

SSMS에서 동적 공구할당을 고려한 부품투입 알고리즘의 시물레이션 분석

이충수*, 김성식**, 노형민*

* 한국과학기술연구원 CAD/CAM 연구센터, ** 고려대학교 산업공학과

Abstract

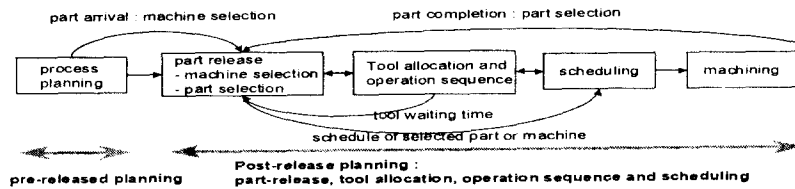
Recently in manufacturing environment, manufacturing order is characterized by unstable market demand, shorter product life cycle, a variety of product and shorter production lead time. In order to adapt this manufacturing order, flexible manufacturing system(FMS) in manufacturing technology advances into the direction that machines become further versatile functionally and that tools are controlled by fast tool delivery device. Unlike conventional FMS to mainly focus on part flow, it is important to control tool flow in single-stage multimachine systems(SSMS), consisting of versatile machines and fast tool delivery device. In this paper, in SSMS under dynamic tool allocation strategy to share tools among machines, we propose real-time part release and tool allocation algorithms which can apply real factory and which can improve system performance.

1. 서론

1980년대 말부터 다기능 공작기계의 개념이 도입되었다. 이 다기능 공작기계는 다양한 작업을 수행할 수 있고, 빠른 공구 이동장치에 의한 지원을 받기 때문에 어떤 부품을 가공하기 위하여 매우 적은 셋업을 필요로 한다. 다기능 공작기계로 구성된 시스템은 병렬기계(parallel machines)[6], SSMS(single-stage multimachine system)[5] 등의 다른 이름으로 불리워졌다. SSMS는 동일한 기능의 다기능 기계들로 구성되기 때문에 각 기계는 필요한 공구만 공급된다면 다양한 작업을 수행할 수 있다. 부품이 필요한 공구를 보유하고 있는 기계로 이동하면서 가공하던 기존 FMS와 달리, SSMS에서는 한 기계에서 필요한 공구를 공급받아 부품의 가공을 완료할 수 있다. 따라서 SSMS에서는 기존 FMS에서 중요시되던 부품 흐름보다 신속한 공구 이동장치를 이용하여 공구 흐름을 제어하는 것이 중요한 문제이다. Kashyap과 Khator[4]는 FMS에서 총 운영비의 20 ~ 30%는 공구에 관련된 비용이라고 언급했다. Jones 등[3]은 효율적인 공구할당에 의하여 공구재고를 40%까지 줄일 수 있다고 주장했다. 공구할당은 정적전략(static strategy)과 동적전략(dynamic strategy)으로 분류할 수 있다. 정적전략에서 생산계획 기간에 필요한 공구는 가공 전에 해당 기계에 할당되며, 할당된 공구는 그 기간 동안에 이동되지 않는다. 이러한 정적전략들은 공구의 제어가 용이한 반면, 높은 공구 재고가 소요된다. 동적전략에서는 생산계획 기간에 필요한 공구는 초기에 미리 할당되지

않고, 가공중에 필요할 때마다 할당된다. 또한 빠른 공구 이동장치를 이용하여 시스템 내의 다른 기계가 보유하고 있는 공구를 공유함으로써 가상적으로 공구 매거진의 크기를 증가시킴으로써 정적전략보다 공구 재고를 줄일 수 있다.

