Strictly Associative Group Theory using Univalence

Alex Rice[1]

University of Cambridge

HoTT/UF 2023

Outline

Motivation

```
InvUniqueLeft : ∀ {ℓ} (𝒢 : Group ℓ) → Type ℓ
InvUniqueLeft 𝒢 = ∀ g h → h · g ≡ 1g → h ≡ inv g
  where
  open GroupStr (𝒢 .snd)
```

Motivation

```
InvUniqueLeft : ∀ {ℓ} (𝒢 : Group ℓ) → Type ℓ
InvUniqueLeft 𝒢 = ∀ g h → h · g ≡ 1g → h ≡ inv g
  where
  open GroupStr (𝒢 .snd)

inv-unique-left : ∀ {ℓ} (𝒢 : Group ℓ) → InvUniqueLeft 𝒢
inv-unique-left 𝒢 g h p =
  h               ≡⟨ sym (·IdR h) ⟩
  h · 1g          ≡⟨ cong (h ·_) (sym (·InvR g)) ⟩
  h · (g · inv g) ≡⟨ ·Assoc h g (inv g) ⟩
  (h · g) · inv g ≡⟨ cong (_· inv g) p ⟩
  1g · inv g      ≡⟨ ·IdL (inv g) ⟩
  inv g   ∎
  where
    open GroupStr (𝒢 .snd)
```

Motivation

```
InvUniqueLeft : ∀ {ℓ} (𝒢 : Group ℓ) → Type ℓ
InvUniqueLeft 𝒢 = ∀ g h → h · g ≡ 1g → h ≡ inv g
  where
  open GroupStr (𝒢 .snd)

inv-unique-left-strict : ∀ {ℓ} (𝒢 : Group ℓ) → InvUniqueLeft 𝒢
inv-unique-left-strict 𝒢 = strictify InvUniqueLeft
  λ g h p →
    h · 1g          ≡⟨ cong (h ·_) (sym (·InvR g)) ⟩
    h · g · inv g   ≡⟨ cong (_· inv g) p            ⟩
    1g · inv g      ∎
    where
      open GroupStr (RSymGroup 𝒢 .snd)
      open import Groups.Reasoning 𝒢 using (strictify)
```

Strictify

- Given a group $\mathcal{G}$, we create a new group RSymGroup $\mathcal{G}$.

Theorem (Cayley's Theorem)

Every group is isomorphic to a subgroup of a symmetric group.

- In RSymGroup $\mathcal{G}$, various rules hold by reflexivity.
- We show that RSymGroup $\mathcal{G}$ is isomorphic to $\mathcal{G}$.
- By univalence and the structure identity principle, RSymGroup $\mathcal{G}$ is equal to $\mathcal{G}$.
- The strictify function transports a proof from RSymGroup $\mathcal{G}$ back to $\mathcal{G}$.

In the strictified group the following equations hold definitionally:

- $a(bc) = (ab)c$,
- $a1 = a = 1a$,
- ${a^{-1}}^{-1} = a$,
- and $(fg)^{-1} = g^{-1} \cdot f^{-1}$.

Functions compose strictly

Theorem (Cayley's Theorem)

Every group is isomorphic to a subgroup of a symmetric group.

Functions compose strictly

Theorem (Cayley's Theorem)

Every group is isomorphic to a subgroup of a symmetric group.

$$
\begin{aligned}
&\_\circ\_ : (f : B \to C) \to (g : A \to B) \to (A \to C) \\
&(f \circ g)\ x = f\ (g\ x)
\end{aligned}
$$

$$
\begin{aligned}
\text{comp-assoc} :\ & \quad (f : C \to D) \\
& \to (g : B \to C) \\
& \to (h : A \to B) \\
& \to f \circ (g \circ h) \equiv (f \circ g) \circ h \\
\text{comp-assoc}\ & f\ g\ h = \text{refl}
\end{aligned}
$$

Do invertible functions compose strictly?

