Daily Scheduling for R&D Semiconductor Fabrication*

DA-YIN LIAO, SHI-CHUNG CHANG†
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, 10764

SHOU-REN YEN, CHENG-CHUNG CHIEN
Electronics Research & Service Organization
Industrial Technology Research Institute
Hsin-Chu, Taiwan, 301

Abstract

This paper addresses the daily scheduling for a research and development (R&D) pilot line of semiconductor wafer fabrication. An integer programming problem formulation is first given, which captures the salient features such as high variety and very low volume, cyclic process flows, batching at diffusion machines, single mask for each photolithography operation, loop test and engineering splitting and merging of wafer lots. A solution methodology for scheduling flow shops based on Lagrangian relaxation [3] is then extended to solve this class of problems. It may also effectively handle data inaccuracy and cope with production uncertainties. Numerical results demonstrate both the feasibility and potential of this method.

I Introduction

Integrated circuit fabrication is a capital- and technology-intensive business with fierce global competition. It is perhaps one of the most complex manufacturing processes ever found today. In an IC fabrication plant (*fab*), there may be tens of fabrication processes. Each process may contain 200-300 processing steps and over one hundred equipments are involved. There exist high uncertainties in operations such as frequent machine failures and fluctuation of yield rates. Production planning and control of IC fabrication are thus quite complex. Development of effective and efficient production planning and control for IC wafer fabrication remains a very challenging research topic.

Wein [12] pointed out that production control issues in descending order of significance to IC fab performance are (1) wafer release policy, (2) daily scheduling and (3) lot dispatching. On the one hand, wafer release is calculated using one day or one week as a time unit over a long time horizon of two to four months. Daily or weekly wafer outs (throughputs) may also be determined as a byproduct. On the other hand, lot dispatching is done in the shop floor to respond properly to fab states in real time and to meet daily production targets. Daily scheduling bridges between the aforementioned two levels of functions; it breaks daily production targets into a schedule in a time scale of one to three hours over a day by considering fab dynamics. The schedule then sets targets for lot dispatching.

There have been some results on these three issues in the literature [13] such as the Bottleneck Starvation Prevention method for wafer release [8], the stochastic optimal control method for short-term scheduling [9], the hierarchical flow control method for scheduling/dispatching [2] and the target generation and machine allocation (TG&MA) algorithm [5]. Furthermore, Chen *et al.* [4] developed a queueing network model for a semiconductor wafer fab where the presence of both production and development lots are addressed. However, none of them addressed the distinct production control problems in a research and development (R&D) IC pilot line.

An R&D IC pilot line has salient characteristics of

1. very small volume but high varieties of engineering wafers;
2, frequent process changes;
3. frequent engineering splitting and merging;
4. the existence of loop tests, i.e., certain processing steps of lots cannot be done until some loop test lots complete the testing steps;
5. extensive on-line and off-line inspections and lot holding due to experiments for engineering purposes;
6. many dedicated machines without a backup; and
7. inaccurate process data due to lack of historical data.

In this paper, we focus on the daily scheduling problem of an R&D IC pilot line.

The remainder of this paper is organized as follows. The scheduling problem is formulated in Section II and the development of a baseline solution method for the scheduling problem is described in III. Section IV describes further algorithm developments for handling data inaccuracy and for coping with production uncertainties. Computational complexity of the algorithm is also discussed. Numerical results are demonstrated in Section V. Section VI concludes this paper.

*This work was supported in part by the National Science Council of Republic of China under Grant NSC81-0416-E002-567 and by ERSO of ITRI.

†The authors would like to thank Dr. Chi-Ming Chang, Ms. S.-L. Sun, Mr. D.-S. Wu and engineers of ERSO, ITRI for their valuable discussion and data collection.

