges: QCL (Ömer, 2005) and LanQ (Mlnařík, 2006). Then examples representing the introduced notion of quantum labelled systems with predicates will be presented. The first concerns the analysis of the Grover algorithm and the second the superdense coding protocol. We also present the operational proof tree for simulating the β-step using the μ-step in the 1WQC model.

```
var
    q : q16; { declaration of
               quantum variable }
begin
    { reset quantum variable }
    Reset(q);

{ assign 2^16 classical
  states to variable q }
    H(q);

{ show result of measurement
  of q on the screen }

    writeln('q=', q);
end.
```

Fig. 3. Trivial example of the quantum random number generator.

4.1. Operational description for a simple quantum programming language. In this section we define a simple programming language with quantum data types. The language is intentionally similar to Pascal, and therefore we call it Quantum Pascal.

The first simple program depicted in Fig. 3 is an example of the true random number generator based on the quantum mechanics measurement operator.

The first line includes the instruction `Reset(q);`. In other words, the quantum variable is initialised with a state 0,

$$\frac{}{|q\rangle \overset{\text{reset}}{\twoheadrightarrow} |0\rangle_\perp}.$$

The second statement `H(q);` simultaneously initializes

```
var q : qint;

{ oracle procedure computing function f }
procedure fnc(var q : qint);
{ omitted oracle implementation }

begin
    Reset(q); q:=(1);
    q:=Had(q); fnc(q); q:=Had(q);
    writeln('q=',q);
end;
```

Fig. 4. Prototype DJ problem solution written in Quantum Pascal.

the quantum variable q with 2^{16} classical states. In operational semantics, this instruction corresponds to the Hadamard gate applied to a quantum register,

$$\frac{}{|0\rangle_{\perp} \overset{\mathrm{H}}{\twoheadrightarrow} |\psi'\rangle}.$$

Finally, results are displayed with the use of the $writeln$ instruction, which implies a measurement of the quantum variable state,

$$\frac{}{|\psi'\rangle \overset{\mu}{\twoheadrightarrow} |\psi\rangle_{\perp}}.$$

The next example depicted in Fig. 4 is a program solving the Deutch-Jozsa problem (Deutsch and Jozsa, 1992).

Now, we try to formulate the operational rules for our quantum programming language.

Definition 15. A quantum program P is a pair $\langle I, Q\rangle$ where Q is a finite set of quantum variables and I is a sequence of commands (instructions) from the following list:

$$\begin{array}{rcl} C & ::= & skip \\ & | & C_1 \ ; \ C_2 \\ & | & q := q + \mathbb{N} \\ & | & \text{if q then } \mathrm{C}_1 \text{ else } C_2 \\ & | & X(q) \mid Y(q) \mid Z(q) \mid H(q) \mid \ldots \\ & | & \mathrm{CNot}(q_1, q_2) \mid \mathrm{CHad}(q_1, q_2) \mid \ldots \\ & | & \mathrm{Measure}(q_1, q_2, \ldots, q_n). \end{array}$$

Let L represent basic instructions belonging to the program P written in Quantum Pascal:

- the empty instruction denoted by words **skip**, **empty** or by a semicolon,
- the assign statement: $a := a + 2$,
- the deterministic “if” instruction: if $s_0 > 3$ then s_1,
- the function “call” (included system function): $fnc(a)$,
- the measurement procedure applicable to a quantum variable,
- the sequence of two sets of instructions: $s_1; s_2$.

Remark 6. Additional loop constructions can be added to the above list. The bounded loop $repeat$ has the following form:

$$\text{repeat } n \text{ do } s.$$

The value n is the number of iterations of the instruction s. This instruction can be easily decomposed into the list of basic instructions. It is possible to introduce a typical $while$ loop

$$\text{while } test \text{ do } s,$$

where the measuring process examines the expression $test$. This construct is easy to built if we assume that the expression $test$ is built from pure states of eigenstates of observables used in the measurement process.

Definition 16. For the language L there exists the following operational description L_o:

- The empty instruction has the rule

  $$\frac{}{|\psi\rangle \overset{I}{\twoheadrightarrow} |\psi\rangle},$$

  where I represents the identity matrix.

- For the assign statement, we define the transition

  $$\frac{}{|a\rangle \overset{U}{\twoheadrightarrow} |a'\rangle}.$$

  This rule accepts instructions in the following form:

  $$a := a \ \Omega \ c,$$

  where c is the constant or a classical expression and Ω represents the function: $\mathbb{N}^n \to \mathbb{N}^n$. Generally, the assign instruction in this form can be implemented as the permutation matrix which can be replaced by the appropriate set of CNOT gates.

- The deterministic selection instruction defined by the following inference rule:

  $$\frac{}{|q_c q_o\rangle \overset{U(q_o)_{q_c=111\ldots}}{\twoheadrightarrow} |q_c q_o'\rangle}.$$

  The notation $U(q_o)_{q_c=111\ldots}$ means that the operator U is applied to a qubit or qubits denoted by q_o at the state denoted by q_c. This is identical with the CNOT or Toffoli gate definition, which applies the NOT operation to a subspace where the first qubit is in the state 1.

