船舶技術研究所報告　第25巻　第6号　研究報告（昭和63年11月）

# オブジェクト指向言語 SMALLTALK を用いた防撓板の最適設計

青木　元也*

# Optimum Design of Stiffened Plates using the Object-Oriented Programming Language SMALLTALK

By

Genya Aoki

**Abstract**

This paper describes an optimum design of stiffened plates using the object-oriented programming language SMALLTALK. Monte Carlo method is used to obtain an optimum value. The constraint conditions are related to the ultimate stress of panels, the collapse pressure of panels, the buckling stress of stiffeners, the buckling stress of a stiffened plate and the collapse pressure of a stiffened plate. The objective function is the volume of a stiffened plate. Example cases are demonstrated. The following remarks are obtained in the process of developing the program.

(1) SMALLTALK makes it easy to revise or add a program because of its small programming unit, and to change the values of parameters and the search range of variables due to its interactive feature.

(2) Monte Carlo method is not so efficient in search of an optimum value, but gives a satisfactory result in the stage of an initial design.

## 1. まえがき

日本造船業存続のための模索が各分野において行われているが，新しく発展した技術，学問の適用法の検討もその一環であると考えられる。著者は，近年発達の著しい知識工学の構造解析/設計への適用法についての基礎的研究[1)2)]を行っている。知識工学は情報処理分野の新しい工学であり，対象領域の専門知識および専門家のもつ経験的知識を体系化して，知的情報処理システムを開発するための方法論を与えるものである。これに用いられるプログラミング言語には，開発過程における試行錯誤に適した柔軟さが要求され，代表的なものとしては LISP, PROLOG, SMALLTALK 等がある。本報告では，このうちオブジェクト指向言語として知られている SMALLTALK の適用法を検討

* 構造強度部
原稿受付：昭和63年9月1日

した。

適用の対象としては，防撓板の最適設計とした。防撓板は船体構造の一般的な構成要素であるが，ここでは設計荷重として一方向のみの面内荷重と面に垂直な荷重とを考慮した。防撓板全体の縦と横との寸法および面内と垂直との設計荷重が与えられたとき，強度条件を満足し全体重量を最小にする防撓材の本数および寸法を求めるプログラムを検討した。

強度上の制約条件を満たして重量を最小にする寸法を求める問題は最適化問題であり，各種の解法がある。ここでは，問題が非線形であること，変数が離散量であること，プログラムが簡単なこと等を考慮してモンテカルロ法を用いた。

## 2. SMALLTALK の概要

SMALLTALK についての詳細な解説は専門書[3]に譲り，ここではその要点を簡単な例に基づいて説明する。いま矩形平板について考えてみる。変数としては材料定数と寸法とを有し，また，この矩形平板について面内変形，撓み変形，座屈，振動等の挙動を算定する手続きが定義されているとする。抽象的な矩形平板という概念を Plate という名のクラスと考え，その変数に具体的な数値を入れたものをこのクラスのインスタンスと称する。クラスおよびそのインスタンスは変数の集合であるデータと手続きとから構成されており，このデータと手続きとの組合せをオブジェクトと呼んでいる。オブジェクトをプログラミングの基本単位とした言語システムをオブジェクト指向言語と称しており，SMALLTALK はその代表である。

変数に具体的な数値を入れたインスタンスをいま aPlate1，aPlate2，……と表すとする。座屈荷重を求める手続きが buckling というメッセージ名で定義されているとすると，インスタンス aPlate1 の座屈荷重は，bukling というメッセージを aPlate1 に送るという形の次の命令を実行することによって求めることができる。

```
aPlate1  buckling
```

また，面外等分布荷重 q を受ける場合の変形を求める手続きが displacement : q というメッセージ名で定義されているとすると，次の命令を実行することによって aPlate2 の変形を求めることができる。

```
aPlate2  dispalcement : q
```

ここで，q は荷重の値を表す引数である。

SMALLTALK の特徴のひとつとしてクラスの階層構造がある。本報告ではこの機能は利用していないか，プログラムが複雑で大きな場合には非常に利用価値の高いものである。SMALLTALK では，四則演算，配列操作，条件選択，繰返し機能，グラフィックス等についてすでに約250のクラスが定義されており，これらのクラスが保持している手続きは5000にもおよぶ。したがって，これらを有効に利用することが，新たなプログラムの効率的な開発に際して必要となる。

## 3. 防撓板に関する強度算定式

本報告で設計の対象とする防撓板の構造寸法および荷重を Fig. 1 に示す。設計荷重は一方向面内荷重$\sigma$と面外荷重 q である。面内荷重が加わる方向をここでは軸方向と称することにする。防撓板の軸方向辺およびそれに垂直な方向の辺の寸法をそれぞれ A，B とし，板厚を t とする。防撓材については，軸に垂直な方向に矩形断面 $h_1 \times s_1$ のものが m 本，間隔 a で配置されるとし，軸方向には断面 $h_2 \times s_2$ のものが n 本，間隔 b で配置されるとした。