1050-4729/93 $3.00 © 1993 IEEE

II Scheduling Problem Formulation

Consider an IC fab which fabricates IC wafers for the purpose of research and development (R&D). Wafers of the same processing requirements are stored in a cassette as a lot, which has a maximum of 24 pieces of wafers. There are usually tens of wafer types in an R&D pilot line, each having a volume of 2-4 lots. There are machine groups of various functionality; each machine group may consist of a few homogeneous machines. The process flow of each lot can be organized as a sequence of stages. A stage consists of a few processing steps, whose requirements and conditions are specified by recipes. Different lots and stages may use one same machine. There are many revisits to a machine in the process flow of a lot due to the layered nature of IC fabrication. That is, there are cycles in production flow paths. There are relatively large buffer spaces in the shop floor that buffer space availability does not pose a significant constraint on production flows. Based on field engineers' descriptions and data availability, we assume that

1. setup time for changing lots of production at each machine can be estimated and incorporated as part of the processing time;
2. setup cost is negligible;
3. lot transportation time and cost from one machine to the other are negligible;
4. daily wafer releases and production targets are given;
5. machine capacity and process flow data are available; and
6. there are no rework and scrap of wafers.

To focus on the development of the core scheduling algorithm, we postpone the discussion of handling rework, scrap, production uncertainties and data inaccuracy to Section IV.

Let us first define some notations for describing such an R&D IC pilot line.

Notations:

i: wafer type index;

S_i: total number of stages for type-i wafers;

(i,s): stage s of type-i wafers, $s = 1, \cdots, S_i$;

$P_{(i,s)}$: processing time of stage (i,s);

m: machine group index;

$M(i,s)$: the machine group where stage (i,s) can be processed;

T: the scheduling horizon;

t: time period index, $t = 1, \ldots, T$;

C_{mt}: capacity of machine group m at time period t;

$X_{(i,s)t}$: number of type-i wafer lots ready for processing stage (i,s) at the beginning of time period t;

l_{it}: number of type-i wafer lots released at the beginning of time period t;

r: recipe index;

P_r: processing time of recipe r;

$R(i,s)$: the recipe that is needed for processing stage (i,s);

$N(r)$: the diffusion machine where a diffusion recipe r can be processed;

$\overline{B}(\underline{B})$: the maximum (minimum) number of lots in a batch for a diffusion machine;

$DIFF$: the set of diffusion machine groups;

$PHOTO$: the set of photolithography machines;

$\psi_{(i,s)}$: weighting factor for processing a lot at stage (i,s);

$y_{(i,s)}$: daily move target of stage (i,s).

Decision Variables:

$u_{(i,s)t}$: number of type-i wafer lots to be loaded onto machines for processing stage (i,s) at time t;

b_{rt}: number of batches to be formed for processing recipe r at time period t;

In the process flow of type i wafers, the wafers loaded onto machines for processing $(i, s-1)$th stage at period $t - P_{(i,s-1)}$ go to the buffer at stage (i,s) after $P_{(i,s-1)}$ periods of processing. As flows of wafers are not compressible, they must satisfy

Flow Balance Equations for each type i

$$X_{(i,1)(t+1)} = X_{(i,1)t} - u_{(i,1)t} + l_{it}, \ \forall t; \qquad (2.1.a)$$

$$X_{(i,s)(t+1)} = X_{(i,s)t} - u_{(i,s)t} + u_{(i,s-1)(t-P_{(i,s-1)})}, \quad \forall s = 2, 3, \cdots, S_i, \ \forall t; \qquad (2.1.b)$$

$$X_{(i,S_i+1)(t+1)} = X_{(i,S_i+1)t} + u_{(i,S_i)(t-P_{(i,S_i)})}, \ \forall t; \qquad (2.1.c)$$

$$X_{(i,s)1} \text{ given}, \ \forall s; \qquad (2.1.d)$$

$$l_{it} \text{ given}, \ \forall t; \qquad (2.1.e)$$

$$u_{(i,s)(-P_{(i,s)}+1)}, u_{(i,s)(-P_{(i,s)}+2)}, \cdots, u_{(i,s)0} \text{ given}, \ \forall s. \qquad (2.1.f)$$

