テープ・ドライブにおける不定期転送速度低下現象を伴うデータ書き込み所要時間削減策

大石 豊　名倉 寛尚　片桐 隆司　太田 由美子

A method to reducing time to write data on a tape drive with intermittent procrastination server

Yutaka Oishi, Hironobu Nagura, Takashi Katagiri and Yumiko Ohta

LTFS（Linear Tape File System）の登場により，磁気テープ・メディアをファイル・システム経由で利用できるようになり，テープ・ドライブの利用環境は拡大している．本論文では，サーバーとテープ・ドライブが持つ内部データ転送バッファー間の転送速度の変動により，データ書き込み所要時間が増加する課題を解決する手法を提案する．この手法は，サーバーとバッファーの間の転送速度が一定していないことを検知し，磁気テープ・メディアの巻き戻し時間を考慮した走行速度選択アルゴリズムを適用するものである．この手法により，データ書き込み所要時間を 3 割以上削減できる．

Magnetic tape media have much opportunity to be used via file system under LTFS (Linear Tape File System) environment. A new method is proposed to reduce time to write data for a tape drive with intermittent procrastination server. The method detects that a data transfer rate between a server and internal data transfer buffer on a tape drive is not stable and applies tape speed selecting algorithm which considers the time to backhitch magnetic tape media. The method can reduce time to write data more than thirty percents.

Key Words & Phrases：テープ・ドライブ，LTFS，転送速度，書き込み所要時間，不定期転送速度低下
Tape Drive, Linear Tape File System, Data Transfer Rate, Time to Write Data, Intermittent Procrastination

1. はじめに

IBM TS2350 テープ装置などに搭載されている Linear Tape-Open（LTO）テープ･ドライブ（以下，テープ・ドライブ）は従来データのバックアップおよびアーカイブのために用いられてきた．一方，2010 年に登場した Linear Tape File System（LTFS）が広まるにつれ，ビデオ・アーカイブをはじめとする種々の用途において，磁気テープ・メディア（以下，テープ）を光メディアや USB メモリーのようにファイル・システム経由で利用する機会が増えている［1］［2］［3］．LTFS を用いたテープの利用は，第 5 世代 LTO テープ・ドライブからパーティション機能をサポートしたことによって可能となったものである［4］．

テープ・ドライブの利用環境が拡大するにつれ，利用環境によってデータ書き込み所要時間が想定以上に増加する場合があるという新しい課題が顕在化した．これは，サーバーからテープ・ドライブの内部データ転送バッファー（以下，バッファー）へデータを転送する際に不定期に転送速度が低下する環境が存在することに起因している．このような状況では，サーバーとバッファーの間の転送速度がほぼ一定であることを前提とした従来技術では，最適なテープの走行速度を選択することが困難である．最適な走行速度を選択できないと，テープ・ドライブがサーバーから書き込むデータを受け取ることのできないタイミングが発生し，そのためにテープへのデータ書き込み所要時間が増加する．

本論文ではサーバーとバッファーの間の転送速度が一定していない状態の検知手法，およびテープの巻き戻し時間を考慮した新しい走行速度選択手法のアルゴリズムについて述べる．この手法を適用することにより，サーバーとバッファーの間の転送速度が一定ではない場合にもテープ・ドライブがサーバーから書き込むデータを受け取ることのできないタイミングが減少し，従来技術を適用した場合と比較してデータの書き込み所要時間

提出日：2011年9月20日　再提出日：2013年2月25日

を削減することが可能となる.