防撓板の強度評価基準としては，防撓板が用いられている構造部分あるいは荷重条件等によって各種考えられるが，ここでは防撓材に囲まれたパネル，防撓材，防撓板全体がそれぞれ設計荷重に対して耐荷力を有することとした。まず面内荷重を受けるパネルについては，座屈によって面外変形が生じてもまだ十分な耐荷力を有しているが，端部の防撓材部分が降伏応力に達したとき耐荷力を失うとして，そのときの荷重を強度評価基準とした。パネルが面外荷重を受ける場合には，塑性関節線が生じて崩壊機構が形成される荷重を基準とした。軸方向力を受ける防撓材については，3 辺単純支持，1 辺自由の長方形板と考え，座屈によって耐荷力を失うので，そのときの荷重を基準とした。防撓板全体については，面内荷重に対しては全体座屈荷重

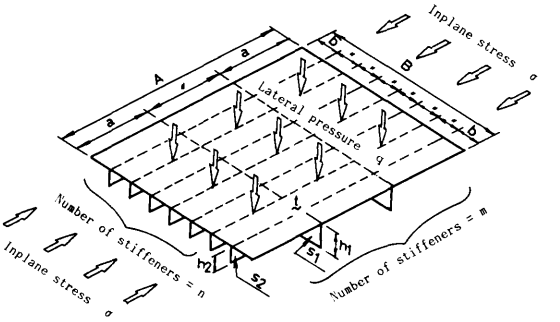


Fig. 1　Stiffened plate under design loads

を基準とし，面外荷重に対しては防撓材を含んだ塑性関節線が生じて崩壊機構が形成される荷重を基準とした。上述したパネル，防撓材，防撓板全体についての強度算定式は現在までに発表されている研究を参照することとし，以下にそれらをほぼ統一した形に書き改めたものを示す。ここで用いた寸法を表す文字は Fig. 1 に示した通りであるが，$\sigma_y$，E，$\nu$はそれぞれ降伏応力，ヤング率，ポアソン比を表わしている。

面内力を受けるパネルの最大平均応力$\sigma_u$は，パラメータ$\beta=(b/t)\sqrt{\sigma_y/E}$を用いて次式で表される[4]。

$$\left.\begin{aligned}\sigma_u/\sigma_y &= 2/\beta-1/\beta^2 \quad , \ \beta\geq 1\\ &= 1 \qquad\qquad , \ \beta<1\end{aligned}\right\} \tag{1}$$

これを図示すると Fig. 2 のようになる。

面外力を受ける周辺支持のパネルの崩壊荷重 $q_c$は，パラメータ$\beta=a/b$ を用いて次式で表される[5]。

$$q_c=\frac{8M_p}{b^2}\frac{\sqrt{1+3\beta^2}+1}{\sqrt{1+3\beta^2}-1}, \quad \beta\geq 1 \tag{2}$$

ここで，$M_p=\sigma_y t^2/4$ は単位幅当りの塑性モーメントである。これを図示すると Fig. 3 のようになる。

面内力を受ける3辺支持，1辺自由の長方形板（防撓材の計算モデル）の座屈荷重$\sigma_{cr}$は，パラメータ$\beta=(h/s)\sqrt{\sigma_y/E}$を用いて次式で表される[6]。

$$\left.\begin{aligned}\frac{\sigma_{cr}}{\sigma_y}&=\frac{\pi^2}{12(1-\nu^2)}\{0.42+(\frac{h}{a})^2\}\frac{1}{\beta^2}, \qquad \beta\geq\beta_0\\ &=1, \qquad\qquad\qquad\qquad \beta<\beta_0\end{aligned}\right\} \tag{3}$$

ここで，$\beta_0=\pi\sqrt{0.42+(h/a)^2}/2\sqrt{3(1-\nu^2)}$ である。これを，h/a=0.20および0.05の場合について図示すると Fig. 4 のようになる。

防撓板の強度算定には有効幅付防撓材の断面係数が必要となる。そこで，Fig. 5 に表す断面寸法についての係数を次に示す。断面2次モーメント I については，

$$\begin{aligned}e_2&=(sh^2+2sht+wt^2)/2(sh+wt)\\ e_1&=h+t-e_2\\ I&=\{we_2^3-(w-s)(e_2-t)^3+se_1^3\}/3\end{aligned} \tag{4}$$

塑性断面係数 z については，

hs≥wt の場合

$$\begin{aligned}e_2&=h/2+t-wt/2s\\ e_1&=h+t-e_2\\ z&=t(w-s)(e_2-t/2)+s(e_2^2+e_1^2)/2\end{aligned} \tag{5}$$

