

大容量記憶システム MSS の性能評価用 シミュレーション・モデルに関する考察†

根 岸 和 義^{††} 米 田 茂^{††}

大容量記憶システム (MSS) の内部処理方式の評価, および性能評価のため, シミュレーション・モデルを作成した. 従来, MSS の性能評価の研究としては, 解析モデルによるもの, シミュレーション・モデルによるもの等が発表されている. われわれのモデルは, ハードウェア, ファームウェア, ソフトウェアのサブモデルより構成され, ハードウェア・モデル中に, コマンド・チェーン処理機能を内蔵している. このモデル構成の特徴は, (1)ファームウェア・モデルの分離によりファームウェア・ロジックの変更, 評価が容易であること, (2) MSS 独特のステージ動作のためにディスク制御装置の使用率が高くなった場合でも, コマンドの処理ごとにチャンネルとディスク制御装置の結合がシミュレートされているので評価精度が保たれること, (3)ファームウェア, ソフトウェアの各モデルからのコマンド・チェーン制御が容易であることである. 本シミュレータの評価結果を実測値と比較検証し, 十分な精度が得られていることを示した. また, 本シミュレータの適用例として, ステージ連続処理を行った場合, ファームウェア処理方式により, その性能が大きく異なる場合のあることを示した.

1. ま え が き

計算機システムのオンライン記憶容量の増大化の要求に対処する手段として, 大容量記憶システム (MSS) が開発された¹⁾. MSS は従来のディスク・サブシステムと異なり, 複数の制御装置が協力して動作を行う, 複雑な分散制御システムである. MSS 全体を制御する大容量記憶制御機構 (MSC) は, ファームウェアとして従来のオペレーティング・システムと同様のデータ管理, アクセス・スケジューリング, MSS 内部資源の管理等の機能を有する. また, そのハードウェアは, CPU, チャンネルと同様のコマンド発行機能を他の制御装置に対してもち, ディスク・アクセス等の処理をコマンド・チェーン* によって行う.

このように, MSS は従来のディスク・サブシステ

ムと異なり, より複雑な処理を行っているため, その評価手段を新しく開発する必要があった. 従来, MSS 性能評価の研究として, Lavenberg²⁾ らのシミュレーション・モデルによるもの, Tsuruho³⁾ らの解析モデルを用いたものが報告されているが, MSS 内部処理に関しては, かなり簡略化したモデルである. また, この他に Hemy⁴⁾ による実機とハードウェア・モニタを使用したファームウェア評価法が報告されている.

われわれの研究目的は, MSS 基本性能, およびファームウェア処理方式の評価である. 評価に当たっては, MSS 特有のデータのステージ処理のため, ステージング・ディスク制御装置*(SDC) の使用率が従来のディスク制御装置に比較して高くなることから, ステージング・ディスクの制御装置におけるアクセスの競合を含めて行う必要がある. 前記, 従来の評価モデルでは, MSS 内部の制御ロジックを簡略化して扱っており, ファームウェア処理方式の詳細な評価は困難である. そこでわれわれは, MSS ファームウェア・ロジック**をファームウェア・モデルとして独立させた, ハードウェア, ファームウェア, ソフトウェアの

† Considerations in Simulation Model for MSS Performance Evaluation by KAZUYOSHI NEGISHI and SHIGERU YONEDA (Hitachi Systems Development Laboratory).

†† (株)日立製作所システム開発研究所

* CPU が DASD に対してデータ・アクセスを行うとき, データの位置付け動作 (シーク, セット・セクタ等), データのリード, ライト動作を, DASD の制御権をもったまま, 連続して実行する必要がある. CPU はこのような一連のコマンドを組にして作成し, チャンネルへ渡す. チャンネルは, コマンド列を受け取り, 制御装置を介して次々と DASD に転送し, 一連のコマンドで要求された動作を行わせる. 制御装置は, 一つのコマンドに対する DASD の動作終了後も, チャンネルから次のコマンドが送られてくるまで, DASD の制御権を確保する. このような, 一連のコマンド処理を, コマンド・チェーンと呼ぶ. MSC の場合は, MSC が CPU のコマンド列の作成機能と同時にチャンネルの機能も代行している.

