

수송 및 준비 시간을 고려한 다제품 회분식 공정의 최적 생산 방안을 위한 혼합 정수 선형 모델

변승헌 · 이해평 · 김경숙* · 류경옥†

한양대학교 공과대학 화학공학과

*동양공업전문대학 화학공학과

(1997년 5월 12일 접수, 1997년 10월 4일 채택)

Mixed Integer Linear Model for the Scheduling of Multiproduct Batch Processes with Transfer and Setup Times

Seung Heon Byun, Hae Pyeong Lee, Kyong Sook Kim and Kyong Ok Yoo†

Dept. of Chem. Eng., College of Eng., Hanyang University

*Dept. of Chem. Eng., Donyang Junior College

(Received 12 May 1997; accepted 4 October 1997)

요 약

본 연구에서는 수송 시간과 준비 시간이 고려된 다제품 회분식 공정의 최적 생산 계획 문제를 이진 변수 Y_{ij} 를 도입하여 MILP(Mixed Integer Linear Programming)모델로 새롭게 나타냄으로써 기존의 MINLP(Mixed Integer Nonlinear Programming) 모델이 최적해를 찾아가는 과정에서 local optimum으로 빠져 최적해를 보장할 수 없다는 단점을 해결하였다. 또한 문제 크기가 클 경우, 기존의 경험 법칙을 이용하여 초기해를 구한 후, 우수한 결과가 모여 있을 것으로 생각되는 해공간(solution group)을 제안된 MILP를 통해 조사해 봄으로써 경험 법칙의 성능을 향상시킬 수 있는 Approximate MILP를 제시하였다.

Abstract—Previous MINLP models in the multiproduct batch scheduling problems with nonzero transfer and setup times have defects in obtaining a local optimal solution. Therefore, a newly formulated MILP models using binary variable Y_{ij} were presented. In case of large size problems, approximate MILP method which can improve the efficiency of previous heuristics was proposed. After obtaining an initial solution by previous heuristics, a solution group which was supposed to obtain good results was investigated using the proposed MILP model.

Key words: MILP, Approximate MILP, Scheduling, UIS, NIS

1. 서 론

1950년대의 화학 공장이 안고 있던 과제는 회분식 공정을 현대적인 연속 공정으로 전환하는 것이었다. 그러나 최근 들어서도 화학공업 산업에서 연속 공정이 보편화되어 가고 중요한 위치를 차지하고 있음에도 불구하고 고부가가치의 정밀 화학제품에 대한 수요가 커짐에 따라 회분식 공정에 대한 관심이 점차 증가되고 있다.

1985년에 실시한 조사에 따르면 회분식 공정의 단지 6%만이 연속 공정으로 바뀌었다고 보고되어 있다. 소품종 대량 생산을 목적으로 하는 연속 공정은 반응 조건이 비교적 간단하고 일년 내내 수요가 있는 제품들에 적합한 반면, 회분식 공정은 약품, 식품, 생물 화학 제품, 전기 물질과 같이 정밀성을 요구하는 고부가가치의 소량 생산 물이나 유사한 생산 경로를 통해 생산되는 다품종 제품의 생산에 적합한 공정으로서 특히, 공정 및 운전상의 유연성 때문에 계절 상품에 유용하다는 특징을 갖고 있다. 이러한 회분식 공정의 조업은 연

속식 공정의 조업과 달리 많은 노동력의 요구, 장치의 효율 저하, 다음 조업을 위한 장치의 제 정비 등 많은 생산 경비가 소요되어 연속식 공정에 비해 조업의 개선 여지가 매우 큰 공정이라 할 수 있다.

이러한 소요 경비의 절감에 대한 연구 중에서도 가장 먼저 연구되어 온 것이 생산스케줄링이다. 효율적 회분식 스케줄링은 일반적으로 회분식 공정에서 생산성 효율의 최적화를 위해 생산되는 제품들의 순서와 그 순서에 의한 제품시간도를 작성하는 것을 포함하고 있다. 회분식 공정 최적화의 목적 함수로는 일반적으로 "makespan" 즉 모든 제품을 생산하는데 소요되는 총 시간을 많이 채택하고 있다. 조업 시간 단축은 기업의 경영적 차원에서 볼 때 원가 절감, 노동력 절감, 생산량 증대의 실질적 이득뿐 아니라, 납기 단축, 원료 및 제품의 원활한 순환 등에 기여하여 기업 경쟁력 강화의 측면에서도 매우 큰 효과를 거둘 수 있기 때문에, 이러한 최소 총 조업 시간을 목적 함수로 하는 생산 계획은 절실히 요구되는 연구 분야라 할 수 있다. 그 외에도 전 제품이 공정을 거치는 평균 시간, 최대 지연 시간, 지불

만기와 재고 비용을 고려한 penalty를 목적 함수로 채택하고 있다.