SSMS에 동적전략을 사용하여 공구할당만을 다른 기존 연구로써, Cantamessa 등[1]은 두 기계를 대상으로 작업장의 공구 수를 최소화하는 알고리즘을 제안하였다. Grieco 등[2]은 빠른 공구 이동장치를 이용하여 공구할당을 하여 시스템의 보유 공구 수를 줄이고도 줄이지 않았을 때와 유사한 성능을 유지할 수 있다는 것을 입증하였다. 상기의 연구들은 부품투입(part release) 규칙에 관한 연구가 아닌, 공구할당만을 수행한 연구이기 때문에 공구 유희시간, 공구 교환횟수 등의 국지적 성능척도(local performance)만을 개선하였다. 동적 공구할당 전략하의 SSMS에서 공구할당과 부품투입을 동시에 고려한다면 납기 지연 부품수, 생산량 등의 시스템의 전역적 성능척도(global performance)를 개선할 수 있을 것이다. SSMS에서 부품투입시 공구 제약조건을 고려한 연구로써 Koo[5]는 다른 기계에서 가공중인 부품에 필요로 하는 공구와 투입하려는 후보부품의 가공에 필요한 공구의 경쟁 정도를 표현한 TA(tool availability)를 계산하였다. Koo는 부품투입을 위하여 부품선택시 URG(highest urgency)와 TA를 조합하여 부품을 선택하였고, 기계선택시 적은 공구 이동횟수(tool movement frequency)가 계산된 기계를 선택하였다. Koo는 성능척도로 생산율(throughput



<그림 1> 시스템의 전 처리계획과 후 처리계획

rate)과 납기지연 부품(tardy part) 등을 이용하였다. Koo가 부품투입시 공구 제약조건을 고려하기 위하여 제안한 TA와 공구 이동횟수로 다른 기계에서 가공중인 공구와 필요한 공구가 경쟁을 피할 수 있는/없는/지 또는 공구 대기시간(tool waiting time)이 얼마나 발생하는지를 정확히 파악할 수 없다. 따라서 동적 공구할당 전략하의 시스템에서는 공구 경쟁에 의한 공구 대기시간을 정확히 정확히 계산하여 반영하는 새로운 부품 투입 규칙에 대한 연구가 필요하다.

본 연구에서는 동적 공구할당 전략하의 SSMS에서 실시간으로 다른 기계의 공구 일정계획을 참조함으로써 공구의 상태를 고려하여 공구 대기시간을 줄이는 방향으로 작업순서와 공구할당 알고리즘을 동시에 수행하고, 이 알고리즘에서 계산된 공구 대기시간을 이용한 실시간 부품투입 알고리즘을 제안한다. 본 연구의 목적은 공구 제약조건을 정확히 고려한 알고리즘이 시스템에서 같은 공구수를 유지하는 상태에서 다른 알고리즘보다 시스템 성능을 높일 수 있다는 것을 입증하는 것이다. 본 연구의 성능척도는 납기지연 부품수를 최소화하는 것과 생산량(throughput)을 최대화하는 것이다.

2. 부품투입 및 공구할당 알고리즘

시스템의 기능은 <그림 1>과 같이 전 처리계획(pre-release planning)과 후 처리계획(post-release planning)으로 구성된다. 전 처리계획은 부품이 시스템에 도착하기 전에 셋업, 작업, 공구, 가공시간 등을 결정하는 공정설계를 수행하고, 후 처리계획은 부품이 시스템에 도착한 후 작업순서, 공구할당, 부품투입, 일정계획 등을 수행한다. 본 논문에서는 후 처리계획만을 다룬다.

본 연구에서는 다른 기계의 일정계획을 참조하여 공구 대기시간을 최소화하는 방향으로 작업순서와 공구할당을 수행하고, 계산된 공구 대기시간을 부품투입에 이용하였다. SSMS에서는 한 기계에서 부품의 가공을 완료하기 때문에, 부품투입 알고리즘을 수행하는 시점은 부품이 시스템에 도착했을 때 한 기계이상인 유휴상태인 경우인 기계선택과 한 부품의 가공을 완료했을 때 시스

템 버퍼에 한 부품 이상이 있는 경우인 부품선택이다. 이 두 경우에 후보 부품 또는 후보 기계에 대하여 공구 대기시간을 최소로 하는 공구할당, 작업순서를 동시에 수립한다. 그 후, 공구 대기시간을 고려한 부품투입 알고리즘에 의하여 부품 또는 기계를 선택한 후, 해당 기계의 일정계획을 수립한다. 부품투입과 공구할당 알고리즘을 설명하기 위하여 필요한 기호는 다음과 같다.

