

## 회분식 공정의 생산계획 문제의 객체지향형 프로그래밍에 의한 표현

김민석, 이인범  
포항공과대학교 화학공학과

### Object-oriented Programming Framework for Batch Process Scheduling Problems

Minseok Kim, In-Beum Lee  
Department of Chemical Engineering, POSTECH

#### 서론

제약, 고분자 등 다품종 소량 생산을 위해 많이 사용되어지고 있는 회분식 조업에는 다제품(multi-product) 회분식 조업과 다목적(multi-purpose) 회분식 조업으로 크게 구분되어지며, 회분식 조업의 주요 특징인 유연한 생산 능력을 최대화하기 위해 생산계획에 대한 연구가 많이 이루어지고 있으며, 주로 중간 저장탱크의 활용 전략에 따른 조업완료시간 결정 알고리즘의 개발, 최적 생산계획 탐색을 위해 MINLP, SA, GA 등의 최적화 알고리즘의 적용, 초기 생산계획으로부터의 조업 이탈에 대한 재생산계획 전략에 관한 연구가 주류를 이루고 있다.

그러나 이러한 연구에서는 대부분 연구의 편의를 위해 공정에 대한 표현 양식을 최대한 간략히 하는 경우가 많았다. 이러한 간략화는 생산계획 문제에 대한 연구의 편의를 가져오는 반면 실제 현장의 실정을 그대로 적용하지 않음으로 인해 연구실을 벗어나 상용화 시스템으로 개발하고자 하는 경우 고려해야 할 새로운 인자들로 인해 개발된 새로운 이론이 제대로 적용되지 못하는 경우가 많았던 것도 사실이다.

따라서, 본 연구에서는 생산계획 문제를 위한 여러 분야에서의 새로운 접근법의 개발 초기 단계에서부터 이러한 현장의 대부분의 영향 요소들을 고려하여 표현함으로써, 실제 현장에 적용 가능한 S/W로의 개발을 용이하게 할 수 있는 방안을 모색하였다. 이에 따라 복잡한 시스템을 표현하여 프로그래밍화 하는데 있어서, 개발자에게 많은 이득을 가져올 수 있는 객체지향형 프로그래밍이 유용한 도구로 사용되어질 수 있다.

#### 이론

객체지향형 프로그래밍은 크고 복잡한 소프트웨어 체제를 개발하는 컴퓨터 프로그래머들에게 효과적으로 시스템을 표현할 수 있는 방안으로 개발되어졌다. 기존의 프로그래밍 언어들이 데이터를 표현하는 것과 단위 기능을 수행하는 각 함수들을 구성하는 것으로 분리되어서 그들 함수들의 수행 순서를 표현함으로써 이루어지는 접근법이었던 것과는 달리, 객체지향형 프로그래밍에서는 전체를 구성하고 있는 작은 객체들에 대해 정의해 나가 이들 객체들간의 관계 수행에 의해 전체 시스템이 이루어지도록 구성하는 접근법

이라 할 수 있다. 이러한 객체지향형 프로그래밍을 지원하는 프로그래밍 언어로는 Ada, CLOS, C++등이 있으며, 최근에는 C++이 가장 선호되는 언어로 사용되어지고 있다.

객체지향형 프로그래밍에서는 시스템을 정의하는 두가지 방안인 순차적인 기능의 정의와 데이터에 대한 표현 정의를 하나의 class의 형태 아래에서 묶어서 구성 요소들을 정의하고 그 class들이 가지게 되는 특성(attributes)들과 수행 기능(behaviors)들을 지정하고, 여러 class들간의 관계를 파악하여 class들간의 계층적 구조로 정리하는 과정을 거치게 된다. 따라서, 다음 그림 1의 class간의 계층구조도에서 볼 수 있듯 각 class들은 하나의 class로부터 그 class의 특성을 그대로 가지면서 그 나름대로의 추가 기능과 특성을 가질 수 있는 하위 class가 유도되어 정의되어질 수 있다. 이때, 상위 class의 특성과 기능을 하위 class가 그대로 가지는 경우 이를 상속(inheritance)라 하며, 어떤 class는 다른 상위 class로부터 상속을 받아 두 class의 특성과 기능을 다 가질 수도 있다. 실제 문제를 표현할 때는 이렇게 정의된 class들로부터 각각의 객체들을 유도하여 사용하게 된다.

