Problems with Fuzzy Allowable Time Constraints between Two Machines Scheduling

Fuzzy Scheduling problems with an extension to multiple criteria and job priorities

Subodh kant (M.Tech software Engineering)

Department of CSE, SRMSCET, Bareilly (U.P.)

Email: Subodhiso.kant@gmail.com

Abstract: we have already introduced many fuzzy concepts to scheduling problems and discussed so called fuzzy scheduling problems. In This paper we introduce fuzzy allowable time concept newly to two identical machines and fuzzy ready times and deadlines are associated to each job, i.e., membership representing a satisfaction degree of start times and completion time are considered to each job and minimal one of them is to be maximized. Further among jobs ,there exists a fuzzy precedence relation, which is a fuzzy relation with membership function representing satisfaction degree of precedence order of each job pair and again minimal one of them is to be maximized .The aim is to maximize of both minimal degree at a time if possible ,but usually there exists no schedule maximizing both of them and so we seek no dominated Schedules.

In real world scheduling Problems, priorities of jobs may change over time. A less important job today may be of high importance tomorrow and vice-versa.

Key words: Fuzzy Scheduling, membership function, fuzzy relation, time constraint , job scheduling, Fuzzy sets.

I. INTRODUCTION

We have already introduced many fuzzy concepts to ordinary scheduling problems and proposed efficient solution procedure for them [3, 4, 5]. For most of scheduling problems, jobs' processing times and due-dates are treated as certain values, but that is not proper to all actual situations. Processing times are not constant because of measurement errors in the data sets for deciding them and/or human actions in the manufacturing process. However, we may be permitted to exceed due-dates slightly. In this paper, we propose a model dealing with uncertain processing times and flexible due-dates in consideration of real situations. Assuming the times and due-dates to be fuzzy numbers, and defining a fuzzy tardiness for a job's due-date, we design to minimize the number of tardy jobs in our problem formulation.

In this paper we introduce fuzzy allowable time constraint to the problem considered in [2], i.e., fuzzy ready time and deadline are associated to each job. Corresponding membership functions of each job represent satisfaction degrees of start time and completion time of job processing. Minimal satisfaction degree of them is to be maximized. In a real world, final deadlines depend upon types production priority of job customers etc. For an example, exports are to be completed rigidly before shipping. But in some cases, slight delay is allowed. Further generally speaking, all customers requirements does not necessarily propose to processing his jobs from a start. Further among jobs, there exists a fuzzy precedence relation, which is a fuzzy relation with membership function representing satisfaction degrees of processing order between each job pair and again minimal one of them is to be maximized. Since usually there exists no schedule maximizing both minimal degrees at a time, we seek nondominated schedules. Section 2 ormulates the problem and defines nondominated schedules. Section 3 briefly review the corresponding nondominated scheduling problem in [2]. Section 4 proposes a solution procedure for nondominated schedules of the problem. Section [5] concludes this paper.

2. LITERATURE REVIEW

In this section we briefly review following fuzzy scheduling problem . H. Ishii [6] introduced the fuzzy scheduling problem with the fuzzy due-date before. Up to now, no paper about fuzzy scheduling has taken up both fuzzy due-dates and ready time yet.

i. There are two identical machines and ***n*** jobs Ji, J2,. -. , Jn that are to be processed by either one of these two machines.
ii. Each job Ji has unit processing time, fuzzy executable start time Ŝi and fuzzy deadline Ďi. Ŝi has the following membership function to represent the satisfaction degree regarding the processing start time Ŝi of the job (supposed as a nonnegative integer).

$$\mu\ (Si) = \sum \begin{cases} \mathbf{0} & \mathbf{(Si \le Ri)} \\ \mathbf{Mi(Si)} & \mathbf{(Ri \le Si \le Ri+Ei);} \\ \mathbf{1} & \mathbf{(Si \ge Ri+Ei);} \end{cases}$$

where si is integer, and mi(si) is nondecreasing and has a value between 0 and 1. In the similar manner, di has the following membership function to represent the satisfaction degree regarding the completion time *C;* of the job.

$$\mu(Ci) = \sum \begin{cases} 1 & (Ci \le Di) \\ K(Ci) & (Di \le Ci \le Di+Fi) \\ 0 & (Ci \ge Di+Fi) \end{cases}$$

where, (Ci; is an integer, and *k(Ci)* is nonincreasing and has a value between 0 and 1.
Further Ri, Ei, Fi, Di > 0, ri + ei < Di;, and they are integers. Roughly speaking,ŕ = ri + ei corresponds to the conventional ready time, while di corresponds to deadline.

iii. H. Ishii [5] has first introduced a flexible processing order of some pairs of jobs as fuzzy precedence relation. It is defined between two jobs Ji and *Jj* and is represented by the satisfaction degree μij when Ji precedes Jj, namely, when Ji is processed before Jj. We assume, μij is a value between 0 and 1, and, μij = 1 in the case of μij > 0, while in the case of μij, = μ_{ji} = 1, Ji and Jj are independent each order. Further simultaneous processing is not allowed except independent jobs.