현재까지 개발되어 발표된 전체 생산 계획 방법론은 크게 공정을 수학적으로 정확히 표현한 후, 최적화 문제를 풀이하는 최적화 기법과 경험적 법칙을 이용하여 비교적 단순하고 간단한 방법으로 준최적 해답을 얻는 방법으로 나눌 수 있다.

Rajagopalan과 Karimi[1]는 전달 시간과 조업 준비 시간을 고려한 혼합 저장 탱크 방침에서의 조업 완료 시간을 구하는 방법을 수식화하였고, Jung 등[2]은 전달 시간과 조업 준비 시간을 고려한 ZW방안에서의 스케줄링 문제를 MINLP모델식으로 풀이하였으며, Kim 등[3]이 전달 시간과 조업 준비 시간을 고려한 다양한 저장 방침의 스케줄링 문제를 MINLP를 통해 해결하였다. 최근에 Moon 등[4]은 전달 시간과 조업 준비 시간을 고려한 ZW방안에서의 스케줄링 문제를 TSP알고리즘을 이용하여 MILP모델식으로 풀이하였다. 경험적 법칙을 이용한 제품의 생산 계획 방법에는 flowshop생산 형태의 공정에서 두 단위공정계의 최적해를 찾는 고전적인 Johnson알고리즘과 Dannenbring[5]이 제안한 것으로서 장치 2개의 Johnson알고리즘을 토대로 다단위 공정에 적용할 수 있도록 개선한 RAES(Rapid Access Extensive Search)알고리즘이 준최적 생산 계획을 얻어낼 수 있는 대표적 방법으로 받아들여지고 있다. 전달 시간과 작업 준비 시간을 고려한 경험 법칙으로는 Cao와 Bedworth[6]가 제안한 QASS알고리즘과 김 등[7]에 의해 발표된 RAES방법을 발전시킨 NHES(New Heuristic Extensive Search)방법이 있다.

그 동안 ZW방안을 제외한 여러 저장 방침에서 제품의 전달 시간과 조업 준비 시간을 고려한 다품종 회분식 공정의 최적 생산 계획을 수립하는데 있어서 MINLP모델식으로밖에 수식화되어지지 않아 이진 변수(binary variable), X_n 는 비선형성을 유발하게 된다. 그러나 비선형일 경우 일반적으로 최적해를 찾아가는 과정에서 local optimum에 빠질 경우, 더 이상의 값을 찾아가지 않고 해답으로 받아들임으로써 최적해를 보장할 수 없다는 단점을 가지고 있다. 그러므로 본 연구에서는 MINLP식들을 MILP로 모델링함으로써 최적해를 찾아내고자 하였다.

수학적 접근 방법 등은 10개 이상의 제품을 갖는 문제에서 해를 못 구하고 기존의 경험 법칙은 제품수에 영향을 받지 않으나 최적해와의 차이가 있는 준최적해라는 단점을 어느 정도 극복하기 위해 경험 법칙에 extensive search방식으로서 본 연구에서 제안한 MILP를 도입함으로써 제품 크기가 클 경우 보다 나은 준최적해를 구할 수 있는 방법을 연구하였다.

2. 수송 시간과 준비 시간을 고려한 조업 완료 시간 결정

여기서는 UIS, NIS, FIS 조업 하에서의 다품종 회분식 공정의 목적 함수로서 makespan 즉 조업 완료 시간을 구하려 한다. 먼저 M개의 장치[여기서는 한 단계(stage)에 하나의 장치만 있다고 생각한다.]를 가지고 N개의 제품을 생산하는 회분식 공정에 대한 가정은 다음과 같다.

- (1) 저장 탱크에서의 작업 준비 시간은 무시된다고 생각한다.
- (2) 모든 제품들은 모든 장치에서 같은 순서로 처리되어진다.
- (3) 진행되고 있는 공정이 잠시 중단되었다가 잠시 후 다시 시작될 수가 없다.
- (4) k장치에서 저장 탱크로 들어가는 전달 시간과 저장 탱크에서 k+1장치로 들어가는 전달 시간은 k장치에서 k+1장치로 들어가는 전달 시간과 같다.
- (5) k+1 장치가 준비되어 있지 않고 저장 탱크가 비어 있으면 k장

치에서 끝난 제품은 저장 탱크로 전달되어진다. 그러나 만약 k+1장치가 준비되어 있으면 저장 탱크를 거치지 않고 바로 k+1장치로 전달되어진다. k번째 장치의 공정이 끝나 있으면 k+1장치가 작업 준비 시간이 끝나자마자 제품을 받아들일 수 있으며 심지어 k번째 장치에서 저장 탱크로 같은 제품을 보내고 있는 도중일 때조차 k+1번째 장치는 그 제품을 저장 탱크로부터 받아들일 수 있다. 이 가정은 저장 탱크 작업이 이중 전달 시간을 줄일 수 있게 설계되었다는 가정에서 출발한다.

