From reversible to irreversible computations

Alexander S. Green [1,2]

Computer Science and IT
The University of Nottingham
Nottingham, UK

Thorsten Altenkirch [3]

Computer Science and IT
The University of Nottingham
Nottingham, UK

Abstract

In this paper we study the relation between reversible and irreversible computation applicable to different models of computation — here we are considering classical and quantum computation.

Abstract models of computation like λ calculus or more abstractly Cartesian closed categories are based on irreversible processes, indeed Cartesian products introduce projections which are irreversible. In contrast, in Physics the more fundamental notions describe processes in closed systems where every action is reversible, i.e. Newtonian Mechanics, Maxwellian electrodynamics and quantum mechanics fit into this pattern. Open systems, which allow irreversible processes, are a derived notion — they can be considered as a subsystem of a closed system. Indeed, an irreversible process can be understood in terms of a reversible one with a particular assignment of boundary conditions, e.g. Feynman's and Wheeler's theory of absobers [WF45].

Key words: Reversible Computation, Irreversible Computation, Quantum Computation.

1 Introduction

Our plan is to follow the physical idea that reversibility is the fundamental notion and irreversibility is a derived notion to model computation. Reversibility

[1] We would like to thank Jonathan Grattage for his help and discussions on this paper, and also the referee whose comments were very interesting and useful.

[2] Email: `asg@cs.nott.ac.uk`

[3] Email: `txa@cs.nott.ac.uk`

This paper is electronically published in
Electronic Notes in Theoretical Computer Science
URL: `www.elsevier.nl/locate/entcs`

has been investigated by Bennet in his classical paper [Ben73], where he shows that reversible computation has the same power as irreversible computation. It has also since been shown that in terms of complexity, reversible space is the same as deterministic space [LMT97]. Recently, Abramsky investigated the notion of reversible computation from a structural perspective [Abr01].

We build on previous work of the 2nd author with Jonathan Grattage on compiling QML [AG05]. QML's design is based on an analogy between classical and quantum computation. To make this precise we introduce two models of computation: FCC for Finite Classical Computation and FQC for Finite Quantum Computation. Both are based on a notion of reversible computation (bijections vs. unitary operators) and introduce irreversible computations as a derived notion by marking certain inputs as preinitialised heap and certain outputs as garbage which is thrown away (i.e. measured in the quantum case) at the end of the computation. We also introduce the notion of extensional equivalence of two irreversible computations which are given by the associated functions on finite sets in the classical case and by an embedding into the category of superoperators on finite dimensional Hilbert spaces in the quantum case. While the choice of extensional equality in the two examples is very natural it is not parametric in the notion of reversible computation. That is, we would like to obtain the notion of irreversible computation as a consequence of our choice of reversible computation.

We attempt to fix this here by introducing three laws which state which algebraic properties a notion of irreversible computation derived from reversible computation must satisfy. Both FCC and FQC satisfy these laws and we show that they are sufficient to derive von Neumann's measurement postulate, which in this setting corresponds to *measuring twice is the same as measuring once*. Currently, we have to leave open the question whether the three laws exactly characterise quantum computation for definable circuits, i.e. whether the equivalence of circuits introduced by the quantum model is on some sense the free model of irreversible computation. This doesn't seem to be equivalent to the question whether the category of completely positive maps is equationally definable from initialisation and measurement which is known not to be the case [?], since we only consider definable circuits.

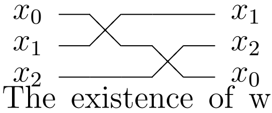
2 Reversible Computation

We model reversible computations by a groupoid $\mathbf{FxC}^{\mathrm{R}}$, that is for every morphism $\psi \in \mathbf{FxC}^{\mathrm{R}}(a,b)$ there is an inverse $\psi^{-1} \in \mathbf{FxC}^{\mathrm{R}}(b,a)$ such that ψ, ψ^{-1} are an isomorphism. We assume that the groupoid is strict, i.e. that any isomorphic objects are equal. This entails that $\mathbf{FxC}^{\mathrm{R}}(a,b)$ is empty, if $a \neq b$, consequently we denote homsets by $\mathbf{FxC}^{\mathrm{R}}\,a = \mathbf{FxC}^{\mathrm{R}}(a,a)$. We also assume that $\mathbf{FxC}^{\mathrm{R}}$ has a strict monoidal structure $I, \otimes$ which corresponds to parallel composition of computations and a special object of Booleans,denoted by 2. Since we are only interested in objects which can be generated from

