Journal of Basrah Researches ((Sciences)) Volume 37. Number 4. C ((2011))

Single machine scheduling to minimize three hierarchically criteria

Haidar younis kawi
Department of math.
College of computer science and mathematics
Al_Qadisiya university

Tariq Salih Abdul-Razaq
Department of math.
College of science
Al_Mustansiriya university

Email: hadirkawi@yahoo.com

Abstract

In this paper, to minimize a function of three cost criteria for scheduling n jobs on a single machine , the problem is discussed :

{Minimizing a function of three criteria maximum tardiness ,maximum earliness and sum square of completion time in hierarchical method }.

A set of *n* independent jobs has to be scheduled on a single machine that is continuously available from time zero onwards and that can handle no more than one job at a time. Job $J_j(j=1, \dots, n)$ requires processing during a given positive uninterrupted time p_j, without loss of generality, we assume p_j to be integral.

For the problem we proposed some algorithms and procedure to find exact and best possible solutions for three criteria maximum tardiness ,maximum earliness and sum square of completion time in hierarchical case.

1. Introduction:

In real life situations, decisions to be made are often constrained by specific requirements. More importantly, these constraints are typically conflicting in nature. The decision making process gets increasingly more complicated with increment in the number of constraints. Modeling and development of solution methodologies for these scenarios have been the challenge for operations researchers from the outset. A variety of algorithms and formulations have been developed for various classes of problems.

Scheduling is one of such classes of problems [1].

Scheduling theory has been developed to solve problems occurring in for instance production facilities. The basic scheduling problem can be described as finding for each of the tasks, which are also called jobs, an execution interval on one of the machines that are able to execute it, such that all side-constraints are met; obviously, this should be done in such a way that the resulting solution, which is called a schedule, is best possible, that is, it minimizes the given objective function[3].

Because the one-machine problem provides a useful laboratory for the development of ideas for heuristics and interactive procedure that may prove to be useful in more general models, we consider the one-machine case in this study.

There are two approaches for the multi-criteria problems; first, the hierarchical approach and the simultaneous approach. In the hierarchical approach, one of the two criteria is considered as the primary criterion and the other one is considered as the secondary criterion. The problem is to minimize the primary criterion while breaking ties in favor of the schedule that has the minimum secondary criterion value. Second, in the simultaneous approach two criteria are considered simultaneously. This approach typically generates all efficient schedules and selects the one that yields the best composite objective function value of the two criteria.

Since the introduction of scheduling theory in the 1950'$_s$, most research has been concentrated on single –criterion optimization. In the real-life problems, multiple and usually conflicting criteria play a role [8].The problems of single machine with three criteria have recently appeared in the literature. Therefore, little has been done in this area of multi-criteria scheduling theory. (see: Nelson, et al. 1986[11], Hoogeveen 1996[6], Erne 2007 [2]).

Nelson, et al. 1986[11] presented some algorithms for the three ,tow –criteria problems utilizing mean flow time F , maximum tardiness T_{max} ,and number of tardy job n_T ; all of these functions are regular function . Hoogeveen 1996[6] presented an algorithm to minimize an objective function that is a nondecreasing function of K performance regular criteria .

Erne 2007 [2] gave a heuristic method for multicriteria scheduling problem with sequencing dependent setup time for the objective function for problem to minimization of the weighted sum of total completion time , maximum tardiness and maximum earliness by integer programming model.

A *schedule* σ defines for each job J_j its completion time $C_j(\sigma)$ such that the jobs do not overlap in their execution; we omit the argument σ if there is no confusion possible as to the schedule we are referring to. The cost of completing J_j at time C_j (j=1, . . . , n)is measured by K ($K=3$) penalty functions f_j^k (k=1, . . . , K); two of these penalty functions are assumed to be ***maximum functions*** and the third one is a ***sum function***.

The multicriteria problem that we consider concerns the hierarchical minimization of the performance measure square completion time $\sum_{i=1}^{n} C_i^2$ and maximum cost f_{max} ,maximum cost is define $f_j(C_j)$,where each f_j denoted a regular or irregular cost function ; regular means that $f_j(C_j)$ does not decrease when C_j is increase such as T_{max} , L_{max} and $\sum_{i=1}^{n} C_i^2$ **.** other wise function called irregular such as E_{max} .