* CPU, 大容量記憶制御機構からのディスク・アクセスの制御, およびデータ記録機構と DASD 間のデータ転送の制御を行う.

** 大容量記憶制御機構のデータ管理, アクセス・スケジューリングおよび, ステージング・ディスク制御装置のステージ処理等を行うため, 各装置のファームウェアによって行われる処理のロジック.

各サブモデルから成るシミュレーション・モデルを開発した。そして、前述のステージング・ディスク制御装置における競合を評価するため、チャンネルと制御装置間のコマンド・チェーンの処理モデルを、ハードウェア・モデルの一部として組み込み、ファームウェア、ソフトウェアの各モデルから、これをコールする形式とした。

モデルの特徴は、ファームウェア・モデルを分離したことにより、ファームウェア・ロジックの変更、評価が容易であること、制御装置の使用率が高い場合にもコマンドの処理ごとにチャンネルと制御装置の結合をシミュレートしていることによって評価精度が保たれることである。

本論文では、われわれが開発した MSS 性能評価用シミュレーション・モデルに関して、モデルの構成とチャンネル、制御装置のインタフェース処理に重点を置き記述する。また、評価結果と実測値の比較検討を行うとともに、その適用例を示し、われわれの採用したインタフェース処理方式の効果を明らかにする。

2. 対象システムの構成と性能評価上の留意事項

MSS の構成と、性能評価上考慮しなければならない点、および従来のシステムとの違いについて以下に示す。

2.1 MSS の構成

MSS は、図 1 に示すように計算機システムの一部を構成する。MSS は、3 種の制御装置(大容量記憶制御機構 (MSC)、ステージング・ディスク制御装置 (SDC)、アクセス制御機構* (ARC))、およびアクセス機構** (ACC)、データ記録機構*** (DRD)、データ記録制御機構**** (DRC)、ディスク接続装置***** (DSC)、DASD より構成される。

MSS 内部では、データは磁気テープ・カートリッジに DASD と同じフォーマットで記憶されており、必要な部分のみを DASD に移して (ステージ) 使用する。データの移動、および DASD 上のデータの管理は、ユーザプログラムからは見えない形で行われるため、ユーザはあたかも多数の DASD を直接使用

* アクセス機構の制御を行う。

** 磁気テープ・カートリッジを、貯蔵庫からデータ記録機構へ、またはその逆へ移動する。

*** 磁気テープ・カートリッジの読み出し、書き込みを行う。

**** データ記録機構の制御、およびこれとステージング・ディスク制御装置間のデータ転送を行う。

***** DASD と、ステージング・ディスク制御装置を接続する。

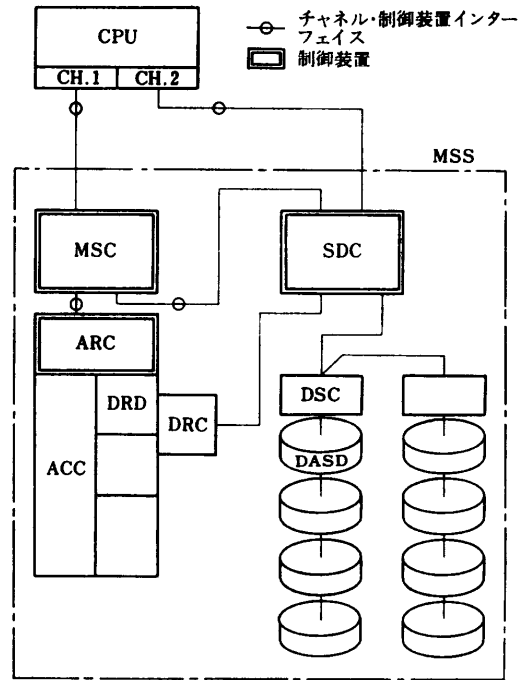


図 1 MSS の構成
Fig. 1 Configuration of MSS.

