Minimizing total cost in proportionate flow shop with controllable processing times by polynomially solvable 0-1 unconstrained Quadratic Program

Byung Cheon Choi*, Sung Pil Hong**, Seung Han Lee***[1)]

* Senior Researcher, Seoul R&BD Program, u-Computing Innovation Center (polytime@hanmail.net)
** Professor, Department of Industrial Engineering, Seoul National University (sphong@snu.ac.kr)
*** Master Course, Department of Industrial Engineering, Seoul National University (shlee17@snu.ac.kr)

Abstract

We consider a proportionate flow shop problem with controllable processing times. The objective is to minimize the sum of total completion time and total compression cost, in which the cost function of compressing the processing times is non-decreasing concave. We show that the problem can be solved in polynomial time by reducing it to a polynomially solvable 0-1 unconstrained quadratic program.

1. Introduction

In a proportionate flow shop environment, the processing times of job j on all machine are equal to p_j (Ow 1985, Pinedo 2002). Shakhlevich et al.(1998) show that the problem of minimizing total weighted completion time in a proportionate flow shop is solvable in $O(n^2)$ time. The proportionate flow shop model cab be generalized to the problem with controllable processing times. By using additional resource such as manpower, fuels and so on, the processing times can be decreased.

Nowicki and Zdrzalka(1990) made a survey of single machine scheduling problems with controllable processing times up to 1990. Janiak(1987) considered single machine scheduling problems with a linear compression cost function. The objective was to minimize regular performance measures, e.g. the maximal job completion time(makespan), maximum lateness and so on, with a constraint on limited total compression amount. He proposed several algorithms that could solve the problems in polynomial time. In particular, many researchers have studied on a single machine scheduling problem with controllable processing times in which the objective was to minimize total weighted job completion time plus total linear cost function of compressing processing times (Hoogeven and Woeginner 2002, Janiak et al. 2005, Vickson 1980, Wan et al. 2001). The problem was initiated by Vickson(1980). He showed that the problem could be solved in polynomial time if all weights were identical. Nowicki and Zdrzalka(1988) considered a two-machine flow shop scheduling problem with controllable processing times in which the objective was to minimize total linear cost function of compressing processing times plus makespan. They showed that the problem is NP-hard. Cheng and Shakhlevich(1999) considered an m-machine proportionate flow shop with controllable processing times in which the cost function of compressing processing times was linear or quadratic. The objective was to construct the Pareto optimal set for minimizing makespan and total compression cost. They developed $O(nlogn)$ algorithm and $O(n^2)$ algorithm for two cases, respectively.

In this paper, we consider the proportionate flow shop with controllable processing times. The objective is to minimize the sum of the total completion times plus the total compression costs which depend on the decreasing concave function. The rest of this paper is organized as follows. In section 2, we present notations and problem definitions. We transform the first problem to a 0-1 quadratic programming (QP) and show that the 0-1 QP can be solved in polynomial time in section 3. Finally, we provide the summary and concluding remarks.

2. Notations and Problem Definition

The following notations and assumptions will be used throughout the paper :

Notations

- e_k : unit vector where the kth component is one and zero elsewhere ;
- p_j : initial processing time of job j ;
- $\sigma=(\sigma(1),...,\sigma(n))$: job sequence, where $\sigma(j)=k$ implies that job k is positioned jth in the sequence ;
- C_j : completion time of job j ;
- $x=(x_1,x_2,...,x_n)$: compression vector where x_j is the compression amount of job j, $0 \le x_j \le p_j$;
- $C_{\sigma(j)}(x)$: completion time of the jth job when all

1) The research was partially supported by the Second Stage of Brain Korea 21 Project in 2007.

the processing times are compressed by x ;
- $f_j(x_j)$: compression cost function of job j ;

Assumptions
- All jobs are available at time zero and preemption is not allowed.
- If job j is compressed by x_j, then all processing times of job j on all machines are compressed by x_j.

