KM 6(1982)
pag. 95-117

CLUSTERING OF PRODUCTS IN THE SINGLE-MACHINE LOT-SIZE SCHEDULING PROBLEM

by

Maria Starmans and Jaap Wessels
University of Technology
Department of Mathematics and Computing Science
Eindhoven, The Netherlands

Abstract. For the single-machine multi-product lot-size scheduling problem very good solutions can be obtained in a very simple way, if the products belong to two homogeneous groups. In this paper it is investigated how, in the general case, the products can be clustered in two groups and how good the solutions are if these groups are treated as homogeneous.

0. Introduction

The problem of planning the production of several products with constant demands on a single machine has attracted a lot of attention over the years, see e.g. the review paper by Elmaghraby [4]. One of the attractive aspects of the problem is the combination of a continuous optimization problem (the lot sizes) and a discrete optimization problem (the scheduling). Although the problem is simply stated, its solution is rather complicated as can be seen for instance in Elmaghraby's paper or in more recent papers as the one by Axsäter [1]. However, if the products belong to two homogeneous groups, then it is very simple to construct rather good solutions, see Hendriks/Wessels [5] and Wessels/Thijssen [7]. We say that a set of products forms a homogeneous group, if all products have the same parameters (demand rate, inventory rate etc.). In fact, also for the case of three or more homogeneous groups of products the problem simplifies, but the situation of two groups is parti-

Key words: multi-products one-machine scheduling, repetitive schemes.

cularly interesting, since it can be analyzed completely.

Knowing this, a sensible approach for the general problem is to design some sort of clustering procedure which separates the products into two groups which are treated as homogeneous groups. Such a clustering procedure and the quality of the resulting production schemes form the subject of this paper. Of course, it is not to be expected that all problems can be solved satisfactorily in this way. However, one may hope that the approach works for a large class of problems. In fact, if the approach leads to a solution with average costs differing much from a theoretical underbound, then it is still sufficiently early to try a more complicated solution technique, since our approach leads to a technique which is executed without much work.

In Section 1 we will present the model and some preliminaries. The clustering procedure is developed in Section 2. The remaining sections are devoted to the analysis of the quality of the production schemes as designed via our approach.

1. The model and some preliminaries

N products, indexed by $i = 1,\ldots,N$, have to be produced on one machine, which can only be manufacturing one product at each time instant. The purpose is to find a production schedule with minimal average costs per time unit over an infinite time horizon. This production schedule has to take into account the following features for product $i \in L = \{1,\ldots,N\}$:

- a constant and deterministic demand rate d_i;
- a constant and deterministic production rate p_i for the time intervals in which product i is manufactured;

- a required service level β_i, i.e. at least during a fraction β_i of time the stock of product i must be nonnegative;
- fixed starting costs F_i for each time interval the machine is manufacturing product i;
- holding cost rate h_i.

In the analysis the following quantities play a role:

- $\rho_i = d_i/\rho_i$, the utilization rate for product i;
- $\rho = \sum_{i=1}^{N} \rho_i$, the machine utilization rate, which is supposed to be less than 1;
- $\alpha_i = \frac{1}{2}(1 - \rho_i)h_i d_i \beta_i^2$; for a production cycle for product i of T time units, with service level β_i, the stock-holding costs per time unit are $\alpha_i T$.

In fact we will only consider the holding costs per time unit $\alpha_i T$, hence also other models with average holding costs proportional to the product cycle time are covered.

Set-up times will not be included, since difficulties caused by the set-up times can always be solved afterwards at a minor cost increase (see [7]).

For each product it is clear that it would be best to produce it with a cycle-time T_i which minimizes

$$\frac{F_i}{T} + \alpha_i T \, .$$

Hence

$$T_i = \sqrt{\frac{F_i}{\alpha_i}} \text{ with average costs } K_i = 2\sqrt{F_i\alpha_i} .$$

This is the classical Camp or Wilson rule.

The ideal production schedule for the range of N products would therefore require an average costs of

$$K = \sum_{i\in L} K_i = 2 \sum_{i\in L} \sqrt{F_i\alpha_i} .$$

If all the T_i would be equal, then it is simple to realize a production schedule with average costs K.