Since a diffusion operation takes a relatively long processing time and a diffusion machine can process many wafers at the same time, lots of the same operating conditions (i.e., lots requiring the same recipe) are usually batched together for diffusion. For a recipe r, $\sum_{\substack{(i,s) \\ R(i,s)=r}} u_{(i,s)t}$ is the total number of lots that can be batched together for processing at time period t. Then,

Batching Constraints

$$\underline{B} \cdot b_{rt} \le \sum_{\substack{(i,s) \\ R(i,s)=r}} u_{(i,s)t} \le \overline{B} \cdot b_{rt}, \quad \forall r \text{ and } \forall t. \qquad (2.2)$$

As the processing capacity of a diffusion machine is also expressed in the unit of batch, the total number of batches being processed by a diffusion machine can not exceed its capacity during each time period, i.e.,
Machine Capacity Constraints (Diffusion Area)

$$\sum_{\substack{r \\ N(r)=m}} \sum_{\tau=t-P_r+1}^{t} b_{rt} \leq C_{mt}, \ \forall m \in DIFF \text{ and } \forall t. \tag{2.3.a}$$

For a non-diffusion stage, the $u_{(i,s)t}$ lots loaded onto the machine group $M(i,s)$ need $P_{(i,s)}$ periods to complete the processing. During a time period t, there is a total of $\sum_{\substack{(i,s) \\ M(i,s)=m}} \sum_{\tau=t-P_{(i,s)}+1}^{t} u_{(i,s)\tau}$ lots being processed by machine group m. This quantity must not exceed the processing capacity of machine group m,
Machine Capacity Constraints (Non-Diffusion Area)

$$\sum_{\substack{(i,s) \\ M(i,s)=m}} \sum_{\tau=t-P_{(i,s)}+1}^{t} u_{(i,s)\tau} \leq C_{mt}, \ \forall m \notin DIFF \text{and } \forall t. \tag{2.3.b}$$

There is only single mask available for each photolithography step, which in turn limits the machine capacity for processing it to at most one machine at a time, i.e.,
Single Mask Constraints

$$u_{(i,s)t} \leq C_{mt}, \ \forall \ m = M(i,s) \in PHOTO. \tag{2.4}$$

Frequent engineering splitting and merging are two important features in an R&D IC pilot line. Let M be a set of stages where several lots are merged into one lot. Let $S^{M}_{(i',s')} \equiv \{(i,s)\}$ be a set of processing stages whose finished lots will be merged into one type-i' lot for further processing of a stage (i',s'). Let $Y_{(i,s)t}$ be the number of wafer lots which have completed stage (i,s) and are ready for the processing of stage (i',s') at time period t. Similar to $\{X_{(i,s)t}\}$, $\{Y_{(i,s)t}\}$ must satisfy the following flow balance equations, i.e.,
Flow Balance Equations for Merging

$$Y_{(i,s)(t+1)} = Y_{(i,s)t} - u_{(i',s')t} + u_{(i,s)(t-P_{(i,s)})},$$

$$\forall (i,s) \in S^{M}_{(i',s')} \text{ and } (i',s') \in M \text{ and } \forall t. \tag{2.5}$$

Furthermore, the number of merged lots cannot exceed what are available for merging,
Merging Constraints

$$u_{(i',s')t} \ \leq \ Y_{(i,s)t}, \ \forall (i,s) \in S^{M}_{(i',s')} \text{ and } (i',s') \in M, \ \forall t. \tag{2.6}$$

The splitting of a lot is just the reverse of merging, where one lot becomes a few lots of different types. Let S be a set of stages whose lots are generated by splitting into several lots. Let $S^{S}_{(i',s')} \equiv \{(i,s)\}$ be the set of stages whose lots are generated by splitting from a lot of stage (i',s'). In a principle similar to that of (2.5), we have
Flow Balance Equations for Splitting

$$X_{(i',s')(t+1)} = X_{(i',s')t} - u_{(i',s')t} + u_{(i,s)(t-P_{(i,s)})},$$

$$\forall (i,s) \in S^{S}_{(i',s')} \text{ and } (i',s') \in S \text{ and } \forall t. \tag{2.7}$$

The model here does not include unexpected splitting or conditional splitting that depends on the processing result of a stage.

