# NOTE ON PERFORMANCE OF HEURISTICS FOR SOLVING THE PRODUCT RATE VARIATION PROBLEM

**Joaquín Bautista, Ramón Companys, Manuel Mateo and Jordi Pereira**
*Dpto. de Organización de Empresas, Polytechnic University of Catalonia, Spain.*
bautista@oe.upc.es, companys@oe.upc.es, mmateo@oe.upc.es, pereira@oe.upc.es

**Abstract**

Sequencing units on a production or assembly line in order to attenuate the variations in the rates of resource consumption is a problem that has received growing attention in recent years. The present study features an analysis of the PRV (Product Rate Variation) problem with quadratic discrepancy function. Taking advantage of the properties satisfied by any optimum sequence, efficiency of several heuristic procedures is increased. Although an efficient exact method exists, these heuristics have a potential utility in developing procedures adapted to the industry. The results of a short computational experience are included.

## 1. Introduction

In mixed assembly production lines, all the units vary affecting the consumption of resources. How to place the units in sequence, to diminish the variations in the rates of resource consumption is a problem that has received attention since 1983, as it is related to JIT. In [10] a description of the state of the art is presented, in which these sequencing problems are classified in two categories: PRV (Product Rate Variation) and ORV (Output Rate Variation).

In the PRV problem the objective is to minimise the rate variation. The problem was presented by [12] and studied by [3], [4], [6], [7], [9] and [11] among others. This study focuses on the PRV problem with quadratic discrepancy function. We provide a computational experience based on the studied heuristic algorithms.

## 2. The PRV problem

The traditional formulation of the PRV problem is the following: units of P different products have to be sequenced in a production or assembly line; the number of units of the product i to be sequenced is $u_i$ ($1 \le i \le P$). The total units to be sequenced are T. The positions in the sequence will be indicated by the index t ($1 \le t \le T$) on account of the implicit supposition that all the units flow at a constant and identical speed in the line. The ideal or mean rate of the product i in the sequence is $r_i = u_i/T$ ($1 \le i \le P$). To define the position of the units in a sequence, values $x_{i,t}$ ($1 \le i \le P$; $1 \le t \le T$) correspond to the number of units of the product i sequenced between the positions 1 and t (both inclusive). For the sake of coherence, $x_{i,0} = 0$ ($1 \le i \le P$). For any value t, the ideal number of units sequenced for the product i between the positions 1 and t is $t.r_i$, while the real value for a sequence is $x_{i,t}$. It seems useful to measure the non-regularity of the sequence through a distance between both sets of values. A quadratic distance is usually used [12].

Given a sequence S, the index of non-regularity will be designated as SDQ(S). The term of the index corresponding to position t (contribution of position t) is designated as $SDQ_t(S)$: $SDQ_t(S) = \Sigma_{1 \le i \le P}\ \varphi (x_{i,t} - t.r_i) = \Sigma_{1 \le i \le P} (x_{i,t} - t.r_i)^2$. These contributions depend on t and the values xi,t, that is, on the number of units sequenced between 1 and t for each product, but not strictly on how these units have been sequenced or how the rest will be sequenced. Consequently, if $X_t$ is the (P,1) vector whose components are $x_{i,t}$, it will often be easier to write $SDQ_t(X_t)$ than $SDQ_t(S)$.

In [7] it is demonstrated that if the objective function can be represented as a sum of the discrepancy at each value of t, the problem of searching for and optimum sequence can be transformed to the search for a minimal path in an associated graph, where the vertices are associated to the vectors $X_t$, where $X_t$ is the (P,1) vector whose components are $x_{i,t}$.

In [11] it is demonstrated that if the objective function can be represented as a sum of the discrepancy measured through a non-negative, symmetrical and convex function, the problem of searching for an optimum sequence can be reduced to an assignment problem.

