Additive Cellular Automata Graded-Monadically

Silvio Capobianco
silvio@cs.ioc.ee
Tallinn University of Technology
Estonia

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

ABSTRACT

Cellular automata are an archetypical comonadic notion of computation in that computation happens in the coKleisli category of a comonad. In this paper, we show that they can also be viewed as graded comonadic—a perspective that turns out to be both more informative and also more basic. We also discuss additive cellular automata to show that they admit both a graded comonadic and a graded monadic view. That these two perspectives are simultaneously available in this special case arises from a graded version of an observation by Kleiner about adjoint comonad-monad pairs.

CCS CONCEPTS

• **Theory of computation** → **Categorical semantics**; **Models of computation**; • **Computer systems organization** → **Cellular architectures**.

KEYWORDS

models of computation, cellular automata, additive cellular automata, comonads, adjoint comonad-monad pairs, graded comonads, graded monads

1 INTRODUCTION

This paper is about cellular automata as a non-purely functional notion of computation to showcase the potential of graded comonads in mathematical semantics and also of adjoint ungraded and graded comonad-monad pairs.

The standard approach to non-purely functional notions of computation is to model their notion of function in some category other than **Set**, often the Kleisli category of some (strong) monad on **Set** or on some other relevant category, e.g., **Cpo**. Such notions of computation can be usefully thought of as "effectful"—in addition to a return value or values, a function produces an "effect".

Notions of computation where a function is a Kleisli map of a suitable (strong) monad are numerous; they are very well known and understood.

Some notions of computation however cannot be modelled in this way, but can be modelled in the coKleisli category of a (lax monoidal) comonad. Intuitively, these notions are "context-dependent"—in addition to the argument, the function consumes some "context". An elegant example of this situation is Lustre-style causal dataflow computation where functions are causal stream functions or, equivalently, coKleisli maps of the nonempty list comonad. A function's return value "now" depends not only on the argument's value "now", but also at some "past instants".

Another neat example of a comonadic notion of computation is that of cellular automata. Here the notion of function is of a uniformly continuous transformation of configurations, the latter being assignments of letters from some alphabet (values from some set) to nodes in a homogeneous grid. A function's return value "here" depends not only on the argument's value "here", but also at some "nearby nodes". Such functions are coKleisli maps of the cowriter comonad on the category **Unif** of uniform spaces, which are a special case of topological spaces for which a good concept of uniformly continuous function arises without introduction of a metric.

In this paper, we revisit cellular automata (CA) in order to make two new points. We show that, in addition to the modelling in the coKleisli category of the cowriter comonad on **Unif**, worked out in [5], they admit an explicitly "resource-aware" modelling in the coKleisli locally graded category of a graded version of this comonad on **Set**. Then we turn to additive cellular automata, whose alphabets have the algebraic structure of commutative monoids and the automaton has to preserve this structure. We show that they can be modelled both in the coKleisli locally graded category of a graded comonad as well as in the Kleisli locally graded category of a graded monad, the graded monad being "pointwise" right adjoint to the graded comonad. The relevant graded monad is that of formal polynomials.

The organization of this paper is the following. First we recall the comonadic approach to ordinary cellular automata, also recalling comonads and the coKleisli and coEilenberg-Moore constructions. Then we refine it into the graded comonadic approach. This requires introduction of graded comonads and some locally graded category theory. Finally we switch to additive CA. We prove a graded version of Kleiner's [15] observation about adjoint (ungraded) comonads and monads and show that additive CA are both graded comonadic and graded monadic.

2 CELLULAR AUTOMATA AS COMONADIC

2.1 Cellular automata

Cellular automata (CA) [6, 13] are a model of computation operating on a homogeneous grid of nodes. A configuration is a labelling of these nodes with letters from some alphabet. When a CA is applied to a configuration, values at all nodes are updated synchronously and in a similar manner depending on some nodes nearby the current node, resulting in a new configuration.

PPDP 2023, October 22–23, 2023, Lisboa, Portugal
 The definitive Version of Record was published in *International Symposium on Principles and Practice of Declarative Programming (PPDP 2023), October 22–23, 2023, Lisboa, Portugal*, https://doi.org/10.1145/3610612.3610625.

We use a somewhat more liberal definition of cellular automaton than is traditional in the area. We take a *cellular automaton* to be parameterized by a monoid $G = (G, 1_G, \cdot)$ (the *grid*, not necessarily a group) and sets X and Y (the *input and output alphabets*, not necessarily finite, not necessarily the same). Further, the main data of a cellular automaton is a local rule.

A *local rule* is a function $k : X^G \to Y$ determining the output configuration letter $k\,c \in Y$ at node 1_G for a given input configuration $c \in X^G$. This function k is required to be *uniformly continuous*: there must exist a finite set $M \subseteq G$ (called a *neighborhood*) such that, for all $c, c' \in X^G$, we have $k\,c = k\,c'$ as soon as c and c' agree on M, i.e., $c\,m = c'\,m$ for all $m \in M$.[1] In other words, $k\,c$ must for any c be determined by the X^M part of c (written $c|_M$), i.e., k must factor through X^M. (Of course, if M is a neighborhood for a cellular automaton, then so is also any finite $M' \supseteq M$.)

Alternatively, instead of a local rule, a CA can be specified with a global rule.

A *global rule* is a function $f : X^G \to Y^G$, returning, for a given input configuration $c \in X^G$, the whole output configuration $f\ c \in Y^G$. It must *commute with translations* in the sense that $f\ (c \rhd_X n) = f\ c \rhd_Y n$ (for $c \in X^G$, $n \in G$) where the family of functions $\rhd_X : X^G \times G \to X^G$ (translation) is defined by $c \rhd_X n = \lambda m \in G.\, c\ (n \cdot m)$ and be *uniformly continuous* in the sense of existence of a finite $M \subseteq G$ such that, for any input configuration $c \in X^G$ and node $n \in G$, the output configuration letter $f\ c\ n \in Y$ at n is determined by the $X^{\{n\} \cdot M}$ part of c.

The Curtis-Hedlund theorem [11] (cf. [6, Theorem 1.8.1]; the version that is adequate for possibly infinite alphabets) states that local rules and global rules as defined above are in bijection, one can equivalently use either a local or a global rule to specify a CA. The bijection is this: the global rule for a local rule k is $f\ c = \lambda n.\, k\ (c \rhd_X n)$; the local rule for a global rule f is $k\ c = f\ c\ 1_G$.

As an example, let us consider Wolfram's Rule 30 (=00011110). It has $(G, 1_G, \cdot) = (\mathbb{Z}, 0, +)$ (the free group on one generator; this group is commutative) as the grid, We use $X = \mathsf{Bool}$ as both the input as well as the output alphabet. The local rule is $k\ c = c\ (-1)\ \mathsf{xor}\ (c\ 0 \vee c\ 1)$. This uses the neighborhood $M = \{-1, 0, 1\}$ (but any finite superset works too). The global rule is $f\ c = \lambda n.\, c\ (n-1)\ \mathsf{xor}\ (c\ n \vee c\ (n+1))$.

Here is the result of iterating this CA on a particular configuration (namely $c = \lambda n.\, (n = 0)$):

...	−4	−3	−2	−1	0	+1	+2	+3	+4	...
...	0	0	0	0	1	0	0	0	0	...
...	0	0	0	1	1	1	0	0	0	...
...	0	0	1	1	0	0	1	0	0	...
...	0	1	1	0	1	1	1	1	0	...
...	1	1	0	0	1	0	0	0	1	...

$\vdots$

Consider also $G = (\{a, b\}^*, [], +\!\!+)$ (words over the alphabet $\{a, b\}$, the free monoid on two generators). A good visualization of this monoid as a grid is provided by its Cayley graph. Use $X = \mathsf{Bool}$ as the input and output alphabet. An input or output configuration is then a labelling of this Cayley graph with Booleans. Let the local rule be $k\,c =$ at least 3 of $\{c\ [], c\ a, c\ (aa), c\ (ab), c\ b\}$ hold. This uses the neighborhood $\{[], a, aa, ab, b\}$. The corresponding global rule is $f\ c = \lambda w.$ at least 3 of $\{c\ w, c\ (wa), c\ (waa), c\ (wab), c\ (wb)\}$ hold.

Traditionally, the input and output alphabet of a CA are required to be the same. This allows the CA to be composed with itself (the global rule to be self-composed as a function). In the formulation that we use here, two CA can be composed if the output alphabet of the first and the input alphabet of the second coincide. The global rule of the composition is given by function composition of the two given global rules. The definition that allows for different input and output alphabets is more general; in this paper, we have no reason to restrict it. Also, traditionally, the grid is required to be a group, but we will have no use for the operation of inverse. Therefore we only ask for a monoid.

