

공적 정보하에서 단일 설비의 다중 에이전트 스케줄링

이용규 · 최유성 · 정인재[†]

한양대학교 산업공학과

Multiagent Scheduling of a Single Machine Under Public Information

Yong-kyu Lee · Yooseong Choi · In-Jae Jeong[†]

Dept. of Industrial Engineering, Hanyang University

This paper considers a multiagent scheduling problem under public information where a machine is shared by multiple agents. Each agent has a local objective among the minimization of total completion time and the minimization of maximum. In this problem, it is assumed that scheduling information is public. Therefore an agent can access to complete information of other agents and pursue efficient schedules in a centralized manner

We propose an enumeration scheme to find Pareto optimal schedules and a multiobjective genetic algorithm as a heuristic approach. Experimental results indicate that the proposed genetic algorithm yields close-to Pareto optimal solution under a variety of experimental conditions.

Keywords : Multiagent Scheduling, Public Information, Genetic Algorithm

1. 서 론

본 연구에서는 모든 작업의 시작 가능 시간이 0이고 preemption이 허용되지 않는 상황에서 m 개의 작업의 집합, J^1, J^2, \dots, J^m 을 단일설비 상에 스케줄링 하는 문제를 다루고 있다. 각각의 집합, $J^i, i=1, 2, \dots, m$ 에 대하여 J^i 를 단일 설비상에 가공하여 어떠한 목적 함수 f^i 를 추구하려고 하는 에이전트 i 가 존재한다. Agnetis (2004)등이 제안한 분류방식에 따르면, 단일 설비의 다중에이전트 문제는 $||f^1 \circ f^2 \circ \dots \circ f^m$ 로 표현될 수 있다. 다중에이전트 문제는 다목적 함수를 고려한다는 측면에서는 다목적 스케줄링 문제라고 볼 수 있다. 그러나 다중에이전트 문제가 다목적 스케줄링과 다른점은 다목적 스케줄링에서는 모든 작업이 모든 다목적 함수에 영향을 미치지만 다중에이전트 문제에서는 일부의 작업만

하나의 목적식에 영향을 미친다는 점이다(Agnetis et al.(2004)). 예를 들면 작업 집합, J^i 는 오직 목적함수 f^i 에 만 영향을 미치는 경우이다.

단일 설비 다중에이전트 스케줄링 문제는 두 가지의 경우로 분류될 수 있다. 하나는 공적 정보하의 단일 설비 다중에이전트 스케줄링 문제(single machine multiagent scheduling problem under public information SMPU) 즉 $||\text{pub.}| f^1 \circ f^2 \circ \dots \circ f^m$ 와 다른 하나는 사적정보하의 단일 설비 다중에이전트 스케줄링 문제(single machine multiagent scheduling problem under private information SMPR), $||\text{pri.}| f^1 \circ f^2 \circ \dots \circ f^m$ 이다.

SMPU에서는 스케줄링 정보가 공적인 정보임을 가정한다. 즉, 모든 에이전트가 다른 에이전트의 정보(가공 시간, 목적함수)에 접근이 가능하다. 반면 SMPR에서는 스케줄링 정보가 사적인 정보로 J^i 와 f^i 의 정보는 오

논문접수일 : 2008년 08월 21일 논문수정일 : 2008년 09월 05일 게재확정일 : 2008년 09월 07일

[†] 교신저자 ijeong@hanyang.ac.kr

※ 본 논문은 2006년도 교육인적자원부 학술연구조성사업비으로 한국학술진흥재단의 지원을 받아 연구되었음(KRF-2006-003-D00674).

로지 에이전트 i 에게만 알려져 있다. SMPR 환경하에서 에이전트들은 독자성을 최대한 유지하면서 합축적이거나 제한된 정보의 교환을 통해 파레토 최적해를 찾기 위해 서로 협력한다. SMPR은 하나의 자원을 서로 다른 조직들이 공유하고 있는 가상기업(virtual enterprise VE) 상황에서 쉽게 발견된다. 가상기업은 공통된 목적을 추구하기 위해 서로 협력하기로 합의한 독립적인 기업들의 모임이나 파트너십이 일시적인 것이기 때문에 사적인 기업정보를 완전히 공유한다는 것을 가정하기 어렵다. 본 연구에서는 두명의 에이전트 $i, i = 1, 2$ 가 작업 집합 J' 에 대해 지역적 목적함수 f^i 와 작업선후행 제약을 가지는 SMPU 즉, $1|prec., pub.| f^1, f^2$ 를 다루고자 한다. 가능한 목적함수로는 총작업 완료 시간의 최소화, 최대 tardiness의 최소화를 고려한다. 또한 이 문제의 최적 파레토 해를 찾기 위한 enumeration 방법과 다목적 유전알고리즘을 제안한다.

