

# 반도체 Wafer Fabrication 공정에서의 Shift 단위 생산 일정계획

예승희\* · 김수영\*

## Shift Scheduling in Semiconductor Wafer Fabrication

Seung-Hee Yea · Soo-Young Kim

### 〈요 약〉

반도체 Wafer Fabrication 공정은 무수한 공정과 복잡한 Lot의 흐름 등으로 다른 제조 형태에 비해 효율적인 관리가 대단히 어려운 부문이다. 본 연구는 반도체 Fab을 대상으로 주어진 생산 소요량과 목표 공기를 효율적으로 달성하기 위한 Shift 단위의 생산 일정계획을 대상으로 하였다. 특히, 전 공정 및 장비를 고려하기보다는 Bottleneck인 Photo공정의 Stepper를 중심으로, 공정을 Layer단위로 묶어, 한 Shift에서 어떻게 Stepper를 할당하고 생산계획을 할 것인가를 결정하기 위한 2단계 방법론을 제시하고, Stepper 할당 및 계획에 필요한 3가지 알고리즘들을 제시하였다. 이 기법들을 소규모의 예제들에 대해 적용한 결과와 최적해와의 비교를 통하여 그 성능을 평가하였다.

〈주요어〉 Wafer Fabrication, 생산 일정계획, Shift Scheduling, Stepper 할당

### 1. 서론

반도체 제조는 다양하고 미세한 다수의 공정들이 반복성을 가지고 수행되며, Wafer가 투입되어 완성되기까지는 일반적으로 두달 여의 기간이 소요된다. 이러한 긴 공기(Cycle time 또는 Turnaround time)와 반복적 공정(Reentrant Process)의 특성 때문에 반도체 산업에서 생산 및 납기 관리는 어렵고 힘든 일이다. 반도체 제품은 크게 고객의 주문에 의한 생산을 하는 주문생산(Build to Order) 제품군(예를 들어, ASIC, Micro 제품들)과, DRAM 제품처럼 계획에 의해 생산하고, 생산된 제품을 판매하는 형식의 계획생산(Build to Plan) 제품군으로 나뉘어진다. 주문생산 제품의 경우 공기의 감축을 통한 고객 만족이 절실히 요구되며, 계

획생산 제품의 경우도 공기가 길게 되면 그만큼 판매 예측이나 수율(Yield)의 오차가 커지게 되어 많은 재공과 재고에 의한 부담을 안게 된다. 이제까지 대부분의 국내 반도체 기업들은 주로 양 위주의 생산량 극대화에 초점을 둔 생산관리 방식을 추구해 왔으나, 최근의 가격 하락과 고객 서비스에 대한 요구 등 새로운 기업 환경의 대두로, 재공 및 재고의 감축과 함께 고객 만족을 위한 공기 단축이 크게 주목을 받고 있다.

반도체 제조의 전체 공정을 크게 분류하여 Wafer Fabrication과 검사(Probe 또는 EDS)를 묶어서 Frontend 공정으로, 조립과 검사(Final test)를 하나로 Backend 공정으로 흔히 분류한다. 본 연구의 대상은 전체 제조공기에서 대부분을 차지하는 초기단계의 Fabrication

\* 포항공과대학교 산업공학과

공정(흔히 Fab이라 칭함)으로, Fab에서의 Shift단위 생산계획을 위한 새로운 방법론을 제시하고자 한다.

Fab 공정에서 Wafer는 Diffusion, Photo, Etch, Thin Film 등의 공정을 한번 거칠 때마다 하나의 Layer가 형성되고, 완전한 회로 완성을 위해서는 보통 10~16개의 Layer를 필요로 한다. 통상 Fab을 통과하는 전 공정을 구분하기 위해 각 Layer를 만드는 과정을 하나의 단위로 끊어, 이를 Layer 1, 2 등으로 구분하여 고려한다. 어떤 layer에서 재공이 과다하게 많은 경우에는 이 Layer에서의 공기가 계속 길어지게 되어 전체적인 계획보다 늦어지게 되며, 반대로 재공이 일정량보다 적을 경우, Production Rate가 떨어지게 된다. 그러므로 각 Layer에서 일정 수준의 재공을 유지하도록 계획을 세우는 것이 중요하다. 이때, 일정 수준의 재공의 양을 결정하는 것과 아울러 결정된 재공의 양을 유지하기 위한 방법이 필요하게 된다. 대부분의 Fab의 경우, 이들 공정 중에서 일반적으로 Photo공정이 Bottleneck공정으로 알려져 있으며, 이 Photo공정에서 사용되는 장비인 Stepper가 Bottleneck으로 규정되어진다. 상당수의 재공들이(일반적으로 전체의 30%이상) 주로 Photo공정에 존재하므로 이 Photo공정에서 Stepper의 할당 및 계획이 공기의 감소와 Line의 밸런스 유지에 중요한 관건이 된다.

반도체 산업에서 공기를 줄이기 위한 목적의 연구들은 주로 Shop Floor Control에 집중되어 왔다. Shop Floor Control에는 크게 두 가지의 방법이 존재하는데, 첫번째는 새로운 Lot을 어느 시점에서 투입할 것인가를 결정하는 투입정책(Release policy)과 각 장비에서 어떤 Lot을 어느 시점에서 가공하게 할 것인가를 결정하는 Scheduling 이나 Dispatching policy가 있다. Lot의 투입정책은 주어진 규칙에 의해서 Lot을 공정에 투입함으로써 안정적이고 짧은 공기를 가지게 하려는 것을 주 목적으로 한다. 반도체 Fab을 고려한 Glassey와 Resende[3]의 Starvation avoidance, Wein[11]의 Input regulation 등의 연구가 있다.

Scheduling 분야의 연구에서 기존의 연구들은 주로 Dispatching Rule에 집중되어 왔다. 상대적으로 일반 제조 형태에 대한 연구보다 반도체에 대한 연구는 적으며, 해외 연구는 Uzsoy 등[9][10]이 체계적으로 정

리하였다. 최근의 공기를 고려한 계획기법이 Lu 등[8]에 발표되었으며, 국내의 연구로는 이근희 등[7]과 조립을 대상으로 권일명 등[5]을 들 수 있다. 실제 선진 반도체 기업에서의 생산계획 현황은 Leachman[6]에 의해 발표되었다.

반도체 Wafer Fab에서 기존의 연구들은 언급한 바와 같이 주로 투입정책과 Scheduling에 집중되어 왔으며, Shift단위의 Scheduling은 미흡하였다. 실제 Fab에 있어서 생산계획의 단위는 Shift단위에 기본을 두고 있으며, 매 Shift마다 적절한 Schedule을 내주는 것이 중요하다. 따라서 본 연구에서는 목표로 하는 공기의 달성과 재공의 변동성을 낮출 수 있는 Shift단위의 Scheduling에 대한 대안을 제시하고, 이들의 비교분석을 통한 성능을 평가하였다. 본 논문에서는 Shift단위 Scheduling의 한 접근으로서 각 제품별 Layer별로 최적으로 생각되는 재공의 양을 결정하는 방법을 제시하고, 이러한 목표를 달성하기 위한 방법으로 Bottleneck 장비인 Stepper를 중심으로 Shift Scheduling의 방법론을 제시하고자 한다.