hs<wt の場合

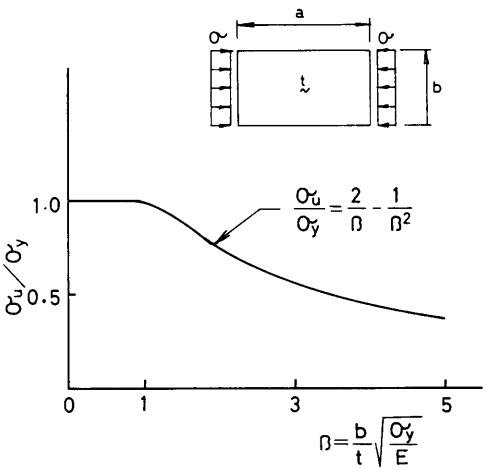


Fig. 2　Ultimate stress of panels

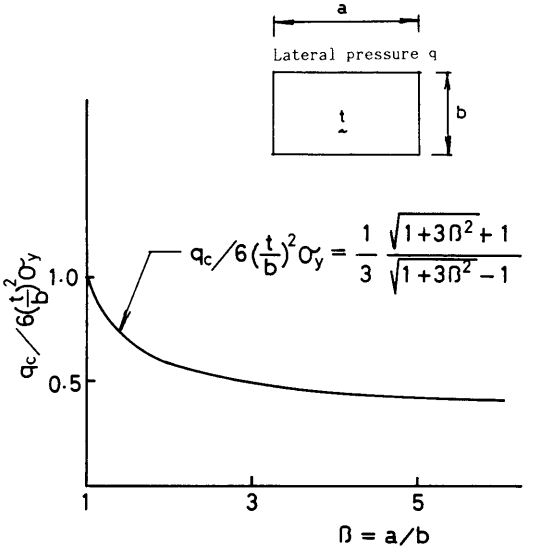


Fig. 3　Collapse pressure of panels

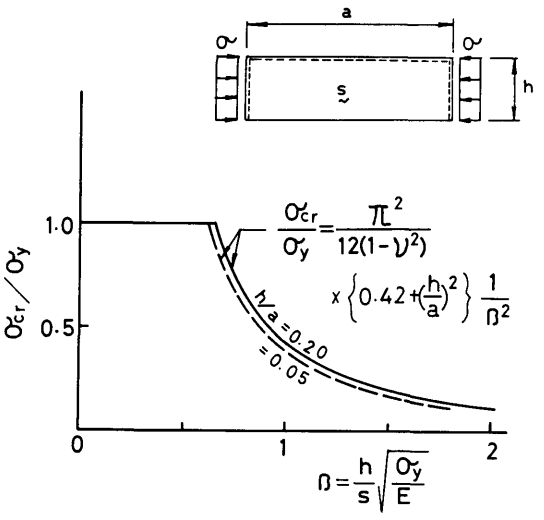


Fig. 4 Buckling stress of stiffeners

$$e_2=(t+sh/w)/2$$
$$e_1=h+t-e_2$$
$$z=w\{e_2^2+(t-e_2)^2\}/2+sh(e_1-h/2) \qquad (6)$$

防撓板が面内荷重を受けるときの座屈荷重$\sigma_{cr}$は，パラメータ$\beta=(B/t_2)\sqrt{t_2^3/i_2}\ \sqrt{\sigma_y/E}$を用いて次式で表される[7]。

$$\left.\begin{aligned}\frac{\sigma_{cr}}{\sigma_y}&=\frac{4\pi^2}{1-\nu^2}\sqrt{\frac{i_1}{i_2}}\frac{1}{\beta^2}\ ,\ \beta\geq\beta_0\\ &=1\qquad\qquad\ ,\ \beta<\beta_0\end{aligned}\right\} \qquad (7)$$

ここで，$\beta_0=2\pi\sqrt[4]{i_1/i_2}/\sqrt{1-\nu^2}$である。また，$t_2=t+ns_2h_2/B$, $i_1=I_1/a$, $i_2=I_2/b$である。$I_1$, $I_2$は有効幅付の防撓材の断面2次モーメントであり，ここでは有効幅を防撓材間隔とした。この式を，$i_1/i_2=1.0$および0.1の場合について図示すると Fig. 6 のようになる。

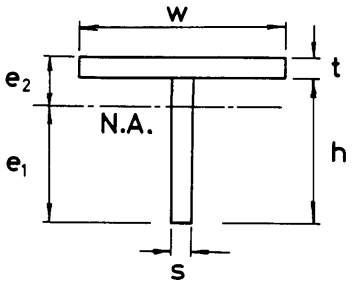


Fig. 5 Section of a stiffener with effective plate

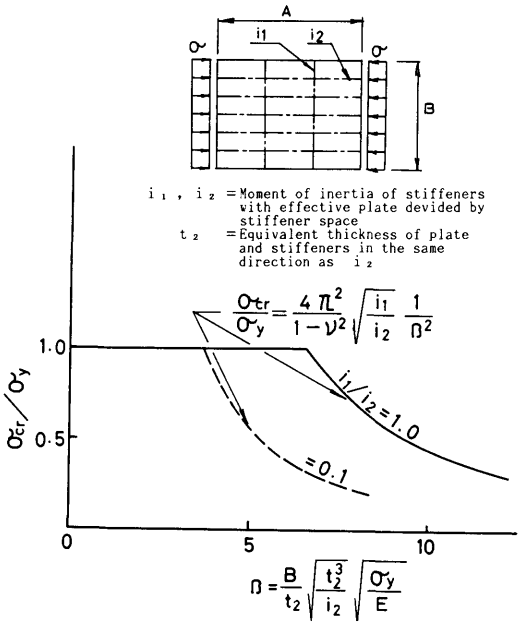


Fig. 6 Buckling stress of stiffened plates

面外荷重を受ける防撓板の崩壊荷重 $q_c$ は，パラメータ $\beta=(A/B)\sqrt{M_{p1}/M_{p2}}$ を用いて次式で与えられる[8]。

$$q_c\Big/\left(\frac{24M_{p1}}{B^2}\right)=\frac{1}{3}\,\frac{\sqrt{1+3\beta^2}+1}{\sqrt{1+3\beta^2}-1},\quad \beta\geq 1 \tag{8}$$

