

향상된 절삭력 모델 기반의 NC 코드 최적화

이한울, 고정훈(포항공대 대학원 기계공학과), 조동우*(포항공대 기계공학과)

NC Code Optimization Based on an Improved Cutting Force Model

H. U. Lee, J. H. Ko, D. W. Cho* (Mech. Eng. Dept, POSTECH)

ABSTRACT

Off-line feed rate scheduling is an advanced methodology to automatically determine optimum feed rates for the optimization of NC code. However, the present feed rate scheduling systems have limitations to generate the optimized NC codes because they use the material removal rate or non-generalized cutting force model. In this paper, a feed rate scheduling system based on an improved cutting force model that can predict cutting forces exactly in general machining was presented. Original blocks of NC code were divided to small ones with the modified feed rates to adjust the peak value of cutting forces to a constant value. The characteristic of acceleration and deceleration for a given machine tool was considered when off-line feed rate scheduling was performed. Software for the NC code optimization was developed and applied to pocket machining simulation.

Key Words : NC Code Optimization (NC 코드 최적화), Feedrate Optimization (이송속도 최적화), Cutting Force (절삭력), Accel./decel. Characteristic (가감속 특성), Z-map (지맵)

1. 서론

최근 CAM 소프트웨어의 개발 동향은 효과적인 이송속도 최적화 방법에 초점이 맞추어져 있다. 기존의 CAM 소프트웨어는 주로 공구경로 생성에 주안점을 두고 있었기 때문에 생성된 NC 코드를 실제 생산 현장에 적용하기 위해서는 추가적인 수정 작업이 필요하기 때문이다. 현재 대부분의 CAM 소프트웨어에서 사용하는 이송속도 최적화 방법은 피삭재와 공구간의 기하학적 관계를 고려한 소재 제거량을 기반으로 하고 있다^{[1][2][3]}. 그러나 이 방법으로는 절삭이라는 난해한 역학적 문제를 효과적으로 해결할 수 없기 때문에 최적의 이송속도를 구하기 힘들다. 따라서 효과적이고 신뢰할 수 있는 이송속도를 얻기 위해서는 절삭력을 기반으로 이송속도를 최적화 시켜주는 시스템이 개발되어야 한다.

일련의 논문들에서 절삭력을 기반으로 한 이송속도 최적화 기법이 제안되었다. Tamg 등^[4]은 칩의 양과 절삭 계수를 이용한 절삭력을 계산하여 기준 이송속도와의 비교를 통해 이송속도를 최적화 하였으며, Lim 등^[5]은 절삭력과 가공오차를 이용하여 이

송속도와 가공방향을 최적화 시키고, 황석 가공과 정삭 가공에 다른 최적화 기준을 적용시켜야 한다는 것을 보여주었다. 최근에 Feng 등^[6]은 볼 엔드밀링의 절삭력 모델을 이용하여 3D 곡면 가공에 대한 최적의 공구 경로와 이송 속도를 결정하는 방법을 발표하였다. 그러나 기존의 연구들에서 사용한 절삭력 모델들은 절삭계수를 가공 조건 또는 공구 회전각의 함수로 가정하여 결정하므로 일반화된 이송속도 최적화를 구현하기 힘들었다.

본 연구에서는 직선 및 원호 가공에서 절삭 조건이나 공구의 회전각에 상관없이 절삭력과 가공 표면 오차의 정확한 예측이 가능한 평 엔드밀링의 절삭력 모델^[7]을 이용하여 임의의 포켓 가공을 위한 절삭력 기반의 이송속도 최적화 기법을 개발하였다. 최적화된 이송속도는 NC 코드를 통하여 가공에 반영된다. 그러므로 NC 코드의 블록을 이송속도에 따라 분할하여 최적의 NC 코드를 만들었고 NC 코드 분할 시에는 공작기계의 가감속을 고려하였다. 개발된 NC 코드 최적화 기법을 구현한 소프트웨어를 개발하였고, 이를 포켓 가공에 적용하여 타당성을 검증하였다.

