Proceedings of the 1999 IEEE
International Symposium on Assembly and Task Planning
Porto, Portugal - July 1999

A Procedure to Solve the CORV Problem

Joaquín Bautista, Manuel Mateo, Ramon Companys, Albert Corominas

Institut d'Organització i Control de Sistemes Industrials (ETSEIB-UPC). Diagonal 647, 11th, 08028 Barcelona, Spain.
Contact e-mails: bautista@ioc.upc.es, mateo@ioc.upc.es

Abstract

Sequencing units on assembly lines in order to attenuate rate variations in resource consumption is a problem that has received growing attention in recent years. In this work, we deal with a particular case, the CORV (Constrained Output Rate Variation) problem, that seems to be better adapted than other views to real industry problems, especially in car production systems. Once given a general introduction and formulation, a procedure is described to obtain the searched sequence.

1 Introduction

In mixed assembly lines, not all the units are identical. All of them are quite similar, but they can vary in several aspects that have influence on the resource consumption associated to these units (workstation load and/or part requirements). How to place units in a sequence, with the objective to reduce extremely rate variations referred to resource consumption, is a problem that has received attention for many years and has been dealt with in the literature since 1983, since it was related to JIT concepts.

A classification for problems of regular sequences in JIT systems that divides them in two groups is presented in [1]: PRV (Product Rate Variation) and ORV (Output Rate Variation). A more detailed classification [2], in order to consider other aspects, is the following:

	Products	Resources		
		Components		Load
		1-level	Multilevel	
Property	CP	CO	CMO	CL
Function	PRV	ORV	MORV	LRV
Mixture	CPRV	CORV	CMORV	CLRV

This double point of view in the table corresponds to the following two aspects:
1) columns consider the object under a certain criterion of regularity: products, components or parts (one or several levels) or load;
2) rows show how regularity can be understood: by properties or constraints of regularity to be satisfied by sequences; by a measure of regularity, or usually non-regularity, that indicates if a sequence is more or less regular than others; and finally, by a combination of both aspects (a function considering properties).

Both aspects, specially the latter, lead to specific exact or heuristic algorithms to determine the searched sequences.

In the PRV problem the objective is to minimise the rate variation for different products in any segment of a sequence, i.e. regularity in manufacturing products. PRV was first presented in [3], and then, several works dealing with heuristic procedures [3,4,5,6,7,8,9,10] and with exact procedures [10,11,12,13] have been proposed among others.

The problem of regularity in the consumption of components was formalised in [14] and called ORV [1]. Later, several heuristic procedures [15,16,17,18] and exact procedures [16,17] have been proposed in order to solve it. We have called it MORV (Multilevel Output Rate Variation) when some levels of parts with different weights must be considered to evaluate sequences and has been treated in [15,16,19], among others.

Moreover, load balancing is the main objective in the LRV (Load Rate Variation) problem, treated in [20,21].

The CO (Constrained Output) problem is presented and solved with CLP (Constraint Logic Programming) in [22]; if products are considered instead of parts, the CO problem can be reduced to the CP (Constrained Product) problem. The extension of these concepts, proposed in [2], leads to define CPRV (Constrained Product Rate Variation) and CORV (Constrained Output Rate Variation) problems.

Several meetings with managers of industries in the automotive sector lead us to conclude that they assume the car sequencing problem closer to a CPRV or CORV problem (even CO) than a PRV or ORV problem, most frequently found in literature.

2 The CP and CO problems

The usual formulation for the CP problem is as follows: units of P different products must be sequenced in an assembly line, being u_i (i=1,2,...,P) the number of units of product i to be sequenced. The total of units to be sequenced is T, i.e. $T = \Sigma_{1 \le i \le P}\, u_i$.

The positions in the sequence are indicated by the index t (t=1,2,...,T) on account of the implicit supposition that units flow on the line at a constant speed for all of them. The values $x_{i,t}$ (i=1,2,..,P; t=0,1,2,..T), defined to know the position of the units in a sequence, correspond to the number

This work was supported by the CICYT Project TAP 98-0494.

of units for product i sequenced between positions 1 and t (both inclusive), being $x_{i,0} = 0$ (i=1,2,...,P).

