

遺伝的アルゴリズムを用いた板金構造物の展開図作成手法

外山 眞也
宮崎県工業技術センター

富田 重幸
宮崎大学

吉富 康成
京都府立大学

春成 嘉弘
サイバーテック (株)

(受理 2000 年 2 月 2 日 ; 再受理 2000 年 12 月 22 日)

和文概要 流体の流れを制御するダクト等の筒型板金構造物の設計製作においては、展開図の作成が重要な工程となっているが、その展開形状のパターンは多様である。しかし、現在利用可能な展開図作成用 CAD システムは、いくつかの展開形状のパターンに対応したプログラムを提供しているだけで、対象構造物の形状がわずかに変化しただけで対応できない場合が多い。

そこで、本研究では展開形状の入口と出口の二つの断面形状のそれぞれを点群で分割近似表現し、単一のアルゴリズムで、多種多様な形状の展開図にも対応できる柔軟な CAD システムを開発した。手法として、二つの断面形状における点群間の結線決定問題を最適化問題として定式化した。ここで、曲げ加工線として取り扱う結線の長さの総和と、形成された側面のなめらかさを表す新たに定義した関数を加重和して作成した目的関数を用いた。側面形成の条件は、制約条件として取り扱った。そして、その近似最適解を得る方法として、遺伝的アルゴリズム (GA) を利用したアルゴリズムを提案した。このとき、断面間の点群の結線状態を示すコード化に「隣接制約パス表現」を提案した。このコード化により、探索効率が著しく向上した。

開発した CAD システムの検証実験の結果、複雑な形状の構造物に対しても実用的な展開図を生成できることを確認した。

1. はじめに

空気や流体などの流れの方向を制御するダクトや配管の形状は多種多様で、また入口の角形状から出口の円形状へと断面形状の変化が要求されたりする。さらには、入口から出口まで三次元的に様々な方向へ形状を変える場合もある。このような複雑な形状となるため、現状の CAD システムでは対応できず、現場では経験やカンに頼って作業を行うことが多く非能率的であるといえる[8]。

現在、利用可能な展開図作成用 CAD システムは、いくつかの展開形状のパターンに対応したプログラムを提供しているだけである。しかし、現実に必要な板金構造物の形状パターンは多種類に及ぶので、わずかな形状の変化にも対応できないことが少なくない。また、製品形状を入力すると曲げ部分を自動的に展開するシステムもあるが[5]、二つの断面間をどのような側面で構成するのが適正なのかを検討されていないのが現状である。しかし、すべての形状パターンに対応したプログラムを従来の個別対応的な手法で開発することは困難である。さらに、多種多様な板金構造物を対象としなければならない業界では、その展開図設計段階で多くの時間を要し、合理化も進められないのが現状である。実際、展開板取の現場作業では、比較的単純な一つの展開図の作成にも数時間を要することが多く、複雑な形状を対象とする場合数日の作業となることも多い。

さらに、展開図自動生成に関しての研究報告例はきわめて少なく、わずかに各種箱形形状の展開図を求める手法[2][4]について報告されている程度であり、その汎用

性は少ない。また、二つの相異なった断面形状を基に、その側面を三角形（単体）平面群で包絡処理する理論も未だ整備されていない状況である。

そこで、本研究では、展開図作成のための結線決定問題を最適化問題として定式化し、上面と下面の二つの断面形状を与えさえすれば、断面間の結線と構造物側面の形状の生成を自動的に行い、その構造物における最適な展開図をも生成する手法を、近年数理計画問題の近似解法に利用されている遺伝的アルゴリズム（GA）[1,3,6,7]を用いて開発した。

2. 対象とする問題

気体や液体などの流体の流れを制御するダクト等の板金構造物には、必ず流体の入口と出口がある。このような板金構造物に関しては、その入口と出口をつなぐ側面を三角形平板で構成し直し、平面に広げることによりその展開図を求めることができる。そして、平板形状のものを折り曲げ加工した板金を組み合わせて、結線部を溶接して板金構造物を製作している。

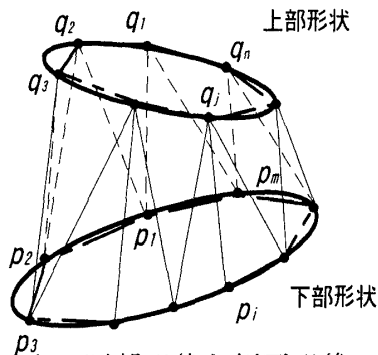


図1 上，下部形状と側面形状の例

表1 結線状態を示すマトリックス X
上断面形状の点群

	1	2	3	4	5	...	n
1	1	1	0	0	0	0	0
2	0	1	1	0	0	0	0
3	0	0	1	0	0	0	0
4	0	0	1	1	0	0	0
5	0	0	0	1	0	0	0
6	0	0	0	1	1	0	0
...	0	0	0	0	1	...	1
m	1	0	0	0	0	0	1

ることになる。そして、各結線は、隣り合う二つの三角形平板の辺として共有される線分となるため、生成される結線の数には三角形平板の数と同一で $m+n$ となる。ただし、上側断面形状の隣り合う2点を結ぶ線分と、対する下側断面形状の隣り合う2点を結ぶ線分が平行であり、この二つの線分を一つずつ辺として含む隣接する三角形平板が二つ生成される場合は、この限りではない。このような場合、二つの三角形平板ではなく一つの四角形として展開図を作成する。

本研究の目的は、結線状態を評価する規準を提案し、その規準を基に結線状態を最適化する問題を数理計画問題として定式化し、その解法を開発することにある。

3. 問題の定式化

展開図作成において、結線は短いほど曲げ加工線の長さは短くなり(曲げ加工部の

そこで、その流体が流れる入口と出口の断面形状を定義し、その断面間の側面を構成する複数の三角形平板を生成することを考える。このとき、各断面形状は点群で直線近似された形状とする。

図1に対象とする形状例を示す。点群の作成方法については、付録1に記す。

下部形状を表現する m 個の点を p_i ($i=1, \dots, m$)、上部形状を表現する n 個の点を q_j ($j=1, \dots, n$) としたとき、結線する線の本数は、 $m+n$ 個となり、結線状態は表1の行列 $X=(x_{ij})$ で表現できる。この表は、上部形状の点群を行、下部形状の点群を列方向に並べている。そして、その対応する2点を結線する場合は行列要素が“1”となる。

そして、一つの断面形状の点のうちの1点と、もう一方の断面形状の点のうちの隣り合う2個の点とのあわせて3点で一つの三角形平板が定義され、全部で $m+n$ 個の三角形平板が生成さ

面積は小さく), 加工の点で好ましい. また, 側面の形状がいびつなものとなることは外観上好ましくない. そこで, 展開図の結線決定問題をこの二つの規準を基にした最適化問題として定式化する.

3.1. 目的関数

3.1.1. 結線長さの総和

二つの断面形状における各節点間の距離 $l_{i,j}$ と, 結線状態を示した行列 X の要素 $x_{i,j}$ により, 結線長さの総和 f_l は,

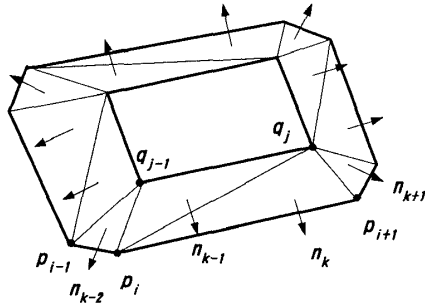


図2 生成された側面の法線ベクトルと結線

$$f_l = \sum_{i=1}^m \sum_{j=1}^n l_{i,j} \cdot x_{i,j} \quad (1)$$

となる. ただし, ここで, 図2に示すように線 $p_i p_{i+1}$ と線 $q_{j-1} q_j$ が平行であるとき, この各線分を構成要素とする各側面三角形の二つの単位法線ベクトル $\mathbf{n}_{k-1}(X)$ と $\mathbf{n}_k(X)$ は一致するので, 二つの側面三角形をあわせて一つの四角形平面として取り扱う. このため, 2点 $p_i q_j$ を結線する線の長さ $l_{i,j}$ は加工曲げ線としては扱わない. 従

って, この場合, 結線長さの総和の計算の際に, この結線部分の長さ $l_{i,j}$ を加算しないこととする.

3.1.2. 生成された側面の法線ベクトルによる外観の評価

図1のように, 2点 p_i と q_j を結ぶ線が生成されたとき, 3点 q_j, p_i, p_{i-1} ができる側面と3点 q_j, p_i, p_{i+1} ができる側面の法線ベクトルをそれぞれ求めることができる. この法線ベクトルが隣接する側面間でなめらかに変化するかどうか評価する以下の関数 f_v を用いて外観の評価を行う.

$$f_v = \sum_{k=1}^{m+n} w_k(X) \cdot |\Delta \theta_k(X)| \quad (2)$$

ここで,

$$w_k(X) = \frac{S_k(X)}{\sum_{k=1}^{m+n} S_k(X)}$$

$$\theta_k(X) = \tan^{-1} \left(\frac{n_{kY}(X)}{n_{kX}(X)} \right)$$

$$\Delta \theta_k(X) = \theta_k(X) - \frac{\theta_{j_k}(X) + \theta_{l_k}(X)}{2}$$

$$\begin{cases} j_k = k-1, l_k = k+1 & (n_k \neq n_{k-1}, n_k \neq n_{k+1}) \\ j_k = k-2, l_k = k+1 & (n_k = n_{k-1}, n_k \neq n_{k+1}) \\ j_k = k-1, l_k = k+2 & (n_k \neq n_{k-1}, n_k = n_{k+1}) \end{cases}$$

$$\text{但し } n_{-1} = n_{m+n-1}, n_0 = n_{m+n}, n_{m+n+1} = n_1, n_{m+n+2} = n_2$$

図3-aに示すよう, 生成された k 番目の側面三角形における面積を $S_k(X)$, その側面三角形の単位法線ベクトルを $\mathbf{n}_k(X)$ とし, このベクトル $\mathbf{n}_k(X)$ の X 成分 $n_{kX}(X)$ 及び Y 成分 $n_{kY}(X)$ を使って, 角度 $\theta_k(X)$ を求める(図4). そして, 偏差 $|\Delta \theta_k(X)|$ の加重和 f_v を求め, 隣接する側面間での単位法線ベクトルの変化の平滑性の評価に基づく外観の評価基準とした. ここで, 各側面の単位法線ベクトルを求めるときに,