2. テープへのデータの書き込み方法

2.1 テープの動き

テープ・ドライブがテープにデータを書き込む場合，テープを長手方向に定速で移動（走行）させ，その間にデータを書き込む．テープは，**図 1** に示すように 4 つのデータ・バンドと呼ばれる領域に分割されている．テープの始端から終端，ないし終端から始端への一度の移動でデータを書き込むことができる領域をラップと呼ぶ．ラップは各データ・バンドに数十本ずつ割り当てられている．データの書き込み時にラップの終端まで移動すると，テープの走行方向を反転し，次のラップ上で書き込みを再開する．

テープが停止している場合や，テープの加減速中に，テープにデータを書き込むことはできない．テープ・ドライブは，あらかじめ定められた複数の走行速度を用い，テープを走行させる．テープにデータを書き込み中に，テープを巻き戻すことなく走行速度の変更を試みると，加減速中はテープにデータを書き込むことができないため，走行速度変更時にはテープ上の記録に空白が生じ，それによってテープに書き込み可能なデータの総容量が低下する．この現象を避けるため，走行速度を変更する場合は，一度テープの走行を止め，テープを巻き戻してから，再度新たな走行速度へテープを加速する．このテープを巻き戻す作業をリポジショニングと呼ぶ[5]．リポジショニングの所要時間は，リポジショニング前後の走行速度と加速度に依存し変動し，数秒の時間が必要となる．

2.2 データの流れと転送速度

サーバーがテープにデータを書き込む場合，サーバーからテープ・ドライブに送られてきたデータは一旦バッファーに格納され，その後，バッファーからテープに書き込まれる．バッファーの容量はテープ・ドライブの世代によって異なるが，数百 MB から数 GB 程度である．バッファーは数 MB の大きさのセグメントと呼ばれる領域に等分されており，サーバーから送られてきた可変長のデータを必要に応じて，固定長のセグメントに分割して順次格納する．テープ・ドライブはセグメントの単位でデータをテープに書き込む．

サーバーからバッファーにデータが送られてくる際の転送速度を，サーバー転送速度と呼ぶ．サーバー転送速度は，セグメントの容量を，あるセグメントがサーバーから送られてきたデータで満たされてからその次のセグメントがサーバーから送られてきたデータで満たされるまでの時間で割った値と定義する．同様に，バッファーからテープにデータを送る際の転送速度を，テープ転送速度と呼ぶ．テープ転送速度は，セグメントの容量を，あるセグメントに格納されたデータをテープに書き終えてからその次のセグメントに格納されたデータをテープに書き終えるまでの時間で割った値と定義する．

テープ・ドライブはテープに規格によって定められた所定の記録密度でデータを書き込むため，各走行速度におけるテープ転送速度は一定である．一方，サーバー転送速度は，書き込むデータを送出するアプリケーション等の影響によって変動することがある．サーバー転送