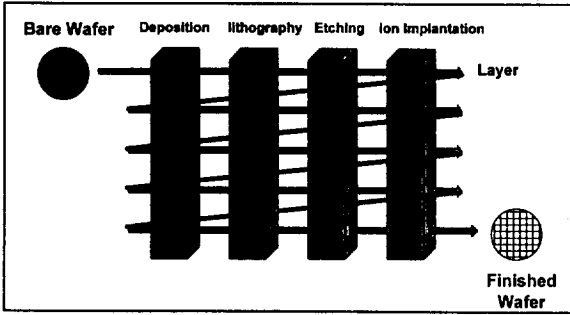
본 논문은 다음과 같이 구성된다. 먼저 다음 장에서 Wafer Fab공정에 대한 개략적인 설명과 함께 문제 설명을 하고, 3장에서는 제안하는 기법들에 대해 자세히 설명한다. 4장에서는 수치 실험을 통한 기법들의 비교 분석을 하고, 마지막으로 5장에서는 결론과 향후 연구과제에 대해 간단히 언급하였다.

## 2. 문제 설명

### 2.1 Wafer Fabrication 공정 개요

Fab공정은 반도체 제조에 있어서 가장 중요한 부분이며 가장 복잡하고 자본 집약적인 공정이다. Fab은 Wafer가 필요로 하는 전기적 특성을 가지는 회로형성을 위해 여러개의 Layer와 패턴이 만들어지는 공정으로 수백개의 세부 공정을 거쳐야 한다. Fab의 기본적인 공정의 흐름을 다음의 <그림 1>에서 나타내었다.

Fab공정이 시작되기 전에 Wafer는 연마공정을 통하여 회로 패턴을 그려넣기 위한 준비 작업을 하며 설계된 회로 패턴을 그릴 때 사용되는 Reticle (혹은



〈그림 1〉 Wafer Fab 공정의 흐름

Mask)이 제작된다. Wafer에 회로패턴을 그리는 작업은 사진 기술을 이용하는데 첫번째 단계로서 Wafer에 산화막을 형성하는 Deposition공정을 거치고, 감광액 도포(Photo Resist), 노광(Exposure), 현상(Development) 등의 일련의 Photo공정을 거친다. 그 후에 필요없는 부분을 선택적으로 제거시키는 식각(Etching)공정, 이온 주입(Ion Implantation)공정, 화학 기상증착(Chemical Vapor Deposition), 금속 배선(Metallization)공정 등 제품에 따라 필요한 공정을 거친다. 이와 같은 일련의 공정은 반도체 칩의 한 Layer를 형성하기 위해 거치는 한 Cycle로서 반도체의 Layer수만큼 Cycling되어 Fab공정을 끝낸다.

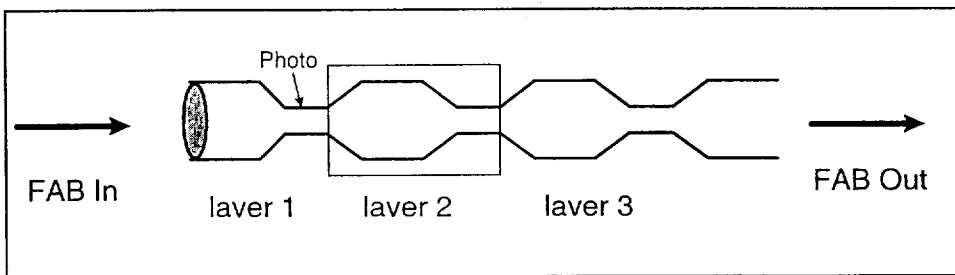
## 2.2 문제 정의

Fab의 수많은 공정을 일일이 열거하여 매 Shift의 생산계획을 세울 수도 있겠으나, 본 연구에서는 문제의 범위를 Bottleneck인 Photo공정의 Stepper를 중심으

로 한 일정계획 문제로 제한하였다. 이는 전체 재공의 분포를 실질적으로 결정하는 것이 Photo에서의 작업이고, 대형 메모리 Fab의 경우 Stepper장비만도 수십대가 되어, 이의 적절한 계획으로 전체 생산의 중요 흐름을 좌우하기 때문이다. 공정은 Layer를 기준으로 구분하며, Layer j에서의 재공의 양은 (j-1) Layer에서의 Photo 공정과 j번째 Photo 공정 사이에 존재하는 재공의 양으로 정의한다. 이것은 아래의 〈그림 2〉에서 나타내었다.

Wafer Fab에는 실제로 여러 종류의 Stepper가 존재하며, 제품/layer에 따라서 사용될 수 있는 Stepper의 종류는 미리 정해져 있으나, 어느 제품 어느 Layer에 몇 대를 할당할 것인가는 매 Shift초기에 결정하게 된다. 또한 Reticle의 수도 제한되어 있는데, 어떤 Stepper가 어떤 제품/layer를 생산하도록 할당되더라도 Reticle이 가용하지 않으면 실제 생산하지 못하고 Idle하게 되는 것이다.

따라서 Fab의 Shift 계획을 수행하기 위해서는 다음과 같은 의사결정이 요구된다. 첫째는 각 Stepper를 어떤 제품의 어떤 Layer에 할당할 것인가 이고, 둘째는 얼마나 오래 생산할 것인가이다. 이를 결정할 때 고려해야 하는 조건들로서는, 사용 가능한 Stepper의 종류, 수, 현재 Setup되어 있는 상황, Reticle의 수, 전체 재공현황 등이다. 계획 시에 추구하는 목표로는, Fab 생산 계획량의 달성, 전체적인 재공의 Balance 유지, 목표하는 공기의 달성 등을 들 수 있다. 추가로 고려되어야 할 것이 Stepper에서의 Setup이다. 다른 제품으로의 변경이나, 같은 제품이라도 다른 Layer로 변경하여 생산할 때 요구되는 Setup은 시간적인 Loss로 인



〈그림 2〉 Layer와 photo process의 관계

하여 생산량에 영향을 주게 된다. 따라서, 너무 잦은 변경은 생산량 손실을, 반대로 너무 적은 Setup은 재공의 편차를 크게 하고 공기 증가를 가져와 이의 적절한 고려도 포함되어야 한다. 공기에 대한 고려를 하기 위해서는 각 제품별로 달성을 원하는 목표 공기가 사전에 설정되었음을 가정하였다.

문제에 대한 이해를 돕기 위해 다음에서 예제를 통하여 간단히 설명한다.