제 2장에서는 기존연구에 대해 소개하고 제 3장에서는 SMPU에 대해 정의한다. 제 4장에서는 알고리즘을 제안하고 제 5장에서 실험결과를 보이고 제 6장에서 결론을 맺는다.

2. 기존 연구

기존의 관련 연구는 대부분이 단일 설비의 다목적 스케줄링에 집중되어 있었다(Chen and Bulgin, 1993, Nagar et al., 1995, Fry et al., 1989). 본 연구에서 다루는 것은 다목적 스케줄링이 아니고 다중에이전트 스케줄링이므로 이와 관련된 기존 연구에 대해서만 알아보도록 하겠다.

Agnētis et al.(2000)는 공적 정보하에서 두 에이전트가 jobshop 설비를 공유하는 상황에서의 스케줄링 문제를 다루었다. Agnētis et al.(2004)는 다양한 성능 척도에 대해 단일 설비에서의 두 명의 에이전트가 존재하는 경우의 complexity를 분석하였다. Babayan and He(2004)는 n-job 3-stage flexible flowshop 스케줄링 문제에 대해 연구하였다.

사유정보하의 다중에이전트 스케줄링에서 가장 중요한 연구는 분산 다이나믹 스케줄링이다. 분산 다이나믹 스케줄링은 설비 고장과 같이 다이나믹하게 변화하는 환경에 잘 적응하는 특성을 가지고 있다. 또한 독자적인 에이전트의 정보의 encapsulation과 에이전트 간의 부분 정보 공유와 같은 중요한 특성을 가지고 있다. Davis and Smith(1983)는 “contract net”이라고 불리우는 프로토콜을 제안하였다. 이 프로토콜은 Shaw(1988), Lin and Solberg (1992), Duffie and Prabhu(1994), Dewan and Joshi(2000)에

의해서 다이나믹 스케줄링에 적용되었는데 작업 에이전트, 설비 에이전트들이 특정한 목적을 달성하기 위해 bidding을 하는 프로세스로 이루어져 있다. 그러나 대부분의 분산 다이나믹 스케줄링에서는 에이전트들이 동종 목적식(homogeneous objective)을 가지고 있음을 가정한다. 예를 들면 Duffie and Prabhu(1994)는 납기 위배 최소화, Dewan and Joshi(2000)는 가중 완료시간 최소화를 고려하였다. Wu et al.(2005)는 flexible job shop에서 earliness와 tardiness를 최소화하는 문제를 고려하였다.

제약탐색기법(constraint-guided heuristic search; CHS)은 정해진 규칙과 제약을 만족하는 하나의 가능해를 찾기 위한 탐색기법이다(Pinedo, 1995). Sycara et al.(1991)은 분산된 형태의 제약탐색기법을 제안하였는데 설비 에이전트들의 코디네이션에 의해 설비 사용에 있어서 발생하는 상충되는 목적을 조정하였다.

3. 공적 정보하의 단일 설비 다중에이전트 스케줄링 문제

이 장에서는 공적 정보하의 단일 설비 다중에이전트 스케줄링 문제(single machine multiagent scheduling problem under public information SMPU), SMPU문제를 정형화 하고 파레토 최적해를 찾기 위한 enumeration방법과 다목적 유전알고리즘 등 두가지 중앙통제식 알고리즘을 제안한다. 문제의 정형화에 필요한 기호와 수식은 아래와 같다.

m : 단일 설비를 공유하는 에이전트의 수.

