Event-Time Models for Supply Chain Scheduling

Ömer S. Benli

California State University, Long Beach
obenli@csulb.edu

Summary This study presents a modeling paradigm for scheduling problems in supply chains. The constituents of a supply chain need to cooperate, rather than compete, in order to achieve maximum respective benefits. To analyze this, it is essential to have a concise but comprehensive formulation of this problem. This formulation, in addition to being computationally viable, must account for special characteristics of supply chains. It is argued that *lot streaming* provides an appropriate paradigm for scheduling problems in supply chains. It is shown that *event-time models* provide a general formulation for lot streaming problems. Furthermore, logic-based modeling framework of constraint programming makes it possible to handle special requirements of these models. This is illustrated by a small example. It is also shown that the fundamental results for single-job lot streaming problems can be systematically obtained as special cases of this generalized formulation. This demonstrates that event-time formulation accurately models all of the features of the problem; and that starting with a concise but comprehensive formulation, solutions for special cases of the problem can be obtained systematically. To that end, a number of additional special case formulations of single-job lot streaming problem is presented. An appendix presents a schema for lot streaming problem classifications.

1 Introduction

Today's supply chains differ from the integrated logistics systems of the past primarily because of the autonomous nature of its constituents. In the traditional approach to integrated logistics, the entire system is treated as a monolithic entity, whereas today's supply chains are usually comprised of components that are autonomous entities with competing interests. These constituents of the supply chain, such as manufacturers, wholesalers, and retailers, will be better off if they operate in co-operation. Similar situation arises in *supply contracts.* (Cachon 2003) Co-operation via supply contracts results in a win-win outcome for all parties concerned. The same is true for supply chain scheduling, which is concerned with timing and amount of material handling moves throughout the supply chain. Supply chain scheduling has replaced the integrated production planning and scheduling systems of traditional logistics; and cooperation is essential in scheduling operations in supply chains.

At *"The Factory Scheduling Conference"* held at Carnegie Institute of Technology in May 1961, William Pounds argued that production scheduling problem was not "... a visible one in many firms because other parts of the firm have absorbed much of the impact of poor scheduling." (Pounds 1961) If the due dates were not routinely met, it was customary to give protracted due dates; if there were a bottleneck machine, the problem was solved by acquiring another machine. Although the changing nature of business competitiveness demands highly advanced production scheduling systems, sufficient emphasis is still not being given to scheduling in supply chains. In the indexes of two recent handbooks on supply chain management, 6 out of 765 pages in Graves & de Kok (2003) and only one page out of 817 pages in Simchi-Levi et al. (2004) directly refer to scheduling. de Kok & Fransoo (2003) suggests the following explanation: *"Decisions with regard to the different components of planning of supply chain operations have traditionally been analyzed independently from one another by the researchers. Research addressing the scheduling problem, the (multi-echelon) inventory problem, and the aggregate capacity planning problem have hardly been interconnected while maintaining their own characteristics."*

Production planning problems customarily are posed as periodic review processes. On the other hand, detailed scheduling problems extend over relatively shorter planning horizons and require continuous time domain. Among the first to comment on this incongruity was Karmarkar (1994); comparing the different modeling perspectives in theory and practice, he stated that:

> [Analytical models] often treat capacity in terms of loading time buckets. However, in *practice*, it is much easier to think in terms of time lines and events and intervals. In some models, such as scheduling with Gantt charts, we use this kind of modeling, but we lack ways of dealing with decomposition or composition of these models. As a result, capacity and planning models are often formulated very differently. Perhaps this is one reason that time interval and release oriented methods like MRP and DRP are used in practice for planning even though they are completely unable to actually deal with resource allocation decisions.

Likewise, Bill Maxwell questioned the use of models that lump operations and events into large time buckets and suggested formulation of *"event-time"* models which are in essence "ordered sequence of time and data." (Maxwell 1997)

2 Lot Streaming Paradigm

A major problem in production planning is how to handle sequencing requirements on resources, whereas models of traditional machine scheduling cannot handle lot sizing. *Lot streaming* may provide the necessary conceptual framework for integrating lot sizing and machine scheduling. Basically, lot streaming is moving some portion of a process batch ahead to begin a downstream operation. Classical machine scheduling theory envision an operation as an elemental task to be performed. It is assumed that "[t]he processing times of successive operations of a particular job cannot be overlapped. A job can be in process on at most one operation at a time" (Conway et al. 1967). This assumption is justified when jobs are monolithic entities. But for scheduling production lots, where each lot consists of a number of units, this assumption may be overly restrictive. The processing time of such a lot is comprised of a (usually "detached") setup time and the sum of the processing times of each unit in the lot. For instance, when the machine is available, it is not reasonable to delay its setup until *all* the items arrive from the upstream machine.

Lot streaming, in this context, was introduced in papers by Baker (1988) and Trietsch (1987). In a later joint work, they discuss the practical importance of this approach. (Trietsch & Baker 1993) A number of manufacturing management innovations, such as *Group Technology* (leading to cell based manufacturing, resulting in shorter lead times and reduced work in progress inventories), *Just-in-Time Systems* ("lot size of one"), and OPT/*Synchronous Manufacturing* (transfer *vs.* process batches) paved the way to lot streaming theory, which provides a rigorous analytical treatment of these issues. In the recent years lot streaming attracted considerable attention in machine scheduling research. Chang & Chiu (2005) present a comprehensive review of lot streaming literature.

There does not seem to exist a unified approach to solving lot streaming problems. Probably this is due to the fact that a general model for lot streaming problems does not exist. The event-time modeling approach, introduced in the next section, attempts to provide such a paradigm.

3 Event-Time Models

In event-time modeling *events* are ordered sequence of material handling moves (interstage material transfers). There are two types of events:

exogenous events whose time of occurrence are given (as parameters of the problem), such as demand occurrences or order deadlines, and

endogenous events whose occurrence times are decision variables of the model, such as WIP movements.

Essentially, the model is formulated as a multi-item periodic review process with *variable period lengths*. Let

$\mathbb{M}$ be the index set of all items i, and

$\mathbb{M}_e$ be the index set of items i that have external (independent) demand in addition to possible internal (dependent) demand. The items $i \in \mathbb{M}_e$ are called *end items*.