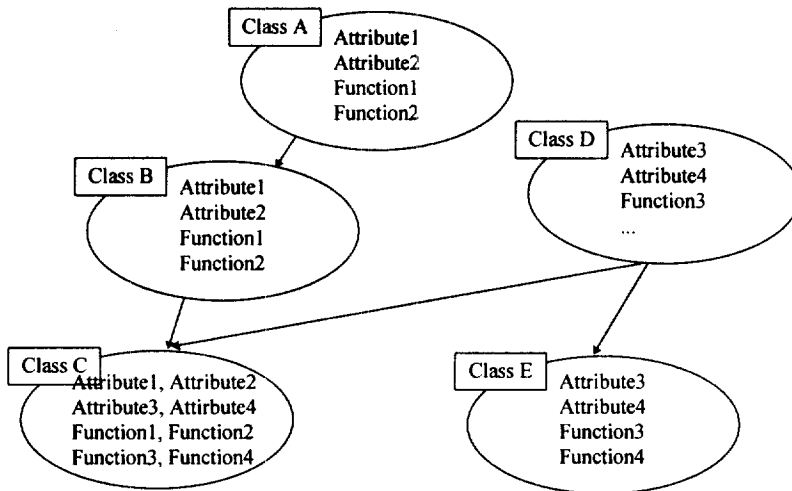


그림 1. Class 계층 구조도

**적용**

회분식 공정을 위한 생산 계획 문제를 객체지향형 프로그래밍 기법으로 표현하기 위해서는 문제를 구성하는 요소들을 파악해 나가야 하는데, 본 연구에서는 공정 자체를 표현하기 위한 class들과 생산해야하는 제품들을 표현하기 위한 class, 고객의 주문 사항을 표현하기 위한 class, 생산 계획을 표현하기 위한 class들로 분류되어 정의하였다.

회분식 공정은 크게 다제품 생산 회분식 공정과 다목적 생산 회분식 공

정으로 나뉘어지는데, 다제품 생산 공정은 모든 제품의 생산에 있어서 동일한 생산 경로를 거치는 공정으로서, 그림 2에서와 같이 (a)의 단순 일렬형과 (b)의 망형 공정으로 다시 나뉘어져서 단순 일렬형은 모든 제품의 생산을 위해 참여하는 장치가 동일한 반면 망형 회분식 공정에서는 각 제품의 생산을 위한 거쳐가게 되는 장치의 Path가 다 달라질 수 있게 된다. 이에 반해 다목적 회분식 공정에서는 각 제품의 생산을 위한 처리 단계가 달라질 수 있는 공정이다.

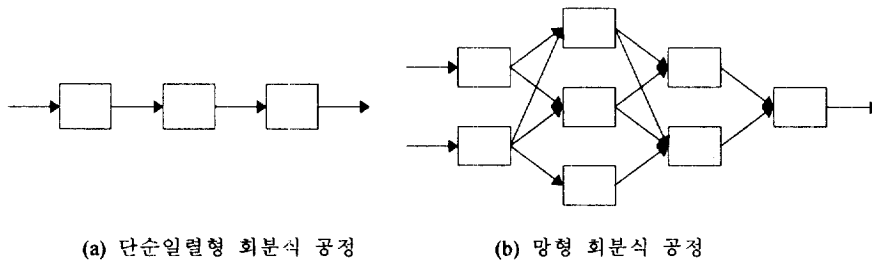


그림 2. 다제품 생산용 회분식 공정

이렇듯, 매우 다양한 형태로 존재하는 회분식 조업을 일반적으로 표현하기 위해서 전체 공정을 구성하는 요소들로는 실제 조업을 진행하는 장치 (unit), 중간체를 임시 저장하는 중간저장탱크(tank), 공정 상에서 각 회분들이 거처나갈 수 있는 생산경로(path), 각 장치들을 연결하고 있는 pipe 등으로 나뉘어지며, 장치를 표현하는 class는 경로를 선정하는 기준이 되는 역할을 하는 그 장치가 한 회분의 조업으로 처리할 수 있는 한계 용량, 이전 제품 생산으로부터 다음 제품 생산으로 변경하기 위해 필요로 되는 cleaning time 등이 해당되며 처리량에 따른 준비시간, 연결된 pipe들의 list 등을 특성 데이터로 가지고, 준비시간을 계산하는 함수등을 내장 함수로 가지게 된다. 중간저장탱크는 장치의 대부분의 특성치들과 기능들을 거의 그대로 유지하기 때문에 이를 위한 class는 장치 class로부터 유도되는 sub-class로 정의하였다. pipe를 정의하는 class는 각 장치들 간의 실제 연결을 표현하고, pipe가 처리할 수 있는 수송시간을 특성치로 가지고 이를 계산할 수 있는 함수 등을 기능을 가지도록 정의하였다.

