

COMTRAC-H 運転整理システムの性能評価†

吉澤 康文† 名内 泰蔵‡‡
 長谷川 豊‡‡‡ 稲田 伸一‡‡‡‡

東海道・山陽新幹線の列車運行管理システムである COMTRAC (COMputer aided TRAffic Control)-H は 1975 年 5 月より稼動を開始している。このシステム構成は、進路制御の機能を担う制御用計算機と運転整理の機能を分担する汎用計算機が階層的に結合されたものである。本報告では、列車ダイヤに乱れが生じたときに、人間と機械の会話により迅速にダイヤを收拾する運転整理システムの性能評価をシステム開発の前段階で行い、設計に反映させた例について述べる。

運転整理システムの特徴は、列車に乱れが生じた際に成される指令行為が計算機に急激な負荷をもたらす点にある。したがって、システム設計にあたっては、指令員の投入したコマンドに対する応答時間とスループットを共に確保する処理方式を確立する必要がある。

この理由から、運転整理システムのシミュレーション・モデルを開発し、想定した入力負荷に対する特性を分析した。この分析結果から、応答時間に占める主要因は、(1)計算機間の交信、(2)異なる優先順位を持つタスク間の交信、そして(3)ファイルの競合にあることが判明した。

以上から、制御用計算機と汎用計算機間の交信周期の最適化を計ること、および、優先度の高いタスクの処理を分割することを提案し、システム設計に反映させた。この結果、運転整理システムは充分な応答時間とスループットを満足するシステムとして現在稼動している。

1. ま え が き

コマンド・アンド・コントロール・システムを実現する計算機システムでは、プロセスの制御を司る制御用計算機と、スケジュールの作成ならびに管理を担う上位計算機が階層的に結合される場合が多い。このようなシステムにおける上位計算機では、プロセスの将来を常に予測し、事前に異常の検出を行い、その対策を迅速に講ずる機能が必要である。したがって、プロセスの異常を予知した際に、計算機システムは、指令員のコマンド入力に対して充分な性能を維持し、本来の機能を果たす必要がある。この理由から、システムを建設する場合、事前にシステムの性能評価を行うことが極めて重要となる。

COMTRAC (Computer aided traffic control system)-H は、先に述べたコマンド・アンド・コントロール・システムの典型的な例であり、東京一博多を結ぶ東海道・山陽新幹線の列車運行管理を行う。このシ

ステムでは、列車の運行がプロセスに相当し、列車の進路を制御する制御用計算機と、列車の運行を予測し、列車の乱れを事前に予知し、その対策を指令員と行う汎用計算機とが階層的に結合されている。図 1 は、COMTRAC-H の機能の概略であり、このなかで、運転整理機能の充実は本システム開発の主眼である。

人間と複数台の計算機によりプロセスを制御する COMTRAC-H のようなシステムでは、以下の問題を解決する必要がある。

- (1) プロセッサ間の交信周期の最適化。
 - (2) プロセッサ間で保有するファイルの一致を行うこと。
 - (3) 指令員に対するコマンドの応答時間の保障。
- 本研究では、これらの性能に関する問題点が、COM-

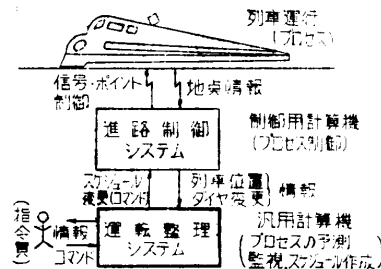


図 1 COMTRAC-H システムの機能概略
 Fig. 1 Schematic diagram of COMTRAC-H system.

† Performance Evaluation for COMTRAC-H Traffic Control System by YASUFUMI YOSHIZAWA (Systems Development Laboratory, Hitachi, Ltd.) TAIZO NAUCHI (Software Works, Hitachi, Ltd.), YUTAKA HASEGAWA, and NOBUKAZU INADA (Railways Technical Research Institute, Japanese National Railways).