- The function "call" understood as an application of the U operation, i.e., Hadamard or any other unitary gate:
$$\frac{}{|\psi\rangle \overset{U}{\twoheadrightarrow} |\psi'\rangle}.$$
- The measurement of states equal to the eigenstate of the observable used:
$$\frac{}{|\psi\rangle_\perp \overset{\mu}{\twoheadrightarrow} |\psi\rangle_\perp}.$$
- The sequence $s_1 \;;\; s_2$ is defined by two recursive rules. The first rule describes the case when the set s_1 represents an empty instruction:
$$\frac{\langle s_1, |\psi\rangle\rangle \overset{I}{\twoheadrightarrow} \langle \text{empty}, |\psi\rangle\rangle}{\langle s_1 s_2, |\psi\rangle\rangle \overset{U}{\twoheadrightarrow} \langle s_2, |\psi\rangle\rangle},$$
and a rule of the second case in the form
$$\frac{\langle s_1, |\psi\rangle\rangle \overset{U}{\twoheadrightarrow} \langle s_1, |\psi^{'}\rangle\rangle}{\langle s_1; s_2, |\psi\rangle\rangle \overset{U}{\twoheadrightarrow} \langle s_2, |\psi^{'}\rangle\rangle}.$$

Now, it can be proved that the language L terminates, i.e., programs built from the mentioned instructions are finite.

Proposition 7. *The language L defined by the operational semantics L_o terminates.*

Proof. To prove that the language L terminates we define a complexity function $\mathrm{Len}(p) \to \mathbb{N}$ defined by the following expressions:

$$\begin{aligned}
\mathrm{Len}(\text{empty}) &= 0,\\
\mathrm{Len}(\text{assign}) &= 1,\\
\mathrm{Len}(\text{if } b \text{ then } s_1) &= s_1 + 1,\\
\mathrm{Len}(\text{call fnc}) &= \mathrm{Len}(\text{definition of fnc}),\\
\mathrm{Len}(\text{measure}) &= 1,\\
\mathrm{Len}(s_1; s_2) &= \mathrm{Len}(s_1) + \mathrm{Len}(s_2).
\end{aligned}$$

The function $Len(p)$ returns a natural number equal to the maximum length of the transition in the program p. Because the set of natural numbers is well founded we may assume that the language L terminates. ■

Since we use a very special case of measurement, it can be also proved that the language L is deterministic.

Proposition 8. *The language L defined by the operational semantics L_o is deterministic.*

Proof. First, we have to prove that for a state $|\psi\rangle$ in each step of the computational process a program written in the language L generates only one transition:

$$\forall_{i\in\mathcal{L}} !\exists_U \langle i, |\psi\rangle\rangle \overset{U}{\twoheadrightarrow} \langle i, |\psi^{'}\rangle\rangle.$$

We prove this property by the structural induction. Two cases are considered:

- The case of the first five instructions in the definition of the language L. It is obvious that all those rules have at most one transition.
- Let $i = s_1; s_2$. In this case semantics are denoted by the following set:
$$\begin{aligned} L_o(s_1; s_2) =&\{\langle s_2, |\psi\rangle\rangle : \langle \text{empty}, |\psi\rangle\rangle \in L_o\} \cup\\ &\{\langle s_1; s_2, |\psi\rangle\rangle : \langle s_1, |\psi^{'}\rangle\rangle \in L_o\}.\end{aligned}$$
By the structural induction, rules generated by the operational function have at most one element. This element can be an empty instruction or any other instruction defined by the semantics L_o .

In this way we proved that the language L has always at most one transition in every rule. ■

4.2. Operational tree proof for a quantum teleportation protocol.

The teleportation protocol first presented in (Bennett *et al.*, 1993) (a physical realisation described in (Boschi *et al.*, 1998; Bouwmeester *et al.*, 1997)) is a good example of an algorithm for which a proof tree graph built over three qubits can be easily constructed. Let the teleported qubit be represented by t, the qubit A be Alice's qubit and the qubit B be Bob's qubit. Using the teleportation protocol we want to transfer the state of t on to Bob's qubit B. The teleportation protocol has the proof tree depicted in Fig. 5.