$J = \cup_i J^i$: 모든 작업의 집합

$|J^i| = n_i$: 에이전트 i 에 속하는 작업의 수 $i = 1, 2, \dots, m$

$N = |J| = \sum_{i=1}^m n_i$: 시스템에 있는 전체 작업의 수

j : 작업 인덱스, $j = 1, \dots, N$

p_j : 작업 j 의 가공시간

$T = \sum_{j=1}^N p_j$: planning horizon

$x_{jt} = \begin{cases} 1, & \text{만약 } j \text{가 시간 } t \text{이전에 시작되었으면, } t=1, \dots, T. \\ 0, & \text{그렇지 않으면.} \end{cases}$

$s_j = \sum_{t=1}^T (1 - x_{jt})$: 작업 j 의 시작시간

$C_j = s_j + p_j = \sum_{t=1}^T (1 - x_{jt}) + p_j$: 작업 j 의 완료시간

$T_j = \max(C_j - d_j, 0) = \max\left(\sum_{t=1}^T (1 - x_{jt}) + p_j - d_j, 0\right)$:

작업 j 의 tardiness

$$\sum_j C_j = \sum_j \left(\sum_{t=1}^T (1-x_{jt}) + p_j \right) : \text{총 작업 완료시간의 합}$$

$$T_{\max} = \max_j \left\{ \max \left(\sum_{t=1}^T (1-x_{jt}) + p_j - d_j, 0 \right) \right\} : \text{최대 tardiness}$$

1|prec., pub.| $f^1 \circ f^2 \circ \dots \circ f^m$ 문제는 아래와 같이 정형화 될 수 있다.

$$\text{Min } (f^1, f^2, \dots, f^m) \quad (1)$$

$$\text{St. } x_{jt+1} \geq x_{jt} \quad \forall j \in J, t = 1, \dots, T-1, \quad (2)$$

$$\sum_{t=1}^{T-p_j+1} x_{jt} \geq 1 \quad \forall j \in J, \quad (3)$$

$$\sum_j (x_{jt} - x_{j-t-p_j}) \leq 1 \quad \forall t \quad (4)$$

$$\sum_{r=1}^T (x_{qr} - x_{rr}) \geq p_q \quad \forall (q, r) \in J, \quad (5)$$

$$x_{jt} = \{0, 1\} \quad \forall j \in J, \forall t. \quad (6)$$

위의 수리모형은 목적식이 (1)에 나타난 바와 같이 다목적적인 것을 제외하면 Jeong and Leon(2005)에서 제안한 것과 같다. 제약 (2)는 작업이 일단 시작되면 전체 계획구간 동안 항상 1이어야 함을 의미한다. 제약 (3)은 모든 작업이 계획구간 안에 완료되어야 함을 의미한다. 제약 (4)는 설비의 capa. 제약을 의미한다. 만약 $x_{jt} - x_{j-t-p_j} = 1$ 이면, 작업 j 는 시간 t 에서 작업중이고 반면에 $x_{jt} - x_{j-t-p_j} = 0$ 이면, 작업은 아직 시작하지 않았거나 작업이 이미 완료된 상태를 의미한다. 그러므로 어떠한 특정한 시점에 가공중인 작업의 수는 하나를 초과할 수 없다. 제약 (5)는 작업 q 와 r 사이의 선행행 관계를 나타낸다. 만약 작업 q 가 작업 r 을 선행하면 이러한 선행행 관계는 (q, r) 로 표현된다. 부등식의 좌측은 job와 의 시작시간의 간격을 나타내고, 이 간격은 선행 가공되어야 할 job의 가공 시간보다 커야 한다. 선행 제약을 가진 단일 설비 문제는 잘 알려진 NP-hard문제이다(Pinedo, 1995).

Agnets et al.(2004)는 두 명의 에이전트로 이루어진 SMPU 문제에서 모든 파레토 최적해를 찾아내는 방법을 제안하였다. 이 방법에서 다른 모든 목적식이 k 를 만족한다는 가정하에 하나의 목적식을 최적화하는 스케줄을 찾는다. 이후에 k 를 조금씩 줄여서 모든 파레토 최적해를 찾아나가는 방법이다. 본 연구에서는 이 방법론을 1|prec., pub.| $f^1 \circ f^2$ 에 대하여 아래와 같이 적용하였다.

3.1 1|prec., pub.| $f^1 \circ f^2$ 의 파레토 최적해 생성 방법

Step 1 : S 는 파레토 최적해의 집합이고 M 은 큰 정수

라고 하자. 또한 k 는 f^2 의 상한이라고 하자. 이때 $S = \emptyset, k = M$ 로 설정한다.

