

평균납기 지연 최소화를 위한 배치생산공정의 실시간 로딩전략

구평희[†]

부경대학교 시스템경영공학과

A Real-Time Loading Strategy of Batch Processing Machines for Average Tardiness Minimization

Pyung-Hoi Koo

Department of Systems Management and Engineering, Pukyong National University

This paper provides a real-time loading strategy for batch processing machines in which a number of jobs are simultaneously processed as a batch. The batch processing machines can be seen in both manufacturing industries (e.g., semiconductor, automobile and metal working) and service industries (transportation vehicles, mail shipment and theme park). This paper focuses on batch processing machines in semiconductor manufacturing. We present a look-ahead loading strategy for tardiness minimization where future arrivals and due dates are taken into consideration. Simulation tests are performed on the presented strategy and some existing loading heuristics under various production settings with different traffic intensities and forecasting errors. Experimental results show that our strategy provides the performance of good quality.

Keywords: Batch Processing Machines, Semiconductor Manufacturing, Real-Time Loading, Dispatching, Batching

1. 서론

반도체 제조공정은 웨이퍼(wafer) 위에 여러 회로층을 입히는 웨이퍼팩(wafer fab) 공정과 회로가 형성된 웨이퍼를 절단하여 IC 칩으로 형상화하는 조립/검사 공정 단계로 구성된다. 이중 웨이퍼팩 공정은 반도체 생산을 위한 핵심 공정으로서 다양한 장비들로 연결된 많은 세부공정으로 이루어져 있다. 웨이퍼팩은 reentrant 흐름, 즉 제품이 시스템 내에서 일련의 공정들을 여러 번 반복적으로 거치는 특징이 있다. <Figure 1>은 반도체 웨이퍼팩 공정을 제품의 흐름 관점에서 간단하게 표현한 것이다. 반도체 공정의 복잡도 증가와 웨이퍼 대구경화로 인해 새로운 웨이퍼팩의 구축을 위한 투자비는 지속적으로 증가하여 300mm 웨이퍼를 생산하는 웨이퍼팩 공장을 신설하는 경우 약 40억달러의 자본 투자가 필요하다(Gupta *et al.*, 2006). 이처럼 막대한 자본이 필요한 반도체 생산시스템의 고효율화와 고생

산성을 달성하기 위해서는 적절한 반도체 생산운영전략을 적용하는 것이 중요하다.

반도체 생산공정은 생산형태에 따라 개별생산공정(DPM : discrete processing machine)과 배치생산공정(BPM : batch processing machine)으로 분류될 수 있다. 개별생산공정은 웨이퍼를 하나씩 날개로 생산하는 공정을 말하고 배치생산공정은 <Figure 2>와 같이 여러 개의 제품을 묶어 배치(batch)로 구성하고, 구성된 배치를 동시에 처리하는 공정을 말한다. 웨이퍼팩에는 확산(diffusion), 습식각(wet etching), 산화막형성(oxidation) 공정 등이 이러한 배치생산공정에 속하며 전체 공정 중 30% 이상을 차지한다. 배치공정은 반도체산업뿐만 아니라 열처리공정, 건조공정, 오븐공정 등을 포함한 타 제조산업과, 셔틀서비스, 놀이동산 기구, 물품 운송 등 서비스 산업에서도 찾아볼 수 있다. 일반적으로 배치생산공정은 생산능력에 제한이 있어 한 번에 수용할 수 있는 최대수용능력이 존재한다. 일단

논문은 2012년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2012R1A1A4A01014897).

[†] 연락저자 : 구평희 교수, 608-737 부산광역시 남구 용소로 45 부경대학교 시스템경영공학과, Tel : 051-629-6485, Fax : 051-629-6478,

E-mail : phkoo@pknu.ac.kr

2013년 7월 26일 접수; 2013년 10월 12일 수정본 접수; 2013년 11월 15일 게재 확정.

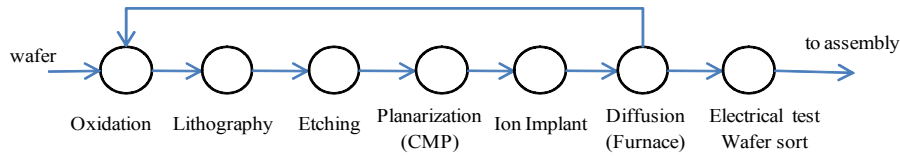


Figure 1. Production flow in semiconductor wafer fab

작업이 시작되면 중단 없이 작업이 진행되며 작업 중간에는 작업물이 새롭게 추가되거나 공정에서 빠져나올 수 없고 모든 작업물이 동시에 작업을 마치게 된다.

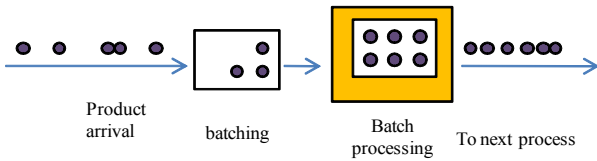


Figure 2. Schematic representation of a batch processing machine

본 논문은 웨이퍼팩에서의 BPM을 대상으로 실시간 로딩 (loading) 전략을 다룬다. BPM이 작업을 완료한 후에 다음 작업을 시작할 준비가 되었을 때 버퍼에 대기 중인 제품이 있는 경우에, 현재의 제품으로 배치를 구성하여 작업을 시작할 것인가 아니면 미래에 도착하는 제품을 기다렸다가 배치를 크게 하여 미래시점에서 작업을 시작할 것인가 하는 로딩 의사결정이 필요하다. 미래에 도착하는 제품을 기다렸다가 이를 포함하여 로딩하는 것은 현재 대기 중인 제품에 대해서는 대기시간이 증가하고 납기준수에 악영향을 줄 수 있다는 단점이 있는 반면에, 미래에 도착하는 제품에 대해서는 도착하면 바로 생산이 시작되므로 대기시간이 감소되고 납기준수율이 높아질 수 있다는 장점이 있다. 본 논문에서는 이처럼, 어떤 제품으로 배치를 구성해서 언제 배치기계에서 생산을 시작할 것인가 하는 실시간 로딩 문제를 다룬다. 실시간 로딩 의사결정 문제는 배치기계가 작업을 완료한 시점 또는 기계가 유향상태에서 새로운 제품도 도착하는 시점 두 가지 경우에 발생한다.