## 3. Heuristics

The procedures known to solve assignment problems find optimal solution to the problem formulated by [12] efficiently. But the industrial environment problem has certain features [8], making it different of the theoretical one. Solutions obtained for the theoretic problem may be useless in real world. These features lead to use other procedures, that are variations of direct heuristic procedures. In these procedures, the sequence or path is built progressively. At each stage in the procedure, a new vertex is added to the segment of built sequence, chosen according to prefixed criteria; and once the choice is made, it is not reconsidered. This procedure coincides with the "goal chasing" proposed by [13] for the ORV problem, which can be solved optimally as shown in [1], [2], and heuristic H1 [12] for the PRV.

Several authors have stated, on the basis of computational experiences, that the behaviour of this heuristic is not efficient because of its short-sightedness. One solution is considering the contribution of more than one arc in the prolongation of the path from a vertex X at level t. The 2-step heuristic considers the two following vertices from the last one added to the segment. This procedure, more efficient than the previous one, coincides with heuristic H2 proposed by [12].

In [9] a modification of H2 (denoted as H1.5) is shown. This heuristic is faster than H2 and also gives good results. Ding and Cheng assure the procedure is a 2-step heuristic, but the claim is denied by computational experience. In fact, the provided demonstration contains a wrong step [3].

A three-step heuristic (H3) may lead to better results than H2, but with a greater expenditure. We also propose a heuristic called H2.5, which coincides substantially with a 3-step heuristic, except the simplified scheme for positions t+2 and t+3, inspired by [9]. Following this criterion, it is also possible to develop a H3.5 heuristic.

## 4. Rules for constructing optimum prolongations

The number of considered vertices following a vertex $X_t$ in a heuristic step may be important; therefore, it will be useful to add the following rules to reduce this number. The aim of these rules is to eliminate those vertices following from $X_t$ that cannot form part of any optimum prolongation. There are three main rules (rules 1, 2 and 3), which can be complemented with three more (rules 4, 5 and 6) if several products share the same rate. These rules are only stated here, as the demonstration was included in [5].

**Rule 1**: If $r_i > r_j$ and $x_{i,t} - x_{j,t} \leq (r_i - r_j).(t+1)$, we can get rid of $X_t + I_j$ in prolongation paths from $X_t$ (product j must not be sequenced at position t+1).

**Rule 2**: If $r_i = r_j$ and $x_{i,t} - x_{j,t} < 0$, we can get rid of $X_t + I_j$ in prolongation paths from $X_t$.

**Rule 3**: If $r_i < r_j$ , $x_{i,t} < u_i$ and $x_{i,t} - x_{j,t} < (r_i - r_j).[t + T - (m + k)(u_i - x_{i,t}) + k]/2$ , we can get rid of $X_t + I_j$ in paths from $X_t$, being m the number of products with rate higher than $r_i$, k the number of products with rate $r_i$, and $I_j$, a (P,1) vector with j-th component is 1 and the rest is 0.

**Rule 4**: If $r_i = r_j$ , $x_{i,t} - x_{j,t} \leq 0$ and i < j we can get rid of $X_t + I_j$ in paths from $X_t$.

**Rule 5**: If $r_i = r_h > r_j$ and $x_{i,t} > x_{h,t}$ , we can get rid of $X_t + I_j$ in prolongation paths from $X_t$.

**Rule 6**. If $r_i > r_j$, i belongs to a family with k products which have the value $x_{i,t} = a$ at t, and $a - x_{j,t} \leq (r_i - r_j).[ t + (1 + k) / 2])$ , we can get rid of $X_t + I_j$ in prolongation paths from $X_t$.

The behaviour of the above heuristics is notably inefficient when there are families of products. As can be observed in the computational experience, filtering generally increases the efficiency of the rules. An additional advantage is the reduction in computing time.

## 5. Computational Experience

In the following Tables the summarised results show the behaviour of several heuristics, without and with rule filtering. Those heuristics have been applied to five sets of 5000 instances each one, with quadratic discrepancy function; each set correspond a one couple of values (T, P) as shown in Table 1. For each couple (T, P) 5000 different instances are randomly generated. The optimal solutions have been obtained solving their associated assignment problem [11].

