An Equational Notion of Lifting Monad*

Anna Bucalo
Dipartimento di Matematica e Informatica
Università di Udine, Italy

Carsten Führmann
School of Computer Science
University of Birmingham, England

Alex Simpson
LFCS, Division of Informatics
University of Edinburgh, Scotland

Abstract

We introduce the notion of an *equational lifting monad*: a commutative strong monad satisfying one additional equation (valid for monads arising from partial map classifiers). We prove that any equational lifting monad has a representation by a partial map classifier such that the Kleisli category of the former fully embeds in the partial category of the latter. Thus equational lifting monads precisely capture the equational properties of partial maps as induced by partial map classifiers. The representation theorem also provides a tool for transferring non-equational properties of partial map classifiers to equational lifting monads. It is proved using a direct axiomatization of Kleisli categories of equational lifting monads. This axiomatization is of interest in its own right.

1 Introduction

Ever since Moggi's work [13, 14], the use of strong monads has provided a structural discipline underpinning the categorical approach to denotational semantics. The underlying idea is to make a denotational distinction between the operational notions of *value* and *computation* by modelling them in two separate categories. The category of values, **C**, is a category of total functions in which the usual datatypes are given their standard universal properties. Programs, however, are interpreted in the category of computations, which is obtained (at least in the call-by-value case) as the Kleisli category of a strong monad on **C** embodying a suitable "notion of computation". The precise nature of the notion of computation varies with the programming language modelled. In general, it will cater for the various kinds of possible computational effect such as: nontermination, nondeterminism, input/output, ... (see [14]).

In this paper we concern ourselves solely with one of the simplest forms of computational effect: deterministic computation with possible nontermination. This notion of computation is naturally modelled using partial functions. Thus, in a categorical setting, one looks for monads embodying notions of partiality.

The study of such "lifting" monads goes back to work of Mulry, Rosolini and Moggi in the 1980s [15, 17, 12]. In particular, in his PhD thesis [17], Rosolini considered a general categorical approach to partiality (based on the associated notions of *dominion*, see Section

*This research was carried out at the LFCS with the support of: a CNR Research Grant (Bucalo); an EU TMR Marie Curie Research Training Grant (Führmann); and EPSRC Research Grant GR/K06109 (Simpson).

3, and *dominance*) and proved representation results into categories of partial functions on presheaf toposes. In computer science, this categorical approach to partiality has proved its value through applications in *axiomatic domain theory* [5, 6] and *synthetic domain theory* [8].

In spite of the above, there are reasons to look for more general approaches to partiality. In particular, the notion of dominion requires every partial map to have its domain of definition represented as an object of the category. There are cases in which it is arguable that such domain objects should not be assumed as primitive. For example, the requirement of their existence prevents one from building syntactic models in which the objects are the types of a programming language and the morphisms are programs (unless either the programs are very simple or the types are very complex). It might be argued that such syntactic models should not lie within the realm of semantics. But one of the elegant features of category theory is that it potentially allows syntax and semantics to be treated on a par, with term models arising as (quotients of) free categories with structure.

In this paper we identify the properties a strong monad must possess in order for its Kleisli category to behave like an induced category of partial maps. The standard examples here are the monads determined by partial map classifiers with respect to a dominion — we review the properties of such *dominical lifting monads* in Section 3. However, as motivated above, we aim for an axiomatization of lifting monads without making reference to any notion of domain. The domain-free axiomatization will explicitly allow the class of models to include natural examples (such as the term models mentioned above) that would otherwise be excluded. In Section 4, we shall present an example of such a term model, based on call-by-value PCF.

Section 4 also contains our principal definition, the notion of *equational lifting monad*, and presents our main results. Theorem 1 shows that Kleisli categories of equational lifting monads fully embed in Kleisli categories of dominical lifting monads via structure-preserving functors. An immediate corollary, Corollary 1, characterises conditions (applying to the original category itself, rather than to the Kleisli category) under which stronger forms of representation hold. These results justify the definition of equational lifting monad. They also have applications to showing that equational lifting monads inherit certain non-equational properties from dominical lifting monads (Corollaries 2 and 3).

The proof of Theorem 1 occupies the remainder of the paper. However, much of the development directed towards its proof is of independent interest.

In Sections 5 and 6 we provide an alternative perspective on equational lifting monads, by giving a direct axiomatization of the categorical structure of their Kleisli categories. This work continues in a tradition, exemplified by Robinson and Rosolini's notion of *p-category* [16], of providing direct, domain-free axiomatizations of categories of partial maps. In the case of p-categories, the axiomatized categories correspond to categories of partial maps with a suitable product structure. The structure we require is that of a p-category that, in addition, is the Kleisli category of an equational lifting monad on its associated total category. Our axiomatization of such categories is obtained by extending the notion of *abstract Kleisli category* [7], which provides a direct axiomatization of categories that arise as Kleisli categories, with the necessary additional structure, to obtain the notion of *abstract Kleisli p-category*.

In Section 7 we characterise when an abstract Kleisli p-category arises as the Kleisli category of a dominical lifting monad (Theorem 2). The characterisation, similar to [16, Theorem 1.7], requires a collection of idempotents in the Kleisli category to split. We then show that the structure of an abstract Kleisli p-category is preserved under a formal idempotent splitting, allowing any abstract Kleisli p-category to be embedded in a dominical one. This preservation result is not *a priori* obvious, as not all the abstract Kleisli p-category structure is natural (in the technical sense of the word).

Finally, in Section 8, we complete the proof of Theorem 1, making crucial use of the results and constructions from the previous sections. We also characterise when an equational lifting monad is dominical (Theorem 3), and discuss other miscellaneous properties of lifting monads.

2 Preliminaries

In this section, we briefly review facts we require about monads [11, 1], monoidal categories [11], strong monads [10, 14] and idempotent splittings. The reader may prefer to skip this section, and refer back to it as and when necessary.

First, some general remarks about our policy towards structure-preserving functors between categories with additional structure. In all cases, the correct general notion of structure-preserving functor should be given by a functor with specified natural transformations satisfying appropriate coherence conditions. However, in this paper, we shall always assume a more restrictive notion of structure-preserving functor, in which the natural transformations are required to be isomorphisms. (In the literature, such special structure-preserving functors are often identified by the adjective *strong*.) In the special case in which the assumed natural isomorphisms are in fact identities, we call the functors *strict*. Throughout the paper, we adopt the policy of stating results for arbitrary structure-preserving functors, mediated by natural isomorphisms, but we give proofs just in the special case of strict functors. This serves to reduce notational clutter in the proofs, which all adapt easily to the non-strict case.

Given a monad (T, η, μ) on a category $\mathbf{C}$, we write $\mathbf{C}_T$ for the Kleisli category, $J : \mathbf{C} \to \mathbf{C}_T$ for the associated left-adjoint functor, and $K : \mathbf{C}_T \to \mathbf{C}$ for its right adjoint. We shall often refer to a monad by just naming its underlying functor.

We say that a monad T satisfies: the *mono property* if all components $A \xrightarrow{\eta} TA$ are mono; and the *equaliser property* if all diagrams $A \xrightarrow{\eta} TA \underset{T\eta}{\overset{\eta}{\rightrightarrows}} T^2A$ are equalisers. The mono property is easily seen to be equivalent to the faithfulness of T, and also to the faithfulness of J. The equaliser property trivially implies the mono property; in fact it is equivalent to every component $A \xrightarrow{\eta} TA$ being a regular mono [1, Lemma 6, p.110].

Suppose that (T, η, μ) is a monad on $\mathbf{C}$ and (T', η', μ') is a monad on $\mathbf{C}'$. We say that a functor $F : \mathbf{C} \to \mathbf{C}'$ is *monad preserving* if it comes with a natural transformation $\iota : FT \Rightarrow T'F$ such that $\eta' = \iota \circ F\eta$ and $\mu' \circ T'\iota \circ \iota = \iota \circ F\mu$, and also (as discussed above) ι is an isomorphism.

A monad-preserving functor $F : \mathbf{C} \to \mathbf{C}'$ determines a functor $F_K : \mathbf{C}_T \to \mathbf{C}'_{T'}$ between Kleisli categories, mapping any $f \in \mathbf{C}_T(A, B)$, given by $A \xrightarrow{f} TB$ in $\mathbf{C}$, to $\iota \circ Ff$ in $\mathbf{C}_{T'}(FA, FB)$. We say F is *Kleisli full* (resp. *Kleisli faithful*) to mean that the induced functor F_K is full (resp. faithful). Observe that full and faithful imply Kleisli full and Kleisli faithful respectively. The converse implications are closely linked to properties of η.

Proposition 2.1 *Suppose T is a monad on $\mathbf{C}$, T' is a monad on $\mathbf{C}'$ and $F : \mathbf{C} \to \mathbf{C}'$ is monad preserving.*

1. *Suppose F is Kleisli faithful and the mono property holds for T'. Then F is faithful if and only if the mono property holds for T.*

2. *Suppose F is Kleisli full and Kleisli faithful and the equaliser property holds for T'. Then F is full and faithful if and only if the equaliser property holds for T.*

Proof As stated above, we give the proof in the case that F is strict monad preserving.

1. Let J' be the functor $\mathbf{C}' \longrightarrow \mathbf{C}'_{T'}$. The mono property on T (resp. T') is equivalent to the faithfulness of J (resp. J'). So the equivalence follows from the equality $F_K J = J'F$.

2. For the right-to-left implication, the faithfulness of F follows from *1*. To prove fullness, take any $f \in \mathbf{C}'(FA, FB)$. We must find g such that $f = Fg$. Consider $\eta' \circ f \in \mathbf{C}'_{T'}(FA, FB)$. As F_K is full, there exists $h \in \mathbf{C}_T(A, B)$ such that $Fh = \eta' \circ f$. Then:

$$FT\eta \circ Fh = T'\eta' \circ \eta' \circ f = \eta' \circ \eta' \circ f = F\eta \circ Fh.$$

   So $T\eta \circ h = \eta \circ h$, by the faithfulness of F_K. Thus, as T satisfies the equaliser property, there exists $g \in \mathbf{C}(A, B)$ such that $\eta \circ g = h$. Finally

$$\eta' \circ Fg = F\eta \circ Fg = Fh = \eta' \circ f$$

   and so, as η' is mono, $Fg = f$.

   For the converse, consider $f \in \mathbf{C}(A, TB)$ such that $\eta \circ f = T\eta \circ f$. Then $\eta' \circ Ff = T'\eta' \circ Ff$. So, as T' satisfies the equaliser property, there exists $h \in \mathbf{C}'(FA, FB)$ such that $Ff = \eta' \circ h$. Now, since F is full, there exists $g \in \mathbf{C}(A, B)$ such that $Fg = h$. So we obtain

$$Ff = \eta' \circ Fg = F\eta \circ Fg = F(\eta \circ g).$$

   As F is faithful, $f = \eta \circ g$. Uniqueness follows from the fact that η is mono, as η' is mono and F is faithful.

□

We shall also be interested in conditions under which monad-preserving functors preserve (existing) pullbacks (although the proposition below applies equally well to any other type of limit).