본 논문의 구성은 제 2장에서 배치공정 로딩전략에 대한 기존의 연구를 정리하고 제시될 새로운 로딩전략의 배경에 대해 설명한다. 제 3장에서는 미래정보를 이용한 배치공정의 실시간 로딩전략을 기술하고, 제 4장에서는 도출된 로딩전략의 성능을 검증하기 위해 수행한 시뮬레이션 실험 결과를 분석한다. 마지막으로 제 5장에서는 연구 결과와 앞으로의 추가 연구 방향에 대해 기술한다.

2. 배치생산공정의 실시간 로딩전략 연구

배치생산공정의 실시간 로딩은 의사결정시 사용하는 정보의 내용에 따라 threshold 전략과 look-ahead 전략으로 분류할 수 있다(Koo and Moon, 2013). Threshold 전략은 BPM에 대기하고

있는 제품의 수가 일정한 수준(threshold) 이상이 되면 해당제품을 배치로 구성하여 작업을 시작하고, 그렇지 않으면 추가로 제품이 도착 할 때까지 기다리는 전략이다. 반면에, look-ahead 전략은 현재 대기중인 제품과 더불어 가까운 미래의 시스템 상태와 도착하는 제품을 고려하여 로딩에 대한 의사결정을 하는 전략이다. 초기의 배치공정 로딩전략에 대한 연구는 주로 배치공정의 로딩문제가 bulk 서비스를 다루는 대기(queueing) 시스템과 유사하므로 대기이론을 기반으로 한 최소배치크기 (MBS : minimum batch size)를 결정하는 threshold 전략을 다루었다(Neuts, 1967; Deb Serfozo, 1973; Neuts and Nadarajan, 1982). Avramidis *et al.*(1998)은 제품의 도착이 포아송 분포를 따르고 생산시간이 IID(independent and identically distributed)인 환경에서 리드타임을 최소로 하는 MBS 크기를 찾는 알고리즘을 소개하였다. 실험을 통하여 MBS 크기는 가동률(utilization)이 높은 경우보다 낮은 경우에 더 민감하게 수행도(performance)에 영향을 준다는 사실을 확인하였고, 특히 가동률이 낮은 경우는 MBS 크기를 낮게 하는 것이 유리하다고 주장하였다. Akcali *et al.*(2000)은 배치기계에서의 실시간 의사결정을 다음에 생산할 제품을 결정하는 디스패칭(dispatching)과 배치크기를 얼마나 할 것인가를 정하는 로딩으로 분류하고, 시뮬레이션 실험을 통하여 이들의 영향을 분석하였다. 실험결과 디스패칭보다 로딩에 대한 결정이 리드타임 및 납기와 연관된 생산성지표에 더 큰 영향을 미친다고 주장하였다. Leachman *et al.*(2002)은 국내 S전자의 웨이퍼팩에 실제로 적용한 사례를 소개하면서, 리드타임 감축을 위한 생산로딩전략에서 배치공정에서의 로딩 전략으로 MBS를 기반으로 한 알고리즘을 소개하였다. 운영되고 있는 배치공정은 비병목공정이므로 세부적인 스케줄에 의해서 운영되는 것이 아니라, 시스템의 상태를 보아가며 실시간으로 로컬정보만 가지고 의사결정을 하고 있다고 기술하고 있다. Fowler *et al.*(2002)은 웨이퍼팩에서 배치공정에 의한 시스템 변동성 증대의 영향을 강조하면서, 제품의 도착과 배치처리시간이 확률적으로 변하는 경우 대기이론을 기반으로 하여 대기시간과 리드타임을 계산하는 모형을 소개하고 이를 기반으로 유전자알고리즘을 활용하여 리드타임을 최소로 하는 배치크기 결정 방안을 제시하였다. Phojanamongkolkij *et al.*(2002)은 다품목, 다기계 상황에서 대기모형을 이용하여 MBS를 결정하는 방안을 제안하였고, 여러 다른 BPM운영 방안과 비교 분석하는 실험을 실제 반도체 공정 데이터를 이용하여 분석하였다.

반도체 웨이퍼팩은 보통 MES(manufacturing execution system)

Table 1. Literature on look-ahead loading strategy

Literature	Product ¹	Station ²	Criterion ³	Extra notes ⁴
Glassey and Weng(1991)	s	s	FT	DBH
Flower <i>et al.</i> (1992)	s, m	s	FT	NACH
Weng and Leachman(1993)	s, m	s	FT	MCR
Robinson <i>et al.</i> (1995)	m	s	FT	RHCR, β - δ serial line
Chun and Hong	s	s	FT	MMCR, setup cost
Van der Zee <i>et al.</i> (1997)	m	m	FT	DJAH
Van der Zee <i>et al.</i> (2001)	m	m	FT	DSH, non-identical machine
Murray <i>et al.</i> (2008)	m	m	FT	TLNA, sequence dependent
Kim <i>et al.</i> (2001)	m	m	TD	MMBS, MDBH, PUCH
Kim <i>et al.</i> (2010)	m	m	TD	PRA, PRALC
Gupta and Sivakumar(2006)	s	s	TD	LAB
Sha <i>et al.</i> (2007)	m/s	s	TD	LBCR
Cerekci and Banerjee(2010)	m	s	TD	NACH-T, NARCH-T
Park and Banerjee(2011)	m	s	TD, ER, FT	MBS-DBU, NACH-DBU

^{1,2} s : single product type(machine); m : multiple product types(machines).

³ FT : flow time; TD : tardiness; ER : earliness.

⁴ β : batch machine, δ : discrete machine.

등과 같은 정보시스템이 구축되어 있어 현장의 상태정보를 실시간으로 얻을 수 있다. 최근의 배치공정 로딩과 관련된 연구는 대부분 실시간 상태정보를 기반으로 하는 미래예측(look-ahead) 기능을 활용하고 있다. <Table 1>은 기 수행된 look-ahead 기반의 로딩전략을 각 전략이 대상으로 하는 시스템의 특성과 목적을 포함하여 정리한 것이다.

