C-045

HPC 分野向け高速・大容量ストレージシステムの省電力化を図るアクセス予知階層ストレージの試作と省電力効果の検証

Experimental Verification of Efficiency of an Energy-Efficient High-Speed Tiered-Storage System (eHiTS) with Power-Aware Proactive Data-Allocation Method for HPC Systems

赤池 洋俊† 藤本 和久‡ 岡田 尚也‡ 三浦 健司‡ 村岡 裕明‡

Hirotoshi Akaike, Kazuhisa Fujimoto, Naoya Okada, Kenji Miura, Hiroaki Muraoka

1. はじめに

近年、IT 機器の消費電力は無視できないほど増加しており[1]、大きな問題となっている。ストレージシステムはその中でも多くの電力を消費するシステムの一つである。特に HPC 分野では、計算機の性能向上に伴いデータ容量が著しく増加していることから、ストレージシステムの大規模化と高性能化の要求が強く、今後さらなる消費電力の増加が予想される。スーパーコンピュータと接続するストレージシステムには大量のデータを高速に入出力することを目的として高い性能が要求される。そのため、性能を維持しながら消費電力を削減するストレージアーキテクチャと、その管理方式が求められている。

この背景のもと、図 1 に示すように階層ストレージ構成においてアクセス予知(図 1 中②)に基づくデータ配置(図 1 中④)と電源 ON/OFF 制御(図 1 中③①)を行う低消費電力化方式を提案した[2,4]。図 2 の①データ使用頻度に基づくデータ配置と②スピンダウン制御を行う従来方式と比較して、システム容量 1024TB の場合での試算では、提案方式は性能を維持しながら消費電力を 50%以上削減する見込みを得た[2,3,4]。

本研究では、提案した低消費電力化方式を試作ストレージシステムに実装し、実際に消費電力を測定することで省電力効果を検証する。この試作ストレージシステムのことをアクセス予知階層ストレージと呼ぶ。

2. 提案方式の概要

提案方式は従来方式と同様に、高性能なオンラインストレージ（以下、OL）と大容量のニアラインストレージ（以下、NL）の階層ストレージ構成をとる(図 1)。スーパーコンピュータではジョブという単位でデータを処理しており、ストレージに対してデータアクセスを行う。

提案方式と従来方式の違いはストレージの管理方式にあり、次の動作を行う。まず、①通常は全データを NL に保存し、ディスクの電源を OFF する。ただし、ユーザログイン時など、ディスクアクセスがある場合は、NL のディスクの電源を ON にする。次に、②アクセス予知により、ジョブ実行開始までの時間的余裕を予測する。③ジョブ実行開始前に、予めジョブに必要なデータを格納するディスクの電源を ON にし、④対象データを性能の低い NL から OL へコピーする。これにより、スーパーコンピュータがアクセスするデータは常に高性能な OL 上にあるため、高速データアクセスが可能で、性能は維持される。さらに⑤ジョブ終了後、ジョブで使用したデータを大容量の NL へ書き戻す。これにより、消費電力の大きい OL の高速 HDD の台数を削減できる。

