# 連続・離散系両用シミュレーションプログラムの開発*

赤塚武昭** 吉田信之** 加藤明憲**

**Abstract**

There are two types of the simulation languages at present, namely, the continuous system type and the discrete system type.

The former is used in the fields of chemistry, physics and engineering. The most well-known one is the "CSMP". The other type is used in operations research areas. The "GPSS" and the "SIMSCRIPT" are the most typical. Nevertheless the actual problems which we must solve are usually interrelated with the discrete and the continuous systems.

Therefore the authors have developed the new simulation program which can handle these complex systems. This program is the powerful tool to analyze the actual problems by means of simulation.

## 1. はじめに

シミュレーションの目的は時間の経過とともにシステムの状態がいかに変化していくかを調べることである．シミュレータは時間を進めながらシステムモデルの状態をもとめるものであるが，この時間の進め方によりそれぞれ連続系，離散系シミュレータに大別されている．前者はシステムの状態変数を時間に関して連続的に変化させる．この範疇のプログラムとして "CSMP" があり，物理，化学，工学の諸問題に現われる微分方程式系のシミュレーションに用いられている．後者の離散系シミュレータでは変化の生じる時刻のみに関してシミュレーションを進めていく．この種のシミュレータには "GPSS", "SIMSCRIPT" などがあり主として O.R. の分野において順序動作，論理判断で構成されるシステム解析に適用されてきた．現実の問題として，われわれがシミュレーション実行の対象とする系は単純に連続系，離散系が独立して，構成されているものではなく，各々の系が複雑に交絡しあっている．生産システムの例を Fig. 1 に示すようにプロセス内部におこる物理・化学現象を順序関係，論理判断やスケジュールにもとづいて運転員が制御し生産活動を行なっている．バッチ反応プロセスをみると原料の仕込みなどの作業員の動作は離散系の事象であり，反応のような化学現象は連続的な現象として取扱うのが自然である．Fig. 2 ではグラフで示した部分が連続系に相当し，下部の作業計画が離散系の事象を表わしている．このような一般的な問題をとり扱う場合，従来は連続系部分と離散系部分とに分離し，別々にシミュレーションをするか，あるいは一方の系で近似的に

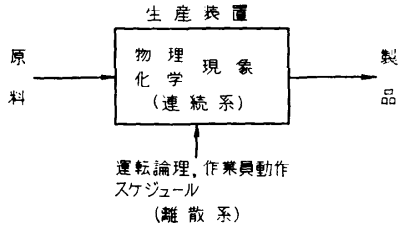



Fig. 1

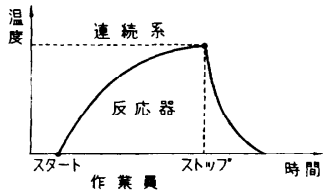



Fig. 2

* Integrated Continuous and Discrete System Simulation Program, by Takeaki Akatsuka, Nobuyuki Yoshida and Akinori Kato (Engineering Labs., Toray Industries, Inc.)

** 東レ株式会社，エンジニアリング研究所

表現するかの方法しかなくシミュレーションの精度がよくなく，モデルを表現するのも困難な場合がしばしば生じていた．筆者らはこれらの欠点を解消し，現状のシステムをより自然な形で精度よく解析するために連続系，離散系両方の機能をインテグレートしたシミュレーションプログラムを開発し，シミュレーションの汎用性をより拡大した．

