Comparing Quotient- and Symmetric Containers

Philipp Joram Niccolò Veltri

Tallinn University of Technology, Estonia

TYPES Conference, June 2024

Containers: syntax for polymorphic programs

We want to study various notions of containers in HoTT.

- containers define syntax for (polymorphic) data types.
- morphisms of containers are syntax for transformations of such data
- A functor $[\![-]\!]$ gives semantics to containers & their morphisms
- $[\![-]\!]$ often lands in endofunctors and natural transformations

Today:

Quotient containers, and how they relate to symmetric containers.

Quotient containers

Idea: Data at certain positions are identified.

A quotient container $(S \triangleleft P/G)$ is...

- a set of *shapes* S,
- a set of *positions* $P(s)$ for each shape s
- a group of permissible *symmetries* $G_s \leq \mathrm{Aut}(P(s))$

Extension of a quotient container: an endofunctor $[\![S \triangleleft P/G]\!]_/ : \mathsf{Set} \to \mathsf{Set}$:

$$[\![S \triangleleft P/G]\!]_/(X) := \sum_{s:S} \frac{P(s) \to X}{\sim_s}$$

$$u \sim_s v :\Longleftrightarrow \begin{tikzcd} P(s) \arrow[rr, "\exists g : G_s"] \arrow[dr, "u"'] & & P(s) \arrow[dl, "v"] \\ & X & \end{tikzcd}$$

Example: unordered pairs

The container of *unorderd* pairs has

one shape, and

$$\mathsf{UPair} := (\, \mathbf{1} \lhd \mathbf{2} \,/\, \mathsf{Aut}(\mathbf{2})\,)$$

two positions

...that identify data by swapping

The extension of UPair gives sets of unordered pairs:

$$[\![\mathsf{UPair}]\!]_{/}(X) = \sum_{s:\mathbf{1}} (\mathbf{2} \to X)/{\sim_s} = \frac{X^2}{(x,y) \sim (y,x)}$$

More examples: Finite multisets, cyclic lists.
Non-examples: Finite sets.

Category of quotient containers

Quotient containers[1] are the objects of a category $\mathcal{Q}$.
Containers are related by *premorphisms*:

$$(\underbrace{u}_{S \to T}, \underbrace{f}_{Q(us) \to P(s) \text{ (contravariantly)}}, \underbrace{\varphi}_{\text{related positions stay related}}) : (S \triangleleft P/G) \rightharpoonup (T \triangleleft Q/H)$$

Morphisms of $\mathcal{Q}$ are premorphisms quotiented by some relation.

[1] Abbott et al., "Constructing Polymorphic Programs with Quotient Types".

Quotient containers in HoTT

We give a definition of $\mathcal{Q}$ in HoTT:

- We assume shapes and positions to be (homotopy) sets, not arbitrary types
- Define extension functor using set-quotients (i.e. via a HIT)

Quotients of sets are nice to work with[2] in HoTT, so...

[2] *Terms and conditions apply*

Properties of quotient containers

Some immediate results, for example:

- $\mathcal{Q}$-Iso(UPair, UPair) is contractible
- $(\mathsf{UPair} = \mathsf{UPair}) \simeq \mathbf{2}$
- $\mathcal{Q}$ is *not* a univalent category

With a little more work, we can port traditional proofs to HoTT:

Theorem
Each $[\![Q]\!]_{/}$ is a left Kan extension.
Universal property of Kan extensions implies that $[\![-]\!]_{/}$ is fully faithful.

A better presentation

These definitions and proofs are quite involved.
There is entirely too much reasoning about symmetry groups.
HoTT promises to help study such symmetries.
Questions:

1. Can we find a more intensional presentation of containers with symmetries?
2. Are quotient containers a subclass of those?

Symmetric containers

A *symmetric container*[3] is...

$$(\; S \triangleleft P \;)$$

an h-groupoid of shapes

a family of h-sets

- morphisms are *what you expect™*
- define a univalent bicategory $\mathcal{S}$ (2-cells = homotopies of morphisms)
- $[\![S \triangleleft P]\!](X) := \sum_s P(s) \to X$ is a pseudofunctor Gpd $\to$ Gpd

[3] Gylterud, "Symmetric Containers".