The proof tree clearly shows nonlinearity (the process of computation needs results from earlier computation steps) of the teleportation algorithm introduced by (Bennett *et al.*, 1993) (the symbol $\sim$ represents the state equality, X represents the Not gate, I is the identity gate (i.e., the identity matrix) and F represents a phase change gate). The measurement is used in the process of the reconstruction of the state t, which is done correctly with probability one. From the proof tree shown in Fig. 5 it can be seen that the described computational process contains four operational traces. Using the notion of the operational trace, we can prove the following proposition:

Proposition 9. *The teleportation protocol given by the proof tree (Fig. 5) is a deterministic quantum process.*

Proof. The proof is very short if the notion of the operational trace is used. The space of the operational traces of the teleportation protocol contains four sequences:

$$\begin{aligned}
\xi_0 &= (\mathrm{CNot}, \mathrm{H}, \mu \to (00)_2, \mathrm{I}),\\
\xi_1 &= (\mathrm{CNot}, \mathrm{H}, \mu \to (01)_2, \mathrm{X}),\\
\xi_2 &= (\mathrm{CNot}, \mathrm{H}, \mu \to (10)_2, \mathrm{F}),\\
\xi_3 &= (\mathrm{CNot}, \mathrm{H}, \mu \to (11)_2, \mathrm{X}, \mathrm{F}).
\end{aligned}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{B \sim t}{\dfrac{|B\rangle \overset{I}{\twoheadrightarrow} |t\rangle}{|tA^3_\perp\rangle = (00)_2} \quad \dfrac{|B\rangle \overset{X}{\twoheadrightarrow} |t\rangle}{|tA^3_\perp\rangle = (01)_2} \quad \dfrac{|B\rangle \overset{F}{\twoheadrightarrow} |t\rangle}{|tA^3_\perp\rangle = (10)_2} \quad \dfrac{|B\rangle \overset{XF}{\twoheadrightarrow} |t\rangle}{|tA^3_\perp\rangle = (11)_2}}}{|tAB^2\rangle \overset{\mu(tA)}{\twoheadrightarrow} |tA^3_\perp\rangle|B^2\rangle}}{|tAB^1\rangle \overset{H(t)}{\twoheadrightarrow} |tAB^2\rangle}}{|tAB^0\rangle \overset{\mathrm{CNot}(t,A)}{\twoheadrightarrow} |tAB^1\rangle}$$

Fig. 5. Operational proof tree of the teleportation protocol.

Therefore, we can conclude that the teleportation protocol is deterministic. ■

The first two operations of each trace are the same, but the difference is in the third operation. The third operation—the μ computational step representing the measure procedure—causes a need for using different operations to achieve the final state. A proper sequence of operations after μ, leading to the final state, is always known regardless of the result of μ (which is probabilistic). It must be stressed that the final state is the same for all four traces. Due to this fact we can conclude that the teleportation protocol is deterministic.

The operational tree depicted in Fig. 6 represents the one-bit teleportation protocol implemented in (Kak, 2003).

Proposition 10. *The one-bit version of the teleportation protocol given by the proof tree (Fig. 6) is a deterministic process.*

Proof. The proof of this proposition is again short and easy if the notion of the operational trace is used. The one-bit teleportation protocol contains only two operational traces depicted below:

$$\xi_0 = (\mathrm{H}, \mathrm{CNot}, \mathrm{CNot}, \mathrm{CNot}, \mathrm{H}, \mu \rightarrow (00)_2 \,\mathrm{or}\, (01)_2, \mathrm{I}),$$
$$\xi_1 = (\mathrm{H}, \mathrm{CNot}, \mathrm{CNot}, \mathrm{CNot}, \mathrm{H}, \mu \rightarrow (11)_2 \,\mathrm{or}\, (10)_2, \mathrm{Z}).$$

The first five operations of each trace are the same, but the difference is in the sixth operation. The sixth operation—the μ computational step representing the measure procedure—causes the need for using different operations to achieve the final state. A proper sequence of operations after μ, leading to the final state, is always known regardless of the result of μ (which is probabilistic). It must be stressed that the final state is the same for two traces. Due to this fact we can conclude that the one-bit teleportation protocol is deterministic. ■

4.3. Superdense coding.

The next example presents the deterministic property of the superdense coding protocol, which is the opposite of the teleportation protocol. To obtain a better legibility, the proof tree is split into two sections. One is for classical states $(00)_2$ and $(01)_2$:

$$\dfrac{\dfrac{|AB\rangle \overset{B}{\rightarrow} |00\rangle_\perp}{(00)_2, |AB\rangle \overset{I(A)}{\rightarrow} |AB\rangle} \quad \dfrac{|A^1B\rangle \overset{B}{\rightarrow} |01\rangle_\perp}{(01)_2, |AB\rangle \overset{X(A)}{\rightarrow} |A^1B\rangle}}{|AB\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)}, \tag{31}$$

and the second for classical states $(10)_2$ and $(11)_2$:

$$\dfrac{\dfrac{|A^1B\rangle \overset{B}{\rightarrow} |10\rangle_\perp}{(10)_2, |AB\rangle \overset{F(A)}{\rightarrow} |A^1B\rangle} \quad \dfrac{|A^1B\rangle \overset{B}{\rightarrow} |11\rangle_\perp}{(11)_2, |AB\rangle \overset{X(A),F(A)}{\rightarrow} |A^1B\rangle}}{|AB\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)}. \tag{32}$$

Proposition 11. *The superdense coding protocol is a deterministic quantum process.*

An additional operation, i.e., a measurement step to obtain classical values, is executed on the states which are eigenstates of the observable. The corresponding measurement operator represents the computational step μ: $|\psi\rangle_\perp \overset{\mu}{\twoheadrightarrow} |\psi\rangle_\perp$.

