特集 8

UDC 681.527.07

仮想ストライピング機能を有するRAID-5型ディスクアレイ

Performance Evaluation of RAID-5 Disk Arrays with Virtual Striping

茂木和彦*・喜連川　優**

Kazuhiko MOGI and Masaru KITSUREGAWA

1. はじめに

外部記憶装置の高性能化・高信頼化を目的としたディスクアレイ[1]の研究・開発が活発に行われている．RAID[6]はデータと冗長情報の記録方式から5つのレベルに分類され（冗長情報を記録せずにデータのストライピングのみを行ったものを含めて6つのレベルに分類されることもある），トランザクション処理やUNIXのファイルシステムのように，アクセスされるデータの大きさは小さいが，そのI/O負荷が大きな用途に対してはレベル5（RAID 5型）がよいとされている．

RAID 5型の構成を図1に示す．各ディスクに配置されたデータからパリティストライプと呼ばれるグループを作る．データを格納するとともにそれらからパリティを計算し，記録することによりミラーディスクに比べて低記録コストで高信頼化が図られているが，パリティの導入により，データの書き込み（更新）時にはパリティの更新が必要となる．

従来の方式では，パリティストライプは物理的に固定的に配置されていた．そのため，パリティストライプ中の一部のデータを更新する時には以下のようにして新しいパリティを計算する必要がある．

New Parity = Old Data ⊕ New Data ⊕ Old Parity

したがって，データの更新時には，パリティの更新に必要な古いデータとパリティを読み出さねばならず，それに伴う性能の低下が大きな問題となっている．

そこで，このデータ更新時の性能低下を抑えるため，「仮想ストライピング」[2)~4)]と名付けたパリティグループの再編成を基本とする方式を現在検討している．この方式

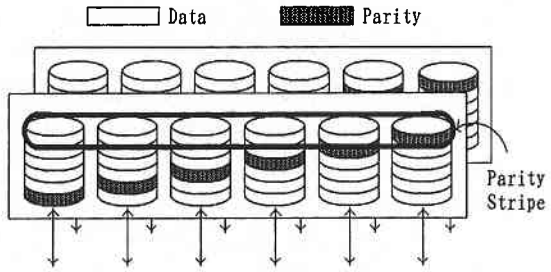

図1　RAID 5型ディスクアレイの構成

ではパリティストライプを仮想化することにより，データの含まれているパリティストライプの動的組み替えを行う．この組み替えにより，書き込まれるデータだけでパリティの計算が可能となり，データ書き込み時の性能が向上することが期待される．仮想ストライピングでは，仮想ストライプテーブルを用いることによって物理ブロックアドレスと論理ブロックアドレスの変換を行う．これにより，ガーベジコレクション時においてもデータの移動は発生せず，テーブルの組み替えのみの操作に帰着する．以下，2章で仮想ストライピングの機構について説明する．3章でソフトウェアシミュレーションによる性能評価を示す．

2. 仮想ストライピング

仮想ストライピングの概念図を図2に示す．パリティストライプは，従来のRAID 5型ディスクアレイと異なり，

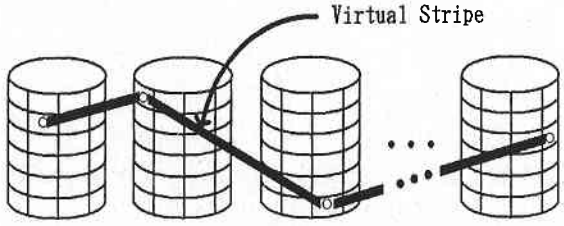

図2　仮想ストライピングの概念図

*東京大学生産技術研究所　第3部
**東京大学生産技術研究所　付属機能エレクトロニクス研究センター

各ディスクの任意のブロックから構成することが出来る．以下，仮想ストライピングのシステム構成について説明する．

2.1 パリティストライプの仮想化

仮想ストライピングを用いた場合の機構上の特徴として，以下の2点が挙げられる．

1．データの物理アドレスと論理アドレスの分離
2．パリティストライプの仮想化

1．はデータの物理的な位置の変更を可能とするために必要な機構である．また，2．によりデータの物理アドレスとその属するパリティストライプは動的に変更可能となる．

仮想ストライピングでは，パリティストライプの仮想化のため，その組み合わせを管理するテーブルが必要になる．これを仮想ストライプテーブルと呼び，図3のような構成をとる．このテーブルには，それぞれのストライプを区別するストライプ番号と，そのストライプを構成するデータのシリンダ番号とブロック番号が記録される．各ストライプのパリティディスクは，ストライプ番号から計算される．ガーベジコレクションのために，各ストライプのダーティブロック（データは別の場所に移動したが，パリティを計算するために内容を保存しておく必要があるブロック）の数も記録する．

