旋削加工における遺伝的アルゴリズムを用いた 同時加工順序の決定法[†]

水 上 裕 登*・松 下 裕 一*・花 木 義 麿*
大 西 昇**・杉 江 昇***

A GA-based Planning of Simultaneous Machining Sequence for Turning

Yuto MIZUKAMI*, Hirokazu MATSUSHITA*, Yoshimaro HANAKI*,
Noboru OHNISHI** and Noboru SUGIE***

Lathes with two turrets are often used to get high efficiency of turning. To utilize them, it is important to plan a machining sequence considering simultaneous machining by two cutting tools, since combinations of processes influence practical cutting time very much. But it is difficult for users to determine an optimal machining sequence, because the number of the combinations is great.

A method proposed in this paper divides processes into cycles, during which spindle speed is constant, and distributes divided processes to two turrets. Genetic algorithm is applied to divide processes into cycles. Genes are assigned to processes. Values of genes represent identified cycle numbers, and processes with the same cycle number are simultaneously machined. A chromosome is a series of all processes, and constitutes a machining sequence. As a criterion of the first fitness, total cutting time by machining sequences represented by a chromosome is adopted. And as that of the second fitness, difference of cutting time on two turrets for each cycle is adopted. Preserving crossover, arbitrary crossover and mutation are used as genetic operations. Preserving crossover is multipoint crossover to preserve cycles with the good second fitness. Mutation is intended to break cycles with the bad second fitness. Preserving crossover is mainly used in the first stage, and arbitrary crossover and mutation are in the second stage. A simulation showed that proper use of two kinds of crossover and mutation produces good results.

Key Words: simultaneous machining sequence, optimization, turning, cycle, genetic algorithm

1. はじめに

機械加工工場においては、NC プログラムや治具などの作成が、慢性的熟練者不足により大きな問題となっている¹⁾。NC プログラム作成に掛かる労力を軽減するため、旋削加工を対象としたCAMシステム、自動プログラミングシステムに関しては、APT, EXAPTを始めと

して多くの研究・開発が行われ、様々なシステムが実用化されている^{1)~3)}。しかし、従来のシステムにはいまだ問題点も多く、ユーザのニーズを完全に満たしているとはいえない⁴⁾。その一つに、システムの出力するNCプログラムの最適性の問題がある。近年、多品種少量生産を実現すべく、工作機械が多機能化される傾向にあるが⁵⁾、特にこれら多機能工作機械に対応した最適なNCプログラム生成手法の確立は大きな課題である⁶⁾。

多機能工作機械の一つに、複数の刃物台や主軸を備えたNC旋盤がある。これは、複数の工具を同時に用いて加工を行うことにより、旋削加工の高効率化を目指したものである⁷⁾。しかし、NC自動プログラミングにおいて同時加工を考慮した最適な加工順序の決定手法がいまだ確立されていないため、同時加工機能を有効に活用できていないことが多い⁸⁾。本研究は、以上のような事情に鑑

[†] 第14回システム工学会・研究会で発表(1994・4)

* オークマ(株) 愛知県丹羽郡大口町下小口5-25-1

** 名古屋大学工学部 名古屋市千種区不老町

*** 名城大学理工学部 名古屋市中天白区塩釜口1-501

* Okuma Corporation, Niwa-gun, Aichi

** School of Engineering, Nagoya University, Nagoya

*** Faculty of Science and Technology Meijo University, Nagoya

(Received July 20, 1994)

(Revised December 8, 1994)

み、旋削自動プログラミングにおける、複数工具を用いた最適な同時加工順序のプランニング手法の確立を目的としている。本報では、1主軸・2刃物台(4軸)旋盤を対象として、遺伝的アルゴリズム(GA)を用いて準最適な同時加工順序を求める方法について述べる。なお、本文中における最適な加工順序とは、加工時間が最短となる加工順序を表わす。

2. 同時加工問題とその性質

2.1 問題の概略

Fig. 1は簡単な旋削加工図面の例である。この例では、加工領域が4個あり、各加工領域に工程が設定される。第1工程としてOD FACE ↓(加工領域1)、第2工程としてOD FACE ↓(加工領域2)、第3工程としてOD ←(加工領域3)、第4工程としてID FACE ↑(加工領域4)である⁹⁾。なお、ODは外径加工、IDは内径加工、FACEは端面加工を表わす。矢印は切削方向である。