## 2. 連続系と離散系

連続系は通常微分方程式系のモデルで表現される物理，化学，工学上の諸問題が対象である．したがってこのモデルを用いてシミュレーションを実行するということは微分方程式を「積分」して時間を進めていくことにほかならない．「積分」ということは時間きざみ幅を無限小にして未来の状態を求めていくことである．この場合すべての系の状態変数は時間の進行にしたがって変化していく，しかも連続的に変数が変化するので表現方法としてはグラフを用いるのが理解しやすい．連続系モデルにおいて主役を果す要素は「積分器」*であり，その機能は現在の系の状態を入力とし未来の状態を出力として時間を進行させる．また「積分器」は一変数入力，一変数出力の要素でもある．一方離散系のモデルは各事象の順序関係，スケジュール，論理判断から構成されており一般的な数式表現は困難である．したがってこの場合の時間の進め方は，離散系事象のモデルを用いて未来に発生する事象を予知しうる範囲で求めこれらの事象を発生時刻の順にスケジュール表に書き込み，このスケジュール表により状態を変化させていく．この場合状態の変化が起るのは事象が発生した時点のみであり，一つの事象発生時刻より次の発生時刻までの間の状態は保持されていて変化しないと考える．つまり状態は時間に関して離散的にしか変化していかない．しかもその状態の変化は特定の事象に関した変数のみが変化し，スケジュール表には現在の状態にもとづいて未来を予想した状態およびその発生すべき時刻を書き，所定の時刻になればその状態がスケジュール表より出力され，その状態にもとづいて事象が実行されていくので，スケジュール表は入力・出力ともに多変数系と考えられる．シミュレーションの結果としては，ある時刻におけるシステムの状態を求めることと同時にある期間中における各変数の平均値，標準偏差，最大値，最小値などの統計量が必要なことが多い．離散系のシミュレーションにおいて重要な役割を果すのは「スケジュール表」であって，これは連続系における「積分器」に相当する．システムの記述方法は連続系では微分方程式を含む数式モデルであるので，より直観的にはブロック線図が用いられる．これはシステム要素の相互の関係を示したものである．一方離散系においては計算機のプログラミングに用いるのと同様にフローチャートが便利である．フローチャートはシステム要素間の順序関係を示したものである．この関係を Fig. 3 に示す．

フローチャートで示される離散系はスケジュール表により時間が進行し，ブロック線図で表現される連続系は積分により時間が進む様子を示す．ここで連続系と離散系の特徴をまとめて Table 1 に示す．両者はそれぞれの特性が相異なっているがその相互関係を Fig. 3 上で考察すると連続系のブロック線図における積分のスタート，ストップは離散系の事象により決定され，逆に離散系の事象は連続系の変数の状態により制御される．

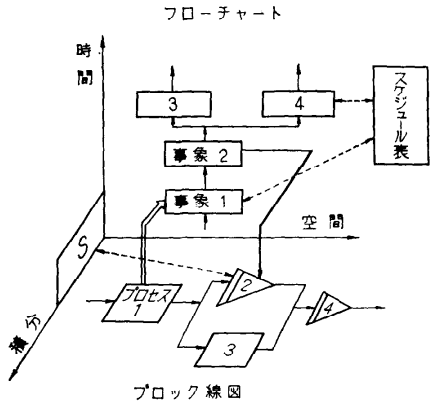



Fig. 3

Table 1

| | 連続系 | 離散系 |
|---|---|---|
| 対象 | 物理,化学,工学 | 管理・人間動作OR |
| モデル | 微分方程式 | 順序論理 |
| 時間要素 | 積分器 | スケジュール表 |
| 出力 | 状態変数変化 | 統計量 |
| 図表現 | ブロック線図 | フローチャート |
| 変数変化 | 同時変化 | 順次変化 |

## 3. 構 成

シミュレーションプログラムの方式としては言語形

* アナログ計算機では多変数入力の積分器も見受けられるがこれは加算器が組合せられたものであり，積分器自身は $x$ を入力とし $x$ を出力とする一変数要素である．