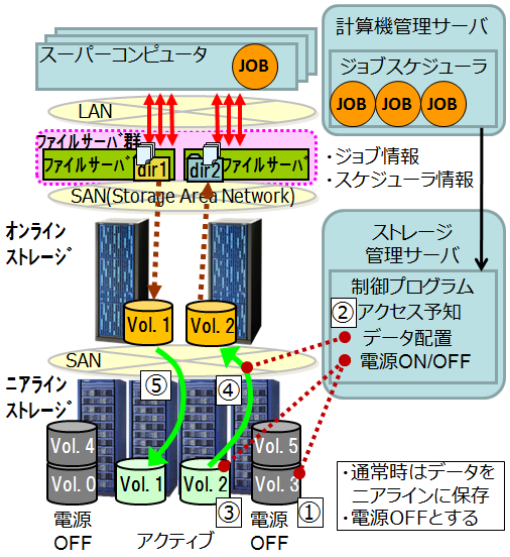


図 1. 提案方式の概要

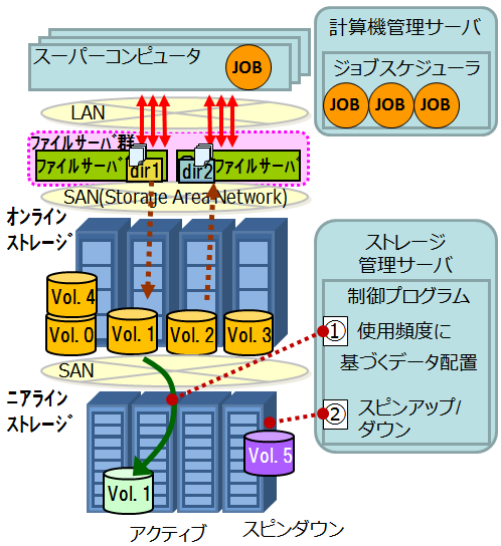
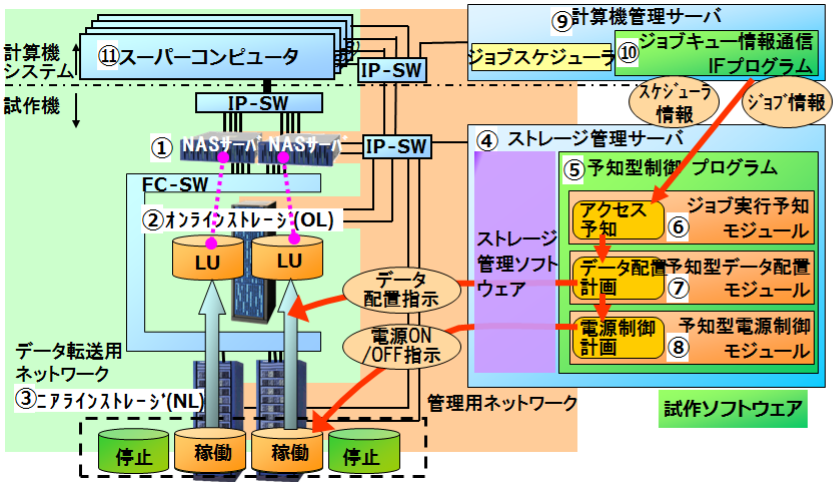


図 2. 従来方式の概要

この管理方式により、スーパーコンピュータがアクセスするデータのみを OL 上に配置することができる。通常全てのデータを OL に保存する従来方式と比べて、提案方式は OL のサイズを減らすことができ、HDD 数削減分の低消費

† （株）日立製作所 システム開発研究所
‡ 東北大学電気通信研究所



主要な構成機器の仕様

#	機器	主な仕様	台数
①	NASｻｰﾊﾞ	IP:1Gbps×12ﾎﾟｰﾄ FC:4Gbps×4ﾎﾟｰﾄ	2
②	OL	高性能エンタープライズストレージ 容量:12.6TB (高速HDD： 300GB-SAS/15krpm 48台)	1
③	NL	大容量ミッドレンジストレージ 容量:32TB (大容量HDD： 1TB-SATA×40台(10台×4筐体)) 機能:HDD電源ON/OFF ,スピンアップ/ダウン制御	2
④	ｽﾄﾚｰｼﾞ管理ｻｰﾊﾞ	Windows Server 2003	1

図 3. 試作機のハードウエアとソフトウエア

電力化が可能となる。そして、アクセスされない NL 上のディスクを電源 OFF にすることで、従来のスピンダウンよりも更なる電力を削減している。ところが、電源 OFF は電力を消費しない一方で、電源 ON に数分の時間が必要になる。スピンアップ時間の十数秒と比較して、電源 ON 時間は長く、応答時間のペナルティが大きい。そこで、提案方式では、アクセス予知によって予め電源 ON することで問題を解消している。

アクセス予知では、ジョブ情報とジョブスケジューラで管理されているジョブのリストや状態を含むスケジューラ情報をヒントにして、ジョブのアクセス先データの特定とジョブ実行開始までの時間的余裕の予測を行う。提案方式は、アクセス予知に基づき、ジョブ実行開始前に予めアクセス先データを NL から OL にデータ配置する。そのため、OL には実行中のジョブのアクセス先データと、予めデータ配置されたデータが保存されている。アクセス予知における時間的余裕の予測精度が高いほど、アクセス先データをジョブ実行の直前に配置することができる。理想的には、OL に実行中のジョブのアクセス先データのみを保存すればよい。この様に時間的余裕の予測精度が高いほど、OL のサイズを削減でき、消費電力削減効果が向上する。データ配置についてみると、従来方式ではアクセス頻度の低いデータを OL から NL へデータ配置している。これに対して、提案方式ではアクセス前に使用するデータを予め NL から OL へデータ配置しており、データ配置のタイミングと方向が従来方式と逆に管理されている点に特徴がある。

