

퍼지 최적화기법을 이용한 유연 흐름 생산시스템의 근사 최적 스케줄링

Near Optimal Scheduling of Flexible Flow Shop using Fuzzy Optimization Technique

박 승 규, 이 창 훈, 장 석 호, 우 광 방

(Seung Kyu Park, Chang Hoon Lee, Seok Ho Jang, Kwang Bang Woo)

Abstract : This paper presents the fuzzy optimization model based scheduling methodology for the efficient production control of a FFS(Flexible Flow Shop) under the uncertain production environment. To develop the methodology, a fuzzy optimization technique is introduced in which the uncertain production capacity caused by the random events like the machine breakdowns or the absence of workers is modeled by fuzzy number. Since the problem is NP hard, the goal of this study is to obtain the near optimal but practical schedule in an efficient way. Thus, Lagrangian relaxation method is used to decompose the problem into a set of subproblems which are easier to solve than the original one. Also, to construct the feasible schedule, a heuristic algorithm was proposed. To evaluate the performance of the proposed method, computational experiments, based on the real factory data, are performed. Then, the results are compared with those of the other methods, the deterministic one and the existing one used in the factory, in the various performance indices. The comparison results demonstrate that the proposed method is more effective than the other methods.

Keywords : fuzzy optimization, near optimal scheduling, Lagrangian relaxation, heuristics, flexible flow shop

I. 서론

생산시스템의 스케줄링에서 고려해야 할 문제점들 중 하나는 기계설비의 고장, 작업자의 결근, 수요의 변동 그리고 제품불량으로 인한 재작업 등의 우발적으로 발생하는 사건들로 인해 야기되는 생산환경의 불확실성에 대처하는 문제이다. 이러한 불확실성을 생산시스템의 스케줄링에 반영하기 위해 그간 많은 연구가 있어왔으며 여기에는 크게 확률이론을 이용한 접근방법과 퍼지이론을 응용한 방법이 있다. 일반적으로 확률이론을 이용한 해석적 접근방법은 두 가지 점에서 비판받고 있다. 첫째로는 이 기법을 이용하여 풀어야 하는 문제는 대상의 규모가 커지고 복잡해질 경우 그 해를 구하기가 매우 어렵다[3]. 둘째로는 이 방법이 적용되기 위해서는 설비나 제품등에 관한 장기간에 걸친 통계 데이터가 요구되는데 반해 실제 생산 현장에서는 이와같은 데이터가 존재하지 않는 경우가 많다[22]. 그렇기 때문에 생산 현장에서는 대부분의 경우 스케줄링 전문가들의 경험적 지식에 의하여 스케줄을 수립하고 있다. 그런데 그들이 가지고 있는 지식은 명확히 표현되기 보다는 대부분 “~정도”와 같이 애매하게 표현되는 경우가 많다[8][21]. 이와 같은 특성에 의해 최근 퍼지 논리를 도입한 스케줄링에 관한 연구가 활발히 진행되고 있다[7][8][21]. 생산시스템의 스케줄링 문제에 퍼지이론을 도입 하는데는 퍼지 규칙 베이스를 이용한 방법과 퍼지 최적화 기법을 이용하는 두가지 방법이 있을 수 있다. 퍼지 규칙 베이스 응용에 관한 연구로는 다음과 같은 것들이 있다. 먼저 Turksen 등[10]이 Fox[6]등에 의하여 제안된 바 있는 ISIS를 바탕으로 퍼지 규칙 베이스와 근사적 추론 기법을 이용하여 job shop 스케줄링 문제를 계층

적으로 해결하는 기법을 제안하였다. Luis 등[13]은 Gershwin 등이 제안한 생산시스템의 계층적제어 개념을 도입하여 작업할당 문제에는 퍼지 다속성 의사 결정(Multi-Attribute Decision Making), 생산을 결정단계에는 퍼지 규칙 베이스를 이용한 퍼지제어기를 적용하여 그 효율성을 입증 하였다. 또한, Chiang 등[14]은 단일기계의 스케줄링 문제에서 기계고장으로 인한 영향을 완화하기 위하여 이미 작성된 스케줄을 재조정하는 작업에 퍼지 집합 개념을 도입하였다. 물론 이들 기법들을 이용하여 생산환경의 불확실 요인을 반영한 스케줄링이 가능하지만 그 성격상 결국 이들도 일종의 휴리스틱이므로 도출된 스케줄링 결과의 최적 여부에 대한 객관적인 평가가 어렵고 더구나 성능 역시 상황에 따라 변화가 심하다는 문제점이 있다. 한편 퍼지 최적화 기법을 응용한 연구들은 다음과 같다. 먼저 Hintz와 Zimmermann 등[20]은 FMS의 스케줄링 문제를 4 계층으로 나누어 최상위 단계인 계획단계의 문제의 해결에 퍼지 최적화 기법을 도입하고, 나머지 세 단계의 문제들(작업분배, 기계할당 문제)은 근사적 추론 기법을 이용하여 해결하는 방안을 제시 하였다. 한편 Hiroaki 등[7]은 n-작업, 2-기계 스케줄링 문제를 퍼지 최적화 문제로 모델링하여 해결하는 기법을 제안 하였다. 이들 방법들은 앞서 퍼지 규칙베이스가 가진 문제점의 해결이 가능하나 이 방법 역시 실재적 문제에 적용하기 어렵다는 문제점이 있다. 또한 이들 연구에서 대상으로 하는 시스템 단일공정으로 이루어진 간단한 경우였다.

본 연구는 다중공정으로 이루어진 유연 흐름 생산시스템을 대상으로 불확실한 생산환경하에서도 효율적인 생산제어가 수행되도록 하기 위한 스케줄링 기법의 개발을 목적으로 한다. 유연 흐름 생산시스템은 기존의 직렬 생산라인에 유연성이 부가되어 확장된 형태의 생산시스템으로서 그 최적 스케줄링에 관한 문제는 매우 난해하기 때문에 최근 이에 관한 많은 연구가 진행되고 있다. 근간의

접수일자 : 1997. 7. 4., 수정완료 : 1998. 4. 3.

박승규 : 삼성 SDS

이창훈 : 한라공과대학교 전기공학과

장석호, 우광방 : 연세대학교 전기공학과

연구들로는 Salvador[17], Wittrock[18][19], Sriskandarajah [1], Kochhar[20]이 있으나 이들 연구들은 대부분 스케줄 발생에 소요되는 시간이 지나치게 길거나[17] 휴리스틱의 사용으로 인해 도출된 결과에 대한 최적 여부의 객관적인 평가가 어렵다는 문제가 있다[1][18]-[20]. 더구나 이들 연구에서는 생산시스템에서 발생할 수 있는 이상상황에 대한 고려를 전혀하지 않았다.

본 연구에서는 스케줄링 기법의 개발을 위하여 기계고장이나 작업자 결원 등과 같은 우발적 사건으로 인한 가공용량의 불확실성을 퍼지수로 반영하는 퍼지 최적화 기법을 이용하였다. 또한 Lagrangian relaxation 기법[4][15][17]을 이용하여 대상 문제를 보다 해결이 용이한 여러 개의 서브 문제들로 분리하여 해결하는 근사 최적의 스케줄링 기법을 제시하였다.

제시된 기법의 성능을 평가하기 위하여 실공장 데이터를 기반으로하는 시뮬레이션을 통해 퍼지화 하지않은 기존의 최적화 기법, 현재 공장에서 사용하고 있는 기존 기법의 결과들과 성능 비교를 수행하였다.

II. 유연 흐름 생산시스템 스케줄링 문제의 정의

1. 대상 공정의 모델링

스케줄링 알고리즘의 개발을 위하여 본 논문에서 대상으로 한 유연 흐름 생산시스템은 중량, 중품종의 제품을 생산할 수 있는 FMS(Flexible Manufacturing System)이다(그림 1).

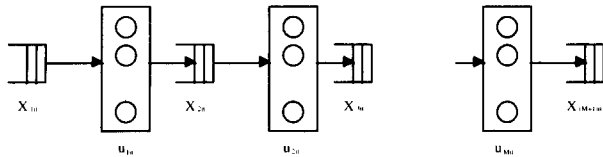


그림 1. 유연 흐름 생산시스템.
Fig. 1. A flexible flow shop.

이와 같은 생산시스템은 보통 수대 내지 수십대의 기계들과 작업자들로 구성된 작업장들이 일렬로 배치된 형태를 취한다. 각 작업장 간에는 전단계의 공정에서 생산된 반제품을 임시로 저장하기 위한 버퍼가 존재한다. 각 제품들은 미리 정해진 가공경로를 따라 가공되며 모든 공정을 다 거친 후 완성된 제품은 출하창고에 저장되었다가 수요에 대응한다. 각 공정 간에 제품 이동은 보통 중간 재고를 통해 이루어 지므로 공정간 이동시간은 무시할 수 있다. 또한 FFS에서는 다양한 제품이 생산되므로 같은 설비에서 여러 종류의 제품을 가공하게 된다. 이와 같은 방식의 생산을 위해서는 각 제품의 가공 전에 작업 준비시간(setup time)이 필요하게 된다. 이와같은 FFS를 표현하기 위해 다음과 같은 용어를 정의한다.

