Cubical Categories

Tanguy Massacrier and Georg Struth

April 18, 2024

Abstract

This AFP entry formalises cubical ω-categories and cubical ω-categories with connections in the style of single-set categories. Cubical categories, and the cubical sets on which they are based, have their origins and main applications in algebraic topology. Applications in computer science include homotopy type theory, higher-dimensional automata in concurrency theory and higher-dimensional rewriting. The single-set axiomatisation, introduced in these components and a companion paper, allows a formalisation based on Isabelle's type classes.

Contents

1 Introductory Remarks

Based on a formalisation of catoids and single-set categories in the AFP [2] we develop single-set axiomatisations for cubical ω-categories with and without connections. A detailed explanation of the single-set approach, the classical approach to cubical ω-categories and the proof of equivalence of the single-set and the classical approach can be found in a companion article [1]. Isabelle, with its high degree of proof automation, has been instrumental for developing the single-set axioms introduced in this article.

2 Indexed Catoids

theory *ICatoids*
imports *Catoids.Catoid*

begin

All categories considered in this component are single-set categories.

no-notation *src* (σ)

notation *True* (*tt*)
notation *False* (*ff*)

abbreviation *Fix* :: $('a \Rightarrow 'a) \Rightarrow 'a\ set$ **where**
$Fix\ f \equiv \{x.\ f\ x = x\}$

First we lift locality to powersets.

lemma (**in** *local-catoid*) *locality-lifting*: $(X \star Y \neq \{\}) = (Tgt\ X \cap Src\ Y \neq \{\})$
$\langle proof \rangle$

The following lemma about functional catoids is useful in proofs.

lemma (**in** *functional-catoid*) *pcomp-def-var4*: $\Delta\ x\ y \Longrightarrow x \odot y = \{x \otimes y\}$
$\langle proof \rangle$

2.1 Indexed catoids and categories

class *face-map-op* =
fixes *fmap* :: $nat \Rightarrow bool \Rightarrow 'a \Rightarrow 'a$ (∂)

begin

abbreviation *Face* :: $nat \Rightarrow bool \Rightarrow 'a\ set \Rightarrow 'a\ set$ ($\partial\partial$) **where**
$\partial\partial\ i\ \alpha \equiv image\ (\partial\ i\ \alpha)$

abbreviation *face-fix* :: $nat \Rightarrow 'a\ set$ **where**
$face\text{-}fix\ i \equiv Fix\ (\partial\ i\ ff)$

abbreviation $fFx\ i\ x \equiv (\partial\ i\ ff\ x = x)$

abbreviation $FFx\ i\ X \equiv (\forall x \in X.\ fFx\ i\ x)$

end

class *icomp-op* =
fixes *icomp* :: $'a \Rightarrow nat \Rightarrow 'a \Rightarrow 'a\ set$ (-$\odot_{\_}$-[70,70,70]70)

class *imultisemigroup* = *icomp-op* +
assumes *iassoc*: $(\bigcup v \in y \odot_i z.\ x \odot_i v) = (\bigcup v \in x \odot_i y.\ v \odot_i z)$

begin

sublocale *ims*: *multisemigroup* $\lambda x\ y.\ x \odot_i y$
⟨*proof*⟩

abbreviation $DD \equiv ims.\Delta$

abbreviation *iconv* :: $'a\ set \Rightarrow nat \Rightarrow 'a\ set \Rightarrow 'a\ set$ (-⋆-[70,70,70]70) **where**
$X \star_i Y \equiv ims.conv\ i\ X\ Y$

end

class *icatoid* = *imultisemigroup* + *face-map-op* +
assumes *iDst*: $DD\ i\ x\ y \Longrightarrow \partial\ i\ tt\ x = \partial\ i\ ff\ y$
and *is-absorb* [*simp*]: $(\partial\ i\ ff\ x) \odot_i x = \{x\}$
and *it-absorb* [*simp*]: $x \odot_i (\partial\ i\ tt\ x) = \{x\}$

begin

Every indexed catoid is a catoid.

sublocale *icid*: *catoid* $\lambda x\ y.\ x \odot_i y\ \partial\ i\ ff\ \partial\ i\ tt$
⟨*proof*⟩

lemma *lFace-Src*: $\partial\partial\ i\ ff = icid.Src\ i$
⟨*proof*⟩

lemma *uFace-Tgt*: $\partial\partial\ i\ tt = icid.Tgt\ i$
⟨*proof*⟩

lemma *face-fix-sfix*: *face-fix* = *icid.sfix*
⟨*proof*⟩

lemma *face-fix-tfix*: *face-fix* = *icid.tfix*
⟨*proof*⟩

lemma *face-fix-prop* [*simp*]: $x \in face\text{-}fix\ i = (\partial\ i\ \alpha\ x = x)$
⟨*proof*⟩

lemma *fFx-prop*: $fFx\ i\ x = (\partial\ i\ \alpha\ x = x)$
⟨*proof*⟩

end

class *icategory* = *icatoid* +
assumes *locality*: $\partial\ i\ tt\ x = \partial\ i\ ff\ y \Longrightarrow DD\ i\ x\ y$
and *functionality*: $z \in x \odot_i y \Longrightarrow z' \in x \odot_i y \Longrightarrow z = z'$

begin

Every indexed category is a (single-set) category.

sublocale *icat*: *single-set-category* $\lambda x\ y.\ x \odot_i y\ \partial\ i\ ff\ \partial\ i\ tt$
⟨*proof*⟩

abbreviation *ipcomp* :: $'a \Rightarrow nat \Rightarrow 'a \Rightarrow 'a$ (-⊗₋-[*70*,*70*,*70*] *70*) **where**
$x \otimes_i y \equiv icat.pcomp\ i\ x\ y$

lemma *iconv-prop*: $X \star_i Y = \{x \otimes_i y \mid x\ y.\ x \in X \wedge y \in Y \wedge DD\ i\ x\ y\}$
⟨*proof*⟩

abbreviation *dim-bound* $k\ x \equiv (\forall\, i.\ k \leq i \longrightarrow fFx\ i\ x)$

abbreviation *fin-dim* $x \equiv (\exists\, k.\ dim\text{-}bound\ k\ x)$

end

end

3 Cubical Categories

theory *CubicalCategories*
imports *ICatoids*

begin

All categories considered in this component are single-set categories.

3.1 Semi-cubical ω-categories

We first define a class for cubical ω-categories without symmetries.

class *semi-cubical-omega-category* = *icategory* +
assumes *face-comm*: $i \neq j \Longrightarrow \partial\ i\ \alpha \circ \partial\ j\ \beta = \partial\ j\ \beta \circ \partial\ i\ \alpha$
and *face-func*: $i \neq j \Longrightarrow DD\ j\ x\ y \Longrightarrow \partial\ i\ \alpha\ (x \otimes_j y) = \partial\ i\ \alpha\ x \otimes_j \partial\ i\ \alpha\ y$
and *interchange*: $i \neq j \Longrightarrow DD\ i\ w\ x \Longrightarrow DD\ i\ y\ z \Longrightarrow DD\ j\ w\ y \Longrightarrow DD\ j\ x\ z$
$\Longrightarrow (w \otimes_i x) \otimes_j (y \otimes_i z) = (w \otimes_j y) \otimes_i (x \otimes_j z)$
and *fin-fix*: $\exists\, k. \forall\, i.\ k \leq i \longrightarrow fFx\ i\ x$

begin

lemma *pcomp-face-func-DD*: $i \neq j \Longrightarrow DD\ j\ x\ y \Longrightarrow DD\ j\ (\partial\ i\ \alpha\ x)\ (\partial\ i\ \alpha\ y)$
⟨*proof*⟩

lemma *comp-face-func*: $i \neq j \Longrightarrow (\partial\partial\ i\ \alpha)\ (x \odot_j y) \subseteq \partial\ i\ \alpha\ x \odot_j \partial\ i\ \alpha\ y$
⟨*proof*⟩

lemma *interchange-var*:
assumes $i \neq j$
and $(w \odot_i x) \star_j (y \odot_i z) \neq \{\}$
and $(w \odot_j y) \star_i (x \odot_j z) \neq \{\}$

shows $(w \odot_i x) \star_j (y \odot_i z) = (w \odot_j y) \star_i (x \odot_j z)$
⟨*proof*⟩