しているかのように、データへのアクセスができる。データの移動および管理を CPU に負担をかけずに実施するため、MSS ではこれらの機能の大部分を図 1 の二重枠で示した制御装置のファームウェアで実現している。

MSS 全体の制御は、大容量記憶制御機構によって行われる。ユーザプログラムがどの磁気テープカートリッジ内のどのデータセットを使用するかは、あらかじめ、オペレーティングシステムによって解析され、大容量記憶制御機構へコマンドの形で指令される。大容量記憶制御機構は、この指令を解釈して、ステージ用の制御ブロックを作成する。次に大容量記憶制御機構は、この制御ブロックに基づいてステージを行う。実際の処理は、大容量記憶制御機構が、アクセス制御機構、ステージング・ディスク制御装置にコマンドを発行して、行わせる。

ステージ用の制御ブロックを用いたカートリッジ処理につき、さらに詳細に述べる。図 2 に示すように、アクセス機構による磁気テープ・カートリッジの取出し、データ記録機構内でのカートリッジの自動ロード、1 回 8 シリンダまでのデータのステージのくり返し、カートリッジのアンロード、およびアクセス機構によるカートリッジの返却により、一連のカートリッ

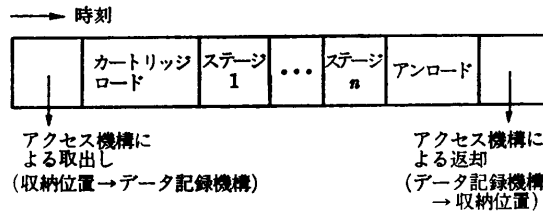


図 2 カートリッジの処理手順

Fig. 2 Process sequence of MSS cartridge.

ジ処理が行われる。

DASD 上で更新されたデータの磁気テープ・カートリッジへの収容 (ダステージ) も同様の処理手順により大容量記憶制御機構によって制御される。ユーザプログラムの終了時、オペレーティング・システムによって使用済データセットを磁気テープ・カートリッジに戻す指令が発せられる。この指令は、大容量記憶制御機構によってステージと同様に処理される。データの移動方向、および更新されたデータを含むシリンダのみが転送されること以外は、ステージの場合と同じ処理が行われる。以後、代表してステージ動作の解析を行う。

ステージ処理、カートリッジのアンロード処理の制御は、ステージング・ディスク制御装置によって行われる。ステージ動作時のデータ転送では、データ記録機構と DASD の転送速度の違いを吸収するため、ステージング・ディスク制御装置のバッファ (32k byte) が使用される。バッファの使用状況に従って、データ記録機構とバッファおよびバッファと DASD の間のデータ転送が非同期に行われるため、全体としてのデータ転送の定量的な解析は困難である。

データ転送は、データ記録機構からステージング・ディスク制御装置へは最大 1 シリンダ、ステージング・ディスク制御装置へは最大 4 トラック連続して行われる。大容量記憶制御機構からのステージ指令は、コマンドによって伝達され、最大 8 シリンダまでの処理を一度のステージ命令で指定できる。ただし、ステージング・ディスク制御装置では、前述の最大連続転送量を上限として、各装置を切離すことにより、同一装置を長時間占有することのないように処理を行う。

2.2 性能評価上の留意事項

従来の周辺記憶システムでは、制御装置は単に CPU から送られてくるコマンドの処理を行うだけであったが、MSS は、独自のデータ管理、転送機能を持っている。このため、性能評価に際し、下記の点に留意する必要がある。

(1) ステージング・ディスク制御装置は、CPU からのコマンド以外に、ステージ、ダステージのコマンドを大容量記憶制御機構から受け取り処理する。したがって、ステージング・ディスク制御装置の使用率は高くなり、コマンドの競合が生じやすくなる。

(2) MSS 独自の処理を行っているファームウェアの処理方式の評価を行う必要がある。

3. シミュレーション・モデルの構成

MSS 性能評価用シミュレータは、前章で述べた事柄を考慮して、性能に関係する動作を詳細にモデル化している。本章では、モデル化の方針、シミュレータの入出力、シミュレーション・モデルの構成、およびモデルの特徴を述べる。