<notation>

- p_{ij} : processing time for machining operation j of part i ($p_i = \sum_j p_{ij}$)
- tw_{ij} : tool waiting time of the required tool for operation j of part i ($tw_i = \sum_j tw_{ij}$)
- ttw_{jt} : tool transporter waiting time of the required tool t for operation j of part i
- ct_t : the time at which the required tool t is completed on a machine
- t_{ijk} : the time at which the operation j of part i is completed on current machine k
- mt_{ijk} : tool moving time from another machine to current machine k
- S_i : slack time of part i
- SS : set of parts which has small slack time
- O_i : set of operations required for part i
- D_i : due-date of part i
- $tnow$: current decision time

2.1 부품투입 알고리즘

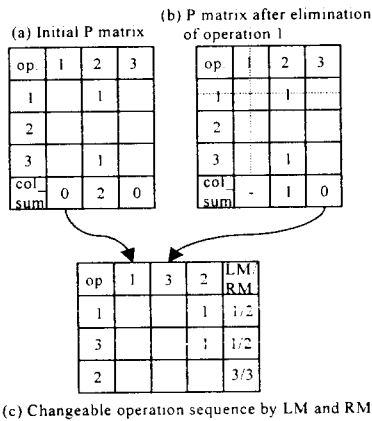
부품투입 알고리즘에서 부품선택의 경우, 해당 기계에 대하여 가장 적은 여유시간(slack time) $S_i = \{(D_i - tnow) - (p_i + tw_i)\}$ 을 갖는 부품을 선택한다. 기계선택의 경우, 해당 부품에 대하여 가장 적은 공구 대기시간 tw_i 를 제공하는 기계를 선택한다. 이 부품투입 알고리즘에서 tw_i 는 2.2 절의 공구할당 알고리즘을 수행하여 계산된다.

2.2 공구할당 알고리즘

어떤 부품의 가공을 위하여 필요한 공구가 해당 기계에 없을 경우, 다른 기계로부터 공구를 가져온다. 이때 그 공구를 사용중이어서 즉시 가져올 수 없는 경우, 공구 대기시간이 일어난다. 이 경우,

선행관계를 만족하는 범위내에서 그 작업의 순서를 바꿈으로써 공구 대기시간을 줄일 수 있다. 이 알고리즘을 수행하는 절차는 다음과 같다.

첫번째 단계에서 선행관계를 만족하는 작업순서를 결정하기 위하여 <그림 2>의 (a)와 같은 선행관계 행렬 $P(i, j)$ 를 이용했다. 이 선행관계 행렬에서 각 열의 col_sum(j)를 계산하여 0인 작업을 <그림 2> (c)의 작업순서에 고정시킨다. <그림 2>의 (a)에서 col_sum(j)가 0인 작업은 작업 1과 작업 3 이이며, 여기서는 작업 1을 임의로 작업순서에 고정시켰다. <그림 2>의 (b)에서는 작업 1을 제거한 후, col_sum(j)를 다시 계산하여 나머지 작업들을 작업순서에 고정시켰다. 어떤 작업 j의 LM(left movement)은 작업순서에서 왼쪽으로의 이동 허용범위를 의미하고, RM(right movement)은 오른쪽으로의 이동 허용범위를 의미한다[7, 8]. <그림 2>의 (c)에는 LM과 RM에 의하여 변동 가능한 작업순서가 나타나 있다. 여기서 작업 1과 작업 3의 LM과 RM은 1과 2이다. 따라서 작업 1과 3은 작업순서에서 첫번째 또는 두번째에 위치할 수 있다. 두번째 단계에서는 <그림 2>의 (c)에서 결정된 LM과 RM 사이에서 변동 가능한 여러 작업순서에서 다른 기계의 일정계획을 참조하여 공구 상태, 공구 이동시간, 공구 이동장치 대기시간 등을 고려하여 공구 대기시간을 최소화 하는 공구 할당과 작업순서를 결정한다.



<그림 2> 선행관계 행렬에서 LM과 RM에 의하여 변경 가능한 작업순서 결정

공구 대기시간 rw_{ij} 는 공구 경쟁에 의한 대기시간($ct_i - t_{ijk}$), 공구 이동시간(mt_{ijk}), 공구 이동장치의 대기시간(tw_{ij})으로 구성된다. 공구 할당 시점에 필요한 공구가 다른 기계에서 사용중이라면 공구 대기시간은 0 이상일 것이다. 그러나 공구 할당시점 전부터 유후

해서 미리 이동했다면 공구 대기시간은 0 일 것이다. 이 공구 대기시간의 계산은 다음과 같다.

$$rw_{ij} = \sum_t \sum_k \{ \max(0, ct_i - t_{ijk} + mt_{ijk} + tw_{ij}) U_{ijk} \} \text{ for all } i, j \in O,$$

$$U_{ijk} = \begin{cases} 1 & \text{if operation } j \text{ of part } i \text{ is processed with tool } t \text{ on machine } k \\ 0 & \text{otherwise} \end{cases}$$

3. 시물레이션 실험

2절에서 제안한 부품투입과 공구 할당 알고리즘의 성능척도에 큰 영향을 주는 요인을 분석하기 위하여 2^k 요인실험(2^k factorial design)을 도입하였다. 시물레이션 언어는 AweSim과 C++을 이용하였다.