lemma *interchange-var2*:
assumes $i \neq j$
and $(\bigcup a \in w \odot_i x. \bigcup b \in y \odot_i z.\ a \odot_j b) \neq \{\}$
and $(\bigcup c \in w \odot_j y. \bigcup d \in x \odot_j z.\ c \odot_i d) \neq \{\}$
shows $(\bigcup a \in w \odot_i x. \bigcup b \in y \odot_i z.\ a \odot_j b) = (\bigcup c \in w \odot_j y. \bigcup d \in x \odot_j z.\ c \odot_i d)$
⟨*proof*⟩

lemma *face-compat*: $\partial\ i\ \alpha \circ \partial\ i\ \beta\ = \partial\ i\ \beta$
⟨*proof*⟩

lemma *face-compat-var* [*simp*]: $\partial\ i\ \alpha\ (\partial\ i\ \beta\ x) = \partial\ i\ \beta\ x$
⟨*proof*⟩

lemma *face-comm-var*: $i \neq j \Longrightarrow \partial\ i\ \alpha\ (\partial\ j\ \beta\ x) = \partial\ j\ \beta\ (\partial\ i\ \alpha\ x)$
⟨*proof*⟩

lemma *face-comm-lift*: $i \neq j \Longrightarrow \partial\partial\ i\ \alpha\ (\partial\partial\ j\ \beta\ X) = \partial\partial\ j\ \beta\ (\partial\partial\ i\ \alpha\ X)$
⟨*proof*⟩

lemma *face-func-lift*: $i \neq j \Longrightarrow (\partial\partial\ i\ \alpha)\ (X \star_j Y) \subseteq \partial\partial\ i\ \alpha\ X \star_j \partial\partial\ \ i\ \alpha\ Y$
⟨*proof*⟩

lemma *pcomp-lface*: $DD\ i\ x\ y \Longrightarrow \partial\ i\ ff\ (x \otimes_i y) = \partial\ i\ ff\ x$
⟨*proof*⟩

lemma *pcomp-uface*: $DD\ i\ x\ y \Longrightarrow \partial\ i\ tt\ (x \otimes_i y) = \partial\ i\ tt\ y$
⟨*proof*⟩

lemma *interchange-DD1*:
assumes $i \neq j$
and $DD\ i\ w\ x$
and $DD\ i\ y\ z$
and $DD\ j\ w\ y$
and $DD\ j\ x\ z$
shows $DD\ j\ (w \otimes_i x)\ (y \otimes_i z)$
⟨*proof*⟩

lemma *interchange-DD2*:
assumes $i \neq j$
and $DD\ i\ w\ x$
and $DD\ i\ y\ z$
and $DD\ j\ w\ y$
and $DD\ j\ x\ z$
shows $DD\ i\ (w \otimes_j y)\ (x \otimes_j z)$
⟨*proof*⟩

lemma *face-idem1*: $\partial\ i\ \alpha\ x = \partial\ i\ \beta\ y \Longrightarrow \partial\ i\ \alpha\ x \odot_i \partial\ i\ \beta\ y = \{\partial\ i\ \alpha\ x\}$
⟨*proof*⟩

lemma *face-pidem1*: $\partial\ i\ \alpha\ x = \partial\ i\ \beta\ y \Longrightarrow \partial\ i\ \alpha\ x \otimes_i \partial\ i\ \beta\ y = \partial\ i\ \alpha\ x$
⟨*proof*⟩

lemma *face-pidem2*: $\partial\ i\ \alpha\ x \neq \partial\ i\ \beta\ y \Longrightarrow \partial\ i\ \alpha\ x \odot_i \partial\ i\ \beta\ y = \{\}$
⟨*proof*⟩

lemma *face-fix-comp-var*: $i \neq j \Longrightarrow \partial\partial\ i\ \alpha\ (\partial\ i\ \alpha\ x \odot_j \partial\ i\ \alpha\ y) = \partial\ i\ \alpha\ x \odot_j \partial\ i\ \alpha\ y$
⟨*proof*⟩

lemma *interchange-lift-aux*: $x \in X \Longrightarrow y \in Y \Longrightarrow DD\ i\ x\ y \Longrightarrow x \otimes_i y \in X \star_i Y$
⟨*proof*⟩

lemma *interchange-lift1*:
assumes $i \neq j$
and $\exists\, w \in W.\ \exists\, x \in X.\ \exists\, y \in Y.\ \exists\, z \in Z.\ DD\ i\ w\ x \wedge DD\ i\ y\ z \wedge DD\ j\ w\ y \wedge DD\ j\ x\ z$
shows $((W \star_i X) \star_j (Y \star_i Z)) \cap ((W \star_j Y) \star_i (X \star_j Z)) \neq \{\}$
⟨*proof*⟩

lemma *interchange-lift2*:
assumes $i \neq j$
and $\forall\, w \in W.\ \forall\, x \in X.\ \forall\, y \in Y.\ \forall\, z \in Z.\ DD\ i\ w\ x \wedge DD\ i\ y\ z \wedge DD\ j\ w\ y \wedge DD\ j\ x\ z$
shows $((W \star_i X) \star_j (Y \star_i Z)) = ((W \star_j Y) \star_i (X \star_j Z))$
⟨*proof*⟩

lemma *double-fix-prop*: $(\partial\ i\ \alpha\ (\partial\ j\ \beta\ x) = x) = (fFx\ i\ x \wedge fFx\ j\ x)$
⟨*proof*⟩

end

3.2 Type classes for cubical ω-categories

abbreviation *diffSup* :: $nat \Rightarrow nat \Rightarrow nat \Rightarrow bool$ **where**
$diffSup\ i\ j\ k \equiv (i - j \geq k \vee j - i \geq k)$

class *symmetry-ops* =
fixes *symmetry* :: $nat \Rightarrow 'a \Rightarrow 'a\ (\sigma)$
and *inv-symmetry* :: $nat \Rightarrow 'a \Rightarrow 'a\ (\vartheta)$

begin

abbreviation $\sigma\sigma\ i \equiv image\ (\sigma\ i)$

abbreviation $\vartheta\vartheta\ i \equiv image\ (\vartheta\ i)$

symcomp i j composes the symmetry maps from index i to index i+j-1.

primrec *symcomp* :: *nat* ⇒ *nat* ⇒ *'a* ⇒ *'a* (Σ) **where**
  $\Sigma\ i\ 0\ x = x$
  $\mid \Sigma\ i\ (Suc\ j)\ x = \sigma\ (i + j)\ (\Sigma\ i\ j\ x)$

inv-symcomp i j composes the inverse symmetries from i+j-1 to i.

primrec *inv-symcomp* :: *nat* ⇒ *nat* ⇒ *'a* ⇒ *'a* (Θ) **where**
  $\Theta\ i\ 0\ x = x$
  $\mid \Theta\ i\ (Suc\ j)\ x = \Theta\ i\ j\ (\vartheta\ (i + j)\ x)$

end

Next we define a class for cubical ω-categories.

class *cubical-omega-category* = *semi-cubical-omega-category* + *symmetry-ops* +
  assumes *sym-type*: $\sigma\sigma\ i\ (face\text{-}fix\ i) \subseteq face\text{-}fix\ (i + 1)$
  and *inv-sym-type*: $\vartheta\vartheta\ i\ (face\text{-}fix\ (i + 1)) \subseteq face\text{-}fix\ i$
  and *sym-inv-sym*: $fFx\ (i + 1)\ x \Longrightarrow \sigma\ i\ (\vartheta\ i\ x) = x$
  and *inv-sym-sym*: $fFx\ i\ x \Longrightarrow \vartheta\ i\ (\sigma\ i\ x) = x$
  and *sym-face1*: $fFx\ i\ x \Longrightarrow \partial\ i\ \alpha\ (\sigma\ i\ x) = \sigma\ i\ (\partial\ (i + 1)\ \alpha\ x)$
  and *sym-face2*: $i \neq j \Longrightarrow i \neq j + 1 \Longrightarrow fFx\ j\ x \Longrightarrow \partial\ i\ \alpha\ (\sigma\ j\ x) = \sigma\ j\ (\partial\ i\ \alpha$
$x)$
  and *sym-func*: $i \neq j \Longrightarrow fFx\ i\ x \Longrightarrow fFx\ i\ y \Longrightarrow DD\ j\ x\ y \Longrightarrow$
    $\sigma\ i\ (x \otimes_j y) = (if\ j = i + 1\ then\ \sigma\ i\ x \otimes_i \sigma\ i\ y\ else\ \sigma\ i\ x \otimes_j \sigma\ i\ y)$
  and *sym-fix*: $fFx\ i\ x \Longrightarrow fFx\ (i + 1)\ x \Longrightarrow \sigma\ i\ x = x$
  and *sym-sym-braid*: $diffSup\ i\ j\ 2 \Longrightarrow fFx\ i\ x \Longrightarrow fFx\ j\ x \Longrightarrow \sigma\ i\ (\sigma\ j\ x) = \sigma\ j$
$(\sigma\ i\ x)$

begin

First we prove variants of the axioms.