(6) 제품은 작업 도중 분리되거나 합쳐질 수 없다.

(7) 저장 탱크는 공유되어지지 않고 장치와 장치 사이에 존재되어지는 것으로 간주한다.

2-1. UIS 조업 하에서의 조업 완료 시간 결정

UIS의 경우는 장치간에 중간 저장 탱크가 무한히 많이 있어서 제품 i가 k번째 장치에서 처리가 끝난 다음에 k+1번째 장치가 비어 있지 않으면 중간 저장 탱크로 옮겨져서 장치 k에 머무르지 않기 때문에 장치 k가 다음 제품의 생산 준비를 할 수 있다. 장치 설치비는 많이 드나 장치의 노는 시간(idle time)을 최소화 할 수 있다. 장치 k에서 제품 i의 공정 완료 시간은 다음과 같은 두 가지 경우를 고려해야 한다.

- (1) 장치 k-1에서 제품 i의 공정이 완료된 시간
- (2) 장치 k에서 제품 i-1을 보내고 작업 준비를 끝낸 시간

첫 번째 경우로 장치 k가 작업 준비가 끝난 상태로 비어 있고 제품 i가 장치 k-1에서 끝난 경우, 제품 i는 장치 k로 들어갈 수 있다. 그러므로 제품 i가 장치 k에서의 조업 완료 시간은 다음과 같다.

$$C_{ik} \geq C_{(k-1)i} + a_{(k-1)i} + t_{ik} \quad (1)$$

두 번째 경우로 제품 i가 장치 k-1에서 조업이 완료된 후에도 장치 k가 제품 i를 처리할 준비를 하지 못한 경우는 장치 k에서 제품 i-1을 처리하고 제품 i를 받아들일 준비가 끝날 때까지 저장 탱크에서 저장되어야만 한다. 그러므로 제품 i의 장치 k에서의 조업 완료 시간은 다음과 같다.

$$C_{ik} \geq [C_{(i-1)k} + a_{(i-1)k} + S_{(i-1)k}] + a_{ik-1} + t_{ik} \quad (2)$$

그러므로 k번째 장치에서 제품 i의 공정을 끝내고 내보내는 시간을 C_{ik} 라 하면 식 (3)이 얻어진다.

$$C_{ik} = \max[C_{(i-1)k} + a_{(i-1)k} + S_{(i-1)k}, C_{(k-1)i}] + a_{ik-1} + t_{ik} \quad (3)$$

2-2. NIS 조업 하에서의 조업 완료 시간 결정

NIS방침의 경우 UIS방침과는 달리 중간 저장 탱크가 없기 때문에 장치 k에서 장치 k+1이 빌 때까지 조업이 끝난 이후에도 기다려야 한다는 제약 조건이 생겨난다. UIS조업 완료 시간에 다음과 같은 항이 추가되어야만 한다.

$$C_{ik} \geq C_{(i-1)(k+1)} + a_{(i-1)(k+1)} + S_{(i-1)(k+1)} \quad (4)$$

그러므로 공정 완료 시간에 대한 수식은 (5)와 같이 나타낼 수 있다.

$$C_{ik} = \max[C_{(i-1)k} + a_{(i-1)k} + S_{(i-1)k}, C_{(k-1)i}, C_{(i-1)(k+1)} + a_{(i-1)(k+1)} + S_{(i-1)(k+1)} - a_{ik-1} - t_{ik}] + a_{ik-1} + t_{ik} \quad (5)$$

3. 혼합 정수 선형 모델

기존의 혼합 정수 비선형 모델이 비선형성으로 인해 최적성을 보

장할 수 없는 소위 near local optimum solution법에 속하므로 이진 변수 X_{ij} 이외에 이진 변수 Y_{ij} 를 도입하여 개선하였다.

이진 변수 Y_{ij} 와 X_{ij} 는 다음과 같이 정의된다.

$$X_{ij} = \begin{cases} 1 & i \text{가 } j \text{위치에 있을 때} \\ 0 & \text{그렇지 않을 때} \end{cases}$$

$$Y_{ij} = \begin{cases} 1 & i \text{가 } j \text{위치에 있고 } l \text{이 } j-1 \text{위치에 있을 때} \\ 0 & \text{그렇지 않을 때} \end{cases}$$

그리고 이진 변수 Y_{ij} 와 X_{ij} 의 제한 조건식들은 다음과 같다.

$$Y_{ij} \geq X_{ij} + X_{l(j-1)} - 1 \quad (6)$$

$$Y_{ij} \leq \frac{1}{2} (X_{ij} + X_{l(j-1)}) \quad (7)$$

식 (6), (7)은 이진 변수 Y_{ij} 가 i 가 j 위치에 있고 l 이 $j-1$ 위치에 있을 때 1을 할당하고, 그렇지 않을 때는 0을 할당할 수 있도록 한다.