2. 이송 속도 최적화 모델

2.1 절삭력과 이송속도와의 관계

평 엔드밀의 절삭력은 식 1 과 같이 나타낼 수 있다⁷⁾.

$$\begin{aligned} F_x(j) &= \sum_k \sum_i [C_1 \cos(\phi - \alpha_r) + K_f C_3 \cos \phi \\ &\quad - K_f C_4 \sin(\phi - \alpha_r)] K_n B_1 t_c(\phi) \\ F_y(j) &= \sum_k \sum_i [C_1 \sin(\phi - \alpha_r) + K_f C_3 \sin \phi \\ &\quad + K_f C_4 \cos(\phi - \alpha_r)] K_n B_1 t_c(\phi) \\ F_z(j) &= \sum_k \sum_i [-C_2 + K_f C_5] K_n B_1 t_c(\phi) \end{aligned} \quad (1)$$

여기서 C_1, C_2 는 일정한 값이며, C_3, C_4, C_5 는 칩 배출각, θ_c 에 의해 결정되는 값으로 다음과 같다.

$$\begin{aligned} C_1 &= \cos \theta_h / \sin \theta_{ik} \\ C_2 &= (\sin \theta_h / \sin \theta_{ik}) \cos \alpha_r \\ C_3 &= \sin \theta_h (\sin \theta_c - \cos \theta_c \cot \theta_{ik}) \\ C_4 &= \cos \theta_c / \sin \theta_{ik} \\ C_5 &= \cos \theta_h (\sin \theta_c - \cos \theta_c \cot \theta_{ik}) \end{aligned}$$

여기서 $\cos \theta_{ik} = \sin \alpha_r \cdot \sin \theta_h$ 이다.

절삭력 모델을 CNC 에 적용하기 위해서는 계산 속도를 빠르게 해야 한다. 따라서 식 2 와 같은 공구의 기하 형상과 가공 조건을 이용한 미변형 칩 두께 모델을 사용하여 가공 중의 절삭력을 간단하게 계산하도록 한다.

$$t_c(\phi) = f_i \sin \phi \quad (2)$$

식 2 를 식 1 에 대입하여 정리하면 평 엔드밀의 절삭력과 이송속도의 관계를 식 3 과 같이 나타낼

$$\begin{aligned} F_x(j) &= f_i \left\{ \sum_k \sum_i [C_1 \cos(\phi - \alpha_r) + K_f C_3 \cos \phi \right. \\ &\quad \left. - K_f C_4 \sin(\phi - \alpha_r)] K_n B_1 \sin \phi \right\} \\ &= f_i \Psi_x(j) \\ F_y(j) &= f_i \left\{ \sum_k \sum_i [C_1 \sin(\phi - \alpha_r) + K_f C_3 \sin \phi \right. \\ &\quad \left. + K_f C_4 \cos(\phi - \alpha_r)] K_n B_1 \sin \phi \right\} \\ &= f_i \Psi_y(j) \\ F_z(j) &= f_i \left\{ \sum_k \sum_i [-C_2 + K_f C_5] K_n B_1 \sin \phi \right\} \\ &= f_i \Psi_z(j) \end{aligned} \quad (3)$$

수 있다.

일반적으로 엔드밀에 의한 가공은 황삭, 중삭 그리고 정삭의 3 단계로 나누어 생각해 볼 수 있으며, 이 가운데 황·중삭 가공의 경우 가공 시간이, 정삭 가공의 경우에는 가공면의 표면 오차가 생산성에 밀접한 관련을 가진다. 따라서 이송속도를 최적화 할 때 황·중삭 가공에서는 가공 시간을 줄이는 방향으로, 정삭 가공에서는 표면 오차를 최소화하는 방향으로 시행해야 한다. 평 엔드밀은 정삭 가공보다는 황삭 가공에 많이 사용되는 공구이므로 여기서는 가공 시간을 줄이는 것으로 이송속도를 최적화 하였다.