Our problem(Problem P1) is defined as follows :

Problem P1 In an n-job, m-machine proportionate flow shop with controllable processing times, determine a schedule (σ,x) that minimizes

$$K=\sum_{j=1}^{n} C_{\sigma(j)}(x)+\sum_{j=1}^{n} f_j(x_j),$$

where $f_j(x_j)$, j=1,2,...,n, is a non-decreasing concave function.

Since the above objective function is regular, there exists a permutation schedule that is optimal (Shakhlevich et al. 1998). Thus, we only consider a permutation schedule throughout the paper.

3. Polynomial Solvability of Problem P1

In this section, we first discuss some optimality conditions for Problem P1.

3.1 Optimality conditions for Problem P1

Proposition 1 *(Shakhlevich et al. 1998)* In an n-job, m-machine proportionate flow shop, for $k=1,...,n$

$$C_{\sigma(k)}(x)=\sum_{j=1}^{k-1}(p_{\sigma(j)}-x_{\sigma(j)})+m(p^{k_{\max}}-x^{k_{\max}}),$$

where $p^k_{\max}-x^k=\max_{j=1,...,k}\{p_{\sigma(j)}-x_{\sigma(j)}\}$.

Proposition 2 *(Panwalkar and Kahn 1976)* In an n-job, m-machine proportionate flow shop, the total job completion time is minimized by the shortest processing time (SPT) order.

By Propositions 1 and 2, K can be formulated as follows :

$$K=\sum_{k=1}^{n}\left(\sum_{j=1}^{k-1}(p_{\sigma(j)}-x_{\sigma(j)})+m(p_{\sigma(k)}-x_{\sigma(k)})\right)+\sum_{j=1}^{n} f_j(x_j),$$

where $p_{\sigma(1)}-x_{\sigma(1)}\le p_{\sigma(2)}-x_{\sigma(2)}\le\cdots\le p_{\sigma(n)}-x_{\sigma(n)}$.

Theorem 3 In Problem P1, there is an optimal schedule such that all processing times are totally compressed $(x_j=p_j)$ or uncompressed $(x_j=0)$, $1\le j\le n$.

Proof For clarity of notations, let T be the set of jobs whose processing times are totally compressed, U be the set of jobs whose processing times are uncompressed, and L be the set of jobs whose processing times are partially compressed. Let (σ,x,T,U,L) be a schedule.

Suppose that there is an optimal schedule $(\sigma^*,x^*,T^*,U^*,L^*)$ such that $J_l\in L^*$. Let $p^*_j=p_j-x^*_j$, $j=1,...,n$ and let

$Q_1=\{J_j|\ p^*_l\le p^*_j\le p_l,\ j\in U^*\cup L^*\setminus\{J_l\}\}$ and

$Q_2=\{J_j|\ p_l<p^{*_j},\ j\in U^*\cup L^*\setminus\{J_l\}\}$.

We can construct the new schedule $(\sigma^1,x^1,T^1,U^1,L^1)$ by letting $x^1_l=0$, $T^1=T^*$, $U^1=U^*\cup J_l$ and $L^1=L^*\setminus\{J_l\}$, and rearranging jobs in the SPT order. Also, the other schedule $(\sigma^2,x^2,T^2,U^2,L^2)$ can be constructed by letting $x^2_l=p_l$, $T^2=T^*\cup J_l$, $U^2=U^*$ and $L^2=L^*\setminus\{J_l\}$, and rearranging jobs in the SPT order. K_1 and K_2, the objective values of $(\sigma^1,x^1,T^1,U^1,L^1)$ and $(\sigma^2,x^2,T^2,U^2,L^2)$, respectively, can be calculated as follows:

$K_1=K^*-|Q_1|p^*_l+\sum_{j\in Q_1}p^*_j+mx^*_l+|Q_2|x^*_l-f_l(x^*_l)$ and

$K_2=K^*-(|Q_1|+|Q_2|)p^*_l-mp^*_l+f_l(p_l)-f_l(x^*_l)$.

