生産環境の変動への柔軟な対応を志向した加工工程設計支援の取組み

森永 英二*・若松 栄史*・荒井 栄司*

1. はじめに

工程設計は、機械加工において加工の手順や方法を決定する過程である。製品設計とか加工の橋渡し的な位置付けにあり、加工の良否に大きな影響を及ぼす重要な過程となっている。製品設計に対するCAD (Computer Aided Design)や加工作業に対するCAM (Computer Aided Manufacturing) と同じく、CAPP (Computer Aided Process Planning) という呼称のもとに、その確実な実施のための計算機支援技術が厚く熱く論じられてい る [1,2]。それは“variant approach”に基づくものと“generative approach”に基づくものに大別される。前者は、初期に検討された対話型のアプローチであり、類似した部品に対して過去に設計された工程を検索し、工程案として出力し、それを設計者が必要に応じて適宜修正することで工程を得るものである。後者は、工程設計の完全自動化を目指すものであり、入力された形状情報に対し、決定アルゴリズムや知識ベースによって工程案を生成するものである。複雑な形状の取扱いを考えると、作業者の介入を必要としない後のアプローチが好ましく、それにに基づく手法が活発に論じられている。しかしながら、暗黙知である設計者の経験、ノウハウを形式化して支援手法に取り込むことは難しく、十分な支援に結びついていない状況にある。

皆が同じものを欲した少品种多量生産の時代が過ぎ、消費者ニーズの多様化および多品種少量生産への対応が、この四半世紀に進められてきた。近年、消費者ニーズのさらなる多様化とニーズの急速な変化によって需要予測が困難な時代となっており、情報の変化に敏捷に対応する柔軟性をもつ、変化に即座に対応するため、高度に自動化および知能化された生産システムの構築が、工　程設計支援においても、これらの要求に応えることが必須といえる。本稿では、生産環境の変動に対する柔軟性の観点から、筆者の所属する研究組織で取り組まれてきた、一定の工程設計支援手法について紹介する。

2. フレキシブル加工工程設計手法

2.1 加工フィーチャと工程設計

計算機が解釈可能な工場製造データの表現および交換の国際規格がISO 10303 シリーズとして策定されている。CADで設計された製品情報をCAMに渡すために使用することを意図して、加工部位を表現するための加工フィーチャが、この規格の中で定義されている。加工フィーチャは、加工部位を平面や溝、ボケットなどに分類して表現したものであり（第1図），同シリーズのISO 14649 シリーズで規定された情報モデルによって、加工工程や設計の詳細内容の情報と関連付けて、データを取り扱うことができる。このため、製品の形状と加工の方法や使用する工具などが関連付けられることになり、加工フィーチャは工程設計の自動化に有効なツールとなる。

2.2 加工フィーチャに基づく工程設計

他のアプローチと同様に、上述の加工フィーチャに基づく工程設計支援のアプローチも活発に論じられ、さまざまな手法が提案されている [2]。しかし、これらの手法は一般的には、得られる工程案が一通りに限られたため、加工中に、工具の破損などの状況の変化が発生した場合に、対応ができない、この問題に対して、白瀬ら [4] は、NCプログラム作成が不要で加工作業の変更に柔軟に対応可能な自律型工作機械の実現に関する研究の中で、複数の工程案を事前に作成し、それらの中から最良のものを選択する手法を提案している。この手法では、以下の手順に沿った情報処理によって工程案が生成される。

(1) 加工除去領域の抽出

被削材と製品の形状から削除すべき領域（加工除去領域）を抽出する（第2図）。

(2) 加工プリミティブの組合せの生成

z 軸に沿って被削材に接近する工具への負荷を考慮し、はじめにz軸に垂直な平面で加工除去領域を分割した後に、以下のルールに則ってさらに分割し、加工プリミティブの組合せを得る（第3図）。