황삭 가공에서 가공시간을 줄이기 위해서는 이송속도를 가능한 빠르게 하여야 한다. 그러나 이송속도가 빨라지면 공구에 가해지는 절삭력의 크기가 커지게 되고, 이에 따라 채터나 공구의 마모, 파손이 발생하게 된다. 따라서 평 엔드밀을 이용한 가공에서의 이송속도를 최적화하기 위해서는 가공 시 공구에 가해지는 절삭력의 크기가 항상 설정된 한계치가 되도록 이송속도를 조절하여야 한다. 이 때 채터나 공구의 마모, 파손은 공구에 작용하는 수직력과 마찰력 모두에 영향을 받으므로 절삭력의 한계치는 F_x 의 F_y 의 합력 값을 사용하도록 한다. 임의의 절삭력 한계치에 따른 이송속도는 식 3 을 이용하여 식 4 와 같이 구할 수 있다.

$$\begin{aligned} (\text{feedrate})_{\text{desired}} \\ &= \frac{F_{\text{lim}}}{\sqrt{\Psi_x(j)^2 + \Psi_y(j)^2}} \cdot N \cdot \text{RPM} \end{aligned} \quad (4)$$

여기서 F_{lim} 은 절삭력 한계치, N 은 공구의 날 수, RPM 은 주축 회전속도이다.

2.2 NC 코드 최적화 방법

NC 코드 최적화는 입력된 NC 코드의 시뮬레이션을 통해 계산되는 가공 중의 절삭력이 항상 일정한 값을 갖도록 이송속도를 조절해 주는 것이다. 황삭 가공의 경우에는 채터나 공구의 마모, 파손이 일어나지 않는 최대 절삭력을 기준으로 사용하며, 정삭 가공에서는 표면 오차가 요구되는 값보다 작아지는 기준 절삭력을 사용하게 된다. 여기에서는 황삭 가공을 기준으로 가공 중 공구가 1 회전할 때 발생하는 F_x 와 F_y 합력의 최대값이 일정한 값을 가지도록 이송속도를 최적화하고, 이 값을 기준으로 NC 코드의 각 블록을 분할하여 NC 코드를 최적화하였다.

가공 시뮬레이션을 통해 계산된 절삭력을 바탕으로 NC 코드를 분할할 때는 다음의 두 가지를 사

용할 수 있다. 첫 번째는 사용자가 NC 코드 최적화를 실행할 구간을 임의로 지정하는 방법이다. 이 방법은 공구 경로 가운데 절삭력의 변화가 클 것으로 예측되는 곳을 사용자가 선택하고, 선택된 구간만을 시뮬레이션 하여 NC 코드를 최적화하는 것이다. 전체 공구 경로가 아닌 일부분만의 이송속도를 최적화하므로 실행 시간을 상당히 단축시킬 수 있는 장점이 있으나, 구간 선택이 적절하지 않을 경우 NC 코드 최적화의 효과를 얻기 힘들기 때문에 숙련된 사용자에게 적합한 방법이다. 두 번째는 미리 절삭력 변화량의 제한 값을 설정하고, 이 값 이상으로 절삭력이 변할 경우 블록을 분할하여 각 블록의 최대 절삭력이 항상 일정한 값을 가지도록 하는 방법이다. 이 방법은 가공 중 절삭력이 항상 일정한 값을 가지게 되고 초보자도 쉽게 NC 코드 최적화를 할 수 있지만, 프로그램 실행 시간이 첫 번째 방법보다 상대적으로 오래 걸린다는 단점이 있다. 여기에서는 사용자의 편의를 위해 두 번째 방법인 변화량에 따라 NC 코드를 분할하는 방법으로 NC 코드 최적화를 구현하였다.

2.3 공작기계의 가감속 특성 고려

2.3.1 공작기계의 가감속 특성

이동하는 물체를 정지시키려고 할 경우 하중에 의한 관성 때문에 정확한 위치에 멈추게 하는 것이 힘들다. 마찬가지로 정지 상태의 물체를 순간적으로 일정 속도로 이송시키는 경우도 그 속도에 도달하는데 필요한 가속 구간이 존재하게 된다. 이와 같은 문제는 공작기계를 이송하는 경우에도 적용되며, 이를 해결하기 위해 공작기계의 제어기는 NC 코드 블록의 시작점과 끝점에서 가감속을 제어한다. 이를 자동 가감속 기능이라 하며, 기계의 종류와 크기에 따라 제어기는 정해진 가감속 시간에 따라 NC 코드의 각 블록에서 가감속을 실행한다.

