Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 319 (2015) 369–385

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Complete Positivity and Natural Representation of Quantum Computations

Mathys Rennela[1]

Institute for Computing and Information Sciences
Radboud University
Nijmegen, The Netherlands

Sam Staton[2]

Department of Computer Science
Oxford University
Oxford, United Kingdom

Abstract

We propose a new 'quantum domain theory' in which Scott-continuous functions are replaced by Scott-continuous natural transformations.

Completely positive maps are widely accepted as a model of first-order quantum computation. We begin by establishing a categorical characterization of completely positive maps as natural families of positive maps. We explore this categorical characterization by building various representations of quantum computation based on different structures: affine maps between cones of positive elements, morphisms of algebras of effects, and affine maps of convex sets of states. By focusing on convex dcpos, we develop a quantum domain theory and show that it supports some important constructions such as tensor products by quantum data, and lifting.

Keywords: Operator algebra, complete positivity, quantum computation, domain theory, convex set

Introduction

This paper is about semantic models of quantum computation. In common with other approaches to programming language semantics, the general idea is to interpret a type A as a space $[\![A]\!]$ of observations about A. One interprets a computation $x : A \vdash t : B$, that produces something t of type B but depends on something x of type A, as a predicate transformer $[\![B]\!] \to [\![A]\!]$, which maps a predicate on B to its weakest precondition. (See e.g. [4,18,3].)

[1] Email: mathysr@cs.ru.nl

[2] Email: sam.staton@cs.ox.ac.uk

http://dx.doi.org/10.1016/j.entcs.2015.12.022

1571-0661/

In more detail, one interprets a type A as a C*-algebra of operators $[\![A]\!]$, and the computations describe maps that are in particular *positive*: it is actually only the positive elements of the algebra that describe the observables, and these must be preserved. Moreover the maps should be *completely positive*. Informally this means that it makes sense to run the computation on a subsystem of a bigger system; for example, we could adjoin an extra qubit to the system and still run the computation. More formally it means that not only does the map $[\![t]\!] : [\![B]\!] \to [\![A]\!]$ preserve positive elements, but also $\mathrm{id}_{[\![\mathrm{qubit}]\!]} \otimes [\![t]\!] : [\![\mathrm{qubit}]\!] \otimes [\![B]\!] \to [\![\mathrm{qubit}]\!] \otimes [\![A]\!]$ preserves positive elements.

The first contribution of this paper (Section 2) is a technique for building representations of quantum computation in terms of completely positive maps. In the second half of the paper (Section 3) we demonstrate our technique by making some first steps in the development of a 'quantum domain theory'.

A technique for building representations

Here, a representation is a full and faithful functor $F : \mathbf{C} \to \mathbf{R}$, that is, a functor for which each function $F_{A,B} : \mathbf{C}(A, B) \to \mathbf{R}(F(A), F(B))$ is a bijection.

From a programming language perspective, where objects interpret types and morphisms interpret programs, a representation result gives two things. Firstly, it gives a way of interpreting types as different mathematical structures, which can be illuminating or convenient, while retaining essentially the same range of interpretable programs. Secondly, since $\mathbf{R}$ may be bigger than $\mathbf{C}$, it gives the chance to interpret more types without altering the interpretation of programs at existing types.

There are several existing representation results which allow us to understand and analyze quantum computations in terms of different structures, such as convex sets (e.g. [11]), domains (e.g. [18]), partial monoids and effect algebras (e.g. [10]). However, many of these representation results are only valid for positive maps, and so they do not fully capture quantum computation. Our contribution is a general method for extending these results to completely positive maps. Roughly, the method allows us to convert a full and faithful functor

$$(\text{positive maps}) \longrightarrow \mathbf{R}$$

(where $\mathbf{R}$ is an arbitrary category) into a full and faithful functor

$$(\text{completely positive maps}) \longrightarrow [\mathbf{N}, \mathbf{R}]$$

into a functor category, where $\mathbf{N}$ is a category whose objects are natural numbers.

Towards a quantum domain theory

In the second part of the paper we demonstrate our technique by making some first steps in the development of a 'quantum domain theory'. The ultimate goal in this line of work is to analyze all kinds of quantum programming by solving