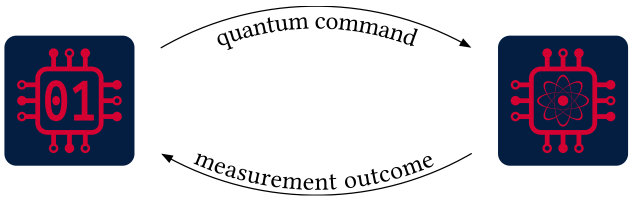
Quantum instruments are a quantum effect monad

Robert I. Booth[1], Dominik Leichtle[2], Alex Rice[2], and Kim Worrall[2]

[1] *University of Oxford, United Kingdom*
[2] *University of Edinburgh, United Kingdom*

In the QRAM model, a classical program instructs an external processor in real time to apply quantum operations which modify quantum states, while measurements return classical data to the program:



Quantum computing under this view is inherently effectful. The general principle of modelling computational effects (such as nondeterminism, local state, and divergence) using monads over a pure functional language [Mog91] has proved foundational in the development of denotational semantics for effectful programs including probabilistic programming languages [SGG15; Sta+16]. This raises the question: can hybrid quantum computation be given a concrete denotational semantics by modelling it as a computational effect added to a purely classical language?

To address this question, we introduce the *quantum instrument monad*, showing that quantum instruments form a strong monad on Set. This monad can be understood as a generalisation to quantum evolutions of the finite distribution monad, cleanly separating the classical and quantum parts of a program's semantics. Quantum instruments are a well-established formalism in quantum information theory [CTU13; MF23; Oza84], which decompose a completely positive trace-preserving (CPTP) map into components indexed by classical outcomes. To our knowledge, their monad structure has not previously been observed.

Existing approaches to the semantics of hybrid quantum computation differ in how classical control and quantum effects are integrated. Measurement has been treated as an algebraic effect over combined quantum-classical programming language [Sta15], added as a classical effect to a purely quantum programming language [HK22], and monadically without a denotational semantics [AG10]. Quantum processes have been given categorical models, such as Selinger's CPM construction [Sel07], which models mixed-state quantum evolution and measurement via completely positive maps in dagger compact categories, or Huot and Staton's result showing that the category of completely positive trace-preserving maps is a canonical completion of the category of isometries [HS21]. The quantum lambda calculus [SV05] uses linear logic to model hybrid quantum computing. Monads have been used to describe circuit-generation in Quipper [Gre+13], and extended to allow mid-circuit measurement [RS18; SCD26]. In contrast, we model quantum computation itself as a monadic effect over a purely classical language, using quantum instruments directly as the denotational structure. Closest to our work, Jia et al. [Jia+22] describe an extension of the probabilistic fixpoint calculus by quantum effects, using the theory of *Kegelspitzen* [KP17] to link the classical part of the language modelled using a valuations monad, and the quantum part modelled with von Neumann algebras. We leave clarifying the connection of their construction to ours as future work.

Beyond the finitely-supported case, we sketch ongoing work in extending the quantum instrument monad to measurable spaces, in analogy with the Giry monad [Gir82], and to directed-complete partial orders (DCPOs), in analogy with the probabilistic powerdomain monad [JP89], so as to allow modelling of classical programs that may diverge.

1 Quantum instruments and measurement

In quantum theory, measurements are typically modelled as a *positive operator-valued measure* (POVM), i.e. a finite set $\{P_i\}$ of positive operators that sum to identity, $\sum_i P_i = 1$, and can be seen as maps from quantum to classical information. Quantum channels, on the other hand, mapping quantum to quantum information, are mathematically captured by CPTP maps. *Quantum instruments* [DL70] unify these objects by describing a more general notion of measurement process, and are given by a finite set $\{\mathcal{E}_i\}$ consisting of completely positive trace non-increasing (CPTNI) channels $\mathcal{E}_i$ that sum to a CPTP channel:

$$\mathrm{tr}\left(\sum_i \mathcal{E}_i(\rho)\right) = \mathrm{tr}(\rho). \tag{1}$$

Any POVM can therefore be realised as a quantum instrument $\mathcal{E}_i(\rho) = P_i \rho P_i$, but one also recovers both unitary and trace-preserving evolutions as singleton instruments, e.g. $\mathcal{E}(\rho) = U\rho U^\dagger$. Given a quantum instrument $\{\mathcal{E}_i\}_{i\in I}$, the map $i \mapsto \mathrm{tr}(\mathcal{E}_i(\rho))$ is a probability distribution on X for every density operator ρ, and any finite probability distribution $\{p_i \in [0,1]\}_{i\in I}$ embeds as the quantum instrument $\mathcal{E}_i(\rho) = p_i \cdot \rho$.