‡ (株)日立製作所システム開発研究所

‡‡ (株)日立製作所ソフトウェア工場

‡‡‡ 日本国有鉄道鉄道技術研究所

TRAC-H において、どのように現れ、また、どのように解決すべきかを探ることが目的であり、その手段として、COMTRAC-H の運転整理システムを中心としたシミュレーション・モデルを開発した。ここでは、COMTRAC-H の概略、シミュレーションによる解析結果、ならびに、システム設計に反映した点について述べる。

2. COMTRAC-H の概略

COMTRAC-H では、列車の進路を制御する計算機を PRC(Programmed Route Control) 系¹⁾と呼び、運転整理を主として担う上位計算機を(汎用計算機を用いるため) EDP 系と呼んでいる。図 2 に、EDP 系を中心とした情報と制御の流れを示す。EDP 系と PRC 系は 1 分ごとに、(1)列車位置変化情報、(2)ダイヤ変更に伴うファイル変更情報、を主とする情報交換を行っている。EDP 系は、この情報に基づき、現在の列車運行の状況を把握し、列車の遅延情報を各所の端末へ定期的に出力する。また、PRC 系で変更されたダイヤを、EDP 系のダイヤ・ファイルに反映させ、その内容をダイヤ変更の指令伝達として端末に送出する。一方、EDP 系では、10 分周期に、現在運行中の列車の 2.5 時間先の状態を予想し、その結果をダイヤ・モニタ・プログラムを通して、グラフィック・ディスプレイ(以後、GD と省略する)に表示する。同時に、警報モニタ・プログラムでは、列車の運行を修正する必要があるか否かを調べ、GD に表示し、指令員に警告する。

指令員は以上の情報を GD から受け、ダイヤの修正を行う必要性を判断する。ダイヤの大幅な修正を必要とする場合は、修正情報や事故に関する情報を GD から入力し、その状況下でダイヤの予測を行う。この

結果は、ダイヤ・モニタおよび変更点抽出プログラムで GD に表示される。指令員は、ダイヤ予測の結果を判断し、満足な結果が得られるまでダイヤの修正作業を繰返す。修正作業の結果は、ダイヤ・ファイルの更新を行うために、変更指示、実行指示、ファイル変更管理プログラムを起動したのちに PRC 系に送られる。PRC 系は、この要求を受け、PRC 系の持つ列車ダイヤ・ファイルの変更を行い、新たに EDP 系に対し、ファイル変更の要求を行う。つまり、プロセスの制御を担う PRC 系は、常に最新のスケジュール・ファイルを持つことにより列車を制御する必要があるため、列車ダイヤの変更は、PRC 系から行い、それを EDP 系に反映させる方針としている。指令員の提案したダイヤ変更の手続きは、上記のごとく、PRC 系からのファイル変更要求として EDP 系に取込まれ、先に述べたファイル変更と同様に EDP 系にて処理される。

3. 運転整理システムのモデル化

シミュレーション・モデルは、GPSS に似たシミュレーション言語を持ち、CSSII^{2),3)} とほぼ同程度の機能を持つ HCSS (Hitachi Computer System Simulator) により記述した。HCSS は、ハードウェアの構成、性能をパラメータで記述し、入出力装置の動作が内蔵されており、プログラムの動作を記述する言語体系を持つ計算機専用シミュレーション言語である。

以下、HCSS による運転整理システムのモデルの概略を説明する。

3.1 ハードウェア構成

シミュレーション・モデルは、運転整理システムの稼動時における必要最低限のハードウェア構成を想定した。図 3 に構成図を示す。これらの性能の設定は、CPU 処理能力として平均一命令実行時間を 3.6 μ sec. ディスクの平均シーク時間を 67 msec., 転送速度 312 kB/sec., 一方、ドラムの回転速度は 20.6 msec., 転送速度 281 kB/sec. である。また、GD の転送速度は 150 kB/sec., 通信回線の転送速度は 200 bps である。

計算機システムの性能評価では、入出力装置の動作を正確に取扱うことが重要である。このため、ファイルの配置をシミュレーション・モデルに組み込み、そのアクセスの正確さを期した。具体的には、ディスク・ファイルへのシーク、レコードまでの回転待ち、データ転送、等の情報を HCSS 言語で記述した。HCSS では、この情報を基に、現在のディスク・アームの位

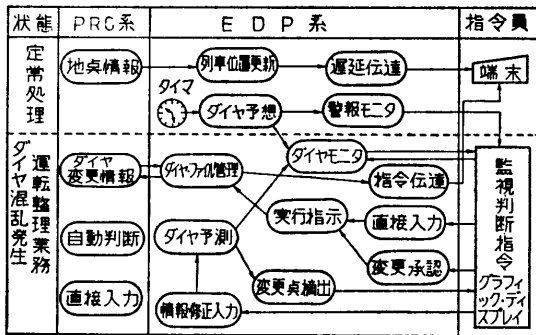


図 2 情報と制御の流れ
Fig. 2 Information and control flows.