ここで，$M_{p1}=\sigma_y t^2/4+mz_1\sigma_y/a$

$M_{p2}=\sigma_y t^2/4+nz_2\sigma_y/b$

$z_1$, $z_2$ は有効幅付の防撓材の塑性断面係数であり，ここでは有効幅を防撓材間隔とした。この式を図示すると Fig. 7 のようになる。

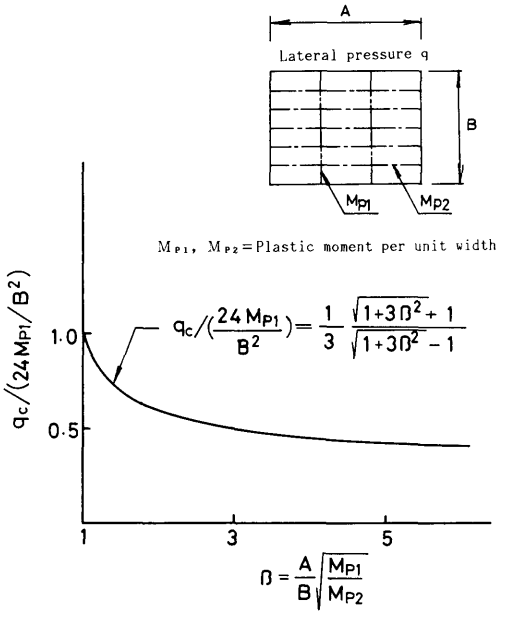


Fig. 7　Collapse pressure of stiffened plates

## 4. 最適設計プログラムの構成

### 4.1 プログラムの概要

構造物の設計においては，設計条件に合致し費用を最も安くする設計諸量を決定することが課題となる。この際に，設計技術者の経験が大きく関係してくるが，最適設計法の利用もかなり盛んに行われてきている。これには各種の手法があるが，まだ最適設計の汎用解法といわれるものはない。ここでは，SMALLTALK の適用法が主題であるので，問題をできるだけ簡単化した。

制約条件としては，前節で述べた座屈荷重あるいは崩壊荷重が設計荷重以上であることとした。また，目的関数は防撓板の重量すなわち体積とし，これを最小にすることが最適設計の目的であるとした。既定値パラメータとしては，面内設計荷重 $\sigma_d$，面外設計荷重 $q_d$，降伏応力 $\sigma_y$，ヤング率 E，ポアソン比 $\nu$，防撓板の軸方向辺長さ A，それに垂直方向辺の長さ B を考えた。設計変数としては，防撓板の厚さ t，軸に垂直方向および軸方向のそれぞれの防撓材の本数，深さ，厚さ，すなわち m, $h_1$, $s_1$, n, $h_2$, $s_2$ とした。

防撓材の深さは連続量を取り得るが，実際の設計寸法ではラウンドナンバーとなるのが普通である。したがって，ここで考えた設計変数は全て離散量であるといえる。また，ここで用いる制約条件および目的関数ともに設計変数に関して非線形である。本報告では初期設計段階における対話型の最適設計を考えているので，変数の探索範囲あるいは収束条件等の変更が容易に行える解法が望ましい。以上のことを考慮して，ここでは最適設計法としてモンテカルロ法を用いた。

モンテカルロ法では，乱数を発生させて設計変数を探索範囲内で無作為に定める。そしてその設計変数の値に対して目的関数（ここでは防撓板の体積）を計算する。これを繰返して目的関数の最小値の近似値およびそれを与える設計変数の値を求める。この方法は計算効率は良くないが，目的関数および制約条件の性質に関係なく用いることができる。プログラムの全体の流れを Fig. 8 に示す。

### 4.2 プログラム・コーディングの要点

プログラムの全体リストは付録に示してあるが，ここではそのコーディングにおける要点を Fig. 8 の流れに沿って述べる。乱数を発生するインスタンスを生成し，（0，1）の乱数を得るには次の命令を実行する。

```
rand←Random new.
rand next
```

前者は乱数のクラス Random にインスタンス作成のメッセージ new を送り，rand という名のインスタンスを作成することを意味する。後者によってインスタンス rand にメッセージ next が送られて乱数が発生する。また，区間 [i, j] の整数の乱数は次の命令から得られる。

```
rand next * (j−i) truncated+i
```