Proposition 2.2 *Suppose T is a monad on $\mathbf{C}$, T' is a monad on $\mathbf{C}'$ and $F : \mathbf{C} \to \mathbf{C}'$ is monad preserving.*

1. *If T and F_K preserve pullbacks and T' reflects pullbacks then F preserves pullbacks.*

2. *If T' and F preserve pullbacks and F_K reflects pullbacks then T preserves pullbacks.*

Proof First, K preserves and reflects limits (it is a right adjoint of descent type [1]). So $T = KJ$ preserves pullbacks if and only if J does. Similarly, T' preserves pullbacks if and only if J' does.

1. If T and F_K preserve pullbacks then so does $K'F_K J = T'F$. So if T' reflects pullbacks, F preserves them.

2. If T' and F preserve pullbacks then so does $J'F = F_K J$. So if F_K reflects pullbacks, J preserves them, hence so does T.

$\square$

Symmetric monoidal structure on $\mathbf{C}$ is given by a functor $\otimes : \mathbf{C} \times \mathbf{C} \to \mathbf{C}$, an object I (the *unit*), and natural isomorphisms $\alpha : (A \otimes B) \otimes C \longrightarrow A \otimes (B \otimes C)$, $\gamma : A \otimes B \longrightarrow B \otimes A$, $\lambda : A \otimes I \longrightarrow A$, and $\rho : I \otimes A \longrightarrow A$ satisfying the coherence diagrams in [11, Ch.VII, § 1 & 7]. Given symmetric monoidal structure $(\otimes, I, \alpha, \gamma, \lambda, \rho)$ on $\mathbf{C}$ and $(\otimes', I', \alpha', \gamma', \lambda', \rho')$ on $\mathbf{C}'$, we say that a functor $F : \mathbf{C} \to \mathbf{C}'$ is *monoidal* if it comes with a natural transformation $\theta_2 : F \otimes F \Rightarrow F(Id \otimes Id)$ and a morphism $\theta_0 : I' \longrightarrow FI$ subject to the coherence conditions in [11, Ch. XI], and in addition (as discussed above) θ_2 and θ_0 are isomorphisms.

Given symmetric monoidal structure on $\mathbf{C}$, a *strong monad* (T, η, μ, t) on $\mathbf{C}$ is a monad together with a natural transformation $A \otimes TB \xrightarrow{t} T(A \otimes B)$ (its *strength*) that satisfies the four equations:

$$
\begin{aligned}
T\rho \circ t &= \rho \\
T\alpha \circ t &= t \circ (id \times t) \circ \alpha \\
t \circ (id \otimes \eta) &= \eta \\
t \circ (id \otimes \mu) &= \mu \circ Tt \circ t
\end{aligned}
$$

A dual *"costrength"* $TA \otimes B \xrightarrow{t'} T(A \otimes B)$ is defined by $t' = T\gamma \circ t \circ \gamma$. A strong monad is *commutative* if $\mu \circ Tt' \circ t \;=\; \mu \circ Tt \circ t'$, in which case we write $TA \otimes TB \xrightarrow{\psi} T(A \otimes B)$ for this map (the *symmetric strength*). Given a strong monad (T', η', μ', t') on $\mathbf{C}'$, we say that $F : \mathbf{C} \to \mathbf{C}'$ is *strong-monad preserving* if it is monad preserving, monoidal and also $\iota \circ Ft \circ \theta_2 = T'\theta_2 \circ t' \circ (id \otimes \iota)$.

In this paper, we shall only be interested in strong monads arising from monoidal structure given by specified finite products, in which case we always take $\alpha, \gamma, \lambda, \rho$ to be the usual natural transformations defined from tuples of projections (e.g. $\alpha = \langle \pi_1 \circ \pi_1, \langle \pi_2 \circ \pi_1, \pi_2 \rangle \rangle$). In such cases, monoidal functors (and hence strong-monad-preserving functors) automatically preserve the finite product structure (this is only true because we are assuming that structure is preserved up to isomorphism).

In any category $\mathbf{C}$, an *idempotent* is an endomorphism $A \xrightarrow{a} A$ such that $a = a \circ a$. An idempotent a is said to *split* if there exist maps $A \xrightarrow{r} A' \xrightarrow{m} A$ such that $r \circ m = id_{A'}$ and $m \circ r = a$. For any collection S of idempotents, the category $\mathbf{Split}_S(\mathbf{C})$ has as objects the idempotents in S, and the morphisms from $A \xrightarrow{a} A$ to $B \xrightarrow{b} B$ are those $A \xrightarrow{f} B$ such that $b \circ f = f = f \circ a$. The identity on an object a is given by a itself. Composition is inherited from $\mathbf{C}$. If S contains all identities in $\mathbf{C}$ then there is a full and faithful functor $I : \mathbf{C} \to \mathbf{Split}_S(\mathbf{C})$ mapping each object A to id_A. Moreover, for every idempotent $a \in S$, the idempotent $I(a)$ splits in $\mathbf{Split}_S(\mathbf{C})$. Further, the functor I preserves all existing limits and colimits (because Yoneda and its dual both factor through I).

3 Dominical lifting monads

In this section we review the standard categorical approach to partiality and the notion of lifting monad it induces. All the definitions and results are contained (at least implicitly) in [17]. A good range of computationally motivated examples can be found in [5].

Definition 3.1 A *dominion* on a category $\mathbf{C}$ is given by a collection $\mathbf{D}$ of monomorphisms in $\mathbf{C}$ that is closed under composition, contains every isomorphism, and is closed under pullback along arbitrary morphisms.

Let $\mathbf{C}$ be a category and let $\mathbf{D}$ be a dominion on it. We use the symbol $\hookrightarrow$ to represent monos in $\mathbf{D}$.

A $\mathbf{D}$-*partial map* from an object A of $\mathbf{C}$ to an object B is given by an equivalence class of spans of the form $A \xleftarrow{m} A' \xrightarrow{f} B$, where m is in $\mathbf{D}$, under the equivalence identifying (f, m) and (f', m') if there exist maps i, i' such that: $m' \circ i = m$, $f' \circ i = f$, $m \circ i' = m'$, and $f \circ i' = f'$. (The equations involving monos imply that i and i' are inverse isomorphisms.)

The conditions imposed on $\mathbf{D}$ are just what is needed for the collection of partial maps to form a category (with the same objects as $\mathbf{C}$). In particular, composition is performed using closure under pullbacks. We write $\mathbf{Ptl_D}(\mathbf{C})$ for the category of partial maps. There is an evident faithful functor $J : \mathbf{C} \to \mathbf{Ptl_D}(\mathbf{C})$, which is the identity on objects.

Definition 3.2 We say that $\mathbf{C}$ has $\mathbf{D}$-*partial map classifiers* if, for every object B, there exists an object LB and a morphism $B \xhookrightarrow{\eta} LB$ in $\mathbf{D}$ such that, for every partial map $A \xleftarrow{m} A' \xrightarrow{f} B$, there exists a unique *characteristic map* $A \xrightarrow{g} LB$ such that the square below is a pullback.

$$\begin{tikzcd}
A' \arrow[r, "f"] \arrow[d, hook, "m"'] \arrow[dr, phantom, "\lrcorner", very near start] & B \arrow[d, hook, "\eta"] \\
A \arrow[r, "g"] & LB
\end{tikzcd}$$

The existence of partial map classifiers is equivalent to the representability in $\mathbf{C}$ of all functors $\mathbf{Ptl_D}(\mathbf{C})(J(-), B) : \mathbf{C}^{\mathrm{op}} \to \mathbf{Set}$, which is, in turn, equivalent to the existence of a functor $K : \mathbf{Ptl_D}(\mathbf{C}) \to \mathbf{C}$ right adjoint to J. When such a right adjoint K exists, its unit η provides $\mathbf{D}$-partial map classifiers, where $\mathbf{D}$ is the collection of all pullbacks of components of η (which is indeed a dominion). As for any adjunction, the composite $L = KJ : \mathbf{C} \to \mathbf{C}$ has an associated monad structure. Further, because $J : \mathbf{C} \to \mathbf{Ptl_D}(\mathbf{C})$ is bijective on objects, the category $\mathbf{Ptl_D}(\mathbf{C})$ of partial maps is (isomorphic to) the Kleisli category of the monad.

The aim of this paper is to analyse what properties a monad must enjoy in order for its Kleisli category to behave like a category of partial maps. Monads derived, as above, from $\mathbf{D}$-partial map classifiers will be our paradigmatic examples of such "lifting" monads. For later purposes, it will be useful to have a name for these.

Definition 3.3 We say that a monad (L, η, μ) on a category $\mathbf{C}$ is a *dominical lifting monad* if there exists a dominion $\mathbf{D}$ on $\mathbf{C}$ such that $\mathbf{C}$ has $\mathbf{D}$-partial map classifiers and (L, η, μ) is the monad determined by the adjunction between $\mathbf{C}$ and $\mathbf{Ptl_D}(\mathbf{C})$.

It is instructive to analyse the structure of a dominical lifting monad (L, η, μ) directly in terms of the associated $\mathbf{D}$-partial map classifiers. The unit is given by the family $A \xrightarrow{\eta} LA$ required by Definition 3.2. The multiplication, $L^2A \xrightarrow{\mu} LA$, is the unique map making the square below a pullback.

$$\begin{tikzcd}
A \arrow[r, "id"] \arrow[d, hook, "\eta \circ \eta"'] \arrow[dr, phantom, very near start, "\lrcorner"] & A \arrow[d, hook, "\eta"] \\
L^2A \arrow[r, "\mu"] & LA
\end{tikzcd} \tag{1}$$

Dominical lifting monads have many properties not shared by arbitrary monads. For example, the natural transformations η and μ are both *cartesian*, i.e. all their naturality squares

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "\eta"'] \arrow[dr, phantom, very near start, "\lrcorner"] & B \arrow[d, "\eta"] \\
LA \arrow[r, "Lf"] & LB
\end{tikzcd}
\qquad
\begin{tikzcd}
L^2A \arrow[r, "L^2f"] \arrow[d, "\mu"'] \arrow[dr, phantom, very near start, "\lrcorner"] & L^2B \arrow[d, "\mu"] \\
LA \arrow[r, "Lf"] & LB
\end{tikzcd} \tag{2}$$

are pullbacks. The cartesianness of η implies that L satisfies the equaliser property. In addition, L preserves existing pullbacks. Also, L reflects pullbacks (using a cube of pullbacks whose four walls are η squares).

Now suppose $\mathbf{C}$ has specified finite products. Then L is a strong monad, with a unique strength $A \times LB \xrightarrow{t} L(A \times B)$, defined as the characteristic map below.

$$\begin{tikzcd}
A \times B \arrow[r, "id"] \arrow[d, hook, "id \times \eta"'] \arrow[dr, phantom, very near start, "\lrcorner"] & A \times B \arrow[d, hook, "\eta"] \\
A \times LB \arrow[r, "t"] & L(A \times B)
\end{tikzcd}$$

The costrength $LA \times B \xrightarrow{t'} L(A \times B)$ can be defined analogously. By pasting pullback squares, it is straightforward to verify that the monad is commutative, with its symmetric strength obtained as the unique morphism making the square below into a pullback.

$$\begin{tikzcd}
A \times B \arrow[r, "id"] \arrow[d, hook, "\eta \times \eta"'] \arrow[dr, phantom, very near start, "\lrcorner"] & A \times B \arrow[d, hook, "\eta"] \\
LA \times LB \arrow[r, "\psi"] & L(A \times B)
\end{tikzcd}$$

An important consequence of the discussion above is that, given $\mathbf{D}$-partial map classifiers $B \overset{\eta}{\hookrightarrow} LB$, there is a unique dominical lifting monad that has η as its unit. The action of L on morphisms is determined by the left-hand pullback of (2). Similarly, the multiplication μ is determined by pullback (1). Moreover, if $\mathbf{C}$ has specified finite products then the unique strength is determined by the defining pullback above.