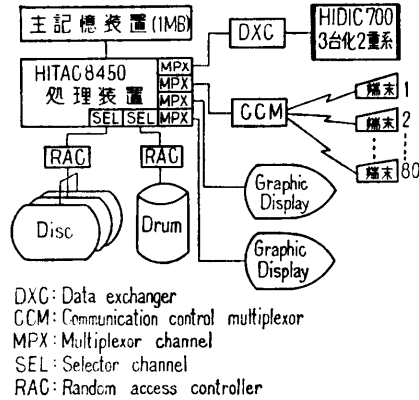


図3 シミュレーション・モデルに反映したハードウェア構成
Fig. 3 Modeled hardware configurations.

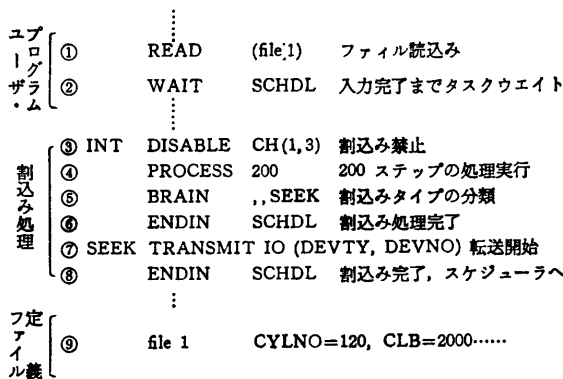


図4 HCSS による I/O 操作の例
Fig. 4 Example of I/O operation by HCSS.

置から目的のシリンダまでのシーク時間を求め、イベントを発生する。転送についても全く同じ動作を行う。これらイベントは割込みであり、コントロールがある特定の HCSS ステートメントに渡る。図4に、HCSS での入出力に関するプログラム例を示す。この例は、ユーザ・プログラムにおいて、①でファイル1を読み込み、その完了を②で待つ。入力オペレーションは、まずファイルの存在する120シリンダまでシークを行い、完了の時点で制御が③に移る。割込み処理では、まず割込み禁止状態にしてから、200ステップの実行を行い④、割込みタイプで分岐を行う。最初は、シーク割込みが発生し、⑦に分岐する。ここで、データの転送を行う命令を発行し、⑧で割込み処理を完了させる。このデータ転送完了割込みは、シークと同様に③に入り、⑤により割込みタイプが分類され、制御が⑥に移る。ここで、入力を要求したタスクは実行可能となり②のウエイトは解除される。

表1 OS のモデル化に用意したマクロ命令
Table 1 Macro instructions in the simulated OS model.

	マクロ記号	機能概略
タスク管理	EXTSK	サブタスクの起動を行う
	TTSK	サブタスクの処理完了を親タスクへ通知する
	XITSK	サブタスク完了割込み処理完了通知を行う
	PR	タスクを待ち状態にし、割込みイベントを待つ
	LPOV	オーバレイ・プログラムのローディング要求
データ管理	EXCPW	入出力実行要求
	HOLD	入出力装置の確保
	FREE	入出力装置の解除
通信管理	CWAIT	回線からの入出力イベントを待つ
	CREAD	端末からのメッセージ入力要求
	CWRT	端末へのメッセージ出力要求
タイマ	STIME	実時間割込み要求
IPC関係	PCSET	IPC マクロ使用の宣言を行う
	PCRD	他ジョブからの情報の読み込み
	PCWRT	他ジョブへの情報伝送要求
	PCWT	PCWRT の完了イベント待ち
ゲックラフ関係	GWRT	GD に対する図形データの出力要求
	GRD	GD から入力データの取込みを行う
	GPR	GD からの入力アテンションを待つ
	GWT	GWRT の完了イベントを待つ

図4に示した HCSS ステートメントで、以下の OS ならびに運転整理のプログラムの記述を行った。HCSS は、上記のごとく、入出力割込みやタイマの割込み機能を備えた計算機専用シミュレーション言語である。

