Single-set cubical categories and their formalisation with a proof assistant

Philippe Malbos[1], Tanguy Massacrier[1], Georg Struth[2]

[1]Université Claude Bernard Lyon 1

[2]University of Sheffield

https://arxiv.org/abs/2401.10553

https://www.isa-afp.org/entries/CubicalCategories.html

Single-set cubical categories

- Cubical sets and categories are a categorical description of cubes, their faces and their compositions. They provide a language used for:
  - studying homotopy,
  - studying higher-dimensional rewriting,
  - modelling concurrency using higher dimensional automata,
  - modelling homotopy type theory.
- We want to study the process of formalisation. We present an alternative, single-set, description of cubical categories.
- A single-set approach is easier to formalise and to compute with. We use the proof assistant **Isabelle** to formalise these categories and to help us find their definition.

The geometry of computations is cubical

We want to study computations in an algebraic structure using rewriting.

Example: expressions in $(\mathbb{N}, +)$.

We replace equalities by oriented arrows:

$$1 + 1 \longrightarrow 2$$

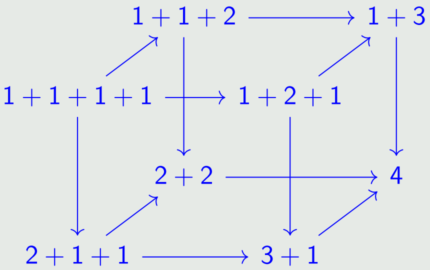
Compare relations:

$$\begin{array}{ccccc} & & 1+1+1 & & \\ & \swarrow & & \searrow & \\ 2+1 & & \Rightarrow & & 1+2 \\ & \searrow & & \swarrow & \\ & & 3 & & \end{array}$$

The geometry of computations is cubical

Example: expressions in $(\mathbb{N}, +)$.

Compare relations between relations:



- Cubical categories provide a language for formalising computations.
- We apply it to abstract rewriting systems in a forthcoming work.

Cubical categories

Definition (Brown, Higgins, 1981):

- A **cubical ω-category** is the data of:
  - a set $\mathcal{C}_k$ of k-cells for $k \in \mathbb{N}$,
  - face maps $\partial^{\alpha}_{k,i} : \mathcal{C}_k \to \mathcal{C}_{k-1}$ for $1 \leq i \leq k$ and $\alpha \in \{-,+\}$,
  - degeneracies $\epsilon_{k,i} : \mathcal{C}_{k-1} \to \mathcal{C}_k$ for $1 \leq i \leq k$,
  - compositions $\star_{k,i} : \mathcal{C}_k \times_{k,i} \mathcal{C}_k \to \mathcal{C}_k$ for $1 \leq i \leq k$,

  satisfying some compatibility conditions.

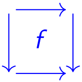
$$\mathcal{C}_0 \underset{\partial^+_{1,1}}{\overset{\partial^-_{1,1}}{\leftleftarrows}} \mathcal{C}_1 \underset{\partial^+_{2,2}}{\overset{\partial^-_{2,1}}{\leftleftarrows}} \mathcal{C}_2 \qquad \dots \qquad \mathcal{C}_{k-1} \underset{\partial^+_{k,k}}{\overset{\partial^-_{k,1}}{\leftleftarrows}} \mathcal{C}_k \qquad \dots$$

- A cubical n-category is the same but we forget the structure after dimension n.

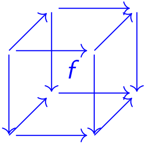
Cells and dimensions

The shapes of cells are as follows:

- 0-cells are points,
- 1-cells, $x \xrightarrow{f} y$
- 2-cells,

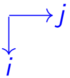


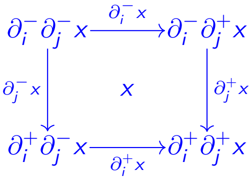
- 3-cells,



Face maps

In every direction i, there are two faces: $\partial^-_{k,i} = \partial^-_i$ (source) and $\partial^+_{k,i} = \partial^+_i$ (target).