iv. When the processing start time of the job Ji under schedule π is represented by Si^{π}:, and the completion time is represented by **Ci** $^{\pi}$*,* then the minimum satisfaction degree $\boldsymbol{\mu rmin^{\pi}}$ regarding the processing start time in π is given as the follows

$\mu rmin^{\pi}$ = min {$\mu si^{-}(Si^{\pi})$ I i= 1,2...n}
And the minimum value of the sat isfaction degree regarding the completion time $\mu dmin^{\pi}$ is given as follows:
$\mu dmin^{\pi}$=min{$\mu di^{-}(Ci^{\pi})$ i=1,2...n}
Therefore, the satisfaction degree regarding the job execution time µ1: is defined as shown below:
µ1 = {min($\mu rmin$, $\mu dmin$}
While, the minimum value of satisfaction degree regarding the job processing order µ2 is given as:
µ2 = {min(μij ICi<Cj}

Under the above setting, the following problem Q is considered.

- Q: µ1→max , µ2→max
  Subject to π € Π

where, Π is a set of all the feasible schedules.
In general, there is no schedule to maximize both µ1and µ2 simultaneously, therefore we seek for a nondominated schedule defined next. There is a possibility that there are many nondominated schedules having an identical schedule vector. However, as for the same nondominated schedule vector, only a single schedule of them is sought.
The investigated scheduling problem of the Sherwood Press Corporation may be modelled as a job shop problem, in which a set of machines M □□{M1,...,Mm} exists that perform operations on a set of jobs J □□{J1,..., Jn} (Błażewicz et al. 1996). Each job Ji ,i □1,...,n consists of an ordered set of tasks Ti □□{Ti1,...,Tit} , and each task Tij ,i □1,...,n, j □1,...,t may be processed on a certain machine with a nonnegative processing time pij . It is assumed that each machine may process at most one task and each task may be processed by at most on machine at a certain time.

V. Nondominated Schedule
First of all, schedule vector
$v\pi=(\mu 1\pi,\mu 2\pi)$
is defined to each schedule π. That the scheduleπ 1dominates $\pi 2$ *means*:
V1 $\pi 1 \geq$ V2 $\pi 2$,
V1 $\pi 2 \geq$ V2 $\pi 1$,
V $\pi 1 \neq$ V $\pi 2$
for the correspond schedule vectors
V $\pi 1$=(V1 $\pi 1$, V2 $\pi 1$) *and* V$\pi 2$=(V1$\pi 2$, V2 $\pi 2$)
That the schedule π is called nondominated schedule when there is no schedule dominating π..*[3]*.

3.NONFUZZY TWO MACHINES PROBLEM WITH PRECEDENCE RELATION,READY TIME AND DEADLINE.

Here we briefly review the solution procedure of two-machine problem with nonfuzzy precedence relation, ready time, and deadline by Garey and Johnson [2]. First, the problem is given by the following (1) to (4):
i.There are two machines and n jobs J1,J 2, …Jn that are to be processed by either of these two machines.

ii. Each job Ji; has unit processing time, and ready timeri¯ and deadline di are defined. Namely, Ji*;* must start processing after ri¯ ,, and complete processing before the deadline di .

iii. Precedence relation ɣ is defined among some jobs. The expression Ji ɣ Jj means that the job Ji must precede the job Jj, in other words, Ji must be processed before the processing of Jj is started. Two jobs not having this relation are called independent each other. To each job Ji ;, the job Jj in the relation of Ji ɣ Jj is called a successor job of Ji.

iv. Under the above setting, a schedule to meet all of ready time, deadline, precedence relation is seeked for.
First, *S(i,*S, d) to each job Ji and ri¯ ≤ S ≤ di≤d is defined as a set of all the jobs Jj *(j* # i) that has dj < d and is either the succeeding job of Ji; or rj≥S. And N(i, S, d) is defined as the number of its elements. Then when it holds that,
N(i, S, d)≥2(d-s) & d- ┌ N(i, S, d)/2┐ < di
deadline of Ji is modified as di = d- ⌈ N(i, S, d)/2 ⌈ where ┌.┐ means the minimum integer not below **. .** When this modification is repeated, it leads to either the case where modification is no longer carried out or the case di < ri¯ + l occurs. Algorithm to carry out this modification efficiently is omitted here, but at the moment when correction is not available, it leads to the following internally consis tent status. (For details, see [2] .)
Internally Consistent View :
We call the deadlines internally consistent whenever the following conditions hold for every job Ji;

i. di < ri¯ + l
ii. For every pair of integers S,d satisfying
Si< S < di < d,if N(i, S,d)≥2(d-s),then d- ┌ N(i, S, d)/2┐ .