Unordered pairs, take II

The symmetric container of *unordered pairs* is $\mathrm{UPair} :\equiv (\mathbf{B}(\mathrm{Aut}(\mathbf{2})) \triangleleft U)$, where

- $\mathbf{B}(\mathrm{Aut}(\mathbf{2}))$ has one point $\bullet$ and one non-trivial path $(\mathrm{swap} : \bullet = \bullet)$
- U is defined by induction:

$$U(\bullet) :\equiv \mathbf{2}$$

$$U(\mathrm{swap}) :\equiv \underbrace{\boxed{\mathrm{ua}\,\mathrm{not}}}_{\text{a path } \mathbf{2} = \mathbf{2}}$$

The path space of shapes encodes all symmetries!

For $x, y : [\![\mathrm{UPair}]\!](X)$:

$$(x = y) = \sum_{\sigma : \mathrm{Aut}(\mathbf{2})} \mathrm{snd}(x) = \mathrm{snd}(y) \circ \sigma$$

Delooping construction

This gives us an idea how to associate a symmetric container to any quotient container.

Definition

Define $\mathbf{B}(S \triangleleft P/G) := (S^{\uparrow} \triangleleft P^{\uparrow})$ where

$$S^{\uparrow} := \sum_{s:S} \mathbf{B}\, G_s$$

$$P^{\uparrow} := \begin{cases} (s : S) & \mapsto P(s) : \mathcal{U} \\ (g : G_s) & \mapsto \mathsf{ua}(g) : P(s) =_{\mathcal{U}} P(s) \end{cases}$$

Note: A version of $P^{\uparrow}$ appears in the definition of $[\![-]\!]_{/}$ as a Kan extension.

Extension of $\mathbf{B}Q$ I

$[\![\mathbf{B}(Q)]\!]$ maps groupoids to groupoids. How does it relate to $[\![Q]\!]_{/}$?

Theorem

The diagram
$$\begin{tikzcd} \mathcal{S} \arrow[r, "{[\![-]\!]}"] & \mathrm{Gpd} \to \mathrm{Gpd} \arrow[d, "\lambda F.\, \|F(-)\|"] \\ \mathcal{Q} \arrow[u, "\mathbf{B}"] \arrow[r, "{[\![-]\!]_{/}}"'] & \mathrm{Set} \to \mathrm{Set} \end{tikzcd}$$
of functions in $\mathcal{U}$ commutes.

Extension of $\mathbf{B}Q$ II

Proof.

Idea: $S^{\uparrow}$ has pointed connected components.

$$
\begin{aligned}
\underbrace{\| \llbracket \mathbf{B}Q \rrbracket(X) \|}_{\text{set truncation}} &\simeq \Big\| \sum_{s:S} \sum_{g:\mathbf{B}(G_s)} P^{\uparrow}(s,g) \to X \Big\| \\
&\simeq \sum_{s:S} \Big\| \sum_{g:\mathbf{B}(G_s)} P^{\uparrow}(s,g) \to X \Big\| && (S \text{ is a set}) \\
&\simeq \sum_{s:S} \frac{P(s) \to X}{\sim_{G_s}} && (\text{by computation}) \\
&\simeq \llbracket Q \rrbracket_{/}(X) && \square
\end{aligned}
$$

$\mathbf{B}(-)$ as a functor?

Does $\mathbf{B}(-)$ extend to an action on $\mathcal{Q}$-morphisms? Not obviously:

- maps of shapes $\sum_s \mathbf{B} G_s \to \sum_t \mathbf{B} H_t$ require a group homomorphism $G_s \to H_t$
- $\mathcal{Q}$-morphisms are equivalence classes
- For a premorphism $(u, f, \varphi) : (S \triangleleft P/G) \rightharpoonup (T \triangleleft Q/H)$, $\varphi : G_s \to H_{ut}$ is not necessarily a group homomorphism

This means that $\mathbf{B}(-)$ cannot be directly defined by induction on $\mathcal{Q}$-morphisms.

Remedies

- Work harder:
  - give alternative presentation of $\mathcal{Q}$-morphisms
  - get rid of quotiented homsets this way
  - maybe it is functorial after all?
- Change definitions:
  - add properties: make premorphisms preserve group structure of symmetries
  - all practical examples satisfy this property
  - is this a parametricity condition?

More questions

Can we go the other way $\mathbf{U} : \mathcal{S} \to \mathcal{Q}$?

- Yes, assuming a form of choice: "Every groupoid has *pointed* connected components."
- In this case, $\mathbf{U}$ is a retract of $\mathbf{B}$: $\mathbf{U}(\mathbf{B}(Q)) = Q$.

Question to TYPES:

Is the above a known choice principle?

Conclusion

- Quotient containers are symmetric containers, in some way
- Less clear how morphisms relate
- Some interesting questions of reverse mathematics arise
- Cubical Agda has been helpful in figuring this all out

Thank you!