

低軸 CNC 환경에서의 효율적인 황삭가공

강자훈, 서석환, 이정재

Computer Automated Manufacturing Lab
포항공과대학교 산업공학과

Abstract

다축가공은 3축 이상의 동시제어축을 이용하여 복잡한 형상을 효율적으로 가공할 수 있는 첨예의 기술인 반면, 가공설비의 고가로 인해 실제현장에 보급되지 못하고 있는 실정이다. 부가축 방식에 의한 低軸化 가공방식은 이러한 현실적 문제에 대처할 수 있는 강력한 방식으로서, 본 연구팀에서는 3축 CNC 공작기계에 부가축 테이블 방식을 이용하여 5축 곡면가공을 구현한 바 있으며, 정삭가공 알고리즘을 개발한 바 있다. 본 연구에서는 부가축 환경하에서 황삭가공 알고리즘을 다루며, 기존의 全軸환경의 황삭가공에 비해 공구자세를 인덱싱 형태로 변화시킬 수 있다는 차이가 있으며, 이에 따라 자세조정횟수의 최소화가 생산성 지표로 부각된다. 본 연구에서 개발된 황삭경로 알고리즘은 자세조정횟수를 포함하여 공구접근영역, 공구교환횟수, 퍼드조정을 통하여 전체적으로 황삭가공시간의 최소화로 접근하였다. 연구된 알고리즘은 컴퓨터시뮬레이션을 통하여 검증하였으며, 실제절삭을 통한 검증이 추진중에 있다.

1. 서 론

황삭가공은 불필요한 영역을 제거하는 과정으로서, 가공시간의 대부분을 차지하며, 가공정확도를 고려하지 않아도 되는 특징을 갖고 있다. 따라서 황삭가공의 효율성은 가공정밀도가 아닌 가공시간에 의해 평가되고, 효율적인 황삭가공이란 소재제거율의 극대화를 통한 가공시간의 최소화로 볼 수 있다. 이를 위해서는 가) 사용공구의 최적선택, 나) 공구 둘 횟수를 최소화 할 수 있는 공구경로, 다) 이송속도의 조절을 통하여 소재제거율을 높이는 방안이 강구되어야 하며, 과거에 수행된 많은 연구가 있다.

기존의 연구를 살펴보면, 대부분의 연구는 3축가공에 국한되고 있으며 [2-5], 공구의 자세변화가 가능한 4축 이상의 가공을 대상으로 한 연구는 소수에 불과하다 [1,6]. 특히, 본 연구에서 다루고자하는 저축화 환경, 즉 3축의 CNC에 부가테이블을 장착함으로써 가공에 요구되는 동시제어축 보다 적은 기계환경에서의 황삭알고리즘에 관한 연구는 전무한 상태이다. 본 연구팀에서는 이전에 3축 CNC에 인덱싱 테이블을 장착한 부가축 환경에서 임펠러와 같은 5축형상을 가공할 수 있는 하드웨어환경을 구현한 바 있으며, 정삭경로 알고리즘을 개발하고 유효성을 실제가공을 통하여 검증한 바 있다[1]. 이러한 환경에서는 공구의 자세가 공작물의 셋업을 통해 이루어 지기 때문에 셋업횟수의 최소화가 가공의 효율성을 좌우하는 주요 문제로 대두되며, 이는 황삭가공에서도 마찬가지이다.

본 연구는 인덱싱 방식에 의한 부가축환경에서 뿐만 아니라, 공구의 자세가 연속적으로 변화될 수 있는 일반적인 4축 이상의 다축가공환경에서 직면하는 황삭가공문제를 대상으로, 전체가공시간을 최소화 할 수 있는 효율적인 알고리즘에 관한다. (일반적인 다축환경과 인덱싱에 의한 부가축환경과의 차이는 셋업변경시간의 차이에 있으며, 이는 비용함수의 조절을 통하여 같은 알고리즘을 적용할 수 있음을 주기 한다.) 근본적으로, 기존의 황삭경로 알고리즘에 비해 본 연-

구에서 다루는 문제와의 차이는 공구자세의 변경을 고려하는 것으로 볼 수 있으며, 공구자세에 따라 가공가능영역이 변하기 때문에 전역을 커버하는 최소자세 결정문제가 주요 이슈로 부각된다.