4.4. Predicate for the Grover algorithm.

In the traditional Grover algorithm (Grover, 1996) for one searched state the predicate has a trivial form of the projector onto a one-dimensional subspace of a Hilbert space of the states of the register. When the Grover algorithm is applied to search for two different states, the predicate cannot be expressed as a simple projector onto such a subspace. We have to use a positive operator to extract important information. For example, consider a quantum register with three qubits. The diffuse operator is denoted by the symbol d and the change of the sign of the amplitude by s. In every iteration step we apply the operators s and d. In our example only two iterations are enough to find the solution.

Suppose that we are searching for two states indexed by the indices 1 and 3. Then the appropriate predicate has the form of a discrete POVM. The first predicate is used to obtain information on the amplitudes of the searched

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{|B^2\rangle \overset{I}{\twoheadrightarrow} |B^2\rangle}{|t^2A^3\rangle = 00 \text{ or } 01} \qquad \dfrac{|B^2\rangle \overset{Z}{\twoheadrightarrow} |B^3\rangle}{|t^2A^3\rangle = 10 \text{ or } 11}}{|t^1A^2B^2\rangle \overset{\mu(t^1A^2)}{\twoheadrightarrow} |t^2A^3\rangle_\perp |B^2\rangle}}{|tA^1B^1\rangle \overset{\mathrm{CNot}(t,A^1)}{\twoheadrightarrow} |tA^2B^1\rangle \overset{\mathrm{CNot}(A^2,B^1)}{\twoheadrightarrow} |tA^2B^2\rangle \overset{H(t)}{\twoheadrightarrow} |t^1A^2B^2\rangle}}{|tA^1B\rangle \overset{\mathrm{CNot}(A^1,B)}{\twoheadrightarrow} |tA^1B^1\rangle}}{|tAB\rangle \overset{H(A)}{\twoheadrightarrow} |tA^1B\rangle}
$$

Fig. 6. Operational proof tree of one-bit the teleportation protocol.

states:

$$
F_{1,3} = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix} \tag{33}
$$

and the predicate for other states has the form

$$
F_{\mathrm{oth}} = \begin{pmatrix}
\frac{1}{6} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{6} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{6} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{6} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{6} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{6}
\end{pmatrix}. \tag{34}
$$

The matrices $F_{1,3}$ and F_{oth} obey the positive operator-valued measurement assumption:

$$
\begin{cases} F_{1,3} \geq 0, \ F_{\mathrm{oth}} \geq 0 \\ 2F_{1,3} + 6F_{\mathrm{oth}} = \mathbb{I}. \end{cases} \tag{35}
$$

>From the last formula it follows that

$$
2\mathrm{Tr}(F_{1,3}\rho) + 6\mathrm{Tr}(F_{\mathrm{oth}}\rho) = 1 \tag{36}
$$

for any density matrx ρ.

These operators do not commute, which means that these predicate give us mutual information about the state of the quantum register. In other words, these predicates are unambiguous. The application of $F_{1,3}$ gives an answer as to whether the quantum register is in the searched state, and application of the second predicate F_{oth} allows us to obtain information about the probability of collapsing into a state different than the searched one.

The following table shows values attained by the predicates corresponding to the Grover algorithm running for the first four iterations on the three qubits register:

	$\mathrm{Tr}(F_{1,3}\rho)$	$\mathrm{Tr}(F_{\mathrm{oth}}\rho)$
i_1	0.25	0.75
i_2	1	0
i_3	0.25	0.75
i_4	$\dots$	$\dots$

.

The first iteration gives equal superposition states and both predicates are satisfied, but the trace of applying the second predicate is greater that the first one. In the second iteration the situation is completely different. The first predicate is true and the second one is false, which means that the Grover algorithm has found the searched state with probability 1. This example covers the case where the Grover algorithm is fully deterministic. More information about this deterministic case for the Grover algorithm can be found, e.g., in (Hirvensalo, 2001).

4.4.1. General form of predicates in the Grover algorithm. In a general case for n qudit registers with d degrees of freedom, it is possible to formulate two types of predicates. First, the set P^S represents the success predicates

$$
P^S = \{p_1^s, p_2^s, \dots, p_i^s\}.
$$

The set P^F represents the failure predicates

$$
P^F = \{p_1^f, p_2^f, \dots, p_j^s\},
$$

where

$$
i + j \leq d^n.
$$

For both types of predicates we have

$$
\sum_i p_i^s + \sum_j p_j^f = \mathbb{I}, \quad \sum_i p_i^s \perp \sum_j p_j^f, \tag{37}
$$

and the given predicates satisfy the relation

$$\mathrm{sat}_{PS}(\rho) + \mathrm{sat}_{PF}(\rho) = 1. \tag{38}$$

4.5. Sketch of a natural algorithm of quantum programs synthesis.

Our algorithm is based on the classical predicate calculus. For the assertion $\{?\}\ x \leftarrow 1 + y\ \{x > 5\}$, where we try to find an entry predicate, after an elementary transformation the entry predicate $1 + y > 5, \quad y > 5 - 1, \quad \underline{y > 4}$ was attained. On the other hand, if we know the entry and the postpredicate, we may try to find a transformation action. We must add to x some value fulfilling the postpredicate. We change the value of x with y: $x := y$, but we must use the value in the conditions $x > 5$ and $y > 4$. Because we know that y is greater than four, we can add $(5 - 4)$ to fulfil the postpredicate. Finally, $x := y + 1$ is attained.

