A formal verification of the theory of parity complexes
Mitchell Buckley
April 10, 2015

Abstract

We formalise, in Coq, the opening sections of Parity Complexes [Str91] up to and including the all important excision of extremals algorithm. Parity complexes describe the essential combinatorial structure exhibited by simplexes, cubes and globes, that enable the construction of free $\omega$-categories on such objects. The excision of extremals is a recursive algorithm that presents every cell in such a category as a composite of atomic cells, this is the sense in which the $\omega$-category is free. Due to the complicated multi-dimensional nature of this work, the detail of definitions and proofs can be hard to follow and verify. Indeed, some corrections [Str94] were required some years following the original publication. Our formalisation verifies that all cases of each result operate as stated. In particular, we indicate which portions of the theory can be proved directly from definitions, and which require more subtle and complex arguments. By identifying results that require the most complicated proofs, we are able to investigate where this theory might benefit from further study and which results need to be considered most carefully in future work.

1 Introduction

An $n$-simplex $\Delta_n$ is a geometric figure that generalises the notion of triangle or tetrahedron to $n$-dimensional space. Simplexes have a number of properties that make them useful in algebraic topology, algebraic geometry and homotopy theory where they often play a foundational role. Each $n$-simplex can be oriented in such a way that it forms an $n$-category. We include below the cases for $n = 1$, 2 and 3.

This orientation is easily described at all dimensions, but as $n$ increases beyond three it becomes very difficult to describe the $n$-category structure.
Beginning in the late 1970’s Ross Street, together with John Roberts and Jack Duskin, began investigating how this process could be rigorously extended to any $n$. This was achieved in [Str87] where the process was described for the simplexes and the corresponding categories were dubbed the orientals (referring to the fact that they are oriented). The main motivation at this time stemmed from non-abelian cohomology where various constructions rely on the orientals.

At the same time, Iain Aitcheson was developing a similar series of results for $n$-cubes: that each cube could be given an orientation in such a way that it forms an $n$-category or even an $\omega$-category [Ait10]. A third example of this phenomenon is found in $n$-globes where the corresponding $n$-categories have a very simple description. For more on the usefulness of simplexes and cubes, see Street’s survey [Str95].

Following these successes, the goal was then to describe the general structure of all oriented multi-dimensional structures for which it is possible to extract free $\omega$-categories in the style of these three examples. The early 1990’s yielded a number of related solutions. Ross Street defined a structure called a parity complex and gave an explicit description of the $\omega$-category associated to each [Str91]. Some minor corrections were added in [Str94]. Richard Steiner contributed directed complexes as a generalisation of directed graph. He showed that loop-free directed complexes generated free $\omega$-categories in the appropriate way [Ste93]. Both of these authors also showed that their respective structures were closed under product and join and covered three main examples of simplexes, cubes, and globes. Around the same period Mike Johnson was working on a formal description of pasting scheme for $\omega$-categories [Joh88, Joh89], and was able to describe the free $\omega$-category on such structures. He included the simplexes as his primary example, and there is a strong sense in which this addressed the same problem. Further related work can be found in [AAS93, Ste04] and also in [Ver08] where a conjecture of Street–Roberts is proved in the closing chapter.

Our interest centres on Parity Complexes which takes a particularly ‘hands-on’ approach and describes the combinatorics of this construction in full detail. Our goal is to encode and verify the opening sections of this text up to the excision of extremals algorithm. The theory shows how to build, for any parity complex $C$ an $\omega$-category $O(C)$. The excision of extremals algorithm shows that each cell can be presented as a composite of atomic cells; this is the sense in which $O(C)$ is free. The algorithm can also be used to generate explicit algebraic descriptions of the cells in $O(C)$.

Our motivation is two-fold. First, some of the combinatorial arguments in Street’s text can be difficult to follow and can easily conceal errors; this is illustrated by the fact that corrections were later required. We will provide some confirmation that the corrections have addressed all issues. Second, a computer-verified encoding provides a good resource for understanding the intricacies of these complicated structures and opens a path to further refinements of the material. The excision of extremals algorithm provides a natural breaking point in the text as it is central to showing that the relevant categories are freely generated.

From this point on we often refer to [Str91] as the ‘original text’, and
to [Str94] as ‘the corrigenda’.

We programmed everything in Coq [Coq] and the code is freely available for inspection at the following location.

https://github.com/MitchellBuckley/Parity-Complexes

In Section 1 we give a brief introduction to the history of parity complexes and explain our goals in this paper. In Section 2 we outline the foundational mathematics that needs to be introduced for an encoding of parity complexes. We also outline how we chose to implement this foundation. In Section 3 we outline the content of [Str91] section-by-section. At each stage we comment on the intuition underlying each result and discuss our implementation of the definitions and results. We pay particular attention to those sections of the material that were difficult to translate into Coq. In Section 4 we discuss how formalisation has shed light on the material and make suggestions for how future work might proceed. In Section 5 we outline the few lessons we have learned in computer-verified encoding of mathematics. Section 6 contains concluding remarks.

2 Required Foundations

Parity complexes are described using basic set theory and partially ordered sets. In particular, we need to implement:

- sets;
- set union, set intersection, set difference etc.;
- finite sets;
- cardinality of finite sets;
- partial orders; and
- segments of partial orders.

Many of these structures are already encoded in the Coq standard library.

2.1 Sets

We implement sets using the Ensembles standard library. This involves a universe type \( U : \text{Type} \) on which all our sets will be based. Then a set is an ensemble: an indexed proposition \( U \rightarrow \text{Prop} \). An element of the universe \( x : U \) is a member of a set \( A : U \rightarrow \text{Prop} \) when the corresponding proposition \( A \ x \) is true. Inclusion of sets relies on logical implication.

```coq
Definition Ensemble := U -> Prop.
Definition In (A:Ensemble) (x:U) : Prop := A x.
Definition Included (B C:Ensemble) : Prop :=
  forall x:U, In B x -> In C x.
```
Set operations union, intersection, and set difference are all implemented using point-wise logical operations:

\[
\begin{align*}
\text{Union } A \, B & := \ \text{fun } x \mapsto (A \, x \lor B \, x) \\
\text{Intersection } A \, B & := \ \text{fun } x \mapsto (A \, x \land B \, x) \\
\text{Setminus } A \, B & := \ \text{fun } x \mapsto (A \, x \land \neg(B \, x))
\end{align*}
\]

We will henceforth suppose that we are working within a fixed universe \( U \).

The Coq language has a convenient feature that allows us to introduce notation for these operations.

\[
\begin{align*}
\text{Notation } "x \in B" & := (\text{In } A \, x) \ (\text{at level } 71). \\
\text{Notation } "A \subseteq B" & := (\text{Included } A \, B) \ (\text{at level } 71). \\
\text{Notation } "A \cup B" & := (\text{Union } A \, B) \ (\text{at level } 61). \\
\text{Notation } "A \cap B" & := (\text{Intersection } A \, B) \ (\text{at level } 61). \\
\text{Notation } "A \setminus B" & := (\text{Setminus } A \, B) \ (\text{at level } 61).
\end{align*}
\]

Each special symbol is introduced as a utf-8 character which Coq has no problem recognising. This feature makes the code much more readable.