본 논문에서는 이 문제와 더불어 황삭가공의 기본적인 이슈인 공구크기 선정, 공구교환횟수, 공구경로산출 문제를 통합적인 차원에서 최적화를 기할 수 있는 방식을 다음과 같이 단계별로 접근한다. 가) 주어진 곡면의 3차원convex hull 오퍼레이션을 통한 간접가능영역 탐색, 나) 간접가능영역을 대상으로 network graph 이론과 branch and bound 기법을 이용하여 가공시간을 최소화 할 수 있는 다수의 공구규격 및 공구자세를 도출하고, 다) 선정된 공구자세 및 공구크기에 따른 가공영역의 결정 및 황삭경로를 산출하고, 라) 최대 절삭력 범위내에서 이송속도를 극대화할 수 있는 이송속도를 조정한다.

2. 간접가능성에 따른 가공영역 구분

2.1. 간접가능영역 및 간접미발생영역 산출

황삭가공시 전체 가공영역은 공구몸체간접 가능성에 따라 간접이 발생할 수 있는 영역(간접가능영역)과 간접이 발생하지 않는 영역(간접미발생영역)으로 구분할 수 있다. 일반적으로, 간접가능영역은 합물된 부위에 존재하며, 간접미발생 영역은 합물된 지역을 제외한 가공영역에 존재한다. 예컨대, 그림 1에서 a 영역은 공구 t1 과 자세 o1 에 의해 전체를 가공하는 것이 가능하나, 공구 및 자세 (t1, o2), (t2, o1), (t2, o2) 에서는 공구몸체간접에 의해 전체영역을 가공하는 것이 불가능함을 보여준다 (따라서, a 지역은 간접가능영역). 반면에, b 영역은 공구크기와 자세에 상관없이 (예컨대, (t1, o1), (t1, o2), (t2, o1) 과 (t2, o2)) 간접없이 가공할 수 있기 때문에 간접미발생영역으로 분류된다.

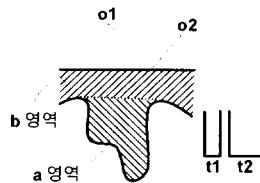


그림 1. 간접가능성에 따른 가공영역의 구분

간접가능영역은 convex hull과 오일러 연산을 이용하여 구할 수 있다. 식 (1)은 오일러 연산을 이용하여 convex hull에서 간접가능영역과 간접미발생영역을 구하는 과정을 나타낸다.

$$\begin{aligned} \text{간접가능영역}(l, m, n) &= \\ \text{convex_hull}(l, m, n) - \text{가공형상}(l, m, n) & \quad (1) \end{aligned}$$

$$\begin{aligned} \text{간접미발생영역}(l, m, n) &= \\ \text{초기피삭재}(l, m, n) - \text{convex_hull}(l, m, n) \end{aligned}$$

convex hull은 가공형상을 폴리헤드론으로 근사시킨 모델에서 폴리곤을 구성하는 모든 모서리 점을 추출한 후, devide-and-conquer type 방법[7]을 적용하여 구할 수 있으며, convex hull과 가공형상 사이의 오일러 연산은 가공형상의 복셀화(voxelization)를 통해 쉽게 구현될 수 있다 [8]. 이상에 의해 간접가능영역 및 간접미발생영역을 산출하는 과정은 다음과 같다.

- 단계 1. 형상을 폴리헤드론 모델로 나타낸다.
- 단계 2. 폴리헤드론 모델에서 모든 모서리 점을 추출한 후, 추출점들로부터 convex hull을 구하고 이를 폴리헤드론 모델로 표현한다.
- 단계 3. 초기피삭재, 가공형상 및 convex hull의 폴리헤드론 모델을 복셀화한다.
- 단계 4. 식 (1)을 이용하여 간접가능영역과 간접미발생영역을 구한다.

2.2. 간접가능영역의 세분화

위와 같이 가공영역을 구분하는 것은: 가) 공구크기 및 자세산출시 대상지역을 전체가공영역에서 간접가능영역으로 축소시킬 수 있으며, 나) 간접미발생 영역에 대해서는 (주어진 공구집합 중) 가장 큰공구를 배치시킴으로써 소제제거율을 극대화 할 수 있다. 본 연구에서는 간접가능영역을 다시 몇개의 독립된 영역(형상의 특징에 따라, 스텝, 포켓, 슬롯, 구멍 등)으로 세분화 시키는데 (그림 2), 이렇게 함으로써 특징형상에 따른 가공경로를 산출할 수 있고, 아울러 영역간 이동거리를 줄일 수 있도록 전체경로를 설계할 수 있는 장점이 있다.