2.2 Comonads

A *comonad* on a category $\mathcal{C}$ is a functor $D : \mathcal{C} \to \mathcal{C}$ together with natural transformations $\varepsilon : D \to \mathsf{Id}$ (the *counit*) and $\delta : D \to D \cdot D$ (the *comultiplication*) such that

$$\begin{tikzcd} D \arrow[r, "\delta"] \arrow[d, "\delta"'] \arrow[dr, equal] & D \cdot D \arrow[d, "\varepsilon \cdot D"] \\ D \cdot D \arrow[r, "D \cdot \varepsilon"'] & D \end{tikzcd} \qquad \begin{tikzcd} D \arrow[r, "\delta"] \arrow[d, "\delta"'] & D \cdot D \arrow[d, "\delta \cdot D"] \\ D \cdot D \arrow[r, "D \cdot \delta"'] & D \cdot D \cdot D \end{tikzcd}$$

Equivalently but shorter, a *comonad* $\mathcal{C}$ is a comonoid in the strict monoidal category $[\mathcal{C}, \mathcal{C}]$ of endofunctors with composition as the tensor.

The *coKleisli category* $\mathbf{CoKl}(D)$ of a comonad $D = (D, \varepsilon, \delta)$ on $\mathcal{C}$ has as objects objects of $\mathcal{C}$ and as maps from X to Y maps $DX \to Y$ of $\mathcal{C}$. The identity on X is $\varepsilon_X : DX \to X$ and the composition of $k : DX \to Y$ and $\ell : DY \to Z$ is $\ell \circ k^\dagger : DX \to Z$ where $k^\dagger = Dk \circ \delta_X : DX \to DY$ (the *coKleisli extension* of k).

A *coalgebra* of a comonad D on $\mathcal{C}$ is an object X of $\mathcal{C}$ together with a map $\xi : X \to DX$ of $\mathcal{C}$ such that

$$\begin{tikzcd} X \arrow[r, "\xi"] \arrow[dr, equal] & DX \arrow[d, "\varepsilon_X"] \\ & X \end{tikzcd} \qquad \begin{tikzcd} X \arrow[r, "\xi"] \arrow[d, "\xi"'] & DX \arrow[d, "\delta_X"] \\ DX \arrow[r, "D\xi"'] & DDX \end{tikzcd}$$

A *map between coalgebras* (X, ξ) and (Y, χ) is a map $f : X \to Y$ such that

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "\xi"'] & Y \arrow[d, "\chi"] \\ DX \arrow[r, "Df"'] & DY \end{tikzcd}$$

The *coEilenberg-Moore category* $\mathbf{CoEM}(D)$ of D has as objects coalgebras of D and as maps coalgebra maps, with the identities and composition inherited from $\mathcal{C}$.

A *resolution* of a comonad D on $\mathcal{C}$ is a category $\mathcal{D}$ with functors $R : \mathcal{C} \to \mathcal{D}$ and $L : \mathcal{D} \to \mathcal{C}$ such that L is left adjoint to R, $D = L \cdot R$, $\varepsilon = \mathsf{e}$ and $\delta = L \cdot \mathsf{h} \cdot R$ where e and h are the counit and unit of the adjunction. A resolution map between $(\mathcal{D}, R, L)$ and $(\mathcal{D}', R', L')$ is

[1] Uniform continuity is strictly stronger than continuity, which requires that, for any $c \in X^G$, there exists a finite set $M \subseteq G$ (generally depending on c) such that, for all $c' \in X^G$, one has $k\,c = k\,c'$ when $c|_M = c'|_{M'}$.

a functor $F : \mathcal{D} \to \mathcal{D}'$ such that $R = R' \cdot F$ and $F \cdot L = L'$. The resolutions of a comonad form a category.

The initial resolution is given by the coKleisli category $\mathbf{CoKl}(D)$ with functors J, K where $JX = X$, $Jf = f \circ \varepsilon$, $KX = DX$, $Kk = k^\dagger$. The final resolution is provided by the coEilenberg-Moore category $\mathbf{CoEM}(D)$ with functors F, U where $FX = (DX, \delta_X)$ (the cofree coalgebra of D on X), $Ff = Df$, $U(X, \xi) = X$, $Uf = f$. The unique map between these two resolutions is the functor $E : \mathbf{CoKl}(D) \to \mathbf{CoEM}(D)$ defined on objects by $EX = (DX, \delta_X)$ and maps by $Ek = k^\dagger$. This functor is fully-faithful. Hence, $\mathbf{CoKl}(D)$ is isomorphic to the full subcategory of $\mathbf{CoEM}(D)$ corresponding to cofree coalgebras.

2.3 CA as comonadic

The approach of Capobianco and Uustalu [5] (and also of Piponi [23]) to CA as comonadic is in its basic form based on the set-theoretic *cowriter comonad* for the monoid G. This is the comonad $D = (D, \varepsilon, \delta)$ on **Set** defined by

- $DX = X^G$,
  $D(f : X \to Y)(c \in X^G) = f \circ c \in Y^G$,
- $\varepsilon_X(c \in X^G) = c\,1_G \in X$,
- $\delta_X(c \in X^G) = \lambda n \in G.\, c \rhd_X n$
  $= \lambda n \in G.\, \lambda m \in G.\, c\,(n \cdot m) \in (X^G)^G$.

CoKleisli maps of this comonad D are functions $k : X^G \to Y$, i.e., precisely CA local rules, except for the uniform continuity requirement. Maps between cofree coalgebras of D are functions $f : X^G \to Y^G$ such that $f\,(c \rhd_X n) = f\,c \rhd_Y n$, i.e., precisely CA global rules, again except for the uniform continuity requirement. Moreover, these two types of maps compose precisely as CA local resp. global rules compose. We noted above that the coKleisli category and the full subcategory of the coEilenberg-Moore category given by the cofree coalgebras are isomorphic for any comonad on any category. This means that, once we accept the categorical perspective, the Curtis-Hedlund theorem is immediate from some generalities.

This modelling of CA with a comonad on **Set** has the flaw that it ignores the uniform continuity requirement on CA local and global rules.

Capobianco and Uustalu [5] showed that this issue can be fixed in a principled way by switching from **Set** as the base category to the category **Unif** of uniform spaces, endowing X^G (the G-fold product of the underlying set of a given uniform space X with itself) with the product uniformity induced by the uniformity of X. This move solves the problem readily. Classical CA local and global rules amount precisely to coKleisli maps resp. cofree coalgebra maps for X, Y discrete of the cowriter comonad for G on **Unif**. But there is nothing forbidding one to contemplate also CA with non-discrete alphabets X, Y.

3 CELLULAR AUTOMATA AS GRADED COMONADIC

An alternative way to incorporate uniform continuity, which we promote here in this paper, is to be constructive about "there being an M such that…" in the definition of CA: to use a particular M as a data rather than existence of some M as a property in this definition.

This move allows one to remain in **Set** at the expense of switching from a comonad to a graded comonad. Here by graded comonads we mean the concept dual to the graded monads of Smirnov [25], Katsumata [14] and Melliès [18]. For the coKleisli and coEilenberg-Moore constructions for graded comonads, we adopt the approach of McDermott and Uustalu [17], for graded monads, based on locally graded category theory.

3.1 Graded comonads

A *preordered set* $\mathcal{M}$ is a set $|\mathcal{M}|$ with a preorder, i.e., a reflexive and transitive binary relation $\leq$, in other words a thin category (between any two objects there is at most one map). A *preordered monoid* is a preordered set whose underlying set carries a monoid structure $(1, \cdot)$ with $\cdot$ monotone wrt. $\leq$ in both arguments, in other words a thin strictly monoidal category.

Given a preordered monoid $\mathcal{M} = (|\mathcal{M}|, \leq, 1, \cdot)$, an $\mathcal{M}$-*graded comonad* is given by

- a family of functors $D_M : \mathcal{C} \to \mathcal{C}$ functorial in M together with
- a family of natural transformations $D_{M \leq M'} : D_M \to D_{M'}$,
- a natural transformation $\varepsilon : D_1 \to \mathsf{Id}$ and
- a family of natural transformations $\delta_{N,M} : D_{N \cdot M} \to D_N \cdot D_M$ natural in N, M

such that

$$\begin{tikzcd} D_M \arrow[r, "\delta_{1,M}"] \arrow[d, "\delta_{M,1}"'] \arrow[dr, equal] & D_1 \cdot D_M \arrow[d, "\varepsilon \cdot D_M"] \\ D_M \cdot D_1 \arrow[r, "D_M \cdot \varepsilon"'] & D_M \end{tikzcd} \qquad \begin{tikzcd} D_{P \cdot N \cdot M} \arrow[r, "\delta_{P \cdot N, M}"] \arrow[d, "\delta_{P, N \cdot M}"'] & D_{P \cdot N} \cdot D_M \arrow[d, "\delta_{P,N} \cdot D_M"] \\ D_P \cdot D_{N \cdot M} \arrow[r, "D_P \cdot \delta_{N,M}"'] & D_P \cdot D_N \cdot D_M \end{tikzcd}$$

Functoriality of D_M in M means that

$$D_{M \leq M} = \mathsf{id}_{D_M} \qquad D_{M' \leq M''} \circ D_{M \leq M'} = D_{M \leq M''}$$

and naturality of $\delta_{N,M}$ in N and M means that

$$\begin{tikzcd} D_{N \cdot M} \arrow[r, "D_{N \cdot M \leq N' \cdot M'}"] \arrow[d, "\delta_{M,N}"'] & D_{N' \cdot M'} \arrow[d, "\delta_{N', M'}"] \\ D_N \cdot D_M \arrow[r, "D_{N \leq N'} \cdot D_{M \leq M'}"'] & D_{N'} \cdot D_{M'} \end{tikzcd}$$

A more concise definition is that an $\mathcal{M}$-*graded comonad* on a category $\mathcal{C}$ is an oplax monoidal functor from $\mathcal{M}$ as a strict monoidal category to the strict monoidal category $[\mathcal{C}, \mathcal{C}]$. Equivalently, one can also say that it is a oplax monoidal action of $\mathcal{M}$ on $\mathcal{C}$.[2]

A comonad is a **1**-graded comonad where **1** is the singleton preordered monoid.