$I, 2, \otimes$ we can use natural numbers $a \in \mathbb{N}$ to denote the object 2^a. We write $[a] = \{i \in \mathbb{N} \mid i < a\}$ for the initial segment of $\mathbb{N}$.

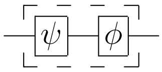
We characterise the morphisms, i.e. circuits, in $\mathbf{FxC}^{\mathrm{R}}a$ inductively and also give the inverses:

wires Given a bijection on initial segments $\phi : [a] \simeq [a]$ we write wires $\phi \in \mathbf{FxC}^{\mathrm{R}}\,a$ for the associated *rewiring*. For example, the rewiring denoted pictorially as

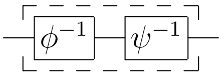


would have $\phi = [1, 2, 0]$. The existence of wires follows from the strict monoidal structure, with the identity (id_a) being a special case of wires.

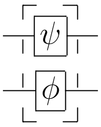
sequential composition combines two circuits of equal size (ie. with the same number of wires) in sequence. That is, given $\psi, \phi \in \mathbf{FxC}^{\mathrm{R}}a$ we construct $\phi \circ \psi \in \mathbf{FxC}^{\mathrm{R}}a$.



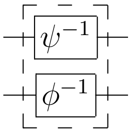
we can construct the inverse using ϕ^{-1} and ψ^{-1} to give $\psi^{-1} \circ \phi^{-1}$.



parallel composition combines any two circuits in parallel, and can be thought of as the tensor product. The size of the new circuit constructed is equal to the sum of the sizes of the original two circuits. That is, given $\psi \in \mathbf{FxC}^{\mathrm{R}}a$ and $\phi \in \mathbf{FxC}^{\mathrm{R}}b$ we can construct $\psi \otimes \phi \in \mathbf{FxC}^{\mathrm{R}}(a + b)$.

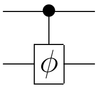


again we can construct the inverse using ψ^{-1} and ϕ^{-1}, this time to give $\psi^{-1} \otimes \phi^{-1}$.

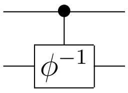


rotations count as any 1 "bit" operations. That is a rotation is any element of $\mathbf{FxC}^{\mathrm{R}}1$, and in the case of classical reversible circuits the only rotation available is the Not operation. So we have $\neg \in \mathbf{FxC}^{\mathrm{R}}1$ with $\neg^{-1} = \neg$. In the quantum case this would obviously be any single qubit rotation.(i.e. a unitary operation in $U(2)$)

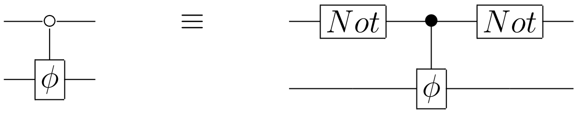
conditionals use a control wire to decide whether a computation should be performed. That is, given $\phi \in \mathbf{FxC}^{\mathrm{R}}a$ we can construct $id_a \mid \phi \in \mathbf{FxC}^{\mathrm{R}}(1 + a)$.



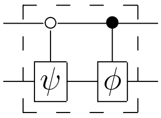
the inverse is again constructed using ϕ^{-1} giving $id_a \mid \phi^{-1}$.