그림 2. 간접가능영역의 세분화

3. 공구크기와 자세에 따른 절삭영역 및 미절삭영역 산출

추출된 간접가능영역은 식 (1)에서 구해진 복셀의 집합으로 표현되며, 각 복셀은 일정한 부피를 지닌 단위영역이 된다. 간접가능영역에 대한 가공가능성은 단위영역별로 모든 공구크기 및 자세에 대한 검색형태로 이루어 진다. 단위영역을 기본으로 하는 것은 가공볼륨의 이산화를 위한 것으로서, 그림 3과 같이 단위영역의 간격으로 인한 오차를 감수할 수밖에 없다. 이 오차는 단위간격을 줄임으로써 (메모리의 증가와 trade-off) 줄일 수 있고, 단위영역 안쪽으로 들어오는 미세영역과 바깥쪽으로 나가는 미세영역이 평균적으로 같다는 (그림 3) 매크로 한 관점에서 본다면 큰 문제는 아니라고 볼 수 있다.

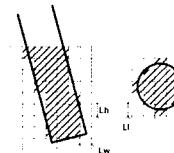


그림 3. 절삭영역의 이산화

각 단위영역에 대한 가공가능성 여부는 1 또는 0에 의해 표현이 가능하다. 이는 모든 공구, 자세 및 단위영역의 조합에 대해 이진값으로 표현하고 EBMT (Extended Binary Machinability Table)의 형태로 구체화시킬 수 있다. 이하에서는 접근가능자세의 산출 및 EBMT의 생성방법을 기술한다.

3.1. 접근가능자세의 산출

접근가능자세는 공구가 취할 수 있는 모든 자세집합의 부분집합으로서 간접가능영역에 접근할 수 있는 자세집합을 뜻한다. 그림 4는 간접가능영역에 대한 접근가능자세집합 예로서, 다음의 과정을 거쳐 단위구(unit sphere)상에 표현할 수 있다.

- 단계 1. 간접가능영역에 포함된 복셀 중 간접미발생영역과 접하는 복셀들을 찾고, 그 중심점을 구한다.
- 단계 2. 구해진 중심점에서 표면에 수직인 근사화된 법선벡터를 구한다.
- 단계 3. 법선 벡터를 중심으로 하는 단위 반구(unit hemisphere)를 구한다.
- 단계 4. 단위반구들의 합집합을 접근가능자세로 한다.

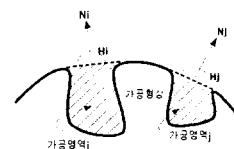


그림 4. 접근가능자세 예

3.2. 간접가능영역에 대한 EBMT의 생성

EBMT는 개념적으로 공구, 자세, 간접가능영역을 인덱스화한 3축 좌표계에 1 또는 0으로 표기한 3차원 행렬로 볼 수 있다. 3차원 그리드상의 값은 식(2)와 같이 공구크기 i로

단위영역 k 를 공구자세 j 로 가공가능하면 1의 값을 갖는다.

$$EBMT(i, j, k) = \begin{cases} 1, & \text{if unit area } k \text{ can be machined without interference, by tool } i, \text{ orientation of } j \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

모든 조합에 대해 생성된 EBMT는 이후의 최적화 과정에서 필요한 정보를 제공하며, 기본적으로 가) 자세 및 공구에 따른 가공가능성 여부와, 나) 가공가능한 단위영역의 갯수를 조사함으로서 가공가능한 부피를 쉽게 구할 수 있다 (식 (3)).

$$MV(i, j) = dV * \sum_{k=1}^1 EBMT(i, j, k) \quad (3)$$

where, dV = Volume of voxel

공구 i , 자세 j 에서 단위영역 k 의 가공가능성 여부, 즉 $EBMT(i, j, k)$ 값을 도출하는 과정은 다음과 같다 (그림 5 참고).