To talk about the coKleisli, coEilenberg-Moore constructions and resolutions, we need locally graded categories. These were introduced by Wood [28] under the name of wide categories (see also Levy [16]).

Suppose given a preordered monoid $\mathcal{M} = (|\mathcal{M}|, \leq, 1, \cdot)$. A *locally* $\mathcal{M}$-*graded category* is given by:

- a set $|\mathcal{C}|$ of objects;

[2]These definitions make perfect sense also if $\mathcal{M}$ is a general monoidal category. But restricting to a preordered monoid simplifies the unpacking of the compact definitions significantly and is general enough for our purposes here.

- for any $M \in |\mathcal{M}|$ and $X, Y \in |\mathcal{C}|$, a set $\mathcal{C}_M(X, Y)$ of maps of grade M (we write $f : X \to_M Y$ for $f \in \mathcal{C}_M(X, Y)$);
- if $M' \le M$, then, for any map $f : X \to_{M'} Y$, a map $(M' \le M)^* f : X \to_M Y$ (the coercion);
- for any $X \in |\mathcal{C}|$, a map $\mathrm{id}_X : X \to_1 X$ (the identity);
- for any maps $f : X \to_M Y$, $g : Y \to_N Z$, a map $g \circ f : X \to_{M \cdot N} Z$ (the composition)

such that

- $(M \le M)^* f = f$,
  $(M'' \le M)^* f = (M' \le M)^*((M'' \le M')^* f)$,
- $f \circ \mathrm{id} = f = \mathrm{id} \circ f$,
  $h \circ (g \circ f) = (h \circ g) \circ f$,
- $(N' \le N)^* g \circ (M' \le M)^* f = (M' \cdot N' \le M \cdot N)^*(g \circ f)$.

A *functor* between two locally $\mathcal{M}$-graded categories $\mathcal{C}$ and $\mathcal{D}$ is a mapping $F : |\mathcal{C}| \to |\mathcal{D}|$ with, for any $M \in |\mathcal{M}|$, $X, Y \in |\mathcal{C}|$, a mapping $F : \mathcal{C}_M(X, Y) \to \mathcal{D}_M(FX, FY)$ such that $F\mathrm{id}_X = \mathrm{id}_{FX}$ and $F(g \circ f) = Fg \circ Ff$. A *natural transformation* between functors F, G between locally $\mathcal{M}$-graded categories $\mathcal{C}, \mathcal{D}$ is, for any $X \in |\mathcal{C}|$, a map $\tau_X : FX \to_1 GX$ of $\mathcal{D}$ such that, for any map $f : X \to_M Y$ of $\mathcal{C}$, one has $Gf \circ \tau_X = \tau_Y \circ Ff$.

For a preordered monoid $\mathcal{M} = (|\mathcal{M}|, \le, 1, \cdot)$, we write $\mathcal{M}^{\mathrm{oprev}}$ for the preordered monoid $(|\mathcal{M}|, \ge, 1, \cdot^{\mathrm{rev}})$ where $\ge$ is the converse of $\le$ and $M \cdot^{\mathrm{rev}} N = N \cdot M$.

The *coKleisli locally $\mathcal{M}^{\mathrm{oprev}}$-graded category* $\mathbf{CoKl}(D)$ of an $\mathcal{M}$-graded comonad $D = (D, \varepsilon, \delta)$ on $\mathcal{C}$ has as objects objects of $\mathcal{C}$ and as maps of grade M from X to Y maps $D_M X \to Y$ of $\mathcal{C}$. The coercion $X \to_M Y$ of $k : X \to_{M'} Y$ along $M' \ge M$ is $k \circ D_{M \le M', X} : D_M X \to Y$ (note that $k : D_{M'} X \to Y$). The identity on X is $\varepsilon_X : D_1 X \to X$ and the composition $X \to_{M \cdot^{\mathrm{rev}} N} Z$ of $k : X \to_M Y$ and $\ell : Y \to_N Z$ is $\ell \circ k^\dagger_N : D_{N \cdot M} X \to Z$ where $k^\dagger_N = D_N k \circ \delta_{N,M,X} : D_{N \cdot M} X \to D_N Y$ (the coKleisli extension of k) (note that $k : D_M X \to Y$ and $\ell : D_N Y \to Z$).

A *coalgebra* of D is a functor X from $\mathcal{M}$ to $\mathcal{C}$ together with a family of maps $\xi_{N,M} : X_{N \cdot M} \to D_N X_M$ of $\mathcal{C}$ natural in N, M such that

$$
\begin{tikzcd}
X_M \arrow[r, "\xi_{1,M}"] \arrow[dr, equal] & D_1 X_M \arrow[d, "\varepsilon_{X_M}"] \\
& X_M
\end{tikzcd}
\qquad
\begin{tikzcd}
X_{P \cdot N \cdot M} \arrow[r, "\xi_{P \cdot N, M}"] \arrow[d, "\xi_{P, N \cdot M}"'] & D_{P \cdot N} X_M \arrow[d, "\delta_{P,N,X_M}"] \\
D_P X_{N \cdot M} \arrow[r, "D_P \xi_{N,M}"] & D_P D_N X_M
\end{tikzcd}
$$

A *map* of grade M *between coalgebras* (X, ξ) and (Y, χ) is a family of maps $f_N : X_{N \cdot M} \to Y_N$ of $\mathcal{C}$ natural in N such that

$$
\begin{tikzcd}
X_{P \cdot N \cdot M} \arrow[r, "f_{P \cdot N}"] \arrow[d, "\xi_{P, N \cdot M}"'] & Y_{P \cdot N} \arrow[d, "\chi_{P,N}"] \\
D_P X_{N \cdot M} \arrow[r, "D_P f_N"] & D_P Y_N
\end{tikzcd}
$$

The *coEilenberg-Moore locally $\mathcal{M}^{\mathrm{oprev}}$-graded category* $\mathbf{CoEM}(D)$ of an $\mathcal{M}$-graded comonad D on $\mathcal{C}$ has as objects coalgebras of D and as maps of grade M coalgebra maps of grade M. The N-component of the coercion $(X, \xi) \to_M (Y, \chi)$ of $f : (X, \xi) \to_{M'} (Y, \chi)$ along $M' \ge M$ is $f_N \circ X_{N \cdot M \le N \cdot M'} : X_{N \cdot M} \to Y_N$ (note that $f_N : X_{N \cdot M'} \to Y_N$). The identity on (X, ξ) is id_X. The P-component of the composition $(X, \xi) \to_{N \cdot^{\mathrm{rev}} M} (Z, \zeta)$ of $f : (X, \xi) \to_M (Y, \chi)$ and $g : (Y, \chi) \to_N (Z, \zeta)$ is $g_P \circ f_{P \cdot N} : X_{P \cdot N \cdot M} \to Z_P$ (note that $f_{P \cdot N} : X_{P \cdot N \cdot M} \to Y_{P \cdot N}$ and $g_P : Y_{P \cdot N} \to Z_P$).

We omit the definition of the concept of resolution of a graded comonad, but the coKleisli and coEilenberg-Moore locally graded categories of D form its initial and final resolutions. The unique resolution map between them is the locally graded functor $E : \mathbf{CoKl}(D) \to \mathbf{CoEM}(D)$ defined on objects by $EX = (D_{-}X, \delta_{-,-,X})$, on maps of grade M by $E(k : X \to_M Y) = k^\dagger : (D_{-}X, \delta_{-,-,X}) \to_M (D_{-}Y, \delta_{-,-,Y})$ (note that $k : D_M X \to Y$, so $k^\dagger_N : D_{N \cdot M} X \to D_N Y$).

We finish this minimal introduction to graded comonads and locally graded category theory with three remarks.