lemma *sym-type-var*: $fFx\ i\ x \Longrightarrow fFx\ (i + 1)\ (\sigma\ i\ x)$
  ⟨*proof*⟩

lemma *sym-type-var1* [*simp*]: $\partial\ (i + 1)\ \alpha\ (\sigma\ i\ (\partial\ i\ \alpha\ x)) = \sigma\ i\ (\partial\ i\ \alpha\ x)$
  ⟨*proof*⟩

lemma *sym-type-var2* [*simp*]: $\partial\ (i + 1)\ \alpha \circ \sigma\ i \circ \partial\ i\ \alpha = \sigma\ i \circ \partial\ i\ \alpha$
  ⟨*proof*⟩

lemma *sym-type-var-lift-var* [*simp*]: $\partial\partial\ (i + 1)\ \alpha\ (\sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ X)) = \sigma\sigma\ i\ (\partial\partial$
$i\ \alpha\ X)$
  ⟨*proof*⟩

lemma *sym-type-var-lift* [*simp*]:
  assumes $FFx\ i\ X$
  shows $\partial\partial\ (i + 1)\ \alpha\ (\sigma\sigma\ i\ X) = \sigma\sigma\ i\ X$

⟨*proof*⟩

lemma *inv-sym-type-var*: $fFx\ (i + 1)\ x \Longrightarrow fFx\ i\ (\vartheta\ i\ x)$
⟨*proof*⟩

lemma *inv-sym-type-var1* [*simp*]: $\partial\ i\ \alpha\ (\vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x)) = \vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x)$
⟨*proof*⟩

lemma *inv-sym-type-var2* [*simp*]: $\partial\ i\ \alpha \circ \vartheta\ i \circ \partial\ (i + 1)\ \alpha = \vartheta\ i \circ \partial\ (i + 1)\ \alpha$
⟨*proof*⟩

lemma *inv-sym-type-lift-var* [*simp*]: $\partial\partial\ i\ \alpha\ (\vartheta\vartheta\ i\ (\partial\partial\ (i + 1)\ \alpha\ X)) = \vartheta\vartheta\ i\ (\partial\partial\ (i + 1)\ \alpha\ X)$
⟨*proof*⟩

lemma *inv-sym-type-lift*:
assumes $FFx\ (i + 1)\ X$
shows $\partial\partial\ i\ \alpha\ (\vartheta\vartheta\ i\ X) = \vartheta\vartheta\ i\ X$
⟨*proof*⟩

lemma *sym-inv-sym-var1* [*simp*]: $\sigma\ i\ (\vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x)) = \partial\ (i + 1)\ \alpha\ x$
⟨*proof*⟩

lemma *sym-inv-sym-var2* [*simp*]: $\sigma\ i \circ \vartheta\ i \circ \partial\ (i + 1)\ \alpha = \partial\ (i + 1)\ \alpha$
⟨*proof*⟩

lemma *sym-inv-sym-lift-var*: $\sigma\sigma\ i\ (\vartheta\vartheta\ i\ (\partial\partial\ (i + 1)\ \alpha\ X)) = \partial\partial\ (i + 1)\ \alpha\ X$
⟨*proof*⟩

lemma *sym-inv-sym-lift*:
assumes $FFx\ (i + 1)\ X$
shows $\sigma\sigma\ i\ (\vartheta\vartheta\ i\ X) = X$
⟨*proof*⟩

lemma *inv-sym-sym-var1* [*simp*]: $\vartheta\ i\ (\sigma\ i\ (\partial\ i\ \alpha\ x)) = \partial\ i\ \alpha\ x$
⟨*proof*⟩

lemma *inv-sym-sym-var2* [*simp*]: $\vartheta\ i \circ \sigma\ i \circ \partial\ i\ \alpha = \partial\ i\ \alpha$
⟨*proof*⟩

lemma *inv-sym-sym-lift-var* [*simp*]: $\vartheta\vartheta\ i\ (\sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ X)) = \partial\partial\ i\ \alpha\ X$
⟨*proof*⟩

lemma *inv-sym-sym-lift*:
assumes $FFx\ i\ X$
shows $\vartheta\vartheta\ i\ (\sigma\sigma\ i\ X) = X$
⟨*proof*⟩

lemma *sym-fix-var1* [*simp*]: $\sigma\ i\ (\partial\ i\ \alpha\ (\partial\ (i + 1)\ \beta\ x)) = \partial\ i\ \alpha\ (\partial\ (i + 1)\ \beta\ x)$
⟨*proof*⟩

lemma *sym-fix-var2* [*simp*]: $\sigma\ i \circ \partial\ i\ \alpha \circ \partial\ (i + 1)\ \beta = \partial\ i\ \alpha \circ \partial\ (i + 1)\ \beta$
⟨*proof*⟩

lemma *sym-fix-lift-var*: $\sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ (\partial\partial\ (i + 1)\ \beta\ X)) = \partial\partial\ i\ \alpha\ (\partial\partial\ (i + 1)\ \beta\ X)$
⟨*proof*⟩

lemma *sym-fix-lift*:
assumes $FFx\ i\ X$
and $FFx\ (i + 1)\ X$
shows $\sigma\sigma\ i\ X = X$
⟨*proof*⟩

lemma *sym-face1-var1*: $\partial\ i\ \alpha\ (\sigma\ i\ (\partial\ i\ \beta\ x)) = \sigma\ i\ (\partial\ (i + 1)\ \alpha\ (\partial\ i\ \beta\ x))$
⟨*proof*⟩

lemma *sym-face1-var2*: $\partial\ i\ \alpha \circ \sigma\ i \circ \partial\ i\ \beta = \sigma\ i \circ \partial\ (i + 1)\ \alpha \circ \partial\ i\ \beta$
⟨*proof*⟩

lemma *sym-face1-lift-var*: $\partial\partial\ i\ \alpha\ (\sigma\sigma\ i\ (\partial\partial\ i\ \beta\ X)) = \sigma\sigma\ i\ (\partial\partial\ (i + 1)\ \alpha\ (\partial\partial\ i\ \beta\ X))$
⟨*proof*⟩

lemma *sym-face1-lift*:
assumes $FFx\ i\ X$
shows $\partial\partial\ i\ \alpha\ (\sigma\sigma\ i\ X) = \sigma\sigma\ i\ (\partial\partial\ (i + 1)\ \alpha\ X)$
⟨*proof*⟩

lemma *sym-face2-var1*:
assumes $i \neq j$
and $i \neq j + 1$
shows $\partial\ i\ \alpha\ (\sigma\ j\ (\partial\ j\ \beta\ x)) = \sigma\ j\ (\partial\ i\ \alpha\ (\partial\ j\ \beta\ x))$
⟨*proof*⟩

lemma *sym-face2-var2*:
assumes $i \neq j$
and $i \neq j + 1$
shows $\partial\ i\ \alpha \circ \sigma\ j \circ \partial\ j\ \beta = \sigma\ j \circ \partial\ i\ \alpha \circ \partial\ j\ \beta$
⟨*proof*⟩

lemma *sym-face2-lift-var*:
assumes $i \neq j$
and $i \neq j + 1$
shows $\partial\partial\ i\ \alpha\ (\sigma\sigma\ j\ (\partial\partial\ j\ \beta\ X)) = \sigma\sigma\ j\ (\partial\partial\ i\ \alpha\ (\partial\partial\ j\ \beta\ X))$
⟨*proof*⟩

lemma *sym-face2-lift*:
  assumes $i \neq j$
  and $i \neq j + 1$
  and *FFx j X*
  shows $\partial\partial\ i\ \alpha\ (\sigma\sigma\ j\ X) = \sigma\sigma\ j\ (\partial\partial\ i\ \alpha\ X)$
  ⟨*proof*⟩

lemma *sym-sym-braid-var1*:
  assumes *diffSup i j 2*
  shows $\sigma\ i\ (\sigma\ j\ (\partial\ i\ \alpha\ (\partial\ j\ \beta\ x))) = \sigma\ j\ (\sigma\ i\ (\partial\ i\ \alpha\ (\partial\ j\ \beta\ x)))$
  ⟨*proof*⟩

lemma *sym-sym-braid-var2*:
  assumes *diffSup i j 2*
  shows $\sigma\ i \circ \sigma\ j \circ \partial\ i\ \alpha \circ \partial\ j\ \beta = \sigma\ j \circ \sigma\ i \circ \partial\ i\ \alpha \circ \partial\ j\ \beta$
  ⟨*proof*⟩

