Apprd in: the Proceedings of the Fourth International Conference on Expert Systems in Production and Operations Management, Hilton Head, South Carolina, May 1990

PROTECTION AGAINST UNCERTAINTY IN A DETERMINISTIC SCHEDULE

Whay-Yu Chiang and Mark S. Fox

Center for Integrated Manufacturing Systems
The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

It is a fact of manufacturing life that machines inevitably malfunction. This paper discusses how to attach a flexible bound by fuzzy algebra to the deterministic processing time to prevent a predetermined schedule from temporal deviation caused by such machine failures. A simulation of a simple job shop is conducted where we vary specific parameters of shop load, uncertainty variance for four different scheduling methods. The method using type-2 bounds outperforms the other three methods (original, mean, upperbound) with less total cost resulted from work-in-process and tardiness. The upperbound method occasionally results in minimum total cost; yet it lacks the flexibility as the type-2 method since it uses the bound to the highest extent and ends up in larger tardiness when there are tardy jobs.

To test the sensitivity of the result to different unit cost values, several cost structures are used to find the best condition for the method of type-2 bounds scheduling. The best condition, which is typical in manufacture industry, is when the unit cost of tardiness is higher than the unit cost of work-in-process. Furthermore, we explore the feasibility of using standard deviations for the bounds while both the time between machine failures and the duration of machine failure have normal distributions.

1.0 OVERVIEW

It is a fact of manufacturing life that machines inevitably malfunction. Of many effects of such an event, operational schedule disruptions are perhaps the most visible ramifications which may send tremor throughout the entire manufacturing process. A disrupted schedule not only leaves a shop in turmoil, but also incurs tremendous cost in not meeting promised due dates and cost invested in holding the inventory.

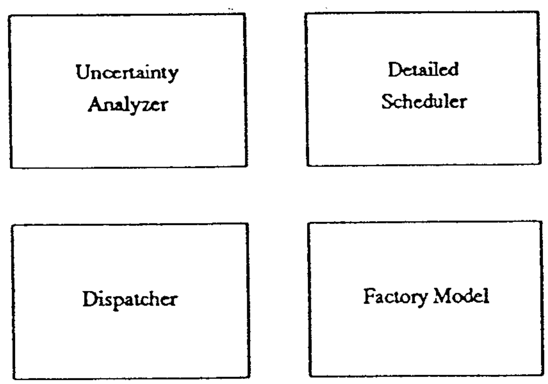
The goal of our uncertainty management research is to develop a model of uncertainty in a manufacturing setting so that we can alter the precision with which plans and schedules are formulated. As uncertainty increases, we would like to decrease precision in the predictive element (i.e., planning and scheduling) of the system, while at the same time providing the reactive elements (i.e., dispatching) with greater flexibility in their ability to react to change. Uncertainty may arise from a number of places, including external sources such as order arrival types and rates, material quality, and personnel availability, and internal sources of uncertainty such as operation quality, product yield, and material availability.

In general, a predictive schedule is sketched in advance according to orders. When machines fail, a dispatcher either gives up the present schedule and reschedules the rest of the orders all over or needs a reactive scheme to determine the scope of the uncertain effect and patches the schedule locally if possible. Three common metrics used to evaluate adjusted schedules are (1) the resulting work-in-process levels, (2) the tardiness induced by the schedule, and (3) rescheduling effort. The last criterion is dependent on a managerial control decision to abandon a schedule or not and that decision in turn depends on implementation issues in the shop such as the amount of time that rescheduling activity takes, how often and so on. It is not easy to draw a line between predictive and reactive scheduling. Moreover, it is difficult to define the rules for the reactive scheme. Worst thing that can happen is a schedule losing track of the events and under tremendous revision introducing more complexity rather than being a guidance. Under such circumstances there is no point maintaining a precise schedule. The relevant question becomes how to represent a schedule and to what precision.

The problem domain is a job shop where breakdowns occur frequently. The problem of interest in this paper deals with deviations in scheduled operation time due to the uncertainty of machine breakdown. We investigate which representation of protection allowance can yield enough temporal slack to tolerate machine uncertainty. The basic concept is to add slack to protect the job. With some known behavior of machine uncertainty, such as the time between failures and the duration of the failure, we can build up a predictive schedule with protection allowance and a dispatcher can react to uncertain events as long as they are within the predictive range of time bounds. Thus, the flexibility of a schedule is maintained to the precision of the time bounds. The same metrics are used as the criteria of evaluating the performance of protection against machine uncertainties. In our research, the scheduling method of using temporal slack generates a schedule that eliminates the need of rescheduling. As a consequence, we focus on the impacts that machine interrupts act on a predictive schedule with protection allowance and we compare different temporal slack performances based on the first two criteria: minimum work-in-process and tardiness.