First, locally graded categories are a special case of enriched categories. If $\mathcal{M}$ is small, then a locally $\mathcal{M}$-graded category is the same thing as a category enriched in $[\mathcal{M}, \mathbf{Set}]$, for a particular monoidal structure on $[\mathcal{M}, \mathbf{Set}]$ given by Day convolution.

Second, the fact that the coKleisli and coEilenberg-Moore constructions for graded comonads need locally graded category theory is motivated by the fact that graded comonads are a special case of relative comonads in the sense of locally graded category theory. For details, worked out for graded monads rather than graded comonads, see McDermott and Uustalu [17].

Third, the $^{\mathrm{oprev}}$'s are a bit of annoyance. They would go away if we took an $\mathcal{M}$-graded comonad to be an oplax monoidal functor from $\mathcal{M}^{\mathrm{oprev}}$ (instead of $\mathcal{M}$) to $[\mathcal{C}, \mathcal{C}]$; this would change the type of $D_{M \le M'}$ to $D_{M'} \to D_M$ and that of $\delta_{N,M}$ to $D_{M \cdot N} \to D_N \cdot D_M$. The coKleisli and coEilenberg-Moore categories of an $\mathcal{M}$-graded comonad D would then be locally $\mathcal{M}$-graded and D itself would be a relative comonad in the locally $\mathcal{M}$-graded categorical sense. But the existing works [4, 18, 21] on graded comonads (calling them positive actions, indexed comonads or parameterized comonads) have made the other choice. (One could also change the types of coercion and composition in the definition of locally $\mathcal{M}$-graded category. We are using the definition from [17], which is local $\mathcal{M}^{\mathrm{op}}$-gradedness according to [16, 28].)

3.2 CA as graded comonadic

We use a graded version of the cowriter comonad for G.

From the monoid $G = (G, 1_G, \cdot)$, we build a preordered monoid $\mathcal{M} = (|\mathcal{M}|, \le, 1, \cdot)$ by taking $|\mathcal{M}| = \mathcal{P}_{\mathrm{f}}(G)$, $\le \; = \; \supseteq$, $1 = \{1_G\}$, $N \cdot M = \{n \cdot m \mid n \in N, m \in M\}$.

Then we define an $\mathcal{M}$-graded comonad $D = (D, \varepsilon, \delta)$ on $\mathbf{Set}$ by

- $D_M X = X^M$,
  $D_M(f : X \to Y)(c \in X^M) = f \circ c \in Y^M$,
- $\varepsilon_X(c \in X^1) = c\, 1_G \in X$,
- $\delta_{N,M,X}(c \in X^{N \cdot M}) = \lambda n \in N.\, c \rhd_{N,M,X} n$
  $= \lambda n \in N.\, \lambda m \in M.\, c\,(n \cdot m) \in (X^M)^N$,
- $D_{M \le M', X}(c \in X^M) = c|_{M'}$
  $= \lambda m \in M'.\, c\, m \in X^{M'}$ (note that $M' \subseteq M$).

Here $\rhd_{N,M,X} : X^{N \cdot M} \times N \to X^M$ is defined by $c \rhd_{N,M,X} n = \lambda m \in M.\, c\,(n \cdot m)$; this is a resource-aware version of translation. The configuration $D_{M \le M', X}\; c$ is defined as the restriction of c.

Differently from the ungraded cowriter comonad case, coKleisli and cofree coalgebra maps of the graded cowriter comonad are not exactly the same as CA local and global rules like we defined them above. Yet they are still very close.

CoKleisli maps of grade M are functions $k : X^M \to Y$. These have X^M instead of X^G as the domain: this is for the relevant part of an input configuration instead of all of it. Local rules are often defined in this equivalent format.

Cofree coalgebra maps of grade M are families of functions $f_N : X^{N \cdot M} \to Y^N$ that are natural in N (which says just that, if $N' \subseteq N$, then $(f_N\, c)|_{N'} = f_{N'}\, (c|_{N' \cdot M}) \in Y^{N'}$ for $c \in X^{N \cdot M}$) and satisfy the coalgebra homomorphism equation

$$f_N\, (c \rhd_{P, N \cdot M, X}\, p) = f_{P \cdot N}\, c \rhd_{P, N, Y}\, p \in Y^N$$

for $c \in X^{P \cdot N \cdot M}$, $p \in P$. These are an equivalent version of global rules. Such families of functions allow one to find (at once) any finite part of the output configuration using a large enough part of a given input configuration.

4 ADDITIVE CELLULAR AUTOMATA

4.1 Additive CA—comonadic and graded comonadic

Additive cellular automata [7] are an example of variations of the concept of cellular automaton where the alphabets (and by pointwise extension, also configurations) are equipped with some algebraic structure and this structure is to be preserved by the local and global rules.

An additive cellular automaton has commutative monoids instead of just sets as input and output alphabets. We will use additive notation for these commutative monoids.

For any commutative monoid $X = (X, 0_X, +_X)$, the set X^G (the G-fold product of the underlying set of X with itself) also carries an obvious pointwise commutative monoid structure: $0_{X^G}\, g = 0_X$ and $(c +_{X^G} d)\, n = c\, n +_X d\, n$.

An additive CA local rule is a CA local rule $k : X^G \to Y$ for the underlying sets of X and Y that is additive (a commutative monoid homomorphism), i.e., $k\, 0_{X^G} = 0_Y$ and $k\, (c +_{X^G} d) = k\, c +_Y k\, d$. Ditto for additive CA global rules: they are CA global rules for the underlying sets that are additive.

For an example, consider $(G, 1_G, \cdot) = (\mathbb{Z}, 0, +)$ as the grid and let us use $(X, 0_X, +_X) = (\mathbb{Q}, 0, +)$ as both the input and output alphabet. Choose the local rule to be $k\, c = \frac{1}{3} * c\, (-1) + \frac{2}{3} * c\, 0$, then we can take $M = \{-1, 0\}$ and k is additive. The global rule is $f\, c = \lambda n.\, \frac{1}{3} * c\, (n-1) + \frac{2}{3} * c\, n$.

Here is the result of iterating this CA on a particular configuration:

...	−2	−1	0	+1	+2	+3	+4	...
...	0	0	1	0	0	0	0	...
...	0	0	$\frac{2}{3}$	$\frac{1}{3}$	0	0	0	...
...	0	0	$\frac{4}{9}$	$\frac{4}{9}$	$\frac{1}{9}$	0	0	...
...	0	0	$\frac{8}{27}$	$\frac{12}{27}$	$\frac{6}{27}$	$\frac{1}{27}$	0	...
...	0	0	$\frac{16}{81}$	$\frac{32}{81}$	$\frac{24}{81}$	$\frac{8}{81}$	$\frac{1}{81}$	...

$\vdots$

It is immediate that additive CA local and global rules are precisely coKleisli maps and cofree coalgebra maps of the cowriter comonad for G on **UnifCommMon** (the category of uniform commutative monoids) and, in the version with the explicit modulus M, the same as coKleisli maps and cofree coalgebra maps of the $\mathcal{M}$-graded cowriter comonad for G on **CommMon**.

But additive CA also admit a different account, their local rules are also Kleisli maps of a graded *monad*. This is because of the extra structure of the additive setting and a general fact about adjoint graded comonads and monads.

4.2 Adjoint (graded) comonads and monads

We need to introduce adjoint (also called conjugate) natural transformations [8], also called mates.

Given categories $\mathcal{C}$ and $\mathcal{D}$. Recall that functors $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ are called *adjoint* (notation $L \dashv R$) if they are equipped with a bijection $(-)^{\succ} : \mathcal{D}(LX, Y) \to \mathcal{C}(X, RY)$ natural in $X \in |\mathcal{C}|$ and $Y \in |\mathcal{D}|$; the bijection is called the *right transpose*. Its inverse $(-)^{\prec}$ is called the *left transpose*. Alternatively, one can ask for natural transformations $\mathsf{h} : \mathsf{Id} \to R \cdot L$ and $\mathsf{e} : L \cdot R \to \mathsf{Id}$, called the *unit* and *counit*, subject to what are known as the triangle equations, or work with one the two hybrid formats, e.g., with $(-)^{\succ}$ and e as the primitive data.

Now, given functors $L, L' : \mathcal{C} \to \mathcal{D}$ and $R, R' : \mathcal{D} \to \mathcal{C}$ such that $L \dashv R$ and $L' \dashv R'$. Natural transformations $\tau : L' \to L$ and $\theta : R \to R'$ are called *adjoint* (notation $\tau \dashv \theta$) if, for any $f : LX \to Y$, it is the case that $(f \circ \tau_X)^{\succ'} = \theta_Y \circ f^{\succ} : X \to R'Y$. Left and right adjoints in this sense always exist. A right adjoint of τ is defined by $\theta_X = (\mathsf{id}_{RX}{}^{\prec} \circ \tau_{RX})^{\succ'}$ or $\theta = R' \cdot \mathsf{e} \circ R' \cdot \tau \cdot R \circ \mathsf{h}' \cdot R$; a left adjoint of θ is symmetrically defined by $\tau_X = (\theta_{LX} \circ \mathsf{id}_{LX}{}^{\succ})^{\prec'}$ or $\tau = \mathsf{e}' \cdot L \circ L' \cdot \tau \cdot L \circ L' \cdot \mathsf{h}$. These right and left adjoints are also the only ones, hence adjoint natural transformations are in bijection.