제품(product)을 표현하는 class는 고객의 주문과는 별개로 대상 회분식 공정이 생산할 수 있는 제품들에 대한 특성치들을 미리 입력시켜 두기 위한 class로서, 각 제품의 이름, 생산 recipe, 각 단계에서 단위 물량당 생산에 소요되는 생산시간 등의 특성치들을 가지게 된다. recipe는 다시 새로운 class로 정의하여 ZW policy를 필요로 하는 그 제품의 중간체 list, 가능한 특정 경로 또는 진행되어서는 안되는 특정 경로 list 등과 그 제품이 생산되어진 후 창고 또는 저장 탱크에 재고로 쌓아둘 때 생기는 재고 비용 penalty parameter를 특성치로 가지게 정의하였다. 회분(batch)을 위한 class는 모아진 고객의 주문들로부터 공정에 투입되는 기본 단위가 되며, 각 회분들은 회분

량(batch size)등의 특성치를 가진다. 본 연구에서 회분에 관련된 변수들을 표현할 때에는  $i$  를 index 로 사용한다.

단계(Stage)는 각 제품의 생산을 위해 수행해야할 task 들의 list 들을 정의하는 class 로서, 그 기능을 수행할 수 있는 장치들을 list 형태로 저장하도록 정의하였다. 단순 일렬형에서는 단지 소속 장치가 하나인 경우로 표현되며, 다목적 회분식 공정에서는 에서 모든 제품의 생산을 위해 거쳐야 하는 각 생산 절차를 의미한다.

고객으로부터 받게되는 주문(order)을 위한 class 는 주문 코드와 고객이 원하는 납기일, 납품량, 납기일 위반에 대한 penalty parameter 값등을 특성치로 가지게 된다.

각 회분이 각 장치에서 체제하는 시간은 다음 그림 3과 같이 준비시간, 처리시간, 수송시간으로 이루어지는 형태로 나타나게 되므로 이를 생산계획의 가장 기초 단위로 사용하는 것이 유리하여 이를 class 로 정의하였다.

이외에도 많은 class 들이 정의되어져 회분식 공정을 위한 생산계획문제를 구성하게 된다.

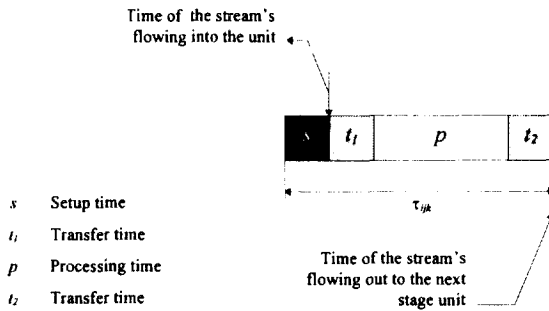


그림 3. 체제시간의 정의

**결론**

본 연구에서는 회분식 공정에 대한 생산 계획 문제에 대한 접근법을 개발함에 있어서 실제 현장에 적용하는데 큰 무리 없이 표현하기 위한 방안으로서 객체지향형 프로그래밍 기법을 적용하였다. 이러한 객체지향형 표현 양식은 G2 와 같은 전문가 시스템 개발 도구들 중에서도 대상 시스템의 표현을 위해 적용되고 있기 때문에 객체지향형 언어로 작성된 표현 양식은 전문가 시스템으로의 이식이 용이해지는 장점을 가지게 된다. 따라서 이를 통해 생산계획의 기존의 어려웠던 문제에 대해 전문가 시스템 기법을 도입하는데도 유리해지게 된다. 본 연구를 통해 개발된 객체지향형 언어로 개발된 class library 는 향후의 회분식 조업을 위한 scheduling S/W package 개발에 사용되어질 예정이다.