日本機械学会論文集(C編)
76巻764号(2010-4)

論文 No. 09-0650

# GAとバーチャルファクトリを用いた組み立てセル生産の部品配置決定システム*

山本秀彦*1, 山田貴孝*1, 中村昌弘*2

# Parts Layout Decision for Assembly Cell-Production by GA and Virtual Factory System

Hidehiko YAMAMOTO*3, Takayoshi YAMADA and Masahiro NAKAMURA

*3 Faculty of Engineering, Gifu University,
1-1 Yanagido, Gifu-shi, Gifu, 501-1193 Japan

This paper describes the system that can decide the efficient parts layout for assembly cell-production before setting up the real cell-production line in a factory. The system is called Virtual Assembly Cell-production System (VACS). VACS consists of two module, a parts layout decision GA system and a virtual production system. Especially, the GA system adopts the original crossover method called Twice Transformation Crossover. It is ascertained that VACS is useful by applying it to a cell-production line of a personal computer assembly.

***Key Words***: Production System, Virtual Factory, Assembly Line, Cell Production, GA

## 1. はじめに

ユーザーニーズの多様化と言われ久しくなり，多くの多品種生産システムが，実際の工場で稼動してきた．特に，最近のIT関連製品の組み立てには，セル生産[1][2]が有効とされ，各企業で実際に採用されてきている．ところが，このセル生産の設置には，トライ&エラーによる部品配置の決定作業が必要となる．すなわち，実際に部品を配置して，作業者が不便と感じないように，徐々に部品配置場所を修正していく調整期間が必要で，この調整期間に多くの日数を費やしている．

トライ&エラーの調整が必要な理由は，セル生産での部品配置場所決定には，非常に多くの不確定要因が重なるためである．すなわち，多品種を取り扱い，製品によって，使用する部品が異なり，かつ製品生産比率が異なるため，どの部品をどこに置くのが効率よいのか，実際にライントライしてみないとわからない．このため，今までシステム的にこの問題を解決した研究はほとんどない．

本研究は，この問題を解消するため，組み立てセル生産を対象にして，その計画時に効率のよい2次元的な部品配置を決定する遺伝アルゴリズム(GA)と，バーチャル・ファクトリを結合したシステムの開発に取り組む．特に，GAにおいては，特有の交叉方法を採用する．そして，開発したシステムを，パソコンの組み立てセル生産に適用し，開発システムの有効性を検証する．

## 2. 組み立てセル生産

本論文が対象とする生産ラインは，複数の製品を，1人の作業者が組み立てを行う，セル生産現場である．この組み立てセル生産現場の特徴は以下である．

[1] 組み立て部品（以降単に部品）を，作業者の周りの複数の棚に配置し，作業者1人が作業テーブルに向かい組み立てる．
[2] 製品によって，使用する部品が異なる場合と，同じ部品（共通部品）がある．
[3] 各製品の組み立て生産比率が異なる．

* 原稿受付 2009年7月6日．
*1 正員，岐阜大学工学部人間情報システム工学科(〒501-1193 岐阜市柳戸1-1)．
*2 (株)レクサー・リサーチ(〒680-0911 鳥取市千代水2-98)．
E-mail: yam-h@gifu-u.ac.jp

[4] 部品の大きさにより，片手で2個持てる部品と，両手で1個しかもてない部品がある．
[5] 1個流し生産[3]である．

上記のセル生産現場の略図を図1に示す．この図は，1人の作業者に対して，周りに5つの棚S1～S5があり，各棚から部品を取り出し，作業テーブルに置いて組み立て作業を行うセル生産である．

作業者は，特徴[1]により，近くの棚はほとんど歩かずに部品を取り出せるが，離れた棚は振り返り足を踏み出し部品を取り出す必要がある．すなわち，どの部品を近くの棚に置き，どの部品を離れた棚に置くかが，作業者に負担をかけない効率良い作業かの決め手となる．

特徴[5]から，製品1つづつ組み付けるが，その製品の組み立て順番がランダムなため，さらに，特徴[2]の要因も加わり，どの部品をどの棚に置くのが適当か断言できない．