3. アクセス予知階層ストレージの試作機開発

3.1 試作機のハードウエアとソフトウエア

試作機のハードウエアを図 3 に示す。OL は高速 HDD を搭載した高性能エンタープライズストレージ(図 3 中②、容量 12.6TB)、NL は大容量 HDD を搭載した大容量ミッドレンジストレージ(図 3 中③、容量 32TB) 2 台で構成した。NL には HDD の電源制御機能があり、ディスク筐体という単位で電源 ON/OFF, スピンアップ/ダウンの制御が可能である。試作機には 2 台の①NAS(Network Attached Storage)サーバを配置しており、これがスーパーコンピュータに対してファイルサービスを提供する。④のストレージ管理サーバは、提案方式を実装した試作ソフトウエアを実行するとともに、試作機の各構成機器を管理するサーバである。試作機外観を図 4 に示す。

図 3 に示すように、試作ソフトウエアは⑤予知型制御プログラムと⑩ジョブキュー情報通信 IF プログラムから構成する。計算機システム上の⑨計算機管理サーバで動作するジョブキュー情報通信 IF は、提案方式のアクセス予知に用いるスケジューラ情報とジョブ情報をスケジューラから取得し、予知型制御プログラムに送信する。予知型制御プログラムは、アクセス予知と予知に基づくデータ配置と電源制御を行うプログラムで、ジョブ実行予知モジュール、予知型データ配置モジュール、予知型電源制御モジュールの 3 つから構成している。

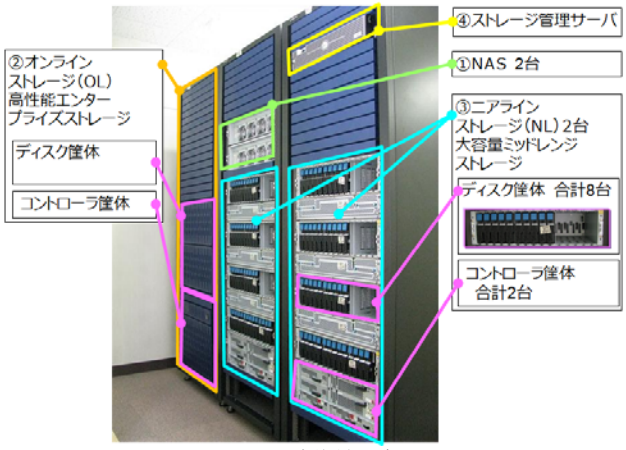


図 4. 試作機外観

3.2 予知型制御プログラムの動作概要

試作ソフトウエアは、スーパーコンピュータでのジョブ実行をスケジューリングするジョブスケジューラに連動して動作する。スーパーコンピュータのユーザは図 5 に示す様なジョブスクリプトでジョブを定義し、ジョブスケジューラに投入する。図 6 に示すように、ジョブスケジューラは投入されたジョブに id を付け、ジョブをキュー内に待機させる。スーパーコンピュータで前のジョブが終了すると、キュー内の先頭ジョブを、すなわち FIFO(First In First Out)で、スーパーコンピュータに転送し、ジョブを実行する。予知型制御プログラムは、ジョブスケジューラのキューで待機中のジョブが使用するデータを、そのジョ

ブ実行前に予め NL から OL へデータ配置する。今回の試作では提案方式の省電力効果を検証するため、ジョブ投入直後にデータ配置を実行する、最も簡単な実装を行った。同時に、ジョブ投入時には常に数個のジョブが待機している状態となるように、ジョブの投入条件を選んだ。この様な実装と、ジョブ投入条件の選択により、ジョブのアクセス先データがジョブ実行前に OL 上にデータ配置済みになる。

予知型制御プログラムの動作を図 7 に示す。まず、予知型制御プログラムは、アクセス予知に必要となるジョブ情報とスケジューラ情報を取得する。これは⑩ジョブキュー情報通信 IF(Interface)プログラムが行う。ジョブキュー情報通信 IF プログラムは、ユーザがジョブスケジューラに投入したジョブスクリプトを読み取ってジョブ情報を取得し、そしてジョブスケジューラに要求コマンドを出してスケジューラ情報を取得する。取得したジョブ情報とスケジューラ情報は予知型制御プログラムへ送信する。

