

A Category Theoretic Interpretation of Gandy's Principles for Mechanisms

Joseph Razavi

Andrea Schalk

School of Computer Science
University of Manchester
Manchester, UK

joseph.razavi@manchester.ac.uk

andrea.schalk@manchester.ac.uk

Based on Gandy's principles for models of computation we give category-theoretic axioms describing locally deterministic updates to finite objects. Rather than fixing a particular category of states, we describe what properties such a category should have. The computation is modelled by a functor that encodes updating the computation, and we give an abstract account of such functors. We show that every updating functor satisfying our conditions is computable.

1 Introduction

In a well-known paper [3], Gandy sets out principles which aim to characterize the possible behaviours of a discrete, deterministic mechanical computing device which could be realized in the physical world. Although in Gandy's detailed axiomatization there are four principles, they can be summarized by two conceptual insights (as emphasized by, for instance, [1, 11]): first, that states of computation should be finite objects with a bounded amount of local detail; second, that changes should only propagate with a bounded velocity, and thus their effects on a given location should be determined by a neighbourhood of finite size.

Gandy's technical realization of these principles are axioms which are set-theoretic in nature, the idea being that every mathematical description of a computing machine ought to correspond to a set-theoretical one. As long as this is done sensibly, and the original model satisfies Gandy's conceptual principles, the resulting formalization will satisfy Gandy's axioms. Although some later studies keep within this set-theoretic framework [11], other work inspired by Gandy's principles replaces his arbitrary hereditarily finite sets with mathematical objects for which one has more direct spatial intuitions, such as graphs [1, 7, 10] or simplicial surfaces [2]. Indeed, when one has a concrete model of computation in mind, it is often easier to conduct detailed investigations into the operation of Gandy's principles by working with the models directly, in an environment which respects their extra structure, rather than their set-theoretic encodings.

In many of these models the dynamics are given in terms of a colimit of updated versions of certain neighbourhoods of a state. This is explicit in [7], and suggested by the use of unions in [1, 2]. Indeed, if one heuristically reads Gandy's notion of restriction of a state to a part of a state as analogous to a set of morphisms, then Gandy's own description strongly suggests a colimit. This is evidence that the ambient categorical structure can play an explanatory role.

Rather than describing specific categories whose objects can serve as states of computation, one might ask what kinds of category could serve as a setting for Gandy-like models. We know that objects like graphs and simplicial complexes are characterized by being colimits of a set of generating objects. We might wonder whether the colimit description of the dynamics follows from this, given some more direct axiomatization of local determinism.

In this paper, we attempt to give just such a description of a class of categories suitable for describing states of computation in the style of Gandy, along with a categorical description of locally deterministic dynamics. We came to this axiomatization in an attempt to better understand the phenomena of information flow in cellular automata as described in the first author's thesis [8]. For this purpose, they seem to be useful for getting at the relevant aspects of the causal structure, and we hope, therefore, that they may be of use to workers engaged in similar enquiry.

2 Preliminaries: Comma Categories and Kan Extensions

In this section we briefly sketch some category theoretic background, loosely following [5, 12]; the reader interested in a more comprehensive overview of category theory is directed to [6].

In order to talk about things happening locally, as is required to model Gandy's ideas, it is useful to think of a morphism $p : A \rightarrow X$ in \mathbb{C} as a 'shape A located at position p in X '. More generally, one often wants to restrict attention to the case where the source and target of the morphisms are the output of two functors $F : \mathbb{X} \rightarrow \mathbb{C}$ and $G : \mathbb{Y} \rightarrow \mathbb{C}$. In this paper, we study such morphisms in the standard setting: the **comma category** F/G . The objects of F/G are triples (X, p, Y) where X is an object of \mathbb{X} , Y is an object of \mathbb{Y} , and $p : FX \rightarrow GY$ in \mathbb{C} . A morphism from (X, p, Y) to (X', p', Y') is a pair (f, g) such that $Gg \circ p = p' \circ Ff$.

We may think of such an equation as telling us that the way the shape FX is located at position p in GY is compatible with that of FX' at p' in GY' via Ff and Gg .

