シャッフラーにおける証明方式の比較

概要

ブラウザ側で差分プライバシー(DP)乱数化を行うシステムにおけるシャッフラーは、ユーザレポートの並べ替えによって匿名化を強化します。

シャッフラーの正当性を担保する証明方式には、

1. **Sum Preservation**証明 :コミットメントの合計が不変であることを示す
2. **Permutation**証明 :出力列が入力列の完全な置換であることを示す

- **Sum Preservation**
  - 加法同型性で「総和が変わらない」ことを高速に検証
  - 同じ和でも異なる集合になり得る → 改竄を見逃す恐れ
- **Permutation証明**
  - 多重集合の同一性をゼロ知識で保証
  - 強い担保だが高コスト
    (Bulletproofs: Short Proofs for Confidential Transactions and More, Efficient Zero-Knowledge Argument for Correctness of a Shuffle)

1. Sum Preservation証明

定義と仕組み

- Pedersenコミットメント

$$C_i = r_i H + v_i G$$

- 加法同型性により

$$\sum_i C_i = C\left(\sum_i v_i, \sum_i r_i\right)$$

- 証明：入出力コミットメント列の総和一致を非対話型ZKPで示す (論文)

限界

- 例：{1,3} と {2,2} は同じ合計4 → 改竄か置き換えか判別不可 (論文)
- 追加・削除・不正操作を完全には防げない

2. Permutation(並べ替え)証明

定義と仕組み

- 入力列 ({C_i}) と出力列 ({C'_j}) が同一多重集合であることを保証
- ランダムチャレンジ (z) に対し：

$$\prod_i (C_i - z) = \prod_j (C'_j - z)$$

  をゼロ知識で証明
  (MinimalShuffle, Bulletproofs)

利点

- 完全性：並べ替え以外の改竄を一切許さない
  (Distributed Differential Privacy via Shuffling)
- 応用例：選挙集計、金融ソルベンシー証明など

3. DPシャッフルモデルでの選択基準

プライバシー重視

- LDP乱数化自体が強力 → シャッフラーは匿名化(Linkability遮断)を担う
  (Private Summation in the Multi-Message Shuffle Model)
- 集計精度重視なら **Sum Preservation** の軽量性を活かす
  (A Shuffling Framework for Local Differential Privacy)

インテグリティ重視

- 改竄防止や不正レポート検知には **Permutation**証明 を推奨
  (Bulletproofs, MinimalShuffle)
- PROCHLOなどでは両立実装例として採用

まとめ

- **Sum Preservation**
  - 証明コスト低、集計用途に適するが改竄検出は不可
- **Permutation証明**
  - 証明コスト高、完全一致保証で改竄リスク排除
- 使い分け：
  - プライバシー重視 → Sum Preservation
  - インテグリティ重視 → Permutation