1刃物台(2軸)旋盤では各時点で1本の工具による加工(単独加工)しか行えないため、加工順序によって正味切削時間は変化しない。ゆえに、加工順序の決定は困難な問題ではなく、Fig. 1でも第1工程から順に加工すれば良い。

一方、2刃物台(4軸)旋盤では2本の工具による同時加工が可能である。旋削加工における加工時間 T (sec)は、切削距離 L (mm)、送り f (mm/rev)、回転数 n (rps)を用いて、

$$T = \frac{L}{fn} \quad (1)$$

で表わされる。 f や n などの切削条件は、被加工物の材質や工具刃先の材質、切削方向などにより決まり、データベースに記述される。ある工程の切削条件を決定する場合、単独加工ならばデータベースの値をそのまま用い

て容易に決定することができるが、2刃物台旋盤における同時加工では、切削条件、特に回転数が同時加工の相手工程に依存する。(1)式のように、加工時間は回転数に反比例するので、工程の組み合わせ方により加工時間が変化する。ゆえに、2本の工具を用いて同時加工しても加工時間が半減するとは限らず⁹⁾、工程の組み合わせ方によっては、逆に加工時間が長くなってしまうこともある。また、単独加工で機械の許容負荷一杯で加工するように切削条件を設定してある工程を同時加工する際には、切込み量を減らすなど、切削条件を落とさなければならない。

よって、最適な加工順序を決定するには加工時間の最も短くなる同時加工の組合せを求める必要がある。しかし、被加工物の形状が複雑になると工程数が増加し、同時加工の組合せ数も爆発的に増大する。したがって、ユーザは膨大な労力を注いで計算を繰り返して加工時間の短い加工順序を見つけるか、勘と経験によって同時加工順序を決定してプログラミングしているのが実情である。このように、2刃物台旋盤においては加工順序の決定が重要な問題であるにもかかわらず、ユーザが最適な加工順序を見つけるのは非常に困難である¹⁰⁾。

熟練者が同時加工順序を考える場合、1刃物台旋盤にて加工するのと同様に切削方向ごとに加工領域を分割し、それを基に同時加工順序を決めることが多い。そして、同時加工可能な組合せが存在しない場合には、1刃物台では単一の工程となる加工領域を複数に分割して同時加工可能な組合せを作り出し、同時加工する。本論文では加工する順序のプランニングに重点を置くため、加工領域の分割方法は1刃物台旋盤でのそれと同じとして以下の論を進める。

2.2 サイクルの導入

同時加工の最も基本的な形態は、Fig. 2のように同時加工する双方の工程が終了してからつぎの工程を開始するものである。この場合は、同時加工する2個の工程から回転数が決定される。一方、Fig. 3のように追込み加工する場合、一つの工程が各時点で異なる工程と同時加工していることがあり得る。Fig. 3の例では、第1工程は最初第2工程と、ついで第4工程と同時加工している。このような場合、ある時点で加工している工程は高々2工

Fig. 1 Example of divided machining areas

Fig. 2 Basic simultaneous machining

程であるが、切削条件は3工程以上を考慮して決定しなければならない。このような場合を処理するため、サイクルという考え方を導入する。サイクルとは、ある時点から2個の刃物台により同時に、またはどちらか一方の刃物台により工程を開始し、工程を開始した時点から見て最初に双方の刃物台において工程が終了した状態になるまでの単位を表わす。そして1サイクル内では、回転数は一定である。たとえば Fig. 3 では、4工程で1サイクルを構成し、1サイクルで1個の同時加工と考える。全体の工程数を n とすると、サイクルを構成する工程の数は1個から n 個まで考えられる。サイクルを構成する工程数が1個の場合は単独加工にほかならないが、便宜上1工程による同時加工と表現する。

旋削加工には、定周速加工(切削速度を常に一定に保つように、加工径の変化に伴い主軸回転数を変化させる加工)もしくは定回転加工が用いられる。同時加工の場合、双方の工程を定周速加工で行うのは不可能である。ゆえに本論文では、同時加工は定回転加工とし、双方の工程の回転数のうち低い回転数を同時加工の回転数として採用する。