A loop test bears much resemblance to the planned engineering splitting and merging; it is initiated by the completion of certain stages of a few lots, which can therefore be viewed as a pseudo merging; when the loop test finishes, these lots may resume their individual processing flows, which corresponds to a splitting. The techniques for modeling merging and splitting developed above can thus be applied to model loop tests.

Unlike mass production IC fabs, engineering experimentations and inspection stages are frequently added to the original processes of a pilot line on a daily basis. Such a feature can be easily handled by updating the process flow database and requires no extra modeling efforts. However, the processing time data for these stages is usually estimated roughly by experienced engineers. Treatment of such data inaccuracy will be discussed in Section IV.

Another objective of daily scheduling is to meet the daily production targets and is modeled as an inequality constraint,
Daily Production Target Constraints

$$\sum_{t} u_{(i,s)t} \geq y_{(i,s)}, \ \forall \ (i,s). \tag{2.8}$$

Integrality Constrains
$u_{(i,s)t}$, $X_{(i,s)t}$ and b_{rt} are nonnegative integers,

$$\forall (i,s), \forall r \text{ and } \forall t. \tag{2.9}$$

Our objective of daily scheduling is to maximize the total weighted production (or *moves*) of wafer lots in the fab. The complete daily scheduling problem is formulated as

$$\max_{\mathbf{u},\mathbf{b}} \quad \sum_{(i,s)} \sum_{t} \psi_{(i,s)} u_{(i,s)t}$$

subject to (2.1-2.9), or equivalently,

$$(P) \qquad \min_{\mathbf{u},\mathbf{b}} \quad -\sum_{(i,s)} \sum_{t} \psi_{(i,s)} u_{(i,s)t}$$

subject to (2.1-2.9).

III A Baseline Solution Method

The scheduling problem (P) formulated in Section II is an integer programming problem of NP-hard computational complexity [11]. We now develop a solution algorithm by extending the previous results on flow shop scheduling based on the methodology of Lagrangian relaxation [3, 7].

III.1 Relaxation and Decomposition

In problem (P), the coupling among production flows of different wafer types is caused by their competition for processing resources, i.e., through the machine capacity constraints and batching constraints. Constraints (2.5-2.7) reflect coupling among lots due to merging and splitting. Constraint (2.8) represents a coupling of production decisions over time periods for each wafer type. Based on these observations, we apply Lagrangian relaxation to relax machine capacity constraints (2.3.a) and (2.3.b), batching constraints (2.2), merging constraints (2.5) and (2.6), splitting constraints (2.7) and daily target constraints (2.8), exploit the separability of the relaxed problem and form a dual problem as

$$(D) \qquad \max_{\substack{\lambda\geq 0,\pi\geq 0,\mu\geq 0\\ \nu\geq 0,\delta\geq 0,\sigma\geq 0\\ \zeta\geq 0,\eta\geq 0}} \Phi(\lambda,\pi,\mu,\nu,\delta,\sigma,\zeta,\eta),$$

where $\boldsymbol{\lambda}= \{\lambda_{mt}\}$, $\boldsymbol{\pi}= \{\pi_{mt}\}$, $(\boldsymbol{\mu}= \{\mu_{rt}\}$, $\boldsymbol{\nu}= \{\nu_{rt}\})$, $\boldsymbol{\delta}= \{\delta_{(i,s)}\}$, $\boldsymbol{\sigma}= \{\sigma_{(i,s)}\}$, $\boldsymbol{\zeta}= \{\zeta_(i,s)\}$ and $\boldsymbol{\eta} = \{\eta_{(i,s)}\}$ are the Lagrange multipliers for relaxing constraints (2.3.a), (2.3.b), (2.2), (2,5), (2.7), (2.6) and (2.8) respectively, and