lemma *sym-sym-braid-lift-var*:
  assumes *diffSup i j 2*
  shows $\sigma\sigma\ i\ (\sigma\sigma\ j\ (\partial\partial\ i\ \alpha\ (\partial\partial\ j\ \beta\ X))) = \sigma\sigma\ j\ (\sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ (\partial\partial\ j\ \beta\ X)))$
⟨*proof*⟩

lemma *sym-sym-braid-lift*:
  assumes *diffSup i j 2*
  and *FFx i X*
  and *FFx j X*
  shows $\sigma\sigma\ i\ (\sigma\sigma\ j\ X) = \sigma\sigma\ j\ (\sigma\sigma\ i\ X)$
  ⟨*proof*⟩

lemma *sym-func2*:
  assumes *fFx i x*
  and *fFx i y*
  and *DD (i + 1) x y*
  shows $\sigma\ i\ (x \otimes_{(i\ +\ 1)}\ y) = \sigma\ i\ x \otimes_{i}\ \sigma\ i\ y$
  ⟨*proof*⟩

lemma *sym-func3*:
  assumes $i \neq j$
  and $j \neq i + 1$
  and *fFx i x*
  and *fFx i y*
  and *DD j x y*
  shows $\sigma\ i\ (x \otimes_{j}\ y) = \sigma\ i\ x \otimes_{j}\ \sigma\ i\ y$
  ⟨*proof*⟩

lemma *sym-func2-var1*:
  assumes $DD\ (i + 1)\ (\partial\ i\ \alpha\ x)\ (\partial\ i\ \beta\ y)$
  shows $\sigma\ i\ (\partial\ i\ \alpha\ x \otimes_{(i\ +\ 1)}\ \partial\ i\ \beta\ y) = \sigma\ i\ (\partial\ i\ \alpha\ x) \otimes_{i}\ \sigma\ i\ (\partial\ i\ \beta\ y)$
  ⟨*proof*⟩

lemma *sym-func3-var1*:
  assumes $i \neq j$
  and $j \neq i + 1$
  and $DD\ j\ (\partial\ i\ \alpha\ x)\ (\partial\ i\ \beta\ y)$
  shows $\sigma\ i\ (\partial\ i\ \alpha\ x \otimes_j \partial\ i\ \beta\ y) = \sigma\ i\ (\partial\ i\ \alpha\ x) \otimes_j \sigma\ i\ (\partial\ i\ \beta\ y)$
  ⟨*proof*⟩

lemma *sym-func2-DD*:
  assumes $fFx\ i\ x$
  and $fFx\ i\ y$
  shows $DD\ (i + 1)\ x\ y = DD\ i\ (\sigma\ i\ x)\ (\sigma\ i\ y)$
  ⟨*proof*⟩

lemma *func2-var2*: $\sigma\sigma\ i\ (\partial\ i\ \alpha\ x \odot_{(i + 1)} \partial\ i\ \beta\ y) = \sigma\ i\ (\partial\ i\ \alpha\ x) \odot_i \sigma\ i\ (\partial\ i\ \beta\ y)$
⟨*proof*⟩

lemma *sym-func2-lift-var1*: $\sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ X \star_{(i + 1)} \partial\partial\ i\ \beta\ Y) = \sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ X) \star_i \sigma\sigma\ i\ (\partial\partial\ i\ \beta\ Y)$
⟨*proof*⟩

lemma *sym-func2-lift*:
  assumes $FFx\ i\ X$
  and $FFx\ i\ Y$
  shows $\sigma\sigma\ i\ (X \star_{(i + 1)} Y) = \sigma\sigma\ i\ X \star_i \sigma\sigma\ i\ Y$
⟨*proof*⟩

lemma *func3-var1*:
  assumes $i \neq j$
  and $j \neq i + 1$
  shows $\sigma\sigma\ i\ (\partial\ i\ \alpha\ x \odot_j \partial\ i\ \beta\ y) = \sigma\ i\ (\partial\ i\ \alpha\ x) \odot_j \sigma\ i\ (\partial\ i\ \beta\ y)$
⟨*proof*⟩

lemma *sym-func3-lift-var1*:
  assumes $i \neq j$
  and $j \neq i + 1$
  shows $\sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ X \star_j \partial\partial\ i\ \beta\ Y) = \sigma\sigma\ i\ (\partial\partial\ i\ \alpha\ X) \star_j \sigma\sigma\ i\ (\partial\partial\ i\ \beta\ Y)$
⟨*proof*⟩

lemma *sym-func3-lift*:
  assumes $i \neq j$
  and $j \neq i + 1$
  and $FFx\ i\ X$
  and $FFx\ i\ Y$
  shows $\sigma\sigma\ i\ (X \star_j Y) = \sigma\sigma\ i\ X \star_j \sigma\sigma\ i\ Y$
⟨*proof*⟩

lemma *sym-func3-var2*: $i \neq j \Longrightarrow \sigma\sigma\ i\ (\partial\ i\ \alpha\ x \odot_j \partial\ i\ \beta\ y) = (\textit{if}\ j = i + 1\ \textit{then}$

$\sigma\ i\ (\partial\ i\ \alpha\ x) \odot_i \sigma\ i\ (\partial\ i\ \beta\ y)$ *else* $\sigma\ i\ (\partial\ i\ \alpha\ x) \odot_j \sigma\ i\ (\partial\ i\ \beta\ y))$
⟨*proof*⟩

Symmetries and inverse symmetries form a bijective pair on suitable fix-points of the face maps.

lemma *sym-inj*: *inj-on* $(\sigma\ i)$ (*face-fix* i)
⟨*proof*⟩

lemma *sym-inj-var*:
assumes *fFx* $i\ x$
and *fFx* $i\ y$
and $\sigma\ i\ x = \sigma\ i\ y$
shows $x = y$
⟨*proof*⟩

lemma *inv-sym-inj*: *inj-on* $(\vartheta\ i)$ (*face-fix* $(i + 1)$)
⟨*proof*⟩

lemma *inv-sym-inj-var*:
assumes *fFx* $(i + 1)\ x$
and *fFx* $(i + 1)\ y$
and $\vartheta\ i\ x = \vartheta\ i\ y$
shows $x = y$
⟨*proof*⟩

lemma *surj-sym*: *image* $(\sigma\ i)$ (*face-fix* i) = *face-fix* $(i + 1)$
⟨*proof*⟩

lemma *surj-inv-sym*: *image* $(\vartheta\ i)$ (*face-fix* $(i + 1)$) = *face-fix* i
⟨*proof*⟩

lemma *sym-adj*:
assumes *fFx* $i\ x$
and *fFx* $(i + 1)\ y$
shows $(\sigma\ i\ x = y) = (x = \vartheta\ i\ y)$
⟨*proof*⟩

Next we list properties for inverse symmetries corresponding to the axioms.

lemma *inv-sym*:
assumes *fFx* $i\ x$
and *fFx* $(i + 1)\ x$
shows $\vartheta\ i\ x = x$
⟨*proof*⟩

lemma *inv-sym-face2*:
assumes $i \neq j$
and $i \neq j + 1$
and *fFx* $(j + 1)\ x$
shows $\partial\ i\ \alpha\ (\vartheta\ j\ x) = \vartheta\ j\ (\partial\ i\ \alpha\ x)$