Two positive integer values a_i and b_i, with a_i<b_i, are associated to each product i. A sequence is considered regular if, for any product i, at most a_i positions are occupied by units of product i in any segment of b_i consecutive positions. Therefore: $x_{i,t+bi} - x_{i,t} \leq a_i$ for i=1,2,...,P; $t = 0,1,..,T - b_i$.

Given a set of values (P; u_i, a_i, b_i; i=1,2,...,P), i.e. an instance of the CP problem may have no feasible solution (none of the possible sequences satisfies the conditions of regularity), one solution or several solutions. In order to establish the necessary conditions for the existence of feasible solutions, it is convenient to define previously the ideal or mean rate of the product i in the sequence: r_i (i=1,2,..,P): $r_i = u_i/T$.

To assign the u_i units of product i, the following expression must be accomplished:

$$r_i \leq \frac{a_i}{b_i} + \frac{a_i}{T}\left(1 - \frac{a_i}{b_i}\right)$$

whose value tends to: $r_i \leq \frac{a_i}{b_i}$ when T is increased.

The fulfilment of the previous condition guarantees that at most a_i units of product i can be allocated in any segment of b_i consecutive positions in the sequence. But the fulfilment of this condition for two products i and h does not guarantee that it is possible to place units of both products in the sequence satisfying simultaneously the constraints for both of them.

To formulate the CO problem, it is necessary to consider that each product i consumes $n_{j,i}$ units of the component or part j (j=1,2,...,C). Let $y_{j,t}$ be the number of units of component j consumed by the products sequenced in the first t positions, whose value can be calculated through the expression: $y_{j,t} = \Sigma_{1\leq i\leq P}\, n_{j,i} x_{i,t}$, that, expressed in a matrix form, is: $Y = N \cdot X$

Two positive integers a_j and b_j are associated to each component j. A sequence is considered regular if, for all components j, the units assigned to these positions have a requirement not higher than a_j units in any segment of b_j consecutive positions. The constraints on the components may be expressed in the following way:

$$y_{j,t+bj} - y_{j,t} \leq a_j, \text{ for } j=1,2,...,C;\ t = 0,1,...,T - b_j$$

The ideal or mean rate of the consumption of component j is:

$$r_j = \frac{\sum_{i=1}^{P} n_{j,i} \cdot u_i}{T}$$

necessary to establish conditions for the existence of feasible solutions.

In many real circumstances, $n_{j,i}$ adopts only the values 0 or 1 (absence or presence of an option, related to several components, i.e. air conditioning plus cables and screws). This happens precisely in the kind of problem presented in [22], like Example 1. A situation with 100 units to be sequenced, that can be found in [16,24,25,26], is presented in Figure 1 and below described.

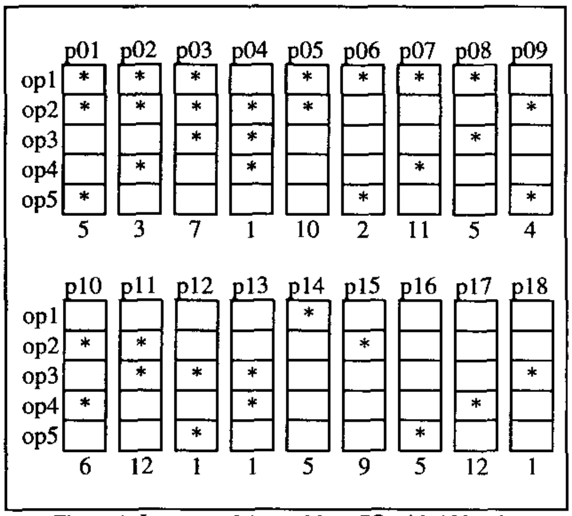

Figure 1: Instance of the problem CO with 100 units to be sequenced. The symbol * supposes to apply the special option in a module.

Example 1

18 types of products (P=18), or varieties of a product, made from 5 modules or components (C=5) have to be sequenced. Each module may adopt two values: 0 corresponds to the basic option, and 1 to an special option that generates workstation overload, and/or can be related to the consumption of certain components.

