8. Shop problems

8.1 Two-machine flowshop problems

In this section we are interested in multicriteria flowshop scheduling problems with two machines. Each job J_i is processed on the machine M_1 for a duration $p_{i,1}$, then on the machine M_2 for a duration $p_{i,2}$. In this context, the multicriteria scheduling problem which is addressed the most in the literature involves the minimisation of the criteria $\overline{C}$ and C_{max}. Different scheduling problems are derived according to the form of the considered objective function.

8.1.1 The $F2|prmu|Lex(C_{max}, \overline{C})$ problem

[Rajendran, 1992]

This problem is strongly $\mathcal{NP}$-hard ([Chen and Bulfin, 1994]) and Rajendran proposes two heuristics and one exact algorithm to solve it. These two heuristics, denoted by HCR1 and HCR2, use the algorithm ESJ1 of [Johnson, 1954] to obtain an optimal initial sequence for the criterion C_{max}. Next they perform adjacent job permutations which are chosen by two indicators D_i and D'_i, generally defined for a sequence S and a position r by:

$$D_{S[r]} = p_{S[r],1} + p_{S[r],2} - p_{S[r+1],1} - p_{S[r+1],2}$$
$$D'_{S[r]} = 2p_{S[r],1} + p_{S[r],2} - 2p_{S[r+1],1} - p_{S[r+1],2}$$

The aim of these permutations is to reduce the value of the criterion $\overline{C}$ of the schedule S. The next job to be permutated is determined according to these indicators. The heuristic HCR1 uses the indicator D_i and ties between several jobs are broken using D'_i. The heuristic HCR2 uses the indicator D'_i and ties between several jobs are broken using D_i. The heuristic HCR1 is presented in figure 8.1 and the heuristic HCR2 is similar. We frequently refer in the literature to the best schedule calculated by HCR1 and HCR2.

Example.
We consider a problem for which $n = 10$ and we apply the heuristic HCR1.

ALGORITHM HCR1

/* T is the set of the n jobs to schedule */
/* ESJ1 is the algorithm of [Johnson, 1954] for the $F2|prmu|C_{max}$ problem */

Step 1: Apply algorithm ESJ1 to obtain the schedule S;

Step 2: Let k be the index in S such that

$$\sum_{r=1}^{k} p_{S[r],1} - \sum_{r=1}^{k-1} p_{S[r],2} = \max_{u=1,\dots,n} \left(\sum_{r=1}^{u} p_{S[r],1} - \sum_{r=1}^{u-1} p_{S[r],2}\right);$$

For $r = 1$ to n Do
  If $(r = k-1)$ or $(r = k)$ or $(r = n)$ Then
    $D_{S[r]} = -1$;
    $D'_{S[r]} = -1$;
  Else
    $D_{S[r]} = p_{S[r],1} + p_{S[r],2} - p_{S[r+1],1} - p_{S[r+1],2}$;
    $D'_{S[r]} = 2p_{S[r],1} + p_{S[r],2} - 2p_{S[r+1],1} - p_{S[r+1],2}$;
  End If;
End For;
$L = \{r / D_{S[r]} \geq 0\}$;
Sort L by decreasing order of values D_i (break ties by choosing the job with the greatest value D'_i);

Step 3: While $(L \neq \emptyset)$ Do
  $r = L[1]$;
  $S' = S$ after permutation of jobs in positions r and $(r+1)$;
  If $\left((C_{max}(S') = C_{max}(S))\text{ and }(\overline{C}(S') < \overline{C}(S))\right)$ Then
    $S = S'$;
    Goto Step 2;
  End If;
  $L = L - \{r\}$;
End While;

Step 4: Print $S, C_{max}(S)$ and $\overline{C}(S)$;

[Rajendran, 1992]

Fig. 8.1. An heuristic algorithm for the $F2|prmu|Lex(C_{max}, \overline{C})$ problem

i	1	2	3	4	5	6	7	8	9	10
$p_{i,1}$	5	6	7	10	10	8	13	7	10	2
$p_{i,2}$	10	8	11	10	9	7	5	4	2	1

(i) The schedule obtained by the algorithm $ESJ1$ is $S = (J_1, J_2, J_3, J_4, J_5, J_6, J_7, J_8, J_9, J_{10})$ and we have $C^*_{max} = C_{max}(S) = 79$ and $\overline{C}(S) = 521$.
$k = 10$.
(ii) Calculation of the initial indicators

r	1	2	3	4	5	6	7	8	9	10
$D_{S[r]}$	1	−4	−2	1	4	−3	7	−1	−1	−1
$D'_{S[r]}$	0	−5	−5	1	6	−8	13	−4	−1	−1

$L = \{J_7, J_5, J_4, J_1\}$.
(iii) $S' = (J_1, J_2, J_3, J_4, J_5, J_6, J_8, J_7, J_9, J_{10})$, $C_{max}(S') = 79 = C^*_{max}$ and $\overline{C}(S') = 521$.
$L = \{J_5, J_4, J_1\}$.