⟨*proof*⟩

lemma *sym-braid*:
assumes *fFx i x*
and *fFx* $(i + 1)$ *x*
shows $\sigma\ i\ (\sigma\ (i + 1)\ (\sigma\ i\ x)) = \sigma\ (i + 1)\ (\sigma\ i\ (\sigma\ (i + 1)\ x))$
⟨*proof*⟩

lemma *inv-sym-braid*:
assumes *fFx* $(i + 1)$ *x*
and *fFx* $(i + 2)$ *x*
shows $\vartheta\ i\ (\vartheta\ (i + 1)\ (\vartheta\ i\ x)) = \vartheta\ (i + 1)\ (\vartheta\ i\ (\vartheta\ (i + 1)\ x))$
⟨*proof*⟩

lemma *sym-inv-sym-braid*:
assumes *diffSup i j 2*
and *fFx* $(j + 1)$ *x*
and *fFx i x*
shows $\sigma\ i\ (\vartheta\ j\ x) = \vartheta\ j\ (\sigma\ i\ x)$
⟨*proof*⟩

lemma *sym-func1*:
assumes *fFx i x*
and *fFx i y*
and *DD i x y*
shows $\sigma\ i\ (x \otimes_i y) = \sigma\ i\ x \otimes_{(i + 1)} \sigma\ i\ y$
⟨*proof*⟩

lemma *sym-func1-var1*: $\sigma\sigma\ i\ (\partial\ i\ \alpha\ x \odot_i \partial\ i\ \beta\ y) = \sigma\ i\ (\partial\ i\ \alpha\ x) \odot_{(i + 1)} \sigma\ i\ (\partial\ i\ \beta\ y)$
⟨*proof*⟩

lemma *inv-sym-func2-var1*: $\vartheta\vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x \odot_i \partial\ (i + 1)\ \beta\ y) = \vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x) \odot_{(i + 1)} \vartheta\ i\ (\partial\ (i + 1)\ \beta\ y)$
⟨*proof*⟩

lemma *inv-sym-func3-var1*: $\vartheta\vartheta\ i\ ((\partial\ (i + 1)\ \alpha\ x) \odot_{(i + 1)} (\partial\ (i + 1)\ \beta\ y)) = \vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x) \odot_i \vartheta\ i\ (\partial\ (i + 1)\ \beta\ y)$
⟨*proof*⟩

lemma *inv-sym-func-var1*:
assumes $i \neq j$
and $j \neq i + 1$
shows $\vartheta\vartheta\ i\ ((\partial\ (i + 1)\ \alpha\ x) \odot_j (\partial\ (i + 1)\ \beta\ y)) = \vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x) \odot_j \vartheta\ i\ (\partial\ (i + 1)\ \beta\ y)$
⟨*proof*⟩

lemma *inv-sym-func2*:
assumes *fFx* $(i + 1)$ *x*

and *fFx* $(i + 1)$ *y*
and *DD i x y*
shows $\vartheta\ i\ (x \otimes_i y) = \vartheta\ i\ x \otimes_{(i + 1)} \vartheta\ i\ y$
⟨*proof*⟩

lemma *inv-sym-func3*:
assumes *fFx* $(i + 1)$ *x*
and *fFx* $(i + 1)$ *y*
and *DD* $(i + 1)$ *x y*
shows $\vartheta\ i\ (x \otimes_{(i + 1)} y) = \vartheta\ i\ x \otimes_i \vartheta\ i\ y$
⟨*proof*⟩

lemma *inv-sym-func*:
assumes $i \neq j$
and $j \neq i + 1$
and *fFx* $(i + 1)$ *x*
and *fFx* $(i + 1)$ *y*
and *DD j x y*
shows $\vartheta\ i\ (x \otimes_j y) = \vartheta\ i\ x \otimes_j \vartheta\ i\ y$
⟨*proof*⟩

The following properties are related to faces and braids.

lemma *sym-face3*:
assumes *fFx i x*
shows $\partial\ (i + 1)\ \alpha\ (\sigma\ i\ x) = \sigma\ i\ (\partial\ i\ \alpha\ x)$
⟨*proof*⟩

lemma *sym-face3-var1*: $\partial\ (i + 1)\ \alpha\ (\sigma\ i\ (\partial\ i\ \beta\ x)) = \sigma\ i\ (\partial\ i\ \alpha\ (\partial\ i\ \beta\ x))$
⟨*proof*⟩

lemma *sym-face3-simp* [*simp*]:
assumes *fFx i x*
shows $\partial\ (i + 1)\ \alpha\ (\sigma\ i\ x) = \sigma\ i\ x$
⟨*proof*⟩

lemma *sym-face3-simp-var1* [*simp*]: $\partial\ (i + 1)\ \alpha\ (\sigma\ i\ (\partial\ i\ \beta\ x)) = \sigma\ i\ (\partial\ i\ \beta\ x)$
⟨*proof*⟩

lemma *inv-sym-face3*:
assumes *fFx* $(i + 1)$ *x*
shows $\partial\ i\ \alpha\ (\vartheta\ i\ x) = \vartheta\ i\ (\partial\ (i + 1)\ \alpha\ x)$
⟨*proof*⟩

lemma *inv-sym-face3-var1*: $\partial\ i\ \alpha\ (\vartheta\ i\ (\partial\ (i + 1)\ \beta\ x)) = \vartheta\ i\ (\partial\ (i + 1)\ \alpha\ (\partial\ (i + 1)\ \beta\ x))$
⟨*proof*⟩

lemma *inv-sym-face3-simp*:
assumes *fFx* $(i + 1)$ *x*

shows $\partial\ i\ \alpha\ (\vartheta\ i\ x) = \vartheta\ i\ x$
⟨*proof*⟩

lemma *inv-sym-face3-simp-var1* [*simp*]: $\partial\ i\ \alpha\ (\vartheta\ i\ (\partial\ (i + 1)\ \beta\ x)) = \vartheta\ i\ (\partial\ (i + 1)\ \beta\ x)$
⟨*proof*⟩

lemma *inv-sym-face1*:
assumes *fFx* $(i + 1)\ x$
shows $\partial\ (i + 1)\ \alpha\ (\vartheta\ i\ x) = \vartheta\ i\ (\partial\ i\ \alpha\ x)$
⟨*proof*⟩

lemma *inv-sym-face1-var1*: $\partial\ (i + 1)\ \alpha\ (\vartheta\ i\ (\partial\ (i + 1)\ \beta\ x)) = \vartheta\ i\ (\partial\ i\ \alpha\ (\partial\ (i + 1)\ \beta\ x))$
⟨*proof*⟩

lemma *inv-sym-sym-braid*:
assumes *diffSup i j 2*
and *fFx j x*
and *fFx* $(i + 1)\ x$
shows $\vartheta\ i\ (\sigma\ j\ x) = \sigma\ j\ (\vartheta\ i\ x)$
⟨*proof*⟩

lemma *inv-sym-sym-braid-var1*: *diffSup i j 2* $\Longrightarrow \vartheta\ i\ (\sigma\ j\ (\partial\ (i + 1)\ \alpha\ (\partial\ j\ \beta\ x))) = \sigma\ j\ (\vartheta\ i\ (\partial\ (i + 1)\ \alpha\ (\partial\ j\ \beta\ x)))$
⟨*proof*⟩

lemma *inv-sym-inv-sym-braid*:
assumes *diffSup i j 2*
and *fFx* $(i + 1)\ x$
and *fFx* $(j + 1)\ x$
shows $\vartheta\ i\ (\vartheta\ j\ x) = \vartheta\ j\ (\vartheta\ i\ x)$
⟨*proof*⟩

lemma *inv-sym-inv-sym-braid-var1*: *diffSup i j 2* $\Longrightarrow \vartheta\ i\ (\vartheta\ j\ (\partial\ (i + 1)\ \alpha\ (\partial\ (j + 1)\ \beta\ x))) = \vartheta\ j\ (\vartheta\ i\ (\partial\ (i + 1)\ \alpha\ (\partial\ (j + 1)\ \beta\ x)))$
⟨*proof*⟩

The following properties are related to symcomp and inv-symcomp.

lemma *symcomp-type-var*:
assumes *fFx i x*
shows *fFx* $(i + j)\ (\Sigma\ i\ j\ x)$ ⟨*proof*⟩

lemma *symcomp-type*: *image* $(\Sigma\ i\ j)$ (*face-fix i*) $\subseteq$ *face-fix* $(i + j)$
⟨*proof*⟩

lemma *symcomp-type-var1* [*simp*]: $\partial\ (i + j)\ \alpha\ (\Sigma\ i\ j\ (\partial\ i\ \beta\ x)) = \Sigma\ i\ j\ (\partial\ i\ \beta\ x)$
⟨*proof*⟩

lemma *inv-symcomp-type-var*:
 assumes *fFx* $(i + j)$ x
 shows *fFx* i $(\Theta\ i\ j\ x)$ ⟨*proof*⟩

lemma *inv-symcomp-type*: *image* $(\Theta\ i\ j)$ (*face-fix* $(i + j)$) $\subseteq$ *face-fix* i
 ⟨*proof*⟩

lemma *inv-symcomp-type-var1* [*simp*]: $\partial\ i\ \alpha\ (\Theta\ i\ j\ (\partial\ (i + j)\ \beta\ x)) = \Theta\ i\ j\ (\partial\ (i + j)\ \beta\ x)$
 ⟨*proof*⟩

lemma *symcomp-inv-symcomp*:
 assumes *fFx* $(i + j)$ x
 shows $\Sigma\ i\ j\ (\Theta\ i\ j\ x) = x$ ⟨*proof*⟩

lemma *inv-symcomp-symcomp*:
 assumes *fFx* i x
 shows $\Theta\ i\ j\ (\Sigma\ i\ j\ x) = x$ ⟨*proof*⟩