$$\begin{aligned}\Phi(\lambda,\pi,\mu,\nu,\zeta,\delta,\sigma,\eta) \equiv & \sum_i PL_i(\lambda,\mu,\nu,\zeta,\delta,\sigma,\eta)\\ + & \sum_r\sum_t BL_{rt}(\pi,\mu,\nu)\\ - & \Bigg(\sum_{\substack{m\\ m\notin DIFF}} \lambda_{mt} + \sum_{\substack{m\\ m\in DIFF}} \pi_{mt}\Bigg)C_{mt}\\ + & \sum_{(i,s)} \eta_{(i,s)}y_{(i,s)},\end{aligned}$$

with *(1) production scheduling subproblem for type-i wafers*
$(PS-i)$ $PL_i(\lambda,\mu,\nu,\zeta,\delta,\sigma,\eta) \boxminus$

$$\begin{aligned}\min_{u_i}\Bigg\{\sum_s\sum_t\Bigg[\Bigg(&\sum_{\substack{\tau=t\\ m=M(i,s)}}^{t+P_{(i,s)}-1} \lambda_{m\tau} - \psi_{(i,s)} - \eta_{(i,s)}\Bigg)\\ &+ \sum_{\substack{r\\ r=R(i,s)}} \Bigg(\nu_{rt} - \mu_{rt}\Bigg)\Bigg]\cdot u_{(i,s)t}\\ + \sum_{\substack{s\\ (i,s)\in M}} \sum_{(i',s')\in S^M_{(i,s)}} &\delta_{(i,s)}\Bigg[Y_{(i',s')(t+1)} - Y_{(i',s')}\\ &+u_{(i,s)t} - u_{(i',s')(t-P_{(i',s')})}\Bigg]\\ + \sum_{\substack{s\\ (i,s)\in S}} \sum_{(i',s')\in S^S_{(i,s)}} &\sigma_{(i,s)}\Bigg[X_{(i,s)(t+1)} - X_{(i,s)}\\ &+u_{(i,s)t} - u_{(i',s')(t-P_{(i',s')})}\Bigg]\\ + \sum_{\substack{s\\ (i,s)\in M}} \sum_{(i',s')\in S^M_{(i,s)}} &\zeta_{(i,s)}\Bigg[u_{(i,s)t} - Y_{(i',s')t}\Bigg]\Bigg\},\end{aligned}$$

subject to (2.1), (2.4) and (2.9);
and *(2) batch allocation subproblem for recipe r at time period t*
$(BA-rt)$ $BL_{rt}(\pi,\mu,\nu) \equiv$

$$\min_{b_{rt}}\sum_r\sum_t\Bigg(\sum_{\substack{\tau=t\\ m=N(r)}}^{t+P_r-1} \pi_{m\tau} + \underline{B}\cdot\mu_{rt} - \overline{B}\cdot\nu_{rt}\Bigg)\cdot b_{rt},$$

subject to (2.9).

Note that for a given set of Lagrange multipliers $\boldsymbol{\lambda}$, $\boldsymbol{\pi}$, $\boldsymbol{\mu}$, $\boldsymbol{\nu}$, $\boldsymbol{\zeta}$, $\boldsymbol{\delta}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\eta}$, $(PS-i)$ corresponds to production scheduling of type i wafer lots with no capacity limitation and $(BA-rt)$ corresponds to the batching for a diffusion recipe r at time t, respectively. Subproblems are independent of each others and can be solved individually.

III.2 Solutions for Subproblems

The set of flow balance equations (2.1) of $(PS-i)$ render themselves naturally to a network representation, where each node in the network is associated with a flow balance equation and the flow on an arc between two nodes corresponds to a wafer flow. The flow conservation at each node then represents one of the flow balance equations. The capacity of an arc in the network is bounded by its corresponding machine or buffer capacity. Arc costs are set according to the cost coefficients in $(PS-i)$. Subproblem $(PS-i)$ is essentially a minimum cost linear network flow (MCLNF) problem, whose integer optimal solution can be found by polynomial time algorithms [11]. We adopt the RELAX code [1] to solve each $(PS-i)$.