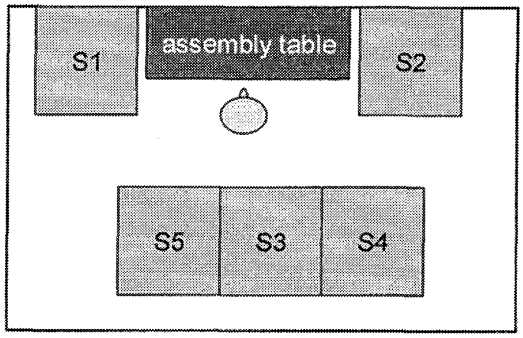


Fig.1 Example of cell production

一般に，2個持てる小さな部品を近くに置いておけば，無駄な動作なしに2つの部品を同時に取ることができ，効率的と思える．しかし，特徴[3]より，組み立てる製品の比率が異なり，さらに特徴[2]より，共通部品も存在するため，本当に2個持てる部品を近くに置くことが良いのか，断言ができない．たとえば，連続して使う小さな部品P1とP2を，図1の棚S1とS2にそれぞれ置くことがふさわしいか考える．ここで，部品P1とP2は，頻繁に使用する部品とすると，作業者の近くの棚，S1とS2，に置いても一般的に良いと予想できる．しかし，部品P1は3回に1回は使用する部品で，部品P2は1日に5回しか使用しない部品となると，すなわち，部品P2はめったに使わない部品である場合，部品P2を作業者近くの棚S1かS2に置くのは非効率と予想される．では，使われ方が，どのくらいの頻度になればよいのか？

このように，組み立てセル生産は，多くの不確定要因が重なり，事前に効率の良い部品の配置を決定することは容易でない．

## 3．開発システム VACS

**3・1 VACS の概要** 本研究は，2章で述べた問題点を解決するシステム，仮想組み立てセル生産システム Virtual Assembly Cell-production System (VACS)を提案し，事前に，各部品の配置場所を決定する問題に取り組む．VACS の特徴は，図2に示すように，バーチャル・ファクトリ・モジュールと部品配置 GA モジュールとの結合にある．これにより，VACS は次の4つの機能を持つ．
<1>バーチャル・ファクトリ・モジュール上で対象とする組み立てセル生産現場を仮構築する（仮生産現場）．
<2>仮生産現場から，棚名と距離の2元データを，部品配置 GA モジュールに転送する．
<3>GA モジュールは，2元データのうち，距離データとさらに部品名を用いて，効率よい部品の配置座標を求める．
<4>この配置座標をバーチャル・ファクトリ・モジュールに転送し，3次元の仮想セル生産現場を再構築し（最終生産現場），この最終生産現場を稼動し，目で見る最終確認を行う．

以上，4つの機能がバーチャル・ファクトリ・モジュールと部品配置 GA モジュールの結合により可能となる．

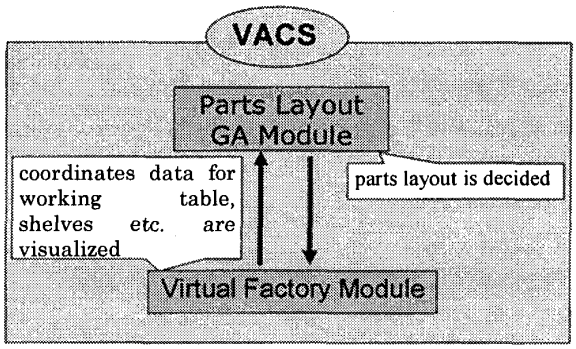


Fig.2 VACS

**3・2 2元データ** 上述の4つの機能のうち，<1>の仮ラインの構築では，3次元 CG を用い仮想組み立てセル生産現場を実際に設計する．特に，作業テーブルや，各棚を設計し，3次元空間のバーチャル・ファクトリーに配置・作図する．作図したこの仮生産現場から，棚名と棚の位置座標を抽出する．すなわち，棚

の数n個の棚名S(n)と，n個の棚座標$C_n=(x_n, y_n)$を抽出する．

さらに，この棚座標$C_n$と，作業者の組み立て作業立ち位置座標$(a, b)$を用いて，組み立て作業立ち位置から各棚への距離$(L_n)$を求める．この距離は，距離集合$\boldsymbol{L}$として式(1)で表現する．

$$\boldsymbol{L}=\{L_1,L_2,\cdots,L_n\}$$
$$=\{\sqrt{(a-x_1)^2+(b-y_1)^2},\ \sqrt{(a-x_2)^2+(b-y_2)^2}\cdots,\ \sqrt{(a-x_n)^2+(b-y_n)^2}\} \quad (1)$$

