UDC 681.322.06

# 東京大学生産技術研究所における計算機運用システムの開発について（その 2）

Development of Computer Job-shop Operating System

古谷千恵*・柴田碧**

Chie FURUTANI and Heki SHIBATA

## 4. システム開発の方法

### （1） OS II のシステム管理機能

ここで言う運用システムの開発とは，OS II のジョブ管理に含まれている制御パラメータを使用の実態に合せて決定することである．これら制御パラメータには，多重度，優先度，グループ制御パラメータがある．OS II のように複雑なシステムになると，各々が競合関係を持ち，いわゆるマニュアルで述べられている以上に複雑な機能を有するようになる．ここでは，それらを実験的に解明しながら進んだ．以下その要点について述べる．

（a） ジョブの多重度

ジョブの多重度とは，マルチジョブ処理に際して同時に走るジョブ・ストリームの数のことである．この多重度について，大型計算機システムのバッチ処理の場合3が最適であるとうい報告がある[2]．この報告は，テスト用ジョブでの効率測定と待ち行列による理論解析に基いている．われわれは，本所のベンチマークテストジョブを用いて最適の多重度を探索した．その結果，われわれのテストデータにおいても図5のように3が最良であることが確かめられた．一方，メーカでも別途これを確認したという報告があった．さらに，利用者ジョブによるオンラインのテストによっても現在の主記憶容量（192 K語）では多重度4以上では実効が上がらないことが確認された．（多重度を3より増加させても処理能力があがらないのは，主として主記憶容量に制限されて実効的な多重度が増加しないことを意味する．）

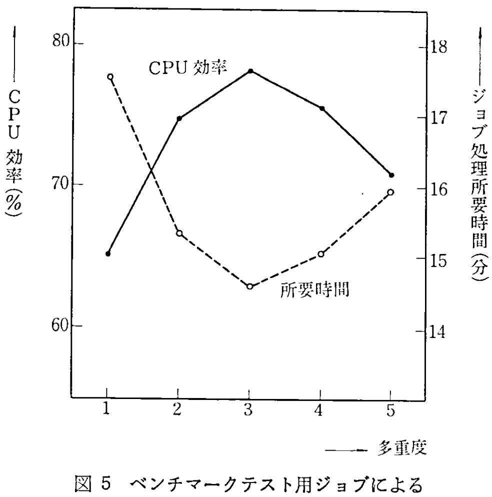



図 5　ベンチマークテスト用ジョブによる多重度テスト結果（ジョブ数 10）

（b） ジョブの優先順位に関する制御

マルチプログラミングでは，資源（特に処理装置）をいくつかのプログラムが競合して使用する場合の実行の順序が問題になる．この実行の順序は，優先度（レベル）というプログラムの持つ資格によって決められる．優先度には，ジョブ起動の優先順位とジョブ・ステップ実行の優先順位がある．

〔i〕 ジョブ起動の優先順位

ジョブ起動の優先順位は，0から3まで4レベルの指定ができる．値の大きい程優先度が高い。入力待ち行列はこの優先順位に従って並べられる．

〔ii〕 ジョブ・ステップ実行の優先順位

ジョブ・ステップ実行の優先順位とは，ジョブ・ステップに対して指定するものであってジョブ・ステップの使用する資源（特に処理装置）使用の優先度である．OS II では各プログラムの CPU 連続処理時間をもとに，自動的にジョブ間の CPU 配分を最適に保つダイナミック・ディスパッチング機能があるので，ここではこれを制御パラメータとしない．

優先度の問題としては，急行，普通，長時間ジョブ間のジョブ開始優先度，センタとリモートバッチ・ステーション間のジョブ開始償先度を決定しなければならない．

（c） ジョブ・グループ制御

本システムにおいてはグループは3であり，グループ名は EPG, TSG, LNG とした．ジョブ開始優先度に従ってジョブの入力待ち行列が作られる．この三つの入力待行列の中から（多重度が3なので）三つのジョブ・ストリームを作るときにグループ間優先度が働く．