If the T_i are not equal a first attempt to find a good production schedule would be to use the same cycle time T_L for all products. The best choice for T_L would be the value of T which minimizes

$$\sum_{i\in L} \frac{F_i}{T} + \alpha_i T \qquad \text{or} \qquad \frac{F_L}{T} + \alpha_L T ,$$

with

$$F_L = \sum_{i\in L} F_i , \alpha_L = \sum_{i\in L} \alpha_i .$$

So

$$T_L = \sqrt{\frac{F_L}{\alpha_L}} \text{ with average costs } K_L = 2\sqrt{F_L\alpha_L} .$$

This is the well-known rotation scheme.

It will be clear that this rotational production scheme is not always very good. However, if the T_i lie not far apart, then the scheme works quite well, since the costs for a product are not very sensitive to small deviations of the cycle time. In fact the average costs for product i with a cycle time γT_i are

$$\frac{F_i}{\gamma T_i} + \alpha_i \gamma T_i = \frac{1}{\gamma}\sqrt{F_i \alpha_i} + \gamma\sqrt{F_i \alpha_i} = \frac{1}{2}(\gamma + \frac{1}{\gamma})K_i \ .$$

In this formula $\frac{1}{2}(\gamma + \frac{1}{\gamma})$ changes only slowly for γ around 1 .
Therefore we try a slightly more sophisticated type of scheme which has been called repetitive production scheme [5] . For this scheme, two clusters of products are determined and both clusters get their own cycle times with the only provision that one is a multiple of the other.

Suppose for a moment that we have split up the set $L = \{1,2,\dots,N\}$ into two subsets I and J. Then the problem remains to determine the cycle times T and kT for I and J respectively (here we have supposed, without loss of generality, that J deserves to have the larger cycle time). For given T and k the average costs are

$$\frac{F_I}{T} + \alpha_I T + \frac{F_J}{kT} + \alpha_I kT = \frac{F_I + \frac{F_J}{k}}{T} + (\alpha_I + k\alpha_J)T$$

where

$$F_I = \sum_{i \in I} F_i \quad \text{etc.}$$

For fixed k we find - as before - the optimal value of T:

$$T_{IJ}(k) = \sqrt{\frac{F_I + \frac{F_J}{k}}{\alpha_I + k\alpha_J}} \text{ with average costs } K_{IJ}(k) = 2\sqrt{(F_I + \frac{F_J}{k})(\alpha_I + k\alpha_J)}$$

$$K_{IJ}(k) = \quad 2k^{-\frac{1}{2}}\sqrt{[(k-1)F_I + F_L][\alpha_L + (k-1)\alpha_J]}$$

$$= \quad K_L k^{-\frac{1}{2}}\sqrt{[(k-1)f_I + 1][(k-1)A_J + 1]}\ ,$$

where

$$f_I = \frac{F_I}{F_L} \quad , \quad A_J = \frac{\alpha_J}{\alpha_L}\ .$$

The optimal k, for fixed clustering, can be determined by considering the inequality

$$K_{IJ}(k) \le K_{IJ}(k+1)$$

or

$$\frac{1}{k}\,[(k-1)f_I + 1][(k-1)A_J + 1] \le \frac{1}{k+1}\,[kf_I + 1][kA_J + 1]\ .$$

This inequality is equivalent to

$$k(k+1) \ge \frac{(1-f_I)(1-A_J)}{f_I A_J} = \frac{f_J A_I}{f_I A_J} = \left(\frac{T_J}{T_I}\right)^2\ .$$

So, k is optimal if it satisfies

$$k(k-1) \le \left(\frac{T_J}{T_I}\right)^2 < k(k+1) \ .$$

Roughly the optimal k is the ratio of T_J and T_I, the optimal rotation cycle times for the sets of products J and I respectively.

Before going into the details of choosing the clusters, some remarks will be made about the realization of a production scheme for two clusters with cycle times T and kT.
The cluster J has to be split up further into k subsets and then the production is scheduled in such a way that in each cycle of length T the whole cluster I is produced together with one of the subsets of J. See Fig. 1 for a 2-repetitive production scheme for the case $I = \{1,2,3\}$, $J = \{4,5,6,7\}$.

