3T-03

# 個人間分散バックアップにおけるブロックデータの重複排除手法

石田 大明† 打矢 隆弘‡ 内匠 逸‡

†名古屋工業大学 工学部 情報工学科
〒446-8555 愛知県 名古屋市 昭和区 御器所町

‡名古屋工業大学 大学院 工学研究科
〒446-8555 愛知県 名古屋市 昭和区 御器所町

## 1 はじめに

近年，災害の発生によるデータ消失に備えるためにバックアップの需要が高まっている．堅牢なバックアップ手段として，地理的に分散してデータを保存する分散バックアップが注目されている．分散バックアップにより，災害の発生時にデータが消失する可能性が低くなる．低コストで運用することができる既存の個人間分散バックアップシステムには，所要時間*，即時性†，保存容量‡などの問題点が存在する．この問題点の解決を目標とした先行研究として，エージェントを用いた個人間分散バックアップシステム [1] が存在する．本研究では，ブロックデータの重複排除手法を新たに提案し，これをシステムに組み込むことでバックアップデータの容量削減を目指す．

## 2 先行研究

先行研究では，エージェントフレームワーク DASH[2] を用いて個人間分散バックアップシステムを構築している．このシステムでは，エージェントを用いて複数のユーザの計算機にデータを分散配置し，冗長化を施すことで，リストア時に全ての計算機を用いることなくデータの復元を可能とする．このシステムにより，既存の個人間分散バックアップシステムに比べて所要時間の削減と即時性の向上を実現している．しかし，ファイルの更新に従い複数回バックアップを行うにつれて，保存先のユーザの計算機に保存されるバックアップデータの容量が大きくなるという問題点が存在する．

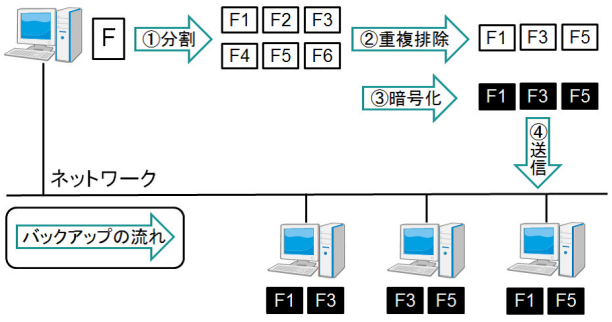


図 1: 提案手法の概要図

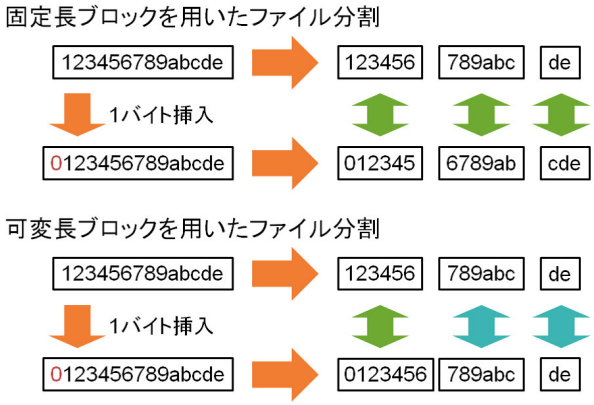


図 2: ファイル分割の概要図

## 3 提案手法

上記で指摘した問題点を解決するために，バックアップデータを可変長ブロックに分割し，過去に保存されたブロックとの重複排除を行う（図 1）．

### 3.1 ファイル分割

ファイルを可変長ブロックに分割するために，CDC（Content Defined Chunking）[3] を用いる．CDC のアルゴリズムの流れについて説明する．

1. ファイルをバイト列に変換し，ブロックの境界を決定するために固定長のウィンドウを設置する．
2. ウィンドウ内のデータをハッシュ値に変換する．
3. 変換したハッシュ値の下位数ビットを特定の値と比較する．値が一致した場合，ウィンドウの終点をブロックの境界とする．
4. ウィンドウを 1 バイトだけスライドさせる．
5. 2.〜4. の作業をファイルの終点まで繰り返す．

固定長ブロックを用いた場合，ファイルの一部を変更し，ブロックに分割すると，変更前に分割したブロッ

---

De-duplication of Block Data in the Distributed Backup System among Individuals
†Hiroaki ISHIDA ‡Takahiro UCHIYA ‡Ichi TAKUMI