Before discussing the results on multi-criteria scheduling that have appeared in the literature , we look at the basic concepts of multi-criteria scheduling.

2. Basic Concepts and Notation

We start with introducing some important notation where we concentrate on the performance criteria without elaborating on the machine environment etc. We assume that there are n jobs, which we denoted by $j_1,\dots,j_n$ these jobs are to be scheduled on a set of machines that are continuously available from time zero on words and that can handle only one job at a time .

In this paper , we only state here the notation that is used for single machine , jobs $J_i(i=1,\dots,n)$ has:

N: set of jobs.

n: The number of jobs in a known sequence.

$\mathbf{P}_j$: which means that it has to be processed for a period of length p_j .

$\mathbf{d}_j$: a due date ,the date when the jobs should ideally be completed , the completion of job after its due date is allowed ,but a penalty is incurred . When the due date absolutely must be met , it is referred to as deadline $\bar{d}_j$, and when due date is constant for all jobs ,then called common due date.

$\mathbf{r}_j$: a release date of job j ,i.e. the earliness time at which the processing of job can begin.

- The completion time C_j
- The flow time $F_j=C_j-r_j$

- The lateness $L_j=C_j - d_j$
- The tardiness $T_j=\max\{ 0, C_j-d_j\}$
- The earliness $E_j= \max\{ 0, d_j-C_j\}$
- The unite penalty $U_j=1$ if $C_j>d_j$, $U_j=0$ if $C_j\le d_j$

For a given schedule σ we compute.

- $C_{max}(\sigma) =\max_j(C_j)$
- $E_{max}(\sigma)=\max_j(E_j)$
- $L_{max}(\sigma)=\max_j(L_j)$
- $T_{max}(\sigma)=\max_j(T_j)$

Theorem(1)(Jackson 1955)[9]. The $1/ / L_{max}$ problem is minimized by sequencing the jobs according to the earliest-due- date (EDD) rule, that is, in order of non-decreasing d_i.■

Theorem (2)(Smith 1956)[12]. The $1/ / \Sigma C_i$ problem is minimized by sequencing the jobs according to the shortest –processing-time (SPT) rule ,that is, in order of non-decreasing p_i.■

Theorem(3)(Lawler 1973)[10].The $1//f_{max}$ problem, f_{max} is minimized as follows: while there are unassigned jobs, assign the job that has minimum cost when scheduled in the last unassigned position in that position.■

Theorem(4)[4]. The $1/ /E_{max}$ problem is solved by sequencing the jobs according to the minimum slack time (MST) rule ,that is ,in order non-decreasing d_i-p_i .■

In order to describe the multi-objective optimization in general. Consider a program with k ($k\ge 2$) conflicting objective functions

($f_i : R^n \longrightarrow R$) that are to be minimized simultaneously. That is, we wish to find a solution, $x=(x_1,x_2,\ldots,x_k)$, from the set of feasible solutions, X, that solves the problem $\underset{x\in X}{Min}\, f(x)=\{f_1(x),f_2(x),\ldots,f_k(x)\}$. Since it is assumed that the objectives conflict, there is no single value of x that minimizes all objectives simultaneously in the single objective sense, so ‘‘optimal’’ must be defined differently .

***Definition (1)[**6**]**: A point $x=(x_1, \ldots , x_k)$is Pareto optimal within a given set S, if S does not contain any other point $y=(y_1, \ldots , y_k)$ with $y_i \le x_i$ for $i =1, \ldots , k$.
Correspondingly, a schedule σ corresponds to a Pareto optimal point if there is no feasible schedule σ' with $f^k_{max}(\sigma')\le f^k_{max}(\sigma)$ for $k=1, \ldots , K$, where at least one of the inequalities is strict; in this case, we say that σ is not dominated.*

***Definition(2)[**6**]**: hierarchical minimization :The performance criteria $f_1, \ldots , f_k$ are indexed in order of decreasing importance. First, f_1 is minimized. Next, f_2 is minimized subject to the constraint that the schedule has minimal f_1 value. If necessary, f_3 is minimized subject to the constraint that the values for f_1 and f_2 are equal to the values determined in the previous step.*