下断面形状の重心から上断面形状の重心へ向かうベクトルを三次元直交座標 X - Y - Z の Z 軸に一致するように座標変換している. そのため, 隣接する三角形側面間の法線ベクトルの変化の平滑さについては, 通常の板金構造物では各側面の法線ベクトルの Z 成分の変化は無視できるので, 偏差 $|\angle \theta_k(X)|$ は単位法線ベクトル $\mathbf{n}_k(X)$ の, X 成分 $n_{kX}(X)$ と Y 成分 $n_{kY}(X)$ のみを用いて算出している. ただし, $\angle \theta_k(X)$ を求めるときに, 通常 $j_k = k-1$, $l_k = k+1$ であるが, 隣接する三角形が同一平面内となる場合, つまり側面の法線ベクトル $\mathbf{n}_k(X)$ と $\mathbf{n}_{k-1}(X)$ が一致する場合(図 3-b)は $j_k = k-2$, 同様に $\mathbf{n}_k(X)$ と $\mathbf{n}_{k+1}(X)$ が一致する場合は $l_k = k+2$ となる. この場合, 隣接する二つの三角形側面は, 一つの四角形平面として展開できる. $\angle \theta_k(X)$ が 0 に近い程, 隣接する側面間で単位法線ベクトルがより平滑に変化すると判断できる. そして, (2)式では, 側面の面積が大きいほど外観への影響が大きいという判断から加重 $w_k(X)$ を導入している. ここで, $S_k(X)$, $w_k(X)$, $\angle \theta_k(X)$ は, 結線状態を表す行列 $X=(x_{i,j})$ の各要素を変数とする関数となっている.

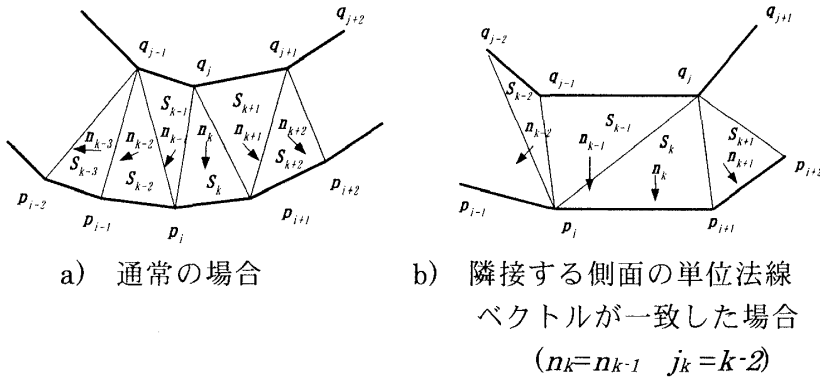


図3 生成側面の法線ベクトルによる判定

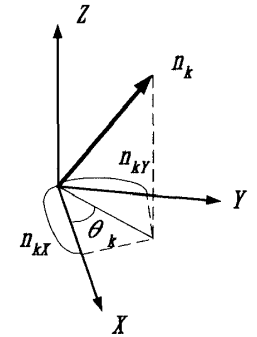


図4 角度 $\theta_k(X)$

3.1.3. ウェイト法を用いた目的関数の作成

上記結線長さの総和 f_l と側面の外観の評価関数 f_v の加重和をとり, 目的関数を作成した.

3.2. 制約条件

下断面形状を表現する点数を m , 上断面形状を表現する点数を n とするとき, 展開図作成可否を考慮しなければ, 断面間の各点を結ぶ線の数の上限は, mn となる. 一方, 展開図を生成するのに必要な結線数は $m+n$ (制約条件1) であり, 下断面形状の線分を三角形側面の一辺とする側面の数は m (制約条件2), 上断面形状の線分を三角形側面の一辺とする側面の数は n (制約条件3) となる. また, ある断面形状の1点と相対する断面形状の隣り合う2点を結んでできる三角形平板は, 最大で $2mn$ 個まで生成することが可能であるが, その三角形平板は, その二つの断面間を結んで側平面を構成する要素とはならない場合がある. そこで, その不適当な構成要素をあらかじめ削除するため行列 C (制約条件4で使用) を以下の式で求める.

$$c_{i,j} = a_{i,j,+1} + a_{i,j,-1} + b_{i,j,+1} + b_{i,j,-1} \quad (\text{ブール代数演算})$$

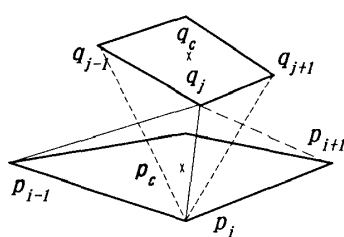
$$a_{i,j,\pm 1} = \begin{cases} 1, (z_{i,j,p,a,\pm 1} \cdot z_{i,j,q,a,\pm 1} \geq 0) \\ 0, (z_{i,j,p,a,\pm 1} \cdot z_{i,j,q,a,\pm 1} < 0) \end{cases}, \quad b_{i,j,\pm 1} = \begin{cases} 1, (z_{i,j,p,b,\pm 1} \cdot z_{i,j,q,b,\pm 1} \geq 0) \\ 0, (z_{i,j,p,b,\pm 1} \cdot z_{i,j,q,b,\pm 1} < 0) \end{cases}$$

$$(i=1, \dots, m), (j=1, \dots, n) \quad (\text{複号同順})$$

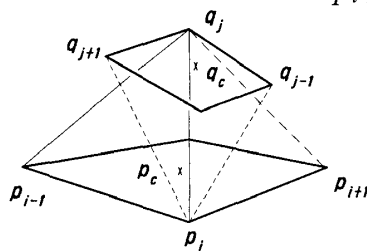
上断面形状を表すある点から下断面形状の隣り合う2点を選択して形成される三

角形平板と、上断面の中央点(q_c)と下断面形状の中央点(p_c)との位置関係をあらかじめ調べ、形成された三角形平板を含む平面からみて q_c と p_c が同じ方向にない場合は、側面を構成する三角形平板としては不適当であると判定するようにした(図5). 同様に、下断面形状を表す点から、上断面形状の隣り合う2点を選択して形成される三角形平板と上下断面形状の中央点 q_c , p_c との位置関係を調べた(図5). なお、下断面を含む平面を $X'-Y'$ 平面とする $X'-Y'-Z'$ 三次元直交座標系において、各断面形状の各点の X', Y', Z' 座標における最大最小の値を求め、その各座標の平均値を X', Y', Z' 座標とする点として中央点を、あらかじめ求めておく.

$a_{i,j,\pm 1}$ は、点 p_i と q_j を結ぶ線を一辺とする三角形 ($p_i, q_j, q_{j\pm 1}$) (複号同順) を作り、

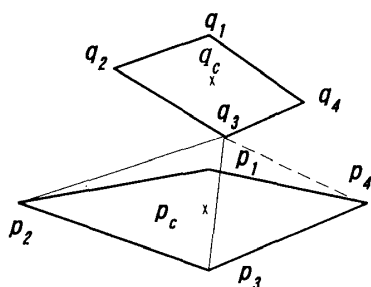


$c_{i,j} = 1$

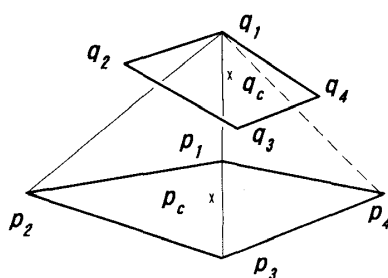


$c_{i,j} = 0$

図5 三角形側面の展開可能性の判定



a) 三角形平板が側面として
適正な場合



b) 三角形平板が側面として
適正でない場合

図6 生成された側面三角形平板の展開適性判定

その三角形が側面として適正かどうかの判定結果を表す 0,1 定数である (0:不適, 1:適). このとき、その三角形を含む平面を $x-y$ 平面とし、法線ベクトルが z 軸方向になるように座標変換したときに、下部形状の中央点 p_c , 上部形状の中央点 q_c の z 座標を各々 $z_{i,j,p,a,\pm 1}$, $z_{i,j,q,a,\pm 1}$ とし、 $a_{i,j,\pm 1}$ を求めた. また、同様に $b_{i,j,\pm 1}$ は、点 p_i と q_j を結ぶ線を一辺とする三角形 ($p_i, p_{i\pm 1}, q_j$) (複号同順) での判定結果を表す, 0,1 定数である.

表2 4×4 の場合の
適正判定行列 C

		上断面形状の点群			
下断面形状の点群		1	2	3	4
	1	1	1	0	1
	2	1	1	1	0
	3	0	1	1	1
	4	1	0	1	1

例えば、上、下部断面形状を表現する点とともに4点の場合(図6)において、生成された三角形平板が側面として適正かどうかを判定した結果、表2のような行列 C ができる. この行列では図6-bで示すように、 p_3 と q_1 を結ぶ線を三角形の一辺とする平板(三角形 $p_3q_1p_2$, $p_3q_1p_4$, $p_3q_1q_2$, $p_3q_1q_4$) は側面として適正でないことを $c_{3,1} = 0$ で表現する.

この側面適性判定定数 $c_{i,j}$ の提案は、展開図作成時の経験則に基づいており、すべての展開図作成において有効であるとは言えない. 実際、凹型断面形状のように中央点とその断面形状内部に存在しない特殊な場合は、この側面の適性判定は有効でない.