The definition of objects of the comma category as triples allows us to define two functors, dom and cod , to \mathbb{X} and \mathbb{Y} respectively, by projecting onto the first and last components. The morphisms in the middle components of these triples assemble themselves into a natural transformation

$$\begin{array}{ccc} F/G & \xrightarrow{\text{dom}} & \mathbb{X} \\ \text{cod} \downarrow & \swarrow_{\text{mid}} & \downarrow F \\ \mathbb{Y} & \xrightarrow{G} & \mathbb{C}, \end{array}$$

and this square has a universal property which defines the comma category up to isomorphism.

When giving a comma category, if we have a subcategory \mathbb{G} of \mathbb{C} , we often write \mathbb{G} to mean its inclusion into \mathbb{C} , writing for example \mathbb{G}/\mathbb{C} to mean the comma category from the inclusion of \mathbb{G} to the identity on \mathbb{C} . Similarly, if A is an object of \mathbb{C} , we may write A to mean the constant functor from the one-object category to A . As a first approximation, one can think of a comma category F/G as a sort of asymmetric pullback, asking not what F and G have in common, but what the result of probing G with F is. Certain properties are then unsurprising. The domain and codomain functors on \mathbb{C}/\mathbb{C} both have a section, given by sending each object to its identity morphism. Moreover, if F has a section, then so does the codomain functor of F/G .

In the introduction, we mentioned that in many examples we encounter functors which can be calculated by colimits. This may indicate the presence of a Kan extension, which is defined as follows. Suppose we have functors $F : \mathbb{X} \rightarrow \mathbb{D}$ and $G : \mathbb{X} \rightarrow \mathbb{C}$. A **(left) Kan extension** of G along F is a functor $L : \mathbb{D} \rightarrow \mathbb{C}$ equipped with a natural transformation $\kappa : G \rightarrow L \circ F$ which is universal in the sense that for every $K : \mathbb{D} \rightarrow \mathbb{C}$ with $\alpha : G \rightarrow K \circ F$ there exists a unique mediator $\mu : L \rightarrow K$ such that α is

equal to the pasting

$$\begin{array}{ccc}
 \mathbb{X} & \xrightarrow{G} & \mathbb{C} \\
 \downarrow F & \swarrow \kappa & \parallel \\
 \mathbb{D} & \xrightarrow{L} & \mathbb{C} \\
 \parallel & \swarrow \mu & \parallel \\
 \mathbb{D} & \xrightarrow{K} & \mathbb{C}.
 \end{array}$$

On first encountering the definition, it is common to wonder whether when $G = L \circ F$, the identity transformation exhibits L as a Kan extension. One soon realizes, however, that far away from the image of F , the values of L may have little to do with G ; this will preclude having the universal property. A natural thing to demand to improve this situation is that F have a section; one easily verifies that in this case every functor that post-composes with F to give G is a Kan extension of the latter along the former. We are interested in Kan extensions with additional properties.

A (left) Kan extension L of G along F as below is called **absolute** if for all $H : \mathbb{C} \rightarrow \mathbb{E}$ the pasting

$$\begin{array}{ccccc}
 \mathbb{X} & \xrightarrow{G} & \mathbb{C} & \xrightarrow{H} & \mathbb{E} \\
 F \downarrow & \swarrow \kappa & \parallel & & \parallel \\
 \mathbb{D} & \xrightarrow{L} & \mathbb{C} & \xrightarrow{H} & \mathbb{E}
 \end{array}$$

is a Kan extension. This means that having a Kan extension for G gives us a Kan extension for any composite of G . For example, suppose $F : \mathbb{C} \rightarrow \mathbb{D}$, $U : \mathbb{D} \rightarrow \mathbb{C}$ with a natural transformation $\eta : 1_{\mathbb{C}} \rightarrow U \circ F$. Then η is the unit of an adjunction with left adjoint F and right adjoint U if and only if it exhibits U as the Kan extension of the identity along F and this extension is absolute.

