

航空宇宙技術研究所資料

TECHNICAL MEMORANDUM OF NATIONAL AEROSPACE LABORATORY

TM-543

航技研計算機システム用トランザクション・
プロセッシング・シミュレータ

土屋雅子・畑山茂樹

1985年4月

航空宇宙技術研究所
NATIONAL AEROSPACE LABORATORY

航技研計算機システム用トランザクション・ プロセッシング・シミュレータ*

土 屋 雅 子** 畑 山 茂 樹**

<主要略語一覧表>

CPU	:	中央処理装置
DUR	:	プログラムの応答性を変更させるためのCPU使用時間
FEシステム	:	フロントエンドシステム
FWKL	:	Fix ワークロードレベル
IOP	:	入出力処理装置に対する優先順位
ISV	:	CPU使用時間の最小値
JES/E	:	ジョブエントリサブシステム
MAINシステム	:	ローカルシステム
PDL/PDA	:	Performance Data Logger/Performance Data Analyzer
PFNG	:	パフォーマンスグループ番号
PGN	:	プライオリティグループ番号
PRTY	:	中央処理装置に対する優先順位
SDM	:	System Decision Manager
SMF	:	System Management Facility
TUNIT	:	タイムスライス
UIC	:	実記憶装置に対する優先順位
VAR	:	多重度の事前調整値

1. 緒 言

近年、電子計算機システムの利用は拡大の一途をたどり、規模の大型化、処理内容の高度化、ユーザーニーズの多様化等、変化の様相には激しいものがある。これに伴い、計算機システムの管理運用も複雑化し、システムの効率的運用を図ることが増々困難の度を深める傾向にある。したがって、ハードウェア、ソフトウェアおよびユーザーへの木目細かいサービス等に対する総合的に高度なシステムパフォーマンス目標を実現する運用法の追求は重要な課題となっている。

しかし、システムパフォーマンス評価の技法やツールは未だ確立されていないのが現状である。昭和58年2月に運用を開始した航技研の大型電子計算機 FACOM M-380 + M-180 II ADシステムも、規模、性能、機能ともに高度で、かつ、大変複雑なシステムである。本システムは、疎結合多重処理(Loosely Coupled Multi Processing)方式で運用しており¹⁾、M-380の処理能力が航空宇宙技術の研究開発に特徴的に現われる大規模計算に振り向けられ、かつ、M-180 II ADが各種ソフトウェアの開発に必須な TSS 処理機能を十分発揮するとともに、MT スプール、ユーザコンソール、図形出力モニター機能等²⁾のマン・マシン・インタフェースの強化に対しても十分応答性を保証するようシステム構築が行われた。

* 昭和60年2月25日受付

** 計算センター

このようなシステム設計における用途はかなりの程度達成されていることが実稼動状況の解析から実証された。しかし、何分にも複雑なシステムであるから、三大システム資源である中央処理装置（CPU という）、実記憶装置および入出力処理装置（I/O という）は適切に運用されているか（運用パラメータ設定法の有効性の可否）、疎結合方式における共用磁気ディスクに対する入出力処理競合はどの程度か、サポート可能な TSS 端末台数はどのくらいか、TSS の応答性とジョブクラスごとのバッチジョブのスループットはどのようにして改善すべきか等、システムパフォーマンス評価に関する疑問に答えるのは容易ではない。これらの疑問に定量的な解答を与え、システムの効率的運用に対する諸方策を検討するた

めには計算機システムの稼動状況を模擬するシミュレータを開発することが最良な方法と考えられる。本報告はこのような目的で開発したトランザクション・プロセッシング・シミュレータ（Transaction Processing Simulator, TPS と略称する）の詳細について記述するとともに、TPS を使用して行ったシミュレーション実験結果について述べ、かつ、新たな計算機システム運用法を提案し、その有用性をシミュレーション実験で確認する。

2. 航技研計算機システムの概要

2.1 ハードウェア構成の概要

航技研計算機システムのハードウェア構成の概念図を図 2.1 に示す。同図において、中央計算機は疎

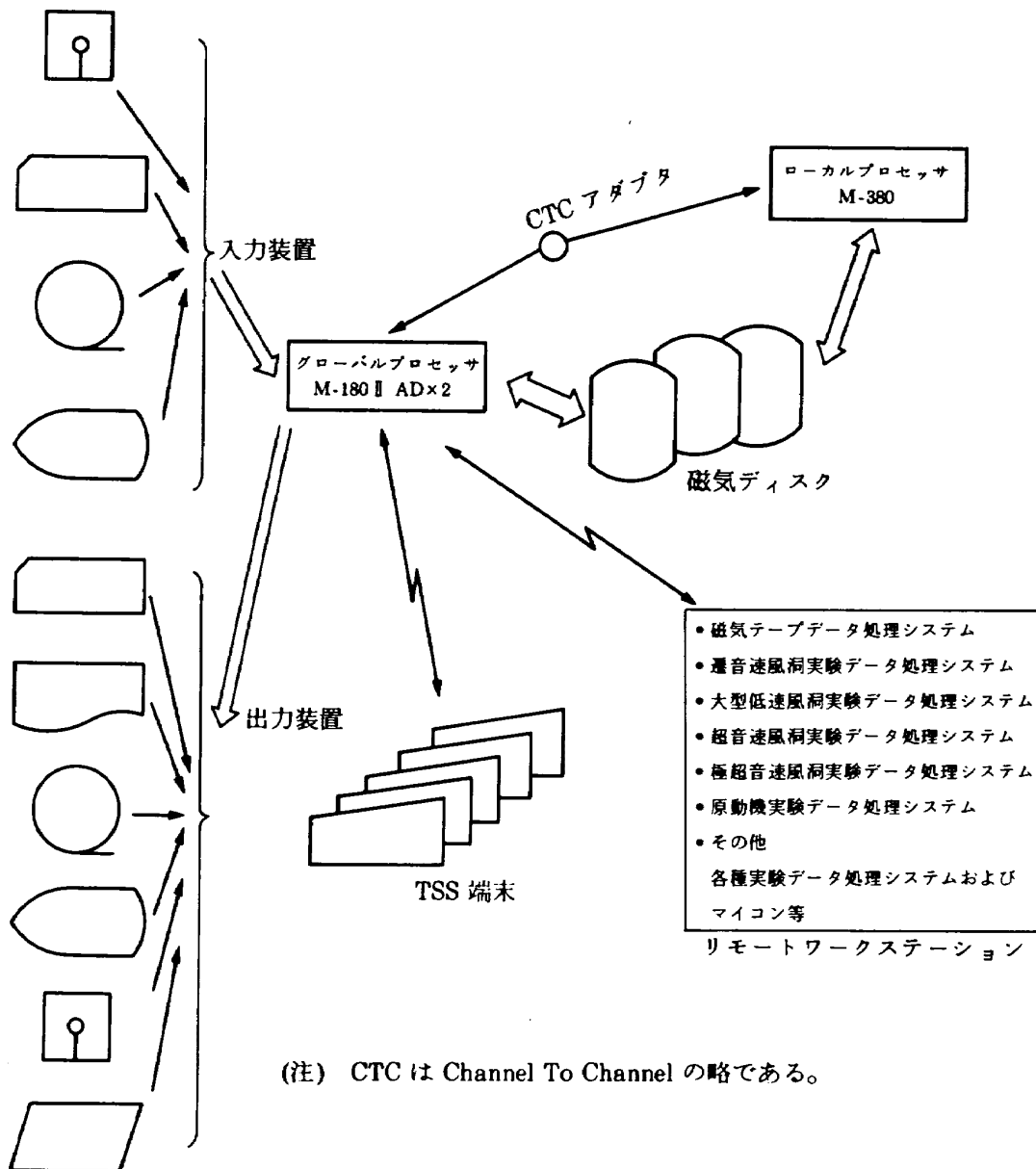


図 2.1 航技研計算機システムの概念図

結合方式複合システムとして運用し、M-180 ■AD × 2をグローバルプロセッサ (FEシステムという)、M-380をローカルプロセッサ (MAINシステムという)としている。磁気ディスク装置は両システムで共用し、各種入出力装置はFEシステム側に接続されている。FEシステムはジョブの入出力処理、TSS処理、小規模バッチ処理および各種のシステム運用上必要な処理を行う。一方、MAINシステムはバッチジョブ処理のみに専念する。また、各種のリモートワークステーションがFEシステムに接続されている。なお、搭載オペレーティングシステム (OSと略称する)はOSN/F4 (F4と略称する)である。

2.2 実稼動状況調査ツール

F4システムが提供している実稼動調査用ツールとして、アカウントティング・ログ (SMF: System Management Facility)³⁾とソフトウェアモニタおよびその分析ソフトウェア (PDL/PDA: Performance Data Logger/Performance Data Analyzer)⁴⁾がある。SMFはシステムの利用状況あるいはジョブに関する情報を収集記録するものである。表2.1はSMF情報の主な内容を示す。PDL/PDAはCPU、実記憶、チャンネル、I/Oデバイス等ハードウェアの利用実績データを収集し、分析してリスト等に編集するものである。図2.2はPDL/PDAの概要を示す。また、表2.2はPDL/PDAが収集・編集

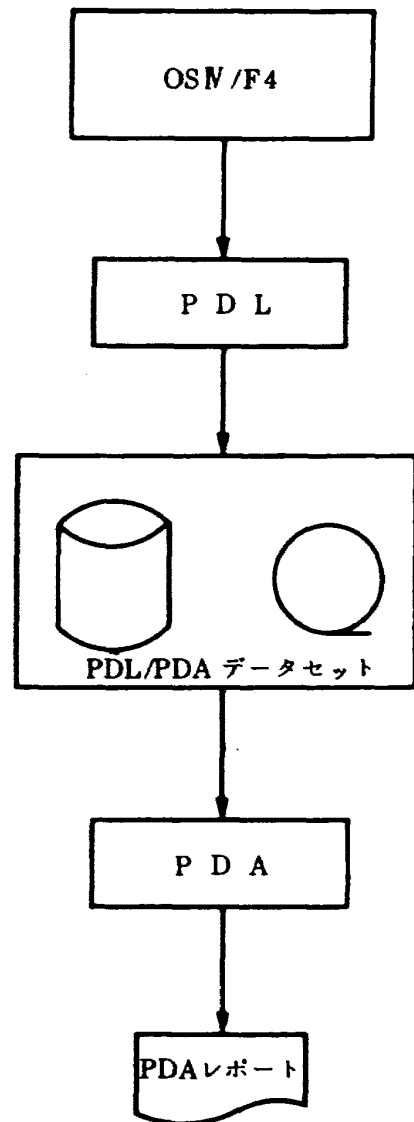


図2.2 PDL/PDAの概要

表2.1 SMFデータレコードの主な内容

レコードタイプ	データ収集時期	内 容
0	IPL時	システム利用情報
4	ジョブステップ終了	ジョブステップ情報
5	ジョブ終了	ジョブ情報
6	ジョブ終了	出力データセット情報
14	データセット入力終了	入力データセット情報
15	データセット出力終了	出力データセット情報
19	磁気ディスククローズ	磁気ディスクボリューム使用情報
21	磁気テープクローズ	磁気テープボリューム使用情報
25	ジョブ資源割当	ジョブの資源割当情報
32	コマンド終了	コマンド情報
34	セッション終了	セッション情報
43	JES/E起動	サブシステム情報

表 2.2 PDI/PDA収集リスト

(1) CPU稼働状況

B1. CPU UTILIZATION																
CPU MODEL		0180		0180												
NUMBER OF CPU'S		2														
PERCENT OF TOTAL TIME						0	10	20	30	40	50	60	70	80	90	100
SYSTEM ACTIVE TIME		100.00%		----- ----- ----- ----- ----- ----- ----- ----- ----- ----- (**=2%)												
SYSTEM IDLE TIME		0.00%		----- ----- ----- ----- ----- ----- ----- ----- ----- -----												
CPU NO.	MASTER SCHEDULER	SYSTEM SPACES	USER SPACES	TOTAL BUSY	WAIT											
0	10	20	30	40	50	60	70	80	90	100						
0180	0.79%	11.86%	60.34%	72.99%	27.01%	----- ----- ----- ----- ----- ----- ----- ----- ----- ----- (**=2%)										
0181	0.50%	10.88%	57.91%	69.29%	30.71%	----- ----- ----- ----- ----- ----- ----- ----- ----- -----										
ANY CPU BUSY						89.26%		----- ----- ----- ----- ----- ----- ----- ----- ----- ----- (**=2%)								
ALL CPU BUSY						53.05%		----- ----- ----- ----- ----- ----- ----- ----- ----- -----								
--PERCENT OF DISABLED--											----- DISABLED TIME -----					
		MAXIMUM		MINIMUM		AVERAGE										
0.61%		1702.809MS		0.252MS		6.191MS										
----- CPU BUSY -----																
----- SUPERVISOR MODE -----						----- PROBLEM PROGRAM MODE -----										
SYSTEM KEY		USER KEY		SYSTEM KEY		USER KEY										
33.38 %		0.32 %		2.15 %		64.15 %										

(2) 実記憶利用状況

H1. REAL STORAGE UTILIZATION						
A1. REAL MEMORY SIZE		3,072 PAGES				
B1. AREA ATTRIBUTE	V/R RATIO	MAXIMUM PAGES (PERCENT)	MINIMUM PAGES (PERCENT)	AVERAGE PAGES (PERCENT)	STANDARD DEVIATION	
B10. NUCLEUS (FIXED)	***	145 (4.72%)	145 (4.72%)	145.0 (4.72%)	0.00	
B11. REGION	39.25 %	1,755 (57.12%)	194 (6.31%)	1,298.2 (42.26%)	187.65	
B12. PLS0A	12.85 %	537 (17.48%)	36 (1.17%)	381.1 (12.40%)	71.04	
B13. LSOA (FIXED)	***	247 (8.04%)	39 (1.26%)	193.7 (6.30%)	30.81	
B14. CSA	52.32 %	250 (9.44%)	104 (3.38%)	153.9 (5.01%)	13.10	
B15. PLPA	39.87 %	834 (27.14%)	346 (11.26%)	619.2 (20.15%)	113.72	
B16. SWA	***	151 (4.26%)	116 (3.77%)	124.7 (4.05%)	2.44	
B17. AVAILABLE	***	829 (26.98%)	2 (0.06%)	152.3 (4.95%)	155.96	
C1. FIXED ATTRIBUTE		PAGES (PERCENT)				
C10. ACTIVE = FIXED + PAGEABLE		2,449				
C11. FIXED		97 (3.96%)				
C12. PAGEABLE		2,352 (96.03%)				
C13. DETAIL		FIXED	PAGEABLE			
PLPA		40 PAGES	579 PAGES			
CSA		41 PAGES	111 PAGES			
PLS0A/REGION		16 PAGES	1,662 PAGES			

表 2.2 PDL/PDA収集リスト

(3) ページング状況

12. PAGING CHARACTERISTICS		
A1. TOTAL NUMBER OF PAGES PAGED IN/OUT	99,285 PAGES	12.56 PAGES/SECOND
A2. TOTAL NUMBER OF PAGES SWAPPED IN/OUT	41,198 PAGES	5.21 PAGES/SECOND
A3. TOTAL NUMBER OF SWAPS	595 TIMES	0.07 TIMES/SECOND
B1. NUMBER OF PAGES PAGED IN BY NON VIO	73,707 PAGES	9.32 PAGES/SECOND
B2. NUMBER OF PAGES PAGED OUT BY NON VIO	25,578 PAGES	3.23 PAGES/SECOND
B3. NUMBER OF PAGES RECLAIMED BY NON VIO	2,832 PAGES	0.35 PAGES/SECOND
C1. NUMBER OF PAGES PAGED IN BY VIO	0 PAGES	0.00 PAGES/SECOND
C2. NUMBER OF PAGES PAGED OUT BY VIO	0 PAGES	0.00 PAGES/SECOND
C3. NUMBER OF PAGES RECLAIMED BY VIO	0 PAGES	0.00 PAGES/SECOND
D1. NUMBER OF PAGES SWAPPED IN	22,304 PAGES	2.82 PAGES/SECOND
D2. NUMBER OF PAGES SWAPPED OUT	18,894 PAGES	2.39 PAGES/SECOND
D3. NUMBER OF PAGES RECLAIMED WHEN SWAP IN	6,094 PAGES	0.77 PAGES/SECOND
D4. NUMBER OF FREED PAGES NOT SWAPPED WHEN SWAP OUT	13,828 PAGES	1.74 PAGES/SECOND
E1. NUMBER OF PAGES PAGED IN (CSA,PLPA)	36,445 PAGES	4.61 PAGES/SECOND
E2. NUMBER OF PAGES PAGED OUT (CSA,PLPA)	1,141 PAGES	0.14 PAGES/SECOND
E3. NUMBER OF PAGES PAGED IN (PLPA)	32,847 PAGES	4.15 PAGES/SECOND
E4. NUMBER OF PAGES RECLAIMED (PLPA)	949 PAGES	0.12 PAGES/SECOND

(4) トランザクション処理状況

V1. WORKLOAD ACTIVITY (CONTINUES)													
----- SYSTEM TUNING PARAMETER -----				----- SERVICE DEFINITION COEFFICIENTS -----				- INTERVAL -		- IPS NAME -			
IUC=100 CPU=100 ERV= 500 CRV= 500				IUC= 0.0 CPU= 1.0 MSU= 0.0 SRB= 0.0				2H 5M 7S		KAAIP500			
GROUP NUMBER	PGP	DNJ	DMN	SERVICE	ABSORB RATE	SERVICE RATE	WORKLOAD LEVEL	NUMBER OF TRANSACTIONS	LIFE TIME (SEC)	OUT TIME (SEC)	SWAP COUNT		
								ACTIVE RESIDENT ENDED					
15	1	1	15	IUC= 0	14	13	191.92	1.97	1.89	10,105	2.7	0.0	11,802
				CPU= 195,789									
				MSU= 0									
				SRB= 0									
				TOT= 195,789									
				PER SEC= 26									
2	1	14		IUC= 0	21	21	154.08	0.20	0.20	438	11.4	0.0	696
				CPU= 31,763									
				MSU= 0									
				SRB= 0									
				TOT= 31,763									
				PER SEC= 4									
3	1	13		IUC= 0	17	17	88.65	0.20	0.20	61	35.3	0.0	783
				CPU= 24,936									
				MSU= 0									
				SRB= 0									
				TOT= 24,936									
				PER SEC= 3									
4	1	12		IUC= 0	7	7	191.97	1.37	1.37	33	160.5	0.0	367
				CPU= 69,825									
				MSU= 0									
				SRB= 0									
				TOT= 69,825									
				PER SEC= 9									
16	1	1	30	IUC= 0	8	8	82.89	0.19	0.19	124	13.5	0.0	130
				CPU= 11,016									
				MSU= 0									
				SRB= 0									
				TOT= 11,016									
				PER SEC= 1									
17	1	1	40	IUC= 0	0	0	0.00	0.00	0.00	0	0.0	0.0	0
				CPU= 0									
				MSU= 0									
				SRB= 0									
				TOT= 0									
				PER SEC= 0									

したデータリストの一例である。SMFとPDL/PDAからシステム稼動状況をかなり詳細に知ることができる。

2.3 ジョブ処理の概要

計算機システムに投入されたジョブの入出力処理はJES/Eと呼ばれるジョブエントリサブシステムにより行われ、ジョブの実行はJES/Eにより起動された各プロセッサのイニシエータで行われる(図2.3参照)^{1),5)}