3.1 モデル化の方針

モデル化に当たっては、MSS の性能 (応答時間、スループット) に着目し、これら进行评估することができるよう考慮した。具体的には、MSS 動作のうち、データ・アクセス動作、データ転送動作、およびディスク上のデータやリソースを管理するファームウェア・ロジック (エラー回復等を除く通常処理部分のみ) を抽出してモデル化した。CPU、チャンネルに関しては、MSS へのアクセスに関係する部分のみを、モデル化した。

3.2 シミュレータの入出力

シミュレータは、システム構成、負荷となるジョブのパラメータ、データ・セットの配置等を入力パラメータとして定義テーブルを生成し、これらとあらかじめ用意してあるロジックの処理ルーチンを前処理プログラムにより結合し、シミュレーション・モデルを生成する。このモデルを、GPSS⁵⁾ 形式のシミュレータでシミュレートする。この結果から、後処理プログラムによって応答時間、リソース使用率、トレース情報等を出力する。モデルの処理ルーチン部分は、必要に応じて前処理時に変更が可能である。

3.3 モデルの構成

モデルは、実機の変更等を反映することが容易なように、実機のハードウェア、ファームウェア (制御装置のマイクロコード)、ソフトウェアに対応した、ハードウェア・モデル、ファームウェア・モデル、ソフトウェア・モデルの三つのサブモデルより構成されている (図 3)。以下に、各サブモデルの詳細を示す。

3.3.1 ハードウェア・モデル

CPU、チャンネル、制御装置、その他のハードウェア

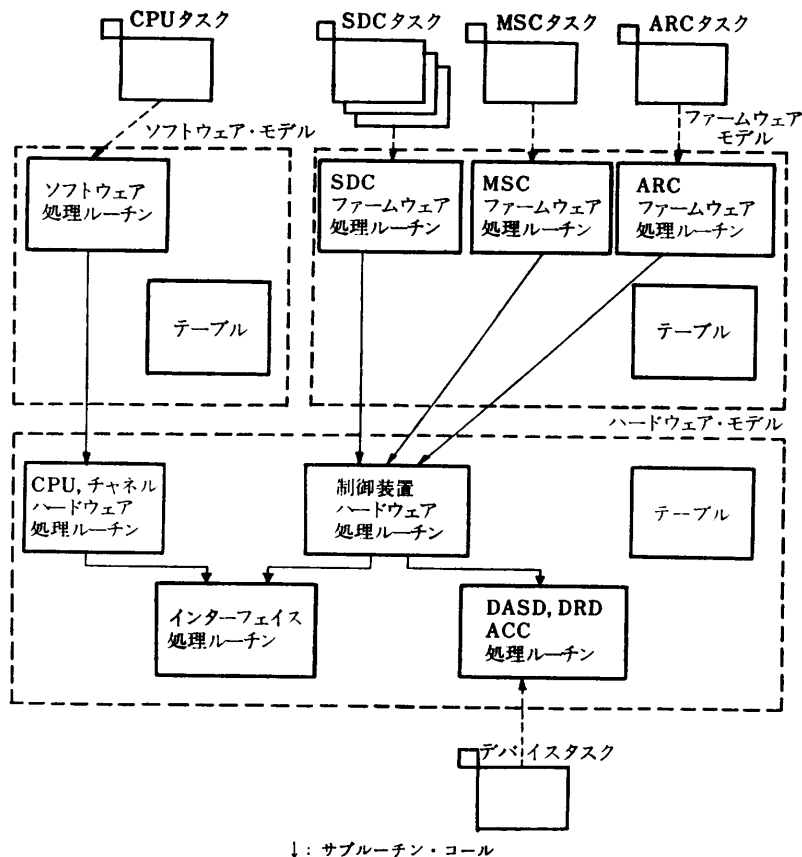


図3 MSS性能評価モデルの構成とタスク
Fig. 3 Configuration and tasks of MSS performance evaluation model.

