

Electronic Notes in Theoretical Computer Science 230 (2009) 3–21

[www.elsevier.com/locate/entcs](http://www.elsevier.com/locate/entcs)

Context for Models of Concurrency

Peter Bubenik [1](#_bookmark0) *,*[2](#_bookmark0)

*Institut de G´eom´etrie, Alg`ebre et Topologie Ecole Polytechnique F´ed´erale de Lausanne Lausanne, Switzerland*

Abstract

Many categories have been used to model concurrency. Using any of these, the challenge is to reduce a given model to a smaller representation which nevertheless preserves the relevant computer-scientific information. That is, one wants to replace a given model with a simpler model with the same directed homotopy-type. Unfortunately, the obvious definition of directed homotopy equivalence is too coarse. This paper introduces the notion of *context* to refine this definition.

*Keywords:* models for concurrency, po-space (pospace), directed homotopy (dihomotopy), context, fundamental category, model category, pushout, local po-space (local pospace)

# Introduction

Various topological models are being used for studying concurrency. Among them are precubical complexes [[7](#_bookmark34)], d-spaces [[9](#_bookmark39),[10](#_bookmark41)], local po-spaces [[4](#_bookmark35),[8](#_bookmark40)], and FLOW [[6](#_bookmark36)]. For a given concurrent system, each of these categories provides a model which captures the relevant computer-scientific properties of the system.

These categories are large in two senses. They are large ‘locally’ in that a given model contains many paths which correspond to executions which are essentially equivalent. They are also large ‘globally’ in that a given concurrent system has a large number of models within the category. The size of these categories is a strength in terms of their descriptive power. However, to aid in calculations one would like to reduce these models to a smaller, possibly even discrete, representation.

A major goal of current research in this area is to introduce equivalences to obtain such smaller representations, which nevertheless still retain the relevant computer-scientific properties.

1 This work was funded by the Swiss National Science Foundation grant 200020-105383.

2 Email: [peter.bubenik@epfl.ch](mailto:peter.bubenik@epfl.ch)

1571-0661/© 2009 Elsevier B.V. Open access under [CC BY-NC-ND license.](http://creativecommons.org/licenses/by-nc-nd/3.0/)

doi:10.1016/j.entcs.2009.02.014

On the local front progress has been made in reducing the path space of a given model using directed homotopies of paths and the fundamental category [[9](#_bookmark39)]. One global approach is to pass to the component category [[5](#_bookmark37),[13](#_bookmark44)]. In this paper we introduce another global approach, which is perhaps more geometric and which is compatible with the model categorical approach of [[1](#_bookmark32)].

In the classical (undirected) topological case, the solution to this ‘global’ problem is well-understood. The equivalent spaces are the (weak) homotopy equivalent ones. So for example, all of the contractible spaces (those homotopy equivalent to a point) are equivalent.

In the directed case there is a similar notion of directed homotopy equivalence (abbreviated to *dihomotopy equivalence*, which will be defined in the next section). However this notion is too coarse.

−→ −→ −→ −→ −→

Fig. 1. *I* , *I* × *I* , *X* , and *O*

Example 1.1 Let

−→

*I* be the unit interval [0*,* 1] with a direction given by the usual

−→ −→

ordering of the real numbers. Let *I* × *I* be [0*,* 1] × [0*,* 1] with the ordering (*x, y*) ≤

' ' '

' −→

(*x ,y* ) if and only if *x* ≤ *x*

−→

and *y* ≤ *y* . Let *X* be the space in Figure [1](#_bookmark1) given by

attaching two copies of *I* at their midpoints. Then as will be shown explicitly in

−→ −→ −→ −→

Example [2.7](#_bookmark11), *I* ,

−→

*I* × *I* and *X* are all dihomotopy equivalent to a point. However

−→

*I* models an execution with one initial state and one final state while *X* models

an execution with two initial states and two final states.

−→ −→ −→

Clearly a stronger notion of equivalence is needed. Since

*I* and *I* × *I* both have

one initial state and one final state and all execution paths seem to be essentially equivalent it seems natural that we should look for a definition of equivalence under which these are equivalent. However even this ‘equivalence’ has a pitfall.

For a notion of equivalence to be practical it should continue to hold under certain ‘pastings’. Our philosophy is the following. If we make the same addition to equivalent models we should still have equivalent models.

~ ?

*y*'

*x*'

*x*

*y*

Fig. 2. A hypothetical equivalence

−→ −→ −→

Example 1.2 Assume we have an equivalence

*I* × *I* → *I* as in Figure [2](#_bookmark2). Consider

the following pasting on

−→ −→

*I* × *I* . Let

−→

*O* be the space in Figure [1](#_bookmark1) constructed by

*b*2

*a*1 *a*1

*b*2

Fig. 3. A map *B* → *C* which should not be an equivalence

−→

[3](#_bookmark5) −→

attaching two copies of *I* at their initial points and at the final points.

Let *O* 1

−→ −→ −→

and *O* 2 be two copies of *O* . For *i* = 1*,* 2 let *ai, bi* ∈ *Oi* denote the initial and final

points of

−→

*Oi*. Now choose two points *x, y* ∈

−→ −→

*I* × *I* such that neither *x* ≤ *y* nor

' ' −→

*y* ≤ *x*. Let *x ,y* ∈ *I* be the images of *x* and *y* under the assumed equivalence

(Figure [2](#_bookmark2)). Then either *x*'

≤ *y*'

or *y*'

≤ *x*', since

−→

*I* is totally ordered.

If *x*' ≤ *y*' then identify *b*1 and *x* and identify *a*2 and *y*. Call this space *B*

and denote *C* the space obtained by collapsing

−→ −→

*I* × *I* ⊂ *B* to

−→

*I* using the given

equivalence (Figure [3](#_bookmark4)). Then there is an execution path from *a*1 to *b*2 in *C* but not

in *B*. So the concurrent systems modeled by *B* and *C* are not equivalent. A similar

construction is possible if *y*'

should not be equivalent.

≤ *x*'. Thus from this point of view

−→ −→ −→

*I* × *I* and *I*

This gives a good indication of the current state of affairs for determining a

global notion equivalence. We don’t even know whether or not should be equivalent.

−→ −→ −→

*I* × *I* and *I*

−→ −→ −→

In this paper we introduce the idea of *context*. Whether or not

*I* and *I* × *I* are

equivalent depends on the context. If we permit pastings as in Example [1.2](#_bookmark3), then

they are not equivalent. However if we only permit pastings to the initial and final

points of

−→

*I* and

−→ −→

*I* × *I* then they are equivalent. ¿From the computer-scientific

point of view this can be interpreted as follows. We cannot expect equivalent con- current systems to still be equivalent after arbitrary (but equal) changes. However, if equal additions are made in a suitably modular way, then the resulting systems should still be equivalent.

It should be noted that in the examples in this paper the context is chosen ‘by hand’. The problem of choosing the context is related to the components of the fundamental category [[5](#_bookmark37)] and to the universal dicovering space [[3](#_bookmark38)]. A procedure for choosing the context is a subject for future research.

