Enriched graphs with applications to organic chemistry and trees

Egbert Rijke

University of Ljubljana

April 23 2023

Prelude (First of two unrelated questions)

Consider a type family B over A. The W-type $\mathbb{W}(A, B)$ is inductively generated by one constructor

$$\mathsf{tree} : \prod_{(x:A)} (B(x) \to \mathbb{W}(A, B)) \to \mathbb{W}(A, B).$$

Question 1 (A question that shouldn't be asked)

Elements of W-types are often said to be trees. According to what concept of tree can we indeed view elements of W-types as trees?

Goals

- Understand various concepts of trees and how they are related.
- Establish a precise way in which to view elements of W-types as trees, i.e., to establish an embedding

$$\mathbb{W}(A, B) \hookrightarrow \mathsf{Tree}$$

for an appropriate notion of tree.

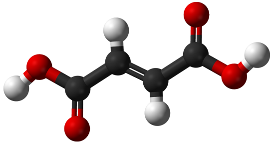
Prelude (Second of two unrelated questions)

Isomerism

There are molecules that have the same underlying graph, but nevertheless we can distinguish them based on their spatial arrangement. Such pairs of molecules are called **isomers**.

Question 2

How do we define hydrocarbons in univalent mathematics in such a way that distinct isomers are correctly distinguished?



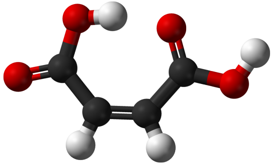


Image credit:
Ben Mills, Wikipedia (Public Domain)

Overview

Goals

- To show how to be very careful about how we define concepts in univalent mathematics.
- To demonstrate the usefulness of higher group theory.
- To break symmetries and make sure our concepts have the correct notion of equality.

Formalization

The material presented today has been formalized in the agda-unimath library:

- Graphs and enriched graphs can be found in the `graph-theory` folder.
- Trees and W-types can be found in the `trees` folder.
- The type of hydrocarbons can be found in the `organic-chemistry` folder.

Overview

Goals

- To show how to be very careful about how we define concepts in univalent mathematics.
- To demonstrate the usefulness of higher group theory.
- To break symmetries and make sure our concepts have the correct notion of equality.

Evil goals

My real goal is to convince you that trees should be allowed to have multiple edges between nodes.

Formalization

The material presented today has been formalized in the agda-unimath library:

- Graphs and enriched graphs can be found in the `graph-theory` folder.
- Trees and W-types can be found in the `trees` folder.
- The type of hydrocarbons can be found in the `organic-chemistry` folder.

Basic information on the agda-unimath library

Purpose of the agda-unimath library

- The agda-unimath library aims to be a general purpose library of formalized mathematics from a univalent point of view.
- The agda-unimath library aims to be an informative resource for mathematicians.
- The agda-unimath library contains currently only constructive univalent mathematics, but we are also open to contributions of classical univalent mathematics.

Where to find us

- Repository: `https://github.com/UniMath/agda-unimath`
- Website: `https://unimath.github.io/agda-unimath/`
- Discord: Univalent Agda (Community with over 425 members, discussing four univalent agda libraries)

Contributions in any area of mathematics are welcome.

Contributors

The agda-unimath library is the largest and fastest growing library in Agda, with over 180,000 lines of code over more than 1100 files, compiled into a markdown book of approximately 3500 pages.

28 people have contributed so far (Thank you all!!)

- Egbert Rijke (349,358 ++, 207,170 --)
- Fredrik Bakke (30,138 ++, 22,397 --)
- Éléonore Mangel (25,885 ++, 12,287 --)
- Elisabeth Bonnevier (10,220 ++, 7,258 --)
- Jonathan Prieto-Cubides (107,296 ++, 101,727 --)
- Raymond Baker (2,250 ++, 1,005 --)
- Bryan Lu (4,479 ++, 2,494 --)
- Fernando Chu (6,856 ++, 912 --)
- Elif Uskuplu (1,384 ++, 747 --)
- Victor Blanchi (10,378 ++, 1,474 --)
- ...