Since K^* is optimal, the following inequalities should hold.

$-|Q_1|p^*_l+\sum_{j\in Q_1}p^*_j+mx^*_l+|Q_2|x^*_l-f_l(x^*_l)\ge 0$ and

$-(|Q_1|+|Q_2|)p^*_l-mp^*_l+f_l(p_l)-f_l(x^*_l)\ge 0$.

Since $-|Q_1|p^*_l+\sum_{j\in Q_1}p^*_j\le|Q_1|x^*_l$, the above equations can be rewritten as follows :

$$\frac{f_l(x^*_l)-f_l(0)}{x^*_l}\le m+|Q_1|+|Q_2|\le\frac{f_l(p_l)-f_l(x^*_l)}{p^*_l}.$$

By the above equation and the concavity of f_l, $K_1=K_2=K^*$. There is an other optimal schedule $(\overline{\sigma},\overline{x},\overline{T},\overline{U},\overline{L})$ such that $|\overline{L}|=|L^*|-1$. By continuing to apply the above procedure until $L=\varnothing$, we can construct an optimal schedule such that all processing times are totally compressed or uncompressed. ■

3.2 Reduction to 0-1 quadratic programming

Assume that $p_1\le\cdots\le p_n$.

0-1 Quadratic Programming(QP) Given an $n\times n$ matrix Q, determine a vector $y\in R^n$ such that

$Min\ y^TQy$ s.t. $y_j\in\{0,1\}$, $j=1,...,n$,

By Theorem 3, our problem can be transformed as follows :

$$Min\ z(y)=\sum_{k=1}^{n}\left(\sum_{j=1}^{k-1}p_jy_j+mp_k\right)y_k+\sum_{j=1}^{n}\overline{p}_j(1-y_j)$$

s.t. $y_j\in\{0,1\}$, $j=1,...,n$,

where $\overline{p}_j=f_j(p_j)$, $j=1,...,n$. Let $y_j=1$ if $x_j=0$, and $y_j=0$ if $x_j=p_j$. Since $y^2_j=y_j$,

$$z(y)=\sum_{k=1}^{n}\sum_{j=1}^{k-1}p_jy_jy_k+\sum_{j=1}^{n}(mp_j-\overline{p}_j)y^2_j+\sum_{j=1}^{n}\overline{p}_j=y^TQy+\sum_{j=1}^{n}\overline{p}_j$$

where $Q=(q_{ij})_{i=1,...,n,\ j=1,...,n}$ is defined as follow :

$$q_{ij}=\begin{cases}p_i & if\ i<j\\ mp_i-\overline{p}_i & if\ i=j\\ 0 & othewise\end{cases}$$

completes proof. ■

3.3 *Efficient Algorithm for Problem P1*

Definition Let a row-ordered matrix(ROM) be defined as $Q=(q_{ij})_{i=1,...,n,\ j=1,...,n}$ such that $q_{ij}=0$ for $i>j$, $q_{ij}\le q_{i+1,j}$, $i=1,...n-1$, $j=1,...n-1$ and $q_{ij}=k_i$, $i=1,...n$, $j=i+1,...,n$, where $k_i\ge 0$, $i=1,...n$.

Lemma 4 In QP with ROM, there is an optimal solution with $y_l=1$ such that $q_{ll}<0$ and $q_{ll}\le q_{jj}$, $\forall j\ne l$.