4 Equational lifting monads

In the previous section, we investigated some of the properties of dominical lifting monads. The main definition of the paper aims to precisely identify their equational properties.

Definition 4.1 We say that a strong monad (L, η, μ, t), on a category $\mathbf{C}$ with specified finite products, is an *equational lifting monad* if it is commutative and the diagram below commutes

$$\begin{tikzcd}
LA \arrow[r, "\Delta"] \arrow[dr, "L\langle \eta,\, id \rangle"'] & LA \times LA \arrow[d, "t"] \\
& L(LA \times A)
\end{tikzcd} \tag{3}$$

where $LA \xrightarrow{\Delta} LA \times LA$ is the diagonal of the product.

In the previous section we established that every dominical lifting monad is a commutative strong monad. Moreover, Diagram (3) commutes because both diagrams below are pullbacks (the second because η is cartesian).

$$\begin{tikzcd}
A \arrow[r, "\langle \eta,\, id \rangle"] \arrow[d, hook, "\eta"'] \arrow[dr, phantom, "\lrcorner", very near start] & LA \times A \arrow[r, "id"] \arrow[d, hook, "id \times \eta"'] \arrow[dr, phantom, "\lrcorner", very near start] & LA \times A \arrow[d, hook, "\eta"] \\
LA \arrow[r, "\Delta"] & LA \times LA \arrow[r, "t"] & L(LA \times A)
\end{tikzcd}
\qquad
\begin{tikzcd}
A \arrow[r, "\langle \eta,\, id \rangle"] \arrow[d, hook, "\eta"'] \arrow[dr, phantom, "\lrcorner", very near start] & LA \times A \arrow[d, hook, "\eta"] \\
LA \arrow[r, "L\langle \eta,\, id \rangle"] & L(LA \times A)
\end{tikzcd}$$

So, by the uniqueness of characteristic maps, $L\langle \eta,\, id \rangle = t \circ \Delta$. Thus every dominical lifting monad, on a category with finite products, is an equational lifting monad. We give an example of an equational lifting monad that is not dominical at the end of this section.

The goal of this paper is twofold: to show that equational lifting monads possess all the equational properties of dominical lifting monads; and to characterise the precise conditions under which the other (non-equational) properties of dominical lifting monads hold also for equational lifting monads.

We start with a simple proposition establishing two basic equational properties.

Proposition 4.2 *For an equational lifting monad (L, η, μ, t), the diagrams below commute.*

$$\begin{tikzcd}
LA \arrow[r, "\Delta"] \arrow[dr, "L\Delta"'] & LA \times LA \arrow[d, "\psi"] \\
& L(A \times A)
\end{tikzcd}
\qquad
\begin{tikzcd}
L^2A \arrow[r, "id"] \arrow[d, "\langle \mu,\, L! \rangle"'] & L^2A \\
LA \times L1 \arrow[r, "t"] & L(LA \times 1) \arrow[u, "L\pi_1"']
\end{tikzcd} \tag{4}$$

Proof The first equality follows from Equation (3) and the strong monad equations, by expanding $\psi = \mu \circ Lt' \circ t$. For the second equality, using (3) and the naturality of t, one obtains:

$$id = L\mu \circ L\eta = L\mu \circ L\pi_1 \circ L\langle \eta, id \rangle = L\pi_1 \circ L(\mu \times !) \circ t \circ \Delta = L\pi_1 \circ t \circ \langle \mu, L! \rangle$$

$\square$

The left-hand diagram above expresses that any equational lifting monad is *relevant* in the sense of Jacobs [9]. (Not every commutative relevant monad is an equational lifting monad. A simple counterexample is the $(-)^2$ monad on **Set**.)

The right-hand diagram has a couple of interesting consequences. One easy consequence is equation (7) of [2]: $L\pi_1 \circ t \circ \langle id, L! \rangle = L\eta : LA \longrightarrow L^2 A$, which was used there as part of a non-equational axiomatization of lifting monads. Another consequence is that the arrows $L1 \xleftarrow{L!} L^2 A \xrightarrow{\mu} LA$ are jointly monic (because $\langle \mu, L! \rangle$ is a split mono).

Our main theorem will state that every equational lifting monad can be sufficiently well "represented" by a dominical lifting monad.

Definition 4.3 Let L be a strong monad on $\mathbf{C}$. A *dominical representation* of L is given by a category $\mathbf{C}'$ with finite products and dominical lifting monad L' together with a strong-monad-preserving functor $F : \mathbf{C} \to \mathbf{C}'$.

We refer to a representation as being (Kleisli) full/faithful if the property mentioned holds of the representing functor F.

Both the commutativity equation and Diagram (3) can be viewed as equations between Kleisli category morphisms. Also, these equations hold in any dominical lifting monad. It follows that any strong monad with a Kleisli faithful dominical representation is necessarily an equational lifting monad. Our principal theorem gives a strengthened converse to this observation.

Theorem 1 *If L is an equational lifting monad then L has a dominical representation F that is Kleisli full and Kleisli faithful such that F_K preserves all limits and colimits existing in $\mathbf{C}_L$.*

The proof of Theorem 1 will be given in Section 8, after preparatory work occupying the remainder of the paper. As an immediate corollary, we obtain precise conditions for strengthened forms of representation to be possible.

Corollary 1 *Let L be an equational lifting monad.*

1. *L has a faithful and Kleisli full dominical representation if and only if it satisfies the mono property.*

2. *L has a full and faithful dominical representation if and only if it satisfies the equaliser property.*

3. *L has a full and faithful dominical representation that preserves existing finite limits if and only if L satisfies the equaliser property and preserves existing pullbacks.*

Proof Given Theorem 1, statements 1 and 2 are immediate from Proposition 2.1. Statement 3 follows from Proposition 2.2, using: the underlying functor of a dominical lifting monad preserves and reflects pullbacks (see Section 3); the functor F_K given by Theorem 1 preserves and reflects limits (it reflects them because it is full and faithful); and, as a dominical representation is assumed to preserve finite products, it preserves existing finite limits if and only if it preserves existing pullbacks. $\square$

It is worth explaining the significance of Theorem 1 and Corollary 1. The notion of dominical lifting monad corresponds to an accepted categorical notion of partiality, with the associated category of partial maps obtained as the Kleisli category. Our aim is to establish that, for equational lifting monads, the Kleisli category also acts just like a category of partial maps determined by a dominical lifting monad. Theorem 1 states in what sense this is indeed the case. Corollary 1 quantifies how well the base category of the equational lifting monad can be related to that of the dominical lifting monad.

We now give some applications of Theorem 1 and Corollary 1, illustrating how they can be used to establish non-equational properties of equational lifting monads. Recall the definition of a *cartesian* natural transformation (see the text before Diagram (2)).

Corollary 2 *For any equational lifting monad, μ is cartesian.*

Proof It suffices to apply the representing functor F to the right-hand diagram of (2) and note that all arrows involved are actually arrows in the Kleisli category $\mathbf{C}'_{L'}$. So the fullness and faithfulness of F_K and the cartesianness of μ' together imply the result. $\square$

The pullback property of μ can be exhibited explicitly. Consider the bottom-right cospan in the right-hand square of (2), and let $LA \xleftarrow{g} C \xrightarrow{h} L^2B$ be a cone for this diagram. The required universal map is $L\pi_1 \circ t \circ \langle g, h\rangle : C \longrightarrow L^2A$. In fact, in our original proof that μ is cartesian, we directly verified that this map has all the required properties. We shall not present this direct proof here, as it is extremely long (especially the verification of the equation $\mu \circ L\pi_1 \circ t \circ \langle g, h\rangle = g$). In contrast, it is pleasing that the cartesian property of μ falls out so easily from our representation theorems. Moreover, in theory, it is possible to extract a direct verification that $L\pi_1 \circ t \circ \langle g, h\rangle : C \longrightarrow L^2A$ is the required map, by unwinding the proof of Theorem 1.

Theorem 1 can be used in an analogous way to transfer other pullback properties of dominical lifting monads to equational lifting monads, for example, to establish that Diagram (1) is also a pullback for any equational lifting monad. However, not all pullbacks transfer automatically.

Corollary 3 *For any equational lifting monad, η is a cartesian natural transformation if and only if the equaliser property holds.*

Proof The left-to-right implication is easily proved for an arbitrary monad. The right-to-left implication, follows from Theorem 1 and Corollary 1, as the full and faithful dominical representation reflects pullbacks. $\square$

We end this section with the promised example of a computationally natural term category with an equational lifting monad that is not dominical.

Example 4.4 Consider a call-by-value version of PCF with types:

$$\tau \quad ::= \quad \mathit{unit} \mid \mathit{int} \mid \tau \times \tau' \mid \tau \to \tau'$$

and with terms as in, for example, [20, Ch. 11], but extended (in the obvious way) with a "singleton" type, *unit*. We define a category $(\mathbf{PCFv})_{\mathbf{t}}$ whose objects are types. To define the morphisms, we say that a closed term $t : \tau \to \tau'$ is *total* if, for all *values* (i.e. *canonical forms* in the terminology of [20, Ch. 11]) $c : \tau$, the evaluation of the application term $t(c)$ terminates (i.e. $t(c) \downarrow^e$ in the notation of [20, Ch. 11]). The morphisms from τ to τ' are equivalence classes of total closed terms of type $\tau \to \tau'$ modulo operational (contextual) equivalence. The composition of $\tau \xrightarrow{[t]} \tau'$ and $\tau' \xrightarrow{[t']} \tau''$ is given by $[\lambda x^\tau.\, t'(t(x))]$. Identities are obvious.

This category, $(\mathbf{PCFv})_{\mathbf{t}}$, has finite products given by the product operation on types. There is also a natural (syntactically defined) equational lifting monad. The underlying functor maps a type τ to the type $unit \to \tau$. The action on morphisms maps $\tau \xrightarrow{[t]} \tau'$ to $[\lambda e^{unit\to\tau}.\, (\lambda x^{unit}.\, t(e(x)))]$. The unit of the monad is given by $[\lambda x^\tau.\, \lambda y^{unit}.\, x]$ and the counit is given by $[\lambda e^{unit\to unit\to\tau}.\, \lambda x^{unit}.\, e(x)(x)]$. The strength of the monad is given by $[\lambda p^{\tau\times(unit\to\tau')}.\, (\lambda x^{unit}.\, \langle fst(p),\, snd(p)(x)\rangle)]$. With a bit of work, the commutativity of the monad corresponds to the operational equivalence:

$$(\lambda x.\, \lambda y.\, t(x)(y))(t_1)(t_2) \quad \equiv_o \quad (\lambda y.\, \lambda x.\, t(x)(y))(t_2)(t_1)$$

where t, t_1 and t_2 range over arbitrary terms (such that the expressions above are well-typed). Finally, the equational lifting monad equation (3) amounts to the operational equivalence:

$$\lambda e^{unit\to\tau}.\, \lambda x^{unit}.\, \langle e,\, e(x)\rangle \quad \equiv_o \quad \lambda e^{unit\to\tau}.\, \lambda x^{unit}.\, (\lambda y^\tau.\langle \lambda z^{unit}.y,\, y\rangle)(e(x))$$

Masochistically-inclined readers may wish to verify the above claims for themselves. There are essentially two tasks involved: first, translating all the required structure on the category and properties of the monad to operational equivalences; second, verifying the operational equivalences.

