# Concurrent Monads as Monads in a 2-Category

Bjarki Gunnarsson
University of Iceland, Reykjavik, Iceland
bjarkihg@gmail.com

Exequiel Rivas
Tallinn University of Technology, Estonia
exequiel.rivas@taltech.ee

Tarmo Uustalu
Reykjavik University, Iceland
Tallinn University of Technology, Estonia
tarmo@ru.is

We discuss the concurrent monads of Rivas and Jaskelioff, a structure axiomatizing effectful notions of computation supporting not only sequential, but also parallel composition. We have previously analyzed accessible concurrent monads on given ordered monoidal category as concurrent monoid objects in a certain ordered duoidal category. Here we explain the different perspective that concurrent monads are monads in a certain 2-category and what this entails.

Concurrent monads by Rivas and Jaskelioff [3] offer a categorification of concurrent monoids from the work of Hoare et al. [2] on program algebras for Moggi's monad-based analysis of functional programming with effects. Concurrent monads determine not just sequential composition of effectful functions, as monads do, but also parallel composition. The equational and inequational axioms of a concurrent monad stipulate that sequential and parallel composition are unital and associative and obey certain interchange inequations.

Concurrent monads capture various models of sequential-parallel effectful programming with preemptive scheduling as instances, including both interleaving and truly concurrent models, in particular interleaving models for shared memory (collections of traces, resumptions), including also transactional memory for example.

The original work Rivas and Jaskelioff [3] used ordered monads in the sense of monoidal topology; here we adopt the somewhat more principled approach of Rivas and Uustalu [4] based on ordered (=**Poset**-enriched) category theory. A concurrent monad on an ordered monoidal category is given by an ordered monad and ordered lax-monoidal functor that share the underlying ordered functor and satisfy certain inequations of compatibility between the two structures. Explicitly, a concurrent monad on a ordered monoidal category $(\mathbb{C}, \mathsf{I}, \otimes)$ is a 5-tuple $(T, \eta, \mu, \psi^0, \psi)$ where $(T, \eta, \mu)$ is an ordered monad, $(T, \psi^0, \psi)$ is an ordered lax-monoidal functor and the following inequations hold: $\eta_\mathsf{I} \leq \psi^0$, $\mu_\mathsf{I} \circ T\psi^0 \circ \psi^0 \leq \psi^0$, $\eta_{X\otimes Y} \leq \psi_{X,Y} \circ \eta_X \otimes \eta_Y$ and $\mu_{X\otimes Y} \circ T\psi_{X,Y} \circ \psi_{TX,TY} \leq \psi_{X,Y} \circ \mu_X \otimes \mu_Y$. Concurrent monads in this sense are a generalization of lax-monoidal monads (which are the same as commutative monads) for ordered category theory, relaxing their compatibility equations to inequations.

The Kleisli ordered category of (the underlying ordered monad of) a concurrent monad, together with its inclusion ordered functor from the base category, forms what was called in [4] a concurrent category. This is a "typed" generalization of the concept of concurrent monoid. Sequential composition of effectful functions is given by the composition structure of the concurrent category, parallel composition by its lax-functorial ordered monoidal structure; the inclusion ordered functor includes the pure functions among the effectful. As soon as the inclusion ordered functor of a concurrent category has a right adjoint, the concurrent category is in fact the Kleisli ordered category of a concurrent monad.

Submitted to:
MSFP 2026

Monads on a category $\mathbb{C}$ are monoid objects in the monoidal category $([\mathbb{C},\mathbb{C}],\mathsf{I},\otimes)$. It was shown in [4] that accessible concurrent monads on a ordered monoidal category $(\mathbb{C},\mathsf{I},\otimes)$ are concurrent monoid objects in the ordered duoidal category $([\mathbb{C},\mathbb{C}]_{\mathrm{a}},\mathsf{Id},\cdot,\mathsf{Jd},\star)$ where $(\mathsf{Jd},\star)$ is the Day convolution ordered monoidal structure on $[\mathbb{C},\mathbb{C}]_{\mathrm{a}}$ defined by the ordered monoidal structure $(\mathsf{I},\otimes)$ on $\mathbb{C}$. This is analogous to accessible lax-monoidal monads on a monoidal category being duoid objects.

A different perspective on monads, offering further insights, considers them as monad objects in a 2-category. As reported by Gunnarsson [1], this turns out to apply to concurrent monads too. In this talk, we present this observation together with some conclusions that can be drawn from it. We also discuss some questions that it raises.

