

Multi-Dimensional Dynamic Programming Algorithm for Input Lot Formation in a Semiconductor Wafer Fabrication Facility

June-Young Bang* · Seung-Kil Lim* · Jae-Gon Kim**†

*Sungkyul University, Gyeonggi-do, Korea

**Incheon National University, Incheon City, Korea

반도체 팹에서의 투입 로트 구성을 위한 다차원 동적계획 알고리즘

방준영* · 임승길* · 김재곤**†

*성결대학교 산업경영공학부

**인천대학교 산업경영공학과

This study focuses on the formation of input release lots in a semiconductor wafer fabrication facility. After the order-lot pegging process assigns lots in the fab to orders and calculates the required quantity of wafers for each product type to meet customers' orders, the decisions on the formation of input release lots should be made to minimize the production costs of the release lots. Since the number of lots being processed in the wafer fab directly is related to the productivity of the wafer fab, the input lot formation is crucial process to reduce the production costs as well as to improve the efficiency of the wafer fab. Here, the input lot formation occurs before every shift begins in the semiconductor wafer fab. When input quantities (of wafers) for product types are given from results of the order-lot pegging process, lots to be released into the wafer fab should be formed satisfying the lot size requirements. Here, the production cost of a homogeneous lot of the same type of product is less than that of a heterogeneous lot that will be split into the number of lots according to their product types after passing the branch point during the wafer fabrication process. Also, more production cost occurs if a lot becomes more heterogeneous. We developed a multi-dimensional dynamic programming algorithm for the input lot formation problem and showed how to apply the algorithm to solve the problem optimally with an example problem instance. It is necessary to reduce the number of states at each stage in the DP algorithm for practical use. Also, we can apply the proposed DP algorithm together with lot release rules such as CONWIP and UNIFORM.

Keywords : Input Lot Formation, Semiconductor Wafer Fabrication, Multi-Dimensional Dynamic Programming

1. 서 론

최근의 반도체 시장은 공급자에게는 경쟁이 심화되어

가고 있으며, 애플이나 델 등의 주요 고객들의 대량 구매력을 바탕으로 한 가격 협상력과 생산 및 공급 일정에 대한 영향력이 커져가고 있다. 또한, 메모리 반도체 치킨 게임이라 불리는 반도체 공급경쟁이 마무리되어 메모리 반도체 시장이 겨우 안정화되었으나, 중국의 공격적인 M&A로 촉발되는 2차 치킨 게임이 일어날 것으로 예상

Received 25 January 2016; Finally Revised 4 March 2016;

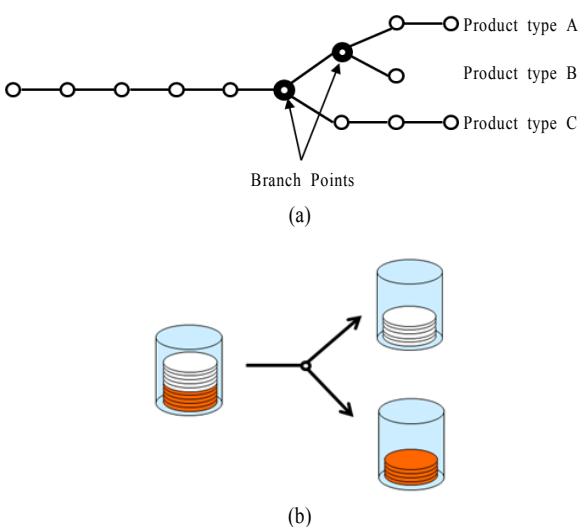
Accepted 5 March 2016

† Corresponding Author : jaegkim@inu.ac.kr

되고 있다. 이러한 어려운 시장 상황에서 살아남기 위해서는 반도체 설계 기술의 우위를 유지하는 동시에 반도체 생산 효율의 우위도 지켜가야 한다. 반도체 생산 효율 증대를 위한 반도체 공급망 관리에서 고객의 주문을 충족시키기 위한 생산 운영 기술은 가장 기본적이고 중요한 기술이나, 반도체 생산 공정의 복잡성으로 인해 난이도가 매우 높다.

메모리 반도체나 어플리케이션 프로세서(AP)와 같은 고사양 반도체는 일반적으로 다음과 같은 4단계의 공정을 거쳐서 생산한다. (1) 실리콘 웨이퍼에 회로를 생성하는 웨이퍼 fabrication 공정(이하, 웨이퍼 펌 공정); (2) 웨이퍼 상에 열/전기적 스트레스를 가한 후 웨이퍼 상 반도체 소자의 정상 동작 여부를 테스트하는 웨이퍼 테스트 혹은 웨이퍼 프로빙 공정; (3) 웨이퍼를 소자 단위로 절단하여 PCB에 접착 및 배선을 연결한 후 플라스틱 수지로 보호하는 조립 공정; (4) 반도체 칩을 다양한 온도에서 소비전력, 동작 속도를 측정하여 분류하는 테스트 공정. 반도체 펌 공정에서 95% 이상의 부가가치가 일어나고 30~40일의 생산 기간이 소요되므로 반도체 펌에서의 생산 효율의 증대가 제품의 생산 단가 경쟁력 증진에 가장 중요한 부분이다.

반도체 웨이퍼가 반도체 펌에 투입될 때 웨이퍼 제조사에서 공급하는 웨이퍼의 전기적 특성에 따라 1~3종의 웨이퍼가 투입되지만 공정이 진행됨에 따라 공정 조건이 상이하여 수십 종의 제품으로 분기된다. <Figure 1>(a)와 같이 생산되고 있는 웨이퍼 로트가 제품별로 분기점을 지나기 전까지는 동일 조건으로 각 세부 공정을 진행하나 제품 분기점을 지난 후에는 로트 내에 있는 웨이퍼들을 각각의 제품별로 로트를 나누어야 한다. 이러한 과정을 로트 분리(lot split)라고 한다(<Figure 1>(b) 참조).