さて，VACSは，機能<2>で，仮生産現場から2元データを転送するが，この2元データは，式(2)で示す，棚名S(n)と距離集合$\boldsymbol{L}$の各要素の組み合わせを1つの要素とする集合$\boldsymbol{Q}$である．

$$\boldsymbol{Q}=\{(S(1), L_1), (S(2), L_2),\cdots(S(n), L_n)\} \quad (2)$$

**3・3 部品配置の決定** 棚の2次元の部品配置場所決定は部品配置GAモジュールで行う．このモジュールの第1番目の仕事は，バーチャル・ファクトリ・モジュールから転送された2元データのうち，棚名S(n)と，さらに部品名を用い，初期個体を生成することである．この個体は，配置する場所となる棚名と配置する部品との組み合わせを要素とする集合である．

すなわち，仮生産現場の棚名S(n)と，この棚名に配置するn個の部品名M(j)で1つの要素を表し，この要素の集合体となる式(3)の個体Iを生成する．

$$I=\{(S(n), M(j)),\ |\ n\text{は1から順にnまでの数},\ j=1\sim n\text{の重複しない数}\} \quad (3)$$

さて，部品配置GAモジュールは，式(3)に相当する個体を複数作成し，初期個体群を生成する．その後，適応度計算，交叉，突然変異のサイクルである，GA操作[4)-6)]を繰り返し，最後の仕事である部品配置を決定する．

ここで，VACSの扱うセル生産問題を，従来のGA操作で処理すると問題が生じる．すなわち，従来の交叉法[4)7)8)]を用いると，致死遺伝子を多数生成してしまう．例えば，図3で示す2つの親個体I(1)，I(2)を考える．交叉点を図3に示すように1箇所選び交叉すると，図4の次世代個体I(1')とI(2')ができる．すると，I(1')に2つの要素9が生成され，重複した要素を持つ次世代個体ができてしまう．式(3)から，個体の要素は，重複が許されないため，このI(1')は致死遺伝子となってしまう．このように，従来の交叉法をVACSに用いると，致死遺伝子を多く生成してしまう．そこで，この問題点を解消するため，次に示す2度の変換を行う交叉法(Twice Transformation Crossover: TTC)を提案する．

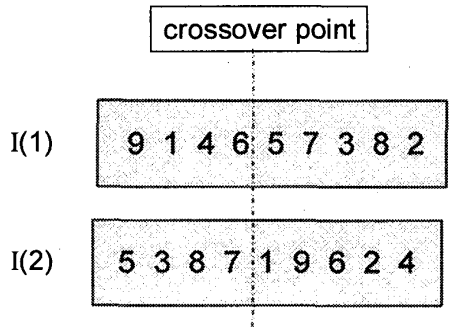


Fig. 3 Parent individuals

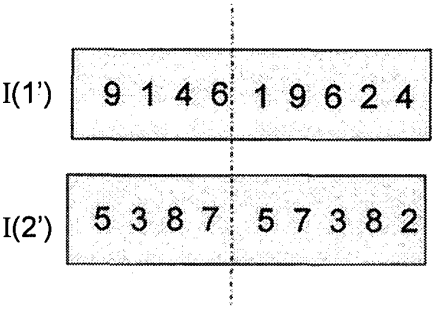


Fig. 4 Children individuals

TTCは，次の手順で，対象とする2つの個体を2度の変換処理を行い交叉させる．

手順[1]基準配列を1つ作成する．

手順[2]基準配列を用い2つの個体$I_1$, $I_2$それぞれを変換し，個体を代替して表す，2つの集合$T_1$', $T_2$'を生成する．

手順[3]新しい2つの集合$T_1$', $T_2$'どうしで交叉操作し，2つの集合$T_1$", $T_2$"を生成する．

手順[4]集合$T_1$", $T_2$"を，基準配列を用い逆変換し，棚名と部品名を要素とする個体表現を獲得し，これを次世代個体とする．

手順[1]の基準配列(Standard Arrangement; ***SA***)は，式(4)に示す整数と部品名を1要素とする集合で表現する．すなわち，配列の左から，1から始まり，棚の数に相当する整数値nまでの順に並ぶ番号(Order)と，ランダム

に配置した部品名の組み合わせを 1 要素とする集合である.

$SA$={ (Order, M(j') ) ｜ j'はランダムに選択する 1~n の重複しない整数} (4)

手順[2]の変換は，個体と基準配列を用いて，以下の操作を行う．ただし，この操作で用いる k と $x$ の初期値は 1 とする.

Step2-1: 個体 $I_1$ の遺伝子座 $x$ 番目の要素の部品名 M(j) と同じ部品名 M(j')を基準配列の要素から探し，その基準配列の並び番号 Order(k)を探す.