3.3. 最適化問題

下部形状を表現する点と上部形状を表現する点において、点間距離が最小となる点を、それぞれ p_1 , q_1 として、各点 p_i , q_j 間の距離 $l_{i,j}$ をあらかじめ求め、行列 $L=(l_{i,j})$ を作成した. この板金構造物の展開図の結線決定問題を、結線状態を表す行列

$X=(x_{i,j})$ の各要素の値を決定する 0-1 計画問題として、以下のように定式化する。

$$\text{Minimize} \quad \alpha_0 \sum_{i=1}^m \sum_{j=1}^n l_{i,j} \cdot x_{i,j} + \beta_0 \sum_{k=1}^{m+n} w_k(X) \cdot |\Delta \theta_k(X)| \quad (3)$$

$$\text{Subject to} \quad \sum_{i=1}^m \sum_{j=1}^n x_{i,j} = m + n \quad (4)$$

$$\sum_{i=1}^m \sum_{j=1}^n x_{i,j} \cdot x_{i+1,j} = m \quad (5)$$

$$\sum_{i=1}^m \sum_{j=1}^n x_{i,j} \cdot x_{i,j+1} = n \quad (6)$$

$$\sum_{i=1}^m \sum_{j=1}^n c_{i,j} \cdot x_{i,j} = m + n \quad (7)$$

$$x_{i,j} \in \{0,1\}$$

$$(x_{i,n+1} = x_{i,1}, x_{m+1,j} = x_{1,j})$$

目的関数の第一項は結線長さの総和を表し、第二項は側面の形状の外観を評価する関数である。 α_0, β_0 は正または0の定数であり、二つの目的関数の加重である。 $w_k(X), \Delta \theta_k(X)$ は結線状態で決まるため、行列 X の各要素の関数となる。 $S_k(X)$ は生成された三角形側面の面積、 $n_{kX}(X), n_{kY}(X)$ は下部断面形状の重心から上部断面形状の重心へ向かうベクトルを三次元直交座標 X - Y - Z の Z 軸に一致させたときの側面の単位法線ベクトル $n_k(X)$ の X, Y 成分である。

ここで、(4)は結線数の制約条件(3.2. における制約条件 1)、(5)及び(6)は結線で側面が形成されるための条件(3.2. における制約条件 2 及び 3)、(7)は各三角形で側面が形成できるための制約条件(3.2. における制約条件 4)である。これらの制約条件を満たすことにより側面は上下断面形状のどちらか一方の一点と他方の隣り合う2点との三点で決定される三角形で構成される。 $c_{i,j}$ は、生成された側面が展開図の側面として適正かどうかをあらかじめ判定した 0,1 定数である。

4. GA の適用

上記定式化された最適化問題の近似解法として、GA の適用を試みる。

4.1. コード化

行列 X が表現型の個体となる。そこで要素 $x_{i,j}$ の"1"の並びに着目し、第一行での"1"の開始位置 j と、 $x_{1,j}$ からはじめて"1"の位置を追跡することにより遺伝子型の個体を作成した。ここで個体の染色体のうち、"1"の位置を追跡する部分では、遺伝子の値を列移動（配列で右への移動）の場合が1、行移動（配列で下方向への移動）の場合に0とし、上、下部形状の点数 $m+n$ 個のビット列で行列 X における"1"の並びを表した。このことにより、順序よく断面間の結線を表現することができる。この表現方法を「隣接制約パス表現」と呼ぶこととする。

コード化の例を表3に、その遺伝子型個体の作成手順を図7に示す。下断面形状を表現する4個の点群と上断面形状を表現する4個の点群に対する染色体は、位置を示す2ビットと行列 X における"1"の配置を示す8ビットの和10ビットのデータとなる。はじめの2ビットは、第一行の何列目から結線を始めているかを示すデ

表3 コード化の例

表現型の個体 :	$\begin{bmatrix} 1,1,0,0 \\ 0,1,1,0 \\ 0,0,1,1 \\ 1,0,0,1 \end{bmatrix}$
遺伝子型の個体 :	0010101010

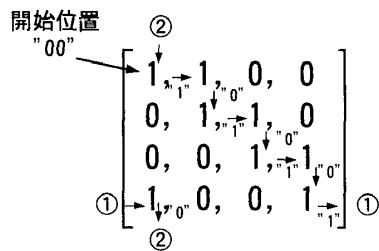


図7 遺伝子型個体作成手順

一タで、この例では上断面形状の点数が4個なので、開始位置は0から3までの数値で表現できる。

そのため、2ビットの領域を確保している。そして、その2ビットのデータ"00"は、第一列目から結線が始まっていることを示している。この「隣接制約パス表現」において、データの一意性を保つために、第一行の開始位置は、"1"の内、最左端位置にあるものを起点として行うようにした。例えば、図7で示す行列 X に対する遺伝子型の個体は、"0010101010"と"0101010101"の二通りとなるが、一意性の確保のために、"0101010101"は"0010101010"に変更される。この処理は適応度を求める時点で行った。

下断面形状の点が m 個、上断面形状の点が n 個の場合、この「隣接制約パス表現」により $m+n$ 個の結線を決定するすべての組み合わせの数が ${}_m C_{m+n}$ から ${}_n C_{m+n}$ に減少される。

4.2. 適応度

上断面形状上の点 q_j と下断面形状上の点 p_i との距離 l_{ij} を要素とする行列を $L=(l_{ij})$ とすると、各点間を結ぶ線分の長さの総和 f_l を(1)式で求める。また、生成された三角形側面の単位法線ベクトルの変化を基にした外観の評価関数 f_v を(2)式で求める。そして、目的関数の逆数を適応度とし、適応度を

$$f = \frac{1}{\left(\alpha \frac{f_l}{f_{lave}} + \beta \frac{f_v}{f_{vave}} \right)}$$

(f_{lave} は初期集団の個体の f_l の平均値)

(f_{vave} は初期集団の個体の f_v の平均値)

として求めた。ここで、 $\alpha, \beta (\geq 0)$ は二つの評価値に対しての加重で、定数である。二つの評価値を初期集団での値で規格化しているのは、二つの評価値の大きさに大きな差がある場合でも、本法が適用できるようにするためである。ここで、 α, β は(3)式における α_0, β_0 と以下の関係がある。

$$\alpha_0 = \frac{\alpha}{f_{lave}}, \quad \beta_0 = \frac{\beta}{f_{vave}}$$

4.3. 選択, 交差, 突然変異

選択方法として、ルーレット戦略を用い、一点交叉、一点突然変異を行った。交叉や突然変異などにより制約条件を満足しなくなった場合、致死遺伝子とした。

4.4. 計算処理アルゴリズム

展開図作成のための曲げ加工線の配置問題を解く計算処理は、次のような流れで実施している。

Step 0 : データの読み込みと定数諸量の計算。

- (1) 上下断面形状の点群の座標の読み込み.
- (2) 上下断面形状の点群の中央点の座標を求める.
- (3) 各点間距離の計算及び行列 L の作成.
- (4) 三角形平板の側面形成可否判定行列 C の作成.

Step 1 : 初期集団の生成.

Step 2 : 適応度の計算.

Step 3 : 選択.

Step 4 : 交叉.

Step 5 : 突然変異.

Step 6 : 終了処理.

最高適応度の値が所定の世代数更新されなければ終了.

または, 世代交代回数が設定値に達したら終了.

それ以外は, Step 2 へ戻る.

4.5. GA の各種条件の検討

4.5.1. 加重 α , β と結線状態との関係

本研究では, 二つの評価関数の加重 α , β は, 対象に応じて決定すべきパラメータとして位置づけている. ここでは, 適応度関数における加重 α , β と最終的に決定される結線状態の関係を典型的な例をもとに検討した. このとき, 交叉率を 0.2, 突然変異率を 0.04 とした. そして, 具体的には, まず下断面形状の点 44 点, 上断面形状の点 24 点の場合について調べた. 世代交代数を 20000 回とし, 最高適応度が 5000 の世代数で更新されなかったときは処理を終了することにして, 各例での適応度を求めた結果を表 4 に示す. α は結線長さの総和 f_l の係数, β は側面の滑らかさ f_v の係数, $MaxFit$ は GA での最終的な最高適応度の値である. このうち, a) $\alpha=0.0, \beta=1.0$, b) $\alpha=0.1, \beta=0.9$, c) $\alpha=0.9, \beta=0.1$, d) $\alpha=1.0, \beta=0.0$ の場合の結線状態を図 8 に示す. このとき, 結線長さの総和だけを評価すると, ほぼ同等の長さのデータが多いために, 最高適応度の収束が悪かった. また, この形状においては, ($\alpha=0.1, \beta=0.9$) または ($\alpha=0.2, \beta=0.8$) の条件が f_l , f_v とも低い値となった.

表 4 加重 $\alpha \beta$ と GA の処理結果(その 1)

α	β	$MaxFit$	f_l	f_v
0.0	1.0	20.6381	3552.8	0.0129
0.1	0.9	7.9050	3449.7	0.0129
0.2	0.8	5.3493	3449.2	0.0104
0.3	0.7	3.5631	3450.1	0.0155
0.4	0.6	2.8713	3449.2	0.0104
0.5	0.5	2.3011	3449.9	0.0145
0.6	0.4	1.9837	3449.9	0.0145
0.7	0.3	1.6720	3450.1	0.0170
0.8	0.2	1.4967	3457.4	0.0258
0.9	0.1	1.3200	3452.1	0.0291
1.0	0.0	1.2236	3451.5	0.0386

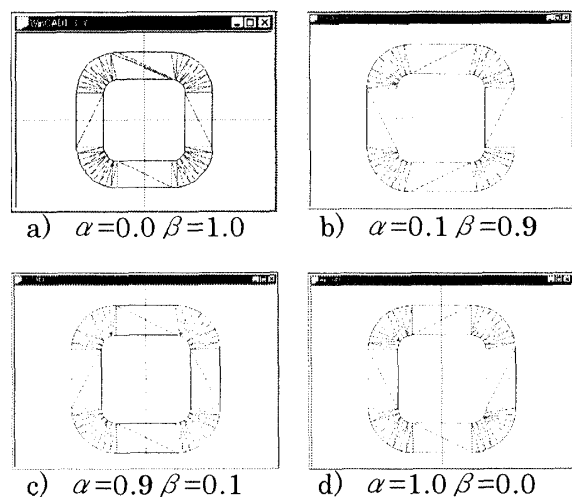


図 8 加重と結線状態(その 1)

表5 加重 α β と GA の処理結果(その 2)

α	β	MaxFit	f_l	f_v
0.0	1.0	13.0200	7269.2	0.0211
0.1	0.9	6.6971	7305.7	0.0169
0.2	0.8	3.8317	7271.1	0.0254
0.3	0.7	3.1185	7305.7	0.0169
0.4	0.6	2.4811	6878.5	0.0280
0.5	0.5	1.9349	7258.1	0.0266
0.6	0.4	1.6937	6700.1	0.0499
0.7	0.3	1.5412	6871.1	0.0351
0.8	0.2	1.4379	6538.7	0.0440
0.9	0.1	1.3909	6075.8	0.0668
1.0	0.0	1.3165	5946.7	0.1391

