ラージスケールシステムのグラフ論的分割

恒川　純吉

1. はじめに

本稿では非線形方程式と常微分方程式とからなる大次元の連立方程式を解くための，グラフ理論を応用した算法について紹介する．

筆者の属する㈱日本科学技術研修所では昭和41年度に化学プロセスの定常解を求めるためのシミュレーション言語 JUSE GIFS を開発し，昭和49年度には，これを拡張して化学プロセスの動的挙動を求めるためのシミュレーション言語ＤＰＳを開発した．化学プロセスは混合，反応，抽出など種々の機能をもつ化学装置と，それをつなぐストリーム(パイプに相当)から成り立っており，さらに制御装置がついていることが多い．ストリームには各成分の流量，圧力，温度などの変数があり，装置にはたとえばタンクの中の液面の高さや各成分のモル分率などの変数がある．装置を数式モデルで表現することによって．これらの変数の間の関数関係が非線形方程式あるいは常微分方程式で記述される．一般に小さな化学プロセスでも数百元の連立方程式となり，大きなものは数千元の連立方程式となる．このような大規模なシステムの解を求めるためには所要時間の面でも，計算機の内部記憶装置の面でも多くの工夫が必要になる．ことにダイナミックスを求めるシミュレーションにおいては，数千元の連立方程式をくり返し解くことになるので，方程式が線形であったとしても，まともに解くならば超大型計算機であっても，気の遠くなるほどの時間がかかるであろうことが想像されるであろう．そこで，変数間の関数関係をグラフで表現したうえで，できるだけ計算を効率的に行なえるように，不要の計算を省いたり，反復計算の部分を最小限に限定するなどして，最適な計算順序を定めているのである．グラフの表現を用いることにより，種々の算法はグラフの処理となり，ＯＲに関心をもつ本誌読者にとっておなじみの算法が適用できることになるのである．

ここで説明する手法は，ＤＰＳで実際にプログラム化され稼動しているものである．JUSE-GIFS は23社，ＤＰＳは７社によって利用されている実績をもち，ＤＰＳの場合は会話型プログラムへ変更することによって海外への販売も具体化している．

この手法は化学工業の面だけでなく，他の分野にも応用できる一般的なものであると考えている．本手法の開発にあたりご協力をいただいた電気通信大学の森口先生，東京大学の伊理先生，東京工業大学の大島先生ほかの諸氏に厚くお礼を申し上げる．

つねかわ　じゅんきち　㈱日本科学技術研修所

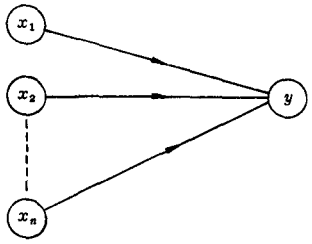


図 1　関数関係のグラフ

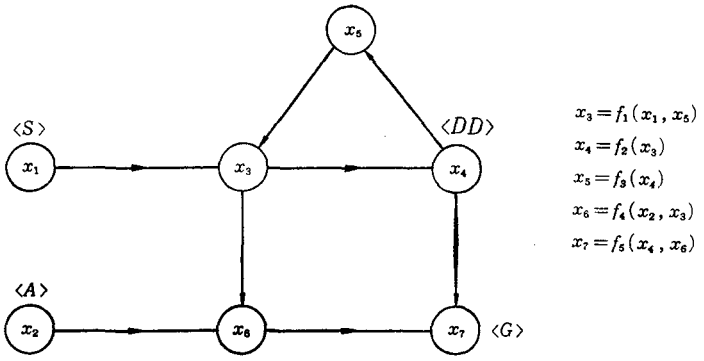


図 2　グラフの例

2.　方程式の記述とグラフによる表現

2.1　非線形方程式と静的グラフ

非線形方程式を表現する関数関係を1つの変数を求める形式で記述する．

$$y=f(x_1, x_2, \cdots, x_n) \tag{1}$$

方程式が1つの変数を陽に解く形で表わせないとき，すなわち，C を定数として，

$$f(x_1, x_2, \cdots, x_n)=C$$

の形で表わされている場合は，変数 ξ を追加し，

$$\xi=f(x_1, x_2, \cdots, x_n)$$

と表現する．そのうえで，変数 ξ の値は所与の値 C であるとして解けばよい．こうして，方程式はつねに (1) 式の形で記述できる．ここで y を左辺変数，$x_1, x_2, \cdots, x_n$ を右辺変数とよぶことにする．