Acknowledgments. The author would like to thank the referee for simplifying the proof of Proposition [3.5](#_bookmark14).

# Context for directed homotopy equivalences

In this section we make precise the intuitive ideas presented in the introduction.

3 This is M.Grandis’ *ordered circle* ↑*O*1 [[9](#_bookmark39), Section 1.2].

Definition 2.1 • A *partial order* on a topological space *U* is a reflexive, transitive, anti-symmetric relation ≤. If *U* has a partial order ≤ which is a closed subset of *U* × *U* under the product topology, then call *U* a *po-space* or *pospace*.

* A *dimap f* : (*U*1*,* ≤1) → (*U*2*,* ≤2) is a continuous map *f* : *U*1 → *U*2 such that

*x* ≤1 *y* implies that *f* (*x*) ≤2 *f* (*y*).

* A product of pospaces (*U*1*,* ≤1) and (*U*2*,* ≤2) is a pospace whose underlying topo- logical space is *U*1 × *U*2 and whose order relation is given by (*x, y*) ≤ (*x*'*, y*') if and only if *x* ≤1 *x*' and *y* ≤2 *y*'.
* A subspace *A* of a pospace *U* inherits a pospace structure under the definition

*x* ≤*A y* if and only if *x* ≤*U y*. This is called a *sub-pospace*.

Definition 2.2 Let Pospace be the the category whose objects are pospaces and whose morphisms are dimaps.

For the sake of simplicity we will work with pospaces but one should be able to easily extend or adapt the constructions presented here for other models of concur- rency.

Let

−→

*I* = ([0*,* 1]*,* ≤) where ≤ is the usual ordering of R. This is a pospace. A

−→

*dipath* in a pospace *B* is a dimap *I* → *B*.

−→

Definition 2.3 • Given dimaps *f, g* : *B* → *C* ∈ Pospace, *φ* : *B* × *I* → *C* ∈

Pospace is a *dihomotopy* [4](#_bookmark10) from *f* to *g* if *φ*|*B*×{0} = *f* and *φ*|*B*×{1} = *g*. In this case write *φ* : *f* → *g*.

* Write *f* *g* if there is a chain of dihomotopies *f* → *f*1 ← *f*2 → *...* ← *fn* → *g*. This is an equivalence relation.
* A dimap *f* : *B* → *C* is a *dihomotopy equivalence* if there is a dimap *g* : *C* → *B*

such that *g* ◦ *f* Id*B* and *f* ◦ *g* Id*C*. In this case write *B* *C*.

Our explicit dihomotopies will often be of the following form.

Definition 2.4 Assume that *C* is a pospace whose underlying topological spaces is a subspace of R*n* for some *n*. Assume *f, g* : *B* → *C* are two dimaps. Let the

*linear interpolation* between *f* and *g* be the map *H* : *B* ×

*H*(*b, t*)= (1 − *t*)*f* (*b*)+ *tg*(*b*).

−→

*I* → R*n*

given by

Remark 2.5 *Note that there is no guarantee that the image of such a map is in*

*C. However one can check that it is for the cases we will consider.*

Lemma 2.6 *Assume that C is a pospace whose underlying topological spaces is a* *subspace of* R*n for some n. If f, g* : *B* → *C are dimaps such that for all b* ∈ *B, f* (*b*) ≤ *g*(*b*) *then if the image of the linear interpolation H between f and g is in C then H is a dihomotopy from f to g.*

Proof. That *H* restricts to *f* and *g* follows from the definition of linear interpola- tion. It remains to check that *H* is a dimap.

4 This is the notion of dihomotopy in [[9](#_bookmark39)] which is stronger than the notion of dihomotopy in [[4](#_bookmark35)] (which uses

−→

*I* = [0*,* 1] with the trivial ordering *x* ≤*I y* ⇔ *x* = *y*, instead of *I* ).

Let *b* ≤*B b*' and *t* ≤−→ *t*'. Then

*I*

*H*(*b, t*)= *f* (*b*)+ *t*(*g*(*b*) − *f* (*b*))

≤ *f* (*b*)+ *t*'(*g*(*b*) − *f* (*b*))

= (1 − *t*')*f* (*b*)+ *t*'*g*(*b*)

≤ (1 − *t*')*f* (*b*')+ *t*'*g*(*b*')

= *H*(*b*'*, t*')

−→ −→

−→ −→ −→ −→

Example 2.7 If *I* 1 and *I* 2 are two copies of *I* , then let *X* = ( *I* 1 M *I* 2)*/* ∼

1 1 −→

where ( 2 )1 ∼ ( 2 )2 (see Figure [1](#_bookmark1)). We will show that under Definition [2.3](#_bookmark7), *I* ,

−→ −→ −→ −→ −→

*I* × *I* , and *X* are dihomotopy equivalent to a point. Let *f* :

−→

*I* → ∗, *g* : ∗→ *I* be

the constant map and the inclusion of the point to 1 ∈ *I* . Then *f* ◦ *g* = Id∗ and it

−→ −→ −→

remains to show that Id−→ *g* ◦ *f* . Let *H* :

*I*

between Id−→ and *g* ◦ *f* . That is,

*I*

*H*(*x, t*)= (1 − *t*)*x* + *t*

= *x* + *t*(1 − *x*)

*I* × *I* → *I* be the linear interpolation

Then *H* is a dimap and is the desired homotopy Id−→ → *g* ◦ *f* .

*I*

In exactly the same way one can show that the constant map *f* :

−→ −→

*I* × *I* → ∗ is

−→ −→

a dihomotopy equivalence with *g* : ∗→ *I* × *I* given by *g*(∗)= (1*,* 1).

−→

To show that the constant map *f* : *X* → ∗ is a dihomotopy equivalence with

*g*(∗) = ( 1 )1 = ( 1 )2 is slightly more complicated. Again *f* ◦ *g* = Id∗. To show

2 2

*H*1 *H*2

Id−→ *g* ◦ *f* we will construct a chain of dihomotopies Id−→ −−→ *h* ←−− *g* ◦ *f* . Let *h* be

*X* −→*X*

−→ −→

the map that collapses the lower two line segments of *X* . That is, let *h* : *X* → *X*

be given by

*x* '→

⎧⎨ 1

1

if *x <*

2

⎩ *x* otherwise

2

Let *H*1 be the linear interpolation between Id−→ and *h* and let *H*2 be the linear

*X*

interpolation between *g* ◦ *f* and *h*. Then *H*1 and *H*2 are dimaps and are the desired dihomotopies.

−→ −→ −→

−→ We will show that in the right *context* it is no longer true that *I* , *I* × *I* , and

*X* are dihomotopy equivalent to a point.

Definition 2.8 Let the *context* be an object *A* ∈ Pospace. Instead of working in the category Pospace we will work in the category A ↓ Pospace of pospaces under

*ιB*

*A*. The objects of A ↓ Pospace are dimaps *A* −→ *B* where *B* ∈ Ob Pospace. The

morphisms in A ↓ Pospace are dimaps

*A* ¸¸

*ιB*

¸¸*ιC*

¸¸¸

*j* *f* v*z*

*B C*

such that *f* ◦ *ιB* = *ιC* .