の動作のうち MSS の性能に関係するものをモデル化している。モデルは、制御装置、CPU、チャンネル等のハードウェア処理ルーチン、および状態管理用のテーブルより構成され、テーブルの書き換えにより任意のシステム接続状態をモデル化することができる。また、制御装置ハードウェアは性能評価の観点からは、すべて共通の動作をすると考えられるので、統一した処理ルーチンにまとめてある。モデルの処理ルーチンは、基本的なハードウェアの動作をモデル化したものであり、変更されることはほとんどないと考えている。本モデルの処理ルーチンの一つとして、前述の、チャンネルと制御装置間のインターフェイスの処理ルーチンが組み込まれている。

3.3.2 ファームウェア・モデル

ファームウェア・モデルは、制御装置のファームウェアのうち性能に関係する部分をモデル化したものであり、おのおのの制御装置のタイプごとの処理ルーチンとテーブルより構成される。制御装置のハードウ

アを用いた動作、たとえばディスクのシーク開始等は、本モデルにより、ハードウェア処理ルーチンをコールすることによってシミュレートする。ファームウェア処理方式の検討に際しては、本モデルの処理ルーチンを変更することによって行う。

3.3.3 ソフトウェア・モデル

オペレーティング・システム、およびアプリケーション・プログラムの動作のうち性能に関係する部分をモデル化したものであり、処理ルーチンとテーブルより構成される。コマンド・チェーンは、本モデルにより作成され起動される。アプリケーション・プログラムの部分は、評価対象システムに応じて作成する必要があるが、シーケンシャル・ファイルのマルチジョブによる処理に対しては、モデルがあらかじめ用意しており、この場合には、パラメータの設定だけで使用することができるようにしている。

3.4 モデルの動作

シミュレーション・モデルは、前述のように、GPSS系の言語上で実現されており、複数のタスク（トランザクション）が並行して動作することにより、モデルをシミュレートする。モデルの動作のうち、コマンド・チェーンに関する部分を以下に示す。

3.4.1 タスクの並行動作

図3に示すように、CPU、各制御装置ごとにシミュレーション用タスクが用意されており、これらの並行動作をシミュレートする。また、これとは別に、ハードウェアの動作をシミュレートするために特殊なタスク（デバイスタスク）が別に用意されている。ハードウェアの動作をシミュレートするために、これらのタスクは、ソフトウェア処理ルーチン、ファームウェア処理ルーチンから、ハードウェア処理ルーチンをコールする。各制御装置、CPU間の伝達、同期は、ハードウェア・モデル内で、インターフェイス処理ルーチンを排他的にコールすることによって実現している。また、インターフェイス処理ルーチンから、各タスクの起動、割込みを行うことができる。

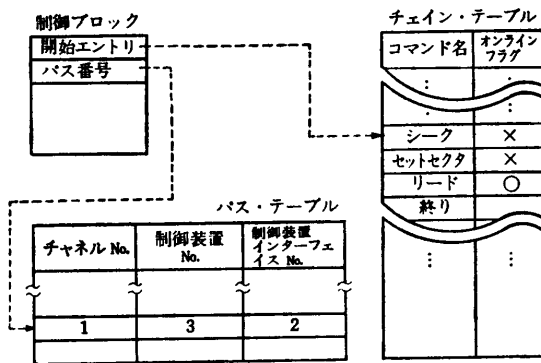


図 4 コマンド・チェーンの制御ブロックとテーブル

Fig. 4 Tables and control block for command chain.

3.4.2 コマンドの処理

コマンドの作成は、ソフトウェア処理ルーチン内で、実機のチャンネルコマンド語 (CCW) に対応する図 4 のような制御ブロックを作成することでシミュレートする。この制御ブロックを、インタフェース処理ルーチンを使用して、相手方制御装置タスクに渡すことにより、コマンドの伝送をモデル化している。インタフェース処理ルーチンは、制御ブロックの内容により使用するバスのチャンネル、制御装置の使用可能性を確認する。

