I －26　等式制約探索法による探索プログラム

群馬高専　正員　平田恭久
東京都立大学　正員　伊藤文人

1. 等式制約探索法の考え方

等式制約探索法とは不等式制約付き最小化問題に対し，不等式制約の中から等式制約として与えられる制約条件を選び，最小値を探索しようとするものである。筆者らは以前から活荷重合成桁の断面決定を例にして最適解の探索方法について考察を進めてきたが，その考察から得られた方法を等式制約探索法と略称した。もし選んだ等式制約が最適解での等式制約（最適解が存在する活性な制約面）なら，選んだ等式制約付き最小化問題を解けば最適解が得られるが，一般に探索を開始する時点では最適解での等式制約は不明である。よって任意に選んだ等式制約から出発し，等式制約が具備しなければならない条件を満足するように等式制約を入換えながら探索していけば，探索が終了した時点では最適解に到達している（Kuhn-Tucker 必要条件を満足している）とするのがこの方法の考え方である。

探索の途中で選んだm個の等式制約（全体の制約式はM個，変数はn個）が具備しなければならない条件(1)式である。目的関数と制約条件によるLagrange 関数$L(x,\lambda)=f(x)+\lambda^T g_m(x)$について，勾配ベクトルを探索変数$x_l$と制約変数$x_m$で分けて表わすと(2)式になり，(2)式第2行よりLagrange 乗数λ_mが計算できる。(2)式第1行は探索方向ベクトル（－を付けて）として用いることができるが，もし最小値が探索できたとすると(2)式第1行＝0になるので探索での収束判定にも使用できる。

$$\left.\begin{array}{ll} g_{M-m}(x)<0 & \text{：他の制約式を不等号で満足} \\ \lambda_m \geqq 0 & \text{：有利な制約式} \\ \dfrac{\partial g_m}{\partial x_m}\text{の}\ \mathrm{rank}=m & \text{：}g_m(x)=0\ \text{の解が存在} \end{array}\right\} \quad (1)$$

$$\left.\begin{array}{l} \dfrac{\partial L(x,\lambda)}{\partial x_l}=\left(\dfrac{\partial f(x)}{\partial x_l}\right)^T+\lambda_m^T\dfrac{\partial g_m(x)}{\partial x_l} \ \text{：}n-m\text{個} \\ \dfrac{\partial L(x,\lambda)}{\partial x_m}=\left(\dfrac{\partial f(x)}{\partial x_m}\right)^T+\lambda_m^T\dfrac{\partial g_m(x)}{\partial x_m}=0\text{：}m\text{個} \end{array}\right\} \quad (2)$$

制約式g_iに抵触したら($g_i(x)>0$)等式制約g_mにg_iを加えるが，このとき変数x_jを制約変数x_mに加える　(3)

不利な制約式g_iを生じたら($\lambda_i<0$)等式制約g_mからg_iをはずすが，このとき変数x_jを制約変数x_mからはずす　(4)

2. 制約式および変数の入換え規則

探索の途中で常に(1)式を満足するように等式制約を入換えていくには制約式および変数の入換え規則を作成しておかねばならない。この入換え規則が等式制約探索法によるプログラムでのアルゴリズムの中心になるが，ここでは活荷重合成桁の主桁断面の決定を例にして入換え規則を説明する。探索を進めていくとき，①制約式に抵触したらその制約式を等式制約g_mに取入れ，②$g_m(x)=0$を解いてλ_mを計算し，③$\lambda_i<0$の制約式があればg_mからはずす，ことにより等式制約の入換えが行われるが，これに付随して制約変数x_mの入換えが行われる。制約式の入換え規則は比較的簡単であり(3)，(4)で示されるが，変数の入換え規則すなわち(3)，(4)での変数x_jをどのように選ぶかが問題である。

制約式番号

0	1	・・・・・・・	17	18
		・・・・・・・		

対応変数番号が入いる　(5)

変数番号

	0	1	2	3	4	5	6
断面番号 0							
1							
2							
3							
4							
	h_w	t_w	b_c	b_t	t_c	t_t	l_x

探索変数なら －1
制約変数なら制約式番号　(6)

x_jの選び方としては(1)式第3行の$g_m(x)=0$の解が存在するすなわち方程式が解けるような変数をx_mに残すことが第一である。このため制約式に対応した変数を定めることとし，制約式番号0～18の19個の制約式についての対応変数を(5)の"対応表"で表わした。また探索変数と制約変数の区別，等式制約g_mの制約式番号を示すため(6)の"変数表"を用意した。変数欄のh_w，t_wは腹板

x_mに加える変数x_jは：抵触制約式の対応変数が探索変数なら対応変数を，対応変数が制約変数なら探索変数の一つを選ぶ　(7)

x_mからはずす変数x_jは：g_mの対応変数が制約変数として残るようにする：抵触してλ_mを計算するとき加えたx_jをx_mからはずす　(8)