まず、カードリーダ等の入力装置から読み込まれたジョブはJES/Eサブタスクにより、SYSINデータと制御文に分けられ、制御文は即時に解釈されて入カスプールに格納される。ひき続いて制御文をもとにジョブに必要なボリューム、データセット等の資源セットアップが行われ、実行待ちジョブを管理するキュー(セレクトキューという)に登録される。セレクトキューはジョブクラスごとにあり、ジョブは先着順(FIFOという)に管理される。セレクトキューの先頭ジョブはイニシエータにより処理が実行される。イニシエータはジョブクラスごとに起動多重度分だけ存在する。ジョブクラスに応じてどのプロセッサのイニシエータかが定まる。イニシエータは起動したジョブの制御文を仮想空間に格納し、この情報にもとづいて、実記憶へのローディングおよび資源の割当等スケジューリングを行い、ジョブステップの実行を開始する。ジョブステップが実行終了するとイニシエータはそのジョブステップが使用していた実記憶、入出力装置、データセット等を解放し、先行ジョブステップの終了時の状態に応じて次のジョブステップの実行を開始する。このように先頭ジョブステップより順次実行し、ジョブ内の最後のジョブステップが終了したとき、ジョブの実行は終了する。終了したジョブは再びJES/Eサブタスクにより出力待ちキューにつながれ、出力結果の取り出しを待ち、ジョブ出力が行われた後に消滅する。なお、JES/Eサブタスクの処理はFEシステムで行われる。

2.4 TSS処理の概要

TSS処理はユーザが通信回線を介し、遠隔地の

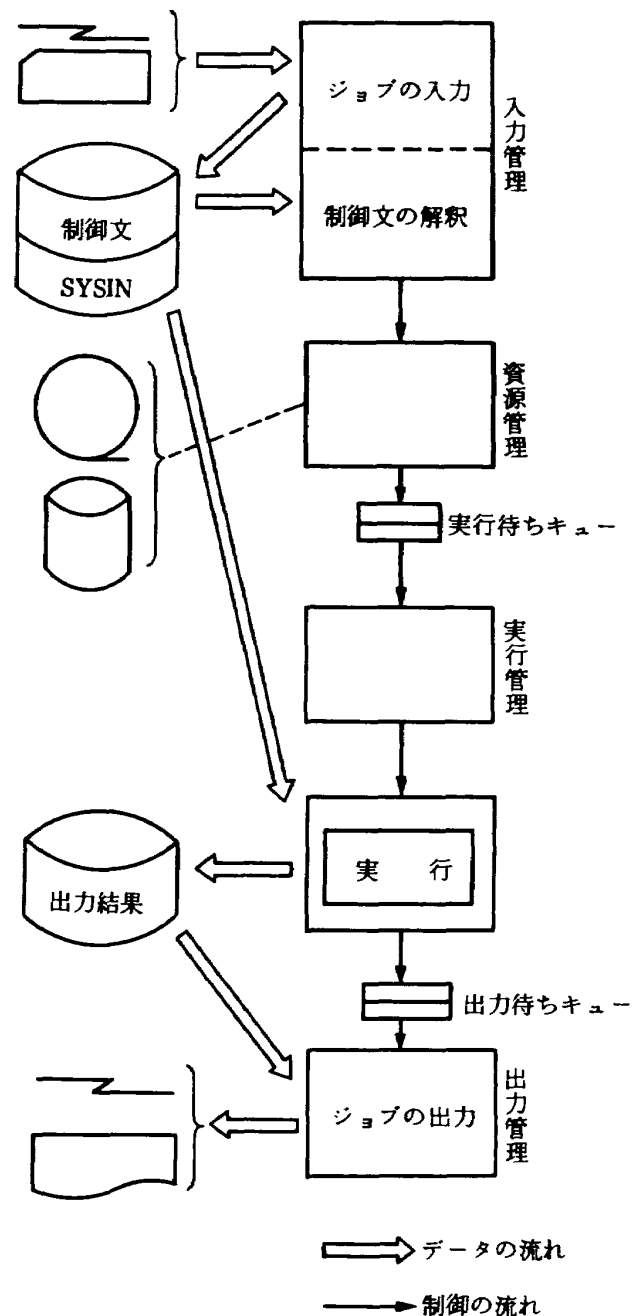


図 2.3 ジョブ処理の概要

TSS端末を利用して計算機システムと会話形式で処理を行う形態である。⁶⁾ TSS処理ではLOGONコマンドを投入して処理開始を宣言し、LOGOFFコマンドを入力して処理終了をシステムに告げる。ユーザがLOGONコマンドを入力してからLOGOFFコマンドを入力するまでをセッションという。セッションごとにTSSユーザ仮想空間(次節参照)が開設され、運用中には、セッション中のユーザ数(セッション数)に等しい仮想空間が開設されている。大多数のTSSコマンドはTSSユーザ仮想空間で処理が行われる。LOGONコマンドが入力される

と、システムはユーザ登録名やパスワード等の資格チェックや各種資源の割当等のセッション開設処理を行う。この処理が終了すると、端末には“READY”メッセージが出力され、ユーザにコマンド入力を促す。ユーザがコマンドを入力するとコマンド開設処理（OS処理）により、ユーザ空間は実行可能状態となり、該当するコマンド処理プログラムが主記憶上にローディングされる。この後、コマンド処理はトランザクション（第2.6節参照）としてバッチジョブのジョブステップと同様に、CPU、実ページ、I/O等各種資源を利用して動作する。トランザクションが終了するとコマンド開設処理（OS処理）により、各種資源の解放、およびユーザ空間を待ち状態にする処理が行われ、再び“READY”のメッセージを出力する。なお、コマンドによっては更にくつかのサブコマンドを有するものがあり、サブコマンド処理もコマンド処理と同様に行われるが、処理終了後は“READY”メッセージの代わりにサブコマンドの入力を促すサブコマンドメッセージが出力される。“READY”メッセージ出力からコマンド入力まで、またはサブコマンドモードメッセージからサブコマンド投入まではユーザの思考時間（端末思考時間という）である。コマンドおよびサブコマンド処理とユーザ思考を繰返して、ユーザはプログラムの作成、デバッグ、実行、またはデータセットの作成、消去、更新およびバッチ処理依頼等のTSS処理を行う。コマンド、サブコマンド処理に要する時間（TSSの応答時間という）はユーザの思考の妨げにならないような応答性が要求される。

2.5 仮想記憶方式の概要

F4システムの記録管理はジョブごとに独立した16メガバイトの仮想空間を割当る多重仮想記憶方式で行われる。図2.4に仮想空間の領域構成を示す。図中、共通領域は制御プログラムの一部や制御プログラムが使用するシステム内で共通な制御表などの領域が含まれている。システム領域は、主記憶上に常駐していなければならない制御プログラムやデータ等が入っている。システム内に存在するすべての仮想空間は独立であるが、システム領域と共通領域に対する内容、大きさは同一であり、これを共有す

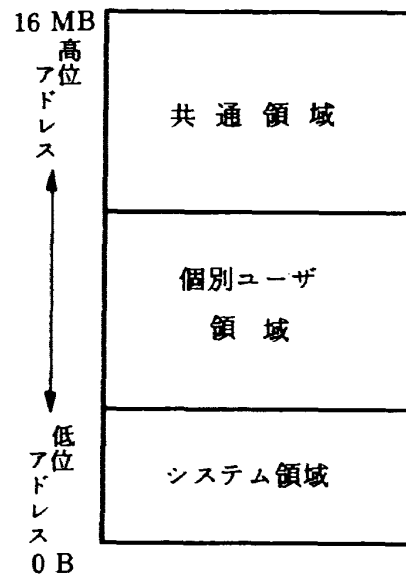


図 2.4 仮想空間の領域構成

る。個別ユーザ領域は各仮想空間ごとに固有な内容をもった領域として存在し、記憶装置管理上の制御表やユーザプログラムがローディングされ、実行される領域である。領域は4キロバイトの大きさのページ単位で管理され、また、仮想空間上で連続した16ページの領域を1セグメントと呼び、1仮想空間は連続した256セグメントの領域で構成される。実記憶上には、プログラムが使用する領域はページ単位に分散して割当られる。このため、仮想記憶上の仮想アドレスと実記憶上の実アドレスとの対応づけのために、セグメント・テーブルとページ・テーブルを用い、ハードウェアが動的にアドレス変換を行っている。仮想記憶方式では、すべての領域が実記憶上にとられている必要がなく、もし実記憶上にない仮想ページが実行されるとページフォルトという事象が発生し、その仮想ページは実記憶上にローディング（ページイン処理という）されるので、プログラムは再び実行可能となる。実ページの管理はアベイラブルキューとメモリキューによって行われる。前者は使用可能実ページ（これをアベイラブルページという）を、後者は実記憶割付要求を管理している。割付要求が発生するとメモリキューに登録され、実ページの割付を待つ。実ページの割付は通常、アベイラブルキューの先頭から行われるが、要求ページがアベイラブルキュー上に存在する場合にはその実ページを割り付ける（これを実ページのリクレームという）。アベイラブルページは実ページ

割当要求に対して即時に応えられるように、一定値以上が常時貯えられている（これをページプールという）。このプールが不足すると、その時点で割当済の実ページの中から適当な実ページが取り上げられ（これをページスティールという）、それを外部ページに移す処理（これをページアウト処理という）が行われる。この使用可能な実ページを作り出す処理をページリプレースメント処理という。このときに、どの実ページを取り上げるかというページ選択アルゴリズムは、LRU(Least Recently Used)方式で行われる。これは最も過去の時点でしか使用されたことがない実ページから順番に取り上げる方式である。取り上げ実ページが決定されると、通常、ページアウト処理が起動されるが、取り上げる実ページの内容と同じ内容が既に外部ページデータセットに存在していれば、ページアウト処理は省略される。また、リクレイムされたページに対してはページイン処理が省略される。

2.6 SDMの概要

SDM(System Decision Manager)はシステム資源へのアクセス権制御とプログラムの応答性制御を司るシステムプログラムであり、各種の運用パラメータを有し、センタ独自のシステム運用の意図を直接的に反映させる上で非常に重要なものである⁵⁾。SDMでは、その制御の対象となるシステム内で動作中のジョブステップ、TSSコマンド等をトランザクションと呼称している（図2.5参照）。アクセ

ス権制御は中央処理装置に対する優先順位（PRTYという）、入出力処理装置に対する優先順位（IOPという）および実記憶装置に対する優先順位（UICという）の値にもとづいて行われる。一方、プログラム応答性制御は実行中トランザクションの多重度制御と、CPU連続使用可能時間の最大値（TUNIT値という）、I/Oバウンド（I/O処理が多い）なトランザクションのCPU処理を優先させるか否かの属性（ROUND属性といい、可をラウンド、否をノンラウンドという）、多重度制御を受けずに実行が保証されるCPU使用時間の最小値（ISV値という）、多重度の事前調整値（VAR値という）および、プログラムの応答性を変更させるためのCPU使用時間（DUR値という）にもとづいて行われる。ここで、事前調整とは、急激なトランザクション数の増大に対処するため、前もって実行を可能とする多重度を増やしておく処理をいう。多重度制御は起動されているトランザクションを実行可能状態とするか、実行不可能状態にするか（スワップイン、スワップアウト状態という）により行われる。スワップイン/アウトには論理スワップと物理スワップとがあり、前者にはページイン/アウト処理が伴わず、後者にはページイン/アウト処理が伴う。多重度には実際に実行可能状態にあるトランザクション数を表わす実行多重度と、SDMが判断して一定時間間隔ごとに定める目標多重度とがある。目標多重度の値は運用パラメータとして定めた上限多重度と下限多重度の範囲内で決定される。SDMでは T_1 秒間隔

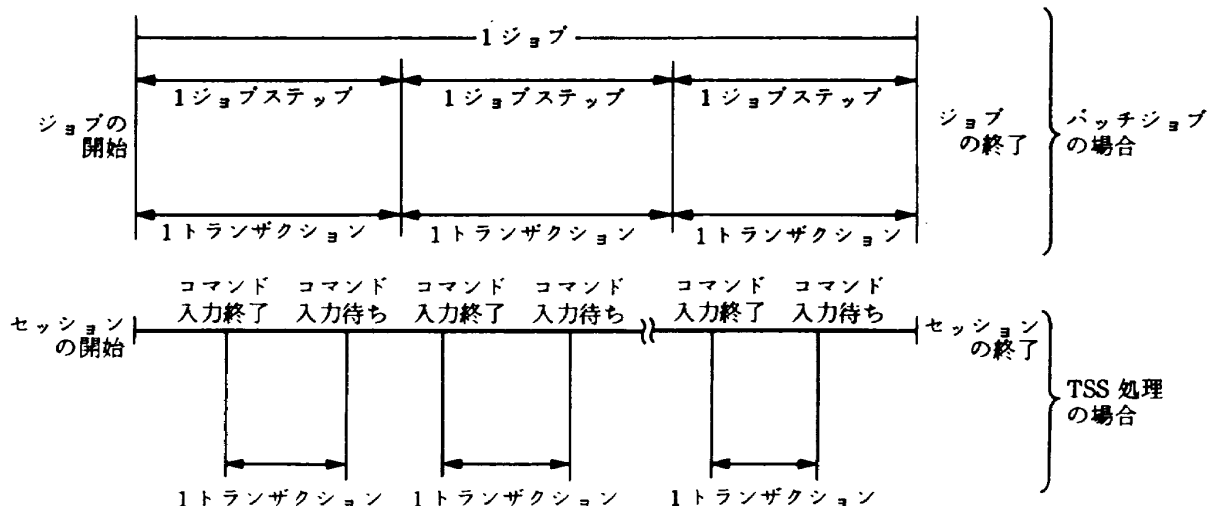


図 2.5 トランザクション動作の概念図

でその間におけるシステムの負荷状態（過負荷状態・安定状態・疎負荷状態がある）にもとずき目標多重度の調整を行っている。また、 $T_2 (T_2 \ll T_1)$ 秒間隔で実行多重度と目標多重度の値を一致させるための多重度調整を行っている。多重度調整周期において、スワップインまたはスワップアウトするトランザクションはFIFOで決定されるか、または、サービス率制御で決定される。サービス率制御は次のようにして行われる。サービス率はトランザクションごとに一定時間間隔でSDMが算出しているシステム資源使用の割合であるが、その値を直接使用せず、トランザクションごとにシステムで定めたパフォーマンス目標、およびサービス率をそれに対応したワークロードレベル値に変換する。例えば図 2.6 の場合、パフォーマンス目標 1 のトランザクション A の受けたサービス率 50、パフォーマンス目標 2 のトランザクション B の受けたサービス率 80、パフォーマンス目標 3 のトランザクション C の受けたサービス率を 90 としたとき、ワークロードレベル値はそれぞれ、30、60、70 となる。このようにして求められたワークロードレベル値の高いもの程スワップインし易く、低いもの程スワップアウトし易く制御を行う。なお、図 2.6 に示したパフォーマンス目標 3 のようにサービス率によらず一定のワークロードレベル値を指定することを特にFWKL属性と

いう。SDMでは各トランザクションをドメインというグループに分けて制御する。各ドメインには下限・上限多重度、FWKL値、およびVAR値を定義することができる。また、パフォーマンスグループが定義でき（PFGNという）、PFGNごとにパフォーマンス目標、ドメイン番号、PRTY、ISV値、UIC値、IOP値、DUR値を指定することができる。ここで複数のDUR値を定義した場合には、特に、パフォーマンスピリオドが定義されたという。さらに、プライオリティグループが定義でき（PGNという）、PGNごとにPRTY値の属する範囲、TUNIT値、およびROUND属性を指定することができる。したがって、多重度調整周期ごとにトランザクションの受けたサービス率にもとづいたスワッピング制御が行われ、目標多重度調整周期ごとにドメインの受けたサービス率にもとづいて、ドメインごとの目標多重度が調整される。目標多重度調整周期においては、平均アベイラブルページ数がある閾値と比較し、その値より少ければ実ページを確保したまま論理スワップアウト状態のトランザクションを物理スワップアウトし、アベイラブルページを増やす処理を行って、実ページが枯渇することを防止している。なお、SDMが実行可能状態と判定したトランザクションやOSタスクも含めて、最もCPU実行優先順位の高いものを見つけて、それにCPU

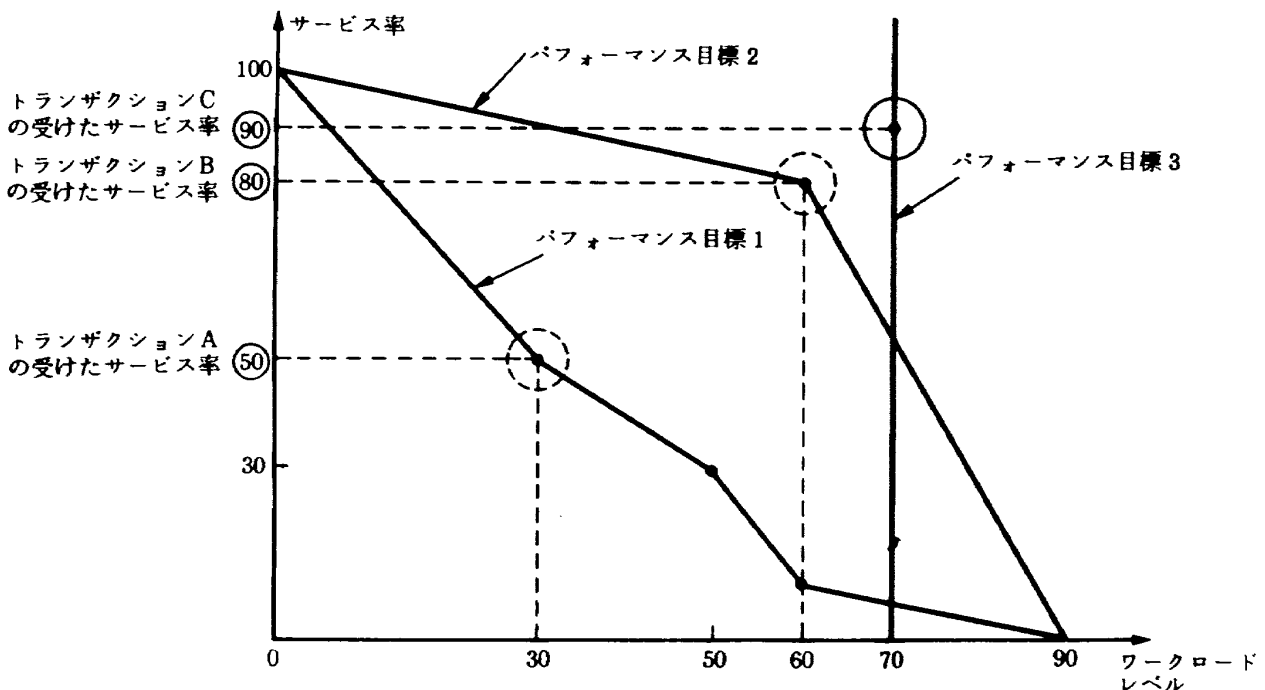


図 2.6 パフォーマンス目標曲線

使用権を与える制御を行うスーパーバイザルーチンをディスパッチャと呼び、ディスパッチャの行う機能をディスパッチングという。ディスパッチャはその管理をディスパッチキューを用いて行っている。

注) 同図のとおり、I/O処理装置はチャンネル処理装置(CHPまたはCHCという)、物理チャンネル装置(BMCという)、磁気ディスク制御装置(DIRという)、および磁気ディスク装置で構成されている。