The production plan, whose total number is 100 units (T=100), is defined by the quantity of units for each type of product to be sequenced:

{5,3,7,1,10,2,11,5,4,6,12,1,1,5,9,5,12,1}

The constraints on special options in the five modules considered (or components) are the following ones:

Option 1: No more than 1 out of 2 units.
Option 2: No more than 2 out of 3 units.
Option 3: No more than 1 out of 3 units.
Option 4: No more than 2 out of 5 units.
Option 5: No more than 1 out of 5 units.

The solution proposed in [26], obtained using CLP, accomplishes the imposed constraints. That solution is shown in Figure 2. In Figure 3, there is another solution obtained by the procedure described in Section 5.

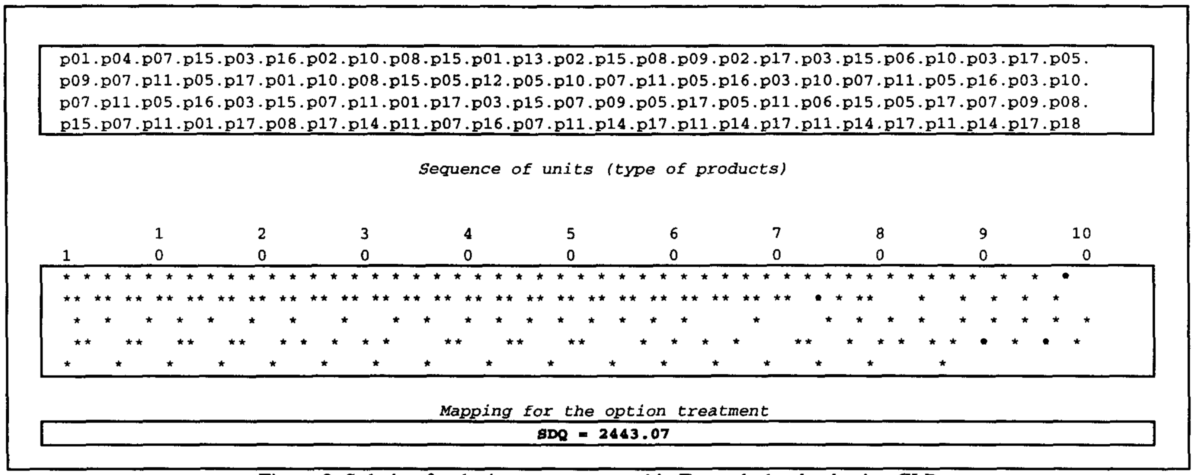

Figure 2: Solution for the instance presented in Example 1 solved using CLP

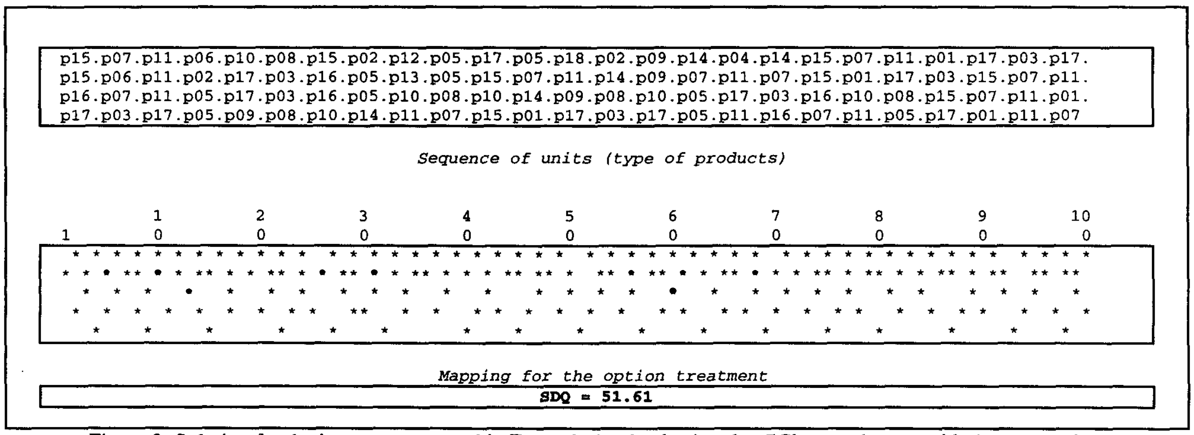

Figure 3: Solution for the instance presented in Example 1 solved using the *GCh* procedure considering constraints