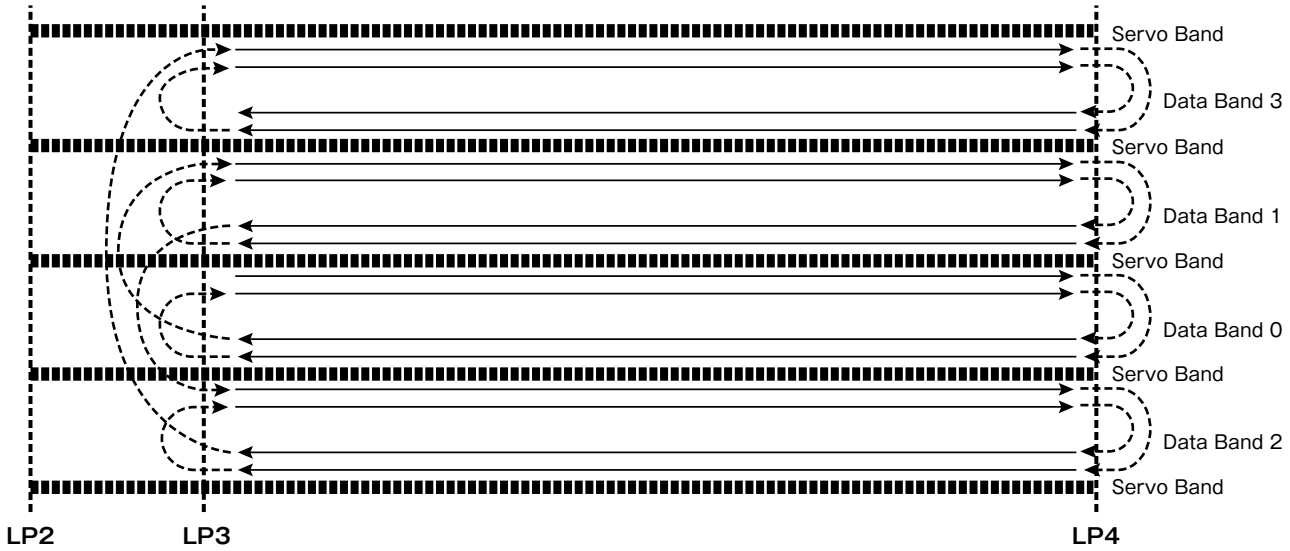

図1. テープのフォーマット

速度とテープ転送速度の差異が一定の範囲内であれば,差異がデータ書き込み所要時間へ与える影響はバッファーによって吸収される.

テープ転送速度に比べ，サーバー転送速度が著しく速い場合，バッファーはテープへ書き込み待ちのデータで満たされる．その後バッファーに空きが生じるまでテープ・ドライブはサーバーから書き込みデータを受け取ることはできない．逆に，テープ転送速度に比べ，サーバー転送速度が著しく遅い場合，書き込み待ちのデータをすべてテープに書き終えるとバッファーは空になる．その後サーバーから次のデータが送られてくるまでリポジショニングを発生させて待つことになる.

リポジショニングを実施している間，サーバーから送られてくるデータをテープ・ドライブが受け取り続けられるかどうかは，バッファーの容量，サーバー転送速度，それにリポジショニングの所要時間に依存する．リポジショニングの所要時間内にサーバーから送られてくるデータ量がバッファーの容量を超える場合は，バッファーがサーバーから送られてきたデータで満たされる．その後リポジショニングが終了しバッファー内のデータがテープに書き込み始めるまでの間，テープ・ドライブはサーバーから送られてくるデータを受け取ることはできない．このような，テープ・ドライブがサーバーから送られてくるデータを受け取ることができないタイミングの発生は，データ書き込み所要時間の増加につながる.

2.3 スピード・マッチング機能

走行速度を切り替える主なタイミングは，テープをラップの終端まで移動させた後走行方向を反転する場合，およびリポジショニングが発生した場合である．走行速度は従来，スピード・マッチング機能と呼ばれる走行速度選択手法を用いて決定している[6][7].スピード･マッチング機能は，サーバー転送速度の履歴を元に走行速度を選択する．直近のサーバー転送速度のみを採用すると，サーバー転送速度の一時的な変動に影響されて，サーバー転送速度の平均値と乖離した走行速度の選択につながる．一方，過去の履歴をすべて採用すると，サーバー転送速度が変化した場合に，履歴情報が多ければ多いほど，直近のサーバー転送速度の変化に追従できなくなるという特性がある．これらの不具合を避けるため，スピード・マッチング機能は，過去の履歴を尊重しつつも，より直近に近い情報を重用するようにサーバー転送速度の加重平均値を算出して採用する仕組みになっている.

サーバー転送速度と，テープ転送速度の乖離が大きい場合，積極的にリポジショニングを実行し走行速度を切り替える．リポジショニングを実行するかどうかの判定は，ラップの終端まで現在の走行速度で書き続けた場合の所要時間と，即座にリポジショニングを実行し新たな走行速度を用いてラップの終端まで書き続けた場合の所要時間を比較することによって行う.

