Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science
ELSEVIER
Electronic Notes in Theoretical Computer Science 210 (2008) 139–152
www.elsevier.com/locate/entcs

Quantum Arrows in Haskell

Juliana Kaizer Vizzotto[1], Antônio Carlos da Rocha Costa[2]

Federal University of Rio Grande do Sul, Brazil
Catholic University of Pelotas, Brazil

and

Amr Sabry[3]

Indiana University, USA

Abstract

We argue that a realistic model for quantum computations should be *general* with respect to measurements, and *complete* with respect to the information flow between the quantum and classical worlds. We discuss two alternative models for general and complete quantum computations based on probability distributions of quantum state vectors and on density matrices with classical outputs. We show that both models can be structured using a generalization of monads called arrows.

Keywords: Quantum programming, density matrices, probabilities.

1 Introduction

In recent work [13] we established that a *general* model of quantum computing (including measurements), based on density matrices and superoperators, is an instance of a generalization of monads called *arrows* [5]. That work is strictly based on quantum data (any classical value must be represented as quantum). The model cannot express the passage of information between the classical and quantum worlds.

However, various quantum algorithms are explained in terms of the *interchanging* of quantum and classical information [4]. For example, quantum teleportation is a

[1] Email: jkv@atlas.ucpel.tche.br
[2] Email: rocha@atlas.ucpel.tche.br
[3] Email: sabry@indiana.edu

[4] By interchanging we mean, for instance, a measurement in the middle of the computation.

doi:10.1016/j.entcs.2008.04.023

traditional algorithm which is based on two quantum processes communicating via *classical data*. There is interest to consider *measurements* and the *information flow* between quantum and classical processes as essential components of quantum computations (for instance, see [11,6,9,4,12]).

On the other hand, the finding of a representation that is suitable for representing both the results of unitary transformations and measurement operations should also be put into perspective, in order to uniformly fit with the requirements of generality and completeness.

That is, we would like that the same representational framework be able to take care of both: (1) the task of representing the *quantum state* resulting from a unitary operation applied to a given quantum state, and (2) the task of representing the pair of information coming out from a measurement, namely: (2a) that corresponding to the *measure* produced by the measurement (one of the eigenvalues of the measurement operator), and (2b) the *quantum state* that results from the projection imposed on the original quantum state by the measurement (one of the eigenvectors of the measurement operator).

The main problem introduced by the need of that uniformity is that measurement results (both value and state results) are of a probabilistic kind, needing *sets of possible results* for their representation. The usual alternative solution to such problem is the density matrix formalism.

However, there is a (possibly not minor) conceptual problem in the adoption of the density matrix formalism, namely: a density matrix is supposed to represent a set (*ensemble*) of quantum systems whose probability distribution of states the density matrix represents; however, from a programming theoretic point of view, one usually thinks of a quantum algorithm as being performed by one single quantum system, not an ensemble of quantum systems each possibly behaving in a different way according to a probability distribution.

We feel that the quantum programmer's intuition of programming one single quantum system at a time, while elaborating his algorithms, may happen to be not appropriately captured by the density matrix formalism. We feel (but we have no definite argument) that a representation modelled on the usual set-theoretic representation of states of non-deterministic machines, adjusted to explicitly represent the probability of occurrence of each deterministic state, may happen to capture in a better way the quantum programmer's intuition.

So, in the paper, we introduce two ways to deal with *combined* quantum and classical computations, which are based on different ways of representing states which result from measurements, one based on density matrices, the other based on explicit probability distributions over sets of quantum states.

The paper is organised as follows. In Section 2 we introduce *indexed monads* and *indexed arrows* in the context of Haskell. Section 3 briefly reviews our previous work [13] on modelling superoperators as arrows. We then show two alternative general and complete models for combined quantum and classical computations structured as indexed arrows in Section 4. Section 5 concludes.