We shall return to the above example at the end of the next section.

5 Abstract Kleisli categories

In the previous two sections, the category $\mathbf{C}$ together with its monad was taken as primitive, and the notion of equational lifting monad was defined in order to capture the idea of when the Kleisli category is an interesting category of partial maps. On occasion, however, it is arguably more natural to consider the partial category itself as the primitive category. For example, in the case of a category with partial functions and associated product structure, this was the approach adopted by Robinson and Rosolini with their *p-categories* [16].

In the next two sections we provide such a direct axiomatization of the Kleisli categories of equational lifting monads, which are our partial categories of interest. The axiomatization will be obtained by extending Führmann's *abstract Kleisli categories* [7], which directly axiomatize the structure of Kleisli categories of monads. In this section we review the definitions and results that we shall need from [7], in the special case of a *commutative precartesian abstract Kleisli category*, which is of most relevance to us. In Section 6 we shall extend the axiomatization with an equation corresponding to the additional properties of equational lifting monads, to give us the notion of an *abstract Kleisli p-category*.

Definition 5.1 An *abstract Kleisli category* is a category $\mathbf{K}$ together with:

1. A functor $G : \mathbf{K} \to \mathbf{K}$;

2. A transformation[1] $A \xrightarrow{\vartheta_A} GA$ (called *thunk*);

3. A natural transformation $GA \xrightarrow{\varepsilon_A} A$ (called *force*);

such that $\vartheta_G : G \to G^2$ is a natural transformation, and the following diagrams commute:

$$\begin{tikzcd} A \arrow[r, "\vartheta"] \arrow[d, "\vartheta"'] & GA \arrow[d, "G\vartheta"] \\ GA \arrow[r, "\vartheta"] & G^2A \end{tikzcd} \qquad \begin{tikzcd} A \arrow[r, "\vartheta"] \arrow[dr, "id"'] & GA \arrow[d, "\varepsilon"] \\ & A \end{tikzcd} \qquad \begin{tikzcd} GA \arrow[r, "\vartheta"] \arrow[dr, "id"'] & G^2A \arrow[d, "G\varepsilon"] \\ & GA \end{tikzcd}$$

Given any category $\mathbf{C}$ with a monad L, the Kleisli category $\mathbf{C}_L$ forms an abstract Kleisli category. The endofunctor $G_L : \mathbf{C}_L \to \mathbf{C}_L$ is obtained as the composite $\mathbf{C}_L \xrightarrow{K} \mathbf{C} \xrightarrow{J} \mathbf{C}_L$ around the adjunction determined by the monad. Thus on objects we have $G_LA = LA$. The thunk morphism $A \xrightarrow{\vartheta_L} G_LA$ in $\mathbf{C}_L$ is given by $A \xrightarrow{\eta\circ\eta} L^2A$ in $\mathbf{C}$. The force map $G_LA \xrightarrow{\varepsilon_L} A$ in $\mathbf{C}_L$ is just the counit of the adjunction, which is explicitly given by the identity $LA \xrightarrow{id} LA$ in $\mathbf{C}$.

We next show that, conversely, an abstract Kleisli category $\mathbf{K}$ determines a category with monad such that $\mathbf{K}$ is isomorphic to the Kleisli category of the monad.

Definition 5.2 In an abstract Kleisli category $\mathbf{K}$, a morphism $A \xrightarrow{f} B$ is said to be *thunkable* if the diagram below commutes:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "\vartheta"'] & B \arrow[d, "\vartheta"] \\ GA \arrow[r, "Gf"] & GB \end{tikzcd}$$

The collection of thunkable maps forms a subcategory of $\mathbf{K}$. We write $\mathbf{K}_t$ for this subcategory and $J : \mathbf{K}_t \to \mathbf{K}$ for the inclusion functor. This functor has a right adjoint given by the natural isomorphism: $[-] : \mathbf{K}(A, B) \cong \mathbf{K}_t(A, GB)$ defined by:

$$[f] \quad = \quad Gf \circ \vartheta$$

As for any adjunction, the composite determines a monad on $\mathbf{K}_t$. It is shown in [7] that this monad satisfies the equaliser property and, moreover, its Kleisli category is isomorphic, as an abstract Kleisli category, to $\mathbf{K}$. To explain this, we must define functors between abstract Kleisli categories.

Given abstract Kleisli categories $(\mathbf{K}, G, \vartheta, \varepsilon)$ and $(\mathbf{K}', G', \vartheta', \varepsilon')$, a functor $F_K : \mathbf{K} \to \mathbf{K}'$ is said to *preserve abstract Kleisli structure* if it maps thunkable morphisms to thunkable morphisms and, in addition, the induced natural transformation $[F_K\varepsilon] : F_KG \Rightarrow G'F_K$ is

[1] By transformation we mean an arbitrary family of arrows indexed by objects.

an isomorphism. (The general discussion on structure-preserving functors from Section 2 applies to this definition.) Functors preserving abstract Kleisli structure are in one-to-one correspondence with monad-preserving functors from $\mathbf{K}_t$ to $\mathbf{K}'_t$.

The following result, from [7], summarises the discussion so far.

Proposition 5.3 *Given an abstract Kleisli category* $\mathbf{K}$*, the inclusion functor* $J : \mathbf{K}_t \to \mathbf{K}$ *has a right adjoint, inducing a monad* L *on* $\mathbf{K}_t$ *such that* $\mathbf{K}$ *is strictly isomorphic, as an abstract Kleisli category, to* $(\mathbf{K}_t)_L$*. Moreover,* L *satisfies the equaliser property.*

The above constructions map any category and monad $(\mathbf{C}, L, \eta, \mu)$ to an induced abstract Kleisli category $(\mathbf{C}_L, G_L, \vartheta_L, \varepsilon_L)$, and map any abstract Kleisli category $(\mathbf{K}, G, \vartheta, \varepsilon)$ to an induced category $\mathbf{K}_t$ with a monad satisfying the equaliser property.

Suppose we start with a category $\mathbf{C}$ with monad L. We write $(\mathbf{C}_L)_t$ for the category obtained by extracting the thunkable maps from the Kleisli category construed as an abstract Kleisli category. There is a monad-preserving functor $U : \mathbf{C} \to (\mathbf{C}_L)_t$, mapping any morphism $A \xrightarrow{f} B$ in $\mathbf{C}$ to the thunkable morphism $A \xrightarrow{\eta \circ f} B$ in $\mathbf{C}_L$. Moreover, U is an isomorphism of categories if and only if L satisfies the equaliser property. In fact U is a component of the unit of an adjunction exhibiting the (large) category of (small) abstract Kleisli categories (and structure-preserving functors) as a full reflective subcategory of the (large) category of (small) categories with monads and monad-preserving functors [7].

The functor U above maps any monad to one satisfying the equaliser property. To apply this construction to equational lifting monads, we also need to ensure that strength, commutativity and Equation (3) are all preserved.

In [7], the notion of *precartesian abstract Kleisli category* is introduced to extend the reflection theorem to strong monads. In this paper, we are interested only in commutative strong monads. This allows some of the complications of general strong monads to be avoided (in particular, we can work with a monoidal structure rather than a premonoidal structure).

Definition 5.4 A *commutative precartesian abstract Kleisli category* is given by an abstract Kleisli category $\mathbf{K}$ together with a symmetric monoidal structure such that $\mathbf{K}_t$ has distinguished finite products whose induced symmetric monoidal structure is strictly preserved by the inclusion functor $J : \mathbf{K}_t \to \mathbf{K}$.

A functor $F_K : \mathbf{K} \to \mathbf{K}'$, between commutative precartesian abstract Kleisli categories, is *structure preserving* if it preserves both the abstract Kleisli structure and the symmetric monoidal structure. Such functors automatically preserve the finite product structure on the subcategories of thunkable maps.

Let $\pi_i : A_1 \otimes A_2 \longrightarrow A_i$ be the projections given by the finite products on $\mathbf{K}_t$, let $\Delta_A : A \longrightarrow A \otimes A$ be the diagonal, and let $!_A : A \longrightarrow I$ be the unique thunkable arrow. For arbitrary maps $f_1 : A \longrightarrow A_1$ and $f_2 : A \longrightarrow A_2$ of K, define $\langle f_1, f_2 \rangle$ to be

$$A \xrightarrow{\Delta} A \otimes A \xrightarrow{f_1 \otimes f_2} A_1 \otimes A_2$$

So, in the special case where f_1 and f_2 are thunkable, $\langle f_1, f_2 \rangle$ is the unique thunkable morphism $A \xrightarrow{h} A_1 \otimes A_2$ such that $\pi_i \circ h = f_i$. It is important to note, however, that the equalities $\pi_1 \circ \langle f, g \rangle = f$ and $\pi_2 \circ \langle f, g \rangle = g$ do *not* hold for arbitrary f and g in $\mathbf{K}$. Nonetheless, each projection is natural in the non-discarded argument:

Lemma 5.5 *In any commutative precartesian abstract Kleisli category, for arbitrary morphisms $A \xrightarrow{f} A'$ and $B \xrightarrow{g} B'$, we have*

$$\begin{array}{rcl} \pi_1 \circ (f \otimes g) & = & f \circ \pi_1 \circ (id \otimes g) \\ \pi_2 \circ (f \otimes g) & = & g \circ \pi_2 \circ (f \otimes id) \end{array}$$

Given a category $\mathbf{C}$ with finite products and a commutative strong monad L, the induced abstract Kleisli category $\mathbf{C}_L$ forms a commutative precartesian abstract Kleisli category, with $\otimes$ on $\mathbf{C}_L$ generated by the product and strength in $\mathbf{C}$ (see e.g. [9]). Conversely, given a commutative precartesian abstract Kleisli category $\mathbf{K}$, the inclusion $J : \mathbf{K}_t \to \mathbf{K}$ has a right adjoint determining a commutative strong monad on $\mathbf{K}_t$ such that $\mathbf{K}$ is the induced commutative precartesian Kleisli category (up to isomorphism). Once again, the constructions can be combined to obtain, for any category $\mathbf{C}$ with commutative strong monad, a category $(\mathbf{C}_L)_t$ with commutative strong monad satisfying the equaliser property, and a strong-monad-preserving functor $U : \mathbf{C} \to (\mathbf{C}_L)_t$. Note that $(\mathbf{C}_L)_t$ and U are constructed exactly as in the case without strength, so Proposition 5.3 applies *mutatis mutandis*.

We end this section by returning to Example 4.4, showing one can obtain an arguably simpler presentation by first presenting the Kleisli category of $(\mathbf{PCFv})_\mathbf{t}$ directly as an abstract Kleisli category.