Glassey and Weng(1991)은 단일제품, 단일배치기계에서 미래 도착제품 정보를 고려하여 대기시간을 최소로 하는 DBH(dynamic batching heuristic)를 제시하고, 시뮬레이션 실험을 통하여 여러 생산환경에서 DBH의 수행도를 분석하였다. DBH는 현 시점에 로딩하는 경우와 미래에 도착하는 제품을 기다렸다가 배치를 구성하여 로딩하는 경우의 총 대기시간을 각각 계산하고, 총 대기시간이 최소가 되는 시점에 로딩하는 전략이다. 실험을 통하여 가동률이 중간(약 30~70%) 정도인 경우에 제시된 look-ahead 방식이 일반적인 threshold 전략보다 수행도가 좋게 나왔지만, 가동률이 아주 낮거나 높으면 대기시간에 있어 차이가 미미하다고 주장하였다. Fowler *et al.*(1992)은 DBH에서 절감된 대기시간에 대해 바로 다음에 도착하는 제품의 영향력이 그 이후에 도착하는 제품들을 모두 합한 영향력 만큼 된다는 실험결과를 토대로, 다음에 도착하는 한 개의 제품만을 고려하여 현재 배치공정을 시작할 것인지 다음 도착시점으로 의사결정시점을 미룰 것인지 결정하는 NACH(next arrival control heuristic)를 제시하였다. 다음 도착시점에 로딩해야 하는 것으로 결정되면 다음 도착 시점에서 다시 의사결정 과정을 새롭게 진행하는 rolling horizon 방식을 적용하고 있다. NACH는 DBH보다 적은 정보를 필요로 하고 미래정보의 오류에도 강건하다는 장점이 있다. 후에 Weng and Leachman(1993)

은 DBH의 planning horizon을 변경한 수정된 휴리스틱인 MCR(minimum cost rate)을 제시하였다. MCR에서는 총 대기시간의 절감대신에 단위 시간당 대기시간을 기반으로 의사결정을 수행한다. Chun and Hong(1996)은 작업준비시간이 존재하는 경우 준비 비용을 고려한 MMCR(Modified MCR)을 제안하였다. Robinson *et al.*(1995)은 다중제품에 대해서 NACH와 MCR을 혼합한 RHCR(rolling horizon cost rate heuristic)을 제시하였다. RHCR에서는 전공정(upstream)과 후공정(downstream)의 정보를 모두 활용하는 디스패칭 방안을 제시하고, 중간 정도의 부하율인 시스템에서 추가적으로 생산성이 향상될 수 있다고 주장하였다, 특히 부하율이 높은 시스템에서는 개별생산공정의 스케줄링보다 배치생산공정의 스케줄링이 시스템 수행도에 더 큰 영향을 준다는 것을 실험을 통하여 확인하였다. Van der Zee *et al.*(1997)은 복수의 배치기계에서 대기 비용과 셋업 비용으로 이루어진 logistics 비용을 최소로 하는 실시간 스케줄링 방안인 DJAH(dynamic job assignment heuristic)을 제안하고, 후에 서로 상이한 복수배치기계에서의 문제로 연구를 확장 하였다(Van der Zee *et al.*, 2001). Murray *et al.*(2008)은 여러 대로 구성된 BPM에서 대기시간 및 기계의 운영시간과 연관된 비용의 합을 최소로 하는 TLNA(time-limit next arrival heuristic)를 제시하였다. TLNA에서는 제품의 종류가 변경될 때 셋업시간이 존재하는 경우를 다루고 있으며, 특히 제품의 생산순서에 따라 준비시간이 다른 경우를 고려하여 작업순서를 결정한다.

위에서 기술한 threshold를 기반으로 하는 로딩전략과 look-ahead 로딩전략에 관한 연구들은 대부분 리드타임과 대기시간의 최소화를 목적으로 하고 있다. 최근에는 납기를 고려하여 납기지연(tardiness)을 최소로 하는 배치공정 로딩전략에 대한

연구가 증가하고 있다. Kim *et al.*(2001)은 기존의 DBH 전략에 기반으로 납기를 추가로 고려한 MDBH(modified DBH)를 제시하였고 후에 이를 보완하여 look-ahead 기간을 확장시킨 PRALC(priority rule-based algorithm with look-ahead checks)를 소개하였다(Kim *et al.*, 2010). 여기서 로딩에 대한 의사결정은 납기의 긴급도(urgency)를 고려한 가중대기시간을 고려하여 이루어진다. Gupta and Sivakumar(2006)는 린(lean) 생산시스템 환경에서 적시공급의 중요성을 주장하고, 제품의 도착이 동적인 배치생산 공정에서 납기지연을 생산성지표로 하는 LAB(look-ahead batching)를 제시하였다. 우선 미래의 도착시점에서 배치를 구성하여 시나리오로 구성하고, 각 시나리오 별로 여유시간을 계산하여 총여유시간이 가장 적은 시나리오를 선택하여 해당 시점을 로딩시간으로 정하는 방법이다. 이 방법에서는 로딩시점 이후에 도착하는 제품의 긴급도는 고려하지 않는다. Sha *et al.*(2007)은 고객의 만족도와 제품의 적시공급이 중요해지는 환경에서 납기와 관련된 평가지표가 중요하다고 강조하고, BPM에서 납기기반의 평가지표를 최적화하는 디스패칭 방안인 LBCR(due-date oriented look-ahead batching rule)을 제시하였다. Cerekci and Banerjee(2010)는 복수의 배치기계에서 평균 납기지연시간을 최소화 하는 로딩전략인 NACH-T를 제시하였다. 로딩하는 시점을 결정하기 위하여 각 미래도착시점 별로 총 납기지연시간을 구하고 그 중 가장 적은 납기지연을 유발하는 시점을 로딩시점으로 선택한다. Park and Banerjee(2011)는 납기를 고려한 효용함수(utility function)를 제시하고, 이를 이용하여 MBS와 NACH를 수정한 MBS-SUE(MBS-Stochastic Evaluation Utility)와 NACH-SUE를 제시하였다. 여기서는 납기를 만족하는 제품에 대해서는 기존의 MBS와 NACH를 그대로 사용하고, 납기를 초과한 제품에 대해서는 납기지연을 고려하여 로딩여부를 결정하는 방안을 제시하였다.