A Kan extension L of G along F as above is called **pointwise** if whenever we have a functor $J : \mathbb{Y} \rightarrow \mathbb{D}$ and we consider the comma category F/J , the pasting

$$\begin{array}{ccccc}
 F/J & \xrightarrow{\text{dom}} & \mathbb{X} & \xrightarrow{G} & \mathbb{C} \\
 \text{cod} \downarrow & \swarrow \text{mid} & \downarrow F & \swarrow \kappa & \parallel \\
 \mathbb{Y} & \xrightarrow{J} & \mathbb{D} & \xrightarrow{L} & \mathbb{C}
 \end{array}$$

is a Kan extension. We can think of a pointwise Kan extension as being ‘locally a Kan extension’ in that whenever we ‘probe’ L with a functor J as above, we still get a Kan extension. One reason pointwise Kan extensions are useful is that for any object D of \mathbb{D} we can compute LD as the colimit of the functor given by precomposing G with $\text{dom} : F/D \rightarrow \mathbb{X}$, which means that if we take the big diagram with one copy of each object X of \mathbb{X} for every morphism from $FX \rightarrow D$, and arrows between them making commuting triangles, then D is the colimit.

One family of examples is given by the fact that every absolute Kan extension is pointwise. Another example which one frequently encounters is the idea of a **dense subcategory**. A subcategory \mathbb{G} of \mathbb{C} is called **dense** if and only if the identity on \mathbb{C} is the pointwise Kan extension of the inclusion of \mathbb{G} into \mathbb{C} along itself, meaning that every object of \mathbb{C} is the colimit of all ways of mapping objects of \mathbb{G} into it. We think of objects in \mathbb{C} as regions generated by the shapes in the dense subcategory \mathbb{G} . In the sequel, this is used to capture the notion of ‘finite detail’. Since every morphism out of a colimit is uniquely determined by suitable morphisms out of the constituent parts this also characterizes the morphisms in \mathbb{C} , which are determined by the way they act on morphisms out of \mathbb{G} by postcomposition. Similar reasoning shows that the domain of every object of \mathbb{C}/\mathbb{C} is the colimit of all the domains of morphisms of \mathbb{G}/\mathbb{C} into it (to see that one does not end up with spurious extra copies of objects of \mathbb{G} , the important observation is that every morphism in \mathbb{C}/\mathbb{C} factors as a morphism with an identity of \mathbb{C} in its domain, followed by one with an

identity of \mathbb{C} in its codomain). This implies that the identity natural transformation corresponding to the commuting diagram

$$\begin{array}{ccccc} \mathbb{G}/\mathbb{C} & \hookrightarrow & \mathbb{C}/\mathbb{C} & \xrightarrow{\text{dom}} & \mathbb{C} \\ \downarrow & & \parallel & & \parallel \\ \mathbb{C}/\mathbb{C} & \xlongequal{\quad} & \mathbb{C}/\mathbb{C} & \xrightarrow{\text{dom}} & \mathbb{C} \end{array}$$

is a pointwise Kan extension. The codomains of morphisms play little role in this argument, and indeed one can replace \mathbb{G}/\mathbb{C} and \mathbb{C}/\mathbb{C} in the above with \mathbb{G}/U and \mathbb{G}/U respectively, for any functor U into \mathbb{C} .

Kan extensions enjoy the property that if one vertically composes a collection of Kan extension squares, the result is again a Kan extension. Moreover, if every component of the pasting is absolute or pointwise, so is the result.

3 Finite Objects and Local Determinism

Suppose one has in mind a collection of spatially extended states, and incomplete parts of states, for a ‘mechanical’ model of computation, along with a notion for how the various parts fit together to form larger parts and complete states. These constitute a category \mathbb{C} . Following Gandy, we stipulate that, if the model is to be truly mechanical, these states must be finitely big and have a finite amount of possible local detail.

Let us first consider the restriction on \mathbb{C} corresponding to the objects being ‘finitely big’. Given two finite combinatorial objects, we expect that the number of ways in which one fits into the other to be finite. This means that all hom-sets in \mathbb{C} should be finite; one says that \mathbb{C} is ‘locally finite’.

Only slightly more subtle is the issue of finite local detail. The idea is that there should be a finite set of objects such that any object X is determined by the ways in which these objects map into X . This is exactly the condition described above for a dense sub-category, so this is the notion we use here.

One example of a locally finite category with a finite dense subcategory is the category of finite graphs: there are finitely many graph homomorphisms between any two graphs, and every graph can be constructed as a colimit of nodes and edges. This is, however, a somewhat unnatural example for our purposes, since allowing nodes of arbitrarily high degree means that the notion of ‘locally finitely detailed’ can not mean ‘locally’ in the usual sense for graphs. It is more natural to take graphs with a fixed bound on the degree.