### 2.2 Finiteness and cardinality

Finiteness is implemented using the `Finite_sets` standard library. This contains an inductively defined proposition `Finite` stating that a set \( S \) is finite when \( S = \emptyset \), or \( S = \{x\} \cup S' \) where \( S' \) is finite. Cardinality is implemented in a similar way, using the same library. There is an inductively defined proposition `cardinal` stating that a set \( S \) has cardinality 0 when it is empty and has cardinality \( n + 1 \) when \( S = \{x\} \cup S' \) and \( S' \) has cardinality \( n \).

\[
\begin{align*}
\text{Inductive } \text{Finite} : \text{Ensemble } U \rightarrow \text{Prop} := \\
| \text{Empty_is_finite} : \text{Finite} (\text{Empty_set } U) \\
| \text{Union_is_finite} : \\
\quad \forall A:\text{Ensemble } U, \\
\quad \text{Finite } A \rightarrow \forall x:U, \neg \text{In } U \ A \ x \rightarrow \\
\quad \text{Finite} (\text{Add } U \ A \ x).
\end{align*}
\]

\[
\begin{align*}
\text{Inductive } \text{cardinal} : \text{Ensemble } U \rightarrow \text{nat} \rightarrow \text{Prop} := \\
| \text{card_empty} : \text{cardinal} (\text{Empty_set } U) 0 \\
| \text{card_add} : \\
\quad \forall (A:\text{Ensemble } U) (n:\text{nat}), \\
\quad \text{cardinal } A \ n \rightarrow \forall x:U, \neg \text{In } U \ A \ x \rightarrow \\
\quad \text{cardinal} (\text{Add } U \ A \ x) (S \ n).
\end{align*}
\]

When our universe has decidable equality we can show that finiteness interacts well with set operations, for example \( \forall A \, B, \text{Finite } A \land \text{Finite } B \rightarrow \text{Finite } (A \cup B) \). Cardinality and finiteness are related by the result \( \forall S, (\text{Finite } S \leftrightarrow \exists n, \text{cardinal } S \ n) \).

### 2.3 Partial orders

Some material on partial orders is available in the `Relations` standard library. Our particular requirements for orders were slightly more complicated than that library could help us with. We found it simpler to explicitly prove basic results as they were needed.
2.4 Equality of sets

We say that two sets $S$ and $T$ are equal when they are equal as terms of the type $\text{Ensemble } U$; in that case $\forall x, S \ x = T \ x$. We write $S = T$ to indicate that $S$ and $T$ are equal. This is the standard notion that is built into Coq and allows us to replace $S$ with $T$ in any expression.

There is another notion of equality: we say that $S$ and $T$ are the same when they are equivalent as propositions, that is $\forall x, S \ x \leftrightarrow T \ x$. Equivalently, $S \subseteq T \land T \subseteq S$. We write $\text{Same_set } S \ T$ or $S == T$ to indicate that $S$ and $T$ are the same.

The standard library $\text{Ensembles}$ contains an extensionality axiom stating that $\forall A \ B, A == B \rightarrow A = B$. In order to keep our formalisation as constructive as possible we are careful never to the axiom in our formalization.

While the ‘same set’ relation is clearly an equivalence, Coq does not automatically allow us to rewrite $S$ with $T$ whenever $S == T$. We use the standard library $\text{Setoid}$ and setoid rewrite functionality to account for this. Once we have proved $\text{Same_set}$ is an equivalence relation and that the appropriate set operations preserve sameness, we are free to rewrite $S$ for $T$ whenever $S == T$.

Without the extensionality axiom it is not possible to prove that $\text{Finite } S$ and $S == T$ implies $\text{Finite } T$. Something similar happens with the definition of finite cardinality. The problem occurs when we try to show that $T == \text{Empty_set}$ implies that $\text{Finite } T$. In that case, we find that neither $T = \text{Empty_set}$ nor $T = \{x\} \cup T'$ and so neither constructor will show that $T$ is finite. This problem can be solved in more than one way. We chose to solve this by adding a third constructor for $\text{Finite}$ that explicitly introduces the property that $\text{Finite } S \land S == T \rightarrow \text{Finite } T$. This modification allows us to recover this basic property of finite sets without the extensionality axiom. This illustrates how careful one must be with even the most basic of definitions.

2.5 More on finiteness

In many cases we augmented the standard library with extra results about finite sets that were not already present. We found that setting up this basic theory was often tedious, but occasionally a fun exercise in constructive mathematics. For instance, it became clear at some point that certain basic results about sets could not be proved without supposing that equality in $U$ is decidable, i.e. $\forall a \ b : U, (a = b) \lor \neg(a = b)$. We have made this a further assumption in our implementation. It does not force our arguments to be non-constructive, but it does place a mild restriction on the kinds of universe we can work within.

We have now covered the essential mathematical foundations required for a formalization of parity complexes. More details can be found by examining the code itself.
3 Parity Complexes

In this section we summarise Sections 1 to 4 of [Str91] together with modifications given in the corrigenda [Str94]. These sections are sufficient to express the excision of extremals algorithm (Theorem 4.1). As we progress through the material we will usually reproduce definitions and terminology verbatim from [Str91], [Str94]. In each case we will explain the underlying intuition of the material, comment on our implementation, and indicate where our formalisation shed light on the underlying arguments. Unless otherwise stated, we exactly preserve the numbering and labelling of axioms, propositions and theorems from [Str91], [Str94].

3.1 Definitions and the simplex example

We begin by summarising the content of Section 1 of [Str91].

Definition. A parity complex is a graded set

\[ C = \sum_{n=0}^{\infty} C_n \]  

(2)

together with, for each \( x \in C_{n+1} \) two disjoint, non-empty, finite sets \( x^+, x^- \subseteq C_n \) subject to Axioms 1, 2, 3A and 3B which appear below.

Before we list the axioms we will introduce some terminology. Elements of \( x^- \) are called negative faces of \( x \), and those of \( x^+ \) are called positive faces of \( x \). We will sometimes refer to \( x^- \) and \( x^+ \) as face-sets of \( x \). Given \( S \subseteq C \), let \( S^- \) denote the set of elements of \( C \) which occur as negative faces of some \( x \in S \), and similarly for \( S^+ \).

\[ S^- = \bigcup_{w \in S} w^- \quad \text{and} \quad S^+ = \bigcup_{w \in S} w^+ \]  

(3)

Each subset \( S \subseteq C \) is graded via \( S_n = S \cap C_n \). The n-skeleton of \( S \subseteq C \) is defined by

\[ S^n := \sum_{k=0}^{n} S_k \]  

(4)

Call \( S \) n-dimensional when it is equal to its n-skeleton.

The broad intuition is to see this structure as a generalisation of directed graph. Elements of dimension \( C_0 \) are vertices, elements of \( C_1 \) are directed edges, elements of \( C_2 \) are directed ‘faces’, elements of \( C_3 \) are directed ‘volumes’, and so on. The usual notion of source and target are replaced by face-sets \( x^- \) and \( x^+ \).
$x^+$. The following is a basic example of this structure of dimension two.

\[
\begin{array}{c}
\bullet \quad \bullet \quad \bullet \quad \bullet \\
\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
\bullet \quad \bullet \quad \bullet \quad \bullet \\
\end{array}
\]

Notice that elements above dimension 1 can have more than one source-face or target-face.

Without the axioms below, this structure is very general indeed and many unusual examples can be provided. When the axioms are applied, the kinds of examples we can build are much better behaved. Examples of arbitrary dimension can be constructed from simplexes, cubes, and other kinds of polytopes. The simplexes provide the main motivation for understanding these kinds of structure.

So far we have described the data of a parity complex: a graded set with a pair of face-set maps $(-)^-, (-)^+: C_{n+1} \to \mathcal{P}(C_n)$. We also require the following three axioms to ensure that it is somewhat well-behaved.