The following observations were published by Kleiner [15] (but were likely known earlier).

Suppose given two endofunctors D and T on a category $\mathcal{C}$ such that $D \dashv T$. If D carries a comonad structure (ε, δ), then T carries a monad structure (η, μ) that is right adjoint to it in the sense that $\varepsilon \dashv \eta$ and $\delta \dashv \mu$. (These adjunctions are wrt. the adjunction $\mathsf{Id} \dashv \mathsf{Id}$ and the adjunction $D \cdot D \dashv T \cdot T$ canonically induced by the given adjunction $D \dashv T$.) And the other way round, if T carries a monad structure (η, μ), then D carries a left adjoint comonad structure (ε, δ). Because of uniqueness of adjoint natural transformations, the two constructions give a bijection between adjoint comonad structures on D and monad structures on T.

For corresponding comonad and monad structures on D resp. T, the categories $\mathbf{CoKl}(D)$ and $\mathbf{Kl}(T)$ are isomorphic. The isomorphism is identity on objects. On maps, it is the bijection between maps $DX \to Y$ and $X \to TY$ provided by the transpose operations of the adjunction.

A similar relationship occurring between monad structures on T and comonad structures on D when $T \dashv D$ is better known and was first observed by Eilenberg and Moore [8] (see also [24]). In this situation, which is actually not relevant for our application, if either T carries a monad structure or D carries a comonad structure, then the other carries a right adjoint comonad or a left adjoint monad structure and $\mathbf{EM}(T)$ and $\mathbf{CoEM}(D)$ are isomorphic; the isomorphism is then identity on carriers and maps.

That the comonad equations entail the monad equations and the other way around in both of these situations follows from uniqueness of adjoints and from how vertical and horizontal composition preserve adjointness:

- for $L, L', L'' : \mathcal{C} \to \mathcal{D}$, $R, R', R'' : \mathcal{D} \to \mathcal{C}$ such that $L \dashv R$, $L' \dashv R'$, $L'' \dashv R''$ and for $\tau : L' \to L$, $\tau' : L'' \to L'$, $\theta : R \to R'$, $\theta' : R' \to R''$, if $\tau \dashv \theta$ and $\tau' \dashv \theta'$, then $\tau \circ \tau' \dashv \theta' \circ \theta$;
- for $L, L' : \mathcal{C} \to \mathcal{D}$, $R, R' : \mathcal{D} \to \mathcal{C}$, $F, F' : \mathcal{D} \to \mathcal{E}$, $G, G' : \mathcal{E} \to \mathcal{D}$ such that $L \dashv R$, $L' \dashv R'$, $F \dashv G$, $F' \dashv G'$ and for $\tau : L' \to L$, $\theta : R \to R'$, $\phi : F' \to F$, $\psi : G \to G'$, if $\tau \dashv \theta$ and $\phi \dashv \psi$, then $\phi \cdot \tau \dashv \theta \cdot \psi$.

Because of this behavior of adjoint natural transformations wrt. compositions, the above facts about adjoint comonads and monads scale to graded comonads and monads.

Proposition Given two assignments $D, T : |\mathcal{M}| \to |[\mathcal{C}, \mathcal{C}]|$ of endofunctors on $\mathcal{C}$ to elements of $|\mathcal{M}|$ such that $D_M \dashv T_M$.

(1) If D carries an $\mathcal{M}$-graded comonad structure (D, ε, δ), then T carries an $\mathcal{M}^{\mathrm{oprev}}$-graded monad structure (T, η, μ) right adjoint to it in the sense that $D_{M \le M'} \dashv T_{M' \ge M}$ (notice that $D_{M \le M'} : D_M \to D_{M'}$ and $T_{M' \ge M} : T_{M'} \to T_M$), $\varepsilon \dashv \eta$ and $\delta_{N,M} \dashv \mu_{M,N}$. (The latter adjunctions are wrt. the adjunction $\mathsf{Id} \dashv \mathsf{Id}$ and the adjunctions $D_N \cdot D_M \dashv T_M \cdot T_N$ canonically induced by the adjunctions $D_M \dashv T_M$ and $D_N \dashv T_N$.)
And conversely, if T carries an $\mathcal{M}^{\mathrm{oprev}}$-graded monad structure (T, η, μ), then D carries a left adjoint $\mathcal{M}$-graded comonad structure (D, ε, δ).
(2) The two constructions form a bijection.
(3) For corresponding structures on D and T, the locally $\mathcal{M}^{\mathrm{oprev}}$-graded categories $\mathbf{CoKl}(D)$ and $\mathbf{Kl}(T)$ are isomorphic.

Proof sketch

(1) Given an $\mathcal{M}$-graded comonad structure (D, ε, δ), the data $T_{M' \ge M}$, η, $\mu_{M,N}$ can be produced from the data $D_{M \le M'}$, ε, $\delta_{N,M}$ using the explicit right adjoint formula.
That these data satisfy the equations of an $\mathcal{M}^{\mathrm{oprev}}$-graded monad is proved using that compositions are preserved by adjointness.
E.g., by construction, we have $D_{M \le M'} \dashv T_{M' \ge M}$, $D_{M' \le M''} \dashv T_{M'' \ge M'}$ and $D_{M \le M''} \dashv T_{M'' \ge M}$. Thus, $D_{M' \le M''} \circ D_{M \le M'} \dashv T_{M' \ge M} \circ T_{M'' \ge M'}$. Because D_M is functorial in M, we have $D_{M' \le M''} \circ D_{M \le M'} = D_{M \le M''}$, hence also $T_{M' \ge M} \circ T_{M'' \ge M'} = T_{M'' \ge M}$ since one natural transformation cannot have two different right adjoints.
Likewise, by construction, we have $\varepsilon \dashv \eta$ and $\delta_{1,M} \dashv \mu_{M,1}$. This gives $\varepsilon \cdot D_M \dashv T_M \cdot \eta$ and further $\varepsilon \cdot D_M \circ \delta_{1,M} \dashv \mu_{M,1} \circ T_M \cdot \eta$. We also have $\mathsf{id}_D \dashv \mathsf{id}_T$. The left counitality equation of D assures that $\varepsilon \cdot D_M \circ \delta_{1,M} = \mathsf{id}_D$. By uniqueness of right adjointness, this forces $\mu_{M,1} \circ T_M \cdot \eta = \mathsf{id}_T$, i.e., the right unitality equation of T.
The converse direction is symmetric.
(2) This is immediate from adjoint natural transformations being in bijection.
(3) The locally $\mathcal{M}^{\mathrm{oprev}}$-graded categories $\mathbf{CoKl}(D)$ and $\mathbf{Kl}(T)$ have the same objects.
The homsets $\mathbf{CoKl}(D)_M(X, Y)$ and $\mathbf{Kl}(T)_M(X, Y)$ consist of maps $D_M X \to Y$ and $X \to T_M Y$ of $\mathcal{C}$ respectively and these are in bijection via the right transpose of the adjunction $D_M \dashv T_M$.
Given a map $k : X \to_{M'} Y$ of $\mathbf{CoKl}(D)$ (i.e., a map $k : D_{M'} X \to Y$ of $\mathcal{C}$), its coercion along $M' \ge M$ is $k \circ D_{M \le M', X} : D_M X \to Y$. Since $D_{M \le M'} \dashv T_{M' \ge M}$, its right transpose is $(k \circ D_{M \le M', X})^{\succ} = T_{M' \ge M} \circ k^{\succ} : X \to T_M Y$, which is the coercion along $M' \ge M$ of the map $k^{\succ} : X \to_{M'} Y$ of $\mathbf{Kl}(T)$ (i.e., the map $k^{\succ} : X \to T_{M'} Y$ of $\mathcal{C}$).
The identity on X of $\mathbf{CoKl}(D)$ is $\varepsilon_X : D_1 X \to X$. Since $\varepsilon \dashv \eta$, its right transpose is $\varepsilon_X{}^{\succ} = (\mathsf{id}_X \circ \varepsilon_X)^{\succ} = \eta_X \circ \mathsf{id}_X = \eta_X : X \to T_1 X$, which is the identity on X of $\mathbf{Kl}(T)$.
Given maps $k : X \to_M Y$ and $\ell : Y \to_N Z$ of $\mathbf{CoKl}(D)$ (i.e., maps $k : D_M X \to Y$ and $\ell : D_N Y \to Z$ of $\mathcal{C}$), their composition $X \to_{M \cdot^{\mathrm{rev}} N} Z$ is the map $\ell \circ D_N k \circ \delta_{N,M,X} : D_{N \cdot M} X \to Z$. Since $\delta_{N,M} \dashv \mu_{M,N}$, its right transpose is $(\ell \circ D_N k \circ \delta_{N,M,X})^{\succ} = \mu_{M,N,Z} \circ (\ell \circ D_N k)^{\succ\succ} = \mu_{M,N,Z} \circ (\ell^{\succ} \circ k)^{\succ} = \mu_{M,N,Z} \circ T_M \ell^{\succ} \circ k^{\succ} : X \to T_{N \cdot M} Z$. As required, this is the composition $X \to_{M \cdot^{\mathrm{rev}} N} Z$ of $k^{\succ} : X \to_M Y$ and $\ell^{\succ} : Y \to_N Z$ of $\mathbf{Kl}(T)$ (i.e., $k^{\succ} : X \to T_M Y$ and $\ell^{\succ} : Y \to T_N Z$ of $\mathcal{C}$). □