Definition 1. *Given a fixed ambient Hilbert space $\mathcal{H}$, the finite quantum instruments functor $Q : \mathsf{Set} \to \mathsf{Set}$ maps a set X to the set of (finitely supported) quantum instruments indexed by X and acting on $\mathcal{H}$. Given a set function $f : X \to Y$, we define $Qf : QX \to QY$ as $Qf(\{\mathcal{E}_x\}_{x\in X}) \coloneqq \{\sum_{x\in f^{-1}(y)} \mathcal{E}_x\}_{y\in Y}$.*

In analogy to the case of finite distributions, we can give equip this functor with the structure of a monad. The unit of the monad is given by

$$\begin{aligned} \eta_X : X &\longrightarrow QX, \\ x &\longmapsto \{\delta_{x,y} 1_{\mathcal{H}}\}_{y\in X}, \end{aligned} \tag{2}$$

where

$$\delta_{x,y} = \begin{cases} 1 & \text{if} \quad x = y; \\ 0 & \text{otherwise;} \end{cases} \qquad \text{is the Kronecker delta.} \tag{3}$$

The Kleisli extension of $\mathcal{F} : X \to QY$ is given by:

$$\begin{aligned} \mathcal{F}^* : QX &\longrightarrow QY \\ \{\mathcal{E}_x\}_{x\in X} &\longmapsto \left\{ \sum_{x\in X} \mathcal{F}(x)_y \circ \mathcal{E}_x \right\}_{y\in Y} \end{aligned} \tag{4}$$

Theorem 2. *Q forms a strong monad on* Set.

The Kleisli category of Q is a quantum generalisation of the category of stochastic matrices. In fact, morphisms $X \to QY$ are essentially matrices $[\mathcal{E}_{x,y}]$ whose elements are CPTNI maps, and whose columns sum to a CPTP map $\sum_x \mathcal{E}_{x,y}$, i.e. each column describes a quantum instrument.

Remark 3. If we want to model state initialisation and destructive measurements, the CPTNI maps that make up the quantum instrument must have different output/input types. We can capture this by instead as a Hilb-parameterised monad in the sense of [Atk09]: for any Hilbert spaces $\mathcal{H}, \mathcal{J}$ and set X, let $Q_{\mathcal{H},\mathcal{J}}X$ be the set of quantum instruments $\mathcal{H} \to \mathcal{J}$ indexed by X. Then the monad multiplication becomes a family of maps $\mu_X^{\mathcal{H},\mathcal{J},\mathcal{K}} : Q_{\mathcal{J},\mathcal{K}}Q_{\mathcal{H},\mathcal{J}}X \to Q_{\mathcal{H},\mathcal{K}}X$ mirroring composition of the underlying CPTNI maps, and the units are given by maps $\eta_X^{\mathcal{H}} : X \to Q_{\mathcal{H},\mathcal{H}}X$.

While the finite case already yields a strong monad on Set, it is not sufficient for two important reasons: it excludes measurements with infinitely many outcomes, and it does not account for diverging or recursive classical programs. In analogy with classical probability theory, these limitations point respectively toward measure- and domain-theoretic extensions.

2 Beyond finite instruments

Generalising to quantum instruments taking infinitely many values requires significantly more machinery. Recalling the analogy to probability monads, there are two natural candidates to draw inspiration from: the Giry monad on the category of measurable spaces [Gir82], and the probabilistic powerdomain monad on the category of directed-complete partial orders [JP89].