* 大阪大学 大学院工学研究科 マテリアル生産科学研究

Key Words: computer aided process planning, flexibility, machining, agile manufacturing.
（a）yz平面に平行な平面で分割する（分割案1）。
（b）xz平面に平行な平面で分割する（分割案2）。
（c）できるだけ大きい加工プリミティブが得られる
ように，yz平面に平行な平面とxz平面とで分割する（分割案3）。

（3）加工順序の決定
前のステップで得られた加工プリミティブの組合せのそれぞれに対して，各加工プリミティブの加工順序を，以下のルールに則って決定する。
● z座標値の大きい加工プリミティブから（すなわち，上にあるものから）加工する。
● z座標値が同じ場合には，体積の大きい加工プリミティブから加工する。
● 体積も同じ場合には，オープンフェース1の多い加工プリミティブから加工する。
● 上記で順序を決定できない場合には，被削材の中央に近い加工プリミティブから加工する。
● 穴は最後に加工する。

（4）加工フィーチャの認識
各加工プリミティブが有するオープンフェースの数と稜線の関係（第1表）から，その加工プリミティブを加工フィーチャとして認識する。ここまでの操作によって，加工フィーチャの組合せとその加工順序（すなわち工程案）が算出される。

（5）使用工具の選択と工具径の決定
各加工フィーチャに対して，フィーチャの種類と寸法から，第2表のルールに則って，使用可能な工具群の中から使用工具を決定する。

（6）切削条件の決定[5]
過去の加工事例を蓄積したデータベースを参照しながら切削条件の推論する事例ベース推論によって，切削条件を決定する。被削材，工具材種，工具の表面処理，仕上げ条件を入力して検出した類似事例に含まれる切削条件に対し，加工内容の

第1表 加工プリミティブの条件と加工フィーチャとの関係
（OF：オープンフェース）

<table>
<thead>
<tr>
<th>フィーチャ</th>
<th>製品側形状</th>
<th>OF数</th>
</tr>
</thead>
<tbody>
<tr>
<td>Closed Pocket</td>
<td>□</td>
<td>1</td>
</tr>
<tr>
<td>Open Pocket</td>
<td>□</td>
<td>3</td>
</tr>
<tr>
<td>Closed Slot</td>
<td>□</td>
<td>2</td>
</tr>
<tr>
<td>Open Slot</td>
<td>□</td>
<td>3</td>
</tr>
<tr>
<td>Face</td>
<td>□</td>
<td>5</td>
</tr>
<tr>
<td>Step</td>
<td>□</td>
<td>4</td>
</tr>
<tr>
<td>Through Hole</td>
<td>□</td>
<td>2</td>
</tr>
<tr>
<td>Blind Hole</td>
<td>□</td>
<td>1</td>
</tr>
<tr>
<td>Free Form</td>
<td>□</td>
<td>-</td>
</tr>
</tbody>
</table>

第2表 使用工具選択および工具径決定ルール

<table>
<thead>
<tr>
<th>フィーチャ</th>
<th>工具決定ルール（x：工具径）</th>
</tr>
</thead>
<tbody>
<tr>
<td>Closed Pocket</td>
<td>次式を満たす最大径のエンドミル</td>
</tr>
<tr>
<td></td>
<td>5r ≤ l ≤ w (l, wはポケットの幅・奥行)</td>
</tr>
<tr>
<td>Open Pocket</td>
<td>最大径のフラットエンドミル</td>
</tr>
<tr>
<td>Closed Slot</td>
<td>次式を満たす最大径のエンドミル</td>
</tr>
<tr>
<td></td>
<td>3r ≤ w (wはスロットの幅)</td>
</tr>
<tr>
<td>Open Slot</td>
<td>Closed Slotと同じ</td>
</tr>
<tr>
<td>Face</td>
<td>最大径の正面フライス</td>
</tr>
<tr>
<td>Step</td>
<td>Faceと同じ</td>
</tr>
<tr>
<td>Through Hole</td>
<td>穴径に合致するドリル</td>
</tr>
<tr>
<td>Blind Hole</td>
<td>穴径に合致するドリル</td>
</tr>
<tr>
<td>Free Form</td>
<td>最大径のボールエンドミル</td>
</tr>
</tbody>
</table>