Example 2.9 For example if *A* = *S*0 = {*a, b*} then *B* ∈ Ob A ↓ Pospace is a

−→

pospace with two marked points. An important example is *I* with *ι*→− (*a*)=0 and

*I*

*ι*−→(*b*)= 1.

*I*

Definition 2.10 • Given dimaps *f, g* : *B* → *C* ∈ A ↓ Pospace, *φ* is a *dihomotopy*

−→

from *f* to *g* if *φ* : *B* × *I* → *C* ∈ Pospace, *φ*|*B*×{0} = *f* , *φ*|*B*×{1} = *g*, and for all

*a* ∈ *A*, *φ*(*ιB*(*a*)*, t*)= *ιC*(*a*). In this case write *φ* : *f* → *g*.

* Write *f* *g* if there is a chain of dihomotopies *f* → *f*1 ← *f*2 → *...* ← *fn* → *g*. This is an equivalence relation.
* A dimap *f* : *B* → *C* is a *dihomotopy equivalence* if there is a dimap *g* : *C* → *B*

such that *g* ◦ *f* Id*B* and *f* ◦ *g* Id*C*. In this case write *B* *C*.

We can think of this as *dihomotopy rel A*. In case the context *A* is one point or two points we get pointed and bipointed dihomotopies. However we will see that this notion is useful for more general contexts.

Example 2.11 Let us return to the example above. In the context of its end points

−→

*I* is no longer dihomotopic to a point. There is a dimap

*S*0,

*ι*→−

,,,*ι*

*I*

−→ *j* *f*

*I*

∗

,,,

, z *z*

∗

−→

making the diagram commute, but there is no map *g* : ∗→ *I* making the diagram

commute.

Example 2.12 In the context of *S*0 = {*a, b*} let *ι*−→(*a*)= 0, *ι*−→(*b*)= 1, *ι*−→

−→(*a*)=

*I I* −→

−→*I* × *I*−→

(0*,* 0), and *ι*−→ −→(*b*) = (1*,* 1). We claim that in this context *I* and

*I* × *I* are

*I* × *I*

dihomotopy equivalent. Let *f* :

−→ −→

*I* × *I* →

−→ −→

*I* and *g* : *I* →

−→ −→

*I* × *I* be given by

*f* (*x, y*)= max(*x, y*) and *g*(*x*)= (*x, x*). Then *f* and *g* are both dimaps, *f* ◦ *g* = Id−→

*I*

and *g* ◦ *f* (*x, y*)= (max(*x, y*)*,* max(*x, y*)). It remains to construct a dihomotopy rel

*S*0 from Id−→ −→ to *g* ◦ *f* .

*I* × *I*

Let *φ* be the linear interpolation (see Definition [2.4](#_bookmark8)) of Id−→

−→ and *g* ◦ *f* . That

is,

*φ*(*x, y, t*)= (1 − *t*)(*x, y*)+ *t*(max(*x, y*)*,* max(*x, y*))

= (*x* + *t*(max(*x, y*) − *x*)*,y* + *t*(max(*x, y*) − *y*))*.*

Then *φ* is the desired dihomotopy rel {*a, b*}.

*I* × *I*

−→ −→ −→

Hence

*I* × *I* and *I* are dihomotopy equivalent in the given context.

# Context and the fundamental category

We will now introduce some definitions and prove some lemmas that will allow us to relate dihomotopy rel *A* to the *fundamental category*. Furthermore it will enable us to quickly see that certain spaces are not dihomotopy equivalent in a given context.

Definition 3.1 Let *B* ∈ Pospace and let *x, y* ∈ *B*.

−→

* A *dipath* is a dimap *γ* : *I* → *B*.

−→

* Let *γ*1*, γ*2 : *I* → *B* be dipaths such that *γ*1(0) = *γ*2(0) = *x* and *γ*1(1) = *γ*2(2) = *y*.

Then *γ*1 and *γ*2 are dihomotopic if they are dihomotopy equivalent with respect to

their endpoints. That is, *γ*1 *γ*2 in S0 ↓ Pospace where *ι*−→(*a*)= 0, *ι*−→(*b*)= 1,

*I* *I*

*ιB*(*a*)= *x*, and *ιB*(*b*)= *y*. In this case write *γ*1 *γ*2.

* Let −→*π* 1(*B*)(*x, y*) be the set of dihomotopy classes of dipaths from *x* to *y*. The *fundamental category* of *B* is the category −→*π* 1(*B*) whose objects are the points of *B* and whose morphisms between *x* and *y* are the elements of −→*π* 1(*B*)(*x, y*). [5](#_bookmark15)

Lemma 3.2 *Given dihomotopic dipaths γ* *γ*'

*then f* ◦ *γ* *f* ◦ *γ*' *are dihomotopic dipaths.*

−→

: *I* → *B and a dimap f* : *B* → *C,*

' *H*1

*H*2 *H*3 *Hn*

Proof. Since *γ* *γ* there is a chain of dihomotopies *γ* −−→ *γ*1 ←−− *γ*2 −−→ *...* ←−−

*Hn*+1 '

*f* ◦*H*1

*f* ◦*H*2

*f* ◦*H*3

*f* ◦*Hn*

*f* ◦*Hn*+1 '

*γn* −−−→ *γ* . Then *f* ◦*γ* −−−→ *f* ◦*γ*1 ←−−− *f* ◦*γ*2 −−−→ *...* ←−−− *f* ◦*γn* −−−−−→ *f* ◦*γ*

is a chain of dihomotopies from *f* ◦ *γ* to *f* ◦ *γ*'.

Corollary 3.3 *For a dimap f* : *B* → *C and x, y* ∈ *B there is an induced map*

−→*π* 1(*f* ): −→*π* 1(*B*)(*x, y*) → −→*π* 1(*C*)(*f* (*x*)*,f* (*y*)) *mapping* [*γ*] '→ [*f* ◦ *γ*]*. That is, a dimap*

*f* : *B* → *C induces a functor* −→*π* 1(*f* ): −→*π* 1(*B*) → −→*π* 1(*C*)*.*

Lemma 3.4 *Given dihomotopic dimaps f* *g* : *B* → *C* ∈ A ↓ Pospace *and a*

−→

*dipath γ* : *I* → *B such that γ*(0) = *ιB*(*a*) *and γ*(1) = *ιB*(*b*) *where a, b* ∈ *A then*

*f* ◦ *γ* *g* ◦ *γ are dihomotopic dipaths.*

*H*1 *H*2 *H*3 *Hn*

Proof. Since *f* *g* there is a chain of dihomotopies *f* −−→ *f*1 ←−− *f*2 −−→ *...* ←−−

*Hn*+1

' −→

*fn* −−−→ *g*. For 1 ≤ *i* ≤ *n* + 1, let *Hi* = *Hi* ◦ (*γ* × *I* ).