Also define

$\mathbb{P}(i)$ as the predecessor set of item i, which is the index set of items that are required in the procurement of item i, and

$\mathbb{S}(i)$ as the successor set of item i, which is the index set of items that require item i in their procurement.

Interstage transfers can occur at time points T_t, $t \in \mathbb{T}$, where $\mathbb{T} = \{1, \ldots, n\}$ is the index set of all such time points. These time points are decision variables of the problem except for the points corresponding to exogenous events: the given deadlines or other external demand occurrences. Let a subset of $\mathbb{T} \supset \mathbb{T}_e = \{\tau_1, \tau_2, \ldots, \tau_m\}$ denote set of time points corresponding to exogenous events.

A period, which has variable length, is defined as the time interval $[T_{t-1}, T_t], t \in \mathbb{T}$, and referred to as period t. The section of the process depicting the inventory and procurement operations corresponding to item i is called *stage* i. In each period, stage i contains two *buffers*. During period t, the items that will be needed in the procurement of item i are inventoried in *input buffer* and the procured item i, that is not yet transfered to the input buffers of successor items is stored in the *output buffer*. The corresponding material flow is depicted in Figure 1.

We can now define the following variables:

Y_{it} is the amount added to the input buffer of item i in period t, which consists of all predecessor items, in required proportions, needed for the procurement of item i. One unit of item i requires α_{ji} units of item j for all $j \in \mathbb{P}(i)$.

I_{it} is the input inventory level of item i at the end of period t (amount in the input buffer at T_t).

X_{it} is the amount of item i procured during $[T_{t-1}, T_t]$; that is, amount of procurement during period t in stage i.

O_{it} is output inventory level of item i at the end of period t (amount in the output buffer at T_t).

L_{it} is the sum of external (independent) demand for item i at period t and the amount of item i made available for its successor items at time T_t, which is the size of the transfer batch at the end of period t in stage i: $L_{it} = D_{it} + \sum_{j \in \mathbb{S}(i)} \alpha_{ij} Y_{j,t+1}$.

Whether or not a transfer is to take place in stage i at the end of period t is indicated by a set of logical variables,

$$Z_{it} = 1, \text{ if there is a transfer batch from stage } i \text{ to all stages } j \in \mathbb{S}(i) \text{ at the end of period } t,$$
$$= 0, \text{ otherwise.}$$

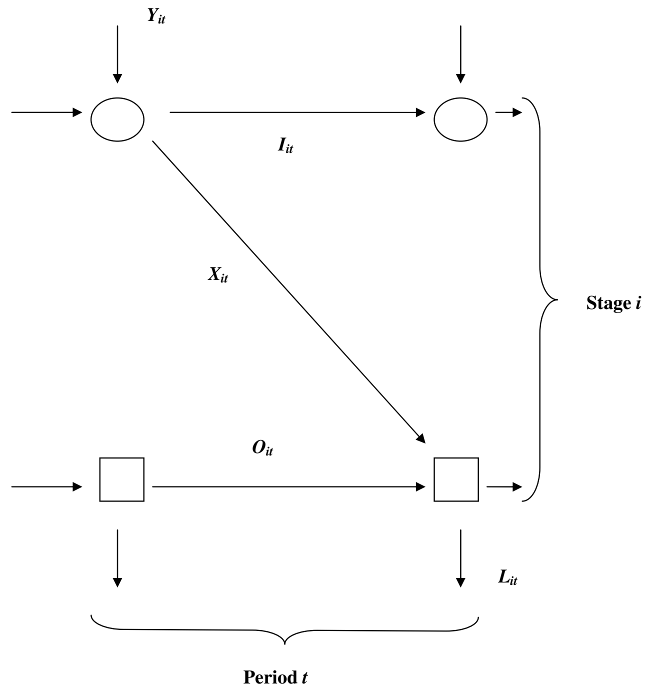

Fig. 1. Material Flow through Buffers.

There are two sets of *inventory balance equations*, for input and output buffers, respectively:

$$
\begin{aligned}
I_{i,t-1} + Y_{i,t} &= I_{it} + X_{it},\ \forall\, i, t,\\
O_{i,t} + X_{it} &= O_{i,t+1} + L_{it},\ \forall\, i, t,\\
\text{where } L_{it} &= D_{it} + \sum_{j \in \mathbb{S}(i)} \alpha_{ij} Y_{j,t+1},\ \forall\, i, t,
\end{aligned}
$$

Recall that demand for end items occur only at time points $t \in \mathbb{T}_e$. D_{it}, is the given external (independent) demand for item $i \in \mathbb{M}_e, t \in \mathbb{T}_e,$ and $D_{it} = 0$ for all $i \notin \mathbb{M}_e, t \notin \mathbb{T}_e$.

There are resource constraints restricting the amount that can be procured during a period,

$$\sum_i \rho_{ki}\ X_{it} \leq r_k \left[T_t - T_{t-1}\right],\ \forall\, k \in \mathbb{K},\ \ t \in \mathbb{T}.$$

where,

$\mathbb{K}$ is the index set of resources used by items, and

ρ_{ki} is the amount of resource $k \in \mathbb{K}$ required for procuring a unit of item i,
r_k is the total availability of resource k per unit time.

Big$-M$ constraints, $L_{it} \leq MZ_{it}$, $\forall\, i \in \mathbb{M}, t \in \mathbb{T}$, can enforce the transfers can take place only if they are indicated to do so.

Transfer times among stages are assumed to be negligible. If these times are significant, as it would be in goods movement in a supply chain, then the inter-stage transfers can be formulated as a separate "stage" with appropriate resource requirements.

The basic framework of logic-based modeling (Hooker 2000) makes event-time models computationally viable utilizing the specific features of constraint programming. In addition to its capability to handle big$-M$ constraints quite efficiently, constraint programming makes it possible to handle the exogenous events whose times are fixed and the endogenous events whose times of occurrence are decision variables in the same model.