また，ジョブ・グループ内パラメータとして MULTI（ジョブ・グループ内のジョブの多重度を指定），SIZE（ジョブ・グループに割り当てる主記憶領域の上限値を指定）がある．よって，各々のグループ間優先度，およびグループ内多重度を決定する必要がある．この外，グループ間の開始条件の制御に関して WAIT パラメータがある．なお，これらの詳細は後に図6によって具体的に説明する．

* 東京大学生産技術研究所　計算機室
** 〃　第2部，計算機室

**(2) システム開発の手順**

長時間ジョブは集中して特定日（たとえば週2日間とか）に流すことを前提に2通りの運用システムの開発を行なった．それぞれ急行，普通ジョブ群と，長時間，急行，普通ジョブ群とである．また，過去の使用実態から平常期と繁忙期で運用システムを変える必要が予想される．この2種類のシステムを繁忙度によって2段にシステム変更を行なう際，計算機室側としてはできるだけ簡単にでき，かつ利用者側からみて変更なしに行えるようにすることが必要である．開発経過において平常期のジョブ形態にあっては，上記，急行，普通ジョブのシステムに必要に応じ長時間ジョブの部分を追加するだけですむことが確認された．

その結果，運用システムの構成を表2のように決定しジョブ制御マクロを活用し新しいマクロを作成した．

表2　運用システムの構成

| 期間＼ジョブ群 | 急行，普通ジョブ | 長時間，急行，普通ジョブ |
|---|---|---|
| 平常期 | *A* | *A* |
| 繁忙期 | *A* | *B* |

まずグループ内多重度，ジョブ開始優先度，グループ間優先度のパラメータの組合せをベンチマークテストで暫定的に決定しその後，利用者用ジョブのテストによってさらに改良を加えた．このパラメータの決定にあたってはコンソール・ディスプレイに表示されるジョブの実行状態，待ち行列の状態，システム資源の使用状況を動的に把握しながら，制御しうるパラメータを変更するという方法をとった．この heuristic な方法は，従来の平均値的なパフォーマンスを用いる方法よりもシステムの動的な動きを把握して，運用方式に反映させることができる点で有効であると思われる．

表3　システムAの制御パラメータ

| ステーション名 | ジョブ種別＼項目 | | グループ制御: グループ名 | グループ内多重度 | CORE占有領域 | 優先度: (1)グループ間優先度 | ジョブ開始優先度 |
|---|---|---|---|---|---|---|---|
| センタ | 普通 | *A* | TSG | 2 | K語 100 | 2 | 2 |
| リモートバッチ・ステーション | 急行 | *D* | EPG | 2 | 90 | 3 | 1 |
| | 普通 | *L* / *C* | TSG | 2 | 100 | 2 | 1 |
| | 長時間 | *Q* / *P* | LNG | 1 | 90 | 1 | 1 |

(1K語=1,024語)

(1) グループ間優先度の数字は，グループ名登録テーブルの先頭からの順序を示しており，大きい数字の順に登録してある．

## 5. 運用システム

**(1) システムA**

本システムは主として，急行，普通ジョブに注目し急行，普通ジョブ間の queue balance をよくすることを目的とする．この際 CPU 使用効率をいちじるしく悪化させないことを前提とする．CORE 占有領域の制限については本方式の目的をスムーズに行うため急行ジョブについてだけ制限する．センタとリモートバッチ・ステーションの優先度については，センタ・クローズドジョブを優先させた．システムAの制御パラメータは表3の通りである．このシステムを用いた場合の使用実態の測定例を表4に示す．（ここに使用したジョブ・ミックスは急行ジョブ，普通ジョブよりなり，その特性は図4の特性分布の平均的特性に近いものである．）CPU 使用効率は83.3%であって，この種の計算機システムとしては十分高い値であると思われる．このようにして決定された制御パラメータに長時間ジョブを追加した方法で平常期に使用可能であることが確かめられた．この場合の長時間ジョブの測定例を表5に示す．CPU 時間と MEMORY 時間の比は1.9以下であり，これは長時間ジョブの性格から言って十分使用にたえるものである．