$$\sum_{i=1}^N Y_{ij} = 0 \quad (j=1, 2, \dots, N) \quad (8)$$

식 (8)은 제품 i 다음 위치에 제품 i 가 할당되지 못하도록 한다.

$$\sum_{i=1}^N X_{ij} = 1 \quad (j=1, 2, \dots, N) \quad (9)$$

식 (9)는 위치 j 에 한 개의 제품만이 할당되도록 한다.

$$\sum_{j=1}^N X_{ij} = 1 \quad (i=1, 2, \dots, N) \quad (10)$$

식 (10)은 제품 i 가 한 위치에만 할당되도록 한다.

위의 식 (6)-(10)은 각각의 저장 탱크 방침의 모델식에 공통적으로 사용되는 제한 조건이다.

3-1. UIS 작업 하에서의 혼합 정수 선형 모델

본 연구는 저장 탱크를 무한한 것으로 보는 UIS방침 하에서의 작업 완료 시간을 기초로 하여 기존의 논문에서 다루고 있는 혼합 정수 비선형 모델이 아닌 정수 선형 모델로 수식화하였다.

Objective Function : Min $C_{NM} + \sum_{i=1}^N a_{iM} X_{iN}$

for unit 1 ($k=1$)

$$C_{11} \geq \sum_{i=1}^N X_{i1} (a_{i0} + t_{i1}) \quad (11)$$

$$C_{j1} \geq C_{(j-1)1} + \sum_{i=1}^N a_{i1} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{li1} + \sum_{i=1}^N X_{ij} (a_{i0} + t_{i1}) \quad (12)$$

for unit k ($k=2, \dots, M-1$)

$$C_{jk} \geq C_{j(k-1)} + \sum_{i=1}^N X_{ij} (a_{i(k-1)} + t_{ik}) \quad (13)$$

$$C_{jk} \geq C_{(j-1)k} + \sum_{i=1}^N a_{ik} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{lik} + \sum_{i=1}^N X_{ij} (a_{i(k-1)} + t_{ik}) \quad (14)$$

for unit M

$$C_{jM} \geq C_{j(M-1)} + \sum_{i=1}^N X_{ij} (a_{i(M-1)} + t_{iM}) \quad (15)$$

$$C_{jM} \geq C_{(j-1)M} + \sum_{i=1}^N a_{iM} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{liM} + \sum_{i=1}^N X_{ij} (a_{i(M-1)} + t_{iM}) \quad (16)$$

3-2. NIS 작업 하에서의 혼합 정수 선형 모델

NIS저장 방침일 경우, 저장 탱크가 없으므로 다음과 같이 수식화된다.

Objective Function : Min $C_{NM} + \sum_{i=1}^N a_{iM} X_{iN}$

for unit 1 ($k=1$)

$$C_{11} \geq \sum_{i=1}^N X_{i1} (a_{i0} + t_{i1}) \quad (17)$$

$$C_{j1} \geq C_{(j-1)1} + \sum_{i=1}^N a_{i1} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{li1} + \sum_{i=1}^N X_{ij} (a_{i0} + t_{i1}) \quad (18)$$

$$C_{j1} \geq C_{(j-1)2} + \sum_{i=1}^N a_{i2} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{li2} \quad (19)$$

for unit k ($k=2, \dots, M-1$)

$$C_{jk} \geq C_{j(k-1)} + \sum_{i=1}^N X_{ij} (a_{i(k-1)} + t_{ik}) \quad (20)$$

$$C_{jk} \geq C_{(j-1)k} + \sum_{i=1}^N a_{ik} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{lik} + \sum_{i=1}^N X_{ij} (a_{i(k-1)} + t_{ik}) \quad (21)$$

$$C_{jk} \geq C_{(j-1)(k+1)} + \sum_{i=1}^N a_{i(k+1)} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{li(k+1)} \quad (22)$$

for unit M ($k=M$)

$$C_{jM} \geq C_{j(M-1)} + \sum_{i=1}^N X_{ij} (a_{i(M-1)} + t_{iM}) \quad (23)$$

$$C_{jM} \geq C_{(j-1)M} + \sum_{i=1}^N a_{iM} X_{l(j-1)} + \sum_{i=1}^N \sum_{l=1}^N Y_{lij} S_{liM} + \sum_{i=1}^N X_{ij} (a_{i(M-1)} + t_{iM}) \quad (24)$$

3-3. 적용 사례

본 연구에서 제안한 MILP모델과 기존의 MINLP모델의 결과값을 비교하기 위해서 제품수와 공정의 수 즉 장치수를 달리하는 6개의 예제를 사용하였으며 각 예제들에 대해서 목적 함수인 makespan값들의 결과값 비교를 Table 1에 제시하였다.