Contents of the agda-unimath library

Mathematical subjects in agda-unimath

- Category theory
- Commutative algebra
- Elementary number theory
- Finite group theory
- Foundation
- Graph theory
- Group theory
- Higher group theory
- Linear algebra
- Lists
- OEIS
- Order theory
- Organic chemistry
- Orth. factorization systems
- Polytopes
- Real numbers
- Ring theory
- Set theory
- Species
- Structured types
- Synthetic homotopy theory
- Trees
- Type theories
- Univalent combinatorics
- Universal algebra

Part I

Undirected graphs and trees

Traditional definitions of trees in graph theory

Definition

A **tree** is an undirected graph G satisfying any of the following equivalent conditions:

1. G is connected and acyclic.
2. G is acyclic, and a simple cycle is formed if any edge is added to G.
3. G is connected, but would become disconnected if any single edge is removed from G.
4. Any two vertices are connected by a unique path (i.e., a walk that does not repeat vertices).

- We would like to give a positive definition of trees. The condition that a graph has no cycles is not positive. Removing an edge requires decidable equality on edges.
- The fourth definition is either negative or wrong. Any two vertices in the following graph are connected by a unique path:

Unordered pairs of elements

Definition

The type of **unordered pairs of elements** in a type A is defined by

$$\text{unordered-pair}(A) := \sum_{(I:BS_2)} A^I.$$

Equality of unordered pairs

For any two unordered pairs (I, a) and (J, b) of elements in A we have

$$((I, a) = (J, b)) \simeq \sum_{(e:I \simeq J)} a \sim b \circ e.$$

Standard unordered pairs

Given two elements $a, b : A$, the **standard unordered pair** $\{a, b\}$ is defined by

$$\{a, b\} := (\mathrm{Fin}_2, (0 \mapsto a; 1 \mapsto b)).$$

Undirected graphs

Definition

An **(undirected) graph** $G \doteq (V, E)$ consists of

- A type V of **vertices**
- A type family

$$E : \text{unordered-pair}(V) \to \mathcal{U}$$

  of **half-edges**, indexed by unordered pairs.

Remark

- If $e : E(\{x, y\})$ is a half-edge, we think of it as the half-edge starting at x pointing in the direction of y. We have $E(\{x, y\}) \simeq E(\{y, x\})$ which relates the two halves of an edge.
- Graphs can have multiple half-edges between vertices.
- Graphs can have loops (half-edges from a vertex pointing to itself).
- We don't make any restriction on the truncation levels of the vertices or the edges.

Two graphs with one vertex

Example

Consider the graph with

$$V := \mathbf{1}$$
$$E(X, f) := \mathbf{1}$$

Then $E(\{*, *\}) := \mathbf{1}$ so this graph has one half-edge. The total space of all edges is

$$\sum_{(X,f)} E(X, f) \simeq BS_2.$$

There is indeed only half an edge! This graph looks like

Example

Consider the graph with

$$V := \mathbf{1}$$
$$E(X, f) := X$$

Then $E(\{*, *\}) := \mathrm{Fin}(2)$ so this graph has two half-edges. The total space of all edges is

$$\sum_{(X,f)} E(X, f) \simeq \mathbf{1}.$$

This graph has a loop on the unique vertex. This graph looks like

Equivalences of undirected graphs

Functoriality of unordered pairs

Given a map $f : A \to B$ we obtain a map

$$\text{unordered-pair}(f) : \text{unordered-pair}(A) \to \text{unordered-pair}(B)$$

given by $\text{unordered-pair}(f)(I, a) := (I, f \circ a)$. If f is an equivalence, then so is unordered-pair(f).

Equivalences of undirected graphs

An **equivalence** of undirected graphs $G \simeq H$ consists of

- An equivalence $e_V : V_G \simeq V_H$.
- A family of equivalences

$$e_E : \prod_{(p:\text{unordered-pair}(V_G))} E_G(p) \simeq E_H(\text{unordered-pair}(e_V, p)).$$

By univalence it follows that $(G = H) \simeq (G \simeq H)$.