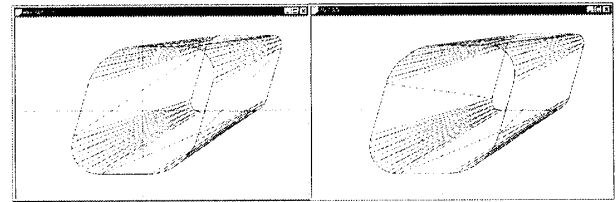
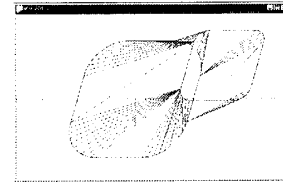
a) $\alpha=0.0$ $\beta=1.0$ b) $\alpha=0.1$ $\beta=0.9$ c) $\alpha=1.0$ $\beta=0.0$

図9 加重と結線状態(その 2)

表6 加重 α β と GA の処理結果(その 3)

α	β	MaxFit	f_l	f_v
0.0	1.0	2.5746	3098.4	0.1787
0.1	0.9	2.1867	3067.9	0.1771
0.2	0.8	1.9693	3061.9	0.1794
0.3	0.7	1.7314	3090.3	0.1933
0.4	0.6	1.6779	3047.2	0.1786
0.5	0.5	1.5543	3067.9	0.1771
0.6	0.4	1.4356	3047.8	0.1777
0.7	0.3	1.3833	3047.8	0.1777
0.8	0.2	1.2608	3051.9	0.1776
0.9	0.1	1.2178	3046.7	0.1794
1.0	0.0	1.1463	3031.2	0.2536

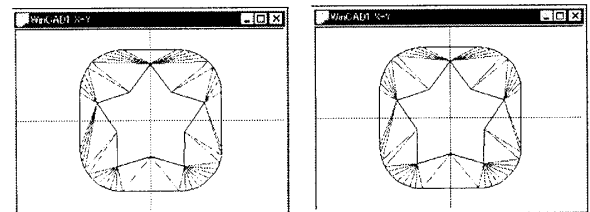
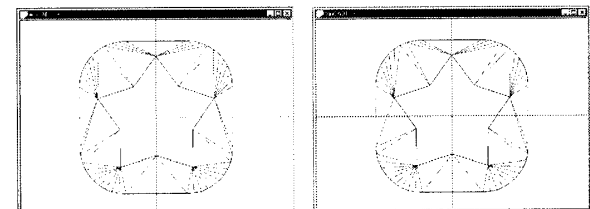
a) $\alpha=0.0$ $\beta=1.0$ b) $\alpha=0.1$ $\beta=0.9$ c) $\alpha=0.9$ $\beta=0.1$ d) $\alpha=1.0$ $\beta=0.0$

図10 加重と結線状態(その 3)

表7 加重 α β と GA の処理結果(その 4)

α	β	MaxFit	f_l	f_v
0.0	1.0	2.9382	6003.8	0.1492
0.1	0.9	2.3819	6001.9	0.1507
0.2	0.8	1.9801	5746.2	0.1772
0.3	0.7	1.9086	5660.2	0.1568
0.4	0.6	1.7635	5610.3	0.1581
0.5	0.5	1.6220	5261.2	0.1795
0.6	0.4	1.4532	5556.1	0.1583
0.7	0.3	1.3964	5305.0	0.1813
0.8	0.2	1.3295	5115.2	0.2073
0.9	0.1	1.2647	5170.4	0.2139
1.0	0.0	1.2545	5003.4	0.4252

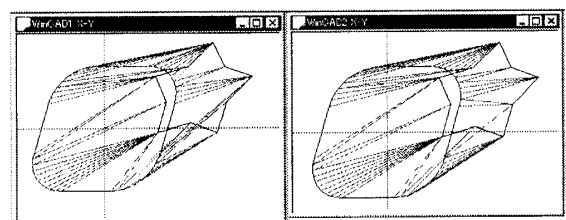
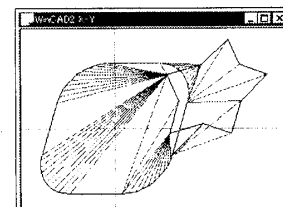
a) $\alpha=0.0$ $\beta=1.0$ b) $\alpha=0.1$ $\beta=0.9$ c) $\alpha=1.0$ $\beta=0.0$

図11 加重と結線状態(その 4)

次に、上記例に対し板金構造物としてある変形の例として、上断面形状が X' 軸方向に 100(下断面形状の幅の 2 倍に相当する大きさ)平行移動した場合について、検討した結果を表 5 に示す。同様に、得られた結線状態を図 9 に示す。図 9 の場合、結線状態が見やすいようにするために、視線方向を変更して表示している。このとき、結線長さの評価のみで結線状態を求めると、外観上展開図としては適正でないもの(図 9-c))が生成された。側面の法線ベクトルを重視する(β の値を高める)ことにより、外観上も適正な展開図が得られるようになった。

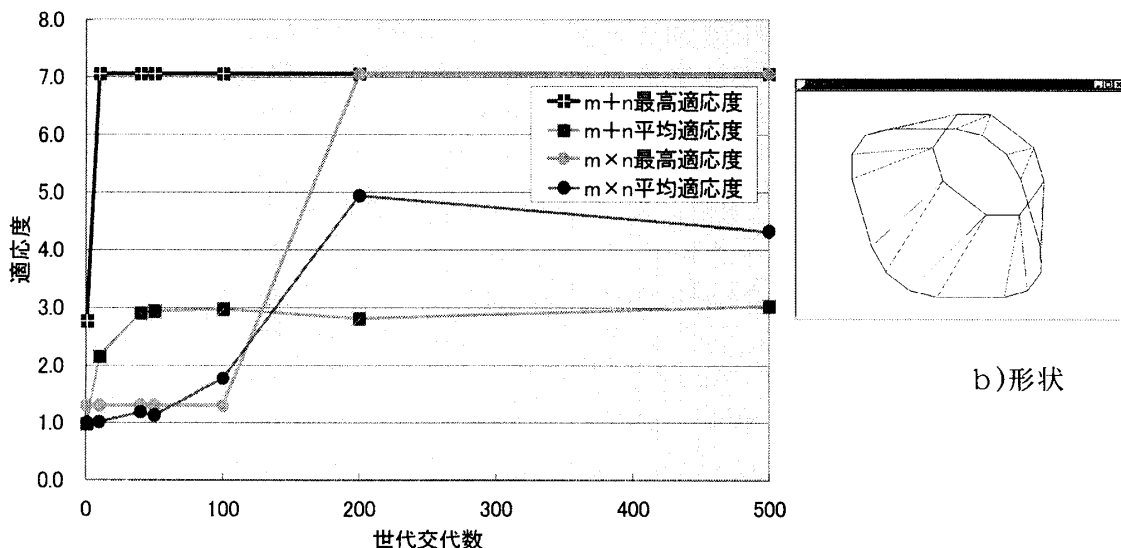
同様に、上断面形状が星形の場合も検討し、その結果を表 6 及び図 10、その上断面形状が X' 軸方向へ 100 平行移動した形状の結果を表 7 及び図 11 に示す。図 11 についても図 9 と同様に視線方向を変更して表示している。

表 6 においては、 $\alpha=0.7, \beta=0.3$, $\alpha=0.8, \beta=0.2$, $\alpha=0.1, \beta=0.9$ 付近の条件において、 f_l , f_v とも低い値となっていることがわかる。

上下の断面形状がずれている場合は、いずれも $\alpha=0.1, \beta=0.9$ に設定された場合で、 f_l , f_v とも低い値となっており、生成される側面の法線ベクトルの変化の滑らかさを重視する方が外観上も良い結果を得ている。

4.5.2. 世代交代数と適応度の関係

隣接制約パス表現を使用することによる計算時間短縮効果を検証するため、世代交代数と最高適応度の関係を調べた。GA の処理条件は、個体数 500, 交叉率 0.2, 突然変異率 0.04 とした。そして、上下断面の例として、上断面形状の節点数は 8, 下断面形状の節点数は 16 とした。そして、上下断面形状の中央点の X' , Y' 座標をほぼ同じとした。目的関数における二つの項の加重和については、結線長さの総和の加重 $\alpha=0.1$, 側面の法線ベクトルによる角度偏差の加重 $\beta=0.9$ とした。その結果を図 12-a)に示す。隣接制約パス表現を使用しない場合(図 12, 13では $m \times n$ と表記), 世代交代数が 200 を超えて収束した。隣接制約パス表現を使用した場合(図 12, 13では $m+n$ と表記)は、世代交代数が 10 を超えると最高適応度が収束している。隣接制約パス表現の使用, 使用しないに関わらず、得られた近似最適解は同一であった。近似最適解を用いた結線状態を図 12-b)に示す。

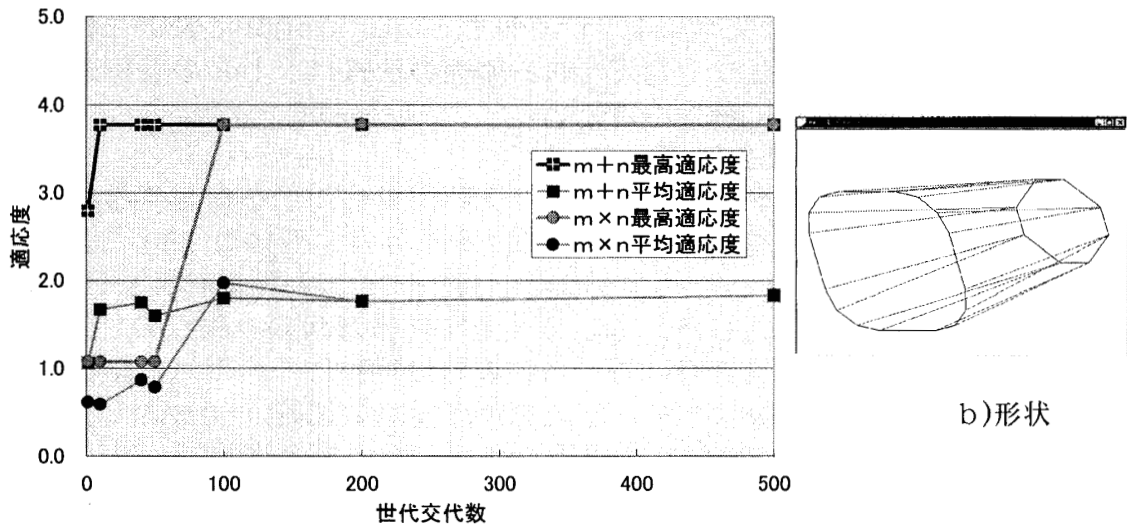


a) 世代交代数と適応度の関係

図 12 上下断面の中央点の x, y 座標がほぼ同じ場合の世代交代数と適応度との関係

さらに、図 1 2 と同じ上断面、下断面形状で、上部断面形状が下部断面形状に対して、 X' 軸方向に 100(下断面形状の幅とほぼ同じ大きさ)だけずれた場合の結果を図 1 3 に示す。この場合も、隣接制約パス表現を使用しない場合、世代交代数が 100 を超えて最高適応度が収束している。隣接制約パス表現を使用した場合は、世代交代数が 10 を超えると最高適応度が収束している。隣接制約パス表現の使用、使用しないに関わらず、得られた近似最適解は同一であった。近似最適解を用いた結線状態を図 1 3 -b)に示す。