3.1 2^k 요인실험

이 시물레이션 실험의 성능척도는 납기 지연 부품수를 최소화하는 것과 생산량을 최대화하는 것이다. SSMS에서 성능척도에 큰 영향을 줄 수 있는 요인은 평균 공구 중복율(tool duplication rate), 부품 종류수, 부품의 평균 작업수, 공구 이동장치수, 동일 것으로 생각했다. 공구 중복율은 시스템에 있는 각 공구 종류별 공구수를 기계수로 나눈 수치이다. 실험은 기계수는 4대, 공구 종류수 25개로 이루어졌다. 이 실험에서 공구 종류수, 부품 종류수, 평균 작업수는 실제 현장의 1/5 정도로 축소하였다. 2^k 요인실험을 위하여 <그림 3>과 같이 각 요인의 고수준과 저수준을 정의했다.

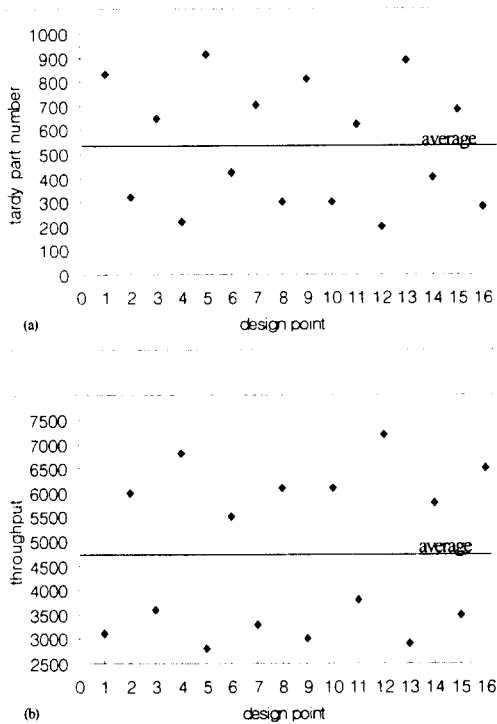
factor	factor name	- (low level)	+ (high level)
1	tool duplication rate	U(0.25, 0.4)	U(0.75, 0.9)
2	part type number	U(4, 5)	U(12, 15)
3	operation number/part	U(5, 7)	U(15, 20)
4	tool transporter number	1	2

<그림 3> 요인의 수준 정의

design point	factor				Response
	1	2	3	4	
1	-	-	-	-	R ₁
2	+	-	-	-	R ₂
3	-	+	-	-	R ₃
4	+	+	-	-	R ₄
5	-	-	+	-	R ₅
6	+	-	+	-	R ₆
7	-	+	+	-	R ₇
8	+	+	+	-	R ₈
9	-	-	-	+	R ₉
10	+	-	-	+	R ₁₀
11	-	+	-	+	R ₁₁
12	+	+	-	+	R ₁₂
13	-	-	+	+	R ₁₃
14	+	-	+	+	R ₁₄
15	-	+	+	+	R ₁₅
16	+	+	+	+	R ₁₆