ここで truncated は小数の 0 方向への丸めを意味する。板厚の配列 thick が次のように与えられているとき，

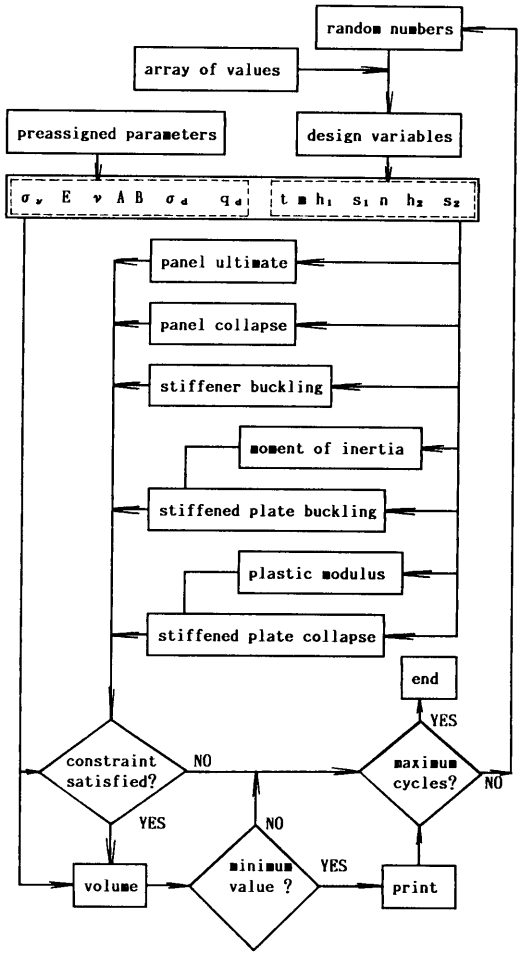


Fig. 8　Flow of the optimum design

thick＝(2.3　3.2　4.5　6.0　9.0)

乱数として3が得られたとすると，配列の3番目の値すなわち4.5がこのときの板厚となる。

パネル，防撓材，防撓板の座屈あるいは崩壊荷重を算定する手続きがそれぞれコーディングされて別々のメッセージ名を付けられる。部材寸法等の変数に具体的な数値を代入された防撓板のインスタンスにこれらのメッセージが送られて各荷重値が求められる。予め与えられた設計荷重に比べて，求められたこれらの各荷重が全て大きい場合にはそのときの体積が計算されるが，そうでない場合には次の乱数発生が行われる。体積が計算された場合には，その値がそのときまでの最小値と比較され，小さい場合には新たな最小値として設定されると同時にそのときの変数値とともにプリント出力される。その後は次の乱数発生のステップへと移る。

## 5.　プログラムの実行例

防撓板の全体寸法2種類についての計算例を示す。A＝5000mm，B＝5000mm および A＝10000mm，B＝5000mm の2ケースである。降伏応力（$\sigma_y$＝25kgf/mm²），ヤング率（E＝21000kgf/mm²），ポアソン比（$\nu$＝0.3），面内設計荷重（$\sigma_d$＝22kgf/mm²），面外設計荷重（$q_d$＝0.02kgf/mm²）についてはそれぞれ同じ値を用いた。目的関数としては防撓板の全体体積を A×B で割った次の値 vol を用いた。

$$vol = t + mh_1s_1/A + nh_2s_2/B \qquad (9)$$

### 5.1　ケース1（A＝5000mm，B＝5000mm の場合）

板厚 thick，防撓材の深さ height および防撓材の本数 stiffNumber についてそれぞれ次の配列を考えた。すなわち，thick＝(3.2　4.5　6　9　12　16　19　22　25　28　30　32　34　36)，height＝(25　50　75　100　125　150　175　200　225　250　275　300　325　350)，stiffNumber＝(0　1　2　3　4　5　6　7　8　9　10　11　12　13　14　15　16　17　18　19　20　21　22　23) である。発生した乱数によって，それぞれの配列の中のひとつの値が選ばれる。計算結果を Table 1 に示す。ここで，i は変数の一組みの値を乱数によって発生させた回数，t，m，$h_1$，$s_1$，n，$h_2$，$s_2$は Fig. 1 に示す部材寸法および防撓材本数，vol は式(9)で示される値である。step I では，変数の値の探索範囲は上述の配列内の全ての値であるが（ただし，m＝0～4，n＝11～23），step II～IVにおいては前の段階の step において得られた変数の値を参照し，探索範囲を狭くして計算効率を上げている。乱数発生回数 i とその回数における目的関数 vol の最小値との関係を Fig. 9 に示す。i が大きい範囲では vol の値の変化が非常に小さくなっていることから，実用上満足すべき値が得られていると考えられる。

### 5.2　ケース2（A＝10000mm，B＝5000mm の場合）

計算結果を Table 2 に示す。step I では前述のケース1における step I と同じ探索範囲を用いた。step I における m，および step II における $h_1$，$s_1$が探索範囲の大きい方の値にかたよっているのがわかる。そこで，それぞれ次の step ではこれらの変数についての探索範囲を大きい値の方へ移動させている。step III～IV では前段階の step の変数値を参照して，探索範囲を小