違いを考慮した修正が施され，それぞれの平均値が算出として用いられて，切削条件として決定される。

（7）加工能率の評価
各工程案に対して，上記2つの処理によって作業設計が完了するため，各工程の総加工時間Tを次式により計算して，各工程案を評価することが可能となる。Tが最小となるものが最適工程として選択される。ここで，nは加工フィーチャの数，mは工具の交換回数，t_eは工具交換時間，v_i，a_i，r_i，f_iはそれぞれi番目の加工フィーチャの体積と，x方向切込み，半径方向切込み，送り速度を表す。
式 (1)で示すように、この結合処理は、加工除去領域の分割を行い、それらの分割領域を組合せることで、最終的に、得られる加工ブリティプの組合せ数を最大化することが可能となる。

2.3 最小分割・再構成による柔軟性の増大

上記の手法では、加工除去領域を分割し、加工ブリティプを生成する段階において、3通りの分割のみを行っていたため、少数の工程案しか得られないことになる。分割の仕方は他にも多数存在し、それらも考慮することで、より優れた工程が得られる可能性がある。そのため、一つのアプローチとして、加工除去領域から直接加工ブリティプを生成するのではなく、いったん加工除去領域を限界で細かく分割した後、それらの細分化された領域に適宜結合させて、加工ブリティプを生成することができる（第3項）。なお、この操作は、前述の (2) の処理を、以下の二つの処理に置き換えることで可能となる。

(2)'1) 加工除去領域の最小分割

加工除去領域を、xy, yz, zx 平面にそれぞれ平行な平面で一気に分割する（第 4 図）。これによって得られた直方体領域を最小凸多面領域とよぶ。

(2)'2) 最小凸多面領域の再結合

最小凸多面領域同士を再結合し、加工ブリティプの組合せを以下の手順で生成する。

(a) 加工時の工具への荷負を考慮し、最小凸多面領域群を、同じ形状をもつ領域群ごとグループ分け（第 5 図）。

(b) 各グループ内で、凸多面領域が得られなくななるまで、隣接する領域同士を結合させていき、それらの凸多面領域を加工ブリティプとする（第 6 図）。

(c) 各グループに対する各再結合結果を、グループ間で結合させて、最終的な加工ブリティプの組合せを生成する。

なお、製品形状の一部に精度情報が設定されている場合、加工後の精度の観点から、その部分に対応する加工除去領域辺の部分は、同じ工程で加工除去されることが望ましい。しかし、このことを考慮せずに最小分割・再結合処理を行うと、本来同じ工程で加工されるべき数の領域が、別々の加工ブリティプに割り当てられてしまうおそれがある。隣接領域との結合処理 (2)'2b) の実行において、隣接する最小凸多面領域が同一仕上げ情報を持つ場合には、これらの結合を優先することで、そのような加工ブリティプの組合せが生成されることは多い。

この再結合操作をすべての結合の順序・組合せによって行うことにより、得られる加工ブリティプの組合せの数を大幅に増加させることができる（第 5 図）。

2.4 凹部分割による計算量の低減

最小分割・再結合処理の導入によって、得られる加工ブリティプの組合せ数の増加は得られる工程案の
数が大幅に増大する。しかし、この結合法に必要となる計算量は大きく、手法全体としての計算負荷が深刻なものとなる。したがって、最小単位での分解と再結合処理を行わず、同等の加工ブリミティブの組合せを得ることが望ましい。この観点から、最小分割・再結合処理を、さらに、次のような処理に置き換える[7]。

(2°-1) z分割
加工時の工具への負荷を考慮して、z 軸に垂直な面で加工除去領域を最初に分割する（第 8 図）。

(2°-2) 四分割
分割された加工除去領域の各小領域に対し、その輪郭の四部分から、yz または zx 平面に平行な切断面を生成して分割を行う。その際、切断面は、それより前に生成された切断面を越えて分割を続けることは許さず、その切断面に接した時点で、当該切断面による分割を終えることとする（第 9 図）。