이 결과값들을 얻기 위해서 SUN/SPARC workstation 시스템을 사용했으며, MINLP의 경우에는 상용 소프트웨어인 GAMS의 DICOPT ++ solver를 사용하였고, MILP의 경우에는 GAMS의 OSL solver를 사용하여 계산을 수행하였다.

Table 1의 예제 3, 5, 6에 대한 결과값들을 살펴본 결과, 기존의 MINLP모델은 local optimum값을 얻게 됨으로써 제안된 MILP모델의 global optimum값과 차이가 있음을 알 수 있었다.

이 예제들 가운데 6개의 제품과 8개의 장치를 사용한 예제 3에 대한 공정 데이터와 결과값들을 Table 2와 3에 나타냈다. 그 결과로서 NIS방침일 경우에는 대부분의 예제에서 두 모델간의 결과값들의 차이가 없었지만 UIS방침에서는 제안된 MILP일 경우, 생산sequence가 p5-p2-p3-p6-p1-p4로서 makespan값이 265였으며, MINLP일 경우에는 생산sequence가 p5-p6-p1-p4-p2-p3으로서 makespan값이 269였기 때문에 MINLP의 경우에는 local optimum에 빠지는 것을 나타내고 있다. 또한, 예제 3의 최적해에 대한 UIS방침과 NIS방침의 Gantt chart를 Fig. 1과 2에 나타내었다.

CPU time은 장치수보다 이진 변수의 수에 직접 관련되는 제품수에 더 큰 영향을 받는다. 제품수 6개인 예제 3에서 제안된 MILP로 풀었을 때, UIS의 경우는 26.09 sec, NIS의 경우 29.15 sec였으며, MINLP로 풀었을 때, UIS의 경우는 42.03 sec, NIS의 경우는 41.86 sec로서 제안된 MILP모델일 경우가 보다 빠른 CPU time을 나타내었다. 그

Table 1. Comparison of makespan between proposed MILP and MINLP

	Proposed MILP		MINLP	
	UIS policy	NIS policy	UIS policy	NIS policy
E×1(4×4)	120	126	120	126
E×2(6×4)	151	153	151	153
E×3(6×8)	265	271	269	271
E×4(8×4)	173	185	173	185
E×5(8×6)	245	255	253	255
E×6(8×8)	311	318	318	324

Table 2. Data for example 3 in this study

(a) Processing time									(b) Transfer time									(c) Setup time (sequence oriented)							
	U1	U2	U3	U4	U5	U6	U7	U8		U0	U1	U2	U3	U4	U5	U6	U7	U8		P1	P2	P3	P4	P5	P6
P1	10	20	5	30	10	35	15	21	P1	2	2	2	2	3	3	3	2	3	P1	0	3	2	1	1	3
P2	15	8	12	10	23	22	17	28	P2	3	3	3	3	1	2	2	3	2	P2	4	0	1	3	3	1
P3	20	7	9	5	18	19	22	13	P3	2	4	2	2	1	3	3	1	2	P3	2	1	0	2	2	2
P4	13	7	17	10	24	7	18	14	P4	2	2	1	4	2	1	1	2	1	P4	4	1	3	0	1	3
P5	8	3	16	7	20	19	25	26	P5	1	1	1	3	2	2	2	2	2	P5	2	2	2	1	0	4
P6	6	9	22	7	10	5	15	18	P6	3	3	2	2	2	1	1	3	4	P6	1	3	4	2	2	0

Table 3. Comparison of proposed MILP and MINLP for example 3

S/W TOOL	Proposed MILP		MINLP	
	OSL solver		DICOPT++ solver	
H/W TOOL	SUN/SPARC workstation		SUN/SPARC workstation	
	UIS	NIS	UIS	NIS
Makespan	265	271	269	271
Sequence	5-2-3-6-1-4	5-3-2-6-1-4	5-6-1-4-2-3	5-3-2-6-1-4
CPU time (sec)	26.0901	29.1501	42.0301	41.8601

러나 MINLP모델의 경우에 이진 변수의 수는 제품수의 제곱 즉 N^2 인데 비해서 제안된 MILP모델의 경우에는 이진 변수의 수가 $N^3 + N^2$ 이기 때문에 예제 4와 같이 제품수가 8개로 증가하면 MINLP모델에서는 이진 변수, X_{ij} 의 수가 64개인 반면에 제안한 MILP모델은 이진 변수 X_{ij} 와 Y_{ij} 의 수가 576개로 급격히 증가함으로써 UIS방침일 경우, 120.68 sec로 CPU time이 증가된 MINLP모델 보다 제안한 MILP 모델은 더 큰 증가없이 395.26 sec를 요구함을 알 수 있었다. 하지만 제안된 MILP모델의 경우에는 기존의 MINLP모델과 마찬가지로 제품수 8개, 장치수 8개 정도의 문제 크기를 안정적으로 처리할 뿐만 아니라 MINLP 경우처럼 local optimum에 빠지지 않고 global optimum을 찾아가기 때문에 본 연구에서 제안한 MILP모델식의 우수성

과 강건성을 입증할 수 있었다.