Step 2 : 다음의 단일 목적함수를 가지는 스케줄링 문제를 풀고 최적 스케줄을 σ 라고 하자.

$$\begin{aligned} &\text{Min } f^1 \\ &\text{St. } f^2 \leq k \end{aligned} \quad (7)$$

(2), (3), (4), (5), (6)

Step 3 : 만약 σ 가 가능해이고 S 의 어떠한 스케줄에 의해서도 dominate되지 않는다면 $S = S \cup \sigma$ 로 하자.

Step 4 : 만약 $k \neq 0$ 이면 $k = k-1$ 로 하고 step 2로 가라. 그렇지 않으면 멈춘다.

step 2에 나타난 단일 설비문제는 그 자체가 NP-hard 이므로 큰 크기의 문제에 대해서 모든 파레토 최적해를 찾으려면 시간이 많이 소요될 수 있다. 그러므로 두 명의 에이전트 이상의 문제에 대해서는 다목적 유전 알고리즘을 다음과 같이 제안한다.

3.2 다목적 유전 알고리즘

$\{j[1], j[2], \dots, j[N]\}$ 를 염색체라 하고 각각의 유전자를 단일설비 상의 작업을 의미한다고 하자.

Step 1 : (초기화) N_{pop} 개의 개체로 이루어진 초기 모집단을 랜덤하게 생성하라.

Step 2 : (평가) 각각의 개체에 대해 m 개의 목적 함수를 평가하라.

Step 3 : (파레토 랭킹 선별(Goldberg, 1989))

Step 3.1 : 모집단에서 비지배해에 1순위를 부여하고 고려대상에서 제외하라. 나머지 개체들 중에서 다시 비지배해를 찾아 2순위를 부여하고 이를 계속 반복하라.

Step 3.2 : r 순위의 개체 선별 확률을 $\frac{q(1-q)^{r-1}}{\sum_r q(1-q)^{r-1}}$

where $0 < q < 1$ 로 하여 N_{pop} 개의 모집단을 생성하라.

Step 4 : (순서 교차)

Step 4.1 : 교차 확률 pc 로 두 개의 부모를 랜덤하게 선택하라. 하나의 부모로부터 유전자의 일부를 자손에게 상속하라.

Step 4.2 : 두 번째 부모의 염색체에서 자손에게 상속된 유전자를 삭제하라.

Step 4.3 : 두 번째 부모의 삭제되지 않은 유전자를 왼쪽에서 오른쪽 순서로 자손의 고정되지 않은 유전자 위치에 놓는다.

Step 5 : (역교환 돌연변이) p_m 의 확률로 두 개의 유전자를 랜덤하게 선택하고 서로의 위치를 교환한다.

Step 6 : (종료 조건) 만약 N_{gen} 의 세대수가 되면 종료하고 그렇지 않으면 step 2로 가라.

파레토 최적해 생성 알고리즘과 다목적 유전 알고리즘은 모든 에이전트의 정보가 하나의 의사결정자에게 알려져 있어야 한다는 측면에서 중앙통제식 알고리즘이다.

4. 실험

이 장에서는 알고리즘의 성능을 평가할 성능 척도를 제안하고 예제를 보여주려고 한다. 마지막으로 실험 결과와 분석에 대해 설명한다.

4.1 성능척도

S_k^{Method} , $k=1, \dots, u_{Method}$ 를 Method를 사용해서 구한 k th 파레토 해라고 하자. 여기서 Method는 최적해(OPT), 유전알고리즘의 해(GA)가 될 수 있다. u_{Method} 는 Method를 사용하여 찾은 파레토 해의 수를 나타낸다. Z_k^{Method} (Z_k^{Method} , Z_{2k}^{Method})는 S_k^{Method} 의 목적함수 값을 나타내는데 여기서 Z_{1k}^{Method} 는 Method를 사용하여 구한 파레토 해의 총완료시간을 나타내고, Z_{2k}^{Method} 는 Method를 사용하여 구한 파레토 해의 최대 tardiness를 나타낸다

S_k^{OPT} 의 성능을 GA로 구한 해와 비교 할 때 먼저 S_k^{OPT} 에서 Euclidean 거리가 가장 가까운 GA의 해를 S_k^{GA} 라 하면, S_k^{OPT} 의 i th 목적함수에 대한 Solution Error(SE)는 $\frac{\max(Z_{ik}^{OPT} - Z_{ik}^{GA}, 0)}{\max(Z_{ik}^{OPT}, Z_{ik}^{GA})} \times 100$ 으로 계산된다 $i=1, 2$. 그러므로 OPT의 GA에 대한 SE는 평균 편차로 아래와 같이 계산된다.