同じ工程によって構成されるサイクルの中でも、たとえば Fig. 3 と Fig. 4 のように刃物台への工程の分配の仕方は複数考えられる。すなわち、最適な加工順序を求めることは、全体の工程を実行可能なサイクルに分割し、サイクル内で加工時間が最短となるように工程を2個の刃物台に分配することに等しい。したがって、加工順序の決定にはサイクルへの分割と、刃物台への分配の2段階を要する。各サイクルを構成する工程を最適に刃物台に分配する問題は、いわゆる多プロセッサ・スケジューリング問題における、2台のプロセッサの場合に等しい。この問題に関しては多くの研究がなされており、良好な解を得る方法が提案されている¹¹⁾。ここで解決すべき問

題は、各工程をサイクルの単位に分割することである。以後、全工程をサイクルの単位に分割した形を、サイクル分割形と呼ぶ。

なお、刃物台に工程を分配する方法として、LPT(longest processing time)法を用いる¹¹⁾。

2.3 グラフ表現

問題を分かりやすく表現することは重要な問題である。そこで、実行可能なサイクル分割形全体を Fig. 5 のようにグラフ表現した。サイクル分割形を構築することは、全工程からなる集合から、空集合になるまでサイクル単位で工程を取り除いていくことである。

破線で描かれた矩形内にあるのが工程集合で、各集合が節になり、枝は工程の1サイクルの確定を表わす。各枝上の楕円はサイクルを構成する工程を表わす。破線で描かれた矩形の下から出ている枝の数が実行可能なサイクルの数である。各工程集合はサイクルを表わす枝で結ばれる。最上位の工程集合から枝をつたって ϕ に至ると、通ってきたサイクルを組み合せることにより一つのサイクル分割形が完成する。

工程集合を表わす破線で描かれた矩形の下部にある数字は各枝のサイクルにより加工した場合の最短加工時間

Fig. 3 Continuous simultaneous machining

Fig. 4 Another way of process distribution

Fig. 5 Graph representation of all plans to divide processes into cycles

である。破線で描かれた矩形の上部の数字は、破線で描かれた矩形の下部にある数字の最小値、すなわち当該工程集合に対する最短の加工時間である。 ϕ は全領域の加工が終了したことを表わす。なお、図中に示した加工時間の単位はすべて秒である。

3. 遺伝的アルゴリズムによる解法

Fig. 5 のようにグラフ表現される工程のサイクルへの分割問題は、動的計画法 (DP) の最適性の原理を応用して最適解を求めることができる。それによれば、最適解探索時の加算および比較回数を全数探索に比較して大きく削減できる¹⁰⁾。しかし、工程数が多くなると実際の計算時間は非常に長くなる。これは、Fig. 5 のグラフの中の楕円の数が増え、各工程の加工時間の計算回数が膨大になるためである。したがって、ヒューリスティックな手法により加工時間の計算回数を減らすことが求められる。そこで、ヒューリスティックな手法として遺伝的アルゴリズム (GA) を用いた方法について述べる¹²⁾。

3.1 問題の表現方法

問題をつぎのように表現する。各工程に遺伝子を割り当て、全工程の遺伝子を 1 次元に配列したものを染色体とする。染色体がサイクル分割形を表わす。遺伝子の採る値は $1 \sim n$ (n : 工程数) である。遺伝子の値は、それに割り当てられた工程の分類番号を表わし、同じ分類番号を付された工程の集合がサイクルを構成する。

3.2 初期染色体

初期染色体は、ランダムに設定する。同一染色体が生じないようにする。

3.3 適応度の計算

適応度として、染色体の表わすサイクル分割形による全体の加工時間 T_c の逆数を採用する。最終的には、適応度の最も大きい染色体が最適な加工順序を表わす。適応度 f_1 は以下の式で表現される。