Each subproblem $(BA-rt)$ is a simply constrained, static, linear integer optimization problem. The complementary slackness conditions [10] are used to determine its solution. Given $\boldsymbol{\pi}$, $\boldsymbol{\mu}$, and $\boldsymbol{\nu}$, we set b_{rt}=0 or C_{mt} depending on whether the sign of the cost coefficient of b_{rt} in $(BA-rt)$ is nonnegative or negative.

III.3 Dual Problem Solution

After solving all the subproblems for a given set of $(\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\nu},\boldsymbol{\zeta},\boldsymbol{\delta},\boldsymbol{\sigma},\boldsymbol{\eta})$, we use the resulting solution to update $(\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\nu},\boldsymbol{\zeta},\boldsymbol{\delta},\boldsymbol{\sigma},\boldsymbol{\eta})$. Aware of the nondifferentiability of the dual objective function Φ due to the integrality constraints in subproblems, we calculate the subgradient of Φ with respect to $(\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\nu},\boldsymbol{\zeta},\boldsymbol{\delta},\boldsymbol{\sigma},\boldsymbol{\eta})$ and update $(\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\nu},\boldsymbol{\zeta},\boldsymbol{\delta},\boldsymbol{\sigma},\boldsymbol{\eta})$ according to the subgradient method of [6]. The dual problem (D) is then solved iteratively by solving subproblems and updating Lagrange multipliers in each iteration.

Remark:

As the primal problem (P) is not a convex optimization problem because of integer decision variables, the solution to the dual problem generally results in an infeasible schedule, i.e., some of the relaxed constraints may be violated. However, the dual cost does provide a lower bound to the optimal cost of (P).

III.4 Construction of a Feasible Schedule

An iterative, model-based heuristic algorithm is then developed that adjusts the dual solution to a near-optimal, feasible schedule by taking advantage of the marginal cost interpretation of Lagrange multipliers and the network structure of the flow balance equations. The heuristic algorithm checks all facilities and all recipes over all time periods to see whether their respective capacity and batching constraints are satisfied. When there is a constraint violation, some excessively scheduled lots have to be removed and rescheduled to resolve the violation. Once the excessive lots are removed, they are then rescheduled by utilizing the residual capacities or slacks. Both the removing and rescheduling are done by following the principle of minimizing the change of production cost in order to stay close to the lower bound (dual) cost, i.e., to achieve a near-optimal solution. The realization of these two heuristics is first through proper parameter modification of the involved subproblem(s) and then by solving a MCLNF or complementary slackness problem of the modified subproblem(s).

IV Further Developments

Many salient features of an R&D pilot line have been captured in the problem formulation to which Section III provides a baseline solution method. This section describes further algorithm developments for handling data inaccuracy and for coping with production uncertainties. As these further developments are based on the baseline solution method, a simple computational complexity analysis of the baseline algorithm is first given.

IV.1 Complexity Analysis

The major computational loads of the baseline solution algorithm lie in solving subproblem $(PS-i)$'s and subgradient iterations. It is known that the computational complexity of the RELAX code is $O(\mathrm{N}^3 log \mathrm{NC})$, where N is the number of nodes in the network and C is the range of arc cost coefficients and that the convergence of subgradient iterations slows down as the number of Lagrange multipliers increases.

In an R&D pilot line, each type of wafers is of a small volume ranging from two to four lots. These different lots of the same type are usually distributed in the wafer-in-process (WIPs) of nearby processing stages at the beginning of a day. Each lot may only go through a few (no more than 5, empirically) consecutive stages of processing during one day because of the long processing and/or setup times plus time of waiting. These facts imply that only a small portion of the processing stages (e.g., 10 out of 100) need to be considered in each subproblem $(PS-i)$. Namely, the network for representing $(PS-i)$ to an hourly resolution has approximately 24×10 nodes and at most 4 units (lots) of flows on it. A $(PS-i)$ can therefore be solved very efficiently by RELAX.