Walks in undirected graphs

Idea

A walk from a to b in a graph G is a sequence

$$a = x_0 \overset{e_1}{\text{———}} x_1 \overset{e_2}{\text{———}} x_2 \overset{e_3}{\text{———}} x_3 \overset{e_4}{\text{———}} \cdots \overset{e_n}{\text{———}} x_n = b.$$

Definition

Given a vertex a of an undirected graph $G \doteq (V, E)$, the type family

$$\mathsf{walk}(a) : V \to \mathcal{U}$$

of **walks** in G starting at a is defined inductively by

$$\begin{aligned} \mathsf{refl} &: \mathsf{walk}(a, a) \\ \mathsf{cons} &: \prod_{((I,b):\mathsf{unordered\text{-}pair}(V))} \prod_{(i:I)} E(I, b) \to \mathsf{walk}(a, b_i) \to \mathsf{walk}(a, b_{\sigma(i)}), \end{aligned}$$

where $\sigma : I \to I$ is the swap function.

The edges on a walk

Definition

Consider an unordered pair p of vertices and an edge $e : E(p)$ in a graph G. We define the type family

$$\text{is-edge-on-walk}(e) : \prod_{(b:V)} \text{walk}(a, b) \to \mathcal{U}$$

by

$$\begin{aligned} \text{is-edge-on-walk}(e, \text{refl}) &:= \varnothing, \\ \text{is-edge-on-walk}(e, \text{cons}(p', i, e', w)) &:= ((p, e) = (p', e')) + \text{is-edge-on-walk}(e, w). \end{aligned}$$

Then we define

$$\text{edge-on-walk}(w) := \sum_{(p:\text{unordered-pair}(V))} \sum_{(e:E(p))} \text{is-edge-on-walk}(e, w).$$

Observation

One easily verifies that $\text{edge-on-walk}(w) \simeq \text{Fin}_n$ where n is the length of w.

Undirected trees

Definition

There is an obvious projection

$$\pi_w : \text{edge-on-walk}(w) \to \sum\nolimits_{(p:\text{unordered-pair}(V))} E(p).$$

We say that the walk w is a **trail** if π_w is injective.

Observation

- Being a trail is a proposition, because edge-on-walk(w) is a standard finite type, and hence a set.
- Why don't we ask that π_w is an embedding?

Definition

An **undirected tree** is an undirected graph T such that for every two vertices a and b in T the type

$$\text{trail}(a, b)$$

of trails from a to b is contractible.

A very unfortunate theorem

Theorem (R)

The type of nodes (i.e. vertices) of an undirected tree has decidable equality.

Proof.

Consider two nodes a and b in T.

1. One routinely verifies that the type $a = b$ is equivalent to the type of walks of length 0 from a to b.
2. Furthermore, every walk of length 0 is a trail since the type of edges on such walks is empty.
3. Then $a = b$ if and only if the unique trail from a to b has length 0, which is decidable. □

Symmetric identity types

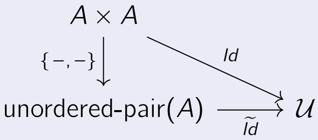
Definition

The **symmetric identity type** on a type A is the type family $\widetilde{Id} : \text{unordered-pair}(A) \to \mathcal{U}$ defined by

$$\widetilde{Id}(I, a) := \sum_{(x:A)} \prod_{(i:I)} x = a_i$$

Proposition

The symmetric identity type equips the identity type with a fully coherent $\mathbb{Z}/2$-action in the sense that the following diagram commutes



Acyclic undirected graphs

Definition

A **geometric realization** of an undirected graph $G \doteq (V, E)$ is a homotopy initial type X equipped with

$$
\begin{aligned}
i &: V \to X \\
p &: \prod_{(I,x):\mathsf{unordered\text{-}pair}(V)} E(I,x) \to \widetilde{Id}(I,x)
\end{aligned}
$$

Definition

An **acyclic undirected graph** is an undirected graph of which the geometric realization is contractible.

Example

Since there are nontrivial acyclic types, there are nontrivial acyclic undirected graphs which are not trees in the previous sense.

Half-time recap

Elements of W-types as trees?

Suddenly our quest of figuring out in what sense elements of W-types are trees looks rather impossible.

- The trees we defined above in the graph theoretical sense must have decidable equality. In particular they are sets.
- W-types can have arbitrarily high truncation levels, so we don't expect their elements to be set-level objects.

Conclusions

- We don't expect elements of W-types to be trees in the above sense.
- Let's try our luck with organic chemistry.