- 1) for $\forall t_i, t_i \in \{tool\}$
- 2) for $\forall o_j, o_j \in \{\text{접근가능자세}\}$
- 3) 자세 O_j 를 (ϕ, θ) 값으로 환산한 후, $(-\phi, -\theta)$ 만큼 형상을 회전한다.
- 4) x, y 평면상에서 회전된 가공영역의 최대 x, y 값을 계산한다.
- 5) 최대, 최소범위내에서 일정한 간격으로 x, y 값을 증가하면서 공구를 위치시킨다.

$$\text{for } \forall x, x \in \{x | x = \max_x + u * (\max_x - \min_x), 0 \leq u \leq 1\}$$

$$\text{for } \forall y, y \in \{y | y = \max_y + v * (\max_y - \min_y), 0 \leq v \leq 1\}$$

- 6) 위치 x, y 에서 가공형상과 간섭이 발생하지 않는 최소높이 z 를 찾고 현재위치를 $p(x, y, z)$ 로 한다.
- 7) 단계 6)에서 찾아진 연속한 두점 사이를 지나는 공구 불륨과 만나는 단위영역에 대해 $EBMT$ 값을 1로 설정한다.

$$EBMT(i, j, k) = 1, \text{ if } E(P_1, P_2) \cap \text{단위영역}(k) = 1$$

where, $E(P_1, P_2) : \int_{P_1}^{P_2} T(x) dx$

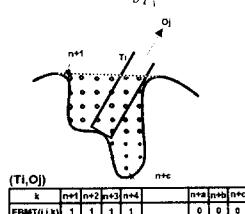


그림 5. $EBMT(i, j, k)$ 의 결정.

4. 최적자세 및 공구셋 산출

최적자세 및 공구셋은 M 개의 공구와 N 개의 공구자세가 주어진 경우: 가) 간섭가능 전영역을 제거하는 가공시간과, 나) 공구교환시간 다) 공구자세변경시간과, 라) 황삭가공후 남은 영역을 축삭 또는 정삭공구로 가공할 때 필요한 가공시간의 합을 최소로 하는 공구 및 자세로 정의할 수 있다. 따라서, 황삭가공의 최적화문제는 수행도 지표 식(4)를 최소화하는 공구집합과 공구자세집합을 구하는 것이다. 가공시간은 가공영역의 부피와 소재제거율값에 의해 근사화되어 산출된다. 소재제거율은 Machining Data Handbook[11]에서 제시되는 수치를 이용한다.

$$\text{Min} \left\{ \sum_{i=1}^m \left[\frac{\bigcup_{j=1}^n \alpha_{i,j} * MV(i, j)}{\text{MRR}(i)} \right] + \text{Min}(1, \text{Max}(\sum_j \alpha_{i,j}, 0)) * dT + \sum_j \alpha_{ij} * dO + \beta * UV \right\} \quad (4)$$

where

$$\alpha_{ij} : \begin{cases} 1, & \text{if tool } i, \text{ orientation } j \text{ is used} \\ 0, & \text{otherwise} \end{cases}$$

$$dT : \text{Tool change time}$$

$$dO : \text{Setup change time}$$

$$UV : \text{Uncut volume}$$

$$\beta : \text{Uncut volume} \text{에 대한 축삭시간고려 weight}$$

식 (4)는 비선형 문제로서 분석적 방법에 의해서 직접 정확한 해를 구하기가 매우 힘들다. 따라서 본 연구에서는 실용적인 축면을 강조하여 식 (4)를 선정될 공구 및 자세 수에 따라 네가지의 문제로 세분화하여 접근하였다: 가) 한 공구와 한 자세를 선정하는 경우, 나) 한 공구와 다수 자세를 선정하는 경우, 다) 다수 공구와 한 자세를 선정하는 경우, 라) 다수 공구와 다수 자세를 선정하는 경우. 현재 본 연구에서는 라) 문제에 대한 연구가 진행중에 있어, 다음에서는 나머지 3 개 문제에 대한 접근방식을 제시한다.