次に、ジョブ情報とスケジューラ情報を元にアクセス予知を行う。アクセス予知では、⑥ジョブ実行予知モジュールが、ジョブのアクセス先データの特定とジョブ実行開始までの時間的余裕の予測を行う。まず、ジョブのアクセス先データの特定は、以下の様に行う。ジョブキュー情報通信 IF から受信したジョブ情報(図 5)を解析して、入力ファイルと出力ファイルの名前および格納場所を示す入力ファイル・出力ファイルの情報を取り出し、ジョブごとに入出力ファイル情報を作成する。そして、ジョブごとに、その入出力ファイルを格納する NL の論理ボリューム(Volume)を「ファイル-ボリューム対応テーブル」から導き、ジョブのアクセス先データを論理ボリューム単位で特定する。これを、ジョブのアクセス先ボリュームと呼ぶ。なお「ファイル-ボリューム対応テーブル」は、ファイルと、それを格納するボリュームを対応付けるテーブルである。続いて、ジョブ実行開始までの時間的余裕の予測は、以下の様に行う。ジョブキュー情報通信 IF から受信したスケジューラ情報(図 6)のジョブの並びから、待機中(ジョブ状態=Q)のジョブを集めて待機ジョブの並びに変換する。この待機ジョブの並びを元に、ジョブの実行開始までの時間的余裕を予測する。前述の通り、今回のジョブ投入条件により、ジョブ実行開始までにジョブのアクセス先ボリュームを NL から OL にデータ配置する時間的余裕が常にあるため、ジョブ投入直後にデータ配置を実行する。

そこで、待機ジョブの並びに新しいジョブが追加されると、⑦予知型データ配置モジュールは NL から OL へのデータ配置の指示を出す。現在の試作機の仕様では、データ配置を論理ボリューム単位で実行する。そのため、アクセスされる NL のボリュームを OL の空きボリュームに移動することで、データ配置する。配置先の OL の空きボリュームは、「OL 上空き Vol リスト」から決定する。予知型データ配置モジュールは、新しいジョブが追加されるごとに、この配置先と配置元ボリュームの組をデータ配置計画として作成し、ストレージ管理ソフトウエアを通してデータ配置指示を出す。

NL のボリュームは通常電源 OFF のため、データ配置前に予め対象ボリュームの電源 ON が必要になる。これに対して⑧予知型電源制御モジュールは、データ配置計画の対象ボリュームに対応する NL ディスク筐体に、ストレージ管理ソフトウエアを通して電源 ON 指示を出す。NL ディスク筐体は、「ボリューム-ディスク筐体対応テーブル」から導く。「ボリューム-ディスク筐体対応テーブル」は、ボリュームとそれを格納するディスク筐体を対応付けるテーブルである。データ配置完了後は、再び NL ディスク筐体の電源を OFF にして消費電力を削減する。

```
#!/bin/sh  (ジョブスクリプト)
inputfile=/home/USER01/TEST1/input.txt      ←入力ファイル情報
outputfile=/home/USER01/TEST1/output.txt    ←出力ファイル情報

omprun /home/USER01/TEST1/simulation1.exe ¥←実行ファイル
            ${inputfile} ¥
            ${outputfile}
```

図 5. ジョブ情報の例（ジョブスクリプト）

```
2009/03/25 21:02:01
Job id          Name    User      Time Use S Queue
--------------- ------- --------  -------- - -----
00018           job_03  USER01    01:11:34 R PX
00022           job_04  USER02           0 Q PX
00026           job_05  USER03           0 Q PX
00030           job_06  USER04           0 Q PX
00032           job_07  USER05           0 Q PX
00037           job_08  USER06           0 Q PX
00038           job_09  USER07           0 Q PX
```

ジョブの並び　　ジョブ状態 (R=実行状態、Q=待機状態)