3.2 オペレーティング・システムのモデル

OS は、アプリケーション・プログラムの制御方式を左右する重要な位置を占める。また、システム建設時には、OS の機能や処理ステップ数が正確に把握できた。そこで、運転整理システムで使用する OS の主機能を中心としたモデル化を行った。表1に、運転整理システムが使用する主な OS の機能であるマクロと、その機能の概略を示す。HCSS で運転整理システムのプログラムを記述する場合は、表1のマクロをそのまま使用できるようにし、プログラムの記述が容易に行えるようにした。

表1における IPC (Inter-Program Communication) 関係マクロとは、ジョブ間の交信手段である。

3.3 運転整理プログラムのモデル

運転整理システムのシミュレーション・モデルは、システムの建設と同時に開始したため、各プログラムの正確な処理ステップ数の把握は困難であり、過去の開発経験 (COMTRAC-フェーズ⁴⁾1) を基に推定を行う方法を取った。しかし、入出力の回数や、オーバレイ構造の設計、ならびに、プログラム間の制御方式に

については正確な情報を得ることができ、運転整理処理プログラムの動作ならびに、OSの動作をシミュレーションモデルに反映させた。図5は運転整理システムのプログラム構成の概念図であり、CMP, GMP, TMPと名付けたジョブで構成されている。これらのジョブは、互いにIPCというスーパーバイザ・コールで交信を行い有機的に制御情報の交換を行っている。ジョブ間のデータ交信はファイルを経由して行っている。例えば、CMPがPRC系から受取った情報をファイルに格納し、IPCによりその処理をGMPに指令する。GMPはCMPからの起動で要求された処理を行うために、ファイルから情報を読み込む。つまり、図5における実線は制御情報の流れであり、二重線は情報の流れを意味する。

次に、運転整理システムにおける処理のモデルは図6に示すようなネットワーク・モデルである。システムへの入力は、指令員操作によるグラフィック・ディスプレイ、PRC系からの入力、端末からの問合せ、そして定期的に起動されるタイマ入力処理である。これらは、三つのジョブのいずれかで処理されるため

に、ジョブの待ち行列に入る。これをジョブ（あるいはタスク）待ちと呼ぶ。ジョブは一台のCPUで、発生した入力を処理を行うために、CPUの待ち行列に入る。一般に、処理の過程でファイル入出力が行われるため、その場合はファイルの待ち行列に入り処理される。

運転整理システムでは、五つのタスク（GMPのみ二個のサブタスクを持つ）が互いに交信を行い処理を進めるために、ジョブの待ち行列には、外部からの割込み以外に、内部的に発生する入力がある。各ジョブは異なった優先度を持っており、ジョブ間の交信が円滑に行われるか否かが、このようなジョブ・モデルにおいては問題になる。

3.3.1 CMP (Communication Management Program)

PRC系および端末との交信が主機能である。CMPは単一タスクで動作し、以下の処理を行う。

(1) PRC系との交信: 通常は1分周期でPRC系にポーリングを行い、PRC系から情報を得る。PRC系への情報送信は任意の時点で行う。

(2) 端末入出力管理: 列車遅延、指令の伝達と受領の確認に関するメッセージの送受信管理を行う。

(3) GMPとの交信: GMPは運転整理案の結果生じたダイヤ変更情報をPRC系に送信することをCMPに通知する。逆に、PRC系の要求するファイル変更は、CMPによりGMPに知らされる。

(4) タイマによる起動: PRC系へのポーリング、列車遅延情報作成プログラムの起動を行う。

CMPのシミュレーション・モデルでは、上記の処理は、すべてOSモデルからの割込みイベントにより起動される。

3.3.2 GMP (Graphic Management Program)

GMPは、指令員の列車運行監視、列車乱れの収束を行う業務プログラムの起動を含む、全運転整理機能を制御する。つまり、EDP系における中核となるジョブである。GMPは、2台のGDの制御を行い、複数のコマンドの処理を並列に処理する必要があるため、マルチ・タスク構成となっている。ここで、メイン・タスクは指令員からのコマンドの受付と、その処理の実行を管理し、2つのサブタスクは、入力画面の処理と、業務プログラム処理を行う。

指令員の起動する業務プログラムは、一般に複数個のプログラムで実行される。これらのプログラムは、使用する計算機リソースの特性から、同一タスクで実

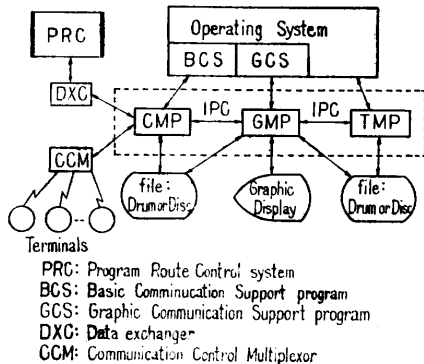


図5 シミュレーション・モデルに反映したソフトウェア構成

Fig. 5 Modeled software configurations.