式を用いてモデルを記述していく方法や，固定プログラムを作り，データによりプログラムの流れを制御する方法などがあるが，ここではFORTRANで記述されたサブルーチンによって基本要素群を構成し，ユーザがこれらの要素を組み合せてモデルを構成する方式を採った．シミュレータはこれらのモデルを所定の順序で実行していく．したがって2，3の例外を除けばFORTRANのステートメントを自由に用いることができる．基本要素としては従来の連続系，離散系それぞれの要素のみでは不十分で，連続系と離散系を結合させるための要素が必要になる．例えばある準備作業の終了後反応をスタートさせる場合のように離散系の事象により連続系の積分をスタートさせたり，逆に反応の状態が所定の値になればストップさせて，製品の取り出し作業を実行する場合のように，連続系の状態により離散系の事象を発生させる必要がある．以上のように要素としては離散系用，連続系用，結合用を備えている．これらの要素を用いれば系のモデルは容易に記述できるが，シミュレーションを実行する場合は各要素間の時間，順序関係が複雑であり，ユーザがこれらの時間，順序関係を意識せずに自動的にシミュレーションを実行するために，ユーザの記述したモデルをコントロールしながら時間を進めていくタイミングルーチンが必要である．これらと標準入力データの読み込みおよび標準出力タイプアウトルーチンとからこのシミュレーションプログラムが構成されている．プログラムの流れをFig. 4に示す．プログラムの実行がはじまるとまず標準の入力データを読む．次にモデルに必要なデータの読み込みを実行するが，このプログラムはユーザが作成しておかねばならない．初期ルーチンはGPSSのGenerateに相当する機能のプログラムである．これらの初期設定が終了するとタイミングルーチンにはいるわけである．ここでは現在時刻，あるいは状態変数の値が設定値に等しくなったかどうかを判定し，もし条件が満たされていれば指定された離散系の事象を実行する．満足されていなければ連続系モデルの計算を行ない，その値を用いて積分を行なう．積分ルーチンは積分時間の短縮のためきざみ幅自動変更法を考慮したが不連続制御やスタート，ストップやむだ時間を含む系であると収束が悪くなり，取扱いも非常に不便になるのできざみ幅固定のRunge-Kutta-Gill法を用いた．

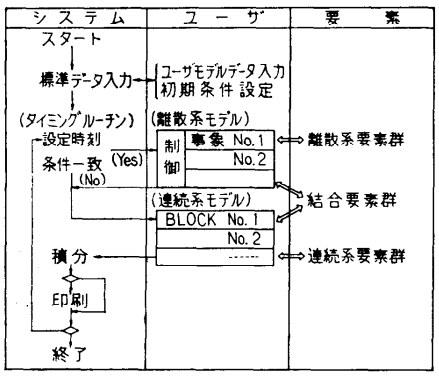



Fig. 4

積分のあとタイプアウト時刻であればタイプルーチンへ，またシミュレーション終了時刻であれば終了のルーチンへとぶ．それ以外の場合には積分により時間を進めたあと，最初の設定時刻，条件一致の判定へもどり，このループをくりかえしてシミュレーションが進行していく．

各モデルはユーザがそれぞれの要素を組み合せて記述する．連続系モデルはブロック線図の順序に要素をかいていけばよいが，ループになっているところは積分要素またはImplicit 要素*から書きはじめる．離散系モデルにおいてはEvent Orientedの表現をとる．すなわち，離散系モデルにおいて時間の経過するところでフローチャートを区切り，これによって作られたブロックを一つのEventとし，このEventを離散系要素の組合せにより記述していく．シミュレーション実行時にはこのEventごとに流れがコントロールされる．

## 4. 要　素

連続系，離散系の各要素は従来のCSMP，GPSS，SIMSCRIPTに準拠した．結合用の要素は連続系モデルから離散系へ移行するに必要な要素と離散系より連続系を制御するための要素とに分類される．主要要素をTable 2, 3, 4, 5に示す．

## 5. データ　ファイル

FORTRANのサブルーチンで要素を構成したのでシステムプログラムとのデータの授受が問題となるが，システムプログラム内のみで用いている変数はラベル付Commonを用いる．ユーザプログラムとシステムプログラム両方に用いる変数で，どの要素サブル

* Implicit 要素（連続系のブロック線図中の閉ループ中に積分器が存在しないとき，この閉ループ中のある時刻における各変数の値を求めるには一般に代数方程式を解かねばならない．このため適当な初期値を与えて繰返し計算により求める解に収束させていく．このための要素がImplicit（陰関数）要素である．Table 2 参照）