図 6. スケジューラ情報の例
（ジョブキュー内のジョブ状態の情報）

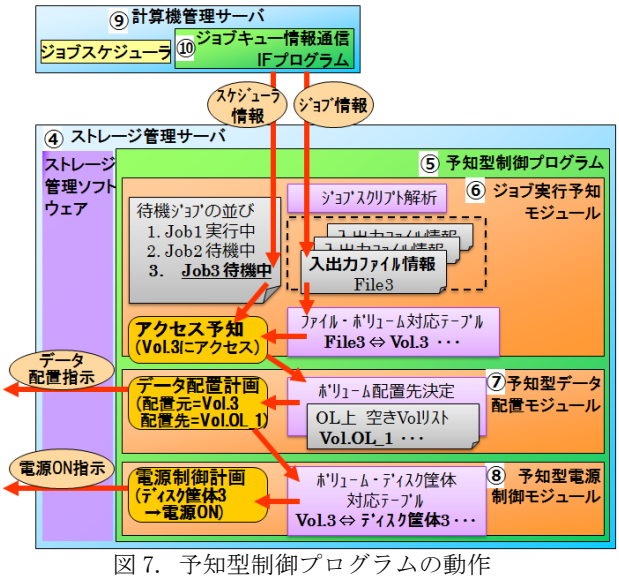


図 7. 予知型制御プログラムの動作

3.3 本試作でのデータ配置

図 3 に示すとおり、NAS サーバはスーパーコンピュータに対してファイルサービスを提供している。データ配置によりデータの保存場所が移動することになるが、これによってファイルサービスが停止することは望ましくない。そこで本試作では、データ配置中においても配置中のボリュームに保存されているファイルのファイルサービスを常時継続できるように、OL の仮想ボリューム機能とボリューム移動機能を用いてデータ配置を実装した。図 8 にデータ配置の動作を示す。なお、提案方式のデータ配置の実装は、本試作の方法に限定されるわけではない。

①通常時、NL のボリュームは仮想ボリューム機能によっ

て OL 上の仮想ボリュームとして管理されている。仮想ボリュームには論理ボリュームへのパスが設定されている。NAS は OL 上の論理ボリュームをマウントして、データを Read/Write する。アクセス先の実データは実際には NL の論理ボリュームに保存されるが、NAS はそのことを関知せず、透過的にデータアクセスできる。

NL→OL へのデータ配置指示があると、OL はボリューム移動機能により実データを NL の論理ボリュームから OL の実ボリュームへ②データコピーを行い、データコピー終了後に③OL の論理ボリュームへのパスを OL の仮想ボリュームから実ボリュームへ切り替える。ボリューム移動機能には、データ移動時でもボリュームへのデータアクセスが可能という特徴がある。NAS はパス切替までの間は NL の論理ボリュームの実データにアクセスを行うが、パス切替後は OL の実ボリュームの実データにアクセスする。仮想ボリューム機能とボリューム移動機能により、NAS はボリューム配置毎にボリュームのマウント先を切り替える必要はなく、OL 上の論理ボリュームをマウントしておくだけで良い。さらに、アクセス予知毎に実行されるデータ配置の最中でも NAS はファイルサービスを継続できるというメリットがある。

ジョブ実行中、OL の論理ボリュームは④OL の実ボリュームにパスがつながっている。ジョブの実行終了後、OL→NL へのデータ配置指示があると、OL はボリューム移動機能により⑤データコピーを実行し、元の NL の論理ボリュームに実データを書き戻す。データコピー完了後、⑥OL の論理ボリュームへのパスを OL の実ボリュームから仮想ボリュームへ切り替える。NAS はパス切替までの間は OL の実ボリュームの実データにアクセスを行うが、パス切替後は NL の論理ボリュームの実データにアクセスする。

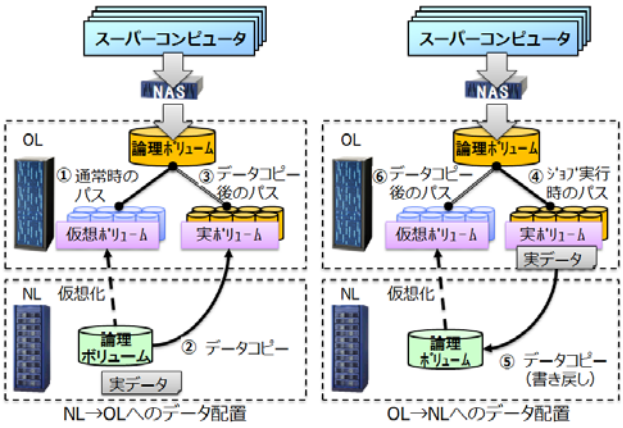


図 8. データ配置の動作

4. 省電力効果の検証

4.1 検証方法

省電力効果の検証では、まず 4.2 節で試作機に従来方式と提案方式を適用した時の消費電力を測定し、従来方式に対する提案方式の消費電力量削減率を評価する。

次に、4.3 節では 4.2 節で測定した消費電力のデータに基づいて、実用的なシステム容量 1024TB の場合で従来方式および提案方式の消費電力量を試算し、同様に消費電力量の削減率を評価する。