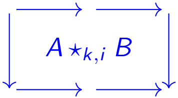




Compositions

When faces $\partial^{+}_{k,i}A = \partial^{-}_{k,i}B$ of two k-cells coincide

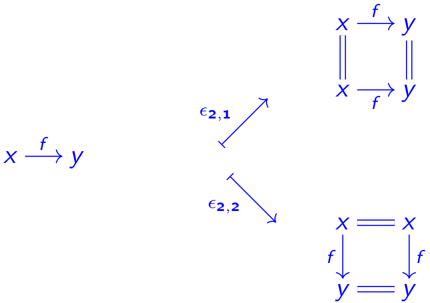
then we can $\star_{k,i}$-compose them by 'glueing' them along direction i.



Degeneracies

From a 0-cell x we get an identity 1-cell.

From a 1-cell we get two degenerate 2-cells.



The degeneracies $\epsilon_{k,i}$ are the identities for the $\star_{k,i}$-composition.

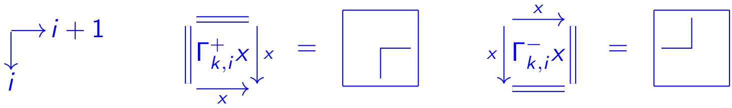
The categories Cub_n

A functor $F : \mathcal{C} \to \mathcal{D}$ is a family of maps $F_k : \mathcal{C}_k \to \mathcal{D}_k$ preserving the structure (face maps, degeneracies, compositions).

This defines the categories Cub_n for $n \in \mathbb{N} \cup \{\omega\}$.

Connections and equivalence with globular categories

Connections are 'twisted' degeneracies. They bend the wires between different directions.



By adding functors, this defines the categories Cub_n^{Γ} for $n \in \mathbb{N} \cup \{\omega\}$.

Theorem (Al-Agl, Brown, Steiner, 2001)

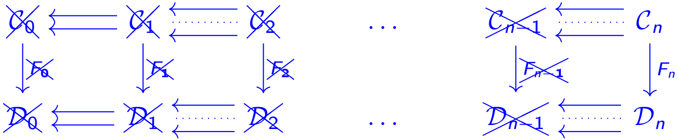
For $n \in \mathbb{N} \cup \{\omega\}$, $\mathrm{Cub}_n^{\Gamma} \simeq \mathrm{Cat}_n$.

From classical to single-set

- Alternative model of cubical categories.
- Idea: the low dimensional cells are already encoded in higher dimensions as identity cells, the degeneracies.

From classical to single-set

- Easier to compute with, because we only have one set containing all the cells. We don't treat the cells of different dimensions separately, but all at once.
    - *For instance:* a functor is only a function $F : \mathcal{S} \to \mathcal{T}$ respecting face maps, symmetries and compositions.



- Easier to formalise in a proof assistant, because the dimension of cells is not captured with types but functionally through fixed-point properties.
- We formalised single-set cubical categories in Isabelle. Isabelle used to develop their axiomatisation.

Dimension 1

Definition (MacLane, 1971):

A **single-set category** $\mathcal{S}$ is the data of:

- a set $\mathcal{S}$ of cells,
- face maps $\delta^- : \mathcal{S} \to \mathcal{S}$ (source) and $\delta^+ : \mathcal{S} \to \mathcal{S}$ (target),
- partially defined composition map $\otimes$ from $\mathcal{S} \times \mathcal{S}$ to $\mathcal{S}$,

where $x \otimes y$ is defined if and only if $\delta^+ x = \delta^- y$, in which case

$$\delta^-(x \otimes y) = \delta^- x, \qquad \delta^+(x \otimes y) = \delta^+ y,$$

$$x \otimes \delta^+ x = x, \qquad \delta^- x \otimes x = x,$$

$$x \otimes (y \otimes z) = x \otimes (y \otimes z)$$

$$\delta^- \delta^- x = \delta^- x = \delta^+ \delta^- x,$$

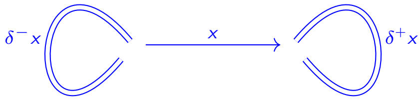
$$\delta^+ \delta^+ x = \delta^+ x = \delta^- \delta^+ x.$$