**Axiom 1.** For all $x$,

\[ x^{++} \cup x^{--} = x^{-+} \cup x^{+-} . \]

This is a kind of globularity condition that ensures various face-sets are appropriately related. The following diagram is an example where $x \in C_2$ and both $x^-$ and $x^+$ have four elements.

\[
\begin{array}{c}
\bigcirc \quad \bigcirc \quad \bigcirc \\
\downarrow \quad \downarrow \\
\bigcirc \quad \bigcirc \\
\end{array}
\]

Edges marked with a dotted line belong to $x^-$, the other edges belong to $x^+$. Vertices marked with a $\bullet$ belong to $x^{++}$, those marked with a $\circ$ belong to $x^{--}$, those marked with a $\bigcirc$ belong to $x^{-+}$, and those marked with a $\bigcirc$ belong to $x^{+-}$. In particular, this axiom implies that $x^{++} \subseteq x^{-+} \cup x^{+-}$, that is, positive faces of positive faces must be the negative face of a positive face, or the positive face of a negative face.

Notice that in each of these examples the set of source (target) faces have all elements aligned in a common direction, and they do not branch apart. This behaviour is guaranteed by introducing Axiom 2 below.

Write $S \perp T$ when $S^- \cap T^- = S^+ \cap T^+ = \emptyset$. This extends to elements by $x \perp y$ when $x^- \cap y^- = x^+ \cap y^+ = \emptyset$. A subset $S \subseteq C$ is called *well-formed* when $S_0$ has at most one element, and, for all $x, y \in S_n$ ($n > 0$), if $x \neq y$ then $x \perp y$. 

7
Broadly speaking, a set is well-formed when it doesn’t contain any branchings

\[ \bullet \xrightarrow{x} \bullet \quad \text{or} \quad \bullet \xrightarrow{y} \bullet, \]  

and it contains at most one element of dimension 0. The diagram in (7) depicts a branching in dimension 1, but well-formedness prevents branching in all dimensions.

**Axiom 2.** For all \( x \), \( x^- \) and \( x^+ \) are well-formed.

If we think of the union \( x^- \cup x^+ \) as forming a boundary of \( x \), as in (6) above, then this axiom ensures that the boundary looks something like the boundary of a polytope. For those familiar with higher categories, this condition ensures that the face-sets look like valid pasting diagrams. In particular, it ensures that elements of dimension 1 have a single source vertex and a single target vertex.

Write \( x < y \) whenever \( x^+ \cap y^- \) is non-empty. That is, when \( x \) and \( y \) abut by having a common element in their respective sets of positive and negative faces. This implies \( x \neq y \) since \( x^- \) and \( x^+ \) are disjoint. We then let \( < \) be the reflexive transitive closure of \( < \). An example is

\[ \downarrow \bullet \xrightarrow{x} \bullet \downarrow \quad \downarrow \bullet \xrightarrow{y} \bullet \downarrow \quad \downarrow \bullet \xrightarrow{z} \bullet \]  

where \( x < y \) and \( y < z \). In this case we often say that there is a path from \( x \) to \( z \). For all \( S \subseteq C \) we let \( <_S \) denote the reflexive transitive closure of \( < \) restricted to \( S \). When \( x <_S z \) we say there is a path from \( x \) to \( z \) in \( S \).

While Axioms 1 and 2 can be seen as imposing some of the basic structural behaviour of graphs, the following axiom restricts us to certain ‘loop-free’ graphs.

**Axiom 3.**

A. \( x < y < x \) implies \( x = y \).

B. if \( x < y \) then \( \forall z, \neg(x \in z^+ \land y \in z^-) \text{ and } \neg(y \in z^+ \land x \in z^-) \).

Axiom 3.A says that \( < \) is anti-symmetric, or, that there are no paths that loop within a fixed dimension. Axiom 3.B says that there are no paths that cross between the face-sets of any element \( z \). That is, we avoid circumstances
where a path can cross from one face-set to the other face-set of an element \( z \) as in the diagram below.

These are all the axioms for a parity complex. The following examples come from p.318–319 of [Str91].

**Example.** A 1-dimensional parity complex is precisely a directed graph with no circuits.

**Example.** The \( \omega \)-glob is the parity complex \( G \) defined by \( G_n = \{ (\epsilon, n) : \epsilon = - \text{ or } + \} \), and \( (\epsilon, n + 1)^- = \{ (-, n) \} \) and \( (\epsilon, n + 1)^+ = \{ (+, n) \} \). Elements of dimension 0, 1, and 2 are 'n-discs'.

**Example.** The \( \omega \)-simplex is the parity complex \( \Delta \) described as follows. Let \( \Delta_n \) denote the set of \( (n + 1) \)-element subsets of the set of natural numbers \( N = \{ 0, 1, 2, \ldots \} \). Each \( x \in \Delta_n \) is written as \( (x_0, x_1, \ldots, x_n) \) where \( x_0 < x_1 < \cdots < x_n \). Let \( x\delta_i \) denote the set obtained from \( x \) by deleting \( x_i \). Take \( x^- \) to be \( \{ x\delta_i : i \text{ odd} \} \) and \( x^+ \) to be \( \{ x\delta_i : i \text{ even} \} \). Elements of dimension 0, 1, and 2 are 'n-simplexes'.

**Example.** The \( \omega \)-cube is the parity complex \( Q \) described as follows. The elements are infinite sequences of the three symbols \(-, 0, +\) containing a finite number of 0's. The dimension of an element is the number of 0's appearing in it. Let \( x\delta_i^- \) denote the sequence obtained from \( x \) by replacing the \( i \)-th 0 by \(-\) when \( i \) is odd and by \(+\) when \( i \) is even. Similarly, \( x\delta_i^+ \) is defined by interchanging \(-\) and \(+\) in the previous sentence. For \( x \in Q_n \), define \( x^\prime = \{ x\delta_i^\prime : 1 < i < n \} \).
Elements of dimension 1, 2, and 3 are ‘$n$-cubes’.

Before continuing our exposition of Section 1 of [Str91], we will comment briefly on our implementation.

**Implementation 3.2.** The basic data for a parity complex without the axioms is sometimes called a *pre-parity complex*. We chose to implement this concept first, as there are many trivial results about preparity complexes that we will later use. A preparity complex is implemented as the following data:

\[
\begin{align*}
C & : \text{Type} \\
\text{dim} & : C \rightarrow \text{nat} \\
\text{plus} & : C \rightarrow \text{Ensemble} \ C \\
\text{minus} & : C \rightarrow \text{Ensemble} \ C
\end{align*}
\]

This data is technically different from our description above, but the essential structure is identical. There is a collection of objects $C$, each member of which has a dimension and two face-sets. A few axioms are introduced to ensure that face-sets are finite, non-empty, and disjoint, and that they interact with dimension correctly.

\[
\begin{align*}
\forall (x \ y : C), \ x & \in (\text{plus} \ y) \rightarrow \text{dim} \ y = \text{dim} \ x + 1 \\
\forall (x \ y : C), \ x & \in (\text{minus} \ y) \rightarrow \text{dim} \ y = \text{dim} \ x + 1 \\
\forall (x : C), \ \text{Finite} \ (\text{plus} \ x) \\
\forall (x : C), \ \text{Finite} \ (\text{minus} \ x) \\
\forall (x : C), \ \text{dim} \ x > 0 \rightarrow \text{Inhabited} \ (\text{plus} \ x) \\
\forall (x : C), \ \text{dim} \ x > 0 \rightarrow \text{Inhabited} \ (\text{minus} \ x) \\
\forall (x : C), \ \text{dim} \ x = 0 \rightarrow \text{plus} \ x = \text{Empty_set} \\
\forall (x : C), \ \text{dim} \ x = 0 \rightarrow \text{minus} \ x = \text{Empty_set} \\
\forall (x : C), \ \text{Disjoint} \ (\text{plus} \ x) \ (\text{minus} \ x)
\end{align*}
\]

These are given meaningful names such as `plus_Finite`, `plus_dim`, and `plus_Inhabited`.