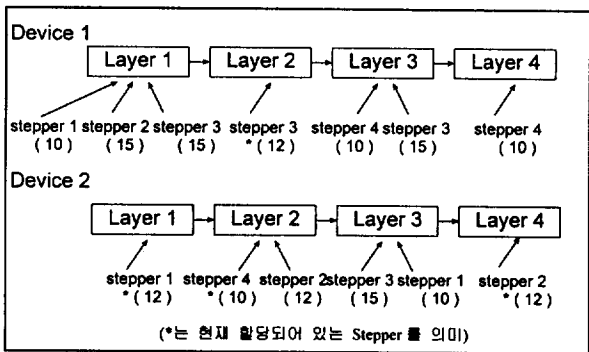
### 2.3 예제

예제에서 2개의 제품이 존재하고 각 제품은 4개의 Layer를 가지는 것을 가정하였다. 또, 각 제품/Layer에서 현재 가지고 있는 Layer와 Stepper의 재공량(Layer 재공은 Stepper 재공 + 전공정 재공)은 다음과 같이 가정하였다.

〈표 1〉 예제 Data

		Layer 1	Layer 2	Layer 3	Layer 4
제품 1	Layer 재공 (lot)	120	100	100	120
	Stepper 재공 (lot)	60	60	60	60
제품 2	Layer 재공 (lot)	90	100	200	100
	Stepper 재공 (lot)	60	60	60	60

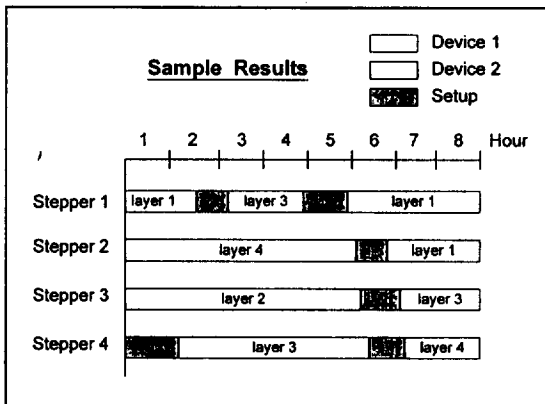
4개의 Stepper의 종류가 있다고 가정하고, 각 제품/layer에서 각 Stepper를 사용했을 때의 표준 가공 시간은 다음과 같이 가정하였다.



〈그림 3〉 Sample 문제의 구조

예를 들어, 위의 그림에서 Stepper 1, Stepper 2, Stepper 3은 제품 1의 Layer 1 공정에서 쓰일 수 있으며, Stepper 1을 사용했을 때 1 Lot을 가공하는데 10분이 걸리며, Stepper 2는 15분, Stepper 3은 15분이 걸린다.

이런 가정하에서 Shift Scheduling의 결과의 한 예는 다음과 같다.



〈그림 4〉 Shift Schedule의 결과 예

이 결과에서 볼 때, Stepper 1은 제품 1/ Layer 1을 1시간 30분 정도 생산한 다음 제품 1/ Layer 3을 생산하기 위해 Setup을 한 후, 2시간 정도 제품 1/ Layer 3을 생산하고, 다음으로 제품 2/ Layer 1을 생산하기 위해 Setup을 하고 나머지 시간 동안 제품 2/ Layer 1을 생산하도록 되어 있다.

### 3. 2단계 Shift Scheduling

본 논문에서는 Shift Scheduling의 방법으로 2단계 알고리즘을 제안한다. 첫번째 단계는 각 제품/Layer에서의 목표 재공의 양을 결정하고, 두번째 단계에서는 이번 Shift가 끝난 시점에서 실제 재공상황이 앞 단계에서 결정된 목표 재공에 근접할 수 있는 Stepper의 Schedule을 만들어 낸다. 이와 같은 2단계의 접근을 제시하는 이유는, 앞의 문제 설명에서 열거한 목표들과 제약을 모두 만족하기 위해 하나의 단계로 전체 계획을 수립하기란 어려우며, 현실적으로 적당한 목표 재공을 설정하고 이를 유지하기 위한 계획이 합리적인 현장에서 파악하였기 때문이다. 아래에서 각 단

계에 대한 자세한 설명을 한다.

$$\text{where } R_{ij} = \frac{(D_{ij} - P_{ij})}{8 \times N}$$

### 단계 1. 목표 재공의 결정

단계 1은 생산 수요량과 목표 공기 등을 고려해서 각 제품/Layer에서의 목표 재공의 양을 결정한다. 일반적으로 공기는 가공 시간과 이동 시간 및 대기 시간의 합으로 결정된다. 여기서 가공 시간은 고정되어 있는 값이고 대기시간은 재공의 양이 많을 경우엔 증가하고, 재공의 양이 적을 경우엔 감소하는 변수이다. 그러므로 공기를 최소한으로 줄이면서 수요량을 만족하고, Stepper의 사용률을 높일 수 있는 목표 재공의 양을 결정하고자 한다.

목표 재공의 양을 결정하는 방법을 제시하기 전에 먼저 본 논문에서 쓰이는 기호들을 다음과 같이 정의하였다.

#### Notation

$i$  : 제품 기호,  $i \in I$ ,  $I$ 는 FAB에 있는 모든 제품의 Set.

$j$  : Layer 번호,  $j = 1, \dots, J_i$ ,  $J_i$ 는 제품  $i$ 의 Layer 수.

$k_i$  : 제품  $i$ 의 상수

$CT_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 목표 공기.

$TCT_i$  : 제품  $i$ 의 Fab-in에서 Fab-out까지의 전체 목표 공기.

$TW_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 목표 재공.

$St_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 표준 가공 시간.

$\sigma_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 공기의 표준 편차.

$R_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 목표 Production Rate.

$D_{ij}$  : 이번 계획 구간에서 제품  $i$ 의 Layer  $j$ 에서의 수요량.

$P_{ij}$  : 이번 계획 구간에서 지금까지 제품  $i$ 의 Layer  $j$ 에서의 생산된 양의 합.

$N$  : 이번 계획 구간에서 남은 Shift의 수.

다음으로 목표재공의 양을 결정하는데 쓰이는 식들을 다음과 같이 정의하였다.

$$TW_{ij} = R_{ij} \times CT_{ij} \quad (1)$$

$$CT_{ij} = ST_{ij} + k_i \sigma_{ij} \quad (2)$$

$$TCT_i = \sum_j CT_{ij} \quad (3)$$

먼저 식 (1)에서 전통적인 Little's Law에 의해 각 제품의 Layer별 재공의 양을 해당 Layer의 공기와 생산률의 곱으로 표현했다. 식 (2)는 재고 모델에서 많이 쓰이는 식을 변형시킨 것인데 공기를 가공 시간과 여유 시간의 합으로 표현한 것이다. 즉, 각 제품/Layer에서의 공기는 필수 시간인 가공 시간외에도 대기 시간, 이동 시간을 포함하는데 고정된 가공 시간외에 나머지 여유 시간을 공기의 표준편차의 배수로 정의하였다. 마지막으로 식 (3)은 모든 Layer에서의 공기의 합은 미리 정의된 전체 공기와 같아야 한다는 식이다.