ILOG's constraint programming software, OPL STUDIO (ILOG 2005) allows for *variable subscripts*, making it possible to handle the following *conditional constraint*. Let ν_ℓ denote the period at the end of which an exogenous event ℓ occurs. Recalling that $\tau_\ell \in \mathbb{T}_e = \{\tau_1, \tau_2, \ldots, \tau_m\}$ are the times at which the exogenous events occur, the *conditional constraint*

$$(\nu_\ell = t) \rightarrow (T_t = \tau_\ell), \quad t \in \mathbb{T} = \{1, \ldots, n\},$$

for each $\ell = 1, \ldots, m$, ensures that the exogenous event ℓ occurs at T_t. The following example illustrated how this constraint is incorporated in the program.

Example

Consider a planning horizon of 10 time units ("`horizon`"). Assume that there will be 5 time periods in the problem ("`nbPeriods`"). Suppose that there are following two exogenous events ("`nbEvents`"): external demands occurring at times 3 and 8. Then the following lines in the program make sure that two of the periods end exactly at time points 3 and 8. Timing of remaining 3 periods are determined based on other constraints and optimality criterion.

In terms of the notation and terminology used in model formulation, this example assumes that there will be five periods (`nbPeriods`, or $n = 5$) in the model. The end points of these periods correspond to *events* times, $T_1, T_2, \ldots, T_5$,. They can assume any (integer) values within the planning horizon of 10 time units ("`horizon`"). Two of these events are exogenous (`nbPeriods`, $m = 2$) whose times are given as are fixed at $\tau_1 = 3 and \tau_2 = 8$. The other three are endogenous events whose times are to be determined.

For example, a feasible ordering would be $\{T_1, T_2 = 3, T_3, T_4 = 8, T_5\}$; another would be $\{T_1 = 3, T_2, T_3, T_4, T_5 = 8\}$. These orderings and possible assignments are shown as `Solution[22]` and `Solution[56]` in the following illustration of the OPL code and its partial output.

```
int nbPeriods = 5;
int nbEvents = 2;
int horizon = 10;

range period 1..nbPeriods;
range exoEvent 1..nbEvents;

int givenTime[exoEvent] = [3,8];

var int time[period] in 1..horizon;
var int index[exoEvent] in 1..nbPeriods;

solve{

  forall(t in 2..nbPeriods)
     time[t-1] < time[t];

  forall(i in exoEvent)
  time[index[i]] = givenTime[i];
}
;
```

```
     Solution [22]              Solution [56]

     time[1] = 1                time[1] = 1
     time[2] = 3                time[2] = 2
     time[3] = 4                time[3] = 3
     time[4] = 8                time[4] = 7
     time[5] = 9                time[5] = 8

     index[1] = 2               index[1] = 3
     index[2] = 4               index[2] = 5
```

This section presented a generalized model for lot streaming problems. Furthermore, it is shown that constraint programming framework provides a computationally viable approach to these problems. The next section shows that the two fundamental results for single-job lot streaming problems can be systematically obtained as special cases of this generalized formulation. The purpose of that section is twofold. First, it is to validate, at least for a special class of lot streaming problems, that event-time formulation accurately models all of the features of the problem. Secondly, it maintains that, starting with a concise but comprehensive formulation, solutions for special cases of the problem can be obtained systematically. To that end, a number of additional special case formulations of single-job lot streaming problem is presented.

4 Single Job Lot Streaming Problems

This section will show that two fundamental results for single-job lot streaming problems can be systematically obtained as special cases of event-time modeling approach presented in the previous section. The special cases of single-job lot streaming problem are analyzed by Trietsch & Baker (1993) and Glass et al. (1994). Consider the following operational situation. At each *stage* of production an *item* is produced. For example, the raw material enters stage 1 in which item 1 is produced. Item 1 goes into stage 2 for the production of item 2, and so on, until the finished good, item m, is produced in stage m. It has a very simple, “series”, product structure in which a single unit of item i is required for a unit production of item $i+1$.

In each stage, there are s_i time points at which transfers in between stages can occur. Since, in general, these time points may not necessarily occur at the same instances in each stage, there are, at most, $n = \sum_{i=1}^{m} s_i$ time points at which the transfers can occur. Denote these time points by $T_t,\ t = 1, \ldots, n$ and let T_0 be the time at which the job starts on stage 1. The *periods* are defined by the time intervals $[T_{t-1}, T_t],\ t = 1, \ldots, n$. Note that $T_t \leq T_{t+1},\ t = 1, \ldots, n-1$, are decision variables in the model. Define the following variables:

X_{it} : Amount of item i produced during $[T_{t-1}, T_t]$, i.e. amount of production in period t in stage i.

Y_{it} : Amount of item i made available for the production of item i at time T_{t-1}, i.e. the size of the transfer batch at the end of period $t-1$ in stage $i-1$.

In stage i, the unprocessed material (item $i-1$) that has been transferred from stage $i-1$ is stored in an input buffer and the processed material (item i), if not transferred to stage $i+1$, is stored in an output buffer. In order to achieve the inventory balance, define two sets of inventory variables:

I_{it} : Input inventory level (amount in the input buffer at T_{t-1}) in stage i at the beginning of period t.

O_{it} : Output inventory level (amount in the output buffer at T_{t-1}) in stage i at the beginning of period t.

The earliest time period in stage i in which production can take place is period $t = i$ and the latest time period in which production can take place is $t = n - m + i$. For stage i to begin processing, a sublot must have been processed at $(i-1)$ upstream stages. That is, at least, $(i-1)$ transfers are required until the first sublot to reach stage i. Since each transfer takes place at the end of a stage, earliest time period at which processing can start in stage

i is period i. Similarly, at least $(m-i)$ transfers, and, thus, time periods, are required for the last sublot in stage i to complete its processing in the downstream stages. Define the periods $t = i, \ldots, n-m+i$ as *active periods* for stage i. Thus, there are $n-m+1$ active periods for each stage.

Finally, define a set of binary variables, one for each active period in every stage, in order to indicate whether or not a transfer takes place at stage i at the end of period t:

$Z_{it} = 1$, if there is a transfer batch from stage i to stage $i+1$ at the end of period t,
$= 0$, otherwise.