Proof Let l be the index such that $q_{ll}<0$ and $q_{ll}\le q_{jj}$, $\forall j\ne l$. Suppose that there exists an optimal solution y^* such that $y_l^*=0$. Let $r=\max\{j|y_j^*=1\}$. We can construct a new solution $\hat{y}$ by letting $\hat{y}_l=1$, $\hat{y}_r=0$ and $\hat{y}_j=y_j^*$, $\forall j\ne l,r$. Since $(\hat{y}-e_l)^TQ(\hat{y}-e_l)=(y^*-e_r)^TQ(y^*-e_r)$,

$$\hat{y}^TQ\hat{y}=(y^*)^TQy^*-e_r^T(Q^T+Q)y^*+e_l^T(Q^T+Q)\hat{y}+q_{rr}-q_{ll}.$$

Let $S=\{1,...,n\}\setminus\{l,r\}$. Let $(Q^T+Q)_S$ be submatrix constructed by column vectors $(Q^T+Q)_i$, $i\in S$. Let $(Q^T+Q)_l$ and $(Q^T+Q)_r$ be the lth column and rth column of (Q^T+Q), respectively. Let y_S be subvector constructed by y_i, $i\in S$. Clearly, $y_S^*=\hat{y}_S$.

Since $y_r^*=\hat{y}_l=1$, $y_l^*=\hat{y}_r=0$, $e_r^T(Q^T+Q)_r y_r^*=2q_{rr}$ and $e_l^T(Q^T+Q)_l\hat{y}_l=2q_{ll}$,

$$e_r^T(Q^T+Q)y^*=e_r^T(Q^T+Q)_S y_S^*+2q_{rr} \text{ and}$$
$$e_l^T(Q^T+Q)\hat{y}.=e_l^T(Q^T+Q)_S\hat{y}_S+2q_{ll}.$$

Thus,

$$\hat{y}^TQ\hat{y}=(y^*)^TQy^*-e_r^T(Q^T+Q)_S y_S^*-q_{rr}+e_l^T(Q^T+Q)_S\hat{y}_S+q_{ll}.$$

Claim 1 For $k=1,...,n$,

$$(Q^T+Q)_{1k}\le...\le(Q^T+Q)_{k-1,k}\le(Q^T+Q)_{k+1,k}=...=(Q^T+Q)_{nk},$$

where let $(Q^T+Q)_{ij}$ be the (i,j)th element of the matrix (Q^T+Q).

Proof of Claim 1 It immediately holds from the definition of ROM. ■

Claim 2 $e_r^T(Q^T+Q)_S y_S^*\ge e_l^T(Q^T+Q)_S\hat{y}_S$.

Proof of Claim 2 Two cases are considered.

Case 1 : $r>1$

Since $e_r^T(Q^T+Q)_j\ge e_l^T(Q^T+Q)_j$, $\forall j\in S$ by Claim 1 and $y_S^*=\hat{y}_S$, Claim 2 holds.

Case 2 : $r<1$

Clearly, $(Q^T+Q)_S y_S^*=(Q^T+Q)_S\hat{y}_S$. Let $((Q^T+Q)_S y_S^*)_j$ be the jth element of $(Q^T+Q)_S y_S^*$. Since $r=\max\{j\mid y_j^*=1\}$,

$$((Q^T+Q)_S y_S^*)_j=((Q^T+Q)_S\hat{y}_S)_{j+1},\ j=r,...,n-1.$$

Thus, $e_r^T(Q^T+Q)_S y_S^*=e_l^T(Q^T+Q)_S\hat{y}_S$ ■

By Claim 2 and $q_{rr}\ge q_{ll}$, $\hat{y}^TQ\hat{y}\le(y^*)^TQy^*$. This completes proof. ■

If $y_l=1$ in the optimal solution, then QP with $n\times n$ ROM can be reduced to QP with $\hat{Q}\in R^{(n-1)\times(n-1)}$, constructed by letting

$$\hat{q}_{jj}=\begin{cases} q_{jj}+q_{jl} & j=1,...,l-1 \\ q_{ll} & j=l \\ q_{jj}+q_{lj} & j=l+1,...,n \end{cases}$$

$\hat{q}_{ij}=q_{ij}$, $\forall i\ne j$ and eliminating the lth row and the lth column from the $n\times n$ matrix Q. Clearly, $\hat{Q}$ is a ROM.