3.4.3 コマンド・チェーン処理

実際のコマンド・チェーン処理をシミュレートするために、あらかじめチェーン・テーブルに登録されたチェーンのどのエントリを使用するかを、ソフトウェア処理ルーチンが、制御ブロック上で指定する。実機と同様に、一度チェーンが起動されれば、チェーンの終了、または制御装置やチャンネルが使用中のために起動ができなかった場合のほかは制御が戻ってくることはない。コマンドチェーン中のコマンドのなかには、処理終了まで一時的にインタフェースを切り離す形のコマンド (オフライン・コマンド) と接続したままのコマンド (オンライン・コマンド) があり、これらの区別を指定するフラグが、チェーン・テーブル上にコマンドごとに定義されている。インタフェース処理ルーチンがこのフラグを見てコマンド伝達後のインタフェースの状態を、接続中、または切離し状態とするので、ファームウェア・モデル、ソフトウェア・モデルでは、接続、切離しのための処理は不要である。

3.5 モデルの特徴

本方式は、従来の解析モデルに比較すると、シーク、セットセクタ時のチャンネル再結合を正確にモデル

化しており、再結合失敗による遅れを反映することができる。また、従来のシミュレーション・モデルでは、チェーンのパターンを個別にモデル化していたのに対し、これをテーブルとして統一的に扱うようにした。

本方式によれば、2.2 節で述べた問題点は下記のように解決されている。

(1) モデル内でコマンド・チェーンの中断、再結合を詳細にシミュレートしており、制御装置の使用率が高く、チェーン処理が中断される確率が大きくても、精度を落とさず評価が可能である。

(2) ファームウェア、ソフトウェアの変更が容易である。インタフェースの操作は、別モジュールとなっているため、変更の影響をあまり受けない。

4. 精度検証と適用例

作成したモデルの精度を検証するため、MSS にステージ動作を連続して行わせた場合の、実機ベンチマーク・ジョブによる単位時間当りのステージ件数、および同一条件におけるシミュレーションによるステージ件数を測定した。ステージの単位は、MSS に収容されると予想されるファイルの大きさを考慮のうえ、4 シリンダ、16 シリンダ、32 シリンダの 3 通りとした。図 5 にステージ単位に対する上記ステージ件数の相対値による比較結果を示す。図 5 の実線がシミュレーション結果であり、破線がベンチマーク・ジョブの結果である。実測結果に対するシミュレーション結果

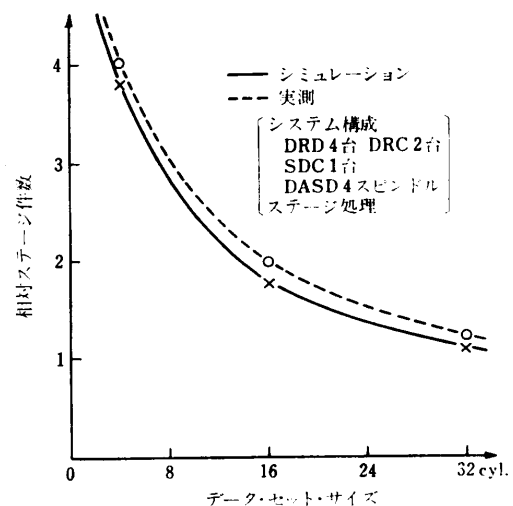


図 5 シミュレーションと実測によるステージ処理能力の比較

Fig. 5 Comparison between measured and simulated processing ability to stage data.

の違いは5~7%であり、ほぼ一致している。また、このケース以外に、データ記録機構、データ記録制御機構の台数を半分にしたシステム構成で同様の比較を行い、同等の結果を得ている。デステージに関しては、とくに比較を行っていないが、更新のあったシリンダのみが転送されること、データ転送が逆方向に行われることを除きステージの場合と同様の処理手順であることから、同等の精度であると推定される。

次に、シミュレータの適用例として、MSSのファームウェア・ロジックの変更によるMSS性能の違いを評価した例を示す。図6は、ファームウェア処理方式比較のため、図5と同じ評価項目に関して行ったシミュレーションの結果である。ここで、方式1は大容量記憶制御機構がステージ、アンロードのコマンドを各ステージング・ディスク制御装置にシリアルかつ排他的に処理させ、コマンド作成時に使用するパス(ステージング・ディスク制御装置とデータ記録制御機構)を1度決定したら固定したままにしておく方式、方式3は大容量記憶制御機構がステージング・ディスク制御装置に、発行するコマンド・チェーンのうちデータ記録機構のサーチ動作を伴うコマンドを切り離して先に発行することにより、同一のステージング・ディスク制御装置にサーチ動作を複数個並行して行わ