The transfers can take place only if they are indicated to do so,

$$Y_{it} \leq M Z_{it}, \; i = 1, \ldots, m, \; t = i, \ldots, n-m+i. \tag{1}$$

where M is a very large number.

The total number of transfer batches in stage i is bounded by s_i,

$$\sum_{t=i}^{n-m+i} Z_{it} \leq s_i, \; i = 1, \ldots, m. \tag{2}$$

There are two sets of *inventory balance* equations, for input and output buffers, respectively,

$$I_{i,t-1} + Y_{i,t} = I_{it} + X_{it}, \; i = 1, \ldots, m, \; t = i, \ldots, n-m+i, \tag{3}$$

$$O_{i,t} + X_{it} = O_{i,t+1} + L_{i+1,t+1}, \; i = 1, \ldots, m, \; t = i, \ldots, n-m+i, \tag{4}$$

There are *capacity* constraints restricting the amount that can be produced in a period,

$$\rho_i \, X_{it} \leq T_t - T_{t-1}, \; i = 1, \ldots, m, \; t = i, \ldots, n-m+i \tag{5}$$

with $T_0 = 0$.

Finally, the nonnegativity, integrality, and initial restrictions complete the model:

$$T_t \geq 0, \; t = 1, \ldots, n \tag{6}$$

$$X_{it}, \; Y_{it}, \; I_{it}, \; O_{it} \geq 0, \; i = 1, \ldots, m, \; t = i, \ldots, n-m+i \tag{7}$$

$$Y_{it} = 0 \text{ or } 1, \; i = 1, \ldots, m, \; t = i, \ldots, n-m+i. \tag{8}$$

and $T_0 = I_{i,i-1} = I_{i,h-m+i} = O_{i,i-1} = O_{i,h-m+i} = 0$, $\sum_{t=1}^{h-m+1} Y_{0,t} = \sum_{t=m}^{n} Y_{mt} = U$.

If the entire job, U units, are available at time T_0, then $Y_{0,0} = U$ and $\sum_{t=2}^{n-m+1} Y_{0,t-1} = 0$. For each stage and for each active period, production (X_{it}), transfer (Y_{it}), input buffer (I_{it}), and output buffer (O_{it}) variables are defined. Since the ending inventories are zero, two less inventory variables are needed for each stage. Thus, together with the n variables (T_t) defining the end points of the periods, there are altogether $4m(n-m+1)-2m+n = 2m(2(n-m)-1)+n$ continuous variables and $m(n-m+1)$ binary variables ($Z_{i,t}$), where $n = \sum_{i=1}^{m} s_i$. There are two inventory balance and two capacity constraints for every active period and an additional m constraints restricting the maximum number of transfers in each stage, resulting in $4(n-m+1)+m = 4n-3m+4$ constraints.

4.1 Extensions of the Single Job Model

The constraints described in the above general formulation describe the physics of the problem of lot streaming a single job in a flow shop. It should be noted that the setup times do not complicate the formulation, as long as the setup is *not* "attached" to the job and can be done while the job is not physically at the machine. Since there is a single job, it can be assumed that the setups are completed prior to the start of first sublot in each machine.

In the case that there is a constraint on the total number of transfers, s, that can take place, rather than a separate restriction for each stage, it is sufficient to replace the set of Constraints (2) by

$$\sum_{i=1}^{m} \sum_{t=i}^{n-m+i} Z_{it} \leq s. \tag{9}$$

By defining a specific objective function, restricting values of some variables, and appending additional configurational constraints, we can obtain the specific formulations of different problem types. Some examples are given below.

Minimization of Makespan

Since there is a single job, this criterion is equivalent to minimizing the completion time of this job. The completion time of a job is determined by the time at which its last unit completes processing in the last machine. Thus, there is no need to have more than one sublot on the last machine, that is,

$$s_m = 1, \tag{10}$$

and thus $Z_{mt} = 0,\ t = m, \ldots, n-1$, and $Z_{mn} = 1$. The problem is

$$\min T_n,$$

subject to constraints (3) through (8) and (10).

Special cases of this problem are discussed in Trietsch & Baker (1993) and Glass et al. (1994).

Minimization of Mean Flow Time

In the traditional scheduling theory, the spirit of this criterion is to maximize "the rate at which jobs are completed". In the context of lot streaming with finite number of sublots, a reasonable interpretation would be to "weigh" each sublot with its size, i.e. the number of units in the sublot. These are not arbitrary weights that can be imposed in the problem data but intrinsic property of the problem instance. Additional weights may be imposed as in the case of weighted mean flow time, only in the presence of multiple jobs. With this interpretation, the minimization of mean flow time in a single job lot streaming problem is minimization of total sublot completion time where each sublot is weighed by its size. Hence the problem becomes,

$$\min \sum_{t=m}^{h} Y_{mt} T_t,$$

subject to constraints (3) through (8).

Note that the objective function is quadratic. When the sublots are *consistent* (i.e. $L_{ij} = L_j, \ \forall \ i, j$; see Appendix.) Kropp & Smunt (1990) give a quadratic programming formulation. In Sen et al. (1998) specific results for the special cases of this problem are presented.

Minimization of Mean Unit Completion Time

This objective aims to maximize the rate at which the units complete their processing in the last stage. Practically, this means that there are as many transfer batches at the end of last stage as there are units in the lot. Since stage $(m-1)$ sends s_{m-1} sublots to stage m, we can think of stage m to process, uninterrupted, s_{m-1} sublots. Each unit, as it completes processing in the last stage, leaves the system. Each unit's contribution to the objective function value is its completion time on stage m. Consider the sublot that leaves stage m at time T_t of size Y_{mt}. This sublot must have started its processing on stage m at time $(T_t - \rho_m Y_{mt})$, and the first unit in this sublot must have been completed at time $(T_t - p_m Y_{mt} + \rho_m)$. In order to find the sum of completion times of all the units in this sublot, we have to compute the sum of Y_{mt} units in an *arithmetic progression*, in which each number differs from the previous number by a *common difference* of ρ_m and which has $T_t - \rho_m(Y_{mt} - 1)$ as its first term. This sum is equal to $Y_{mt}[T_t - \frac{1}{2}\rho_m(Y_{mt} - 1)]$. Since the earliest time at which a sublot can arrive to the last stage is T_m, the objective function becomes,

$$\min \sum_{t=m}^{n} Y_{mt}[T_t - (Y_{mt} - 1)\rho_m/2],$$

subject to Constraints (3) through (8).