Dimension 1

The fixed points of the face maps

$$\mathcal{S}^{\delta} = \{x \in \mathcal{S} \mid \delta^{-}x = x\} = \{x \in \mathcal{S} \mid \delta^{+}x = x\}$$

are the identity arrows for the composition, and they correspond to the 0-cells in classical categories.



Single-set cubical categories

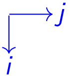
Definition:

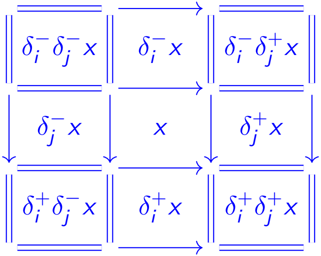
- A **single-set cubical ω-category** $\mathcal{S}$ is the data of:
  - a family of single-set categories $(\mathcal{S}, \delta_i^-, \delta_i^+, \otimes_i)_{i \geq 1}$,
  - symmetries $s_i : \mathcal{S} \to \mathcal{S}$ for $i \geq 1$,
  - reverse symmetries $\tilde{s}_i : \mathcal{S} \to \mathcal{S}$ for $i \geq 1$,

  satisfying some compatibility conditions.
- A single-set cubical n-category is the same but we forget the structure after dimension n.

Faces

The faces are themselves cells:





Symmetries

Let's recall that every 1-cell can be seen as a degenerate 2-cell in two ways.

We need a way to identify them: symmetries.

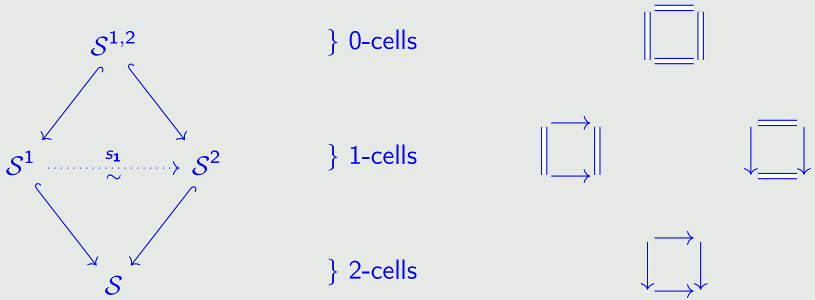
The symmetries exchange directions: s_i sends identities for $\otimes_i$-composition to identities for $\otimes_{i+1}$-composition.

Lattice of fixed points

How do we recover low-dimensional cells?

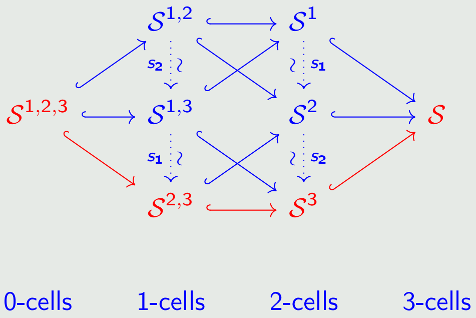
Define $\mathcal{S}^i = \{x \in \mathcal{S} \mid \delta_i^{\pm} x = x\}$, the set of fixed points for the face maps in direction i, and $\mathcal{S}^I = \bigcap_{i \in I} \mathcal{S}^i$. We get inclusions forming a lattice.

Example in dimension 2.



Lattice of fixed points

Example in dimension 3.