Table 2

| 機 能 | 数学的内容 | 一 般 形 | 要 素 の 説 明 |
|---|---|---|---|
| 積 分 | $\frac{1}{S}$ | Y=FINTE(Y0,X) | $Y=\int_0^t X dt + Y0$ |
| むだ時間 | $e^{-P\cdot S}$ | Y=DELAY(P,X) | Y(t)=X(t−P), t≧P Y(t)=0, t<P |
| 一次遅れ | $\frac{1}{C\cdot S+1}$ | Y=REALPL(Y0,C,X) | $C\dot{Y}+Y=X$ Y(0)=Y0 |
| 二次遅れ | $\frac{1}{S^2+2C1\cdot C2\cdot S+C2^2}$ | Y=CMPXPL(Y1,Y2,C1,C2,X) | $\ddot{Y}+2C1\cdot C2\cdot\dot{Y}+C2^2Y=X$ Y(0)=Y2, Ẏ(0)=Y1 |
| リードラグ | $\frac{C1\cdot S+1}{C2\cdot S+1}$ | Y=FLDLAG(C1,C2,X) | $C2\cdot\dot{Y}+Y=C1\cdot\dot{X}+X$ |
| Implicit | ループに積分器がない | Y=FIMPL(Y0,E,X) | \|Y−X\|≦F Y0 初期値, E 許容誤差 |
| ランプ入力 |  | Y=RAMP(P) | Y=t−P, t≧P Y=0, t<P |
| ステップ入力 |  | Y=STEP(P) | Y=1, t≧P Y=0, t<P |
| 比較器 |  | Y=CMPAR(X1,X2) | Y=1, X1≧X2 Y=0, X1<X2 |
| 2点リレー |  | Y=RELAY(P,X1,X2) | Y=X1, P≦0 Y=X2, P>0 |
| AND |  | Y=AND(X1,X2) | Y=1, X1>0. X2>0 Y=0 その他 |
| OR |  | Y=OR(X1,X2) | Y=0, X1>0. X2>0 Y=1 その他 |
| NOT |  | Y=NOT(X) | Y=1, X≦0 Y=0, X>0 |

Table 3

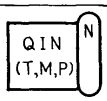
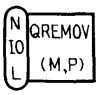
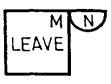
| 機 能 | 要 素 | 要 素 の 説 明 | 記 号 |
|---|---|---|---|
| スケジュール表の書き込み | SCHEDL(T,M,P) | スケジュール表に事象の発生時刻T, 事象番号 M, 情報 P(配列)を書き込む. |  |
| スケジュール表への割込み | CANCEL(X,IP,M,P,L) | スケジュール表を探索し, IP番目の情報がXに等しいものを取り出す, L=1 該当有, L=0 無し. |  |
| 待ち行列にはいる | QIN(N,T,M,P) | N番目の待ち行列に待ちからはずされた時の発生事象 Mと情報Pを持って入る. T, 参照データ. |  |
| 待ち行列から出す | QREMOV(N,IO,M,P,L) | N番目の待ち行列から事象M, 情報PをIOで指定した順に出す. IO=1 参照データの大きい順, IO=−1 小さい順. |  |
| STORAGEの確保 | ENTR(N,M,L) | N番目のSTORAGEからM個確保する. L=1 確保できた, L=0 確保できず. |  |
| STORAGEの解放 | LEAVE(N,M) | N番目のSTORAGEをN個解放する. |  |