lemma *symcomp-adj*:
 assumes *fFx* i x
 and *fFx* $(i + j)$ y
 shows $(\Sigma\ i\ j\ x = y) = (x = \Theta\ i\ j\ y)$
 ⟨*proof*⟩

lemma *decomp-symcomp1*:
 assumes $k \leq j$
 and *fFx* i x
 shows $\Sigma\ i\ j\ x = \Sigma\ (i + k)\ (j - k)\ (\Sigma\ i\ k\ x)$ ⟨*proof*⟩

lemma *decomp-symcomp2*:
 assumes $1 \leq k$
 and $k \leq j$
 and *fFx* i x
 shows $\Sigma\ i\ j\ x = \Sigma\ (i + k)\ (j - k)\ (\sigma\ (i + k - 1)\ (\Sigma\ i\ (k - 1)\ x))$
 ⟨*proof*⟩

lemma *decomp-symcomp3*:
 assumes $i \leq l$
 and $l + 1 \leq i + j$
 and *fFx* i x
 shows $\Sigma\ i\ j\ x = \Sigma\ (l + 1)\ (i + j - l - 1)\ (\sigma\ l\ (\Sigma\ i\ (l - i)\ x))$
 ⟨*proof*⟩

lemma *symcomp-face2*:
 assumes $l < i \vee i + j < l$
 and *fFx* i x
 shows $\partial\ l\ \alpha\ (\Sigma\ i\ j\ x) = \Sigma\ i\ j\ (\partial\ l\ \alpha\ x)$ ⟨*proof*⟩

lemma *symcomp-face3*: $fFx\ i\ x \Longrightarrow \partial\ (i + j)\ \alpha\ (\Sigma\ i\ j\ x) = \Sigma\ i\ j\ (\partial\ i\ \alpha\ x)$
⟨*proof*⟩

lemma *symcomp-face1*:
assumes $i \leq l$
and $l < i + j$
and $fFx\ i\ x$
shows $\partial\ l\ \alpha\ (\Sigma\ i\ j\ x) = \Sigma\ i\ j\ (\partial\ (l + 1)\ \alpha\ x)$
⟨*proof*⟩

lemma *inv-symcomp-face2*:
assumes $l < i \vee i + j < l$
and $fFx\ (i + j)\ x$
shows $\partial\ l\ \alpha\ (\Theta\ i\ j\ x) = \Theta\ i\ j\ (\partial\ l\ \alpha\ x)$ ⟨*proof*⟩

lemma *inv-symcomp-face3*: $fFx\ (i + j)\ x \Longrightarrow \partial\ i\ \alpha\ (\Theta\ i\ j\ x) = \Theta\ i\ j\ (\partial\ (i + j)\ \alpha\ x)$
⟨*proof*⟩

lemma *inv-symcomp-face1*:
assumes $i < l$
and $l \leq i + j$
and $fFx\ (i + j)\ x$
shows $\partial\ l\ \alpha\ (\Theta\ i\ j\ x) = \Theta\ i\ j\ (\partial\ (l - 1)\ \alpha\ x)$
⟨*proof*⟩

lemma *symcomp-comp1*:
assumes $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ i\ x\ y$
shows $\Sigma\ i\ j\ (x \otimes_i y) = \Sigma\ i\ j\ x \otimes_{(i + j)} \Sigma\ i\ j\ y$
⟨*proof*⟩

lemma *symcomp-comp2*:
assumes $k < i$
and $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ k\ x\ y$
shows $\Sigma\ i\ j\ (x \otimes_k y) = \Sigma\ i\ j\ x \otimes_k \Sigma\ i\ j\ y$
⟨*proof*⟩

lemma *symcomp-comp3*:
assumes $i + j < k$
and $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ k\ x\ y$
shows $\Sigma\ i\ j\ (x \otimes_k y) = \Sigma\ i\ j\ x \otimes_k \Sigma\ i\ j\ y$ ⟨*proof*⟩

lemma *fix-comp*:

assumes $i \neq j$
and $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ j\ x\ y$
shows $fFx\ i\ (x \otimes_j y)$
⟨*proof*⟩

lemma *symcomp-comp4*:
assumes $i < k$
and $k \leq i + j$
and $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ k\ x\ y$
shows $\Sigma\ i\ j\ (x \otimes_k y) = \Sigma\ i\ j\ x \otimes_{(k\ -\ 1)} \Sigma\ i\ j\ y$
⟨*proof*⟩

lemma *symcomp-comp*:
assumes $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ k\ x\ y$
shows $\Sigma\ i\ j\ (x \otimes_k y) = (\mathit{if}\ k = i\ \mathit{then}\ \Sigma\ i\ j\ x \otimes_{(i\ +\ j)} \Sigma\ i\ j\ y$
$\mathit{else}\ (\mathit{if}\ (i < k \wedge k \leq i + j)\ \mathit{then}\ \Sigma\ i\ j\ x \otimes_{(k\ -\ 1)} \Sigma\ i\ j\ y$
$\mathit{else}\ \Sigma\ i\ j\ x \otimes_k \Sigma\ i\ j\ y))$
⟨*proof*⟩

lemma *inv-symcomp-comp1*:
assumes $fFx\ (i + j)\ x$
and $fFx\ (i + j)\ y$
and $DD\ (i + j)\ x\ y$
shows $\Theta\ i\ j\ (x \otimes_{(i\ +\ j)} y) = \Theta\ i\ j\ x \otimes_i \Theta\ i\ j\ y$
⟨*proof*⟩

lemma *inv-symcomp-comp2*:
assumes $k < i$
and $fFx\ (i + j)\ x$
and $fFx\ (i + j)\ y$
and $DD\ k\ x\ y$
shows $\Theta\ i\ j\ (x \otimes_k y) = \Theta\ i\ j\ x \otimes_k \Theta\ i\ j\ y$
⟨*proof*⟩

lemma *inv-symcomp-comp3*:
assumes $i + j < k$
and $fFx\ (i + j)\ x$
and $fFx\ (i + j)\ y$
and $DD\ k\ x\ y$
shows $\Theta\ i\ j\ (x \otimes_k y) = \Theta\ i\ j\ x \otimes_k \Theta\ i\ j\ y$
⟨*proof*⟩

lemma *inv-symcomp-comp4*:

assumes $i \leq k$
and $k < i + j$
and $fFx\ (i + j)\ x$
and $fFx\ (i + j)\ y$
and $DD\ k\ x\ y$
shows $\Theta\ i\ j\ (x \otimes_k y) = \Theta\ i\ j\ x \otimes_{(k\ +\ 1)} \Theta\ i\ j\ y$
⟨*proof*⟩

end

end

4 Cubical Categories with Connections

theory *CubicalCategoriesConnections*
imports *CubicalCategories*

begin

All categories considered in this component are single-set categories.

class *connection-ops* =
fixes *connection* :: $nat \Rightarrow bool \Rightarrow 'a \Rightarrow 'a$ (Γ)

abbreviation (**in** *connection-ops*) $\Gamma\Gamma\ i\ \alpha \equiv image\ (\Gamma\ i\ \alpha)$

We define a class for cubical ω-categories with connections.