図 1 2, 1 3 に示すように、隣接制約パス表現を使用することによる計算時間短縮効果は極めて大きい。また、隣接制約パス表現を使用しない場合は、上下断面形状を表現する点数が多くなると、良好な近似最適解を探索できないことが少くない。



a) 世代交代数と適応度の関係

図 1 3 上下断面が大きくずれた場合の世代交代数と適応度との関係

4.5.3. ランダムに発生させた個体と GA での近似最適解との適応度の比較

図 1 4, 表 8 -a), b)は、4.5.2 の図 1 2 に対応する上断面、下断面の形状の場合で、 $\alpha=0.1$, $\beta=0.9$ としたとき、ランダムに発生させた個体における適応度の分布を示したものである。個体数は 5000 個の場合である。隣接制約パス表現を使用せずに、 16×8 のマトリックスに $16+8$ 個の "1" をランダムに発生させた場合を Random1 と表記する。この場合、データを 5000 個発生させて、そのほとんどの個体で致死遺伝子が生じてしまい、わずか 1 個の個体しか生成できず、その最高適応度が 1.2998 であった。隣接制約パス表現に基づきランダムに個体を発生させた場合を Random2 と表記する。このとき、約 $7/10$ の 3676 個の個体で致死遺伝子が生じているが、その他の個体のほとんどは、0.5 から 2.5 の適応度の範囲に存在し、その個体の最高適応度は 4.2679 となった。GA では、個体数 500, 交叉率 0.20, 突然変異率 0.04 の条件で行った場合、世代交代数 10 で最高適応度 7.0584 が得られている。この結果からも、「隣接制約パス表現」は、適応度の高い個体の発生を容易にし、GA は探索手法として有効に機能していることがわかった。同様に、図 1 5, 表 9 -a), b)は、4.5.2 の図 1 3 に対応する上断面、下断面の形状の場合で、 $\alpha=0.1$, $\beta=0.9$ としたとき、ランダムに発生させた個体における適応度の分布を示

したものである。このとき、隣接制約パス表現を使用しない Random1 の場合、ほとんどの個体で致死遺伝子が生じ、その他の個体（2 個）の最高適応度が 1.07561

表 8 - a) ランダムデータと GA の適応度

	最高適応度	平均適応度
Random1	1.2998	1.2998
Random2	4.2679	1.1029
GA	7.0584	2.1577

表 9 - a) ランダムデータと GA の適応度

	最高適応度	平均適応度
Random1	1.0756	1.0756
Random2	3.0576	1.1114
GA	3.7794	1.6716

表 8 - b) ランダムデータの度数分布

	Random1 ($m \times n$)	Random2 ($m+n$)
致死遺伝子	4999	3676
0.0~0.5	0	5
0.5~1.0	0	615
1.0~1.5	1	550
1.5~2.0	0	115
2.0~2.5	0	30
2.5~3.0	0	5
3.0~3.5	0	2
3.5~4.0	0	0
4.0~4.5	0	2
4.5~5.0	0	0
5.0~5.5	0	0
5.5~6.0	0	0
6.0~6.5	0	0
6.5~7.0	0	0
7.0~7.5	0	0
7.5~8.0	0	0

表 9 - b) ランダムデータの度数分布

	Random1 ($m \times n$)	Random2 ($m+n$)
致死遺伝子	4998	3667
0.0~0.5	0	4
0.5~1.0	0	571
1.0~1.5	2	580
1.5~2.0	0	149
2.0~2.5	0	27
2.5~3.0	0	1
3.0~3.5	0	1
3.5~4.0	0	0
4.0~4.5	0	0
4.5~5.0	0	0
5.0~5.5	0	0
5.5~6.0	0	0
6.0~6.5	0	0
6.5~7.0	0	0
7.0~7.5	0	0
7.5~8.0	0	0

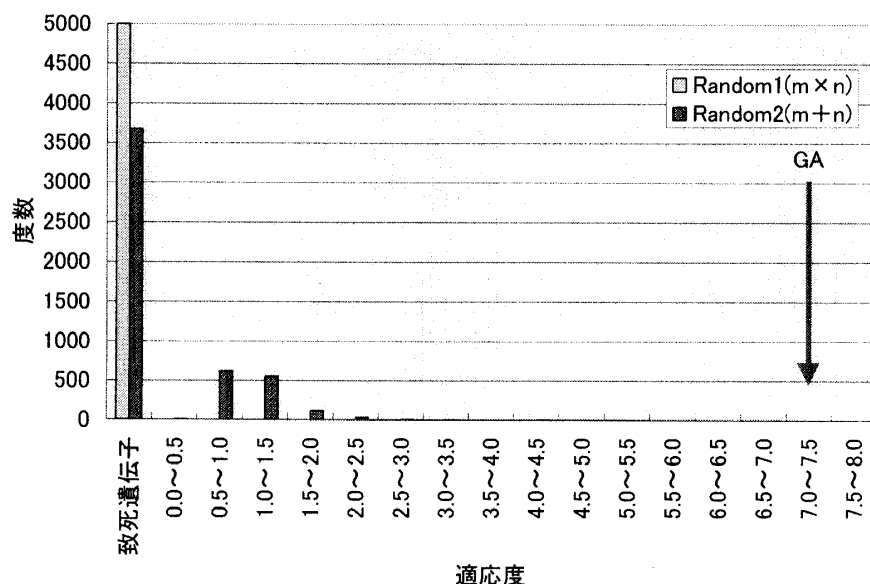


図 14 ランダムに発生させた個体の適応度の分布と GA での最高適応度(その 1)

であった。Random2 の場合、隣接制約パス表現に基づき個体を発生させている。このとき、約 7/10 の 3667 個の個体で致死遺伝子が生じており、その他の個体で得られた最高適応度は 3.0576 である。GA では、個体数 500、交叉率 0.20、突然変異率 0.04 の条件で行った場合、世代交代数 10 で最高適応度 3.7794 が得られている。この場合にも、「隣接制約パス表現」は、適応度の高い個体の発生を容易にし、GA は、探索手法として有効に機能していることがわかった。また、隣接制約パス表現を使用しない場合は、個体の発生も容易でないことがわかった。

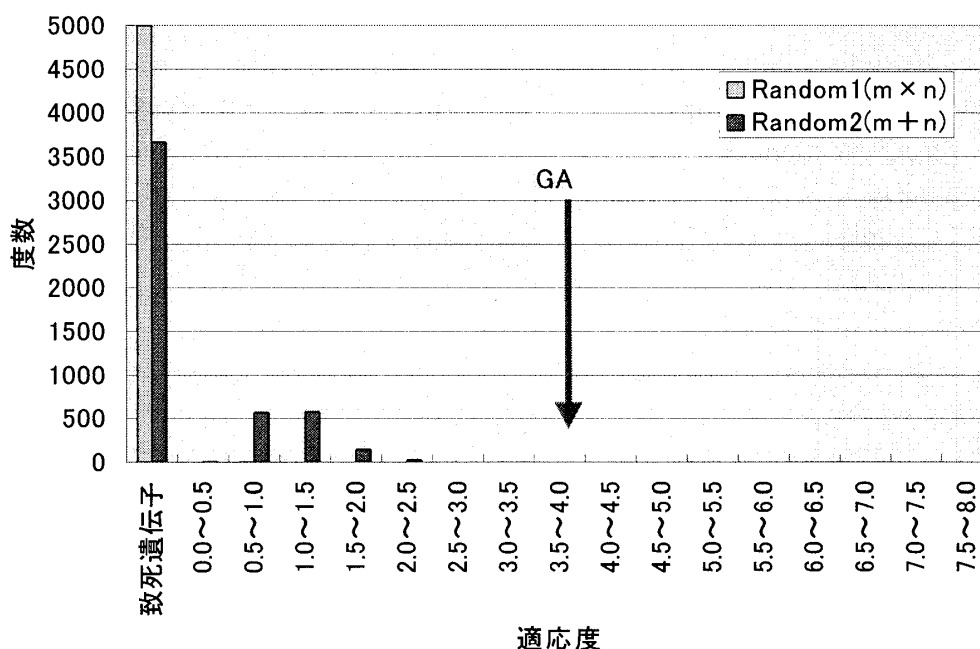


図 15 ランダムに発生させた個体の適応度の分布と GA での最高適応度(その 2)

4.5.4. 最適解と GA での近似最適解の比較

次に、上下断面形状の節点数の種々の条件で、最適解と GA での近似最適解の比

表 10 上下断面形状の節点数と最高適応度の関係

	下断面 節点数	上断面 節点数	α	β	GAによる最高適応度			最適解の適応度		
					MaxFit	f_l	f_v	MaxFit	f_l	f_v
a	4	4	0.9	0.1	1.0000	229.78	1.5708	1.0000	229.78	1.5708
b	4	10	0.9	0.1	1.3365	622.24	1.9000	1.3365	622.24	1.9000
c	4	16	0.9	0.1	1.3234	1014.72	1.1247	1.3234	1014.72	1.1247
d	14	4	0.9	0.1	1.2923	759.75	0.3142	1.2923	759.75	0.3142
e	14	10	0.9	0.1	1.2395	999.55	0.2614	1.2395	999.55	0.2614
f	24	4	0.9	0.1	1.4100	1289.72	0.3180	1.4100	1289.72	0.3140