Fundamental definitions for sets such as $S_n$ and $S^n$ are also given and some trivial statements are also proved here. For example,

\[
\begin{align*}
\text{Definition sub} \ (R : \text{Ensemble} \ C) \ (n : \text{nat}) & : \text{Ensemble} \ C \\
& := \text{fun} \ (x : C) \Rightarrow (x \in R \land (\text{dim} \ x) = n).
\end{align*}
\]

\[
\begin{align*}
\text{Lemma sub_Union} & : \\
& \forall T \ R \ n, \\
& \quad \text{sub} \ (T \cup R) \ n = (\text{sub} \ T \ n) \cup (\text{sub} \ R \ n).
\end{align*}
\]

More complicated definitions like well-formedness are also given and more powerful (though almost trivial) results are also proved here. For example,
Definition well_formed \(X : \text{Ensemble carrier}\) : Prop :=
\[
\text{(forall (x y : carrier), x ∈ X ∧ y ∈ X}
\quad \rightarrow \, \text{dim x = 0 → dim y = 0}
\quad \rightarrow \, x = y)
\]
\[
\lor
\quad \text{(forall (x y : carrier), x ∈ X ∧ y ∈ X}
\quad \rightarrow \, \text{(forall (n : nat), dim x = S n → dim y = S n}
\quad \rightarrow \, \neg \, \text{(perp x y) → x = y)}).
\]

Lemma well_formed_by_dimension :
\[
\text{forall X,}
\quad \text{well_formed X <-> forall n, well_formed (sub X n).}
\]

All other basic definitions and trivial results are encoded in a similar fashion.

We now look at some basic properties of parity complexes.

Given \(S ⊆ C\), let \(S^-\) denote the set of negative faces of elements of \(S\) which are not positive faces of any element of \(S\), and similarly for \(S^+\). So
\[
S^- = S^- \setminus S^+ \text{ and } S^+ = S^+ \setminus S^-.
\]
This extends to individual elements by \(x^± := \{x\}^±\) and \(x^\mp := \{x\}^\mp\). These sets capture the notion of purely positive and purely negative faces of an element \(x\) or set \(S\).

The following propositions follow from Axioms 1, 2 and 3.

**Proposition 1.1.** For all \(x\),
\[
x^{++} \cap x^{--} = x^{+-} \cap x^{+--} = \emptyset \tag{13}
\]
\[
x^{-\mp} = x^{+\mp} = x^{--} \cap x^{+--} \tag{14}
\]
\[
x^{-\pm} = x^{+\pm} = x^{-+} \cap x^{++}. \tag{15}
\]

Proposition 1.1 contains identities that one would expect from a polytope-like structure and are much like Axiom 1. The meaning is reasonably clear when the various face-sets are highlighted in an example like (6) above.

**Proposition 1.2.** For all \(u, v, x, u \triangleleft v\) and \(v \in x^+\) imply
\[
u^- \cap x^{+-} = \emptyset. \tag{16}
\]

Proposition 1.2 indicates that if \(u\) branches out from the source of \(x\) then a path from \(u\) to \(v\) can not end in the target of \(x\). This is a consequence of Axiom 3.B. This has three duals obtained by reversing the roles of \(u\) and \(v\) and reversing the roles of \(x^-\) and \(x^+\). Proposition 1.2 and its duals are together equivalent to Axiom 3.B.

The following observation describes a convenient technical property of well-formed sets.

**Observation A (page 322 in [Str91]).** For all \(T, Z\), if \(T \cup Z\) is well-formed and \(T \cap Z = \emptyset\), then \(T \perp Z\).
We say a set $R$ is tight when, for all $u, v$, $u \triangleleft v$ and $v \in R$ implies $u^- \cap R^\pm$ is empty. This condition prevents a path from starting in $R^\pm$ and ending in $R$.

The following two results are required for somewhat technical reasons.

**Proposition 1.4.** For all $R, S$, if $R$ is tight, $S$ is well-formed, and $R \subseteq S$, then $R$ is a segment of $S$.

**Observation B (page 359 in [Str94]).** For all $x$, $x^+$ and $x^-$ are tight.

This concludes our exposition of Section 1.

**Remark 3.3.** The notion of tightness was introduce in the Corrigenda [Str94]. It appears to be entirely necessary, but we do not understand the full significance of the concept (see our discussion on page 24).

**Implementation 3.4.** Each axiom and proposition is readily encoded, for example

**Axiom axiom1 :**

```plaintext```
forall (x : C),
(Plus (plus x)) \cup (Minus (minus x)) ==
(Plus (minus x)) \cup (Minus (plus x)).
```

**Lemma Prop_1_2 :**

```plaintext```
forall u v x,
triangle u v \rightarrow
v \in (plus x) \rightarrow
(minus u) \cap (Plus (minus x)) == Empty_set.
```

We were able to prove each result from basic definitions and axioms. This is exactly as described in the original work. The proof of Proposition 1.4 makes use of Propositions 1.1 and 1.2.

When we look ahead we find that axioms 1 and 2 are used frequently throughout the material. Axiom 3.A is only used to prove that $\triangleleft_S$ is decidable and that finite non-empty sets $X$ have minimal and maximal elements under $\triangleleft_X$. Axiom 3.B is used only to prove Propositions 1.1 and 1.2.

### 3.5 Movement

In Section 2 of [Str91] the concept of movement is introduced. It is a concept that is fundamental to describing cells in $n$-categories generated from parity complexes.

For three sets $S, M, P$, we say that $S$ moves $M$ to $P$, or $M \xrightarrow{S} P$, when

$$M = (P \cup S^-) \setminus S^+ \quad \text{and} \quad P = (M \cup S^+) \setminus S^-.$$  \hspace{1cm} (17)

Here are two examples of movement at dimensions 2 and 1:

```
\begin{center}
\begin{tikzpicture}
\draw (0,0) node [right] {$s$};
\draw (1.5,0) node [right] {$m$};
\draw (3,0) node [right] {$p$};
\draw (4.5,0) node [right] {$m$};
\draw (6,0) node [right] {$p$};
\draw (7.5,0) node [right] {$m$};
\end{tikzpicture}
\end{center}
```

```
\begin{center}
\begin{tikzpicture}
\draw (0,0) node [right] {$s$};
\draw (1.5,0) node [right] {$m$};
\draw (3,0) node [right] {$p$};
\draw (4.5,0) node [right] {$m$};
\draw (6,0) node [right] {$p$};
\draw (7.5,0) node [right] {$m$};
\end{tikzpicture}
\end{center}
```

12
and

where lowercase labels $m, p, s$ indicate which set each component belongs to (unlabelled elements do not belong to $M, P, S$). This condition guarantees that the face-sets of $S, M$ and $P$ are related in the basic way we would expect of pastings in $n$-categories. Certain well-formedness conditions are currently lacking and will be introduced at a later stage.

It is helpful to recognise that movement is a condition that applies dimension-by-dimension, that is, $M \xrightarrow{S} P$ if and only if $M_n \xrightarrow{S_{n+1}} P_n$ for all $n$. This not only aids in various proofs, but it indicates there is nothing complicated happening across dimensions.

Proposition 2.1. For all $S, M$, there exists $P$ with $M \xrightarrow{S} P$ if and only if

$$S^\ominus \subseteq M \quad \text{and} \quad M \cap S^+ = \emptyset .$$  \hspace{1cm} (20)

Proposition 2.1 illuminates a fundamental meaning of movement: that $M$ contains the purely negative faces of $S$ and none of the positive faces. This is illustrated below where elements of $S^\ominus$ are indicated by squiggly arrows and those of $S^+_2$ are indicated by dashed arrows.