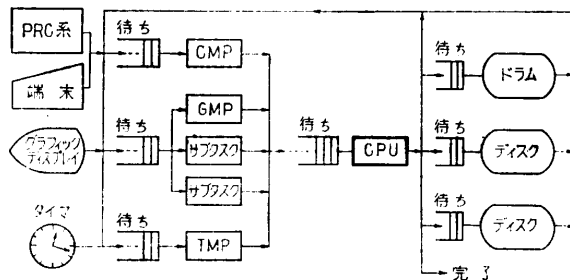


図6 COMTRAC-Hシステムの待ち行列モデル
Fig. 6 Queuing network model of COMTRAC-H system.

行すると、処理能力の低下を招くため、機能分担されたタスクのもとで処理される。この理由から、各々のプログラムの処理単位をジョブ・ステップと呼ぶ。

シミュレーション・モデルでは、GD からの入力とジョブ・ステップの制御を以下の通り処理する。

(1) 指令員を1台のプロセッサでシミュレートし、コマンドやパラメータの入力をタイマ割込みを利用し、EDP 系のプロセッサに送信する方法とした。GMP はこの割込みで、指定されたジョブ・ステップの列を設定する。

(2) GMP は、ジョブ・ステップの列から先頭のプログラムを実行するタスクを求め、そのタスクの起動を行う。該当するタスクが実行中の場合は、そのタスクの待ち行列にプログラムを並べる。

(3) 実行すべきプログラムが CMP や TMP の場合には、該当するジョブに起動用マクロを出す。

(4) サブタスクの終了時に、GMP メイン・タスクはそのサブタスクの待ち行列の処理を行う。

(5) 指令員のパラメータ入力は、平均2秒の指数分布に従うものとした。

運転整理システムでは、指令員からの入力以外に、次の2つの業務が起動される。

(6) 列車ダイヤ予想プログラムの10分周期の起動。

(7) PRC 系からのダイヤ変更要求時に、CMP から通知を受け、ジョブ・ステップの設定を行う。

3.3.3 TMP (Traffic Management Program)

TMP はシングル・タスクで、GMP の起動要求によりプログラムの実行を行う。TMP で実行されるプログラムは、列車ダイヤの予想、予測、ならびにダイヤ・ファイル変更プログラムである。

3.4 PRC 系モデル

運転整理業務には、PRC 系との通信を含むコマンドが存在する。したがって、プロセッサ(図3における HITAC 8450 と HIDIC 700)間の通信が、コマンドの処理時間に占める割合を求めるために、PRC 系のモデル化を行った。その機能は以下に示す。

(1) 列車位置変化情報を EDP 系に送出する。

(2) ダイヤ変更情報を発生させる。発生間隔は指数分布に従わせる。

(3) 指令員の入力したダイヤ変更を EDP 系から受取り、PRC 系で処理した後に、その結果として、ファイル変更情報を EDP 系に送出する。

(4) PRC 系のデュアル運転方式である、3秒周

期の入出力動作。EDP との通信は、3秒周期で行う。

4. シミュレーション結果と考察

HCSS によるシミュレーション・モデルの規模は、約3,500ステップである。シミュレーションは、モデル内のクロックが1時間に至るまで行った。

4.1 入力トラフィックの設定

COMTRAC-I (東京—岡山) の使用経験を基にし、COMTRAC-H で対処すべき最大の列車の乱れを推定した。表2はその結果であり、1時間あたりの最大負荷量としてモデルの入力パラメータとして使用した。

4.2 計算機資源の利用率

(1) CPU 負荷とタスク稼働率

CPU の負荷とタスクの稼働率を表3に示す。タスクの稼働率は、OS からみてタスクが何らかのプログラムを実行している割合である。

表3から、CMP はタスクの稼働率が非常に高い。CMP の稼働率の大部分は、プログラムのオーバーレ

表2 想定した最大の入力トラフィック量
Table 2 Expected maximum transactions.