2.7 I/O処理の概要

図 2.7 に I/O 処理装置のハードウェア構成を示

注) 磁気ディスク以外の入出力装置(カードリーダー、ラインプリンタ等)は省略している。

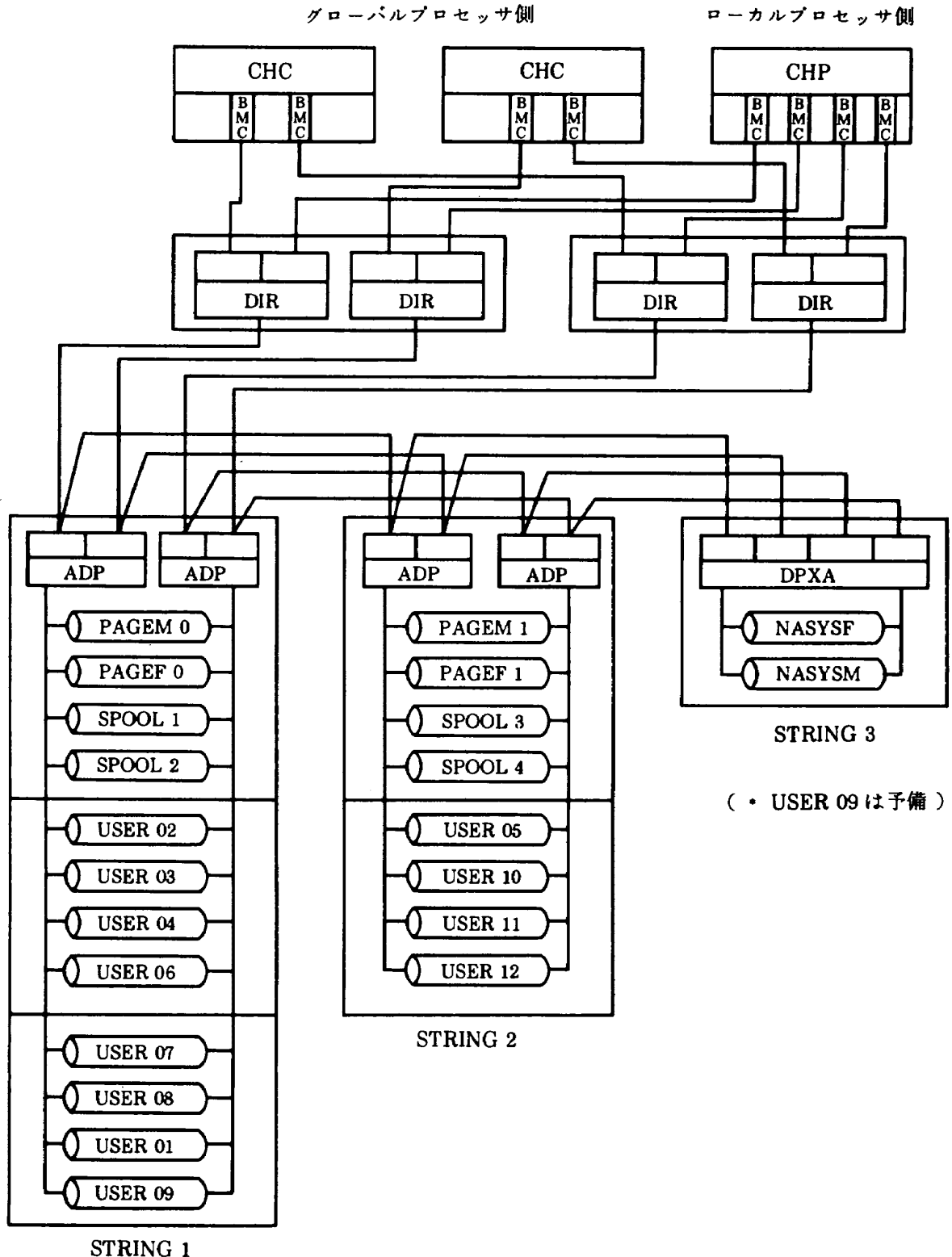


図 2.7 I/O処理装置のハードウェア構成

磁気ディスク装置はアダプタ部 (ADPまたはDPXA という) が内蔵されているものを先頭にして複数個接続でき、一つのストリングを構成している。アダプタ部にはストリング切換機構とクロスコール機構を取り付け、どのDIRからも、どのストリングの磁気ディスクにもアクセスできるようにしていると同時に、1ストリング内で同時に2台の磁気ディスクが読み書き動作できるようにしている。各UCBには同図に示したとおりのボリューム名を付けている。I/O処理を司るOSルーチンには、I/Oスーパーバイザとページングスーパーバイザがあり⁵⁾、前者は通常のI/O処理を制御している。後者はページフォルト、スワッピング、ページリプレースメント処理に伴うページイン/アウト処理を制御している。特に、ページイン/アウト処理においては、一定時間 (バースト時間という) を設定し、その時間内に可能で、かつ、最適なページング処理のスケジューリングを行い、その処理の高速化を図っている。これをバースト処理というが、この処理において、ページインの処理はページアウトの処理に優先される。I/O処理の管理は論理チャネルというキューを用いて行われる。図2.7から分るように、航技研システムの場合、M-380側からも、M-180 IAD側からも任意のBMCを介して、任意の磁気ディスクにアクセスすることが可能である。なお、I/O処理を開始した後、BMCと磁気ディスクを結ぶパスがビジーであることがある。これをパスビジーというが、この事象が発生した場合は、ビジーが解除するまでI/O処理は中断する。

3. シミュレータの概要

3.1 ジョブ処理のモデル化

T(1)時間後に発生が約束されているジョブの処理過程は次のようにモデル化する。シミュレーション時間が進行して、 $T(1)=0$ となると、事象EV(1)が起り、発生ジョブの資源要求量等のジョブ諸属性が決定されると同時に、JES/Eサブタスク (RDR, ISDVR, CI等を指す)¹⁾が発生し、このジョブをセレクトキューに登録する処理が開始される。この処理に要するCPU使用時間をT(2)で表す。シミュレーション時間が進行して、 $T(2)=0$ となると、事

象EV(2)が発生し、そのジョブのジョブクラスのセレクトキューにFIFOで登録される。そのジョブの処理はキューの先頭になるまで待たされる。各ジョブクラスのジョブ起動用プログラム (イニシエータ) が空くと、ジョブ開設処理および第1ジョブステップ開設処理 (OSタスク) が発生する。この処理の中で、第1ジョブステップが要求する仮想記憶に対して実記憶が割り当てられる (第4.1節参照)。この処理に要するCPU使用時間をT(3)で表す。シミュレーション時間が進行して、 $T(3)=0$ となると、事象EV(3)が発生し、CPU使用時間がT(4)で表わされるジョブの第1ジョブステップタスクが発生し、ディスパッチキューに登録され (第4.3節参照)、CPU使用権を待つ。このトランザクションはCPUサービスおよびI/Oサービスを繰返し、シミュレーション時間が進行して、 $T(4)=0$ となったときに終了となる。と同時に、ジョブステップ開設処理と後続ジョブステップの開設処理 (OSタスク) が発生する。この処理に要するCPU使用時間をT(5)で表す。もし、後続ジョブステップが無い場合、ジョブステップ開設処理およびジョブ開設処理 (OSタスク) が発生する。この処理に要するCPU使用時間をT(6)で表す。シミュレーション時間が進行して、 $T(5)=0$ となったときには、第1ジョブステップと同じくシミュレーションは進行する。 $T(6)=0$ となったとき、JES/Eサブタスク (OUTSERV, WTR, PURGE等を指す)¹⁾が発生する。この処理に要するCPU使用時間をT(7)で表す。シミュレーション時間が進行し、 $T(7)=0$ となったとき、ジョブは消滅する。なお、実際にはOUTSERVタスク終了後、ジョブは出力キューに登録され、ユーザの取り出しを待つが、本シミュレータでは上記のようにラインプリンタ出力リストは自動出力としている。

3.2 TSS処理のモデル化

T(11)時間後に発生が約束されているTSSセッションの処理過程は次のようにモデル化する。シミュレーション時間が進行して、 $T(11)=0$ となると、事象EV(11)が起り、セッション開設タスク (OSタスク) が発生する。この処理に要するCPU使用時間をT(12)で表す。シミュレーション時間が進行して、 $T(12)=$

0 となると、事象 EV(12)が発生し、セッションが“READY”状態となるとともに T(13)で表わされる端末思考時間と、T(17)で表わされるセッション閉設時間がセットされる。シミュレーション時間が進行して、T(13)=0 となると、事象 EV(13)が発生し、コマンド開設処理(OS タスク)が開始する。本シミュレータでは、この処理に要する CPU 使用時間は 0 で、I/O 処理のみに時間を要する。この処理が終了すると、EV(14)が発生し、投入されたコマンドの資源要求量等の諸属性が決定され、コマンドタスクはディスパッチキューに登録される。このコマンド処理に要する CPU 使用時間を T(15)で表す。このトランザクションは、CPU サービス、I/O サービスを繰返して、シミュレーション時間が進行し、T(15)=0 となると、事象 EV(15)が発生し、コマンド終了閉設処理(OS タスク)が開始される。この処理に要する CPU 使用時間は 0 で、I/O 処理のみに時間を要する。この処理が終了すると、事象 EV(16)が発生し、端末は“READY”状態となり、新たな T(13)で表わされる端末思考時間がセットされる。以後の処理は上記と同じである。事象 EV(16)の処理の中では、T(17)を比較し、T(17)>T(13)である限り、ユーザ思考とコマンドの処理を繰返す。シミュレーション時間が進行し、T(17)=0 となると、事象 EV(17)が発生し、セッション閉設処理(OS タスク)が開始される。ただし、このとき、セッションが“READY”状態でなければ“READY”状態になるまで、セッション閉設処理を延期している。この処理に要する CPU 使用時間を T(18)で表す。シミュレーション時間が進行して、T(18)=0 となると、事象 EV(18)が発生し、セッションが消滅する。

3.3 CPU 処理と I/O 処理のモデル化

上記の説明の中で述べた OS タスク、ジョブステップタスク、およびコマンドタスクの処理は次のようにモデル化する。なお、以下においてこれらのタスクを総称してトランザクションと呼ぶ。トランザクションが発生すると、まず、そのトランザクションの CPU 優先権に対応するディスパッチキューに登録されるとともに、ディスパッチャはディスパッチキュー上で最も高い CPU 優先権をもつトランザ

クションを選択し、それに CPU 使用権を与える。CPU 使用権を与えられたトランザクションは T(22)=0 で表わされる TUNIT 切れ、T(23)=0 で表わされる CPU サービスエンド、T(25)=0 で表わされる I/O 割込、または T(27)=0 で表わされる SDM 割込のいずれかが発生するまで CPU を使用する。もし、シミュレーション時間が進行して、T(22)=0 であれば、その CPU 優先権に対応するディスパッチキューに登録され、再び CPU 使用権を与えられるまで待つ。もし、T(23)=0 であれば、このトランザクションは I/O サービスを要求するか、トランザクションの終了となる。もし、T(25)=0 であれば、そのまま、CPU を使用し続ける場合と、T(22)と同様な処理となる場合がある(詳細は第 4.3 節参照)。もし、T(27)=0 であれば、T(22)と同様な処理となる。もし、T(23)=0 で I/O サービスを要求する場合には、論理チャネルキューに登録され、I/O サービスの開始を待つ。この I/O サービスの要求する UCB が割り当てられると、T(25)で表わされる処理時間を要する I/O サービスが開始される。ただし、UCB が割り当てられても、T(24)>0 のため、チャネル I/O パスがビジー状態であると、T(24)=0 となるまで I/O サービスの開始は待たされる。また、T(25)=0 となると、I/O サービスが完了し、再びディスパッチキューに登録される。ただし、この場合、チャネルの I/O パスがビジー状態であると、ビジー解除となるまで、磁気ディスクの 1 回転に相当する時間だけ、I/O サービス完了が引き延ばされる。なお、この他に、I/O サービスとして、ページイン/アウト、スワップイン/アウト処理があるが第 4 章で詳しく述べる。

3.4 シミュレーション時間の進行

シミュレータの高速化を図るため、シミュレーション時間の進行は次の二つに分けて行う。一つは次式で決定されるシミュレーション進行時間(TP)である。

$$TP = \text{Min} \left[\begin{array}{l} T(0) \\ T(1) \\ T(11) \\ \text{Min}_{i=1,11} \left\{ \begin{array}{l} T(13)_i \\ T(17)_i \end{array} \right\} \end{array} \right]$$

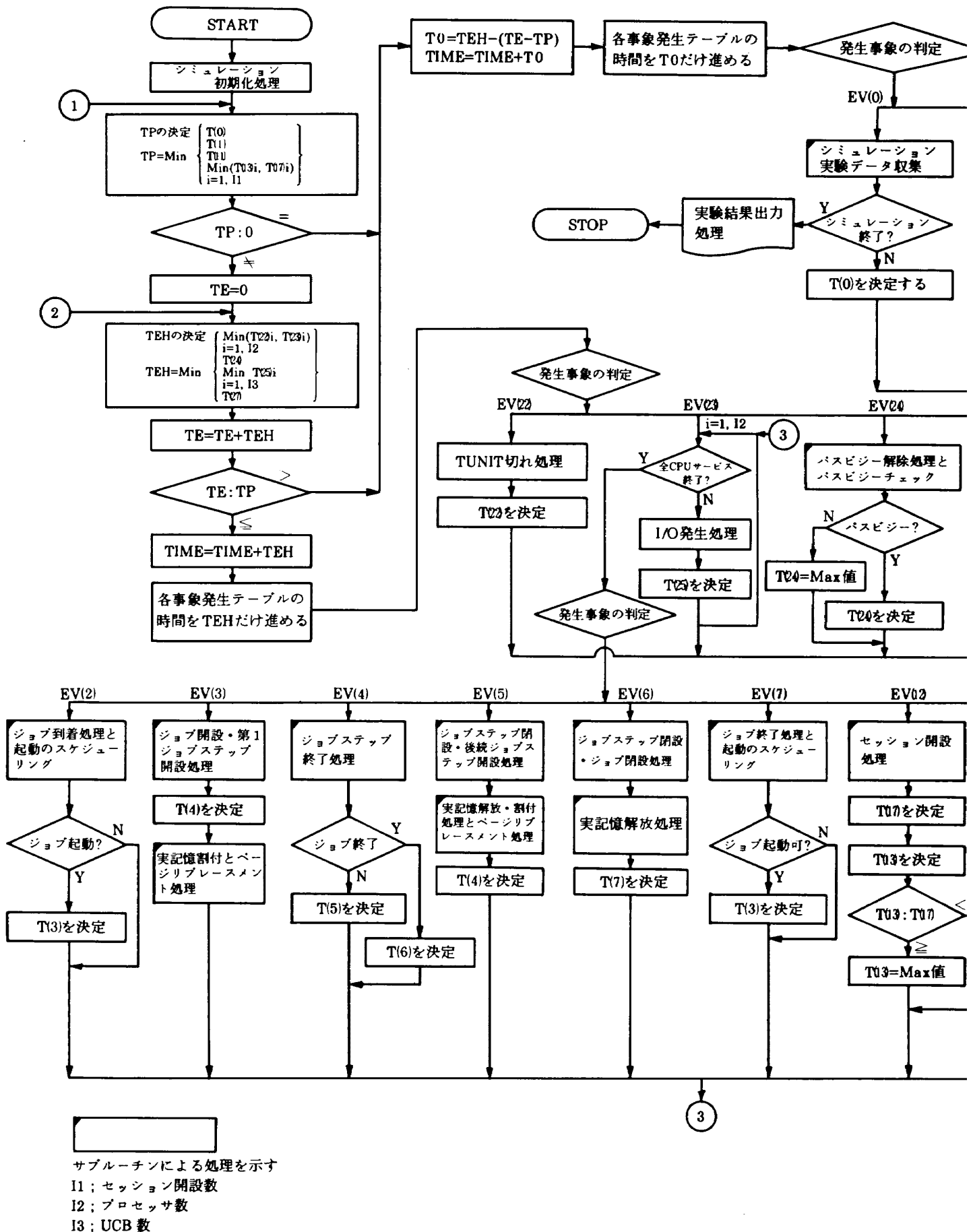
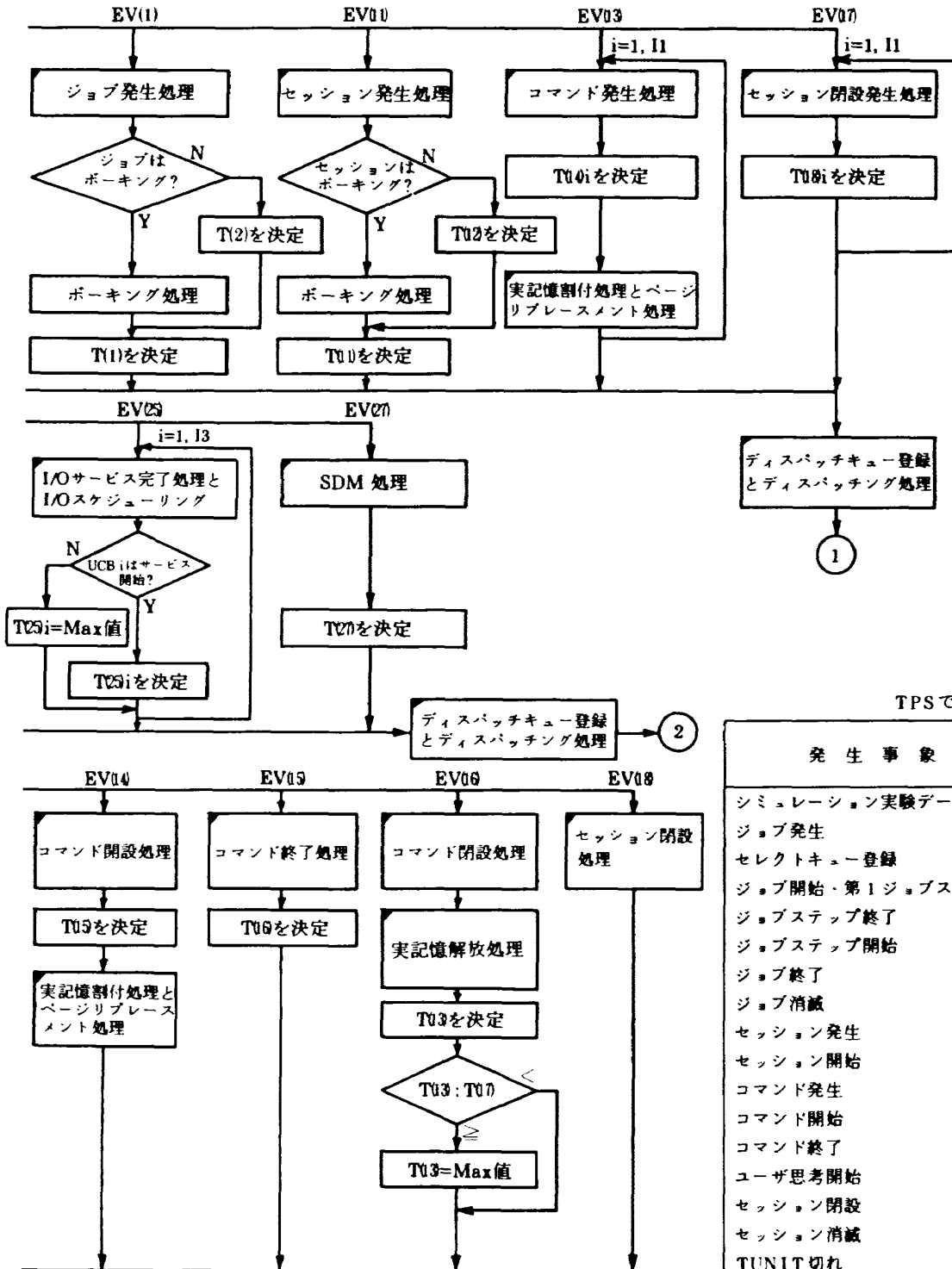


