

# DRM(動的資源管理)の制御方式とその評価

入保秀士, 大野直哉, 大鷹正之, 土井根敏明  
(日本電気)

## はじめに

コンピュータシステムの大規模化が進み、一つのシステムを様々な利用目的、利用形態で多くのユーザが同時に使用している。このようなシステムを、システム管理者・エンドユーザのすべてが満足するように運用できるようにするには強力な資源管理機能が必要とされる。我々はこの機能が達成すべき目標を明確にし、それを実現する方式の研究を行って DRM(Dynamic Resources Manager)方式を確立した。<sup>1)~4)</sup>そして、統合次元システムを管理する中大型機用 OS である ACOS-4/MVP の中核をなすコンポーネントとして DRM3.5 を開発した。<sup>5)</sup>ここでは DRM における管理の狙い、それを実現するためにとっている制御方式の概要を述べ、これらの効果を実測により評価した結果について報告する。

## 2. DRM における管理目標

コンピュータシステムの資源管理には次の3つの目標があると考えられる(図1)。

- ・ドメイン(ジョブのグループ)に対する処理能力の適切な配分
- ・個々のトランザクションに対する応答性の保証
- ・システムスループットの向上

システムは多くの組織、業務、複数の処理次元により共用される。これらに属する処理要求はサービスをシステムに要求し、処理能力の配分を求めて競合する。この競合に解決を与えるために DRM は処理要求の主体の集合をドメインとして定義可能にし、システム設置目的に応じて規定された運用ポリシーに従って、実測した負荷状況を考慮しつつドメインに対するサービス量の配分を行う。<sup>2)</sup>

様々な緊急度のジョブがシステム内に混在するため統一的な応答性管理が必要とされる。DRM は最も直接的にエンドユーザの期待を反映する管理目標を採用した。トランザクションのグループ(応答クラス)に対して目標応答時間と達成率目標を指定できるようにしこれの達成を目指した管理を行う。<sup>3)</sup>

スループットの向上のためには設置されているハードウェア構成をできるだけフルに稼働させるような仕事の手え方を行うように制御する必要がある。<sup>1)</sup>

これらの3つの目標は互いに相反する面を含んでおり、目標間のバランスのとり方に関するポリシーもシステム管理者が与えられるようになっている。

## 3. 管理目標を実現するための制御方法

2. で述べた目標を達成するために、DRM は基本的には動的なフィードバック制御を行っている。図1に示すように制御は階層化されている。ユーザインタフェースとしてポリシーの設定層があり、これを受けてポリシーの論理的実行階層がシステムと各ジョブの状況に応じて、スワッピング、ディスパッチング、I/O プ

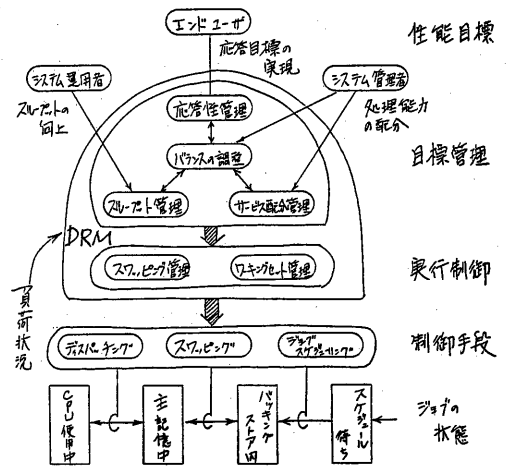


図1. DRM の概要

ライオリティ，ワーキングセット制御等の手段により，リソース割当てを制御する。

サービス配分管理はドメインに対するサービス率の配分を管理する。ドメインのサービス率は次の式で計算される。a, b, c は重み係数である。

$$\text{実サービス率}(i) = a \times \text{CPU使用率}(i) + b \times \text{I/O使用率}(i) + c \times \text{主記憶使用率}(i)$$

さらに，応答性を優先する制御に

よってロスとなるサービス率も含めた実効サービス率を定義する。システムのサービス容量は実効サービス率で計ると一定でありこれを各ドメインへ分配する。ドメインの処理要求量を表わす概念として潜在負荷量を導入し，この負荷量に応じた実効