*Table 1*: Size of 5.000 instances of each set and computer time per instance to attain optimum solution (sec.)

| **Set** | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| **T/P/Time** | 500/10/0.22 | 1000/10/1.52 | 1500/10/4.58 | 2000/10/11.08 | 2500/10/19.90 |

*Table 2*: Left, mean relative deviation (%) and, right, maximum relative deviation (%)

| **Heuristic\set** | **1** | **2** | **3** | **4** | **5** | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|---|---|---|---|---|
| **H1** | 2.62 | 2.95 | 2.89 | 3.06 | 3.06 | 13.95 | 14.60 | 16.61 | 22.90 | 18.33 |
| **H1.5** | 1.18 | 1.37 | 1.25 | 1.37 | 1.35 | 13.95 | 11.61 | 9.24 | 12.94 | 10.79 |
| **H2** | 1.44 | 1.67 | 1.56 | 1.68 | 1.65 | 13.95 | 11.61 | 10.89 | 12.94 | 12.06 |
| **H2.5** | 1.00 | 1.20 | 1.03 | 1.14 | 1.11 | 13.95 | 11.61 | 9.39 | 12.94 | 10.79 |
| **H3** | 0.97 | 1.17 | 1.01 | 1.12 | 1.11 | 13.95 | 11.61 | 9.24 | 12.94 | 10.79 |
| **H3.5** | 0.77 | 0.96 | 0.80 | 0.88 | 0.86 | 13.95 | 11.61 | 8.49 | 12.94 | 10.79 |
| **H1+** | 2.50 | 2.89 | 2.85 | 3.03 | 3.03 | 13.95 | 14.60 | 16.61 | 22.90 | 18.33 |
| **H1.5+** | 1.12 | 1.34 | 1.24 | 1.36 | 1.34 | 13.95 | 11.61 | 9.24 | 12.94 | 10.79 |
| **H2+** | 1.42 | 1.67 | 1.56 | 1.67 | 1.65 | 13.95 | 11.61 | 10.89 | 12.94 | 12.06 |
| **H2.5+** | 0.97 | 1.17 | 1.01 | 1.12 | 1.11 | 13.95 | 11.61 | 9.24 | 12.94 | 10.79 |
| **H3+** | 0.97 | 1.17 | 1.01 | 1.12 | 1.11 | 13.95 | 11.61 | 9.24 | 12.94 | 10.79 |
| **H3.5+** | 0.77 | 0.96 | 0.80 | 0.88 | 0.86 | 13.95 | 11.61 | 8.49 | 12.94 | 10.79 |

*Table 3*: Left, rate of optimum solutions (%) and, right, computer time per instance (sec.)

| **Heuristic\set** | **1** | **2** | **3** | **4** | **5** | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|---|---|---|---|---|
| **H1** | 1.66 | 1.16 | 0.42 | 0.38 | 0.36 | 0.02 | 0.04 | 0.06 | 0.08 | 0.10 |
| **H1.5** | 1.98 | 1.24 | 0.44 | 0.38 | 0.36 | 0.03 | 0.07 | 0.10 | 0.14 | 0.17 |
| **H2** | 2.26 | 1.32 | 0.54 | 0.42 | 0.44 | 0.15 | 0.30 | 0.47 | 0.63 | 0.78 |
| **H2.5** | 4.94 | 1.68 | 1.10 | 0.62 | 0.62 | 0.29 | 0.59 | 0.91 | 1.21 | 1.52 |
| **H3** | 5.70 | 1.82 | 1.14 | 0.64 | 0.68 | 1.36 | 2.84 | 4.41 | 5.92 | 7.45 |
| **H3.5** | 7.68 | 2.60 | 2.12 | 1.40 | 1.12 | 2.70 | 5.62 | 8.79 | 11.79 | 14.86 |
| **H1+** | 1.68 | 1.16 | 0.42 | 0.38 | 0.36 | 0.04 | 0.08 | 0.13 | 0.18 | 0.22 |
| **H1.5+** | 2.06 | 1.26 | 0.44 | 0.38 | 0.36 | 0.08 | 0.17 | 0.26 | 0.35 | 0.44 |
| **H2+** | 2.30 | 1.32 | 0.54 | 0.42 | 0.44 | 0.10 | 0.21 | 0.37 | 0.49 | 0.62 |
| **H2.5+** | 5.70 | 1.80 | 1.12 | 0.64 | 0.68 | 0.17 | 0.34 | 0.61 | 0.82 | 1.04 |
| **H3+** | 5.70 | 1.82 | 1.14 | 0.64 | 0.68 | 0.23 | 0.46 | 0.90 | 1.20 | 1.52 |
| **H3.5+** | 7.78 | 2.66 | 2.18 | 1.98 | 1.10 | 0.35 | 0.71 | 1.45 | 1.93 | 2.46 |