Table 1 The result of Case 1

| step | i | t | m | $h_1$ | $s_1$ | n | $h_2$ | $s_2$ | vol |
|---|---|---|---|---|---|---|---|---|---|
| I | 1 | 30 | 0 | 150 | 32 | 10 | 225 | 34 | 45.30 |
| | 10 | 28 | 4 | 125 | 16 | 16 | 150 | 22 | 40.16 |
| | 46 | 19 | 4 | 25 | 30 | 12 | 250 | 32 | 38.80 |
| | 92 | 16 | 1 | 225 | 34 | 17 | 175 | 32 | 36.57 |
| | 99 | 9 | 4 | 275 | 3.2 | 22 | 300 | 16 | 30.82 |
| | 147 | 6 | 4 | 175 | 22 | 20 | 275 | 19 | 29.98 |
| | 246 | 9 | 2 | 275 | 25 | 14 | 300 | 16 | 25.19 |
| | 476 | 6 | 3 | 350 | 30 | 19 | 300 | 9 | 22.56 |
| II | 477 | 9 | 2 | 225 | 25 | 14 | 300 | 12 | 21.33 |
| | 558 | 6 | 4 | 250 | 28 | 18 | 300 | 9 | 21.32 |
| | 577 | 6 | 2 | 200 | 28 | 20 | 325 | 9 | 19.94 |
| | 711 | 6 | 4 | 275 | 22 | 18 | 275 | 9 | 19.75 |
| | 743 | 6 | 2 | 225 | 25 | 18 | 325 | 9 | 18.78 |
| III | 809 | 6 | 4 | 275 | 25 | 19 | 300 | 6 | 18.34 |
| IV | 1029 | 6 | 4 | 275 | 25 | 20 | 275 | 6 | 18.10 |
| | 1142 | 6 | 4 | 250 | 25 | 19 | 300 | 6 | 17.84 |

Table 2 The result of Case 2

| step | i | t | m | $h_1$ | $s_1$ | n | $h_2$ | $s_2$ | vol |
|---|---|---|---|---|---|---|---|---|---|
| I | 3 | 36 | 4 | 325 | 30 | 11 | 325 | 22 | 55.63 |
| | 45 | 36 | 4 | 300 | 32 | 15 | 325 | 9 | 48.62 |
| | 359 | 25 | 4 | 325 | 34 | 19 | 275 | 16 | 46.14 |
| II | 505 | 19 | 5 | 350 | 28 | 10 | 325 | 28 | 42.10 |
| | 521 | 22 | 5 | 350 | 36 | 10 | 250 | 25 | 40.80 |
| | 548 | 19 | 6 | 325 | 28 | 15 | 275 | 19 | 40.14 |
| III | 556 | 19 | 8 | 325 | 32 | 8 | 250 | 19 | 34.92 |
| | 639 | 16 | 4 | 425 | 32 | 11 | 275 | 19 | 32.94 |
| | 715 | 19 | 5 | 375 | 42 | 8 | 275 | 12 | 32.16 |
| | 735 | 19 | 4 | 425 | 38 | 8 | 325 | 12 | 31.70 |
| | 758 | 12 | 4 | 425 | 32 | 10 | 350 | 16 | 28.64 |
| | 790 | 12 | 6 | 400 | 34 | 14 | 300 | 9 | 27.72 |
| | 795 | 9 | 5 | 375 | 42 | 12 | 250 | 16 | 26.48 |
| | 816 | 9 | 4 | 425 | 40 | 14 | 350 | 9 | 24.62 |
| | 949 | 9 | 5 | 400 | 42 | 12 | 275 | 9 | 23.34 |
| IV | 979 | 9 | 6 | 375 | 42 | 13 | 275 | 6 | 22.74 |
| | 1150 | 9 | 6 | 400 | 36 | 13 | 275 | 6 | 21.93 |

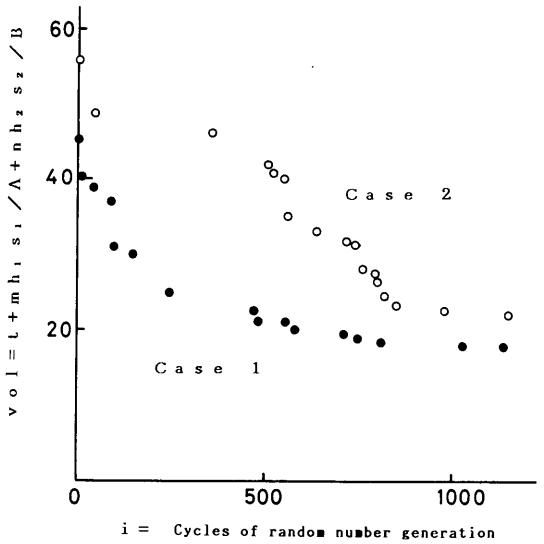


Fig. 9 Relation between minimum volume and cycle numbers

さくしている。回数 i と最小値 vol との関係を Fig. 9 に示したが，ケース 1 の場合を参照して，この場合にも実用上満足すべき最小値が得られていると考えられる。