Step2-2 : Order(k)を集合 $T_1$'の第 k 番目の要素とする.

Step2-3 : 基準配列を，次のように更新する．基準配列の要素の中から, M(j')とその Order(k)の要素を削除し，この削除要素以降の要素の Order(k)を 1 づつ繰り下げて，新しい基準配列として更新する.

Step2-3 : 個体 $I_1$ の遺伝子座 2 番目以降すべての要素について，k←k+1, $x$←$x$+1 を繰り返し，Step2-1～Step2-3 を行い，新しい集合 $T_1$'を完成する.

$T_1$'={Order(k) ｜ k は 1～n の整数} (5)

個体$I_2$に対しても, 同様に新しい集合$T_2$'を完成する.

手順[4]の逆変換は，以下の操作を行う．ただし，手順[3]での交叉操作後の集合を $T_1$"= {Order(k') ｜ k'は 1～n の重複しない整数} とする．また初期値 y=1 とする.

Step4-1 : 集合 $T_1$"から，並び番号 y 番の要素 Order(k') を探す.

Step4-2 : 基準配列から並び番号 k'番目の要素を探索し，棚名 S(y)と部品名 M(k')を y 番目の要素とする次式の集合 $newI_1$ を生成する.

$newI_1$= {(S(y), M(k') ｜ k'は 1～n の重複しない整数} (6)

Step-4-3 : 基準配列を，次のように更新する．基準配列の要素の中から, M(k')とその Order(k')の要素を削除し，この削除要素以降の要素の Order(k')を 1 づつ繰り下げて，新しい基準配列として更新する.

Step4-4 : 集合 $T_1$"の並び番号 2 番以降すべての要素について， y←y+1 として，Step4-1～Step4-3 を行い，$newI_1$ を完成する.

例えば，図 5 に示す文字列が基準配列，図 6 の 2 つの個体を交叉する場合を考える．個体 I(1)について，Step2-1 の個体 I(1)の遺伝子座 1 番目の要素の部品名 M(j)を探索すると，M(2)となり，この M(2)を基準配列の並び番号で探すと 3 番目，すなわち Order(3)となる. Step2-2 では，集合 T'(1)={Order(3)} を生成する. Step2-3 で，M(2)を含む要素を削除した新しい基準配列 {(1, M(4)), (2, M(5)), (3, M(1)), (4, M(3))} を生成する. Step-2-4 から Step2-1 にもどり，遺伝子座 2 番目の要素の部品名 M(1)より，このM(1)を基準配列の並び番号を探すと 3 番目，すなわち Order(3)となり，T'(1)= {Order(3), Order(3)} を生成する．新しい基準配列は {(1, M(4)), (2, M(5)), (3, M(3))} となる．以下，同様に M(5), M(3), M(4) について行い，T'(1)= {Order(3), Order(3), Order(2), Order(1), Order(1)} が完成する.

個体 I(2)についても，同様にして，T'(2)= {Order(2), Order(2), Order(2), Order(2), Order(2)} を得る.

$SA$ = {(1, M(4)), (2, M(5)), (3, M(2)), (4, M(1)), (5, M(3))}

Fig. 5 Standard arrangement

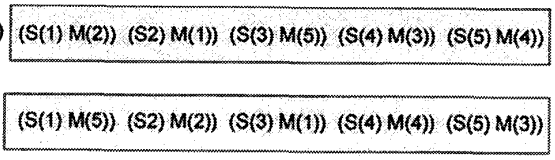


Fig.6 Individuals

手順[3]は，T'の要素を前から 2 番目と 3 番目で交叉させると，T'(1)= {Order(3), Order(3), Order(2), Order(2), Order(2)} と T"(2)= {Order(2), Order(2), Order(2), Order(1), Order(1)} を得る.