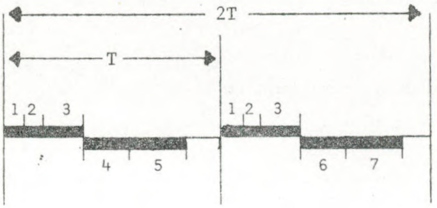

Fig. 1: a 2-repetitive scheme for 7 products

The only problem might be that the subsets don't require equal production times. This might have as consequence that for some of the cycles T there is not sufficient time to produce the set of products I and the appropriate subset of J, whereas other cycles T have spare production time. Of course the first objective when splitting J in k subsets should be to determine the subsets in such a way that they require nearly equal production times. Particularly in cases where the utilization rate is very close to 1, this may not be sufficient. If there remains a fitting problem, then that can be solved by minor changes in the scheme, like bringing one product to the other cluster or adapting part of the cycle time (compare [5]). Since these types of changes are simple to construct and usually only require a very small extra cost, this point will not be worked out here in detail. We will suppose further that any relevant k-repetitive scheme can be realized for any relevant clustering.

2. The clustering procedure

In the preceding section we have seen how an optimal repetitive scheme is determined for a given clustering I,J. Now the only problem remaining is to find a good - preferably the best - clustering.

The natural procedure would be to order the products according to their camp cycle times T_i and to determine a clustering by a caesura. Regrettably, the average costs do not behave nice as a function of the caesura (neither for fixed k, nor for optimal k at given caesura). It is even not necessary that such a type of clustering is optimal as may be shown by a counter example. For the case with equal fixed costs for all products, it can easily be proved that making a caesura in the ordered set of α's gives an optimal clustering. Even in that case the average costs for fixed or for optimal k do not depend necessarily in a unimodal way on the caesura. However, the computations are rather simple, hence by varying the caesura in the set of camp cycle times one obtains an efficient procedure for finding a relatively good clustering:

1. Order the products in such a way that $\frac{f_\ell}{A_\ell} \le \frac{f_{\ell+1}}{S_{\ell+1}}$;

2. Compute for each clustering $I = \{1,\ldots,\ell\}$, $J = \{\ell + 1,\ldots,N\}$ the value of k with

$$(k - 1)k \le \frac{f_J A_I}{f_I A_J} < k(k + 1)$$

and for this value of k

$$C_\ell = \frac{1}{k}\,[(k - 1)f_I + 1][(k - 1)A_J + 1] \;;$$

3. Choose an ℓ which minimizes $\{C_\ell, \ell = 1,\ldots,N - 1\}$;

Choose the I,J belonging to the minimizing ℓ as clustering.

Summarizing: the clustering obtained in this way is optimal in the case of equal set-up costs, however, in the general case it may be suboptimal.

The following approach is even simpler. In the preceding procedure we try to minimize

$$\frac{1}{k}\,[(k - 1)f_I + 1][(k - 1)A_J + 1]$$

over k and over the partitioning I,J. Or, we try to minimize

$$\frac{1}{k}\,[(k - 1)^2 f_I A_J + (k - 1)(f_I + A_J) + 1] \;.$$

At least for relatively small values of k, the second term is the most important (f_I,A_J are at most 1). Therefore, a sensible partitioning might be obtained by minimizing $f_I + A_J$. This idea leads to the following procedure.

1. the products with $f_i \le A_i$ form set I;
2. the products with $f_j > A_j$ form set J;
3. choose the value of k such that

$$(k - 1)k \le \frac{f_J A_I}{f_I A_J} < k(k + 1) .$$

So, in the latter procedure the caesura in the set L is determined by the value 1 for f_i/A_i, i.e. I contains the products with $f_i/A_i \le 1$ and J those with $f_i/A_i > 1$.

The last procedure is very simple to execute and therefore it is always worthwhile to try it before embarking on more elaborate procedures. If the value of $K_{IJ}(k)$ which follows from this or any procedure is only slightly higher than K, then it is not worthwhile to investigate further. If it is much higher, then the clustering often provides hints for improvements.