変数を点とし，右辺変数から左辺変数へ有向枝を引くことにより，(1) 式は図1のグラフで表わすことができる．微分方程式を含まないグラフを静的グラフとよぶことにする．

一般に非線形方程式からなる連立方程式は，

$$\left.\begin{aligned} y_1&=f_1(x_{11}, x_{12}, \cdots, x_{1n_1}) \\ y_2&=f_2(x_{21}, x_{22}, \cdots, x_{2n_2}) \\ y_m&=f_m(x_{m1}, x_{m2}, \cdots, x_{mn_m}) \end{aligned}\right\} \tag{2}$$

の形で記述される．ここに左辺変数 $y_1, y_2, \cdots, y_m$ には同じ変数が2回以上現われないものとする．連立方程式 (2) に対して，静的グラフを作成することができる．図2にその一例を示す．

2.2　変数の性質

連立方程式の中の変数は，値の指定およびグラフ上の特性によっていくつかの性質に分類される．図2のグラフで説明しよう．

(1)　S形変数

値が設定 (set) されていて，以後の計算には設定値が使われる．通常はグラフの極小点に設定されるが，極小点以外の点に設定された場合は，その変数を求める式は無視される．変数 x_1 がS形である．

(2)　A形変数

グラフの極小点である点の値が設定されていない場合，その変数は未知変数であって，値を仮定 (Assume) して解かなければならない．

(3)　G形変数

関数の左辺に現われる変数の値が所与 (Given) であり，その関数値を評価したとき，所与の値に等しくなるように他の変数の値が調整される．図1ではA形変数 x_2 の値を変えて，G形変数 x_7 の値を所与の値に等しくすることになる．その結果 x_2 の値が求められる．

(4)　DD形変数

グラフ上に有向閉路が存在する場合，その有向閉路の中の変数の1つを仮定し，有向閉路に沿ってひと回りしてその値を求めたとき，仮定値と等しくなることが必要である．このような変数をDD形(Divided)の変数とよぶ．図2では x_3, x_4, x_5 が有向閉路を形成しているが，その中で x_5 を

DD形変数としている．

2.3 簡約化された方程式

連立方程式(2)を解く場合，S形変数は既知の定数であり，一般の変数は代入計算によって消去できるので，基本的な未知変数はA形変数とDD形変数だけとなる．一般に $x_1, x_2, \cdots$ をA形変数，$z_1, z_2, \cdots$ をDD形変数，$c_1, c_2, \cdots$ をG形変数の値とするとき，次の方程式に帰着できる．これを簡約化された方程式とよぶ．

$$\left.\begin{aligned} c_1&=F_1(x_1, x_2, \cdots, z_1, z_2, \cdots)\\ c_2&=F_2(x_1, x_2, \cdots, z_1, z_2, \cdots)\\ &\cdots\cdots\cdots\\ z_1&=G_1(x_1, x_2, \cdots, z_1, z_2, \cdots)\\ z_2&=G_2(x_1, x_2, \cdots, z_1, z_2, \cdots)\\ &\cdots\cdots\cdots \end{aligned}\right\} \qquad (3)$$

図2の例では次の方程式となる．

$$\left.\begin{aligned} C_7&=F_1(x_2, x_5)\\ x_5&=G_1(x_5) \end{aligned}\right\}$$

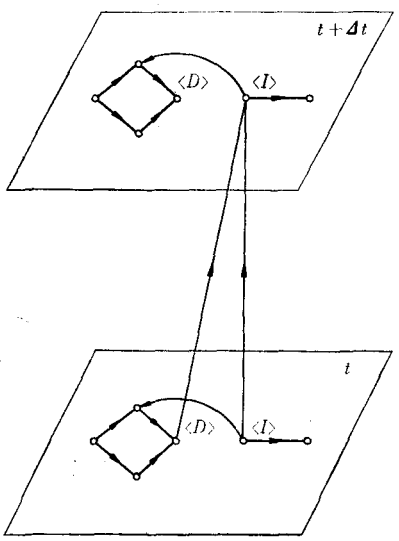
もともと数百元，数千元の連立方程式であっても，簡約化された方程式では小さな次元の連立方程式となることが多い．この簡約化された方程式は3.4で示すように，さらに小さな元数の方程式群に分けることができる．簡約化された方程式を解く方法は，非線形であるため反復解法を用いることになるが，DPSでは差分近似を用いてヤコビ行列を作成し，ニュートン法を用いて解いている．