Observe that $S^\ominus_2 \subseteq M_1$ and $M_1 \cap S^+_2 = \emptyset$ as indicated by the proposition. Proposition 2.1 has an obvious dual where $M$ and $P$ play opposite roles.

Proposition 2.2. Suppose $M \xrightarrow{S} P$ and $X \subseteq M$ has $S^\ominus \cap X = \emptyset$. If $Y \cap S^+ = \emptyset$, and $Y \cap S^- = \emptyset$, then $(M \cup Y) \setminus \neg X \xrightarrow{S} (P \cup Y) \setminus \neg X$.

Proposition 2.2 indicates that some elements of $M$ and $P$ can be added or removed without disturbing the movement condition. The conditions on $X$ and $Y$ indicate that they are disjoint from the faces of $S$ in a suitable way. Sets
X and Y should be thought of as sets that are added to or removed from the movement as below.

\[
\begin{array}{c}
\bullet \quad \bullet \quad \bullet \\
| \quad | \quad |
\end{array}
\]

Proposition 2.3. Suppose \(M \xrightarrow{S} P\) and \(P \xrightarrow{T} Q\). If \(S^+ \cap T^+ = \emptyset\) then \(M \xrightarrow{S \cup T} Q\).

Proposition 2.3 describes the condition under which movements can be ‘composed’ or ‘pasted’ together. The following diagram depicts an example. Elements of sets \(M, S, P, T, Q\) are labelled with the corresponding lower-case letters.

\[
\begin{array}{c}
\bullet \quad \bullet \quad \bullet \\
\bullet \quad \bullet \\
\bullet \quad \bullet \\
\end{array}
\]

Proposition 2.4. Suppose \(M \xrightarrow{T \cup Z} P\) with \(Z^\pm \subseteq P\). If \(T \perp Z\) then there exists \(N\) such that \(M \xrightarrow{T} N \xrightarrow{Z} P\).

Proposition 2.4 describes a condition under which movement can be decomposed. In particular, if \(T \cup Z\) is well-formed then \(T \perp Z\) as required in the proposition.

Implementation 3.6. The definition of movement and the propositions above are readily encoded. For example

\[
\text{Definition moves_def (S M P : Ensemble C) : Prop :=}
\]

\[
\begin{align*}
P &= ((M \cup (\text{Plus}\ S)) \cap (\text{Complement}(\text{Minus}\ S))) \\
M &= ((P \cup (\text{Minus}\ S)) \cap (\text{Complement}(\text{Plus}\ S))).
\end{align*}
\]

Notation "S 'moves' M 'to' P" := (moves_def S M P) (at level 89).

Lemma Prop_2_3 : forall (S M P T Q : Ensemble C),
S moves M to P →
T moves P to Q →
(Disjoint (Minus S) (Plus T)) →
(S ∪ T) moves M to Q.
The Notation command in Coq allows us to use the statement \( S \) moves \( M \) to \( P \) in place of the somewhat awkward \texttt{moves_def S M P}.

It did not take long to verify that the proofs in this section proceed precisely as indicated in the original text.

Proposition 2.1 is proved by appealing to definitions and basic manipulation of sets. Propositions 2.2, 2.3, and 2.4 are proved using 2.1 and basic manipulation of sets. Propositions 2.1 and 2.4 have duals that are not displayed here but are required later; they are implemented separately in our code. It is worth noting that none of these results require Axioms 1, 2 or 3. In our implementation, we prove these results before the axioms are even introduced.

We also implement a more specific version of Proposition 2.1 that states: For all \( S, M, S \) moves \( M \) to \((M \cup S^+) \cap \neg S^-\) if and only if \( S^- \subseteq M \) and \( M \cap S^+ = \emptyset \). This not a stronger statement, but it is sometime more convenient than the original proposition.

This concludes our exposition of Section 2 of [Str91].

3.7 The \( \omega \)-category of a parity complex

Having described the basic properties of parity complexes and the more advanced notion of movement, in Section 3 we describe the cells of an \( \omega \)-category \( O(C) \) associated with any parity complex \( C \).

**Definition.** A cell of a parity complex \( C \) is a pair \((M, P)\) of non-empty, well-formed, finite, subsets of \( C \) with the property that \( M \) and \( P \) both move \( M \) to \( P \).

If this is interpreted dimension by dimension, we get the following picture at dimension 2,

![Diagram](image)

where lowercase labels \( m, p, s \) indicate which set the elements belong to. Notice that \( M_1 \) and \( P_1 \) are neither equal nor disjoint, but each move \( M_0 \) to \( P_0 \). Notice also that \( M_2 = P_2 \). This kind of behaviour is generally repeated through all dimensions. Notice also that, aside from the movement condition, we only require that \( M \) and \( P \) be non-empty, well-formed and finite. Call \((M, P)\) an \( n \)-cell when \( M \cup P \) is \( n \)-dimensional. In this case we have \( M_n = P_n \) as above.

**Definition.** The \( n \)-source and \( n \)-target of a pair \((M, P)\) are defined by

\[
s_n(M, P) = (M^{n-1} \cup M_n, P^{n-1} \cup M_n)
\]
and
\[ t_n(M, P) = (M^{n-1} \cup P_n, P^{n-1} \cup P_n) \, . \] (26)

If \((M, P)\) is a cell we can show that \(s_n(M, P)\) and \(t_n(M, P)\) are also cells, and that they are \(n\)-dimensional. Notice that \((M, P), s_n(M, P)\) and \(t_n(M, P)\) contain exactly the same elements in dimension \(n-1\) and below. We encourage the reader to consider the 1-source and 1-target of the cell depicted in (24).

**Definition.** A pair of cells \((M, P), (N, Q)\) are \(n\)-composable when
\[ t_n(M, P) = s_n(N, Q) \, , \] (27)
in which case their \(n\)-composite is
\[ (M \cup (N \cap \neg N_n), (P \cap \neg P_n) \cup Q) \, . \] (28)

Notice that (27) implies that the two cells agree from dimensions 0 to \(n-1\) and that \(P_n = N_n\) at dimension \(n\). The resulting composite is almost exactly the pair-wise union of \((M, P)\) and \((N, Q)\); the set-difference ensures correct behaviour at dimension \(n\). It is not surprising that some form of set-difference is required since most forms of composition will forget the point of contact: \(A \rightarrow B \rightarrow C\) composes to \(A \rightarrow C\). We will show later that the composite of any two cells is also a cell.

For any parity complex \(C\), let \(\mathcal{O}(C)\) be the set of cells of \(C\). We will see later (Theorem 3.6) that \(\mathcal{O}(C)\) is an \(\omega\)-category. Before this can be achieved, we need to establish some basic properties of cells.

**Definition.** A set is receptive when for all \(x\),
\[ \text{if } x^+ \cap x^{++} \subseteq S \text{ and } S \cap x^{-} = \emptyset \text{ then } S \cap x^{+} = \emptyset \]
and
\[ \text{if } x^{-} \cap x^{--} \subseteq S \text{ and } S \cap x^{+} = \emptyset \text{ then } S \cap x^{-} = \emptyset \, . \]

A cell is receptive when it is receptive at every dimension.

**Remark 3.8.** The notion of receptivity is somehow important, we find later that all cells are receptive and it is a necessary condition for some central results. It appears to be entirely necessary, but we do not have an intuitive understanding of its meaning (see our discussion on page 24).