$$f_1 = \frac{1}{T_c} \quad (2)$$

ここで、染色体の表わすサイクル分割形におけるサイクル数を m とすると、

$$T_c = \sum_{k=1}^m C_k \quad (3)$$

である。 C_k は、 k 番目のサイクルの加工時間で、

$$C_k = \max(CT_a, CT_b) \quad (4)$$

と表現される。ここで、2 個の刃物台を A 刃物台、B 刃物台とすると、 CT_a, CT_b はおのおの A 刃物台、B 刃物台の加工時間である。

3.4 遺伝子操作

遺伝子操作として、交叉および突然変異を用いる。

Fig. 6 An example of arbitrary crossover

Fig. 7 An example of preserving crossover

Fig. 8 An example of mutation

3.4.1 交叉

任意の箇所での一点交叉を行う方法 (任意交叉と呼ぶ) と、サイクルを保存するために多点交叉を行う方法 (保存交叉と呼ぶ) を用いる。任意交叉は、2 染色体を親染色体として選び、交叉の位置をランダムに決定して 2 個の親染色体から 2 個の子染色体を生成する。なお、淘汰されずに残った染色体をすべて任意に組み合わせ、親染色体とする。任意交叉の例を Fig. 6 に示す。

多点交叉である保存交叉は、サイクルを構成する遺伝子を親から子へ継承するように用いる¹³⁾。一般に、同時加工実行時において両刃物台の加工時間差が小さければ、その同時加工は好適であると考えられる。なぜなら、加工時間差が小さければ、双方の刃物台が効率良く稼働していると考えられるからである。そこで、両刃物台の加工時間差の逆数を第 2 の適応度 f_2 として採用し、加工時間の均衡度を評価する。すなわち、

$$f_2 = \frac{1}{|CT_a - CT_b|} \quad (5)$$

とする。 f_2 が基準値以上のサイクルが、保存交叉により親から子へ継承される。保存交叉では、2 個の親から 1 個の子を生成する。保存交叉の例を Fig. 7 に示す。保存交叉は、第 2 の適応度の高いサイクルをもつ全個体に対し、ランダムに相手を選んで行う。

3.4.2 突然変異

任意の 1 工程に対応する遺伝子の値を任意の値に変更する。その一例を Fig. 8 に示す。突然変異する遺伝子は、ランダムに選択する。

4. 予備実験

はじめに、上述の方法で初期染色体数や交叉の割合などのパラメータを決定するための予備実験を行った。結果はすべて10回の試行の平均である。遺伝子操作の繰返しは100世代とし、最適解に達しなくとも100世代で終了した。予備実験に用いた図面をFig. 10に示す。1~13の番号の付いた部分が加工領域である。ただし、Fig. 10は見やすいように変形してある。その工程数は13で、すべて荒加工のための工程である。保存交叉における第2の適応度の基準値は0.1(加工時間差10.0秒)とした。

予備実験を含むシミュレーションに用いた計算機は、SUN SPARC station 2である。

4.1 初期染色体の影響

初期染色体として500個生成し、淘汰により100個に減らした。初期染色体の質の影響を調べるため、初期染色体に対する淘汰のみ、適応度の低いものから100個残し、それ以後は適応度の高いものから残す場合と、常に適応度の高いものから残す場合とを比較した。遺伝子操作は、任意交叉と確率0.2の突然変異である。最終的に最適解に達するまでに要した世代数は、前者が28.8、後者が31.8であった。いずれの試行も最適なサイクル分割形に達していた。

結果からわかるように、初期染色体の淘汰において意

Fig. 9 Comparison between arbitrary and preserving crossover

識的に悪いものだけ残すようにしても、収束が遅くなることはない。すなわち、この問題に関しては、初期染色体の適応度の善し悪しが、最終的に得られる解の最適性、収束の速さに影響を与えることはほとんどないと判断できる。

4.2 交叉の効果

任意交叉と保存交叉の、収束の速さの比較をFig. 9に示す。双方の場合とも、突然変異を確率0.2で併用している。初期段階においては後者が明らかに速く収束する。しかしながら、保存交叉の場合、最適解に収束するまでに要した世代数は43.7世代で、任意交叉の場合の25.7世代に比較して局所解から中々抜け出せない傾向が窺える。そこで、初期段階ではサイクルを保存する交叉を行い、ある程度まで達したらランダムな交叉に切り替える方法が有効であると考えられる。

4.3 突然変異の効果

遺伝子操作に突然変異のみを用いた場合(確率1で適用)、100世代以内に最適なサイクル分割形に到達したのは、10回のうち5回であった。また、任意交叉のみを用いた場合、10回試行のうち、最適なサイクル分割形に到達したのは7回であった。一方、任意交叉と突然変異を用いた場合は10回とも最適なサイクル分割形に到達した。また最適なサイクル分割形に達するまでに要した世代数は、突然変異のみが81.4世代、任意交叉のみが37.2世代に対し、任意交叉・突然変異併用は25.7世代であった。結果から、この問題においては交叉に突然変異を適度に併用することが大変有効であると考えられる。