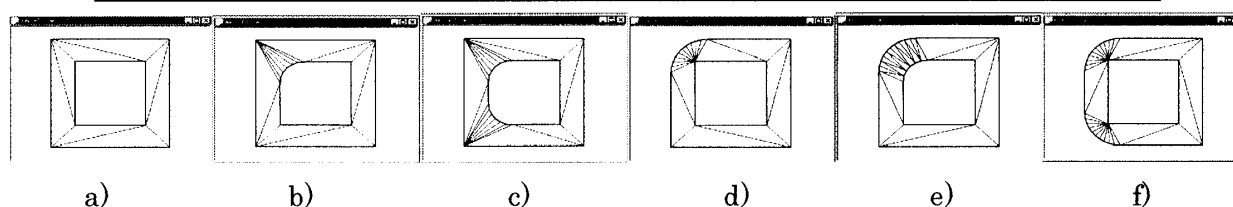


図 16 GA で求められた最適な結線状態 (例)

較を行った。最適解は、すべての実行可能解の中から求めた。その最適解と GA による近似最適解の適応度の比較を表 10 に示す。そして、GA で求められた結線状態の例を図 16 に示す。ここに示す形状においては、いずれも 16 通りの最適解がある。また表 11 には、表 10-a の場合の最適解の結線状態、表現型の個体、遺伝子型の個体の例を示す。表 10-a の場合において、GA により求められた最高適応度の個体は 1111001100 であった。

GA はそもそも近似最適解を得る手法であり、すべての場合で最適解が得られる保証はないが、表 10 に示した条件では、GA により最適解が得られた。

表11 表10-aの場合の最適解

結線状態	表現型個体	遺伝子型個体	結線状態	表現型個体	遺伝子型個体																																
	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	1	1	0	0	0	1	1	0	0	0	1	1	1	0	0	1	0010101010		<table><tr><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	1	0	0	0	1	1	1	0	0	0	1	1	1	0	0	1	0001101001
1	1	0	0																																		
0	1	1	0																																		
0	0	1	1																																		
1	0	0	1																																		
1	0	0	0																																		
1	1	1	0																																		
0	0	1	1																																		
1	0	0	1																																		
	<table><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr></table>	1	1	0	1	0	1	1	0	0	0	1	1	0	0	0	1	1111010100		<table><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr></table>	1	0	0	1	1	1	1	0	0	0	1	1	0	0	0	1	1110110100
1	1	0	1																																		
0	1	1	0																																		
0	0	1	1																																		
0	0	0	1																																		
1	0	0	1																																		
1	1	1	0																																		
0	0	1	1																																		
0	0	0	1																																		
	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	0	1	1	0	0	0	1	0	1	0	1	1	0010100011		<table><tr><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	0	0	0	1	1	1	0	0	0	1	0	1	0	1	1	0001100110
1	1	0	0																																		
0	1	1	0																																		
0	0	1	0																																		
1	0	1	1																																		
1	0	0	0																																		
1	1	1	0																																		
0	0	1	0																																		
1	0	1	1																																		
	<table><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	1	0	1	1	0	0	0	1	0	0	0	1	1	1111010010		<table><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr></table>	1	0	0	1	1	1	1	0	0	0	1	0	0	0	1	1	1110110010
1	1	0	1																																		
0	1	1	0																																		
0	0	1	0																																		
0	0	1	1																																		
1	0	0	1																																		
1	1	1	0																																		
0	0	1	0																																		
0	0	1	1																																		
	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	1	1	0	0	0	1	0	0	0	1	1	1	1	0	0	1	0010011010		<table><tr><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	1	0	0	0	1	1	0	0	0	1	1	1	1	0	0	1	0001011010
1	1	0	0																																		
0	1	0	0																																		
0	1	1	1																																		
1	0	0	1																																		
1	0	0	0																																		
1	1	0	0																																		
0	1	1	1																																		
1	0	0	1																																		
	<table><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr></table>	1	1	0	1	0	1	0	0	0	1	1	1	0	0	0	1	1111001100		<table><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr></table>	1	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1110101100
1	1	0	1																																		
0	1	0	0																																		
0	1	1	1																																		
0	0	0	1																																		
1	0	0	1																																		
1	1	0	0																																		
0	1	1	1																																		
0	0	0	1																																		
	<table><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	0	0	1	0	0	0	1	1	0	1	0	1	1	0010010110		<table><tr><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>1</td></tr></table>	1	0	0	0	1	1	0	0	0	1	1	0	1	0	1	1	0001010110
1	1	0	0																																		
0	1	0	0																																		
0	1	1	0																																		
1	0	1	1																																		
1	0	0	0																																		
1	1	0	0																																		
0	1	1	0																																		
1	0	1	1																																		
	<table><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr></table>	1	1	0	1	0	1	0	0	0	1	1	0	0	0	1	1	1111001010		<table><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td></tr></table>	1	0	0	1	1	1	0	0	0	1	1	0	0	0	1	1	1110101010
1	1	0	1																																		
0	1	0	0																																		
0	1	1	0																																		
0	0	1	1																																		
1	0	0	1																																		
1	1	0	0																																		
0	1	1	0																																		
0	0	1	1																																		

5. 適用例

5.1. GA の設定条件

展開図を求めるにあたり，上下の断面形状を表現する節点間の結線状態を GA で求めるための処理条件は，個体数 500 個，交叉確率 0.2，突然変異確率 0.04，世代数 20000 回とした．また，世代交代の繰返しにおいて最高適応度の更新が 5000 世代以上行われないうきは処理を終了するようにした．以下，得られた結果を示す．本研究では，DEC 製 Personal Workstation Pentium2 CPU400MHz を計算に用いた．

5.2. 結果

以下に結果例を示す．なお，GA で得られた近似最適解で決定される結線状態を用いて，展開図を作成する手法については，付録 2 に記す．

5.2.1. 上下形状ともに角 R（コーナー）を有する場合

上断面形状の節点数 24，下断面形状の節点数 44 の多節点の場合で展開図を求めた．このとき， $\alpha=0.1, \beta=0.9$ とした．得られた結線状態および展開図を図 17 に示す．この展開図では，四角形平面内に生成された不要な結線部分は削除されている．

5.2.2. 5.2.1. の上下断面形状で上断面形状が X' 軸方向に 100 (下断面の幅とほぼ同じ大きさ) だけずれた場合

上下断面において，上断面の中央位置が下断面に対して X' 軸方向に 100 だけずれている場合で展開図を求めた．この場合も， $\alpha=0.1, \beta=0.9$ で実施した．この結果，得られた結線状態と展開図を図 18 に示す．外観も含め，適正な展開図が得られている．

5.2.3. 上断面形状が星型形状の場合

上断面形状の節点数 10，下断面形状の節点数 44 の多節点の場合で展開図を求めた．このとき， $\alpha=0.9, \beta=0.1$ とした．得られた結線状態と展開図を図 19 に示す．

5.2.4. 5.2.3. の場合で上断面が X' 軸方向に 100 だけずれた場合

5.2.3. で示した形状において，上断面形状が X' 軸方向に 100 だけずれた場合について展開図を求めた．このとき， $\alpha=0.1, \beta=0.9$ とした．得られた結線状態と展開図を図 20 に示す．

5.2.5. 複雑な形状の場合

上断面形状の節点数 23，下断面形状の節点数 33 の場合で，凹部分がある複雑な形状において展開図を求めた．従来，このような形状は展開図としては希な例であるが，従来行われている定型パターンでの展開図の作成[8]では対応が困難であった形状である．

このとき， $\alpha=0.1, \beta=0.9$ とした．得られた結線状態と展開図を図 21 に示す．

5.2.6. 自由曲線で定義された形状の場合

断面形状が定型形状では表現できない場合であっても，節点表現を導入したことにより展開図の作成は容易に実現できるようになった．この例では，上下断面形状がともに，傾斜角度を有している物についての GA による展開図作成例を示すものである．

下断面形状は，円錐型形状をある平面で斜めに切断した断面形状を点群に分割し座標を求めている．また，上断面形状も，下断面の場合と同様に円錐型形状を平面で斜めに切断した断面形状を点群に分割し座標を求めている．このとき，下断面に

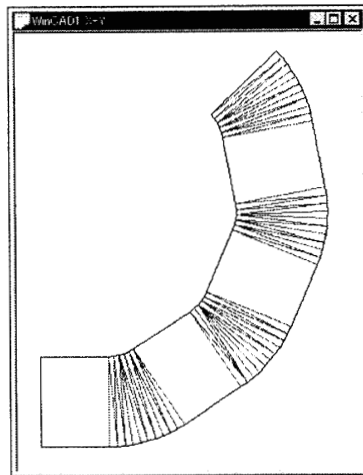
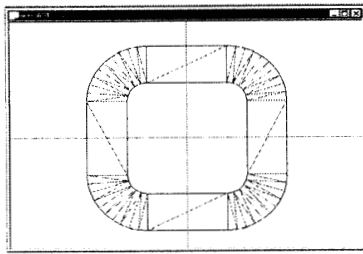


図 17 結線状態及び展開図
Type A ($\alpha=0.1, \beta=0.9$)

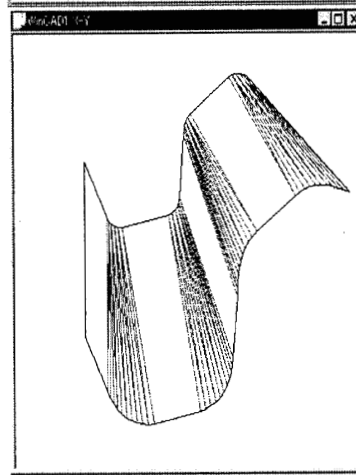
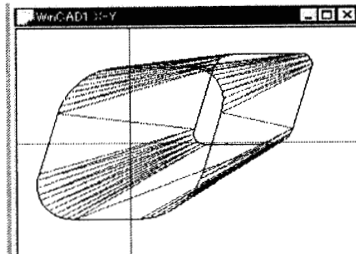


図 18 結線状態及び展開図
Type B ($\alpha=0.1, \beta=0.9$)