4.1 한공구, 한자세 산출

한공구, 한자세의 사용은 가공시 공구교환 및 자세변경이 발생하지 않는다. 따라서 식 (4)에서 공구교환요소 및 자세변경요소를 고려하지 않아도 되며, 이는 식 (4)를 아래와 같이 단순화시킨다.

$$\text{Min} \left\{ \frac{MV(i, j)}{\text{MRR}(i)} + \beta * UV \right\} \quad (5)$$

식 (5)는 공구집합 및 자세집합에 포함된 각 원소들의 조합(I, j)에 대해, 1) 전체가공시간을 계산하고($T(i, j)$), 2) 최소가공시간을 갖는 I^* , j^* 를 찾는다. 이때, I^* , j^* 는 구하고자 하는 최적공구 및 자세가 된다.

4.2 다수 공구, 한 자세 산출

고정된 공구자세에서의 처리할 수 있는 미절삭영역은 공구 크기 간섭으로 인하여 접근할 수 없는 영역이다. 이런 미절삭영역은 보다 작은 공구를 사용하여 제거될 수 있다. 미절삭영역을 제거하기 위해 공구 i 를 사용한 후 공구 j 를 사용할 때 걸리는 시간은 식 (6)에 의해 표현되

어진다.

$$\text{Min} \left\{ \sum_{i=1}^n \left[\frac{\alpha_{i,j} * \text{MV}(i,j)}{\text{MRR}(i)} \right] + \text{Min}(1, \text{Max}(\sum_j \alpha_{i,j}, 0)) * dT + \beta * UV \right\} \quad (6)$$

공구 i 와 공구 j 사이의 가공시간 관계는 DAG(Directed Acyclic Graph)[9]로 표현할 수 있다(그림 10). node(i)는 공구 i의 사용을 의미하고, arc(i,j)는 공구 i 사용 후 남은 영역을 공구 j를 사용하여 가공할 때 걸리는 공구교환시간을 포함한 가공시간을 나타낸다. Arc의 방향은 공구 사용순서를 의미한다. 따라서 arc의 방향은 직경이 큰 공구에서 작은 공구로 향하게 된다.

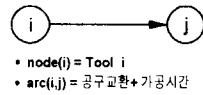


그림 6. 공구 i 와 공구 j 관계

고정된 공구자세에서 공구수가 n 개인 경우 공구사이의 가공시간 관계는 그림 6과 같이 DAG로 표현된다. 여기서 node(S)는 공작물의 초기상태를 의미하고, node(F)는 정착(또는 중식)가공 전의 상태를 의미한다. 따라서 고정된 공구자세에서의 최소 가공시간을 가지는 공구셋은 node(S)에서 node(F)를 연결하는 최단경로상에 존재하는 node들이 된다.

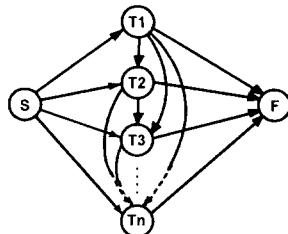


그림 7. 공구수 n 경우 형성된 DAG

따라서 다수 공구와 한 공구자세 선정방법은 가) 우선 주어진 모든 자세에 대해 위에서 제시한 DAG를 이용하여 최소 가공시간을 가지는 최적 공구셋을 구하고, 나) 다음 이들 중 최소 가공시간을 가지는 공구자세 및 공구셋을 최적 자세 및 공구셋으로 한다.

4.3 한 공구와 다수 자세 선정

한 공구와 다수 자세를 선정과정은 (다수 공구, 한자세) 문제와 유사하게 가) 우선 주어진 모든 공구들에 대해 최소 가공시간을 가지는 최적 자세집합을 구하고, 나) 다음 이들 중 최소 가공시간을 가지는 공구를 구하여, 이를 최적 공구 및 자세들로 선정한다.

한 공구와 다수의 자세가 주어진 경우, 최소 가공시간은 식 (4)로부터 다음과 같이 표현될 수 있다.

$$\text{Min} \left\{ \left[\frac{\bigcup_{i=1}^n \alpha_{i,j} * \text{MV}(i,j)}{\text{MRR}(i)} \right] + \sum_j \alpha_{ij} * dO + \beta * UV \right\} \quad (7)$$

식 (7)의 해법으로 본 연구에서는 branch-and-bound 기법[10]을 이용하였다(그림 8). 구체적인 과정은 다음과 같다.