4.4 淘汰により残す個体数の影響

個体の淘汰により、適応度の高い上位100個の個体を確率1で残すようにした。適応度がそれ以下の個体は確率0.1で残した。一般的には、ランク戦略やルーレット戦略など、適応度に比例した確率で個体を残す方法が知られているが¹²⁾、本問題に適用したところ、上記手法と同等の結果しか得られなかった。そこで、よりシンプルな方法を採用することとした。

淘汰の段階において確率1で残す個体の数は100とした。シミュレーションの結果、この数が少ないと局所解

Fig. 10 Example of machining charts for simulation

に陥りやすく、多いと計算時間が多く掛かる傾向が見られた。結局その兼ね合いであるが、100 程度が最も良好な結果を示した。

5. シミュレーション結果と考察

5.1 シミュレーション方法

予備実験結果を基に、遺伝子操作方法を、前半において保存交叉と 0.2 の確率による突然変異、後半において任意交叉と 1.0 による突然変異とした。保存交叉・任意交叉の親の選び方は 3.4 節の記述と同様である。前半と後半で突然変異の確率を変えたのは、初期段階で有効な保存交叉の割合を多くするためである。前半・後半の切り替えは、最良解の更新が 3 世代行われなかった場合に行うこととした。初期染色体の質は解に大きな影響を与えないことが認められたので、初期染色体は 100 個のみ生成し、初期染色体に対する淘汰は行わないこととした。20 世代に渡り最良染色体の更新が行われない場合に、収束と判定し処理を終了した。

5.2 結果と考察

この構成によりシミュレーションを行い、DP による方法と処理時間を比較した例を Table 1 に示す。ここで用いた DP による方法は、必ず最適なサイクル分割形が求められる方法である。使用した図面は、予備実験で用いた図面を含む 3 図面である。Table 1 における GA の処理時間は、予備実験と同様に 10 回の試行の平均である。

GA においても、各図面とも 10 回の試行すべてにおいて最適なサイクル分割形に到達した。すなわち、本問題に対するこのシミュレーションでは、DP および GA による方法によって得られた解の間に違いはなかった。しかし、GA による方法では DP による方法に比べ処理時間が大幅に短縮されている。これは、DP が必ず全体の中から最適なサイクル分割形を求めるのに対し、GA は保存交叉により良好なサイクルを残し、それを含むサイクル分割形を中心に探索することになるためと考えられる。すなわち、保存交叉は Fig. 5 のグラフにおいて示されるサイクルのうち、良好なものを含むサイクル分割形を中心に探索すると考えることができる。

GA は一般には解の最適性を保証するものではないが、本問題に対してはシミュレーション試行時にすべて最適解が求められ、GA が有効であると考えられる。

Solution の欄に示した文字列は最適なサイクル分割形を表わす染色体である。文字列の左方から第 1 工程、第 2 工程の順に遺伝子が割り当てられている。同じ値を採っている工程群で 1 サイクルを構成する。Fig. 10 に示した図面 3 では、第 1 工程のみで 1 サイクル、第 2・3・

Table 1 Comparison between GA and DP on processing time

Chart No.	Number of processes	Solution	GA	DP
1	11	01122132321	8 sec	5 min
2	12	000112323144	15 sec	30 min
3	13	0112213232144	28 sec	210 min

Solution: Optimal constitution of cycles

6・11 工程で 1 サイクル、第 4・5・8・10 工程で 1 サイクル、第 7・9 工程で 1 サイクル、第 12・13 工程で 1 サイクルとなる。そして各サイクル内で、LPT 法を用いて工程の刃物台への分配が行われる。

6. ま と め

旋削加工図面のデータから加工方法を自動的に決定し、NC プログラムを生成するシステムにおいて、2 工具による同時加工を考慮した加工順序を遺伝的アルゴリズム (GA) を用いて求める方法について提案した。同時加工するグループであるサイクルという単位に全工程を分割し、各サイクル内で LPT 法を用いて 2 個の刃物台に工程を分配するようにした。これにより、GA を用いやすい形に問題を帰着できた。シミュレーションの結果、2 種類の交叉を使い分けることにより、工程数が多くなっても短い時間で最適な形に工程を分割できることを確認できた。