図 3.2 TPS のゼネラルフロー



TPSで発生する事象

発生事象名	事象番号	処理に要するCPU使用時間
シミュレーション実験データ収集	EV(0)	T(0)
ジョブ発生	EV(1)	T(1)
セレクトキュー登録	EV(2)	T(2)
ジョブ開始・第1ジョブステップ開始	EV(3)	T(3)
ジョブステップ終了	EV(4)	T(4)
ジョブステップ開始	EV(5)	T(5)
ジョブ終了	EV(6)	T(6)
ジョブ消滅	EV(7)	T(7)
セッション発生	EV(01)	T(01)
セッション開始	EV(02)	T(02)
コマンド発生	EV(03)	T(03)
コマンド開始	EV(04)	T(04)
コマンド終了	EV(05)	T(05)
ユーザ思考開始	EV(06)	T(06)
セッション閉設	EV(07)	T(07)
セッション消滅	EV(08)	T(08)
TUNIT切れ	EV(02)	T(02)
CPUサービスエンド	EV(23)	T(23)
I/Oパスビジョ解除	EV(24)	T(24)
I/Oサービスエンド	EV(25)	T(25)
SDM	EV(27)	T(27)

4. シミュレータの詳細

4.1 記憶装置管理の方法

4.1.1 記憶装置管理のモデル化

本シミュレータでは、仮想記憶の管理をトランザクションが要求する仮想記憶のユーザ個別領域に対してのみモデル化し、仮想記憶のシステム領域、共通領域およびシステムタスクが要求する個別領域容量は管理の対象としていない。後者全体で要求する実記憶容量は PDL/PDA から推測した一定値としている。したがって、管理の対象となる実記憶容量 MEM は、実装記憶容量を TMEM、管理の対象外としているシステム領域等を OMEM とすると、

$$MEM = TMEM - OMEM$$

である。

4.1.2 実記憶の割付・解放処理

実記憶割付処理は、

- (1) トランザクション開始時
- (2) 物理スワップイン処理発生時
- (3) ページフォルト発生時

に行われる。割付処理は、まず、要求仮想ページ数を満たすアベイラブルページが存在すれば、全ページが割り付けられ、(1)の場合には、新トランザクションが発生する。(2)、(3)は、スワップイン、ページイン処理が開始され、その完了を待ってトランザクションは実行可能になる。要求仮想ページを満たすアベイラブルページが存在しない場合には、FIFO でメモリキューにおいて要求を満たすまで待つ。実記憶解放処理は、

- (1) トランザクション終了時
- (2) 物理スワップアウト処理完了時
- (3) リプレースメントによるページアウト処理完了時

に行われる。解放されたページは、アベイラブルキューに登録され、もし、メモリキューに割付待ちトランザクションが存在すると、引き続き上記の割付処理が行われる。(1)の場合には、トランザクションの消滅である。(2)の対象となるトランザクションは SDM で決定される。(3)でページアウトの対象となるページはアベイラブルページ(ページプール)が一定値以下になると、次節で述べるページリプレ

ースメント処理で決定される。

4.1.3 ページリプレースメント処理

ページリプレースメント処理は、アベイラブルページ数が P_1 ページ以下となると開始され、アベイラブルページ数を P_2 ($P_2 > P_1$) ページにする処理であり、全トランザクション(メモリキューでトランザクション開始待ちのものも含む)を対象にしてページスティーラは以下のように行われる。

トランザクション i の要求仮想ページ数を V_i とするとき、その割付実ページ数 R_i は、

$$R_i \geq V_i \times WS$$

WS : ワーキングセット係数で 0 と 1 の間の値が保証される。特に、 $R_i = V_i \times WS$ の値をワーキングセット値と呼ぶ。いま、ページスティーラ数を PG とするとき、まず、ワーキングセット値以上のページ数をもった論理スワップアウトトランザクションを対象として各トランザクションから、次式のページ数を取り上げる。ただし、ワーキングセット値以下のものからは取り上げない。

$$A_i = \left\{ \frac{1}{UIC_i} \times \frac{R_i}{V_i} / \left(\sum_{i=1}^{11} \frac{1}{UIC_i} \times \frac{R_i}{V_i} \right) \right\} \times PG$$

I1 : ワーキングセット値以上のページ数をもつ論理スワップアウトトランザクションの数

UIC_{*i*} : トランザクション i の実記憶優先権の値

$\sum_{i=1}^{11} A_i = PG$ ならば、各トランザクションからのページスティーラページ数の決定を終了する。

$\sum_{i=1}^{11} A_i < PG$ ならば、ワーキングセット値以上のページ数を持った、実行可能トランザクションをも含めた全トランザクションを対象として、スティーラページ数を次式から決定する。

$$A_i = \left\{ \frac{1}{UIC_i} \times \frac{R_i}{V_i} / \left(\sum_{i=1}^{12} \frac{1}{UIC_i} \times \frac{R_i}{V_i} \right) \right\} \times \left(PG - \sum_{i=1}^{11} A_i \right)$$

I2 : ワーキングセット値以上のページ数をもった全トランザクションの数

この処理において、ページスティールの対象となる全トランザクションの割付ページ数がワーキングセット値になってしまい、かつ、アベイラブルページ数が P_1 ページ以下になってしまうことがある。この場合には、ページリプレースメント処理は起動されることはない。スティールされたページはページアウト処理にまわされ、I/O処理の完了後にアベイラブルページとなる。

4.1.4 アベイラブルページの管理

アベイラブルページは、未割当ページ、トランザクション終了による解放ページ、スワップアウトページおよびスティールページの集りであり、実記憶割付の対象となるページである。アベイラブルページの管理は後述のアベイラブルキューを用いている。実記憶の解放があると、その解放ページはアベイラブルキューの先頭に返却される。一方、実記憶の割付はキューの先頭から行いが、スワップインとページフォルト時には、リクレイムページ数を算出するように管理している。

4.1.4.1 アベイラブルページ管理テーブル

アベイラブルページの管理は表 4.1 の AVLQP 表、AVLQ 表で行う。IS をシステム名としたとき、AVLQP(1, IS) には、アベイラブルキューの先頭の内容を示す AVLQ 表の番地が格納される。AVLQ 表にはアベイラブルページの属性、解放トランザクション名、解放ページ数、および次のキューイングの内容を示す AVLQ 表の番地が格納されている。これにより、アベイラブルキューの先頭から最後まで のキューイングの内容を知ることができる。ここで、スワップアウトおよびページスティールの場合には、

その区別(アベイラブルページの属性)とそのトランザクション名を格納し、未割当ページとトランザクション終了による解放ページを示す場合には、トランザクション名を格納しない。このトランザクション名は、第 4.1.4.4 節で述べるスワップイン、ページフォルト時のリクレイム処理に利用される。

4.1.4.2 ページ解放時の処理

解放ページはキューの先頭に返却されるので、ページ解放時には、管理テーブルを次のように更新すればよい。まず、AVLQ 表の空き番地を探し、そこに解放ページの内容を格納する。この際、次のキューイング内容を示す AVLQ 表の番地として、AVLQP(1, IS) の内容を格納してから、AVLQP(1, IS) をここで格納した AVLQ 表の番地に更新する。

4.1.4.3 新規割付時の処理

割付要求ページ数が、AVLQP(1, IS) で示されるキューの先頭の解放ページ数以内の場合には、格納されているページ数の値を、この値から要求ページ数を引いた値に更新する。そうでなければ、キューの先頭から、次々に割り当て、割当ページ数を満したキューイング番地のページ数を残りページ数に更新し、AVLQP(1, IS) をこのキューイング番地に更新する。

4.1.4.4 スワップイン、ページフォルト時の処理

スワップインが決定されたトランザクションの実記憶割付要求ページ数はスワップアウト直前に有していたページ数である。この値は表 4.2 に示す TASK 表に格納されている。アベイラブルキューの先頭か

表 4.1 アベイラブルページ管理テーブル

AVLQP(4, IS) IS; システム名に対応

1	アベイラブルキューの先頭を示す AVLQ 表の番地
2	アベイラブルキューの最後を示す AVLQ 表の番地
3	キューイング個数
4	制御表を更新した時間

I; キューイング数に対応
AVLQ(I, 4, IS) IS; システム名に対応

1	アベイラブルページの属性
2	解放ページ数
3	解放トランザクション名
4	次にキューイングするアベイラブルページの AVLQ 表の番地

表 4.2 TASK 表
TASK(57, I) I ; トランザクション名に対応

1	トランザクションの種類	30	物理スワップアウトページ数の和
2	プログラム名	31	スワップアウト時にページアウトしないで解放された実ページ数の和
3	バッチジョブの場合にはジョブクラス	32	論理スワップインページ数の和
4	セッション名またはジョブ名	33	物理スワップインページ数の和
5	CPU使用時間の残り	34	スワップイン時リクレームされたページ数の和
6	平均CPUサービス時間	35	ページフォルトの発生回数
7	TUNIT	36	ページフォルト発生時リクレームされた回数
8	仮想記憶容量(ページ)	37	ページスティーラ回数
9	PGN	38	スティーラページ数
10	PRTY	39	ページアウトページ数の和
11	ドメイン名	40	CPU使用時間
12	TUNITの残り時間	41	UCB使用時間
13	CPUサービス時間の残り	42	VIOサービス中の表示
14	スワップアウト時の実ページ数	43	VIO発生回数
15	処理プロセッサ名	44	前回のスワップアウト継続時間
16	トランザクション発生時間	45	実ページ使用時間の和
17	VIO以外のI/O発生回数	46	(実ページ×使用時間)の和
18	割付実ページ数	47	スワップイン処理中のページ数
19	割付実ページ更新時間	48	前パフォーマンスピリオドにおけるCPU使用時間
20	VIO制御表の番地	49	ワーキングセットの値
21	トランザクションの状態	50	DPQ表の番地
22	トランザクション開始またはスワップインしてからのCPUサービス時間	51	最大割付実ページ
23	最近のスワップインした時間	52	前回の端末思考時間
24	最近のスワップアウトした時間	53	リクレーム回数
25	論理スワップアウト回数	54	空き
26	物理スワップアウト回数	55	空き
27	論理スワップアウト時間の和	56	空き
28	物理スワップアウト時間の和	57	空き
29	論理スワップアウトページ数の和		

(注) VIO機能について文献(5)参照

ら、そのトランザクション名の最初のものを探し、スワップアウトの識別がなければ、第4.1.4.3節の処理を行う。識別がなければ、そのページはリクレームされるので(スワップイン要求ページ数-リクレームページ数)だけの実記憶割付を第4.1.4.3節の処理で行う。アベイラブルキュー

の更新については、リクレームしたページが先頭の場合には、AVLQ表の次のキューイング番地を示す内容をAVLQP(1, IS)に格納する。n番目の場合には、n-1番目のAVLQ表の次のキューイング番地を示す内容をn+1番目のもので更新する。ページフォルトの場合には、ページフォルトを発生し

たトランザクション名をアベイラブルキューの先頭から全て探し、そのページ数の累積を求める。累積ページ数が0であれば、1ページの割付要求を第4.1.4.3節の処理で行う。累積ページ数が1ページ以上の場合には、

$$\frac{\text{累積ページ数}}{\text{不在ページ数}} \geq AR$$

不在ページ数：要求ページ数－割付ページ数

AR：一様乱数の値で $0 < AR < 1$

のとき、リクレイム可能、

$$\frac{\text{累積ページ数}}{\text{不在ページ数}} < AR$$

のとき、リクレイム不可能により、リクレイムの可・不可を判定する。リクレイム不可能の場合の処理は、1ページの割付要求を第4.1.4.3節の処理で行う。リクレイム可能な場合には、そのトランザクション名でキューの最初に現われるAVLQ表のページ数を1減じる。もし、1減じた結果、ページ数が0となる場合には、上記スワップイン時と同じ処理を行う。

4.2 SDM

SDMの処理の内容は多重度調整と目標多重度調整の二つに分けられる。

4.2.1 多重度調整の処理

多重度調整処理は、(1)実行多重度を目標多重度に調整し、(2)パフォーマンスピリオドの移行、(3)論理スワップアウトから物理スワップアウトへの切り換えの決定などを、一定周期ならびにTSSコマンドの発生、および終了時に行う。

4.2.1.1 実行多重度調整

(1) サービス率制御を受けるドメインの場合

システムで定義されているドメインごとに、実行多重度を目標多重度に合せる処理を行う。まず、実行多重度が目標多重度より少ない場合には、スワップアウトトランザクションでサービス率の小さいものから目標多重度に達するまでスワップインする。実行多重度が目標多重度に等しい場合には、スワップインしているトランザクションでサービス率が最大なもの、スワップアウトしているトランザクションでサービス率が最小のものと比較する。前者が後

者より大の場合には、前者をスワップアウトして、後者をスワップインする交換スワップを行う。次に、実行多重度が目標多重度より大きい場合には、目標多重度になるまで、スワップイントランザクションの中で、サービス率の大きいトランザクションからスワップアウトする。なお、サービス率は次のように計算する。

- トランザクション開始以来、スワップイン状態の場合

$$\text{サービス率} = \frac{\text{全CPUサービス時間}}{\text{現在時刻} - \text{トランザクション開始時刻}}$$

- スワップアウト状態からスワップイン状態になった場合

$$\text{サービス率} = \frac{\text{スワップインしてから受けたCPUサービス時間}}{\text{現在時刻} - \text{スワップアウト時刻}}$$

- スワップアウト状態の場合

$$\text{サービス率} = \frac{\text{スワップインしてからスワップアウトするまでに受けたCPUサービス時間}}{\text{現在時刻} - \text{スワップイン時刻}}$$

ただし、上記のCPUサービス時間がISV値より小さい場合には、CPUサービス時間を0として計算する。また、複数のトランザクションが同じサービス率の場合、スワップアウトするときにはLIFO（後着順）で、スワップインする場合にはFIFO（先着順）で行われる。

(2) FIFO制御を受けるドメインの場合

スワップイン、スワップアウトトランザクションをサービス開始時間の早い順番に目標多重度に達するまでスワップインする。当然ながら、スワップイントランザクションでスワップアウト処理すべきものと、スワップアウトトランザクションでスワップイン処理すべきものがある。

4.2.1.2 パフォーマンスピリオド移行

複数のパフォーマンスピリオドが定義されているパフォーマンスグループで動作するトランザクションに対しては、次のようなパフォーマンスピリオドの移行処理を行う。トランザクションが現パフォーマンスピリオドで受けたサービス量はそのDUR値より小さい場合には、現パフォーマンスピリオドで走行を継続させる。もし、大きければ、次のパフォ

ーマンスピリオドに移行させ、新たなパフォーマンスピリオドが属するプライオリティグループにトランザクションをキューイングしなおし、第 4.2.1.1 節の処理を行う。

4.2.1.3 論理スワップアウトから物理スワップアウトへの切り換え

論理スワップアウト状態が T_3 分以上経過しているトランザクションは、全て物理スワップアウトへ切り換える。もし、アベイラブルページが P_3 ($P_3 > P_2$) ページ以下の場合には、前回の論理スワップアウト時間が後述するシステム思考時間を超えているトランザクションのうち、最近スワップアウトされたトランザクションを一つだけ物理スワップアウトへ切り換える。なお、前回の論理スワップアウト時間がないトランザクションについては、この値をシステムで定められた 1 秒とする。

4.2.2 目標多重度調整処理

目標多重度調整処理では、システムの負荷状態（過負荷・安定・疎負荷）を判定して、目標多重度の調整、目標多重度の事前調整、およびシステム思考時間の調整を行う。

4.2.2.1 システムの負荷状態の判定

(1) 過負荷状態

リプレースメント処理が起動不可能な場合、または、平均ページング処理所要時間が T_4 秒以上、または、リプレースメント起動回数が 1 秒当り N 回以上のとき、システム状態を過負荷状態と判定する。

(2) 疎負荷状態

平均ページング処理所要時間が T_5 ($T_5 < T_4$) 秒以下、かつ、平均アベイラブルページ数が P_3 ページ以上の場合には、システム状態を疎負荷状態と判定する。

(3) 安定状態

(1)でも、(2)でもなければ、安定状態と判定する。

4.2.2.2 目標多重度の調整

(1) 過負荷状態の調整処理

過去 T_1 秒間の多重度が高すぎたと判断した場合に行う処理であり、下限多重度より大きな目標多重度をもつドメインで、一番低いワークロードレベル値のドメインの目標多重度を次式の値に低下させる。

$$\text{MAX}(\text{下限多重度}, \text{MIN}(\text{現目標多重度}-1,$$

$$\text{最高実行可能空間数}-1))$$

ここで、最高実行可能空間数は過去 T_1 秒間における実行可能トランザクション数の最大値を意味する。

(2) 疎負荷状態の調整処理

過去 T_1 秒間の多重度が低すぎたと判断した場合に行う処理であり、上限多重度未満の目標多重度をもち、かつ、目標多重度以上の平均実行可能空間数が存在するドメインで、一番大きいワークロードレベル値をもつドメインの目標多重度を 1 上げる。ここで、平均実行可能空間数は過去 T_1 秒間における実行可能トランザクション数の平均値を意味する。

(3) 安定状態の最適調整処理

過去 T_1 秒間の多重度は適正であったと判断した場合に行う処理であり、下限多重度以上の目標多重度をもつドメインのうち、最も小さいワークロードレベル値をもつドメインの目標多重度を次の値にする。

$$\text{MAX}(\text{下限多重度}, \text{MIN}(\text{現目標多重度}-1, \text{最高実行可能空間数}-1))$$

また、上限多重度未満の目標多重度をもち、かつ、目標多重度以上の平均実行可能空間数が存在するドメインのうち、上記で調整するドメインより大きなワークロードレベル値をもつドメインがあれば、最大のワークロードレベル値のドメインの目標多重度を 1 上げる。なければ、目標多重度の調整処理は行わない。

4.2.2.3 事前調整の処理

VAR 値を定義したドメインの目標多重度は事前に、システム状態が疎負荷状態と安定状態のとき、次のように調整することができる。

上限多重度未満の目標多重度をもつドメインで（目標多重度-VAR）以上の実行可能空間数をもつワークロードレベル値最大のドメインの目標多重度を 1 上げる。ただし、この処理は、疎負荷または安定状態の調整において、目標多重度を 1 上げたドメインのワークロードレベル値より大きな値である場合のみ行う。

4.2.2.4 システム思考時間の調整

システム思考時間は実記憶の負荷にもとづいて、1 秒と T_6 秒のあいだで調整する。平均アベイラ

ルページ数が P_4 以上の場合には、0.5 秒増加させ、 P_1 以下の場合には、1 秒減少させる。初期値は 1 秒である。

4.3 ディスパッチング

4.3.1 シングルプロセッサの場合

(1) TUNIT切れ、CPUサービスエンド事象発生時、ディスパッチャは事象が起きたトランザクションのCPUプライオリティ以下で最高のCPUプライオリティをもつ実行待ちトランザクションを探し、ディスパッチングを行う。ただし、このプライオリティがノンラウンドプライオリティグループに属する場合には、そのプライオリティグループの最高プライオリティのものから、実行待ちトランザクションを探し、ディスパッチングを行う。