On the left side of Table 2, the mean of the relative deviation of the solutions obtained by each heuristic is shown, being the relative deviation of a heuristic solution: (solution value of heuristic – optimum value)/ optimum value. On the right side, the maximum of those relative deviation of the solutions is shown. On the left side of Table 3 the rate of optimum solutions obtained by each heuristic is shown, while on the right side the computational time is shown.

Heuristics H1, H1.5, H2, H2.5, H3 and H3.5 are the heuristics described in Section 3. Heuristics H1+, H1.5+, H2+, H2.5+ H3+ and H3.5+ coincide with the above ones plus the rule filtering at all the steps, as proposed in Section 4.

## 6. Conclusions

The graph associated to PRV problem, with convex discrepancy objective function, have some properties regarding any optimum path or prolongation. These properties can be completed with additional, generally stricter, properties in case of functions with quadratic discrepancy. The properties can be used efficiently in constructive heuristic algorithms to determine sequences (or paths) close to the optimum solution. Constructive algorithms with filtering of candidates for the prolongation of a partial sequence, besides improving the relative deviation of the solutions, run faster than without filtering, for significant instance sizes.


## Acknowledgements

This research is supported by CICYT grant TAP95-0420.


## References


[1] Bautista, J. (1993), Procedimientos heurísticos y exactos para la secuenciación en sistemas productivos de unidades homogéneas (contexto JIT), Doctoral Thesis, DOE, ETSEIB-UPC.

[2] Bautista, J., Companys, R. and Corominas, A. (1996a), "Heuristic and exact algorithms for solving the Monden problem", *Eur. J. Opl. Res*. **88**, 101-113.

[3] Bautista, J., Companys, R. and Corominas, A. (1996b), *A coment on 'A comment on 'A simple sequencing algorithm for mixed-model assembly lines in just-in-time production systems''*, D.I.T. 96/02, DOE, ETSEIB-UPC.

[4] Bautista J., Companys R. and Corominas, A. (1996c), "A note on the relation between the product rate variation (PRV) problem and the apportionment problem", *J. Opl. Res. Soc*. **47**, 1410-1414.

[5] Bautista, J., Companys, R. and Corominas, A. (1997a), "Resolution of the PRV problem", DIT 97/25, DOE, ETSEIB-UPC.

[6] Bautista, J., Companys, R. and Corominas, A. (1997b), "Modelling and solving the production rate variation problem (PRVP)", *Top* **5** (2), 221-239.

[7] Bautista, J., Companys, R. and Corominas, A. (1999), "Note on cyclic sequences in the product rate variation problem", *Eur. J. Opl. Res*. 124 (3), 468-477.

[8] Companys, R., Bautista, J. and Mateo, M. (2000), "Mixed-model assembly line level scheduling at automobile companies", invited paper at *I World conference on production and operations management, POM Sevilla 2000.*

[9] Ding, F.-Y. and Cheng, L. (1993), "A simple sequencing algorithm for mixed-model assembly lines in just-in-time production systems", *Opns. Res. Letters* **13** (1), 27-36.

[10] Kubiak, W. (1993), Minimization of production rates in just-in-time systems: A survey, *Eur. J. Opl. Res*. **66**, 159-271.

[11] Kubiak, W. and Sethi, S. (1991), "A note on 'Level schedules for mixed-model assembly lines in just-in-time production systems'", *Mgt. Sci*. **3**7 (1), 121-122.

[12] Miltenburg, J. (1989), "Level schedules for mixed-model assembly lines in just-in-time production systems", *Mgmt. Sci*. **32** (2), 192-207.

[13] Monden, Y. (1983), *Toyota production system*, Institute of Industrial Engineers Press, Norcross, GA.