4.2 試作機の消費電力測定と測定結果

消費電力測定時の試作機構成を表 1 に示す。試作機の NL 容量 64TB に対して、従来方式では OL 容量を実環境の比率に合わせて 4.2TB に設定した。提案方式では、予知型のデータ配置により OL 容量を従来方式比 1/2 に削減できると仮定して、OL 容量を 2.1TB に設定した。

消費電力測定の条件を表 2 に示す。試作機へのデータアクセスは、スーパーコンピュータのジョブとユーザのアクセスがある。ここでは、ユーザのログイン時間を 1 日 1 回連続 8 時間と仮定した。提案方式では、データを通常 NL に保存するため、ログイン時のディスクアクセスを行う 8 時間は NL のディスク筐体の電源を ON 状態とする。その一方で、従来方式ではデータは通常 OL に保存されるため、ログインとは無関係に NL のディスク筐体は通常スタンバイ状態としている。

測定方法を図 9 に示す。試作機にスーパーコンピュータ(図 9 中⑪)を接続し、試作機にファイルを入出力する複数の計算ジョブを 24 時間投入した時の消費電力を電力モニタで測定した。今回は、100%全てのジョブにおいて、予知型データ配置モジュールがジョブ実行開始前に必要なデータを NL から OL にデータ配置するように、理想的な条件でジョブを投入した。ジョブには流体シミュレーションを行う実際のジョブ用いており、スーパーコンピュータで実行されたジョブは実際に試作機へファイルアクセスを行う。表 2 の測定条件に示す通り、1 時間毎に 1 ジョブを投入し、ジョブ実行時間は平均 1 時間とした。提案方式では、アクセス予知に基づき、データ配置を自動的に実行する。従来方式では使用頻度の低いデータを OL から NL へデータ配置する。ここでは、1 日 1 回 8 時間かけてデータ配置するものとした。

表 1 消費電力測定時の試作機構成

	従来方式	提案方式
試作機構成	OL容量:4.2TB NL容量:64TB OL:NL容量比=1:15	OL容量:2.1TB NL容量:64TB OL:NL容量比=1:30

表 2 消費電力測定条件

ユーザ数	8
ユーザ容量	1ユーザに8TBを割り当て (1ユーザにディスク筐体を 1つ割り当て)
ユーザログイン時間	1日1回連続8時間
ジョブ投入	1時間毎に1ジョブ投入
ジョブ実行時間	平均1時間
測定時間	24時間

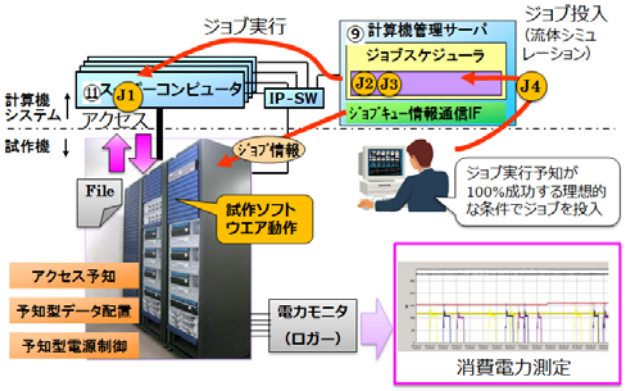


図 9. 測定方法の説明

測定の結果、提案方式を実装した予知型制御プログラムは正常に動作し、従来方式の消費電力量 71.6[kWh]に対して提案方式は 57.6[kWh](従来方式比 19.6%削減)だった。測定では OL、NL の消費電力を測定するとともに、ボリューム配置状態、ジョブ投入/実行のイベントの監視を行った。この中から提案方式と従来方式の主要な動作部分として、2 台ある NL の内の 1 台分の消費電力とそれに関連するイベントを取り出し纏めたものが図 10（提案手法）と図 11（従来手法）である。関連のあるジョブは Job1, 2, 3, 7, 8, 9, 13, 14 である。

図 10 の提案方式の測定結果をみると、(A)Job1 の投入後ディスク筐体 1 の電源 ON、次に(B)NL→OL へのデータ配置開始、データ配置完了後(C)再びディスク筐体 1 の電源が OFF になり、消費電力 0[W]が確認できる。図では、ボリューム配置状態として、データが OL に配置したことを、OL と記した横棒で示した。(D)ジョブ実行前にボリューム移動が完了しており、アクセス予知に基づくデータ配置に成功している。ジョブ実行後、(E)ディスク筐体 1 の電源 ON、(F)OL→NL へのデータ配置開始、そしてデータ配置完了後(G)再びディスク筐体 1 の電源 OFF が確認できる。この様に、予知型制御プログラムの正常動作を確認した。