This research is intended to be one of the modules in the distributed manufacturing project CORTES (Fox et. al. [5]). As shown in Figure 1-1, the system consists of the following modules: (1) Uncertainty Analyzer, (2) Detailed Scheduler, (3) Factory Model, and (4) Dispatcher that are distributed across many workstations and are connected by a communication network. Concentrated on machine failures, this paper as well as its later extension is to be the core of the Uncertainty Analyzer module.

Figure 1-1: Modules of Cortes Project



2.0 LITERATURE SURVEY

Anthony [3] classified the model of control into three broad categories. In brief, these three categories are described as managerial decisions at three hierarchical levels (Hax and Candea [8]): (1) **strategic planning**: top level decision of plans for acquisition of resources, (2) **tactical planning**: middle level of plans for utilization of resources, and (3) **operation control**: low level of detailed execution of schedules.

The effects of uncertainty to the manufacturing environment have been investigated at the middle level of the model of control such as well-known analytical approaches to the inventory problem of lot yielding and safety stock (Gerchak et. al. [6]; Grave [7]). Yet, the temporal deviation from machine failures in the job shop scheduling was not explicitly addressed. Our focus is to examine the effects of uncertainty at the lowest level of operations control and scheduling.

Sources of uncertainty can also be described in these three levels. At the top level, the uncertain market environment can change the product emphasis and labor supply which can in turn change the plan of capacity acquisition. At the middle level, changes in forecast and seasonal demand, yield, raw material quality and quantity can impact the production plan. And, finally at the lowest level, change of time duration for operations (transition, setup, processing), change of capacity from machine downtime or tool availability and so on can easily invalid a schedule.

The lowest level of operation control provides the day-to-day flexibility needed to meet customer requirements on a daily basis within the guidelines established by the more aggregate plans from the middle level. Taking orders directly from customers, or as generated by the inventory decision system, detailed schedules are drawn up in advance for a week, then a day, and finally to a shift. Decisions at the lowest level are dynamic in nature since at this level a shop faces with various sources of uncertainty at a shorter decision cycle. Unanticipated causes as well as scheduled events contributes to the shop uncertainty at this level of operations control. Raw materials are not always available. Aged tools wear out and affect the precision quality. In particular, machines break down from time to time. Uncertainties in machine performance often cause reality to deviate from schedules. How to tolerate these temporal deviation is the theme of this research.

One of the previous approaches to scheduling at this level is the Sched-star package by Morton et. al. [11] [12] that dynamically adjusts to the uncertain environment as it can redecide the urgency index for the orders at that time from the imputed (dual) prices of the machines. The dual prices of machines are passed down from an aggregate level for its relative value from these aggregated resources in the inventory and production level. Then, the lowest level can have a narrower focus and perform local optimization. The reactive scheme can be either locally as a recomputing minor price changes as in dispatcher mode, or leaving to rescheduling as in replanning model (Morton [11]). The prices for this model are computed using heuristics. The disadvantage is that it is hard to return good price estimates under all conditions for heuristics. An analytical approach with a hierarchical control has been studied for a flexible manufacturing environment (Akella et. al. [1, 2]). The objective of this algorithm is to calculate times at which to **dispatch parts** into a system in a manner which limits the disruptive effects of machine failures. With this approach, three levels of control are addressed. Its middle level is the major decision level of the scheduler and determines the production rate within capacity limits and achieves the objective function computed off-line by its higher level. Its lowest level decides the actual times at which parts are loaded into the system according to that production rate. However, it is for cumulative demand instead of being addressed to the job shop scheduling. The prior approaches have focused on dynamicly redeciding the urgency index or loading decisions at the time of uncertainty. There are industries with expensive machines so that building predictive schedules with protection allowance ahead is necessary for the day-to-day operation. In our approach, we explicitly take environmental uncertainty into account in order to produce schedules that tolerate temporal deviations and minimize work-in-process and job tardiness.

3.0 MODEL DESCRIPTION

The simulation is based on a model of a job shop environment with one part type, one machine and orders with requested due-dates and arrival-dates. The orders are fulfilled by a make-to-order policy. A scheduler decides its own production schedule for several weeks ahead.

In this simple model, there is one operation per order and as all orders are of the same type there is no setup required. An *order* is composed of several number of *units*. The number of units in an order is different from one order to another. The *unit* processing time for the operation is a constant, so the processing time for an *order* without any machine failure is the unit processing time multiplied by the number of units in the order. The processing time without machine interruptions for an order in turn is different from one order to another. The units in one order are completed together and there is no preemption among orders.