IV.2 Coping with Uncertainties

Production uncertainties such as unscheduled machine failures, process changes, rework and scrap, etc., occur quite frequently in an R&D pilot line. The schedule by the baseline algorithm provides a nominal schedule for a day. Along a line of thinking similar to [3], we develop an optimization-based, fast rescheduling algorithm for timely adjusting the nominal schedule to cope with small disturbances and perform periodic re-scheduling using the baseline algorithm to adjust to large or cumulative disturbances. The fast rescheduling algorithm exploits ideas and steps of the heuristic algorithm in Section III.4 by removing unrealized portions of the original schedule and then properly re-scheduling them according to the actual system status after the occurrence of an uncertain event. The periodic re-scheduling utilizes the original profiles of Lagrange multipliers as a starting point of the iterations and aims at responding to disturbances in a longer time scale. As long as the system status does not vary too much, re-scheduling with this initialization will much reduce the computation time of converging to a new schedule.

IV.3 Handling Data Inaccuracy

Much of the processing data for R&D wafer lots is estimated either by process engineers or line operators due to lack of historical data. Computationally efficient sensitivity analysis of how processing data affects the performance of a schedule can be developed by exploiting the network structure and the dual problem formulation of the baseline method. For example, it has been recognized that the optimal Lagrange multipliers associated with machine capacity constraints represent the sensitivity of the cost function with respect to the level of machine resources if the integrality requirement is relaxed [10]. A multiplier can therefore be interpreted as a *shadow price* for using a unit of machine capacity. Such a view point can be extended to evaluate the impacts of a wide variety of parameter changes. An interactive software environment will be developed to help supervisors of the pilot line identifying the critical processing data estimates and performing what if analysis (e.g., the effects of hot lots). Data inaccuracy will make the actual evolution of the line during a day deviate from the nominal schedule even when there are no uncertainties of the types described in IV.2. Application of the adjustment schemes of IV.2 helps alleviating the deviation due to data inaccuracy.

V Numerical Results

Numerical experimentation of the daily scheduling algorithm are performed on a SUN/SPARC-II workstation.

V.1 A Simple Example

Consider a simple example where there are four machines D, I, E and P, for steps of diffusion, implantation, etching and photolithography, respectively. Capacities of I, E and P are 1 lot/time period. The maximum capacity of D is 4 lots per batch. There are two lots (A and B) to be scheduled. Processing requirements are given in Tables V.1. Lot A is initially ready for processing stage (A,P1) and lot B ready for processing stage (B,E). The weighting factor and daily production target for each stage of the two lots are given in Table V.2. The scheduling horizon is 8 time periods. The resultant feasible schedule by applying our daily scheduling algorithm is shown in Table V.3, with a weighted production cost of -690.0 and computation time of 3.21 seconds. The dual cost is also -690.0, which implies that the optimal solution is achieved for this simple example.

Table V.1: Processing Times of Lots A & B

Stage	P1	E	D1	P2	I	D2
Mach.	P	E	D	P	I	D
Lot A	1	2	4	1	1	5
Lot B	1	1	4	1	1	5

Table V.2: Weighting Factors and Targets

Stage	P1	E	D1	P2	I	D2
ϕ_A	20	10	10	10	0	0
y_A	1	1	1	1	0	0
ϕ_B	0	200	200	100	100	10
y_B	0	1	1	1	1	1

Table V.3: Daily Schedule of Lots A and B

Time	1	2	3	4
P	(A,P1)			
E	(B,E)	(A,E)		
I				
D		(B,D1)	(B,D1)	(B,D1)
Time	5	6	7	8
P		(B,P2)		
E				
I			(B,I)	(B,I)
D	(B,D1)	(A,D1)	(A,D1)	(A,D1)

V.2 A Realistic Case

In this subsection, the daily scheduling algorithm is applied to a realistic case of a local R&D IC fab. There are 60 machine groups and 41 lots in the fab to be scheduled. The time unit of scheduling is an hours and the horizon is 24 hoursl. The daily production targets and weighting factors are set by field engineers. Our algorithm takes 215.270 CPU seconds to obtain a feasible schedule with a cost of -33700. The corresponding dual cost is -33927.8 and the resultant relative duality gap is 0.676%. Such a solution can be considered near-optimal. The detailed schedule is considered very reasonable after the review of field engineers. Moverover, it takes much less time to generate a better schedule than that currently generated by the production meeting every morning.