Table 4

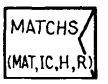
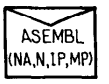
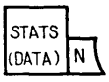
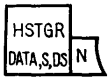
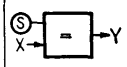
| 機　能 | 要　素 | 要　素　の　説　明 | 記　号 |
|---|---|---|---|
| 待ちから出しスケジュール表に書く | QSCHDL(N,IORD,L) | QREMOVとSCHEDLの機能を合せたもの．現在時刻でスケジュール表に書く．L＝1で該当するもの有，L＝0で無し． |  |
| 乱数発生 | RNOML(X,S) | 平均X，標準偏差Sの正規分布に従う乱数を発生する． | |
| 他のENTITYとの待ち合せ | MATCHS(MAT,IC,M,P) | マッチ表で対応するENTITY(MAT)をさがし，あれば現在時刻でスケジュール表に書く，ない場合，自分をマッチ表に書く． |  |
| ENTITYの合併 | ASEMBL(NA,N,IP,M,P) | IP番目の情報P(IP)が等しいN個のENTITYの合併． |  |
| 統計量の収集 | STATS(N,DATA) | 統計素のN行目がDATAなる値で書き変えられる． |  |
| ヒストグラムの作成 | HSTGR(N,DATA,S,DS) | ヒストグラム表のN行目をDATAの値によって度数を増加させる．S出発値，DSきざみ幅． |  |

Table 5

| 機　能 | 要　素 | 要　素　の　説　明 | 記　号 |
|---|---|---|---|
| 連続系モデルのスタート，リセット | BSW(I) | 連続系モデルをブロック化する．I，ブロックの番号． |  |
| | BSWSET(I,IS) | 連続系ブロックの状態指定．I，ブロック番号．IS＝1，スタート，IS＝0，ストップ，IS＝−1，ホールド． |  |
| 連続系モデルから離散系事象への移行 | CDSW(N,S) | 連続系から離散系へ移行のためのスイッチ．N，スイッチの番号　S＝1のとき移行． |  |
| | CTBLIN(N,M,P) | 移行のための情報の書き込み．M，移行先　N，スイッチ番号　P，情報 |  |
| 条件一致の判定 | EQUL(X,S,IS) | 条件一致の判定　S，設定値　X＝Sで1，X≠Sで0　IS，一致の条件 |  |

ーチンにも共通のものはラベルなし Common で，特定のサブルーチンとユーザプログラムにのみ関係する変数は引数を利用する．ユーザプログラム同志のデータの受渡しはラベルなし Common のあとに続ける．

データファイルとしてはスケジュール表，Entity の待ち合せ表，待ち行列表などをもっているが，これら

は空白番地とともにまとめられてリスト構造にして表の書込み，読出しを容易に迅速にできるようにしている．この他にストレッジの状態を示す表をもち，ストレッジの稼動状況を把握する．

## 6. 入出力

入力データの読込みはシステムが必要とする標準データとユーザのシステムモデルに必要なデータに分けられる．標準データは所定の形式できめられた順序で読込むようになっている．これらは JOB 番号，シミュレーション終了時刻，レポート打出しの時間間隔，使用するストレッジの種類とその種類に属するストレッジの数，積分のきざみ幅などである．ユーザモデルのためのデータはユーザが Start サブルーチンを作り，これに必要なデータの読込み命令を入れておく．

結果の出力は連続的な打出しと定時打出し機能をもっている．連続的な打出しとしては連続系システムの任意の変数を任意の時間分プリンタに印字または X-Y プロッタへプロットできる．X-Y プロッタへ出力するときは自動スケーリングを行うようになっている．また離散系のシステムについては各事象の実行ごとにその時刻，事象番号などをタイプすることができる．この機能もタイプアウトの時間幅を自由に指定できる．定時刻のタイプアウトは時刻，スケジュール，マッチ，待ち行列等各表の内容，待ち行列の最大，平均長さ，統計，ヒストグラムサブルーチンで予め収集されている変数の統計表，ストレッジの稼動率，使用回数，連続系積分器の値である．定時のタイプアウトはタイプアウト間隔をスタート時に標準データとして読込ませるだけでタイミングルーチンが自動的にこの出力ルーチンを call する．連続的タイプアウトはスタートルーチンでユーザが call しないと働かない．