A similar situation exists with a quantum transformation for two given states, denoted by $|\psi_{\mathrm{in}}\rangle$ and $|\psi_{\mathrm{out}}\rangle$. The transformation B might be found with the use of the following transformation:

$$B|\psi_{\mathrm{in}}\rangle = |\psi_{\mathrm{out}}\rangle, \quad B^\dagger|\psi_{\mathrm{out}}\rangle = |\psi_{\mathrm{in}}\rangle, \quad BB^\dagger = I. \tag{39}$$

We use the leftmost equation in the case where $B = B^\dagger$ or the right most equation in the case where $B \neq B^\dagger$. Additionally, to verify the result, we can use the predicate to prove that the obtained state is proper.

4.5.1. Superdense coding for three qubits as an example of a synthesis unitary operation.

In this example, we try to find a B gate in the superdense coding protocol for three qubits. We use maximaly entangled states, e.g., the GHZ state $|GHZ\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$. The superdense coding protocol is based on the transformation between two orthonormal bases. The first base $|\mathrm{in}\rangle$ is formed from the following entangled states:

$$\begin{aligned}
|\mathrm{in}_{0,7}\rangle &= \tfrac{1}{\sqrt{2}}(|000\rangle \pm |111\rangle),\\
|\mathrm{in}_{3,4}\rangle &= \tfrac{1}{\sqrt{2}}(|100\rangle \pm |011\rangle),\\
|\mathrm{in}_{2,5}\rangle &= \tfrac{1}{\sqrt{2}}(|010\rangle \pm |101\rangle),\\
|\mathrm{in}_{1,6}\rangle &= \tfrac{1}{\sqrt{2}}(|110\rangle \pm |001\rangle).
\end{aligned}$$

The second base $|\mathrm{out}\rangle$ consists of the following states:

$$\begin{aligned}
|\mathrm{out}_0\rangle &= |000\rangle, & |\mathrm{out}_1\rangle &= |001\rangle,\\
|\mathrm{out}_2\rangle &= |010\rangle, & |\mathrm{out}_3\rangle &= |011\rangle,\\
|\mathrm{out}_4\rangle &= |100\rangle, & |\mathrm{out}_5\rangle &= |101\rangle,\\
|\mathrm{out}_6\rangle &= |110\rangle, & |\mathrm{out}_7\rangle &= |111\rangle.
\end{aligned}$$

It is easy to check that the elementary set of predicates is given by the density matrices built from the base states, e.g., for states 0 and 6 we have

$$F_0 = |000\rangle\langle 000|, \quad F_6 = |110\rangle\langle 110|.$$

The set of predicates $\{F_0, F_1, \ldots, F_7\}$ is the invariant set of predicates for the three-qubit superdense coding protocol.

Proposition 12. *$P_{\mathrm{sdp}} = \{F_0, F_1, \ldots, F_7\}$ is the set of invariant predicates for the superdense coding protocol*

$$\sum_{i=0}^{7} F_i = \sum_i |i\rangle\langle i| = \mathbb{I}$$

and

$$\forall_{i=0,1,\ldots,7}\ \mathrm{Tr}(F_i|\mathrm{in}_i\rangle) > 0 \ \ \textit{and}\ \ \mathrm{Tr}(F_i|\mathrm{out}_i\rangle) > 0.$$

Proof. It is obtained by direct calculations. ■

This proposition can be easily generalised to a superdense coding protocol for n qubits.

Proposition 13. *$P_{sdp} = \{F_0, F_1, \ldots, F_{n-1}\}$ is the set of invariant predicates for a superdense coding protocol on n qubits,*

$$\sum_{i=0}^{n-1} F_i = \sum_i |i\rangle\langle i| = \mathbb{I}$$

and

$$\forall_{i=0,1,\ldots,n-1}\ \mathrm{Tr}(F_i|\mathrm{in}_i\rangle) > 0 \ \ \text{and}\ \ \mathrm{Tr}(F_i|\mathrm{out}_i\rangle) > 0.$$

The B transformation can be found by solving the eight linear equations in the form $B|\mathrm{out}\rangle = |\mathrm{in}\rangle$. Let B_x be the matrix defined as the following one (where each value is treated as an unknown variable):

$$B_x = \begin{pmatrix} x_{11} & \ldots & x_{18} \\ \vdots & \ddots & \vdots \\ x_{81} & \ldots & x_{88} \end{pmatrix}.$$