3 Similarity between CO and ORV problems

A certain similarity between the CO and ORV problems can be easily noticed. In both cases, the proposed objective is to make regular the appearance of special options or the consumption of components. This fact is revealed if the mean rates of consumption Rj (j=1,..,C) are compared with the ratios fixed by the constraints:

Option 1: $r_1 = 0.48$ ratio 1 = 1/2 = 0.50
Option 2: $r_2 = 0.57$ ratio 2 = 2/3 = 0.67
Option 3: $r_3 = 0.28$ ratio 3 = 1/3 = 0.33
Option 4: $r_4 = 0.34$ ratio 4 = 2/5 = 0.40
Option 5: $r_5 = 0.17$ ratio 5 = 1/5 = 0.20

Logically, the ratios are not lower than the rates; otherwise, a solution satisfying the constraints could not be generally feasible.

The nearness between ratios and rates leads to approach Example 1 as a case of ORV problem. To reach a solution, two procedures have been used: the heuristic *GCh* (Goal Chasing) [14] and *BDP* (Bounded Dynamic Programming) [16,17,24]. *GCh* is a greedy procedure that builds the sequence adding units in order to minimise *SDQ*. *BDP* uses bounds to reduce the set of possible solutions, under a scheme of dynamic programming. The *SDQ* function, referred to components, which has been used as objective function, is the following:

$$SDQ = \sum_{t=1}^{T} \sum_{j=1}^{C} \left(\sum_{i=1}^{P} n_{j,i} x_{i,t} - t r_j \right)^2$$

The sequence from *GCh*, whose objective function *SDQ* is 52.97, does not satisfy the constraints in six times for option 1 (segments: 29-30, 71-72, 79-80, 83-84, 87-88 and 96-97); once for option 3 (segment 47-49); and once for option 5 (segment 93-97). *BDP* obtains better solutions, but without satisfying only the constraint for option 1, to be precise in the segments 25-26 and 75-76, reaching *SDQ* = 44.49. Looking for a minimum *SDQ* favours the constraint accomplishment, as it was foreseeable.

4 The CPRV and CORV problems

The CPRV problem is a natural extension of the CP problem. Two positive integers a_i and b_i, with $a_i<b_i$ are associated to each product i. A sequence is considered regular if, for all product i, at most a_i positions are occupied by units of product i, in any segment of b_i consecutive positions. Moreover, the objective is to minimise the product rate variation for each product located in any segment of the sequence.

The CORV problem is a natural extension of the CO problem and consists in sequencing T units, from which u_i are of type or product i (i=1,..,P), being as close as possible to regularity. Products (i=1,..,P) reflect a unitary consumption of components (j=1,..,C), expressed by the relations $n_{j,i}$. Moreover, components or options can be subjected to constraints of maximum load (a_j) in segments or subsequences of prefixed length (b_j). It can be formalised as follows:

Let:

$N_j = \sum_{i=1}^{P} n_{j,i} u_i$, the total demand for component j.

$r_j = N_j / T$, the ideal rate of consumption for component j (mean rate).

$x_{i,t}$, the actual production of product i after t units made.

$y_{j,t} = \sum_{i=1}^{P} n_{j,i} x_{i,t}$, the actual consumption of component j after t units made.

One pretends an actual consumption for all components j ($y_{j,t}$) in any instant t adjusted as good as possible to the ideal consumption tr_j. As a measure of non-regularity, several formulations are possible, among which are the following:

$$SDQ = \sum_{t=1}^{T} \sum_{j=1}^{C} \left(\sum_{i=1}^{P} n_{j,i} x_{i,t} - tr_j \right)^2$$

$$SDR = \sum_{t=1}^{T} \sum_{j=1}^{C} \left| \sum_{i=1}^{P} n_{j,i} x_{i,t} - tr_j \right|$$