$$SE(OPT, GA) = (SE_1, SE_2)$$

$$= \left(\frac{\sum_{k=1}^{U_M} \left[\frac{\max(Z_{1k}^{OPT} - Z_{1k}^{GA}, 0)}{\max(Z_{1k}^{OPT}, Z_{1k}^{GA})} \times 100 \right]}{U_M}, \frac{\sum_{k=1}^{U_M} \left[\frac{\max(Z_{2k}^{OPT} - Z_{2k}^{GA}, 0)}{\max(Z_{2k}^{OPT}, Z_{2k}^{GA})} \times 100 \right]}{U_M} \right)$$

4.2 실험 결과

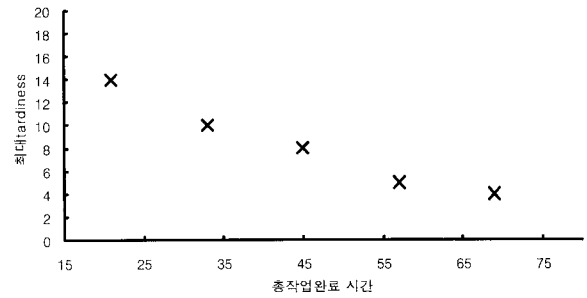
예제

두 명의 에이전트가 하나의 설비를 공유하고 있고 총 완료 시간의 최소화 (f1)와 최대 tardiness의 최소화 (f2)를 목적함수로 가지는 문제를 고려하자. 문제의 데이터는 아래의 <표 1>에 주어져 있다.

<표 1> 예제의 데이터

	$i=1$	$i=2$
n_i	4	4
p_j	(1, 2, 3, 4)	(4, 2, 5, 1)
d_j	(4, 13, 12, 10)	(4, 6, 7, 8)
(q, r)	(1, 4), (3, 2)	(1, 2), (3, 4)

Enumeration scheme은 아래의 <그림 1>에 나타난 바와 같이 4개의 최적 파레토 해(f1, f2) = {(21, 14), (33, 10), (45, 8), (57, 5), (69, 4)}를 찾았고 유전알고리즘도 동일한 4개의 파레토 해를 찾았다.



<그림 1> 예제의 파레토 최적해

<표 2> 실험에서 고려한 문제유형

문제 유형	m	n_i	선행행 제약수	p_j	d_j
1	2	2	$U(1, \frac{n_i}{2})$	$U(1, 5)$	$U(p_j, \sum_j p_j)$
2	2	4			
3	2	6			

<표 2>에 나타난 바와 같이 실험에서는 에이전트의 수와 작업의 수 그리고 목적함수의 종류에 따라 총 10가지 종류의 문제를 고려한다. 예를 들어 문제유형1은 두 명의 에이전트가 존재하고 각각은 총 작업 완료시간 최소화 와 최대 tardiness 최소화의 목적함수를 가지는 스

케줄링 문제이다.

또한 두 명의 에이전트는 각각 두 개의 작업을 가지고 있다. 에이전트 i 가 가지는 선후행 제약의 수는 $U(1, \frac{n_i}{2})$ 으로부터 랜덤하게 발생하였고, 작업 j 의 작업시간은 $U(1, 5)$, 납기일은 $U(p_j, \sum_j p_j)$ 에서 랜덤하게 발생하였다. 각각의 문제 유형에 대해 20개의 랜덤한 문제를 발생시켜 문제를 풀었다.

파일럿 테스트를 통해서 유전알고리즘의 모수를 다음과 같이 설정하였다. 최대 세대수, $N_{gen} = 100$, 모집단 크기, $N_{pop} = 100$, 선별확률, $q = 0.4$, 교차율, $p_c = 0.6$, 돌연변이율, $p_m = 0.01$. 모든 알고리즘은 C++로 코딩되었고 CPU-3.0, 512 RAM의 PC에서 실행되었다.

<표 3>은 다목적 GA와 OPT를 비교한 결과가 나와 있다. SE1과 SE2는 각각 총 작업 완료시간과 최대tardiness의 solution error를 나타낸다. 결과적으로 GA가 최적 파레토 해에 근접한 것으로 보인다. 왜냐하면 최대 SE

가 5.2%에 불과하기 때문이다.