<Figure 1> Illustration of Branch Points and Lot-Split Process

한편, 투입 로트를 구성하고 실제 펌에 투입하는 과정은 다음과 같다. 우선 주문-로트 페깅(Order-lot Pegging)을 통해서 고객 주문을 만족시키기 위하여 추가로 투입해야 하는 제품별 웨이퍼 수량을 계산한다. 이때, 제품별 투입 필요량에 따라 비용을 최소화하도록 투입 로트들을 구성해야 한다. 투입되는 로트의 유형은 단일 제품으로 구성된 로트, 유사성이 높은 제품군으로 구성된 로트, 적정 로트 사이즈를 만족하지 못하는 잔량에 대해서 소량 웨이퍼로 구성되는 부분 로트(partial lot)가 존재한다. 펌 공정 중 작업 조건(Recipe; 사용 설비, 작업 시간, 작업 시 소요되는 가스, 작업 온도 등)이 모두 동일한 경우에는 위의 두 번째 로트 유형과 같이 여러 제품 유형들을 하나의 로트로 구성하여 투입한다[2, 5]. 결과적으로 반도체 펌의 투입 단에서 투입 로트를 구성하는 상황에서 다음의 세 가지 경우를 고려할 수 있다.

- Case 1 : 단일 제품 종으로 25매 로트 구성(단일 제품 Full 로트 구성)
- Case 2 : 유사한 제품 2종 이상으로 25매 로트 구성(이종 제품 Full 로트 구성)
- Case 3 : 단일 제품 종만으로 25매 이하로 로트 구성(단일 제품 Partial 로트 구성)

일반적으로 펌내에서 생산하는 로트 수를 최소화하는 것이 생산 효율에 유리하다. 로트 1개당 장비에서의 소요 시간은 ‘로딩시간+언로딩시간+(웨이퍼 수×웨이퍼 당 작업시간)’으로 근사하여 계산할 수 있다. 또한 배치 장비에서는 ‘로딩시간+언로딩시간+로트 작업시간’으로 결정된다. 따라서 로딩/언로딩 시간은 로트의 수에 비례하고, 로트 수가 많아지면 펌 내 저장 공간을 차지하여 다음 공정으로 이동하는데 걸리는 물류시간도 증가하게 된다. 다종 제품으로 구성된 로트의 경우 로트가 <Figure 1>(a)와 같이 분기점을 지나는 경우, 로트의 분할을 하기 위해 소트 장비(wafer sorting)에서 약 30분에서 1시간 가량의 시간을 소모하게 되고, 이후 공정에서도 로트 수의 증가로 인한 추가적인 비용이 발생하게 된다. 따라서, Case 1의 경우가 생산 비용이 가장 적게 발생하고, Case 2, Case 3 순으로 생산 비용이 커진다. Case 2의 경우에는 다수의 고객과 연결된 로트이므로 높은 수준의 관리가 필요하다. 따라서 주요 공정이 종료될 때마다 웨이퍼에 이상이 없는지 확인하기 위해 Case 1의 경우보다 자주 검사(Quality Inspection)를 시행해야 하고, 분기점 이후에는 두 개 이상으로 로트로 분리하여 나머지 공정을 진행하므로 제품의 분기나 잦은 검사가 필요 없는 Case 1보다 큰 비용이 발생하게 되는 것이다. 또한, Case 3과 같이 처음부터 로트를 분리하여 생산하는 경우에는 생산 시간 및 물류 시간

이 더욱 증가하여 Case 2보다 큰 비용이 발생하게 된다. 따라서 비용을 줄이기 위해서는 투입 로트를 구성할 때, 로트 수를 최소화하도록 하나의 로트를 최대한 동일 제품으로 구성하거나 함께 구성되는 제품 종류의 수를 최소화해야 한다.

반도체 팹 내의 로트를 주문에 할당하고, 주문을 만족시키는 투입 필요량을 결정하는 주문-로트 할당 문제(Lot-Order Assignment Problem; LOAP)에 대한 연구는 반도체 전체 공정의 후반부 공정인 조립 및 테스트 공에 대해서 주로 수행되어 왔다[3, 4, 6, 13]. 최근 들어서는 반도체 팹에서의 LOAP에 대한 연구들이 이루어지고 있다. Wu [17]와 Bang et al.[1]는 반도체 팹 내에서의 불확실한 상황을 휴리스틱 기반의 빠른 할당을 통해 일정 기간마다 반복하는 소프트 폐경 방식을 제안하고 방법론을 제시하였다. Kim and Lim[8]과 Lim et al.[11]은 수리적 모델을 기반으로 최적 할당을 도출하였다. Kim et al.[9]은 LOAP를 단일 설비에서 작업시간이 작업순서에 따라 변경되는 일정계획 문제로 변환될 수 있음을 보였고, 이 문제에 대한 휴리스틱 방법론을 제시하였다. Seo and Bang[14]은 반도체 팹에서 조립 및 테스트 공정 전체에 대한 LOAP에 대해서 생산 가용량을 고려한 알고리즘을 제시하여 폐경 단계에서 로트의 생산 완성 시점을 예측하여 실제 적용 사례를 보였다.

한편, 주문-로트 할당의 결과로 신규 투입 필요량에 따라 신규 로트를 구성한 후에는 팹에 투입해야 하는 로트의 순서와 시점을 결정해야 한다. 로트 투입 문제에 대해서는 크게 네 가지의 연구가 대표적이다. 일정한 간격으로 로트를 투입하는 UNIF(uniform release rule)[7], 팹 내의 전체 로트 수가 일정하도록 투입하는 CONWIP(constant WIP rule)[15], 병목 공정의 대기 로트의 총 작업시간이 일정 수준 이하로 낮아진 경우 신규 로트를 투입하는 WR (workload regulating rule)이다[16]. 위와 같이 주문-로트 할당 문제인 폐경 문제와 구성된 투입 로트의 투입 시점의 결정 문제에 관하여는 많은 연구가 진행되어 왔으나, 투입 로트 구성에 관한 연구는 현재까지 본 연구가 유일하다.