앞에서 언급하였듯이 웨이퍼팹의 특징은 개별기계와 배치 공정 등 여러 종류의 장비를 거치면서 작업이 진행되고, 이러한 작업은 반복적으로 수행된다는 것이다. 일반적으로 웨이퍼팹의 병목공정은 노광(lithography)장비이다. Reentrant 흐름을 특징으로 하는 웨이퍼팹에서 배치공정에서 처리되어 흐르는 제품은 다시 노광공정을 거쳐야 하므로 비병목공정인 배치 공정에서는 병목공정인 노광장비가 중단되지 않도록 필요한 시점에 맞추어 노광공정에 제품을 공급해 주는 것이 무엇보다도 중요하다. 병목공정은 시스템의 능력을 결정하는 중요한 공정이기 때문에 보통 세부적인 일정계획이 수립되어 작업이 진행된다. 따라서 노광공정에서는 각 제품의 생산시점이 알려져 있으므로 이 생산시점이 배치공정에서는 납기의 역할을 한다. 따라서 전체 시스템적인 측면에서 보자면 배치공정 로딩전략은 기존에 널리 사용되고 있는 리드타임 및 대기시간의 최소화 보다는 납기를 기반으로 하는 납기지연의 최소화가 중요하다. 또한, 기존의 연구에서는 배치공정이 유희상태로 되었을 때 대기하고 있는 제품의 수가 최대수용능력 이상이면 생산지연을 최소화 하고 기계의 가동률을 최대화 하기 위하여 바로

로딩하여 생산을 시작한다. 그러나 각 제품별로 납기가 존재하는 경우에는 급하게 생산해야 하는 제품이 시간적으로 잠시 후에 도착하고 현재 버퍼에 있는 제품 중 일부는 납기에 여유가 있다면 잠시 기다렸다가 급한 제품을 배치로 묶어 생산을 시작하는 것이 전체 수행도에 도움을 줄 수 있다. 이때 여러 종류의 제품이 기다리는 경우는 어떤 종류의 제품을 생산할 것인가를 결정해야 한다. 기존연구 중 Gupta and Sivakumar(2006)의 LAB 만이 이러한 문제를 다루고 있다. 하지만 LAB에서는 로딩에 대한 의사 결정 시 로딩시점 이후에 도착하는 제품에 대한 영향을 고려하지 않으므로 생산성에서 손실을 볼 수 있다. 이러한 문제를 해결하기 위하여 본 논문에서는 아래와 같이 배치생산공정에서 납기지연을 최소로 하는 새로운 로딩전략을 제시한다.

3. 새로운 배치공정 로딩 전략 : LBT(Look-Ahead Batching for Average Tardiness Minimization)

본 논문에서 제시하는 배치공정의 실시간 로딩전략은 평균 납기지연의 최소화를 목적으로 배치를 구성할 제품을 선정하고 생산시간을 결정하는 전략으로 LBT(look-ahead batching for average tardiness minimization)라고 칭한다. 배치공정은 최대수용능력이 정해져 있고 일정한 처리시간을 가지는 단일기계를 대상으로 한다. Look-ahead 기간 내에 도착하는 제품의 도착시간 및 납기정보는 알려져 있고, 배치는 제품과 무관하게 구성할 수 있으며, 배치를 구성하여 제품을 생산하기 시작하면 중간에 제품을 추가할 수 없는 상황을 대상으로 한다. LBT에서 사용하는 기호는 아래와 같다.

- t_0 : 의사결정 시점(기계에서 생산 완료시점 또는 새로운 제품 도착 시점)
- t_i : 시점 t_0 이후 i 번째로 도착하는 제품의 도착시점
- q : 시점 t_0 에서 대기 중인 제품의 수
- T : 배치공정의 생산시간
- la : look-ahead 제품의 수
- n^* : (t_0+T) 이내에 도착하는 제품의 수
- n : 고려대상인 미래 도착제품의 수($n = \min(la, n^*)$)
- d_j : 제품 j 의 납기
- td_{ji} : 시점 t_i 에서 제품 j 의 납기지연시간
- sl_{ji} : 시점 t_i 에서 제품 j 의 생산여유시간
- B_i^- : 시점 t_i 에 이미 도착하여 대기하고 있는 제품의 집합
- B_i^0 : 시점 t_i 에서 배치로 구성되는 제품의 집합
- B_i^+ : 시점 t_i 와 t_n 사이에 도착하는 제품의 집합
- TT_i : 시점 t_i 에서의 총 납기지연시간

로딩에 대한 의사결정은 시점 t_0 에서 발생하고 이 시점은 대기 중인 제품이 존재하는 상태에서 배치공정이 작업을 끝내는

시간 또는 배치공정이 유희중인 상태에서 새로운 제품이 도착하는 시간 두 경우 중 하나이다. <Figure 3>은 LBT 로딩전략의 전체적인 의사결정 절차를 보여주고 있다. LBT 로딩전략은 로딩 가능 시점별로 배치를 구성하고 이를 바탕으로 총 납기지연시간을 구한 후, 총 납기지연시간이 최소인 시점을 선택하여 로딩하는 것이다. 세부적인 절차는 다음과 같다.

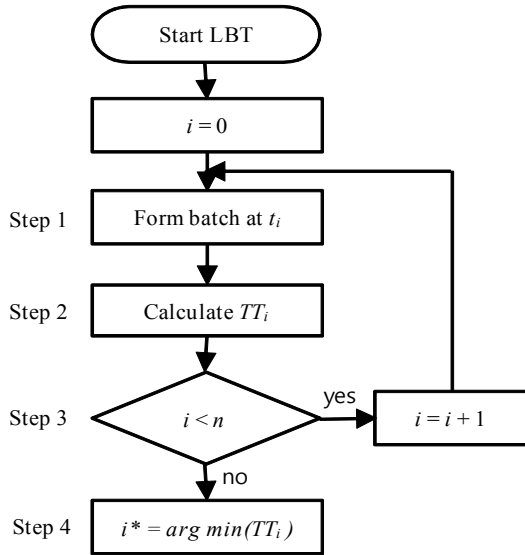


Figure 3. Flowchart of the LBT loading strategy

Step 1 : 로딩 가능 시점별 배치(B_i^0) 구성 : 시점 t_i 에서의 배치, B_i^0 의 구성은 t_i 시점에 버퍼에 대기 중인 제품이 배치공정의 최대수용능력보다 적은 경우는 모든 제품으로 배치를 구성하고, 최대수용능력보다 많은 경우는 각 제품에 대해 여유시간을 계산하고 여유시간이 적은 순서대로 최대수용능력 만큼을 배치로 구성한다. 제품 j 에 대한 t_i 시점에서의 여유시간은 아래와 같이 계산된다.