スピード・マッチング機能が選択する走行速度は，サーバー転送速度よりもテープ転送速度が若干速くなる走行速度ないし若干遅くなる走行速度のいずれかである．リポジショニングの間にサーバーから送られてくるデータを受け取り続けられる場合，サーバー転送速度よりも若干速くなる走行速度を選択する．これにより，サーバーを待たせることなく，リポジショニングの発生頻度を抑制できる．逆に，リポジショニングの間にサーバーから送られてくるデータを受け取り続けられない場合は，若干速くなる走行速度を選択した場合と，若干遅くなる走行速度を選択した場合の予測されるデータ書き込み所要時間を比較し，データ書き込み所要時間が短くなる走行速度を選択する.

3. データ書き込み所要時間の増加

テープ・ドライブ利用時にデータ書き込み所要時間が予測よりも長くなる現象は，サーバーに起因する場合と，テープ･ドライブに起因する場合に大別できる．データ書き込み所要時間が増加する現象の要因を解析するために，各セグメントがサーバーから送られてきたデータで満たされる間隔を調査した結果，次のことが明らかとなった.

1. サーバーとバッファーの間の平均転送速度は十分速い
2. リポジショニングが頻繁に発生
3. リポジショニング中にテープ・ドライブのバッファーがいっぱいになる
4. バッファーがいっぱいの期間は，サーバーの書き込みが待たされる

サーバーとバッファーの間の平均転送速度,つまりバッファーのセグメントごとに算出したサーバー転送速度の平均値が十分速いことは，データ書き込み所要時間の増加の要因がテープ・ドライブ側にあることを示している．また，リポジショニングが頻繁に発生することは，スピード・マッチング機能によって選択された走行速度に対応するテープ転送速度とサーバー転送速度の乖離が大きいことを示している．最大テープ転送速度より最

大サーバー転送速度が速いこともあり，リポジショニング中にバッファーがいっぱいになることがある．その間サーバーの書き込みが待たされることは，デザイン通りの挙動である※．一方，スピード・マッチング機能が期待通りに動作していれば，リポジショニングの頻度は十分低く，データ書き込み所要時間への影響は十分小さくなるはずである．

リポジショニングが頻発する要因をさらに調査したところ，サーバー転送速度が一時的に低下する現象が不定期に発生していることを見出した．例えば，ファイル・サイズが 1GB のファイルを連続して書き込む場合，図２に示すように，大半の期間はバッファーのセグメントごとに算出したサーバー転送速度は 100MB/sec 程度であるが，セグメント数十個分程度のデータをサーバーからテープ・ドライブに送信する間のサーバー転送速度が連続して 60MB/sec 程度に低下する現象が不規則に発生していた．サーバー転送速度が低下するタイミングに規則性はなく，また，書き込むファイルのファイル・サイズへの依存性も見られなかった．

スピード・マッチング機能が，サーバー転送速度の履歴から求めた加重平均値を基に選択した走行速度を利用していると，サーバー転送速度の一時的な低下が不定期に発生したときにバッファーが空になり，リポジショニングが発生していることが明らかになった．