せ、また、使用するパスの決定をパスが空く都度やりなおす方式である。方式2は、方式1と方式3の中間の処理方式である。

図6の例では、パス選択方式の違いによる効果を明確にするため、データ記録機構からDASDへのパスは完全デュアル構成となっている。図6より、シミュレーションの結果、処理方式によりその処理能力が、7~60%異なることが把握できる。

計算機使用時間は、図6の場合、前処理と後処理を含めて約4時間である。また、モデルはファームウェアを詳細に記述してあるため、約12kステップを要している。このうち、図6の方式検討時の変更部分は約1kステップである。ファームウェア・モデルが分離し、モジュール化しているため変更は局所的なものが大部分である。

5. むすび

MSS性能評価のためのシミュレーション・モデルを作成した。モデルは、ハードウェア、ファームウェア、ソフトウェアの各モデルから構成され、ハードウェア・モデル中にコマンド・チェーン処理機能を含んでいる。コマンド・チェーン処理モデルは、コマンド処理ごとにチャンネルと制御装置の結合、切離しを効率的に処理する。作成したシミュレーション・モデルをステージ連続処理の評価に適用し、ファームウェア・ロジックの違いにより、ステージ処理能力が大きく異なる場合のあることを示した。

本論文のモデルの特長は、ファームウェア・モデルが分離されているためファームウェア・ロジックの変更、評価が容易であること、コマンド・チェーンのパターンを自由に設定できること、ファームウェア、ソフトウェアの両モデルからのコマンド・チェーン制御が容易であることである。また、MSS特有のステージ動作のため、ステージング・ディスク制御装置の使用率が高くなった場合でも、その空き待ちを正しく評価できる点も特長となっている。

今後の課題として、モデルの一部に解析モデルを導入すること等によるシミュレータの高速化、およびシミュレータの適用による性能評価技法の確立がある。

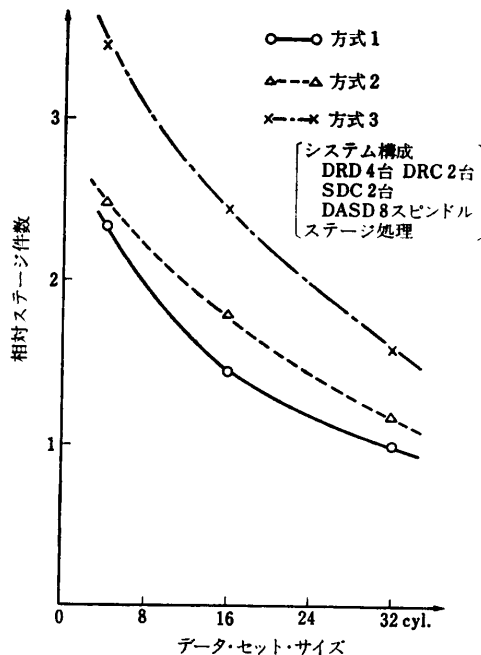


図6 ファームウェア処理方式によるステージ処理能力の比較

Fig. 6 Comparison with firmware logics of processing ability to stage data.

参考文献

- 1) 日立製作所: 大容量記憶システム (MSS) 概説・解説書, 8080-1-004 (1979).

- 2) Lavenberg, S. S. and Slutz, D. R.: Regenerative Simulation of a Queuing Model of an Automated Tape Library, *IBM J. Res. Dev.*, Vol. 19, No. 5, pp. 463-475 (1975).
- 3) Tsuruho, S. et al.: Mass Storage System Performance Analysis Using a Queuing Model, Proc. 3rd UJCC, pp. 320-324 (1978).
- 4) Hempy, H.: IBM 3850 Mass Storage System

Performance Evaluation Using Channel Monitor, in *Computer Performance*, pp. 177-196, North Holland (1977).

- 5) 日立製作所: 離散型シミュレーション・システム GPSS 入門編, 8080-7-001 (1979).

(昭和56年10月16日受付)

(昭和57年2月16日採録)