2.2 パリティストライプの動的割付け

データの更新時には書き換えるデータのみで新しいパリティストライプを構成し，ディスク上の空きストライプに記録していくこととし，これをパリティストライプの動的割り付けと呼ぶ．

ストライプの動的割り付けを実行しnブロックからパリティが計算されている場合，nケのデータの書き込みに必要なディスクへのアクセス数は総計n×D+Pアクセスで済む．（ここでDはデータディスク，Pはパリティディスクを指す．）一方，従来の方式では古いデータとパリティの読み出しのために総計n×(2D+2P)アクセスが必要であった．したがって，パリティストライプの動的割り付けを行うことによりアクセス数が大きく減少し，データ更新時の処理負荷の軽減が図れる．

書き込み可能な空きストライプが存在しない場合，通常の4アクセスを必要とする書き込みを行う．ただし，書き込むデータは最もダーティブロックの多いストライプのダーティブロックへと書き込むようにストライプの変更を行う．同じストライプへの書き込みが多数あった場合，パリティアクセスの共有化を行うことができアクセス数を減らすことができる．

2.3 ガーベジコレクション

書き込み時に新たなパリティグループを作り続けていくと，空ストライプは最終的には消費しつくされてしまう．一方，更新前のデータはパリティの計算に使用されているためダーティブロックとなっている．したがって，そのままでは新たなデータを書き込むことは不可能であり，ダーティブロックをまとめ，新しく書き込み可能な空きストライプを作る必要がある．この操作をガーベジコレクションと呼ぶ．

新しく空きストライプを作るには，その元となるストライプ（以下ヴィクティムストライプと呼ぶ）を決め，その中のアクティブブロック（データが存在するブロック）を他のストライプのダーティブロックと置き換える必要がある（図4）．これに必要な動作は，置換を行うダーティブロックの内容を，ヴィクティムストライプ上のアクティブブロックの内容に置き換えることである．したがって，仮想ストライピングを用いない場合のデータ更新と同様に，この実行にはパリティの更新が必要である．このとき，ス