A similar relationship holds between $\mathcal{M}^{\mathrm{oprev}}$-graded monad structures on T and $\mathcal{M}$-graded comonad structures on $T, D : |\mathcal{M}| \to |[\mathcal{C}, \mathcal{C}]|$ when $T_M \dashv D_M$. In this case, for corresponding structures on T and D, the locally $\mathcal{M}^{\mathrm{oprev}}$-graded categories $\mathbf{EM}(T)$ and $\mathbf{CoEM}(D)$ are isomorphic.

4.3 Additive CA as graded monadic

A straightforward fact relevant for additive CA with uniform continuity modulus $M \subseteq G$ is that an additive function $k : X^M \to Y$ is fully determined by what it does on relevant *point configurations*, i.e., partial configurations $[m \mapsto x]_M \in X^M$ defined by

$$[m \mapsto x]_M = \lambda m' \in M.\, \mathsf{if}\ m' = m\ \mathsf{then}\ x\ \mathsf{else}\ 0_X$$

for $m \in M$, $x \in X$. Indeed, if $k : X^M \to Y$ is additive, then, for any $c \in X^M$, one has

$$k\, c = k\, (\bigoplus_{m \in M} [m \mapsto c\, m]) = \bigoplus_{m \in M} k\, [m \mapsto c\, m]$$

Notice that, since M is finite and addition is commutative, these sums are well-defined.

It follows that additive functions $k : X^M \to Y$ are in bijection with additive functions $\phi : X \to Y^M$ via

$$k^{\succ}\, x = \lambda m \in M.\, k\, [m \mapsto x]_M$$

$$\phi^{\prec}\, c = \bigoplus_{m \in M} \phi\, (c\, m)\, m$$

Indeed, it is straightforward to verify that, if ϕ is additive, then $\phi^{\prec}$ is additive and, vice versa, if k is additive, then so is $k^{\succ}$.

Moreover, given an additive function ϕ, we have

$$\begin{array}{lcl} \phi^{\prec\succ}\, x & = & \lambda m \in M.\, \phi^{\prec}\, [m \mapsto x] \\ & = & \lambda m \in M.\, \bigoplus_{m' \in M} \phi\, ([m \mapsto x]_M\, m')\, m' \\ & = & \lambda m \in M.\, \bigoplus_{m' \in M} \phi\, (\text{if } m' = m \text{ then } x \text{ else } 0_X)\, m' \\ & = & \lambda m \in M.\, \bigoplus_{m' \in M} \text{if } m' = m \text{ then } \phi\, x\, m' \text{ else } \phi\, 0_X\, m' \\ & = & \lambda m \in M.\, \bigoplus_{m' \in M} \text{if } m' = m \text{ then } \phi\, x\, m' \text{ else } 0_Y \\ & = & \lambda m \in M.\, \phi\, x\, m \end{array}$$

and, given an additive function k, we have

$$\begin{array}{lcl} k^{\succ\prec}\, c & = & \bigoplus_{m \in M} k^{\succ}\, (c\, m)\, m \\ & = & \bigoplus_{m \in M} k\, [m \mapsto c\, m]_M \\ & = & k\, (\bigoplus_{m \in M} [m \mapsto c\, m]_M) \\ & = & k\, c \end{array}$$

Recall that a function k (a local rule) determines the letter at node 1_G of the output configuration from a given partial input configuration $c \in X^M$. The corresponding function ϕ specifies the same CA in a different way: it describes the contribution that a given letter x makes to the output letter at node 1_G if placed at a node $m \in M$ in an input configuration.[3]

In the case of our example from above, where the local rule $k : X^M \longrightarrow Y$ is $k\, c = \frac{1}{3} * c\, (-1) + \frac{2}{3} * c\, 0$, the corresponding function $\phi : X \longrightarrow Y^M$ is $\phi\, x = \lambda\{-1.\, \frac{1}{3} * x; 0.\, \frac{2}{3} * x\}$.

The above reasoning, together with the observation that $(-)^{\succ} : \mathbf{CommMon}(X^M, Y) \to \mathbf{CommMon}(X, Y^M)$ is natural in X and Y, witnesses an adjunction $D_M \dashv T_M$ between $D_M, T_M : \mathbf{CommMon} \to \mathbf{CommMon}$ where

- $T_M X = X^M$,
  $T_M(f : X \longrightarrow Y)(s \in X^M) = f \circ s \in Y^M$

so T_M is the same functor as D_M.

We know that D is equipped with the structure of an $\mathcal{M}$-graded comonad on $\mathbf{CommMon}$. By the proposition about adjoint graded comonads and monads, T consequently carries a right adjoint $\mathcal{M}^{\mathrm{oprev}}$-graded monad structure and the locally $\mathcal{M}^{\mathrm{oprev}}$-graded categories $\mathbf{CoKl}(D)$ and $\mathbf{Kl}(T)$ are isomorphic.

The functorial action of T is explicitly defined by

- $T_{M' \geq M,X}(s \in X^{M'}) = s|^M$
  $= \lambda m \in M.\, \text{if } m \in M' \text{ then } s\, m \text{ else } 0_X \in X^M$

(note that $M' \subseteq M$), so $T_{M' \geq M,X}\, s$ is obtained by "padding out" s.

The equations of functoriality hold because $M \setminus M = \emptyset$ and $(M \setminus M') \cup (M' \setminus M'') = M \setminus M''$ when $M' \subseteq M$ and $M'' \subseteq M'$.

[3] If G is a group, this is equivalent to saying that ϕ describes the contribution that an input letter $x \in X$ at node 1_G makes to the output letter at node m^{-1} for any $m \in M$; the influence of the input letter at node 1_G is limited to the $Y^{M^{-1}}$ part of the output configuration. The formulation we have given above is more robust. For it to work, it suffices that G is a monoid.

In order for $D_{M \leq M'}$ and $T_{M' \geq M}$ to be adjoint whenever well-defined (i.e., when $M' \subseteq M$), it must be that, for any $k : X^{M'} \longrightarrow Y$,

$$(k \circ D_{M \leq M',X})^{\succ} = T_{M' \geq M,Y} \circ k^{\succ} : X \longrightarrow Y^M$$

We verify

$$\begin{array}{ll} \multicolumn{2}{l}{(k \circ D_{M \leq M',X})^{\succ}\, x} \\ = & \lambda m \in M.\, k\, (D_{M \leq M',X}\, [m \mapsto x]_M) \\ = & \lambda m \in M.\, k\, (\lambda m' : M'.\, [m \mapsto x]_M\, m') \\ = & \lambda m \in M.\, \text{if } m \in M' \text{ then } k\, (\lambda m' : M'.\, [m \mapsto x]_M\, m') \\ & \qquad\qquad\qquad \text{else } \ k\, (\lambda m' : M'.\, [m \mapsto x]_M\, m') \\ = & \lambda m \in M.\, \text{if } m \in M' \text{ then } k\, [m \mapsto x]_{M'} \text{ else } k\, 0_{X^{M'}} \\ = & \lambda m \in M.\, \text{if } m \in M' \text{ then } k\, [m \mapsto x]_{M'} \text{ else } 0_Y \\ = & T_{M' \geq M,Y}(\lambda m \in M'.\, k\, [m \mapsto x]_{M'}) \\ = & T_{M' \geq M,Y}(k^{\succ}\, x) \end{array}$$

Note that adjointness is automatic (does not need proof) if one has calculated the definition of $T_{M' \geq M}$ from the definition of $D_{M \leq M'}$ and the general formula for the right adjoint natural transformation. The equations of functoriality T are automatic from the equations of functoriality of D once adjointness has been established.