Considering the function *SDQ*, the whole problem can be formulated through a mathematical program:

$$[min] SDQ = \sum_{t=1}^{T} \sum_{j=1}^{C} \left(\sum_{i=1}^{P} n_{j,i} x_{i,t} - tr_j \right)^2 \qquad (1)$$

$$\sum_{i=1}^{P} x_{i,t} = t \qquad 1 \le t \le T \qquad (2)$$

$$x_{i,T} = u_i \qquad 1 \le i \le P \qquad (3)$$

$$0 \le x_{i,t} - x_{i,t-1} \le 1; \qquad 1 \le i \le P, \; 1 \le t \le T \qquad (4)$$

$$\sum_{i=1}^{P} n_{ji} \left(x_{i,t+b_j} - x_{i,t} \right) \le a_j \qquad 1 \le j \le C, \; 0 \le t \le T - b_j \qquad (5)$$

$x_{i,t}$ *integer variables*

The constraints (2) indicate that exactly t units must be sequenced after t units; (3) determine that the final sequence contains the given number of units for each product; (4) impose that productions in consecutive instants are coherent; finally, (5) establish the upper bounds for the consumption of components (options) in a segment.

It is remarkable that if P=C and $n_{j,i}$=1 (i=j) and $n_{j,i}$=0 ($i \neq j$), the CPRV problem can be observed as a particular case of the CORV problem; moreover, the PRV problem also can be treated as a particular case of ORV problem. On the other hand, if constraints (5) are cancelled in the CORV problem, then it appears as the ORV problem. In conclusion, the resolution of the CORV problem implies the resolution of ORV, CO, CPRV, PRV and CP problems.

5 Resolution of the CORV problem

5.1 Graph associated to the problem

In order to represent the problem, an acyclic graph G_0 with T+2 levels can be defined. A vertex at level t (t=1,..,T) is characterised by:

1) a vector of P components $X(t) = (x_{1,t},...,x_{P,t})$, such that: $\sum_{\forall i} x_{i,t} = t$ and $0 \le x_{i,t} \le u_i$

2) a subsequence $S(t)=(s_{1,t},...,s_{l(t),t})$, with $l(t)$ units in $X(t)$, which represent the units added in the last $l(t)$ instants, being $l(t) = \min\{t, \max(b_j)\}$.

At level 0 there is only a vertex α associated to the production $X(0)$ = (0,...,0) and an empty subsequence. The level T is constituted by vertices with a exclusive vector $X(T) = (u_1,...,u_P)$ and for all the subsequences of $l(T)$ units that can be built according to the production $X(T)$. Finally, at level T+1 there is only a vertex ω.

An arc exists between the vertices $V_{t-1}[X(t-1), S(t-1)]$ and $V_t[X(t), S(t)]$, at levels t-1 and t (t=1,..,T) respectively, if:

(1) $X(t$-1) and $X(t)$ satisfy the constraints (4), and

(2) $S(t$-1) and $S(t)$ are compatible subsequences:

$s_{1,t} = s_{2,t-1}, \ldots, s_{l(t)-1,t} = s_{l(t),t-1}$.

P arcs emerge from the vertex α (one for each type of product) and as many arcs as vertices are at level T pointing to vertex ω. The number of vertices in G_0 can be very high (about $1.5 \cdot 10^{13}$ for Example 1); nevertheless, many states are not reachable, so the graph G_0 admits a pruning. That action consists in eliminating vertices associated to subsequences that cannot satisfy the constraints (5) and, consequently, eliminating all the emergent and incident arcs to those vertices.

Let G_1 be the graph after the indicated pruning. An index of non-regularity can be assigned to each vertex $V_t[X(t), S(t)]$ (or to all its incident arcs) of G_1, as follows:

$$f(V_t) = \sum_{j=1}^{C} \left(\sum_{i=1}^{P} n_{j,i} x_{i,t} - t r_j \right)^2 \qquad (6)$$

In these conditions, if there is no path between the vertices α and ω in G_1, the problem has no solution, unless a constraint is not satisfied. Otherwise, if there is some path between vertices α and ω, finding a sequence with minimum *SDQ* is equivalent to find the minimal path from the vertex α to the vertex ω. A heuristic procedure to solve the proposed problem is below presented.