4. Approximate MILP Heuristic

제안된 MILP모델은 작업 준비시간과 제품의 전달 시간을 고려한 스케줄링 문제를 풀어 최적값을 찾는데 매우 효과적인 방법 중의 하나이다. 그러나 조업 순서가 일정한 다품종 회분식 생산의 조업 방법은 N!의 해공간을 갖기 때문에 제품수가 1개 늘면서 해공간은 N (제품수)배씩 커진다. 이러한 이유로 다품종 생산 계획 문제는 NP-complete하므로 큰 문제일 경우, 해공간 크기는 천문학적 수치를 나타내게 된다. MILP모델과 기존의 MINLP모델의 경우는 제품수와 장치수의 증가에 따라 CPU time이 지수적으로 증가하고, 제품수 8개 정도가 한계였지만 제안된 MILP모델은 모든 저장 방침을 고려해 줄 수 있으며, 그 정확성으로 인해 dispatch rule을 이용한 기존의 경험 법칙에 local search방식으로서 적용하는데 다른 접근 방법보다 효율적인 방법이 된다. 생산 계획 문제의 특징 가운데 하나는 많은 경험 법칙에서 나타나는 것처럼 특정의 해공간 지역은 좋은 해가 없을 가능성이 매우 높고, 또 다른 특정의 해공간에는 좋은 해들이 모여 있을 가능성이 높다는 점이다. 본 연구에서 사용한 경험 법칙인 NHES 역시 dispatch rule로서 NH(New Heuristic)를 개발하고 local search방법으로 ES(Extensive Search)를 적용시킨 방법이다. 제안된 MILP모델의 경우, 제품이 존재하는 범위를 모든 위치로 두고 풀어 나아가기 때문에 본 연구에서는 NHES를 통해 구한 준최적해를 초기해로 둬으로써 상당히 우수한 결과가 모여 있을 가능성이 높은 지역을 찾아내어 제품의 존재 범위를 제한하는 방법을 사용하였다. 그 예로서 만약 제품수 N이 6일 경우 경험 법칙을 통해 구한 값 중 첫 번째 위치에 해당하는 제품이 MILP에서 고려해 줘야 할 위치는 1, 2, 3 위치로 한정된다.

Approximate MILP Heuristic알고리즘은 다음과 같이 요약될 수 있다.

- (1) NHES를 통해 초기해를 구해 낸다.
- (2) NHES를 통해 각각의 제품에 대한 존재 범위를 설정한다.
- (3) Table 4의 존재 범위에서 생산 순서를 MILP모델을 통해 계산한다.

Approximate MILP Heuristic은 본 연구에서 제안한 MILP모델이 갖고 있는 제품수의 한계를 극복하기 위해서 약간의 정확도 감소를 감수해야 하지만 더 큰 문제를 풀어야 할 경우에 매우 효율적이다. 또 NH가 UIS 저장 방침에 기초하여 제품 순서를 결정한 후, 각 저장 탱크 방침에 적용하는 방법이므로 이때 생길 수 있는 오차를 어느 정도 극복할 수도 있다.

4.1. 적용 사례

본 연구에서 제안한 Approximate MILP모델의 결과를 살펴보기

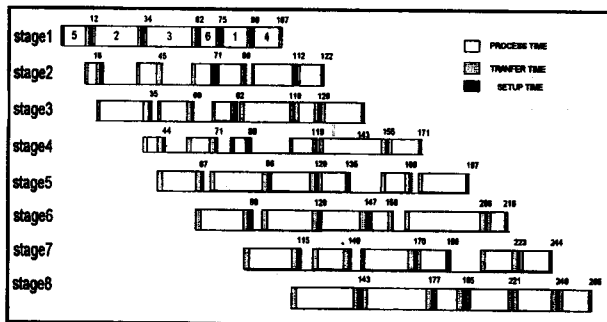


Fig. 1. Gantt chart of example 3 with UIS policy.

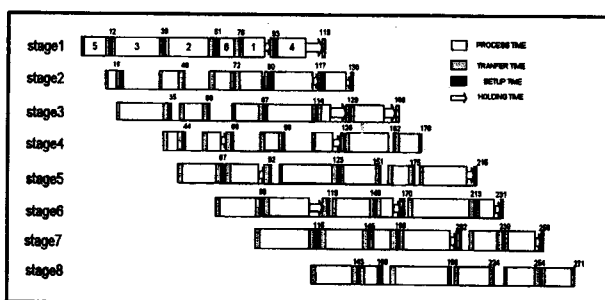


Fig. 2. Gantt chart of example 3 with NIS policy.