Another example, which we return to below, is the category **Tape** whose objects are strings over the alphabet $\{\blacksquare, \square\}$, and where a morphism from A to B is a position in B at which A occurs as a substring. For example, there are two morphisms from $\blacksquare \square \blacksquare \blacksquare$ to $\blacksquare \square \blacksquare \square \blacksquare \blacksquare$, since it occurs at positions 0 and 3. For technical reasons, it is useful to assume that the empty string is an initial object, occurring just once in every string. The full subcategory on the objects $\{\blacksquare, \square, \blacksquare\blacksquare, \blacksquare\square, \square\blacksquare, \square\square\}$ is dense, because any string is determined by knowing its consecutive pairs of letters, and which pairs share individual letters. In a similar vein, one might think of objects made of a finite number of tiles (even of, say, a Penrose tiling), generated by gluing together neighbourhoods where they share tiles.

The issue of local determinism is more interesting. First, we suppose that the way in which states are updated by the dynamics is given by a functor $U : \mathbb{C} \rightarrow \mathbb{C}$. Functoriality is a reasonable requirement, meaning that the updated versions of the parts of a state should fit together in the updated version of that state. It loosely corresponds to Gandy’s idea that the update function in his model should be ‘structural’, able only to observe the way atomic parts of the state fit together, not the names we have given them.

As an example, let us consider the one dimensional cellular automaton on the alphabet $\{\blacksquare, \square\}$ in which a white cell becomes black if any of its neighbours is black. We can model this as a functor

$U : \text{Tape} \rightarrow \text{Tape}$ which removes the two outer cells in a string (returning the empty string if there are too few cells) and updates those in the middle according to the rule. For example $U(\blacksquare \square \square \blacksquare) = \blacksquare \blacksquare \blacksquare$. Functoriality of U corresponds roughly to the fact that the update can not make use of, for example, the position of a cell in the sequence to determine its new colour.

Now we come to local determinism itself. Suppose we update a state, and then look at a part of the updated situation. This amounts to considering a map $p : A \rightarrow UX$ for some A and X in \mathbb{C} . We want to remember the state X we are updating, so it is best to consider p as an object of \mathbb{C}/U . Since we postulate that the action of U is locally deterministic, this local effect must have had some local cause $f : N \rightarrow Y$, for some N and Y in \mathbb{C} , which explains why p occurs in UX in the following sense. If we update f to get $Uf : UN \rightarrow UY$, we should find p inside the result via a morphism $\eta : p \rightarrow Uf$ in \mathbb{C}/U . Because of the way the comma category is defined, this implies that there is a morphism $\text{cod}(\eta) : X \rightarrow Y$ in \mathbb{C} . We think of f as a ‘causal neighbourhood’ of p . Although we could, if we wished, choose any causal neighbourhood, it is best to choose one which comes with a reasoning principle. Since we are supposing that ‘changes propagate with a bounded velocity’, there ought to be a minimal causal neighbourhood such that any sequence of changes which could have contributed to the updated state p must have passed into it. This amounts to saying that for every other causal neighbourhood, that is for every $g : M \rightarrow Z$ in \mathbb{C} with a morphism $\gamma : p \rightarrow Ug$ in \mathbb{C}/U , there is a unique $\hat{\gamma} : f \rightarrow g$ in \mathbb{C}/\mathbb{C} such that $\gamma = U\hat{\gamma} \circ \eta$.

Since we want this assignment of a causal neighbourhood to vary naturally as we vary p , it amounts to an adjunction; but an adjunction where? The simplest option is to consider the functor $U_{\triangleleft} : \mathbb{C}/\mathbb{C} \rightarrow \mathbb{C}/U$ which takes an object $(X, f : X \rightarrow Y, Y)$ of \mathbb{C}/\mathbb{C} to (UX, Uf, Y) in \mathbb{C}/U . Note that the target component Y is unchanged. It is for this functor that we demand a left adjoint, say F_{\triangleleft} .