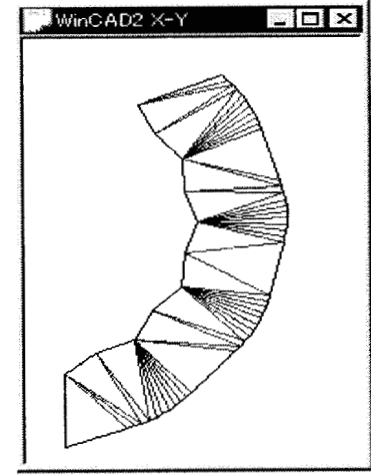
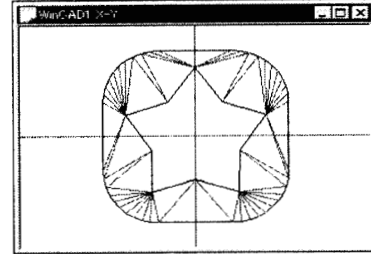


図 19 結線状態及び展開図
Type C ($\alpha=0.9, \beta=0.1$)

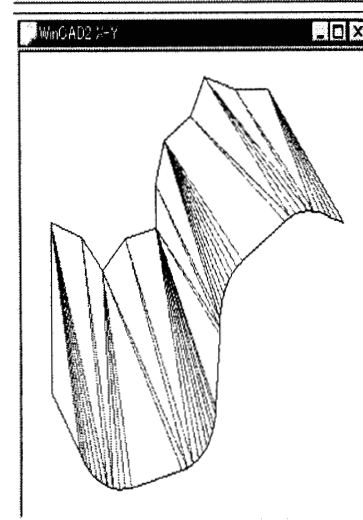
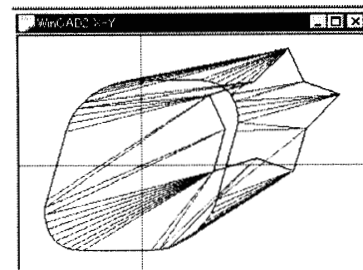


図 20 結線状態及び展開図
Type D ($\alpha=0.1, \beta=0.9$)

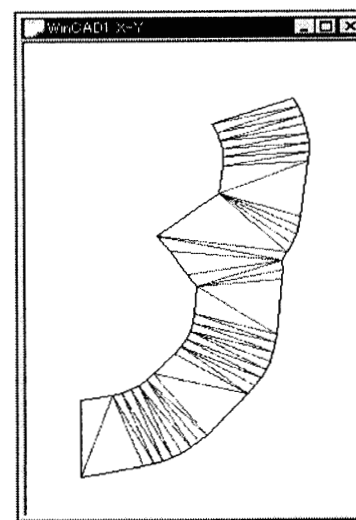
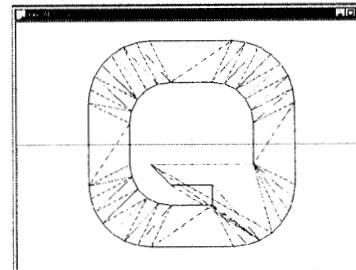


図 21 結線状態及び展開図
Type E ($\alpha=0.9, \beta=0.1$)

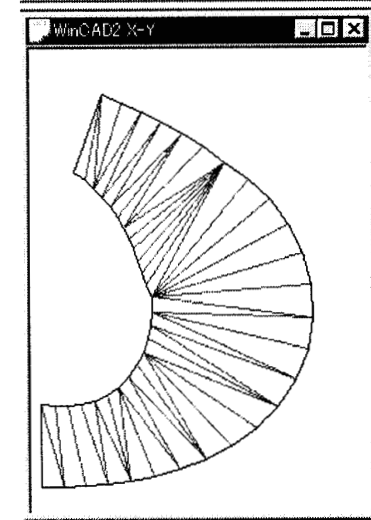
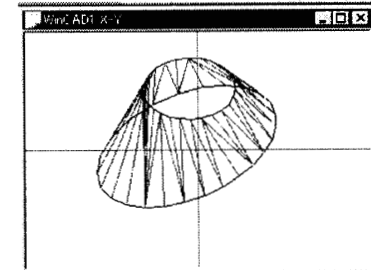


図 22 結線状態及び展開
Type F ($\alpha=0.9, \beta=0.1$)

おける円錐の底面を含む平面を $X'-Y'$ 平面とする $X'-Y'-Z'$ 三次元直交座標系において、節点の座標、下断面及び上断面の中央点の座標を求めている。この場合、 $\alpha=0.1$, $\beta=0.9$ として展開図を求め、図 2 2 に示す結果が得られた。

5.3. 考察

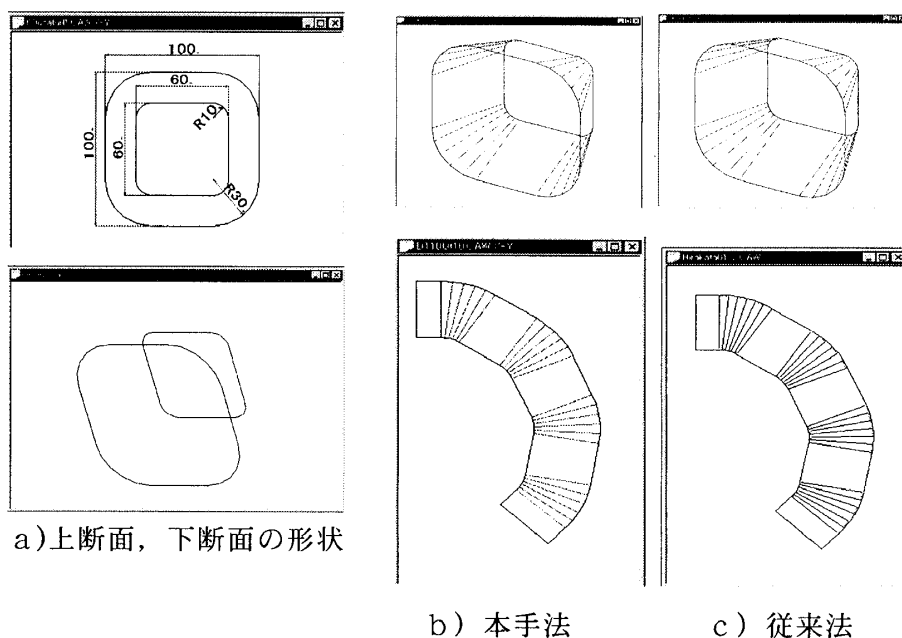


図 2 3 本手法と従来法による結線状態と展開図の比較 (その 1)

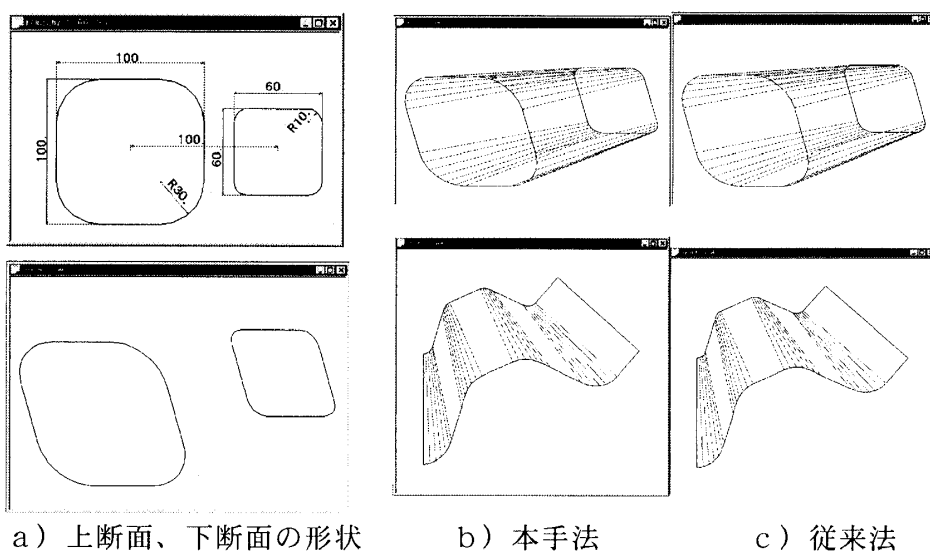


図 2 4 本手法と従来法による結線状態と展開図の比較 (その 2)

実際に企業で利用されている展開図生成プログラムでは、斜めに切断された円筒や円錐、上部円形で下部四角形などの複数のパターンに個別に対応した展開図を求めており、断面形状が円や四角形に限定されている。また、その展開図を求める手法は、側面を三角形に分割し、それらの三角形を平面に広げていく三角形法である。それに対して、本報では、上下断面形状を点群で表現し、より適切な展開図を求めるためにどのように点群間を結線し、三角形側面を構成すれば良いかを求めるアルゴリズムを提案している。そして、その後、側面を展開する工程は、この三角形法

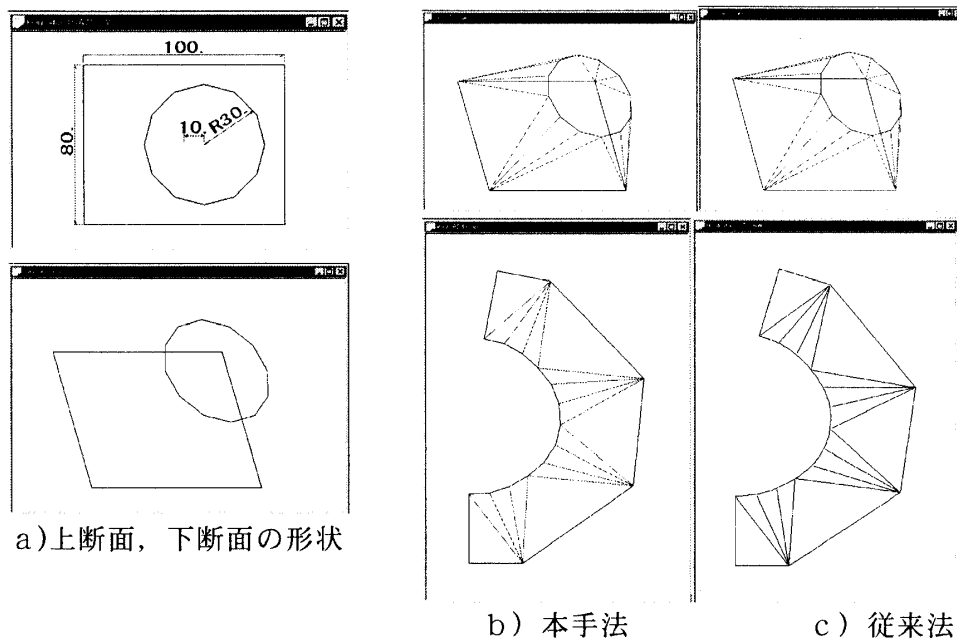


図 2.5 本手法と従来法による結線状態と展開図の比較 (その 3)