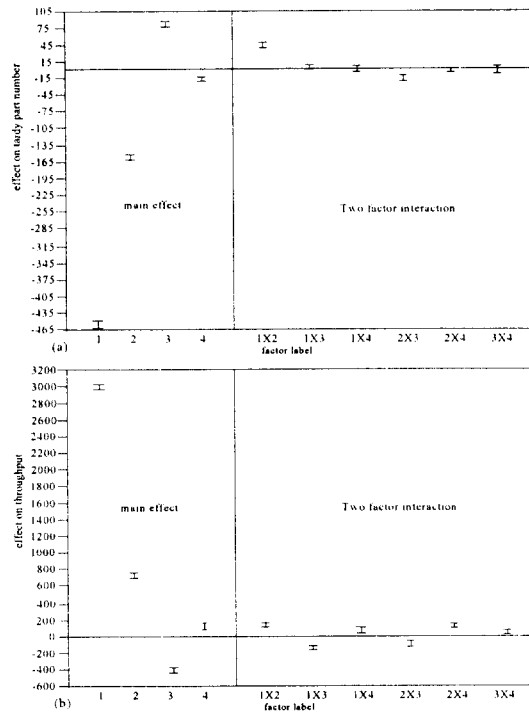
<그림 4> 2^4 요인실험에서 실험점의 조건

<그림 4>는 각 실험점(design point)에서 요인의 조건을 나타내고 있으며 R_i는 성능 척도를 나타낸다. <그림 5>의 (a)는 5000 개의 부품을 완성할 때 측정된 납기지연 부품 수이며, <그림 5>의 (b)는 400,000 분을 시물레이션하여 측정된 생산량이다. 시물레이션 초기에 각 기계는 비어있는 상태에서 시작하기 때문에 처음 10000 분의 통계량은 버렸다. <그림 5>의 (a)와 (b)에서 각 실험점의 성능척도는 10 번의 시물레이션을 수행한 평균이다. <그림 6>의 (a)와 (b)에는 납기지연 부품수와 생산량에 대한 요인의 주효과와 요인간의 2원 상호작용효과를 계산하여 정리하였다. 3 원이상의 상호작용효과를 계산하였으나 0 에 근사했기 때문에 표시하지 않았다. 실제로 <그림 6>의 (a)와 (b)에서 2 원 상호작용 효과는 주효과와 비교할 때 무시할 정도임을 알 수 있다. 이로부터 시물레이션 실험에 가장 큰 영향을 주는 요인은 평균 공구 중복율이고, 그 다음으로 부품 종류수, 평균 작업수, 공구 이동장치수임을 알 수 있다. 그리고 최적의 실험조건은 요인 1, 2, 4 가 고수준이고, 요인 3 이 저수준인 경우임을 알 수 있다.



<그림 5> 실험점들의 납기지연 부품수(a)와 생산량(b)

3.2 다른 알고리즘과 시물레이션 비교 분석



<그림 6> 납기지연 부품수(a)와 생산량(b) 대한 주효과와 2원 상호작용효과

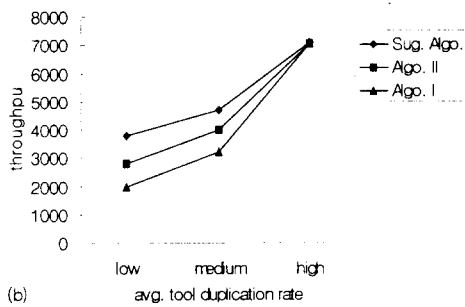
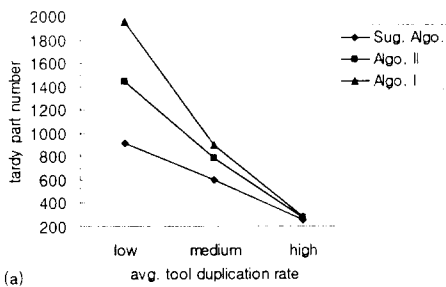
제안된 부품투입 알고리즘의 우수성을 입증하기 위하여 성능척도를 다른 알고리즘들과 비교하였다. 비교대상은 부품투입시 공구 제약조건을 전혀 고려하지 않은 알고리즘과 TA와 공구 이동횟수를 고려한 알고리즘이다. 이 알고리즘들은 다음과 같다.

알고리즘 I: 부품투입시 공구 제약조건을 고려하지 않고 EDD로 기계에 부품을 할당한 후, 그 부품의 가공을 위하여 필요한 공구를 할당하는 알고리즘이다.

알고리즘 II: 부품선택시, EDD와 TA를 조합하여 부품을 선택하고, 기계선택시 적은 공구 이동횟수가 계산된 기계를 선택하여 부품투입을 수행하는 알고리즘이다. 이 알고리즘은 Koo의 부품투입 알고리즘과 유사한 것으로, 부품선택시 Koo의 URG/TA 규칙을 EDD/TA 규칙으로 수정한 것이다. 알고리즘 I과 II에서 공구할당은 작업순서를 고려하지 않고 한 공구를 사용하여 가공을 완료한 직후에 선행관계에 의하여 가능한 공구 중 적절한 다음 공구를 선택한다.