3. **Analysis of the Three-Criteria .**

In this section, we present the main multi-criteria scheduling results that have appeared in the literature , we analyze the three-criteria problem. Hoogeveen and Van de

Valde [7] solve the problem $1//F(\sum C_i, f_{max})$. Hoogeveen [6] solves the problem $1//F(f_{max}, g_{max})$ and the problem $1//F(f_{max}, g_{max}, h_{max})$. He presented a polynomial algorithm for both problems and he showed that these can be used if precedence constraints exist between the jobs or if all penalty functions are non-decreasing in the job completion times. Hoogeveen[5] solves the general problem $1//F(f^1_{max}, \ldots, f^k_{max})$,k finite integer number and each one of these penalty functions is assumed to be non-decreasing in the job completion time .

There are two methods for dealing with conflicting criteria, the first one is hierarchical minimization of the performance criteria $f^1, f^2, \ldots, f^k$ which are indexed in order of decreasing importance . The second method is simultaneous minimization . The criteria are aggregated into a single composite objective function $F(f^1, f^2, \ldots, f^k)$,which minimized.

4. The $1//F(\sum_{i=1}^{n} C_i^2, T_{max}, E_{max})$ problem

The problems and the algorithms considered in this section belongs to first class of multi-criteria (hierarchical optimization).All of these problems are special case of the general problem $1//F(\sum_{i=1}^{n} C_i^2, T_{max}, E_{max})$ .Hence we have the following:

1- $1//Lex(\sum_{i=1}^{n} C_i^2, T_{max}, E_{max})$ problem.

2- $1//Lex(\sum_{i=1}^{n} C_i^2, E_{max}, T_{max})$ problem.

3- $1//Lex(T_{max}, \sum_{i=1}^{n} C_i^2, E_{max})$ problem.

4- $1//Lex(E_{max}, \sum_{i=1}^{n} C_i^2, T_{max})$ problem.

5- $1//Lex(E_{max}, T_{max}, \sum_{i=1}^{n} C_i^2)$ problem.

4.1. The $1//Lex(\sum_{i=1}^{n} C_i^2, T_{max}, E_{max})$ problem.

This problem can be defined as:

$$\left.\begin{array}{l} \text{Min } E_{max} \\ \text{s.t.} \\ \sum_{i=1}^{n} C_i^2 = C^*, \quad C^* = \sum_{i=1}^{n} C_i^2 \text{ (SPT)} \\ T_{max} \le T, \quad T \in [T_{max}(EDD), T_{max}(SPT)] \end{array}\right\} \ldots (P1)$$

Since – in this problem (P1)- the $\sum_{i=1}^{n} C_i^2$ is the more important function and should be optimal, then the following simple algorithm (CTE) gives the optimal required result.

Algorithm (CTE):

Step(0): Order the jobs by SPT rule and calculate $(\sum_{i=1}^{n} C_i^2, T_{max}, E_{max})$ point .

Step(1): If there exist a tie (jobs with equal processing times) ,order these jobs by EDD rule ,and if a tie is still, order these jobs by MST rule.

Example(1): Consider the problem (P1) with the following data.

i	1	2	3	4	5
P_i	1	1	7	2	9
d_i	18	8	7	6	9
s_i	17	7	0	4	0

The SPT rule gives the schedules (1,2,4,3,5) and(2,1,4,3,5) according to algorithm (CTE), we choose the schedule(2,1,4,3,5) that gives minimum $\sum_{i=1}^{n} C_i^2$ with $(\sum_{i=1}^{n} C_i^2$, T_{max}, $E_{max})$=(542,11,16).

4.2. The 1//Lex($\sum_{i=1}^{n} C_i^2$, E_{max}, T_{max}) problem.

This problem can be defined as:

$$\left.\begin{array}{l} \text{Min } T_{max} \\ \text{s.t.} \\ \sum_{i=1}^{n} C_i^2 = C^* \ , \ C^* = \sum_{i=1}^{n} C_i^2 \text{ (SPT)} \\ E_{max} \le E \quad , E \in [E_{max}(MST), E_{max}(SPT)] \end{array}\right\} \ldots (P2)$$

Since – in this problem (P2)- the $\sum_{i=1}^{n} C_i^2$ is the more important function and should be optimal ,then the following simple algorithm (CET) gives the optimal required result.