In the subsequent sections the last and simplest procedure will be evaluated in different ways. One should keep in mind, that the first procedure is at least as good and often better and one should keep in mind that one may not expect that partitioning in two pseudo-homogeneous groups is efficient for all cases.

3. Worst-case analysis

Since we are not able to compare the effectiveness of optimal schemes and schemes as computed with our procedure, we will replace in the comparison the optimal average costs by K, a lower bound.

Therefore, we are interested in

$$K^{-1}K_L k^{-\frac{1}{2}} \left[(k-1)f_I + 1\right]^{\frac{1}{2}}\left[(k-1)A_J + 1\right]^{\frac{1}{2}} =$$

$$= \Big[\sum_{\ell\in L} \sqrt{f_\ell A_\ell}\Big]^{-1} k^{-\frac{1}{2}} \left[(k-1)f_I + 1\right]^{\frac{1}{2}} \left[(k-1)A_J + 1\right]^{\frac{1}{2}}$$

with $f_\ell, A_\ell \geq 0$, $\sum_{\ell\in L} f_\ell = 1$, $\sum_{\ell\in L} A_\ell = 1$ and k, I, J satisfying

$$i \in I \Rightarrow f_i \leq A_i \ ,$$
$$j \in J \Rightarrow f_j \geq A_j \ , \quad I \cap J = \emptyset, \quad I \cup J = L \, .$$
$$(k-1)k \leq \frac{f_J A_I}{f_I A_J} < k(k+1) \, .$$

Note that for reasons of symmetry products with $f_\ell = A_\ell$ may be put either in I or in J.

After the transformation

$$x_\ell = \sqrt{f_\ell} \ , \ y_\ell = \sqrt{A_\ell} \ ,$$

we are interested in the behaviour of

$$(*) \qquad [\sum_{\ell\in L} x_\ell y_\ell]^{-1} k^{-\frac{1}{2}} [(k-1)\sum_{i\in I} x_i^2 + 1]^{\frac{1}{2}} [(k-1)\sum_{j\in J} y_j^2 + 1]^{\frac{1}{2}}$$

$$A \left\{ \begin{array}{l} \text{with} \\ x_\ell, y_\ell \geq 0, \quad \sum_{\ell\in L} x_\ell^2 = \sum_{\ell\in L} y_\ell^2 = 1 \quad \text{and } k, I, J \\ \text{satisfying} \\ I = \{i \mid x_i \leq y_i\} \\ J = \{j \mid x_j \geq y_j\}, \quad I \cap J = \emptyset, \quad I \cup J = L \\ (k-1)k \leq \dfrac{\sum_{j\in J} x_j^2 \sum_{i\in I} y_i^2}{\sum_{i\in I} x_i^2 \sum_{j\in J} y_j^2} < k(k+1) . \end{array} \right.$$

This form can easily be maximized as a function of the parameters if the conditions are further strengthened by fixing

$$t = \sum_{j\in J} y_j^2 \quad \text{and} \quad r = \frac{\sum_{j\in J} x_j^2 \sum_{i\in I} y_j^2}{\sum_{i\in I} x_i^2 \sum_{j\in J} y_j^2} .$$

This leads to the maximization problem

$$\max_{\{x_\ell, y_\ell\}} [\sum_{\ell\in L} x_\ell y_\ell]^{-1} k^{-\frac{1}{2}} [\frac{(k-1)(1-t)}{1+t(r-1)} + 1]^{\frac{1}{2}} [(k-1)t + 1]^{\frac{1}{2}}$$

with $\quad x_\ell, y_\ell \geq 0, \quad \sum_{\ell\in L} x_\ell^2 = \sum_{\ell\in L} y_\ell^2 = 1,$

$$\sum_{j\in J} y_j^2 = t, \quad \sum_{i\in I} x_i^2 = \frac{1-t}{1+t(r-1)}$$

and k,I,J satisfying

$$i \in I \Rightarrow x_i \leq y_i, \quad j \in J \Rightarrow x_j \geq y_j, \quad I \cap J = \emptyset, \quad I \cup J = L$$

$$(k - 1)k \leq r < k(k + 1) .$$

For fixed t,r, this maximization requires the minimization of the innerproduct of two unit vectors with given Euclidean lengths of the parts of these vectors satisfying $x_i \leq y_i$ and $x_j \geq y_j$ respectively. This is a simple geometric problem. The solution runs as follows:

For every allowed set of x_ℓ, y_ℓ one has

$$\sum_{\ell \in L} x_\ell y_\ell = \sum_{i \in I} x_i y_i + \sum_{j \in J} x_j y_j \geq \sum_{i \in I} x_i^2 + \sum_{j \in J} y_j^2 =$$

$$\sum_{\ell \in L} x_\ell y_\ell \geq \frac{1 - t}{1 + t(r - 1)} + t .$$

Indeed, this lower bound can be attained (if $N \geq 4$), viz. choose

$$x_1^2 = y_1^2 = \frac{1 - t}{1 + t(r - 1)}, \qquad 1 \in I$$

$$x_N^2 = y_N^2 = t \qquad\qquad , \qquad N \in J$$

$$x_i = 0, \ y_j = 0 \qquad \text{for the other} \qquad i \in I, j \in J .$$

So, the maximum for fixed t,r is

$$\left[\frac{1 - t}{1 + t(r - 1)} + t\right]^{-1} k^{-\frac{1}{2}}\left[\frac{(k - 1)(1 - t)}{1 + t(r - 1)} + 1\right]^{\frac{1}{2}} \left[(k - 1)t + 1\right]^{\frac{1}{2}}$$

$$= \left[1 + t^2(r - 1)\right]^{-1} \left[1 + t(\frac{r}{k} - 1)\right]^{\frac{1}{2}}\left[1 + t(k - 1)\right]^{\frac{1}{2}} \left[1 + t(r - 1)\right]^{-1}$$

with k satisfying $(k - 1)k \leq r < k(k + 1)$.

This form can easily be maximized numerically over $t \in [0,1]$ for fixed r. The result is depicted in Fig. 2 as a function of $r^{\frac{1}{2}}$ for the relevant values of r, i.e. $r \geq 1$.

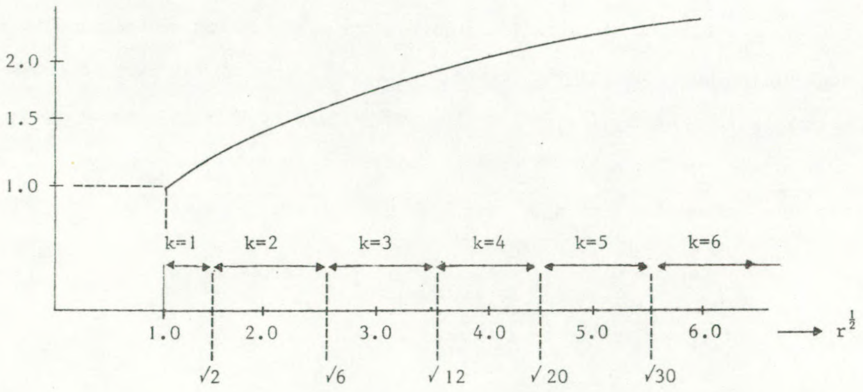

Fig. 2: worst case behaviour of the k-repetitive scheme obtained by a simple clustering procedure depicted as function of $r^{\frac{1}{2}}$, i.e. the ratio of the optimal cycle times of both clusters. The figure gives the worst ratio of the costs of the repetitive scheme (obtained according to the procedure) and K (the sum of the costs for individually optimal cycle times).

Note that the comparison in Fig. 2 takes place with respect to K which is only a lower bound of the real optimum.

From the construction of the worst cases, it is clear that these worst cases are really exceptional. So it might still be true that in the majority of the cases the procedure has a relatively good performance. This point will be investigated from different points of view in subsequent sections. Also will be shown, that in the extreme cases a substantial improvement can be obtained by a simple change of the production scheme.

4. Worst case analysis for the equal fixed costs case

In many practical situations, the fixed costs for the different products are of the same order of magnitude. In the extreme cases of the preceding section, however, some of the fixed costs have to be zero. Therefore, we will investigate in this section the quality of the performance for situations where all fixed costs are equal.