What does this all mean in our example? Consider the occurrence of \blacksquare at position 2 in $\blacksquare \square \blacksquare$. We must find a causal neighbourhood which explains why it occurs. We choose the occurrence of $\square \square \blacksquare$ at the end of $\blacksquare \square \square \blacksquare$. This updates precisely to the part we wanted, and it must be the universal explanation since anything smaller would only be updated to a single cell. Now the reader may be troubled by a subtle point: one might have expected that the explanation for the above should be the occurrence of $\square \blacksquare$ at position 3 in the input, since we know intuitively that its left-hand square will turn black. If we had defined U to make use of this knowledge, however, then we could not have defined F_{\triangleleft} in a functorial manner, since we would have had $U(\blacksquare \square) = \blacksquare \blacksquare$. But then the middle \blacksquare could be explained just as well by $\blacksquare \square$ on the left as by $\square \blacksquare$ on the right. This is why we think in terms of neighbourhoods through which *all* causal influences on an updated part must have passed. This issue is related to the problem of ‘overdetermination’ familiar to philosophers (see e.g. [9]).

To enforce the bounded speed of propagation, we need to stipulate that F_{\triangleleft} have a certain finiteness property. For any given object A , there are many possible shapes of causal neighbourhood for parts of shape A , depending on the context in which we find them. However, since there is a bound on the size of these, and there can only be finitely many objects of this size, we know that any part $p : A \rightarrow UX$ has only finitely many different shapes of possible causal neighbourhoods. Therefore we demand whenever we have a subcategory \mathbb{S} of \mathbb{C}/U whose image under the domain functor, $\text{dom}[\mathbb{S}]$, is finite, the domain of its image under F_{\triangleleft} , which is $(\text{dom} \circ F_{\triangleleft})[\mathbb{S}]$, is also finite.

In our example, consider all morphisms out of the object \blacksquare into outputs of U . Although this is an infinite set of morphisms (because there are an infinite number of strings containing \blacksquare), each of these occurrences is explained by a substring in the input of one of the forms $\blacksquare \blacksquare \blacksquare$, $\blacksquare \square$, $\blacksquare \square \square$, $\square \square \blacksquare$, $\square \blacksquare \blacksquare$, or $\square \square \blacksquare$. If we did not stipulate that this collection be finite, then different occurrences of \blacksquare in different outputs of U could have been explained by substrings of arbitrary length in the input. This would allow such behaviour as outputting a single cell which is black if and only if the input codes a halting Turing machine!

The example we have given is somewhat restrictive, since the partial states we mention always shrink, and otherwise do not change shape very much. This is not a necessary limitation. For instance, we could introduce special ‘blank’ cells on the ends of the strings which, rather than being removed by U , are duplicated to allow the working area to grow. More radical changes to the intuitive shape of parts are possible, but harder to describe. Putting all our conditions together we get the following.

Definition 1. A **categorical Gandy machine** is an endofunctor $U : \mathbb{C} \rightarrow \mathbb{C}$ where \mathbb{C} is a locally finite category with a finite dense subcategory, such that:

- the induced functor $U^\triangleleft : \mathbb{C}/\mathbb{C} \rightarrow \mathbb{C}/U$ has a left adjoint F_{\triangleleft} , and
- for all subcategories \mathbb{S} of \mathbb{C}/U such that $\text{dom}[\mathbb{S}]$, is finite, we have that $(\text{dom} \circ F_{\triangleleft})[\mathbb{S}]$ is also finite.

We are now in a position to prove that every U satisfying these conditions is computable. We do this in two steps.

Proposition 1. *Let $U : \mathbb{C} \rightarrow \mathbb{C}$ be a categorical Gandy machine. Let \mathbb{G} be a dense subcategory of \mathbb{C} which gives an induced inclusion $\mathbb{G}/U \hookrightarrow \mathbb{C}/U$. Then U is the pointwise (left) Kan extension of dom along $(\text{dom} \circ F_{\triangleleft})$ when both are precomposed with this inclusion.*