The remaining data of the graded monad are explicitly defined by

- $\eta_X(x \in X) = \lambda\_ \in 1.\, x \in X^1$,
- $\mu_{N,M,X}(s \in (X^M)^N)$
  $= \lambda o \in M \cdot N.\, \bigoplus_{m \in M, n \in N, o = m \cdot n} s\, n\, m \in X^{N \cdot^{\mathrm{rev}} M}$.

Kleisli maps of grade M of T correspond to coKleisli maps of grade M of D: they are maps $\phi : X \longrightarrow Y^M$ in bijection with maps $k : X^M \longrightarrow Y$ as detailed above. Free algebra maps of grade M of T however are families of maps $h_N : X^N \longrightarrow Y^{M \cdot N}$ that are natural in N (meaning that, if $N' \subseteq N$, then $(h_{N'} s)|^{M \cdot N} = h_N(s|^N) \in Y^{M \cdot N}$ for $s \in X^{N'}$) and satisfy the algebra homomorphism equation

$$\begin{array}{ll} \multicolumn{2}{l}{h_{N \cdot P}\, (\lambda q \in N \cdot P.\, \bigoplus_{n \in N, p \in P, q = n \cdot p} s\, p\, n)} \\ = & \lambda r \in M \cdot N \cdot P.\, \bigoplus_{o \in M \cdot N, p \in P, r = o \cdot p} h_N\, (s\, p)\, o \in Y^{M \cdot N \cdot P} \end{array}$$

for $s \in (X^N)^P$.

Explicitly, the bijection between Kleisli maps and free algebra maps is the following: given $\phi : X \longrightarrow Y^M$, define $h_N = \mu_{N,M,Y} \circ \phi^N$; given $h_N : X^N \longrightarrow Y^{M \cdot N}$, define $\phi = h_1 \circ \eta_X$.

Formal polynomials. It does not make much sense to think of elements s of $T_M X = X^M$ as partial configurations. But they can be viewed as *formal polynomials* with *exponents* from $M \subseteq G$ and *coefficients* from X (or shorter, formal polynomials over X of degree M). That is, a function $s \in X^M$ is identified with the one-variable polynomial $\lambda \mathbf{y}.\, \sum_{m \in M} s\, m \times \mathbf{y}^m$ where the variable, exponentiation, multiplication and summation are purely formal. The monoid $(G, 1_G, \cdot)$ is best thought of as additive rather than multiplicative here, the archetypical example being $(\mathbb{N}, 0, +)$. A polynomial of degree n is a polynomial of degree $\{0..n\} \subseteq \mathbb{N}$.

The data of the graded monad T represent important operations on polynomials. The unit η sends an element of X to the corresponding constant polynomial:

$$\eta_X\, x = \lambda \mathbf{y}.\, x \times \mathbf{y}^{1_G} = \lambda \mathbf{y}.\, x$$

The multiplication μ sends a polynomial whose coefficients are polynomials with coefficients from X to a polynomial with coefficients from X by evaluating the inner polynomials at the outer variable:

$$\begin{aligned} &\mu_{N,M,X}\,(\lambda \mathbf{y}.\, \sum_{n \in N} (\lambda \mathbf{y}_n.\, \sum_{m \in M} s\,n\,m \times \mathbf{y}_n^m) \times \mathbf{y}^n) \\ = \;& \lambda \mathbf{y}.\, \sum_{n \in N} (\sum_{m \in M} s\,n\,m \times \mathbf{y}^m) \times \mathbf{y}^n \\ = \;& \lambda \mathbf{y}.\, \sum_{n \in N} \sum_{m \in M} s\,n\,m \times \mathbf{y}^{m \cdot n} \\ = \;& \lambda \mathbf{y}.\, \sum_{o \in M \cdot N} (\textstyle\bigoplus_{m \in M, n \in N, o = m \cdot n} s\,n\,m) \times \mathbf{y}^o \end{aligned}$$

(Remember it is best to think of G as additive rather than multiplicative here, so 1_G is zero and $\cdot$ is addition.)

5 ADDITIVE CA AS (UNGRADED) MONADIC?

Could we also view additive CA as (ungraded) monadic? Only with reservations.

If we ignore the uniform continuity requirement, then we can restrict the alphabets to be *complete* monoids—where any family of elements can be summed—and define a *formal series* monad on the category **CmplMon** of complete monoids by

- $TX = X^G$,
- $\eta_X (x \in X) = \lambda\_ \in G.\, x \in X^G$,
- $\mu_X (s \in (X^G)^G) = \lambda o \in G.\, \bigoplus_{m,n \in G, o = m \cdot n} s\,n\,m \in X^G$.

The sum in the definition of μ is then well-defined although it is generally infinite.

Alternatively, if G has the property that any element $o \in G$ only admits finitely many decompositions $o = m \cdot n$ (this is true, e.g., for free monoids), then we get a monad already on **CommMon** since the sum in the definition of μ is then necessarily finite.

If we insist on the uniform continuity requirement, we may aim to work in **UnifCommMon**. But uniform continuity of a map $k : X^G \to Y$ is generally not the same as uniform continuity of the corresponding map $\phi : X \to Y^G$ unless X is finite. We can restrict to the full subcategory of the Kleisli category of T given by finite uniform spaces.

6 RELATED WORK

To model impure notions of computation of the "effectful" flavor in Kleisli categories of (strong) monads is standard since Moggi [19]. From the beginning, this tradition includes a programming language design for this model, the λ_c-calculus.

The use of coKleisli categories of (lax monoidal) comonads to model a different flavor of impure notions of computation—"context-dependent" notions of computation— goes back to at least Brookes, Geva and Van Stone [2, 3]; a more systematic study, together with a programming language, was provided by Uustalu and Vene [27]. Uustalu and Vene [26] studied causal (à la Lustre) and non-causal (à la Lucid) dataflow computation; Capobianco and Uustalu [5] treated cellular automata.

Graded monads were invented by Smirnov [25], but also by Borceux, Janelidze and Kelly [1], who studied lax actions of a monoidal category on a category, which are the same thing, but did not view them as a generalization of monads.

Katsumata [14] and Melliès [18] introduced graded monads to programming semantics and demonstrated that they can be used for a finer analysis of effectful notions of computation where one can talk about degrees of effectfulness of functions.

Graded comonads were first employed in programming semantics by Petricek et al. [21] and Brunel et al. [4], for finer analysis of context-dependent or resource-consuming notions of computation. For this, one typically wants graded comonads on a symmetric monoidal category, with grades from a preordered semiring.

Fujii, Katsumata and Melliès [10] were the first to introduce analogues of the Kleisli and Eilenberg-Moore constructions for graded monads. In their work, the Kleisli and Eilenberg-Moore adjunctions are adjunctions of ordinary category theory. Their constructions suffer from certain deficiencies. McDermott and Uustalu [17] showed that those can be avoided by moving to locally graded category theory. The intuitive reason why this is the right thing to do is that a graded monad is precisely a relative monad on a particular canonical functor, in the sense of locally graded category theory.

Adjoint comonad-monad pairs and adjoint monad-comonad pairs are rare in computation structures. The main adjoint comonad-monad pair example is given by the coreader comonad and the reader monad while the main adjoint monad-comonad pair example is given by the writer monad and the cowriter comonad. Hinze [12] and Orchard [20] have discussed such examples.

Category theory is not a commonly used tool among cellular automata theorists, but Salo and Törmä [22] studied what monos, epis, split monos, split epis, regular monos, regular epis amount to in categories of various classes of subshifts and block maps; subshifts are subsets of $X^{\mathbb{Z}}$ closed under translation and block maps are maps between two subshifts commuting with translations. Also, Fernandez, Maignan and Spicher [9] showed that the global rule of a CA arises as the left Kan extension of the local rule in the 2-category of posets, monotone functions and comparison of monotone functions, under a particular view of the global and local rule as monotone functions. In our opinion, their categorical analysis of cellular automata is less illuminating than ours. Its shortcoming is that, while it explains the relationship of local and global rules in categorical terms, it gives no such explanation for local or global rules themselves: those are just special cases of monotone functions between special cases of posets. Trancón y Widemann and Hauhs [29] attempted to adapt Turi and Plotkin's bialgebraic take on structural operational semantics to CA. They modelled configurations of a CA as syntax, with terms for a signature containing constants for letters of the alphabet, and a form of configuration-labelled nonwellfounded trees (from where infinite sequences of configurations could be extracted) as behaviors. The semantic function, assigning such trees to configurations, modelled infinite iteration of the CA rule from a given initial configuration.

7 CONCLUSIONS AND FUTURE WORK

We have shown that, while CA are a nice example of comonadic notions of computation, they exemplify more in fact. In particular, it is natural to consider grading in this example and this makes the analysis more informative in that it is explicitly resource-aware. It also makes it more elementary in one aspect: instead of an involved category like **Unif**, we can work in **Set**.

The CA example of graded comonadic notions of computation is instructive in that it helps with intuitions for the sophistication involved in the locally graded coKleisli and coEilenberg-Moore constructions as compared to their ordinary counterparts.