See Sen et al. (1998) for specific results for special cases of this problem.

Minimization of Number of Transfer Batches Subject to a Deadline

Suppose the job *must* be completed by a deadline $\bar{d}$. If the objective is to achieve this deadline using minimum number of transfer batches, then set

$$T_n = \bar{d} \tag{11}$$

and optimize the objective function,

$$\min \sum_{i=1}^{m} \sum_{t=i}^{n-m+i} Z_{it},$$

subject to constraints (3) through (8), and (11).

Maximization of Throughput During a Planning Horizon

Suppose we have a planning horizon of $[0, \bar{d}]$. Given the maximum number of transfers between the stages, the problem is to process as many units as possible. Let U be the variable amount to be processed, then the objective function becomes,

$$\max U,$$

subject to Constraints (3) through (8), and (11).

The next two subsections show that it is possible to obtain the well known results available in the literature as a direct consequence of the special cases of above general formulation. Notation used for the classification scheme for lot streaming problems is explained in the Appendix.

4.2 Single Job $F2|s|C_{max}$ Problem

This problem is discussed in Trietsch (1987), Baker (1988) and Potts & Baker (1989), and shown that the optimal sublot sizes are given by a geometric pattern. Using the notation introduced above, they have proved that the optimal sublot sizes are,

$$X^*_{i,t+i-1} = [(\pi^{t+i-2} - \pi^{t+i-1})/(1 - \pi^s)],\ i = 1, 2;\ t = 1, \ldots, s,$$

where $\pi \equiv \rho_2/\rho_1$. Here, it will be shown that the same results can be obtained by restrictions on the general model.

The fractional values for sublot sizes are allowed, then, without loss of generality, for ease of notation, we can set $U \equiv 1$. Since the objective is makespan minimization, $s_2 = 1$ and let $s_1 \equiv s$. Then $n = \sum_{i=1}^{2} s_i = s + 1$, and the *active periods* for stages 1 and 2 are, respectively, $\{1, 2, \ldots, s\}$ and $\{2, 3, \ldots, s + 1\}$. Hence, we can set $Z_{1t} = 1,\ t = 1, \ldots, s$. The Constraints (1) and (2), for $i = 1$, are redundant and therefore can be omitted. The resulting problem is a linear program, and therefore *the single job* $F2|s, idlg|C_{max}$ is polynomially solvable.

The formulation can be further simplified. Since $\{Y_{1t} \geq 0,\ t = 1, \ldots, s\}$ are no longer restricted, and a *transfer* can take place at the end of every active period, there is no need for the output buffer in Stage 1. The amount produced in Stage 1 during any (active) period can be transferred to (the input buffer of) Stage 2 at the end of that period. Setting $O_{1t} = 0,\ t = 1, \ldots, s$, in Constraints (4) implies $X_{1t} = Y_{1t},\ t = 1, \ldots, s$, thus eliminating the Y_{1t} variables from the formulation.

The resulting inventory balance equations can be equivalently represented as,

$$\sum_{k=1}^{t} X_{1k} \geq \sum_{k=1}^{t} X_{2,k+1},\ t = 1, \ldots, s \tag{12}$$

$$\sum_{t=2}^{s+1} X_{2t} = 1, \tag{13}$$

Moreover, since the objective function is minimizing T_{s+1}, the capacity constraints (5) simply require

$$T_{s+1} = \rho_1 X_{11} + \sum_{t=2}^{s} [\ \max_{1 \leq i \leq 2} \{\rho_i X_{it}\}\ ] + \rho_2 X_{2,s+1}$$

Thus the LP formulation of *the single job* $F2|s, idlg|C_{max}$ problem becomes,

$$\min\ T_{s+1} = \rho_1 X_{11} + \sum_{t=2}^{s} [\max_{1 \leq i \leq 2} \{\rho_i X_{it}\}] + \rho_2 X_{2,s+1} \tag{14}$$

subject to:

$$\sum_{k=1}^{t} X_{1k} \geq \sum_{k=1}^{t} X_{2,k+1},\ t = 1, \ldots, s \tag{15}$$

$$\sum_{t=2}^{s+1} X_{2t} = 1, \tag{16}$$

$X_{it} \geq 0,\ i = 1, 2;\ t = i, \ldots, s + i - 1.$

We need the following result.

Result: There exists an optimal solution $\{X^*_{it},\ i = 1, 2;\ t = i, \ldots, s+i-1\}$ to the *single job* $F2|s, idlg|C_{max}$ problem such that

$$X^*_{1t} = X^*_{2,t+1},\ t = 1, \ldots, s,$$

and,

$$\rho_1 X^*_{1t} = \rho_2 X^*_{2t},\ t = 2, \ldots, s.$$

Proof: Clearly, this holds for $\rho_1 = \rho_2$. Therefore, assume $\rho_1 \neq \rho_2$. For convenience in notation let,

$$\begin{aligned} z &\equiv T_{s+1}, \\ x_t &\equiv X_{1t},\ t = 1, \ldots, s \\ x_{s+1} &\equiv 0 \\ y_1 &\equiv 0 \\ y_t &\equiv X_{2t},\ t = 2, \ldots, s+1 \\ a &\equiv \rho_1 \\ b &\equiv \rho_2 \end{aligned}$$

Rewriting the problem in this notation,

$$\min\ z = \sum_{t=1}^{s+1} [\max\{ax_t, by_t\}] \tag{17}$$

subject to:

$$\sum_{k=1}^{t} x_k \geq \sum_{k=1}^{t} y_{k+1},\ t = 1, \ldots, s \tag{18}$$

$$\sum_{t=1}^{s+1} y_t = 1, \tag{19}$$

with $x_t, y_t \geq 0,\ t = 1, \ldots, s+1$.