입력데이터

- I : 품종 수
- i : 품종 번호, $i=1, \dots, I$
- D_i : 품종 i 의 요구량
- p_{im} : 품종 i 의 공정 m 에서의 가공시간 (cycle time)
- S_{im} : 품종 i 를 공정 m 에서 생산하기 위해 필요한 작업 준비 시간
- d_i : 품종 i 의 납기일
- M : 공정의 총 수, $m=1, \dots, M$
- T : 스케줄링 총 주기
- t : 시간 주기 번호, $t=1, \dots, T$

- C_{mt} : 주기 t 에 공정 m 의 가공용량
- h_{im} : 버퍼 m 에서 품종 i 의 재고유지 비용
- l_{mt} : 주기 t 에 공정 m 에서 생산을 외주화할 경우의 비용

의사결정 변수

- X_{imt} : 주기 t 의 시작에 버퍼 m 에 존재하는 품종 i 의 재고량
- u_{imt} : 주기 t 동안 가공을 위해 작업장 m 에 투입되는 품종 i 의 총량
- O_{mt} : 주기 t 동안 공정 m 에서 외주화 해야할 용량

모든 가공품들은 1 주기 이내에 모두 가공이 끝난다고 가정할 경우 시간 주기 t 시작에 임의의 공정 m 에 투입된 가공품들은 시간 주기 $t+1$ 시작까지 모두 버퍼 $m+1$ 에 투입되게 된다. 그림 1에서 알 수 있는 바와 같이 라인 맨 앞의 버퍼($m=1$)는 원료만을 계속 라인에 투입하는 원료 창고(source)가 되고 맨 마지막 버퍼($m=M+1$)는 가공이 모두 끝난 완성된 제품들을 저장하는 발주업체의 창고(sink)의 역할을 하게 된다. 따라서 대상 생산시스템에서 가공품의 흐름과 상태는 다음과 같은 방정식들을 통해 표현 되어질 수 있다.

$$X_{i1} = D_i \tag{1}$$

$$X_{iml} \text{ given, } m = 2, \dots, M \tag{2}$$

$$X_{i(t+1)} = X_{it} - u_{it} \tag{3}$$

$$X_{im(t+1)} = X_{imt} - u_{imt} + u_{i(m-1)t}, m=2, \dots, M \tag{4}$$

$$X_{i(M+1)(t+1)} = X_{i(M+1)t} + u_{imt} \tag{5}$$

for $i=1, \dots, I$ and $t=1, \dots, T-1$

위와 같은 형식의 방정식들을 통해 제품들의 가공 선후 관계가 쉽게 표시됨을 알 수 있다. 임의의 시간 주기 t 에 공정 m 에 추가되는 부하의 총량은 $\sum_{i=1}^I p_{im} u_{imt}$ 가 되고 여기에 각 품종 간의 교체 생산을 위해 필요한 작업 준비 시간 S_{im} 을 합한 양은 그 주기 동안에 해당 공정에서 사용할 수 있는 최대 가공 용량과 외주화 용량의 합을 초과해서는 안된다. 이것은 다음과 같은 제약식으로 표시될 수 있다.

공정에서의 가공용량에 관한 제약

$$\sum_{i=1}^I (p_{im} u_{imt} + S_{im}) \leq C_{mt} + O_{mt}, \quad \forall m, \forall t \tag{6}$$

또한 외주화할 수 있는 용량에도 한계가 있으며 이것은 다음과 같은 제약식으로 표현된다.

최대 외주용량에 관한 제약

$$0 \leq O_{mt} \leq C_{omt}, \quad \forall m, \forall t \tag{7}$$

앞에서 언급하였듯이 $M+1$ 번째 버퍼는 발주업체의 창고를 표시한다. 여기서 F_{it} 를 주기 t 동안에 출하되는 제품의 양을 이라 하면 다음의 등식이 성립된다.

$$F_{it} = u_{imt}, \quad \forall i, \forall t \tag{8}$$

또한 다음과 같이 각 공정에서의 제품의 생산량은 항상 음이 아닌 정수이어야 한다.

$$u_{imt} \geq 0, \text{ integer } \forall i, \forall m, \forall t \tag{9}$$

각 공정간에 제품 전달은 재고를 통해 이루어지므로 공정간에는 일정량 이상의 제품이 항상 존재하여야 한다. 따라서 다음과 같은 제약식이 정의된다.

$$X_{imt} \geq X_{ims}, (X_{ims} \geq 0) \text{ integer } \forall i, \forall m, \forall t \tag{10}$$

X_{ims} 는 안전재고량에 해당하는 것이며 스케줄링 담당자에 의해 임의로 결정되는 상수이다.

한편, 본 논문에서의 스케줄링 목적은 다음의 세가지로 정리될 수 있다.

- 1) 제품의 납기준수
 - 2) 외주에 소요되는 비용의 최소화
 - 3) 재공(work in process)유지에 소요되는 비용의 최소화
- 재공유지나 외주화에 의해 소요되는 비용은 앞에서 정의한 상수 h_{im} 과 l_{mt} 에 의해 쉽게 표시될 수 있다. 주기 t 에 완성된 각 제품 i 에 대해 납기 준수에 관한 항을 표시하기 위하여 다음과 같은 계수 ψ_{it} 를 정의한다.

$$\psi_{it} = \begin{cases} B_i, & t \leq d_i \\ A_i, & t > d_i \end{cases} \quad (11)$$

여기서 계수 A_i, B_i 는 공장의 생산방식, 각각의 제품들이 갖는 가공 우선순위 그리고 상대적 중요도 등에 따라 각각 다르게 정의될 수 있다. 여기서 제품이 납기 이전에 완성 되었을 경우 ψ_{it} 는 재고비용이 되며 납기를 초과했을 경우에는 납기초과에 대한 벌칙비용이 된다.

이상을 정리해보면 생산시스템에서의 스케줄링 문제는 모든 생산 제약조건들의 제한하에서 전체 생산비용이 최소화 되도록 공장과 협력업체간에 작업부하를 분배하는 것이 되며 이를 수리적으로 나타내면 다음과 같이 된다.

$$(P) \min_{(u, O)} \sum_{i=1}^I \sum_{t=1}^T [\sum_{m=2}^M h_{im} X_{imt} + \psi_{it} F_{it}] + \sum_{m=1}^M \sum_{t=1}^T l_{mt} O_{mt} \quad (12)$$

subject to the constraints (1)~(10).

2. 퍼지 최적화 문제로의 변환

앞절에서 정의된 스케줄링 문제는 생산용량이 항상 일정상태라고 가정한 경우이다. 그러나 실제의 경우 기계고장이 항상 발생한다. 이러한 사건들은 우발적인 성격을 갖기 때문에 스케줄 작성시에 임의의 시간주기 t 동안에 가용할 수 있는 생산용량을 미리 산정할 수 없게되며 이로 인해 생산시스템의 스케줄링 문제는 더욱 난해하게 된다. 이와 같이 미래의 임의의 주기 t 에서의 불확실한 가공용량을 모델링하기 위해 퍼지수 C_{Mt} 를 도입한다. 이것은 임의의 주기 t 에 공정 m 에서 가용할 수 있는 가공용량을 의미하며 소속 함수(membership function)는 다음과 같다.

$$\mu_{C_{Mt}}(x) = \begin{cases} 1, & 0 \leq x \leq C_{Mt}^f \\ (C_{Mt}^f + C_{Mt}^v + 1 - x) / (C_{Mt}^v + 1), & C_{Mt}^f \leq x \leq C_{Mt}^f + C_{Mt}^v + 1 \\ 0, & otherwise \end{cases} \quad (13)$$

이와같은 소속 함수를 그래프로 표시하면 그림 2와 같다.

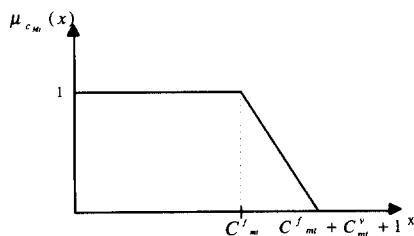


그림 2. C_{Mt} 의 소속 함수.
Fig. 2. Membership function of C_{Mt} .

여기서 x 가 가공용량을 나타내는 변수라할 때 C_{Mt} 는

임의의 주기 t 에 공정 m 에서 있을 수 있는 최소 가용 가공용량을 나타내며 C_{Mt} 는 생산환경의 불확실 요인들에 의해 변동될 수 있는 가변 용량을 나타낸다. 따라서 $C_{Mt}^f + C_{Mt}^v + 1$ 는 최대 가용 가공용량을 나타내게 된다. 소속 함수의 의미는 다음과 같다. 스케줄 작성시 각 주기별 작업부하를 계산하였을때 계산된 부하가 C_{Mt} 보다 작으면 부하된 부하를 모두 생산할 수 있다(소속 값=1)는 것이다. 만일 부하가 C_{Mt} 보다 클경우 생산용량의 불확실성 때문에 100% 생산할 수 있다고 확실히 단정할 수 없다는 것 ($0 < \text{소속 값} < 1$)이고 $C_{Mt}^f + C_{Mt}^v + 1$ 이상이 되는 경우는 100% 모두를 생산할 수 없음이 확실하다는 의미이다 (소속 값=0). 한편, 협력업체의 생산용량에는 불확실 요소가 없다고 가정한다.

이와같이 가공용량을 모델링할 경우 앞 절에서 정의된 가공용량에 관한 제약식 (6)은 다음과 같이 변형된다.

$$\sum_{i=1}^I (p_{im} u_{imt} + S_{im}) - O_{mt} \leq C_{Mt}, \quad \forall m, \forall t \quad (14)$$

정의된 스케줄링 문제는 퍼지 최적화 문제가 되고 이의 해결을 위하여 본 논문에서는 대칭기법(symmetric approach)을 사용한다[8]. 이는 목적함수 자체를 하나의 퍼지 제약조건화 하여 각 제약조건들의 최소 만족도를 최대화 시키는 방법이다. 이 기법의 적용에 의해 스케줄링 목적함수는 다음과 같은 제약식으로 변형될 수 있다.

$$\min_{(u, O)} \sum_{i=1}^I \sum_{t=1}^T [\sum_{m=2}^M h_{im} X_{imt} + \psi_{it} F_{it}] + \sum_{m=1}^M \sum_{t=1}^T l_{mt} O_{mt} \leq J_0$$

$$\sum_{i=1}^I (p_{im} u_{imt} + S_{im}) - O_{mt} \leq C_{Mt}^f, \quad \forall m, \forall t$$

subject to the constraints (1) ~ (5), (7) ~ (10).

제약식화 된 목적함수의 소속함수는 다음과 같이 정의된다(그림 3).

$$\mu_f(x) = \begin{cases} 1, & 0 \leq x \leq J_0 \\ (J_0 + \Delta_J - x) / \Delta_J, & J_0 \leq x \leq J_0 + \Delta_J \\ 0, & otherwise \end{cases} \quad (15)$$

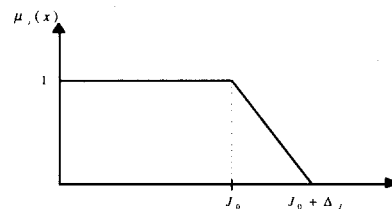


그림 3. 목적함수의 소속 함수.
Fig. 3. Membership function of objective function.