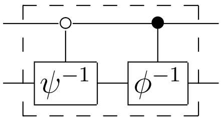
For ease of notation we shall also introduce the conditional that acts when the control wire is set to true. This conditional can be constructed from the conditional already given, and the Not operation (or rotation) as follows:



which for $\phi \in \mathbf{FxC}^{\mathrm{R}}a$ can be denoted $\phi \mid id_a \in \mathbf{FxC}^{\mathrm{R}}(1+a)$. This naturally leads us to a choice operator, such that given two computations of the same size, the value of the control wire is used to govern which computation is done. That is, given $\psi, \phi \in \mathbf{FxC}^{\mathrm{R}}a$ we can construct $\psi \mid \phi \in \mathbf{FxC}^{\mathrm{R}}(1+a)$, as follow:

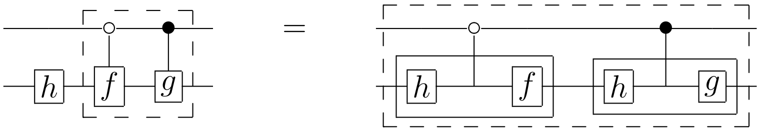


the inverse is once again given by ψ^{-1} and ϕ^{-1}, and constructed as $\psi^{-1} \mid \phi^{-1}$:

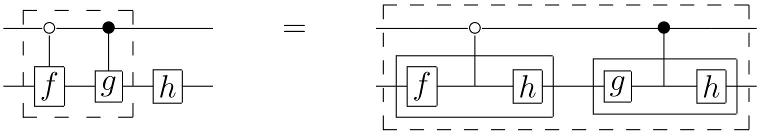


The laws governing wires, sequential composition and parallel composition follow from the categorical infrastructure. Additionally, we assume that the following equalities hold for conditionals:

Firstly we have for $f, g, h \in \mathbf{FxC}^{\mathrm{R}}a$ that $(f \mid g) \circ (2 \otimes h) = f \circ h \mid g \circ h$ pictorially this can be shown as:

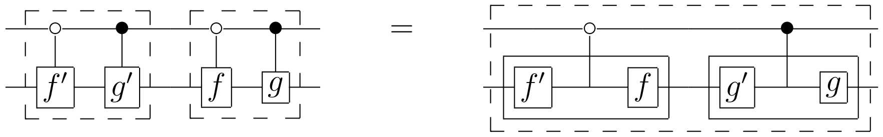


Secondly we have for $f, g, h \in \mathbf{FxC}^{\mathrm{R}}a$ that $(2 \otimes h) \circ (f \mid g) = h \circ f \mid h \circ g$ pictorially this can be shown as:



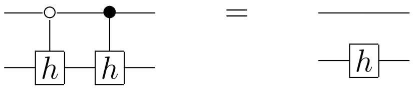
and thirdly we have that for $f, f', g, g' \in \mathbf{FxC}^{\mathrm{R}}a$ that $(f \mid g) \circ (f' \mid g') =$

$(f \circ f') \mid (g \circ g')$ again the pictorial representation for this would be:

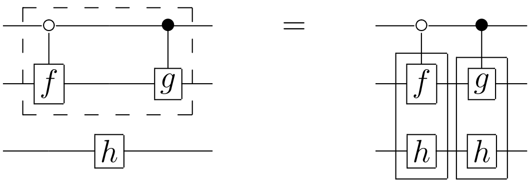


using this last axiom it is possible to simplify the first two to just be that $(h \mid h) = (id_1 \otimes h)$ or pictorially:

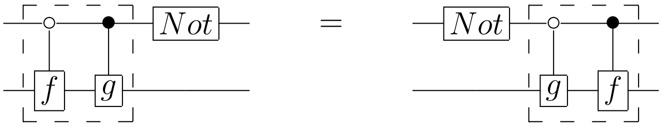
Alex: Check this is true!



We also have distributivity over $\otimes$ and $|$, such that given $f, g \in \mathbf{FxC}^{\mathrm{R}}a$ and $h \in \mathbf{FxC}^{\mathrm{R}}b$ we have that $(f \mid g) \otimes h = (f \otimes h) \mid (g \otimes h)$. This can again be given pictorially.