4. SOLUTION PROCEDURE FOR FUZZY VERSION AND SCHEDULING METHOD

First sort all µij such that 0 < µij < 1 in the fuzzy precedence relation, and let the results be as shown below:
µ0=1 > µ1> µ2…… µa>0
where, a is the number of the different µij And, also sort
µSi¯(Si) (ri≤ Si≤ ri+ ei; integer 1,2,3….n)
and
µai(Ci) (di≤ Ci≤ di+ fi; integer 1,2,3….n)

and let following be obtained;
µ0=1> µ1> µ2>.... µb>0= µb+1
where b is the number of different
µSi¯(Si), µdi¯(Ci) in (0,1)

Algorithm

[1].calculate $\mu 0=1> \mu 1> \mu 2.....> \mu b+1=0$

Set l=0 ,construct pg0 and search μt *, t=0,1,2....b*

$\lceil \mu_{si}^{-1}(\mu+) \rceil$, $\lfloor \mu_{si}^{-1}(\mu+) \rfloor$ *,respectively*

And find find the maximum $\mu t= \mu to$ *to make it feasible and obtain corresponding schedule*

π 0. Set ,DS={ π 0},l=1 and go to step 2

[2]. Construct *pg1 search* μt *, t=0,1,2.... tl-1-1*

and set ready time, deadline of

each job $\lceil \mu_{si}^{-1}(\mu+) \rceil$, $\lfloor \mu_{si}^{-1}(\mu+) \rfloor$ *,respectively And find find the maximum* $\mu t= \mu to$ *to make it feasible and obtain executable schedule π l. Set ,DS=DS U{ π l}, and go to step 3 Otherwise, set* $\mu t = \mu tl-1$ *and go to step 3*

[3]. Set l = l + 1. Terminated in the case of l = a + 1. Otherwise, return to step 2.
(Herein, $\lceil \ \rceil$ means the minimum integer not less than the content, and $\lfloor \ \rfloor$ means the maximum integer not greater than the content.)

Theorem : The above algorithm finds out no dominated schedule in

$O(\sum_{i=1}^{n}(fi + ei))n^3))$

Proof: Computational complexity is
equal to calculation amount of the above result in order since each μi , i = 0,1, - - - , b + 1 is checked once, and the check on each feasibility is made in $0(^{n^3})$ computational time.

Theorem : Dominant set of points in O(n) if two points A(x1,y1) and B(x2,y2) are given, and if x1 <= x2 and y1<= y2, then we say B dominates A. Now, given a lot of points, I wish to find out all the non-dominated points. Trivial approach is compare every point with others and get all non-dominated points. But it's O(n^2). So I tried divide and conquer, pretty straightforward and I get to find that in O(nlogn). Our professor says, it can still be done in O(n). I kind of think it's really not possible. I'm not asking you to solve this for me, but would like to know if there's any 'obvious' way through which I can be sure that it can't be done in O(n) under any conditions?

5. CONCLUSION AND FUTURE WORK

In this paper, we have discussed some key points that, in our opinion, motivate a constructive Problems with Fuzzy Allowable Time Constraints between Two Machines Scheduling . The concept of fuzzy allowable time may be introduced into many other scheduling problems. Further, it may be possible to consider a tri-criteria scheduling problem taking fuzzy processing time intoconsideration such as [7].

We extend some Scheduling algorithm for corresponding nonfuzzy problem to fuzzy version and proposed an efficient algorithm. But our above algorithm is straightforward and so may be refined more. For most of scheduling problems, jobs' processing times and due-dates are treated as certain values, but that is not proper to all actual situations. Processing times are not constant because of measurement errors in the data sets for deciding them and/or human actions in the manufacturing process. However, we may be permitted to exceed due-dates slightly. A study of a scheduling problem with fuzzy due dates and dynamically changing preferences has been presented, based upon an investigation of a real world problem. A fuzzy membership function has been used in order to describe the imprecise due dates and
tardiness.

References

[1]. M. Fujii, T. Kasami and K. Ninomiya: Optimal sequencing of two equivalent processors. SIAM J. Appl. Math., 17 (4) (1969) 784-789.

[2]. M.R. Garey and D.S. Johnson: Two-processor scheduling with start-time and deadlines. SIAM J. Comput., 6 (3) (1977) 416-426.

[3]. H. Ishii: Two machine Lmax minimization problem with fuzzy precedence constraint. Proceeding of APORS, 94 (1994).

[4]. H. Ishii and T. Konno: An open shop scheduling with fuzzy allowable time and fuzzy resource constraint. Fuzzy Sets And Systems (accept).

[5]. H. Ishii and M. Tada: Single machine scheduling with fuzzy precedence relation. European Journal of Operational Research, 87 (1995) 284-288.

[6]. H. Ishii, M. Tada and T. Masuda: Two scheduling problems with fuzzy duedates. Fuzzy Sets and Systems, 46 (1992) 339-347.

[7]. M. Tada and H. Ishii: Single machine scheduling with fuzzy precedence relations and Fuzzy processing times (to be submitted).

[8]. Beasley, J.E. (1990) OR-Library: Distributing test problems by electronic mail. Journal of the Operational Research Society, 41 (11), 1069-1072.

[9]. //www.**sciencedirect.com**