Machine failures occur from time to time during processing. The mean time between failure and mean duration of the failure are assumed known. Downtimes is assumed to be in the interrupt-resume regime, that is, once the downtime duration is completed (i.e. the machine is fixed), processing continues at the point of interruption and no rework is required. Consequently, machine failures cause a variation in the processing *time* and not in the scheduling *order sequence*. Feeding a fixed processing time to a deterministic scheduler without any allowance for uncertainty creates a fixed schedule which is vulnerable to the uncertain environment of disrupted machines. Therefore, we need a different scheme to take care of the case when the length of the processing time varies as a result of machine failures. The problem here is whenever a machine fails, the predetermined schedule is no longer valid. However, a predetermined schedule is needed for the shop to have control over its processing. With this uncertainty, a certain amount of time can be attached to the processing time in the deterministic schedule to accommodate the machine interruptions. Our research seeks how to allocate such time and to what amount to reduce the fragility of generated schedules. The key to reducing the fragility of a schedule in this manner is by explicitly representing the uncertainty in the processing time in a way that can be addressed by a scheduling algorithm.

4.0 UNCERTAINTY REPRESENTATION

Before dealing with machine uncertainty in the scheduling algorithm, we need an adequate representation to express the effect of uncertainty to processing time explicitly from the duration of machine failure and the time between failures. Let the original processing time be P, which is deterministic in the model. Let the time between machine failure be a random variable F and the duration of interrupt be a random variable D. If these two means are known as F and D, then a direct extension of the processing time to include the machine interruptions is given as $P+(P/F)\times D$, where P/F gives the number of interrupts that may occur during the processing and $(P/F)\times D$ gives the total length of the machine downtime. Thus the protection allowance is implicitly absorbed into the extended processing time.

Instead of being random variables of known distribution, the duration of the failure and the time between failure may be only known to be bounded approximately. The development of fuzzy number theory has made it possible to express these imprecise informations. Let the bounds are (D_{lb},D_{ub}) for D and (F_{lb},F_{ub}) for F with the means D and F, we can therefore determine the extended processing time using fuzzy algebra (Kaufmann and Gupta [9]).

- **Fuzzy Number of Type-1** Let A be a number with upper and lower bounds defining a confidence interval noted as $[a_1,a_2]$ where $a_1 \leq a_2$. Similarly, let B be a number associated with an interval $[b_1,b_2]$ (where $b_1 \leq b_2$) representing an interval of confidence for B.
- **Fuzzy addition**: Assuming two intervals of confidence in real numbers $R{\cdot}A=[a_1,a_2]$ and $B=[b_1,b_2]$. Hence if $x \in [a_1,a_2]$ and $y \in [b_1,b_2]$, then $x+y \in [a_1+b_1,a_2+b_2]$. Symbolically, we write it as $A(+)B=[a_1,a_2]+[b_1,b_2]=[a_1+b_1,a_2+b_2]$.
- **Fuzzy Subtraction**: $A(-)B=[a_1-b_2,a_2-b_1]$.
- **Fuzzy Multiplication**: $A(\times)B=[a_1\times b_1,a_2\times b_2]$.
- **Fuzzy Division**: $A(/)B=[a_1/b_2,a_2/b_1]$ when A and B are defined in R^+.

From the above fuzzy algebra, the bounds of the extended processing time can be given as follows: $P+P(/)F(\times)D=P+P(\times)D(/)F$ where $D(/)F=[D_{lb}/F_{ub},D_{ub}/F_{lb}]$, so that the *upper* and *lower* bounds of processing time would be: $(P+P\times D_{lb}/F_{ub},P+P\times D_{ub}/F_{lb})$.

The values of Fuzzy bounds may originate from subjective known processing characteristics described by a shop operator, or perhaps from known distributions described by shop statistics. The representation of such a uncertain duration is called a **type-1** fuzzy representation as the bounds are known in advance.

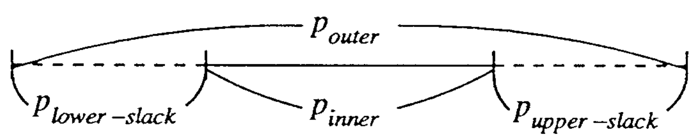
From the knowledge of the domain, the actual processing time is within the specified uncertain bounds. In an extension to representation, we can hypothesize the bounds as **type-2 bounds** similar to the fuzzy number of type-2 (Prade [14]) representation.

- **Fuzzy number of Type-2:** The lower and upper bounds of an interval of confidence, instead of being ordinary numbers, are fuzzy numbers that themselves have interval of confidence. That is $A=[[a_1,a_1'],[a_2,a_2']]$. When $a_1=a_1'$ and $a_2=a_2'$, the interval of confidence of type 2 becomes an interval of **type 1**. If $a_1=a_1'=a_2=a_2'$, we obtain an interval of confidence of **type 0,** an ordinary number.