$$
\begin{array}{l}
\textsf{record Inverse } (A : \textsf{Type})\ (B : \textsf{Type}) : \textsf{Type where} \\
\quad \textsf{field} \\
\qquad \uparrow\ : A \to B \\
\qquad \downarrow\ : B \to A \\
\qquad \varepsilon : \forall\ x \to\ \downarrow (\uparrow x) \equiv x \\
\qquad \eta : \forall\ y \to\ \uparrow (\downarrow y) \equiv y
\end{array}
$$

Strict invertible functions

```
record Inverse (A : Type) (B : Type) : Type where
  constructor ⌊_,_,_,_⌋
  field
    ↑ : A → B
    ↓ : B → A
    ε : ∀ b {x} → x ≡ ↓ b → ↑ x ≡ b
    η : ∀ a {y} → y ≡ ↑ a → ↓ y ≡ a

_∘_ : Inverse B C → Inverse A B → Inverse A C
_∘_ ⌊ f , g , p , q ⌋ ⌊ f' , g' , p' , q' ⌋ =
  ⌊ (λ x → f (f' x)) ,
    (λ y → g' (g y)) ,
    (λ b r → p b (p' (g b) r)) ,
    (λ a r → q' a (q (f' a) r)) ⌋
```

Strict invertible functions

$$
\begin{aligned}
\mathsf{assoc} :\ & (f : \mathsf{Inverse}\ C\ D)\\
& \to (g : \mathsf{Inverse}\ B\ C)\\
& \to (h : \mathsf{Inverse}\ A\ B)\\
& \to f \circ (g \circ h) \equiv (f \circ g) \circ h\\
\mathsf{assoc}\ & f\ g\ h = \mathsf{refl}
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{id\text{-}inv} : \mathsf{Inverse}\ A\ A\\
&\mathsf{id\text{-}inv} = \lfloor\ (\lambda\ x \to x)\ ,\ (\lambda\ x \to x)\ ,\\
&\qquad\qquad (\lambda\ b\ r \to r)\ ,\ (\lambda\ a\ r \to r)\ \rfloor
\end{aligned}
$$

$$
\begin{aligned}
\mathsf{id\text{-}unit\text{-}left} :\ & (f : \mathsf{Inverse}\ A\ B)\\
& \to \mathsf{id\text{-}inv} \circ f \equiv f\\
\mathsf{id\text{-}unit\text{-}left}\ & f = \mathsf{refl}
\end{aligned}
$$

$$
\begin{aligned}
\mathsf{id\text{-}unit\text{-}right} :\ & (f : \mathsf{Inverse}\ A\ B)\\
& \to f \circ \mathsf{id\text{-}inv} \equiv f\\
\mathsf{id\text{-}unit\text{-}right}\ & f = \mathsf{refl}
\end{aligned}
$$

Strict invertible functions

$$
\begin{aligned}
&\text{inv-inv} : \text{Inverse}\ A\ B \to \text{Inverse}\ B\ A \\
&\text{inv-inv}\ \lfloor\ f\ ,\ g\ ,\ \varepsilon\ ,\ \eta\ \rfloor = \lfloor\ g\ ,\ f\ ,\ \eta\ ,\ \varepsilon\ \rfloor
\end{aligned}
$$

$$
\begin{aligned}
&\text{inv-involution} : \quad (f : \text{Inverse}\ A\ B) \\
&\qquad\qquad\quad \to \text{inv-inv}\ (\text{inv-inv}\ f) \equiv f \\
&\text{inv-involution}\ f = \text{refl}
\end{aligned}
$$

$$
\begin{aligned}
&\text{inv-comp} : \quad (f : \text{Inverse}\ B\ C) \\
&\qquad\quad \to (g : \text{Inverse}\ A\ B) \\
&\qquad\quad \to \text{inv-inv}\ (f \circ g) \equiv \text{inv-inv}\ g \circ \text{inv-inv}\ f \\
&\text{inv-comp}\ f\ g = \text{refl}
\end{aligned}
$$

Representable functions

The map $\iota : g \mapsto g \cdot \_$ includes the group $\mathcal{G}$ in the symmetric group. We now want to restrict the symmetric group to those functions that are in the image of ι.