表5　システムAを用いた場合の長時間ジョブの測定結果

| 長時間ジョブ名＼項目 | CPU 時間 (*C*) | MEMORY 時間 (*M*) | *M*/*C* | CORE 占有領域 (実行) |
|---|---|---|---|---|
| $P_1$ | 30 m 21 s | 37 m 36 s | 1.238 | 63K 語 |
| $P_2$ | 31 m 57 s | 33 m 43 s | 1.055 | 74 〃 |
| $P_3$ | 23 m 00 s | 43 m 31 s | 1.892 | 71 〃 |
| $P_4$ | 26 m 59 s | 48 m 46 s | 1.807 | 71 〃 |

次に，システムAの場合についてジョブ・スケジューリングの詳細を図6によって説明する．

〔i〕 待ち行列の構成

入力されたジョブはグループ毎に別個の入力待ち行列

表4　システムAを用いた場合の測定結果

| 処理件数 | CPU 時間の合計 (*A*) | (1) ジョブ処理所要時間 (*T*) | (2) $P_{CPU\text{-}U}=A/T$ | 1件当りの平均値: CPU 時間 | MEMORY 時間 | CORE 占有領域 | OUTPUT 枚数 | CHANNEL 使用回数 |
|---|---|---|---|---|---|---|---|---|
| 45件 | 1 h 47 m 26 s | 2 h 8 m 54 s | 0.833 | 2 m 23 s | 5 m 12 s | 38K 語 | 21枚 | 486回 |

(1) ジョブ処理所要時間(*T*)として，最小限1つの利用者ジョブが処理されている時間をとるため，ある日の一定時間（約2時間）を限って，データを採集した．

(2) $P_{CPU\text{-}U}$ は CPU 使用効率．

を構成する．本システムでは三つのグループに分類しているので，EPG 待ち行列，TSG 待ち行列，LNG 待ち行列が作られる．ジョブが入力されると，それぞれに属するグループの待ち行列に先入先出 (FIFO) の原則に従って入れられる．ただし，同一グループ内でもジョブ開始優先度の高いジョブの順に配置される．たとえば，TSG の場合「$A$」の優先度=2,「$C$」,「$L$」, の優先度=1 であるから，図のように $A_4$ が入力されると $A$ の行列の最後尾 $L_1$ の前に配置される．そして $C_4$ と $L_5$ は最後尾に並ぶ．図の斜線でかこまれた部分が状態表示機能 $S$ によりコンソール・ディスプレイに現われる．

〔ii〕 ジョブ・ストリームの構成

コンソール・ディスプレイには現在実行中のジョブ ($D_0, A_0, Q_0$) が状態表示機能 $J$ により現れる．いま，$Q_0$ のジョブが終了したとする．この図の場合他の二つのストリームが EPG, TSG であり各グループの多重度は 2 であるから EPG (または TSG) を選ぶことになる．待ち行列を見ると EPG があるから $D_1$ が選ばれて起動を開始する． $A_0$ が終了した時点をみると他の二つのストリームに EPG が走っており EPG の多重度は満たされているので，TSG の筆頭ジョブ $A_1$ を起動する． もし，ストリーム 1, ストリーム 2 が EPG または TSG で EPG, TSG の少なくとも一つの待ち行列が 0 の場合，LNG の筆頭の $Q_1$ を起動する．

〔iii〕 ジョブ・ストリーム上のジョブ・ステップの実行

三つのストリームの中から適宜 CPU または入出力装置 (I/O) を用いての実行制御は，主にスーパバイザが行う．特にジョブ・ステップの実行優先度が各ストリーム共，同一レベルのときは，ダイナミック・ディスパッチング機能が働いて，より効率的に資源の活用が行われる．なお，この時各グループの CORE 占有領域制限が働いて使用可能な範囲内でジョブが実行される．