## 7. 例 題

ゴミの焼却炉のプロセスを例に問題を構成してみよう．

（1） プロセスの説明

ゴミを満載したトラックが平均 15 分，標準偏差 5 分の正規分布で焼却場に到着する．焼却炉は 3 基あり，それぞれ同時にトラック 1 台分のゴミを処理できるゴミ仕込口がある．焼却場に到着したトラックは仕込口が空いていれば仕込口に車をつけるが，そのために 5 分間必要である．炉へのゴミの仕込みは総仕込み量の最小の炉を優先し，作業員 1 人で作業時間は 10 分である．焼却は焼却残量を $y$ とすると次の式で表わされるものとする．

$$\frac{dy}{dt}=-Ky(b-y).$$

ここで $K$ は燃焼速度，$b$ は炉容積を表わす．

ゴミは一定の率の不燃性物質を含むため一連の燃焼バッチにおいて総仕込量がトラック 4 台分になると次の仕込みを行わず，炉内残量が 5 %になるまで燃焼させ不燃性物質をとり出す．またトラック到着が遅れ，炉内残量が 5 %以下になったときも不燃性物質をとり出す．この作業は作業員 1 名で 10 分必要である．

（2） 目的

この問題ではいくらの量のゴミを処理することが可能であるか，また作業員の忙しさはどの程度であるか，作業の流れの状態，たとえばトラックの待ち状態はどうなるかをシミュレーションによって求める．

（3） 計算例

この例題のフローチャートを Fig. 5 に示す．ここでは Event 単位でフローを示しているが各 Event が数個の要素サブルーチンより構成されている．定時打出しによる標準レポートの例を Fig. 6 に示す．この例では，Match, Assemble は使用していないので，Empty となっている．Queue Table はタイプアウト時に偶然待ちがなかったため Empty になった．し

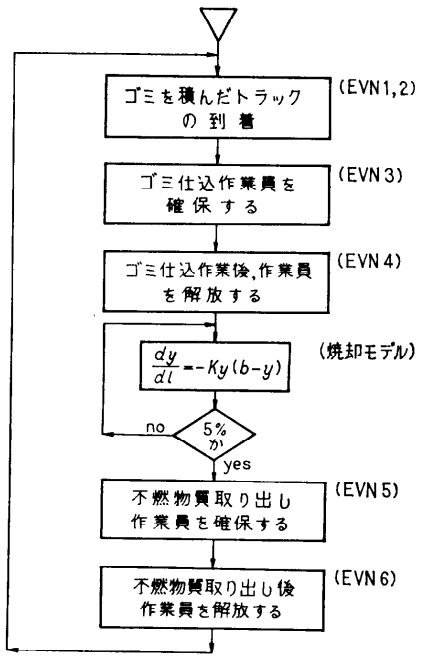



Fig. 5

たがって過去の待ち行列の統計はタイプアウトされている．炉の燃焼状態を X-Y プロッタにかかせた結果を Fig. 7 に示す．計算時間は(4)項記載の計算機を用いてシミュレーション 24 時間分で 150 秒である．

```
          JOB NO.   1   CLOCK   8.000

SCHEDULE TABLE
  TIME      EVENT NO.        INFORMATION
   8.000        1000       0.000    0.000
   8.011           4     250.000    1.000
   8.049           1     250.000    0.000

C-TABLE
  CDSW NO.      EVENT NO.   INFORMATION
     2              5        2.000   0.000
     3              5        3.000   0.000
     1              5        1.000   0.000

MATCH TABLE
  MATCH NO.  MATCH INF.  EVENT NO.  INFORMATION
   EMPTY

ASSEMBLE TABLE
  ASSEMBLE NO.  ASSEMBLE INF.  ASSEMBLE NO.  INFORMATION
   EMPTY

QUEUE TABLE   1
  TIME      EVENT NO.      INFORMATION
   EMPTY

Q-NO.   AVERAGE      MAXIMUM      AVERAGE
        LENGTH       LENGTH       TIME/TRANS.
   1     0.087          1           0.139
   2     0.003          1           0.023
   3     0.058          1           0.465

STORAGE
  STORAGE  CAPACITY  AVERAGE  MAXIMUM  AVERAGE  ENTRIES
    NO.             CONTENTS CONTENTS UTILIZATION
     1         1       0.62       1      0.6236     25
     2         1       0.37       1      0.3750      4

OUTPUT OF INTEGRATOR
  F(1)=0.  F(2)=0.7232E02  F(3)=0.4100E03
```