Part II

Enriched graphs and univalent hydrocarbons

The basic idea in the definition of the hydrocarbons

Main points

- A hydrocarbon consists of hydrogen atoms and carbon atoms.
- Hydrogen atoms form exactly one bond.
- Carbon atoms form exactly four bonds.
- To account for the spatial arrangement of a hydrocarbon, we must restrict the symmetry group of each carbon atom:
  - Any symmetry that fixes one point, must preserve the cyclic ordering of the remaining three points.
  - Any symmetry that fixes two points also fixes the remaining two points.

In other words, the symmetry group of a carbon atom in 3-space is A_4.

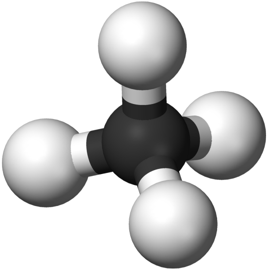


Image credit:
Ben Mills, Wikipedia (Public Domain)

Higher groups

Definition

A **higher group** G consists of a pointed connected type BG.

- The type BG is called the **delooping** of G.
- The base point of BG is called the **shape** of G.
- The **underlying type** of G is defined to be ΩBG. We often write G for ΩBG.

Definition

A G-**action** on a type X consists of a type family $Y : BG \to \mathcal{U}$ equipped with an equivalence $e : X \simeq Y(*)$. Given such a G-action on X, we define the **action** μ of G on X such that $\mu(g) : X \to X$ is the unique map equipped with a homotopy

$$\begin{array}{ccc} X & \xrightarrow{e} & Y(*) \\ {\scriptstyle \mu(g)}\downarrow & & \downarrow{\scriptstyle \mathrm{tr}_G(g)} \\ X & \xrightarrow[e]{} & Y(*). \end{array}$$

Enriched undirected graphs

Definition

Consider an undirected graph $G \doteq (V, E)$, and let $v : V$. Then we define the **neighborhood** of v by

$$\mathrm{Neighborhood}_G(v) := \sum_{(x:V)} E(\{v, x\})$$

Definition

Consider a type A and a type family B over A. An (A, B)**-enriched undirected graph** consists of

- An undirected graph $G \doteq (V, E)$.
- A map $\mathrm{sh} : V \to A$. We call $\mathrm{sh}(v)$ the **shape** of v.
- For each vertex $v : V$ an equivalence

$$e_v : B(\mathrm{sh}(v)) \simeq \mathrm{Neighborhood}_G(v).$$

∞-group actions of (A, B)-enriched graphs

Remark

Consider an (A, B)-enriched graph (V, E, sh, e).

- For every vertex $v : V$ we obtain an ∞-group

  $$BG_v := \sum_{(x:A)} \|\mathsf{sh}(v) = x\|.$$

  Its base point is $\mathsf{sh}(v)$, and we write G_v for its underlying type. G_v is called the **symmetry group** of v.
- For every vertex $v : V$ we obtain a G_v-type

  $$B : BG(v) \to \mathcal{U}$$

  given by restricting B.
- By the equivalence $B(\mathsf{sh}(v)) \simeq \mathsf{Neighborhood}(v)$ we obtain an action

  $$G_v \to (\mathsf{Neighborhood}(v) \to \mathsf{Neighborhood}(v))$$

  of the symmetry group of v on its neighborhood.

Equivalences of (A, B)-enriched graphs

Definition

An equivalence between two (A, B)-enriched graphs (V, E, sh, e) and $(V', E', \mathsf{sh}', e')$ consists of

- An equivalence $\alpha : V \simeq V'$
- A family of equivalences $\beta : E(p) \simeq E'(\text{unordered-pair}(\alpha, p))$ for each $p : \text{unordered-pair}(V)$
- A homotopy $\gamma : \mathsf{sh} \sim \mathsf{sh}' \circ \alpha$
- For each vertex v a commuting square

$$\begin{array}{ccc} B(\mathsf{sh}(v)) & \xrightarrow{\mathsf{tr}_B(\gamma(v))} & B(\mathsf{sh}'(\alpha(v))) \\ {\scriptstyle e(v)}\downarrow & & \downarrow{\scriptstyle e(\alpha(v))} \\ \text{Neighborhood}_{(V,E)}(v) & \xrightarrow[\text{Neighborhood}_{(\alpha,\beta)}(v)]{} & \text{Neighborhood}_{(V',E')}(\alpha(v)) \end{array}$$

Remark

Equivalences of (A, B)-enriched graphs are shape-preserving equivalences of graphs that also preserve the action of the symmetry group of a vertex on its neighborhood.