An equivalent representation of the problem is,

$$\min z = ax_1 + \sum_{t=2}^{s} w_t + by_{s+1}$$

subject to:

$$w_t - ax_t \geq 0,\ t = 2, \ldots, s, \tag{20}$$

$$w_t - by_t \geq 0,\ t = 2, \ldots, s, \tag{21}$$

and Constraints (18), (19) and $x_t \geq 0,\ t = 1, \ldots, s;\ y_t \geq 0,\ t = 2, \ldots, s+1;\ w_t \geq 0,\ t = 2, \ldots, s$.

Let the dual variables $\gamma_t, \delta, \alpha_t, \beta_t$, respectively, correspond to the primal constraints (18) through (21). The the dual problem is:

$$\max\ \delta$$

subject to:

$$\sum_{k=t}^{s} \gamma_k - a\alpha_t \geq 0,\ t = 1, \ldots, s, \tag{22}$$

$$-\sum_{k=t}^{s} \gamma_k + \delta - b\beta_{t+1} \geq 0,\ t = 2, \ldots, s, \tag{23}$$

$$\alpha_t + \beta_t \leq 1,\ t = 2, \ldots, s, \tag{24}$$

and $\alpha_t, \beta_t \geq 0,\ t = 2, \ldots, s;\ \gamma_t \geq 0,\ t = 1, \ldots, s$; and setting $\alpha_1 = 1, \beta_{s+1} = 1$.

The resulting complementary slackness conditions are:

$$x_t \left[\sum_{k=t}^{s} \gamma_k - a\alpha_t \right] = 0,\ t = 1, \ldots, s, \tag{25}$$

$$y_t \left[-\sum_{k=t-1}^{s} \gamma_k + \delta - b\beta_t \right] = 0,\ t = 2, \ldots, s+1, \tag{26}$$

$$w_t[1 - \alpha_t - \beta_t] = 0,\ t = 2, \ldots, s, \tag{27}$$

$$\alpha_t[w_t - ax_t] = 0,\ t = 2, \ldots, s, \tag{28}$$

$$\beta_t[w_t - by_t] = 0,\ t = 2, \ldots, s, \tag{29}$$

$$\gamma_t \left[\sum_{k=1}^{t} x_t - \sum_{k=1}^{t} y_{t+1} \right] = 0,\ t = 1, \ldots, s. \tag{30}$$

To show that there exists an optimal solution to the primal such that

$$x_t > 0,\ t = 1, \ldots, s, \tag{31}$$

$$y_t > 0,\ t = 2, \ldots, s+1, \tag{32}$$

$$x_t = y_{t+1},\ t = 1, \ldots, s, \tag{33}$$

$$ax_t = by_t,\ t = 2, \ldots, s, \tag{34}$$

it is sufficient to show the existence of a dual feasible solution that satisfies the complementary slackness conditions.

Because of (31) and (32), the Conditions (25) and (26), respectively, require,

$$\sum_{k=t}^{s} \gamma_k = a\alpha_t,\ t = 1, \ldots, s, \tag{35}$$

$$\sum_{k=t}^{s} \gamma_t = \delta - b\beta_{t+1},\ t = 1, \ldots, s. \tag{36}$$

Finally, because of (31), (32), and (34), Condition (3) requires

$$\alpha_t + \beta_t = 1. \tag{37}$$

Recalling that $\alpha_t, \beta_{s+1} = 1$ and using Equations (35) and (36),

$$a\alpha_t = \delta - b\beta_{t+1},\ t = 1, \ldots, s.$$

Then, using Equations (37),

$$\begin{aligned} -a\alpha_t + b\beta_{t+1} &= b - \delta,\ t = 1, \ldots, s-1, \\ -a\alpha_s &= b - \delta. \end{aligned}$$

Let $\pi = b/a$,

$$\begin{aligned} \alpha_t &= \delta/a - \pi + \pi\alpha_{t+1},\ t = 1, \ldots, s-1, \\ \alpha_s &= \delta/a - \pi. \end{aligned}$$

Defining $\varphi = \delta/a - \pi$,

$$\begin{aligned} \alpha_s &= \varphi, \\ \alpha_t &= \varphi + \pi\alpha_{t+1},\ t = s-1, \ldots, 1. \end{aligned}$$

Solving the above equations of φ,

$$\alpha_t = \varphi \left[\sum_{j=0}^{s-t} \pi^j \right],\ t = 1, \ldots, s.$$

Recalling that $\alpha_1 = 1$,

$$1 = \varphi \left[\sum_{j=0}^{s-1} \pi^j \right] = \varphi \left[\frac{1-\pi^s}{1-\pi} \right].$$

Then,

$$\varphi = \frac{1-\pi}{1-\pi^s} > 0.$$

Thus,

$$0 < \alpha_t = \frac{1-\pi^{s-t+1}}{1-\pi^s} \leq 1,\ t = 1, \ldots, s,$$

and

$$0 < \beta_t = \frac{\pi^{s-t+1} - \pi^s}{1-\pi^s} \leq 1\ t = 2, \ldots, s+1,$$

and finally, using Equations (35), $\gamma_t \geq 0,\ t = 1, \ldots, s$.

The above result also shows that there will be no intermittent idling in an optimal solution to *single job* $F2|s, idlg|C_{max}$ problem, thus it simply can be stated as *single job* $F2|s|C_{max}$. Furthermore, there exists an optimal solution with consistent sublots.

Now, to show that

$$X^*_{i,t+i-1} = [(\pi^{t+i-2} - \pi^{t+i-1})/(1-\pi^s)],\ i = 1,2;\ t = 1,\ldots,s,$$

where $\pi \equiv \rho_2/\rho_1$, solves the *single job* $F2|s|C_{max}$ optimally, note that $X_{1t} = \pi X_{2t} = \pi X_{1,t-1} = \pi^{t-1} X_{11},\ t = 1,\ldots,s$. But since $\sum_{t=1}^{s} X_{1t} = 1$,

$$1 = X_{11} \sum_{t=0}^{s-1} \pi^t = [(1-\pi^s)/(1-\pi)]X_{11}$$

That is, $X_{11} = [(1-\pi)/(1-\pi^s)]$. Thus,

$$\begin{aligned} X_{1t} &= \pi^{t-1}[(1-\pi)/(1-\pi^s)],\ t = 1,\ldots,s, \\ &= [(\pi^{t-1} - \pi^t)/(1-\pi^s)], t = 1,\ldots,s. \end{aligned}$$

And since, $X_{2,t+1} = X_{1t}$,

$$X_{2t} = [(\pi^{t-2} - \pi^{t-1})/(1-\pi^s)],\ t = 2,\ldots,s+1.$$

4.3 Single Job $F|s_i|C_{max}$

The single job $F|s_i|C_{max}$ problem, where idling is not permitted, can be solved in polynomial time by applying the result for two stage problem to adjacent machine pairs Trietsch & Baker (1993). In the following, the same result is derived as a special case of the general formulation.