今後の課題としては、より効率の良い手法の考案が挙げられる。この問題には 2 章で述べたような種々の制約条件があり、これを利用した染色体の表現方法やほかのヒューリスティックスも有効であると思われる。さらに、2 スピンドル機への手法の拡張、ユーザノウハウの学習、切削条件の最適化なども重要な課題である。

日ごろ貴重な御助言を頂くオークマ (株) 営業技術部日比明主査に感謝いたします。また熱心に討論して頂く名古屋大学工学部大西研究室の皆様にも感謝いたします。

本研究の一部は、(財)人工知能研究振興財団の助成による。

参 考 文 献

- 1) 花木, 日比, 深谷, 水上: AI 応用自動プログラミングシステム, 日本機械学会第 2 回 FAN シンポジウム講演論文集, 59/62 (1992)
- 2) 井上, 本多: 加工用ソフトウェア, 精密機械, 47-11, 1359/1365 (1981)
- 3) 人見勝人: 生産システム工学, 共立出版 (1990)
- 4) CAD/CAM/CAE 利用実態調査, 日経メカニカル, 407, 60/66 (1993)
- 5) 平成元年度生産技術高度化に関する調査研究報告書 (基盤加工技術体系化), 日本機械工業連合会・機械技術協会 (1990)

- 6) 花木, 山本: 人間中心の CIM, 平成 4 年電気学会産業応用部門全国大会講演論文集, S. 278/S.283 (1992)
- 7) 山下: NC 旋盤における多機能化技術, 機械と工具, 33-4, 49 (1989)
- 8) 河端, 白瀬, 笹山, 藤村, 安井: NC 旋盤の生産性向上に関する研究, 精密工学会誌, 60-4, 576/580 (1994)
- 9) NC 旋盤活用マニュアル, 大河出版 (1988)
- 10) Y. Mizukami, Y. Fukaya, H. Matsushita, Y. Hayakawa, Y. Hanaki, N. Ohnishi and N. Sugie: A DP-Based Method of Planning Simultaneous Machining Sequence for Turning, Proceedings of 1994 JAPAN-U. S.A. SYMPOSIUM ON FLEXIBLE AUTOMATION, 513/519 (1994)
- 11) 西川, 三宮, 茨木: 最適化, 岩波書店, 125/126 (1985)
- 12) 北野宏明: 遺伝的アルゴリズム, 産業図書 (1993)
- 13) 小林重信: 遺伝的アルゴリズムの現状と課題, 計測と制御, 32-1, 2/9 (1993)

[著者紹介]

水 上 裕 登 (正会員)

1987 年, 名古屋大学大学院工学研究科情報工学専攻修士課程修了。同年(株)大隈鐵工所((現)オークマ(株))入社。CNC 旋盤の自動プログラミングシステムの研究・開発に従事。精密工学会の会員。

松 下 裕 一

1979 年, 名古屋大学大学院電気系専攻修士課程修了。同年(株)大隈鐵工所((現)オークマ(株))入社。主に数値制御装置と数値制御情報作成装置の開発に従事。

花 木 義 麿

1965 年, 名古屋大学工学部電気工学科卒業。同年(株)大隈鐵工所((現)オークマ(株))入社。数値制御装置, サーボシステム, FMS などの開発に従事。

大 西 昇 (正会員)

1975 年, 名古屋大学大学院工学研究科電気系専攻修士課程修了。同年労働福祉事業団労災リハビリテーション工学センター研究員。86 年名古屋大学工学部講師。89 年同助教授。94 年同大学工学部情報工学科教授。生体工学, 福祉工学, 人工知能などの研究・教育に従事(工学博士)。

杉 江 昇 (正会員)

1957 年, 名古屋大学工学部電気学科卒業。同年通商産業省電子技術総合研究所入所。62~64 年カナダ・マギル大学客員研究員。79 年名古屋大学大学院工学研究科情報工学専攻教授。94 年名城大学理工学部電気電子工学科教授。バイオニクス, 医用工学, コンピュータビジョン, 自然言語処理などの研究・教育に従事(工学博士)。