Broken symmetries

Theorem

Given two (A, B)-enriched graphs (V, E, sh, e) and $(V', E', \mathsf{sh}', e')$, we have an equivalence

$$((V, E, \mathsf{sh}, e) = (V', E', \mathsf{sh}, e')) \simeq ((V, E, \mathsf{sh}, e) \simeq (V', E', \mathsf{sh}', e')).$$

The sign homomorphism

Theorem (Mangel, R.)

For any $n : \mathbb{N}$ there is a map $\sigma : BS_n \to BS_2$ such that the square of group homomorphisms

$$\begin{array}{ccc} S_n & \xrightarrow{\text{sign}} & S_2 \\ \cong\downarrow & & \downarrow\cong \\ \Omega BS_n & \xrightarrow[\Omega\sigma]{} & \Omega BS_2 \end{array}$$

commutes.

Remark

The construction of σ involves:

- A functorial construction that turns an arbitrary n-element set X into a 2-element set $\sigma(X)$.
- As a functor, $\sigma : BS_n \to BS_2$ must be full (surjective on morphisms).
- In HoTT we say that σ must be 0-connected.

The alternating groups

Definition

We define BA_n as the pullback

$$\begin{array}{ccc} BA_n & \longrightarrow & \mathbf{1} \\ \downarrow & & \downarrow \\ BS_n & \xrightarrow[\sigma]{} & BS_2. \end{array}$$

Definition

- Define the type of **hydrogen atoms** to be

$$\mathcal{H} := \mathbf{1}.$$

  Note that there is a canonical map

$$\gamma_{\mathcal{H}} : \mathcal{H} \to BS_1.$$

- Define the type of **carbon atoms** to be

$$\mathcal{C} := BA_4,$$

  where BA_4 is the classifying type of the alternating group A_4. Note that there is a canonical map

$$\gamma_{\mathcal{C}} : \mathcal{C} \to BS_4.$$

Definition

The **type of hydrocarbons** is defined to be the type of (A, B)-enriched graphs where

$$\begin{aligned} A &:= \mathcal{H} + \mathcal{C} && : \mathcal{U} \\ B &:= [\gamma_{\mathcal{H}}, \gamma_{\mathcal{C}}] && : A \to \mathcal{U} \end{aligned}$$

such that

- The type of vertices is finite.
- The underlying graph is connected.
- The underlying graph has no loops.

Part III

Elements of W-types as enriched directed trees

Elements of W-types as graphs

Definition

Consider a W-type $\mathbb{W}(A, B)$. We define a relation $\in : \mathbb{W}(A, B) \to \mathbb{W}(A, B) \to \mathcal{U}$ inductively by

$$(x \in \mathrm{tree}(a, \alpha)) := \sum_{(b:B(a))} \alpha(b) = x.$$

Definition

Consider a W-type $\mathbb{W}(A, B)$. We define $\mathrm{node} : \mathbb{W}(A, B) \to \mathcal{U}$ inductively by

$$r : \prod_{(w:\mathbb{W}(A,B))} \mathrm{node}(w)$$
$$i : \prod_{(u,w:\mathbb{W}(A,B))} u \in w \to \mathrm{node}(u) \to \mathrm{node}(w).$$

and define $\mathrm{edge} : \prod_{(w:\mathbb{W}(A,B))} \mathrm{node}(w) \to \mathrm{node}(w) \to \mathcal{U}$ by

$$r : \prod_{u,w:\mathbb{W}(A,B)} u \in w \to \mathrm{edge}(w, i(r(u)), r(w))$$
$$i : \prod_{u,w:\mathbb{W}(A,B)} \prod_{(H:u\in w)} \prod_{(x,y:\mathrm{node}(u))} \mathrm{edge}(u, x, y) \to \mathrm{edge}(w, i(x), i(y)).$$

Directed trees

Definition

A **directed graph** $G \doteq (V, E)$ consists of a type V of vertices and a binary type-valued relation $E : V \to V \to \mathcal{U}$ of edges. **Walks** in directed graphs are defined analogously to the undirected case.