Oftentimes, there is more uncertainty in a scheduling problem that can be handled by a type-1 bound representation. As a consequence, it is more accurate to represent additional uncertainty and less accurate not to; therefore, the upper and lower bounds must reflect uncertainty in their representation as type-2 bounds. The procedure of constructing the **type-2 bounds** in this paper is described through the following steps (see Figure 4-1):

1. Use the mean processing time and use it as the inner bound of the type-2 representation. Denote it as $p_{inner}=P+P\times D/F$.
2. Use the upper bound $(P+P\times D_{lb}/F_{ub})$, as the outer bound of the type-2 representation. Denote the length of the outer bound as p_{outer}.
3. Divide the slack between p_{outer} and p_{inner} into two segments and denote them as $p_{lower-slack}$ and $p_{upper-slack}$. That is, $p_{lower-slack}=p_{upper-slack}=(p_{outer}-p_{inner})/2=P\times(D_{ub}/F_{lb}-D/F)/2$ as shown in Figure 4-1. The $p_{lower-slack}$ and $p_{upper-slack}$, temporal slacks of equal amount, are designed to protect against uncertainty for possible delays of previous operations or possible delays of consequent operations.

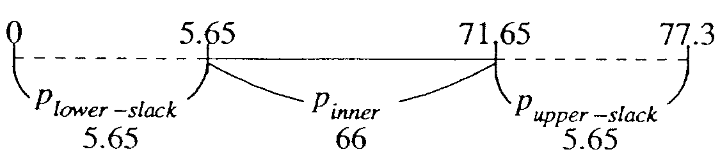
Figure 4-1: Illustration for the type-2 bound



Thus, **type-2 bounds** representation to include processing time with protection allowance for a job is constructed with values $A=[[a_1,a_1'],[a_2,a_2']]=[[0,p_{lower-slack}],[p_{lower-slack}+p_{inner},p_{outer}]]$ if the operation it represented starts at time 0. Or, $A=[[a_1,a_1'],[a_2,a_2']]=[[t,t+p_{lower-slack}],[t+p_{lower-slack}+p_{inner},t+p_{outer}]]$ if it starts at time t. The type-2 bounds of a job can then be used to reserve time block in a schedule for the job.

A numerical example can be used to illustrate the concepts. In the example with intervals and means available through estimations that $(D_{lb},D_{ub})=(12,14)$, $D=13$, $(F_{lb},F_{ub})=(15,25)$, $F=20$ and $P=40$, where all time units are in minutes. The bounds of the protection allowance (including the processing time and interruption estimates) are calculated as: $(P+P\times D_{lb}/F_{ub},P+P\times D_{ub}/F_{lb})=(40+40\times12/25,40+40\times14/15)=(59.2,77.3)$. While using D and F gives us the value of $(P+P\times D/F)=66$. Following the steps, we get $p_{lower-slack}=p_{upper-slack}=(77.3-66)/2=5.65$. If the operation starts at time 0, the type-2 bounds representation is [[0,5.65],[71.65,77.3]]. The results are shown in Figure 4-2.

Figure 4-2: Numerical example for the type-2 bound

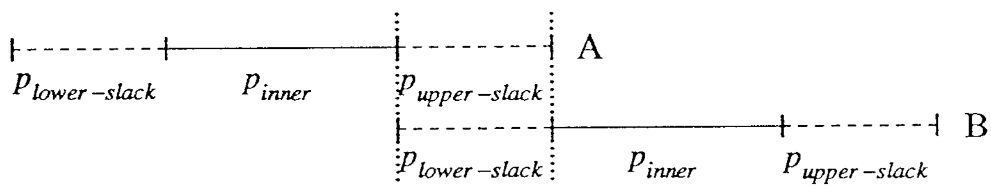


Thus, extended processing times with temporal allowance for uncertainty can be expressed by type-2 bounds. With these representations at hand we can focus on devising its scheduling method.

5.0 SCHEDULING WITH BOUNDS

In scheduling method for type-2 bounds, the mean time is used as a reservation (p_{inner}) necessary for the operation while the protection bounds ($p_{lower-slack}$ or $p_{upper-slack}$) may be overlapped with the protection bounds of other possible consequent jobs. As in Figure 5-1, operation A is designed to be overlapped with operation B. If A completes earlier, then B may start earlier to the extent of its lower bound. If A completes later, then B may start later, again within its bounds. The overlapped section is the $p_{upper-slack}$ part of the operation A and $p_{lower-slack}$ part of the operation B as the bounds are depicted in Figure 5-1. The inner bound, p_{inner} is the reservation to protect the processing for its completion in the bounds.

Figure 5-1: Illustration for two overlapped type-2 bound operations



When uncertainty increases, the precision of this predictive schedule is decreased as the slack segments are larger. These overlapped segments give human dispatchers the flexibility to start the job at any time within the slack bounds.