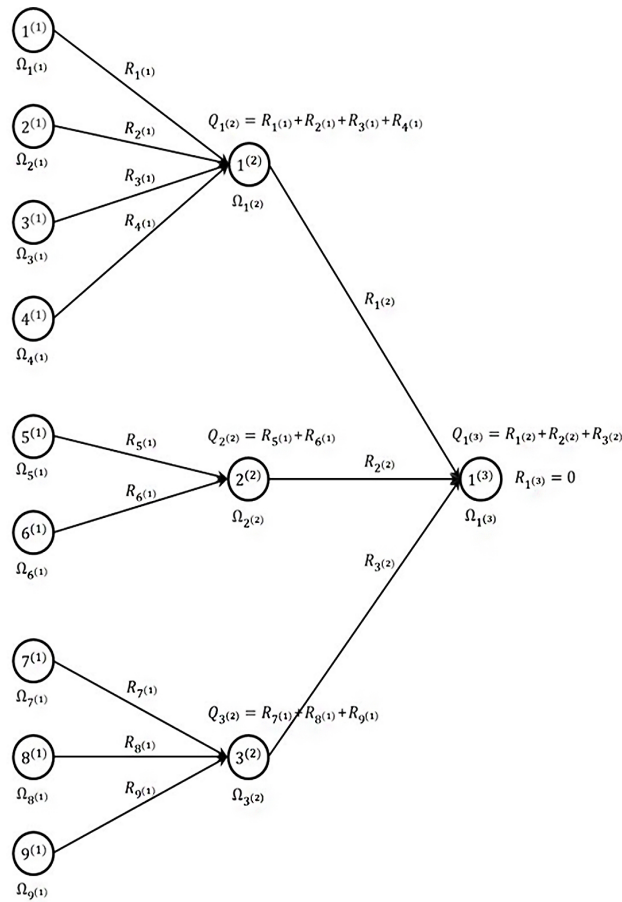
본 연구에서는 반도체 팹에서 주문-로트 폐경의 결과로 투입 필요량이 주어졌을 때, 반도체 팹으로 투입하기 위한 투입 로트를 구성하는 문제(Input Lot Formation Problem; ILFP)를 다루었다. 투입 로트의 생산 비용은 로트를 구성하는 제품 종류의 수나 제품의 유사성에 따라 n 개의 수준으로 등급화되어 있고, 이 비용들은 생산 원가 분석을 통해 사전에 주어져 있다. ILFP의 최적해를 도출하기 위하여 다차원 동적 계획 알고리즘을 개발하였고, 실제 문제를 간소화하여 간단한 예제를 만들고 알고리즘이 작동하는 과정을 설명하였다.

2. 투입 로트 구성 문제

이제 본 연구에서 다루는 반도체 팹에서의 투입 로트 구성 문제(Input Lot Formation Problem; ILFP)에 대해 자세히 설명하고자 한다. 다음은 사용되는 기호들에 대한 설명이다.

파라미터

- n 이질성 수준을 나타내는 기호($n = 1, 2, \dots, N$)(높은 이질성 수준을 가지는 로트는 생산 과정에서 다양한 제품 유형들로 나뉘어지며 이로 인해 생산비용 또한 커진다. 보다 단순한 표현을 위해 이후에서는 ‘이질성 수준’을 간략히 ‘수준’이라고만 표기하고자 한다).
- $l^{(n)}$ 수준 n 에 포함된 제품 유형 그룹들을 나타내는데 사용하는 기호(수준 n 에서의 하나의 제품 유형 그룹은 수준 $n-1$ 에서의 여러 제품 유형 그룹들을 묶어서 정의된다. 여기서, 수준 1에서의 제품 유형인 $l^{(1)}$ 은 아직 그룹으로 묶이지 않은 원래의 개별 제품 유형을 나타낸다. 보다 단순한 표현을 위해 이후에서는 ‘제품 유형 그룹’을 ‘그룹’이라고만 표기하고자 한다).
- $\Lambda^{(n)}$ 수준 n 에 포함되는 모든 그룹들의 집합(여기서, $\Lambda^{(1)}$ 은 아직 그룹으로 묶이지 않은 원래의 개별 제품 유형들의 집합이다).
- $\Delta_{l^{(n)}}$ 수준 n 의 그룹 $l^{(n)}$ 을 구성하는 수준 $n-1$ 에서의 그룹들의 집합(여기서, 모든 n 에 대하여 수준 n 에 속하는 그룹 $l_1^{(n)}$ 과 $l_2^{(n)}$ 가 $l_1^{(n)} \neq l_2^{(n)}$ 이면 $\Delta_{l_1^{(n)}} \cap \Delta_{l_2^{(n)}} = \phi$ 이고 $\cup_{l^{(n)}} \Delta_{l^{(n)}} = \Lambda^{(n-1)}$ 이다. 이때, 수준 $n-1$ 의 임의의 그룹은 수준 n 의 오직 하나의 그룹에만 포함된다).
- $W_{l^{(n)}}$ 그룹 $l^{(n)}$ 에 대해 로트를 구성할 때의 최대 로트 사이즈 제한
- $D_{l^{(n)}}$ 그룹 $l^{(n)}$ 에 대해 로트를 구성할 때의 최소 로트 사이즈 제한
- $C_{l^{(n)}}^U$ 그룹 $l^{(n)}$ 에 대해 구성된 로트에 포함된 웨이퍼 1매당 단위 생산 비용(여기서, 수준 n 이 클수록 단위 생산 비용은 증가한다. 다만, 동일 그룹에 속한 서로 다른 제품 유형들의 단위 생산 비용은 동일하다고 가정한다).
- $Q_{l^{(n)}}$ 그룹 $l^{(n)}$ 에 대해 로트들로 구성해야 하는 웨이퍼 총 매수
- $\Omega_{l^{(n)}}$ 그룹 $l^{(n)}$ 에 대해 로트들로 구성된 웨이퍼 총 매수
- $R_{l^{(n)}}$ 그룹 $l^{(n)}$ 에 대해 수준 n 에서 로트들로 구성되지 못한 웨이퍼 총 매수(여기서, $R_{l^{(n)}} = Q_{l^{(n)}} - \Omega_{l^{(n)}}$ 이며 $Q_{l^{(n)}} = \sum R_{l^{(n)}}$ 이다. 또한, $R_{l^{(0)}} = Q_{l^{(n)}}$ 이다).