Both constructions require moving from finite summation to integration:

$$\sum_{x \in X} f(x) \cdot p_x \quad \text{becomes} \quad \int_{x \in X} f(x) \, \mathrm{d}p. \tag{5}$$

In the case of the Giry monad, the *Lebesgue* integral plays this role; the probabilistic powerdomain monad uses a *lower* integral on valuations. Comparing with equation (4), we informally observe that we need some kind of *non-commutative integral* to define a multiplication,

$$\sum_{x \in X} \mathcal{F}(x)_y \circ \mathcal{E}_x \quad \text{becomes} \quad \int_{x \in X} \mathcal{F}(x)_y \circ \mathrm{d}\mathcal{E}. \tag{6}$$

Both the integrand $\mathcal{F}$ and "measure" $\mathcal{E}$ are valued in completely positive maps, and the integration must respect the order of composition.

To the best of our knowledge, such non-commutative integrals have not previously appeared in the literature. These questions motivate the development of such constructions: how should we define them, and how much of classical integration theory carries over? In the remainder of this abstract, we detail some work-in-progress for constructing these integrals.

2.1 Towards a quantum Giry monad

The classical setting of probability theory is the Lebesgue theory of measures, in which finite probability distributions are generalised to probability measures. There is an analogous measure-theoretic generalisation of quantum instruments originally due to Davies and Lewis [DL70].

Definition 4. *Let $\mathcal{H}, \mathcal{J}$ be complex Hilbert spaces, then a* quantum instrument $\mathcal{H} \to \mathcal{J}$ *over a measurable space (X, σ_X) is a mapping $\mu : \sigma_X \to \mathsf{CPTNI}(\mathcal{H}, \mathcal{J})$ which is*

- strict*: $\mu(\varnothing) = 0$;*
- normalised*: $\mu(X)$ is trace-preserving;*

- countably additive*: for any countable collection* $\{E_j\} \subseteq \sigma_X$ *of pairwise disjoint sets,* $\mu(\bigcup_j E_j) = \sum_j \mu(E_j)$.

Each of these properties mirror the definition of probability measures. In particular, for any density operator ρ on $\mathcal{H}$, the mapping $\mu_\rho : \sigma_X \to [0,1] : A \mapsto \mathrm{tr}(\mu(A)(\rho))$ defines a probability measure on X. As a result, it is natural to try and construct an integral of the form (6) by "evaluating" on density operators to reduce the problem to the real-valued case:

$$\int_{x\in X} f(x) \circ \mu(\mathrm{d}x) \longmapsto \mathrm{tr}\left(\int_{x\in X} f(x) \circ \mu(\mathrm{d}x)(\rho)\right). \tag{7}$$

The problem is that the trace doesn't commute with the integral, since the quantity $\mathrm{tr}(f(x) \circ \mu(\mathrm{d}x)(\rho))$ is not well-defined. It's therefore not obvious how to relate this integral to the probability measure μ_ρ. In order to do so, we rely on the following result.

Theorem 5 (Informal)**.** *Given a quantum instrument* $\mathcal{H} \to \mathcal{J}$ *over a measurable space* (X, σ_X)*, there is a probability measure* Δ_μ *on* σ_X *and a map* $d_\mu : X \to \mathsf{CPTNI}(\mathcal{H}, \mathcal{J})$ *such that*

$$\mu_\rho(E) = \int_{x\in E} \mathrm{tr}(d_\mu(x)(\rho))\ \Delta_\mu(\mathrm{d}x)), \tag{8}$$

for all density operators ρ *on* $\mathcal{H}$ *and all measurable sets* $E \in \sigma_X$*.*

We can then define the integral of a map $f : X \to \mathsf{CPTNI}(\mathcal{J}, \mathcal{K})$ with respect to μ by setting

$$\mathrm{tr}\left(\int_{x\in X} f(x) \circ \mu(\mathrm{d}x)(\rho)\right) = \int_{x\in X} \mathrm{tr}(f(x) \circ d_\mu(x)(\rho))\ \Delta_\mu(\mathrm{d}x). \tag{9}$$

The structure of the Giry monad can then be syntactically carried over to this setting, but the monad laws require more (ongoing) work to prove.

2.2 A quantum continuations monad on DCPO

Many algorithms, for example Grover's algorithm or variational algorithms [Cer+21], and techniques such as repeat-until-success circuits [BRS15], utilise classical loops. Typically, such loops are modelled by moving to a category which supports fixed points, such as the category of directed complete partial orders (DCPO). Extending the instrument monad to this category immediately requires more than finite instruments, as these are not closed under suprema.