−→ −→

*I* × *I*

−→ ¸ ¸ *H*'

*i*

*γ*× *I* J¸ ¸

*C*

−→ *Hi*

¸z

*B* × *I*

' ' ' '

*H*

*H*

*H*

*H*

1 2 3 *n*

'

*n*+1

*H*

Then *f* ◦ *γ*

−−→ *f*1 ◦ *γ* ←−− *f*2 ◦ *γ* −−→ *...* ←−− *fn* ◦ *γ*

−−−→ *g* ◦ *γ* is a chain of

dihomotopies from the dipath *f* ◦ *γ* to the dipath *g* ◦ *γ*.

5 This differs from the definition of fundamental category in [[5](#_bookmark37)] where the dihomotopy classes of dimaps

−→

use *I* and not *I* .

Proposition 3.5 *If f* : *B* → *C* ∈ A ↓ Pospace *is a dihomotopy equivalence* *then for all a, b* ∈ *A the induced set map* −→*π* 1(*f* )(*a, b*) : −→*π* 1(*B*)(*ιB*(*a*)*, ιB*(*b*)) →

−→*π* 1(*C*)(*ιC*(*a*)*, ιC* (*b*)) *is a bijection.*

Proof. By definition there is a dimap *g* : *C* → *B* such that *g* ◦ *f* Id*B* and

−→

*f* ◦ *g* Id*C*. So by Lemma [3.4](#_bookmark13), for any *a, b* ∈ *A*, any dipath *γ* : *I* → *B* such that

*γ*(0) = *ιB*(*a*) and *γ*(1) = *ιB*(*b*) and any dipath *γ*'

−→

: *I* → *C* such that *γ*(0) = *ιC*(*a*)

and *γ*(1) = *ιC*(*b*), *g* ◦ *f* ◦ *γ* *γ* and *f* ◦ *g* ◦ *γ*' *γ*'. Hence −→*π* 1(*g*)(*a, b*) is an inverse

for −→*π* 1(*f* )(*a, b*).

0 −→ −→

Example 3.6 Let *A* = *S*

= {*a, b*} and choose any points *x, y* ∈ *I* × *I* such that

*x* # *y* and *y* #

−→ −→ −→

−→ −→ −→

*x*. Then the sets

*π* 1( *I* × *I* )(*x, y*) and

*π* 1( *I* × *I* )(*y, x*) are empty.

However for any dimap *f* :

−→ −→

*I* × *I* →

−→

*I* (see Figure [2](#_bookmark2)), either *f* (*x*) ≤ *f* (*y*) or

−→ −→

*f* (*y*) ≤ *f* (*x*) since *I* is totally ordered. Furthermore one of *π* 1( *I* )(*f* (*x*)*,f* (*y*)) and

−→ −→

−→ −→(*a*)= *x* and *ι*−→

−→(*b*)=

*π* 1( *I* )(*f* (*y*)*,f* (*y*)) is nonempty. So in the context of *ι I* × *I*

*I* × *I*

−→

*y*, *I* ×

−→

*I* is not dihomotopy equivalent to

−→

*I* since there can be no dihomotopy

−→ −→ −→ −→

equivalence *f* : *I* × *I* → *I* such that *π* 1(*f* )(*a, b*) is an isomorphism.

−→

Example 3.7 Let *X* be the space defined earlier (see Figure [1](#_bookmark1)). In the context

−→

of its four endpoints (0)1, (0)2, (1)1, and (1)2, *X* is not dihomotopy equivalent to

−→

*I* (taking any four not necessarily distinct points as the context for

−→

*I* ). Indeed,

there are no dipaths from (0)1 to (0)2 and vice versa (similarly for (1)1 and (1)2),

−→

whereas the same is not true for the corresponding points in *I* .

# Finding simpler models using context

In this section we look at two two-dimensional pospaces with a given context. We show how each of them can be replaced with an equivalent one-dimensional pospace by constructing explicit directed homotopy equivalences.

−→ −→

Fig. 4. *I* × *I* with a square removed

Example 4.1 In this example we show that in the context of the points (0*,* 0) and

(1*,* 1),

−→ −→

*I* × *I* with a square removed from its interior is dihomotopy equivalent to

its boundary.

Let *A* = *S*0

= {*a, b*}. Let *B* be the sub-pospace of

−→ −→

*I* × *I* in Figure [4](#_bookmark17) given by

−→ −→

1 2

1 2

*I* × *I* −

3 *,* 3

−→

→

−× 3 *,* 3

. Let *ιB*(*a*)= (0*,* 0) and let *ιB*(*b*)= (1*,* 1). Let *C* be the

boundary of *I* × *I* with *ιC* (*a*)= (0*,* 0) and *ιC*(*b*)= (1*,* 1).

Intuitively we will contract *B* to *C* in two stages. First we will expand the missing square ( 1 *,* 1 ) × ( 1 *,* 1 ) to ( 1 *,* 1) × ( 1 *,* 1) and then to (0*,* 1) × (0*,* 1). The first

3 3 3 3 3 3

will be done by a map *h* which we define below and the composite of the two will yield the desired dihomotopy equivalence *f* .

// /

// /

// /

/ /*F*2

/

0

2

3 3

1

Fig. 5. The graphs of *F*1, *F*2, and *F*2 ◦ *F*1.

Let *F*1 : [0*,* 1] → [0*,* 1] be given by the mapping

1

⎧

⎪ *x* if *x <*

*x* '→

⎪⎪⎨

⎪

1

2*x* −

3

3

1 2

if ≤ *x* ≤

3 3

2

⎪⎩ 1 if *x >*

3

Let *F*2 : [0*,* 1] → [0*,* 1] be given by the mapping

⎧⎪⎨

1

0 if *x <*

*x* '→ 3 1 1 3

⎪⎩ 2 *x* − 2 if 3 ≤ *x* ≤ 1

See Figure [5](#_bookmark18) for graphs of *F*1, *F*2, and *F*2 ◦ *F*1.

Let *f* : *B* → *C* and *g* : *C* → *B* be given by *f* (*x, y*) = (*F*2 ◦ *F*1(*x*)*, F*2 ◦ *F*1(*y*)) and *g*(*x, y*)= (*x, y*). Also let *h* : *B* → *B* be given by *h*(*x, y*)= (*F*1(*x*)*, F*1(*y*)). Since *F*2 ◦ *F*1, Id*I* , and *F*1 are increasing maps, *f* , *g*, and *h* are dimaps.