The algorithm for QP with ROM is proposed below :

Algorithm ROM

Step 0 : $k\leftarrow 0$, $Q^{(k)}\leftarrow Q$, $I=\varnothing$, $J=\{1,...,n\}$.

Step 1 : Select an index l such that $q_{ll}^k\le q_{jj}^k$, $\forall j\ne l$ and $q_{ll}^{(k)}<0$. If there exists no such index, go to Step 2. Otherwise, go to Step 3.

Step 2 : Let $y_j=0$, $j\in J$ and $y_i=1$, $i\in I$. STOP.

Step 3 : Find the index l_k corresponding to the index l from Q. $I=I\cup\{l_k\}$ and $J=J\setminus\{l_k\}$.

Step 4 : Construct an $(n-k-1)\times(n-k-1)$ matrix $Q^{(k+1)}$ by letting

$$q_{jj}^{(k+1)}=\begin{cases} q_{jj}^{(k)}+q_{jl}^{(k)} & j=1,...,l-1 \\ q_{ll}^{(k)} & j=1 \\ q_{jj}^{(k)}+q_{lj}^{(k)} & j=l+1,...,n-k \end{cases}$$

$q_{ij}^{(k+1)}=q_{ij}^{(k)}$, $\forall i\ne j$ and eliminating the lth row and the lth column from $Q^{(k)}$.

Step 5 : $k\leftarrow k+1$ and go to Step 1.

The time complexity of Algorithm ROM is $O(n^2)$.

Lemma 5 QP with ROM can be solved in polynomial time by Algorithm ROM.

Proof If $Q^{(k)}$ is a ROM, then $Q^{(k+1)}$ is a ROM. By Lemma 4, Algorithm ROM produces an optimal solution. ■

Theorem 6 Problem P1 can be solved in polynomial time.

Proof Since Problem P1 can be reduced to the 0-1 QP with ROM in polynomial time, it immediately follows from Lemma 5. ■

4. Summary

We show the polynomiality of a proportionate flow shop with controllable processing times by reducing it to 0-1 QP which is solved in polynomial time.

References

S.S. Panwalkar, A.W. Khan (1956), An ordered flow-shop sequencing problem with mean completion time criterion, *International Journal of Production Research*, **14**, 631-635.

R.G. Vickson (1980), Choosing the job sequence and processing times to minimize total processing plus flow cost on a single machine, *Operations Research*, **28**, 1155-1167.

P.S. Ow (1985), Focused Scheduling in Proportionate Flowshops, *Management Science, 31*, 852-869.

A. Janiak (1987), One-machine scheduling with allocation of continuously-divisible resource and with no precedence constraints, *Kybenetika*, **23**, 289-293.

E. Nowicki, S. Zdrzalka (1988), A two-machine flow shop scheduling problem with controllable job processing times, *European Journal of Operational Research*, **34**, 208-220.

E. Nowicki, S. Zdrzalka (1990), A survey of results for sequencing problems with controllable processing times, *Discrete Applied Mathematics*, **26**, 271-287.

N. Shakhlevich, H. Hoogeveen, M. Pinedo (1998), Minimizing total weighted completion time in a proportionate flow shop, *Journal of Scheduling, 1*, 157-168.

T.C.E. Cheng, N. Shakhlevich (1999), Proportionate flow shop with controllable processing times, *Journal of Scheduling 2*, 253-265.

G. Wan, B.P.C. Yen, C.L. Li (2001), Single machine scheduling to minimize total compression plus weighted flow cost is NP-hard, *Information Processing Letters*, **29**, 41-47.

H. Hoogeveen, G.J. Woeginner (2002), Some comments on sequencing with controllable processing times, *Computing*, **68**, 181-192.

M. Pinedo (2002), Scheduling : Theory, Algorithms and Systems (2nd), Prentice Hall.

A. Janiak, M.Y. Kovalyov, W. Kubiak, F. Werner. (2005), Positive half-products and scheduling with controllable processing times, *European Journal of Operational Research*, **165**, 416-422.