Example 5.6 We define a category, $\mathbf{PCFv}$, whose objects are the same as those of $(\mathbf{PCFv})_\mathbf{t}$, but whose morphisms from τ to τ' are equivalence classes of *arbitrary* closed terms of type $\tau \to \tau'$ modulo operational (contextual) equivalence. Identities and composition are as before. For the abstract Kleisli structure, the required endofunctor is defined in the same way as the monad functor of Example 4.4, and thunk is defined in the same way as the unit of the monad (it is no longer natural because of the inclusion of non-total terms in the category). Force is simply $\lambda e^{unit \to \tau}.\, e(*)$ where $*$ is the unique constant of *unit* type.

In order to verify the commutative precartesian structure, the following observation is crucial. A morphism $\tau \xrightarrow{[t]} \tau'$ is thunkable if and only if the following operational equivalence holds:

$$\lambda x^\tau.\, \lambda y^{unit}.\, t(x) \quad \equiv_o \quad \lambda x^\tau.\, (\lambda z^{\tau'}.\, \lambda y^{unit}.\, z)(t(x))$$

Using the "Context Lemma" for call-by-value PCF, one verifies that the above equivalence holds if and only if t is total. Thus, as the notation suggests, $(\mathbf{PCFv})_\mathbf{t}$ is indeed the subcategory of thunkable maps in $\mathbf{PCFv}$.

6 Abstract Kleisli p-categories

We have now given a direct axiomatization of the Kleisli categories of commutative strong monads. It remains to deal with the extra equation (3) of an equational lifting monad. This equation can be expressed in terms of the structure of the commutative precartesian abstract Kleisli category.

Definition 6.1 An *abstract Kleisli p-category* is given by a commutative precartesian abstract Kleisli category in which the diagram below commutes:

$$\begin{tikzcd} GA \arrow[d, "\varepsilon"'] \arrow[dr, "{\langle id,\, \varepsilon\rangle}"] & \\ A \arrow[r, "{\langle \vartheta,\, id\rangle}"] & GA \otimes A \end{tikzcd} \tag{5}$$

Proposition 6.2 *Let L be a commutative strong monad on $\mathbf{C}$. Then L is an equational lifting monad if and only if $\mathbf{C}_L$ (construed as a commutative precartesian abstract Kleisli category) is an abstract Kleisli p-category.*

Thus, unsurprisingly, **PCFv** of Example 5.6 is an abstract Kleisli p-category.

In the remainder of this section, we shall investigate properties of abstract Kleisli p-categories. In particular, we shall show that abstract Kleisli p-categories all form p-categories in the sense of [16] — hence the terminology.

Following [7], it is useful to identify classes of maps that enjoy some of the properties of thunkable maps with respect to the commutative precartesian structure.

Definition 6.3 Suppose that $\mathbf{K}$ is a commutative precartesian abstract Kleisli category. A morphism $A \xrightarrow{f} B$ of $\mathbf{K}$ is called *copyable* if the left-hand diagram below commutes. It is called *discardable* if the right-hand diagram commutes.

$$\begin{tikzcd} A \arrow[r, "\Delta"] \arrow[d, "f"'] & A \otimes A \arrow[d, "f \otimes f"] \\ B \arrow[r, "\Delta"] & B \otimes B \end{tikzcd} \qquad \begin{tikzcd} A \arrow[d, "f"'] \arrow[dr, "!"] & \\ B \arrow[r, "!"] & I \end{tikzcd} \tag{6}$$

The above notions are used in the following proposition, about abstract Kleisli p-categories, which contains the crucial fact that all morphisms are copyable, and provides two very useful ways of proving the equality of morphisms by using joint monicity.

Proposition 6.4 *For a commutative precartesian abstract Kleisli category $\mathbf{K}$, the following are equivalent:*

1. *$\mathbf{K}$ is an abstract Kleisli p-category.*

2. *Every morphism of $\mathbf{K}$ is copyable, and for each object A, the morphisms $GA \xrightarrow{\varepsilon} A$ and $GA \xrightarrow{!} I$ are jointly monic.*

3. *For all objects A and B, $A \otimes B \xrightarrow{\pi_1} A$ and $A \otimes B \xrightarrow{\pi_2} B$ are jointly monic, and $\pi_1 \circ \langle id_A,\, \varepsilon_A\rangle = \vartheta_A \circ \varepsilon_A$.*

Proof For 1⇒2, first observe that for every morphism $f : A \longrightarrow B$ we have $f = \varepsilon \circ [f]$. Because $[f]$ is in $\mathbf{K}_t$, it is copyable, so if ε is copyable then so is f. To see that ε is copyable, consider

$$\begin{aligned}
\Delta \circ \varepsilon &= (\varepsilon \otimes id) \circ \langle \vartheta,\, id \rangle \circ \varepsilon \\
&= (\varepsilon \otimes id) \circ \langle id,\, \varepsilon \rangle && \text{Diagram (5)} \\
&= (\varepsilon \otimes \varepsilon) \circ \Delta
\end{aligned}$$

To see that ε_A and $!_{LA}$ are jointly monic, let $f, g : A \longrightarrow GB$ be such that $\varepsilon \circ f = \varepsilon \circ g$ and $! \circ f = ! \circ g$. Because $[-]$ is natural, we have $G\varepsilon \circ [g] = [\varepsilon \circ g] = [\varepsilon \circ f] = G\varepsilon \circ [f]$, and analogously, $G! \circ [f] = G! \circ [g]$. Because $[f]$ and $[g]$ are copyable, we have $\langle G\varepsilon,\, G! \rangle \circ [f] = \langle G\varepsilon,\, G! \rangle \circ [g]$. The morphism $\langle G\varepsilon,\, G! \rangle$ is a split mono because

$$\begin{aligned}
[\pi_1 \circ (id \otimes \varepsilon)] \circ \langle G\varepsilon,\, G! \rangle &= [\pi_1 \circ (id \otimes \varepsilon) \circ \langle G\varepsilon,\, G! \rangle] \\
&= [\pi_1 \circ ((G\varepsilon) \otimes !) \circ \langle id,\, \varepsilon \rangle] \\
&= [G\varepsilon \circ \pi_1 \circ \langle id,\, \varepsilon \rangle] \\
&= [G\varepsilon \circ \pi_1 \circ \langle \vartheta,\, id \rangle \circ \varepsilon] && \text{Diagram (5)} \\
&= [G\varepsilon \circ \vartheta \circ \varepsilon] = [\varepsilon] = id
\end{aligned}$$

Therefore we have $[f] = [g]$, which implies $f = g$.

Now for 2⇒3. To see that π_1 and π_2 are jointly monic, suppose that $\pi_1 \circ f_1 = \pi_1 \circ f_2$ and $\pi_2 \circ f_1 = \pi_2 \circ f_2$. Because every morphism is copyable, we have $f_i = \langle \pi_1 \circ f_i,\, \pi_2 \circ f_i \rangle$, and therefore $f_1 = f_2$. To prove $\pi_1 \circ \langle id_A,\, \varepsilon_A \rangle = \vartheta_A \circ \varepsilon_A$ we use that ε and $!$ are jointly monic. We have $\varepsilon \circ \vartheta \circ \varepsilon = \varepsilon = \pi_1 \circ \Delta \circ \varepsilon = \pi_1 \circ \langle \varepsilon,\, \varepsilon \rangle = \varepsilon \circ \pi_1 \circ \langle id,\, \varepsilon \rangle$, the penultimate step being an application of Lemma 5.5. And we have $! \circ \pi_1 \circ \langle id,\, \varepsilon \rangle = ! \circ \pi_2 \circ \langle id,\, \varepsilon \rangle = ! \circ \vartheta \circ \varepsilon$, where the last step follows again from Lemma 5.5.

For 3⇒1, Diagram (5) follows from the joint monicity of π_1 and π_2 because:

$$\begin{aligned}
\pi_1 \circ \langle id,\, \varepsilon \rangle &= \vartheta \circ \varepsilon = \pi_1 \circ \langle \vartheta,\, id \rangle \circ \varepsilon \\
\pi_2 \circ \langle id,\, \varepsilon \rangle &= \varepsilon = \pi_2 \circ \langle \vartheta,\, id \rangle \circ \varepsilon
\end{aligned}$$

□

Next we shall explain why every abstract Kleisli p-category forms a p-category. We refer to the axiomatization of p-categories in [16, p.101].

Let $\mathbf{K}$ be an abstract Kleisli p-category. This provides the functor $\otimes : \mathbf{K} \times \mathbf{K} \to \mathbf{K}$ and transformations $A \xrightarrow{\Delta} A \otimes A$ and $A_1 \otimes A_2 \xrightarrow{\pi_i} A_i$ required by a p-category. The required naturality of Δ is equivalent to the fact that every morphism is copyable, as given by Proposition 6.4. The six equations in the definition of p-category involving the projections and the diagonal hold because these transformations are inherited from the finite products on $\mathbf{K}_t$. The associativity and twist maps formed by Δ and the π_i are natural because the inclusion functor $J : \mathbf{K}_t \to \mathbf{K}$ sends them to the symmetric monoidal isomorphisms of $\mathbf{K}$. Finally, the naturality of projections in the non-discarded argument is Lemma 5.5.

From the above discussion, one sees that a commutative precartesian abstract Kleisli category $\mathbf{K}$ forms a p-category if and only if every morphism in $\mathbf{K}$ is copyable. Such categories are exactly the Kleisli categories of commutative relevant monads in the sense of [9] — see the discussion after Proposition 4.2. However, the Kleisli category of the $(-)^2$ monad on **Set** forms a p-category (via its induced commutative precartesian abstract Kleisli structure) but *not* an abstract Kleisli p-category, as ε and ! are not jointly monic.

To appreciate the need for the joint monicity of ε and !, let us see how far we can get without this property. Suppose then that $\mathbf{K}$ is a commutative precartesian abstract Kleisli category in which every map is copyable (i.e. it is a p-category). As in [16, p. 102], we associate a *domain* map $A \xrightarrow{\overline{f}} A$ to any map $A \xrightarrow{f} B$, by defining:

$$\overline{f} \;=\; A \xrightarrow{\langle id, f\rangle} A \otimes B \xrightarrow{\pi_1} A$$

The importance of domain maps is apparent throughout [16]. In fact, Cockett and Lack [3], have recently based their *restriction categories*, which provide a very general axiomatization of categories of partial maps, entirely on equational properties of domain maps. From such properties, we single out the following, which can also be found in [16, pp. 102–104].

Proposition 6.5 *For any morphism* $A \xrightarrow{f} B$ *in* $\mathbf{K}$*,* $f \circ \overline{f} = f$ *and* $\overline{f} = \overline{\overline{f}}$*.*

It follows immediately that every domain map is idempotent.

In any p-category, domain maps determine a notion of total morphism [16, p. 104].

Definition 6.6 A morphism $A \xrightarrow{f} B$ is said to be *p-total* if $\overline{f} = id_A$.

The p-total maps form a subcategory $\mathbf{K}_{pt}$ of $\mathbf{K}$. Moreover, a straightforward consequence of Proposition 6.5 is that every mono in $\mathbf{K}$ is p-total. Further, p-totality corresponds to a notion we have already met.