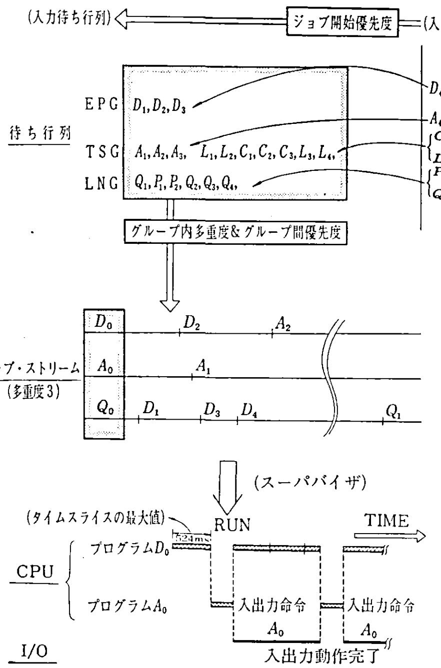



図 6　ジョブ・スケジューリングの説明図

(2) システム B

本システムでは急行，普通，長時間ジョブの queue balance をとりながら，しかも長時間ジョブのターン・アラウンド時間を短くするということが目的である．ここでは急行，普通ジョブが混んでいるという状態を前提としているので，長時間ジョブが急行，普通ジョブを圧迫するという現象に注目しながら諸パラメータの決定を行った．まず，長時間，普通，急行の三つのグループを作る．急行，普通ジョブが混んでいる条件とジョブの数が (急行)＞(普通)＞(長時間) という条件を考えれば優先度は(急行)＞(普通)＞(長時間)，グループ内多重度は(急行)＝(普通)＝(長時間)とするのが妥当ということになる．しかし，この方法で行うと (長時間) は必らず走るので(普通)を圧迫する．すなわち，普通ジョブ待ち行列がたまった状態で長時間ジョブが走るのは不都合である．したがって，普通ジョブの待ち行列がたまっている時は，長時間ジョブのジョブ開始起動を待たせればよいわけである．このような制御を長時間ジョブの interval run と名付けることにする．この方法を徹底して行うにはグループ制御の WAIT パラメータを使用することが考えられる．徹底して行うという意味は急行ジョブ，普通ジョブの待ち行列もなく，その上に実行ジョブもないときにのみ長時間ジョブを走らせることである (これを hard interval run という)．その外に優先度とグループ内多重度を適切に組合せることによって，それに近い現象を生み出すことができる (これを soft interval run

表 6　システム B の制御パラメータ

| ステーション別 | ジョブ種別 | 項目 | グループ制御 グループ名 | グループ制御 グループ内多重度 | グループ制御 CORE 占有領域 | 優先度 グループ間優先度 | 優先度 ジョブ開始優先度 |
|---|---|---|---|---|---|---|---|
| センタ | 普 通 | $A$ | TSG | 1 | K語 100 | 3 | 2 |
| リモートバッチ・ステーション | 急 行 | $D$ | EPG | 2 | 90 | 2 | 1 |
| | 普 通 | $L$ / $C$ | TSG | 1 | 100 | 3 | 1 |
| | 長時間 | $Q$ / $P$ | LNG | 1 | 90 | 1 | 1 |

(1 K 語=1, 024 語)

表 7　システムBを使用した場合の測定結果

| 処理件数 | CPU 時間の合計 (A) | ジョブ処理所要時間 (T) | (1) $P_{CPU-U}$ $=A/T$ | 1件当りの平均値 | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | CPU 時間 | MEMORY 時 間 | CORE 占有領域 | OUTPUT 枚 数 | CHANNEL 使用回数 |
| 46件 内 (LNG: 2件) | 1 h 43 m 56 s | 2 h 2 m | 0.852 | 1 m 7 s | 2 m 51 s | 44 K 語 | 26 枚 | 441 回 |

(1) $P_{CPU-U}$ は CPU 使用効率

という). すなわち,グループ間優先度は(普通)>(急行)>(長時間),グループ内多重度は(急行)>(普通)≧(長時間) とすればよい. 本システムでは soft interval run を採用した. こうして決められた制御パラメータは, 表6の通りである. 本システムを使用した場合の使用状態の測定例を表7に示す. 表6の「1件当りの平均値」は長時間ジョブを除いた 44 件のジョブの平均特性を表わしている. queue balance, ターン・アラウンド時間, CPU 使用効率からみて, システムBは初期の目的を達成していると判断される.