제어기에 의한 자동 가감속은 Fig. 1의 A, C

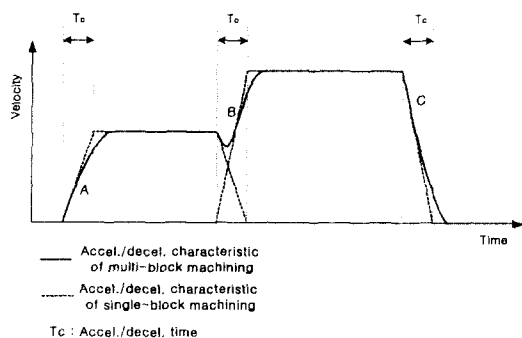


Fig. 1 Accel./decel. characteristic of machine tool.

구간과 같이 지수함수 형태로 이루어진다. 따라서 각 블록이 시작된 다음 지정된 이송속도에 도달하는데 걸리는 시간은 정해진 가감속 시간, T_c 보다 조금 길어지게 된다. 그리고 블록과 블록 사이에서는 원활한 속도 변화를 위해 Fig. 1의 B와 같이 두 블록의 감속 구간과 가속 구간을 중첩시켜 이송속도를 구하게 된다. 따라서 NC 코드의 모든 블록에서는 가공 시 항상 T_c 정도의 가감속 시간이 소요된다. 최근 개발되고 있는 공작기계에서는 Look Ahead 가감속 제어를 이용하여 이러한 문제를 해결하고 있지만, 현재 사용 중인 대부분의 공작기계에서는 가공 중에 이러한 가감속 특성을 반드시 고려해야 한다.

2.3.2 가감속 특성 실험 결과

NC 코드를 최적화할 때 공작기계의 가감속 특성을 어떻게 고려해야 하는지 알아보기 위해 다음과 같은 실험을 수행하였다. 실험에 사용된 공작기계는 대우중공업의 수직형 머시닝 센터 ACE-V30이며, Look Ahead 기능이 없는 공작기계이다. 대상 공작기계의 절삭 최대 이송속도는 4000 mm/min 이며 가감속 시간은 0.06 초이다. 실험은 (0, 0)에서 (10, 10)으로 X, Y 축을 동시에 대각선 방향으로 이송하여 각 축의 속도를 측정하였다. 대각선 방향으로 F300, F600, F1200, F2400의 속도로 이송시키며 실험하였고, 각각의 경우 X 축 방향으로의 이송 속도와 실험 조건은 Table 1 과 같다.

Fig. 2 와 3 에는 F300 과 F2400 에서 블록 수에 따른 이송속도가 나타나 있다. F300 의 경우에는 한 블록을 움직이는데 걸리는 시간이 가감속 시간인 0.06 초보다 모두 크기 때문에 20 블록으로 나누어진 경우에도 원하는 속도인 212 mm/min 으로 움직이는 것을 볼 수 있다. 그러나 가감속 구간 때문에 전체 이송시간이 1 블록으로 이송한 경우에 비해 약간 길어진 것을 알 수 있다. F2400 의 경우에는 20 블록으로 나누어진 경우 한 블록을 움직이는데 걸리는 시간이 0.0125 초로써 가감속에 걸리는 시간보다 짧기 때문에 1697 mm/min 까지 가속하지 못하고, 1260 mm/min 의 속도에 머물러 있는 것을 볼 수 있다. 이는 블록의 크기가 작으면, 충분히 가속

Table 1 Conditions of accel./decel. characteristic test.