Lattice of fixed points

Example in dimension ω.

$$\mathcal{S}$$

$$\mathcal{S}^{1,2,3,\dots} \hookrightarrow \mathcal{S}^{2,3,4,\dots} \hookrightarrow \mathcal{S}^{3,4,5,\dots} \hookrightarrow \dots$$

0-cells 1-cells 2-cells ω-cells

The categories SCub_n and SCub_n^γ

- A functor $F : \mathcal{S} \to \mathcal{T}$ is a map preserving the structure.
  This defines the categories SCub_n for $n \in \mathbb{N} \cup \{\omega\}$.
- As before, we can add connections to the structure: maps $\gamma_i^\alpha : \mathcal{S} \to \mathcal{S}$ for $i \geq 1$ and $\alpha \in \{-, +\}$ satisfying some conditions.
  Adding functors, we get categories SCub_n^γ for $n \in \mathbb{N} \cup \{\omega\}$.

Equivalence with classical cubical categories

- We can recover the low-dimensional structure using the fixed points of the face maps, hence:

Theorem

For $n \in \mathbb{N} \cup \{\omega\}$, $\mathsf{SCub}_n \simeq \mathsf{Cub}_n$ *and* $\mathsf{SCub}_n^\gamma \simeq \mathsf{Cub}_n^\Gamma$.

- Isabelle was used to find the definition of single-set cubical categories and to prove the above equivalence.

Experimental mathematics with Isabelle

How did we find the definition of single-set cubical categories?

- We tried to copy the classical cubical axioms. Example:

$$\partial^{\alpha}_{k,i+1}\Gamma^{\alpha}_{k,i}x = x \quad \text{for } x \text{ in } \mathcal{C}_{k-1} \qquad \rightsquigarrow \qquad \delta^{\alpha}_{i+1}\gamma^{\alpha}_{i}x = s_i x \quad \text{for } x \text{ in } \mathcal{S}^i$$

↑ symmetries had to be introduced

- We added all the axioms needed to show the equivalence of categories.
- We used Isabelle automated proof search tools to show the redundancy of some axioms, and removed them from the definition. Example:

$$\partial^{\alpha}_{k,i}\Gamma^{-\alpha}_{k,i} = \epsilon_{k-1,i}\partial^{\alpha}_{k-1,i} \qquad \rightsquigarrow \qquad \text{not needed in single-set}$$

- It simplified the proof of the equivalence.

Conclusion and perspectives

- We introduced a single-set axiomatisation of cubical categories.
- We showed they are equivalent to classical cubical categories.
- We implemented their definition in Isabelle.
- We can require our cells to be invertible. This gives the categories $\mathsf{Cub}^{\Gamma}_{(n,p)}$ and $\mathsf{SCub}^{\gamma}_{(n,p)}$, to which we extended our results.
- These (n, p)-categories are used in rewriting. Indeed an (ω, p)-category $\mathcal{C}$ presents the p-category obtained by quotienting by the equivalence generated by the $(p+1)$-cells.

$$x \longrightarrow \quad \longleftarrow \quad \longrightarrow \quad \dots \quad \longrightarrow y$$

Finding such presentations by free well-behaved (ω, p)-categories, called resolutions, is a goal of higher-dimensional rewriting. In a forthcoming work we study the case $p = 0$, that is abstract rewriting systems.

Conclusion and perspectives

Next we want to study higher-dimensional rewriting properties in the single-set cubical setting:

- normalisation strategies, which are deterministic choices of reduction paths from one cell to another reduced one,
- proofs of Church-Rosser theorem and Newman's lemma, which characterise confluence properties of rewriting systems,
- polygraphic resolutions of higher categories.

Thank you.