One may imagine that the bounds work as the earliest start time and the latest start time with the earliest finished time and the latest finished times as the activity representation in project management. However, these slacks are designed for a different purpose to protect against uncertainty. Therefore, when time progresses to the point of the lower bound, an operation should start, if the machine is available, to avoid possible subsequent delays in the operation. In project management, slacks exist to indicate the possible earliest start time and the possible finished time and allow a dispatcher to schedule other works in the remaining slack. In the context of uncertainty, the mean processing time (p_{inner}) is reserved for the operation and the slack time ($p_{upper-slack}$ or $p_{lower-slack}$) is reserved for protection against machine uncertainty. For instance, let a job scheduled to start at time t_0 have a type-2 protection allowance as $[[t_0,t_0+p_{lower-slack}],[t_0+p_{lower-slack}+p_{inner},t_0+p_{outer}]]$. At that particular time t_0, the work is immediately released to the shop and the work-in-process time starts counting from time t_0. Therefore, in Figure5-1, the work-in-process time starts from the beginning point of $p_{lower-slack}$ for either operation A or B. Once an operation is ready for processing, a dispatcher should follow the schedule within the prescribed bounds.

Further computational study is needed to measure the performance of type-2 bound scheduling method and its sensitivity to different structures. In the next section, we have included results from different experiments to verify the advantage of using the type-2 method.

6.0 EXPERIMENTS

In order to determine the effectiveness of this representation on schedule fragility, we conducted a series of experiments. The first series, *Experiment 1*, compares type-2 bound scheduling method with three other fixed processing time bound scheduling methods. The second series, *Experiment 2*, plots the total cost performance under various cost structures to investigate its sensitivity to cost. The third series, *Experiment 3*, examines different approaches to obtain the bounds from known normal distributions D and F.

6.1 Experiment 1: Four Scheduling Methods

Independent variables: Four methods of representing time bounds are selected: type-2, original, mean and upperbound. Two levels of shop load are selected: heavy (eight orders) and light (five orders). The distribution of the interrupt is simulated by a triangular distribution, so that it has a range of [min, max] and the mode which is the peak of the triangular distribution. Two levels of uncertainty variance in F and D are chosen yet with the same mean F and D. The experiment is a fully-crossed, factorial design (4×2×2). The cost structure in the first experiment is assumed known and fixed. Orders are taken from the OPIS experiments (Ow [13]).

Scheduling Methods: The type-2 method that includes the mean processing time into the largest bounds is compared with the other three methods in our experiments:

1. **type-2 method**: using fuzzy type-2 bounds as $[[t, t+p_{lowerslack}], [t+p_{lower-slack}+p_{inner}, t+p_{outer}]]$.
2. **original method**: using the original processing time P as the fixed processing time,
3. **mean method**: using the mean processing time as the fixed processing time (i.e. $P+(P/F)\times D$),
4. **upperbound method**: using the upper bound as the fixed processing time (i.e. using $p_{outer}=P+P\times D_{lb}/F_{ub}$).

Each of the methods proposes different time bounds for a deterministic schedule. The first method has fuzzy type-2 bounds and the other three has fixed values of various length for the processing time.

Scheduling Rule: The scheduling rule used in this work is a *dynamic* version of Jackson's algorithm to minimize the maximum lateness (Baker et. al. [4, 10]). For the static version of the n-job single machine problem, L_{max} is minimized by the sequence of EDD according to Jackson's algorithm. We have it revised to accommodate non-simultaneous order arrivals.

- **Jackson's Algorithm (dynamic version)**: At each job completion the job with the minimum due date b_j among available jobs is selected to begin processing. Let S be the set of unscheduled jobs. The algorithm is as the following:
  1. Set t to 0.
  2. Is there at least one job $\in S$ such that the arrival time $a_i \leq$ t? If so, go to 4.
  3. Set t=min a_i.
  4. Among all jobs $\in S$ such that $a_i \leq$ t choose the job j that has the smallest due date b_j; break ties on due date by selecting the job with the largest duration d_j.
  5. Schedule the chosen job next and update t to t+d_j,
  6. If S is empty, go to 2. Otherwise, the schedule is complete.

At a time when the rest of the jobs have all arrived, the dynamic version of the Jackson's algorithm is equivalent to the static version that is the optimal procedure since all orders are now simultaneously available.

The adaptation of dynamic Jackson's algorithm is for the methods with the extended processing time assumed to be of fixed length. Only the method of type-2 bounds requires further attention to the overlapped segments, since it is desirable to overlap the uncertain slack segments of consequential operations and undesirable to overlap the reservation segment.

Dependent Variables: All data are represented in rows of:

- work-in-process = $\sum$ (actual finish time - planned release time),
- tardiness = $\sum$ (actual finish time - requested due date)$^+$,

where the summations are for all orders. The first dependent variable is the cost component invested in work-in-process, resulted from the difference between the actual completion time and planned release time. The second dependent variable, is the cost component occurred from not meeting the due date, the absolute value of the subtraction between the actual completion time and requested due date. We also use a total cost measurement including both cost components.