Suppose no intermittent idling is allowed in between sublots in any stage. This means that there is continuous production in a stage once the production starts. Let the period at which the production starts in a stage $i+1$ is k_{i+1} where,

$$k_{i+1} = \min_{i \le t \le h-m+i} \{t | Y_{it} = 1\} + 1,$$

and the production is completed at the end of period l_{i+1}, where,

$$l_{i+1} = \max_{i \le t \le h-m+i} \{t | Y_{it} = 1\} + 1,$$

Thus the continuous production takes place in stage $(i+1)$ throughout the periods $\{k_{i+1},\ldots,l_{i+1}\}$. The makespan is minimized only if T_{l_i} is as small as possible in every stage $i = 1,\ldots,m$.

There will be exactly s_i transfer batches in between stages i and $i+1$. Recalling the definitions above, the first transfer batch occurs at the end of period $(k_{i+1} - 1)$, so that the production can start at period k_{i+1} in stage $(i+1)$.

For notational convenience, re-index the periods at which the transfers take place from stage i to $i+1$, as $i(\tau),\ \tau = 1,\ldots,s_i$, where $i(1) = k_{i+1} - 1$. This results in fixing the values of the indicator variables,

$$\begin{aligned} Y_{it} &= 1,\ t = i(\tau),\ \tau = 1, \ldots, s_i, \\ &= 0, \text{otherwise.} \end{aligned}$$

Furthermore, the sizes of the transfer batches are given by

$$\rho_{i+1} L_{i,i(\tau)} = \rho_i L_{i,i(\tau+1)},\ \tau = 1, \ldots, s_i - 1.$$

That is,

$$L_{i,i(\tau)} = \pi_i^{\tau-1} L_{i,i(1)},\ \tau = 1, \ldots, s_i - 1.$$

where $\pi_i \equiv \rho_{i+1}/\rho_i$. But, since $\sum_{\tau=1}^{s_i} L_{i,i(\tau)} = 1$,

$$\begin{aligned} 1 &= L_{i,i(1)} \sum_{\tau=1}^{s_i} \pi_i^{\tau-1} \\ &= L_{i,i(1)} \sum_{\tau=0}^{s_i-1} \pi_i^{\tau} \\ &= [(1 - \pi_i^{s_i})/(1 - \pi_i)] L_{i,i(1)} \end{aligned}$$

or

$$\begin{aligned} L_{i,i(1)} &= [(1 - \pi_i)/(1 - \pi_i^{s_i})] \\ L_{i,i(\tau)} &= \pi_i^{\tau-1}[(1 - \pi_i)/(1 - \pi_i^{s_i})] \\ &= [(\pi_i^{\tau-1} - \pi_i^{\tau})/(1 - \pi_i^{s_i})],\ \tau = 1, \ldots, s_i, \end{aligned}$$

which is the desired result.

5 Conclusions

The problems that we have dealt with in this study are scheduling problems in supply chains. A modeling paradigm is proposed for scheduling problems in supply chains. The primary contention has been that the constituents of a supply chain need to cooperate, rather than compete, in order to achieve overall, as well as individual, maximum benefits. In order to analyze this, it is essential to have a concise but comprehensive formulation. The formulation approach proposed, event-time models, in addition to being computationally viable, account for exogenous events, such as demand occurrences and other deadlines, as well as the endogenous events that are decision variables in the model.

After providing the motivation for this study in Section 1, Section 2 argues that lot streaming is a suitable paradigm for supply chain scheduling. Section 3 introduces event-time modeling as a general modeling approach for lot streaming problems. Furthermore, logic-based modeling framework of constraint programming makes it possible to handle exogenous and endogenous models in the same model. This is demonstrated by an example using OPL (Optimization Programming Language), which can handle variable subscripts in conditional constraints. In Section 4, it is shown that the event-time modeling approach can account for all features of a single job lot streaming problem, that are available for this problem in the literature. This section also presents a number of modeling extensions for single job problem.

A The Notation

There is no widely accepted classification for lot streaming problems. Being basically scheduling problems, lot streaming problems can be classified by using, with possible extensions, the established scheme of Lawler et al. (1993). In this scheme a problem type is specified in terms of a three-field classification $\alpha|\beta|\gamma$, where α specifies the machine environment, β indicates a number of job characteristics, and γ refers to the optimality criterion. Using the definitions given in Potts & Baker (1989) and Trietsch & Baker (1993), the following addendum to the list of job characteristics can be proposed for lot streaming problems.

In Lawler et al. (1993), the second field is defined as $\beta \subset \{\beta_1, \ldots, \beta_4\}$, where the elements $\beta_1, \beta_2, \beta_3$, and β_4 are concerned, respectively, with preemption, precedence, release dates, and unit processing requirements.

Further streaming characteristics can be incorporated as $\{\beta_5, \ldots, \beta_8\}$ where $\beta_5 \in \{s_i, s, s = \kappa, \circ\}$: (The symbol $\circ$ denotes an empty symbol which is omitted in the statement of problem types.)

$\beta_5 = s_i$: *Streaming* is allowed, i.e., any job may be split into sublots so that its operations can be overlapped and its progress accelerated, furthermore the maximum number of transfer batches ("sublots"), s_i, allowed from machine i to machine $i+1$ is variable and can be different for all $i = 1, \ldots, m$.

$\beta_5 = s$: *Streaming* is allowed, but the maximum number of transfer batches ("sublots") allowed from machine i to machine $i+1$ is variable but same for all $i = 1, \ldots, m$.