Appendix 1: single-set cubical ω-category

A **single-set cubical ω-category** consists of a family of single-set categories $(\mathcal{S}, \delta_i^-, \delta_i^+, \otimes_i)_{i \in \mathbb{N}_+}$ with *symmetry maps* $s_i : \mathcal{S} \to \mathcal{S}$ and *reverse symmetry maps* $\tilde{s}_i : \mathcal{S} \to \mathcal{S}$ for each $i \in \mathbb{N}_+$. These satisfy, for all $w, x, y, z \in \mathcal{S}$ and $i, j \in \mathbb{N}_+$,

- **(i)** $\delta_i^\alpha \delta_j^\beta = \delta_j^\beta \delta_i^\alpha$ if $i \neq j$,
- **(ii)** $\delta_i^\alpha (x \otimes_j y) = \delta_i^\alpha x \otimes_j \delta_i^\alpha y$ if $i \neq j$ and $\Delta_j(x, y)$,
- **(iii)** $(w \otimes_i x) \otimes_j (y \otimes_i z) = (w \otimes_j y) \otimes_i (x \otimes_j z)$ if $i \neq j$, $\Delta_i(w, x)$, $\Delta_i(y, z)$, $\Delta_j(w, y)$ and $\Delta_j(x, z)$,
- **(iv)** $s_i(\mathcal{S}^i) \subseteq \mathcal{S}^{i+1}$ and $\tilde{s}_i(\mathcal{S}^{i+1}) \subseteq \mathcal{S}^i$,
- **(v)** $\tilde{s}_i s_i x = x$ and $s_i \tilde{s}_i y = y$ if $x \in \mathcal{S}^i$ and $y \in \mathcal{S}^{i+1}$,
- **(vi)** $\delta_j^\alpha s_j x = s_j \delta_{j+1}^\alpha x$ and $\delta_i^\alpha s_j x = s_j \delta_i^\alpha x$ if $i \neq j, j+1$ and $x \in \mathcal{S}^j$,
- **(vii)** $s_i(x \otimes_{i+1} y) = s_i x \otimes_i s_i y$ and $s_i(x \otimes_j y) = s_i x \otimes_j s_i y$ if $j \neq i, i+1$, $x, y \in \mathcal{S}^i$ and $\Delta_j(x, y)$,
- **(viii)** $s_i x = x$ if $x \in \mathcal{S}^i \cap \mathcal{S}^{i+1}$,
- **(ix)** $s_i s_j x = s_j s_i x$ if $|i - j| \geq 2$ and $x \in \mathcal{S}^i \cap \mathcal{S}^j$,
- **(x)** $\exists k \in \mathbb{N} \ \forall i \geq k+1, \ x \in \mathcal{S}^i$.

Appendix 2: with connections

A **single-set cubical ω-category with connections** is a single-set cubical ω-category $\mathcal{S}$ with *connection maps* $\gamma_i^\alpha : \mathcal{S} \to \mathcal{S}$, for all $i \in \mathbb{N}_+$ and $\alpha \in \{-,+\}$. These satisfy, for all $i, j \in \mathbb{N}_+$,

(i) $\delta_j^\alpha \gamma_j^\alpha x = x$, $\delta_{j+1}^\alpha \gamma_j^\alpha x = s_j x$ and $\delta_i^\alpha \gamma_j^\beta x = \gamma_j^\beta \delta_i^\alpha x$ if $i \neq j, j+1$ and $x \in \mathcal{S}^j$,

(ii) if $j \neq i, i+1$ and $x, y \in \mathcal{S}^i$, then

$$\Delta_{i+1}(x,y) \Rightarrow \gamma_i^+(x \otimes_{i+1} y) = (\gamma_i^+ x \otimes_{i+1} s_i x) \otimes_i (x \otimes_{i+1} \gamma_i^+ y),$$
$$\Delta_{i+1}(x,y) \Rightarrow \gamma_i^-(x \otimes_{i+1} y) = (\gamma_i^- x \otimes_{i+1} y) \otimes_i (s_i y \otimes_{i+1} \gamma_i^- y),$$
$$\Delta_j(x,y) \Rightarrow \gamma_i^\alpha(x \otimes_j y) = \gamma_i^\alpha x \otimes_j \gamma_i^\alpha y,$$