Definition

Consider a directed graph $G \doteq (V, E)$ equipped with a distinguised vertex $r : V$. We say that G is a **directed tree** if the type

$$\mathsf{walk}(x, r)$$

of walks from x to r is contractible for every $x : V$. The vertices of a tree are called **nodes** and the distinguished node r is called the **root**.

Example

For each element $w : \mathbb{W}(A, B)$, the directed graph $G_w := (\mathsf{node}(w), \mathsf{edge}(w))$ is a directed tree.

Equivalent definitions of directed trees

Proposition

Consider a directed graph $G = (V, E)$ equipped with a distinguished vertex r. The following conditions are equivalent:

1. *The graph G is a tree with root r.*
2. *For every vertex x there is a walk from x to r, and furthermore the type*

$$(r = x) + \sum_{(y:V)} E(x, y)$$

   is contractible.

Theorem

The type of directed trees is equivalent to the type of infinite sequences

$$\cdots \longrightarrow A_2 \longrightarrow A_1 \longrightarrow A_0$$

where A_0 is contractible. In particular, the type of nodes of a directed tree does not have to be a set.

Comparison between directed and undirected trees

- The root in a directed tree is always an isolated point. Indeed, being the root is decidable by checking that the unique trail to the root has length 0.
- In a directed tree, the type

$$\sum_{(y:V)} E(x, y)$$

  is contractible for every $x \neq r$, while $E(x, y)$ itself needs not be subterminal.
- In an undirected tree there is at most one edge on every unordered pair $\{x, y\}$ of nodes.
- Every undirected rooted tree is a directed rooted tree in a canonical way. The edges can be directed towards the root.
- Nevertheless, the undirected tree condition is much stronger than the directed tree condition, since it asks for a unique undirected trail between every two nodes, while the directed tree condition only asks for a unique trail from every node to the root.
- Elements of W-types are directed trees. However, they have a bit more structure!

Enriched directed trees

Definition

An (A, B)-**enriched directed tree** consists of

- A directed tree $T \doteq (N, E)$.
- A map $\mathrm{sh} : N \to A$. The value $\mathrm{sh}(x)$ is said to be the **shape** of the node x.
- For each node $x : N$ an equivalence

$$B(\mathrm{sh}(x)) \simeq \sum_{(y:N)} E(y, x)$$

Equality of enriched directed trees

Elements of W-types as enriched directed trees

The underlying tree of an element $w : W(A, B)$ has the structure of an (A, B)-enriched directed tree:

- We define $\mathsf{sh}(w)$ recursively by

$$\begin{aligned} \mathsf{sh}(\mathsf{tree}(a, \alpha))(r(w)) &:= a \\ \mathsf{sh}(\mathsf{tree}(a, \alpha))(i(x)) &:= \mathsf{sh}(\alpha(b), x) \end{aligned}$$

  for any $b : B(a)$ and $x : \mathsf{node}(\alpha(b))$.

- We define $B(\mathsf{sh}(w, x)) \simeq \sum_{(y:\mathsf{node}(w))} \mathsf{edge}(w, y, x)$ recursively by

$$\begin{aligned} B(a) &\simeq \sum_{u:\mathbb{W}(A,B)} u \in w \\ &\simeq \sum_{u:\mathbb{W}(A,B)} \sum_{H:u \in w} \sum_{y:\mathsf{node}(u)} \mathsf{edge}(w, i(y), r(w)) \\ &\simeq \sum_{(y:\mathsf{node}(w))} \mathsf{edge}(w, y, r(w)). \end{aligned}$$

  The recursive step is handled similarly.

Elements of W-types are enriched directed trees

Goal

There is an embedding

$$\mathbb{W}(A, B) \hookrightarrow (A, B)\text{-Enriched-Directed-Tree}$$

- This requires us to match the identity type of W-types with equivalences of enriched directed trees
- I thought it would be formalizable in a week or two. Alas, it wasn't! Due to the homotopical nature of enriched directed trees it is not possible to pattern match your way out.
- I am now developing a lot of theory of (enriched) directed trees in agda-unimath in a draft PR of ∼5000 LOC.
- Even though I have an informal argument to believe that this is true, I'm not calling the above goal a theorem until it is formalized. Too much work in homotopy type theory, including my own, hasn't been formalized and contains inaccuracies!