We obtain eight equations,

$$\begin{aligned}
B_x|\mathrm{out}_0\rangle &= |\mathrm{in}_0\rangle, & B_x|\mathrm{out}_1\rangle &= |\mathrm{in}_1\rangle,\\
B_x|\mathrm{out}_2\rangle &= |\mathrm{in}_2\rangle, & B_x|\mathrm{out}_3\rangle &= |\mathrm{in}_3\rangle,\\
B_x|\mathrm{out}_4\rangle &= |\mathrm{in}_4\rangle, & B_x|\mathrm{out}_5\rangle &= |\mathrm{in}_5\rangle,\\
B_x|\mathrm{out}_6\rangle &= |\mathrm{in}_6\rangle, & B_x|\mathrm{out}_7\rangle &= |\mathrm{in}_7\rangle.
\end{aligned}$$

After solving each of these linear equations, we found the entries of the matrix B. For example, from the first equation for state 0, $B_x|\mathrm{out}_0\rangle = |\mathrm{in}_0\rangle$, we obtain the values forming the first row of the transformation matrix B:

$$x_{11} = \frac{1}{\sqrt{2}}, \quad x_{12} = 0, \quad \ldots, \quad x_{17} = 0, \quad x_{18} = \frac{1}{\sqrt{2}}.$$

The first part of operational tree:

$$\dfrac{\dfrac{\dfrac{|\psi\rangle_{000}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,I}{|\psi\rangle_{0000}}\quad |\psi\rangle_{000}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,I}{|\psi\rangle_{0001}}}{|\psi\rangle_{00}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{000}}\qquad\dfrac{|\psi\rangle_{001}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,Z}{|\psi\rangle_{0010}}\quad |\psi\rangle_{001}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,Z}{|\psi\rangle_{0011}}}{|\psi\rangle_{00}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{001}}}{|\psi\rangle_{0}\overset{\mu_{s_2}}{\twoheadrightarrow}|\psi\rangle_{00}}\qquad\dfrac{\dfrac{|\psi\rangle_{010}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,I}{|\psi\rangle_{0100}}\quad |\psi\rangle_{010}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,I}{|\psi\rangle_{0101}}}{|\psi\rangle_{01}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{010}}\qquad\dfrac{|\psi\rangle_{011}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,Z}{|\psi\rangle_{0110}}\quad |\psi\rangle_{011}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,Z}{|\psi\rangle_{0111}}}{|\psi\rangle_{01}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{011}}}{|\psi\rangle_{0}\overset{\mu_{s_2}}{\twoheadrightarrow}|\psi\rangle_{01}}}{|\psi\rangle\overset{\mu_{s_1}}{\twoheadrightarrow}|\psi\rangle_{0}}$$

The second part of operational tree:

$$\dfrac{\dfrac{\dfrac{|\psi\rangle_{100}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,Z}{|\psi\rangle_{1000}}\quad |\psi\rangle_{100}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,Z}{|\psi\rangle_{1001}}}{|\psi\rangle_{10}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{100}}\qquad\dfrac{|\psi\rangle_{101}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,I}{|\psi\rangle_{1010}}\quad |\psi\rangle_{101}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,I}{|\psi\rangle_{1011}}}{|\psi\rangle_{10}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{101}}}{|\psi\rangle_{1}\overset{\mu_{s_2}}{\twoheadrightarrow}|\psi\rangle_{10}}\qquad\dfrac{\dfrac{|\psi\rangle_{110}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,Z}{|\psi\rangle_{1100}}\quad |\psi\rangle_{110}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,Z}{|\psi\rangle_{1101}}}{|\psi\rangle_{11}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{110}}\qquad\dfrac{|\psi\rangle_{111}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{X,I}{|\psi\rangle_{1110}}\quad |\psi\rangle_{111}\overset{\mu_{s_4}}{\twoheadrightarrow}\overset{I,I}{|\psi\rangle_{1111}}}{|\psi\rangle_{11}\overset{\mu_{s_3}}{\twoheadrightarrow}|\psi\rangle_{111}}}{|\psi\rangle_{1}\overset{\mu_{s_2}}{\twoheadrightarrow}|\psi\rangle_{11}}}{|\psi\rangle\overset{\mu_{s_1}}{\twoheadrightarrow}|\psi\rangle_{1}}$$

Fig. 7. Simulation of the unitary matrix u by a measurement pattern. We use four μ steps with appropriate parameters which are specified for the simulated β unitary step. After measurements we apply Pauli gates (X, Y, Z, I) to correct the final state.

Finally, the representation of the unitary operation U for superdense coding can be expressed by the following matrix:

$$B = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 & 0 \\ \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & \frac{-1}{\sqrt{2}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & \frac{-1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 & 0 & 0 \end{pmatrix}. \tag{40}$$

It is easy to check that the obtained transformation B is unitary, i.e., $BB^{\dagger} = \mathbb{I}$.