Instead of considering only powers of 2 we could have modelled arbitrary sized computations by introducing a strict bimonoidal structure (as defined by Laplaza in [?]) with $Z, \oplus$. Defining $2 = I \oplus I$ the conditionals and their laws are derivable from the bimonoidal structure. $\neg$ then becomes a witness of the fact that the additive structure is symmetric. We can also derive for $f, g \in \mathbf{FxC}^{\mathrm{R}}a$ that $(\neg \otimes id_a) \circ (f \mid g) = (g \mid f) \circ (\neg \otimes id_a)$, or pictorially that would be:

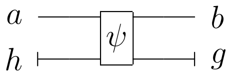


Examples of $\mathbf{FxC}^{\mathrm{R}}$ *categories*

There are two obvious examples that can be given of $\mathbf{FxC}^{\mathrm{R}}$ categories, firstly there is the $\mathbf{FCC}^{\mathrm{R}}$ category of classical reversible circuits, and secondly there is the $\mathbf{FQC}^{\mathrm{R}}$ of quantum circuits. The difference mainly being in the rotations that are available. Interestingly we have that $\mathbf{FCC}^{\mathrm{R}} \hookrightarrow \mathbf{FQC}^{\mathrm{R}}$ and therefore that to show equality in $\mathbf{FCC}^{\mathrm{R}}$ it is enough to show equality (for the same circuit) in $\mathbf{FQC}^{\mathrm{R}}$ (and vice versa).

3 Irreversible computations

We derive a notion of irreversible computations from the given notion of reversible computation by defining the category $\mathbf{FxC}^{\mathrm{ir}}$, where every morphism of the category represents an irreversible computation, but is in fact of the form $\psi' = (h, g, \psi)$ where h is a set of heap inputs, g is a set of garbage outputs, and ψ is the underlying reversible computation. So a morphism in $\mathbf{FxC}^{\mathrm{ir}}(a, b)$ can be given as a morphism in $\mathbf{FxC}^{\mathrm{R}}((a \otimes h), (b \otimes g))$ with the requirement that $(a \otimes h) = (b \otimes g)$. Pictorially we can represent an irreversible computation (h, g, ψ) as the reversible computation ψ where we mark heap and garbage explicitely:

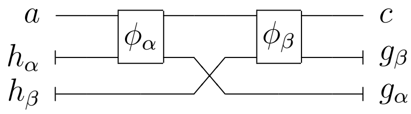


We also have that for any $\psi \in \mathbf{FxC}^{\mathrm{R}} a$ there is an equivalent circuit $\widehat{\psi} \in \mathbf{FxC}^{\mathrm{ir}}(a, a)$, more precisely this is given by the predicate:

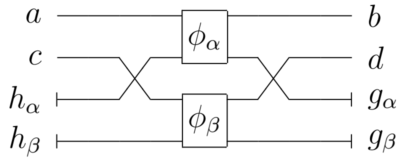
$$\frac{\psi \in \mathbf{FxC}^{\mathrm{R}} a}{\widehat{\psi} \in \mathbf{FxC}^{\mathrm{ir}}(a, a)}$$

such that $\widehat{\psi} = (0, 0, \psi)$, i.e. there is no heap or garbage.

We note that we can define sequential composition for irreversible computations, i.e. given $\alpha = (h_\alpha, g_\alpha, \phi_\alpha) \in \mathbf{FxC}^{\mathrm{ir}}(a, b)$ and $\beta = (h_\beta, g_\beta, \phi_\beta) \in \mathbf{FxC}^{\mathrm{ir}}(b, c)$ we define $\beta \circ \alpha \in \mathbf{FxC}^{\mathrm{ir}}(a, c)$ as:



The identity can be obtained by lifting the reversible identity $id_a^{\mathbf{FxC}^{\mathrm{ir}}} = \widehat{id_a^{\mathbf{FxC}^{\mathrm{R}}}}$. It is straightforward to verify that $\mathbf{FxC}^{\mathrm{ir}}$ thus constructed is a category by using the monoidal indentities in the underlying category of reversible computations. Moreover, $\mathbf{FxC}^{\mathrm{ir}}$ inherits the monoidal structure from $\mathbf{FxC}^{\mathrm{R}}$, e.g. given $\alpha = (h_\alpha, g_\alpha, \phi_\alpha) \in \mathbf{FxC}^{\mathrm{ir}}(a, b)$ and $\beta = (h_\beta, g_\beta, \phi_\beta) \in \mathbf{FxC}^{\mathrm{ir}}(c, d)$, we obtain $\alpha \otimes \beta \in \mathbf{FxC}^{\mathrm{ir}}(a \otimes c, b \otimes d)$ as:



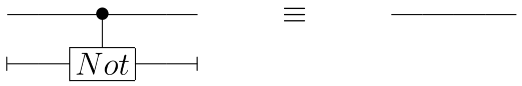
The neutral element of the tensor, i.e. the empty circuit, can be obtained by lifting $I^{\mathbf{FxC}^{\mathrm{ir}}} = \widehat{I^{\mathbf{FxC}^{\mathrm{R}}}}$.

Examples of $\mathbf{FxC}^{\mathrm{ir}}$ *categories*

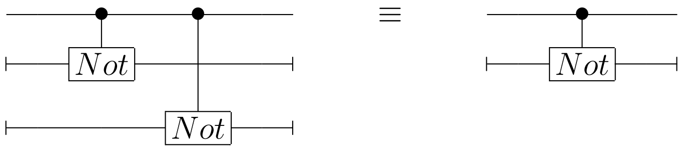
We can now extend our two example $\mathbf{FxC}^{\mathrm{R}}$ categories to $\mathbf{FxC}^{\mathrm{ir}}$ categories. We shall call these **FCC** for the category of finite classical computations, and **FQC** for finite quantum computations. We have informally that $\mathbf{FCC} \simeq$ *Finite Sets*, and that $\mathbf{FQC} \simeq$ *Superoperators*.

4 Equivalence

In the reversible case the equality of definable circuits is the same in the classical case and in the quantum case, but this doesn't hold for irreversible computations. E.g. in the classical case the following two circuits would be equivalent:

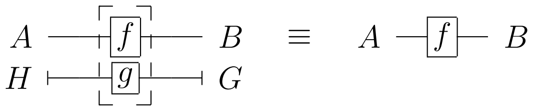


However, this equivalence does not hold when we move into the category of finite quantum computations (**FQC**). This is because, in quantum computation, the control wire (or qubit) can become entangled with the target wire (qubit). All is not lost though as there is another similar equivalence that holds in **FQC** that is (von Neumann's measurement postulate):

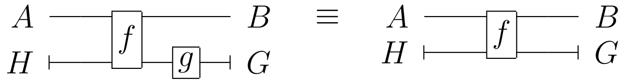


so, how now can we characterise the equivalences which should always hold?

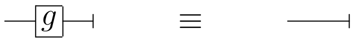
We have come up with three laws to try and characterise these equivalences, that hold in both **FCC** and **FQC**. The first law is that of garbage collection, and it is states that if a circuit can be reduced into two smaller circuits such that one part of the circuit only acts on heap inputs, and on garbage outputs, then that part of the circuit can be removed.



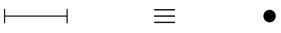
The second law is of the uselessness of garbage processing, and states that if a circuit can be reduced into two smaller circuits such that one part of the circuit only has an effect on garbage outputs, then that can be removed.



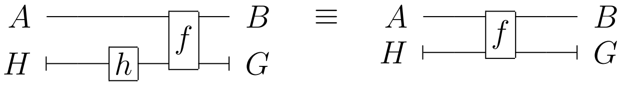
this can be alternately stated as saying that if the only outputs of (part of) a circuit are garbage outputs, then this is equivalent to just having garbage.



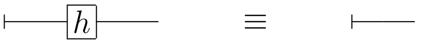
and similarly we can now simplify the first law to state that a wire that simply connects the heap to the garbage is equivalent to having nothing.