- 1) 자세수 n 을 1로 하고, 전체자세집합
 $O = \{o_1, \dots, o_N\}$ 를 초기화 한다.
- 2) n 개의 서로다른 원소로 구성되는, P 의 모든 부분집합 P_{ni} 를 구한다.
- 3) 2)에서 구해진 모든 O_i^n 에 대해 절삭영역을 산출하고 최대절삭영역을 갖는 $O^{n\max}$ 를 찾는다.
- 4) 찾아진 $O^{n\max}$ 에 대해 식 (8)를 이용하여 총가공시간 T_n 을 구한다.
- 5) $T_n > T(n-1)$ 이거나 $n = N$ 인 경우 step 6로 간다.
아닌 경우 n 을 1 증가시켜 step 2로 간다.
- 6) 찾아진 T_1, \dots, T_n 중 최소값을 가지는 n 을 찾고, $O^{n\max}$ 를 최적자세집합으로 한다.

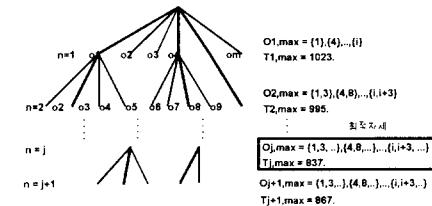


그림 8. 최적자세집합 선정 과정 (branch)

5. 공구경로 산출 및 이송속도 보정

공구경로는 계층방식을 사용하며, 각 계층(절삭평면)에 대해 지그재그 경로 형태를 취한다. 경로 생성은 4장에서 구해진 최적 공구 및 자세를 이용하여, 각 공구 및 자세에 따른 가공영역은 3장에서 구해진 EBMT를 이용하여 인식한다. 이송속도는 경로상의 두 가공점을 단위로 하여 보정되며, 표준 소재제거율을 기준값으로 한다.

5.1 공구경로 산출

공구 및 자세공구경로산출시 가공영역의 인식이 선행되어져야 한다. 공구 i, 자세 j 가 주어졌을 때 가능가능한 영역의 인식은 EBMT를 통하여 용이하게 산출된다(식 (3)). 인식된 가공영역에 대한 공구경로산출 과정을 요약하면 그림 9와 같다. : 1) MV(i,j)에 대해 자세방향으로 최고점과 최저점을 구하고, 2) 자세방향을 법선으로 하는 절삭평면을 구하여, 최고점부터 최저점까지 절삭깊이만큼 아래로 이동하면서, 4) 각각의 절삭평면에 대해 지그재그 형태의 경로를 생성한다.

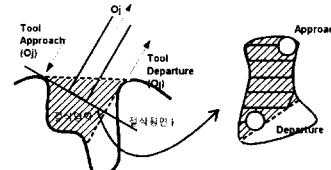


그림 9. 공구경로 산출

5.2 이송속도 보정

이송속도보정은 두 가공점사이에서 발생하는 최대절삭력을 기준으로 보정한다. 절삭력은 두 가공점사이에서 공구방향으로의 수직절단면의 면적값을 이용하여 근사화한다. 이는 평균절삭력과 소재제거율이 비례하고, 소재제거율은 절삭폭과 절삭거리에 의해 산출되므로, 순간절삭력은 순간소재제거율, 즉 절삭폭에 의해 표현될 수 있기 때문이다. 그러나 두 가공점사이의 모든 순간에 대한 절삭력산출은 불가능하므로, 두 가공점사이를 이산화하고 이산된 각 지점에서의 절삭력을 구한다. 산출된 절삭력의 범위가 큰 경우 두 가공점을 분할하여 새로운 가공점을 생성하여 미세 보정을 한다. 그림 10은 이산된 각 지점에서의 절삭력 산출과정을 보여준다.

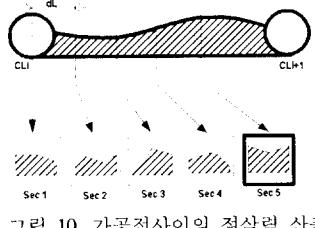


그림 10. 가공점사이의 절삭력 산출

- 1) CL_{i+1}과 CL_i를 연결하는 단위벡터(\vec{t}_d)를 구한다.
 - 2) CL_{i+1}과 CL_i 사이에 \vec{t}_d 를 법선벡터로 하는 평면을 일정한 간격으로 위치시킨다.
 - 3) 위치된 평면과 접하는 가공단위영역들을 평면에 투영시킨 후, 투영된 면적을 구한다.
 - 4) 각 평면상에서 구해진 면적중 최대값 maxS과 최소값 minS를 구한다.
- 식 (9)는 최대 면적을 이용하여 이송속도를 보정하는 것을 나타낸다.