여기서 J_0 는 이상적인 경우의 스케줄링 비용으로써 작성된 스케줄에 의해 계산된 비용 (12)가 이것 이하이면 만족도가 좋다는 것이며 그 이상이 될 경우, 목적함수의 최대 허용 변화량을 Δ_J 라 할 때, $J_0 + \Delta_J$ 가 될 때까지 만족도가 감소한다는 것이다. $J_0 + \Delta_J$ 는 목적함수의 한계값으로써 경험이나 실험에 의해 정해질 수 있는 값이다. 대칭 모델로 변환 했을때 이 문제는 u_{imt} 와 O_{mt} 를 적절히 결정하여 목적함수와 생산용량에 관한 퍼지 제약식들의 만족도를 가능한한 최대화 되도록 하는 문제가 된다.

$$z^* = \max_{(u, O)} \min (\mu_f(J(u, O)), \mu_{11}(\sum_{i=1}^I (p_{i1} u_{i1} + S_{i1}) - O_{11}),$$

$$\mu_{21}(\sum_{i=1}^I (p_{i2} u_{i2} + S_{i2}) - O_{21}), \dots, \mu_{M1}(\sum_{i=1}^I (p_{iM} u_{iM} + S_{iM}))$$

$$-O_M), \dots, \mu_{1T}(\sum_{i=1}^I (p_{i1} u_{i1T} + S_{i1}) - O_{1T}), \dots, \\ \mu_{(M-1)T}(\sum_{i=1}^I (p_{i(M-1)} u_{i(M-1)T} + S_{i(M-1)}) - O_{(M-1)T}), \\ \mu_{MT}(\sum_{i=1}^I (p_{iM} u_{iMT} + S_{iM}) - O_{MT})$$

subject to the constraints (1)-(5), (7)-(10).

여기서 위의 식은 다음과 같은 식으로 변환할 수 있다.

$$(FP) \max z \tag{16}$$

$$\text{subject to } J(\mathbf{u}_i, \mathbf{O}) + z\Delta_J \leq J_0 + \Delta_J, \tag{17}$$

$$\sum_{i=1}^I (p_{im} u_{im} + S_{im}) - O_{mt} + z(C^f_{mt} + C^v_{mt} + 1) \leq C^f_{mt} + C^v_{mt} + 1, \tag{18}$$

$$0 \leq z \leq 1 \quad m = 1, \dots, M, \quad t = 1, \dots, T$$

subject to the constraints (1)-(5), (7)-(10).

III. Lagrangian relaxation 기법을 이용한 부최적 스케줄링 알고리즘

앞 장에서 정의된 문제 (FP)는 일반적으로 NP 복잡성을 갖는 문제로써 이것의 최적해를 구하는 것은 매우 난해하다[11]. 따라서 본 논문에서는 그 대안으로 계산상으로 보다 효율적인 부최적의 해를 찾는 알고리즘 개발을 목적으로 한다. 알고리즘의 전체 구성은 다음과 같다.

Step 1 : Lagrangian relaxation 기법을 이용하여 제약식을 완화하고 dual 문제로 변환한다.

Step 2 : 해결이 보다 쉬운 부문제들로 분리한다.

- 1) 생산스케줄링 부문제; (PS) Production Scheduling Subproblem.
- 2) 멤버십 부문제; (MS) Membership Subproblem.
- 3) 외주용량 할당 부문제; (OCA) Outside Capacity Allocation Subproblem.

Step 3 : 최소비용 네트워크 흐름 문제 알고리즘을 이용하여 부문제 (PS)의 해를 구한다.

Step 4 : 부문제 (MS)의 해를 구한다.

Step 5 : 부문제 (OCA)의 해를 구한다.

Step 6 : Subgradient 알고리즘을 이용하여 dual 문제의 최적해를 구한다.

Step 7 : Step 1)~Step 6)에 의해 구해진 스케줄이 비가능(infeasible)해이므로 휴리스틱 알고리즘을 통해 가능해로 재구성한다.

1. Lagrangian relaxation과 부문제로의 분할

퍼지 최적화 문제 (FP)에 Lagrangian relaxation 기법을 이용하여 제약식 (17)과 (18)을 완화(relax)시켜 Lagrangian 함수를 구한다. 여기서 $\max z$ 는 구간 $[0, 1]$ 에서 $\max(-(1-z)^2+1)$ 와 같으므로 본 논문에서는 후자의 식을 사용한다. 그 이유는 전자의 식을 사용할 경우 해인 z 의 값이 극대(=1), 극소(=0)값을 번갈아 갖게되는 진동 현상이 심하며 이러한 현상은 해의 수렴을 어렵게 만들기 때문이다. 후자의 식이 이용될 경우 이와 같은 문제를 피할 수 있게 된다.

$$(R) : \min_{\{\mathbf{u}_i, \mathbf{O}, z\}} ((1-z)^2 + \lambda_J (J(\mathbf{u}_i, \mathbf{O}) + z\Delta_J - (J_0 + \Delta_J))) \tag{19} \\ + \sum_{t=1}^T \sum_{m=1}^M \pi_{mt} (\sum_{i=1}^I (p_{im} u_{im} + S_{im}) - O_{mt} + z(C^f_{mt} + C^v_{mt} + 1) - (C^f_{mt} + C^v_{mt} + 1)) - 1 \\ , \lambda_J, \pi_{mt} > 0, \quad 0 \leq z \leq 1$$

subject to the constraints (1)-(5), (7)-(10).

여기서 λ_J 와 $\{\pi_{mt}\}$ 는 Lagrangian 승수(multiplier)이며 항상 양의 값을 갖는다. 여기서 $\mathbf{u}_i = \{u_{imt}, \forall m, \forall t\}$ 라 할때 (19)는 주어진 Lagrangian 승수 λ_J 와 $\pi = \{\pi_{mt}, m=2 \dots M, \forall t\}$ 에 대하여 \mathbf{u}_i 와 O_{mt} 그리고 z 에 관련된 항들로 분리가능한 함수이다. 먼저 품종 i 에 관련된 항들을 묶어 다음과 같은 함수를 정의한다.

$$P_i(\mathbf{u}_i, \lambda_J, \pi) \equiv \sum_{t=1}^T (\sum_{m=2}^M \lambda_J h_{im} X_{imt} + \lambda_J \psi_{it} F_{it}) \tag{20} \\ + \sum_{m=1}^M \pi_{mt} p_{im} u_{im}$$

또한, z 와 관련된 항들을 묶어 다음의 함수를 정의한다.

$$P_z(z, \lambda_J, \pi) \equiv ((1-z)^2 + z \lambda_J \Delta_J + z \sum_{t=1}^T \sum_{m=1}^M \pi_{mt} (C^f_{mt} + C^v_{mt} + 1)) \tag{21}$$

마지막으로 주기 t 에 공정 m 에 대하여 다음의 함수를 정의한다.

$$P_{omt}(O_{mt}, \lambda_J, \pi) \equiv (\lambda_J l_{mt} - \pi_{mt}) O_{mt} \tag{22}$$

한편 local duality 정리에 의하여 문제 (R)의 dual 문제 (D)를 다음과 같이 정의할 수 있다

$$(D) : \max_{\lambda_J, \pi \geq 0} \Phi(\lambda_J, \pi) \equiv \sum_{i=1}^I \min_{\mathbf{u}_i} P_i(\mathbf{u}_i, \lambda_J, \pi) \tag{23}$$

$$+ \sum_{m=1}^M \sum_{t=1}^T \min_{O_{mt}} P_{omt}(O_{mt}, \lambda_J, \pi) + \min_z P_z(z, \lambda_J, \pi) \\ - \lambda_J (J_0 + \Delta_J) - \sum_{t=1}^T \sum_{m=1}^M \pi_{mt} (C^f_{mt} + C^v_{mt} + 1) - 1$$

subject to the constraints (1)-(5), (7)-(10).

(23)을 보면 dual 문제에는 다음과 같은 3가지의 독립적인 문제들이 존재함을 알 수 있다.

1) 품종 각각에 대해 가공용량에 관한 제약이 없는 생산 스케줄링 부문제

$$(PS-i) \min_{\mathbf{u}_i} P_i(\mathbf{u}_i, \lambda_J, \pi), \quad i=1 \dots I$$

subject to the constraints (1)-(5), (8)-(10).

2) 소속도 부문제

$$(MS) \min_z P_z(z, \lambda_J, \pi)$$

3) 주기 t 에 공정 m 에서의 외주 용량 할당 부문제

$$(OCA-mt) \min_{O_{mt}} P_{omt}(O_{mt}, \lambda_J, \pi)$$

subject to the constraints (7) and (9).

1.1 생산 스케줄링 부문제 해결을 위한 최소 비용 네트워크 유통 알고리즘

2장에서 정의된 (1)~(5)에 의해 첫번째 부문제는 최소 비용 네트워크 유통 문제의 형태임을 알 수 있다. 즉, 여기서 노드 n_{mt} 를 m 번째 버퍼의 시각 t 에서의 상태라 할때 노드 n_{mt} 와 노드 $n_{m(t+1)}$ 을 연결하는 아크 $(n_{mt}, n_{m(t+1)})$ 는 주기 t 의 생산량 u_{imt} 로 표시될 수 있음을 알 수 있다. 또한 생산량 u_{imt} 만큼의 양이 흘러 나간뒤 버퍼 n_{mt} 에 남아있는 가공품들은 다음 주기에서의 버퍼의 상태 $n_{m(t+1)}$ 가 되게 되고 이 두 노드를 연결하는 아크 $(n_{mt}, n_{m(t+1)})$ 는 재공품을 나타내는 X_{imt} 로 표시될 수 있음을 알 수 있다. 여기서 소스(source) 노드를 S, 싱크(sink) 노드를 T라 할때 소스 노드 S는 노드 n_{11} 와 연결되어 여기에는 일정량의

흐름 D_j 가 흐르도록 되어 있으며 노드 n_{im} , ($m=2, 3, \dots, M$)들과는 버퍼의 초기 상태에 해당하는 X_{im1} 이 흐르도록 되게 된다. 또한 노드 n_{im} 와 T 를 연결하는 아크 (n_{im}, T)는 주기 t 에서 발주업체로의 완제품 출하량을 나타내는 것으로 F_{it} 로 표시할 수 있다. 이와 같이 정의된 네트워크의 각 노드에서 유통량 보존(flow conservation)법칙을 적용하면 2장에서 정의된 (1)~(5)들 중에 하나가 도출됨을 알 수 있다. 각 노드를 연결하는 아크를 통해 흐를 때의 비용은 부문제 (PS-i)를 나타내는 식에서 쉽게 알 수 있다. 따라서 이 문제는 전형적인 네트워크 유통 문제임을 알 수 있다. 이 문제의 해를 구하는 기법에 관한 여러 알고리즘이 기존에 나와 있다. 본 연구에서는 Klein 법을 이용한다 [2][12].