	Feedrate (mm/min)	Time interval of one block (sec.)		
		1 block	Divided 5 blocks	Divided 20 blocks
F300	212	2	0.4	0.1
F600	424	1	0.2	0.05
F1200	848	0.5	0.1	0.025
F2400	1697	0.25	0.05	0.0125

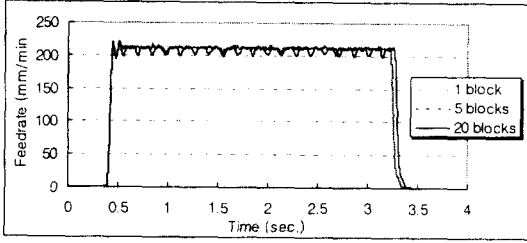


Fig. 2 Comparison of feedrate from NC codes with different time interval of one block for F300

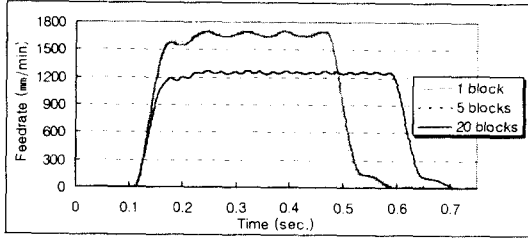


Fig. 3 Comparison of feedrate from NC codes with different time interval of one block for F2400

되지 않은 상태로 블록이 끝나고 감속하기 때문이다. 그리고, 같은 이유로 5 블록으로 나누어진 경우에도 최대 이송속도는 원하는 속도에 도달하고 있지만, 이송속도가 안정화되는 구간은 없다는 것을 볼 수 있다.

2.3.3 가감속을 고려한 NC 코드 최적화

제어기가 Look Ahead 기능을 지원하지 않으면, 동작기계의 가감속 시간보다 작은 블록에서는 원하는 이송속도를 얻을 수 없다. 그러므로 NC 코드를 분할하여 이송속도를 최적화 할 경우에는 반드시 블록의 크기를 가감속 시간보다 크게 해야한다. 물론 한 블록을 일정한 이송속도로 가공하기 위해서는 적어도 블록의 크기가 가감속 시간의 2~3 배 이상이 되어야 한다. 따라서 본 연구에서는 절삭의 변화량이 일정한 제한 값을 넘어갈 경우 NC 코드의 블록을 분할하되, 분할되는 블록의 크기가 제한된 크기보다 작을 경우에는 블록이 제한된 크기보다 커지는 부분에서 분할되도록 NC 코드 최적화를 수행하였다.

3. 소프트웨어 구현 기법

3.1 Z-map 을 이용한 절삭공정 시뮬레이션

NC 코드 최적화를 일반화하여 임의의 NC 코드에 적용할 수 있도록 Z-map 을 이용하여 피삭체를 모델링하였다. Z-map 은 xy-평면상에 정의된 일정한

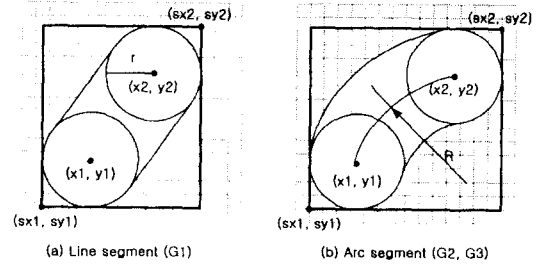


Fig. 4 Considered area for cutting process analysis in Z-map

간격의 격자들에서만 z 값을 갖기 때문에 자료구조가 간단하므로, 임의의 절삭 공정을 쉽고 빠르게 해석할 수 있다. 가공이 진행됨에 따라 임의의 위치 (x, y) 에서 공구의 높이와 공작물의 높이를 비교하여 공구와 공작물의 접촉 관계를 얻을 수 있으며, 이를 이용하여 가공 중의 절삭력을 계산할 수 있다.

Z-map 을 이용하여 절삭 공정을 시뮬레이션 할 경우, 공구가 움직일 때마다 xy-평면상의 모든 격자점에서 공구와의 접촉 관계를 해석하려면 많은 시간이 소요된다. 따라서 여기서는 일정 시간동안 공구가 움직일 때 공구의 시작점과 끝점을 이용하여 해석 영역을 설정하고, 해석 영역 내부의 격자점과 공구의 접촉 관계만을 해석하였다.