*バックアップ開始からバックアップ終了までの時間
†ファイル更新後からバックアップ終了までの時間
‡ユーザの計算機に保存されるバックアップデータの容量の総量

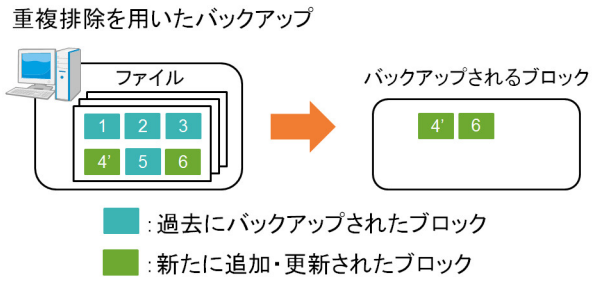


図 3: 重複排除の概要図

クとの間にズレが発生し，重複排除を行うことができない．ファイルを可変長ブロックに分割することで，固定長ブロックに分割する手法と比べてファイルの変更時に発生するブロックのズレを少なくできる（図 2）．

### 3.2 重複排除

重複したブロックのバックアップを避けるために，重複排除を用いる（図 3）．初回のバックアップでは，分割されたブロック全てをバックアップする．2 回目以降のバックアップでは，新たに追加・更新されたブロックのみをバックアップする．追加・更新されたブロックの判別には，ブロックのハッシュ値を用いる．これによりバックアップデータの容量をファイル単位でなくブロック単位で削減できる．

### 3.3 暗号化

重複排除を行い，新たに追加・更新されたと判断された各ブロックを暗号化するために，ブロック暗号 AES（Advanced Encryption Standard）を用いる．AES を用いることにより他人の計算機にデータをバックアップしても保存された情報を保護できる．ただし，ブロック暗号はパディング*によりファイルサイズが若干増加する欠点を持つため，複数のブロックを暗号化する提案手法においても有効であるかを検証する必要がある．

## 4 実験と考察

実装した手法の有効性検証のために，重複排除，暗号化の実験を行った．その結果を以下に示す．

### 4.1 重複排除

重複排除を 1GB の mp4 ファイルに対して行った実験結果を表 1 に示す．なお，実験に用いた重複排除対象ファイルの概要は以下の通りである．

*データ長を固定長の倍数に合わせる処理

**初回の重複排除**

ビデオカメラにより撮影された mp4 ファイル

**2 回目の重複排除**

撮影ファイルに BGM を追加した mp4 ファイル

表 1: 重複排除の実験結果

| | ブロック数 | バックアップ数 |
|---|---|---|
| 初回の重複排除 | 26907 | 26907 |
| 2 回目の重複排除 | 26909 | 9553 |

表 1 より，重複排除を行うことによりバックアップ対象ブロック数を本来の 26909 個から 9553 個まで削減できていると言える．これは，ファイルサイズに換算すると 52%の削減であったため，提案手法において重複排除は有効であると考えられる．

### 4.2 暗号化

500KB，10MB，100MB，1GB の各サイズ 5 種類のファイルを分割し，生成されたブロックに対して AES を施した実験結果を表 2 に示す．

表 2: 暗号化の実験結果

| ファイルサイズ | 平均増加サイズ（KB） |
|---|---|
| 500KB | 0.52 |
| 10MB | 10.6 |
| 100MB | 107.0 |
| 1GB | 1115.3 |

表 2 より，暗号化を施すことによる総増加サイズは元のファイルサイズに比べて 1000 分の 1 程度であるので，パディングにより各ブロックのサイズが若干増加する欠点は，提案手法において問題ないと考えられる．

## 5 まとめ

先行研究におけるバックアップデータの容量の問題点を解決するために，ブロックデータの重複排除手法を提案した．今後は，提案手法を組み込んだバックアップシステムを用いて，評価実験により有用性を検証する．

## 参考文献

[1] Motohiro Shibakawa et al., “Development of the Distributed Backup System using Multi-Agent Technology”, ICCI*CC2013, pp.268-273, 2013.

[2] 打矢 隆弘 他, “リポジトリ型 AF の開発と評価”, 情報技術レターズ, Vol.2, pp.139-141, 2003.

[3] Athicha Muthitacharoen et al., “A Low-bandwidth Network File System”, ACM SIGOPS Operating Systems Review, Vol.35, No.5, pp.174-187, 2001.