次に、図 11 の従来方式の測定結果をみると、まず(A)で Job1 の投入イベントが発生している。通常データは OL に保存されているため、図の OL と記した横棒(ボリューム配置状態を示す)は全時刻に渡っている。ディスク筐体は通常スタンバイ状態で 112[W]の電力を消費している。従来方式のデータ配置のため、(B)スピンアップを行い、(C)データ配置を開始する。約 8 時間後の(D)データ配置完了の後に、(E)で再びスピンダウンする。

4.3 容量 1024TB のシステムの場合の試算

4.2 節の消費電力測定結果を元に、容量 1024TB のシステムに従来方式と提案方式を適用した場合の消費電力量をそれぞれ試算した。ただし、各方式で OL:NL 容量比は消費電力測定と同じ設定とした。結果を図 12 に示す。 1024TB の場合、提案方式は従来方式と比較して 52.9%の消費電力量削減が可能であることが分かった。

システム全体の容量が 512TB, 256TB の場合で同様の評価を行った。3 つのどの容量でも、提案方式は省電力効果があることを示している。そして、システム容量が大きいほど、従来法と比較して消費電力量削減の割合が大きくなることが分かる。提案方式はディスク筐体の消費電力を削減する方式なので、コントローラで消費される電力の割合が小さくなると、その分 OL と NL のディスク筐体の消費電力の割合が増加し、消費電力削減の寄与する割合が増すためである。

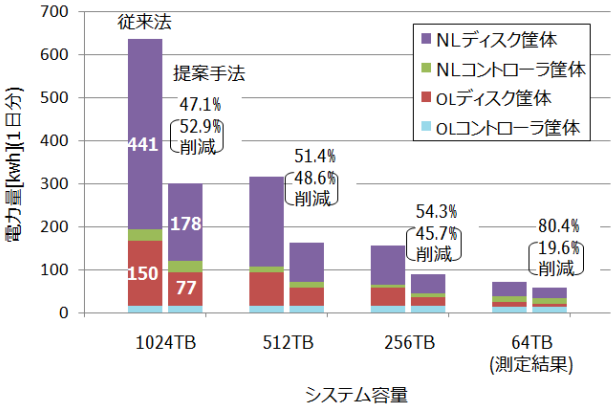


図 12. システム容量と電力量の関係

5. まとめ

提案する低消費電力化方式を実装したアクセス予知階層ストレージの試作機を開発した。スーパーコンピュータを接続した実環境で、システム容量 64TB の試作機の消費電力を測定したところ、提案方式は正常に動作し、従来方式比で消費電力量 19.5%の削減を確認した。実用的な容量 1024TB のシステムに換算した場合、消費電力量を従来比 50%以上削減する見込みを得た。今後は、OL・NL 間でデータ配置を効率化することで更に消費電力削減を目指す。そして、HPC 分野以外の高いデータ処理性能が要求される他のアプリケーションへの適用を検討する。

謝辞 本研究は、文部科学省の委託研究「高機能・超低消費電力スピンデバイス・ストレージ基盤技術の開発」の成果の一部である。

参考文献

[1] “Report to Congress on Server and Data Center Energy Efficiency Public Law 109-431”, U.S. Environmental Protection Agency, ENERGY STAR Program, Aug. 2007.

[2] 赤池洋俊, 藤本和久, 岡田尚也, 三浦健司, 村岡裕明, “HPC 向け高速・大容量ストレージの省電力化を図る階層ストレージアーキテクチャと階層管理方式の提案”, 第 71 回情報処理学会全国大会, 2009 年 3 月

[3] 岡田尚也, 藤本和久, 赤池洋俊, 三浦健司, 村岡裕明, “アクセス予測を利用した HPC 向け高速・大容量ストレージの階層管理方式における予測確率に関する検討”, 第 71 回情報処理学会全国大会, 2009 年 3 月

[4] Kazuhisa Fujimoto, Hirotoshi Akaike, Naoya Okada, Kenji Miura, and Hiroaki Muraoka, “Power-Aware Storage-Tiering Management for High-Speed Tiered-Storage Systems”, http://www.usenix.org/events/fast09/wips_posters/fujimoto_poster.pdf, USENIX Conference on File and Storage Technologies (FAST'09), Feb, 2009.