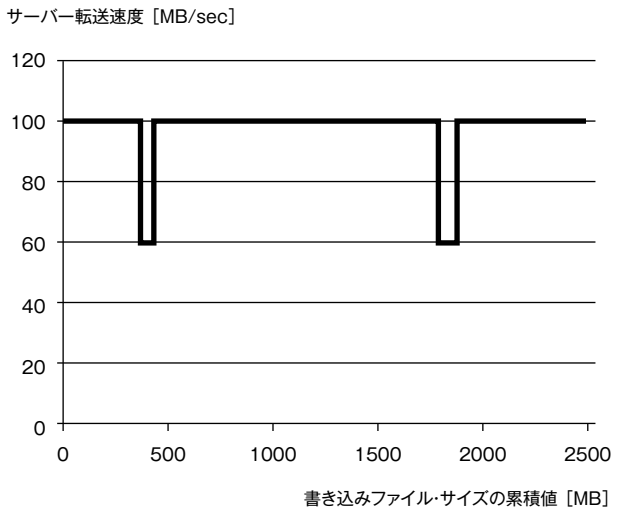

図2. サーバー転送速度の不定期な低下現象

※ テープ・ドライブはサーバーから送られてきたデータを圧縮し，テープに書き込む．最大サーバー転送速度が最大テープ転送速度の数倍に設定されているのは，テキスト・データのように圧縮の効くデータを扱う場合に対応するためである．動画ファイルのように圧縮の効かないデータを扱う場合にサーバーから最大サーバー転送速度でデータを送付されると，リポジショニングの発生の有無によらずバッファーが一杯になることがある．

4. サーバー転送速度の不定期な低下を伴う場合の書き込み所要時間削減策

4.1 サーバー転送速度の不定期な低下の検知

従来のスピード・マッチング機能では，各セグメントがサーバーから送られてきたデータで満たされるタイミングを基に，サーバー転送速度を算出している．一方，サーバー転送速度の一時的な低下が不定期に発生していることを，セグメントがデータで満たされるタイミングから検知することは困難である．それは，セグメントごとに評価すると，サーバーから送られてくるデータの容量とセグメントの容量が一致しないことから必然的に発生するタイミングのずれによるばらつきが無視できず，誤検出を避けることが難しいからである．また，毎回直近の数十セグメントの転送速度を評価すると，前述のばらつきは無視できるものの，数十 MB にわたり急激に転送速度が変わる場合では転送速度がなだらかに変動しているように見えるため，サーバー転送速度の一時的な低下の発生を検知するための適切な閾値を定めにくい．セグメントをいくつかの領域ごとに分け，領域ごとの転送速度を評価すると，転送速度が遅い部分が複数の領域にまたがるような場合の転送速度の変化の幅を見積もるのが困難となる．

サーバー転送速度の一時的な低下を検知するため，セグメント毎にサーバーから送られてきたデータで満たされるタイミングから判断するのではなく，より大局的にバッファーが空になるタイミングによって判断することにする．サーバー転送速度が一時的に低下したからといって，常にバッファーが空になるわけではない．バッファーが空になる前にサーバー転送速度が元の速い状態に戻れば，バッファーが空になる現象は発生しない．今回のサーバー転送速度の一時的な低下を検知する目的は，サーバー転送速度の一時的な低下によってバッファーが空になり，それによってリポジショニングが頻発する現象に対処することである．バッファーが空にならない程度のサーバー転送速度の一時的な低下はむしろ積極的に無視することが求められる．

通常，サーバー転送速度とテープ転送速度に若干のずれが存在するため，サーバー転送速度が一時的に低下する現象が発生しなくてもバッファーが空になることがある．従来のバッファーが空になる現象とサーバー転送速度の一時的な低下によるバッファーが空になる現象を区別するため，バッファーが空になる間隔に着目

する．そして，バッファーからテープへデータの転送を開始後，一定の間隔よりも短い間隔でバッファーが空になった場合には，サーバー転送速度の一時的な低下が不定期に発生しているとみなすことにする．

ここまでデータをテープに書き込む場合について解説したが，サーバー転送速度の一時的な低下の検知はデータをテープから読み出す場合にも適用可能である．読み出し時にサーバー転送速度が一時的に低下すると，書き込み時とは逆にバッファーがサーバーへ転送待ちのデータで満たされ，リポジショニングが発生する．データをテープから読み出す場合は，バッファーが満たされる間隔を観察することにより，サーバー転送速度の一時的な低下が不定期に発生していることを検知できる．

4.2 スピード・マッチング機能の不定期転送速度低下モード

サーバー転送速度の一時的な低下が不定期に発生している場合には，従来スピード・マッチング機能が選択する走行速度よりも遅い走行速度を選択することにする．これは，サーバー転送速度の一時的な低下が不定期に発生することによって引き起こされるリポジショニングが，データ書き込み所要時間に与える影響を考慮したものである．