**Lemma 3.1.** If \(M \xrightarrow{x^+} P\) and \(M\) is receptive, then \(M \xrightarrow{x^-} P\).

Lemma 3.1 is proved using definitions, basic manipulation of sets and Propositions 2.1 and 1.1. It has a dual which we implement in our code. We will find later that, since all cells are receptive, this movement condition is true whenever \(M\) is part of a cell. If one chooses to draw a basic example, then one will find that this is not a surprising result. One might even consider it surprising that the lemma cannot be proved by more trivial means.
Lemma 3.2. Suppose all cells are receptive and suppose \((M, P)\) is an \(n\)-cell. Suppose \(X\) is finite and well-formed with \(X \subseteq C_{n+1}\) and \(X^\pm \subseteq M_n\). Put \(Y = (M_n \cup X^-) \cap \neg X^+\), then:

\[\begin{align*}
B. \quad (M^n \cup Y, P^n \cup Y) \text{ is a cell and and } X^- \cap M_n &= \emptyset. \\
C. \quad (M^n \cup Y \cup X, P \cup X) \text{ is a cell.}
\end{align*}\]

Lemma 3.2.A is contained in [Str91] but was removed in [Str94]. Lemma 3.2.C indicates that, if \(X\) is a well-formed set of dimension \(n + 1\), \((M, P)\) is an \(n\)-cell, and \(X\) abuts \((M, P)\) in the sense that \(X^\pm \subseteq M_n\), then we can form an \((n + 1)\)-cell whose top-dimension elements are those of \(X\) and whose target is \((M, P)\). The source of this cell has \(Y\) at its top dimension. The following diagram is labelled to illustrate this scenario.

\[
\begin{array}{ccc}
\bullet & m & \bullet \\
\downarrow & y & \downarrow \\
\bullet & \bullet & \bullet \\
\downarrow & \downarrow & \downarrow \\
\bullet & m & \bullet \\
\end{array}
\]

There is a dual lemma obtained by reversing the direction of \(X\) in the diagram above.

This kind of result does not seem unusual, but it is surprisingly hard to prove (see Implementation 3.10 below). The proof itself is done in three steps. To quickly summarise:

1. 3.2.B implies 3.2.C. The proof is somewhat direct and proceeds as indicated in the original paper.

2. 3.2.B with \(m = 1\) implies 3.2.B in general. This is done by induction on \(m\) and follows from basic definitions and axioms.

3. 3.2.B holds for \(m = 1\). This is done by induction on \(n\) and the argument relies on Proposition 3.3. The construction works as indicated, though it is not a short argument. There are particular disjointness conditions that must be established (p327 of [Str91]) and require their own special argument.

Proposition 3.3. All cells are receptive.

This is a somewhat technical result, it is not immediately clear to us how the notion of receptivity fits naturally into the combinatorics. The proof of this result relies on Lemma 3.2.B.

Theorem 3.6. If \(C\) is any parity complex then \(O(C)\) is an \(\omega\)-category. Furthermore, if \((M, P)\), \((N, Q)\) are \(n\)-composable cells then \((M_k \cup P_k)^- \cap (N_k \cup Q_k)^+ = \emptyset\) for all \(k > n\).
Theorem 3.6 is a central result in [Str91] since it achieves one of the main goals of the paper. In order to implement Theorem 3.6 we would need to implement a notion of $\omega$-category which is not trivial. Since there is little question that this result holds, and it is not required to prove Theorem 4.1, we have chosen not to implement it. We similarly omit Propositions 3.4 and 3.5 which are preliminary results leading up to Theorem 3.6.

**Remark 3.9.** Perceptive readers will have noticed that Lemma 3.2.B and Proposition 3.3 seem to logically rely on one another. At first glance this appears to be a circular argument and therefore unsound. However, if we look closely we can see that each result proceeds by induction and that the two proofs can be woven together to produce a proof of both results simultaneously. Proposition 3.3 is restated as *for all n, every n-dimensional cell $(M, P)$ is receptive.* The two results are proved by mutual induction on $n$, the dimension $(M, P)$. Included in that argument is an induction on $m = |X|$. The following statements hold and are enough to show that both results hold for all $n$ and $m$.

i. Lemma 3.2.B holds when $m = 1$ and $n = 0$.

ii. For a fixed $n$, if Lemma 3.2.B holds when $m = 1$ then it holds when $m > 1$ (by induction on $m$).

iii. Proposition 3.3 holds when $n = 0$.

iv. If Lemmas 3.2.B and Proposition 3.3 hold for $n = k$, then Lemma 3.2.B holds for $n = k + 1$ and $m = 1$.

v. If Lemma 3.2.B holds for $n = k + 1$ and Proposition 3.3 holds for $n = k$, then Proposition 3.3 holds for $n = k + 1$.

This understanding is not explicit in [Str91].

**Implementation 3.10.** As in earlier sections, the definitions and statement of results are readily encoded. The main difficulty arises in encoding the proofs.

The proofs of Lemma 3.2 and Proposition 3.3 are by far the most difficult part of the entire project and consumed most of our programming effort. Consider the components of the proof given above. Each of the components follow the argument provided by Street in his paper. However the disjointness condition in iv has a dual, and i, ii, iv each have duals. Finally, we needed to uncover the logical dependence that allows us to weave these things together to produce a non-cyclic argument.

It is worth noting that the original proof of Proposition 3.3 uses an argument about skeletons of parity complexes (treating separate parity complexes as objects of the argument). We have translated the argument so that it is internal to any given parity complex.

### 3.11 Freeness of the $\omega$-category

Having built the $\omega$-category $O(C)$ from a parity complex $C$, we now prove that it is a free category.
Definition. For each \( x \in C_p \), two subsets \( \mu(x), \pi(x) \in C^p \) are defined inductively as follows

\[
\begin{align*}
\mu(x)_p &= \{x\} & \mu(x)_{k-1} &= \mu(x)_k^\mp, & 1 \leq k \leq p \\
\pi(x)_p &= \{x\} & \pi(x)_{k-1} &= \pi(x)_k^\pm, & 1 \leq k \leq p
\end{align*}
\]

The pair \((\mu(x), \pi(x))\) is denoted by \( \langle x \rangle \).

This gives us a way to build a cell-like pair from an individual element \( x \).

Definition. An element \( x \in C_p \) is called relevant when \( \langle x \rangle \) is a cell. This amounts to saying that \( \mu(x)_n \) and \( \pi(x)_n \) are well-formed for \( 0 \leq n < p-1 \), and

\[
\mu(x)_{n-1} = \pi(x)_n^\mp, \quad \pi(x)_{n-1} = \mu(x)_n^\pm
\]

for \( 0 < n < p-1 \). Call a cell \((M, P)\) an atom when it is equal to \( \langle x \rangle \) for some \( x \). In that case we say that \((M, P)\) is atomic.

In all of our main examples, every \( \langle x \rangle \) is a cell (all elements are relevant). For the following theorem suppose that, for all \( x \), \( \mu(x) \) is tight.

Theorem 4.1 (excision of extremals). Suppose \((M, P)\) is an \( n \)-cell and \( u \in M_n (= P_n) \) is such that \((M, P) \neq \langle u \rangle \). Then \((M, P)\) can be decomposed as

\[
(M, P) = (N, Q) *_m (L, R)
\]

(30)

where \( m < n \), and \((N, Q)\) and \((L, R)\) are \( n \)-cells of dimension greater than \( m \).

This is another central result of the paper. If this algorithm is applied recursively then it shows how to present an arbitrary \( n \)-cell as a composite of atoms. Thus \( \mathcal{O}(C) \) is not only an \( \omega \)-category, but it is freely generated from its atoms.

The algorithm takes an \( n \)-cell \((M, P)\) and runs as follows.