$$sl_{ji} = d_j - t_i - T, j \in B_i^-$$

Step 2 : 로딩 가능 시점 별 총 납기지연시간(TT_i) 계산 : 시점 t_i 에서 우선 각 제품별로 납기지연을 계산하고 이를 이용하여 총 납기지연시간, TT_i 를 구한다. 납기 지연 시간은 현재 대기하는 제품과 고려대상인 미래도착 제품 모두에 대해 계산한다. 각 제품은 t_i 시점에 로딩되는 배치에 포함되는 제품, B_i^0 과 배치에 포함되지 않은 제품, $(B_i^- - B_i^0) \cup B_i^+$ 으로 나누어진다. 시점 t_i 에서 제품별 납기지연시간(td_{ji})과 총 납기지연시간(TT_i)은 아래와 같이 구한다.

$$td_{ji} = \max(0, t_i + T - d_j), j \in B_i^0$$

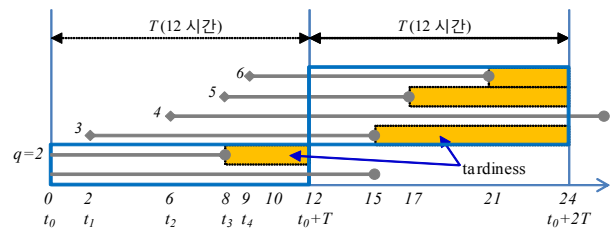
$$td_{ji} = \max(0, t_i + 2T - d_j), j \in (B_i^- - B_i^0) \cup B_i^+$$

$$TT_i = \sum_{j=1}^{q+n} td_{ji}$$

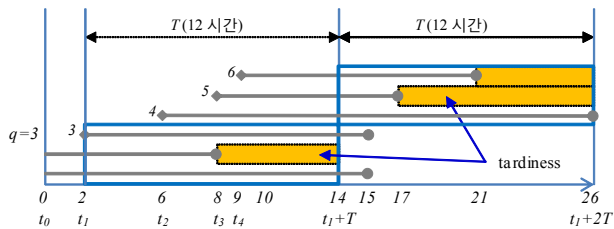
Step 3 : 위의 Step 1과 Step 2를 t_0 에서부터 t_n 까지 $n+1$ 회 반복하여 계산한다.

Step 4 : 각 미래도착시점에 대한 총납기지연시간을 구한 후 모든 경우 중에서 총 납기지연시간이 가장 적은 도착시점을 선택하여 이를 다음 로딩시간(i^*)으로 정한다 : $i^* = \arg_i \min(TT_i)$.

<Figure 4>는 위 알고리즘이 어떻게 적용되는지 설명하기 위한 예를 보여주고 있다. 여기서 x 축은 시간을 나타내고 y 축은 로딩해야 할 제품을 나타낸다. 현 시점 t_0 (시간 0)에서 로딩 여부에 대한 의사결정을 해야 하는 상황이다. 배치공정의 표준생산시간은 12시간 소요되고 이 기간 동안에 도착할 것으로 예상되는 제품은 4개가 있다고 가정한다. 그림에서 각 제품의 도착시간과 납기는 선의 시작과 끝으로 표현하였고, 선의 시작지점에 제품번호가 나타나 있다. <Figure 4>(a)는 시점 t_0 에 로딩하는 경우로서, 현재 대기 중인 q 개(이 예에서는 2개)의 제품으로 배치를 구성해서 생산을 시작하여 T 시간 즉 12시간 뒤에 생산이 완료된다. 이때 2번 제품은 4시간의 납기지연이 발생한다. 시간간격 $[t_0, t_0 + T]$ 동안에 도착하는 제품들은 (이 예에서는 4개의 제품) 현 작업이 끝난 후에야 로딩이 가능하므로 $(t_0 + 2T)$ 즉 24시간 뒤에 생산이 완료될 수 있고 이에 따라 제품별로 납기지연이 발생할 수 있다. 여기서는 3, 5, 6번 제품에 대해 각각 9시간, 7시간, 3시간의 납기지연이 발생한다. 따라서 t_0 에 로딩작업을 하는 경우의 총 납기지연시간은 23시간이 된다. <Figure 4>(b)의 경우는 시점 t_1 (2시간 후)에 로딩하는 경우의 생산 상황이다. 여기서는, 현재 대기 중인 제품 2개와 미래에



(a) Loading at time t_0 (the number of parts loaded : 2)



(b) Loading at time t_1 (the number of parts loaded : 3)

Figure 4. Tardiness when the batch is loaded at time t_0 (a) and t_1 (b)

도착하는 1개의 제품을 합하여 3개로 배치를 구성하여 시점 t_1 에 생산을 시작한다. 이 경우 제품 2의 납기지연시간은 6분으로 증가하지만 제품 3에 대해서는 납기지연이 발생하지 않는다. 이후에 도착하는 4, 5, 6번 제품은 14시간 지나서 로딩될 수 있고 이때 제품 5와 6은 각각 9시간과 5시간의 납기지연이 발생한다. 따라서 t_1 에 로딩작업을 하는 경우의 총 납기지연시간은 20시간이 된다. 이와 같은 과정을 반복적으로 수행하여 각 도착시점별로 총 납기지연시간을 구하고 이중 가장 적은 총납기지연시간을 가져다주는 로딩시간을 선택하여 로딩작업을 수행한다. 이 예에서는 2시간을 기다린 후 시점 t_1 에 로딩하는 것이 납기지연시간을 최소로 하며 따라서 LBT 알고리즘은 이 시점을 로딩시간으로 결정한다.