1.2 소속도 부문제의 해결을 위한 알고리즘

소속도 부문제 (MS)는 그 형태상 간단한 2차 함수이므로 꼭지점에 해당하는 z 값이 최소가 되나 z 가 취할 수 있는 값이 구간 $[0, 1]$ 사이의 값으로 제한되어 있으므로 다음과 같이 구한다.

- i) if $(\lambda_j \Delta_j + \sum_{m=1}^T \sum_{i=1}^M \pi_{mi} (C^v_{mi} + 1))/2 \leq 0$, then $z=0$
- ii) if $0 < (\lambda_j \Delta_j + \sum_{m=1}^T \sum_{i=1}^M \pi_{mi} (C^v_{mi} + 1))/2 < 1$, then $z = 1 - (\lambda_j \Delta_j + \sum_{m=1}^T \sum_{i=1}^M \pi_{mi} (C^v_{mi} + 1))/2$
- iii) if $(\lambda_j \Delta_j + \sum_{m=1}^T \sum_{i=1}^M \pi_{mi} (C^v_{mi} + 1))/2 \geq 1$, then $z=1$

1.3 서브 문제 (OCA-mt)의 해결을 위한 알고리즘

임의의 $\lambda_j, \pi_{mi}, u_i, z$ 에 대하여 부문제 (OCA-mt)의 해 O_{mt} 는 Kuhn-Tucker의 최적화 조건[5]과 제약식 (7)에 의하여 다음과 같이 구할 수 있다.

- 1) if $(\lambda_j l_{mt} - \pi_{mi}) > 0$ then $O_{mt} = 0$
- 2) if $(\lambda_j l_{mt} - \pi_{mi}) < 0$ then $O_{mt} = C_{omt}$
- 3) if $(\lambda_j l_{mt} - \pi_{mi}) = 0$ then

$$O_{mt} = \min \{ \max \{ 0, \sum_{i=1}^M (p_{im} u_{imt} + S_{im}) - (C^f_{mt} + (1-z^*)(C^v_{mt} + 1)) \}, C_{omt} \}$$

2. Dual 문제의 해결을 위한 subgradient 알고리즘

임의의 주어진 (λ_j, π) 에 대하여 모든 부문제들의 해를 찾은 후에 dual 함수의 값을 평가할 수 있게 된다. 그런데 dual 함수의 미분이 불가능하므로 dual 문제의 해결을 위해 본 논문에서는 subgradient 기법을 사용한다[16]. 여기서 u_i^*, O_{mi}^*, z^* 가 임의의 Lagrangian 승수 λ_j 와 π_{mi} 에 대응하는 최적해라하면 dual 함수 $\Phi(\lambda, \pi)$ 의 subgradient를 다음과 같이 정의된다.

$$\frac{\partial}{\partial \lambda_j} \Phi(\lambda_j, \pi) = \sum_{i=1}^T \sum_{m=1}^M (\sum_{i=2}^M h_{im} X_{imt} + \psi_{it} F_{it}) + z^* \Delta_j - (J_0 + \Delta_j)$$

$$\frac{\partial}{\partial \pi_{mi}} \Phi(\lambda_j, \pi) = \sum_{i=1}^T (p_{im} u_{imt}^* + S_{im}) - O_{mi}^* - (C^f_{mt} + (1-z^*)(C^v_{mt} + 1))$$

Subgradient 방법에 의하여 Lagrangian 승수 λ_j 는 다음 식에 의해 반복적으로 수정된다.

$$\lambda_j^{n+1} = \lambda_j^n + \alpha^n g(\lambda_j)^n$$

여기서 n 은 반복횟수를 나타내고 α^n 는 현단계의 스텝 사이즈, $g(\lambda_j)^n$ 는 현단계의 탐색 방향이다. 스텝 사이즈 α^n 는 매 반복때마다 다음과 같이 구해진다.

$$\alpha^n = \beta \frac{\Phi^U - \Phi^n}{(g(\lambda_j)^n)^T (g(\lambda_j)^n)}, \quad 0 < \beta < 1$$

여기서 Φ^U 는 dual 함수 $\Phi(\lambda, \pi)$ 가 갖는 최적해의 상한 추정치이고 이 값은 반복 초기에 충분히 큰 값을 주게 된다. Φ^n 는 반복횟수가 n 일때 dual 함수의 값이다. 한편 승수 π_{mi} 역시 같은 방식으로 수정된다.

3. 가능(feasible)해를 구하기 위한 휴리스틱 알고리즘

Dual 문제의 해를 통해 구성된 스케줄은 보통 비가능해인 경우가 많다. 즉, 어떤 주기 t 에 공정 m 에서 작성된 스케줄을 통해 부가되는 부하가 그 공정의 추정 가공용량을 초과하는 경우가 생길 수 있다. 여기서, 추정 가공용량 (Estimated Production Capacity: EPC)이란 소속도 문제의 해를 통해 구한 z^* 에 의해 그 주기에 있으리라 추정되는 가공용량으로 그 값은 아래와 같다.

$$EPC = C^f_{mt} + (1-z^*)(C^v_{mt} + 1) + O_{mi}^*$$

이와 같은 문제 때문에 초기에 작성된 스케줄을 조정해 주어야 하며 이것을 휴리스틱 알고리즘을 통하여 수행한다. 먼저 스케줄링 전체 주기에 걸쳐서 가공용량을 초과하는 공정이 있는가를 검사한다. 만일 가공용량의 제약을 위배하는 경우가 생기면 적당한 가공품을 선택하여 작성된 초기 스케줄에서 우선 제외시킨 후 제약을 준수하도록 재스케줄 해야한다. 주기 t 에 가공용량 제약을 위배한 공정에 할당된 가공품들의 배치(batch) 크기가 다양하므로 우선 각 배치의 우선순위를 계산한 후에 우선순위에 따라 스케줄을 조정해 나간다. 각 배치의 우선순위(Priority Factor: PF)를 다음과 같이 정의한다.

$$PF = \frac{\text{time available from now to due date}}{\text{residual processing time of parts in the batch}} \quad (24)$$

PF 값이 작을수록 그 배치의 우선순위가 높은 것이다. 이와같이 조정할 배치가 결정되게 되면 휴리스틱 알고리즘에 의해 해당배치의 재 스케줄링(Rescheduling)이 시작된다. 알고리즘내용은 아래와 같다.

Step 1 : dual 문제의 해를 통해 초기 스케줄을 작성한다.
Do for from $t=1$ to T

Do for from $m=1$ to M

Step 2 : 각 공정 m 에서 주기 t 에 추정된 가공용량에 관한 제약이 위반되는가를 검사한다.

Step 3 : 가공용량이 위배되는 공정이 있는 경우 계산된 우선순위에 따라서 초과된 양을 제거하기 위해 초기 스케줄로부터 제외시킬 품종을 결정한다.

Step 4 : 우선순위 낮은 품종부터 재스케줄링을 실시한다.

- a. 재스케줄링 대상 품종의 네트워크를 구성한다.
- b. 구성된 네트워크의 각 아크 용량을 다음과 같이 수정한다.

① 구간 $[0, t]$ 에 속하는 아크의 용량은 해당 공정의 여유 가공 용량으로 세트한다.

② 구간 $[t+1, T]$ 에 속하는 아크의 용량은 해당 공정의 추정 가공용량으로 세트한다.

c. 최소비용 유통문제를 풀어 수정된 네트워크내에서의 재배치품종의 최소비용경로를 설정한다.

Enddo

Enddo

아크 u_{im} 의 용량을 구간 $[0, t]$ 에서 Step 4에서와 같이 설정한 이유는 재스케줄링될 제품의 흐름이 이 구간내에서는 각 공정이 가지고 있는 가공용량을 절대로 초과하지 못하게하여 가능한 스케줄의 작성을 보장하기 위해서이다. 한편 구간 $[t+1, T]$ 에서 새로이 용량제약의 위반이 발

생할 수 있으나 위의 휴리스틱 알고리즘의 진행순서가 선행 공정에서 후행공정(m의 오름차순)순으로 시간상으로는 t의 오름차순으로 진행되므로 결국 최종적으로는 가능한 스케줄을 구성하게 된다. 그러나 이와같이 구성된 스케줄은 가능해 이기는하나 최적 스케줄이라 보장할 수 없으며 단지 부최적의 스케줄이 된다.

IV. 시뮬레이션 결과 및 고찰

본 장에서는 제안된 스케줄링 알고리즘의 성능평가를 수행한다. 이를 위하여 먼저 본 연구에서 대상으로 하는 공장의 시뮬레이션 모델을 구성하였다. 시뮬레이션 프로그램은 C 언어를 이용하여 SUN Sparc 10 상에서 작성, 실행 하였으며 이를 대상으로 1) 기존의 운영기법(부록 A 참조), 2) 제안된 퍼지최적화 기법, 3) 퍼지화하지 않은 crisp 기법의 세 가지 기법을 적용하여 각 기법들간의 성능 비교를 실시하였다. 시뮬레이션은 가공용량의 최대 가변도를 3단계로 구분하여 실시하였으며 이때 각 기법들간의 성능비교 지표는 2장에서 정의된 스케줄링 목적함수에 따라 1) 재공 및 재고 수준, 2) 작업 지연 정도(tardness), 3) 외주량(outside order amount)으로 한다.