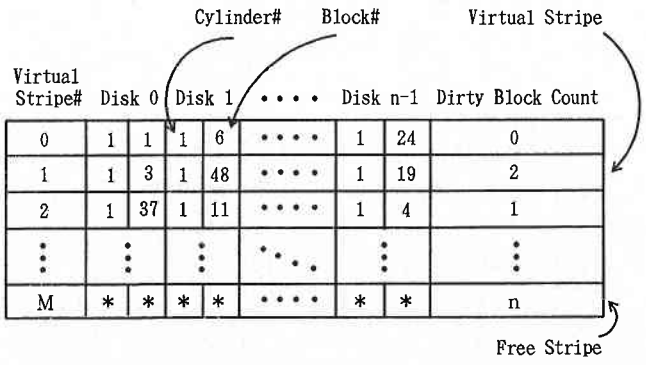

図3　仮想ストライプテーブル

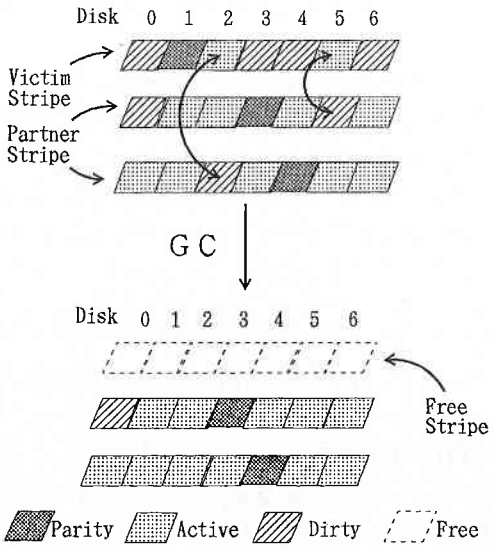

図4　ガーベジコレクション

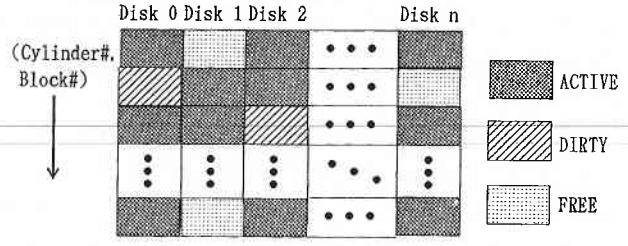

図5　物理ブロック状態テーブル

トライプの仮想化によりパリティストライプは論理的に組み替えが可能であり，実際にデータの移動は行う必要がない．

このとき，ヴィクティムストライプ中のダーティブロックの置換は必要がない．つまり，ダーティブロックが多いストライプをヴィクティムストライプとして選択することにより，空きストライプを作成するために必要なアクセス数を減少させることが可能となる．したがって，ガーベジコレクションの負荷を軽くするため，ヴィクティムストライプはダーティブロックを多く含むものから選択する必要がある．このダーティブロックの多いストライプの検索は，仮想ストライプテーブルに記録されたダーティブロック数を利用して行う．

ガーベジコレクションで置換を実行するブロックを選択するため，各ブロックがアクティブ，ダーティ，フリーのいずれの状態にあるか知る必要がある．そのため，各ブロックの状態を物理ブロック状態テーブルに記録しておき（図5），必要な時にこのテーブルからブロックの状態を調べ，置換を実行するダーティブロックを選択する．

ガーベジコレクションには，LFSにみられるごとくコピーを主操作とするものと，スレッド化するものが考えられるが，ここではガーベジコレクションコストの低減を図るべく，後者の方式を採用している．

2.4　フローティング/アクセススケジューリング

フローティング[5]，アクセススケジューリング[7]共にこれまでにディスクアクセスの高性能化手法として考案されているものである．仮想ストライピングでは，仮想化を行う際にフローティングの考えを利用するとともにアクセススケジューリングを組み込んでいる．

3．シミュレーションによる評価

ソフトウェアシミュレーションにより，仮想ストライピングの効果とその性能を調べる．

3.1　シリンダ固定仮想ストライピング

仮想ストライピングの基本機構については2章で述べたとおりであるが，種々の変形が考えられる．今回は以下に述べるような方式でシミュレーションを行った．

ガーベジコレクションにおける1ブロックの置換には4アクセスが必要であり，その処理は大きな負荷となることが予想される．そこで，パリティストライプはシリンダに固定して仮想化するという制約条件を付ける．また，パリティの配置はシリンダ単位で各ディスクを巡回させる．1回のガーベジコレクションで，対象とする1シリンダ中で実行可能な処理をすべて実行する．この方式により，ガーベジコレクションのアクセスはすべて同じシリンダへのアクセスとなり，スケジューリングによりそのコストを下げることが可能となる．

ガーベジコレクション処理が通常の負荷へ与える影響を減らすため，その実行法については種々の変形が可能であるが，今回はガーベジコレクション処理のアクセスを一括してすべてのディスクへと発行してフリーストライプを作成した．パリティストライプの組合せはシリンダ内で固定されているため，一括処理によりパリティへのアクセスに関してスケジューリングの効果が大きく出ることが期待できる．しかし，一括処理のため，ガーベジコレクションの実行はすべてのディスクに影響が及ぶ．通常負荷への影響を最小限に押さえるため，ガーベジコレクションはできるだけ多くのディスクがアイドル状態の時に実行する必要がある．そこで今回は，i台以下のディスクのみビジー状態の時にガーベジコレクションを実行することとした．（GC−iと名付ける．）

3.2　シミュレーション環境

ディスクのモデルとしてIBM 0661 Model 370（Lightning）を用いる（表1）．ディスクアレイのデータのストライピングは4 KBを1ブロックとして行う．ディスク中の空きエリアは初期状態では各シリンダにほぼ均等

表1

capacity	318 MB
cylinders/disk	949
tracks/cylinder	14
sectors/track	48
sector size	512 bytes
revolution time	13.8ms
seek time	2 ms (min)
	12.5ms (avg)
	25 ms (max)
seek time model	seek(d)=
	$2.0 + 0.01 \cdot d + 0.46 \cdot \sqrt{d}$
track skew	4 sectors