<u>Algorithm (CET):</u>

Step(0): Order the jobs by SPT rule and calculate ($\sum_{i=1}^{n} C_i^2$, E_{max}, T_{max}) point .

Step(1): If there exists a tie (jobs with equal processing times) ,order these jobs by MST rule ,and if a tie is still, order these jobs by EDD rule.

Example(2): Consider the problem (P2) with the following data.

i	1	2	3	4	5
P_i	2	2	5	9	5
d_i	10	10	9	19	5
s_i	8	8	4	10	0

The SPT rule gives the schedules (1,2,3,5,4), (2,1,3,5,4),(1,2,5,3,4),and(2,1,5,3,4); according to algorithm (CET), we choose the schedule(1,2,5,3,4) that gives :

($\sum_{i=1}^{n} C_i^2$, E_{max}, T_{max})=(826,8,5).

4.3. The 1//Lex(T_{max}, $\sum_{i=1}^{n} C_i^2$, E_{max}) problem.

This problem can be defined as:

Min E_{max}

s.t.
$T_{max}=T^*$,$T^*=T_{max}(EDD)$ …(P3)
$\sum_{i=1}^{n} C_i^2 \leq C^*$, $C^* \in [\sum_{i=1}^{n} C_i^2 (SPT), \sum_{i=1}^{n} C_i^2 (EDD)]$

Since – in this problem (P3)- the T_{max} is the more important function and should be optimal ,then the following algorithm (TCE) gives the best possible solution.

<u>Algorithm (TCE):</u>

Step(0): Solve $1//T_{max}$ problem to find T* .

Step(1): Determined $\bar{d}_j=d_j+T^*$,and $s_j=d_j-p_j$ $\forall j \in N$, N={1,…,n} unscheduled jobs ,and σ=(φ) for schedule jobs.

Step(2): Let $t=\sum_{j=1}^{n} p_j$, k=n.

Step(3): Find a job $j \in N$ satisfy $\bar{d}_j \geq t$ (if there exists a tie choose the job with biggest processing time and if a tie is still, choose the job with biggest slack time).

Step(4): Set $t=t-p_{j^*}$,k=k-1,N=N-{j*},σ=(σ(k),σ) if N=φ goes to step(5) ,else goes to step (3) .

Step(5) : For a schedule σ find T_{max} , $\sum_{i=1}^{n} C_i^2$,and E_{max} .

Example(3): consider the problem (P3) with the following data.

i	1	2	3	4	5
P_i	3	1	7	10	10
d_i	4	12	14	11	10
s_i	1	11	7	1	0

T*=17 , t=31

$\bar{d}_1=21$, $\bar{d}_2=29$, $\bar{d}_3=31$, $\bar{d}_4=28$, $\bar{d}_5=27$

i	t	j*
1	31	3
2	24	4
3	14	5
4	4	1
5	1	2

Hence the schedule (2,1,5,4,3) gives (17,1750,11).

4.4. The $1//Lex(E_{max}, \sum_{i=1}^{n} C_i^2, T_{max})$ problem.

This problem can be defined as:

$$\left.\begin{array}{l} \text{Min } T_{max} \\ \text{s.t.} \\ E_{max}=E^* ,E^*=E_{max}(MST) \\ \sum_{i=1}^{n} C_i^2 \leq C^* ,\ C^* \in [\sum_{i=1}^{n} C_i^2 (SPT), \sum_{i=1}^{n} C_i^2 (MST)\quad] \end{array}\right\} \ldots(P4)$$

Since – in this problem (P4)- the E_{max} is the more important function and should be optimal ,then the following algorithm (ECT) gives the best possible solution.

Algorithm (ECT):

Step(0): Order the jobs by MST rule and calculate $E_{max}(MST)= E^*$.

Step(1): Let k=1,and calculate $r_j=\max\{s_j - E^*, 0\}$for every job $j \in N=\{1,..,n\}$ of unscheduled jobs, $\sigma = (\varphi)$, σ be the schedule jobs.

Step(2): Find a job $j^* \in N$ with minimum r_j such that $r_{j^*} \le C_{k-1}$(if there exists a tie, choose the job j^* with smallest p_{j^*},if a tie is still, choose the job with smallest due date),$C_0=0$ when k=1.