従来と比べて遅い走行速度を選択することにより，サーバー転送速度が低下していない期間は，バッファーがテープに書き込み待ちのデータで満たされるようになる．これは，テープ・ドライブがサーバーから書き込みデータを受け取れない期間が発生することを意味している．各走行速度の間隔は 5MB/sec 程度であり，従来と比べ遅い走行速度の選択がデータ書き込み所要時間に与える影響は，サーバー転送速度が 100MB/sec 程度の場合はたかだか 5% 程度であると言える．

一方，サーバーからのデータ書き込み時にサーバー転送速度が一時的に低下した場合には，バッファーが空になりにくくなる．なぜなら，その時点でバッファーがいっぱいであることが期待され，またテープ転送速度が比較的遅いからである．その結果，リポジショニングの頻度が減少することにより，データ書き込み所要時間が削減される．この削減量は，サーバー転送速度が低下していない場合に，テープ・ドライブがサーバーから書き込みデータを受け取れない期間が発生することによるデータ書き込み所要時間の増加量を有意に上回ることが期待される．

不定期転送速度低下モードにおいて，従来のスピード・マッチング機能が選択する走行速度との差分は，バッファーが空になる間隔が一定値を超える頻度によって変動させる．具体的には，サーバー転送速度の一時的な低下が不定期に発生している場合には，従来のスピード・マッチング機能によって選択された走行速度よりも一段階遅い走行速度を選択する．さらに，直近 4 回のバッファーが空になった場合のそれぞれの間隔である 3 回の間隔のうち 2 回以上の間隔が一定値以下であった場合には，もう一段階遅い走行速度を選択する．段階的に走行速度を調整することにより，必要以上に走行速度を低下させることなく，サーバー転送速度の一時的な低下が不定期に発生することに起因するリポジショニングの発生頻度を抑制できる．このアルゴリズムをフローチャートで示すと図 3 のようになる．

5. パフォーマンス評価

5.1 評価方法

データ書き込み所要時間が増加する問題がみられる場合に対する改善策を評価するためには，評価方法を確立する必要がある．テープをファイル・システム経由で利用する場合，利用環境によって書き込むファイルのサイズ，数，頻度がまちまちなため，典型的な使

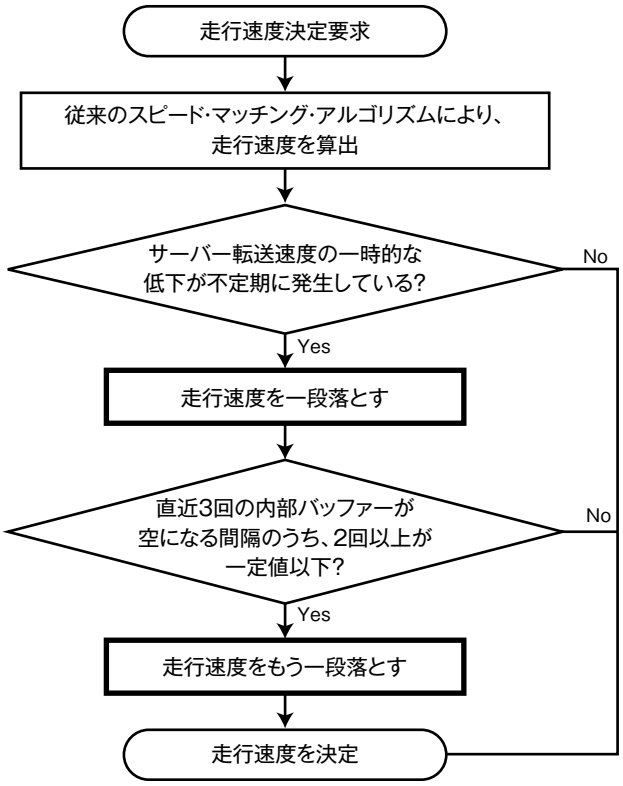

図3. 走行速度決定アルゴリズム

用例を定義することが難しい．また，テープ・ドライブがテープにデータを書き込む際は，テープを長手方向に往復させるが，書き込み中にテープの走行方向を切り替えが発生すると加減速中にデータを書き込めないことからデータ書き込み所要時間が増加するという特徴がある．