Proof. Let η be the unit of the adjunction between U^\triangleleft and F_{\triangleleft} , and consider the diagram

$$\begin{array}{ccccc}
 \mathbb{G}/U & \hookrightarrow & \mathbb{C}/U & \xrightarrow{\text{dom}} & \mathbb{C} \\
 \downarrow & & \parallel & & \parallel \\
 \mathbb{C}/U & \xlongequal{\quad} & \mathbb{C}/U & \xrightarrow{\text{dom}} & \mathbb{C} \\
 \downarrow F_{\triangleleft} & \swarrow \eta & \parallel & & \parallel \\
 \mathbb{C}/\mathbb{C} & \xrightarrow{U^\triangleleft} & \mathbb{C}/U & \xrightarrow{\text{dom}} & \mathbb{C} \\
 \downarrow \text{dom} & & & & \parallel \\
 \mathbb{C} & \xrightarrow{\quad U \quad} & & & \mathbb{C}.
 \end{array}$$

The top row is a pointwise Kan extension by density of \mathbb{G} . The middle row is a pointwise Kan extension since its left-hand square comes from an adjunction. It is thus an absolute Kan extension, preserved by dom . The bottom row is a pointwise Kan extension since it commutes and dom has a section. Then the result follows by pasting of pointwise Kan extensions. \square

Corollary 2. *Let $U : \mathbb{C} \rightarrow \mathbb{C}$ be a categorical Gandy machine. Suppose that objects of \mathbb{C} are represented by diagrams in the finite dense subcategory for which they are the colimit. Then U is computable.*

Proof. Suppose for some object X of \mathbb{C} , we want to compute UX . The above Proposition implies that UX is the colimit of all objects which can occur as the domain of an object g of \mathbb{G}/U such that the domain of $F_{\triangleleft}g$ admits a morphism into X . Since \mathbb{G} is finite, $\text{dom}[\mathbb{G}/U] = \mathbb{G}$ is finite. Hence, $\text{dom}[F_{\triangleleft}[\mathbb{G}/U]]$ is finite. This is the category in which the diagram whose colimit is UX must take its values. Note that, in addition to being finite, it is independent of X . Therefore, it can be ‘precomputed’ for use in evaluating U .

In order to evaluate UX , all commuting triangles with one side in this precomputed category and apex X must be computed. Since \mathbb{C} is locally finite, this is a finite diagram (and it can actually be computed since a morphism into X is determined by its action on morphisms in \mathbb{G} , and one can ‘try all possibilities’ for such actions; if desired, a similar idea can be used to ensure we have the ‘standard’ diagram whose colimit is X). \square

4 Discussion and Future Work

The present work is the first step in a programme to study spatial models of computation from a categorical perspective. Our original motivation was to study the flow of information in such models along the lines of [8]. We are particularly interested in the question of where in the spatially extended state is stored which piece of information about the computation. In previous work we lacked a good axiomatization of suitable structures which we now provide. We believe this framework to be quite robust, and in this paper we show how its definition may be thought to arise from Gandy’s principles.

A natural next step is to strengthen the connections with these principles by investigating whether every example satisfying Gandy’s original axioms can be interpreted in this framework. One obvious hurdle to overcome is that one has to choose a sensible category of states. This can not always have Gandy’s set of states as objects. For example, Gandy points out that the set of finite structures for a first-order logical signature satisfies his axioms. However, if the signature contains function symbols, then the category of structures, while locally finite, does not have a finite dense subcategory. For example, consider a signature with a single function symbol, and interpret this by successor taken *modulo* n on the set $\{0, \dots, n - 1\}$. These structures each admit no incoming morphisms from any other structure, and form an infinite family. The solution is to consider some notion of partial structure. This corresponds to adding objects not only for Gandy’s states, but also enough parts to cover at least the ones important for the structure of the machine.

It may also happen that the definition we give corresponds to a well-behaved subclass of Gandy’s machines (and perhaps to well-behaved classes of examples of the other models inspired by Gandy). An example of the limitations of the present model is given by the functor on the category of finite directed graphs which, wherever it finds a path of length two $x \rightarrow y \rightarrow z$ in its input, adds a direct edge $x \rightarrow z$, taking a single step carried out by the obvious algorithm for transitive closure. This is not an instance of the present framework, since it will add the diagonal to a rectangle like

$$\begin{array}{ccc} \bullet & \longrightarrow & \bullet \\ \downarrow & & \downarrow \\ \bullet & \longrightarrow & \bullet \end{array}$$

but there is no unique smallest causal neighbourhood. This is related to the problem of overdetermination mentioned above. This operation is, intuitively, locally deterministic for the usual notion of ‘locally’ in a graph. However, as we discussed earlier, if nodes of arbitrarily high degree are allowed, then our definition takes a strange view of the meaning of ‘local’.