절삭 공정에서 공구가 한 점에서 다른 점으로 이동하는 방법으로는 Fig. 4 의 (a), (b)와 같이 직선 가공과 원호 가공이 있다. 이 때 해석 영역을 그림과 같이 $(sx1, sy1)$ 과 $(sx2, sy2)$ 로 나타낼 수 있는 사각형으로 정하면, $(sx1, sy1)$ 과 $(sx2, sy2)$ 는 다음과 같이 나타낼 수 있다.

$$\begin{aligned} \square \text{ 직선 가공일 경우} \\ sx1 = \min(x1, x2) - r, \quad sy1 = \min(y1, y2) - r \\ sx2 = \max(x1, x2) + r, \quad sy2 = \max(y1, y2) + r \end{aligned} \quad (5)$$

□ 원호 가공일 경우

i) 식 5 와 같이 $sx1, sy1, sx2, sy2$ 의 값을 정한다.

ii) 원호의 중심점에서 시작점과 끝점까지의 각을 각각 θ_s, θ_e 라고 하면,

$$\begin{aligned} \text{if}(\theta_s < 0 < \theta_e) \quad & sx2 = cx + R + r \\ \text{if}(\theta_s < \pi/2 < \theta_e) \quad & sy2 = cy + R + r \\ \text{if}(\theta_s < \pi < \theta_e) \quad & sx1 = cx - R - r \\ \text{if}(\theta_s < 3\pi/2 < \theta_e) \quad & sy1 = cy - R - r \end{aligned} \quad (6)$$

여기서 (cx, cy) 는 원호 가공의 중심점이다.

3.2 알고리즘

NC 코드 최적화는 Fig. 5 와 같은 과정을 거쳐 이루어진다. 먼저 NC 코드를 읽어들이고, 현재 블록을 이송속도 스케줄러에 넣는다. 스케줄러는 블록의 절삭력을 시뮬레이션 하여 절삭력이 제한 값보다 크면서 블록의 크기가 가감속 시간보다 클 경우 현재 위치를 이용하여 새로운 블록을 만든다. 새로 만들어지는 블록의 이송속도를 블록의 최대 절삭력 값으로 구하고, 이 블록을 현재 블록의 앞에 입력한다. 그리고 공구가 블록의 끝점에 도달하면 블록의 최적 이송속도를 구해 현재 블록의 이송속도를 수정한다. 이 과정을 전체 NC 코드에 대하여 반복함으로써 NC 코드 최적화를 구현하였다.

4. NC 코드 최적화 결과 및 검증

앞에서 기술한 방법으로 NC 코드 최적화 소프트웨어를 구현하였으며, 이를 포켓가공에 적용하여 보았다. Fig. 6 과 같은 임의의 형상을 포켓가공 하였으며, CNC Software 사의 CAD/CAM 소프트웨어인 Mastercam 을 이용하여 같은 공구 경로를 생성하였

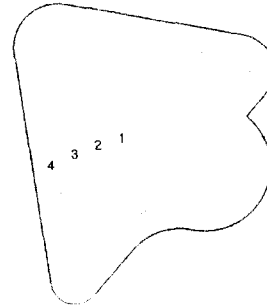


Fig. 6 Tool path for pocket machining

Table 2 Cutting force coefficient & NC code optimization parameter

Kn	1020.0
Kf	0.973
Chip flow angle	28.648°
Maximum force	300 N
Minimum time interval of one block	0.5 sec.
Force variation for block partition	2 N