これらのことを考慮し,評価方法を次のように定義する．

1. テープをテープ・ドライブに挿入
2. テープを LTFS フォーマットに初期化
3. ディスク上に保存されたファイル・サイズ 100MB のファイルを，ステップ 2 でフォーマットしたテープ上へ 10GB 分書き込む（コピーする）所要時間を計測
4. ステップ 3 を 5 回繰り返す
5. ファイル・サイズ 1MB と 32KB のファイルについても，ステップ 3 とステップ 4 を実行
6. ステップ 3 で得られた，各ファイル・サイズの 5 回のデータ書き込み所要時間のうち最短の所要時間と最長の所要時間を除いた 3 回の所要時間の平均値を算出
7. 改善策を適用したテープ・ドライブを用いてステップ 1 ～ステップ 6 を繰り返し，ステップ 6 で算出された各ファイル・サイズにおけるデータ書き込み所要時間の平均値を改善策適用前に算出した値と比較

5.2 評価結果

第 5 世代ハーフハイト･テープ･ドライブと第 5 世代テープを用い，前節で定義した評価方法を基に，評価を実施した．本手法を適用前後のデータ書き込み所要時間を評価した結果を，各ファイル・サイズにおいて改善策適用前のデータ書き込み所要時間を 1 と相対化してグラフ化したのが**図 4** である．適用後の値が小さいほど，より短時間でデータを書き込めたことを意味している．実装を単純にするために，評価のために本手法を適用したテープ･ドライブは，常に不定期転送速度低下モードを適用するよう実装した．本手法を適用することにより，ファイル・サイズが 100MB の場合にはデータ書き込み所要時間が 32%，1MB の場合には 35%，32KB の場合には 64% 削減できることが確認できた．

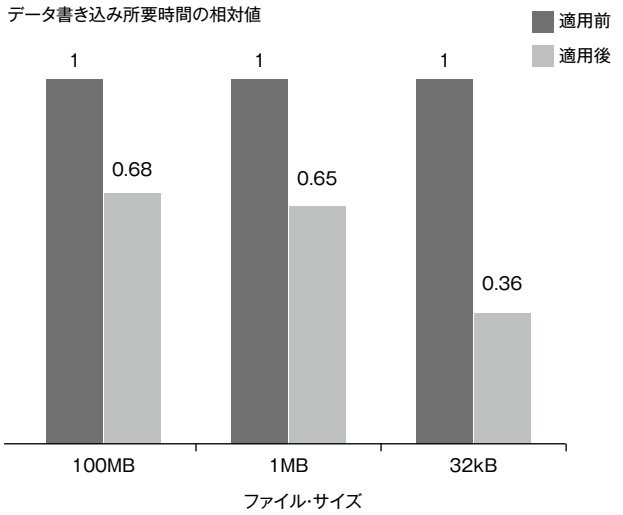

図4. 本手法適用前後のデータ書き込み所要時間

ファイル・サイズが小さい方が本手法を適用する効果が大きいことは，容量当たりのコピー所要時間が長いこと，すなわち転送速度が低いことに起因している．これは，ファイル・サイズが小さいほど，ファイル名を始めとするファイル毎のメタデータを処理する所要時間の影響が効いてくるためである．リポジショニングの所要時間は走行速度が速いほど，つまり転送速度が速いほど長くなるので，本手法を適用することによってリポジショニングの発生頻度が低下することによる転送速度の差分（改善幅）としてはファイル・サイズが大きいほど大きくなる．一方，ファイル・サイズが小さいほど転送速度が低いため，転送速度の改善幅が当初の転送速度に占める割合（改善率）は改善幅とは逆にファイル・サイズが小さいほど高くなる．これが，ファイル・サイズが小さいほど，データ書き込み所要時間の削減効果が大きくなる理由である．