Comparisons with later models may be more direct. For example, in the graph-based model of [1], a definition of local determinism is given in which a phenomenon in an updated graph is explained by a sub-graph of bounded radius around an ‘antecedent’ in the previous state. This is shown to be equivalent to taking a union of local updates. The analogy between this result and Proposition 1 above raises the question whether the other results of [1] have counterparts in the present abstract setting.

Other important questions are suggested by the key role played by finite categories in Definition 1. It would be interesting if the present extrinsic notion of finite category could be replaced by a more intrinsic condition. In [4], the impact on models of computation of varying the notion of finiteness is discussed.

Beyond this, in future work we plan to return to our original aims, and put this axiomatization to use to give a cleaner account of [8], and the flow of information in spatial models.

Acknowledgements. We would like to thank Pouya Adrom for a timely pointer to Kan extensions and Francisco Lobo for a number of very fruitful discussions about the structures we use.

References

- [1] Pablo Arrighi & Gilles Dowek (2012): *Causal Graph Dynamics*. In Artur Czumaj, Kurt Mehlhorn, Andrew Pitts & Roger Wattenhofer, editors: *Automata, Languages, and Programming*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 54–66. Available at <http://arxiv.org/abs/1202.1098>.
- [2] Pablo Arrighi, Simon Martiel & Zizhu Wang (2013): *Causal Dynamics of Discrete Surfaces*. In: *Proceedings 9th International Workshop on Developments in Computational Models, DCM 2013, Buenos Aires, Argentina, 26 August 2013.*, pp. 30–40, doi:10.4204/EPTCS.144.3. Available at <https://doi.org/10.4204/EPTCS.144.3>.
- [3] Robin Gandy (1980): *Church's Thesis and Principles for Mechanisms*. In: *The Kleene Symposium*, North-Holland, pp. 123–148.
- [4] Edward Hermann Haeusler (2015): *Finiteness and Computation in Toposes*. In: *Proceedings of the Eleventh International Workshop on Developments in Computational Models, DCM 2015, Cali, Colombia, October 28, 2015.*, pp. 61–77, doi:10.4204/EPTCS.204.6. Available at <https://doi.org/10.4204/EPTCS.204.6>.
- [5] Marina Christina Lehner (2014): *All Concepts are Kan Extensions: Kan Extensions as the Most Universal of the Universal Constructions*. Undergraduate senior thesis, Harvard College. Available at www.math.harvard.edu/theses/senior/lehner/lehner.pdf.
- [6] Saunders MacLane (1971): *Categories for the Working Mathematician*. Springer-Verlag, New York. Graduate Texts in Mathematics, Vol. 5.
- [7] Adam Obtułowicz (2012): *Generalized Gandy-Păun-Rozenberg Machines for Tile Systems and Cellular Automata*. In Marian Gheorghe, Gheorghe Păun, Grzegorz Rozenberg, Arto Salomaa & Sergey Verlan, editors: *Membrane Computing*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 314–332.
- [8] Joseph Razavi (2017): *Information Flow in Spatial Models of Computation*. Ph.D. thesis, The University of Manchester. Available at https://www.research.manchester.ac.uk/portal/files/59970154/FULL_TEXT.PDF.
- [9] Jonathan Schaffer (2003): *Overdetermining Causes*. *Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition* 114(1/2), pp. 23–45. Available at <http://www.jstor.org/stable/4321372>.
- [10] Wilfried Sieg & John Byrnes (1996): *K-graph machines: generalizing Turing's machines and arguments*, pp. 98–119. *Lecture Notes in Logic* 6, Springer-Verlag, Berlin. Available at <https://projecteuclid.org/euclid.lnl/1235417017>.
- [11] Wilfried Sieg & John Byrnes (1999): *An Abstract Model For Parallel Computations: Gandy's Thesis*. *The Monist* 82(1), pp. 150–164.
- [12] Ross Street (1974): *Fibrations and Yoneda's lemma in a 2-category*. In: *Category Seminar, Lecture Notes in Mathematics* 420, Springer Verlag, pp. 104–133.