class *cubical-omega-category-connections* = *cubical-omega-category* + *connection-ops* +
assumes *conn-face1*: $fFx\ j\ x \Longrightarrow \partial\ j\ \alpha\ (\Gamma\ j\ \alpha\ x) = x$
and *conn-face2*: $fFx\ j\ x \Longrightarrow \partial\ (j + 1)\ \alpha\ (\Gamma\ j\ \alpha\ x) = \sigma\ j\ x$
and *conn-face3*: $i \neq j \Longrightarrow i \neq j + 1 \Longrightarrow fFx\ j\ x \Longrightarrow \partial\ i\ \alpha\ (\Gamma\ j\ \beta\ x) = \Gamma\ j\ \beta\ (\partial\ i\ \alpha\ x)$
and *conn-corner1*: $fFx\ i\ x \Longrightarrow fFx\ i\ y \Longrightarrow DD\ (i + 1)\ x\ y \Longrightarrow \Gamma\ i\ tt\ (x \otimes_{(i\ +\ 1)} y) = (\Gamma\ i\ tt\ x \otimes_{(i\ +\ 1)} \sigma\ i\ x) \otimes_i (x \otimes_{(i\ +\ 1)} \Gamma\ i\ tt\ y)$
and *conn-corner2*: $fFx\ i\ x \Longrightarrow fFx\ i\ y \Longrightarrow DD\ (i + 1)\ x\ y \Longrightarrow \Gamma\ i\ ff\ (x \otimes_{(i\ +\ 1)} y) = (\Gamma\ i\ ff\ x \otimes_{(i\ +\ 1)} y) \otimes_i (\sigma\ i\ y \otimes_{(i\ +\ 1)} \Gamma\ i\ ff\ y)$
and *conn-corner3*: $j \neq i \wedge j \neq i + 1 \Longrightarrow fFx\ i\ x \Longrightarrow fFx\ i\ y \Longrightarrow DD\ j\ x\ y \Longrightarrow \Gamma\ i\ \alpha\ (x \otimes_j y) = \Gamma\ i\ \alpha\ x \otimes_j \Gamma\ i\ \alpha\ y$
and *conn-fix*: $fFx\ i\ x \Longrightarrow fFx\ (i + 1)\ x \Longrightarrow \Gamma\ i\ \alpha\ x = x$
and *conn-zigzag1*: $fFx\ i\ x \Longrightarrow \Gamma\ i\ tt\ x \otimes_{(i\ +\ 1)} \Gamma\ i\ ff\ x = x$
and *conn-zigzag2*: $fFx\ i\ x \Longrightarrow \Gamma\ i\ tt\ x \otimes_i \Gamma\ i\ ff\ x = \sigma\ i\ x$
and *conn-conn-braid*: $diffSup\ i\ j\ 2 \Longrightarrow fFx\ j\ x \Longrightarrow fFx\ i\ x \Longrightarrow \Gamma\ i\ \alpha\ (\Gamma\ j\ \beta\ x) = \Gamma\ j\ \beta\ (\Gamma\ i\ \alpha\ x)$
and *conn-shift*: $fFx\ i\ x \Longrightarrow fFx\ (i + 1)\ x \Longrightarrow \sigma\ (i + 1)\ (\sigma\ i\ (\Gamma\ (i + 1)\ \alpha\ x)) = \Gamma\ i\ \alpha\ (\sigma\ (i + 1)\ x)$

begin

lemma *conn-face4*: $fFx\ j\ x \Longrightarrow \partial\ j\ \alpha\ (\Gamma\ j\ (\neg\alpha)\ x) = \partial\ (j + 1)\ \alpha\ x$
⟨*proof*⟩

lemma *conn-face1-lift*: $FFx\ j\ X \Longrightarrow \partial\partial\ j\ \alpha\ (\Gamma\Gamma\ j\ \alpha\ X) = X$
⟨*proof*⟩

lemma *conn-face4-lift*: $FFx\ j\ X \Longrightarrow \partial\partial\ j\ \alpha\ (\Gamma\Gamma\ j\ (\neg\alpha)\ X) = \partial\partial\ (j + 1)\ \alpha\ X$
⟨*proof*⟩

lemma *conn-face2-lift*: $FFx\ j\ X \Longrightarrow \partial\partial\ (j + 1)\ \alpha\ (\Gamma\Gamma\ j\ \alpha\ X) = \sigma\sigma\ j\ X$
⟨*proof*⟩

lemma *conn-face3-lift*: $i \neq j \Longrightarrow i \neq j + 1 \Longrightarrow FFx\ j\ X \Longrightarrow \partial\partial\ i\ \alpha\ (\Gamma\Gamma\ j\ \beta\ X) = \Gamma\Gamma\ j\ \beta\ (\partial\partial\ i\ \alpha\ X)$
⟨*proof*⟩

lemma *conn-fix-lift*: $FFx\ i\ X \Longrightarrow FFx\ (i + 1)\ X \Longrightarrow \Gamma\Gamma\ i\ \alpha\ X = X$
⟨*proof*⟩

lemma *conn-conn-braid-lift*:
assumes *diffSup i j 2*
and *FFx j X*
and *FFx i X*
shows $\Gamma\Gamma\ i\ \alpha\ (\Gamma\Gamma\ j\ \beta\ X) = \Gamma\Gamma\ j\ \beta\ (\Gamma\Gamma\ i\ \alpha\ X)$
⟨*proof*⟩

lemma *conn-sym-braid*:
assumes *diffSup i j 2*
and *fFx i x*
and *fFx j x*
shows $\Gamma\ i\ \alpha\ (\sigma\ j\ x) = \sigma\ j\ (\Gamma\ i\ \alpha\ x)$
⟨*proof*⟩

lemma *conn-zigzag1-var* [*simp*]: $\Gamma\ i\ tt\ (\partial\ i\ \alpha\ x) \odot_{(i + 1)} \Gamma\ i\ ff\ (\partial\ i\ \alpha\ x) = \{\partial\ i\ \alpha\ x\}$
⟨*proof*⟩

lemma *conn-zigzag1-lift*:
assumes *FFx i X*
shows $\Gamma\Gamma\ i\ tt\ X \star_{(i + 1)} \Gamma\Gamma\ i\ ff\ X = X$
⟨*proof*⟩

lemma *conn-zigzag2-var*: $\Gamma\ i\ tt\ (\partial\ i\ \alpha\ x) \odot_i \Gamma\ i\ ff\ (\partial\ i\ \alpha\ x) = \{\sigma\ i\ (\partial\ i\ \alpha\ x)\}$
⟨*proof*⟩

lemma *conn-zigzag2-lift*:
assumes *FFx i X*
shows $\Gamma\Gamma\ i\ tt\ X \star_i \Gamma\Gamma\ i\ ff\ X = \sigma\sigma\ i\ X$

$\langle proof \rangle$

lemma *conn-sym-braid-lift*: $diffSup\ i\ j\ 2 \Longrightarrow FFx\ i\ X \Longrightarrow FFx\ j\ X \Longrightarrow \Gamma\Gamma\ i\ \alpha\ (\sigma\sigma\ j\ X) = \sigma\sigma\ j\ (\Gamma\Gamma\ i\ \alpha\ X)$
$\langle proof \rangle$

lemma *conn-corner1-DD*:
assumes $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ (i + 1)\ x\ y$
shows $DD\ i\ (\Gamma\ i\ tt\ x \otimes_{(i + 1)} \sigma\ i\ x)\ (x \otimes_{(i + 1)} \Gamma\ i\ tt\ y)$
$\langle proof \rangle$

lemma *conn-corner1-var*: $\Gamma\Gamma\ i\ tt\ (\partial\ i\ \alpha\ x \odot_{(i + 1)} \partial\ i\ \beta\ y) = (\Gamma\ i\ tt\ (\partial\ i\ \alpha\ x) \odot_{(i + 1)} \sigma\ i\ (\partial\ i\ \alpha\ x)) \star_i (\partial\ i\ \alpha\ x \odot_{(i + 1)} \Gamma\ i\ tt\ (\partial\ i\ \beta\ y))$
$\langle proof \rangle$

lemma *conn-corner1-lift-aux*: $fFx\ i\ x \Longrightarrow \partial\ (i + 1)\ ff\ (\Gamma\ i\ tt\ x) = \partial\ (i + 1)\ ff\ x$
$\langle proof \rangle$

lemma *conn-corner1-lift*:
assumes $FFx\ i\ X$
and $FFx\ i\ Y$
shows $\Gamma\Gamma\ i\ tt\ (X \star_{(i + 1)} Y) = (\Gamma\Gamma\ i\ tt\ X \star_{(i + 1)} \sigma\sigma\ i\ X) \star_i (X \star_{(i + 1)} \Gamma\Gamma\ i\ tt\ Y)$
$\langle proof \rangle$

lemma *conn-corner2-DD*:
assumes $fFx\ i\ x$
and $fFx\ i\ y$
and $DD\ (i + 1)\ x\ y$
shows $DD\ i\ (\Gamma\ i\ ff\ x \otimes_{(i + 1)} y)\ (\sigma\ i\ y \otimes_{(i + 1)} \Gamma\ i\ ff\ y)$
$\langle proof \rangle$

lemma *conn-corner2-var*: $\Gamma\Gamma\ i\ ff\ (\partial\ i\ \alpha\ x \odot_{(i + 1)} \partial\ i\ \beta\ y) = (\Gamma\ i\ ff\ (\partial\ i\ \alpha\ x) \odot_{(i + 1)} \partial\ i\ \beta\ y) \star_i (\sigma\ i\ (\partial\ i\ \beta\ y) \odot_{(i + 1)} \Gamma\ i\ ff\ (\partial\ i\ \beta\ y))$
$\langle proof \rangle$

lemma *conn-corner2-lift*:
assumes $FFx\ i\ X$
and $FFx\ i\ Y$
shows $\Gamma\Gamma\ i\ ff\ (X \star_{(i + 1)} Y) = (\Gamma\Gamma\ i\ ff\ X \star_{(i + 1)} Y) \star_i (\sigma\sigma\ i\ Y \star_{(i + 1)} \Gamma\Gamma\ i\ ff\ Y)$
$\langle proof \rangle$

lemma *conn-corner3-var*:
assumes $j \neq i \wedge j \neq i + 1$
shows $\Gamma\Gamma\ i\ \alpha\ (\partial\ i\ \beta\ x \odot_j \partial\ i\ \gamma\ y) = \Gamma\ i\ \alpha\ (\partial\ i\ \beta\ x) \odot_j \Gamma\ i\ \alpha\ (\partial\ i\ \gamma\ y)$