4. 실험 분석

제시된 LBT 알고리즘과 기존의 방법들을 비교하기 위하여 시뮬레이션 모형을 구성하고 여러 생산환경 하에서 시뮬레이션 실험을 수행하였다. 시뮬레이션은 ARENA 패키지를 이용하였고 의사결정 로직을 모델링 하기 위해서는 VBA(Visual Basic Application)를 추가로 활용하였다. 실험결과와 신뢰성과 안정된 통계치 확보를 위하여 시뮬레이션 초기 2,000단위시간은 예비실행(Warm-Up) 기간으로 하였고 그 이후 10,000시간단위의 실험결과를 통계치로 수집하였다. 통계적인 오류를 최소화하기 위하여 100회 반복 실험하였다. 비교를 위한 수행도 지표는 평균 납기지연시간으로 하였다. 납기와 관련된 수행도 지표는 납기의 긴급도 및 배치공정 부하율(TI : traffic intensity)에 영향을 받는다. 배치공정의 TI 는 아래와 같이 정의된다.

$$TI = \frac{T}{\text{평균도착시간} \times \text{배치공정 수용능력}}$$

시뮬레이션을 위한 기본적인 데이터는 Gupta and Sivakumar (2006)에서 사용한 생산환경을 대부분 참조하였다. 시뮬레이션 실험 모형은 앞서 보인 <Figure 2>와 같다. 제품은 배치공정에 동적으로 도착하고 임시대기장소인 버퍼(buffer)에서 대기한다. 도착하는 각 제품의 도착시간과 납기는 미리 알려져 있다. 배치공정에는 한대의 배치기계가 작업하고, 배치기계의 최대수용능력은 6이다. 도착하는 제품들은 어떠한 제품과도 배치구성이 가능하고, 배치기계에서의 생산시간, T 는 변동성 없이 항상 일정하게 60 단위시간이라고 가정한다. 배치공정의 TI 는 배치공정의 수행도에 영향을 주는 주요 요인이다. 본 논문에서는 0.4~0.8범위에서 배치공정의 TI 를 변경해 가며 실험을 수행하였다. 배치공정의 TI 는 도착시간 간격을 변경하여 조정하였다. 예를 들어 배치공정 $TI = 0.5$ 상황으로 실험하기 위해서 도착시간간격은 20시간단위 $\{60/(0.5 \times 6)\}$ 로 하였다. 각 제품의 납기(d_j)는 $d_j = \text{제품}j\text{의 도착시간} + \text{Uniform}[0, 4] \times T$ 로 설정

하였다. 제시한 전략의 수행도 평가를 위해서 기존에 납기지연 최소화를 목적으로 하는 LAB(Gupta and Sivakumar, 2006), NACH-T(Cerekci and Banerjee, 2010), PRALC(Kim *et al.*, 2010) 그리고 납기일을 고려하여 우선적으로 로딩하는 전략을 사용하도록 수정한 MBS 전략을 벤치마크로서 활용하였다. MBS에서는 threshold 값에 따라 다른 결과를 주는데 본 실험에서는 예비실험결과 TI 별로 납기지연을 적게 하는 MBS = 1($TI = 0.4$ 인 경우), MBS = 2($TI = 0.5 \sim 0.7$ 인 경우), MBS = 3($TI = 0.8$ 인 경우) 값을 택하였다.

<Figure 5>는 배치공정 TI 가 0.4~0.8로 변하는 경우 각 로딩 전략의 평균 납기지연시간을 보여주고 있다. 모든 TI 에서 MBS, LAB, PRAL, NACH-T, LBT의 순서로 납기지연시간이 적게 나온 것을 볼 수 있다. 기존의 방법 중에서는 NACH-T가 가장 좋은 결과를 보여주고 특히 TI 가 0.8인 경우는 LBT와 유사한 결과를 나타낸다. 본 논문에서 제시한 LBT는 전체적으로 모든 TI 에서 납기시간이 가장 적은 결과를 보였고, 특히 TI 가 중간(0.6~0.7) 정도일 때 상대적으로 더욱 좋은 결과를 보이고 있다. Gupta and Sivakumar(2006)가 기술하였듯이 반도체 웨이퍼 팸에서 배치공정은 TI 가 0.7이내로 제한하는 것이 일반적이므로 제시된 LBT는 반도체 공정에서 활용성이 높을 것으로 기대된다.

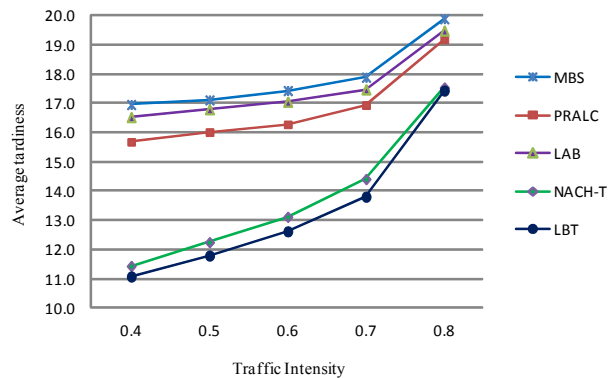


Figure 5. Comparison of loading strategies under different traffic intensities

Table 2. 95% confidence interval of tardiness for NACH-T and LBT

TI	NACH-T	LBT
0.4	(10.83, 11.31)	(11.18, 11.68)
0.5	(11.56, 12.02)	(12.01, 12.47)
0.6	(12.38, 12.84)	(12.87, 13.35)
0.7	(13.50, 14.10)	(14.12, 14.68)
0.8	(16.55, 18.31)	(16.75, 18.31)

<Table 2>은 NACH-T와 LBT를 좀더 세밀히 분석하기 위하여 95% 신뢰구간을 비교한 것이다. TI 가 0.5~0.7인 경우 LBT

가 NACH-T의 수행도보다 우수한 결과를 주는 것을 확인할 수 있다. 그러나 TI 가 0.8과 같이 상당히 높거나 0.4와 같이 낮은 경우는 NACH-T도 우수한 결과를 주고 있는 것을 확인할 수 있다.