Just as the finite instrument monad generalises the finitely-supported distributions monad, we could expect such DCPO monad to generalise the probabilistic powerdomain monad [JP89], by considering quantum valuations μ on a DCPO X of the form

$$\mu : \mathcal{O}(X) \to \mathsf{CPTNI},$$

where $\mathcal{O}(X)$ is the set of Scott-open subsets of X. Such valuations can be given the usual *stochastic ordering*, where $\mu \le \nu$ if $\mu(U) \le \nu(U)$ for all open sets U, making the set of quantum valuations a DCPO (leveraging that CPTNI is itself a DCPO [Cho16]).

Unfortunately, in contrast to the classical case, it is unclear how to define an integral with the correct order-theoretic properties. When restricted to simple quantum valuations, those that arise from finite instruments, the integral must be given by a summation. It is already unknown whether this summation is monotone in its valuation argument, as the classical case typically uses a max-flow min-cut theorem which does not hold for CPTNI maps.

We solve both the integral definition problem and order-theoretic problems by moving from a valuation monad to a continuation monad

$$\mathcal{Q}(X) = \mathsf{DCPO}(X, \mathsf{CPTNI}) \to \mathsf{CPTNI},$$

noting that any such $\mu \in \mathcal{Q}(X)$ restricts to a valuation by defining the action on an open set U to be $\mu(\chi_U)$ (where χ_U is the indicator function on U).

The definition of the integral has now been repackaged into the data of the monad, delaying the challenge to the definition of primitives. This poses no problem for including finite instruments as primitives, where the definition of the integral is clear.

This also resolves the issue of finding the right ordering of quantum instruments, as the pointwise ordering is coarser than the stochastic ordering. In particular, the pointwise ordering defines $\mu \leq \nu$ exactly when

$$\int f d\mu \leq \int f d\nu \quad \text{for all continuous } f,$$

instead of testing only with indicators of open sets, as is the case with the stochastic ordering. We conjecture that when restricting to the probabilistic case, this order coincides with the stochastic ordering, allowing this continuation monad to be seen as a generalisation of the probabilistic powerdomain monad.

References

[AG10] Thorsten Altenkirch and Alexander S Green. "The quantum IO monad". In: *Semantic Techniques in Quantum Computation* (2010), pp. 173–205.

[Atk09] Robert Atkey. "Parameterised notions of computation". In: *Journal of Functional Programming* 19.3 (July 2009), pp. 335–376. DOI: 10.1017/S095679680900728X.

[BRS15] Alex Bocharov, Martin Roetteler, and Krysta M. Svore. "Efficient Synthesis of Universal Repeat-Until-Success Quantum Circuits". In: *Phys. Rev. Lett.* 114 (8 Feb. 2015), p. 080502. DOI: 10.1103/PhysRevLett.114.080502.

[Cer+21] Marco Cerezo et al. "Variational quantum algorithms". In: *Nature Reviews Physics* 3.9 (2021), pp. 625–644.

[CTU13] Giulio Chiribella, Alessandro Toigo, and Veronica Umanità. "Normal Completely Positive Maps on the Space of Quantum Operations". In: *Open Systems & Information Dynamics* 20.01 (2013), p. 1350003. DOI: 10.1142/S1230161213500030.

[Cho16] Kenta Cho. "Semantics for a quantum programming language by operator algebras". In: *New Generation Computing* 34.1 (2016), pp. 25–68.

[DL70] E Brian Davies and John T Lewis. "An operational approach to quantum probability". In: *Communications in Mathematical Physics* 17.3 (1970), pp. 239–260.

[Gir82] Michèle Giry. "A categorical approach to probability theory". In: *Categorical Aspects of Topology and Analysis*. Ed. by B. Banaschewski. Berlin, Heidelberg: Springer Berlin Heidelberg, 1982, pp. 68–85.

[Gre+13] Alexander S. Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron. "Quipper: a scalable quantum programming language". In: *ACM SIGPLAN Notices* 48.6 (June 2013), pp. 333–342. DOI: 10.1145/2499370.2462177.