이 3가지의 식으로부터 각 제품/Layer에서의 목표 공기를 정하기 위해서 다음과 같은 과정을 수행한다.

**STEP 1.** (2)식과 (3)식을 이용해서 각 제품별로 상수  $k$ 를 결정한다.

**STEP 2.** Step 1에서 정의된 상수를 (2)식에 대입하여 제품/Layer별 목표 Cycle Time을 정한다.

**STEP 3.** 목표 공기를 (1)식에 대입하여 최종적으로 제품/Layer별 목표 재공의 양을 결정한다.

### 단계 2. Stepper Scheduling

단계 1에서 제품/Layer별로 목표 재공의 양이 결정되면 단계 2에서는 현재의 재공 상황을 고려해서 한 Shift가 끝났을 때 재공상황이 목표 재공에 근접하기 위한 Stepper Schedule을 세운다.

이러한 Stepper 할당 및 계획 방법으로 아래와 같은 3가지 방법을 제안한다.

#### (1) Linear Programming with Adjustment(LPA)

재공의 목표와의 편차를 줄이고, 가급적 Setup을 적게 하고자 하는 선형계획 Model을 세우고,

오차값을 보정하기 위한 Adjustment 알고리즘을 제안한다.

### (2) Heuristic Stepper Allocation(HSA)

(1)과 같은 목적식을 가지면서 간단한 휴리스틱에 의한 Stepper 할당 방법이다.

### (3) Minimum Setup Allocation(MSA)

실제 현장에서 쓰이는 Stepper 할당 방법과 유사하게 생산량을 극대화하기 위해 단순히 Setup time을 줄이는 것을 목적으로 하는 방법이다.

위의 3가지 Stepper 할당 방법을 설명하기 전에 단계 2에서 쓰이는 기호들을 다음과 같이 정리하였다.

#### Indices

$i$  : 제품 기호,  $i \in I$ ,  $I$ 는 FAB에 있는 모든 제품의 Set.

$j$  : Layer 번호,  $j = 1, \dots, J_i$ ,  $J_i$ 는 제품  $i$ 의 Layer 수.

$k$  : Stepper의 종류 번호,  $k \in K$ ,  $K$ 는 모든 Stepper 종류의 Set.

#### Parameters

$TW_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 목표 재공의 양.

$CW_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 실제 재공의 양.

$I_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 Photo공정에 있는 재공의 양.

$M_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 목표 생산량.

$D_i$  : 제품  $i$ 의 수요량.

$g_{ijk}$  : 제품  $i$ 의 Layer  $j$ 에서의 Stepper  $k$ 를 사용할 때의 표준 가공 시간.

$C_k$  : Stepper  $k$ 의 Capacity.

$S_{ijk}$  : Stepper  $k$ 에서 제품  $i$ 의 Layer  $j$ 를 생산할 때 걸리는 Setup시간.

$y_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 Yield.

$br_{ijk}$  : 이전 Shift에서의 Stepper Setting. 제품  $i$ 의 Layer  $j$ 에서 Stepper  $k$ 가 Setting되어 있으면 1, 아니면 0.

$w_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서의 중요도에 따른 Weight.

$R_j$  : 제품  $i$ 의 Layer  $j$ 에서 사용 가능한 Reticle의 수.

$Q$  : Big Number.

#### Variables

$r_{ijk}$  : Stepper  $k$ 의 전체 Capacity 중 제품  $i$ 의 Layer  $j$ 에 할당된 비율.

$P_{ijk}$  : Stepper  $k$ 가 제품  $i$ 의 Layer  $j$ 에 할당된 비율에 의한 이번 Shift에서의 생산량.

$IB_{ij}$  : 제품  $i$ 의 Layer  $j$ 에서 재공 Imbalance의 양.

$dr_{ijk}^+$  : 이전 Shift에서의 할당과 이번 Shift에서 Stepper  $k$ 가 제품  $i$ 의 Layer  $j$ 에서 할당된 양과의 차이 중 양수 부분.

$dr_{ijk}^-$  : 이전 Shift에서의 Setting과 이번 Shift에서 Stepper  $k$ 가 제품  $i$ 의 Layer  $j$ 에서 할당된 양과의 차이 중 음수 부분.

$d_{ijk}$  : 제품  $i$ 의 Layer  $j$ 에서 Stepper  $k$ 가 사용되는지 아닌지를 나타내는 Indicate Variable, 0 혹은 1 (사전에 결정됨).

#### 전처리 단계

Stepper Allocation 알고리즘에서 필요로 하는 목표 생산량을 정하기 위해 먼저 간단한 계산을 통해 단계 1에서 결정된 목표 재공을 고려하여 이번 Shift에서 얼마만큼의 양을 생산해야 할 것인가를 결정하게 된다.

목표 생산량을 정하기 위해 다음과 같은 식을 사용한다.

$$M_{i,J_i} = D_i \quad (4)$$

$$CW_{ij} - M_{ij} + y_{i,j-1} \times M_{i,j-1} = TW_{ij} \quad (5)$$

$$\text{If } M_{ij} < 0, \text{ Then } M_{ij} = 0$$

위에서  $D_i = (\text{현 기간내 계획된 Fab 생산 수요량} - \text{기 달성량}) / (\text{남은 Shift 수})$ .

먼저 마지막 Layer에서의 목표 생산량을 최종 수요량과 같다고 가정한다. 그리고 각 Layer에서의 목표 생산량은 Shift가 끝나는 시점에서 목표 재공과 실제 재공이 일치하기 위해 앞 Layer에서의 생산량과 연결되어 결정되어진다. 즉, Layer  $j$ 의 재공에서 Layer  $(j-1)$

에서 생산되어 넘어온 양을 더하고 그 Layer에서 생산해서 다음 Layer로 넘어가는 양을 빼 것이 목표 재공과 일치해야 한다는 가정에서 목표 생산량이 결정된다. 위의 계산은 마지막 Layer에서부터 시작하여 Layer를 거꾸로 올라오면서 계산하게 된다.

### 3.1 Linear Programming with Adjustment (LPA)

Stepper의 할당 문제는 Setup과 Reticle의 제약 등을 정확히 고려하기 위해서 정수계획법(Integer Programming)으로 모델링될 수 있다. 그러나 IP를 푸는 데 있어서의 시간적 경제적 제약 때문에 이 방법은 실제로 사용될 수가 없다. 따라서, IP 모델의 제약을 완화시킨 LP 모델을 통하여 근사적인 방법을 적용한 LPA 휴리스틱을 제안한다. LPA의 기본 개념은 IP만큼 정확한 식은 아니지만 근사치의 값을 낼 수 있게 LP로 정형화한 다음, 이 LP에서 나온 해를 Adjustment 알고리즘으로 값을 보정하는 것이다. 이 개념에 따라 기본적으로 LPA는 두 부분으로 나뉘게 되며, 첫번째는 LP를 풀어 초기해를 내는 과정이며, 두번째는 초기해를 보정하여 올바른 값을 계산한다.