(2°-3) 異なる z 座標値に対する分割結果同士を結合させて、加工ブリミティブの組合せを生成する。

第 8 図 z分割

第 9 図 四分割

第 2 図の例に対して、この方法で生成された加工ブリミティブの組合せ群を第 10 図に示す。最小分割・再結合を介して生成した第 7 図の結果と比較すると、ブリミティブの数が最小となる組合せはすべて生成されているが、全組合せを生成することはできていない。

第 10 図 四分割法で得られた加工ブリミティブの組合せ群

つぎに、計算量の点での評価のため、第 11 図に示す階段状の形状を有する加工除去領域を想定して、最小分割・再結合による手法と四分割による手法のそれぞれの処理に必要となる計算回数を比較する。両方で処理が異なり共通の評価指標の設定が困難であるため、ここでは、組合せを内包している処理（最小分割・再結合法では再結合処理、四分割法では四分割処理）における計算回数を比較する。第 12 図の結果が示すように、四分割法の方が計算負荷を大幅に抑えられることがわかる。

第 11 図 計算量評価用モデル

第 12 図 計算量の比較結果

ブリミティブの数が少なければ、サイズの大きい加工ブリミティブが得られて往の大きな工具で効率よく加工できると期待できる。また、工具交換回数も減少することが期待でき、かつ、工具交換や位置決めにおける誤差要因を含むリスクを低減することが期待できる。このため、第 10 図の結果は、十分な結果であるといえる。しかし、多くの工程を生成して不都合な事態に対する柔軟な対応を可能にするということを考えると、第 10 図の結果は、満足には至らない結果といえ、さらなる検討が必要であるという。一方で、計算負荷の点では、四分割法による手法は大きな効果があり、複雑な形状の取扱いが可能である。実際の加工への適用を考えると、この手法は大きなポテンシャルを有しているといえる。

3. 多軸加工に向けた拡張

前節の手法は 3 軸の工作機械によるプライス加工を対象としており、加工方向を 1 方向に固定した場合の最適工程が出力されることになる。近年、多軸加工機の普及が進んでいることを考慮すると、これらの手法を、多方向からの加工を考慮したものに拡張することが望ましい。本節では、この観点での組合せ[7]について述べる。

ここでは、被削材および製品形状が xyz 直交座標系のいずれかの軸に垂直な平面によって形成されるものに限定し、工具はいずれかの軸の方向に沿って被削材に接近・離脱するものとする。このような前提のもと、複数方向からの加工を考慮に入れた手法に拡張するためには、以
下の3項目への対応が必要となる。

- 加工クリティフの組合せを生成する過程において、3次元的な分割を行うこと。
- 加工方向の情報を処理に組み込むこと。
- 加工順序によっては不可能な加工方向が存在するため、方向と順序の情報を一体で処理に組み込むこと。

これらの要因への対応を施した処理手順を以下に示す。

(1) 加工除去領域の抽出
(2) 3次元四部分割による加工クリティフの生成
(3) 加工方向と加工順序の組合せの生成
(4) 加工フィーチャの認識
(5) 使用工具の選択と工具径の決定
(6) 切割条件の決定
(7) 加工能率の評価

手順 (1) と (4) 以降は前節の手法と同一である。以下では、残りの手順 (2) と (3) の詳細については述べる。

3.1 3次元四部分割による加工クリティフ生成

前節では、2軸に沿って被削材に接続する工具の荷重を考慮し、先に2軸に垂直な平面で加工除去領域を分割した後に、4部分割処理を行っていた。ここではそのような制約はないと、加工除去領域そのものに対し、その軸の4方向から、xy, yz, zx平面のどれかに平行な切断面を生成して分割を行う。切断面は、それより前に生成された切断面を越えて分割を継続することはできず、その切断面に接した時点での状態を生成する。すなわち、3次元四部分割を施して得られる加工クリティフの組合せを第3図に示す。x, y軸に沿った方向からの工具接近を考慮することで得られる。組合せ11〜15が新たに得られている。