2.4 微分方程式と動的グラフ

連立方程式の中に常微分方程式が含まれる場合には，関数の中に次の積分関数を含むことになる．

$$y_{t+\Delta t}=f(\tilde{y}, \tilde{d}, \Delta t) \qquad (4)$$

ここに y は積分変数，d は微分変数(微係数)，Δt は積分のきざみ幅である．積分公式によってどの時点の $\tilde{y}, \tilde{d}$ を用いるかが異なっている．また数時点の値を使用する公式もある．$\tilde{y}, \tilde{d}$ に $t+\Delta t$ 時点の値を含まないものが陽解法，含むものが陰解法とよばれる．陽解法か陰解法かの区別によって方程式のグラフ表現が異なることを注意しておく．



〈I〉 積分変数
〈D〉 微分変数

図3 Euler 法における動的グラフ

(1) 陽解法の場合

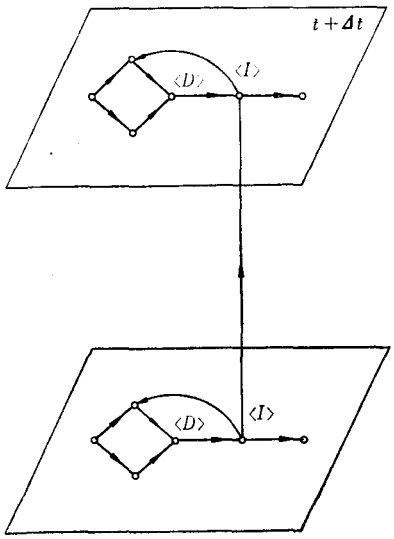
陽解法の場合は，(4)式の右辺の $\tilde{y}, \tilde{d}$ の値は $t+\Delta t$ より以前の値を使用するが，それらの値は既知であるとみなしてよい．したがって，時刻を t から $t+\Delta t$ に進めたとき，すぐに積分公式によって $y_{t+\Delta t}$ を求めることができる．その後で $t+\Delta t$ 時点の非線形方程式を解く手順となる．最も簡単な公式はオイラー法であり，

$$y_{t+\Delta t}=y_t+d_t\Delta t$$

と表現されるが，これをグラフで示すと図3のようになる．このようなグラフを動的グラフとよぶことにする．y_t, d_t は静的グラフ上ではS形変数として扱うことができる．

(2) 陰解法の場合

(4)式の右辺に $t+\Delta t$ 時点の値が用いられている場合は，その値は未知である．したがって，一般には反復計算が必要になる．DPSでは，この



〈I〉 積分変数
〈D〉 微分変数

図 4 Backward Euler 法の動的グラフ

部分を非線形方程式のグラフと同一に取扱っている．簡単な例としてバックワード・オイラー法では，

$$y_{t+\Delta t}=y_t+d_{t+\Delta t}\Delta t$$

と表現されるが，対応する動的グラフは図4のようになる．$d_{t+\Delta t}$ を求める関数が間接的にでも $y_{t+\Delta t}$ を含むときは，有向閉路を形成することになり，2.3で示した手順で反復的に解を求めることになる．

3. グラフ理論を応用した算法

3.1 不必要な変数の除去

DPSでは化学装置のモデルをあらかじめ用意しておき，与えられた装置間の接続関係を用いて，自動的に変数間のグラフを作成する．しかし記述された数式モデル中の式を必ずしも全部計算する必要はない．たとえばモデルの中に物質収支と熱収支の数式の記述があっても，物質収支だけを求める目的のシミュレーションであれば，熱収支の式は計算しなくてよい．出力が指示されている変数およびシミュレーション終了条件に関係する変数をR(Required)形変数とよぶ．このとき，次のようにR形変数を求めるために必要な変数だけを選び，不必要な変数を除去する．

(1) G形変数について，その上位のA形変数が下位にR形変数をもつならば，そのG形変数をR形変数とみなす．

(2) R形変数でない変数が，その下位にR形変数をもたないとき，その変数を除去する．

この他，DPSではバルブ操作などの操作条件によって，装置のブロック全体を計算不要とする機能ももっている．

3.2 変数と方程式の適合性

非線形連立方程式においては，未知変数の個数と方程式の個数とが一致しなければならない．この条件はA形変数の個数とG形変数の個数が一致することと等価である．またこの条件を満たしたとしても，図5の例のような場合には，方程式を解くことができない．この点については次の定理でチェックすることができる．