Group	Level 1									Level 2			Level 3
	1 ⁽¹⁾	2 ⁽¹⁾	3 ⁽¹⁾	4 ⁽¹⁾	5 ⁽¹⁾	6 ⁽¹⁾	7 ⁽¹⁾	8 ⁽¹⁾	9 ⁽¹⁾	1 ⁽²⁾	2 ⁽²⁾	3 ⁽²⁾	1 ⁽³⁾
Product Type	1	2	3	4	5	6	7	8	9	1 2 3 4	5 6 8 9	7 8 9	1, 2 3, 4 5, 6 7, 8 9
Minimum Lot size	20	20	20	20	20	20	20	20	20	20	20	20	1
Maximum Lot size	25	25	25	25	25	25	25	25	25	25	25	25	25

<Figure 2> Example of Three Level Input Lot Formation Problem

<Figure 2>는 총 9개의 제품 유형들에 대한 투입 로트 구성 문제의 예를 보여주고 있다. 수준 1에서는 동일한 제품 유형으로 만들어질 웨이퍼들로만 로트들이 구성되며, 수준 2에서는 제품 유형 1, 2, 3, 4로 나뉘어지는 웨이퍼들로 로트를 구성할 수도 있고, 제품 유형 5, 6으로 나뉘어지는 웨이퍼들로 로트들을 구성할 수도 있으며, 제품 유형 7, 8, 9로 나뉘어지는 웨이퍼들로 로트들을 구성할 수도 있다. 마지막으로 수준 3에서는 9개의 제품 유형들로 나뉠 수 있는 웨이퍼들로 로트들을 구성할 수 있다. 수준 1에서 구성된 로트들보다 수준 2에서 구성된 로트들에 포함된 웨이퍼 1매당 생산 비용이 크며, 마친

가지로 수준 3에서 구성된 로트들에 포함된 웨이퍼 1매당 생산 비용이 수준 2에서 구성된 로트들에 포함된 웨이퍼보다 더 크다. 수준 1과 수준 2에 대하여 최소 로트 사이즈는 20매, 최대 로트 사이즈는 25매로 주어져 있고, 수준 3에서는 최소 로트 사이즈는 1매, 최대 로트 사이즈는 25매이다. 한편, 이질성 수준이 높아짐에 따라 단위 생산 비용이 증가하므로 로트 구성은 수준 1에서부터 수준 3까지 순차적으로 이루어진다. 즉, 로트 사이즈에 대한 최소 및 최대 제약을 만족하며 수준 1에서 제일 먼저 로트들을 구성하고, 수준 1에서 로트로 구성되지 못한 웨이퍼들을 수준 2에서 다시 로트들로 구성하며, 수준 2에서도 로트들로 구성되지 못한 웨이퍼들로 수준 3에서 마지막으로 로트들을 구성하게 된다.

투입 로트 구성 문제의 목적은 총 생산 비용의 합을 최소화하는 것으로 총 생산 비용은 $\sum_{n=1}^N \sum_{l^{(n)} \in A^{(n)}} C_{l^{(n)}}^U \Omega_{l^{(n)}}$ 과 같이 표현된다. 본 연구에서는 각 수준에서의 로트 사이즈 제약을 만족하면서 총 생산 비용을 최소화하는 투입 로트 구성을 만들 수 있는 다차원 동적계획 알고리즘을 개발하였다. Lim et al.[12]은 3수준으로 제한된 상황에서 단위 생산 비용이 증가하는 폭이 수준이 증가함에 따라 점점 감소하는 구조를 가지는 특수한 문제에 대한 최적해를 개발하고 다량 우편물 선 구분 문제에 적용하는 연구를 수행한 바 있다. 본 연구는 임의의 N개의 수준에 대해서 적용 가능한 동적계획법을 개발하여 반도체 팹 투입 로트 구성 문제에 적용하였다는 점에서 이전 연구와 큰 차이를 가진다고 할 수 있다.

3. ILFP를 위한 다차원 동적계획 알고리즘

동적계획 알고리즘을 개발하기 위해서는 먼저, 단계들(Stages)과 각 단계에서의 상태(State), 그리고 각 단계에서의 가능한 대안들(Possible Alternatives)을 정의해야 한다. ILFP 문제에서는 수준 n을 동적계획법의 단계로, 단계 n에서의 상태를 수준 n의 그룹 l⁽ⁿ⁾에 대해 수준 n에 이를 때까지(즉, 수준 n-1까지 로트를 구성하였을 때) 아직 로트들로 구성되지 못한 웨이퍼 총 매수 즉, R_{1⁽ⁿ⁾}, R_{2⁽ⁿ⁾}, ..., R_{l⁽ⁿ⁾}로 정의할 수 있다. 또한, 단계 n에서의 가능한 대안은 최대 및 최소 로트 사이즈 제약을 충족하는 그룹 l⁽ⁿ⁾에 대해 수준 n에서 로트들로 구성된 웨이퍼 총 매수 즉, Ω_{1⁽ⁿ⁾}, Ω_{2⁽ⁿ⁾}, ..., Ω_{l⁽ⁿ⁾}로 정의할 수 있다. 이제, 정의한 단계, 상태, 가능한 대안들로부터 동적 계획을 위한 순환관계식(DP recursive equations)을 만드는데 사용하는 기호들을 설명하고자 한다.

$$\Psi_n(R_{1^{(n)}}, R_{2^{(n)}}, \dots, R_{l^{(n)}}):$$

수준 n 에서 로트로 구성되지 못한 각 그룹 별 웨이퍼 수량이 $R_{1^{(n)}}, R_{2^{(n)}}, \dots, R_{l^{(n)}}$ 일 때, 수준 1에서부터 수준 n 까지 로트로 구성된 웨이퍼들에 대한 최소생산비용

$$\Gamma_n(\Omega_{1^{(n)}}, \Omega_{2^{(n)}}, \dots, \Omega_{l^{(n)}}):$$

수준 n 에서 로트로 구성한 각 그룹 별 웨이퍼 수량이 $\Omega_{1^{(n)}}, \Omega_{2^{(n)}}, \dots, \Omega_{l^{(n)}}$ 일 때, 수준 n 에서 로트로 구성된 웨이퍼들에 대한 생산비용(앞서 언급하였듯이 $\Gamma_n(\Omega_{1^{(n)}}, \Omega_{2^{(n)}}, \dots, \Omega_{l^{(n)}}) = \sum_{l^{(n)} \in A^{(n)}} C_{l^{(n)}}^U \Omega_{l^{(n)}}$ 이 된다).