5.2 GCh dealing with constraints

How to adapt the *GCh* (*Goal Chasing*) procedure of TOYOTA [14] is next presented. Let a^-_{ijt} be the total load of component j in the segment $[t-b_j+1,t]$ with product i in the position t; if going back any option j has a total load (a^-_{ijt}) higher than the maximum allowed (a_j) in the segment of length b_j, then the product of type i is rejected as candidate for the position t (*rule-1*).

At a position t, all the products with pending production are yet not satisfying some load constraint, two decisions are available: 1) a hole may be left in the t-th position of the sequence (omitting constraints of type 2) and go ahead to the next position; 2) adding to the sequence under construction the type of product with pending production that minimises (6). The consideration of *rule-1* to the *Goal Chasing* procedure leads to the following algorithm.

Algorithm A1

0. Initialise:

   $t \leftarrow 1$, $X_i \leftarrow 0$ $(1 \le i \le P)$;

   Determine:

   $$r_j = \frac{N_j}{T} = \frac{1}{T} \sum_{i=1}^{P} n_{j,i} u_i \quad 1 \le j \le C$$

1. Load computing:

   Let X_i $(1 \le i \le P)$ be the sequenced units for the type of product i up to position t-1. For any i such that $X_i < P_i$ determine:

   $$d_i = \sum_{j=1}^{C} \left(\sum_{h=1}^{P} n_{j,h} (X_h + \delta_{ih}) - t r_j \right)^2$$

   $$where: \ \delta_{ih} = 1 \ if \ i = h, \quad \delta_{ih} = 0 \ if \ i \ne h$$

   $$a^-_{ijt} = n_{j,i} + \sum_{\tau=t_\alpha}^{t-1} n_{j[\tau]} \qquad \forall i,j$$

   $$where: \quad t_\alpha = max\{0, t - b_j + 1\}$$

   $$[\tau] \quad is \quad \tau - th \ product \ in \ the \ subsequence$$

2. Selection of a product to be sequenced:

   Let Π be the set of types of product such that $X_i < u_i$ and $a^-_{ijt} \le b_j$.
   - If $\Pi = \varnothing$, leave hole in position t; go to 3.
   - If $\Pi \ne \varnothing$, choose the type of product s such that: $d_s = \min\{d_i\}$ among the elements, i.e. a unit of type s is sequenced in the t-th position.

3. Updating of solutions:

   $X_s \leftarrow X_s + 1$
   If $t = T$, end;
   else $t \leftarrow t+1$; go to 1.

A procedure for backtracking has been added to the algorithm if $\Pi = \varnothing$ in step 2. In that case, the previous vertex in the path is considered, the unsuccessful path on the graph G_1 is left, and a new continuation is taken.

In order to avoid unproductive searching through paths that lead to unfeasible solutions, a procedure to reject vertices has been included. This rejection takes care of a possible incompatibility of the option load for the units not yet sequenced, based on an upper bound for allowable load in the segment to be completed. Indeed, let a^+_{jt} be the total load for the option j that will be caused by the T-t+1 units not yet sequenced when a segment of length t-1 is built; i.e.:

$$a^+_{jt} = \sum_{i=1}^{P} n_{j,i} \cdot (u_i - x_{i,t-1})$$

An upper bound can be easily established, obtained from the allowable load in the segment $[t,T]$ to be completed, according to the type of product i candidate for the t-th position. The proposed bound, K_{ijt}, is determined as follows:

$$\text{if } n_{j,i} = 0 \text{ then}$$

$$K_{ijt} = C_0 b_j + \min\left\{b_j, R_0 \cdot \max_h(n_{j,h})\right\}$$

$$C_0 = \left\lfloor \frac{T-t}{a_j} \right\rfloor; R_0 = (T-t).MOD.(a_j)$$

$$\text{if } n_{j,i} \geq 1 \text{ then}$$

$$K_{ijt} = C_1 b_j + \min\left\{b_j, R_1 \cdot \max_h(n_{j,h})\right\}$$

$$C_1 = \left\lfloor \frac{T-t+1}{a_j} \right\rfloor; R_1 = (T-t+1).MOD.(a_j)$$

Summarising, the rules to eliminate candidates implemented in the procedure with backtracking are:

Rule-1: If $a^-_{ijt} > a_j$, then the type i is rejected as candidate for the t-th position in the sequence.