This implies that we perform the same maximization as in Section 3, but now under the extra constraint that $f_i = x_i^2 = \frac{1}{N}$ for $i = 1,\dots,N$.

So the function (*) and de conditions A are replaced by (**) and B:

$$(**) \qquad [\sum_{\ell\in L} y_\ell]^{-1} k^{-\frac{1}{2}} [(k-1)(N-n)+N]^{\frac{1}{2}} [(k-1)\sum_{j\in J} y_j^2 + 1]^{\frac{1}{2}}$$

$$B \left\{ \begin{array}{l} \text{with} \\ y_\ell \geq 0, \quad \sum_{\ell\in L} y_\ell^2 = 1 \quad \text{and} \quad n,k,J \\ \text{satisfying} \\ J = \{j \mid y_j \leq N^{-\frac{1}{2}}\}, \quad \# J = n \\ (k-1)k \leq \frac{n}{N-n} \dfrac{1-\sum_{j\in J} y_j^2}{\sum_{j\in J} y_j^2} < k(k+1) . \end{array} \right.$$

Again - as in Section 3 - we introduce the parameter r, such that $r^{\frac{1}{2}}$ gives the ratio between the ideal cycle times for the groups

$$r = \frac{n}{N-n} \frac{1-\sum_{j\in J} y_j^2}{\sum_{j\in J} y_j^2} .$$

A good idea of the behaviour of (**) under B may be obtained by maximizing (worst case) (**) under B for fixed n and r. So the problem becomes

$$\max_{\{y_\ell\}} \left[\sum_{\ell \in L} y_\ell\right]^{-1} k^{-\frac{1}{2}} \left[(k-1)(N-n)+N\right]^{\frac{1}{2}} \left[\frac{(k-1)n}{n+r(N-n)}+1\right]^{\frac{1}{2}}$$

with $\quad y_\ell \geq 0, \quad \sum_{\ell \in L} y_\ell^2 = 1, \quad \sum_{j \in J} y_j^2 = \frac{n}{n+r(N-n)}$

and k,J satisfying

$$J = \{j \mid y_j \leq N^{-\frac{1}{2}}\},\ \# J = n, \quad (k-1)k \leq r < k(k+1)\ .$$

This maximization problem (for fixed n), can be split up into two independent parts:

I. $\quad \min\limits_{y_1,\ldots,y_n} \sum_{j=1}^{n} y_j \quad$ under $\quad y_j \geq 0, \quad \sum_{j=1}^{n} y_j^2 = \frac{n}{n+r(N-n)}\ .$

II. $\quad \min\limits_{y_{n+1},\ldots,y_N} \sum_{i=n+1}^{N} y_i \quad$ under $\quad y_i \geq 0, \quad \sum_{i=n+1}^{N} y_i^2 = \frac{r(N-n)}{n+r(N-n)}\ .$

Intuitively the solution of both problems is simple. Formal proof that the intuitive solutions are correct may be obtained by standard methods from non-linear programming theory, for instance by verifying the conditions of Griffith and Stewart (see Cooper and Steinberg [3]).

The intuitive solutions are:

I. $$\left.\begin{aligned} &y_1 = \dots = y_{q-1} = 0 \\ &y_{q+1} = \dots = y_n = N^{-\frac{1}{2}} \\ &y_q = \left[\frac{n}{n + r(N - n)} - \frac{n - q}{N}\right]^{\frac{1}{2}} \end{aligned}\right\}$$ with natural q $(1 \leq q \leq n)$ such that: $$q - 1 \leq \frac{n(r - 1)(N - n)}{n + r(N - n)} < q \ .$$

II. $$y_{n+1} = \dots = y_{N-1} = N^{-\frac{1}{2}}$$

$$y_N = \left[\frac{r(N - n)}{n + r(N - n)} - \frac{N - n - 1}{N}\right]^{\frac{1}{2}} .$$

This gives the solution of the original maximization problem. The result can be easily maximized numerically over n for $n = 1,2,\dots N - 1$. In this way one obtains a worst case behaviour of the repetitive scheme for the case of equal fixed costs. This behaviour is essentially better than in the general case of Fig. 2 as is shown by Fig. 3.