ILFP를 위한 순방향 동적 계획 순환관계식(Forward DP recursive equations)은 식 (1)~식 (3)과 같다.

$$\begin{aligned} & \min \\ \Psi_1(R_{1^{(1)}}, R_{2^{(1)}}, \dots, R_{l^{(1)}}) &= \begin{matrix} \Omega_{l^{(1)}} = Q_{l^{(1)}} - R_{l^{(1)}} \text{ for all } l^{(1)} \\ \text{Lots from } \Omega_{l^{(1)}} \text{ should satisfy} \\ \text{the lot size constraints.} \end{matrix} \Gamma_1(\Omega_{1^{(1)}}, \Omega_{2^{(1)}}, \dots, \Omega_{l^{(1)}}) \end{aligned} \quad (1)$$

$$\begin{aligned} & \min \\ \Psi_n(R_{1^{(n)}}, R_{2^{(n)}}, \dots, R_{l^{(n)}}) &= \begin{matrix} \Omega_{l^{(n)}} = \sum_{l^{(n-1)} \in \Delta_{l^{(n)}}} R_{l^{(n-1)}} - R_{l^{(n)}} \\ \text{for all } l^{(n)} \text{ Lots from } \Omega_{l^{(n)}} \text{ should satisfy} \\ \text{the lot size constraints.} \end{matrix} \Gamma_n(\Omega_{1^{(n)}}, \Omega_{2^{(n)}}, \dots, \Omega_{l^{(n)}}) \\ & + \Psi_{n-1}(R_{1^{(n-1)}}, R_{2^{(n-1)}}, \dots, R_{l^{(n-1)}}) \quad \text{for } n = 2, 3, \dots, N-1 \end{aligned} \quad (2)$$

$$\begin{aligned} & \min \\ \Psi_N(R_{1^{(N)}}) &= \begin{matrix} \Omega_{1^{(N)}} = \sum_{l^{(N)} \in \Delta_{1^{(N)}}} R_{l^{(N-1)}} \\ \text{Lots from } \Omega_{1^{(N)}} \text{ should satisfy} \\ \text{the lot size constraints.} \end{matrix} \Gamma_N(\Omega_{1^{(N)}}) \\ & + \Psi_{N-1}(R_{1^{(N-1)}}, R_{2^{(N-1)}}, \dots, R_{l^{(N-1)}}) \end{aligned} \quad (3)$$

위의 순환관계식을 이용하여 최적해를 구하면 최적 목적 함수 값은 $\Psi_N(R_{1^{(N)}}) = \Psi_N(0)$ 이 된다. 위의 관계식은 $R_{1^{(N)}} = 0$ 을 가정하고 있다. 즉, 마지막 수준 N 에서는 오직 하나의 그룹만 존재하고 로트로 구성되지 않고 남은 웨이퍼는 하나도 없다는 것이다.

4. 다차원 동적계획 알고리즘의 적용 예제

이제 앞서 개발한 동적 계획 순환관계식을 적용하는 과정을 예제를 통해 살펴보고자 한다. <Figure 2>과 동일한 3수준의 ILFP에 대한 예제에 사용되는 데이터가 <Table 1>에 주어져 있다. 9개 제품 유형 별로 투입 로트를 구성해야 할 수량과 각 수준에서 로트를 구성하였을 때의 단위 당 생산 비용이 맨 아래 두 줄에 주어져 있다. 앞서 설명한 바와 같이 이질성수준이 높은 로트들에 포함된 웨이퍼들에 대해서는 더 많은 생산 비용이 발생한다.

<Table 1> Problem Data

Group	Level 1									Level 2			Level 3
	1 ⁽¹⁾	2 ⁽¹⁾	3 ⁽¹⁾	4 ⁽¹⁾	5 ⁽¹⁾	6 ⁽¹⁾	7 ⁽¹⁾	8 ⁽¹⁾	9 ⁽¹⁾	1 ⁽²⁾	2 ⁽²⁾	3 ⁽²⁾	1 ⁽³⁾
Product Type	1	2	3	4	5	6	7	8	9	1	5	7	1,2
										2	6	8	3,4
										3	9	9	5,6
										4			7,8
													9
Minimum Lot size	20	20	20	20	20	20	20	20	20	20	20	20	1
Maximum Lot size	25	25	25	25	25	25	25	25	25	25	25	25	25
Unit Production Cost	105	120	110	105	100	110	110	105	100	130	120	125	140
Input Quantity	28	38	18	30	27	24	31	15	21	-	-	-	-

<Table 2>는 단계 1에서의 순환관계식 계산을 보여주고 있다. 단계 1에서 가질 수 있는 상태 $(R_{1^{(1)}}, R_{2^{(1)}}, \dots, R_{9^{(1)}})$ 은 투입 로트를 구성해야 할 수량($\Omega_{1^{(1)}}, \Omega_{2^{(1)}}, \dots, \Omega_{9^{(1)}}$)으로 부터 구할 수 있다. 예를 들어 제품 유형 1의 경우, 28개의 웨이퍼들을 투입해야 하는데 단계 1에서는 최소 20개에서 최대 25개를 하나의 로트로 구성할 수 있기 때문에 $R_{1^{(1)}}$ 은 3에서부터 8까지의 값을 가질 수 있게 된다. 마찬가지로 제품 유형 2의 경우에는 $R_{2^{(1)}}$ 가 13에서 18까지 가능하며 $R_{3^{(1)}}$ 의 경우에는 로트 사이즈 제약을 만족하는 로트를 단계 1에서는 구성할 수 없기 때문에 오직 18만을 상태로 가질 수 있다. 또한, 앞서 언급한 대로 $(\Gamma_1(\Omega_{1^{(1)}}, \Omega_{2^{(1)}}, \dots, \Omega_{9^{(1)}}))$ 이 주어질 때 단계 1에서의 생산 비용은 다음과 같이 계산된다 :