An ordinary monad on some category is a monad in the 2-category **Cat** of categories, functors and natural transformations. The extra structure present in a concurrent monad can be provided by the richer structure of the ambient 2-category. Namely, a concurrent monad on some ordered monoidal category is precisely a monad in the 2-category $\mathbf{OrdMonCat}_{l,opl}$, with

- as objects ordered monoidal categories,
- as 1-morphisms ordered lax-monoidal functors,
- as 2-morphisms <u>oplax-monoidal</u> natural transformations

where we call a natural transformation $\tau$ between ordered lax-monoidal functors $(F,\psi^0,\psi),(G,\psi^{0\prime},\psi') : (\mathbb{C},\mathsf{I},\otimes)\to(\mathbb{D},\mathsf{I}',\otimes')$ *oplax-monoidal* if $\tau_{\mathsf{I}}\circ\psi^0\le\psi^{0\prime}$ and $\tau_{X\otimes Y}\circ\psi_{X,Y}\le\psi'_{X,Y}\circ\tau_X\otimes'\tau_Y$.

This is analogous to lax-monoidal monads being the same as monads in the 2-category $\mathbf{MonCat}_l$ of monoidal categories, lax-monoidal functors and monoidal natural transformations. The difference is that we need to switch from ordinary to ordered category theory and the 2-morphisms need to be relaxed from monoidal to oplax-monoidal: exactly this relaxes the compatibility equations of a lax-monoidal monad to the compatibility inequalities of a concurrent monad.

Differently from the case of lax-monoidal monads, where the Kleisli category of the underlying monad is monoidal and the Kleisli adjunction lax-monoidal, the Kleisli ordered category of the underlying ordered monad of a concurrent monad and the associated ordered adjunction do not fit into the same 2-category as the concurrent monad itself. To also accommodate them as well, we need to move to a larger 2-category. This has

- as objects <u>lax-functorially</u> ordered monoidal categories,
- as 1-morphisms <u>oplax-naturally</u> ordered lax-monoidal functors,
- as 2-morphisms oplax-monoidal natural transformations

where lax-functorial monoidality of $(\mathbb{C},\mathsf{I},\otimes)$ requires $\mathsf{I}$, $\otimes$ to be lax-functors and oplax-natural lax-monoidality of $(F,\phi^0,\phi)$ requires $\phi^0$ and $\phi$ to be oplax-natural transformations. (We rely on the definitions that $F$ is a lax-functor if $\mathsf{id}_{FX}\le F\mathsf{id}_X$ and $Fg\circ Ff\le F(g\circ f)$ and that $\tau : F\to G$ is an oplax-natural transformation if $Gf\circ\tau_X\le\tau_Y\circ Ff$ for $f : X\to Y$.) It is not clear to us yet which the right (tightest) statement is for the coherence laws for such structures.

It is known that the Eilenberg-Moore category of the underlying monad of a lax-monoidal monad is monoidal under reasonable conditions [5]. We conjecture that the Eilenberg-Moore ordered category of the underlying ordered monad of a concurrent monad is lax-functorially ordered monoidal under some adjusted conditions and are working to identify those.

**Acknowledgement** This work was supported by the Iceproof project of the Collaboration Fund of the Icelandic Ministry of Culture, Innovation and Higher Education and by the Icelandic Research Fund grant 228684-053.

# References


[1] B. Gunnarsson. Concurrent monads in 2-categories. MSc thesis, University of Iceland, 2026. `https://hdl.handle.net/1946/52063`

[2] C. A. R. Hoare, B. Möller, G. Struth, I. Wehrman. Concurrent Kleene algebra and its foundations. *J. Log. Algebraic Program.* 80(6), pp. 266–296, 2011. doi:10.1016/j.jlap.2011.04.005

[3] E. Rivas, Jaskelioff. Monads with merging. HAL eprint hal-02150199. `https://inria.hal.science/hal-02150199`

[4] E. Rivas, T. Uustalu. Concurrent monads for shared state. In *Proc. of 26th Int. Symp. on Principles and Practice of Declarative Programming, PPDP '24 (Milan, Sept. 2024)*, *ACM Int. Conf. Proc. Series*, art. 19, 13 pp. ACM Press, 2024. doi:10.1145/3678232.3678249

[5] G. Seal. Tensors, monads and actions. *Theory Appl. Categ.* 28(15), pp. 403–434. `http://www.tac.mta.ca/tac/volumes/28/15/28-15abs.html`