## 6. あとがき

オブジェクト指向言語 SMALLTALK によって，防撓板に関してモンテカルロ法を用いた最適設計法の検討を行った。SMALLTALK は手続きとデータとの組合せから成るオブジェクトというものをプログラムの基本単位としている。乱数の発生，変数への値の代入，各種の強度基準の計算，制約条件の判定等はそれぞれ手続き群として別々のプログラム単位を構成する。防撓板の寸法を表すデータは各手続きで共通に用いることができる。モンテカルロ法による最適設計は，目的関数および制約条件の性質に無関係に用いることができる。本研究におけるプログラムの作成過程で得られた知見をまとめると次のようになる。

(1) SMALLTALK ではプログラム単位が小さいので，その修正，追加等が容易に行える。また，対話型を基本としているので，設定パラメータ，変数の探索範囲等についての変更を，計算結果を参照しつつ行うことができる。

(2) モンテカルロ法による最適設計は計算効率はあまり良くないが，初期設計の段階では実用上満足できる値を得ることができる。

現在発展段階にある知識処理技術の適用範囲は広い。そして，この技術をいかに利用するかは，それぞれ各分野に属する技術者に課せられた命題であると思う。それと関連して，構造解析/設計の分野でも，今までの analysis を統合する structural synthesis に関する研究がいっそう重要になってくると考えられる。

SMALLTALK によるプログラミングおよび計算の実行はテクトロニクス TEK 4406 を用いて行った。また，本研究は科学技術庁の科学技術振興調整費による重点基礎研究課題として行ったものである。

## 参 考 文 献

1) 青木元也：LISP による対話型構造解析，昭和62年度春季船舶技術研究所研究発表会講演集，昭和62年.

2) 青木元也：プロダクション・システム OPS 5 を用いた構造解析支援システム，船舶技術研究所報告，第25巻第 5 号，昭和63年.

3) Adele Goldberg and David Robson : SMALLTALK - 80 The language and its implementation, Addison - Wesley Publishing Company, 1985.

4) Douglas Faulkner : A Review of Effective Plating for Use in the Analysis of Stiffened Plating in Bending and Compression, Journal of Ship Research, Vol. 19, No. 1, 1975.

5) 楠田忠雄：垂直荷重を受ける防撓板の塑性設計，造船協会論文集，第110号，昭和36年.

6) 長柱研究委員会：弾性安定要覧，コロナ社，昭和44年.

7) A. E. Mansour : Approximate formula for preliminary design of stiffened plates, Proc. 5th Int. OMAE, 1986.

8) 長沢準，青木元也：横荷重を受ける防撓板の塑性強度について，造船協会論文集，第114号，昭和38年.