<표 4>에는 OPT, GA의 평균 계산시간이 나타나 있는데 문제의 크기가 커질수록 OPT가 GA에 비해 파레토 해를 찾는 데 시간이 많이 걸리는 것으로 보인다.

5. 결론

본 연구에서는 공적인 정보의 공유가 가능한 단일설비에서의 다중에이전트 스케줄링문제에서 최적 파레토 해를 찾기 위한 방법과 다목적 유전알고리즘을 제안하였다. 이 연구 후속 연구로 사적 정보상황하의 단일설비 다중에이전트 문제를 고려해 볼 수 있다. 이 연구의 경우 에이전트간의 협업체계 설계를 통해 전체 최적화를 위해서는 어떠한 정보를 공유하고 있어야 하고, 어떠한 방법으로 협력을 해야 하는지 정의 한다. 또한 협업 체계 하에서의 의사결정 문제를 수리적으로 모델링

<표 3> 문제유형 1, 2, 3의 경우 OPT와 비교한 GA의 SE

문제유형 1	SE1	SE2	문제유형 2	SE1	SE2	문제유형 3	SE1	SE2
1	0.0	0.0	1	0.0%	0.0%	1	0.7%	3.4%
2	0.0	0.0	2	0.0%	0.0%	2	0.0%	1.3%
3	0.0	0.0	3	0.0%	0.0%	3	0.0%	0.7%
4	0.0	0.0	4	0.0%	0.0%	4	0.4%	2.7%
5	0.0	0.0	5	0.0%	0.0%	5	0.0%	0.9%
6	0.0	0.0	6	0.0%	0.0%	6	1.0%	3.8%
7	0.0	0.0	7	0.0%	2.0%	7	0.5%	0.6%
8	0.0	0.0	8	0.0%	0.0%	8	0.0%	1.2%
9	0.0	0.0	9	0.0%	0.0%	9	2.3%	5.2%
10	0.0	0.0	10	0.0%	0.0%	10	0.0%	0.0%
11	0.0	0.0	11	0.0%	0.0%	11	0.0%	0.4%
12	0.0	0.0	12	0.0%	0.0%	12	0.7%	0.9%
13	0.0	0.0	13	0.0%	0.0%	13	0.2%	1.4%
14	0.0	0.0	14	0.0%	0.0%	14	0.0%	1.9%
15	0.0	0.0	15	0.0%	0.0%	15	0.0%	0.0%
16	0.0	0.0	16	0.0%	0.0%	16	0.2%	0.0%
17	0.0	0.0	17	0.5%	0.0%	17	0.3%	2.3%
18	0.0	0.0	18	0.0%	0.0%	18	0.4%	0.6%
19	0.0	0.0	19	0.0%	0.0%	19	0.1%	0.7%
20	0.0	0.0	20	0.0%	0.0%	20	0.3%	0.7%
min	0.0	0.0	min	0.0%	0.0%	min	0.0%	0.0%
max	0.0	0.0	max	0.5%	2.0%	max	2.3%	5.2%
avg	0.0	0.0	avg	0.0%	0.1%	avg	0.4%	1.4%

<표 4> 평균 계산시간(초)

문제유형 1	OPT	GA	문제유형 2	OPT	GA	문제유형 3	OPT	GA
1	0.42	0.61	1	0.89	0.72	1	1.30	0.95
2	0.11	0.58	2	0.53	0.69	2	5.97	0.91
3	0.13	0.55	3	0.48	0.74	3	2.02	0.98
4	0.19	0.61	4	0.75	1.02	4	3.17	0.98
5	0.11	0.56	5	0.95	1.11	5	5.88	0.89
6	0.14	0.56	6	0.69	0.77	6	6.44	0.92
7	0.11	0.58	7	0.77	0.72	7	9.17	0.89
8	0.11	0.56	8	0.83	0.72	8	3.41	0.84
9	0.13	0.56	9	0.41	0.70	9	2.81	0.94
10	0.13	0.56	10	0.44	0.72	10	1.75	0.86
11	0.11	0.56	11	0.41	0.73	11	1.94	0.89
12	0.13	0.56	12	0.48	0.72	12	1.38	0.88
13	0.14	0.56	13	0.58	0.77	13	5.28	0.88
14	0.14	0.56	14	0.66	0.74	14	1.23	0.88
15	0.13	0.56	15	0.53	0.77	15	0.92	0.84
16	0.13	0.58	16	0.59	0.74	16	3.11	0.89
17	0.13	0.58	17	0.67	0.81	17	2.83	0.88
18	0.11	0.56	18	0.42	0.77	18	3.27	0.95
19	0.13	0.56	19	0.72	0.80	19	3.61	0.88
20	0.13	0.56	20	1.08	0.75	20	2.52	0.94
min	0.11	0.55	min	0.41	0.69	min	0.92	0.84
max	0.42	0.61	max	1.08	1.11	max	9.17	0.98
avg	0.14	0.57	avg	0.64	0.77	avg	3.40	0.90