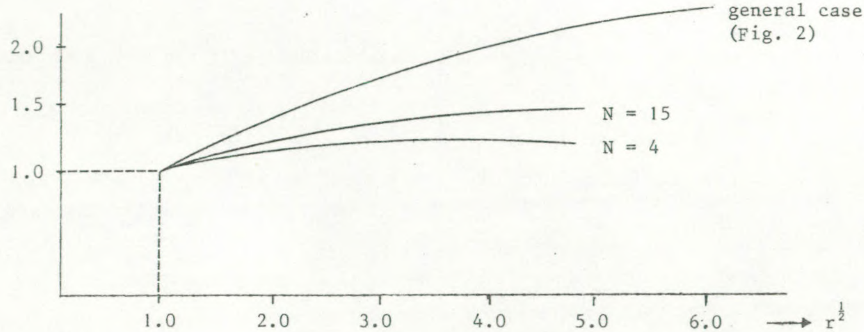

Fig. 3: worst case behaviour of the cost ratio for the case of equal starting costs with 4 and 15 products respectively; in close detail the graphs are relatively irregular.

5. Performance for randomly generated problems

For a small number of randomly generated problems with 6 and 9 products respectively the simplest clustering procedure of Section 2 has been used. In Table 1 the ratio of the resulting costs and the lower bound $\underset{\cdot}{K}$ is given for each of those problems. The F_i and α_i have been generated by random selections from $[0,1]$.

From this table one sees that usually the computed schedule only costs a couple of percents more than the lowerbound $\underset{\cdot}{K}$.

N = 6			
problem	r	k	Q
1	1.922	1	1.031
2	6.952	3	1.039
3	2.596	2	1.027
4	3.838	2	1.054
5	3.776	2	1.022
6	2.744	2	1.023
7	4.792	2	1.021
8	7.608	3	1.053
9	14.557	4	1.041

(a)

N = 9			
problem	r	k	Q
1	7.041	3	1.052
2	4.186	2	1.058
3	13.498	4	1.093
4	2.530	2	1.015
5	3.017	2	1.031

(b)

Table 1: performance of the simple clustering procedure for randomly generated problems; r gives the ratio of the optimal cycletimes of the clusters, k the optimal number of repetitions for the chosen clustering and Q gives the cost ratio (comparison with K).

6. Examples from the literature

For the problem of Rogers [6] we have:

$$N = 5, \quad r = 6.944, \quad k = 3, \quad Q = 1.007 .$$

For the problem of Bomberger [2] we have:

$$N = 10, \quad r = 11.88, \quad k = 3, \quad Q = 1.090 .$$

Remarks:

For the Bomberger problem one indeed meets the fitting problem as mentioned in Section 1. This requires some adaptation of the scheme as described in [5]. A rough way of meeting this difficulty is taking $k = 2$, this gives $Q = 1.128$. For the Bomberger problem a clustering into 4 pseudo-homogeneous groups each with cycletime two times the preceding one gives really good results: $Q = 1.014$ (compare [5]).

7. Worst case analysis for essentially positive F and α

In Section 4 it appeared already that requirement of equal fixed costs considerably improves the results of a worst case analysis. In this section we will investigate the effect of a similar requirement, which is also practically sensible. Namely, we will only consider those configurations where F_i and α_i are bounded away from zero. This is done by adding to requirements in the worst case analysis of Section 3:

$$f_\ell \geq N^{-3} \ , \quad A_\ell \geq N^{-3} \ .$$

This implies that the conditions A in Section 3 have to be amplified with

$$x_\ell \geq N^{-3/2} \ , \quad y_\ell \geq N^{-3/2} \ .$$

With similar techniques as in Sections 3 and 4, this can be worked out. In Fig. 4 the results for the worst case cost ratio (compared with K) are given for $N = 5$ and $N = 10$.