(2) I/O割込、トランザクション発生時、ディスパッチャはCPUがアイドルの場合には、事象が発生したトランザクションのうち最高のCPUプライオリティのトランザクションを探し、ディスパッチングを行う。また、アイドルでない場合には、実行中のトランザクションのCPUプライオリティ事象発生トランザクションの最高CPUプライオリティを比較し、前者が後者より高いか、等しければ実行中のトランザクションにCPUを連続して使用させる。もし、後者が前者より高ければ、実行中トランザクションをディスパッチキューのそのCPUプライオリティの先頭にキューイングし、後者の属するCPUプライオリティから実行待ちトランザクションを探し、ディスパッチングを行う。ただし、両者が同じプライオリティグループに属し、かつ、それがノンラウンドプライオリティグループである場合には、実行中トランザクションは連続してCPUサービスを受けることができる。

(3) I/O割込、トランザクション発生と同時にTUNIT切れ、CPUサービスエンド事象が発生した場合、ディスパッチャは同時に事象発生したトランザクションの中で最高CPUプライオリティのプライオリティキューから、実行待ちトランザクションを探し、ディスパッチングを行う。ただし、最高CPUプライオリティがノンラウンドプライオリティグループに属する場合には、そのグループの中で

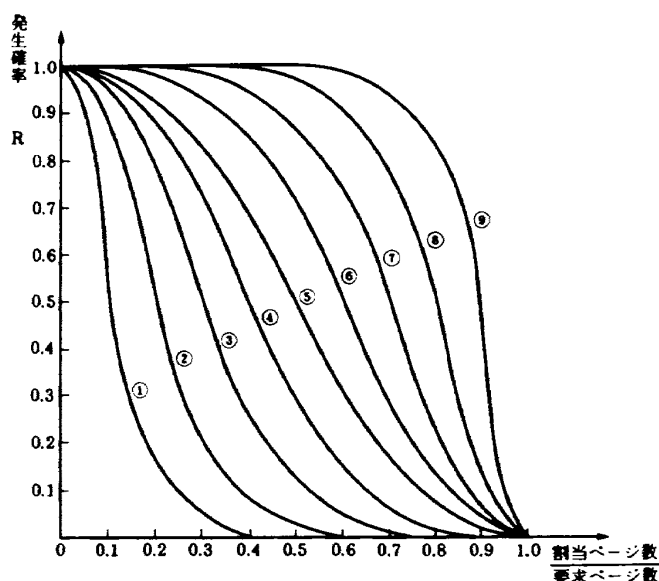


図 4.1 ページフォルト曲線

最高のCPUプライオリティのものを探し、ディスパッチングを行う。

(4) SDM発生、TUNIT切れ、CPUサービスエンド、I/O割込、トランザクション発生事象が同時に発生した場合、ディスパッチャはこれらのトランザクションの中で、最高CPUプライオリティのプライオリティキューから実行待ちトランザクションを探し、ディスパッチングを行う。

(5) ディスパッチング対象となったトランザクションについてページフォルトが発生するか、否かを次の方法で決定する。

トランザクションの内容によって、図 4.1 に示したページフォルト曲線を選択し、ページフォルト発生率 R を求める。一様乱数表から、乱数 A を取り出し、 R と比較する。このとき、 $R \leq A$ のときは、ページフォルトを発生させないでCPUをこのトランザクションに割り付ける。もし、 $R > A$ のときはページフォルトを発生させ、第 4.1.4.4 節の処理を行う。このとき、リクレイム可能となった場合には、CPUをこのトランザクションに割り付けることができるが、リクレイム不可の場合には、このトランザクションのプライオリティキューから、ディスパッチング対象となるトランザクションを再度探す。

4.3.2 マルチプロセッサの場合

マルチプロセッサの場合、各プロセッサはNEXTトランザクションテーブルを有し、次に実行すべき

ランザクションが指示される。それをNEXTランザクションと呼ぶ。

(1) I/O割込, トランザクション発生時, ディスパッチャは, まず, 一様乱数表から乱数を取り出し, これが0.5未満のときには, CPU0を自CPU, CPU1を他CPUとする。もし, 0.5以上のときには, CPU0を他CPU, CPU1を自CPUとする。次に, 他CPUがアイドルか, 否かを調べ, もし, アイドルならば, 事象発生したランザクションのうち最高CPUプライオリティのランザクションを探し, 他CPUのディスパッチングを行う。アイドルでない場合には, 事象発生ランザクションのCPUプライオリティが自CPUのNEXTランザクション(NEXTランザクションの指定がない場合には, 実行中ランザクション)より高ければ, CPU使用権を移行させる。自CPUでディスパッチングが行われない場合には, 他CPUのNEXTランザクション(NEXTランザクションがなければ実行中ランザクション)と事象発生ランザクションのCPUプライオリティを比較し, 後者が高ければ, NEXTランザクションとして, このランザクションをNEXTテーブルに登録する。この場合, 他CPUのディスパッチングは行わない。

(2) TUNIT切れ, CPUサービスエンド事象発生時, ディスパッチャは, 事象が起きたCPUに対して, 次に走行させるNEXTランザクションがある場合には, このランザクションにディスパッチングを行う。NEXTランザクションの指定がなければ事象が起きたランザクションのCPUプライオリティ以下で, 最高プライオリティをもつ実行待ちランザクションを探し, ディスパッチングを行う。

(3) I/O割込, トランザクション発生事象とTUNIT切れ, CPUサービスエンド事象が同時に発生した場合, ディスパッチャは, まず, (1)と同じ方法で自CPUと他CPUを決定する。他CPUがアイドルの場合には, 割込ランザクションおよび発生ランザクションのうち, 最高プライオリティのものを探し, (1)の処理を行うとともに, 自CPUに(2)の処理を行わせる。他CPUがアイドルでなければ, 自CPUでNEXTランザクション(NEXTト

ランザクションがなければ, 直前に走行していたランザクション)と割込ランザクションおよび発生ランザクションのCPUプライオリティを比較して, 最高CPUプライオリティのものを探す。もし後者が高ければ, 後者にディスパッチングを行う。そうでなければ, 自CPUではディスパッチングを行わず, 他CPUの実行中ランザクションとCPUプライオリティを比較し, もし, 割込ランザクション, または, 発生ランザクションのCPUプライオリティが高ければ, NEXTランザクションとして, NEXTテーブルに登録しておく。このランザクションは他CPUで実行中のランザクションがTUNIT切れ, または, CPUサービスエンドとなったときに, すぐディスパッチングされる。なお, 自CPUで切り換えチェックを行うときにNEXTランザクションが指定されているときには, まず, このランザクションと割込および発生ランザクションのCPUプライオリティを比較し, どちらかに必ずディスパッチングが行われる。

(4) SDM発生, TUNIT切れ, CPUサービスエンド, I/O割込, タスク発生事象が同時に発生した場合, ディスパッチャは最高プライオリティのプライオリティキューから実行待ちランザクションを順次, 2つ選択し, CPU0およびCPU1にディスパッチングを行う。

(5) ディスパッチング対象となったランザクションについてページフォルトが発生するか, 否かについて, 前節の(5)と同様な処理を行う。

4.3.3 ディスパッチキューの管理

ディスパッチキューは, トランザクションが発生してから消滅するまで, トランザクションが属するCPUプライオリティごとに実行待ち状態, スワップイン状態, スワップアウト状態を管理する待ち行列であり, ディスパッチャがディスパッチングの対象にするランザクションを, また, SDMがスワップイン, スワップアウトの対象にするランザクションを容易にサーチできるように工夫している。

4.3.3.1 ディスパッチキュー管理テーブル

CPUプライオリティごとのディスパッチキューの管理は, 表4.3のDPQ表, PNT表で行う。Pをプライオリティ, ISをシステム名とすると, PNT

表 4.3 ディスパッチキュー管理テーブル

PNT(3, 3, P, IS) P;プライオリティ番号に対応
IS;システム名に対応

	1	2	3
1	先頭実行待ちトランザクションを示すDPQ表の番地	先頭スワップイントランザクションを示すDPQ表の番地	先頭スワップアウトトランザクションを示すDPQ表の番地
2	最後の実行待ちトランザクションを示すDPQ表の番地	最後のスワップイントランザクションを示すDPQ表の番地	最後のスワップアウトトランザクションを示すDPQ表の番地
3	実行待ちトランザクションの数	スワップイントランザクションの数	スワップアウトトランザクションの数

DPQ(I, 3, P, IS) I;キューイング数に対応
P;プライオリティ番号に対応
IS;システム名に対応

1	システムに滞在するトランザクション名
2	次にキューイングする実行待ちトランザクションのDPQ表の番地
3	次にキューイングするスワップインまたはスワップアウトトランザクションのDPQ表の番地

(1, 1, P, IS)には実行待ちトランザクションの先頭トランザクション名を示すDPQ表の番地が格納される。DPQ(1, 1, P, IS)には、発生したトランザクション名、DPQ(1, 2, P, IS)には実行待ちトランザクションの次のキューイングの内容を示すDPQ表の番地が格納される。これにより、実行待ちトランザクションの先頭から最後までキューの内容を知ることができる。PNT(1, 2, P, IS)には、実行待ちトランザクション、およびI/O処理中トランザクション等スワップイン状態のトランザクションの先頭を示すDPQ表の番地が格納される。DPQ(1, 3, P, IS)には、スワップイン状態トランザクションの次のキューイングの内容を示すDPQ表の番地が格納されている。これにより、スワップイントランザクションの先頭から最後までキューの内容を知ることができる。PNT(1, 3, P, IS)には、論理スワップアウトおよび物理スワップアウト状態トランザクションの先頭を示すDPQ表の番地が格納されている。DPQ(1, 3, P, IS)には、スワップアウト状態トランザクションの次のキューイングの内容を示すDPQ表の番地が格納されている。これにより、スワップアウト状態トランザクションの先頭から最後までキューの内容を知ることができる。

(1) ディスパッチキューにトランザクションをキューイングする場合には、まず、トランザクションAが発生すると、DPQ表の空きの番地を登録している制御表から、番地Eを探し、このトランザクションAに関するTASK表にこの番地を格納し、DPQ(E, 1, P, IS)には、トランザクション名Aを格納する。ピリオド移行がないドメインのトランザクションはトランザクションが終了するまでこの番地を使用する。トランザクションが実行可能状態になると、まず、PNT(2, 1, P, IS)には実行待ちキューの最後のトランザクションを示すDPQ表の番地Rが格納されているので、DPQ(R, 2, P, IS)にEを格納し、このトランザクションを待ち行列の最後にキューイングしてから、PNT(2, 1, P, IS)をEに更新する。トランザクションの数を示すPNT(3, 1, P, IS)を1増加する。もし、キューイングするときに、キューにトランザクションが存在しない場合には、実行待ちキューの先頭を示すPNT(1, 1, P, IS)にEを格納する。なお、CPUサービス中に、TUNIT切れ事象発生 of トランザクションとI/Oサービス完了事象発生 of トランザクションも前記と同様に実行待ちキューの最後にキューイングするが、TUNITを使い切らないうちにCPU使用権を奪われたトランザクションはキューの先頭につ

なげる。この場合には、このトランザクションの次にキューイングする内容として、 $DPQ(E, 2, P, IS)$ にこのときの先頭トランザクションを示す $PNT(1, 1, P, IS)$ の内容を格納してから、 $PNT(1, 1, P, IS)$ をEで更新する。同時に、トランザクションの数を示す $PNT(3, 1, P, IS)$ を1増加する。また、上記と同様に処置を行い、発生トランザクションをスワップインキューにキューイングする。

(2) ディスパッチングの場合には、キューの先頭のトランザクションがドロースされる。この場合、先頭トランザクション(DPQ 表の番地をEとする)の次にキューイングするトランザクションがあるならば、このトランザクションの番地(これは $DPQ(E, 2, P, IS)$ で示される)を $PNT(1, 1, P, IS)$ に格納し、先頭トランザクションを更新する。同時に、トランザクション数を示す $PNT(3, 1, P, IS)$ を1減ずる。

(3) スワップイン状態のトランザクションAが、SDMによりスワップアウトされた場合、このトランザクションの DPQ 表の番地EをTASK表より求め、このトランザクションが実行待ち状態であれば、次のようにディスパッチキューを更新する。まず、このトランザクションがキューの先頭である場合には、(2)の処理を行う。 n 番目の場合には、 $n-1$ 番目の DPQ 表の次のキューイング番地を示す内容を、 $n+1$ 番目のもので更新する。このとき、このトランザクションが実行待ちキューの最後のトランザクションである場合には、 $PNT(2, 1, P, IS)$ の内容を $n-1$ 番目のものと更新し、トランザクションの数を示す $PNT(3, 1, P, IS)$ を1減ずる。トランザクションがI/O処理中の場合には、既に、実行待ちキューからはドロースされているので、前記の処理は省略する。次にスワップインキューからこのトランザクションを取り出す(ドロースという)処理とスワップアウトキューの最後にキューイングする処理を上記と同様の方法で行い、 DPQ 表と PNT 表を更新する。なお、スワップアウト状態のトランザクションをスワップインした場合には、上記と同様の方法でスワップアウトキューよりドロースし、スワップインキューの最後にキューイングし、実行可能状態であれば、実行待ちキューの最後にキューイングす

るように DPQ 表と PNT 表を更新する。

(4) トランザクション消滅時のキューの更新は次のように行う。トランザクション消滅は、必ず、CPUサービスエンド事象発生時に起り、このトランザクションはスワップイン状態で、かつ、実行待ちキューからは既にドロースされているので、(3)の処理でスワップインキューからドロースする処理のみを行う。さらに、このトランザクションが使用していた DPQ 表の番地を、 DPQ 表の空き番地を登録する制御表に格納し、それを使用可能な番地とする処理を行う。

(5) ピリオド移行時におけるキューの更新は、次のように行う。ピリオド移行時のトランザクションは、スワップイン状態であるので、実行待ちキューに存在するか、または、ドロースされて、I/O処理中であるかをTASK表より判定する。もし、実行待ちキューに存在する場合には、(2)の処理を行い、トランザクションをキューからドロースする。また、(3)の処理を行い、スワップインキューからもドロースする。その後、新たに移行したドメインのCPUプライオリティを調べ、(1)の処理を行って、このトランザクションを実行待ちキューとスワップインキューにキューイングする。

4.4 I/O処理装置管理の方法

本シミュレータでは、I/O処理装置の管理をトランザクションが実行中に発生するI/O処理要求の内、磁気ディスクボリュームへの要求のみをモデル化し、他のI/O機器に対するI/O処理は省略している。なお、ハードウェア構成は、図2.7に示したとおりである。

4.4.1 使用ボリュームの決定

I/O処理には、CPUサービスエンドの事象発生に起因する通常I/Oと、SDM処理時に発生したスワッピングや、ディスパッチング処理時に発生したページフォルトに起因するページングI/Oとがある。通常I/Oの場合に各UCBに対する使用確率は、後述の表5.6のとおりである。したがって、使用UCBはI/O発生のために、予め定められた値をもとに決定される。ページングI/Oの場合は、ページイン/アウト、スワップイン/アウトによりUCB

の決定が異なる。ページインの場合には、ページボリュームに対して定義された使用確率をもとに決定される。スワップインの場合には、ページインページ数より、まず、バースト回数を決定し、各バーストに対するUCBはページインの場合と同様に決定される。スワップアウトの場合には、ページアウトページ数より、まず、バースト回数決定される。各バーストに対する使用UCBはUCBリザーブ処理時にアイドルであるページボリュームが割当られる。ページアウトの場合には、リプレースメント処理ごとに発生するページアウトページ数より、まず、バースト回数決定される。各バーストに対する使用UCBはスワップアウトの場合と同様に割当られる。

4.4.2 論理チャンネルのモデル

本シミュレータでは、論理チャンネルをUCBごとに定義する。なお、共用UCBに対する論理チャンネルは1コで模擬し、それに対する各システムからのI/O要求は先着順に管理する。各UCB(ページボリュームは除く)がアイドルになると、その論理チャンネルの先頭からI/O優先権が最高のI/O要求を探し、それに対するI/Oサービス時間を決定し、UCBリザーブ処理を行う。これにより、I/O完了までこのUCBはビジー状態となる。ページボリュームに対しては、I/O要求のスケジューリングを次のように行う。通常I/OとページングI/Oは交互に行う。通常I/Oの場合はその論理チャンネルの先頭からI/O優先権が最高のI/O要求を探す。ページングI/Oの場合には、ページインとスワップインがページアウトおよびスワップアウトよりも優先する。ページインの場合は、I/Oサービス時間を決定し、スワップイン、スワップアウトおよびページアウトの場合には、I/Oサービス時間はバースト時間とする。

4.4.3 I/Oパスのモデル化

UCBリザーブが行われるたびに、パスビジー状態でなければ、I/OパスビジーのためにI/Oサービスの開始が待たされるか、否かの決定を次のように行う。もし、パスがビジー状態であれば、パスビジー解除の事象発生までI/Oパスビジーチェックの決定が待たされる。

(1) チャンネルビジーのチェック

$$B = \left(\text{データ転送時間} \times \frac{\text{システムごとの単位時間当りのI/O処理中UCB数}}{\text{システムごとのチャンネル数}} \right)^4$$

$B \geq 1$ 様乱数のときにパスビジーと判定する。

(2) DIRビジーのチェック

$$B = \left(\text{データ転送時間} \times \frac{\text{単位時間当りのI/O処理中UCB数}}{\text{DIRの数}} \right)^4$$

$B \geq 1$ 様乱数のときにパスビジーと判定する。

(3) ADPビジーのチェック

$$B = \left(\text{データ転送時間} \times \frac{\text{ストリングごとの単位時間当りの処理中UCB数}}{\text{ADPの数}} \right)^2$$

$B \geq 1$ 様乱数のときにパスビジーと判定する。

上記三つのチェックをクリアした場合にはUCBサービスを開始する。もし、いずれかのチェックでパスビジーと判定された場合には、一定時間パスビジーとする処置を行う。この場合には、パスビジー解除の事象発生まで待たされた後、上記のパスビジーチェックが再び行われる。なお、図2.7から分るようにDPXAビジーは発生しない。

4.4.4 I/Oパスビジー解除発生時の処理

パスビジーのためにI/Oサービス開始が待たされているI/O要求に対し、第4.4.3節と同じ処理を行う。

4.4.5 I/Oサービス完了事象発生時の処理

I/Oサービスが完了すると、まず、パスビジー状態を調べる。もし、パスビジーであればこのI/Oサービスを完了とせず、磁気ディスクの一回転待ち時間だけサービス時間が延長する処置を行う。もし、パスビジー状態でなければ、第4.4.3節と同じ処理でパスビジーの判定を行う。このとき、パスビジーとならなかった場合には、I/Oサービス完了処理を行い、このUCBに対するスケジューリングを第4.4.2節にもとづいて行う。I/Oサービス完了処理は、以下のようである。

(1) 通常I/Oサービス完了の場合には、I/Oサービス完了したトランザクションがスワップイン状態であれば、これをディスパッチキューの実行待ち

キューに登録し、割込表を更新する。

(2) ページイン I/O サービス完了の場合には、I/O サービス完了したトランザクションが、スワップイン状態のトランザクションであれば、これをディスパッチキューに登録し、割込表を更新する。もし、そのトランザクションが、論理スワップアウトが決定されていれば、論理スワップアウトする処理を行う。そのトランザクションが、物理スワップアウトが決定されていれば、スワップアウト処理を発生させる。