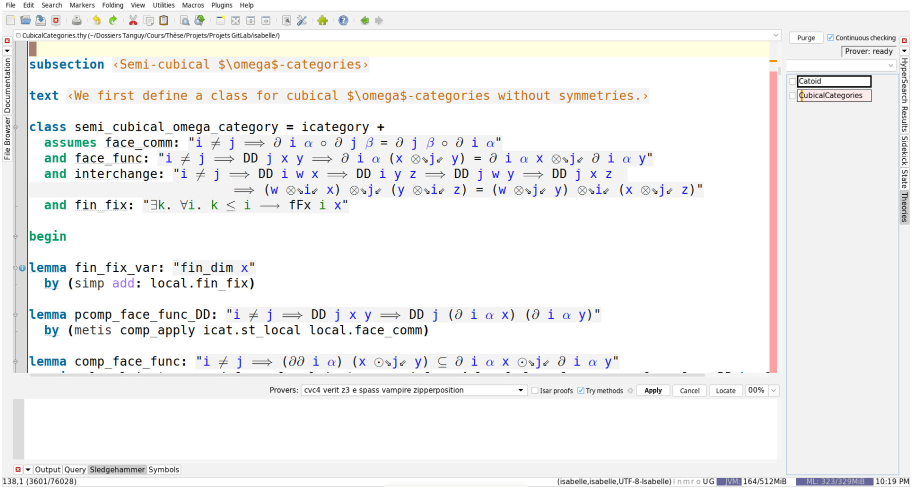
(iii) $\gamma_i^\alpha x = x$ if $x \in \mathcal{S}^{i,i+1}$,

(iv) $\gamma_i^+ x \otimes_{i+1} \gamma_i^- x = x$ and $\gamma_i^+ x \otimes_i \gamma_i^- x = s_i x$ if $x \in \mathcal{S}^i$,

(v) $\gamma_i^\alpha \gamma_j^\beta x = \gamma_j^\beta \gamma_i^\alpha x$ if $|i-j| \geq 2$ and $x \in \mathcal{S}^{i,j}$,

(vi) $s_{i+1} s_i \gamma_{i+1}^\alpha x = \gamma_i^\alpha s_{i+1} x$ if $x \in \mathcal{S}^{i,i+1}$.

Appendix 3: Isabelle



Appendix 3: Isabelle

File Edit Search Markers Folding View Utilities Macros Plugins Help

CubicalCategories.thy (~/Dossiers Tanguy/Cours/Thèse/Projets/Projets GitLab/isabelle/)

```
text ‹Next we define a class for cubical $\omega$-categories.›

class cubical_omega_category = semi_cubical_omega_category + symmetry_ops +
  assumes sym_type: "σσ i (face_fix i) ⊆ face_fix (i + 1)"
  and inv_sym_type: "ϑϑ i (face_fix (i + 1)) ⊆ face_fix i"
  and sym_inv_sym: "fFx (i + 1) x ⟹ σ i (ϑ i x) = x"
  and inv_sym_sym: "fFx i x  ⟹ ϑ i (σ i x) = x"
  and sym_face1: "fFx i x ⟹ ∂ i α (σ i x) = σ i (∂ (i + 1) α x)"
  and sym_face2: "i ≠ j ⟹ i ≠ j + 1 ⟹ fFx j x ⟹ ∂ i α (σ j x) = σ j (∂ i α x)"
  and sym_func: "i ≠ j ⟹ fFx i x ⟹ fFx i y ⟹ DD j x y ⟹
                σ i (x ⊗↘j↙ y) = (if j = i + 1 then σ i x ⊗↘i↙ σ i y else σ i x ⊗↘j↙ σ i y)"
  and sym_fix: "fFx i x ⟹ fFx (i + 1) x ⟹ σ i x = x"
  and sym_sym_braid: "diffSup i j 2 ⟹ fFx i x ⟹ fFx j x  ⟹ σ i (σ j x) = σ j (σ i x)"

begin

text ‹First we prove variants of the axioms.›

lemma sym_type_var: "fFx i x ⟹ fFx (i + 1) (σ i x)"
  by (meson image_subset_iff local.face_fix_prop local.sym_type)
```