#### Part 1. LP 모델

LP는 목표 재공과 비교했을 때의 재공의 편차를 최소화하고, Stepper의 Setup을 적게 하려는 목적식을 가진다. 앞서서도 밝혔듯이 실제 재공의 상황이 한 Shift가 끝났을 때 목표 재공에 근접하기 위한 Stepper schedule을 생성하는 것이 본 논문의 최종 목표이므로 첫번째 목적은 이를 반영한 것이라 하겠다. 두번째 목표인 Stepper setup의 최소화는 시간이 많이 걸리는 Stepper setup을 가급적 적게 하는 것이 생산량 측면에서 보았을 때 더 좋다고 할 수 있기 때문이다.

LP 모델은 다음과 같다.

#### Objective

$$\text{Min. } \sum_i \sum_j w_{ij} (IB_{ij}) + \sum_i \sum_j \sum_{k \in K_{ij}} \frac{C_k}{g_{ijk}} (dr_{ijk}^+ + dr_{ijk}^-)$$

#### Subject to:

(1) 할당률과 생산량의 정의

$$br_{ijk} \times C_k \times r_{ijk} + (1-br_{ijk})[(C_k - S_{ijk})r_{ijk}] = P_{ijk} \times g_{ijk},$$

for  $i \in I, j = 1, \dots, J_i, k \in K_{ij}$ .

(2) 할당률의 합에 대한 제약

$$\sum_i \sum_j r_{ijk} \leq 1, \text{ for } k \in K.$$

(3) 생산량 제약

$$\sum_{k \in K_{ij}} P_{ijk} \leq I_{ij}, \text{ for } i \in I, j = 1, \dots, J_i.$$

(4) 재공 편차의 정의

$$M_{ij} - \sum_{k \in K_{ij}} P_{ijk} \leq IB_{ij}, \text{ for } i \in I, j = 1, \dots, J_i.$$

(5) 전 Setup과 이번 Setup의 관계 정의

$$dr_{ijk}^+ - dr_{ijk}^- = r_{ijk} - br_{ijk}, \text{ for } i \in I, j = 1, \dots, J_i, k \in K_{ij}$$

(6) Non-negativity

all variables  $\geq 0$ .

목적식의 첫번째 항은 가중치를 부여한 재공의 목표 대비 편차를 의미하며, 두번째 항은 기존 Setup(0이나 1)과 새로운 할당률과의 차이(Stepper k가 ij에 Setup되지 않았으나 이번 Shift에 할당되면 양수 부분, 그 반대의 경우는 음수 부분의 값이 됨)에 Shift내 가능한 최대 생산량을 곱한 것이다. 그 의미는, 우선 정수형태의 변수를 배제한 상황에서 근사한 Setup의 수를 나타내기 위해서 할당률과 현재 Setup의 차이를 사용하였고, 그 차이에 생산량을 곱함으로써 첫 항과의 단위를 맞추어 준 것이다. 따라서, 두번째 항은 정확한 Setup의 수를 나타내는 것은 아니나, 현재의 Setup에서 이번 Shift의 할당이 달라질수록 그 값이 커지므로 근사하게 그 역할을 하게 된다.

(1)번 제약식은 각 제품/layer에 Stepper의 할당률과

그에 따른 생산량의 관계를 정의한다. Stepper의 Setup이 필요없는 경우는 Stepper의 총 생산 가능시간에서 할당된 시간 만큼 생산을 하는데 쓰게 되며, Stepper의 Setup이 필요한 경우에는 Stepper에 할당된 양은 생산하는 데 쓰이는 시간과 Setup에 드는 시간의 합이 된다. Setup이 필요한 경우 정확한 식이 되기 위해서는 정수를 도입해야 하는데, LP에서는 정확하게 그 관계를 표현하지 못하므로 Setup Time의 영향을 고려하기 위해 Setup이 필요할 경우의 Stepper의 할당율은 Stepper의 총 가용시간에서 Setup Time을 뺀 것의 비율이라고 정의하였다. 이는 정확하게 Setup이 필요한 경우 Setup Time을 제외한 나머지 가용시간을 고려해서 생산량을 정의하지는 못하므로, LPA의 두번째 부분인 Adjustment부분에서 이 값을 보정하게 된다.

제약식(2)는 Stepper의 이번 Shift동안 할당된 할당율의 합은 1보다 작아야 한다는 것을 표현하고, 제약식(3)은 각 제품/layer에서의 생산량이 Photo공정에서 가지고 있는 재공의 양보다 작아야 한다는 것이다. 제약식(4)와 (5)는 각각 목적식에서 쓰이는 재공 편차와 Setup차이를 정의하기 위해 사용되었다.

LP의 제약식에서 앞에서 설명한 Reticle에 대한 제약은 빠져 있다. 그 이유는 Reticle에 대한 제약을 고려하기 위해서는 정수값을 가지는 변수가 추가로 필요하게 되기 때문이다. 하지만 실제 문제에 있어서는 Reticle의 제약이 고려되어야 하므로, 다음의 Adjustment 알고리즘에서 Setup의 정확한 고려와 함께 휴리스틱한 방법으로 Reticle의 제약을 고려한다.

## Part 2. Adjustment 알고리즘

### 생산량의 재조정

앞에서 언급했듯이 LP에서 Stepper의 Setup이 필요한 경우 Setup시간을 정확히 고려해 주지 못한 결과로 LP의 결과는 실제보다 더 큰 생산량으로 계산한다. 그러므로 정확한 값을 갖게 하기 위해 다음 식을 사용해서 보정한다.

STEP 0. 예상 생산량의 값을 재조정.

For all  $i \in I, j = 1, \dots, J_i, k \in K_{ij}$

IF  $br_{ijk} = 0$

$$rp_{ijk} = \frac{C_k \times r_{ijk} - S_{ijk}}{C_k}$$

where  $rp_{ijk}$  = stepper allocation ratio to process

$$rs_{ijk} = \frac{S_{ijk}}{C_k}$$

where  $rs_{ijk}$  = stepper allocation ratio to setup

$$P_{ijk} = \frac{C_k \times rp_{ijk}}{g_{ijk}}$$

Reticle 제약을 고려한 Stepper의 재할당

위의 스텝에서 생산량의 값이 정확한 값으로 보정되었지만, 이는 Reticle제약식을 고려하지 않은 상황에서 나온 결과 이므로 이 결과를 초기해로 두고 모든 제품/layer에 대해서 Reticle을 고려해서 실제 사용 가능한지를 검증한 후, 만약 이를 어기면(사용 가능한 Reticle의 수보다 많은 Stepper가 할당되었으면) Reticle의 수와 할당된 Stepper의 수가 같을 때까지 그 제품/layer에 할당된 Stepper 중 하나를 선택해서 다른 제품/layer로 재할당한다.