1. 대상 생산시스템

본 논문에서 대상으로 하는 생산시스템은 자동차 생산에 필요한 다양한 프라스틱 사출품들을 생산하는 L사의 사출 성형 공장이다.

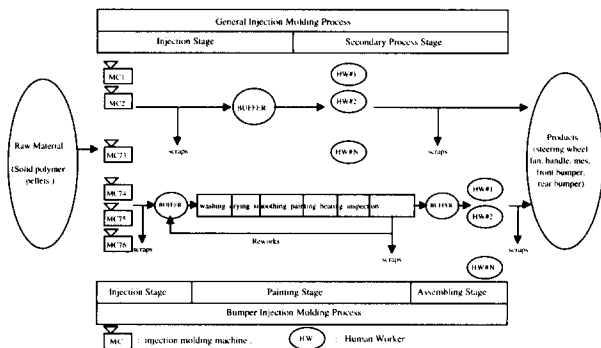


그림 4. 사출 성형 공장의 전체 개요도.

Fig. 4. Layout of the injection molding plant.

사출 성형이란 원료인 공업용 플라스틱을 가열, 용해시킨 후 일정한 모양의 틀(이를 금형이라함)에 압력을 가하여 주입한 후 냉각시켜 원하는 모양의 제품을 제조하는 것을 의미한다. 대상 공장은 크게 사출 1 공장과 사출 2 공장으로 나뉘어 있다.

사출 1 공장에서는 범퍼 사출공정이 진행되며 여기서는 자동차의 앞, 뒤 범퍼가 생산된다. 제품의 생산과정을 보면 먼저 사출단계에는 2대의 초대형 사출기들이 설치되어 범퍼 커버를 찍어낸다. 사출되어 나온 범퍼 커버들은 도장 단계를 거치면서 정해진 색채로 도장되어져 나온다. 도장작업은 위의 그림에서 보는 바와 같이 세척, 건조, 외면 다듬질, 도색, 열처리, 검사의 6단계로 구성되어져 있는데 이와 같은 일련의 과정이 페루프(closed loop) 형태의 도장 라인에 설치된 걸개(hanger)에 앞, 뒤 범퍼 범퍼 커버가 쌍으로 걸려 순환되면서 수행된다. 도장라인에는 210개의 걸개가 설치되어 있다. 도장이 끝난 범퍼 커버들은 조립단계로 전달되며 수작업에의해 몇 가지 부속품들이 결합되어 범퍼 완제품이 만들어지고 출하를 위해 창고에 적재된다. 사출 2 공장에서는 일반 사출공정이 진행되며 주로 일반 사출물들(예를 들어 손잡이, 운전대, 계기판, 냉각팬 등)이 생산된다. 사출단계에서는 70 여대의 사출기

들이 있고 여기서 자동차 생산에 필요한 여러 다양한 부품들이 생산된다. 사출기들은 특성에 따라 몇 개의 그룹으로 나뉘어지며 하나의 그룹을 기종이라 칭한다.

2. 시뮬레이션 환경

앞에서 설명한 사출공장의 경우 언급한 바와 같이 하루 생산되는 품종의 수가 800 여종이 넘으며 사출기의 수 76 대 그리고 작업자의 인원이 작업장당 수십 명이 넘는 대규모 공장이므로 본 연구에서는 제안된 기법의 검증을 위해 규모를 축소한 시뮬레이션 모델을 구성하였다. 구성된 시뮬레이션 모델에 관한 데이터는 다음과 같다.

2.1 작업장에 관한 데이터

- 1) 품종수(총 34 종)
 - 일반사출; 22 종,
 - 범퍼사출; 12 종
- 2) 공정별 설비 및 작업인원
 - 일반사출; 사출기 4 대,
 - 2차 가공; 작업자 6 명,
 - 범퍼사출; 사출기 2 대,
 - 범퍼도장; 도장라인 1 대,
 - 범퍼조립; 100 개
- 3) 도장 라인상의 걸개 수; 100 개
- 4) 1 주기; 1일 (2000 cycle)
- 5) 금형 교체시간; 2000/48≐40 cycle
- 6) 공정별 가공용량

현재 공장에서 기계로 작업이 진행되는 공정(일반사출, 범퍼사출, 범퍼도장)은 3 교대로 하루 24 시간 작업을 진행하고 수작업으로 수행되는 단계는 하루 8시간(1/3일) 작업을 원칙으로 한다. 따라서 각 공정별 가공용량은 아래와 같이 된다.

- 일반사출; 사출기 4 대 × 2000 = 8000 cycle,
- 2 차가공; 작업인원 6 명 × 2000 / 3 = 4000 cycle,
- 범퍼사출; 사출기 2 대 × 2000 = 4000 cycle,
- 범퍼도장; 1개 / 1cycle × 2000 = 2000 개,
- 범퍼조립; 작업인원 4 명 × 2000 / 3 = 2800 cycle.

7) 공정별 외주업체의 가공용량

외주업체의 가공용량은 각 생산단계의 가공용량의 1/2로 한다.

2.2 제품에 관한 데이터 및 주문 데이터

표 1. 생산 데이터.

Table 1. Product data.

Part	Alternative Machines				Processing Time				Min Lot Size	
	1 st	2 nd	3 rd	4 th	Injection	2 nd Process	Bumper Painting	Bumper Assem.	Injection	Bumper Painting
G1	M1	M2	M3	M4	2	2	-	-	20	-
G2	M2	M1	M4	M1	3	3	-	-	30	-
G3	M3	M4	M1	M2	3	3	-	-	20	-
G4	M4	M1	M2	M3	3	3	-	-	20	-
G5	M1	M2	M3	M4	2	3	-	-	30	-
G6	M2	M3	M4	M1	3	3	-	-	20	-
G7	M3	M4	M1	M2	3	3	-	-	20	-
G8	M4	M1	M2	M3	3	2	-	-	30	-
G9	M1	M2	M3	M4	3	3	-	-	20	-
G10	M2	M3	M4	M1	2	2	-	-	30	-
G11	M3	M4	M1	M2	3	3	-	-	20	-
G12	M4	M1	M2	M3	2	3	-	-	30	-
G13	M1	M2	M3	M4	3	3	-	-	20	-
G14	M2	M3	M4	M1	3	3	-	-	20	-
G15	M3	M4	M1	M2	3	2	-	-	30	-
G16	M4	M1	M2	M3	3	3	-	-	20	-
G17	M1	M2	M3	M4	2	2	-	-	30	-
G18	M2	M3	M4	M1	3	3	-	-	20	-
G19	M3	M4	M1	M2	2	3	-	-	30	-
G20	M4	M1	M2	M3	3	2	-	-	20	-
G21	M1	M2	M3	M4	3	3	-	-	20	-
G22	M2	M3	M4	M1	3	2	-	-	30	-
B1	M5	M6	M5	M5	3	-	-	2	30	20
B2	M5	M6	M5	M5	3	-	-	2	30	20
B3	M5	M6	M5	M5	3	-	-	2	30	20
B4	M6	M5	M6	M6	3	-	-	2	30	20
B5	M6	M5	M6	M6	3	-	-	2	30	20
B6	M6	M5	M6	M6	3	-	-	2	30	20
B7	M5	M6	M5	M5	3	-	-	2	30	20
B8	M5	M6	M5	M5	3	-	-	2	30	20
B9	M5	M6	M5	M5	3	-	-	2	30	20
B10	M6	M5	M6	M6	3	-	-	2	30	20
B11	M6	M5	M6	M6	3	-	-	2	30	20
B12	M6	M5	M6	M6	3	-	-	2	30	20

표 1에는 제품이 거쳐야하는 가공경로와 각 작업장에서 요구되는 가공시간, 주 기계 및 대체 기계, 최소 lot 사이즈 등에 관한 데이터가 있다. 한편, 총 34종의 1주일 간 주문량이 아래의 표 2에 나와있다. 여기서 G는 일반사출 공정 제품을 의미하고 B는 범퍼사출 공정 제품임을 의미한다.

3. 각 기법들간의 성능 비교

표 2의 주문 데이터를 이용하여 스케줄링 주기 t에 임의의 공정 m에서의 최대 가공용량의 가변도 C^v_{mt} 의 값이 그 작업장이 가진 전체 용량의 25%인 경우, 15%인 경우 그리고 5%인 경우의 3 단계로 나누어 시뮬레이션을 수행하였다. 성능비교 결과는 다음과 같다.

표 2. 주문 데이터.

Table 2. Order data.

Day	Day1	Day2	Day3	Day4	Day5	Day6	Day7	Total
G1	95	95	95	95	95	95	95	665
G2	95	95	95	95	95	95	95	665
G3	134	134	134	134	134	134	134	938
G4	93	93	93	93	93	93	93	651
G5	111	111	111	111	111	111	111	777
G6	83	83	83	83	83	83	83	581
G7	129	129	129	129	129	129	129	903
G8	118	118	118	118	118	118	118	826
G9	96	96	96	96	96	96	96	672
G10	163	163	163	163	163	163	163	1141
G11	200	200	200	200	200	200	200	1400
G12	198	198	198	198	198	198	198	1386
G13	185	185	185	185	185	185	185	1295
G14	84	84	84	84	84	84	84	588
G15	158	158	158	158	158	158	158	1106
G16	115	115	115	115	115	115	115	805
G17	188	188	188	188	188	188	188	1316
G18	189	189	189	189	189	189	189	1323
G19	88	88	88	88	88	88	88	616
G20	83	83	83	83	83	83	83	581
G21	68	68	68	68	68	68	68	476
G22	125	125	125	125	125	125	125	875
B1	150	150	150	150	150	150	150	1050
B2	145	145	145	145	145	145	145	1015
B3	133	133	133	133	133	133	133	931
B4	161	161	161	161	161	161	161	1127
B5	145	145	145	145	145	145	145	1015
B6	163	163	163	163	163	163	163	1141
B7	150	150	150	150	150	150	150	1050
B8	136	136	136	136	136	136	136	952
B9	161	161	161	161	161	161	161	1127
B10	165	165	165	165	165	165	165	1155
B11	151	151	151	151	151	151	151	1057
B12	148	148	148	148	148	148	148	1036