(3) スワップイン処理完了の場合には、(2)と同じ処理を行う。

(4) スワップアウト処理完了の場合には、そのトランザクションのスワップインが決定されていれば、第 4.1.2 節の処理を行う。

(5) ページアウト処理完了の場合には、第 4.1.2 節の処理を行う。

4.5 各事象発生時の処理

4.5.1 ジョブ発生処理

ジョブ発生事象が発生したならば、ジョブを 1 件発生させる。ジョブの属性として定義された確率より、ジョブクラス、ジョブ入力量、出力量を決定し、ジョブ発生時間、ジョブ番号とともに、表 4.4 に示す JOB 表を作成する。次に、ジョブステップ構成列を定義された確率より決定し、各ジョブステップごとに、CPU 使用時間、仮想記憶使用量、平均

表 4.4 JOB 表

JOB(10, I) I ; ジョブ名に対応

1	ジョブ発生番号
2	ジョブクラス
3	ジョブステップの数
4	入力カード枚数
5	出力リスト行数
6	ジョブ発生時間
7	セレクトキュー到着時間
8	ジョブ開設時間
9	実行中ジョブステップ番号
10	プロセッサ名

表 4.5 JSP 表

I ; ジョブステップ番号に対応
JSP(6, I, J) J ; ジョブ名に対応

1	プログラム名
2	CUP 使用時間
3	仮想記憶使用量
4	平均 CPU サービス時間
5	VIO READ/WRITE ページ数
6	VIO 発生回数

(注) VIO 機能については文献(5)参照

CPU サービス時間等のシステム資源要求量を定義された確率により決定して、表 4.5 に示す JSP 表を作成する。この後、ジョブ入力処理を行う JES/E サブタスクを発生させ、このタスクをディスパッチキューに登録するとともに、次のジョブ発生時間をポアソン確率分布関数により求め、制御表にセットする。また、事象発生時に各ジョブクラスのジョブが規定値以上にセレクトキューにキューイングされた場合、または各種制御表に空きが不足して登録不可能となった場合には、ジョブ発生処理を行わない(ボーキング処理という)。トランザクション発生時の主な処理は、表 4.2 に示す TASK 表に、トランザクションの種類(ジョブステップタスク、コマンドタスクまたは OS タスクの区別)、プログラム名(ジョブステップ名、またはコマンド種別)、このトランザクションが使用するシステム資源要求量(CPU 使用時間、仮想記憶使用量、平均 CPU サービス時間)、各種 SDM 属性値、処理を行うプロセッサ名、トランザクション発生時間、指数分布関数より決定された第 1 回 CPU サービス時間等を格納することと、ドメインの多重度に関する統計処理と多重度の更新、第 4.1.2 節で述べた実記憶割付処理、および第 4.3.3.2 節で述べたディスパッチキューへの登録処理を行うことである。

4.5.2 セッション発生処理

セッション発生事象が発生したならば、まず、既に開設されているセッション数が設定したセッション開設数に至っているか、否かを判定する。もし、設定値まで開設されている場合には、このセッションのボーキング処理を行う。設定値に至っていない

表 4.6 SSN表

SSN(5, I) I ;セッション名に対応

1	セッション発生番号
2	コマンド発生時間の残り
3	セッション開設発生時間の残り
4	セッション発生時間
5	最近の端末思考開始時間

れば、表 4.6 に示す SSN 表にセッション番号、セッション発生時間を格納し、セッション開設処理タスクを発生させ、このタスクをディスパッチキューに登録する。次に、セッション開設数を更新し、次のセッション発生時間を定義されたポアソン確率分布関数により決定し、制御表にセットする。

4.5.3 コマンド発生処理

この事象が発生したならば、定義された確率より、発生コマンドの種別を決定し、そのシステム資源使用量 (CPU 使用時間、仮想記憶使用量、平均 CPU サービス時間) を定義された確率にもとづき決定して、表 4.7 に示す CMD 表に格納する。次に、コマンド開設処理タスクを発生させ、このタスクをディスパッチキューに登録する。さらに、このセッションの SSN 表より端末思考時間の統計処理を行う。また、SSN 表にコマンド処理中を示す値を格納する。

4.5.4 セッション開設処理

この事象が発生したならば、事象発生セッションの SSN 表により、セッションがユーザ思考中であるか、コマンド処理中であるかを確認する。後者の場合、事象発生はコマンド処理終了まで遅らせ、SSN 表にこれを示す値を格納して、この処理を終らせる。セッションがユーザ思考中であれば、まず、SSN 表より、端末思考時間の統計処理計算を行った後にセッション開設処理タスクを発生させ、このタスクをディスパッチキューに登録する。同時に、SSN 表にセッション開設処理中を示す値を格納する。

4.5.5 CPU サービスエンド発生処理

この事象を発生したトランザクションの残り CPU 使用時間が 0 であれば、トランザクション終了とな

表 4.7 CMD表

CMD(4, I) I ;セッション名に対応

1	コマンド名
2	CPU 使用時間
3	仮想記憶使用量
4	平均 CPU サービス時間

る。0 でなければ、CPU サービスエンド事象は I/O 発生事象と同義であり、次の CPU サービス時間を定義された確率分布関数により決定し、さらに、TUNIT 時間を初期値にセットする。次に、第 4.4 節の I/O 処理のスケジューリングが行われる。前者の場合、まず、トランザクションの終了処理を行い、トランザクションを消滅させる。この処理の主な内容は、以下のとおりである。TASK 表よりシステム資源使用量等の統計処理、トランザクションの経過時間の計算、およびトランザクション実行中におけるページングやスワッピング状況等の統計処理を行う。さらに、第 4.3.3.2 節の処理を行った後に TASK 表を全部または一部 0 クリアする。次に CPU サービスエンド事象を発生したトランザクションの種類により、以下に述べる新たなタスクを発生させる。

(1) ジョブ入力処理 (JES/E サブタスク) の終了により、EV(2) が発生したならば、到着ジョブはそのジョブクラスのセレクトキューにキューイングされる。その際、キューの更新前に、キューの統計処理を行う。次にジョブ起動のスケジューリング処理を行い、起動の場合には、多重度の更新とジョブ開設・第 1 ジョブステップ開設処理タスクを発生させ、ディスパッチキューにこのタスクをキューイングする。

(2) ジョブ開設・第 1 ジョブステップ開設処理タスクの終了により、EV(3) が発生したならば、ジョブステップの処理が開始される (トランザクションの発生)。このトランザクションは要求する実記憶の割付が行われれば、ディスパッチキューに登録される。

(3) ジョブステップのトランザクション終了により、EV(4) が発生する。もし、後続ジョブステップ

があれば、ジョブステップ閉設・後続ジョブステップ開設処理タスクを発生させ、ディスパッチキューに登録する。後続ジョブステップがなければ、ジョブステップ閉設・ジョブ閉設処理タスクを発生させ、ディスパッチキューに登録する。

(4) ジョブステップ閉設・後続ジョブステップ開設処理タスク終了により、EV(5)が発生したならば、EV(3)発生時と同じ処理を行う。

(5) ジョブステップ閉設・ジョブ閉設処理タスク終了により、EV(6)が発生したならば、ジョブ消滅タスクを発生させ、ディスパッチキューにキューイングする。

(6) ジョブ消滅JES/Eサブタスク終了により、EV(7)が発生したならば、ジョブが使用したシステム資源使用量、経過時間および実行中のページング、スワッピング情報を統計処理し、JOB表、JSP表を0クリアする。同時に、ジョブの多重度を更新し、ジョブ起動のスケジューリング処理をEV(2)発生時と同様に行う。

(7) セッション開設処理タスク終了により、EV(12)が発生したならば、セッションが端末思考時間に入ったことを示すために、SSN表に事象発生時間を格納する。また、定義されたポアソン確率分布関数により、セッション閉設時間とコマンド発生時間を決定し、SSN表に格納する。さらに、実行不可能空間数を更新する。

(8) コマンド開設処理タスク終了により、EV(14)が発生したならば、コマンドの処理を開始させる(トランザクションの発生)とともに、SDMに、第4.2.1節で述べた多重度調整処理を行わせる。

(9) コマンドのトランザクション終了により、EV(15)が発生したならば、SDMにより、第4.2.1節で述べた多重度調整処理を行わせる。続いて、コマンド閉設処理タスクを発生させ、ディスパッチキューに登録する。

(10) コマンド閉設処理タスク終了により、EV(16)が発生したならば、SSN表より、セッション閉設事象が保留になっているか、否かを判定し、保留になっていれば、セッション閉設処理タスクを発生させる。保留になっていなければ、(7)と同様に端末思考時間を決定する。この値がセッション閉設時間よ

り小のときは、これをSSN表に格納するが、もし、大の場合には、以降、このセッションから、コマンドの発生はないことを示す値をSSN表に格納する。

4.5.6 TUNIT切れ発生処理

TUNIT切れを発生したトランザクションのTUNIT値をTASK表より求め、同表にこの値を格納してから、このトランザクションをディスパッチキューに登録する。

4.5.7 SDM発生処理

SDM事象発生時の処理については、第4.2節に述べた。

5. シミュレーション実験

5.1 実稼動状況

シミュレーション実験によりシステムの稼動状況を評価するため、まず、実稼動状況を把握しなければならない。ここでは、PDL/PDAにもとづいた平均的実稼動状況について述べる。

システムは、概ね、午前10時から12時の時間帯は、バッチジョブ処理要求、TSS処理要求ともに、非常に混雑している。そこで、この時間帯にPDLを流し、多数の実稼動データを収集し、PDAで解析した。その中から利用率がコンスタントに高いデータを5つ抽出し、混雑時における平均的実稼動状況を求めた。その結果を表5.1に示す(ドメイン15~12の定義については第5.2.5節参照)。この値を以後、シミュレーション実験結果に対する評価基準とする。なお、MAINシステムでは、実記憶競合が発生しないように運用を行っているので、ページング処理に関する情報はFEシステムに対してだけ示した。

5.2 実験環境の設定

5.2.1 ジョブの属性

ジョブの発生時間間隔にもとづき、ポアソン分布でTPSへ投入されたジョブは、ジョブ母集団のジョブクラス構成比率、ジョブクラス内ジョブステップ列構成比率およびジョブステップごとのシステム資源使用量の確率分布にもとづき、まず、ジョブクラスが決定され、次に、そのジョブステップ列が決定され、このジョブステップ列を構成する各ジョブステ

表 5.1 混雑時の平均的システム稼働状況

FE システム	CPU稼働率	67.95 %
	システムオーバヘッド率	16.61 %
	運用上のオーバヘッド率	13.66 %
	ページング回数	12.48 回 / 秒
	スワッピングページ数	5.31 ページ / 秒
	稼働端末数	29.66 台
	終了トランザクションの数	
	コマンド	5466 件 / 時間
	バッチジョブ	31.5 件 / 時間
	アクティブトランザクションの数	
コマンド	4.6 件	
バッチジョブ	5.7 件	
コマンドの応答時間		
ドメイン 15	1.99 秒	
ドメイン 14	12.37 秒	
ドメイン 13	35.93 秒	
ドメイン 12	200.83 秒	
コマンド発生間隔	0.659 秒	
MAIN システム	CPU稼働率	99.99 %
	システムオーバヘッド率	3.18 %
	CPU寄与率	
	SMAL	40.22 %
	LARG	25.92 %
LONG	28.16 %	
ジョブ発生間隔	63 秒	

ップに対し、CPU使用時間、仮想記憶使用量、平均CPUサービス時間が決定される。なお、これらの値が運用制限値を超えている場合には、制限値に置き換えられる。昭和58年度のアカウント情報統計解析結果にもとづいたジョブ母集団を規定する各種パラメータの推定値は以下のとおりである。表5.2はジョブ母集団におけるジョブクラス構成比率を示している。発生したジョブは、この構成比率にもとづいてジョブクラスが決定される。表5.3はジョブ母集団によるジョブステップ構成比率を示している。ジョブクラスが決定されたジョブは、この構

成比率にもとづいて、ジョブステップ列が決定される。なお、ジョブクラスとジョブステップ列の決定はすべてジョブが発生するたびに一様乱数を用いて行う。表5.4は各種ジョブステップのシステム資源使用量を決定する分布関数とその母数の推定値を示している。各ジョブステップの各種装置使用時間は、同表に示した確率分布に従って算出される。また、仮想記憶使用量とファイルアクセス回数は同表の回帰式にもとづき、決定されたCPU使用時間から算出される。平均CPUサービス時間はCPU使用時間/ファイルアクセス回数から算出する。シミュレー

表 5.2 ジョブクラス構成比率と各種制限値

ジョブクラス		項目	構成比率 (%)	処理装置使用時間制限値 (秒)	仮想記憶使用量制限値 (MB)
FE システム	SHRT		49.3	300	1
	FSML		6.3	1800	2
	FRNT		10.3	120	2
MAIN システム	SMAL		20.6	600	2
	LARG		10.6	600	4
	LONG		2.9	3600	8

表 5.3 ジョブステップ列構成比率

ジョブステップ列番号	ジョブステップ構成	ジョブステップ数	構成比率 (%)
1	FORTRAN-LIED-GO	3	55.27
2	GO	1	11.25
3	FORTRAN	1	9.28
4	FORTRAN-FORTRAN-LIED-GO	4	5.90
5	FORTRAN-FORTRAN-FORTRAN-LIED-GO	5	3.91
6	FORTRAN-LIED	2	3.68
7	UTILITY	1	3.26
8	FORTRAN(8 ジョブステップ)	8	1.45
9	FORTRAN(6 ジョブステップ)-LIED-GO	8	1.03
10	UTILITY-FORTRAN-LIED-GO	4	0.98
11	FORTRAN(4 ジョブステップ)-LIED-GO	6	0.89
12	FORTRAN(5 ジョブステップ)-LIED-GO	7	0.89
13	(UTILITY-FORTRAN)を4回繰返す	8	0.83
14	UTILITY-UTILITY-UTILITY	3	0.70
15	FORTRAN(4 ジョブステップ)-LIED-GO-FORTRAN(2 ジョブステップ)	8	0.68

ション時間を進める一つの要因である CPU サービス時間はこの平均 CPU サービス時間を平均値とした指数確率分布にしたがって、毎回決定される。なお、LIED、UTILITYにおける仮想記憶使用量は一定値である。シミュレーション実験では、ジョブ発生時間間隔を表 5.1 より多少短くし、30 秒としている。

5.2.2 TSS コマンドの属性

TSS セッションの発生間隔にもとづき、ポアン

ン分布で開設された TSS セッションから端末思考時間間隔にもとづき、コマンドが発生する。TSS コマンドのシステム資源使用量の確率分布については、まだ、解析がなされていない。このため、本シミュレーション実験では、SMF から収集することができるコマンドごとの CPU 使用時間、仮想記憶使用量、ファイルアクセス回数にもとづき、TSS コマンドの属性を次のように決定した値を使用する。TSS はその処理の即応性を保証するため、四つの

表 5.4 ジョブステップのシステム資源使用量の分布

ジョブステップ名	項目	CP使用時間 $\log_{10}(\text{CP})$ の分布	CP使用時間 $\log_{10}(\text{CP})$ と 仮想記録容量 $\log_{10}(V)$ の回帰式	CP使用時間 $\log_{10}(\text{CP})$ とファイルアクセス回数 $\log_{10}(\text{EXCP})$ の回帰式
FE システム	FORTRAN	平均	$\log_{10}(V) = 1.76 + 0.1916(\log_{10}(\text{CP}) - 0.344)$	$\log_{10}(\text{EXCP}) = 1.84 + 0.5262(\log_{10}(\text{CP}) - 0.344)$
		標準偏差 の正規分布		
	LIED	平均	$V = 32$	$\log_{10}(\text{EXCP}) = 2.8 + 1.0584(\log_{10}(\text{CP}) + 0.0228)$
		標準偏差 の正規分布		
	UTILITY	平均	$V = 13$	$\log_{10}(\text{EXCP}) = 1.81 + 1.7103(\log_{10}(\text{CP}) + 0.623)$
		標準偏差		
SHRT-RUN	平均	$\log_{10}(V) = 1.95 + 0.26355(\log_{10}(\text{CP}) - 0.802)$	$\log_{10}(\text{EXCP}/\text{CP}) = 1.18 - 0.81173(\log_{10}(\text{CP}) - 0.802)$	
	標準偏差 の正規分布			
FSML-RUN	平均			
FORTRAN	平均	$\log_{10}(V) = 1.86 + 0.17665(\log_{10}(\text{CP}) - 0.191)$	$\log_{10}(\text{EXCP}) = 2.09 + 0.69453(\log_{10}(\text{CP}) + 0.37)$	
	標準偏差 の正規分布			
	平均	$V1 = 32$ $V2 = 80$ ($V1 : 80\%$, $V2 : 20\%$)	$\log_{10}(\text{EXCP}) = 2.99 + 0.95514(\log_{10}(\text{CP}) - 0.191)$	
	標準偏差 の正規分布			
	SMAL-RUN	平均	$\log_{10}(V) = 2.46 + 0.27261(\log_{10}(\text{CP}) - 1.48)$	$\log_{10}(\text{EXCP}/\text{CP}) = 1.2 - 0.72075(\log_{10}(\text{CP}) - 1.48)$
		標準偏差 の正規分布		
LARG-RUN	平均			
	標準偏差 の正規分布			
LONG-RUN	平均			
	標準偏差 の正規分布			

表 5.5 コマンドの平均的システム資源使用量と発生比率

項目 コマンド種	CPU使用時間 (秒)	仮想記憶使用量 (KB)	ファイルアクセス 回数	発生比率
1 種	0.117	50	48.07	95.0 %
2 種	0.755	70	221.92	4.3 %
3 種	1.790	227	439.93	0.5 %
4 種	12.870	305	2424.40	0.2 %

表 5.6 トランザクションの磁気ディスクアクセス率

トランザクション の種類 ボリューム名	FORTRAN (%)	LIED (%)	GO (%)	UTILITY (%)	TSS コマンド (%)
USER01	0.272	0.007	0.001	0.0	0.888
USER02	4.537	9.472	3.352	0.532	5.806
USER03	4.638	2.159	5.125	12.388	7.500
USER04	5.862	2.296	5.800	1.584	7.216
USER05	2.739	1.225	3.404	0.444	4.970
USER06	1.495	1.424	1.008	0.050	5.628
USER07	2.054	20.039	2.018	0.121	12.144
USER08	8.868	3.182	3.493	2.395	9.592
USER09	0.0	0.0	0.0	0.0	0.0
USER10	6.983	3.070	3.443	11.612	8.675
USER11	4.635	2.313	9.600	0.448	7.215
USER12	5.386	1.667	2.658	9.729	6.248
PAGEF0	0.242	0.811	1.352	0.187	0.052
PAGEF1	2.647	5.067	2.586	1.805	0.218
PAGEM0	0.572	2.679	5.081	0.687	0.0
PAGEM1	1.148	3.759	7.522	0.852	0.002
NASYSF	0.0	0.023	0.0	0.0	0.634
NASYSM	0.0	0.0	0.0	0.0	0.030
SPOOL1	10.669	9.540	11.627	11.053	5.561
SPOOL2	13.130	10.195	11.145	17.930	5.473
SPOOL3	11.335	10.538	10.392	13.093	5.965
SPOOL4	12.788	10.534	10.392	15.091	6.182