STEP 1. Reticle 제약을 만족하는지를 검사.

For all  $i \in I, j = 1, \dots, J_i$

IF  $\sum_{k \in K_{ij}} d_{ijk} > R_{ij}$ , go to Step 2.

$$\text{where } d_{ijk} \begin{cases} = 1 & r_{ijk} > 0 \\ = 0 & \text{otherwise} \end{cases}$$

STEP 2. 다른 제품/layer로 재할당되어야 할 Stepper를 선택.

Select(k) if  $br_{ijk} = 0$  and

$$P_{ijk} > P_{ijl} \text{ for all } l \in K_{ij}$$



STEP 3. Step 2에서 선택된 Stepper가 사용될 수 있는 모든 제품/layer에 대해 더 할당될 수 있는 ratio를 계산.

For all (p,q), (p,q) ∈ S<sub>k</sub>  
 where S<sub>k</sub> = Set of all device/layer where stepper k can be used

$$\text{Set } ar_{pqk} = \frac{g_{pqk}(I_{ij} - \sum_k P_{pqk})}{C_k}$$

STEP 4. Stepper를 제공 Imbalance Weight가 큰 제품/layer에 할당

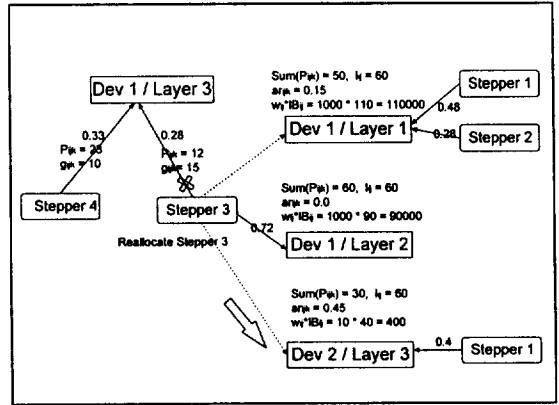
Select (m,n) whose (ar<sub>mnk</sub> > 0) and  
 (W<sub>mn</sub> × IB<sub>mn</sub>) > W<sub>pq</sub> × IB<sub>pq</sub>, (m,n) ∈ S<sub>k</sub>, for all (p,q) ∈ S<sub>k</sub>  
 allocation\_ratio = min[r<sub>ijk</sub>, ar<sub>mnk</sub>]  
 r<sub>mnk</sub> = r<sub>mnk</sub> + allocation\_ratio  
 r<sub>ijk</sub> = r<sub>ijk</sub> - allocation\_ratio

STEP 5. 종료 조건의 검사.

IF all ar<sub>pqk</sub> = 0 or r<sub>ijk</sub> = 0, go to Step 1.  
 ELSE go to Step 4.

STEP 1에서는 Reticle 제약이 만족하는지를 검사하고 만약 이를 어기면 STEP 2에서 생산량이 적은 Stepper가 재할당되어야 할 Stepper로 선택된다. STEP 3에서는 STEP 2에서 선택된 Stepper가 사용 가능한 제품/layer에 대해서 어느 정도 할당할 수 있는지를 계산하고, STEP 4에서 재할당될 제품/layer를 선택한 후 가능한 만큼 할당하게 된다. 이때, 재할당될 제품/layer는 제공 편차가 큰 것이 우선 순위를 갖게 된다. 이 알고리즘은 다음의 <그림 5>에서 예를 들어 설명하였다.

이 예제에서 우선 LP에 의한 결과로 제품 1/ layer 3에 Stepper 3과 4가 각각 할당되어 있음을 알 수 있다. 그러나 제품 1/ layer 3에서 사용 가능한 Reticle의 수는 1이어서 둘 중 한 Stepper는 어쩔 수 없이 생산을 할 수 없게 되므로 어느 한 Stepper를 다른 제



<그림 5> Adjustment 알고리즘의 예제

품/layer로 재할당하여 자원의 낭비가 없도록 해야 한다. 그런데 Stepper 3에 의한 생산량이 Stepper 4에 의한 생산량보다 적으므로 Stepper 3를 재할당해야 하는데, Stepper 3를 사용할 수 있는 제품/layer는 모두 3군데가 있다. 그 중에서 제공 편차가 110000으로 제일 큰 제품 1/ layer 1의 경우 0.15만큼을 더 할당할 수 있지만 Stepper 1과 Stepper 2가 먼저 할당되어 있고, 사용가능한 Reticle의 수가 2이므로 제품 1/ layer 1에 할당할 수 없다. 다음으로 제품 1/ layer 2가 제공 편차가 크지만, 더 이상 할당할 수 있는 제공이 없으므로 불가능하다. 마지막으로 제공 편차는 가장 작지만 할당할 수 있는 양이 0.45이고 또 할당했을 때 Reticle제약식을 깨뜨리지 않는 제품 2/ layer 3에 Stepper 3이 0.28만큼 할당된다.

### 3.2 Heuristic Stepper Allocation (HSA)

HSA 알고리즘은 LPA와 마찬가지로 제공 편차와 Setup을 최소화하는 것을 목적으로 한다. 그러나 LPA와는 다르게 LP를 사용하는 대신 휴리스틱한 방법으로 Stepper할당 Schedule을 결정한다.

STEP 0. 변수의 초기화.

$$\text{All } r_{ijk} = 0, P_{ijk} = 0.$$

STEP 1. 이전 Shift에서 미리 Setup된 제품/layer에 대해 Stepper의 Capacity를 할당.

For (i, j, k) whose  $br_{ijk} = 1$

$$\text{required\_ratio}(i, j, k) = \frac{g_{ijk}(\min[I_{ij}, M_{ij}])}{C_k}$$

$$r_{ijk} = \min[\text{required\_ratio}(i, j, k), 1]$$

$$P_{ijk} = \frac{C_k \times r_{ijk}}{g_{ijk}}$$

Modify  $M_{ij}, I_{ij}$

IF there is no more (i, j, k) whose  $br_{ijk} = 1$

Then, go to Step 2.

ELSE go to Step 1.

STEP 2. Setup되어 있지 않은 제품/layer 중 목표 생산량이 많은 제품/layer에 Stepper의 Capacity를 할당.