3.1 각 기법에 의한 재고수준 비교

그림 5에서 7까지의 각 기법에 따른 재고수준 비교결과를 분석해 보면 다음과 같다. 기존기법의 경우 작업지시시에 항상 미래의 일어날수 있는 고장에 대비하여 각 공정의 사출 단계에서는 하루분에 해당하는 수량을 더 부가하여 내리고 그 밖의 공정에서는 일정비율의 정해진 안전재고율을 곱한 수량을 생산토록 지시하게 된다(부록 A 참조). 반면 crisp 알고리즘의 경우는 작업지시시에 기기의 고장등과 같은 불확실 요인을 고려하지 않고 항상 공장의 생산 용량이 100% 가용하다고 생각하여 작업지시를 하게 된다. 한편 퍼지 최적화 알고리즘의 경우에는 공장의 생산용량이 불확실하므로 이를 고려하여 먼저 스케줄링 주기동안 사용 가능한 생산용량을 추정하여 이 한계범위 내에서 작업지시량을 결정하게 된다. 따라서 그림에서 보듯이 기존 기법의 경우는 작업지시량이 항상 과다하게 된다. 즉, 미래의 고장이거나 결원으로 인한 불안정 요인을 재고수준을 높게 유지하여 해결하려는 것이다. 반면 crisp 알고리즘은 고장에 대한 고려가 전혀 없으므로 임의의 작업장에서 고장이거나 작업자 결원으로 인해 정상적 생산이 불가능할 때 반제품 재고나 완제품 재고가 있는 경우는 이를 이용해 다음공정의 생산이나 출하가 계속 지속되도록 하지만 재고가 없을 경우 그만한 고장시간에 해당하는 만큼의 제품의 생산 및 출하가 불가능해 지게 된다. 그런데

두 가지 경우 의해서 초래되는 결과는 고장이 나면 날수록 재공이나 재고 수준이 낮아지게 된다는 것이다. 한편 퍼지 최적화 알고리즘은 생산용량에 있어서의 불확실 요인을 외주화 통해 처리하므로 각일에 지시된 작업량을 가능하면 지시된 대로 생산하게 되므로 crisp의 경우보다 재고량이 약간 높게되게 된다. 지표상으로 볼때 crisp 알고리즘이 재고 측면에서는 가장 낮은 재고를 유지하므로 가장 우수한 것 같으나 이것은 다른 지표에서의 성능을 저하 시키는 원인이 된다.

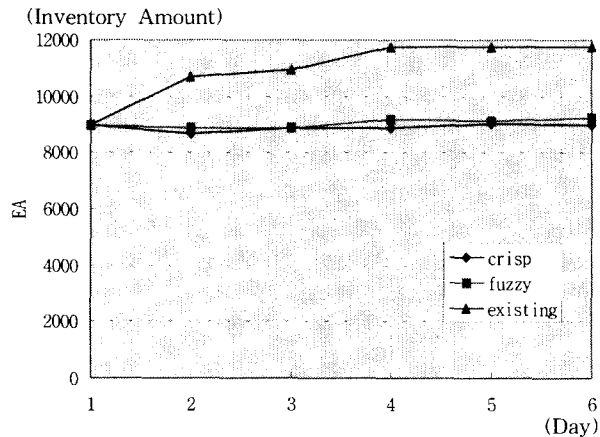


그림 5. 각 기법에 의한 재고수준 ($C^v_{mt} = 0.05 \times C_{mt}$).
Fig. 5. Inventory levels of each method ($C^v_{mt} = 0.05 \times C_{mt}$).

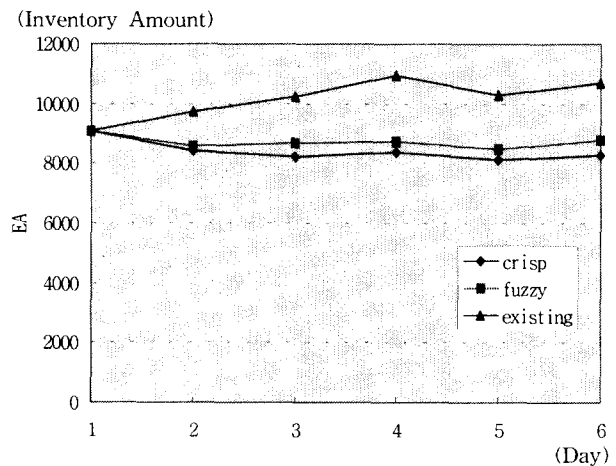


그림 6. 각 기법에 의한 재고수준 ($C^v_{mt} = 0.15 \times C_{mt}$).
Fig. 6. Inventory levels of each method ($C^v_{mt} = 0.15 \times C_{mt}$).

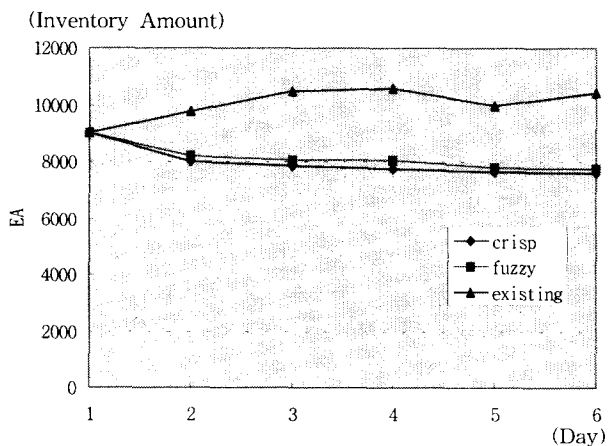


그림 7. 각 기법에 의한 재고수준 ($C^v_{mt} = 0.25 \times C_{mt}$).
Fig. 7. Inventory levels of each method ($C^v_{mt} = 0.25 \times C_{mt}$).

3.2 각 기법에 의한 외주량 비교

그림 8에서 10에 각 기법을 이용했을 때의 각 공정에서 외주화된 제품들의 일별 총 합계가 가공용량 변화도 각각에 대하여 보여져 있다. 먼저 전반적으로 보면 기존 기법에서 전체 스케줄링 주기 초반 부분에서 상당히 많은 수의 제품이 외주화 되었음을 알 수 있다. 이것은 앞서의 재고수준 비교에서도 밝혔듯이 기존 기법의 경우 어떤 제품의 작업량 결정시에 그 제품의 당일 작업요구량에 생산용량 가변에 대응하기 위한 수량을 더하여 스케줄링 초기 단계부터 재고량을 최대로 확보한 뒤 재고가 감소하여 일정비율 이하가 되면 다시 최대 안전재고를 확보하는 방식으로 생산을 진행하기 때문이다. 이것은 그림 8에서 이 방식에 해당하는 곡선이 하루 지날때 마다 증가와 감소를 반복한다는 것을 보면 이 사실은 확인할 수 있다. 그러므로 각 작업장의 초기 작업부하가 상당히 크게 되며 따라서 외주량 역시 크게 증가하게 되는 것이다. 25%의 경우와 15%에서 초기 이틀간이 평탄하게 되는 것은 초반에 작업요구량이 지나치게 많아 최대 외주용량으로 할당을 하여도 전체 가용 생산용량이 작업부하량에 미달되어 생산이 안되고 다음날로 작업이 연기되어 다음날의 작업부하를 증가시키기 때문이다. crisp과 퍼지 알고리즘을 보면 우선 가변도가 작은 5%의 경우 1 주간 각 일별 외주량을 보면 crisp과 퍼지 알고리즘은 일별로 거의 평탄한 곡선이고 차이가 별로 크지 않음을 알 수 있다. 이는 시스템이 안정한 상태이기 때문에 이 경우 작업지시가 내려간 대로 생산이 진행될 가능성이 높으므로 앞서의 영향에 의한 차이는 적다는 의미이다. 그러나 15%인 경우를 보면 이때는 용량 가변도가 크편이고 그 결과 crisp 알고리즘은 고장에 의한 영향으로 외주량 곡선이 점차 증가함을 보이고 있다. 이에 비해 퍼지 알고리즘은 고장 등에 의해 축소될 가능성이 있는 생산용량 만큼의 제품을 미리 외주화 하므로 여전히 평탄한 형태의 곡선을 보임을 알 수 있다. 한편 25%인 경우를 보면 이때는 시스템의 상당히 불안한 경우이다. 이럴 경우에는 양쪽 방법 모두에서 고장 등에 의한 영향을 크게 받음을 알 수 있다. 그러나 crisp 알고리즘은 외주량이 극단적으로 증가하는데 반해 퍼지 알고리즘은 고장에 의한 영향을 사전에 고려하므로 crisp 알고리즘에 비해 낮은 외주량을 유지함을 알 수 있다. 이와같은 결과의 원인은 고장에 의하여 앞서의 주기에 처리되지 못한 작업 때문에 다음 주기의 작업 lot의 크기가 커

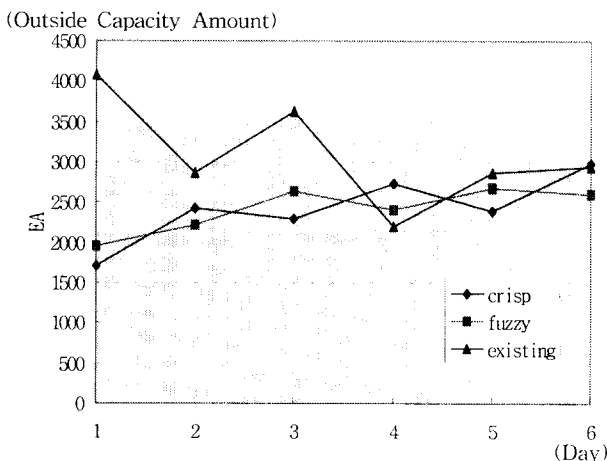


그림 8. 각 기법에 의한 외주화량 ($C'_{mt} = 0.05 \times C_{mt}$).
Fig. 8. Outside capacity amount allocated by each method ($C'_{mt} = 0.05 \times C_{mt}$).