<Table 2> DP Calculations at Stage 1

$(R_{1^{(1)}}, R_{2^{(1)}}, \dots, R_{9^{(1)}})$	$(\Omega_{1^{(1)}}, \Omega_{2^{(1)}}, \dots, \Omega_{9^{(1)}})$	$\Gamma_1(\Omega_{1^{(1)}}, \Omega_{2^{(1)}}, \dots, \Omega_{9^{(1)}})$	$\Psi_1(R_{1^{(1)}}, R_{2^{(1)}}, \dots, R_{9^{(1)}})$
(3, 13, 18, 5, 2, 0, 6, 15, 0)	(25, 25, 0, 25, 25, 24, 25, 0, 21)	$(25 \times 105) + (25 \times 120) + \dots + (21 \times 100) = 18,240$	18,240
(3, 13, 18, 5, 2, 0, 6, 15, 1)	(25, 25, 0, 25, 25, 24, 25, 0, 20)	$(25 \times 105) + (25 \times 120) + \dots + (20 \times 100) = 18,140$	18,140
...
(8, 18, 18, 10, 7, 4, 11, 15, 1)	(20, 20, 0, 20, 20, 20, 20, 0, 20)	$(20 \times 105) + (20 \times 120) + \dots + (20 \times 100) = 15,000$	15,000

<Table 3> DP Calculations at Stage 2

$(R_{1(1)}, R_{2(1)}, \dots, R_{9(1)})$	$(R_{1(2)}, R_{2(2)}, \dots, R_{3(2)})$	$(\Omega_{1(2)}, \Omega_{2(2)}, \Omega_{3(2)})$	$\Gamma_2(\Omega_{1(2)}, \Omega_{2(2)}, \Omega_{3(2)}) + \Psi_1(R_{1(1)}, R_{2(1)}, \dots, R_{9(1)})$
(3, 13, 18, 5, 2, 0, 6, 15, 0)	(14, 2, 0)	(25, 0, 21)	$(130 \times 25) + (125 \times 21) + 1,8240 = 24,115$
	(14, 2, 1)	(25, 0, 20)	$(130 \times 25) + (125 \times 20) + 18,240 = 23,990$

(3, 13, 18, 5, 2, 0, 6, 15, 1)	(19, 2, 1)	(20, 0, 20)	$(130 \times 20) + (125 \times 20) + 18,240 = 23,340$
	(14, 2, 0)	(25, 0, 22)	$(130 \times 25) + (125 \times 22) + 18,140 = 24,140$
	(14, 2, 1)	(25, 0, 21)	$(130 \times 25) + (125 \times 21) + 18,140 = 24,015$
...
(3, 14, 18, 5, 2, 0, 6, 15, 0)	(19, 2, 2)	(20, 0, 20)	$(130 \times 20) + (125 \times 20) + 18,140 = 23,240$
	(15, 2, 0)	(25, 0, 21)	$(130 \times 25) + (125 \times 21) + 18,120 = 23,995$
	(15, 2, 1)	(25, 0, 20)	$(130 \times 25) + (125 \times 20) + 18,120 = 23,870$
...
(8, 18, 18, 10, 7, 4, 11, 15, 1)	(19, 2, 1)	(21, 0, 20)	$(130 \times 21) + (125 \times 20) + 18,120 = 23,350$
	(4, 11, 2)	(50, 0, 25)	$(130 \times 50) + (125 \times 25) + 15,000 = 24,625$
	(4, 11, 3)	(50, 0, 24)	...
...
	(34, 11, 7)	(20, 0, 20)	$(130 \times 20) + (125 \times 20) + 15,000 = 20,100$

$$\Gamma_1(\Omega_{1(1)}, \Omega_{2(1)}, \dots, \Omega_{9(1)}) = \sum_{l=1}^9 C_{f(1)}^U \Omega_{f(1)}$$

또한, 단계 1에서 주어진 $(R_{1(1)}, R_{2(1)}, \dots, R_{9(1)})$ 에 대하여 최소생산비용은 다음과 같이 계산된다 :

$$\min_{\substack{\Omega_{f(1)} = Q_{f(1)} - R_{f(1)} \text{ for all } f(1) \\ \text{Lots from } \Omega_{f(1)} \text{ should satisfy} \\ \text{the lot size constraints.}}} \Psi_1(R_{1(1)}, R_{2(1)}, \dots, R_{9(1)}) = \Gamma_1(\Omega_{1(1)}, \Omega_{2(1)}, \dots, \Omega_{9(1)})$$

<Table 3>은 단계 2에서의 순환관계식 계산을 보여주고 있다. 단계 2에서 가질 수 있는 상태 $(R_{1(2)}, R_{2(2)}, R_{3(2)})$ 은 단계 1에서와 마찬가지로 투입 로트를 구성해야 할 수량 $(Q_{1(2)}, Q_{2(2)}, Q_{3(2)})$ 으로부터 구할 수 있다. 여기서 $Q_{f(2)} = \sum_{i^{(1)} \in \Delta_{f(2)}} R_{i^{(1)}}$ 이다. 또한, 앞서 언급한 대로 $(\Omega_{1(2)}, \Omega_{2(2)}, \Omega_{3(2)})$ 이 주어질 때 단계 2에서의 생산 비용은 다음과 같이 계산된다 :

$$\Gamma_2(\Omega_{1(2)}, \Omega_{2(2)}, \Omega_{3(2)}) = \sum_{l=1}^3 C_{f(2)}^U \Omega_{f(2)}$$

