

2024/10/28

# 15個のレンズ則について

東北大学院情報科学研究科  
中野研究室  
修士一年  
佐藤龍之介

# 目次

- 01 初めに
- 02 今回の発表の目的
- 03 **CORE Lens Laws**
  - ・ **StrongGetPut** と **GetPut** の違い
  - ・ レンズ則と一貫性
- 04 **Weaker Lens Laws**
  - ・ **PutGet** と **WeakPutGet** の違い

# 01 今回の発表の目的

今後行う研究(予定)

- 15個のレンズ則が全域の場合の証明をCoqで行う
- 16個のレンズ則が部分間数の場合の証明において未だ証明されていない4個のレンズ則についての証明

今回の発表の目的

- ”Lens Laws Zoo”に出てくるレンズ則の理解

## 02 初めに

レンズを構成する2つの関数getとputは全域関数

→ ソースSとビューV間の変換が確実に成立し、一貫性が保たれやすくなる。逆に、getとputが全域関数でないといくつかの入力について出力が定義されてない為、getとput間の一貫性が保証できないことがある

レンズ則により、ソースとビューの間での双方向変換が整合性を持って行われ、データの一貫性や信頼性が維持される

# 03 CORE Lens Laws

- ・ソースとビュー間の一貫性を保証する特性 (well-behaved, very-well-behaved, bijective) に重要

## StrongGetPut

$$\forall s, s' \in S, \text{put}(s, \text{get}(s')) = s'$$

任意のgetで生成されたビューに対して、putを適用すると、元のソースに依らず同じソースが得られる

## PutGet

$$\forall s \in S, \forall v \in V, \text{get}(\text{put}(s, v)) = v$$

更新されたビューvによって得られたソースsに対応するビューvが元の更新されたビューvと等しい

## GetPut

$$\forall s \in S, \text{put}(s, \text{get}(s)) = s$$

元のソースデータにget関数を適用して生成したビューをputに再度適用すると、ソースが変わらない

## PutPut

$$\forall s \in S, \forall v, v' \in V, \text{put}(\text{put}(s, v), v') = \text{put}(s, v')$$

## - StrongGetPut とGetPut の違い

例.

$$\begin{aligned}\text{get}(s) &= 2s \ (\in V), \\ \text{put}(s, v) &= v - s \ (\in S),\end{aligned}$$

**StrongGetPut** を満たすか

(左辺)  
 $= \text{put}(s, \text{get}(s')) = \text{put}(s, 2s')$   
 $= 2s' - s$   
(右辺)  
 $= s'$   
よって、StrongGetPut則を満たさない□

**StrongGetPut**

$$\cdot \text{put}(s, \text{get}(s')) = s'$$

**GetPut**

$$\cdot \text{put}(s, \text{get}(s)) = s$$

**GetPut** を満たすか(調べる)

(左辺)  
 $= \text{put}(s, \text{get}(s)) = \text{put}(s, 2s) = 2s - s$   
 $= s$   
(右辺)  
 $= s$   
よって、GetPut則を満たす□

# - レンズ則と一貫性

Well-Behaved, Very-Well-Behaved, bijective の違い

## Well-Behaved

満たすべきレンズ則：

GetPut

•  $\text{put}(s, \text{get}(s)) = s$

PutGet

•  $\text{get}(\text{put}(s, v)) = v$

特徴：

ソースとビュー間でデータ  
の変換が矛盾なく行える

## Very-Well-Behaved

満たすべきレンズ則：

GetPut

PutGet

PutPut

•  $\text{put}(\text{put}(s, v), v') = \text{put}(s, v')$

特徴：

ビューが何度更新されても同じ結  
果になる。過去の更新が影響しな  
い安定性がある。

## Bijjective

満たすべきレンズ則：

StrongGetPut

•  $\text{put}(s, \text{get}(s')) = s'$

PutGet

特徴：

ソースとビューの間に一対一  
の関係があり、ビューの更新  
には元のソースが不要。

# 04 Weaker Lens Laws – (1)

## WeakPutGet

$$\forall s \in S, \forall v \in V, \\ \text{put}(s, \underline{\text{get}(\text{put}(s, v))}) = \text{put}(s, v)$$

## PutGetPut

$$\forall s \in S, \forall v \in V, \\ \text{put}(\text{put}(s, v), \text{get}(\text{put}(s, v))) = \text{put}(s, v)$$

## GetPutGet

$$\forall s \in S, \text{get}(\text{put}(s, \text{get}(s))) = \text{get}(s)$$

## Undoability

$$\forall s \in S, \forall v \in V, \\ \text{put}(\text{put}(s, v), \text{get}(s)) = s$$



## - PutGet と WeakPutGet の違い

例.

$\text{get}(s) = \langle s, s \rangle,$   
 $\text{put}(s, \langle v1, v2 \rangle) = v1,$   
 $\langle s, s \rangle = \langle 1, 2 \rangle$  のとき

**PutGet** を満たすか (調べる)

(左辺)  
 $= \text{get}(\text{put}(s, \langle 1, 2 \rangle)) = \text{get}(1) = \langle 1, 1 \rangle$

(右辺)  
 $= \langle 1, 2 \rangle$

よって、PutGet 則を満たさない □

**PutGet**

•  $\text{get}(\text{put}(s, v)) = v$

**WeakPutGet**

•  $\text{put}(s, \text{get}(\text{put}(s, v))) = \text{put}(s, v)$

**\*WeakPutGet** を満たすか (調べる)

(左辺)  
 $= \text{put}(s, \text{get}(\text{put}(s, \langle 1, 2 \rangle))) = \text{put}(s, \text{get}(1))$   
 $= \text{put}(s, \langle 1, 1 \rangle) = 1$

(右辺)  
 $= \text{put}(s, \langle 1, 2 \rangle) = 1$

よって、WeakPutGet 則を満たす □

## 04 Weaker Lens Laws – (2)

### GetInjectivity

$$\forall s, s' \in S, \\ \text{get}(s) = \text{get}(s') \Rightarrow s = s'$$

### GetSurjectivity

$$\forall v \in V, \exists s \in S, \\ \text{get}(s) = v$$

### PutTwice

$$\forall s \in S, \forall v \in V, \\ \text{put}(\text{put}(s, v), v) = \text{put}(s, v)$$

### SourceStability

$$\forall s \in S, \exists v \in V, \\ \text{put}(s, v) = s$$

# 04 Weaker Lens Laws – (3)

## ViewDetermination

$$\forall s, s' \in S, \forall v, v' \in V, \\ \text{put}(s, v) = \text{put}(s', v') \Rightarrow v = v'$$

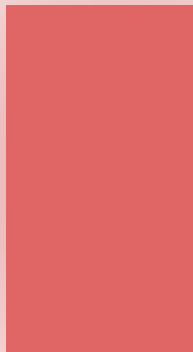
## PutInjectivity

$$\forall s \in S, \forall v, v' \in V, \\ \text{put}(s, v) = \text{put}(s, v') \Rightarrow v = v'$$

## PutSurjectivity

$$\forall s \in S, \exists s' \in S, \exists v \in V, \\ \text{put}(s', v) = s$$

(部分関数の場合について未だ証明されていない4つのレンズ則)



# 引用

- Lens Laws Zoo(P1 - 10) - Nakano Keisuke
- Characterizations of Partial Well-Behaved Lenses - Keishi Hashiba