[HK22] Chris Heunen and Robin Kaarsgaard. "Quantum information effects". In: *Proc. ACM Program. Lang.* 6.POPL (Jan. 2022). DOI: 10.1145/3498663.

[HS21] Mathieu Huot and Sam Staton. "Quantum channels as a categorical completion". In: *Proceedings of the 34th Annual ACM/IEEE Symposium on Logic in Computer Science*. LICS '19. Vancouver, Canada: IEEE Press, 2021.

[Jia+22] Xiaodong Jia, Andre Kornell, Bert Lindenhovius, Michael Mislove, and Vladimir Zamdzhiev. “Semantics for Variational Quantum Programming”. In: *Proceedings of the ACM on Programming Languages* 6 (POPL Jan. 16, 2022), pp. 1–31. DOI: 10.1145/3498687.

[JP89] C. Jones and G. Plotkin. “A probabilistic powerdomain of evaluations”. In: *Proceedings of the Fourth Annual Symposium on Logic in Computer Science.* Pacific Grove, California, USA: IEEE Press, 1989, pp. 186–195.

[KP17] Klaus Keimel and Gordon D. Plotkin. “Mixed powerdomains for probability and nondeterminism”. In: *Logical Methods in Computer Science* Volume 13, Issue 1 (Jan. 24, 2017). DOI: 10.23638/LMCS-13(1:2)2017.

[MF23] Arindam Mitra and Máté Farkas. “Characterizing and quantifying the incompatibility of quantum instruments”. In: *Phys. Rev. A* 107 (3 Mar. 2023), p. 032217. DOI: 10.1103/PhysRevA.107.032217.

[Mog91] Eugenio Moggi. “Notions of computation and monads”. In: *Inf. Comput.* 93.1 (July 1991), pp. 55–92. DOI: 10.1016/0890-5401(91)90052-4.

[Oza84] Masanao Ozawa. “Quantum measuring processes of continuous observables”. In: *Journal of Mathematical Physics* 25.1 (Jan. 1984), pp. 79–87. DOI: 10.1063/1.526000.

[RS18] Francisco Rios and Peter Selinger. “A Categorical Model for a Quantum Circuit Description Language (Extended Abstract)”. In: *Electronic Proceedings in Theoretical Computer Science* 266 (Feb. 2018), pp. 164–178. DOI: 10.4204/eptcs.266.11.

[SCD26] Ken Sakayori, Andrea Colledan, and Ugo Dal Lago. “On Circuit Description Languages, Indexed Monads, and Resource Analysis”. In: *Proc. ACM Program. Lang.* 10.POPL (Jan. 2026). DOI: 10.1145/3776666.

[SGG15] Adam Scibior, Zoubin Ghahramani, and Andrew D. Gordon. “Practical probabilistic programming with monads”. In: *SIGPLAN Not.* 50.12 (Aug. 2015), pp. 165–176. DOI: 10.1145/2887747.2804317.

[Sel07] Peter Selinger. “Dagger Compact Closed Categories and Completely Positive Maps: (Extended Abstract)”. In: *Electronic Notes in Theoretical Computer Science* 170 (2007). Proceedings of the 3rd International Workshop on Quantum Programming Languages (QPL 2005), pp. 139–163. DOI: https://doi.org/10.1016/j.entcs.2006.12.018.

[SV05] Peter Selinger and Benoît Valiron. “A Lambda Calculus for Quantum Computation with Classical Control”. In: *Typed Lambda Calculi and Applications.* Springer Berlin Heidelberg, 2005, pp. 354–368. DOI: 10.1007/11417170_26.

[Sta15] Sam Staton. “Algebraic Effects, Linearity, and Quantum Programming Languages”. In: *SIGPLAN Not.* 50.1 (Jan. 2015), pp. 395–406. DOI: 10.1145/2775051.2676999.

[Sta+16] Sam Staton, Hongseok Yang, Frank Wood, Chris Heunen, and Ohad Kammar. “Semantics for probabilistic programming: higher-order functions, continuous distributions, and soft constraints”. In: *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science.* LICS ’16. New York, NY, USA: Association for Computing Machinery, 2016, pp. 525–534. DOI: 10.1145/2933575.2935313.