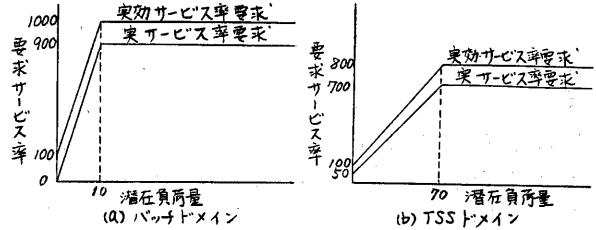


図2. 潜在負荷とサービス率要求の関係の例

サービス率要求を規定できる(図2)。全ドメインの要求量の和がシステム容量を超える場合は要求を削減して割り当てるが，その基準も与えうる(図3)。これにより一定時間間隔で配分目標の見直しが行われる。配分された実効サービス率の実サービス率と応答性向上への振分けも規定できる。図2で実効と実の比E(i)は応答性優先の度合を表わしており，応答負荷レベル目標に変換される(図4)。

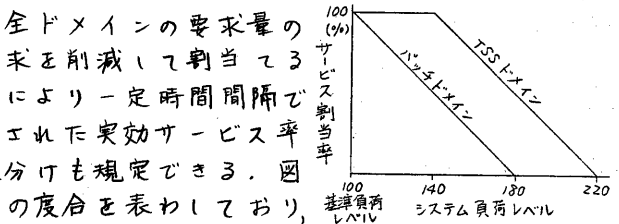


図3. サービス率指定の例

実サービス率の配分目標達成のためには，各ドメインの主記憶多重度，同時接続端末数をフィードバック制御する。応答負荷レベルの目標を達成するためには，ドメインのGP枠(後述)と最小主記憶量を制御する。

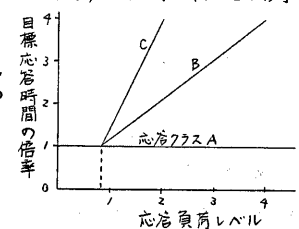


図4. 応答クラス間遅延の配分

応答時間管理はTSS，オンラインに関する応答性の保証を実現するために，トランザクションに与える応答優先度GPを導入している。GPが高いと主記憶，CPU，I/Oの割当てにおいて優先的に扱われる。GPに関する制御として目標応答時間が近づくにつれてGPを上げていくデッドライン制御を行う。また，応答クラス間に目標達成度のばらつきがある時，これをバランスさせるようなフィードバック制御も行う。応答負荷レベルの高い(E(i)の小さい)ドメインについては応答目標を遅らせるが，その度合いはクラス別に指定可能である(図4)。

スループット向上のためには，GPの低いジョブのみを対象として，CPUの割当てにおいてはタイタミックディスプレイパッチングを，主記憶の割当てにおいては主記憶上のジョブが各入出力装置及びCPUの使用率をできるだけバランスさせる組合せになるようにジョブミックス制御を行う。このため，ジョブごとに次の式でスワッピング対象として適合度の判定を行う。ジョブ番号をjとする。

$$F(j) = w_0 \beta_0 x_{0j} + \sum_{i=1}^n w_i \beta_i x_{ij} + w_2 y_j$$

$w_0, w_1, w_2$  は重み係数であり， $x_{ij}$  はリソースi (i=0はCPU，1...nはディスプレイ装置) をジョブjが使用した割合， $\beta_i$  はリソースiのシステム全体としての使用率を反映した値とする。y\_j はjによって専有される装置の数である。

以上の他に，全般的な性能向上のために，主記憶量の論理的な配分管理(ワーキングセット管理，ページプール管理)も行っている。(4), (5)

#### 4. 測定及び評価

ここでは DRM 運用の一例となるシステムにテスト用のバッチ及び TSS 負荷を与え、その負荷の処理結果からスループット・レスポンス特性を採取することによって DRM の評価を行っている。測定は性能向上の観点とポリシー達成の観点から行っており、最後に総合的に DRM の性能評価をしている。

##### (1) 測定環境

測定の対象となる機器構成を表 1 に示す。1 台の低速ディスク及び 3 台の高速ディスクにバックスタックが割り付けられて、残りの 6 台のディスクがテストジョブのアクセス対象となっている。

バッチ負荷としては、平均 7ms の周期でファイル入出力を繰り返すジョブを複数走らせている。一つのバッチジョブは同時に 3 つのファイルをそれぞれ 6:3:1 の割合でアクセスする。バッチジョブ全体で使うファイルは 5 つのディスクに分散されている(表 2)。ジョブスタックは J11, 12, 13, 21, 22, 23, ..., 61, 62, 63 の順に繰り返して行われている。

TSS 負荷はエディターでプログラムテキストを編集し、コンパイルを起動するといった標準的な端末オペレーションを同時に複数擬似できるシミュレータを測定システム上で動かして実現している。TSS 1 端末は思考時間 30 秒で Xact を発生し約 20 分周期でセッションの開設、閉設を繰り返す。1 セッション当りの TSS コマンドの内訳を表 3 に示す。