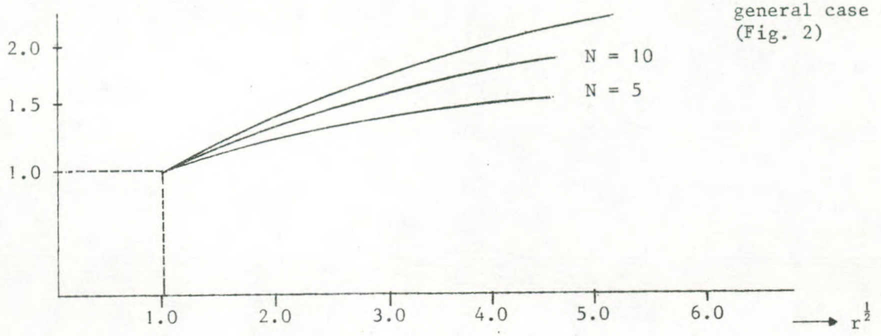

Fig. 4: worst case behaviour of the cost ratio for the situation with $f_\ell \geq N^{-3}$, $A_\ell \geq N^{-3}$.

8. Further analysis of the worst cases

For the worst cases of Section 3 (and also for nearly worst cases and worst cases of Sections 4 and 7) it is clear that the chosen clustering is not very sensible. Let us consider the worst case of Section 3 (for fixed t and r):

$$x_1 = y_1 \text{ and } x_N = y_N \qquad (1 \in I,\ N \in J)$$

moreover $x_i = 0$ for other $i \in I$

$y_j = 0$ for other $j \in J$.

Hence for the worst case products 1 and N both have ideal cycle time 1, but are put in different clusters. Product 1 has to go along with products with ideal cycle time 0 and product N has to go along with products with ideal cycletime ∞. Of course it would be more sensible to let product 1 switch from cluster I to cluster J and adapt the cluster cycle times accordingly. Figure 5 shows the effect on the cost ratio of this simple adaptation. In the adapted schedule the ideal k would be infinite, therefore we give the cost ratio for the choices $k = 5$, $k = 10$ and we give the limit for $k \to \infty$ ($k = \infty$).

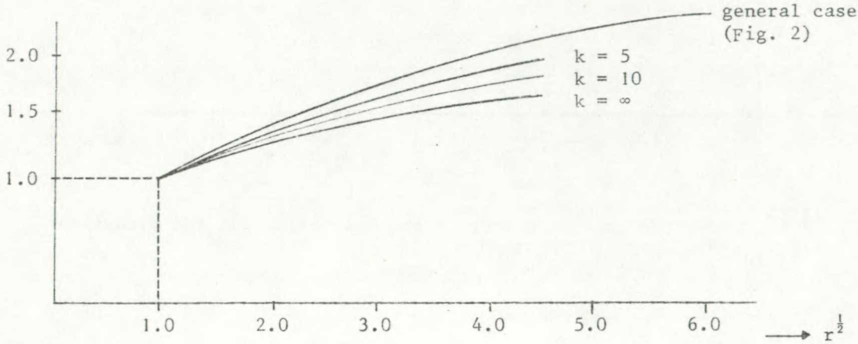

Fig. 5: effect of a simple adaptation on the cost ratio for the worst cases of Section 3.

References

[1] S. Axsäter, Aggregation of data in the economic lot scheduling problem. OR report no. 166 (October 1980), North Carolina State University at Raleigh (USA).

[2] E. Bomberger, A dynamic approach to a lot-size scheduling problem. Management Science 12 (1966) 778-784.

[3] L. Cooper, D. Steinberg, Introduction to methods of optimization. W.B. Saunders Company, 1970.

[4] S.E. Elmaghraby, The economic lot scheduling problem (ELSP): Review and extension. Management Science 24 (1978) 587-598).

[5] T.H.B. Hendriks, J. Wessels, Repetitive schemes for the single-machine multiproduct lot-size scheduling problem. Operations Research Verfahren XXVI (1978) 571-581.

[6] J. Rogers, A computational approach to the economic-lot scheduling problem. Management Science 4 (1958) 264-291.

[7] J. Wessels, N. Thijssen, Repetitive schemes with performance bounds for the economic-lot scheduling problem with critical set-up times. Mitteilungen der Math. Ges. der DDR 1978 (Heft 1) 49-54.