Proposition

A function $f : \mathcal{G} \to \mathcal{G}$ is in the image of ι if and only if for all $g, h \in \mathcal{G}$, $f(g \cdot h) = f(g) \cdot h$.

Representable functions

The map $\iota : g \mapsto g \cdot \_$ includes the group $\mathcal{G}$ in the symmetric group. We now want to restrict the symmetric group to those functions that are in the image of ι.

Proposition

A function $f : \mathcal{G} \to \mathcal{G}$ is in the image of ι if and only if for all $g, h \in \mathcal{G}$, $f(g \cdot h) = f(g) \cdot h$.

$$
\begin{aligned}
&\text{Representable} : \text{Inverse} \langle\ \mathcal{G}\ \rangle \langle\ \mathcal{G}\ \rangle \to \text{Type} \\
&\text{Representable}\ f = \forall\ x\ g\ h \to x \equiv g \cdot h \to \uparrow f\ x \equiv \uparrow f\ g \cdot h
\end{aligned}
$$

$$
\begin{aligned}
&\text{Repr} : \text{Type} \\
&\text{Repr} = \Sigma[\ f \in \text{Inverse} \langle\ \mathcal{G}\ \rangle \langle\ \mathcal{G}\ \rangle\ ]\ \text{Representable}\ f
\end{aligned}
$$

Representable symmetric group

- Let RSymGroup $\mathcal{G}$ be the subgroup of the symmetric group on $\mathcal{G}$ consisting of those functions that are representable.

Representable symmetric group

- Let RSymGroup $\mathcal{G}$ be the subgroup of the symmetric group on $\mathcal{G}$ consisting of those functions that are representable.
- This subgroup still has strict composition.

Representable symmetric group

- Let RSymGroup $\mathcal{G}$ be the subgroup of the symmetric group on $\mathcal{G}$ consisting of those functions that are representable.
- This subgroup still has strict composition.
- The inclusion ι is an isomorphism from $\mathcal{G}$ to the representable symmetric group.

Representable symmetric group

- Let RSymGroup $\mathcal{G}$ be the subgroup of the symmetric group on $\mathcal{G}$ consisting of those functions that are representable.
- This subgroup still has strict composition.
- The inclusion ι is an isomorphism from $\mathcal{G}$ to the representable symmetric group.
- By univalence we get an equality:

$$\iota{\equiv}\ \mathcal{G} : \mathcal{G} \equiv \text{RSymGroup}\ \mathcal{G}$$

Representable symmetric group

- Let RSymGroup $\mathcal{G}$ be the subgroup of the symmetric group on $\mathcal{G}$ consisting of those functions that are representable.
- This subgroup still has strict composition.
- The inclusion ι is an isomorphism from $\mathcal{G}$ to the representable symmetric group.
- By univalence we get an equality:

$$\iota{\equiv}\ \mathcal{G} : \mathcal{G} \equiv \mathsf{RSymGroup}\ \mathcal{G}$$

- This lets us define:

$$\begin{aligned}
\mathsf{strictify} :\ & (\mathcal{G} : \mathsf{Group}\ \ell\text{-zero}) \\
& \to (P : \mathsf{Group}\ \ell\text{-zero} \to \mathsf{Type}) \\
& \to P\ (\mathsf{RSymGroup}\ \mathcal{G}) \\
& \to P\ \mathcal{G} \\
\mathsf{strictify}\ \mathcal{G}\ P\ p & = \mathsf{transport}\ (\mathsf{sym}\ (\mathsf{cong}\ P\ (\iota{\equiv}\ \mathcal{G})))\ p
\end{aligned}$$

Further thoughts

Further thoughts

Does this all work with categories instead of groups?

Conclusion

- For each group $\mathcal{G}$ we can generate an isomorphic group RSymGroup $\mathcal{G}$.
- This group has nice definitional properties
- Univalence allows us to generate an equality between the two groups.
- This allows us to prove theorems about an arbitrary group by instead proving them on the strictified group.
- `https://alexarice.github.io/posts/sgtuf/Strict-Group-Theory-UF.html`