業 務	件数	GD操作数/件	走 動 条 件
直接入力	入力操作	17	指令員
	モニタ操作	6	
	実行指示	2	
整理案ダイヤ	情報・修正入力	53	指令員
	モニタ操作	6	
	変更承認	6	
	実行指示 予 測	12 10	
予 想	6	0	タイマ
PRC ファイル変更	65	0	PRC 要求
列車位置更新	60	0	タイマ
遅延伝達	72	0	タイマ
地点情報取込	60	0	タイマ

表3 タスクと CPU の利用率
Table 3 Tasks and CPU utilizations.

		タスク稼働率	C P U
C M P		71%	1.7%
G M P	メイン・タスク	18%	0.2%
	入力処理タスク	7%	
	出力処理タスク	8%	
T M P		57%	50.6%
オペレーティング・システム			6.4%
合 計			58.9%

イならびに端末メッセージの入出力に必要なディスク入出力で占められる。

CMP は、最も高い優先度で実行されるので、その稼働率が高くなると、PRC 系との交信を含むコマンドの応答性が悪化する原因となる。つまり、GMP と CMP の交信時に、GMP の要求が待たされる。

CPU 利用率は全体で約 60% である。その大部分は、ダイヤの予想、予測を行う TMP で占られる。

4.3 業務の処理時間と GD の応答時間

EDP 系では、列車の運行を監視するため常に最新の列車ダイヤ・ファイルを保持する必要がある。列車ダイヤの変更は、PRC 系より任意の時点で発生すると思われる。したがって、EDP 系はダイヤの変更が通知された時点で、両プロセッサのファイルの一致を計るため、迅速に処理を行う必要がある。このために、EDP 系では、ダイヤ・ファイルの更新時には、他の業務プログラムを中断する制御方式としている。

上記のダイヤ・ファイル競合が、各コマンドの処理時間に及ぼす影響を調べるために、代表的なコマンドである、ダイヤ・モニタの処理時間の分布を図7に示した。図7より、処理時間 40 秒以上のものが、ファイル更新時の影響を受けたもので、全体の 60% に相当している。

一方、優先的に実行されるファイル変更の処理時間の分布は図8に示すとおりである。ファイル変更は、

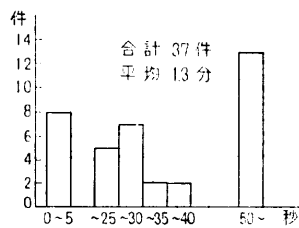


図7 列車ダイヤの作成と表示に要する処理時間の分布
Fig. 7 Processing time distribution for generating and displaying the train diagram images.

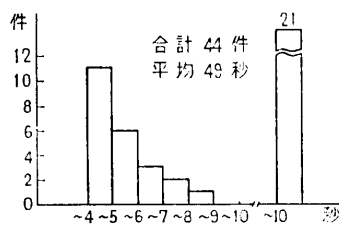


図8 列車ダイヤの更新処理に要する処理時間の分布
Fig. 8 Processing time distribution for updating the train diagram.

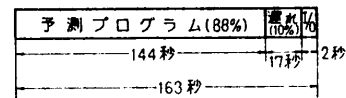


図9 列車ダイヤの予測処理に要する処理時間の内訳
Fig. 9 Component of processing time for the train diagram projection.

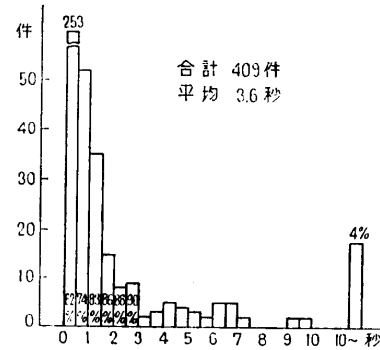


図10 グラフィック・ディスプレイを介して行う会話処理の応答時間の分布
Fig. 10 Distributions of response times for the graphic man-machine interaction.