하고 해법을 제시할 필요가 있다.

참고문헌

[1] Agnetis A., Mirchandani P. B., Pacciarelli D., and Pacifici A.; "Nondominated schedules for a Job-shop with two competing users," *Computational and Mathematical Organization Theory*, 6(2) : 191-217, 2000.

[2] Agnetis A., Mirchandani P. B., Pacciarelli D., and Pacifici A.; "Scheduling problems with two competing Agents," *Operations Research*, 52(2) : 229-242, 2004.

[3] Babayan A. and He D.; "Solving the n-job 3-stage flexible flowshop scheduling problem using an agent-based approach," *International Journal of Production Research*, 42(4) : 779-799, 2004.

[4] Chen C. L. and Bulfin B. L.; "Complexity of single machine, multi-criteria scheduling problems," *European Journal of Operational Research*, 70 : 115-125, 1993.

[5] Davis R. and Smith R. G.; "Negotiation as a metaphor for distributed problem solving," *Artificial Intelligence*, 20 : 63-109, 1983.

[6] Dewan P., Joshi S., Pacciarelli D., and Pacifici A.; "Dynamic single-machine scheduling under distributed decision-making," *International Journal of Production Research*, 38(16) : 3759-3777, 2000.

[7] Duffie N. A. and Prabhu V. V.; "Real-time distributed scheduling of heterarchical manufacturing systems," *Journal of Manufacturing Systems*, 13(2) : 94-107, 1994.

[8] Fry T., Armstrong A., and Lewis H.; "A framework for single machine multiple objective sequencing research," *OMEGA International Journal of Management Science*, 17(4) : 595-607, 1989.

[9] Goldberg D. E.; "Genetic Algorithms in Search, Optimization, and Machine Learning," Addison-wesley publishing company, inc, 1989.

[10] Kutanoglu E., Wu S. D., Pacciarelli D., and Pacifici A.; "On combinatorial auction and Lagrangian relaxation for distributed resource scheduling," *IIE Transactions*, 31(9)

- : 813-826, 1999.
- [11] Jeong I. J., Leon V. J., Pacciarelli D., and Pacifici A.; "Decision making and cooperative interaction via coupling agents in organizationally distributed system," *IIE Transactions*, 34 : 789-802, 2002.
- [12] Jeong I. J. and Leon V. J.; "A single-machine distributed scheduling methodology using cooperative interaction via coupling agents," *IIE Transactions*, 37 : 137-152, 2005.
- [13] Lin G. Y. and Solberg J. J.; "Integrated shop floor control using autonomous agents," *IIE Transactions*, 24(3) : 57-71, 1992.
- [14] Nagar A., Haddock J., and Heragu S.; "Multiple and bi-criteria scheduling : A literature survey," *European Journal of Operational Research*, 81 : 88-104, 1995.
- [15] Pinedo M.; "Scheduling : Theory, algorithms and systems," Prentice Hall, Englewood Cliffs, NJ, 1995.
- [16] Shaw M. J.; "Dynamic scheduling in cellular manufacturing systems : A framework for networked decision making," *Journal of Manufacturing system*, 7(2) : 83-94, 1988.
- [17] Sycara K., Roth S., Sadeh N., and Fox M.; "Distributed constrained heuristic search," *IEEE Transaction on Systems Man and Cybernetics*, 21(6) : 1446-1461, 1991.
- [18] Wu, Z. and Weng M. X.; "Multiagent scheduling method with earliness and tardiness objective in flexible job shops," *IEEE Transactions on Systems, Man and Cybernetics-Part B : Cybernetics*, 35(2) : 293-301, 2005.