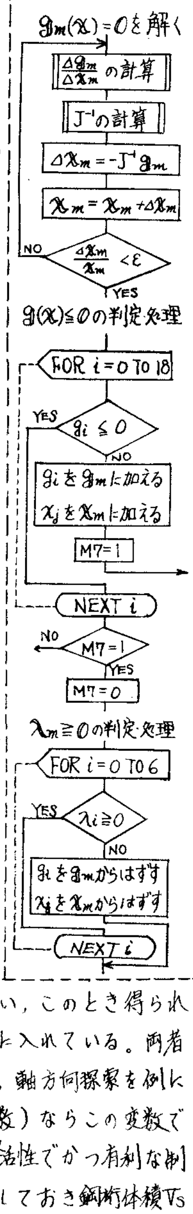
高および腹板厚（各断面共通の変数），bc，btはフランジ幅，tc，ttはフランジ厚，ℓ_xは断面変更点の位置（個数はN）である。

(7)，(8)に変数の入換え規則を示すが，変数の入換えについては，①$g_m(x)=0$が解ければよい，②探索に便利である，の2点が満足できればよいので種々の入換え規則が考えられる。x_mに加える変数およびx_mからはずす変数の選び方により局部的には探索経路が異なってくるが，大域的な探索経路は変数の選び方に関係なく同じになる。プログラムでの制約式および変数の入換えは変数表の変数欄に，①g_m，x_mに加えるときは制約式番号を代入し，②g_m，x_mからはずすときは"−1"を代入する。このときg_mの対応変数がx_mに残るように変数表の中で制約式の移し換えを行うが，対応変数であるかどうかは対応表を参照すれば分かる。

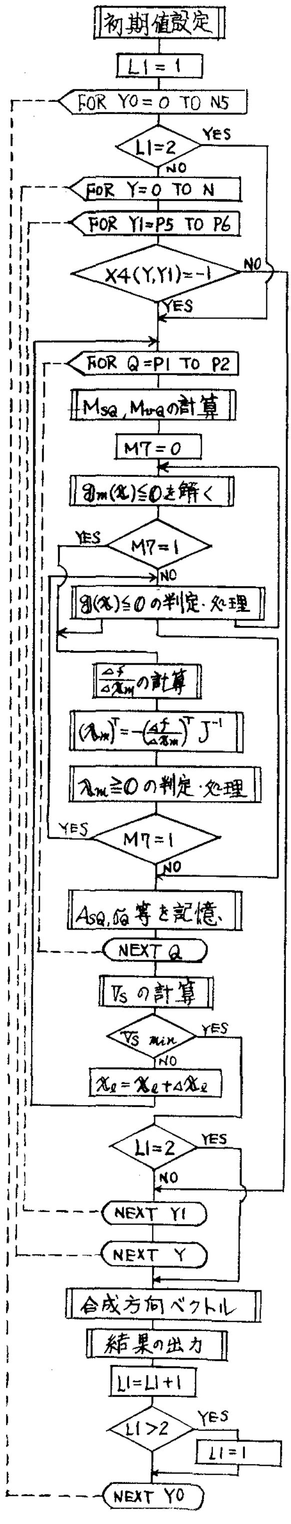
3. 探索プログラム

等式制約探索法による最適解の探索プログラムの概要フローチャートを図－1に示す。このプログラムで最も重要なサブルーチンは，①$g_m(x)=0$を解く，②$g(x)\leqq 0$の判定処理，$\lambda_m\geqq 0$の判定処理，の3個のルーチンでありその内容を図－2に示す。この3個のルーチンについての処理順序はM7で判別しており，上記②，③のルーチンには入換え規則によるアルゴリズムが組込まれている。Newton法でJacobi行列の逆行列J^{-1}を計算しておくとλ_mは簡単に計算できる。$\frac{\Delta g_m}{\Delta x_m}$，$\frac{\Delta f}{\Delta x_m}$の差分の計算では$g_m$の制約式番号，$x_m$の変数番号等は変数表を参照することにより分かる。プログラムはBASICで作成したが大きさは約25KBである。

図－2



図－1



最小値を探索する多次元探索法としては軸方向探索を用い，このとき得られる合成方向ベクトルによる合成方向探索を軸方向探索の間に入れている。両者合わせてN5回反復するがL1=1のときが軸方向探索である。軸方向探索を例にとると，変数表X4(Y,Y1)=−1（Y断面のY1変数が探索変数）ならこの変数で軸方向探索を行うが，この変数が関係するQ断面について"活性でかつ有利な制約面"を見つけ，このときの断面積A_{sq}，応力度σ_q等を記憶しておき鋼桁体積V_sを計算する。V_s最小点が得られたら次の変数に探索が移行する。

4. まとめ

等式制約探索法による探索プログラムを作成し計算を行っているが，探索プログラムとしての機能を十分に発揮できることが判明した。しかしながらプログラムにはまだ細かい点で改良の余地が残っており，プログラムを常に修正しながらアルゴリズムを洗練させていく必要がある。また常に"活性でかつ有利な制約面"上にあるので探索途中の解は設計計算上意味のある値になっており，プログラムを若干修正すれば探索プログラム以外の使い方も可能である。