## 6. おわりに

本運用方式は当面FACOM 230-55機の運用を円滑に行うため必要な点を検討したもので, 細部にわたっては今後もジョブの動向および計算機利用者の意向も含めてレベルアップをはかるつもりである.なお,今後FACOM 230-55 機にはパフォーマンス・データ・ロガ (システム資源の使用状況に関する詳細なデータ採集機能) が追加される予定なので, これより詳細なデータ解析が可能になることが期待される. 最後に, 本システムの検討および実験に際し協力された藤田講師をはじめ,田端助手, 原技官, 柏本技官(現在宇宙開発事業団), 特にマクロの作成作業を担当された鈴木技官, また技術的問題について助言をいただいた富士通株式会社の井坂, 橋本両氏に対し感謝の意を表す.

## 付　録

### 1. 運用システム開発経過

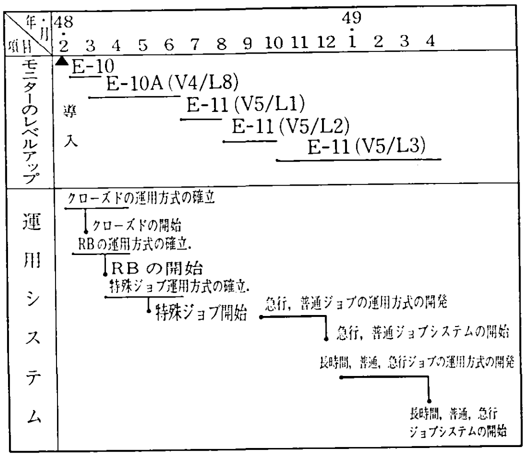



図 7　運用システム開発経過

### 2. FACOM 230-55 機システム構成

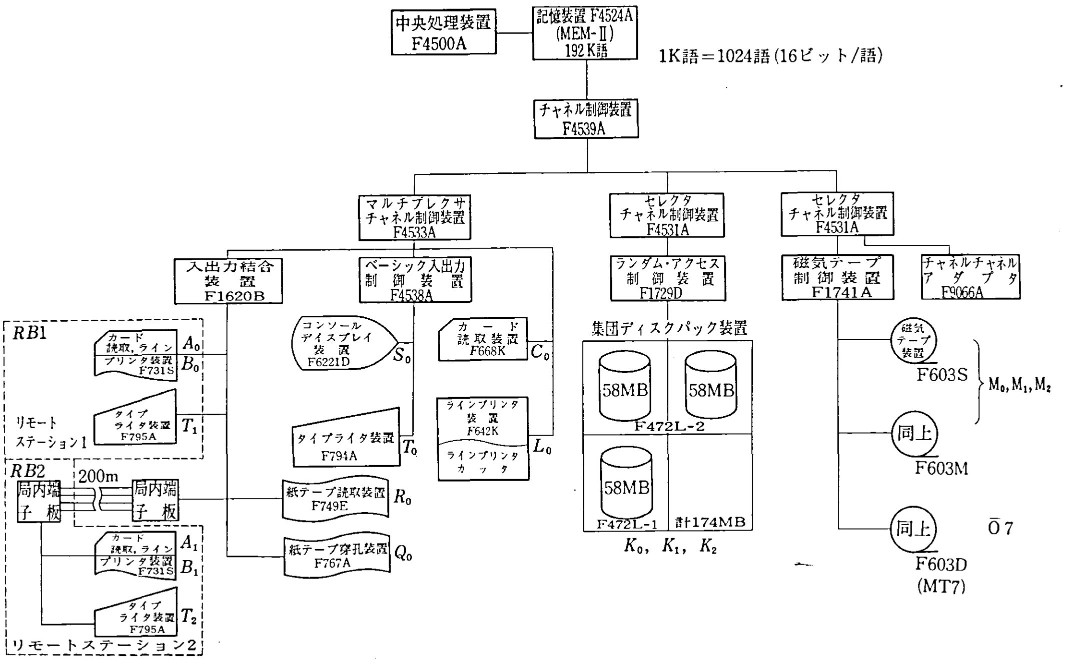