VI Conclusions

We have developed an integer programming problem formulation for the daily scheduling problem of an R&D IC pilot line. The formulation has captured the salient features of this specific type of flexible manufacturing systems. A solution methodology based on Lagrangian relaxation has also been proposed, which is expected not only to be efficient and near-optimal but also to provide an effective method for coping with dynamic changes in the pilot line. Currently, equations (2.5-2.7), which reflect the coupling between wafer flows due to merging and splitting, are handled by a straight forward relaxation. Computational efficiency can be much improved by adopting an advanced multiplier method [7] to handle them. Further algorithmic development and numerical studies using realistic system data will be reported in the future.

References

[1] Bertsekas, D. P., and P. Tseng, "Relaxation Methods for Minimum Cost Ordinary and Generalized Network Flow Problems," *Operations Research*, Vol. 36, No. 1, pp. 93-114, 1988.

[2] Bai, S. X., N. Srivatsan and S. B. Gershwin, "Hierarchical Real-Time Scheduling of a Semiconductor Fabrication Facility," *Proc. of the 9th IEEE Inter. Electronics Manufacturing Technology Symposium*, Washington, D. C., October 1990.

[3] Chang, S.-C., and D.-Y. Liao, "Scheduling Flexible Flow Shops of No Setup Effects," *submitted to IEEE Trans. on Robotics and Automation*, October 1991.

[4] Chen, H., J. M. Harrison, A. Mandelbaum, A. Van Ackere and L. M. Wein, "Empirical Evaluation of a Queueing Network Model for Semiconductor Wafer Fabrication," *Operations Research* 36, No. 2, pp. 202-215, 1988.

[5] Chang, S.-C., L.-H. Lee, L.-S. Pang, Y.-C. Chang, P.-C. Lin and W.-Y. Chen, "Daily Target Generation and Machine Allocation for Integrated Circuit Fabrication," *Proceedings of the 2nd Inter. Conf. on Automation Technology*, July 1992.

[6] Held, M., P. Wolfe, and H. Crowder, "Validation and Subgradient Optimization," *Math. Programming*, Vol. 6, 1974, pp. 62-88.

[7] Hoitomt, D. J., P. B. Luh and K. R. Pattipati, "A Lagrangian Relaxation Approach to Job Shop Scheduling Problems," *Proceeding of 1990 IEEE Intern. Conf. on Robotics and Automation*, Ohio, May 1990, pp. 1944-1949.

[8] Lozinski, C., and C. R. Glassey, "Bottleneck Starvation Indicators for Shop Floor Control," *IEEE Trans. on Semiconductor Manufacturing*, Vol. 1, No. 4, Nov. 1988.

[9] Lou, S. X. C., and P. W. Kager, "A Robust Production Control Policy for VLSI Wafer Fabrication," *IEEE Trans. on Semiconductor Manufacturing*, Vol. 2, No. 4, Nov. 1989.

[10] Luenberger, D. G., *Linear and Nonlinear Programming*, Addison-Wesley, 1984.

[11] Papadimitriou, C. H., and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Inc., 1982.

[12] Wein, L. M., "Scheduling Semiconductor Wafer Fabrication," *IEEE Trans. on Semiconductor Manufacturing*, Vol. 1, No. 3, August 1988, pp. 115-130.

[13] Uzsoy, R., C. Y. Lee and L. A. Martin-Vega, "A Survey of Production Planning and Scheduling Models in the Semiconductor Industry Part I: System Characteristics, Performance Evaluation and Production Planning," *IEE Trans.* Vol. 24, No. 4, pp. 47-61, 1992.