［**定理**］ A形変数とG形変数が同じ k 個だけある連立方程式において，A形変数とG形変数からそれぞれ1個ずつ取り出して，重複することのない k 対を次の条件を満たすように選ぶことができるならば，その連立方程式は一般的に解ける．

(a) 対をなすA形変数からG形変数へ有向路が存在すること．

(b) 異なる対の有向路が同じ点を通らないこと．

ここで一般的に解けるということは，方程式の構造上解けることを意味し，数値的に解けるかど

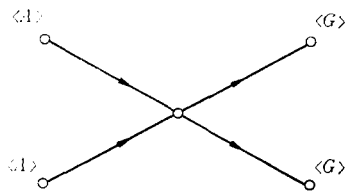


図 5 構造的に解けないグラフ

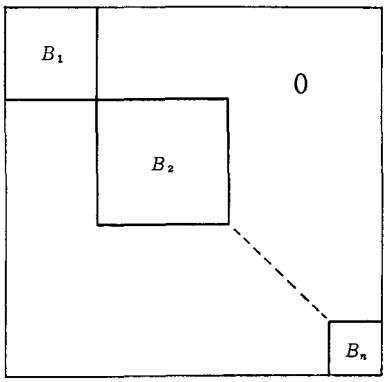


図 6 ブロック三角化された行列

うかは保証しない．この定理は，A形変数に1単位の供給量，G形変数に1単位の需要量があり，各点の容量が1に制限されている輸送問題を解くことと一致する．

3.3 DD形変数の決定

グラフ上に有向閉路が存在する場合，DD形変数を選ぶ必要がある．これは有向閉路の中の変数を切り離して閉路のないグラフとすることに相当するが，切り離す変数の数はできる限り少なくすることが方程式の求解の手間を減らす意味でのぞましい．分割変数の個数を最小化する算法は手間がかかりすぎるので，有向閉路の多くの組に関係する変数からDD形変数に取り入れている．実用的には問題はないようである．

3.4 ブロック三角化と計算順序の決定

簡約化された方程式はブロック三角化によって，さらに小さな元数の方程式群に分割できることが多い．行列Bを定義し，その要素B_{ij}は，第i方程式が第j変数を含むとき1，含まないとき0とする．この行列Bに適当な行置換，列置換を施した結果図6のように対角線上に元数の小さな正方行列を並べ，その右上方向の要素はゼロにすることができる．(小行列に分けられないこともある．)これをブロック三角化とよぶ．図6の場合には上部の小行列B_1をまず解き，次にB_2の部分を解くというふうに，小さな元数の連立方程式を順次解けばよいことになる．これによって，計算の手間を少なくすることができる．

ブロック三角化のための処理も，簡約化された方程式の接続関係を表わすグラフを用いて行なう．すなわち簡約化された方程式の左辺変数（G形およびDD形変数)と右辺変数(A形およびDD形変数，DD形変数は左辺と右辺とを別の変数として扱う）からなる2部グラフを作り，右辺変数から対応する左辺変数に有向枝を引く．このグラフに対して次の処理を行なう．

(1) 右辺変数からなる有向枝で結ばれている左辺変数の1つを選んで対を作り，すべての変数が1つの対に属するようにする．左辺変数から対となる右辺変数に向けて有向枝を加える．

(2) グラフ上に有向閉路が存在するとき，その同じ閉路に属する変数をすべて同じブロックとする．

(3) ブロック間のグラフ上の上位と下位の関係を調べ，上位のものから順に並べる．結局簡約化された方程式は，上位のブロックから順次解けばよいことになる．

3.5 計算時点と計算順序の設定

常微分方程式を解く場合，非線形連立方程式のほうは必ずしも毎時刻に解かなくてもよい．変数ごとに計算時点を4つに分類し，できるだけ計算する回数を減らすようにしている．

(1) 開始時(INITIAL)

時刻に関係なく，最初に1度だけ計算すればよい変数であり，S形変数だけから求められる変数である．

(2) 毎時(EVERY TIME)

積分，微係数，時刻，きざみ幅，シミュレーション終了条件に関係する変数は，積分の各ステップごとに毎時解かなければならない．

(3) 出力時(PRINT TIME)