Table 4. Limited possible sequence position of each product

	Possible position									
	1	2	3	4	5	6	•	•	•	N
1	X	X	X							
2	X	X	X	X						
3	X	X	X	X	X					
4		X	X	X	X	X				
5			X	X	X	X	X			
6				X	X	X	X	X		
•					X	X	X	X	X	
•						X	X	X	X	X
•							X	X	X	X
N								X	X	X

위해서 제품수와 장치수를 달리하는 12개의 예제를 사용하였다. 이 예제들은 기존의 문헌을 통해 발표했던 2개(예제 1, 4)의 예제와 제품수의 한계성을 파악할 목적으로 제품수와 장치수를 증가시켜 무작위로 생성한 10개의 예제들로 구성되어 있다. 그리고 이 예제들에 대한 결과값들(Table 6, 7)을 얻기 위해서 Approximate MILP의 경우는 SUN/SPARC workstation 시스템과 OSL solver를 사용했으며, NHES의 경우는 pentium-100 PC 시스템과 본 연구에서 개발한 프로그램을 사용했다.

이 예제들 가운데 제품수 12개와 장치수 6개를 갖는 예제 10의 공정 데이터를 Table 5에 제시했으며 Approximate MILP와 기존의 NHES의 결과값들을 비교해 보았다. UIS방침에서 Approximate MILP로 풀었을 경우, p5-p7-p2-p4-p8-p10-p6-p1-p12-p11-p9-p3의 sequence와 344의 makespan값을 얻음으로써 NHES를 통해 풀었을 때의 makespan값 348(생산sequence p5-p2-p4-p7-p6-p1-p8-p10-p9-p12-p11-p3)보다 소폭의 향상된 결과를 나타내고 있지만 NIS방침의 경우는 NHES를 통해 풀었을 때 makespan값이 378(생산sequence p2-p4-p5-p1-p6-p7-p8-p10-p9-p11-p12-p3)인데 비해서 Approximate MILP로 풀었을 때의 makespan값은 361(생산sequence p5-p4-p2-p6-p1-p7-p8-p10-p9-p11-p12-p3)로서 더욱 향상된 결과를 얻을 수 있었다.

제품수 8개 이하인 작은 크기의 예제들에서 두 모델의 결과를 살펴보면 NHES모델일 경우, UIS방침일 때 최적해와의 오차가 3.49%, NIS방침일 때 3.36%이고 최대 오차는 10.3%에 이르는데 반해 Approximate MILP모델일 경우, UIS방침일 때 0.64%, NIS방침일 때 1.28%로 많은 향상을 나타냈다. 또한 최적해를 보장할 수 없는 큰 크기의 문제일 경우, 최적해를 찾지 못하므로 두 풀이 방법을 비교했을 때 Approximate MILP가 NHES를 통한 방법보다 UIS방침은

Table 6. Comparison of makespan for each example with UIS policy

	MILP	Approximate MILP	NHES
E×1(4×4)	120	120	120
E×2(6×4)	151	151	151
E×3(6×8)	265	255	288
E×4(8×4)	173	174	182
E×5(8×6)	245	249	252
E×6(8×8)	311	316	324
E×7(10×4)	×	218	230
E×8(10×8)	×	377	385
E×9(12×4)	×	269	269
E×10(12×6)	×	344	348
E×11(12×8)	×	415	420
E×12(14×4)	×	306	306
E×13(14×6)	×	392	408
E×14(14×8)	×	496	519

Table 7. Comparison of makespan for each example with NIS policy

	MILP	Approximate MILP	NHES
E×1(4×4)	126	126	126
E×2(6×4)	153	153	154
E×3(6×8)	271	279	299
E×4(8×4)	185	190	193
E×5(8×6)	255	257	257
E×6(8×8)	318	322	331
E×7(10×4)	×	234	234
E×8(10×8)	×	379	388
E×9(12×4)	×	286	296
E×10(12×6)	×	361	378
E×11(12×8)	×	439	450
E×12(14×4)	×	332	344
E×13(14×6)	×	407	413
E×14(14×8)	×	512	534

2.34%, NIS방침은 2.80%의 향상된 결과를 나타내므로 본 연구를 비롯해 선행 연구 결과[7]에서 밝힌 NHES에서의 오차 허용 한계가 10%정도임을 고려할 때 NHES보다 50%이상의 성능 향상이 있음을 입증할 수 있었다.