Rule-2: If $a^+_{jt} > K_{ijt}$, then the type i is rejected as candidate for the t-th position in the sequence.

Finally, if level 0 in the graph is reached during the backtracking and there is no choice to explore, the problem has no feasible solution.

6 Aplications

A soft package in Visual Basic, ROSINA, has been developed [25], which includes **A1** and a procedure based on *BDP* [23]. The sequence shown in Figure 3, obtained with **A1**, lasts less than 1 second in a PentiumII 233 MHz.

That sequence satisfies all the constraints and the objective function is *SDQ* = 51.61, while for the sequence shown in Figure 2 was 2443.07. Figure 3 shows a better distribution for components or options along the sequence compared with Figure 2, and reaches a better distribution for workloads

	p01	p02	p03	p04
Option.1	*			
Option.2		*		
Option.3			*	
Option.4				*
Prod. plan	24	32	24	16

Option.1: No more than 1 out of 4 units
Option.2: No more than 1 out of 2 units
Option.3: No more than 1 out of 4 units
Option.4: No more than 1 out of 3 units

Figure 4: Instance presented as Example 2

The backtracking procedure in **A1** is not necessary for Example 1; nevertheless, it is different with other instances, like Example 2, which corresponds to a CPRV problem, T=96, P=4, $n_{j,i}$=1 (i=j) and $n_{j,i}$=0 (i≠j), with the scheme presented in Figure 4.

In spite of the simplicity of Example 2, **A1** reaches the stage 95 (with a pending unit of p03) and backtracking is necessary to obtain the solution presented in Figure 5.

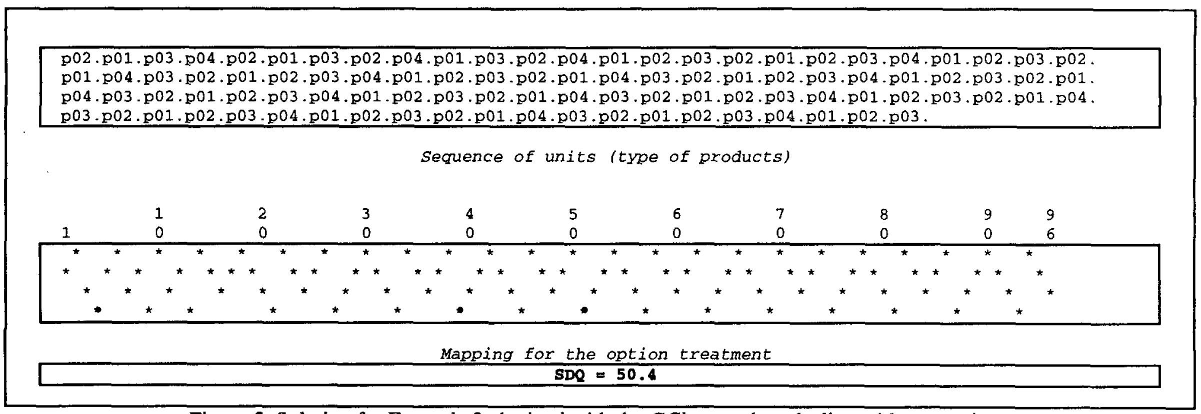

Figure 5: Solution for Example 2 obtained with the *GCh* procedure dealing with constraints.

7 Conclusions

This work presents a kind of problem found sequencing mixed units in assembly lines in which consumption or use of resources to elaborate the final products are subjected to a set of constraints. Each constraint fixes the maximal load for an option in any segment or subsequence of a given length. This way to tackle the problem is closer to problems that managers face up in car factories. They assume their problem closer to a CPRV o CORV problem (even CO) than a PRV o ORV problem.

The constraints considered can reflect different kinds of limitations: physical (impossible to place in a workstation more than a certain number of identical units), operational (impossible to do certain tasks in a higher frequency allowed by the production system), etc. Nevertheless, they can be also useful for sequences with a set of desirable properties in JIT context, such that regularity in consumption of resources or in producing types or varieties of a product.