Total Cost = C_W * Work-in-Process + C_T * Tardiness

Where C_W and C_T are unit cost of work-in-process and unit cost of tardiness, respectively. The unit cost, assumed linear, which we used for experiment 1 and 3 are C_W=2 and C_T=10, as the tardiness cost is generally higher than the holding cost. Results for the one-pass dynamic Jackson's algorithm are listed for four different scheduling methods, namely, type-2, original, mean time and upperbound.

Case 1a: 5-order schedule with smaller uncertainty

D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
10	15	20	30	40	50

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	2019	2550	2297	1944
tardiness	0	0	0	0
total cost	2019	2550	2297	1944

Case 1b: 5-order schedule with larger uncertainty

D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
5	15	25	20	40	60

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	2091	2824	2571	2067
tardiness	0	0	0	516
total cost	2091	2824	2571	2583

Case 1c: 8-order schedule with smaller uncertainty

D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
10	15	20	30	40	50

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	3436	7137	4500	3194
tardiness	2154	2964	2154	2912
total cost	6590	10101	6654	7106

Case 1d: 8-order schedule with larger uncertainty

D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
5	15	25	20	40	60

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	3412	8063	5384	3378
tardiness	4808	4418	3300	10062
total cost	8220	12481	8684	13440

Case 2a: 50-order schedule

D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
10	15	20	30	40	50

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	30685	57675	43712	21002
tardiness	58952	59948	59180	72506
total cost	89637	117623	102892	93508

Case 2b: 100-order schedule

D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
10	15	20	30	40	50

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	131513	250964	197948	40637
tardiness	331812	360362	329428	474948
total cost	463325	611326	527376	515585

The above tables show how the original processing time and mean time methods lack proper protection against uncertainty. They over-estimated the earliness of the completion time in all cases. If rescheduling cost is to be taken into account, the total cost is even more than the result shown in the tables. In case 1a and 1b, where the shop is lightly loaded that no job is tardy, the work-in-process of the type-2 method is larger than that of the upperbound method. As the type-2 method schedules an order to release at the time when the slack segment overlaps the slack segment of its previous order, it ends up in larger work-in-process as orders are released earlier. In case 1c and 1d, where there are some tardiness, the type-2 method gives the minimum total cost from less tardiness cost than upperbound method, since the larger processing time bounds of the upperbound method pushes the completion time forward. Comparing case 1a with case 1b and case 1c with case 1d, it is clear that larger uncertainty results in more total cost of both work-in-process and tardiness. When the number of orders increases as in case 2a and 2b, type-2 method has the ability of reducing the total cost by the flexibility of the overlapped slack segments.

6.2 Experiment 2: Different Cost Structures

In this section, we use different values of the unit cost for the previous results to investigate the conditions for using the type-2 scheduling method. Nine sets of cost structures are used as the additional independent variable. The first one uses the unit costs of the same magnitude. In cost structures of (2), (4), (6) and (8), the unit cost of work-in-process is less than the unit cost of tardiness. In cost structures of (3), (5), (7) and (9), the unit cost of tardiness is less than the unit cost of work-in-process. In the tables, the total costs are listed while the work-in-process parts are listed inside the parentheses.

It shows that the saving from using type-2 bounds is significant when unit cost of tardiness is higher than unit cost of work-in-process. That unit cost condition is typical true in manufacturing settings. For other cost structures with a higher unit cost of work-in-process, there is no motivation to add temporal slack to protect uncertainty as the scheduling method of original processing and mean processing time perform equivalently well or even better. The performance comparison between the type-2 bound and the upperbound scheduling method is listed in Figure6-1 under different cost structure ratios (i.e. C_W/C_T) : the upper figure is for 50 orders, and the lower one is for 100 orders.

Table of Different Cost Structures for 50 and 100 orders:

CS	1	2	3	4	5	6	7	8	9
C_W	1	1	1	1	1	2	5	10	20
C_T	20	10	5	2	1	1	1	1	1

Result for 50 orders:

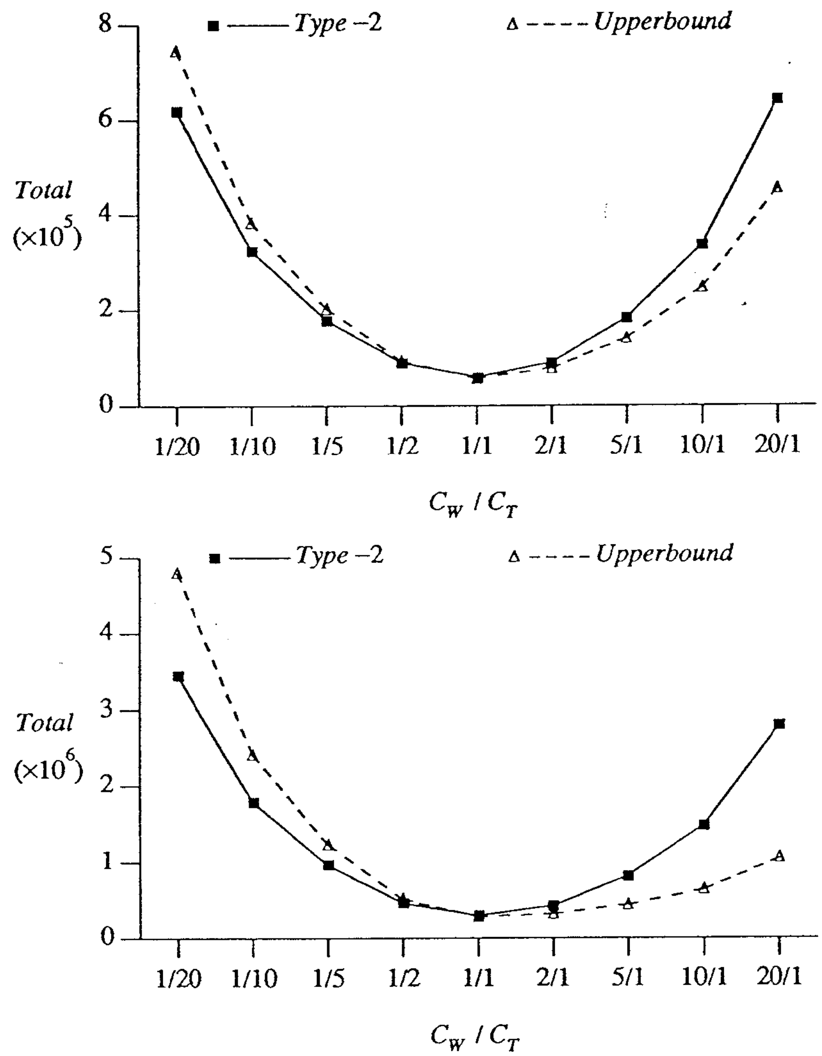
CS	*Type-2*	*Original*	*Mean*	*Upperbound*
1	620205 (30685)	657155 (57675)	635512 (43712)	746062 (21002)
2	325445 (30685)	357415 (57675)	339612 (43712)	383532 (21002)
3	178065 (30685)	207545 (57675)	191662 (43712)	202267 (21002)
4	89637 (30685)	1176231 (57675)	102892 (43712)	93508 (21002)
5	60161 (30685)	87649 (57675)	73320 (43712)	57255 (21002)
6	90846 (61370)	145324 (115350)	117014 (87424)	78257 (42004)
7	182901 (153425)	318349 (288375)	248150 (218560)	141263 (105010)
8	336326 (306850)	606724 (576750)	466710 (437120)	246273 (210020)
9	643176 (613700)	1183474 (1153500)	903830 (874240)	456293 (420040)

Result for 100 orders:

CS	*Type-2*	*Original*	*Mean*	*Upperbound*
1	3449633 (131513)	3854584 (250964)	3492228 (197948)	4790117 (40637)
2	1790573 (131513)	2052774 (250964)	1845088 (197948)	2415377 (40637)
3	961043 (131513)	1151869 (250964)	1021518 (197948)	1228007 (40637)
4	463325 (131513)	611326 (250964)	527376 (197948)	515585 (40637)
5	297419 (131513)	431145 (250964)	362662 (197948)	278111 (40637)
6	428932 (263026)	682109 (501928)	560610 (395896)	318748 (81274)

7	823471 (657565)	1435001 (1254820)	1154454 (989740)	440659 (203185)
8	1481036 (1315130)	2689821 (2509640)	2144194 (1979480)	643844 (406370)
9	2796166 (2630260)	5199461 (5019280)	4123674 (3958960)	1050214 (812740)

Figure 6-1: Total cost of Type-2 and Upperbound for 50 and 100 orders



6.3 Experiment 3: Bounds For Failures Of Normal Distribution

When the two distributions D and F are known to have normal distribution, we can use its standard deviation to decide the bounds for D_{lb},D_{ub},F_{lb} and F_{ub}. Following is an experiment to investigate three different bounds from the standard deviations of the distributions. Independent variables are (1) three different standard deviations: 1, 2 and 3 standard deviations used for the bounds, (2) two different shop loads: heavy (eight orders) and light (five orders) for this 3×2 experiment design. The dependent variable is the total cost under the unit cost structure of (C_W=2 and C_T=10).