6. おわりに

本論文では，サーバー転送速度が変動する場合のデータ書き込み所要時間を削減するために，新たに不定期転送速度低下モードという走行速度選択アルゴリズムを考案し，スピード・マッチング機能に追加した．

本手法をテープ･ドライブ上で実装し評価を行った．本手法を用いない場合と比較すると，少なくとも 3 割，効果が大きい場合には 6 割以上，データ書き込み所要時間を削減できることを確認した．

本手法はデータの書き込み時のみならずデータの読み出し時にも効果を発揮する．本手法は，特許公開 2010 − 113739 として，特許に出願されている．

参考文献

［1］ Kahn, M., MacFarland, A.: Dealing with Cool and Cold Data − and Getting It "Just Right",The Clipper Group Navigator, Report #TCG2010031, http://www.clipper.com/research/TCG2010031.pdf
［2］ 今井 慶太郎：ビッグ・データ時代を制するIBMのバックアップ術 第4章　データのアーカイブ，IBM，http://www.ibm.com/systems/jp/storage/column/backup/04.html
［3］ IBM:IBM Linear Tape File System, http://www.ibm.com/systems/jp/storage/products/tape/ltfs/
［4］ IBM：IBM System Storage TS2350 テープ・ドライブ Express, http://www.ibm.com/systems/jp/storage/products/tape/2350/
［5］ 電子情報技術産業協会テープストレージ専門委員会：ファイルベースワークフローにおけるLTOとLTFSの活用−テープのファイルシステムLTFSとLTOの最新世代 Generation 6の紹介− ,（2013）.
［6］ IBM：第2回 開発の現場にみる進化の軌跡 Chapter2，http://www.ibm.com/systems/jp/storage/column/proventapetech/02/chapter2.html
［7］ Reine, D.: IBM Restores Order to Data Center Storage LTOUltrium Generation 3 on Target,The Clipper Group Navigator, Report #TCG2005009, http://www.clipper.com/research/TCG2005009.pdf

日本アイ・ビー・エム株式会社
システム・テクノロジー開発製造
先進ストレージ開発
アドバイザリー・ソフトウェア・エンジニア

大石 豊 Yutaka Oishi

［プロフィール］
1999年日本IBM入社．ストレージ関連のソフトウェア開発を経て，2003年よりエンタープライズ･テープ･ドライブ，および，LTOテープ・ドライブの開発に従事．現在はLTFSの開発を担当．IBM Master Inventor.

日本アイ・ビー・エム株式会社
システム・テクノロジー開発製造
テープドライブ開発

片桐 隆司 Takashi Katagiri

［プロフィール］
2001年日本IBM入社．2002年よりエンタープライズ・テープ・ドライブ，および，LTOテープ・ドライブの開発に従事．IBM Master Inventor.

日本アイ・ビー・エム株式会社
システム・テクノロジー開発製造
ストレージ・システムズ開発

名倉 寛尚 Hironobu Nagura

［プロフィール］
1991年日本IBM入社．光磁気ディスクドライブの開発および製造を経て，1997年よりエンタープライズ・テープ・ドライブ，および，LTOテープ・ドライブの開発に従事．現在はLTFSの開発およびビジネス開発を担当.

日本アイ・ビー・エム株式会社
システム・テクノロジー開発製造
先進ストレージ開発

太田 由美子 Yumiko Ohta

［プロフィール］
1986年日本IBM入社．ホスト端末エミュレータの開発を経て、NASなどストレージ関連のソフトウェア開発に従事．現在はLTFSのテストを担当.