Oriented rooted binary trees

Definition

An **oriented binary tree** is a directed tree $T \doteq (N, E)$ equipped with a map

$$\mathsf{sh} : N \to \mathsf{Fin}_2$$

and two families of equivalences:

- For every node $x : N$ such that $\mathsf{sh}(x) = 0$ an equivalence $\varnothing \simeq \sum_{(y:N)} E(y, x)$.
- For every node $x : N$ such that $\mathsf{sh}(x) = 1$ an equivalence $\mathsf{Fin}_2 \simeq \sum_{(y:N)} E(y, x)$.

In other words, oriented binary trees are (A, B)-enriched directed trees where $A := \mathsf{Fin}_2$, and $B(0) := \varnothing$, and $B(1) := \mathsf{Fin}_2$.

Consequence of $\mathbb{W}(A, B) \hookrightarrow (A, B)$-Enriched-Directed-Tree

The type $\mathbb{W}(\mathsf{Fin}_2, (0 \mapsto \varnothing; 1 \mapsto \mathsf{Fin}_2))$ is equivalent to the finite oriented binary trees.

Binary trees

Definition

A **binary tree** is a directed tree $T \doteq (N, E)$ equipped with a map

$$\mathsf{sh} : N \to \sum_{(X:\mathcal{U})} \|\mathsf{Fin}_0 \simeq X\| \vee \|\mathsf{Fin}_2 \simeq X\|$$

and for every node $x : N$ an equivalence

$$\mathsf{sh}(x) \simeq \sum_{(y:N)} E(y, x).$$

In other words, binary trees are (A, B)-enriched directed trees where A is the type of shapes defined above, and $B(X) := X$.

Consequence of $\mathbb{W}(A, B) \hookrightarrow (A, B)$-Enriched-Directed-Tree

The type $\mathbb{W}(\sum_{(X:\mathcal{U})} \|\mathsf{Fin}_0 \simeq X\| \vee \|\mathsf{Fin}_2 \simeq X\|, X \mapsto X)$ is equivalent to the finite binary trees.

Note that being a binary tree is a property, while being an oriented binary tree is structure.

Oriented finitely branching trees

Definition

An **oriented finitely branching tree** is a directed tree $T \doteq (N, E)$ equipped with a map

$$\mathsf{sh} : N \to \mathbb{N}$$

and for every node $x : N$ an equivalence

$$\mathsf{Fin}_{\mathsf{sh}(x)} \simeq \sum_{(y:N)} E(y, x).$$

In other words, oriented finitely branching trees are (A, B)-enriched directed trees where $A := \mathbb{N}$, and $B(n) := \mathsf{Fin}_n$.

Consequence of $\mathbb{W}(A, B) \hookrightarrow (A, B)$-Enriched-Directed-Tree

The type $\mathbb{W}(\mathbb{N}, \mathsf{Fin})$ is equivalent to the finite oriented finitely branching trees.

Finitely branching trees

Definition

A **finitely branching tree** is a directed tree $T \doteq (N, E)$ equipped with a map

$$\mathrm{sh} : N \to \mathbb{F}$$

and for every node $x : N$ an equivalence

$$\mathrm{sh}(x) \simeq \sum_{(y:N)} E(y, x).$$

In other words, finitely branching trees are (A, B)-enriched directed trees where $A := \mathbb{F}$, and $B(X) := X$.

Consequence of $\mathbb{W}(A, B) \hookrightarrow (A, B)$-Enriched-Directed-Tree

The type $\mathbb{W}(\mathbb{F}, X \mapsto X)$ is equivalent to the type of finite trees.

Note again that being a finitely branching tree is a property, while being an oriented finitely branching tree is structure.

Conclusion

Closing remarks

Enrichment does not break symmetries if and only if B is a univalent type family over A. Indeed, in this case the enrichment data is a proposition. In particular, if $\mathbb{W}(A, B)$ is an extensional W-type in the sense of Gylterud, then $\mathbb{W}(A, B)$ is the type of all well-founded directed trees with branching in the subuniverse $A \hookrightarrow \mathcal{U}$.

Thank you for your attention!