Fig. 6

なお積分のきざみ幅は 0.12 分である．

(4) 使用計算機

計算機は TOSBAC 3400-31, 16 kW を使用した．記憶容量が少ないのでディスクを利用しオーバレイを繰返して実行した．Fortran JIS 7000 レベルで記述しているのでオーバレイ部分以外は機種に関係なしに利用できる．現在のプログラムでは表の大きさは合計 100 事象まで，ストレッジの種類は 50 個までである．

## 8. むすび

連続・離散系両用シミュレーションプログラムは微分方程式系のモデルに作業員，論理判断などが関係してくる問題，例えばプラントのスタートアップ，シャットダウン，バッチ反応器，交通管理，自動運搬システムなどのシミュレーションに特に適している．これらの問題は事象の発生が時間あるいはシステムの状態によって決定されるので，従来の待ち行列タイプのシミュレーションプログラムはこのような問題に対しても自然な形でシミュレーションができるほか，微分方程式モデル系に用いても系の制御が自由自在にできるという特長を有している．なお本プログラムの大きさはカード枚数で表わすと約 2,200 枚である．

最後に本シミュレーションプログラムの発表を許可された東レ(株)に感謝いたします．

(昭和 47 年 3 月 1 日受付)
(昭和 47 年 4 月 24 日再受付)

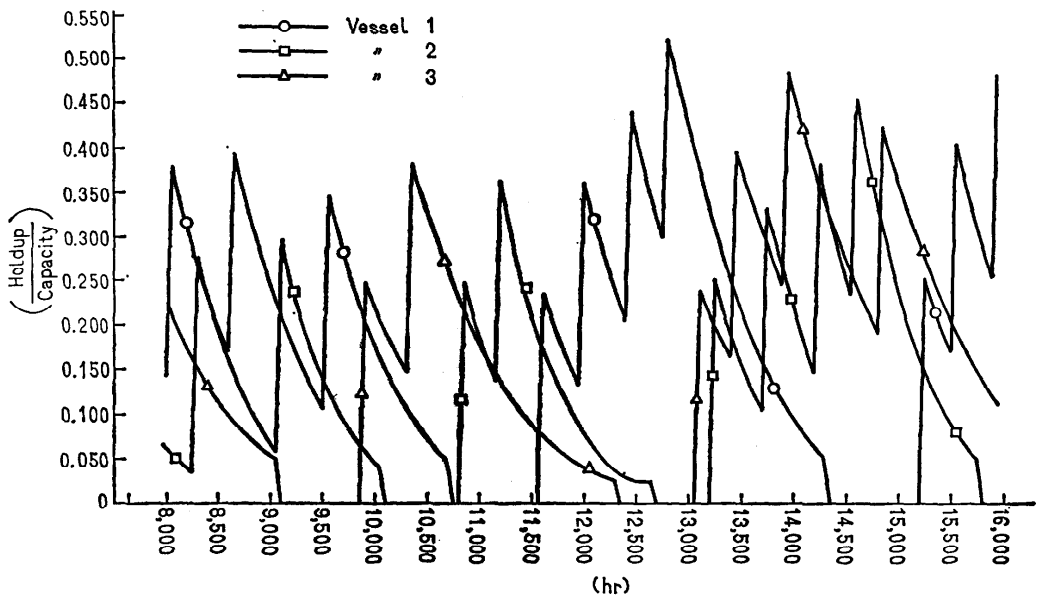



Fig. 7