Case 3a: 3 different bounds for 5 orders

	D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
1 standard dev.	12.5	15	17.5	36	40	44
2 standard dev.	10	15	20	32	40	48
3 standard dev.	7.5	15	22.5	28	40	52

Three standard deviation result:

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	3978	5110	4604	3916
tardiness	0	0	0	0
total cost	3978	5110	4604	3916

Two standard deviation result:

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	4150	5110	4604	3928
tardiness	0	0	0	0
total cost	4150	5110	4604	3928

One standard deviation result:

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	4448	5110	4604	4218
tardiness	0	0	0	0
total cost	4448	5110	4604	4218

Case 3b: 3 different bounds for 8 orders

	D_{lb}	D	D_{ub}	F_{lb}	F	F_{ub}
1 standard dev.	12.5	15	17.5	36	40	44
2 standard dev.	10	15	20	32	40	48
3 standard dev.	7.5	15	22.5	28	40	52

Three standard deviation result:

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	6506	14350	9038	6430
tardiness	12300	15270	10920	21750
total cost	18806	29620	19958	28180

Two standard deviation result:

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*

work-in-process	7214	14350	9038	6378
tardiness	10920	15270	10920	12630
total cost	18134	29620	19958	19008

One standard deviation result:

Cost	*Type-2*	*Original*	*Mean*	*Upperbound*
work-in-process	8214	14350	9038	7490
tardiness	10920	15270	10920	10920
total cost	19134	29620	19958	18410

In case 3a and 3b, the original method and the mean method have the same values in the columns since these two method do not incorporate the uncertainty variance into the bounds. From above experiment cases, we can see that type-2 method with 2 to 3 standard deviations gives desirable protection. One standard deviation gives too short the processing time bounds, thus orders are released too early as larger work-in-process is held.

7.0 CONCLUSION

In summary, we proposed type-2 bounds for the uncertain processing time by bounds and how to schedule it by overlapping slack segments with possible consequent operations. While compared among the other methods (original processing time, mean processing time and upper bounds methods), the result showed its sufficient protection against deviation with less investment in the total cost in general. The best condition to use type-2 bounds method has been found to be the case that the unit cost of tardiness is significantly higher than the unit cost of work-in-process. Also, for the case that the distributions of time between machine failure and the duration of failure have normal distribution, we showed that methods using even only one standard deviation of the uncertainty distributions into the bound (i.e. the type-2 method and the upperbound method), would result in lower total cost than the methods without. And, in general the type-2 method yields better performance than the other methods.

In the future, we will explore different combinations of operations during a predetermined possible bounds as alternate procedures to be delivered to a foreman instead of one operation per bound as in our experiments. We may also reduce the inner bounds for the reservation that leaves less room for the interruptions instead of using the mean processing time as the inner bound.

ACKNOWLEDGE

This research was supported, in part, by the Defense Advance Research Projects Agency under contract #F30602-88-C-0001.

REFERENCE

[1] Akella, R., Choog, Y. and Gershwin, S.
Performance of Hierachical Production Scheduling Policy.
IEEE transactions on Component Hybrids and manufacturing technology 7(3):225-240, 1984.

[2] Akella, R. and Singh, M., and Krogh, B.
Hierachical Control Structures For Failure Prone Multicell Flexible Assembly Systems.
Technical Report, Graduate School of Industrial Administration, Pittsburgh, PA, 1987.

[3] Robert, A.
Planning and Control of systems: A frame work for analysis.
Technical Report, Graduate school of Business Administration, Harvard University, Boston, 1965.

[4] Baker, K.R.
Introduction to Sequencing and Scheduling.
John Wiley & Sons, 1974.

[5] Fox, M.S. and Sycara, K.
Overview of CORTES: A Constraint Based Approach to Production Planning, Scheduling and Control.
In *Proceedings of the Fourth International Conference on Expert Systems in Production and Operations Management.* May, 1990.
Submitted for publication.

[6] Gerchak, Y., Vickson, R.G., and Parlar, M.
Periodic Review Production Models with Variable Yield and Uncertain Demand.
IIE Transactions 20(2):91-103, 1988.

[7] Graves, S.
Safety Stock in Manufacturing Systems.
Technical Report, Sloan School of Management, MIT, Cambridge, 1987.

[8] Hax, A., and Candea, D.
Production and Inventory Management.
Prentice-Hall, New Jersy, 1984.

[9] Kaufmann, A., and Gupta, M. M.
Introduction to Fuzzy Arithmetic.
Van Nostrand Reinhold Company, 1984.

[10] McMahon, G., and Florian, M.
On Scheduling with Ready Times and Due Dates to Minimize Maximum Lateness.
Operations Research 23(3):475-482, 1975.

[11] Morton, T.E., Lawrence, S.R., Rajagopalan, S., and Kekre, S.
MRP-Star: PATRIARCH's Planning Module.
Technical Report, Carnegie-Mellon University, 1986.

[12] Morton, T.E., Lawrence, S.R., Rajagopalan, S., and Kekre, S.
Sched-Star.
Technical Report, GSIA, Carnegie-Mellon University, 1987.

[13] Ow, P-S.
Experiements in Knowledge-based Scheduling.
Technical Report, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, April, 1986.

[14] Prade, H.
Using fuzzy set theory in a scheduling problem.
Fuzzy Sets and Systems 2(2):153-165, 1979.