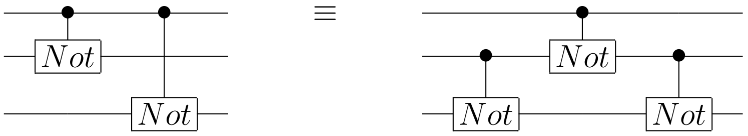
The third law is of the uselessness of heap preprocessing, and states that if a circuit can be reduced into two smaller circuits such that one part of the circuit only has effect on heap inputs, and the effect on the zero vector is the identity, then that part can be removed.
if $h\mathbf{0} = \mathbf{0}$ *then*



the alternate notation for this would again be to state that if (part of) a circuit only has heap inputs, and its effect on the zero vector is the identity, then this is equivalent to just having a heap.
if $h\mathbf{0} = \mathbf{0}$ *then*

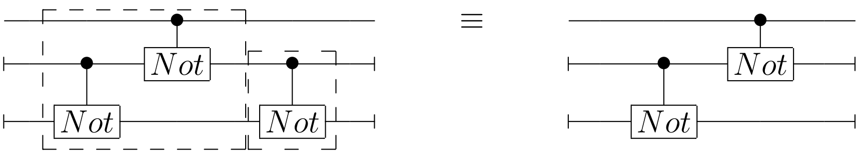


We can already use these laws to give a proof of the measurement postulate. The first step is to show the equivalence of

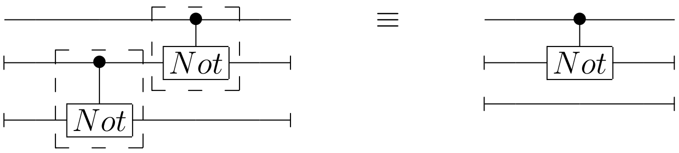


this is simple as you will notice there is no heap or garbage, so we know that the circuits are in $\mathbf{FQC}^{\mathrm{R}}$, and in fact only use the elements from $\mathbf{FCC}^{\mathrm{R}}$ and thus equivalence follows from looking at the truth tables, which are the same.

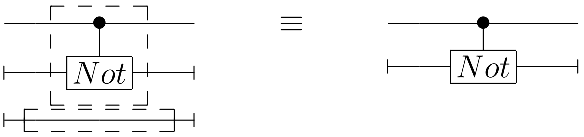
The third controlled not is eliminated using the second law.



The controlled Not's preserve the zero vector so we can eliminate the first one using our third law.



Finally the bottom wire can be removed by use of our first law.



5 Further Work

We are investigating whether we could state the whole development more abstractly using only symmetric strictly bimonoidal categories as the base for the notion of reversible computations. A problem in our current formulation is the last law on heap preprocessing which introduces the precondition that a circuit is 0-preserving. It is not clear how to state this condition abstractly. An alternative would be to drop this condition and to assume that a computation can be carried out provided a correct initialisation. Interestingly our laws would then be symmetric.

Finally, we would like to answer the question whether our laws are complete for quantum computation, i.e. whether we can characterise the equality of definable quantum circuits just by our three laws.

References

[1] S. Abramsky. A structural approach to reversible computation, 2001.

[2] Thorsten Altenkirch and Jonathan Grattage. A functional quantum programming language. In *20th Annual IEEE Symposium on Logic in Computer Science*, 2005.

[3] C. H. Bennett. Logical reversibility of computation. *IBM Journal of Research and Development*, 17(6):525–532, 1973.

[4] Klaus-Jorn Lange, Pierre McKenzie, and Alain Tapp. Reversible space equals deterministic space. In *IEEE Conference on Computational Complexity*, pages 45–50, 1997.

[5] M. Laplaza. Coherence for distributivity. *Lecture Notes in Mathematics*, 281:29–72, 1972.

[6] Peter Selinger. Dagger compact closed categories and completely positive maps. In Peter Selinger, editor, *Proceedings of the 3rd International Workshop on Quantum Programming Languages*, Electronic Notes in Theoretical Computer Science. Elsevier Science, 2005.

[7] John Archibald Wheeler and Richard Phillips Feynman. Interaction with the absorber as the mechanism of radiation. *Rev. Mod. Phys.*, 17(2-3):157–181, Apr 1945.