The use of an objective function and the load constraints as a whole permits a characterisation, both quantitative and qualitative, for regularity. Moreover, it represents a kind of problem (CORV), which combines balancing in consumption or use of resources respect to an ideal prefixed rate with compatibility for loads.

We have proposed a heuristic algorithm to solve the CORV problem, which is also valid to solve the ORV, CO, CPRV, PRV and CP problems. We have used a criterion based on the quadratic distance between real and ideal levels of consumption in production (*SDQ*); nevertheless, all the procedures used can be adapted to other kinds of criteria, as those based on Euclidean (*SDE*) and rectangular distance (*SDR*), among others.

References

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

[2] Bautista, J., Companys, R. and Corominas, A. (1996), "Una visi6n sobre secuencias regulares", *Boletín SEIO* 12 (6), 2-3.

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

[4] Inman, R. R. and Bulfin, R. L. (1991), "Sequencing JIT mixed.model assembly lines", *Mgmt. Sci.* 37 (7), 901-904

[5] Steiner, G. and Yeomans, S. (1993), "Level schedules for mixed-model, just-in-time assembly processes", *Mgt. Sci.* 36 (6), 728-735

[6] Ding, F.-Y. and Cheng, L. (1993), "An effective mixed-model assembly lines sequencing heuristic for just-in-time production systems", J. *Opns. Mgmt.* 11 (1), 45-50

[7] Bautista, J., Companys, R. and Corominas, A. (1995), *Seqüenciació d'unitats en context JIT*, TOE 9, Edicions UPC

[8] Cheng, L. and Ding, F.-Y. (1996), "Modifying mixed-model assembly line sequencing methods to consider weigthed variations for just-in-time production systems", *IIE Transactions* 28, 919-927

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

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

[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.* 37 (1), 121-122

[12] Miltenburg, J., Steiner, G. and Yeomans, S. (1990), "A dynamic programming algorithm for scheduling mixed-model, just-in-time production systems", *Mathl. Comput. Modelling* 13 (3), 57-66

[13] Kubiak, W. and Sethi, S. (1994), "Optimal just-in-time schedules for flexible transfer lines", *J. Flexible Manufacturing Systems* 6, 137-154

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

[15] Miltenburg, J. and Sinnamon, G. (1989), "Scheduling mixed-model multi-level just-in-time production systems", *Int. J. Prod. Res.* 27, 1487-1509

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

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

[18] Duplaga, E. A., Hahn, C. K. and Hur, D. (1996), "Mixed-model assembly line sequencing at Hyundai Motor Company", *Prod. & Inv. Mgmt. J.* 37 (3), 20-26.

[19] Steiner, G. and Yeomans, J. S. (1996), "Optimal level schedules in mixed-model, multi-level JIT assembly systems with pegging", *Eur. J. Op. Res.* 95, 38-52

[20] Yano, C. A. and Rachamadugu, R. (1991), "Sequencing to minimize work overload in assembly lines with product options", *Mgmt. Sci.* 37 (5), 572-586

[21] Tsai, L.- H. (1995), "Mixed-model Sequencing to Minimize Utility Work and the Risk of Conveyor Stoppage", *Mgmt. Sci.* 41 (3), 485-495

[22] Dincbas, M., Simonis, H. and Van Hentenryck, P. (1988), "Solving the Car-Sequencing Problem in Constraint Logic Programming", *Proceedings of the European Conference on Artificial Intelligence* (ECAI-88), 290-295

[23] Bautista, J., Companys, R. and Corominas, A. (1992a), *Introducción al BDP*, D.I.T. 92/04, DOE, ETSEIB-UPC

[24] Bautista, J., Companys, R. and Corominas, A. (1994), *Modelos y algoritmos para la determinación de secuencias regulares en líneas de montaje mixtas con restricciones en la elaboración de productos*, D.I.T. 94/21, DOE, ETSEIB-UPC

[25] Bautista, J., Companys, R. And Corominas, A. (1994), *Rosina Demo*, DOE, ETSEIB-UPC.

[26] Little, J. (1993), "A searching technique", OR Insight, 6 (4), 24-31.