開始時，毎時の計算を必要としない変数のうちで，指定された間隔で結果を出力することを指示

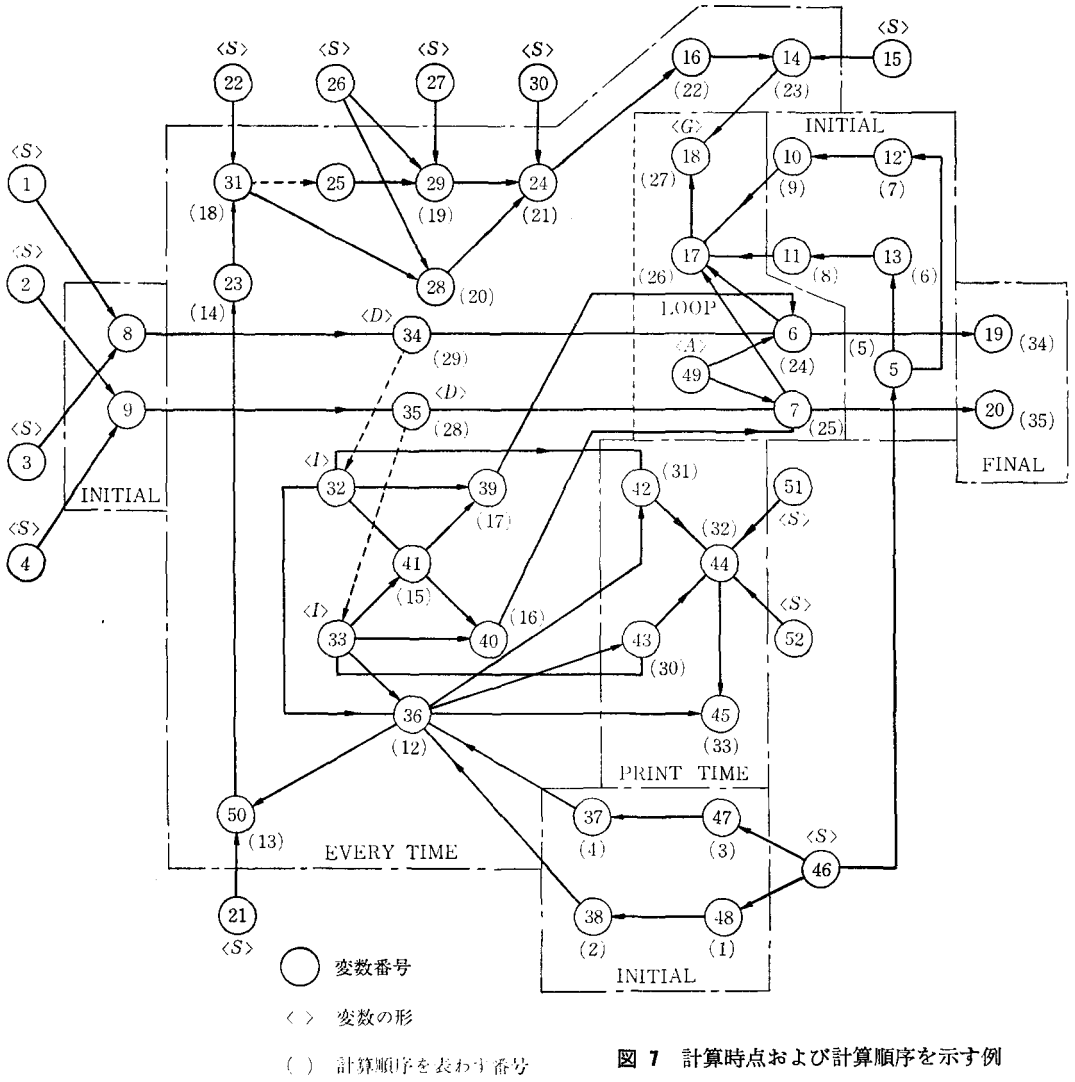


図 7　計算時点および計算順序を示す例

されている変数は，その出力指定時刻だけに計算する．

(4) 最終時(FINAL)

上記(1)，(2)，(3)以外の変数

3.4 で定めた簡約化された方程式の計算順序と本節の計算時点とを用いて，もとの方程式のグラフをたどることにより，最終的に最適化された計算順序を定めることができる．

4. 実　例

図 7 に関数関係のグラフと結果として得られた計算順序の例を示しておく．この例は簡単な化学プロセスのシミュレーションの実例であり，もともとのモデルは 150 個の変数から成り立っていたが，不要な変数を除去した結果52個の変数が計算に用いられている．35個の代数関数と 3 個の積分が評価されるが，代数関数は開始時に11個，毎時に18個，出力時に 4 個，最終時に 2 個が計算される．反復計算に含まれる代数関数は 4 個である．