と同様な処理を行っている。そこで、それらの従来の展開図ソフトと本法の結果比較を行った。比較例としてあげた図 2.3, 2.4, 2.5 の形状は、「板金製缶 展開板取りの実際」[8]で紹介されている「上部円形と下部四すみ丸」、「上部円形と下部四角形」の手法で求められるものである。

まず、図 2.3 に示す形状は、図 1.7 の形状を基に展開図を求めたものである。この場合、従来法では、上下断面ともに同一分割数でしか対応できなかったため、本手法の場合も分割数を 24 に一致させて結果を比較した。このとき、本手法で得た展開図 (図 2.3-b)) と従来法による展開図 (図 2.3-c)) は一致した (展開された座標を確認)。また、図 2.4 においては、図 2.3 で示した上部断面形状が X 軸方向へ +100 だけずれた場合の展開図 (図 1.8 の形状が基本形状) を求めている。この場合も、本法と従来法で得られた展開図の結果は一致した。図 2.5 においては、上断面形状は半径 30 の円で、その中心位置を下断面形状の中心より X 軸方向に +10 だけずらした形状で、下断面形状は横長さ 100、縦長さ 80 の長方形、高さが 50 の形状 (図 2.5-a)) の展開図を求めた。この場合も、本手法で得た展開図 (図 2.5-b)) と従来法による展開図 (図 2.5-c)) は一致した (展開図の座標を確認)。しかし、図 1.9, 2.0 及び 2.1 で示すような凹凸のある非定形の断面形状の場合には、参照できる形状パターンは従来法には準備されていないので、展開図は作成できなかった (表 1.2 参照)。

本手法においては、曲線を含む任意の形状を節点で表現することにより、種々の断面形状を有する展開図を作成することが可能である (図 2.2)。しかも、各断面を表現する点数は、必ずしも一致する必要はなく、任意の各断面点数の条件で適性な展開図を生成できることの意義は大きい。本法は、本比較例では、従来法の結果と一致しており、また、従来法で展開図作成が不可能であった断面形状についても、適用が可能であった。

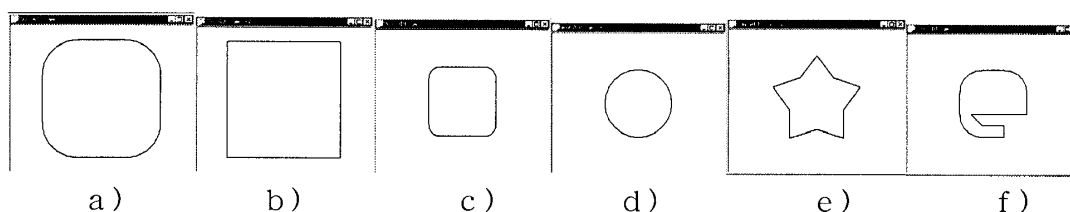
本手法では、上部、下部断面形状を表現する点群が、それぞれ一平面上に存在し、中央点が上断面及び下断面内に存在する条件を前提として、本法が適用でき適正な

展開図が得られることを確認した．上部及び下部が平面でない複雑な形状での板金構造物の展開図作成方法については，今後の課題である．

表 1 2 展開図作成可否の比較例

	形状				展開図作成可否	
	下断面形状	上断面形状	凹凸の有無	X軸方向のずれ	本手法	従来法
A	a	c	無	0	可	可(パターン対応)
B	a	c	無	100	可	可(パターン対応)
C	b	d	無	10	可	可(パターン対応)
D	a	e	有	0	可	不可(パターンなし)
E	a	e	有	100	可	不可(パターンなし)
F	a	f	有	0	可	不可(パターンなし)

断面形状



6. まとめ

- (1)板金構造物の展開図作成において，上下断面形状を表現する節点間の結線を最適化問題として定式化した．
- (2)結線部長さの総和と，隣接する側面が滑らかに方向を変えていくかの評価値の加重和を目的関数とした．展開可能な面の形成については，制約条件として考慮した．
- (3)GAのコード化で「隣接制約パス表現」を提案し，計算処理の高速化を実現できた．
- (4)その結果，上下断面形状が大きくずれた形状においても，適正な展開図を得ることができた．今回の検証実験の範囲では，本法は十分実用レベルにあると考えられる．

参考文献

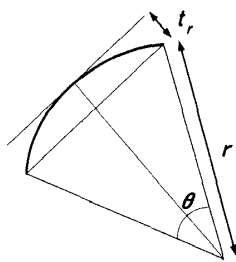
- [1] C.M.Fonseca and P.J.Fleming: Genetic algorithms for multiobjective optimization: formulation,discussion and genralization. *Proc. of 5th International Conf. on Genetic Algorithms*, (1993)416-423.
- [2] M.C.Gan,S.T.Tan and K.W.Chan:Generating and flattening of developable surfaces. *Advances in Design Automation*, 69-1,(1994)359-366.
- [3] J.Horn,N.Nafpliotis and D.E.Goldberg: A niched pareto genetic algorithm for multiobjective optimization. *Proc.of 1st IEEE Conf. on Evolutionary Computation*, (1994)82-87.
- [4] Yu-Long Lin and D.C.H.Yang: Automatic development generation for thin-walled objects. *Advances in Design Automation*, 69-1,(1994)367-377.
- [5] 三谷 由香,入口 健二,加藤 清敬:板金部品の設計システム.公開特許公報,特開

平 7-56974. (1995).

- [6] T.Murata and H.Ishibuchi:Multi-objective genetic algorithms. *Proc. of 2nd IEEE-ICEC Conf.*, (1995)289-294.
- [7] J.D.Scahffer:Multi-objective optimization with vector evaluated genetic algorithms. *Proc.of 1st International Conf. on Genetic Algorithms*,(1985)93-100.
- [8] 繁山 俊雄: 板金製缶 展開板取りの実際(理工学社, 1973).

付録1 点群の作成方法

断面形状を点群に表現し直す手法について説明する. 断面形状は, 主に直線と円弧で表現されている. そして, 直線の場合は, その直線の両端点の2点を点群として作成する. 円弧の場合は, トレランスの値によって円弧をいくつかの点群に分割する. トレランス(t_r)は, 円弧を直線近似するときの許容値である (図A). 円弧の半径 r がわかっていれば, トレランスによって, その円弧の分割角度は次式により決定される.



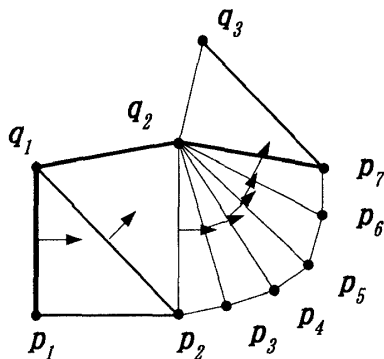
$$\theta = 2 \cos^{-1} \left(1 - \frac{t_r}{r} \right)$$

ここで, 求められた分割角度により, 円弧を分割し, 各分割角度位置で点群を作成する.

付録2 展開図の作成処理

展開図の作成においては, 必ず上断面形状の点から下断面形状の点に向かって曲げ加工線を描く. 展開図はその線の方角に対して反時計回りに各側面要素を開いていくことにより作成される. 図B及びCに示すように, 点 q_1 と点 p_1, p_2 の3点を頂点とする三角形が与えられていて, 次の q_2 の点位置を求めようとするとき, 点 q_1 を中心とし半径を点 q_1 から点 q_2 間の距離とした円と, 点 p_2 を中心とし半径を点 p_2 から点 q_2 間の距離とした円との交点は2つ求められる. このとき, 適正な展開図を得るにはどちらの点を選択すべきかを検討した. 結線状態の探索においては, 定義された上下断面の形状を反時計回りに探索しているので, 側面の生成においても, この手順に従って展開図を作成する. つまり, 上断面形状の一点から結線状態にある下断面形状の点を見て, 上断面形状の点を中心として, 新たに展開される三角形 (または四角形) が反時計回りになるように q_2 の位置を選ぶ. このようにして展開した平板が重なって描画されるのを防いでいる.

図B 展開図作成時の
方向の決定



図C 展開図作成手順
の説明

外山 真也

〒880-0303

宮崎県宮崎郡佐土原町東上那珂 16500-2

宮崎県工業技術センター 機械電子・デザイン部

E-mail:toyama@iri.pref.miyazaki.jp

ABSTRACT

**GENETIC ALGORITHM APPROACH TO GENERATING
THE OPTIMAL DEVELOPMENT OF A THIN-WALLED OBJECT**

Masaya Toyama
*Miyazaki Prefecture
Industrial Technology
Center*

Shigeyuki Tomita
Miyazaki University

Yasunari Yoshitomi
*Kyoto Prefectural
University*

Yoshihiro Harunari
*CyberTech
Corporation*

This paper presents a new method for generating the optimal development of a thin-walled object. The design of developments of thin-walled objects such as ventilation pipes is very important in the industry, where many complex development patterns are required. Although many CAD systems have been developed for this purpose, none to date has been able to satisfy this sort of practical demands because they are only applicable to limited patterns. For example, researches based on the flattening technique applicable to some developable figures have been reported.

This may be the first report with respect to the optimal selection of bending lines for the development of an object. In this work, GA is adopted, where a new coding method and a fitness function are introduced that pool the experience of experts in the field and take into account all physical aspects of this domain. Additional conditions for generating only feasible developments are also considered. The phenotype of each solution is expressed in the form of a 0-1 matrix, which describes the conditions mentioned above, and the genotype is expressed as a starting point and a zigzag (right and down) path in the matrix. In addition, an "Alternative path representation" is proposed for the coding of GA. The fitness function is made up of two objective functions, the first of which describes the total length of bending lines that link two sample points surrounding the entrance and the exit, respectively; the second of which describes the smoothness of changes in directions of normal vectors of the generated side surface.

The coding method of "Alternative path representation" is effective enough for speeding up the calculation. It has been confirmed through several experiments that the method presented here is sufficiently applicable to the generation of a variety of development-patterns required by the industry.