4.6. Simulation of the β-step using the μ-steps.

Figure 7 presents a complete operational description for the simulation process of the β-step with the measurement of the μ-step. The simulation of the β-step is based on a one-way computational model (termed 1WQC) where the measurement plays the main role in the computations. The operational proof tree shows that this procedure is fully deterministic. Figure 7 also shows that the 1WQC and circuit models are the same from the operational point of view. It is easily proven that it is possible to mix both models. The only fundamental difference between those models is the fact that in the circuit mode the unitary gates play the main role and in the 1WQC the measurements represent the main computational steps. Moreover, in both models, the measurement is used to obtain the final result. To proceed further, we introduce the following result.

Proposition 14. *The simulation of the β-step using the μ-steps is deterministic.*

Proof. The proof is easy, because the final state is uniquely determined by the result of all previous μ steps. The fact that the measurement is probabilistic is insignificant here because each final result uniquely determines the required β-step to be used. ■

As a result of the measurement, one gets a set of four binary digits. This means that there are 16 possible results. Each of those uniquely determine the required β operators, see Fig. 7.

The proof tree from Fig. 7 also shows that the algorithm of simulation of the β-step is not computationally stable (i.e., the number of required operations varies depending on the obtained measurement result). There exist four cases where no additional steps is necessary, eight where one is needed, and four cases where two additional steps are required (eight simple β-steps). This shows that the computational complexity of this algorithm (counting the number of operations) is given by

$$4_{\mu} = \mathcal{O}(1) \text{ or } 4_{\mu}+1_{\beta} = \mathcal{O}(1) \text{ or } 4_{\mu}+2_{\beta} = \mathcal{O}(1). \tag{41}$$

For any computation process where we simulate n β-steps, we obtain

$$\begin{aligned} T(n) &= n4_{\mu} = \mathcal{O}(n) && \text{or} \\ T(n) &= n(4_{\mu} + 1_{\beta}) = \mathcal{O}(n) && \text{or} \\ T(n) &= n(4_{\mu} + 2_{\beta}) = \mathcal{O}(n). \end{aligned}$$

In the sense of $\mathcal{O}(\cdot)$ notation, the foregoing cases have the same linear complexity. This fact can be used to construct a very simple proof that the 1WQC and the circuit model belong to the same class of computational complexity.

Proposition 15. *The simulation of n β steps using the 1WQC has the computational complexity of $\mathcal{O}(n)$.*

Proof. If we simulate n β-steps, the number of operations varies from $4n$ to $6n$. In both the cases the complexity is given by $\mathcal{O}(n)$. ■

5. Conclusions

In this paper the notion of quantum labelled transition systems and the predicate for a discrete quantum computation model were presented. The definition of the GSOS format for quantum labelled transition systems was also introduced. An important theorem for finite quantum transition systems for discrete quantum computation models was achieved.

The definition of the predicate notion was achieved by the formulation of a general notion of the predicate as a positive operator-valued measure. The cases of the predicates defined in the literature can be regarded as special cases of our general definition. We also defined a measure of satisfability for predicates defined as positive operators (which are not projectors) and projectors.

In the last part of this text, we presented some simple examples of using the predicates in two well-known quantum algorithms: the Grover method for a search in an unstructured database and the superdense coding for three qubits. The proof tree of 1WQC implementation of a one-qubit unitary gate was presented. The above proof tree allows us to prove two important properties: finiteness and determinism.

One of further tasks would be a more general formulation of the existence theorem of weakest precondition semantics expressed in terms of positive operator-valued measures. The result presented here for finite and discrete quantum systems is the first step in this direction. A slight generalisation of our result can be found in (Gielerak and Sawerwain, 2007).

Acknowledgements

We acknowledge useful discussions on qTLS with the *Q-INFO* group at the Institute of Control and Computation Engineering of the University of Zielona Góra, Poland. The authors would like to thank the anonymous referees for their useful comments and suggestions.

References

Aceto L. (1994). GSOS and finite labelled transition systems, *Theoretical Computer Science* **131**(1): 181–195.

de Bakker J. W., de Roever W. P. (1972). A calculus for recursive programs schemes, *in:* M. Nivat (Ed.), *Automata, Languages, and Programming*, North-Holland, Amsterdam, pp. 167–196.

de Bakker J. W., Meertens, L. G. L. T. (1975). On the completeness of the inductive assertion method, *Journal of Computer and Systems Sciences* **11**(3): 323–357.

Bennett C.H., Brassard G., Crepeau C., Jozsa R., Peres A. and Wooters W.K. (1993). Teleporting an unknown state via dual classical and Einstein-Podolsky-Rosen channels, *Physical Review Letters* **70**(13): 1895–1899.

Birkhoff G. and von Neumann J. (1936). The logic of quantum mechanics, *Annals of Mathematics* **37**(4): 823-843.

Bloom B. (1989): *Ready Simulation, Bisimulation, and the Semantics of CCS-like Languages*, Ph.D. thesis, Massachusetts Institute of Technology.

Bloom B., Istrail S., Meyer A.R. (1989). Bisimulation can't be traced: Preliminary report, *Conference Record of the 15th Annual ACM Symposium on Principles of Programming Languages,* San Diego, CA, USA, pp. 229–239.