Additive CA are among the rare examples of notions of computation that are both (graded) comonadic and monadic: in fact, they make a very elegant such an example. They also demonstrate the real necessity of the graded approach in some situations. In our example, the issue is in potentially infinite sums. The possibility to restrict to finite sums that the graded approach offers makes it more broadly applicable.

ACKNOWLEDGMENTS

We got to know about Kleiner's paper thanks to James McKinna who had learned about it from Marcelo Fiore. We are grateful to Dylan McDermott for useful feedback.

This work was supported by the Estonian Research Council team grant no. PRG1210 and by the Icelandic Research Fund project 228684-051.

REFERENCES

[1] Francis Borceux, George Janelidze, and G. Max Kelly. 2005. Internal Object Actions. *Comment. Math. Univ. Carolin.* 46, 2 (2005), 235–255. http://eudml.org/doc/249553

[2] Stephen Brookes and Shai Geva. 1992. Computational Comonads and Intensional Semantics. In *Applications of Categories in Computer Science*, Michael P. Fourman, Peter T. Johnstone, and Andrew M. Pitts (Eds.). London Math. Soc. Lect. Notes Series, Vol. 177. Cambridge Univ. Press, Cambridge, 1–44.

[3] Stephen Brookes and Kathryn Van Stone. 1993. *Monads and Comonads in Intensional Semantics.* Technical Report CMU-CS-93-140. School of Comput. Sci., Carnegie Mellon Univ.

[4] Aloïs Brunel, Marco Gaboardi, Damiano Mazza, and Steve Zdancewic. 2014. A Core Quantitative Coeffect Calculus. In *Programming Languages and Systems: 23rd Europ. Symp. on Programming, ESOP 2014, Proc. (Lect. Notes in Comput. Sci., Vol. 8410)*, Zhong Shao (Ed.). Springer, Berlin, Heidelberg, 351–370. https://doi.org/10.1007/978-3-642-54833-8_19

[5] Silvio Capobianco and Tarmo Uustalu. 2010. A Categorical Outlook on Cellular Automata. In *Journées Automates Cellulaires 2010*, Jarkko Kari (Ed.). TUCS Lecture Note Series, Vol. 13. University of Turku, Turku, 88–99. https://hal.science/hal-00542015/

[6] Tullio Ceccherini-Silberstein and Michal Coornaert. 2010. *Cellular Automata and Groups.* Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-14034-1

[7] Alberto Dennunzio, Enrico Formenti, Darij Grinberg, and Luciano Margara. 2020. From Linear to Additive Cellular Automata. In *47th Int. Coll. on Automata, Languages, and Programming (ICALP 2020)*, Artur Czumaj, Anuj Dawar, and Emanuela Merelli (Eds.). Dagstuhl Publishing, Saarbrücken/Wadern, 125:1–125:13. https://doi.org/10.4230/lipics.icalp.2020.125

[8] Samuel Eilenberg and John C. Moore. 1965. Adjoint Functors and Triples. *Illinois J. Math.* 9, 3 (1965), 381–398. https://doi.org/10.1215/ijm/1256068141

[9] Alexandre Fernandez, Luidnel Maignan, and Antoine Spicher. to appear. Cellular Automata and Kan Extensions. *Nat. Comput.* (to appear), 15 pages. https://doi.org/10.1007/s11047-022-09931-0

[10] Soichiro Fujii, Shin-ya Katsumata, and Paul-André Melliès. 2016. Towards a Formal Theory of Graded Monads. In *Foundations of Software Science and Computation Structures, 19th Int. Conf., FoSSaCS 2016, Proc.*, Bart Jacobs and Christoph Löding (Eds.). Lect. Notes in Comput. Sci., Vol. 9634. Springer, Cham, 513–530. https://doi.org/10.1007/978-3-662-49630-5_30

[11] Gustav A. Hedlund. 1969. Endomorphisms and Automorphisms of the Shift Dynamical System. *Math. Syst. Theor.* 3, 4 (1969), 320–375. https://doi.org/10.1007/bf01691062

[12] Ralf Hinze. 2011. Monads from Comonads, Comonads from Monads. Notes for an IFIP WG 2.8 Meeting. http://www.cs.ox.ac.uk/ralf.hinze/WG2.8/28/slides/Comonad.pdf

[13] Jarkko Kari. 2005. Theory of Cellular Automata: A Survey. *Theor. Comput. Sci.* 334 (2005), 3–33. https://doi.org/10.1016/j.tcs.2004.11.021

[14] Shin-ya Katsumata. 2014. Parametric Effect Monads and Semantics of Effect Systems. In *Proc. of 41st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '14*. ACM Press, New York, 633–645. https://doi.org/10.1145/2535838.2535846

[15] Mark Kleiner. 1990. Adjoint Monads and an Isomorphism of the Kleisli Category. *J. Algebra* 133, 1 (1990), 79–82. https://doi.org/10.1016/0021-8693(90)90069-Z

[16] Paul B. Levy. 2019. Locally Graded Categories. Slides for talk at Univ. of Cambridge. https://www.cs.bham.ac.uk/~pbl/papers/locgrade.pdf

[17] Dylan McDermott and Tarmo Uustalu. 2022. Flexibly Graded Monads and Graded Algebras. In *Mathematics of Program Construction, 14th Int. Conf., MPC 2022, Proc.*, Ekaterina Komendantskaya (Ed.). Lect. Notes in Comput. Sci., Vol. 13544. Springer, Cham, 102–128. https://doi.org/10.1007/978-3-031-16912-0_4

[18] Paul-André Melliès. 2012. Parametric Monads and Enriched Adjunctions. Manuscript. https://www.irif.fr/mellies/tensorial-logic/8-parametric-monads-and-enriched-adjunctions.pdf

[19] Eugenio Moggi. 1989. Computational Lambda-Calculus and Monads. In *Proc. of 4th Ann. IEEE Symp. on Logic in Computer Science, LICS '89*. IEEE Comput. Soc., Los Alamitos, CA, 14–23. https://doi.org/10.1109/lics.1989.39155

[20] Dominic Orchard. 2011. Should I Use a Monad or a Comonad? Draft. https://www.cs.kent.ac.uk/people/staff/dao7/drafts/monad-or-comonad-orchard11-draft.pdf

[21] Tomas Petricek, Dominic Orchard, and Alan Mycroft. 2013. Coeffects: Unified Static Analysis of Context-Dependence. In *Automata, Languages, and Programming: 40th Int. Coll., ICALP 2013, Proc., Part II (Lect. Notes in Comput. Sci., Vol. 7966)*, Fedor V. Fomin, Rūsiņš Freivalds, Marta Kwiatkowska, and David Peleg (Eds.). Springer, Berlin, Heidelberg, 385–397. https://doi.org/10.1007/978-3-642-39212-2_35

[22] Ville Salo and Ilkka Törmä. 2015. Category Theory of Symbolic Dynamics. *Theor. Comput. Sci.* 567 (2015), 21–45. https://doi.org/10.1016/j.tcs.2014.10.023

[23] Dan (sigfpe) Piponi. 2006. Evaluating Cellular Automata is Comonadic. Blog post. http://blog.sigfpe.com/2006/12/evaluating-cellular-automata-is.html

[24] Zoran Škoda. 2011. Adjoint Monads. ncatlab entry. https://ncatlab.org/nlab/show/adjoint+monad

[25] A. Smirnov. 2008. Graded Monads and Rings of Polynomials. *J. Math. Sci.* 151, 3 (2008), 3032–3051. https://doi.org/10.1007/s10958-008-9013-7

[26] Tarmo Uustalu and Varmo Vene. 2006. The Essence of Dataflow Programming. In *Central European Functional Programming School, 1st Summer School, CEFP 2005, Revised Selected Lectures*, Zoltán Horváth (Ed.). Lect. Notes in Comput. Sci., Vol. 4164. Springer, Berlin, Heidelberg, 135–167. https://doi.org/10.1007/11894100_5

[27] Tarmo Uustalu and Varmo Vene. 2008. Comonadic Notions of Computation. *Electron. Notes Theor. Comput. Sci.* 203, 5 (2008), 263–284. https://doi.org/10.1016/j.entcs.2008.05.029

[28] Richard J. Wood. 1976. *Indicial Methods for Relative Categories.* Ph. D. Dissertation. Dalhousie University. http://hdl.handle.net/10222/55465

[29] Baltasar Trancón y Widemann and Michael Hauhs. 2011. Distributive-Law Semantics for Cellular Automata and Agent-Based Models. In *Algebra and Coalgebra in Computer Science: 4th Int. Conf., CALCO 2011, Proc. (Lect. Notes in Comput. Sci., Vol. 6851)*, Andrea Corradini, Bartek Klin, and Corina Cîrstea (Eds.). Springer, Berlin, Heidelberg, 344–358. https://doi.org/10.1007/978-3-642-22944-2_24