TMP タスクで処理されるため、予測プログラムとタスクの競合が発生する。図8における 10 秒以上を要した処理は、TMP タスクの競合が原因である。

次に、ファイル変更と TMP を競合するダイヤ予測プログラムの処理時間の分析を行った。図9がその結果である。当初、TMP は最も低い優先度で処理されるので、処理の遅れが危惧されたが、他タスクの CPU 使用率が低いため、約 10% 程度の遅れである。

以上、コマンドの処理時間と、それに与える影響を考察したが、運転整理作業を進める過程で、GD の応答性が操作性の観点から重要である。図10に、GD の応答時間の分布を示す。応答時間は、指令員が GD より入力した時点から、その入力に対する最初の画面出力が現れるまでの時間とした。

平均応答時間は 3.6 秒であり、全体の 90% が 3 秒以内に回答されている。応答時間が 10 秒以上のものが約 4% あるが、これは処理の長いコマンドの投入ならびに、先に述べたファイルの競合で処理が中断したものを含むものである。

4.4 タスク間交信時間

運転整理の結果、ダイヤの修正は PRC 系に送られる。この場合、GMP タスクは PRC 系との交信を司る CMP タスクに、修正情報の転送要求を行う指示を行うが、CMP タスクの処理優先度が高いこと、なら

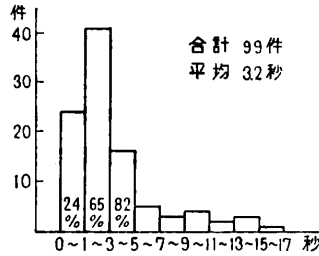


図 11 GMP と CMP タスクの内部通信時間の分布
Fig. 11 Distribution of inter-process communication time between GMP and CMP.

びに、CMP のタスク稼働率が高い、等の影響を受けるため、タスク間の通信時間が問題となる。

そこで、GMP と CMP の通信時間の分布を取得し、図 11 に示した。図 11 より、通信の平均時間は 3.2 秒である。通信時間の長くなる原因は、上記に示した CMP の優先度、タスク稼働率の他に、CMP タスクの処理するイベントである端末入出力が多数存在し、GMP の要求が待たされるためである。

4.5 プロセッサ間通信時間

運転整理の結果発生したダイヤの修正情報は PRC 系に転送されるが、その応答を得るために、EDP 系は一定周期 p で問合せを行う。したがって、この周期 p が、コマンドの処理時間に影響を与える。そこで、両プロセッサ間の通信の特長と、最適な p を求める。

PRC 系はデュアル運転方式の都合上、一定周期 c ごとに外部との通信を行う。図 12 に両系の通信手順を示す。EDP 系の情報転送は、PRC 系とは独立に行われるため、その要求は w だけ待たされて、A 点で受けられる。したがって、 w は $(0, c)$ 間の任意の値を取る。PRC 系は、受け取った要求を処理し、その応答を B 点以降に出力可能とするために、EDP 系からの有効な問合せ要求が来るまでの時間 l が通信時の損失時間となる。

そこで、損失時間 l と問合せの周期 p の関係を求める。時点 B 以降に有効となった問合せが、第 n_0 番であるとすると、 n_0 は、 $n_0 p > w + c$ 、を満たす最小の n である。すると、損失時間は、 $l = n_0 p - (w + c)$ 、で与えられる。ここで、 w は一様分布に従う。



図 12 EDP 系と PRC 系の情報伝達手順
Fig. 12 Data communication procedure between EDP and PRC system.

表 4 プロセッサ間通信の平均損失時間
Table 4 Mean loss time at processor communication.

周期 p	問合せ回数 n_0			損失時間 期待値 l (秒)
	$0 < w \leq 1$	$1 < w \leq 2$	$2 < w \leq 3$	
1	4	5	6	0.5
2	2	3	3	0.5
3	1	2	2	1.5
4	1	2	2	1.8
5	1	1	2	0.83
6	1	1	1	1.5
7	1	1	1	2.5
8	1	1	1	3.5
9	1	1	1	4.5
10	1	1	1	5.5
11	1	1	1	6.5

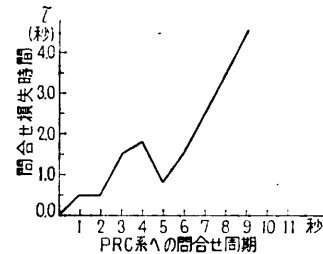


図 13 ポーリング周期とポーリング成功までの平均待ち時間の関係
Fig. 13 Relationship between polling cycle time and expected waiting time for the effective polling.