Find (p, q, k) whose  $I_{pq} \neq 0, \sum_k d_{pqk} < R_{pq}, M_{pq} \geq M_{ij}$   
for all  $i \in I, j = 1, \dots, J_i$ .

$$g_{pqk} \leq g_{pql} \text{ for all } l \in K_{pq}$$

IF (p, q, k)  $\neq$  NULL

$$\text{required\_ratio}(p, q, k) = \frac{g_{pqk} \times (\min[I_{ij}, M_{ij}])}{C_k}$$

$$r_{pqk} = \min[\text{required\_ratio}(p, q, k), \text{remained\_ratio}(k)]$$

$$\text{remained\_ratio}(k) = \text{remained\_ratio}(k) - r_{pqk}$$

$$P_{pqk} = \frac{C_k \times r_{pqk} - S_{pqk}}{g_{pqk}}$$

Modify  $M_{ij}, I_{ij}$ , go to Step 2.

ELSE

Stop !

먼저 STEP 0은 각 변수들의 초기화이고, STEP 1에 서는 현재 Setup되어 있는 제품/layer에 대해서 Stepper

를 가능한 많이 할당한다. 이는 Setup의 변화를 줄이기 위한 목적 때문이다. Setup된 모든 제품/layer에 대해 할당을 한 후, 남은 Capacity를 Setup이 필요한 제품/layer에 할당하게 된다. 이때, 목표 생산량이 많은 제품/layer가 우선권을 가지게 되는데 이는 Imbalance가 큰 것부터 먼저 그 차이를 줄이는 것이 중요하기 때문이다. 남은 Capacity가 없을 때까지 높은 우선권을 가지는 제품/layer부터 최대한 많은 Capacity를 할당해 간다.

### 3.3 Minimum Setup Allocation (MSA)

마지막으로 본 논문에서 다른 알고리즘과 비교를 위하여 현재 Fab에서 이루어지는 상황과 유사한 Stepper할당 규칙을 포함하였다. MSA는 Setup을 줄이는 것, 즉, 생산량을 최대화하는 것을 목적으로 하므로 매우 근시안적일 뿐만 아니라 재공 편차의 개념이 포함되어 있지 않은 알고리즘이다.

이 부분의 자세한 스텝별 서술은 생략하고, 전체적인 수순만을 소개한다. 우선, 초기화 단계에서 Stepper의 할당 순서를 정한다. 이것은 Stepper를 할당하는 순서에 따라 계획 결과가 상당한 차이가 있기 때문에, 가장 좋은 결과를 가져 올 수 있는 Stepper 할당 순서를 정한다. 가장 보편적인 기준으로, 현재 Setup되어 있는 제품/layer의 Photo 재공이 많은 Stepper가 우선 순위에 놓이도록 정하였다. 다음 단계는 앞에서 정의된 할당 순서에 따라 높은 우선순위를 가지는 Stepper부터 재공이 있는 제품과 Layer에 순차적으로 할당해 나간다. MSA는 Setup을 줄이는 것이 목표이므로 기존 Setup된 제품/layer에 먼저 최대한 많이 할당을 한다. 그리고 Capacity가 남을 경우에, 할당 대상 Stepper가 사용 가능하고, 재공이 남아있고, Reticle이 가용한 제품/layer 중에서 Setup time이 가장 작은 것을 선택하여 최대한 할당한다. 이런 식으로 Capacity가 남지 않을 때까지, 혹은 할당 가능한 제품/layer가 없을 때까지 할당을 한 후, 다음 대상 Stepper에 대해서 똑같은 과정을 거친다.

### 4. 비교 평가

본 논문에서 제시한 Stepper 할당 알고리즘을 2장에서 설명한 예제와 비슷한 구조를 갖는 다양한 문제에 대해 적용해 보았다. 물론 실제 Fab과 비슷한 문제에 대해 Stepper 할당 알고리즘을 적용해 보는 것이 가장 좋은 방법이나, 실제 크기의 문제에 대한 최적해를 도출할 수 없는 제약 때문에, 단계 2를 간단한 문제에 대해 적용해 보았다. 총 600개의 예제에 대해 각 알고리즘을 적용했으며 이를 Integer Programming으로 푼 최적해와 비교하였다.

본 논문에서 적용할 Sample들은 다음의 6가지 유형으로 나뉘어진다. 각 유형은 기본 문제에서 매 경우 한 가지 분야를 무작위로 변화시켜 각 경우에 대한 알고리즘들의 성능을 비교하였다.

1. Stepper Capacity를 변화시킨 경우
2. Photo 재공의 수를 변화시킨 경우
3. 목표재공의 수를 변화시킨 경우
4. Stepper Setting을 변화시킨 경우
5. Standard Time을 변화시킨 경우
6. 제품 Weight를 변화시킨 경우

각각의 시나리오에 대해 서로 다른 100가지의 문제를 random하게 생성하였으며, 이에 대해 제안한 3가지의 Stepper 할당 알고리즘을 적용했을 때의 결과를 <표 2>와 <표 3>에 나타내었다. 각 알고리즘의 결과는 최적해와의 % difference로 나타내었는데, 최적해는 MIP(Mixed Integer Programming)으로 풀어서 도출하였으며, % 차는 다음과 같이 계산되었다.

$$\% \text{ 차} = \frac{(\text{휴리스틱의 결과값} - \text{최적해})}{\text{최적해}} \times 100$$

<표 2>의 결과로 볼 때, LPA알고리즘은 최적해에 비해 평균적으로 5%의 차이를 보일 뿐으로 만족할 만한 결과를 보였다. 반면에 HSA 알고리즘은 평균 13.53%, MSA알고리즘은 34.68%의 차이를 보여서 LPA에 비해 많이 떨어지는 결과를 보이고 있다. 이 결과로 볼 때 현재 Fab에서 사용하는 Stepper 할당 방법을

<표 2> 휴리스틱별 각 유형에서 목적식 값에 의한 비교

유형	LPA (% 차)	HSA (% 차)	MSA (% 차)
1	8.70	4.61	20.95
2	2.37	9.90	61.04
3	13.05	42.64	53.09
4	1.84	10.10	12.07
5	2.16	4.83	29.55
6	1.65	9.03	31.39
평균	4.96	13.52	34.68

제안하는 LPA 알고리즘으로 바꿀 경우 전체적인 재공의 Balance면에 있어서 많은 향상을 이룰 수 있고, 따라서 적정 재공의 유지 및 공기의 감소라는 측면에서도 크게 향상을 가져올 것으로 기대된다.

다음으로 3가지 알고리즘을 실제 현장에서 중요하게 생각하는 생산량의 측면에서 MIP에 의한 값과 비교해 보았다. 목적식 값에 대한 비교와 마찬가지로 % 차에 의한 비교를 하였으며 다음의 <표 3>에 나타내었다.