$$t_{\text{new}} = \frac{\max S * d}{MRR_{\text{ref}}} , f_{\text{new}} = \frac{d}{t_{\text{new}}} \quad (9)$$

where

d : CL_{i+1}과 CL_i 사이 거리
 t_{new} : 보정 후 이송시간

6. 결 과

본 연구에서 제시된 알고리즘의 효율성 검증을 위해 임펠러형상을 사용하였으며, 검증예로서 한자세, 다수공구에 대한 해를 제시하였다. 사용공구수는 6개로 하였으며, 초기자세범위는 $\phi : [0,90]$, $\theta : [0,360]$ 으로 지정하였다. 표 1.은 각 공구에 대한 반지름 및 소재제거율을 보여준다.

표 1. 입력공구 데이터

공구 집합	1	2	3	4	5	6
반지름(mm)	8.0	7.0	6.0	5.0	4.0	3.5
MRR(mm ³ /min)	256	196	144	100	64	49

초기형상 및 공구가 입력되면, 초기형상으로부터 간접

가능영역을 추출한다. 그림 11은 복셀로 모델링된 임펠러 두 날개, convex 형상, 그리고 추출된 간접가능영역을 보여준다.

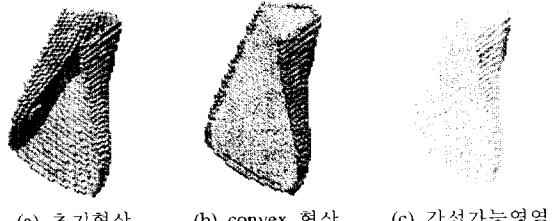


그림 11. 간접가능영역 추출

추출된 간접가능영역에 대해 가공가능성 여부를 나타내는 EBMT를 작성한다. (다수공구,한자세)문제에 대해서 EBMT로부터 각 자세에 대한 DAG를 작성하고, 각각의 DAG에 대해 최단 경로를 찾고, 전체가공시간을 산출한다. 그림 12은 생성된 DAG의 한 예를 보여준다. 이 때 공구교환시간 dT는 30초로 하였다.

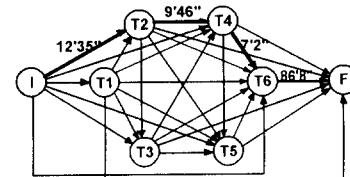


그림 12. 자세 (90, 0) 일때의 DAG 와 최단경로

각각의 자세에 대해 찾아진 최단경로 및 전체가공시간 중 최소시간을 갖는 공구자세는 (90,0)이며 이때 공구셋은 (2,4,6)이 산출(그림 12) 되었으며, 그 결과를 요약하면 표 2 와 같다.

표 2. 다수공구, 한자세 시뮬레이션 결과

	가공시간 (황삭, 중삭)	사용 공구	사용 자세 (ϕ, θ)	절삭 부피 (mm ³)	미절삭 부피 (mm ³)
다수공구	29분25초,	2, 4, 6	(90, 0)	3598.8	782
한자세	87분 08초	(3개)			

다음 그림 13은 위의 결과에 대한 시뮬레이션된 결과를 보여준다. 공구크기에 따라 순차적 가공이 진행되면서 절삭할 수 있는 부분과 최종적으로 남은 부분을 보여준다.



남은 양(2028.8/4380.8)

(a) 공구 2

남은 양(1102.1/4380.8)

(b) 공구 4

그림 13. 다수공구사용시 사용예(continued)



남은 양(782/4380.8)

(c) 공구 6

(d) 미절삭영역

그림 13. 다수공구 사용시 절삭영역

표 3 은 (한공구,한자세) 선정 및 (한공구,다수자세) 선정 결과를 보여준다. 여기서, 공구자세변경 시간을 30초로 하였다.