## 付録　プログラムのリスト

```
Class name : StiffenedPlate
Instance variable names : sA  sB  p  q  yieldP
    youngM  pois  thick  height  stiffNumber
    t  h1  s1  h2  s2  m  n
(instance method)
setVariable
  sA ← 5000.
  sB ← 5000.
  p  ← 22.
  q  ← 0.02.
  yieldP ← 25.
  youngM ← 21000.
  pois ← 0.3
setArray
  thick  ← Array new : 14.
  height ← Array new : 14.
  stiffNumber ← Array new : 24.
  thick  ← #(3.2  4.5  6  9  12  16  19  22  25  28  30  32  34  36)
  height ← #(25  50  75  100  125  150  175  200  225  250  275  300  325  350)
  stiffNumber ← #(0  1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22
                  23)
randomNum : rand
  t  ← thick at : (rand next * 14) truncated+1.
  s1 ← thick at : (rand next * 14) truncated+1.
  s2 ← thick at : (rand next * 14) truncated+1.
  h1 ← height at : (rand next * 14) truncated+1.
  h2 ← height at : (rand next * 14) truncated+1.
  m  ← stiffNumber at : (rand next * 14) truncated+10.
  n  ← stiffNumber at : (rand next * 5) truncated+1
panelUltimate
  | beta  ultimateP  b |
  b  ← sB/(n+1).
  beta ← b/t * ((yieldP/youngM)sqrt).
  beta>1 ifTrue :  [↑ultimateP ← yieldP * (2/beta−(1/(beta raisedToInteger : 2)))]
         ifFalse : [↑ultimateP ← yieldP]
panelCollapse
  | beta  temp  collapseQ  a  b |
  a  ← sA/(m+1).
  b  ← sB/(n+1).
  beta  ← a/b.
  temp  ← (3 * (beta raisedToInteger : 2)+1)sqrt.
  temp  ← (temp+1)/(temp−1).
  ↑collapseQ ← 2 * (t raisedToInteger : 2) * yieldP * temp/(b raisedToInteger : 2)
```

```
stiffBuckling
    | pi  sh  bucklingP  a |
  a  ← sA/(m+1).
  pi ← 3.14159.
  sh ← s2/h2.
  ↑ bucklingP ← (0.42+(h2/a)) * youngM * (pi raisedToInteger : 2)/(12 * (1-(pois raisedToInteger : 2)))
mtOfInertia : ab
    | e1  e2  h3  mtI  w  h  s |
  ab =$a ifTrue : [h ← h1. s ← s1. w ← sA/(m+1)].
  ab =$b ifTrue : [h ← h2. s ← s2. w ← sB/(n+1)].
  e2 ← s * (h raisedToInteger : 2)+(2 * s * h * t)+(w * (t raisedToInteger : 2))/(2 * (s * h+(w * t))).
  e1 ← h+t-e2.
  h3 ← e2-t.
  ↑ mtI ← (w * (e2 raisedToInteger : 3)-((w-s) * (h3 raisedToInteger : 3))+(s * (e1 raisedToInteger :
          3)))/3
stiffPlateBuckling
    | a  b  aI  bI  bSec  temp  temp2  bucklingP |
  temp ← 4 * (3.14159 raisedToInteger : 2) * youngM/(1-(pois raisedToInteger : 2)).
  b  ← sB/(n+1).
  bSec ← h2 * s2.
  temp ← temp/sB/(sB * t+(n * bSec)).
  bI ← self mtOfInertia : $b.
  temp ← temp * ((bI/b) sqrt).
  ((sA<sB) | (m=0))
          ifTrue : [temp 2 ← (t/2) * (t/3 sqrt)]
          ifFalse : [aI ← self mtOfInertia : $a.
                     a ← sA/(m+1).
                     temp 2 ← ((aI/a) sqrt)].
  ↑ bucklingP ← temp * temp 2
plasticModulus : ab
    | h  s  w  temp  e1  e2  z |
  ab =$a ifTrue : [h ← h1. s ← s1. w ← sA/(m+1)].
  ab =$b ifTrue : [h ← h2. s ← s2. w ← sB/(n+1)].
  temp ← h * s-(w * t).
  temp > 0
  ifTrue : [e2 ← h/2+t-(w * t/(2 * s)).
            e1 ← h+t-e2.
            ↑ z ← t * (w-s) * (e2-(t/2))+(s * (e2 raisedToInteger : 2+(e1 raisedToInteger : 2))/2)]
  ifFalse : [e2 ← s * h/w+t/2.
             e1 ← h+t-e2.
             ↑ z ← w * ((t-e2) raisedToInteger : 2+(e2 raisedToInteger : 2))/2+(s * h * (e1-(h/2)))]
stiffPlateCollapse
    | az  bz  aMp  beta  a  b  temp  collapseQ |
  az ← self plfsticModulus : $a.
  bz ← self plasticModulus : $b.
```

```
  a ← sA/(m+1).
  b ← sB/(n+1).
  aMp ← (m * az/sA+(t raisedToInteger : 2)/4) * yieldP.
  bMp ← (n * bz/sB+(t raisedToInteger : 2)/4) * yieldP.
  beta ← (sA/sB) * ((aMp/bMp)sqrt).
  beta > 1
    ifTrue : [temp ← (3 * (beta raisedToInteger : 2)+1)sqrt.
              ↑collapseQ ← 8 * aMp * (temp+1)/(sB raisedToInteger : 2)/(temp−1)]
    ifFalse : [beta ← 1/beta.
               temp ← (3 * (beta raisedToInteger : 2)+1)sqrt.
               ↑collapseQ ← 8 * bMp * (temp+1)/(sA raisedToInteger : 2)/(temp−1)]
constraintOk
  | pUlt pCol sBuck spBuck spCol |
  pUlt ← self panelUltimate.
  pCol ← self panelCollapse.
  sBuck ← self stiffBuckling.
  spBuck ← self stiffPlateBuckling.
  spCol ← self stiffPlateCollapse.
  ((pUlt > p)&(pCol > q)&(sBuck > p)&(spBuck > p)&(spCol > q))
            ifTrue : [↑true]
            ifFalse : [↑false]
volume
  | aSec bSec vol |
  aSec ← h1 * s1.
  bSec ← h2 * s2.
  ↑vol ← t+(m * aSec/sA)+(n * bSec/sB)asFloat
searchMin
  | i j k rand vol volMin |
  i ← j ← k ← 0.
  volMin ← 1000.0.
  rand ← Random new.
  self printHeader.
  [i>500]whileFalse :
    [self randomNum : rand. i ← i+1. (self constraint0k) ifTrue : [vol ← self volume. j ← j+1. (vol<
     volMin) ifTrue : [k ← k+1. volMin ← vol. self printi : i j : j k : k vol : vol]]]
printHeader
  Transcript cr ; show : 'i' ; tab ; show : 'j' ; tab ; show : 'k' ; tab ; show : 't' ; tab ; show : 'm' ; tab ; show : '
  h1' ; tab ; show : 's1' ; tab ; show : 'n' ; tab ; show : 'h2' ; tab ; show : 's2' ; tab ; show : 'vol' ; cr
printi : i j : j k : k vol : vol
  Transcript cr ; show : i printString ; tab ; show : j printString ; tab ; show : k printString ; tab ; show : t
  printString ; tab ; show : m printString ; tab ; show : h1 printString ; tab ; show : s1 printString ; tab ; show :
  n printString ; tab ; show : h2 printString ; tab ; show : s2 printString ; tab ; show : vol printString
(class method)
create
  | aStiffPlate |
```

```
    aStiffPlate ← super new.
    aStiffPlate setVariable.
    aStiffPlate setArray.
    ↑ aStiffPlate
(execution of program)
    | aStiffPlate |
    aStiffPlate ← StiffenedPlate create.
    aStiffPlate searchMin
```