3.1 절의 2^k 요인실험에서 시스템의 성능 척도에 큰 영향을 주는 요인이 평균 공구 중복임을 알았다. 본 절에서는 평균 공구 중복율의 값을 변화시키면서 제안한 알고리즘과 알고리즘 I, II의 성능을 비교하고자 한다.

<그림 7>의 (a)는 평균 공구 중복율의 변화에 따른 납기 지연 부품수를, <그림 7>의 (b)는 생산량을 알고리즘 I, II와 비교한 그래프이다. <그림 7>의 (a)와 (b)에서 평균 공구 중복율이 고수준일 때, 세 알고리즘의 차이는 없었다. 평균 공구 중복율이 저수준과 중수준일 때 제안된 알고리즘, 알고리즘 II, 알고리즘 I의 순서로 성능이 좋았다. 특히 평균 공구 중복율이 저수준일 때 제안된 알고리즘과 다른 알고리즘의 성능과 큰 차이가 있었다. 이 실험으로부터 부품투입시 공구 제약조건을 정확히 고려하면 할수록 시스템의 성능을 높일 수 있다는 것을 입증했다.



<그림 7> 평균 공구 중복율의 변화에 따른 납기 지연 부품수(a)와 생산량(b)

4. 결론

본 연구는 SSMS 에서 동적 공구할당 전략하의 부품투입과 공구할당 문제를 연구했다. 이 연구의 동기는 기존 FMS 에서의 부품중심의 제어가 SSMS 에서는 공구중심의 제어로 변화되어야 한다는 사고로부터 시작되었다.

본 연구에서는 부품투입시, 공구 제약조건을 정확히 고려하여 시스템의 성능을 개선하는 알고리즘을 제안하였다. 이를 위하여 알고리즘의 수행 시점을 실시간으로 이동하였으며, 공구 대기시간을 최소화 하는 작업순서와 공구할당을 수행한 후, 계산된

공구 대기시간을 부품투입 알고리즘에서 이용하였다. 부품투입과 공구할당을 동시에 수행하는 본 연구의 내용과 결과를 요약하면 다음과 같다.

- (1) 동적 공구할당 전략하에서 공구 대기시간을 최소화 하는 실시간 공구할당 알고리즘을 제안하였다.
- (2) 계산된 공구 대기시간으로부터 여유시간을 정확히 계산하여, 이를 이용한 부품투입 알고리즘을 제안하였다.
- (3) 시물레이션 실험을 통하여 부품투입시 공구 제약조건을 정확히 고려하면 할수록 시스템의 성능을 높일 수 있다는 것을 입증하였다. 특히 평균 공구 중복율이 적은 상태에서는 다른 알고리즘보다 월등히 성능을 개선하였다.

Reference

- [1] Cantamessa, M. and Lombardi, F., "Tool flow planning in a flexible manufacturing system", Computer Integrated Manufacturing Systems, Vol. 6(2), pp. 93-102, 1993.
- [2] Grieco, A., Semararo, Q., Tolio, T. and Toma, S., "Simulation of tool and part flow in FMS", International Journal of Production Research, Vol. 33(3), pp. 643-658, 1995.
- [3] Jones, T., "How the big(and little) guys do it, tooling and production", pp. T7-T10, 1994.
- [4] Kashap, A. S. and Khator, S. K., "Control rules for tool sharing in flexible manufacturing systems", Computers and Industrial Manufacturing, Vol. 25(1-4), pp. 507-510, 1993.
- [5] Koo, P-H., "Flow planning and control of single-stage multimachines systems", Ph.D Thesis, Purdue university, 1996.
- [6] Levin, J. B. and Dutta, D., "Computer aided process planning for parallel machines", Journal of Manufacturing Systems, Vol. 11(2), pp. 79-92, 1991.
- [7] Rho, H. M., Geelink, R., van 't Erve, A. H., Kals, H. J. J., "An Integrated Cutting Tool Selection and Operation Sequencing Method", Annals of the CIRP, Vol. 41(1), pp.517-520, 1992.
- [8] Wang, E., Kim, Y. S., Lee, C. S., and Rho, H. M., "Feature-Based Machining Precedence Reasoning and Sequence Planning", Proc. ASME Computers in Engineering Conference, 1998.