본 논문에서 제시한 LBT 로딩전략은 제품의 look-ahead 수를 정하고 이 수만큼 미래에 도착하는 제품의 도착시간을 예측하고 로딩과 관련된 의사결정을 수행한다. <Figure 6>은 TI 가 0.7인 경우 look-ahead 수에 따른 LBT의 수행도의 변화를 보여주고 있다. 그림에서 look-ahead 수가 증가하면 납기지연시간이 줄어든다는 점과 look-ahead 수가 증가할수록 그 감소폭은 감소함을 알 수 있다. Look-ahead 수를 증가하면 더 많은 정보가 필요하고 의사결정이 복잡해 진다는 점에서 수행도를 참고하여 적절한 look-ahead 수를 정하는 것이 필요하다. 본 실험에서는 look-ahead 수가 5 이상이면 수행도의 향상이 미미하다는 판단 하에 look-ahead 수를 5로 정하였다.

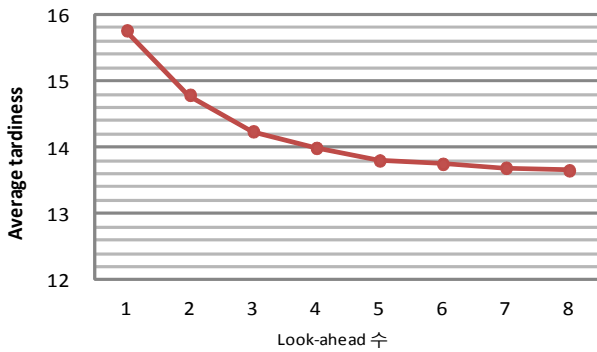


Figure 6. Performance of LBT over different look-ahead spans

Table 3. Tardiness of LBT and LBT-I over different traffic intensities

TI	LBT	LBT-I	차이 %
0.4	11.07	10.94	1.2%
0.5	11.79	11.62	1.4%
0.6	12.61	12.45	1.3%
0.7	13.80	13.71	0.7%
0.8	17.43	17.40	0.2%

제시한 LBT에서는 배치크기와 로딩시간이 정해지면 선택된 로딩시간까지 기다렸다가 작업을 시작한다. 본 논문에서는 이러한 LBT 전략에 시스템의 상황을 동적으로 대응하기 위해 미래 새로운 제품이 도착할 때마다 의사결정을 수행하는 개선된 절차인 LBT-I도 함께 소개한다. LBT-I는 LBT에서 결정된 로딩시점으로 시뮬레이션 시간을 증가시키는 대신에 바로 다음 작업이 도착하는 시점까지 시뮬레이션 시간을 증가시켜 그 시점에서 다시 LBT 절차를 시작한다. <Table 3>는 LBT와 LBT-I의 시뮬레이션 실험 결과를 비교하고 있다. 표에서 LBT-I가 LBT와 비교하여 모든 TI 구간에서 납기지연 시간을 줄인다는

것을 알 수 있고 특히 중간정도의 $TI(0.5 \sim 0.6)$ 에서 개선 폭이 크다는 것을 알 수 있다.

생산시스템에서는 여러 변동성이 존재하기 때문에 제품이 도착할 것으로 예측된 시간에 도착하지 못하는 경우가 자주 발생한다. 도착시간에 대한 예측 오류에 대해서 LBT와 LBT-I가 어떤 영향을 받는지 조사하기 위하여 예측오류 정도에 따라 시뮬레이션 실험을 수행하였다. 여기서 예측오류는 제품의 평균도착시간 간격에 비례한다고 가정하였다. 즉 시간 t 에 도착할 것으로 예상되는 제품의 실제 도착시간은 $[t - k \times \text{평균도착시간간격}, t + k \times \text{평균도착시간간격}]$ 을 범위로 하는 일양분포를 따른다고 가정하였다. <Figure 7>은 TI 가 0.7인 상황에서 $k = 0, 0.2, 0.4, 0.6, 0.8$ 인 경우의 평균 납기지연시간을 비교하고 있다. 그림에서 LBT-I와 LBT가 NACH-T와 비교해서 모든 예측 오류 상황에서 납기지연시간이 낮은 것을 볼 수 있다. 하지만 예측 오류가 아주 큰 경우($k = 0.8$ 인 경우) 비교한 세 전략의 수행도는 그 차이가 줄어들었다는 것을 알 수 있다. LBT와 LBT-I의 전략을 비교해 보면, LBT-I는 예측 오류가 없을 때보다 예측 오류가 발생하는 경우 LBT의 결과보다 상대적으로 더욱 좋은 결과를 보여주고 있다. 따라서 예측오류가 발생하는 상황에서는 로딩에 대한 의사결정을 제품이 도착할 때마다 수행하는 것이 바람직하다고 할 수 있다.

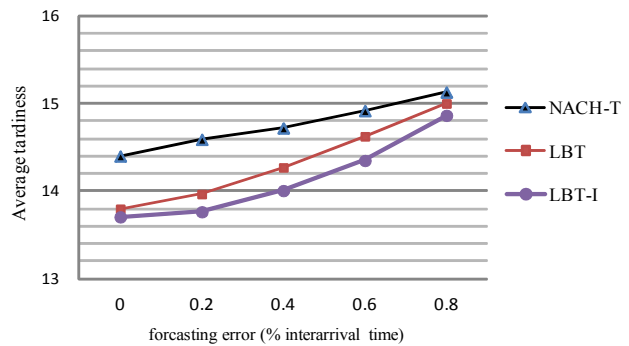


Figure 7. Impact of product arrival forecasting errors on tardiness

5. 결론 및 향후 연구 방향

본 논문은 배치공정에서 납기지연을 최소화 하기 위한 로딩전략인 LBT(look-ahead batching for tardiness minimization)를 제시하고 시뮬레이션 실험을 통하여 그 수행도를 검증하였다. 기존의 납기만족과 관련된 방법과 비교하여 본 논문에서 제시한 방법의 특징은 (1) 제품의 로딩 시점 이후에 도착하는 제품의 납기지연시간을 의사결정 항목에 포함하여 한 시점에서의 로딩이 이후에 도착하는 제품의 납기만족에 어떠한 영향을 주는 지 고려했다는 점과, (2) look-ahead를 일정한 미래도착 제품까지 만으로 제한하여 고려하는 제품의 수에 의해 의사결정에 영향을 받지 않도록 하고, (3) 의사결정시점에 대기하고 있는

제품의 수가 배치공정의 최대수용능력을 넘을 지라도 더 긴급한 제품이 가까운 미래에 도착하면 기다려 이를 우선적으로 처리할 수 있다는 특징이 있다. 시뮬레이션 실험 결과 배치공정의 TI 가 0.5~0.7인 경우에 특히 기존의 타 방법보다 우수한 결과를 얻을 수 있음을 알 수 있었다. LBT를 개선한 LBT-I는 LBT의 수행도를 추가로 향상시킨다는 것을 실험을 통하여 확인하였다. 특히 LBT-I는 제품도착시간에 대한 예측 오차가 발생하는 경우에 LBT보다 향상된 수행도를 나타냄을 알 수 있었다.