Boschi D., Branca S., de Martini F., Hardy L. and Popescu S. (1998). Experimental realization of teleporting an unknown pure quantum state via dual classical and Einstein-Podolsky-Rosen channels, *Physical Review Letters* **80**(6): 1121–1125.

Bouwmeester D., Pan J.W., Mattle K., Eibl M., Weinfurter H. and Zeilinger A. (1997). Experimental quantum teleportation, *Nature* **390**(6660): 575–579.

Choi M.D. (1975). Completely positive linear maps on complex matrices, *Linear Algebra and Its Applications* **10**(3): 285–290.

Coecke B. and Martin K. (2002). *A partial order on classical and quantum states*, Technical report, PRG-RR-02-07, Oxford University.

Deutsch D. and Jozsa R. (1992). Rapid solutions of problems by quantum computation, *Proceedings of the Royal Society of London A*, **439**(1907): 553–558.

Dijkstra E. W. (1976). *A Discipline of Programming*, Prentice-Hall, Englewood Cliffs, NJ.

D'Hondt E. and Panangaden P. (2006). Quantum weakest preconditions, *Mathematical Structures in Computer Science* **16**(3): 429–451.

Gielerak R. and Sawerwain M. (2007). *Generalised quantum weakest preconditions*, available at:
arXiv:quant-ph/0710.5239v1.

Gleason A. M. (1957). Measures on the closed subspaces of a Hilbert space, *Journal of Mathematics and Mechanics* **6**(4): 885–893.

Grover L. K. (1996). A fast quantum-mechanical algorithm for database search, *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing*, Philadelphia, PA, USA, ACM Press, New York, NY, pp. 212–219.

Hirvensalo M. (2001). *Quantum Computing*, Springer-Verlag, Berlin.

Hoare C. (1969). An axiomatic basis for computer programming, *Communications of the ACM* **12**(10): 576–583.

Jozsa R. (2005). *An introduction to measurement based quantum computation,* available at:
arXiv:quant-ph/0508124.

Kraus K. (1983). *State, Effects, and Operations*, Springer, Berlin.

Kak S. (2003). *Teleportation protocols requiring only one classical bit*, available at:
arXiv:quant-ph/0305085v4.

Lalire M., Jorrand P. (2004). A process algebraic approach to concurrent and distributed quantum computation: Operational semantics, *Proceedings of the 2nd International Workshop on Quantum Programming Languages,* Turku, Finland, pp. 109–126.

Löwner K. (1934): Über monotone Matrixfunktionen, *Mathematische Zeitschrift* **38**(1): 177-–216.

Mlnařík H. (2006): *LanQ–Operational Semantics of Quantum Programming Language LanQ,* Technical report FIMU-RS-2006-10, available at:
http://www.muni.cz/research/publications/706560.

Mauerer W. (2005). *Semantics and simulation of communication in quantum programming,* M.Sc. thesis, University Erlangen-Nuremberg Erlangen, Nürnberg, see:
arXiv:quant-ph/0511145.

Ömer B. (2005). Classical concepts in quantum programming, *International Journal of Theoretical Physics*, **44**(7): 943–955, see:
arXiv:quant-ph/0211100.

Peres A. (1995). *Quantum Theory: Concepts and Methods,* Kluwer Academic Publishers, Dordrecht.

Plotkin G.D. (2004). A structural approach to operational semantics, *Journal of Logic and Algebraic Programming* **60**: 17–139.

Raynal P. (2006). *Unambiguous state discrimination of two density matrices in quantum information theory*, Ph.D. thesis, Institut für Optik, Information und Photonik, Max Planck Forschungsgruppe, see: arXiv:quant-ph/0611133.

Rüdiger R. (2007). Quantum programming languages: An introductory overview, *The Computer Journal* **50**(2): 134–150.

Raussendorf R., Briegel H.J. (2001). A one-way quantum computer, *Physical Review Letters* **86**(22): 5188–5191, see: arXiv:quant-ph/0010033.

Raussendorf R., Browne D.E., Briegel H.J. (2003). Measurement-based quantum computation with cluster states, *Physical Review A*, **68**(2), 022312, see:
arXiv:quant-ph/0301052.

Sawerwain M., Gielerak R. and Pilecki J. (2006). Operational semantics for quantum computation, *in:* Węgrzyn S., Znamirowski L., Czachórski T., Kozielski S. (Eds.), *New Technologies in Computer Networks,* WKiŁ, Warsaw, Vol. 1, pp. 69–77, (in Polish).

Selinger P.: (2004): Towards a quantum programming language, *Mathematical Structures in Computer Science* **14**(5): 527–586.

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

Sewell G.: (2005). On the mathematical structure of quantum measurement theory, *Reports on Mathematical Physics* **56**(2): 271–290, see: arXiv:math-ph/0505032.

Shor P. (2004). Progress in quntum algorithms, *Quantum Information Processing* **3**(1): 5–13.

Received: 17 April 2007
Revised: 18 August 2007
Re-revised: 5 May 2008