パフォーマンスピリオドに区切って運用している（詳しくは、第 5.2.5 節参照）。第 1 パフォーマンスピリオドは CPU 使用時間が 500 ミリ秒以内、第 2 パフォーマンスピリオドは CPU 使用時間が 1 秒以内、第 3 パフォーマンスピリオドは 2 秒以内、第 4 パフォーマンスピリオドは 2 秒以上に設定している。そこで、上記収集データをこの四つのグループに分類し、各グループごとの CPU 使用時間、仮想記憶使用量、ファイルアクセス回数の平均値を求めた。その値を表 5.5 に示す。同表には、各グループのコマンドの発生比率も示している。シミュレーション実験では、コマンド発生事象ごとに、この比率にもとづき、どのグループのコマンドかを決定し、決定されたグループのコマンドのシステム資源使用量は表 5.5 に示した一定値を使用する。

5.2.3 磁気ディスクへの I/O 発生頻度

各磁気ディスクへの I/O 発生頻度は、昭和 58 年度の SMF のデータセット使用情報にもとづき解析した結果、トランザクションごとに表 5.6 に示すと

おりである。各トランザクションが I/O を発生するたびに、同表にもとづき、一様乱数を用いて使用磁気ディスクが決定される。

5.2.4 システムオーバヘッド

システムオーバヘッドは OS のオーバヘッドとシステム運用上のオーバヘッドに分類される。前者は、TPS に組込まれているオーバヘッドで、SDM、ページングスーパーバイザ、JES/E (RDR, ISDVR, CI, OUTSERV, WTR, PURGE), TSS (セッション開設、セッション閉設、コマンド開設、コマンド閉設) 等の処理に要するオーバヘッドである。後者は TPS に組込まれていないオーバヘッドで、マスタスケジューラ⁵⁾、AOF⁷⁾、NET⁸⁾、ライタ¹⁾、ユーザコンソール²⁾、PP コンソール²⁾、オンライン XY プロッタ²⁾、テレビディスプレイ²⁾ 等の処理に要するオーバヘッドであり、第 5.1 節で述べたシステム稼動時間帯における、その平均的システム資源使用量を PDL/PDA にもとづいて解析した結果は、表 5.7 に示すとおりである。

表 5.7 システム運用上のオーバヘッド

1 秒当りの CPU 使用時間 (秒)		1 秒当りの 総 IO 回数	IO の 内 訳		備 考	
			磁気ディスク ボリューム名	IO 回数		
MAIN システム	0.018	4 回	PAGEM1	1		
			USER03	1		
			USER08	2		
FE システム	0.041	34 回	NASYSF	2		
			USER03	4		
			USER05	4		
			USER06	4		
			USER07	4		
			USER08	2		
			USER03 } USER06 }	2		{ USER03 (確率 0.7) USER06 (確率 0.3)
			SPOOL1 } SPOOL2 } SPOOL3 }	12		{ SPOOL1 (確率 0.25) SPOOL2 (確率 0.25) SPOOL3 (確率 0.25)
			SPOOL4 }			SPOOL4 (確率 0.25)

5.2.5 SDMパラメータの設定

SDMパラメータは表 5.8 に示すように設定した。同表において、ドメイン番号 1, 3, 5 は LONG, LARG, SMAL クラスのサービスジョブステップを処理するドメインであり、ドメイン番号 2, 4, 6 は LONG, LARG, SMAL クラスのランジョブステップを処理するドメインである。さらに、ドメイン番号 7, 8, 10 は SHRT, FSML, FRNT クラスのジョブを処理するドメインである。ドメイン番号 15, 14, 13, 12 は TSS コマンドを処理するドメインであり、それぞれパフォーマンスピリオド 1, 2, 3, 4 に対応している。このような SDMパラメータの

設定の主旨は、次のとおりである。MAIN システムでは、LONG, LARG, SMAL クラスのサービスジョブステップの処理を優先させ、各クラスのランジョブステップ待ち行列を生成させる²⁾ サービスジョブステップ処理の実行優先順位は LONG > LARG > SMAL とし、大規模ランジョブステップができるだけ実行待ち状態になるようにする。また、ランジョブステップの実行優先順位は LONG = LARG = SMAL とし、各クラスのジョブ処理は公平に行う。FE システムでは、TSS 処理の CPU, 実記憶, I/O 優先順位を高位とし、TSS の応答時間がバッチジョブ処理の影響をできるだけ受けないようにする。

表 5.8 SDMパラメータ

(1) ドメインの設定

設定項目 ドメイン名	下 限 多重度	上 限 多重度	FKWL	VAL
1	1	9	50	0
2	1	1	-	0
3	1	9	45	0
4	1	1	-	0
5	1	9	40	0
6	2	2	-	0
7	2	2	-	0
8	0	1	-	0
10	1	5	50	0
15	4	50	90	10
14	4	50	80	0
13	1	10	70	0
12	1	10	60	0

(2) PFGNの設定

設定項目 パフォーマンス グループ名	ドメイン 名	PRTY	ISV (秒)	UIC	IOP	DUR (秒)
1	1	49	1.0	4	0	∞
2	2	30	∞	1	0	∞
3	3	45	1.0	4	0	∞
4	4	30	∞	1	0	∞
5	5	40	1.0	3	0	∞
6	6	30	∞	2	0	∞
7	7	30	0.5	1	0	∞
8	8	19	0.5	1	0	∞
10	10	50	20.0	2	0	∞
15	15	90	2.0	5	2	0.5
	14	80	2.0	4	1	0.5
	13	70	1.0	3	0	2.0
	12	60	1.0	2	0	∞

(3) PGNの設定

設定項目 プライオリティ グループ番号	PRTY	TUNIT (秒)	ROUND属性
6	1 ~ 19	0.050	NO
7	20 ~ 39	0.050	YES
8	40 ~ 49	0.050	NO
10	50 ~ 59	0.050	NO
11	60 ~ 69	0.250	YES
12	70 ~ 79	0.250	YES
15	80 ~ 89	0.250	YES
20	90 ~ 99	0.250	YES

TSS 処理においては、パフォーマンスピリオド 1, 2, 3, 4 の順に CPU, 実記憶, 並びに I/O 優先順位を高位とし, 即応性コマンドの応答性を保証する。バッチ処理の実行優先順位は, $FRNT > SHRT > FSML$ とし, 小規模バッチジョブを優先的に処理する。FSML クラスのジョブは CPU 稼働率の向上の目的で運用し, TSS 処理, 小規模ジョブにより実記憶の競合が発生した場合には, 実行多重度が 0 となる。実記憶に余裕があると, FSML クラスのジョブは CPU 稼働率の向上に大きく寄与する。以上の主旨のもとに SDM パラメータの設定を行ったが, その有効性はシミュレーション実験で確認しなければならない。

5.2.6 シミュレーション時間

計算機システムの各種統計量が安定するまで, シミュレーションを続行する必要があるが, 実験経験から判断して, シミュレーションは実時間にして 10 時間で打ち切った。

5.3 実験項目

本実験の目的は TPS の妥当性を確認するとともに, 以下に述べる実験を計画し, SDM パラメータ設定値の有効性を示し, システム運用上の問題点を摘出し, システム利用効率向上のための新たな提案を行うことにある。

(1) 実験 I

第 5.1 節で述べた, 混雑時における平均的稼働状況を模擬した実験であり, TPS の妥当性を確認するためのものである。

(2) 実験 II

疎結合複合システムにおける I/O 競合の度合を調べるために, MAIN システムの稼働を停止した場合の実験である。

(3) 実験 III

第 5.2.4 節で述べた, システムオーバヘッドの影響の割合を調べる実験である。

(4) 実験 IV

多重度によらず, ジョブクラスごとの CPU 寄与率を可変にし得る方法として, TUNIT が PGN で固定せず, ドメインごとに設定し得るようになることが考えられる。この新たな方法が実現できれば,

多重度を減少し得, それだけ, 実記憶資源の有効利用が図れる。実験 IV はこの提案の有効性を確認するために行う。

(5) 実験 V

FE システムの CPU 稼働率をバッチ処理の増大により向上させた場合に, TSS の即応性がどのような影響を受けるかを調べる実験である。影響の度合が比較的微小であれば, FE システムの SDM パラメータの設定方法が有効なことを示しており, FE システムの運用効率を著しく向上させることができる。

(6) 実験 VI

M-180 II AD を 1 台にした場合に TSS の即応性がどのような影響を受けるかを調べる実験である。

(7) 実験 VII

アクティブ端末数の増大が TSS の即応性をどの程度低下させるかを調べる実験である。この実験により, FE システムがサポートし得る端末数を予測することができる。

(8) 実験 VIII

端末思考時間の減少が TSS の即応性をどの程度低下させるかを調べる実験である。

(9) 実験 IX

TSS 処理に使用される磁気ディスクの台数を減少すると, TSS 処理における I/O の競合が増大してくる。この影響が, TSS の即応性をどの程度低下させるかを調べる実験である。

(10) 実験 X

使用可能実記憶容量の減少および, TSS 処理における仮想ページ参照の局所性の低下は実記憶の競合を増大させ, ページング I/O 回数を増大させる。ページング I/O の多発は, よく知られているように, スラッシング現象を発生させ, システム運用効率を著しく低下させる。実験 X はこの影響の度合を定性的に調べる目的で行う。

5.4 実験結果

以上の準備のもとで, 10 時間のシミュレーション実験 I ~ X を行った。その結果は, 表 5.9 に示すとおりである。同表の各項目は, TPS の出力ルーチンが, 一定時間間隔ごとに収集する計算機システ

表 5.9(1) 実験結果(実験Ⅰ, 実験Ⅱ, 実験Ⅲ, 実験Ⅳ)

項目	実験名	実験Ⅰ	実験Ⅱ	実験Ⅲ	実験Ⅳ
MAINシステム					
CPU稼働率(%)		99.51		99.06	99.40
システムオーバヘッド率(%)		2.23		4.06	2.25
CPU寄与率(%)					
SMAL		47.46		47.05	48.4
LARG		24.59		24.19	24.8
LONG		26.00		24.45	24.8
ジョブ経過時間(分)					
SMAL		91.2		74.9	87.3
LARG		49.9		36.4	50.2
LONG		223.5		255.2	260.4
FEシステム					
CPU稼働率(%)		54.22	49.62	54.05	89.20
システムオーバヘッド率(%)		20.86	21.40	23.96	25.31
使用可能実ページ数(ページ)		2000	2000	2000	2000
1秒当りのページング回数(回)		0.0	0.0	0.0	0.0
1秒当りのスワッピングページ数(ページ)		0.0	0.0	0.0	0.0
1秒当りのリクレイムページ数(ページ)		0.0	0.0	0.0	0.0
コマンド応答時間(秒)					
ドメイン 15		2.08	2.05	2.18	2.52
ドメイン 14		9.85	9.70	10.32	11.97
ドメイン 13		19.06	18.56	20.27	23.37
ドメイン 12		115.48	112.93	126.69	144.46
コマンド到着間隔(秒)		0.764	0.761	0.764	0.785
ジョブ経過時間(秒)					
SHRT		64.3	60.1	70.8	82.2
FRNT		27.7	29.2	28.4	33.7
FSML		705.3	488.5	661.6	6253.6

ム利用情報記録項目の一部であり、同表の数値は、10時間の収集データの統計量を表わしている。同表の各項目の定義は次のとおりである。

(1) CPUに関する項目

CPU稼働率=総CPU使用時間/シミュレーション実験時間

CPU寄与率:各ジョブクラスのジョブが受け

たCPUサービスの割合

(2) 実記憶に関する項目

ページング回数/秒=総ページング回数/シミュレーション時間(秒)

スワッピングページ数/秒=総スワッピングページ数/シミュレーション時間(秒)

スワッピング回数/秒=総スワッピング回数/

表 5.9(2) 実験結果(実験Ⅳ)

項目 \ 実験	1	2	3	4	5	6	7	8	9	10	11	12
実行多重度												
SMAL	2	2	2	2	2	2	1	1	1	1	1	1
LARG	1	1	1	1	1	1	1	1	1	1	1	1
LONG	1	1	1	1	1	1	1	1	1	1	1	1
CPU寄与率(%)												
SMAL	49.0	49.0	48.7	48.1	47.7	47.2	32.7	48.3	67.4	77.8	81.9	84.2
LARG	24.5	24.5	24.3	24.0	23.9	23.5	32.7	25.0	15.5	10.5	10.8	9.7
LONG	24.3	24.5	25.4	26.8	27.8	28.9	32.4	24.8	15.3	11.4	8.5	7.5
経過時間(秒)												
SMAL(サービス)	16.9	17.8	17.0	18.6	17.8	18.1	18.0	18.0	17.5	17.8	18.5	18.9
SMAL(RUN)	3350	3371	3071	3613	3855	3742	5561	3852	2472	2086	2070	1879
LARG(サービス)	16.4	16.8	16.6	16.2	15.9	16.8	16.5	16.2	16.0	16.7	16.9	16.9
LARG(RUN)	3579	3486	3443	3722	3755	3518	2559	3703	5188	5874	6106	6729
LONG(サービス)	20.4	19.3	17.5	19.8	19.4	19.5	19.3	19.5	19.3	20.7	21.4	21.4
LONG(RUN)	20723	19176	19620	19252	19205	18189	16746	19492	23211	17430	21443	24319
TUNIT(ms)												
SMAL(RUN)	50	100	250	500	750	1000	50	100	250	500	750	1000
LARG(RUN)	50	100	250	500	750	1000	50	50	50	50	50	50
LONG(RUN)	50	100	250	500	750	1000	50	50	50	50	50	50

表 5.9(3) 実験結果(実験Ⅴ; FEシステム)

項目 \ 実験	1	2	3	4	5	6	7	8
バッチジョブ多重度								
SHRT	0	2	2	2	2	2	2	2
FRNT	0	5	5	5	5	5	0	0
FSML	0	0	1	1	1	1	2	2
バッチジョブ発生比率(%)								
SHRT	49.3	49.3	60.0	80.0	80.0	80.0	80.0	70.0
FRNT	10.3	10.3	10.3	10.3	10.3	0.0	0.0	0.0
FSML	6.3	6.3	3.0	3.0	9.7	20.0	20.0	30.0
CPU稼働率(%)	18.6	26.5	38.8	41.8	60.9	63.0	85.3	85.9
コマンド応答時間(秒)								
ドメイン 15	1.98	2.04	2.06	2.07	2.10	2.11	2.10	2.10
ドメイン 14	9.37	9.61	9.76	9.83	9.98	9.99	10.00	9.97
ドメイン 13	17.66	18.44	18.83	18.73	18.98	19.06	19.19	19.07
ドメイン 12	106.18	111.15	112.09	115.10	114.83	115.67	116.69	116.87

表 5.9(4) 実験結果 (実験Ⅶ, 実験Ⅷ; FEシステム)

項目 \ 実験	1	2	3	4	5	6	7	8	9
TSS 端末数	20	30	40	50	30	30	30	30	30
端末思考時間 (秒)	20	20	20	20	30	25	20	15	10
コマンド到着間隔 (秒)	1.13	0.76	0.58	0.47	1.10	0.92	0.76	0.60	0.44
CPU稼働率 (%)	48.1	54.2	56.8	58.6	48.5	50.8	54.2	53.3	59.7
コマンド応答時間 (秒)									
ドメイン 15	1.99	2.08	2.18	2.29	2.00	2.03	2.08	2.15	2.31
ドメイン 14	9.34	9.85	10.43	11.10	9.40	9.57	9.85	10.30	11.30
ドメイン 13	17.96	19.06	20.52	22.26	17.80	18.18	19.06	19.91	22.86
ドメイン 12	105.8	115.48	125.14	139.89	107.38	110.00	115.48	121.88	143.03
使用可能ページ数 (ページ)	2000	2000	2000	2000	2000	2000	2000	2000	2000
1秒当りのページング回数	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.005

表 5.9(5) 実験結果 (実験Ⅸ; FEシステム)

項目 \ 実験	1	2	3	4	5
磁気ディスクボリューム台数 (台)	10	8	6	4	2
CPU稼働率 (%)	54.2	51.8	52.6	51.4	40.19
1秒当りのページング回数					0.149
使用可能実ページ数 (ページ)	2000	2000	2000	2000	2000
コマンド応答時間 (秒)					
ドメイン 15	2.08	2.13	2.22	2.51	4.26
ドメイン 14	9.85	10.14	10.66	12.51	31.36
ドメイン 13	19.06	19.67	20.88	25.46	139.23
ドメイン 12	115.48	119.56	131.39	160.73	1110.10

シミュレーション時間 (秒)

$$\text{リクレイムページ数 / 秒} = \frac{\text{総リクレイムページ数}}{\text{シミュレーション時間 (秒)}}$$

(3) I/Oに関する項目

$$\text{ビジー率} = \frac{\text{総 I/O 使用時間}}{\text{シミュレーション時間}}$$

$$\text{SIO回数 / 秒} = \frac{\text{総 SIO回数}}{\text{シミュレーション時間 (秒)}} \text{ (注)}$$

I/O待ち時間: 論理チャネルでの平均待ち時間

間

$$\text{I/O所要時間: SIOから I/O完了までの時間の平均値}$$

(4) 経過時間

$$\text{ジョブ経過時間: ジョブ発生からジョブ終了までの時間の平均値}$$

$$\text{トランザクション経過時間: トランザクション発生からトランザクション終了までの時間の平均値}$$
(注) SIOはスタートIOの略である。⁹⁾

表 5.9(6) 実験結果(実験 X)

使用可能 実ページ(ページ)	項目 ページフォ ルト発生曲 線の番号	コマンド応答時間 (秒)				CPU稼働率 (%)	1秒当りのペ ージング回数 (回)
		ドメイン15	ドメイン14	ドメイン13	ドメイン12		
2000	7	2.08	9.85	19.06	115.48	54.22	0.0
1600	7	2.08	9.85	19.06	115.48	54.22	0.0
1500	7	2.08	9.80	19.08	114.93	52.49	0.001
1450	7	2.09	9.88	19.56	118.33	53.90	0.23
1400	7	4.76	22.04	81.13	248.59	46.52	15.780
1200	7	11.09	54.55	34.62	155.1	35.99	39.094
1000	7	33.04	155.82	635.07	316.3	22.82	69.926
1500	6	2.08	9.81	19.08	114.27	52.49	0.001
1400	6	2.07	9.84	18.98	114.27	52.29	0.044
1300	6	3.15	17.28	42.85	249.03	51.71	7.002
1200	6	2.09	9.86	19.31	117.76	51.87	0.264
1100	6	8.44	44.43	30.67	376.86	40.21	31.330
1000	6	73.07	403.61	1174.3	140.80	14.01	87.108
900	6	91.74	504.75	1740.5	7252.80	12.78	89.889
1500	5	2.07	9.81	18.94	114.81	52.10	0.001
1400	5	2.08	9.83	18.97	113.39	52.06	0.007
1300	5	2.08	9.84	19.01	116.46	52.27	0.099
1200	5	2.21	10.28	22.39	138.81	50.90	0.752
1100	5	2.24	10.02	21.53	134.59	50.96	1.541
1000	5	2.17	10.12	20.68	139.35	52.50	1.770
900	5	3.02	15.40	32.84	210.59	49.42	10.148
1500	4	2.07	9.81	18.94	114.81	52.09	0.001
1400	4	2.08	9.82	19.01	113.23	52.29	0.022
1300	4	2.08	9.87	19.01	116.34	53.52	0.102
1200	4	2.07	9.83	18.68	114.87	50.67	0.032
1500	1	2.07	9.81	18.94	114.81	52.10	0.001
1400	1	2.08	9.82	19.00	113.23	52.29	0.022
1300	1	2.08	9.88	19.00	115.30	54.11	0.076
1200	1	2.08	9.83	18.68	114.17	50.67	0.032

表 5.9(7) 実験結果(実験Ⅰ, 実験Ⅲ)