$\beta_5 = s = \kappa$ where κ is any positive integer: The number of sublots is a constant and specified as part of the problem.

$\beta_5 = \circ$: No streaming is allowed, i.e. $\kappa = 1$.

$\beta_6 \in \{idlg, \circ\}$:

$\beta_6 = idlg$: Intermittent idling of machines are allowed in between processing of the sublots.

$\beta_6 = \circ$: No idling is allowed, i.e., each machine, once started, must process the entire job continuously, without idling.

$\beta_7 \in \{dscr, \circ\}$:

$\beta_7 = dscr$: Each sublot ("transfer batch") must contain discrete number of units.

$\beta_7 = \circ$: No integer requirement on the size of the sublots.

$\beta_8 \in \{equl, cnst, \circ\}$:

$\beta_8 = equl$: *Equal sublots*, each sublot ("transfer batch") size must be equal, that is, work is allocated equally among all sublots, on all machines.

$\beta_8 = cnst$: *Consistent sublots*, sublot ("transfer batch") size must be same at each machine, that is, the allocation of work to sublots is the same on all machines.

$\beta_8 = \circ$: *Variable sublots*, no restriction, other than that they sum up to lotsize, on the size of the sublots, that is, transfer batches between machines i and $(i+1)$ may differ from the transfer batches between machines $(i+1)$ and $(i+2)$.

In the classical scheduling theory, *single job* problems does not exist in the sense that they are trivial. Other than very few exceptions, the number of jobs practically play no significant role in defining problem *types*. But in lot streaming, the number of jobs is not only important but most of the current research deals with *single job* problems. In order not to further complicate the above schema, an additional field should not be introduced but the statement as to whether the problem type is *single job* or *multiple jobs* can be mentioned along with the three-field classification.

The *dominance* relation among the problems discussed in Trietsch & Baker (1993) corresponds to *relaxation* of the dominating problem resulting in the dominated problem. If problem Q is a relaxed version of problem P, denote the relationship as $P \prec Q$. Thus the objective function value of P can never be better than that of Q. The "job characteristics" $\beta_6, \beta_7, \beta_8$ imply the following dominance relations:

$$\boxed{\text{equl}} \prec \boxed{\text{cnst}} \prec \boxed{\circ}$$

$$\boxed{\text{dscr}} \prec \boxed{\circ}$$

$$\boxed{\circ} \prec \boxed{\text{idlg}}$$

On the other hand, reducibility among the lot streaming problems, in the complexity sense, can be stated for job characteristic β_5. It can be easily verified that if $P \longrightarrow Q$ implies that the decision version of P reduces to the decision version of Q, then:

$$\boxed{s_i} \longrightarrow \boxed{\text{s}} \longrightarrow \boxed{\text{s} = \kappa} \longrightarrow \boxed{\circ}$$

References

Baker, K. R. (1988), Lot streaming to reduce cycle time in a flow shop, Working Paper 203, Amos Tuck School, Dartmouth College.

Cachon, G. P. (2003), Supply chain coordination with contracts, *in* S. C. Graves & A. G. de Kok, eds, 'Supply Chain Management: Design, Coordination and Operation', Vol. 11 of *Handbooks in Operations Research and Management Science*, North-Holland, chapter 6, pp. 229–340.

Chang, J. H. & Chiu, H. N. (2005), 'Comprehensive review of lot streaming', *Int. J. Prodn Res.* **43**, 1515–1536.

Conway, R. W., Maxwell, W. L. & Miller, L. W. (1967), *Theory of Scheduling*, Addison-Wesley.

de Kok, T. G. & Fransoo, J. C. (2003), Planning supply chain operations, *in* S. C. Graves & A. G. de Kok, eds, 'Supply Chain Management: Design, Coordination and Operation', Vol. 11 of *Handbooks in Operations Research and Management Science*, North-Holland, chapter 12, pp. 596–676.

Glass, C. A., Gupta, J. N. D. & Potts, C. N. (1994), 'Lot streaming in three-stage production processes', *European Journal of Operational Research* **75**, 378–394.

Graves, S. C. & de Kok, A. G. (2003), *Supply Chain Management: Design, Coordination and Operation*, North-Holland.

Hooker, J. (2000), *Logic-Based Methods for Optimization*, Wiley.

ILOG (2005), 'ILOG OPL Studio 3.7.1'. `http://www.ilog.fr/`, (5 June 2007).

Karmarkar, U. S. (1994), New directions in modelling production and distribution problems. Paper presented at the 15th International Symposium on Mathematical Programming, August 1994, Ann Arbor.

Kropp, D. H. & Smunt, T. L. (1990), 'Optimal and heuristic models for lot splitting in a flow shop', *Decision Sciences* **21**, 691–709.

Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G. & Shmoys, D. B. (1993), Sequencing and scheduling: Algorithms and complexity, *in* S. C. Graves, A. H. G. Rinnooy Kan & P. H. Zipkin, eds, 'Logistics of Production and Inventory', Vol. 4 of *Handbooks in Operations Research and Management Science*, North-Holland, chapter 9, pp. 445–522.

Maxwell, W. L. (1997). Comments made at the NSF Workshop on Manufacturing Logistics, Lehigh University.

Potts, C. N. & Baker, K. R. (1989), 'Flow shop scheduling with lot streaming', *Operations Research Letters* **8**, 297–303.

Pounds, W. F. (1961), 'The scheduling environment'. Paper presented at The Factory Scheduling Conference, Carnegie Institute of Technology, May 10–12, 1961.

Sen, A., Topaloğlu, E. & Benli, Ö. (1998), 'Optimal streaming of a single job in a two stage flow shop', *European J. Operational Research* **110**, 42–62.

Simchi-Levi, D., Wu, S. D. & Shen, Z.-J. (2004), *Handbook of Quantitative Supply Chain Analysis: Modeling in the E-Business Era*, Springer.

Trietsch, D. (1987), Optimal transfer lots for batch manufacturing: A basic case and extensions, Technical Report NPS-54-87-010, Naval Postgraduate School, Monterey, CA.

Trietsch, D. & Baker, K. R. (1993), 'Basic techniques for lot streaming', *Operations Research* **41**, 1065–1076.