1. Find the largest \( m < n \) with \((M_{m+1}, P_{m+1}) \neq (\mu(u)_{m+1}, \pi(u)_{m+1}) \). This amounts to discovering the highest dimension at which the criterion for being atomic does not hold. In this case, there exists \( w \in M_{m+1} \cap P_{m+1} \).

2. We want to decompose our cell by pulling off a cell of dimension \( m+1 \). Let \( x \) be a minimal element of \( M_{m+1} \) less than \( w \), and let \( y \) be a maximal element of \( M_{m+1} \) greater than \( w \). This relies on the fact that \( \mu(u)_{m+1} \) is a segment of \( M_{m+1} \), which itself relies on \( \mu(u)_{m+1} \) being tight.

\(^\ast\)Alternatively, let \((M, P)\) be a non-atomic \( n \)-cell.

\(^1\)Alternatively, find the largest \( m < n \) with \( M_{m+1} \cap P_{m+1} \neq \emptyset \).
4. If \( x \in M_{m+1} \cap P_{m+1} \) then we get a decomposition of \((M, P)\) as

\[
N = M^m \cup \{x\} \quad Q = P^{m-1} \cup ((M_m \cup x^+) \cap \neg x^-) \cup \{x\} \\
L = ((M \cap \neg \{x\}) \cup x^+) \cap \neg x^- \quad R = P \cap \neg \{x\}
\]

Notice that \((N, Q)\) is an \((m+1)\)-cell whose single element at top dimension is \(x\), and \((L, R)\) is the \(n\)-cell obtained by cutting \(x\) out of \((M, P)\).

5. If \( y \in M_{m+1} \cap P_{m+1} \) then we get a decomposition of \((M, P)\) as

\[
N = M \cap \neg \{y\} \quad Q = ((P \cap \neg \{y\}) \cup y^-) \cap \neg y^+ \\
L = M^{m-1} \cup ((P_m \cup y^-) \cap \neg y^+) \cup \{y\} \quad R = P^m \cup \{y\}
\]

This is dual to the case for \(x\). Notice that \((L, R)\) is an \((m+1)\)-cell whose single element at top dimension is \(y\), and \((N, Q)\) is the cell obtained by cutting \(y\) out of \((M, P)\).

The two hardest parts of this algorithm are parts (3) and (4). In part (3) we must show that either \(x\) or \(y\) belong to \(M_{m+1} \cap P_{m+1} \). This relies on the fact that \(\mu(x)_{m+1}\) is a segment of \(M_{m+1}\), but this follows from Proposition 2.4 and the assumption that each \(\mu(x)\) is tight. In part (4) we need to show that \((N, Q)\) and \((L, R)\) are well-defined cells. The various conditions of finiteness and well-formedness follow quite directly. The difficulty comes in showing that the movement conditions hold. We investigate the cells dimension by dimension and find that the movement conditions can be proved using Proposition 2.4 and Lemma 3.2.

How do we know that this algorithm terminates? The original text defines the rank of an \(n\)-cell \((M, P)\) to be the cardinality of \(M \cup P\). The algorithm produces two cells of smaller rank, so therefore must terminate. It is also possible to define the rank by

\[
\text{rank}(M, P) = \sum_{k=0}^{n} |M_k \cap P_k|
\]

In this case every \(n\)-cell has a rank of at least 1 since \(M_n \cap P_n\) is non-empty. A cell of rank 1 must be atomic. A cell of rank \(k > 1\) can be decomposed using excision of extremals into two cells whose individual ranks are less than or equal to \(k - 1\). Again, this is sufficient to guarantee termination.
Implementation 3.12. As already indicated, Theorem 4.1 is readily proved using the argument given above.

Remark 3.13. There may be some confusion about which conditions of tightness and receptivity are required in which theorems. To summarise, if a parity complex $C$ has the property that $\mu(x)$ is tight for every $x$, then all of the theorems hold; all other loose ends are accounted for.

At this stage, we have not shown that every $\langle x \rangle$ is a cell. In fact, we have no guarantee that any cells exist at all. This is something of a loose end, it is accounted for in the following Section.

3.14 Some comments from Section 5 of [Str91]

Section 5 begins by describing, for any two parity complexes $C$ and $D$, their product $C \times D$ and their join $C \bullet D$. That section also describes two kinds of duals for parity complexes obtained by reversing the roles of $(-)^+$ and $(-)^-$ in all dimensions or in odd dimensions only. This is of particular interest since the diagrams involved in descent are products of globes with simplexes; this is explored in Section 6.

Section 5 also addresses some issues that are as yet unresolved. First, we don’t know that any elements are relevant (consequently we don’t know if any cells exist at all). Second, Lemma 3.2 relies on the fact that all $\mu(x)$ are tight, and this was never established.

Consider the following stronger forms of Axioms 1 and 2.

For all $x$,

\[
\begin{align*}
(R1) & \quad \mu(x)^- \cup \pi(x)^+ = \mu(x)^+ \cup \pi(x)^- \quad \text{and} \\
(R2) & \quad \mu(x)^- \cap \pi(x)^+ = \mu(x)^+ \cap \pi(x) = \emptyset
\end{align*}
\]

These axioms hold for $\Delta$, $G$, and $Q$.

Remark 3.15. If a parity complex $C$ satisfies these axioms then every $\langle x \rangle$ is a cell (every $x$ is relevant). Thus all elements of $\Delta$, $G$, and $Q$ are relevant.

In a parity complex $C$, write $x < y$ when either $y \in x^+$ or $x \in y^-$. Let $\blacktriangleleft$ denote the reflexive transitive closure of the relation $\prec$. Notice that $x < y$ means there exists $z \in x^+ \cap y^-$, so this implies $x \prec y$. Hence, $x \triangleleft y$ implies $x \blacktriangleleft y$. The relation $\preceq$ compares elements of the same dimension, whereas $\blacktriangleleft$ compares elements of all dimensions. We introduce this as an optional axiom.

\[
(AS) \quad \blacktriangleleft \text{ is anti-symmetric.}
\]

This axiom holds in $\Delta$, $G$, and $Q$ where $\blacktriangleleft$ is also total.

Proposition 5.2. If each $x$ is relevant and (AS) holds then each $\mu(x)$ is tight. Thus, every $\mu(x)$ in $\Delta$, $G$, and $Q$ are tight.
Section 5 of \[Str91\] contains examples of parity complexes (motivated by certain pasting diagrams) where ◦ is not antisymmetric. This is why (AS) was not insisted upon in general.

**Remark 3.16.** It might remain unclear which conditions are required for which results (see Remark 3.13). To summarise, if a parity complex $C$ satisfies (R1) and (R2) and (AS) then every theorem and proposition covered in this paper holds. In particular, every theorem and proposition holds for the parity complexes $\Delta, \mathcal{G},$ and $\mathcal{Q}$.

**Remark 3.17.** There seems to be a fundamental relationship between parity complexes and ‘directed graphs of multiple dimension’. Some of the axioms for parity complexes are just those of the graph structure and others restrict us to graphs of a certain kind. Axioms (R1), (R2) and (AS) place further restrictions. Since we are mainly interested in examples that satisfy all of these conditions, we do not need to worry too much about this narrowing of our focus. More generally though, it would be good to know which of these conditions are associated with the graph structure of parity complexes, and which of the conditions allow for the free $\omega$-category construction. This could be the focus of some future research.

## 4 Implications for further work

### 4.1 Confirmed material

The process of formalisation reveals that Sections 1 and 2 and Theorem 4.1 can be implemented with very little deviation from the original text. This is a testament to Street’s insight and suggests that the definitions and results from those sections are well-expressed and useful tools for understanding these complicated combinatorial structures.