以降の評価項目では特にことわらないかぎりこの環境で測定している。

表 2. バッチジョブの使用するファイルとディスクとの対応

ディスク ジョブ	#1	#2	#3	#4	#5	I/O 間隔平均時間 (CPU 時間/I/O 数)	
						標準	2.5MHz ディスクジョブ
J11, 12, 13, 61, 62, 63	F1	F2	F3			7.0msec	6.1msec
J21, 22, 23		F1	F2	F3			4.5.8msec
J31, 32, 33			F1	F2	F3		1.6msec
J41, 42, 43	F3			F1	F2		16.1msec
J51, 52, 53	F2	F3			F1		16.1msec

注. ファイル利用率 F1:60% F2:30% F3:10%

##### (2) スループット評価

DRM で実現されているスループット向上策—ジョブミックス制御、ダイナミックディスクパッチング—の各々に関し単独での評価を行っている。スループットの尺度はバッチジョブから起動される単位時間当りの物理 I/O 完了数としている。

##### ① ジョブミックス制御の効果

ジョブミックス制御は、任意のディスクの使用率がある値を越えたとき起動され、3 秒述べた評価尺度によってスワッピングを行う。このためバッチ負荷を与えるには実行多重度の設定のほかスワップインの候補となるジョブを予めスケジューリングしておく必要がある(スケジューリング多重度)。

本測定では実行多重度 5、スケジューリング多重度 10 に設定し 36 個のバッチジョブが終了するまでを測定対象として評価を行った。ジョブスタックは(1)測定環境で示した通りファイル I/O が特定のディスクに集中する様な構成となっている。

測定対象としてはジョブミックス制御無しと有りの場合、有りでもディスク使

表 1. 機器構成

CPU	ACOS 450 (約 1MIPS)
主記憶	7M バイト
ディスク	高速ディスク x 4 (転送速度 1.2M バイト/秒)
	低速ディスク x 6 (転送速度 0.8M バイト/秒)

表 3. TSS コマンドの内訳とその特性

応答 TSS 名	応答性目標		発生頻度	実行時間 制御	スワップ 制御
	目標到達	達成目標			
TSS 特殊型	—	—	7.5%	×	×
TSS 会話型 1	3 秒	80%	67.5%	○	○
TSS 会話型 2	3 秒	80%	10.0%	○	○
TSS 会話型 3	5 秒	80%	7.5%	○	○
TSS 処理型	10 秒	—	7.5%	○	×

用率換出値の低い場合と高い場合を行って、結果を表4に示す。ジョブミックス制御により12%のスループット向上が得られているがスワッピングがみだりしておりTSSも存在する様なシステムではスワッピング頻度の調節が重要となる。

② デイスパッチング方式

ここでは比較のためにすべてのバッチジョブのデイスパッチングプライオリティを一定にした場合と、DRM=OFFのモード時に採用される方式(1)とDRM=ON

時の方式(2)の3通りについて測定を行った。方式(1)は実行中のジョブを平均I/O間隔の短い順に並べ均等に2つに分割しプライオリティ付けを行う方式で、方式(2)は平均I/O間隔とデイスパッチングレベルを固定的に対応づけて分割する方式で、ここでは2ms, 8ms, 32msと3つの閾値を定めを行った。

バッチジョブは表2に示す様にI/O間隔が様々なものを用い、またジョブスタッフもJ11, 21, ..., 61, 12, 22, ..., 62, 13, 23, ..., 63 という様に異なったI/O間隔のジョブが主記憶上に存在する様な構成をとっている。実行の重度は6でジョブミックス制御は行っていない。測定結果を表5に示す。表より方式(1)では4%, 方式(2)では13%のスループット向上が得られておりCPU, I/Oを効率的に使用していることがわかる。

(3) 応答性能評価

DRMで導入されているデッドライン制御、フィードバック制御は効果として希望の応答性能を達成することはもちろん応答時間の分散を小さくすることが期待される。

① 応答性能の達成

TSS 30 端末の環境下で応答クラス毎の応答目標時間を変化させてその達成状況を測定した。応答目標時間としては表6に示す3つの場合を考えた。case 1を標準と考え、case 2, case 3ではTSS会話型1及び2についてそれぞれ応答目標を-1秒, +1秒だけ変化させた場合と+1秒, -1秒変化させた場合について測定した。得られた応答クラスごとの累積達成率を図5に示す。それぞれ目標を達成する方向に応答分布が動いていることがわかる。TSS会話型3についてはcase 1~3共に同一目標があるが最もXact数の多いTSS会話型1の応答目標をきつくしたcase 2では達成率が悪く、逆に応答目標をゆるくしたcase 3では良い達成率が得られており各クラス間で応答目標を達成するためにシステムからの