Proposition 6.7 *A morphism is p-total if and only if it is discardable.*

Proof If f is p-total, then

$$\begin{aligned} !_B \circ f &= !_B \circ \pi_2 \circ (id_A \otimes f) \circ \Delta && \text{(by Lemma 5.5)} \\ &= !_A \circ \pi_1 \circ (id_A \otimes f) \circ \Delta = !_A \circ \overline{f} = !_A \end{aligned}$$

Conversely, if f is discardable, then

$$\overline{f} = \pi_1 \circ (id_A \otimes !_B) \circ (id_A \otimes f) \circ \Delta = \pi_1 \circ (id_A \otimes !_A) \circ \Delta = id_A$$

□

Theorem 2 will identify a property that characterises those abstract Kleisli p-categories for which the equational lifting monad on the subcategory of thunkable morphisms is dominical. Because we shall obtain a dominion on the subcategory of p-total maps, the thunkable maps and the p-total maps will have to coincide. This property does not hold for an arbitrary commutative precartesian abstract Kleisli-categories in which every morphism is copyable. (for example, it fails in the Kleisli category of the $(-)^2$ monad). Obtaining the coincidence of these classes of maps is the crucial reason for requiring the joint monicity of ε and !.

Proposition 6.8 *In any abstract Kleisli p-category a morphism is discardable if and only if it is thunkable.*

Proof Only the left-to-right implication is in question. This holds because, for every discardable f, the equation $\vartheta \circ f = Gf \circ \vartheta$ follows immediately from the joint monicity of ε and !. □

7 Dominical abstract Kleisli p-categories

The notion of abstract Kleisli p-category captures the Kleisli categories of equational lifting monads. In this section we characterise the Kleisli categories of dominical lifting monads, and show that every abstract Kleisli p-category fully embeds in such a *dominical* abstract Kleisli p-category.

Definition 7.1 We say that a commutative precartesian abstract Kleisli category $\mathbf{K}$ is a *dominical abstract Kleisli p-category* if there exists a category $\mathbf{C}$ with specified finite products, dominion $\mathbf{D}$ and $\mathbf{D}$-partial map classifiers such that $\mathbf{Ptl}_{\mathbf{D}}(\mathbf{C})$ and $\mathbf{K}$ are isomorphic as commutative precartesian abstract Kleisli categories.

Theorem 2 *For a commutative precartesian abstract Kleisli category* $\mathbf{K}$*, the following are equivalent:*

1. $\mathbf{K}$ *is a dominical abstract Kleisli p-category.*
2. *The monad on* $\mathbf{K}_t$ *induced by the abstract Kleisli structure is dominical.*
3. $\mathbf{K}$ *is an abstract Kleisli p-category and every domain map in* $\mathbf{K}$ *splits.*

Proof The 2⇒1 implication is trivial, as $\mathbf{K}$ (qua Kleisli category of the induced monad) is isomorphic to $\mathbf{Ptl}_{\mathbf{D}}(\mathbf{K}_t)$ for the dominion $\mathbf{D}$ inducing the dominical monad by an isomorphism preserving the commutative precartesian abstract Kleisli structure.

For 1⇒3, suppose that $\mathbf{K}$ is dominical. In the category of partial maps of any dominion, the restriction operator $\overline{(\cdot)}$ sends a partial map $A \xhookleftarrow{m} A' \xrightarrow{f} B$ to $A \xhookleftarrow{m} A' \xrightarrow{m} A$. The splitting of this is given by the mono $A' \xhookleftarrow{id} A' \xrightarrow{m} A$ and the epi $A \xhookleftarrow{m} A' \xrightarrow{id} A'$.

For 3⇒2, suppose that $\mathbf{K}$ is an abstract Kleisli p-category and every domain map splits. We first, identify a dominion $\mathbf{D}$ on the category $\mathbf{K}_{pt}$ of p-total maps (see Section 6). The dominion is defined by:

$$\mathbf{D} = \{A' \xrightarrow{m} A \mid \text{some domain map } \overline{f} \text{ splits as } A \longrightarrow A' \xrightarrow{m} A\}$$

Note that each such m is mono and hence in $\mathbf{K}_{pt}$. That this is indeed a dominion follows from [16, p. 104]. The proof uses only the p-category structure on $\mathbf{K}$. For example, to construct the required pullbacks, suppose that $A \xrightarrow{f} B$ is in $\mathbf{K}_{pt}$ and $B' \xrightarrow{m} B$ is in $\mathbf{D}$. Thus $B' \xrightarrow{m} B$ is obtained by splitting some domain map $B \xrightarrow{e} B$ as $B \xrightarrow{r} B' \xrightarrow{m} B$. Let $A \longrightarrow A' \xrightarrow{m'} A$ be obtained by splitting $\overline{f \circ e}$. Then the square below is the required pullback of m along f.

$$\begin{tikzcd}
A' \arrow[r, "r \circ f \circ m'"] \arrow[d, hook, "m'"'] & B' \arrow[d, hook, "m"] \\
A \arrow[r, "f"] & B
\end{tikzcd}$$

We now show that $\mathbf{K} \cong \mathbf{Ptl}_{\mathbf{D}}(\mathbf{K}_{pt})$. Given a partial map $A \xhookleftarrow{m} A' \xrightarrow{f} B$ in $\mathbf{Ptl}_{\mathbf{D}}(\mathbf{K}_{pt})$, let $A \xrightarrow{r} A' \xrightarrow{m} A$ be any splitting witnessing that m is in $\mathbf{D}$. Then the associated map in $\mathbf{K}$ is given by $A \xrightarrow{f \circ r} B$ (this is independent of the choice of splitting). Conversely, given any map $A \xrightarrow{f} B$ in $\mathbf{K}$, let $A \longrightarrow A' \xrightarrow{m} A$ split $\overline{f}$. Then the associated partial map in $\mathbf{Ptl_D}(\mathbf{K}_{pt})$ is given by $A \xleftarrow{m} A' \xrightarrow{f \circ m} B$ (it does indeed hold that $f \circ m$ is in $\mathbf{K}_{pt}$). That these constructions are mutually inverse can again be verified using only the p-category structure on $\mathbf{K}$. The isomorphism $H : \mathbf{K} \cong \mathbf{Ptl_D}(\mathbf{K}_{pt})$ we have constructed is easily shown to commute with the inclusion functors $J' : \mathbf{K}_{pt} \to \mathbf{Ptl_D}(\mathbf{K}_{pt})$ and $J : \mathbf{K}_{pt} \to \mathbf{K}$.

To complete the proof, by Propositions 6.7 and 6.8, the categories $\mathbf{K}_t$ and $\mathbf{K}_{pt}$ coincide. Then $J' = HJ : \mathbf{K}_t \to \mathbf{Ptl_D}(\mathbf{K}_t)$ has right-adjoint KH^{-1} (where $K : \mathbf{K} \to \mathbf{K}_t$ is the right-adjoint to J induced by the abstract Kleisli structure) with unit given by ϑ. By the discussion in Section 3, the monad induced by the adjunction between between $\mathbf{K}_t$ and $\mathbf{Ptl_D}(\mathbf{K}_t)$ is indeed dominical. Moreover, by the definition of the adjunction, this monad is exactly that induced on the functor $KJ : \mathbf{K}_t \to \mathbf{K}_t$ by the abstract Kleisli structure on K. $\square$

In the remainder of this section, we use Theorem 2 to show that every abstract Kleisli p-category has a full and faithful structure-preserving embedding into a dominical abstract Kleisli p-category. Given Theorem 2 and the analogous constructions in, e.g., [16, 3], it is no surprise that we obtain this embedding via idempotent splitting (see Section 2). However, one unexpected element does arise in the verification of the construction. In general, it appears that abstract Kleisli structure does not extend from a category $\mathbf{K}$ to the category $\mathbf{Split}(\mathbf{K})$. However, in the special case that $\mathbf{K}$ is an abstract Kleisli p-category, $\mathbf{Split}(\mathbf{K})$ does inherit all the structure. For example, in the proof below, we make use of the joint monicity of ε and ! both to verify that $\mathbf{Split}(\mathbf{K})$ is an abstract Kleisli category, and also to derive the precartesian structure.

Proposition 7.2 *Suppose that $\mathbf{K}$ is an abstract Kleisli p-category, and S is a collection of idempotents in $\mathbf{K}$ that contains all identities and is closed under the application of G and $\otimes$. Then $\mathbf{Split}_S(\mathbf{K})$ also is an abstract Kleisli p-category, and $I : \mathbf{K} \to \mathbf{Split}_S(\mathbf{K})$ strictly preserves all the commutative precartesian abstract Kleisli category structure.*

Proof First we show that $\mathbf{Split}_S(\mathbf{K})$ forms an abstract Kleisli category. The required structure is defined as follows:

$$
\begin{aligned}
G'a &= Ga && \text{(object part of } G'\text{)} \\
G'f &= Gf && \text{(morphism part of } G'\text{)} \\
\vartheta'_a &= Ga \circ \vartheta \circ a && \\
\varepsilon'_a &= a \circ \varepsilon \circ Ga &&
\end{aligned}
$$

The square in the definition of an abstract Kleisli category can be checked using the joint monicity of ε and !. All other equations are straightforward. For the symmetric monoidal product and unit, define

$$
\begin{aligned}
a \otimes' b &= a \otimes b && \text{(object part of } \otimes'\text{)} \\
f \otimes' g &= f \otimes g && \text{(morphism part of } \otimes'\text{)} \\
I' &= id_I &&
\end{aligned}
$$

The structural isomorphisms are defined by:

$$\begin{aligned}
\alpha'_{a,b,c} &= (a \otimes (b \otimes c)) \circ \alpha \circ ((a \otimes b) \otimes c) \\
\gamma'_{a,b} &= (b \otimes a) \circ \gamma \circ (a \otimes b) \\
\lambda'_a &= a \circ \lambda \circ (a \otimes id_I) \\
\rho'_a &= a \circ \rho \circ (id_I \otimes a)
\end{aligned}$$

The coherence equations are easily verified from those for $\mathbf{K}$.

Next we prove that $\otimes'$ and I' form finite products on $(\mathbf{Split}_S(\mathbf{K}))_t$ with projections $\pi'_i : a_1 \otimes' a_2 \longrightarrow a_i$ given by

$$a_i \circ \pi_i \circ (a_1 \otimes a_2)$$

The required equation $\vartheta' \circ \pi'_i = G\pi'_i \circ \vartheta'$ can be proved with the joint monicity of ε and $!$. (The case for $!$ also needs Lemma 5.5.) Define $!'_a : a \longrightarrow I'$ to be $! \circ a$. To check the finite products, we shall repeatedly use the following lemma:

Lemma 7.3 *For every morphism $f \in \mathbf{Split}_S(\mathbf{K})(a, b)$, if $\vartheta'_b \circ f = G'f \circ \vartheta'_{G'a}$ then $!'_b \circ f = !'_a$.*

Lemma 7.3 holds because the second equation results from postcomposing $!$ on both sides of the first equation.