We will now give explicit dihomotopies rel *A* showing that *g* ◦ *f* Id*B* rel *A* and *f* ◦ *g* Id*C* rel *A*. Let *H*1 and *H*2 be linear interpolations between Id*B* and *h* and between *g* ◦ *f* and *h*. That is,

*H*1(*x, y, t*)= (1 − *t*)(*x, y*)+ *t*(*F*1(*x*)*, F*1(*y*))*,* and

*H*2(*x, y, t*)= (1 − *t*)(*F*2 ◦ *F*1(*x*)*, F*2 ◦ *F*1(*y*)) + *t*(*F*1(*x*)*, F*1(*y*))*.*

Note that *F*1 and *F*2 fix 0 and 1, so *H*0 and *H*1 fix the marked points (0*,* 0) and

(1*,* 1). By Lemma [2.6](#_bookmark9), *H*1 : Id*B* −→

*h* is a dihomotopy rel *A*. *H*2 is a dimap since

*F*2 ◦ *F*1(*x*) ≤ *F*1(*x*) for all *x* ∈ *I*, so *h* ≤ *g* ◦ *f* . Thus by Lemma [2.6](#_bookmark9), *H*2 : *g* ◦ *f* −→ *h*

is a dihomotopy rel *A*. Therefore *g* ◦ *f* Id*B* rel *A* as claimed. Furthermore since *C* is a sub-pospace of *B* and *f* ◦ *g* = *f* = *g* ◦ *f* , the above dihomotopies restrict to *C* showing that *f* ◦ *g* Id*C* rel *A*.

Example 4.2 The Swiss flag.

*b*

*c d*

*a*

Fig. 6. The Swiss flag with labeled points {*a, b, c, d*}

*b*

*c d*

*a*

Fig. 7. A sub-pospace of the Swiss flag with the same labeled points {*a, b, c, d*}



Fig. 8. An intuitive sketch of the dihomotopy equivalence between the Swiss flag and its sub-pospace

In this example we give an explicit dihomotopy between the famous Swiss flag pospace in Figure [6](#_bookmark19) and the one-dimensional sub-pospace in Figure [7](#_bookmark20) in the context

of four points.

Let *A* be the discrete pospace {*a, b, c, d*}. Let *B* be the sub-pospace of

−→ −→

*I* × *I*

given in Figure [6](#_bookmark19) with the (open) cross removed and *ιB*(*a*)= (0*,* 0), *ιB*(*b*)= (1*,* 1),

*ιB*(*c*) = ( 2 *,* 2 ), and *ιB*(*d*) = ( 3 *,* 3 ). Let *C* be the subspace of *B* given in Figure [7](#_bookmark20)

5 5 5 5

with the same marked points.

Intuitively we will contract *B* to *C* be applying four maps which are described in Figure [8](#_bookmark21).

Let *g* : *C* → *B* be the dimap given by *g*(*x, y*) = (*x, y*). Let *f* : *B* → *C* be the dimap given by *f* (*x, y*)= *f*4 ◦ *f*3 ◦ *f*2 ◦ *f*1(*x, y*) where *f*1, *f*2, *f*3, and *f*4 are defined in ([1](#_bookmark22)) below. ¿From the sketches of *f*1*, f*2*, f*3, and *f*4 in Figure [8](#_bookmark21), one can see that

they are dimaps. As in the previous example we will give a chain of dihomotopies rel *A* to show that Id*B* *g* ◦ *f* . Since *C* is a subspace of *B* and *g* ◦ *f* = *f* = *f* ◦ *g* this will restrict to a chain of dihomotopies rel *A* which show that Id*C* *f* ◦ *g*. As a result we will have that *B* *C*.

*f*1(*x, y*)= ⎪⎨

⎪

5 5

1 1 1

⎧⎪ (max(*x, y*)*,* max(*x, y*)) if 0 ≤ *x* ≤ 1 *,* 0 ≤ *y* ≤ 1

( *, y*) if 0 ≤ *x* ≤ *, * *< y*

5 5 5

1 1 1

(1)

(*x,*

⎪⎩

⎧⎪ (min(*x, y*)*,* min(*x, y*)) if 4 ≤ *x* ≤ 1*,* 4 ≤ *y* ≤ 1

(*x, y*) otherwise

) if 0 ≤ *y* ≤ *, < x*

5 5 5

5 5

⎪ 4 4 4

*f*2(*x, y*)= ⎨

( *, y*) if  ≤ *x* ≤ 1*,y < *

5 5 5

⎪⎪⎩

(*x,*

4

) if

5

(*x, y*) otherwise

4 4

≤ *y* ≤ 1*,x <*

5 5

⎧⎪ (max(*x, y* −

⎪

(max(*x, y* +

⎪

2

)*,* max(*x* +

5

2

)*,* max(*x* −

5

2

*, y*)) if

5

2

*, y*)) if

5

1 2 3

≤ *x* ≤ *,*

5 5 5

1 2 3

≤ *y* ≤

*,*

5

5

5

4

≤ *y* ≤

5

4

≤ *x* ≤

5

3

(max(*x, y*)*,* max(*x, y*)) if

⎪⎨

≤ *x* ≤

4 3 4

*,* ≤ *y* ≤

*f*3(*x, y*)=

⎪

2

( + 2(*x* −

5

2

2

)*, y*) if

5

2

5

2

≤ *x* ≤

5

2

5 5 5

3 4

*,y* =

5 5

3 4

⎪⎪⎩ ⎧

(*x,* + 2(*y* − )) if ≤ *y* ≤ *,x* =

5 5 5 5 5

(*x, y*) otherwise

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 |  | 2 1 |  | 2 |
| 5 |  | 5 5 |  | 5 |

(min(*x, y*)*,* min(*x, y*)) if

⎪

≤ *x* ≤ *,*

≤ *y* ≤

*f*4(*x, y*)= ⎪⎨

3 3

( − 2(

5 5

— *x*)*, y*) if

2

≤ *x* ≤

5

3 1

*,y* =

5 5

⎪⎪⎩

3

(*x,*

5

3

— 2(

5

— *y*))

2

if ≤ *y* ≤

5

(*x, y*) otherwise

3 1

*,x* =

5 5

Let *H*1, *H*2, *H*3, and *H*4 be the linear interpolations (see Definition [2.4](#_bookmark8)) between Id*B* and *f*1, *f*2 ◦ *f*1 and *f*1, *f*2 ◦ *f*1 and *f*3 ◦ *f*2 ◦ *f*1, and *f* and *f*3 ◦ *f*2 ◦ *f*1. Since the *fi* fix the labeled points, so do the *Hi*. Furthermore, since *f*1,*f*2,*f*3, and *f*4 are dimaps, *f*1 and *f*3 are increasing and *f*2 and *f*4 are decreasing, by Lemma [2.6](#_bookmark9), the *Hi* form a chain of dihomotopies

*H*1 *H*2 *H*3 *H*4

Id*B* −−→ *f*1 ←−− *f*2 ◦ *f*1 −−→ *f*3 ◦ *f*2 ◦ *f*1 ←−− *f* = *g* ◦ *f.*

Therefore Id*B* *g* ◦ *f* . Restricting to *C* gives a chain of dihomotopies showing

Id*C* *f* = *f* ◦ *g*. Hence *B* is dihomotopy equivalent to *C* rel {*a, b, c, d*}.

# Pushouts of dihomotopy equivalences

In this section we elaborate on the statement made in the introduction that di- homotopy equivalences should be preserved by ‘pastings’. In fact we discuss the construction of a homotopy theory for concurrency. In order that we do not lose focus from the main ideas of this paper, we will defer the details of the definitions and constructions of this section to the appendix.