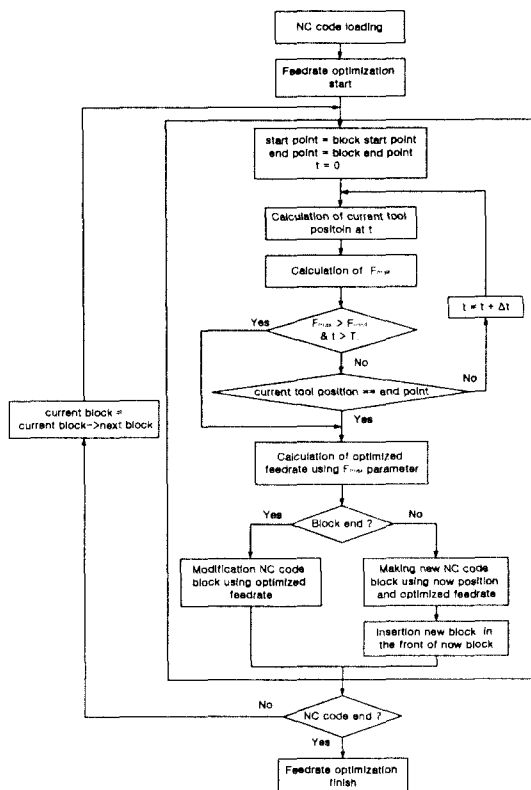


Fig. 5 Algorithm for NC code optimization

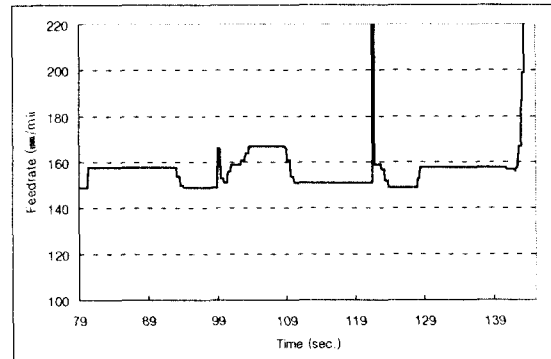


Fig. 7 Optimized feedrate for loop 4

다. 이 때 오프셋량은 5 mm 로 설정하였으며, 이송속도는 F150 이다.

이송속도 최적화 결과 Fig. 6 의 Loop 4 를 가공하기 위한 NC 코드가 Table 3 과 같이 수정되었다. NC 코드는 15 블록에서 42 블록으로 분할되었으며, 최적화된 이송속도는 Fig. 7 과 같다.

Fig. 8 과 9 는 Fig. 6 의 공구 경로 가운데 A, B 구간에서 이송속도 스케줄링 전·후의 절삭력을 비교한 것이다. 구간 A 는 공구가 Loop1 에서 Loop2 로 진입하면서 가공된 부분으로 인해 직선 가공에서는 절삭력이 줄어드는 구간이고, 구간 B 는 원호 가공으로 절삭력이 계속 변하는 구간이다. 이송속도 스케줄링을 통하여 절삭력을 일정하게 유지함으

Table 3 NC code before/after NC code optimization for loop 4

Before optimization	After optimization
N160X13.545Y61.15F150	N160X13.545Y61.15F149 G1X18.574Y28.004F158 G1X18.771Y26.702F154 G1X18.964Y25.433F150
\N162X20.783Y13.442	N162X20.783Y13.442F149 G1X21.581Y14.393F166 G1X22.470Y15.453F153 G1X23.290Y16.450F151 G1X24.099Y17.394F156 G1X26.606Y20.381F159 G1X27.628Y21.599F161 G1X28.490Y22.627F164
\N164X28.843Y23.047	N164X28.843Y23.047F167 G2X41.504Y29.115I13.022J-10.928F167 N166G2X41.866Y29.119R17.F166 G2X43.248Y29.063I-0.000J-17.000F161 G2X44.580Y28.901I-1.385J-16.944F154
N168X45.379Y28.752R17.	N168X45.379Y28.752R17.F151
\N170G3X47.136Y28.569R8.5	\N170G3X47.136Y28.569R8.5F151 G3X55.636Y37.069I0.000J8.500F151 N172X55.636Y37.069R8.5F4000
\N172X55.636Y37.069R8.5	N172X55.636Y37.069R8.5F4000
\N174X47.724Y45.548R8.5	\N174X47.724Y45.548R8.5F151 G1X48.533Y46.512F344 G1X50.376Y48.708F159 G1X51.228Y49.723F157 G1X52.069Y50.725F152 G1X52.883Y51.695F149
\N176G1X55.857Y55.241	\N176G1X55.857Y55.241F149