⟨*proof*⟩

lemma *conn-corner3-lift*:
assumes $j \neq i$
and $j \neq i + 1$
and *FFx i X*
and *FFx i Y*
shows $\Gamma\Gamma\ i\ \alpha\ (X \star_j Y) = \Gamma\Gamma\ i\ \alpha\ X \star_j \Gamma\Gamma\ i\ \alpha\ Y$
⟨*proof*⟩

lemma *conn-face5* [*simp*]: $\partial\ (j + 1)\ \alpha\ (\Gamma\ j\ (\neg\alpha)\ (\partial\ j\ \gamma\ x)) = \partial\ (j + 1)\ \alpha\ (\partial\ j\ \gamma\ x)$
⟨*proof*⟩

lemma *conn-inv-sym-braid*:
assumes *diffSup i j 2*
shows $\Gamma\ i\ \alpha\ (\vartheta\ j\ (\partial\ i\ \beta\ (\partial\ (j + 1)\ \gamma\ x))) = \vartheta\ j\ (\Gamma\ i\ \alpha\ (\partial\ i\ \beta\ (\partial\ (j + 1)\ \gamma\ x)))$
⟨*proof*⟩

lemma *conn-corner4*: $\Gamma\Gamma\ i\ tt\ (\partial\ i\ \alpha\ x \odot_{(i + 1)} \partial\ i\ \beta\ y) = (\Gamma\ i\ tt\ (\partial\ i\ \alpha\ x) \odot_i \partial\ i\ \alpha\ x) \star_{(i + 1)} (\sigma\ i\ (\partial\ i\ \alpha\ x) \odot_i \Gamma\ i\ tt\ (\partial\ i\ \beta\ y))$
⟨*proof*⟩

lemma *conn-corner5*: $\Gamma\Gamma\ i\ ff\ (\partial\ i\ \alpha\ x \odot_{(i + 1)} \partial\ i\ \beta\ y) = (\Gamma\ i\ ff\ (\partial\ i\ \alpha\ x) \odot_i \sigma\ i\ (\partial\ i\ \beta\ y)) \star_{(i + 1)} (\partial\ i\ \beta\ y \odot_i \Gamma\ i\ ff\ (\partial\ i\ \beta\ y))$
⟨*proof*⟩

lemma *conn-corner3-alt*: $j \neq i \Longrightarrow j \neq i + 1 \Longrightarrow \Gamma\Gamma\ i\ \alpha\ (\partial\ i\ \beta\ x \odot_j \partial\ i\ \gamma\ y) = \Gamma\ i\ \alpha\ (\partial\ i\ \beta\ x) \odot_j \Gamma\ i\ \alpha\ (\partial\ i\ \gamma\ y)$
⟨*proof*⟩

lemma *conn-shift2*:
assumes *fFx i x*
and *fFx* $(i + 2)\ x$
shows $\vartheta\ i\ (\vartheta\ (i + 1)\ (\Gamma\ i\ \alpha\ x)) = \Gamma\ (i + 1)\ \alpha\ (\vartheta\ (i + 1)\ x)$

⟨*proof*⟩

end

end

5 Cubical $(\omega, 0)$-Categories with Connections

theory *CubicalOmegaZeroCategoriesConnections*
imports *CubicalCategoriesConnections*

begin

All categories considered in this component are single-set categories.

First we define shell-invertibility.

abbreviation (**in** *cubical-omega-category-connections*) *ri-inv i x y* ≡ (*DD i x y* ∧ *DD i y x* ∧ $x \otimes_i y$ = ∂ *i ff x* ∧ $y \otimes_i x$ = ∂ *i tt x*)

abbreviation (**in** *cubical-omega-category-connections*) *ri-inv-shell k i x* ≡ (∀ *j* α. *j* + *1* ≤ *k* ∧ *j* ≠ *i* ⟶ (∃ *y*. *ri-inv i* (∂ *j* α *x*) *y*))

Next we define the class of cubical $(\omega, 0)$-categories with connections.

class *cubical-omega-zero-category-connections* = *cubical-omega-category-connections* +
assumes *ri-inv*: *k* ≥ *1* ⟹ *i* ≤ *k* − *1* ⟹ *dim-bound k x* ⟹ *ri-inv-shell k i x* ⟹ ∃ *y*. *ri-inv i x y*

begin

Finally, to show our axiomatisation at work we prove Proposition 2.4.7 from our companion paper, namely that every cell in an $(\omega, 0)$-category is ri-invertible for each natural number i. This requires some background theory engineering.

lemma *ri-inv-fix*:
assumes *fFx i x*
shows ∃ *y*. *ri-inv i x y*
⟨*proof*⟩

lemma *ri-inv2*:
assumes *k* ≥ *1*
assumes *dim-bound k x*
and *ri-inv-shell k i x*
shows ∃ *y*. *ri-inv i x y*
⟨*proof*⟩

lemma *ri-inv3*:
assumes *dim-bound k x*
and *ri-inv-shell k i x*
shows ∃ *y*. *ri-inv i x y*
⟨*proof*⟩

lemma *ri-unique*: (∃ *y*. *ri-inv i x y*) = (∃!*y*. *ri-inv i x y*)
⟨*proof*⟩

lemma *ri-unique-var*: *ri-inv i x y* ⟹ *ri-inv i x z* ⟹ *y* = *z*
⟨*proof*⟩

definition *ri i x* = (*THE y*. *ri-inv i x y*)

lemma *ri-inv-ri*: *ri-inv i x y* ⟹ (*y* = *ri i x*)

$\langle proof \rangle$

lemma *ri-def-prop*:
assumes *dim-bound k x*
and *ri-inv-shell k i x*
shows $DD\ i\ x\ (ri\ i\ x) \wedge DD\ i\ (ri\ i\ x)\ x \wedge x \otimes_i (ri\ i\ x) = \partial\ i\ ff\ x \wedge (ri\ i\ x) \otimes_i x = \partial\ i\ tt\ x$
$\langle proof \rangle$

lemma *ri-right*:
assumes *dim-bound k x*
and *ri-inv-shell k i x*
shows $x \otimes_i ri\ i\ x = \partial\ i\ ff\ x$
$\langle proof \rangle$

lemma *ri-right-set*:
assumes *dim-bound k x*
and *ri-inv-shell k i x*
shows $x \odot_i ri\ i\ x = \{\partial\ i\ ff\ x\}$
$\langle proof \rangle$

lemma *ri-left*:
assumes *dim-bound k x*
and *ri-inv-shell k i x*
shows $ri\ i\ x \otimes_i x = \partial\ i\ tt\ x$
$\langle proof \rangle$

lemma *ri-left-set*:
assumes *dim-bound k x*
and *ri-inv-shell k i x*
shows $ri\ i\ x \odot_i x = \{\partial\ i\ tt\ x\}$
$\langle proof \rangle$

lemma *dim-face*: $dim\text{-}bound\ k\ x \Longrightarrow dim\text{-}bound\ k\ (\partial\ i\ \alpha\ x)$
$\langle proof \rangle$

lemma *dim-ri-inv*:
assumes *dim-bound k x*
and *ri-inv i x y*
shows *dim-bound k y*
$\langle proof \rangle$

lemma *every-dim-k-ri-inv*:
assumes *dim-bound k x*
shows $\forall i.\ \exists y.\ ri\text{-}inv\ i\ x\ y$ $\langle proof \rangle$

We can now show that every cell is ri-invertible in every direction i.

lemma *every-ri-inv*: $\exists y.\ ri\text{-}inv\ i\ x\ y$
$\langle proof \rangle$

end

end

References

[1] P. Malbos, T. Massacrier, and G. Struth. Single-set cubical categories and their formalisation with a proof assistant. 2024. http://arxiv.org/abs/2401.10553v1.

[2] G. Struth. Catoids, categories, groupoids. *Arch. Formal Proofs*, 2023, 2023.