An excellent framework for a homotopy theory on a category is given by a *model structure* on the category [[12](#_bookmark43)]. A category with a model structure and all small limits and colimits is called a *model category*. A model structure has three special classes of morphisms: *ﬁbrations*, *coﬁbrations*, and *weak equivalences* which satisfy certain axioms (see Appendix [A](#_bookmark45) for the full definition).

The category Pospace has all small limits and colimits. However it is too re- strictive to model many concurrent systems (for example pospaces cannot contain loops). Though all of our examples are in Pospace a better framework for concur- rency is the category LoPospc of *local pospaces*. A *local pospace* is a topological space such that each point has a neighborhood which is a pospace and that these local orders are compatible (for a precise definition see Appendix [B](#_bookmark46)).

−→

1

Fig. 9. The local pospace *S*

−→

Example 5.1 An example of a local pospace is the directed circle *S*

1

in Figure [9](#_bookmark23)

obtained by identifying the endpoints of

−→ −→

*I* . While *S*

1

does not have a transitive,

−→

anti-symmetric order, locally it has the structure of the pospace *I* .

Unfortunately, unlike Pospace, LoPospc does not contain all small colimits. However there is a formal method of enlarging a category to one with all small limits and colimits. [6](#_bookmark25) Furthermore this larger category has a canonical model struc- ture! [[2](#_bookmark33)] For details on how this theory can be applied to LoPospc see the appendix and [[1](#_bookmark32)]. In the appendix we give a more precise version of the following theorem (Theorem [B.4](#_bookmark48)) which is proved in [[1](#_bookmark32)].

Theorem 5.2 *Let* C = LoPospc*. Then* C *is a subcategory of a model category* UC*. The morphisms in* C *that are coﬁbrations are the monomorphisms and the morphisms in* C *that are weak equivalences are the isomorphisms.*

¿From the point of view of just C, this model structure is almost trivial. However one can *localize* UC with respect to a set *M* of morphisms in C to obtain a new category UC*/*M. UC*/*M has the same objects and cofibrations as UC but the morphisms in *M* are now weak equivalences [[2](#_bookmark33)]. The problem is to choose a good

6 Again more details are provided in the appendix (one passes to the category of simplicial presheaves [[2](#_bookmark33)]).

set of morphisms *M* . For example, we can take *M* to be the set of dihomotopy equivalences in C.

One of the key properties of UC and UC*/*M is that they are *left proper*. That is, the pushout of a weak equivalence over a cofibration is a weak equivalence.

*G*  ∼

J *f C*J

*j*

J∼ J

*D g E*

In particular in UC*/*M if *f* ∈ *M* then *g* is a weak equivalence.

−→ −→ −→

Example 5.3 Recall the dihomotopy equivalence *f* :

*I* × *I* → *I* of Example [2.12](#_bookmark12).

Also recall the inclusions of

−→ −→

*I* × *I* and

−→

*I* into *B* and *C* (see Figure [3](#_bookmark4)) given in

−→ −→

' ' −→

Example [1.2](#_bookmark3) where attachments are made at the points *x, y* ∈

(see Figure [2](#_bookmark2)). We have the following pushout diagram.

*I* × *I* and *x ,y* ∈ *I*

−→ −→ ∼ −→

*I* × *I f I*

JJ

*j*

JJ

*B g*

*C*

Since the inclusion *j* is a cofibration, we get a weak equivalence between *B* and *C*. However as discussed in Example [1.2](#_bookmark3), *B* should not be equivalent to *C*.

The solution to this problem is to work with A ↓ LoPospc instead of LoPospc

where the choice of context *A* ∈ Ob LoPospc depends on the pushouts that one

would like to consider.

In the example above the right context is clearly the points *x, y* ∈

−→ −→

*I* × *I* and

' ' −→

−→ −→(*a*)= *x*, *ι*−→

−→(*b*)= *y*, *ι*−→(*a*)= *x*', and *ι*−→(*b*)= *y*'.

*x ,y* ∈

*I* . So *A* = {*a, b*}, *ι I* × *I*

*I* × *I I* *I*

As discussed in Example [3.6](#_bookmark16) the map *f* is not a dihomotopy equivalence rel *A*. So

we are not forced to conclude that there is a weak equivalence between *B* and *C*.

−→ −→

In the following two examples we examine the ‘pastings’ of two copies of *I* × *I*

with a square removed. We show how choosing the right context allows us to find a one-dimensional sub-pospace which is dihomotopy equivalent to the pushout.

Unlike the previous section, we will not write out the explicit dihomotopy equiv- alences in these two examples.

Example 5.4 Let *A* be the discrete space {*a, b, c*}. Let *B* be the subspace of

−→ −→

1 2 2 3

*I* × *I* in Figure [10](#_bookmark26) with the square {(*x, y*) | 5 *< x <* 5 *,* 5 *< y <* 5 } removed. Let

*ιB*(*a*)= (0*,* 0), *ιB*(*b*)= ( 1 *,* 0), and *ιB*(*c*)= (1*,* 0).

2 −→ −→

Let *C* be the subspace of *I* × *I* in Figure [11](#_bookmark27) with the square {(*x, y*) | 3 *< x <*

5

4 *,* 2 *< y <* 3 } removed. Let *ιB*(*a*)= (0*,* 1), *ιB*(*b*)= ( 1 *,* 1), and *ιB*(*c*)= (1*,* 1).

1

5 5 5

'

Let *B*

be the subspace of

−→ −→ 2

removed and the same marked points. Then there is a dihomotopy equivalence

*I* × *I* in Figure [10](#_bookmark26) with the rectangle

0*,* 2

× ]0*,* 1[

*f* : *B* −→ *B*' rel *A*. One can construct the required dihomotopies by stretching the

a b c a b c

' −→ −→

Fig. 10. The spaces *B* and *B* , which are subspaces of *I* × *I* with a rectangle removed and labeled points

{*a, b, c*}

a b c a b c

' −→ −→

Fig. 11. The spaces *C* and *C* , which are subspaces of *I* × *I* with a square removed and labeled points

{*a, b, c*}

region 2 ≤ *y* ≤ 3

first to *y* = 1 and then to *y* = 0. Next one stretches the region

5 5

1 ≤ *x* ≤ 2 first to *x* = 1 and then to *x* = 0. All this is done while leaving the three

5 5 2

marked points fixed.

Similarly there is a dihomotopy equivalence *g* : *C* −→ *C*' rel *A* where *C*'

is the

−→ −→ 1

subspace of

removed.

*I* × *I* in Figure [11](#_bookmark27) with the rectangle {(*x, y*)| 2 *< x <* 1*,* 0 *< y <* 1}

Let *D* be the space obtained by attaching *B* along its bottom edge to the top edge of *C*. Notice that *D* ∈ Ob A ↓ Pospace and the inclusions *i* : *B* → *D* and *j* : *C* → *D* are dimaps in A ↓ Pospace.

Now take the following pushout.