Purge | Continuous checking | Prover: ready

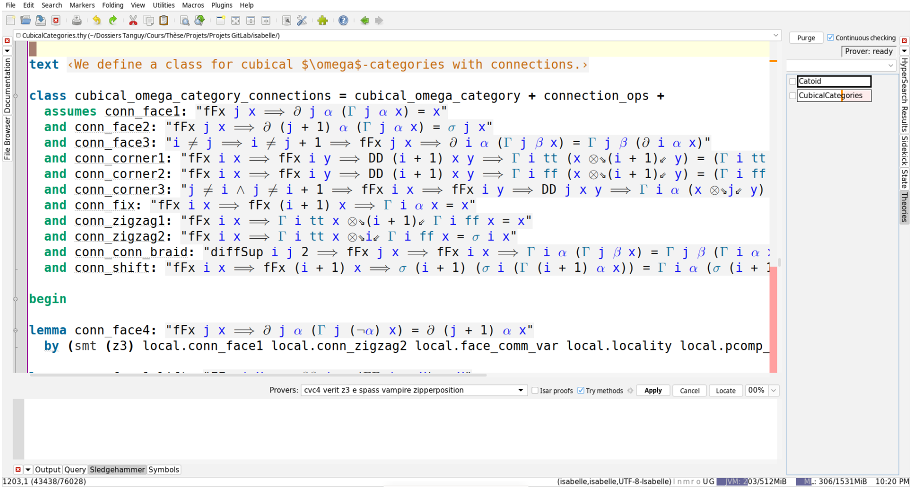
Catoid
CubicalCategories

Provers: cvc4 verit z3 e spass vampire zipperposition | Isar proofs | Try methods | Apply | Cancel | Locate | 00%

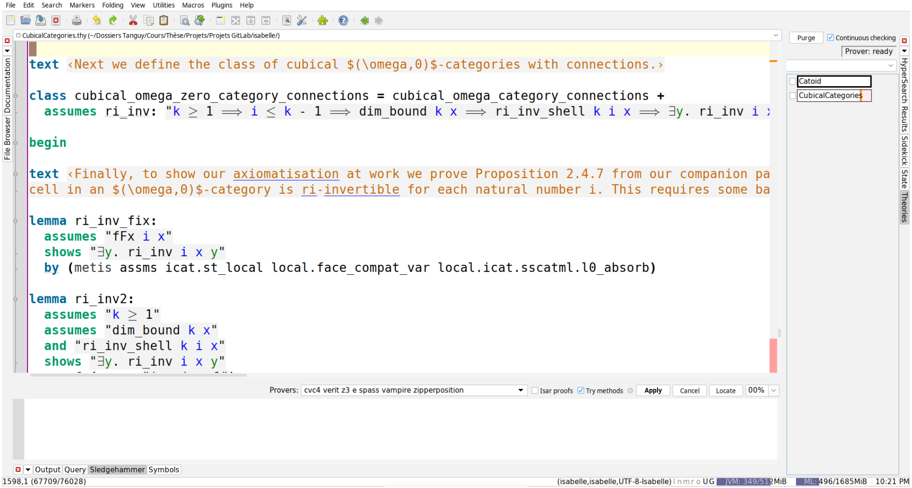
Output Query Sledgehammer Symbols

334,1 (11052/76028) (isabelle,isabelle,UTF-8-Isabelle) | n m r o UG JVM: 160/512MiB ML: 237/990MiB 10:20 PM

Appendix 3: Isabelle



Appendix 3: Isabelle



Appendix 3: Isabelle