또한, 주어진 $(R_{1(2)}, R_{2(2)}, R_{3(2)})$ 에 대하여 단계 2까지 구성된 로트들에 대한 최소생산비용은 다음과 같이 계산된다 :

$$\min_{\substack{\Omega_{f(2)} = \sum_{i^{(1)} \in \Delta_{f(2)}} R_{i^{(1)}} - R_{f(2)} \text{ for all } f(2) \\ \text{Lots from } \Omega_{f(2)} \text{ should satisfy} \\ \text{the lot size constraints.}}} \Psi_2(R_{1(2)}, R_{2(2)}, R_{3(2)}) = \Gamma_2(\Omega_{1(2)}, \Omega_{2(2)}, \Omega_{3(2)}) + \Psi_1(R_{1(1)}, R_{2(1)}, \dots, R_{9(1)})$$

<Table 4> DP Calculations at Stage 2(Continued)

$(R_{1(2)}, R_{2(2)}, R_{3(2)})$	$\Psi_2(R_{1(2)}, R_{2(2)}, R_{3(2)})$
(4, 11, 2)	Should be determined using Equation (2)
(4, 11, 3)	"
...	...
(14, 2, 0)	"
...	...
(34, 11, 7)	"

<Table 4>는 단계 2에서 가능한 $(R_{1(2)}, R_{2(2)}, R_{3(2)})$ 에 대해서 $\Psi_2(R_{1(2)}, R_{2(2)}, R_{3(2)})$ 를 보여주고 있다. <Table 3>에서 볼 수 있듯이 여러 $(R_{1(1)}, R_{2(1)}, \dots, R_{9(1)})$ 로부터 동일한 $(R_{1(2)}, R_{2(2)}, R_{3(2)})$ 가 나타날 수 있으므로 그 중 가장 적은 비용을 발생시키는 것을 찾아야 한다. 예를 들어, 상태 $(R_{1(2)} = 14, R_{2(2)} = 2, R_{3(2)} = 1)$ 은 상태 $(R_{1(1)} = 3, R_{2(1)} = 13, R_{3(1)} = 18, R_{4(1)} = 5, R_{5(1)} = 2, R_{6(1)} = 0, R_{7(1)} = 6, R_{8(1)} = 15, R_{9(1)} = 0)$ 과 상태 $(R_{1(1)} = 3, R_{2(1)} = 13, R_{3(1)} = 18, R_{4(1)} = 5, R_{5(1)} = 2, R_{6(1)} = 0, R_{7(1)} = 6, R_{8(1)} = 15, R_{9(1)} = 1)$ 로부터 발생할 수 있다.

<Table 5>는 단계 3에서의 순환관계식 계산을 보여주고 있다. 앞서 언급한 대로 $\Omega_{1(3)}$ 이 주어질 때 단계 3에서의 생산 비용은 다음과 같이 계산된다 :

$$\Gamma_3(\Omega_{1(3)}) = C_{1(3)}^U \Omega_{1(3)}$$

또한, 주어진 $R_{1(3)}$ 에 대하여 단계 3까지 구성된 로트들에 대한 최소생산비용은 다음과 같이 계산된다 :

<Table 5> DP Calculations at Stage 3

$(R_1^{(2)}, R_2^{(2)}, R_3^{(2)})$	$R_1^{(3)}$	$\Omega_1^{(3)}$	$\Gamma_3(\Omega_1^{(3)}) + \Psi_2(R_1^{(2)}, R_2^{(2)}, R_3^{(2)})$	$\Psi_3(R_1^{(3)})$
(4, 11, 2)	0	17	$(4+11+2) \times 140 + \Psi_2(4, 11, 2)$	Should be determined using Equation (3)
(14, 2, 0)	0	16	$(14+2+0) \times 140 + \Psi_2(14, 2, 0)$	
...	
(34, 11, 7)	0	52	$(34+11+7) \times 140 + \Psi_2(34, 11, 7)$	

$$\Psi_3(R_1^{(3)}) = \Psi_3(0) = \min_{\substack{\Omega_1^{(3)} = \sum_{i \in \Delta_1} R_i^{(2)} \text{ Lots from} \\ \Omega_1^{(3)} \text{ should satisfy} \\ \text{the lot size constraints.}}} \Gamma_3(\Omega_1^{(3)}) + \Psi_2(R_1^{(2)}, R_2^{(2)}, R_3^{(2)})$$

<Table 5>에서 볼 수 있듯이 단계 3에서는 오직 하나의 상태만이 존재한다. 즉, $R_1^{(3)} = 0$. 이는 마지막 단계인 단계 3까지는 투입해야 할 모든 웨이퍼들이 로트로 구성되어야 한다는 것을 의미한다. 이상의 과정을 거쳐 구해진 $\Psi_3(R_1^{(3)}) = \Psi_3(0)$ 가 최소의 생산 비용이 된다. 최소값을 만들어내는 $\Gamma_3(\Omega_1^{(3)}) + \Psi_2(R_1^{(2)}, R_2^{(2)}, R_3^{(2)})$ 로부터 $(R_1^{(2)}, R_2^{(2)}, R_3^{(2)})$ 를 찾고, 다시 $(R_1^{(2)}, R_2^{(2)}, R_3^{(2)})$ 에 대한 최소값을 만들어내는 $\Gamma_2(\Omega_1^{(2)}, \Omega_2^{(2)}, \Omega_3^{(2)}) + \Psi_1(R_1^{(1)}, R_2^{(1)}, \dots, R_9^{(1)})$ 로부터 $(R_1^{(1)}, R_2^{(1)}, \dots, R_9^{(1)})$ 를 찾으면, 최적의 로트 구성을 결정할 수 있게 된다.