表4. ジョブミックス制御の効果

	ジョブミックス無	ジョブミックス有	
		換出値高	換出値低
スループット	1	1.09	1.12
メモリ使用量	1	1.07	1.13
スワッピング回数	0回	29回	67回

表5. デイスパッチング方式の評価

	プライオリティ一定	方式(1)	方式(2)
		スループット	1
CPU利用率	69.4%	71.5%	78.9%

表6. 応答目標時間一覧

応答クラス名	case 1	case 2	case 3
TSS会話型1	3秒	2秒	4秒
TSS会話型2	4秒	5秒	3秒
TSS会話型3	5秒	5秒	5秒

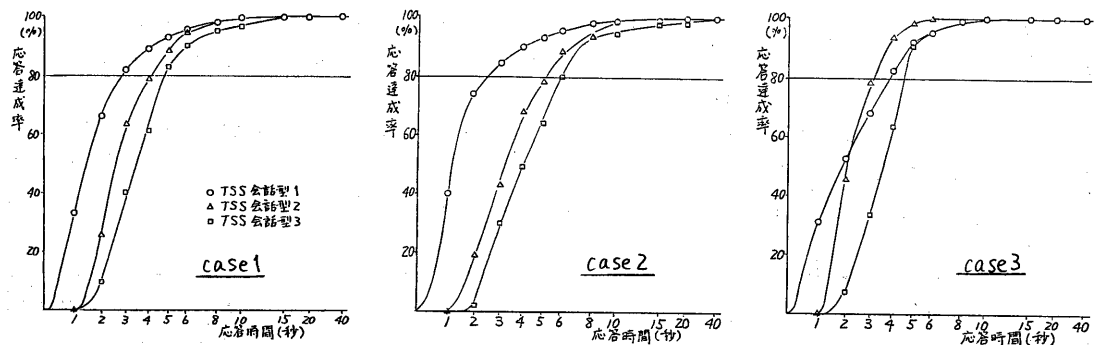


図5 累積応答達成率分布 (応答性能の達成)

サービスを分配しあっている様子が良くわかる。

② 応答分布

TSS20 端末 バッチ多重度6の環境でデッドライン制御の有無により Xact の達成分布がどの様になるかを測定した。ただし対象とする応答クラスの平均応答時間はほぼ等しくなる様にドメインのメモリ枠を調整している。測定結果を図6に示す。デッドライン制御有の時に応答目標時間の直前の分布が多くなっていることから応答時間の分散が小さい傾向にあることがわかる。

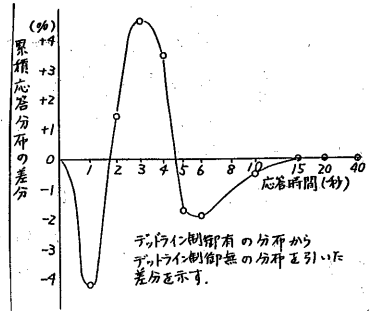
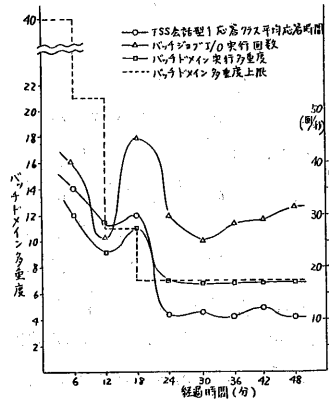
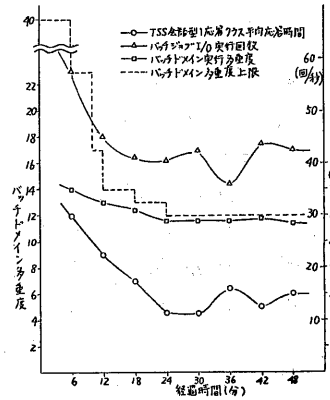


図6. 応答分布の差分(デッドライン制御の有無)

4) サービス配分管理機能

DRMにおけるサービス配分管理機能は負荷の変動に対してもシステム管理者が希望するポリシーの実現を可能にする。ここではこのことを確認するために複数の負荷条件を与えサービス配分機構の多重制御が働いて行く過程を測定した。

サービス配分パラメータは図2.3の通りである。負荷としてはバッチジョブはスケジュールが多重18とし、実行が多重はサービス配分制御により設定される。TSS 端末数としては15, 25 端末の2つを考えた。測定はすべての端末が開設している時にバッチジョブを一度



a) 15 端末

b) 25 端末

図7. サービス配分多重制御

起動しバッチジョブの多重度が安定するまで行った。結果を図7に示す。TSS 15 端末ではバッチ多重度12に、TSS 25 端末では多重度7に収束しレスポンスを達成しスループットを確保している様子がわかる。