*B*  ∼ '

J *f*

*i*

J∼

*B*J

J

1. *f* ' *E*

Then *E* is the pospace obtained by attaching the bottom edge of *B*' to the top edge of *C*. Since *C* includes into *E* we can take the following pushout.

*C*  ∼ '

J *g*

*i*

J∼

*C*J

J

1. *f* '' *F*

Now *F* is the pospace [7](#_bookmark28) in Figure [12](#_bookmark29) obtained by attaching the bottom edge of *B*'

to the top edge of *C*'.

7 Being precise, if we consider the pushout *F* to be a subspace of *I* × *I* then the points [0*,* 1 ] × { 1 } are

5 2

identified as are the points [ 2 *,* 3 ] ×{ 1 } and the points [ 4 *,* 1] ×{ 1 }. However this pospace is dihomotopy

5 5 2

5 2 ' '

equivalent to the pospace obtained by attaching the bottom edge of *B* to the top edge of *C* .

*c*

*b*

*a*

Fig. 12. The pospaces *F* and *G*

*a b*

*c*

Fig. 13. The pospaces *F* ' and *G*'

Finally *F* is dihomotopy equivalent rel *A* to the space *G* in Figure [12](#_bookmark29). Consider

−→ −→

*F* and *G* as sub-pospaces of *I* × *I* . The dihomotopy is obtained by first collapsing

the square [ 1 *,* 1] × [ 1 *,* 1] using (*x, y*) '→ (max(*x, y*)*,* max(*x, y*)), and then collapsing

2 2

the square [0*,* 1 ] × [0*,* 1 ] using (*x, y*) '→ (min(*x, y*)*,* min(*x, y*))

2 2

Thus in the context of *A*, *D* is equivalent to *G*.

Example 5.5 Let *A*, *B*, *C*, *B*' and *C*' be as in the previous example, except that the marked points on *B* and *B*' are taken to be on the top edge, and the marked points on *C* and *C*' are taken to be on the bottom edge. Let *D*' be the space obtained by attaching *C* along its bottom edge to the top edge of *B*.

Then as in the previous example *D*' is dihomotopy equivalent to *F* ' where *F* ' is the pospace in Figure [13](#_bookmark30) obtained by attaching the bottom edge of *C*' to the top edge of *B*'.

Finally *F* ' is dihomotopy equivalent rel *A* to the space *G*' in Figure [13](#_bookmark30). Consider

' ' −→ −→

1. and *G* as sub-pospaces of *I* × *I* . The dihomotopy is obtained by collapsing the

regions [ 1 *,* 1]×[0*,* 1 ] using (*x, y*) '→ (*x,* 1 ), and then collapsing the square [0*,* 1 ]×[ 1 *,* 1]

2 2 2 2 2

using (*x, y*) '→ (*x,* 1 ).

2

Thus in the context of *A*, *D*' is equivalent to *G*'.

Example 5.6 Finally we give an example which requires a non-discrete context.

Let *X* =

−→ −→

1. × *I* . We will show that if we want to use *X* to construct a certain

space *Z* then there is no appropriate finite context.

Let *I* be the unit interval [0*,* 1] together with the trivial partial order given by

*x* ≤*I y* if and only if *x* = *y*.

Fig. 14. *X* and *Y* with the images of *I* marked

Let *ϕ* : *I* → *X* be the inclusion of the anti-diagonal, given by *t* '→ (*t,* 1 − *t*) (see

−→

Figure [14](#_bookmark31)). Let *Y* = *I* × *I* and let *ψ* : *I* → *Y* be the inclusion of the central line,

given by *t* '→ (*t,* 1 ) (see Figure [14](#_bookmark31)). Define the pospace *Z* obtained by gluing *X* and

2

1. together along the images of *I*. That is, *Z* is the following pushout.

*I*  *ϕ*

J

*ψ* J

*Y ιY*

*X*

J

J*ιX*

*Z*

We claim that if we want to consider this pushout then there is no appropriate finite context.

For *α* ∈ *I* let *pα* := *ιX* (*ϕ*(*α*)) = *ιY* (*α,* 1 ), *p*0 = *ιY* (*α,* 0) and *p*1

= *ιY* (*α,* 1).

2 *α α*

Notice that for *s* /= *t* ∈ *I* there does not exist a dipath in *Z* from *p*0 to *p*1.

*s t*

Now let *A* be some context and fix *ιI* : *A* → *I* which determines *ιX* : *A* → *X* and *ιY* : *A* → *Y* . Let *f* : *X* → *X*' be some dihomotopy equivalence rel *A*. Let *Z*' and *g* be defined by the following pushout.

*X*  '

J *f X*J

*ιX* JJ

*Z*  '

*g Z*

Assume there exists *s* /= *t* ∈ *I* such that *f* (*ϕ*(*s*)) = *f* (*ϕ*(*t*)). We claim that there is a dipath from *g*(*p*0) to *g*(*p*1). In *Z* there is a dipath from *p*0 to *ps* and a dipath

*s t s*

from *pt* to *p*1. The concatenation of the images of these paths under *g* gives the

*t*

desired dipath in *Z*'. But this contradicts Proposition [3.5](#_bookmark14).

Therefore there should not have been an equivalence *f* such that *f* (*ϕ*(*s*)) = *f* (*ϕ*(*t*)) for some *s* /= *t* ∈ *I*. We can prevent this difficulty if we use the context *A* = *I* together with *ιI* = Id*I* .

On the other hand with any finite context *A*, we claim that there is always such a dihomotopy equivalence *f* . Assume that *A* is a finite context and fix *ιI* : *A* → *I*. Since *A* is finite there is some interval [*a, b*] ⊂ *I*\*ιI* (*A*). Let *f* : *X* → *X* be the dihomotopy equivalence given by the concatenation of the following two maps. First collapse the region [*a, b*] × *I* to the right. Then collapse the region *I* × [1 — *b,* 1 — *a*] upwards. Then *f* is a dihomotopy equivalence rel *A* but *f* (*ϕ*(*a*)) = *f* (*ϕ*(*b*)).

# Acknowledgement

I would like to thank Eric Goubault, Kathryn Hess, Krzysztof Worytkiewicz, and Emmanuel Haucourt for introducing me to the study of concurrency using topology and category theory and for many helpful discussions.

# References

1. Bubenik, P. and K. Worytkiewicz, *A model category structure for local po-spaces*, arXiv:math.AT/0506352 (2005).

URL <http://arxiv.org/abs/math.AT/0506352>

1. Dugger, D., *Universal homotopy theories*, Adv. Math. 164 (2001), pp. 144–176.
2. Fajstrup, L., *Dicovering spaces*, Homology Homotopy Appl. 5 (2003), pp. 1–17 (electronic), Algebraic topological methods in computer science (Stanford, CA, 2001).
3. Fajstrup, L., E. Goubault and M. Raussen, *Algebraic topology and concurrency*, to appear in Theoretical Computer Science (1999), also preprint R-99-2008, Dept. of Mathematical Sciences, Aalborg University, Aalborg, Denmark.
4. Fajstrup, L., M. Raussen, E. Goubault and E. Haucourt, *Components of the fundamental category*, Appl. Categ. Structures 12 (2004), pp. 81–108.
5. Gaucher, P., *A model category for the homotopy theory of concurrency*, Homology Homotopy Appl. 5

(2003), pp. 549–599.