	ビジー率 (%)		1秒当りのSIO回数 (回)		レスポンスタイム (ms)		処理待ち時間 (ms)	
	実験Ⅰ	実験Ⅲ	実験Ⅰ	実験Ⅲ	実験Ⅰ	実験Ⅲ	実験Ⅰ	実験Ⅲ
USER01	3.52	3.57	1.17	1.19	30.31	30.29	0.26	0.25
USER02	23.84	23.89	7.94	7.95	35.03	34.92	4.99	4.88
USER03	26.25	46.08	8.74	15.35	35.30	47.60	5.26	17.57
USER04	20.30	20.52	6.76	6.84	33.62	33.60	3.58	3.57
USER05	13.00	25.15	4.33	8.38	31.93	34.22	1.89	8.38
USER06	17.29	31.46	5.76	10.46	32.81	39.32	2.77	9.29
USER07	45.61	57.80	15.19	19.25	36.35	51.98	4.20	21.95
USER08	29.14	41.73	1.17	13.89	30.31	39.84	6.31	9.81
USER09	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
USER10	26.01	26.49	8.66	8.82	35.02	35.08	4.99	5.04
USER11	23.29	23.63	7.76	7.87	35.58	35.54	5.54	5.50
USER12	19.90	20.09	6.63	6.69	33.54	33.56	3.50	3.52
PAGEM0	3.85	3.69	1.19	1.13	33.08	33.24	14.52	16.28
PAGEM1	4.66	7.50	1.46	2.40	32.65	32.50	11.92	2.40
PAGEF0	2.08	1.97	0.69	0.66	30.36	30.36	0.32	0.32
PAGEF1	5.73	5.54	1.91	1.84	31.01	30.81	0.97	1.84
SPOOL1	27.42	36.22	9.13	12.06	36.84	44.94	6.81	14.90
SPOOL2	29.27	37.90	9.75	12.62	37.24	44.97	7.20	14.94
SPOOL3	31.11	39.95	10.36	13.31	37.57	45.98	7.53	13.31
SPOOL4	26.56	35.04	8.85	11.67	35.85	43.33	5.82	11.67
NASYSM	7.82	7.55	1.96	1.89	42.73	42.61	2.74	2.63
NASYSF	2.34	10.39	0.59	2.60	40.26	41.01	0.28	1.03

6. 考 察

6.1 実験動状況との比較

表 5.1 で示した平均的実験動状況を, 表 5.9(1)と(7)に示した実験Ⅰの結果を比較する。MAINシステムについてはCPU稼働率, そのオーバーヘッドの割合および各ジョブクラスの寄与率がよい一致を示している。FEシステムについては, CPU稼働率, そのオーバーヘッドの割合, TSSの応答時間が比較的よい一致を示している。また, 比較データはないが, ジョブの経過時間とI/O処理について, シミュレーション実験から次のことが推測できる。

LONG, LARG, SMALクラスのジョブの経過時間はそれぞれ, 約 220分, 50分, 42分であり, SHRT, FRNT, FSMLクラスのジョブの経過時間はそれぞれ, 約 1分, 0.5分, 12分である。これらの値はジョブのワークロードに依存して大幅に変化するが, 本シミュレータではI/O処理を実際的に模擬しているため, ジョブの経過時間の予測がかなり現実的に行えるものと考えられる。I/O処理については, スプールボリューム, ユーザボリュームともに負荷分散が適切になされていることを示しており, このことから, チャンネル, 磁気ディスク制御装置, 磁気ディスク装置のハードウェア構成方法が適切であっ

たといえる。なお、USER07ボリュームの利用頻度が高いのは、システムライブラリデータセットが格納されていることに起因している。一方、実記憶に関しては、実稼動状況と実験上の結果が大きく異なっている。これは、仮想記憶方式のモデル化において、共通領域の模擬を省略し、個別ユーザ領域のみを模擬しているため使用可能実記憶の定め方が大変難しいことを示している。しかし、第6.6節で示すように、使用可能実記憶容量とユーザプログラムのページフォルト率を可変にしてページング回数とシステム利用効率を調べた結果は、いわゆるスラッシング現象をとらえており、使用可能実記憶容量とユーザプログラムのページフォルト率を適切に定めれば、かなり、現実的なシミュレーションが可能である。以上のことにより、TPSはほぼ妥当であると確認し得る。

6.2 疎結合方式の影響

航技研計算機システムは、MAINシステムとFEシステムがI/O資源を共有した疎結合複合システムで運用している。MAINシステムはバッチジョブのみを処理しているため、FEシステムから発生するI/O要求との競合で、経過時間が増大する割合は無視できる。一方、FEシステムはTSS処理を行っているため、MAINシステムから発生するI/O要求との競合でTSSの応答時間が増大するならば、これを無視することはできない。実験Ⅱの結果(表5.9(1)参照)は、MAINシステムから発生するI/O要求がTSSの応答時間に殆んど影響を与えていないことを示している。

6.3 オーバヘッドの影響の度合

以上の実験は表5.7で示したシステム運用上のオーバヘッドをTPSに組み込まないで行った。実験Ⅲの結果(表5.9(1)と(7)参照)は、このオーバヘッドの度合がどの程度かを示している。これによると、MAINシステム、FEシステムともにCPU稼働率は殆んど影響を受けない。I/O資源の利用率は全ボリュームともに増大を示しているが、それがTSSの応答時間に影響を及ぼす度合は、十分許容できる範囲内である。したがって、以後の実験にお

いても、オーバヘッドは除外している。

6.4 CPU寄与率変更のための提案

第6.1節で示したように、MAINシステムにおける各ジョブクラスのCPU寄与率は、実稼動状況とよく一致している。ここでは、まず、TUNIT値に対する寄与率の変化を図6.1(1)に示す。同図から分るように、TUNIT値を変化させても寄与率を殆んど変えることができない。したがって、寄与率を変化させる手段としては多重度を調整する方法しかない。しかし、多重度の変化は寄与率を段階的に変化させ、連続的に変えることはできない。CPU寄与率の連続的変更のためには、TUNIT値をPGNの属性とせず、ドメインの属性とし、ドメインごとにTUNIT値を設定できる方法が考えられる。表5.9(2)は、この新提案が実現した場合のシミュレーション結果を示している。図6.1(2)はTUNIT値に対してCPU寄与率が連続的に変化させ得ることを示している。このことは、各ジョブクラス多重度を1にして、CPU寄与率を変更し得ることを意味しているから、実記憶資源を最大限に有効利用できる。

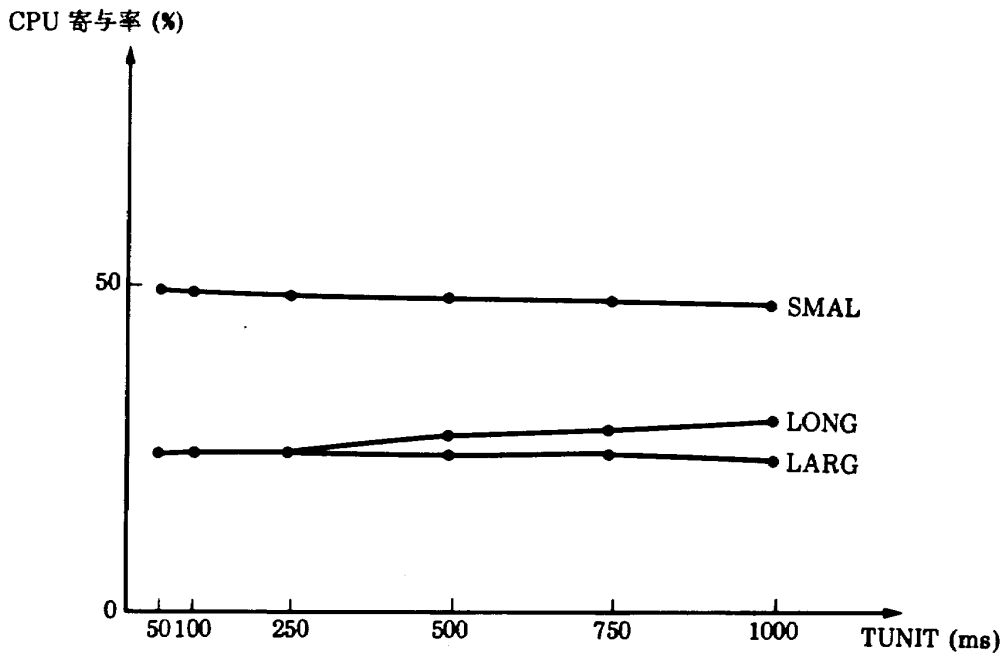
6.5 TSSの即応性について

(1) 実験Ⅴの結果はFEシステムのバッチ処理を増大させ、CPU稼働率を向上させた場合、TSSの即応性がどのような影響を受けるかを示している。表5.9(3)から分るように、CPU稼働率を十分向上させても、TSSの応答時間の増大は十分許容範囲内にある。このことは、SDMパラメータの設定方法の有効性を示している。

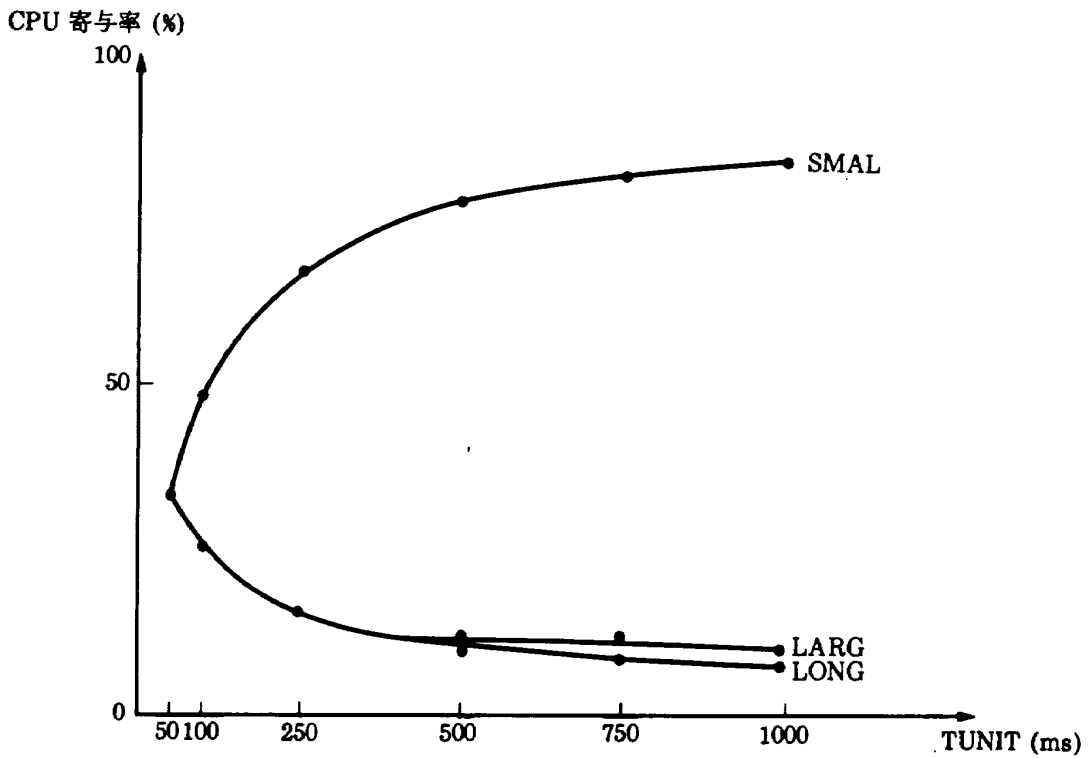
(2) 実験Ⅵの結果は、FEシステムのCPUを1台にした場合のTSSの応答時間に与える影響を示している。表5.9(1)から分るように、TSSの応答時間の増大はほぼ許容範囲内にある。

(3) 実験Ⅶの結果は、アクティブ端末数の増大がTSSの即応性に与える影響を示している。表5.9(4)から分るように、端末思考時間が20秒の場合、本計算機システムはアクティブ端末として50台は十分にサポートできると予測される。

(4) 実験Ⅷの結果は、端末思考時間の減少がTSSの応答時間に及ぼす影響の度合を示している。表



(1) 現状の場合



(2) 提案が実現した場合

図 6.1 TUNITとCPU寄与率

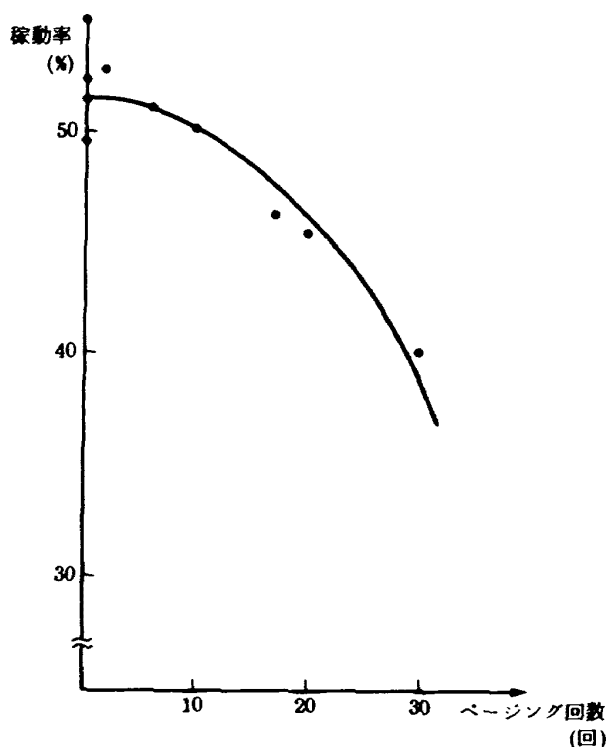
5.9(4)から分るように、アクティブ端末数が30台の場合、端末思考時間が10秒まで減少してもTSSの応答時間の増大はほぼ許容範囲内であるといえる。

(5) 実験Ⅸの結果は、TSS処理に使用される磁気ディスク台数の減少が、TSSの応答時間に及ぼす影響を示している。表5.9(5)から分るように、磁気ディスクの台数を4台以下にするとTSSの応答

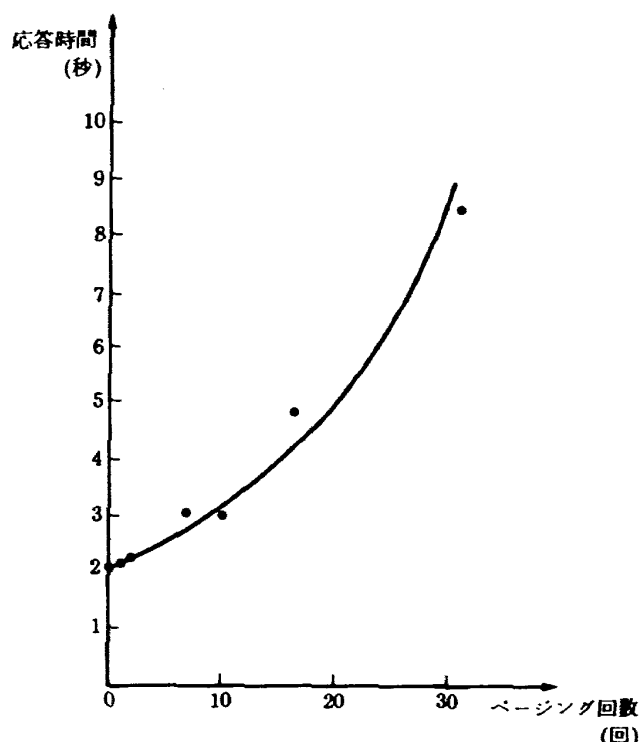
時間が急激に増大し始めるものと予想される。

6.6 スラッシング現象について

実験Ⅹの結果は、使用可能実記憶容量とユーザプログラムのページフォルト率を種々変えた場合に発生するページング処理がシステム利用効率に及ぼす影響を示している。表5.9(6)からページング回数と



(1) ページング回数と稼働率



(2) ページング回数とドメイン 15 の応答時間

図 6.2 スラッシング現象

CPU稼働率、およびTSSの応答時間を取り出し、図示すると、図6.2のとおりである。同図から分るように、TSSの即応性を保証するには、ページング回数は5回/秒以内であることが望まれる。ページング回数が10回/秒以上になると、CPU稼働率がかなり低下し、いわゆるフラッシング現象が現われると予想される。なお、本シミュレータでは共通領域およびシステム運用上の仮想空間から発生するページング処理はモデル化していないので、表5.1の平均的実稼働状況におけるページング回数、約12回/秒とは差異がある。シミュレーション実験結果から予測すると、共通領域、およびシステム運用上の仮想空間から発生するページング回数は約7回/秒と思われる。

7. 結 言

本稿では、航技研計算機システム用に開発したTPS(トランザクション・プロセッシング・シミュレータ)の概要とシミュレーション実験結果について述べた。TPSはCPU資源とI/O処理資源に関してかなり実地的なモデル化を行っており、これ

ら各資源の有効利用の方策や、混雑緩和の方法を検討する道具として有用であると考えられる。実記憶資源に関しては、仮想記憶方式のモデル化において、共通領域の取り扱いを省略したので、実記憶に関しては近似的な結果しか得られないが、仮想記憶方式の本質的問題であるスラッシング現象はとら得ることができた。シミュレーション実験の結果から得られた結論をまとめると以下の様である。

(1) 混雑時の平均的稼働状況とそれを模擬したシミュレーション実験から、TPSの妥当性を確認できた。

(2) 疎結合複合システムにおけるI/O処理の競合は、TSSの応答時間に殆んど影響を及ぼさないことが確認できた。

(3) FEシステムのCPU稼働率を向上させても、TSSの応答時間の増大は、十分許容範囲内であることから、SDMパラメータ設定方法が適切であることが確認できた。

(4) アクティブ端末数、端末思考時間、TSS処理に使用される磁気ディスク台数、CPU処理能力等に関してTSSの即応性に関する予測がTPSで十

分可能であることが確認できた。

(5) ジョブクラスごとのCPU寄与率を連続的に変更し得る新提案を行い、その有効性をTPSで確認した。この提案が実現できれば、実記憶資源は一段と有効利用できる。

今後の問題として、本シミュレータにおける実記憶資源の管理をより実際のモデルに構築し直す予定である。終りにあたり、表 5.4 を提供していただいた計算センタ・計算研究室の中村絹代技官、表 5.1 の作成において助力をいただいた計算課小松増美事務官および、計算機システムのハードウェア並びにソフトウェアに関して多くの情報を提供していただいた富士通(株)とファコム・ハイタック(株)の諸氏に感謝の意を表します。

引用文献

- 1) FACOM OSIV/F4, JES/E運用手引書
- 2) 畑山ら;次期航技研計算機システムの運用計画, 航技研 TM-430, 1981年2月
- 3) FACOM OSIV/F4, SMF説明書
- 4) FACOM OSIV/F4, PDL/PDA使用手引書
- 5) FACOM OSIV/F4, 解説書
- 6) FACOM OSIV/F4, TSS解説書
- 7) FACOM OSIV/F4, AOF説明書
- 8) FACOM OSIV/F4, VTAM-F機能説明書
- 9) FACOM Mシリーズ, ハードウェア機能説明書Ⅱ(機能編)

航空宇宙技術研究所資料 543号

昭和60年4月発行

発行所 航空宇宙技術研究所
東京都調布市深大寺東町7丁目44番地1
電話武蔵野三鷹(0422)47-5911(大代表)〒182

印刷所 株式会社三興印刷
東京都新宿区信濃町12三河ビル