지게 되고 이럴 경우 본 논문의 대상인 배치 생산방식의 공장에서는 작업 할당 시에 각 기계 간은 물론 공장내 협력업체간에 작업분배가 효과적으로 되지 못하게 되어 불필요하게 외주를 증가시키거나 균등한 자원 활용이 어려워지기 때문이다. 따라서 배치 생산방식의 공장에서는 가능하면 각 주기별로 지시된대로 생산이 진행되도록 하는 것이 가장 바람직함을 알 수 있다. 퍼지 알고리즘은 불확실한 요인들에 의한 가공용량의 축소를 어느정도 미리 감안하여 외주량을 결정 하므로 설비 고장 등에 의해 축소된 용량 만큼 공장에서 생산되지 못한 양을 협력업체를 통해 생산하게 되므로 당일 작업지시량을 어느정도 근접하게 생산하는 것이 가능하며 따라서 위와 같은 영향을 최소화할 수 있다.

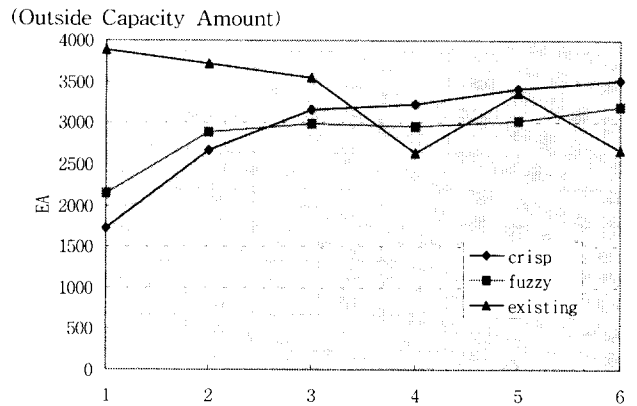


그림 9. 각 기법에 의한 외주화량 ($C'_{mt} = 0.15 \times C_{mt}$).
Fig. 9. Outside capacity amount allocated by each method ($C'_{mt} = 0.15 \times C_{mt}$).

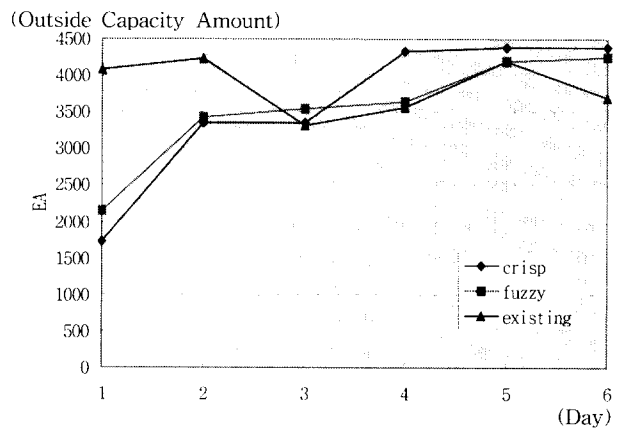


그림 10. 각 기법에 의한 외주량 ($C'_{mt} = 0.25 \times C_{mt}$).
Fig. 10. Outside capacity amount allocated by each method ($C'_{mt} = 0.25 \times C_{mt}$).

3.3 각 기법에 의한 납기 준수도 비교

그림 11부터 13에는 각 기법에 의한 작업지시 결과 주문 납기일을 지나 생산된 제품의 수를 보이고 있다. 먼저 crisp 알고리즘은 세 가지 경우 모두에서 가장 높은 작업지체도를 보이고 있는데 이는 대상 작업장의 생산 용량을 언제나 100%로 간주하여 생산용량의 변동을 고려하지 않았기 때문이다. 즉, crisp 알고리즘은 고장에 대한 고려가 전혀 없으므로 고장으로 인한 가공용량 축소시에 앞서 재고 수준 비교에서 밝혔듯이 반제품이나 완제품 재고가 있을 때는 이를 통해 가공용량 축소에 따른 결손을 보상하나 재고가 없을 경우에는 결손에 해당하는 만큼의 양이 차단이나 궁핍 현상에 의해 생산되지 못하게 되어 납기가

지연되는 것이다. 이와 같은 납기 지연은 crisp 알고리즘에 의한 스케줄링시에 발생하는 문제중 상당히 중요한 부분으로 실 시스템에 적용시에 큰 문제점이 된다.

기존 기법과 퍼지 알고리즘을 보면 생산용량 가변도가 15%(그림 12)에서 25%(그림 13)일 때는 퍼지 알고리즘이 약간 적으나 그 차이는 근소한 것이고 두 기법간에 큰 차이가 없음을 알 수 있다. 이것은 퍼지 알고리즘의 경우는 외주량을 늘려 미래에 일어날 설비고장 등에 대비하고 기존 기법은 높은 재고 수준을 유지하여(물론, 이 경우 외주도 상당히 늘어남을 앞서 결과에서 보았다.) 이에 대응하기 때문에 두 기법 모두 웬만한 생산 용량의 변화에도 생산이 끊김없이 진행되도록 하기 때문이다. 그러나 기존 기법에서는 재고수준을 지나치게 높이기 때문에 불필요한 비용을 증가시키게 된다. 특이한 것은 용량 가변도 5%(그림 11)에서 기존 기법이 가장 낮은 납기 지연도를 보인다는 것이다. 이것은 기존기법에 의해 유지되는 재고 수준이 높기 때문에 시스템이 거의 안정하다 할 수 있는 5% 정도의 용량가변도 에서는 크게 영향을 받지 않음을 의미하며 이 사실로도 기존 기법에 의한 재고 수준이 상당히 높다는 것을 간접적으로 알 수 있다.

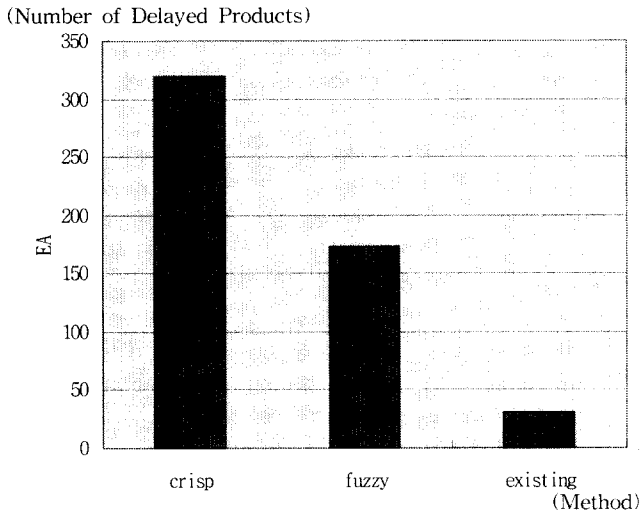


그림 11. 각 기법별 작업 지연량 ($C_{mt}^* = 0.05 \times C_{mt}$).
Fig. 11. The number of delayed products of each method ($C_{mt}^* = 0.05 \times C_{mt}$).

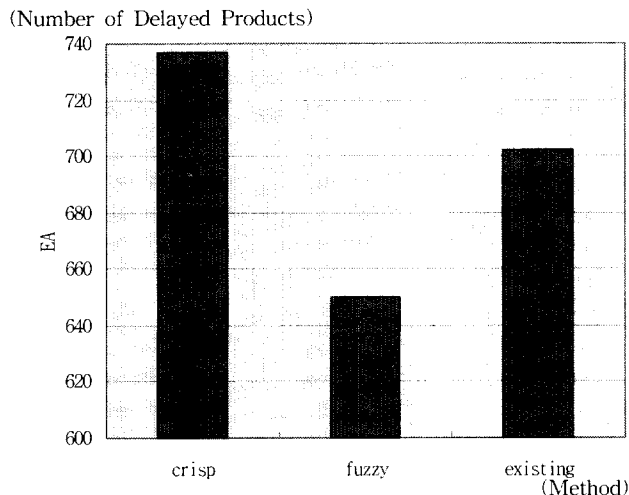


그림 12. 각 기법별 작업 지연량 ($C_{mt}^* = 0.15 \times C_{mt}$).
Fig. 12. The number of delayed product of each method ($C_{mt}^* = 0.15 \times C_{mt}$).

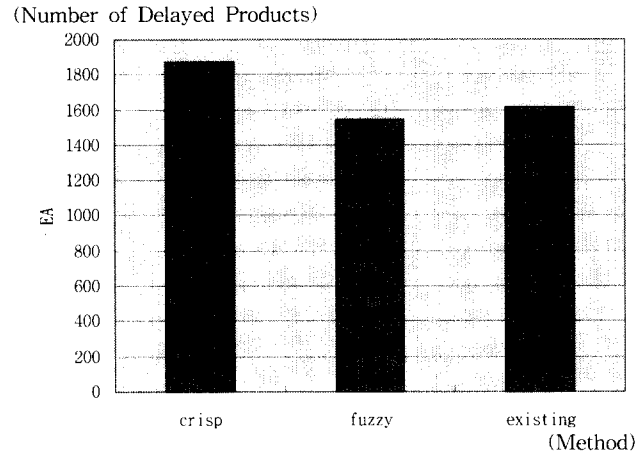


그림 13. 각 기법별 작업 지연량 ($C_{mt}^* = 0.25 \times C_{mt}$).
Fig. 13. The number of delayed products of each method ($C_{mt}^* = 0.25 \times C_{mt}$).

3.4 스케줄링 비용 비교

먼저 앞서의 지표 비교를 위해 시뮬레이션 수행시 사용된 비용 함수의 계수값은 다음과 같다. 재고 유지 비용 $h_{im}=0.1$, 납기 선행(earliness) 비용 $B_{it}=0.1$, 납기 초과(tardness) 비용 $A_{it}=5$, 외주 비용 $l_{mt}=5$. 재고 비용을 나타내는 계수값을 다른 계수값에 비해 상당히 작게 준 이유는 다음과 같다. 실제 대상 공장에서 비용이 발생하는 것은 주로 납기 초과와 외주 발생시로서 이때에는 실제로 비용의 지출이 발생하지만 재고 유지 비용이라는 것은 실질적 비용의 지출이라기 보다는 개념적인 성격이 더 강하기 때문이다. 표 4에 crisp 알고리즘과 기존 알고리즘에 대한 퍼지 알고리즘의 성능 개선도가 보여져 있다.