T'(1)について手順[4]を考える．Step4-1 では，並び番号 1 番目は Order(3)より，k'=3 となり，Step4-2 として基準配列の 3 番目の要素 M(2)が選ばれ, newI(1)={(S(1), M(2))} を生成する．Step4-3 で，要素 M(2)を削除した新しい基準配列 {M(4), M(5), M(1), M(3)} を生成する. Step4-4 で, y=2 とし, Step4-1 にもどる. すなわち, T'(1) の 2 番目の要素は Order(3)で，k'=3 となり，Step4-2 として基準配列の 3 番目の要素 M(1)が選ばれ，newI(1)= {(S(1), M(2)), (S(2), M(1))} を生成する．以下同様にして，newI(1)= {(S(1), M(2)), (S(2), M(1)), (S(3), M(5)), (S(4), M(4)), (S(5), M(3))} となり，交叉後の新個体を生成する. この交叉後の個体要素には，重複した要素は含まれず，TTC を用いることで致死遺伝子の発生をなくす事ができる．致死遺伝子を生成しない交叉法として CX 法や OX 法などが開発されてきたが，TTC は基準配列

を用い交叉を行う点で，これらと異なる手法である．そして，TTCは，交叉点を境に，その前半部分の親遺伝子情報を，子遺伝子が継承しており，CX 法やOX 法と同等の遺伝継承が可能である．

また，適応度は作業者の移動距離の短いほど良いとし，式 (2) の要素である，組み立て作業立ち位置から棚への距離 $L_n$ を用いて計算する．

## 4．シミュレーション応用

VACS を用いて，パソコン組み立てのセル生産シミュレーションを行った．VACS の 3 次元バーチャル生産モジュールは，レクサー・リサーチ社製の GP4 を用いた．

パソコン部品は表 1 に示すように 12 部品で，製品種類は 10 種類とした．製品種類ごとに組み立てる部品は異なる．表 1 は 10 種のうちの 5 種類部品 (P(1)～P(5)) の必要とする部品の数と，その組み立て受注量を示す．また，この部品の配置場所は，図 7 に示すように組み立て作業テーブルの後方で，作業者を囲み込むようにレイアウトされた棚 A～R の 18 箇所である．

GA 操作では，1 世代の集団を 100 個体，ルーレット選択により親個体を選び交叉し，エリート保存個体 5 個体，突然変異率 5%を採用した．また適応度として，作業者の移動距離を評価基準に取り，移動距離が短いほど適応度が高いとした．

シミュレーションでは，AutoCAD で棚や作業台を設計し，それをバーチャル・ファクトリ・モジュールにレイアウトした．このレイアウトに基づき，各棚や作業台の座標値を自動的に読み込み，部品配置 GA モジュールに送り，GA 操作を開始した．図 8 が適応度の変化の一例である．約 200 世代を経過すると，適応度は一定値になった．この時の最高適応度を持つ個体が，求める部品配置である．表2に求めた部品配置を示す．表より，使用個数が多い部品は作業台の近くの棚 A, B や, M~R に集中しており，得られた部品配置の正当性が判断できる．この部品配置をバーチャル・ファクトリ・モジュールに送り，図 9 の最終生産現場が完成した．この最終生産現場で，バーチャル生産を稼動させ，目で見る最終確認を行う事ができた．

## 5．おわりに

本研究は，セル生産で部品組み立てを行う時に，効率のよい 2 次元の部品配置を，実際の生産システムを構築する前に決定するためのシステム開発である．このシステムは，部品配置決定 GA モジュールと 3 次元バーチャル・ファクトリ・モジュールとを統合した VACS である．特に，部品配置決定 GA モジュールでは，固有の交叉方法として，2 度の変換を行う交叉法を採用した．

開発した VACS を用いて，パソコン部品のセル生産を応用事例を行い，従来工場内に配置するまでできなかった効率のよい部品配置が，目で見る管理としてできあがり，VACS の有効性が検証できた．

Table 1 Personal computer parts

| products / parts | P(1) | P(2) | P(3) | P(4) | P( |
|---|---|---|---|---|---|
| case | 1 | 1 | 1 | 1 | 1 |
| power source | 1 | 1 | 1 | 1 | 1 |
| mother board | 1 | 1 | 1 | 1 | 1 |
| CPU | 1 | 1 | 1 | 1 | 1 |
| memory | 2 | 4 | 2 | 4 | 1 |
| | | | | | |
| TV tuner | 0 | 0 | 0 | 1 | 0 |
| fan | 1 | 1 | 1 | 0 | 0 |
| Order number | 20 | 17 | 12 | 9 | 8 |

Table 2 Acquired locations

| Locations | Parts | Parts number |
|---|---|---|
| A | motor | 80 |
| B | case | 80 |
| C | case fan | 35 |
| D | card reader | 49 |
| E | sound card | 46 |
| F | TV tuner | 13 |
| G | other card | 31 |
| H | capture board | 21 |
| I | LAN card | 35 |
| J | other options | 8 |
| K | FD drive | 18 |
| L | CPU fan | 15 |
| M | CPU | 82 |
| N | memory | 169 |
| O | hard disk | 126 |
| P | mother board | 80 |
| Q | video card | 76 |
| R | CD/DVD | 82 |