Step(3): Set $N=N-\{j^*\}$, $\sigma = (\sigma, \sigma(k))$. If $N= \varphi$ goes to step(4),else k=k+1, goes to step(2).

Step(4): Calculate $E_{max}(\sigma)$, $\sum_{i=1}^{n} C_i^2$ (σ)and $T_{max}(\sigma)$.

Example(4): consider the problem (P4) with the following data.

i	1	2	3	4	5
P_i	1	1	7	2	9
d_i	18	8	7	6	9
s_i	17	7	0	4	0

$E^*=0$, $r_1=17, r_2=7, r_3=0, r_4=4, r_5=0$

j	r_j	C_{k-1}	j*
1	17	0	3
2	7	7	2
3	0	8	4
4	4	10	5
5	0	19	1

Hence the schedule (3,2,4,5,1) gives the best solution (0,974,10).

4.5. The $1//Lex(E_{max}, T_{max}, \sum_{i=1}^{n} C_i^2)$ problem.

This problem can be defined as:

$$\left.\begin{aligned} &\text{Min } \sum_{i=1}^{n} C_i^2 \\ &\text{s.t.} \\ &E_{max}=E^* \qquad ,E^*=E_{max}(MST) \\ &T_{max} \le T \quad , T \in [T_{max}(EDD), T_{max}(MST)] \end{aligned}\right\} \ldots (P5)$$

This problem can be written as $1/ E_{max}=E^*, T_{max} \le T / \sum_{i=1}^{n} C_i^2$

Now, we will present the following procedure (ETC) to find the best possible solution for problem (P5).

Procedure (ETC):

Step(0): Solve $1//T_{max}$ and $1//E_{max}$ problem to find T* and E* .

Step(1): Let k=1,and calculate $r_j=\max\{s_j - E^*, 0\}$for every job $j \in N=\{1,..,n\}$ of unscheduled jobs, $\sigma = (\varphi)$, σ be the schedule jobs.

Step(2): Find a job $j^* \in N$ with minimum r_j such that $r_{j^*} \le C_{k-1}$(if there exist a tie, choose the job j^* with smallest p_{j^*},if a tie is still, choose the job with smallest due date d_{j^*}),$C_0=0$ when k=1.

Step(3):$N=N-\{j^*\}$, $\sigma = (\sigma, \sigma(k))$. If $N= \varphi$ goes to step(4),else k=k+1 ,goes to step(2).

Step(4): For the complete schedule σ find $T_{max}(\sigma)$; if $T_{max}(\sigma)=T^*$ then $\sigma_{min}=\sigma$,goes to step(5);else if $E_{max}(\sigma(1))=E^*$,interchange position $\sigma(n)$ and $\sigma(n-1)$ and find $T_{max}(\sigma_{new})$,if $T_{max}(\sigma_{new})=T^*$ then
$\sigma_{min}=\sigma_{new}$,goes to step(5); else choose the schedule from σ and σ_{new} that has minimum T_{max} and called σ_{min} .

Step(5): Reset $N=\{1,\dots,n\}$unscheduled jobs ,and $\pi=(\varphi)$ for schedule jobs, k=1.

Step(6): Find a job $j^* \in N$ with minimum r_j such that $r_{j^*} \le C_{k-1}$(if there exists a tie, choose the job j^* with smallest d_{j^*} ,if a tie is still, choose the job with smallest p_{j^*}),$C_0=0$ when k=1.

Step(7):$N=N-\{j^*\}$, $\pi=(\pi\ ,\pi(k))$. If $N=\varphi$ go to step(8),else k=k+1 ,goes to step(6).

Step(8): For the complete schedule π find $T_{max}(\pi)$; if $T_{max}(\pi)=T^*$ then $\pi_{min}=\pi$,goes to step(9);else if $E_{max}(\pi(1))=E^*$,interchange position $\pi(n)$ and $\pi(n-1)$ and find $T_{max}(\pi_{new})$,if $T_{max}(\pi_{new})=T^*$ then
$\pi_{min}=\pi_{new}$,goes to step(9); else chooses the schedule from π and π_{new} that has minimum T_{max} and called π_{min} .