1. Goubault, E., “The geometry of concurrency,” Ph.D. thesis, Ecole Normale Sup`erieure (1995).
2. Goubault, E., *Some geometric perspectives in concurrency theory*, Homology Homotopy Appl. 5 (2003),

pp. 95–136 (electronic), Algebraic topological methods in computer science (Stanford, CA, 2001).

1. Grandis, M., *Directed homotopy theory. I*, Cah. Topol. G´eom. Diff´er. Cat´eg. 44 (2003), pp. 281–316.
2. Grandis, M., *Directed homotopy theory. II. Homotopy constructs*, Theory Appl. Categ. 10 (2002),

pp. No. 14, 369–391 (electronic).

1. Hirschhorn, P. S., “Model categories and their localizations,” Mathematical Surveys and Monographs

99, American Mathematical Society, Providence, RI, 2003, xvi+457 pp.

1. Hovey, M., “Model categories,” Mathematical Surveys and Monographs 63, American Mathematical Society, Providence, RI, 1999, xii+209 pp.
2. Raussen, M., *State spaces and dipaths up to dihomotopy*, Homology Homotopy Appl. 5 (2003), pp. 257– 280 (electronic), Algebraic topological methods in computer science (Stanford, CA, 2001).

# Model Categories

In this section we define model categories, and show how a given small category can be embedded into a *universal model category*. For more details see [[2](#_bookmark33),[1](#_bookmark32)].

Definition A.1 A *model category* is a category C with three distinguished classes of morphisms: weak equivalences, cofibrations, and fibrations satisfying the follow- ing conditions:

* 1. C contains all small limits and colimits.
  2. If there exist morphisms *f* , *g* and *g* ◦ *f* and two of them are weak equivalences then so is the third.
  3. Weak equivalences, cofibrations, and fibrations are closed under retracts.
  4. Given any commutative diagram

*A*

*X*

*i p*

JJ

*B*

*Y*

such that *i* is a cofibration and *p* is a fibration, then if either *i* or *p* is also a weak equivalence then there exists a map *B* → *X* making the diagram commute.

* 1. Any map may be factored as a cofibration followed by a fibration which is a weak equivalence, and as a cofibration which is a weak equivalence followed by a fibration.

Next we define the category of simplicial presheaves.

Definition A.2 • The simplicial category Δ is the category whose objects are [*n*] = {0*,* 1*,... , n*} for *n* ≥ 0 and whose morphisms are maps *f* : [*n*] → [*k*] such that *x* ≤ *y* implies that *f* (*x*) ≤ *f* (*y*).

* The category of simplicial sets sSet is the category SetΔop whose objects are contravariant functors from Δ to the category of sets Set and whose morphisms are natural transformations.
* Let C be a small category. Then sPre(C) is the category sSetCop whose objects are the contravariant functors from C to sSet and whose morphisms are natural transformations.

Remark A.3 An important fact is that there is an embedding C → sPre(C).

The category sSet has a model structure in which the cofibrations are the monomorphisms and the weak equivalences are the morphisms *f* such that |*f* | the geometric realization of *f* is a weak equivalence in the category of topologi- cal spaces (that is, it induces isomorphisms between homotopy groups). For more details see [[12](#_bookmark43)].

The category of simplicial presheaves has a canonical model structure, called the *coﬁbrant model structure*, where the weak equivalences and the cofibrations are defined objectwise. That is, a morphism *f* in sPre(C) is a weak equivalence or cofibration if and only if for each *X* ∈ Ob C the morphism *f* (*X*) is a weak equivalence or cofibration in sSet.

Now one can localize this model category [[11](#_bookmark42)] with respect some set of morphisms *M* to get a new model category sPre(C)*/*M. This model category has the same objects, but in addition to the previous weak equivalences, the morphisms in *M* are now weak equivalences. For example if C = LoPospc then one could localize with respect to all dihomotopy equivalences (it makes sense to say this because of the embedding of C in sPre(C)).

# Local po-spaces

In this section we give a precise definition of the category LoPospc of local pospaces and use it to give a more precise version of Theorem [5.2](#_bookmark24). Local pospaces are defined in [[4](#_bookmark35),[1](#_bookmark32)]. Here we follow [[1](#_bookmark32)].

Definition B.1 • Given a topological space *M* , an *order atlas* on *M* is an open cover [8](#_bookmark47) *U* = {*Ui*} indexed by a set *I* such that each *Ui* is a pospace and that the orders are compatible. That is, given *x, y* ∈ *Ui* ∩ *Uj*, *x* ≤*i y* if and only if *x* ≤*j y*.

* Let *U* = {*Ui*} and *V* = {*Vj*} be two order atlases. Then *V* is said to be a *reﬁnement* of *U* if for any *Ui* and any *x* ∈ *Ui* there exists a *Vj* containing *x* which is a sub-pospace of *Ui*.

8 That is, each *Ui* is an open subset of *M* , and *M* = ∪*i*∈*I Ui*.

* + Two order atlases are said to be *equivalent* if they have a common refinement. One can check that this defines an equivalence relation.
  + Define a *local pospace* to be a topological space together with an equivalence class of order atlases.
  + Define a *dimap of local pospaces f* : (*M, U*¯ ) → (*N, V*¯ ) to be a continuous map

*f* : *M* → *N* such that for any choice of *V* = {*Vj*} ∈ *V*¯

there is some choice of

*U* = {*Ui*} ∈ *U*¯

pospaces.

such that for all *i, j* the partial map *f* : *Ui* → *Vj* is a dimap of

Definition B.2 Define LoPospc to be the category whose objects are local pospaces whose underlying topological spaces are subsets of R*n* for some *n*, [9](#_bookmark49) and whose mor- phisms are dimaps between local pospaces.

Remark B.3 Notice that we have restricted the class of local pospaces in our category. This is done precisely so that the resulting category LoPospc is a small category, which is used to apply the machinery of Appendix [A](#_bookmark45). For the purposes of concurrency, this does not seem to be a significant limitation. Furthermore, it may be possible that any local pospace can be ‘found’ in sPre(LoPospc).

Nevertheless, a consequence of this, is that the category Pospace in Defini- tion [2.2](#_bookmark6) is not a subcategory of LoPospc. Of course one could define a new category Pospace' whose objects are those pospaces whose underlying topological spaces are subsets of R*n* for some *n*. Then Pospace' is a subcategory of LoPospc. All of our examples are in Pospace'.

We can now give a more precise version of Theorem [5.2](#_bookmark24).

Theorem B.4 ([[1](#_bookmark32)]) *There exists a model structure on* sPre(LoPospc) *such that the coﬁbrations are the monomorphisms. Furthermore the morphisms in* LoPospc *which are weak equivalences in* sPre(LoPospc) *are just the isomorphisms.*

9 The local partial order need not be the one inherited from the usual partial order on R*n*.