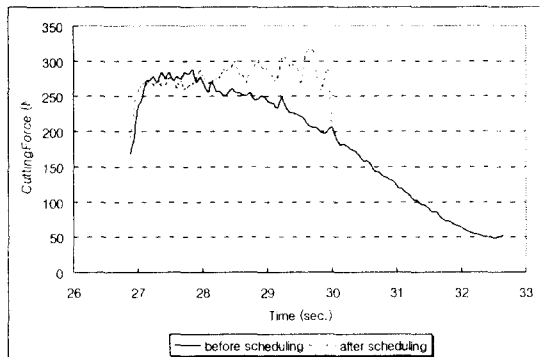


Fig. 8 Comparison of maximum cutting force before/after NC code optimization for path A

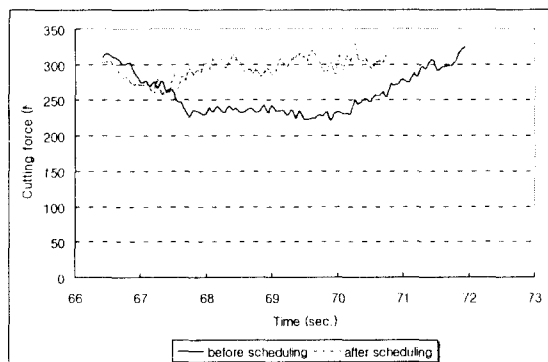


Fig. 9 Comparison of maximum cutting force before/after NC code optimization for path B

로써 가공 속도를 증가시킬 수 있었다. 전체 가공 시간은 스케줄링 전 164.5 초에서 스케줄링 후 144.3 초로 단축되었다.

4. 결론

본 연구에서는 절삭 조건이나 공구의 회전각에 관계없이 절삭력을 예측할 수 있는 평 엔드밀링의 절삭력 모델을 이용하여 포켓가공을 위한 절삭력 기반의 NC 코드 최적화 기법을 개발하였다.

NC 코드의 최적화는 임의의 절삭력에 따른 최적의 이송속도를 구하고, 이를 NC 코드로 구현하는 것이다. 이송속도를 최적화 하기 위해서는 NC 코드를 적절히 분할해야 하며, 이 때 공작 기계의 가감속 특성을 고려하여야 한다. 일반화를 위한 절삭 공정 시뮬레이션에는 Z-map 모델을 사용하였고, Z-map 모델에서 계산 속도를 향상시킬 수 있는 방법을 제안하였다. 제안된 NC 코드 최적화 기법을 소프트웨어로 구현하였으며, 임의의 포켓가공에 적용하여 NC 코드를 최적화한 결과 개발된 방법이 효과적임을 확인하였다.

참고문헌

1. <http://www.cgtech.com>
2. Donggo Jang, Kwangsoo Kim, Jungmin Jung, "Voxel-based Virtual Multi-axis Machining," Int. J. of Advanced Manufacturing Technology, Vol. 16, No. 10, pp. 709-713, 2000.
3. 이기우, 노상도, 신동목, 한형상, "절삭부하 예측을 통한 NC 코드 후처리 시스템," 한국정밀공학회지, 제 17 권, 제 5 호, pp. 116-123, 2000.
4. Y. S. Tarn, Y. Y. Shyr, B. Y. Lee, "Computer-aided generation of the cutting conditions in pocket machining," J. of Materials Processing Technology, Vol. 51, pp. 223-234, 1995.
5. EE Meng Lim, Chia-hsiang Menq, "Integrated Planning for Precision Machining of Complex Surfaces. Part 1: Cutting-path and Feedrate Optimization," Int. J. Mach. Tools & Manufacture, Vol. 37, No. 1, pp. 61-75, 1997.
6. Hsi-Yung Feng, Ning Su, "Integrated Tool Path and Feedrate Optimization for the Finishing Machining of 3D Plane Surfaces," Int. J. Mach. Tools & Manufacture, Vol. 40, pp. 1557-1572, 2000.
7. 고정훈, 조동우, 윤원수, 양희구, "자유 곡면의 고속 절삭 시뮬레이션 시스템 개발," 제 1 회 고속 지능형 가공시스템 기술 Workshop, 서귀포 KAL 호텔, pp. 153-158, 2000.