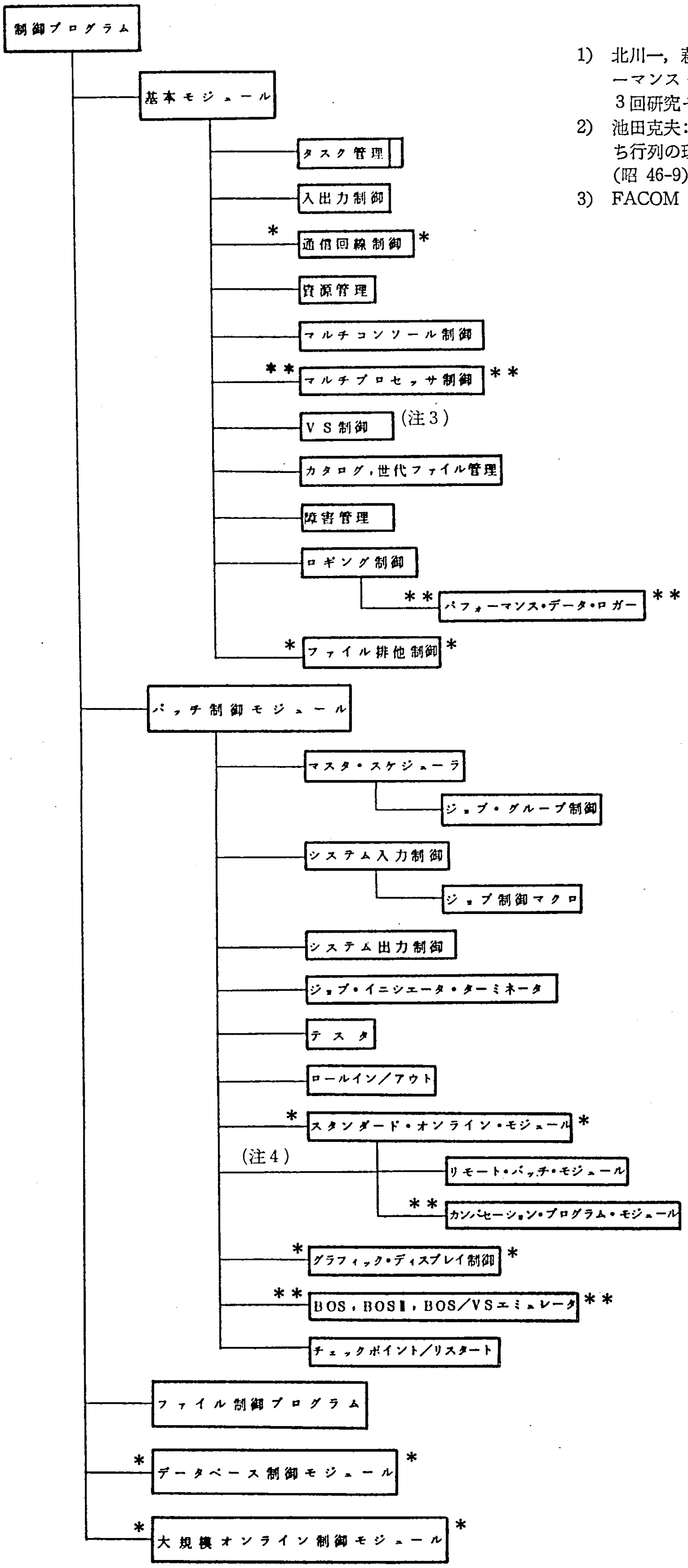



図 8　ハードウェア構成図




注 1) * は現在の本所システムに，くみ込まれていないもの

注 2) **は今後のレベルアップで追加される予定の機能

注 3) FACOM 230-55 には適用されない

注 4) リモート・バッチ・モジュールは単独でくみこまれている

図9　制御プログラムの構成

## 参考文献


1) 北川一，萩原　宏：オペレーティングシステムのパフォーマンス・モニタリング，(京都大学大型計算機センタ第3回研究セミナ報告) (昭 46-12) p. 4〜p. 22.

2) 池田克夫：大型電子計算機システムの効率測定と循環待ち行列の理論による解析，情報処理，Vol. 12, No. 9, (昭 46-9) p. 568〜p. 576.

3) FACOM 230, OS II 解説 (昭 48-3) 富士通.


(1974年6月28日受理)