The uniqueness of $!'_a$ follows directly from Lemma 7.3 with $b = I'$. The equation stating that $!'_a$ is thunkable in $\mathbf{Split}_S(\mathbf{K})$ follows from the joint monicity of ε and $!$. To see that $\otimes'$ forms a cartesian product in $(\mathbf{Split}_S(\mathbf{K}))_t$, let $f_1 : a \longrightarrow b_1$ and $f_2 : a \longrightarrow b_2$ be morphisms of $(\mathbf{Split}_S(\mathbf{K}))_t$. We claim that the unique thunkable $h : a \longrightarrow b_1 \otimes b_2$ such that $\pi'_i \circ h = f_i$ is given by $\langle f_1, f_2 \rangle$ (i.e. the pair formed in K). By Lemma 5.5, we have $\pi'_i \circ h = f_i$ if and only if $\pi_i \circ h = f_i$. Because π_1 and π_2 are jointly monic, h is uniquely determined. Next we check that $\langle f_1, f_2 \rangle \in \mathbf{Split}_S(\mathbf{K})(a, b_1 \otimes b_2)$. Because every morphism of K is copyable, we have $\langle f_1, f_2 \rangle \circ a = \langle f_1 \circ a, f_2 \circ a \rangle = \langle f_1, f_2 \rangle$. Obviously, we have $(b_1 \otimes b_2) \circ \langle f_1, f_2 \rangle = \langle f_1, f_2 \rangle$. To see that $\pi_i \circ \langle f_1, f_2 \rangle = f_i$, observe that we have

$$\begin{aligned}
\pi_1 \circ \langle f_1, f_2 \rangle &= \pi_1 \circ (id \otimes !) \circ \langle f_1, f_2 \rangle \\
&= \pi_1 \circ \langle f_1, ! \circ f_2 \rangle \\
&= \pi_1 \circ \langle f_1, ! \circ a \rangle && \text{Lemma 7.3} \\
&= \pi_1 \circ \langle f_1 \circ a, ! \circ a \rangle \\
&= \pi_1 \circ \langle f_1, ! \rangle \circ a && \text{because every morphism of } K \text{ is copyable} \\
&= f_1 \circ \pi_1 \circ \langle id, ! \rangle \circ a && \text{Lemma 5.5} \\
&= f_1 \circ a = f_1
\end{aligned}$$

Analogously, we get $\pi_2 \circ \langle f_1, f_2 \rangle = f_2$. The required equation $\vartheta'_{b_1 \otimes b_2} \circ \langle f_1, f_2 \rangle = G'\langle f_1, f_2 \rangle \circ \vartheta'_a$ is proved with the joint monicity of ε and $!$: That both sides followed by ε are the same is straightforward. For postcomposition with $!$, consider

$$\begin{aligned}
! \circ \vartheta'_{b_1 \otimes b_2} \circ \langle f_1, f_2 \rangle &= ! \circ \langle f_1, f_2 \rangle = ! \circ \langle ! \circ f_1, ! \circ f_2 \rangle \\
&= ! \circ \langle ! \circ a, ! \circ a \rangle && \text{Lemma 7.3} \\
&= ! \circ \langle a, a \rangle = ! \circ \langle id, id \rangle \circ a \\
&= ! \circ a = ! \circ G'\langle f_1, f_2 \rangle \circ \vartheta'_a
\end{aligned}$$

Although we know now that $a \otimes' b$ is the cartesian product of a and b in $(\mathbf{Split}_S(\mathbf{K}))_t$, we still have to check that the *morphism* part of the induced cartesian-product functor agrees with the functor $\otimes'$ we defined earlier. This is so because

$$\langle f_1 \circ \pi'_1,\, f_2 \circ \pi'_2 \rangle = (f_1 \otimes f_2) \circ \langle \pi'_1,\, \pi'_2 \rangle = f_1 \otimes f_2 = f_1 \otimes' f_2$$

It remains to prove that α', γ', λ', and ρ' coincide with the corresponding maps induced by the finite-product structure on $(\mathbf{Split}_S(\mathbf{K}))_t$. We have

$$\begin{aligned}
&\langle \pi'_1 \circ \pi'_1,\, \langle \pi'_2 \circ \pi'_1,\, \pi'_2 \rangle\rangle \\
&\quad = \langle \pi_1 \circ \pi_1 \circ ((a \otimes b) \otimes c),\, \langle \pi_2 \circ \pi_1 \circ ((a \otimes b) \otimes c),\, \pi_2 \circ ((a \otimes b) \otimes c)\rangle\rangle && \text{Lemma 5.5} \\
&\quad = \langle \pi_1 \circ \pi_1,\, \langle \pi_2 \circ \pi_1,\, \pi_2 \rangle\rangle \circ ((a \otimes b) \otimes c) && \text{copyability in } \mathbf{K} \\
&\quad = \alpha \circ ((a \otimes b) \otimes c) = \alpha \circ ((a \otimes b) \otimes c) \circ ((a \otimes b) \otimes c) \\
&\quad = \alpha' && \text{naturality of } \alpha
\end{aligned}$$

A similar calculation works for γ', and the cases for λ' and ρ' are trivial.

To see that $\mathbf{Split}_S(\mathbf{K})$ is an abstract Kleisli p-category, it suffices to verify Diagram (5), but this follows easily from (5) in $\mathbf{K}$.

It remains to verify that the functor $I : \mathbf{K} \to \mathbf{Split}_S(\mathbf{K})$ strictly preserves all the commutative precartesian abstract Kleisli p-category structure. But this is immediate from the definition of the structure on $\mathbf{Split}_S(\mathbf{K})$. This completes the proof of Proposition 7.2. $\square$

Corollary 4 *Suppose* $\mathbf{K}$ *is an abstract Kleisli p-category. Let* S *be the collection of all idempotents in* S*. Then* $\mathbf{Split}_S(\mathbf{K})$ *is a dominical abstract Kleisli p-category, and* $I : \mathbf{K} \to \mathbf{Split}_S(\mathbf{K})$ *is a full and faithful functor that preserves existing limits and colimits and strictly preserves the commutative precartesian abstract Kleisli category structure.*

Proof By Proposition 7.2, $\mathbf{Split}_S(\mathbf{K})$ is an abstract Kleisli p-category, and the functor $I : \mathbf{K} \to \mathbf{Split}_S(\mathbf{K})$ is a full, faithful and structure-preserving (including limits and colimits). That $\mathbf{Split}_S(\mathbf{K})$ is a dominical follows from Theorem 2, as all idempotents in $\mathbf{Split}_S(\mathbf{K})$ split (not just the domain maps). $\square$

We end the section with a remark about the above proof. In order to expand an abstract Kleisli p-category $\mathbf{K}$ to a dominical abstract Kleisli p-category, it is not really necessary to split *all* idempotents in $\mathbf{K}$. However, in spite of Theorem 2, it does not suffice to split only the domain maps, as these do not satisfy the closure conditions required by Proposition 7.2. What is possible is to define S_0 to be the least collection of maps containing all identities and closed under G, $\otimes$ and the following rule: if $a \in S_0$, and $f \circ a = f$, then $\overline{f} \circ a \in S_0$. (It follows that all maps in S_0 are idempotents.) The dominical abstract Kleisli p-category obtained as $\mathbf{Split}_{S_0}(\mathbf{K})$ has a universal property as the free dominical abstract Kleisli p-category over $\mathbf{K}$. This universal property can be expressed in terms of a 2-adjunction between the 2-category of abstract Kleisli p-categories and its dominical subcategory, along the lines of the 2-adjunctions exhibited in [3].

8 Equational lifting monads revisited

Having taken a thorough look at the abstract Kleisli category account of lifting, we now return to our initial viewpoint, in which the category with monad is taken as primitive. We begin by piecing together the various results we have proved about abstract Kleisli p-categories to obtain a swift proof of Theorem 1.

Proof of Theorem 1. Let L be an equational lifting monad on $\mathbf{C}$. By Proposition 6.2, the Kleisli category $\mathbf{C}_L$ is an abstract Kleisli p-category. Moreover, as in Section 5, there is a strong-monad-preserving functor $U : \mathbf{C} \to (\mathbf{C}_L)_t$ (the thunkable subcategory) that is (trivially) Kleisli full and Kleisli faithful (there is an isomorphism of Kleisli categories).

Let S be the collection of all idempotents in $\mathbf{C}_L$. By Corollary 4, $\mathbf{Split}_S(\mathbf{C}_L)$ is a dominical abstract Kleisli p-category, and there is a full and faithful structure-preserving functor $I : \mathbf{C}_L \to \mathbf{Split}_S(\mathbf{C}_L)$, which also preserves all existing limits and colimits in $\mathbf{C}_L$. The functor I restricts to a strong-monad-preserving functor $I_t : (\mathbf{C}_L)_t \to (\mathbf{Split}_S(\mathbf{C}_L))_t$ which is (Kleisli) full and (Kleisli) faithful. Moreover, by Theorem 2, the induced monad L' on $(\mathbf{Split}_S(\mathbf{C}_L))_t$ is dominical.

Thus $I_t U : \mathbf{C} \to (\mathbf{Split}_S(\mathbf{C}_L))_t$ is the required Kleisli full and Kleisli faithful dominical representation of L. $\Box$

It is worth commenting on the extent to which our extensive detour through abstract Kleisli p-categories facilitated the proof of Theorem 1.

One can view the proof as dividing into two stages. The first, the construction of $(\mathbf{C}_L)_t$, provides a representation of L into an induced monad (that on $(\mathbf{C}_L)_t$) satisfying the equaliser property. This representation is Kleisli full and Kleisli faithful because $(\mathbf{C}_L)_t$ is defined as the thunkable maps within the original Kleisli category. The construction of $(\mathbf{C}_L)_t$ already makes use of abstract Kleisli categories, applying the reflection theorem of [7] to equational lifting monads. However, it would be quite feasible to reformulate this part of the proof entirely in terms of the equational lifting monad structure, by verifying directly that $(\mathbf{C}_L)_t$ carries a suitable equational lifting monad (although this is not entirely trivial as one must verify, e.g., that $(\mathbf{C}_L)_t$ has products)

The second stage of the proof constructs a full and faithful dominical representation of the monad on $(\mathbf{C}_L)_t$. By Theorem 2, it is is necessary to split domain maps in $\mathbf{C}_L$ while retaining the property of being the Kleisli category of an equational lifting monad. In order to verify such a property, it is indispensable to have a direct axiomatization of such Kleisli categories so that the structure that must be preserved under idempotent splitting is identified. The notion of abstract Kleisli p-category provides exactly this.

We now have all the ingredients to characterise when an equational lifting monad is itself a dominical lifting monad.

Theorem 3 *For a monad L on a category $\mathbf{C}$ with finite products, the following are equivalent.*

1. *L is a dominical lifting monad.*
2. *L is an equational lifting monad satisfying the equaliser property and all domain maps split in the Kleisli category $\mathbf{C}_L$.*
3. *L is an equational lifting monad satisfying the equaliser property, all pullbacks of the unit η exist in $\mathbf{C}$, and L preserves existing pullbacks.*

Proof From Section 3, we have that any dominical lifting monad on a category with finite products has a unique strength that makes it an equational lifting monad satisfying the properties in 3. Thus 1 implies 3. Also, it is immediate from Theorem 2 that 1 is equivalent to 2. We complete the proof by showing that 3 implies 2.