표 3. 한공구/한자세, 한공구/다수자세 시뮬레이션 결과

	가공시간 (황삭/ 중삭)	사용 공구	사용 자세 (ϕ, θ)	절삭 부피 (mm ³)	미절삭 부피 (mm ³)
한공구	35분02초/				
한자세	110분9초	4	(90,0)	3453	927.8
한공구	37분59초/				
다수자세	89분06초	4	(90,0) (30,0)	3643	737.8

제시된 알고리즘의 효율성을 비교하기 위해 (한공구,한자세) 선정시와 (다수공구,한자세) 선정 및 (한공구, 다수자세) 선정시의 결과를 비교하였다. 한공구,한자세 사용시에 비해 다수공구,한자세사용시 황삭가공시간은 한공구, 한자세사용에 비해 15.8 % 만큼 감소되었으며, 또한 미절삭영역은 한공구,한자세사용시에 비해 145.8 (mm³)만큼 줄어들어 종삭가공시간도 단축시킬 수 있었다. 다수자세, 한공구사용시는 한자세,한공구사용시에 비해 황삭가공시간은 증가되었으나, 황삭 후 미절삭영역의 감소로 종삭시간이 20분 정도 단축되었으며, 전체적으로 18분 가량의 가공시간이 단축되었다.

7. 결 론

본 연구에서는 다수 공구 다수 자세를 가지는 다축가공을 대상으로 가공시간을 최소화할 수 있도록 공구셋 결정, 공구자세 선정 및 공구경로 산출에 관한 황삭 알고리즘을 제시하였다. 공구셋 및 공구자세 선정 문제는 (한공구, 한자세), (한공구, 다수자세), (다수공구, 한자세), (다수공구, 다수자세)의 문제로 세분화하여 접근하였으며, (한공구, 한자세), (한공구, 다수자세), (다수공구, 다수자세)에 대한 해법을 개발하였다.

제시된 알고리즘은 부가축 환경 뿐만 아니라 다축 환경(3, 4, 5 축을 모두 포함하는)에도 바로 적용될 수 있다. 따라서, 3, 4, 5 축에 공통적으로 적용할 수 있는 황삭 알고리즘이 전무한 상태에서, 본 연구는 이 분야의 이론적인 기반을 제시하리라 기대된다. 본 연구는 추후 (다수공구,

다수자세)에 대한 해법개발과, 공구교환시간 및 공구자세 변화시간의 변화에 따른 공구셋 및 공구자세의 변화에 대한 상세한 분석 등의 방향으로 진행될 것이며, 실제 절삭 실험이 추진중에 있다.

Reference

- [1] 이정재, 서석환, "3축 CNC 를 이용한 5 축 자유곡면 가공," 대한산업공학회지, 21 권, 2 호, 1995, pp. 217-237
- [2] Yoshiro Mizugaki, Minghui Hao, Masafumi Sakamoto, Hiroshi Makino, "Optimal Tool Selection Based on Genetic Algorithm in a Geometric Cutting Simulation," Annals of the CIRP, 43(1), 1994, pp. 433-436
- [3] M.BALA and T.C. CHANG, "Automatic cutter selection and optimal cutter path generation for prismatic parts," International Journal of Production Research, 29(11), 1991, pp. 2163-2176
- [4] Y.S.LEE, B.K.CHOI and T.C.CHANG, "Cut distribution and cutter selection for sculptured surface cavity machining," International Journal of Production Research, 30(6), 1992, pp. 1447-1470
- [5] Kunwoo Lee, Tae Ju Kim and Sung Eui Hong, "Generation of toolpath with selection of proper tools for rough cutting process," Computer-Aided Design, 26(11), 1994, pp. 822-831
- [6] Hiroshi Suzuki, Yoshimasa Kuwano, Katsuyuki Goto, "Development of the CAM System fro 5-Axis Controlled Machine Tool," 日本精密工學會誌, 60(6), 1994, pp. 832-836
- [7] Franco P.Preparata, Michael Ian Shamos, "Computational Geometry An Introduction," Springer-Verlag, 1985, pp. 135-140
- [8] Daniel Cohen and Arie Kaufman, "Scan-Conversion Algorithms for Linear and Quadratic Objects," IEEE, 1991, pp. 280-301
- [9] Katta G. Murty, "Network Programming," Prentice Hall, 1992, pp.251-255
- [10] George F. Luger, William A. Stubblefield, "Artificial Intelligence and the Design of Expert Systems," The Benjamin/Cummings Pub., 1989, pp. 165-170
- [11] Machining Data Handbook, Machinability Data Center, 1972.