COMTRAC-H では、 $c=3$ 秒である。ここで、PRC 系の処理が c 以内に完了した場合の l の期待値 \hat{l} を求める。この場合、 n_0 は上記の制約のもとで、 p ならびに w によって変化するので、この様子を表 4 に示す。 w は一様分布に従うので、 w の取り得る値の範囲は各々 3 分の 1 であるため、 l が計算できる。図 13 に、 \hat{l} と問合せ周期 p の関係を示した。当初 p の設定は 10 秒としていたため、平均損失時間 \hat{l} は 5.5 秒であり、プロセッサ間の通信に多くの時間を要していた。

図 13 より、問合せの周期は小さい程平均損失時間 \hat{l} は小さい。しかし、 p を小さくすることは両 CPU の割込み頻度が高くなり、リアル・タイム性が厳しく要求される PRC 系では大きな負荷となる。この理由から、 $p=5$ 秒が最適な周期であるため、現在はこの値を採用している。

4.6 結果の考察と性能向上策

以上のシミュレーション結果より、運転整理システムのコマンド処理ならびに GD の応答時間に大きな

影響を与える要因は次のとおりである。

(1) PRC 系および端末との交信を担う CMP タスクの稼働率が、71% と高く、GMP タスクとの交信時間が長くなる。

(2) EDP 系と PRC 系の情報交換の周期が当初の設計では 10 秒であり、PRC 系の応答を含むコマンドの処理時間に大きな割合を占めている。

(3) コマンド実行中に、PRC 系よりダイヤの変更が発生すると、その処理のために、コマンドの処理が中断される。

上記3項目が運転整理システムの性能を低下させる主要因であり、その改善策として以下の項目を設計に反映した。

(a) CMP を長く占有するプログラムでは、処理を分割し、処理の単位を完了した際に、イベントの有無を調べ、緊急度の高いイベントの順に処理を実行する。この擬似マルチ・タスク方式により、CMP のイベント処理のレスポンスを向上させる。

(b) PRC 系の応答を必要とする場合の交信周期は前節に示したように、コマンド処理に大きな影響を与える。したがって、PRC 系での処理が1サイクル(3秒)以内で完了するものが多いこと、ならびに、両プロセッサのオーバヘッドを考慮して、プロセッサ間の交信を5秒とし、最適な交信周期を得た。

5. む す び

コマンド・アンド・コントロール・システムとして代表的な COMTRAC-H を、シミュレーションにより解析し、予想される最大負荷時におけるシステムの性能特性を明らかにした。その結果、プロセスの監視、および、異常時のスケジュール変更を行う、運転整理システムの処理性能に与える主要因は、(1)タスク稼働率のアンバランス、(2)プロセッサ間交信周期、そして、(3)ファイルの競合、にある。

以上の解析結果から、性能を向上させる対策として、(a)端末や PRC 系との交信を担う CMP (稼働

率が最も高い) タスクを擬似マルチ・タスク化する。

(b)両プロセッサ間の交信周期の最適化を計る、を考案し、システム設計に反映した。ファイルの競合によるコマンドの処理の遅延については、多大の修正を伴うため、次期システムでの課題として残った。

COMTRAC-H 運転整理システムの性能評価は、システム開発と並行して行われたために、プログラムの処理ステップ数の仮定に誤りがあった。特に、ダイヤ予測の処理ステップ数は、実際の3倍程度を仮定していた。しかし、システムの制御方式のモデルは、ほぼ正確に行われており、事前にシステムのボトルネックを抽出し、その対応策を施したことにより、現システムは使用に耐える性能を示し、稼働中である。

最後に、シミュレーション・モデルならびにその結果について、国鉄本社、鉄道技術研究所の方々に検討をしていただき、また、シミュレーション・モデルの開発には、(株)日立製作所ソフトウェア工場、システム技術本部の方々に協力していただいた。これらの方々に心から感謝の意を表します。

参 考 文 献

- 1) Ishihara, Y., Miyakawa, N., Tanaka, S., Ohno, Y., and Wakabayashi, K.: COMTRAC-A Computer aided traffic control system for Shinkansen; First USA-JAPAN Computer Conference, 1972, Proc. pp. 466-471.
- 2) Seaman, P. H. and Soucy, R. C.: Simulating operating systems: IBM Systems Journal, Vol. 8, No. 4, pp. 264-279 (1969).
- 3) IBM: Computer System Simulator II (CSS/II) Program Description and Operations Manual (SH 20-0875-1) (March 1971).
- 4) Ozeki, M., Naganuma, S., and Inada, N.: Railway Traffic Control System; IFIP Congress 74, 1974, Proc. pp. 802-806.

(昭和52年8月29日受付)

(昭和53年10月11日採録)