### 4.2 Adjusting the axioms

In a discussion with Christopher Nguyen he pointed out that Axiom 3.B is only used to prove Proposition 1.1 and Proposition 1.2 and its duals. We have commented already that Proposition 1.2 and its duals are equivalent to Axiom 3.B. A quick examination of our code then reveals that Proposition 1.2 is only used to prove that $x^+$ is tight and the disjointness condition described on p327. We haven’t investigated this in any detail, but it might be possible to replace Axiom 3.B with something slightly weaker (or stronger) but which has the same implications in the relevant proofs. This is of particular use in light of the fact that Axiom 3.B is not always preserved under products and joins (see remark on page 334).

Note that the stronger axioms (R1) and (R2) subsume Axioms 1 and 2, and the examples of primary interest also have antisymmetry for ◦. So it is worth considering the implications of adding these conditions from the very beginning. There are however good examples of parity complexes that do not have these
stronger properties. It is not yet clear if these examples are for some reason unimportant, or if parity complexes should not always be closed under product and join, or if there is even a third explanation.

4.3 Finding relevant elements

The excision of extremals shows that every cell can be presented as a composite of atomic cells. Unfortunately, not all cells are relevant, so we must explicitly describe some cells before we can use excision of extremals. And we’re not even sure yet that any cells exists. In the presence of (R1) and (R2), the problem is solved since every element is relevant and every \langle x \rangle is a cell. It is not clear whether these stronger conditions are in fact completely natural and should replace axioms 1 and 2, or whether they restrict our examples too much, or in fact, whether they are not strictly any stronger.

4.4 Why do we need \( \mu(x) \) tight?

Theorem 4.1 relies on the fact that each \( \mu(x) \) is tight and therefore a segment in the required place. This is readily proved when the ordering on \( \blacktriangleleft \) is antisymmetric and when (R1) and (R2) hold. So, if we wish to use excision of extremals, we need to have these stronger conditions holding. So we ask, is the tightness condition strictly necessary? Is there another way to ensure that \( \mu(x) \) is a segment in that proof? Or, are we happy to exclude examples of parity complexes where \( \blacktriangleleft \) is not antisymmetric?

4.5 Understanding receptivity and tightness

Section 3 of [Str91] is particularly hard to understand and the proofs there are not always straight-forward. The notions of tightness and receptivity are both a bit opaque and Lemma 3.2 is very hard to prove. This provides some motivation to closely examine Lemma 3.2 and see whether alternative arguments might be made to prove it. In particular, it is not clear whether the stronger properties in \( \Delta, \mathcal{G}, \) and \( \mathcal{Q} \) will allow for a simpler argument. Or whether the various locations where tightness and receptivity are used, a different, more elegant argument might be possible. Or whether, on closer inspection tightness and receptivity can be seen as perfectly natural properties.

5 Some lessons in coded mathematics

5.1 Duals

We were often forced to prove dual results where \( x^+ \) and \( x^- \) were interchanged, or where the direction of a movement \( M \xrightarrow{g} P \) was reversed. In these cases we were forced to explicitly restate and reprove the result, even though the underlying logic had not changed whatsoever. It would have been better if, from the beginning we had encoded plus and minus as duals to each other,
then the theorems would dualise automatically. One way to do this is to define

$$\text{faceset} : \text{bool} \rightarrow \text{C} \rightarrow \text{Ensemble C}$$

and then set $\text{minus} := \text{faceset false}$ and $\text{plus} := \text{faceset true}$. From this base-point, it should be easy to combine dual results into one.

5.2 Notation

Coq has a Notation facility which allows the user to introduce custom notation for specific expressions. We used this to make sets operations easier to read and write. For example, an expression such as $\text{Union A B}$ is displayed as $A \cup B$, and similarly for intersection, inclusion, etc. This made our code much easier to read.

5.3 Tactics

Coq has a tactic language which allows for partial automation of proofs. The language allows the user to describe simple proof strategies that can be automatically applied when little innovative thinking is required. A particular built-in tactic called intuition will automatically deal with simple proofs that require only knowledge of first-order logic. We used the tactic language to describe a proof tactic called basic that automatically applied further logical steps such as $\left( x \in A \cap B \right) \rightarrow \left( x \in A \land x \in B \right)$. In many cases this vastly simplified proofs by applying repeat (basic; intuition) to automatically prove some trivial facts.

5.4 Setoid rewrite

Whenever two terms are definitionally equal ($a = b$), we can use the rewrite command to replace $a$ with $b$ in any expression. Whenever we use a weaker notion of equality such as $\text{Same_set}$, we do not necessarily have definitional equality and we can’t replace $a$ with $b$ in every expression. This problem was solved using setoid rewrites as indicated in subsection 2.4. Given our decision to eliminate the axiom of extensionality, this facility worked very well.

5.5 Axiom of extensionality for sets

We chose to remove the axiom of extensionality because we wanted to deal with sets in a completely constructive fashion. This was a choice of style. In many ways, retaining the axiom would not have weakened our encoding and it would have made manipulation of sets much easier.

5.6 Compiling the excision of extremals algorithm

Our choice to implement sets using ensembles has made it impossible to directly compile an executable version of the excision of extremals. This is unfortunate: we have proved that such an algorithm can run but we can’t actually compile or run it without further coding. The mathematical significance of our work is
not undermined, however some more careful planning could have yielded this pleasant side-effect.

6 Conclusion

We have formalised Ross Street’s Parity Complexes up to the excision of extremals algorithm in Section 4. In particular, Sections 1 and 2 together with Theorem 4.1 are proved as indicated in the original text. Section 3 is also formalised with the same essential arguments as [Str91], but with many additional dual theorems, and a technical but meaningful change to the logical flow of Lemma 3.2 and Proposition 3.3.

We have indicated where the material is most effective at capturing the difficult combinatorics, and where future work might make improvements. We have explicitly outlined the logical dependence of the central results. We have also outlined some lessons learned in the encoding of this mathematics.

References

[AAS93] Fahd Ali Al-Agl and Richard Steiner. Nerves of multiple categories. Proc. London Math. Soc. (3), 66(1):92–128, 1993.

[Ait10] Iain R. Aitchison. The geometry of oriented cubes. 08 2010.

[Joh88] Michael Johnson. Pasting diagrams in n-categories with applications to coherence theorems and categories of paths. PhD thesis, University of Sydney, 1988.

[Joh89] Michael Johnson. The combinatorics of $n$-categorical pasting. J. Pure Appl. Algebra, 62(3):211–225, 1989.

[Coq] The Coq development team. The Coq proof assistant reference manual. LogiCal Project, 2004.

[Ste93] Richard Steiner. The algebra of directed complexes. Appl. Categ. Structures, 1(3):247–284, 1993.

[Ste04] Richard Steiner. Omega-categories and chain complexes. Homology Homotopy Appl., 6(1):175–200, 2004.

[Str87] Ross Street. The algebra of oriented simplexes. Journal of Pure and Applied Algebra, 49(3):283–335, 1987.

[Str91] Ross Street. Parity complexes. Cahiers de Topologie et Géométrie Différentielle Catégoriques, 32(4):315–343, 1991.

[Str94] Ross Street. Parity complexes: Corrigenda. Cahiers de Topologie et Géométrie Différentielle Catégoriques, 35(4):359–361, 1994.
[Str95] Ross Street. Higher categories, strings, cubes and simplex equations. *Appl. Categ. Structures*, 3(1):29–77, 1995.

[Ver08] Dominic Verity. Complicial sets characterising the simplicial nerves of strict $\omega$-categories. *Mem. Amer. Math. Soc.*, 193(905):xvi+184, 2008.