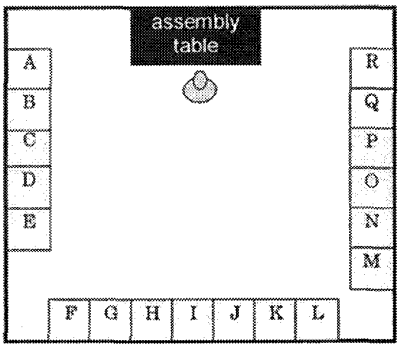


Fig. 7 PC assembly cell production

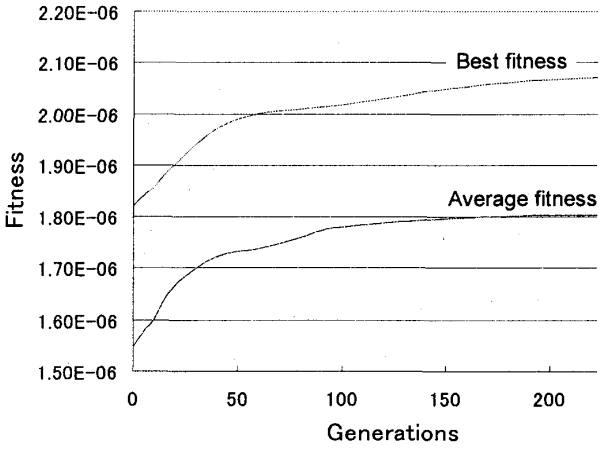


Fig. 8 Fitness curves

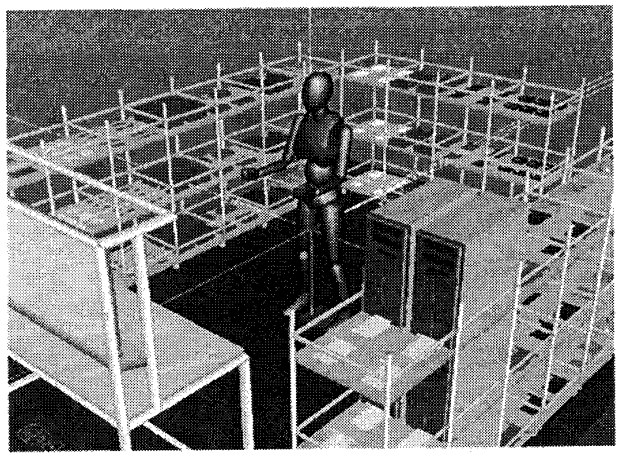

Fig. 9 Example of final production cell

## 文 献

(1) Jie Zhang, Felix T. S. Chan and et. al., Investigation of the reconfigurable control system for an agile manufacturing cell, International Journal of Production Research, Volume 40, Issue 15, (2002), 3709－3723.

(2) Maghsud Solimanpur, Prem Vrat and Ravi Shankar, A multi-objective genetic algorithm approach to the design of cellular manufacturing systems, International Journal of Production Research, Volume 42, Issue 7, (2004), 1419－1441.

(3) Hidehiko Yamamoto and Etsuo Marui, Off-line Simulator to Decide One-by-one Parts Input Sequence of FTL - Method of Keep Production Ratio by Using Recurring Individual Expression -, Journal of the Japan Society for Precision Engineering, Vol.69, No.7, (2003), 981-986.

(4) D.E. Goldberg, Genetic algorithms in search, optimization, and machine learning, Addison-Wesley, 1989.

(5) Hidehiko Yamamoto, One-by-one Production Planning by Knowledge Revised-Type Simulator with GA, Transactions of the Japan Society of Mechanical Engineers, Series C, Vol.63, No.609, (1997), 1803-1810.

(6) S. K. Ong, J. Ding and A. Y. C. Nee, Hybrid GA and SA dynamic set-up planning optimization, International Journal of Production Research, Volume 40, Issue 18, (2002), 4697－4719.

(7) L. Qiao, X. -Y. Wang and S. -C. Wang, A GA-based approach to machining operation sequencing for prismatic parts, International Journal of Production Research, Volume 38, Issue 14, (2000), 3283－3303.

(8) David W. Fanjoy and William A. Crossley, Topology Design of Planar Cross-Sections with a Genetic Algorithm: Part 1–Overcoming the Obstacles, International Journal of Engineering Optimization, Volume 34, Issue 1, (2002), 1－22.