본 논문에서 다룬 배치공정의 로딩 전략은 단일기계상황에서 모든 제품이 하나의 배치를 구성할 수 있는 환경을 가정하고 있다. 실제 반도체 웨이퍼랩 생산현장에는 배치공정을 위한 동일한 기능의 장비가 여러 대 존재하고 제품도 종류에 따라서 서로 동일한 배치로 생산될 수 없는 경우가 존재할 수 있다. 이러한 상황에서의 배치공정 로딩전략에 대한 연구는 본 연구를 기반으로 추가로 수행될 예정이다.

참고문헌

- Akcali, E., Uzsoy, R., Hiscock, D., Moser, A. and Teyner, T. (2000), Alternative loading and dispatching policies for furnace operations in semiconductor manufacturing : a comparison by simulation, *Proceedings of Winter Simulation Conference*, 1428-1435.
- Avramidis, A. N., Healy, K. J., and Uzsoy, R. (1998), Control of a batch-processing machine : A computational approach, *International Journal of Production Research*, **36**(11), 3167-3181.
- Cerecki, A. and Banerjee, A. (2010), Dynamic control of the batch processor in a serial-batch processor system with mean tardiness performance, *International Journal of Production Research*, **48**(5), 1339-1359.
- Chun, K. W. and Hong, Y. (1996), Batch sizing heuristic for batch processing workstations in semiconductor manufacturing, *Journal of the Korean Institute of Industrial Engineers*, **22**(2), 231-245.
- Deb, R. K. and Serfozo, R. F. (1973), Optimal control of batch service queues, *Advances in Applied Probability*, **5**, 340-361.
- Fowler, J. W., Phillips, D. T. and Hogg, G. L. (1992), Real-time control of multiproduct bulk-service semiconductor manufacturing processes, *IEEE Transactions on Semiconductor Manufacturing*, **5**(2), 158-163.
- Fowler, J. W., Phojanamongkolkij, N., Cochran, J. K., and Montgomery, D. C. (2002), Optimal batching in a wafer fabrication facility using a multiproduct G/G/c model with batch processing, *International Journal of Production Research*, **40**(2), 275-292.
- Glasse, C. and Weng, W. (1991), Dynamic batching heuristic for simultaneous processing, *IEEE Transactions on Semiconductor Manufacturing*, **4**(2), 77-82.
- Gupta, A. and Sivakumar, A. (2006), Optimization of due-date objectives in scheduling semiconductor batch manufacturing, *International Journal of Machine Tools and Manufacture*, **46**, 1671-1679.
- Kim, Y. D., Kim, J. G., Choi, B., and Kim, H. U. (2001), Production scheduling in a semiconductor wafer fabrication facility producing multiple product types with distinct due dates, *IEEE Transaction on Robotics and Automation*, **17**(5), 589-598.
- Kim, Y. D., Joo, B. J., and Choi, S. Y. (2010), Scheduling wafer lots on diffusion machines in a semiconductor wafer fabrication facility, *IEEE Transactions on Semiconductor Manufacturing*, **23**(2), 246-254.
- Koo, P. H. and Moon, D. H. (2013), A review on control strategies of batch processing machines in semiconductor manufacturing, *IFAC Proceedings Volumes (IFAC-Papers Online)*, 1690-1695.
- Leachman, R. C., Kang, J., and Lin, V. (2002), SLIM : Short cycle time and low inventory in manufacturing at Samsung Electronics, *Interfaces*, **32**(1), 61-77.
- Murray, S., Geraghry, J., and Young, P. (2008), Time-limited next arrival heuristic for batch processing and setup reduction in a re-entrant environment, *Proceedings of Winter Simulation Conference*, 2109-2117.
- Neuts, M. F. (1967), A general class of bulk queues with Poisson input, *Annals of Mathematical Statistics*, **38**(3), 759-770.
- Neuts, M. F., Nadarajan, R. (1982), A multiserver queue with thresholds for the acceptance of customers into service, *Operations Research*, **30**(5), 948-960.
- Park, H. and Banerjee, A. (2011), A new dynamic scheduling for batch processing systems using stochastic utility evaluation function, *Proceedings of Winter Simulation Conference*, 2307-2319.
- Phojanamongkolkij, N., Fowler, J. W. and Cochran, J. K. (2002), Determining operating criterion of batch processing operations for wafer fabrication, *Journal of Manufacturing Systems*, **21**(5), 363-379.
- Robinson, J. K., Fowler, J. W., Bard, J. F. (1995), The use of upstream and downstream information in scheduling semiconductor batch operations, *International Journal of Production Research*, **33**(7), 1849-1869.
- Sha, D., Hsu, S., and Lai, X. (2007), Design of due-date oriented look-ahead batching rule in wafer fabrication, *International Journal of Advanced Manufacturing Technology*, **35**, 596-609.
- Van Der Zee, D., Harten, A., and Schuur, P. C. (1997), Dynamic job assignment heuristics for multi-server batch operations-A cost-based approach, *International Journal of Production Research*, **35**, 3063-3093.
- Van Der Zee, D. (2001), Harten, A. and Schuur, P.C. On-line scheduling of multi-server batch operations, *IIE Transactions*, **33**, 569-586.
- Weng, W. and Leachman, R. C. (1993), An improved methodology for real-time production decisions at batch-process work stations, *IEEE Transactions on Semiconductor Manufacturing*, **6**(3), 219-225.