본 연구에서 제안한 MILP와 기존의 MINLP방법은 제품수의 한계가 8개 정도인데 반해 Approximate MILP는 제품수 14개와 장치수 8개 정도의 문제에서도 안정적으로 준최적해를 구할 수 있었으며 그 결과값은 기존에 발표한 NHES 경험 법칙[7]보다 우수한 결과값을

Table 5. Data for example 10 in this study

(a) Processing time							(b) Transfer time							(c) Setup time(sequence oriented)													
	u1	u2	u3	u4	u5	u6		u0	u1	u2	u3	u4	u5	u6		p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12
p1	10	20	5	30	8	15	p1	2	2	2	2	3	3	2	p1	0	3	2	1	1	3	3	2	3	3	3	2
p2	15	8	12	10	23	22	p2	3	3	3	3	1	2	2	p2	4	0	1	3	3	1	2	3	3	4	2	3
p3	20	7	9	5	18	13	p3	2	4	2	2	1	3	3	p3	2	1	0	2	2	2	1	2	2	2	2	4
p4	13	7	17	10	24	14	p4	2	2	1	4	2	2	4	p4	4	1	3	0	1	3	1	2	3	2	3	2
p5	8	3	16	7	10	19	p5	1	1	1	3	2	3	2	p5	2	2	2	1	0	4	2	4	2	2	1	2
p6	6	9	22	7	17	17	p6	3	3	2	2	2	2	1	p6	1	3	4	2	2	0	2	1	1	1	2	3
p7	7	5	15	12	12	12	p7	1	2	1	2	1	3	2	p7	2	1	2	1	3	1	0	2	4	3	2	3
p8	14	13	6	4	12	18	p8	2	4	1	1	2	4	3	p8	1	4	2	3	3	3	2	0	2	1	3	2
p9	17	12	10	15	20	23	p9	2	3	2	1	3	2	3	p9	3	2	2	3	1	3	2	4	0	3	2	1
p10	10	9	15	22	12	17	p10	4	2	2	2	1	2	2	p10	2	3	4	2	1	3	3	2	2	0	2	2
p11	13	16	22	9	22	14	p11	3	2	3	2	2	1	2	p11	3	2	3	2	2	2	3	2	1	4	0	2
p12	17	20	19	12	13	18	p12	2	2	2	3	1	3	3	p12	2	2	3	4	1	2	2	3	2	1	1	0

보이고 있다.

5. 결 론

본 연구는 지금까지 다품종 회분식 공정의 최적 생산 계획에서 수송 시간과 준비 시간을 모두 고려할 경우, 비선형성의 발생으로 인해서 local optimum값에 빠지는 것을 방지하기 위하여 MILP모델 형태로 새롭게 나타냈으며, 상용 최적화용 소프트웨어인 GAMS의 OSL solver를 통해 SUN/SPARC workstation에서 제품수 8개, 장치수 8개까지 안정적으로 최적값을 구할 수 있었다.

본 연구에서 제안한 MILP모델로 계산을 수행한 결과, 최적해를 보장하기는 하지만 NP-complete한 다품종 회분식 공정에서 제품수 8개가 넘을 경우에 수학적 생산 기법으로는 구할 수 없기 때문에 기존의 경험 법칙을 이용하여 우수한 결과가 모여 있을 것으로 생각되는 해공간을 구하고, 본 연구에서 제안한 MILP모델을 통해 조사해봄으로써 제품수 14개 정도까지의 문제에서 보다 안정적이고 near optimum값을 구해냄으로써 기존의 경험 법칙보다 약 50%정도의 성능 향상을 얻을 수 있었다.

사용기호

- a_{ik} : transfer time of product i out of batch unit k to batch unit $k+1$
 C_{ik} : completion time of i th product in the sequence in batch unit k where the product is finished and begins to be transferred out of unit k
 M : number of batch units in the plant

- N : number of products to be produced
 S_{ijk} : setup time required for product j after product i in batch unit k
 t_{ik} : processing time of product i in batch unit k
 X_{ij} : binary variable
 Y_{ik} : binary variable

하첨자

- i : index of product
 j : index of product
 k : index of stage
 l : index of product

참고문헌

1. Rajagopalan, D. and Karimi, I. A.: *Computers Chem. Engng.*, **13**, 175(1989).
2. Jung, J. H., Lee, H. K., Yang, D. R. and Lee, I. B.: *Computers Chem. Engng.*, **18**(6), 537(1994).
3. Kim, M. S., Jung, J. H. and Lee, I. B.: *Ind. Eng. Chem. Res.*, **35**, 4058(1996).
4. Moon, S. D., Park, S. W. and Lee, W. K.: *Ind. Eng. Chem. Res.*, **35**, 3458(1996).
5. Dannenbring, D. G.: *Manag. Sci.*, **23**, 1174(1977).
6. Cao, J. and Bedworth, D. D.: *Int. J. Prod. Res.*, **30**(8), 1819(1992).
7. 김경숙, 조영상, 이현구: *화학공학*, **32**(3), 265(1994).