第13図 3次元四部分割によって生成された加工クリティフの組合せ（図中の数字はクリティフのID番号）

3.2 加工方向と加工順序の組合せの生成

加工可能な方向として、±x, ±y, ±zの6方向を考え（被削材の保持による制約はここでは無視する）、加工クリティフ数をnとおくと、加工順序はn!通り存在するため、加工方向と加工順序の組合せの数Nは次式のように

\[ N = n! \cdot 6^n \]  (2)

これらの組合せそれぞれに対して、加工フィーチャ認識と作業設定を行って、最適な工順を抽出することになるが、(2)式が示すように、加工方向と加工順序の組合せは膨大に存在するため、これは計算量の面から現実的ではない。このため、希望可能な組合せ以外はあらかじめ削除する必要がある。そこで、各組合せと製品形状との関係を考えると、これらの組合せの中で、実行不可能なものが多く含まれていることがある。たとえば、第13図中の加工クリティフの組合せにおいては、ID番号1番のクリティフは、3番のクリティフが先に加工されない限り、+y側から工具を接近させて加工することはできない。また、製品形状が存在するため、−x側からはどのような順序でも加工することはできない。このような幾何的制約を、あらかじめCAD情報から作成し、データとして作成しており、加工方向と加工順序の組合せを生成するこの段階でその内容を参照することにより、実行不可能なものすべてを除外できる。筆者らは、このようなデータを以下のような形式でマトリクス情報として作成しておくことを提案し、加工可能方向マトリクスと称している。これは、各クリティフの指定方向に対して、何か存在しているかマトリクス状に表したもので、中の文字列や数字は以下のことを表す。

1 正の整数：
当該クリティフの当該方向には、記載された整数をID番号にもつクリティフが存在する。このため、それらを加工した後でないと、当該クリティフをこの方向から加工することはできない。

2 “OF”：
当該クリティフの当該方向には何も存在せず、したがって、いつの時点でそのクリティフをその方向から削除することが可能である。

3 “P”：
当該クリティフの当該方向には、製品が存在するため、この方向からはどのような順序であっても加工できない。

-26-
上記の加工ブリティプの組合せ1に対しての加工可能方向マトリックスを第3表に示す。この場合、加工方向と加工順序の組合せは全部で31104通り存在するが、このマトリックスを参照することで、28914通りが実行不可能であることが抽出され、2190通りの組合せのみに対して、フィーチャ認識以降の処理を行えばよいことになる。

<table>
<thead>
<tr>
<th>ID番号</th>
<th>方向</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>OF</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>OF</td>
</tr>
<tr>
<td>4</td>
<td>OF</td>
</tr>
</tbody>
</table>

4. おわりに

本稿では、不測の事態に対する柔軟性を考慮した機械加工工程設計支援について、著者が所属する組織で開発されてきた一連の手法を紹介した。これらの手法は、複数の工程案をあらかじめ生成し、大まかな作業設計と工程案の評価を行って、最良の工程を解として出力する仕組みを有しており、工具破損などが発生した場合に、作業設計以降の部分のみを新しい状況に合わせて再実施することで、短時間に新たな最適工程を得ることができる。これら的手法を実用に近づけていくための課題として、以下があげられる。一つは、複雑な形状に適用できるよう、計算負荷のさらなる低減が必要であり、数理計画問題への帰着と求解を介して解決を図る組み立てを進めている[8]。二つ目は、生産スケジュールを同時に考慮した手法への拡張であり、差分則によるジョブ割当てとの融合を進めている[9]。また、近年、複数加工機による工程組み合わせの生産が注目されており、フレキシ加工と旋削加工を同時に扱える手法への拡張も今後の課題としてあげられる。

(2012年11月5日受付)

参考文献