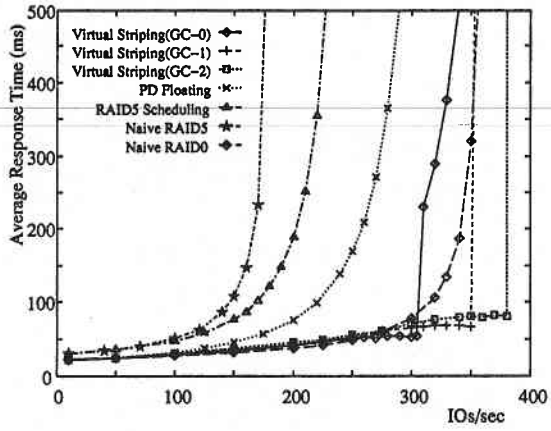

図6　8D+P構成時での性能

に分散しているとする．

読み出し/書き込み（更新）のアクセスの割合は双方とも50%とし，アクセス位置は一様分布をなすと仮定する．アクセスのデータサイズは4KB（1ブロック）の固定長とする．ディスクアレイに到着するアクセス要求の時間間隔は負の指数分布と仮定する．

ディスクアレイのコントローラについては次のような仮定を置く．各ディスクのスケジューリングは独立に行う．アレイコントローラには十分なメモリが存在すると仮定し，そこでのスケジュールのためのオーバーヘッド時間は無視できると仮定する．

パリティメンテナンスの処理はバックグラウンドで行われているとし，データ更新時のレスポンスタイムは，データがディスク上に書き込まれた時点での時間を用いて計算する．

シミュレーションは，仮想ストライピングの場合は，ディスクの台数が8D+Pの構成について，400万アクセス終了後から，その他の場合は（空き領域のブロック数）×2アクセスの終了後から測定を開始し，その値は測定開始後の10万アクセスの平均値を取る．

3.3　静的負荷に対する性能

8台のデータディスクに対して静的負荷を与えた時の平均レスポンスタイムを調べた．ディスクの使用率は80%とした．性能の比較のために以下に示す各方式の性能を測定した．

・仮想ストライピングを用いたRAID5型（GC−i，i=0，1，2）
・データ/パリティフロート＋アクセススケジューリングを行ったRAID5型
・アクセススケジューリングを行ったRAID5型
・RAID5原型
・RAID0原型（データディスクのみの構成）

結果を図6に示す．図の横軸は単位時間あたりにディスクアレイに到着する平均I/O数を表し，縦軸は平均レスポンスタイムを表す．

図6より明らかなように，仮想ストライピングは通常のRAID5型に比べて優れた性能を達成していることがわかる．

4．おわりに

仮想ストライピングを用いたディスクアレイについてその構成法ならびに制御方式を明らかにするとともに，シュミレーションによる性能評価結果を示した．今後より詳細な評価を進めてゆく予定である．　（1993年12月10日受理）

参考文献

1) 喜連川優：最近の二次記憶装置：ディスクアレイ，情報処理，Vol. 34, No. 5, pp. 642-651, 1993.5.
2) 茂木和彦，喜連川優，：動的パリティストライプの再編成によるRAID5型ディスクアレイの高性能化手法（仮想ストライピング）に関する基本検討，SWopp '93コンピュータシステム研究会，pp. 31-38，電子情報通信学会，1993.8.
3) Masaru Kitsuregawa, Kazuhiko Mogi: Virtual Striping: A RAID5 Storage Management Scheme with Robustness for the Peak Access Traffic, Proc. of the Int. Symp. on Next Generation Database Systems and Their Applications, pp. 280-287, 1993. 9.
4) 茂木和彦，喜連川優：仮想ストライピングによるRAID5型ディスクアレイの性能評価，電子情報通信学会　データ工学研究会，pp. 69-75, 1993. 9
5) J. Menon et. al., "Methods for Improved Update Performance of Disk Arrays", Proc. of 25th Hawaii Int. Conf. on System Science Vol. I, pp. 74-83, January 1992.
6) D. Patterson et. al., "A Case for Redundant Arrays of Inexpensive Disks(RAID)", Proc. of ACM SIGMOD, pp. 109-116. June 1988.
7) M. Seltzer et. at., "Disk Scheduling Revisited", Proc. of Winter 1990 USENIX Technical Conf., January 1990.