<표 3> 휴리스틱별 각 유형에서 생산량에 의한 비교

유형	LPA (% 차)	HSA (% 차)	MSA (% 차)
1	-0.34	-9.81	-3.33
2	-1.32	1.21	2.69
3	-1.26	-2.07	-0.46
4	-0.40	-9.40	-2.27
5	-1.99	-6.00	-2.15
6	-3.47	-6.88	-1.74
평균	-1.46	-5.49	-1.21

위의 표에서 알 수 있듯이 MSA알고리즘이 MIP로 푼 결과에 비해 평균 1.21%의 차이(총 생산량은 더 많음을 의미)로 가장 좋은 결과를 내고 있는데, 이는 MSA알고리즘의 목적이 Setup time을 최소화해서 생산량을 최대화하는 것이므로 당연하다 하겠다. 하지만 LPA의 경우 평균 1.46%의 차이를 보이므로 MSA와 거의 유사한 결과라고 볼 수 있다. 또한, MSA의 결정적인 단점인 재공 편차를 상대적으로 크게 함에 따른 전체 재공의 상승과 공기의 증가는 약간의 생산

량 증가보다 더 많은 손실과 폐단을 유발한다. 따라서, 기존의 생산량 중심의 Setup만을 최소화하는 계획 형태에서 전체 재공의 평준화와 목표 공기의 안정적인 달성을 위한 계획이 되기 위해서는 위의 세 가지 기법중에서 LPA가 가장 효과적인 기법이라 하겠다.

## 5. 결론

본 논문은 반도체 Wafer Fabrication공정에서의 Shift 단위 생산 일정 계획에 대한 연구로, 기존의 생산량 중심의 계획과는 다른 측면에서 적정 재공과 공기의 달성을 위한 Stepper중심의 계획 문제를 다루었다. Shift 단위의 생산 일정 계획을 위해 2 단계의 방법을 제시하였으며, 첫번째 단계는 재공의 목표를 설정하고, 두번째 단계에서는 이번 Shift가 끝난 후 실제 재공 상황이 설정된 목표 재공에 근접하도록 하는 Stepper 할당 및 생산 계획을 결정하였다.

Stepper 할당 알고리즘으로 본 논문에서는 3가지의 알고리즘을 제시하였는데, 먼저 LP에 의한 초기해를 얻고 그 해를 Adjustment 알고리즘에 의해 보정하는 LPA알고리즘과, LP를 대신한 휴리스틱 해법으로 Stepper할당량을 결정하는 HSA 알고리즘, 그리고 실제 Fab에서의 상황을 유사하게 적용한 MSA알고리즘을 제시하였다.

3가지의 Stepper 할당 알고리즘을 다양한 경우에 대해 수치 실험을 한 결과 LPA알고리즘이 목적식 값에 있어서 최적해인 MIP에 비해 평균적으로 5%의 차이를 보였고, 생산량의 측면에 있어서도 1.5%의 차이로 실제 적용했을 경우 계산시간이나 결과에서 가장 뛰어난 성능을 보일 것으로 판단된다.

Shift단위 생산 일정 계획을 위해 본 논문에서 Stepper 할당 문제에 대한 연구를 수행하였지만 완전한 생산 일정 계획이 되기 위해서는 Stepper의 Schedule이 생성된 후 생산할 Lot의 순서 결정이 필요하다. 즉, 한 제품/layer에 할당될 Stepper가 정해진 후, 그 제품/layer에 속한 Lot중에서 어떤 것을 먼저 생산할 것인가를 결정해야 한다. 이는 기존의 Dispatching 방식과 같은 형태가 될 수도 있고, Just-In-Time 형태가 될 수도 있을 것이다. 전체적인 Fab의 생산계획은

상위 레벨의 Fab In/Out 계획, Shift단위 계획, 그리고 Dispatching으로, Lot 투입/Scheduling/Dispatching의 3 단계가 유기적 연계하에 이루어져야 하며, 이를 위한 체계적인 산학 연구가 앞으로도 많이 필요한 분야이다.

## 【참고문헌】

- [1] Bitran, G. R. and D. Tirupati, "Development and Implementation of a Scheduling System for a Wafer Fabrication Facility," Operations Research, Vol. 36, No. 3, pp. 377-395, 1988.
- [2] Blackstone, J. H., D. T. Phillips and G. L. Hogg, "A state-of-the-Art Survey of Dispatching Rules for Manufacturing Job Shop Operations," International Journal of Production Research, Vol. 20, No. 1, pp. 27-45, 1982.
- [3] Glassey, C. R. and G. C. Resende, "Closed-Loop Job Release Control for VLSI Circuit Manufacturing," IEEE Transactions on Semiconductor Manufacturing, Vol. 1, No. 1, 36-46, 1988.
- [4] Golovin, J. J., "A Total Framework for Semiconductor Production Planning and Scheduling," Solid State Technology, pp. 322-332, May, 1994.
- [5] 권일명, 최기범, 김성식, "반도체 조립 작업장의 실시간 작업관리 시스템," 산업공학, 8권, 4호, 5-19, 1995.
- [6] Leachman, R. C., "Production Planning and Scheduling Practices Across the Semiconductor Industry," ESRC Report 94-29, Engineering Systems Research Center, University of California, Berkeley, 1994.
- [7] 이군희, 홍유신, 김수영, "반도체 Wafer Fabrication 공정에서의 생산일정계획," 대한산업공학회지, 21권, 3호, 1995.
- [8] Lu, S. C. H., D. Ramaswamy and P. R. Kuma, "Efficient Scheduling Policies to Reduce Mean and Variance of Cycletime in Semiconductor Manufacturing Plants," IEEE Transactions on Semiconductor

Manufacturing, Vol. 7, No. 3, pp. 374-388, 1994.

- [9] Uzsoy, R., C. Y. Lee and L. A. Martin-Vega, "A Review of Production Planning and Scheduling Models in the Semiconductor Industry, Part I: System Characteristics, Performance Evaluation, and Production Planning," IIE Transactions, Vol. 24, No. 4., pp. 47-61, 1992.
- [10] Uzsoy, R., C. Y. Lee and L. A. Martin-Vega, "A Review of Production Planning and Scheduling Models in the Semiconductor Industry, Part II: Shop-Floor Control," IIE Transactions, Vol. 26, No. 5, pp. 44-55, 1994.
- [11] Wein, L. M., "Scheduling Semiconductor Wafer Fabrication," IEEE Transactions on Semiconductor Manufacturing, Vol. 1, No. 3, 115-130, 1988.



예승희

포항공과대학교 산업공학과에서 공학사(1995), 공학석사(1997) 학위를 취득하고, 현재 삼성자동차에서 근무중이다. 관심분야는 생산 및 재고관리, 생산시스템 분석 등이다.



김수영

포항공과대학교 산업공학과 부교수로 재직중으로, 서울대학교 기계공학 학사(1979), 한국과학기술원 석사(1981), 그리고 U.C. Berkeley 산업공학 박사(1988) 학위를 취득하였다. 1989년부터 1993년까지 미국 Rutgers 대학 산업공학과 조교수로 재직하였다. 주요 관심분야는 생산 및 재고관리, 반도체 생산계획 및 생산시스템 최적화, 그리고 기업 정보화 및 Process 혁신 등이다.