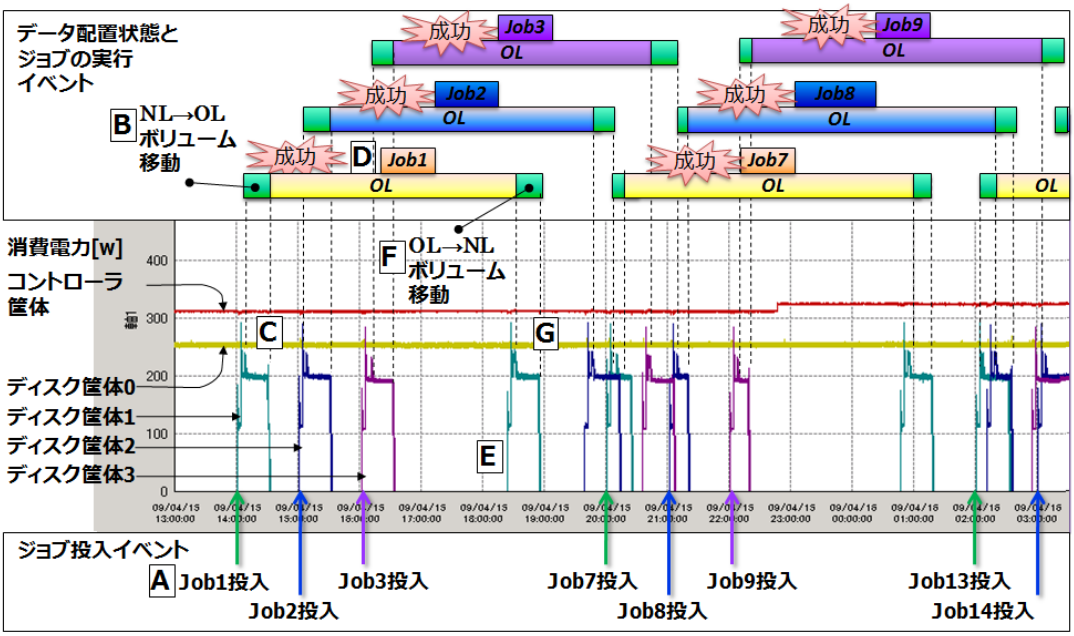


図 10. 提案方式の測定結果

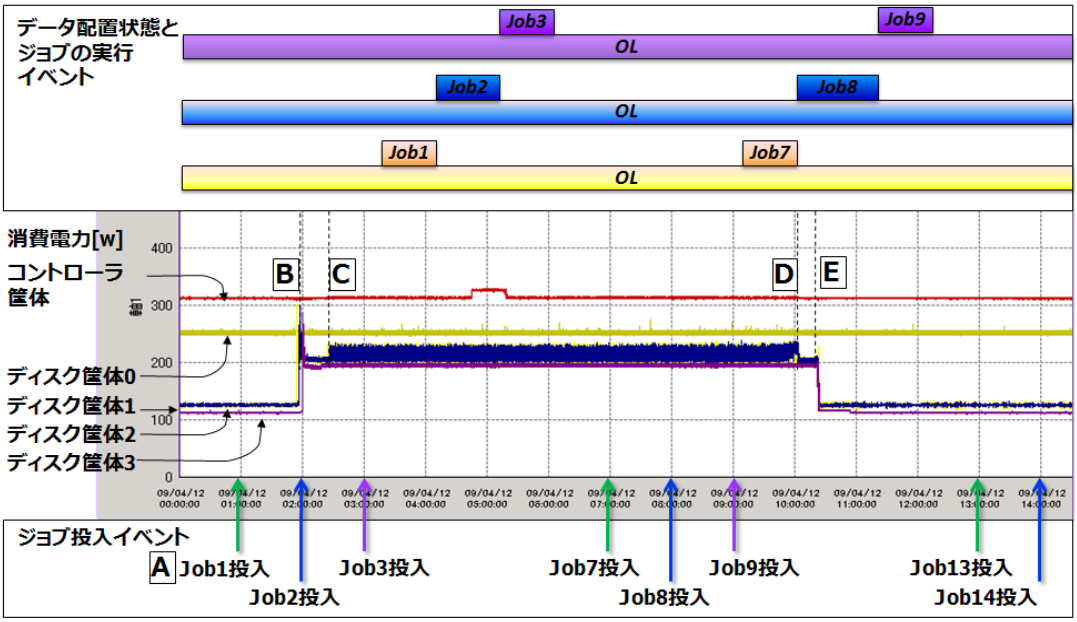


図 11. 従来方式の測定結果