5. 결론 및 추후 연구

본 연구에서는 반도체 웨이퍼 랩에서의 투입 로트 구성 문제를 다루었다. 다양한 제품 유형들로 만들어지는 웨이퍼들을 동일한 로트로 구성하면 이질성 수준에 따라 생산 비용이 달라질 수 있다. 로트 사이즈 제약으로 인해 동질성을 가지도록 모든 로트들을 구성하는 것이 불가능할 수 있기 때문에 로트 사이즈 제약을 충족하면서도 총 생산비용이 최소가 되도록 이질성을 고려한 로트 구성 해법을 다차원 동적계획법으로 개발하였다. 또한, 개발한 해법의 적용 방법을 예제를 통하여 설명하였다. 예제에서 볼 수 있듯이 개발된 동적계획법은 초기 단계에 매우 많은 상태들이 존재할 수 있다. 이 상태들의 수는 단계가 증가함에 따라 크게 감소하기는 하지만, 초기 단계에서의 상태의 수를 줄일 수 있다면 해법의 실행 시간도 크게 줄일 수 있으리라 판단한다. 본 연구에서 제시한 투입 로트 구성 문제의 최적 알고리즘을 실제 랩에 적용하기 위해서는 일간/근무교대간 투입 최대 제한량과 동일한 로트 묶음을 수 있는 제품 유형 제한 등과 같은 현실적 고려 사항들을 반영한 시뮬레이션 테스트와 제시한 알고리즘의 수행시간에 대한 평가가 필요하다. 특히, 앞서 언급한 바와 같이 동적계획 알고리즘의 수행시간을 줄이기 위한 각 단계의 발생 가능 상태들의 수를 줄일 수 있는 방안을 찾

는 것이 매우 중요하다. 이때, CONWIP/UNIFORM 등과 같은 기존의 반도체 랩 투입 정책 연구들의 결과와 로트-오더 페깅 연구 결과들도 함께 활용한다면 실용적인 반도체 랩 투입 방안을 제시해 볼 수 있으리라 기대한다.

Acknowledgement

이 논문은 2014년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2014 R1A1A2058416).

References

- [1] Bang, J.-Y., An, K.-Y., Kim, Y.-D., and Lim, S.-K., A due-date based algorithm for lot-order assignment in a semiconductor wafer fabrication facility, *IEEE Transactions on Semiconductor Manufacturing*, 2008, Vol. 21, No. 2, pp. 209-216.
- [2] Bang, J.-Y., Kim, Y.-D., and Choi, S.-W., Multiproduct Lot Merging-Splitting Algorithms for Semiconductor Wafer Fabrication, *IEEE Transactions on Semiconductor Manufacturing*, 2012, Vol. 25, No. 2, pp. 200-210.
- [3] Boushell, T., Fowler, J., Keha, A., Knutson, K., and Montgomery, D., Evaluation of heuristics for a class-constrained lot-to-order matching problem in semiconductor manufacturing, *International Journal of Production Research*, 2008, Vol. 46, No. 12, pp. 3143-3166.
- [4] Carlyle, M., Knutson, K., and Fowler, J., Bin covering algorithms in the second stage of the lot to order matching problem, *Journal of the Operational Research Society*, 2001, Vol. 52, No. 11, pp. 1232-1243.
- [5] Cha, M.-S. and Jang, J.-S., Effective Operation of SPC System in Semiconductor Manufacturing, *Journal of the Korean Institute of Plant Engineering*, 2009, Vol. 14, No. 4, pp. 95-103.
- [6] Fowler, J., Knutson, K., and Carlyle, M., Comparison and evaluation of lot-to-order matching policies for a semiconductor assembly and test facility, *International Journal of Production Research*, 2000, Vol. 38, No. 8,

- pp. 1841-1853.
- [7] Glassey, C.R. and Resende, M.G.C., A scheduling rule for job release in semiconductor fabrication, *Operations Research Letters*, 1988, Vol. 7, pp. 213-217.
- [8] Kim, J.-G. and Lim, S.-K., Order-lot pegging for minimizing total tardiness in semiconductor wafer fabrication process, *Journal of the Operational Research Society*, 2012, Vol. 63, No. 9, pp. 1258-1270.
- [9] Kim, J.-G., Lim, S.-K., and Bang, J.-Y., Lot-Order Assignment Applying Priority Rules for the Single-Machine Total Tardiness Scheduling with Nonnegative Time-Dependent Processing Times, *Mathematical Problems in Engineering*, 2015, Vol. 2015, Article ID 434653, p. 11.
- [10] Knutson, K., Kempf, K., Fowler, J., and Carlyle, M., Lot-to-order matching for a semiconductor assembly and test facility, *IIE Transactions*, 1999, Vol. 31, No. 11, pp. 1103-1111.
- [11] Lim, S.-K., Kim, J.-G., and Kim, H.-J., Simultaneous order-lot pegging and wafer release planning for semiconductor wafer fabrication facilities, *International Journal of Production Research*, 2014, Vol. 52, No. 12, pp. 3710-3724.
- [12] Lim, S.-K., Kim, J.-G., and Shin, Y., Optimal Three-Level Presort Loading of Commercial Bulk Mail in the Postal Service Industry, *Journal of the Operational Research Society*, 2015, Vol. 66, No. 6, pp. 1007-1022.
- [13] Ng, T.S., Sun, Y., and Fowler, J., Semiconductor lot allocation using robust optimization, *European Journal of Operational Research*, 2010, Vol. 205, No. 3, pp. 557-570.
- [14] Seo, J.-C. and Bang, J.-Y., On-time Production and Delivery Improvements through the Demand-Lot Pegging Framework for a Semiconductor Business, *Journal of the Society of Korea Industrial and Systems Engineering*, 2014, Vol. 37, No. 4, pp. 126-133.
- [15] Spearman, M.L., Woodruff, D.L., and Hopp, W.J., CONWIP : A pull alternative to kanban, *International Journal of Production Research*, 1990, Vol. 28, pp. 879-894.
- [16] Wein, L.M., Scheduling semiconductor wafer fabrication, *IEEE Transactions on Semiconductor Manufacturing*, 1998, Vol. 1, pp. 115-130.
- [17] Wu, T.W., Modular demand and supply pegging mechanism for semiconductor foundry, in *Proceedings of the IEEE International Symposium on Semiconductor Manufacturing San Jose, Calif, USA.*, 2003, pp. 325-328.

ORCID

- June-Young Bang | <http://orcid.org/0000-0003-4999-9161>
- Seung-Kil Lim | <http://orcid.org/0000-0002-9819-9139>
- Jae-Gon Kim | <http://orcid.org/0000-0002-4821-4441>