(5) 達成率目標を変化させた場合

DRM 応答管理のフィードバック制御は実現された応答目標達成率を基に GP 値の上限下限 (GP 枠) を変動させる。ここでは TSS 25 端末バッチ実行多重度5の環境で応答達成率目標を変えることにより図8の様な結果を得た。目標を 90% から 60% に落として行くに従いスループットが向上している。これはフィードバック制御により GP 枠が低く抑えられシステム全体でのスワップアウトが減少したためと考えられる。レスポンスについては目標に違反して行くとする動きが見られるが目標 90% ではスワッピングの多さにレスポンスも逆に落ちてしまうことがわかる。

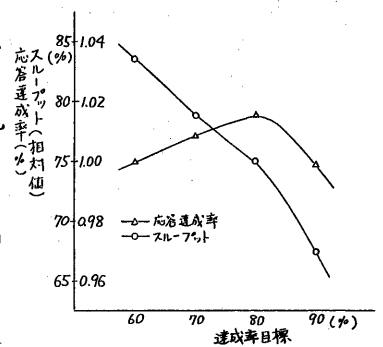


図8. スループット・レスポンス特性(達成率目標)

6) 総合評価

最後に DRM における応答時間管理、スループット管理各種の機能を動作させて

た場合 (DRM=ON) と、これを抑止した場合 (DRM=OFF) について複数の負荷条件でスループット・レスポンスの比較を行った。負荷条件として、TSS 端末数を 10, 20, 30 の3通りに変えて、バッチ実行多重度は 6, スケジュール多重度は 12 としている。

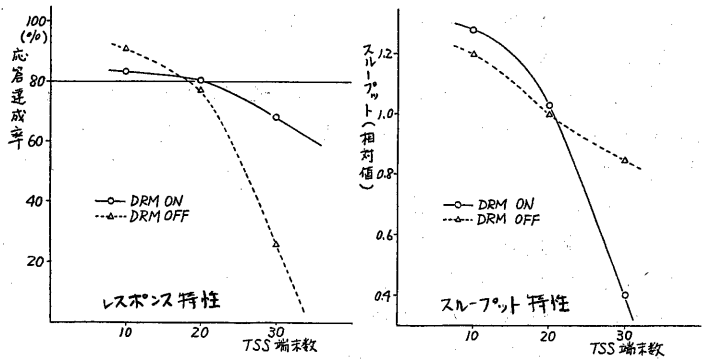


図9 スループット・レスポンス特性 (総合評価)

スループット・レスポンス特性を図9に示す。低負荷の

TSS 10 端末ではレスポンスを抑えスループットを上げているのに対し高負荷の TSS 30 端末ではスループットを抑えレスポンスを保証しており DRM の制御が働いていることがわかる。

### 5. おわりに

DRM は他の OS のこれに対応する機能に比較して次の特徴をもっている。

- ・ エンドユーザの応答性要求を直接的な形で受入れ、その実現を図っている
- ・ きめ細かいスループット向上方式の採用
- ・ システム管理者が直接的にサービス配分を管理できる。
- ・ 一組のシステムパラメータ値で、広い範囲の負荷に対応でき、設定が容易

これらの機能の概要を述べた後に、種々の観点から行った実験結果を評価した。測定結果は DRM の効果を示しており基本的に満足すべきものであった。ただしサービス配分管理の内部制御に不十分さが見られた。今後のシステム規模の大型化に対応するための機能向上と併わせ、さらに強化を行って行く予定である。

最後に、DRM の開発及び評価に関してお世話になった当社基本ソフトウェア開発本部 過部長、磯課長、片岡主任、畑氏に感謝の意を表したい。

### 参考文献

- (1) 久保, 工井根, 大野, 小野 「オペレーティング・システムの動的資源管理方式の評価」 情知学会 性能評価研 17-2, 53年3月.
- (2) 久保, 大野 「多次元システムにおけるサービス配分管理の検討」 情知 20回 全国大会 2J-6, 54年8月.
- (3) 工井根, 大野, 久保 「デッドライン制御をベースとした応答性管理方式とその評価」 情知学会 解析と制御研 6-3 54年9月
- (4) H. Kubo, N. Ohno, T. Doine 「Design and Evaluation of Dynamic Resources Manager (DRM)」 Proc. of IEEE COMPSAC, pp 610-617, 1983年11月
- (5) 日本電気 「ACOS-4/MVP 動的システム資源管理解説書」 DDE55-2, 66no-2 58年1月.