Step(9): Choose the schedule from σ_{min} and π_{min} that satisfies (P5) (i.e. choose a schedule that has minimum E_{max} ,if there is a tie, choose that has minimum T_{max} ,and if a tie is still choose that has minimum $\sum_{i=1}^{n} C_i^2$).

Example(5): consider the problem (P5) with the following data.

i	1	2	3	4	5
P_i	4	6	2	1	5
d_i	4	6	8	3	6
s_i	0	0	6	2	1

$T^*=10$,$E^*=0$, $r_1=0,r_2=0,r_3=6,r_4=2,r_5=1$

j	r_j	C_{k-1}	j*
1	0	0	1
2	0	4	4
3	6	5	5
4	2	10	3
5	1	12	2

Hence $\sigma=(1,4,5,3,2)$

i	1	4	5	3	2
p_i	4	1	5	2	6
d_i	4	3	6	8	6
c_i	4	5	10	12	18
T_i	0	2	4	4	12
E_i	0	0	0	0	0

Hence Interchange position $\sigma_{new}=(1,4,5,2,3)$

i	1	4	5	2	3
p_i	4	1	5	6	2
d_i	4	3	6	6	8
c_i	4	5	10	16	18
T_i	0	2	4	10	10
E_i	0	0	0	0	0

Then $\sigma_{min}=\sigma_{new}$

Now,

j	r_j	C_{k-1}	j*
1	0	0	1
2	0	4	4
3	6	5	5
4	2	10	2
5	1	16	3

Hence π =(1,4,5,2,3) =σ_{min} .

Then the best schedule is (1,4,5,2,3) with **(E_{max} ,T_{max},** $\sum_{i=1}^{n} C_i^2$ **)** =(0,10,721).

5. Conclusions

For the multi-criteria scheduling problem 1//(f_{max} ,g_{max} ,$\sum h_i$) we proposed some algorithms and procedure to find exact and the best possible solution for the hierarchical case.

It is hoped that the contribution of this paper would provide an incentive increased research effort in this multi-criteria field especially three criteria .

An interesting future research topic would involve experimentation with the following machine scheduling problems 1//F(E_{max} ,T_{max} ,$\sum C_i^2$) ,and 1//F(E_{max} ,n_T ,$\sum C_i^2$) .

References:

[1] Divya Prakash ,” Bi-criteria Scheduling Problems on Parallel Machines”, M.Sc. thesis , Virginia Polytechnic Institute and State University (2007).

[2] Eren T., "A Multi-criteria Scheduling with Sequence-Dependent Setup Times" ,Applied Mathematical Sciences, 1, 58, 2883 – 2894(2007).

[3] Hoogeveen J.A, " Invited Review Multi-criteria scheduling ", European Journal of Operational Research 167 , 592–623(2005).

[4] Hoogeveen J.A , ” Minimizing maximum earliness and maximum lateness on a single machine “ ,CWI,BS-R9001 (1990).

[5] Hoogeveen J.A , ” Single machine scheduling to minimize a function of K maximum cost criteria “ ,CWI,BS-R9113 (1991).

[6] Hoogeveen J.A, ” Single machine scheduling to minimize a function of two or three maximum cost criteria “ , Journal of Algorithms, 21,415-433(1996).

[7] Hoogeveen, J.A., van de Velde, S.L., “ Minimizing total completion time and maximum cost simultaneously is solvable in polynomial time”, Operations Research Letters 17, 205–208(1995).

[8] Hoogeveen J.A ,Van de Valde ,” Polynomial time algorithms for single machine multi-criteria scheduling “ ,CWI,BS-R9008 (1990).

[9] Jackson J.R.,“ Scheduling a production line to minimize maximum tardiness”, Research Report 43, Management Sciences Research Project, UCLA(1955).

[10] Lawler E.L.,"Optimal sequencing of a single machine subject to precedence constraints", Management Science 19/5, 544-546(1973).

[11] Nelson R.T., Sarin R.K., and Daniels R.L., "Scheduling with multiple performance measures: The one-machine case", Management Science 32, 464-479(1986).

[12] Smith W.E. , "Various optimizers for single stage production", Naval Research Logistics Quarterly 3/1, 59-66(1956).