표 3. 퍼지 알고리즘의 성능 개선도.

Table 3. Performance improvement of the fuzzy algorithm.

비교대상기법 / 용량가변도	Crisp algorithm	Existing algorithm
5 %	0.9 %	26.2 %
15 %	3.2 %	15.3 %
25 %	2.8 %	9.0 %

표 3의 결과를 분석해 보면 먼저 퍼지 알고리즘과 crisp 알고리즘이 가변율 5%에서는 차이가 거의 없음을 보게되는데 이것은 변동율이 작아지게 되면 퍼지 알고리즘은 crisp 알고리즘과 동일한 것이 된다는($C_{mt}^*=0$ 이면 같은 것이된다.) 사실을 통해 이해할 수 있는 결과이다. 한편 그 이상에서는 대략 3% 정도의 성능향상을 보임을 알 수 있다. 한편 기존 기법과는 반대로 변동율이 작을수록 성능차가 크게되는데 이것은 기존 기법의 적용으로 인해 발생하는 외주량과 재고량이 변동율이 작아질 수록 지나치게 많다는 의미로 해석할 수있다.

이상의 결과들로부터 퍼지 알고리즘은 변동하는 생산환경하에서도 다른 기법들에 비해 대상 생산시스템에서의 운영목적을 가장 잘 만족 시키는 효과적인 방법임을 알 수 있다.

V. 결론

본 논문에서는 불확실한 생산환경 하에서도 시스템 운영목적이 최대한으로 만족 되도록 할 수 있는 근사 최적 스케

줄링 알고리즘을 개발하였다. 개발된 알고리즘은 생산시스템의 스케줄링 문제에서 불확실하게 정의되는 가공용량의 가변성에 대응하기 위해 가공용량에 관한 제약식을 퍼지화한 퍼지 최적화 기법을 기반으로 하였다. 또한 정의된 퍼지 최적화 문제의 효과적인 해결을 위해 Lagrangian relaxation 기법을 도입하여 복잡한 대상문제를 보다 해결이 용이한 수개의 서브 문제들로 분할하여 해결하는 근사적인 기법을 제시하였다. 제시된 알고리즘을 사출 성형 공장에 적용하여 실 생산 데이터를 이용한 시뮬레이션을 통해 타 기법들 즉, 제약식을 퍼지화하지 않은 crisp 알고리즘, 현재 공장에서 활용하고 있는 기존 스케줄링 기법들을 대상으로 공장에서의 각 공정별 가공용량의 가변도를 3단계로 변화 시키면서 성능비교를 수행하였으며 그 결과 제시된 퍼지 알고리즘이 타 기법들에 비해 우수함을 알 수 있었다. 이상의 결과에서 제시된 퍼지 알고리즘을 사용할 경우 가변하는 생산 환경에 적용한 생산 계획의 수립이 가능하며 가공용량의 가변성 이외에 다른 불확실한 요인들까지 감안한 알고리즘으로 발전될 경우 좀더 실용성 있는 스케줄링 시스템의 구현이 가능할 것이다.

부록

부록 A : 기존의 스케줄링 기법

1. 일반사출 단계에서의 일일생산량 결정과정

기본적으로 작업지시수량의 산정은 스케줄링 당일을 기준으로 D_0+2 일치의 재고를 유지하는 것을 기본목적으로 한다. 이러한 전제조건을 기본으로하여 다음의 규칙들을 이용해서 지시수량을 결정하게 된다.

$$PQTY_x = \max(PP_x(D_0), PP_x(D_1), PP_x(D_2)) - (INV_x - \sum_{i=0}^2 PP_x(D_i))$$

$PQTY_x$; 품종 x 의 작업 지시량,
 $PP_x(D_i)$; 품종 x 의 D_i 일의 출하계획량,
 $INV_x = D_0$ 일의 HMC재고+출하재고+반제품재고,

2. BMPR도장 단계에서의 일일생산량 결정과정

일반사출 단계에서는 과부족 수량에 최대 계획량을 더하여 수량을 계산하지만 BMPR도장 단계에서는 과부족수량에 안전재고 비율만을 곱하여 수량을 결정한다.

$$PQTY_x = |INV_x - \sum_{i=0}^2 PP_x(D_i)| \times (1 + \gamma)$$

γ ; 안전재고 비율

3. 일반사출 2차가공 및 BMPR조립 단계에서의 일일생산량 결정과정

일반사출 2차 가공단계에서는 재고비율을 계산할때 일반사출이나 도장단계에서와는 달리 D_1 을 기준으로 재고비율을 계산하고 작업지시수량도 과부족 수량 만큼만을 지시한다.

$$PQTY_x = |INV_x - \sum_{i=0}^1 PP_x(D_i)|$$

참고문헌

[1] C. Sriskandarajah and S. P. Sethi, "Scheduling algorithm for flexible flow shops : worst and average case performance," *European J. of Operations Research*, vol. 43, pp. 143-160, 1989.
 [2] D. P. Bertsekas and P. Tseng, "Relaxation methods for minimum cost ordinary and

generalized network flow problems," *Operations Research*, vol. 36, no. 1, pp. 93-114, 1988.
 [3] M. A. H. Dempster, J. K. Lenstra and A. H. G. Rinnooy Kan, *Deterministic Stochastic Scheduling*, D. Riedel Publishing Company, 1982.
 [4] D. J. Hoinmt, P. B. Luh and K. R. Pattipati, "A practical approach to job-shop scheduling problems," *IEEE Trans. Robotics & Automation*, vol. 9, no. 1, Feb., 1993.
 [5] D. G. Luenberger, *Linear and Nonlinear Programming*, Second Edition, Addison Wesley, 1984.
 [6] M. S. Fox, "Fob shop scheduling : An investigation into constraint directed reasoning," Ph.D. Thesis, CMU, 1982.
 [7] H. Ishii, M. Tada and T. Masuda, "Two scheduling problems with fuzzy due dates," *Fuzzy Sets and Systems*, vol. 46, pp. 339-347, 1992.
 [8] H. J. Zimmermann, *Fuzzy Sets, Decision Making, and Expert Systems*, Kluwer Academic Publishers, 1987.
 [9] G. W. Hintz and H. J. Zimmermann, "A method to control flexible manufacturing systems," *European J. of Operations Research*, vol. 41, pp. 321-334, 1989.
 [10] I. B. Turksen, D. Ulguray and Q. Wang, "Hierarchical scheduling based on approximate reasoning - A comparison with ISIS," *Fuzzy Sets and Systems*, vol. 46, pp. 349-371, 1992.
 [11] J. K. Lenstra, A. H. G. Rinnooy Kan, P. Brucker, "Complexity of machine scheduling problem," *Annals Discrete Math*, I, pp. 343-362, 1977.
 [12] M. Klein, "A primal method for minimal cost flow," *Management Science*, vol. 14, no. 3, pp. 205-220, 1967.
 [13] L. M. M. Custodio, J. J. S. Sentieiro and C. F. G. Bispo, "Production planning and scheduling using a fuzzy decision system," *IEEE Trans. Robotics and Automation*, vol. 10, no. 2, pp. 160-167, 1994.
 [14] M. S. Salvador, "A solution of a special class of flow shop scheduling problems," *Proc. of the Symposium on Theory of Scheduling and Its Application. Berlin: Splinger-Verlag*, pp. 83-91, 1973.
 [15] M. L. Fisher, "The Lagrangian relaxation method for solving integer programming problems," *Management Science*, vol. 27, no. 1, pp. 1-18, 1981.
 [16] M. Heid, P. Wolfe and H. P. Crowdw, "Validation of subgradient optimization," *Mathematical Programming*, vol. 6, pp. 62-88, 1974.
 [17] P. C. Fetterolf and G. Anandalingam, "A Lagrangian relaxation technique for optimizing interconnection of local area networks," *Operations Research*, vol. 40, no. 4, pp. 678-688, 1992.
 [18] R. J. Wittrock, "An adaptable scheduling algorithm for flexible flow lines," *Operation Research*, vol. 36, pp. 445-453, 1985.

- [19] R. J. Wittrock, "Scheduling algorithms for flexible flow lines," *Research Development*, vol. 29, no. 4, pp. 401-412, 1985.
- [20] S. Kochhar and R. J. T. Morris, "Heuristic methods for flexible flow lines scheduling," *J. of Manufacturing Systems*, vol. 6, no. 4, pp. 299-314, 1987.
- [21] Y.-J. Lai and C.-L. Hwang, "Interactive fuzzy linear programming," *Fuzzy Sets and Systems*, vol. 45, pp. 169-183, 1992.
- [22] Y. C. Ho, *Discrete Event Dynamic Systems Analyzing Complexity and Performance in the Modern World*, IEEE Press, pp. 249-263, 1991.

박 승 규

1989년 연세대 전기과 졸업. 동대학원 석사(1991). 동대학원 박사(1996). 현재 삼성SDS 선임연구원, 관심분야 생산시스템, 스케줄링, 인공지능.

이 창 훈

1982년 연세대 전기과 졸업. 동대학원 석사(1986). 동대학원 박사(1989). 현재 한라공대 전기공학과 교수. 관심분야 인공지능, 생산시스템, 최적제어.

장 석 호

1988년 연세대 전기과 졸업. 동대학원 석사(1990). 1997-현재 연세대학교 전기공학과 박사과정. 관심분야 반도체 공정 자동화, 모델링, 자동제어.

우 광 방

1934년 1월 25일생. 1957년 연세대 전기공학과졸(석사). 1962년 및 1964년 오레곤 주립대학전기공학과에서 석사 및 박사학위 취득, 1966년~1971년 워싱턴대 전기공학과 조교수, 1971년 미국립 암연구소 책임연구원, 1980년 재미한국인과학기술자협회장, 1982년~현재 연세대학교 전기공학과 교수, 1992년~현재 연세대학교 자동화기술연구소 소장, 1995년 당학회 회장. 주요관심분야는 반도체공정 자동화 및 스케줄링, 인공지능제어기법, FMS.