Let $A \xrightarrow{f} LA$ in $\mathbf{C}$ be a domain map in $\mathbf{C}_L(A, A)$. We have to show that it splits. By assumption, the pullback below exists in $\mathbf{C}$.

$$\begin{tikzcd}
A' \arrow[r, "h"] \arrow[d, "m"'] \arrow[dr, phantom, "\lrcorner", very near start] & A \arrow[d, "\eta"] \\
A \arrow[r, "f"] & LA
\end{tikzcd}$$

We first show that $\eta \circ h = \eta \circ m$ hence $h = m$.

$$\begin{aligned}
\eta \circ h &= f \circ m \\
&= L\pi_1 \circ t \circ \langle id_A,\, f\rangle \circ m && \text{because } f \text{ is a domain map} \\
&= L\pi_1 \circ t \circ \langle m,\, f \circ m\rangle \\
&= L\pi_1 \circ t \circ \langle m,\, \eta \circ h\rangle \\
&= L\pi_1 \circ \eta \circ \langle m,\, h\rangle \\
&= \eta \circ m
\end{aligned}$$

As L preserves the above pullback, we have a pullback:

$$\begin{tikzcd}
LA' \arrow[r, "Lm"] \arrow[d, "Lm"'] \arrow[dr, phantom, "\lrcorner", very near start] & LA \arrow[d, "L\eta"] \\
LA \arrow[r, "Lf"] & L^2A
\end{tikzcd}$$

But f is a domain map in $\mathbf{C}_L$, hence idempotent, so $\mu \circ Lf \circ f = f = \mu \circ L\eta \circ f$. Also, $L! \circ Lf \circ f = L! \circ L\eta \circ f$ (obviously). So, by the joint monicity of μ and $L!$, we have $Lf \circ f = L\eta \circ f$. Therefore, the pullback above gives a unique $A \xrightarrow{r} LA'$ such that $Lm \circ r = f$.

We now see that $\eta \circ m$ and r provide the required splitting in $\mathbf{C}_L$. That $Lm \circ r = f$ already provides one equality. For the other, we must show that $r \circ m = \eta$. But:

$$Lm \circ r \circ m \;=\; f \circ m \;=\; \eta \circ h \;=\; \eta \circ m \;=\; Lm \circ \eta$$

As Lm is mono, the required equality follows. $\square$

It is instructive to see that the requirement that L preserve pullbacks in statement 3 of the theorem cannot be dropped. Indeed there is a very simple counterexample in **Set**. Define L to be the evident strong monad with underlying functor:

$$LX \quad = \quad \begin{cases} \emptyset & \text{if } X \text{ is empty} \\ 1 + X & \text{otherwise} \end{cases}$$

Then L is an equational lifting monad satisfying the equaliser requirement. Obviously all pullbacks of the unit η exist. However, by exhibiting $\emptyset$ as a pullback of nonempty objects, it is easy to see that L does not preserve pullbacks.

To verify explicitly that L is not dominical, observe that pullbacks of η form the dominion **Inj** of all injective functions. The **Inj**-partial maps are exactly the set-theoretic partial functions. The undefined partial function from 1 to $\emptyset$ is thus an **Inj**-partial map, but it is not represented by a point of $L\emptyset$.

We end this section with a curious fact that arose in discussion with Paul Taylor.

Proposition 8.1 *For an equational lifting monad* $(\mathbf{C}, L)$*, the following are equivalent.*

1. *L satisfies the equaliser property.*
2. *The functor $J : \mathbf{C} \to \mathbf{C}_L$ is comonadic.*

Proof We work with the abstract Kleisli p-category structure on $\mathbf{C}_L$. There is an evident full and faithful functor $(\mathbf{C}_L)_t \to G\text{-Coalg}(\mathbf{C}_L)$, which maps an object A to the comonad coalgebra $A \xrightarrow{\vartheta} GA$ in $\mathbf{C}_L$. The comparison functor from $\mathbf{C}$ to $G\text{-Coalg}(\mathbf{C}_L)$ is given by the composite $\mathbf{C} \to (\mathbf{C}_L)_t \to G\text{-Coalg}(\mathbf{C}_L)$. We must show that this is an isomorphism if and only if the equaliser property holds. But, as remarked in Section 5, the functor $\mathbf{C} \to (\mathbf{C}_L)_t$ is an isomorphism if and only if the equaliser property holds. Thus it suffices to show that the functor $(\mathbf{C}_L)_t \to G\text{-Coalg}(\mathbf{C}_L)$ is always an isomorphism. Accordingly, let $A \xrightarrow{a} GA$ be any comonad coalgebra for G. We show that $a = \vartheta_A$. As a is a comonad coalgebra, $\varepsilon \circ a = id_A$, thus a is mono and hence, by Propositions 6.7 and 6.8, thunkable. But then $a = [\varepsilon \circ a]$ so $a = [id_A] = \vartheta_A$ as required. $\square$

9 Discussion

The work presented in this paper grew out of earlier work by the first author with Pino Rosolini. Their paper, [2], had a similar motivation to ours, but its development was different in detail. In *loc. cit.*, a general notion of lifting was defined for categories with terminal object. In the special case that the category has finite products, as we assume in this paper, the notion of lifting in [2] corresponds to that of an equational lifting monad satisfying the equaliser property whose underlying functor preserves existing pullbacks. So, although, by not assuming finite products, the scope of [2] is more general than ours, the notion of lifting adopted in [2] is strictly stronger than that of an equational lifting monad. In [2], given a category $\mathbf{C}$ with finite products and a lifting (in their stronger sense) L, a category $\mathbf{C}_t$ is constructed, which is the free category with dominion such that $\mathbf{C}_L$ fully embeds in $\mathbf{Ptl}(C_t)$. Also, the functor $\mathbf{C} \to \mathbf{C}_t$ is shown to preserve all existing limits in $\mathbf{C}$ that are preserved by L. This embedding result is similar in spirit to our Theorem 1 and Corollary 1. However, a major difference with the present paper is that the category $\mathbf{C}_t$ of [2] does not itself have a lifting monad acting on it. So, in the present paper, we obtain a stronger representation theorem under weaker conditions. We also separate out precisely the equational properties of lifting monads from the non-equational properties, and establish new nontrivial consequences of the equational properties (e.g. Corollary 2).

At first sight, our equation (3) may seem rather curious. There are many intriguing connections. For example, it seems likely that the *cuboidal category*, introduced in [6], is the free category with equational lifting monad over the empty set of generating objects. Also, although we don't have a precise connection, it is worth remarking that equation (3) is reminiscent of the *Euclidean principle*, recently introduced by Taylor as part of a characterisation of dominances [19]. It is also illuminating to consider the significance of (3) within Moggi's computational lambda-calculus [13]. It appears that equation (3) corresponds to the inter-substitutivity of e with x in the body M of *let* $x = e$ *in* M. It would be interesting to see if the completeness of such a formulation might lead to a simplified meta-theory for the partial λ-calculus [12]. Such an investigation would require extending the results in this paper to cover partial function spaces, which is itself a (probably straightforward) programme of independent interest.

One of the applications we have of the work in this paper is to establish properties of recursion in axiomatic domain theory. A general axiomatic analysis of recursion has been carried out in [18], establishing equational completeness assuming the existence of sufficiently many final coalgebras. In the presence of an equational lifting monad, Kleisli exponentials and a (parameterized) natural numbers object $\mathbf{N}$, all the necessary final coalgebras can be constructed by splitting idempotents on $\mathbf{N}$-fold powers of lifted objects [18].

The work in this paper constitutes an equational analysis of partial map classifiers. There are many other similar projects possible that might be worth investigating. One question is whether there is an equational characterisation of those (subpowerset) monads whose Kleisli categories can be viewed as categories of relations. Such monads would include all lifting monads, but also powerobject monads and various other related notions of free lattice. A natural setting would be to obtain representation theorems with respect to regular categories.

Another possible direction for generalising the work in this paper is dropping the commutativity requirement on equational lifting monads. Many of the calculations in this paper do not make use of commutativity, and thus apply to the more general class of strong monads satisfying Equation (3). One reason for being interested in such a generalisation is that this class also includes "exception" monads of the form $(-) + E$ [14].

Finally, we mention related work by Robin Cockett and Stephen Lack. Building on their still unpublished *restriction categories* [3], they have recently extended their axiomatization to lifting monads [4]. Their work nicely complements ours. We assume finite products in the underlying category, and emphasise equational properties, representation theorems and transference results. They characterise lifting monads without assuming finite products, and provide 2-categorical reflection theorems different relating classes of lifting monad.

Acknowledgements

We thank Robin Cockett, Marcelo Fiore, Gordon Plotkin, Pino Rosolini and Paul Taylor for helpful discussions, and the anonymous referees for their suggestions.

References

[1] M. Barr and C. Wells. *Toposes, Triples and Theories.* Springer-Verlag, 1985.

[2] A. Bucalo and G. Rosolini. Lifting. In *Category Theory and Computer Science, Proceedings of CTCS '97.* Springer LNCS 1290, 1997.

[3] J.R.B. Cockett and S. Lack. Restriction categories I: categories of partial maps. *Theoretical Computer Science*, 271 (to appear), 2001.

[4] J.R.B. Cockett and S. Lack. Restriction categories II: partial map classification. *This volume*, 2002.

[5] M.P. Fiore. *Axiomatic Domain Theory in Categories of Partial Maps.* Distinguished Dissertation Series, Cambridge University Press, 1996.

[6] M.P. Fiore, G.D. Plotkin, and A.J. Power. Complete cuboidal sets in axiomatic domain theory. In *Proceedings of 12th Annual Symposium on Logic in Computer Science*, 1997.

[7] C. Führmann. Direct models for the computational lambda-calculus. Proceedings MFPS XV*, Electronic Notes in Computer Science*, 20, 1999.

[8] J.M.E. Hyland. First steps in synthetic domain theory. In *Category Theory, Proceedings, Como 1990.* Springer LNM 1488, 1990.

[9] B.P.F. Jacobs. Semantics of weakening and contraction. *Annals of Pure and Applied Logic*, 69:73–106, 1994.

[10] A. Kock. Strong functors and monoidal monads. *Archiv der Mathematik*, 23:113–120, 1972.

[11] S. Mac Lane. *Categories for the Working Mathematician.* Graduate Texts in Mathematics. Springer Verlag. Second Edition, 1998.

[12] E. Moggi. *The Partial Lambda-Calculus.* Ph.D. thesis, Department of Computer Science, University of Edinburgh, 1988. Available as LFCS report no. ECS-LFCS-88-63.

[13] E. Moggi. Computational lambda-calculus and monads. In *Proceedings of 4th Annual Symposium on Logic in Computer Science*, pages 14–23, 1989.

[14] E. Moggi. Notions of computation and monads. *Information and Computation*, 93(1), 1991.

[15] P.S. Mulry. Generalized Banach-Mazur functionals in the topos of recursive sets. *Journal of Pure and Applied Algebra*, 26:71–83, 1982.

[16] E.P. Robinson and G. Rosolini. Categories of partial maps. *Information and Computation*, 79:95–130, 1988.

[17] G. Rosolini. *Continuity and Effectivity in Topoi.* PhD thesis, University of Oxford, 1986.

[18] A.K. Simpson and G.D. Plotkin. Complete axioms for categorical fixed-point operators. In *Proceedings of 15th Annual Symposium on Logic in Computer Science*, pages 30–41, 2000.

[19] P. Taylor. *Geometric and Higher Order Logic using Abstract Stone Duality.* Submitted, 2000.

[20] G. Winskel. *The Formal Semantics of Programming Languages.* MIT Press, 1993.