Proceedings of
2009 IEEE International Symposium on Assembly and Manufacturing
17-20 November 2009, Suwon, Korea

Mixed-model sequencing problem with overload minimization considering workstations dependencies

Joaquín Bautista and Raúl Suárez

***Abstract*— This paper reviews the formulation of the Mixed Model Sequencing Problem with Workload minimization (MMSP-W). Two significative models already presented in the literature are describe, showing that they are valid for the case of parallel workstations, but do not properly solve the case of serial workstations. After that, a new model is introduced that is valid for the case of serial workstations. An example is used to illustrate the performance of all the models, and a computational experience was done to verify the applicability of the proposed model using the solver CPLEX and a set of problem instances of small dimension adapted from the literature.**

I. Introduction

Properly sequencing the products is a key point in large number of productive processes with manufacturing lines, particularly when the Just-in-Time and Douki-Seisan concepts are applied. There are several problems related with sequencing, ranging, for instance, from particular problems like the determination of the optimal sequence of robotized part feeding in a set of paralell machines [14] to the determination of an optimal sequence of mixed products in a manufacturing line, which is the topic of this work.

Assembly lines of mixed products allow the manufacturing of a number of different products, avoiding stock related problems. This type of lines can be frequently find in car manufacturing factories, and the proper sequence of products in the line is a relevant point when looking for the optimal efficiency of the line. There are different efficiency measures proposed in the literature, depending basically on the administrative policies of the company [11], [10], [1], [9] [3]. One of these criteria is the minimization of the overload, which is a measurement, in units of time, of the remaining work on a product that is not done in a particular workstation. This overload may appear when the needed time to finish the work on given product is larger than the assigned cycle time [18], which is an average of the time needed by the different products manufactured in the line; thus, if several products with high processing time are consecutively fed in the line at some point one workstation will not have time enough to finish its expected work and, either the line is stopped to allow the workers finishing the product before it leaves the workstation (called conveyor stoppage [16]), or the product exits the workstation partially manufactured. The amount of work not done on a product in a workstation is called with different names: work overload [18] [2], remaining work [4], or utility work [15].

Minimizing the overload is a NP-hard problem [18], and different approaches have been proposed to solve it, like: exact procedures based on branch and bound [6] and dynamic programming [18]. Other procedures are based on a local search [17], greedy with priority rules [5], meta-heuristics [13], or based on beam search [8]. There are also procedures that consider multi-criteria [9], [12]. A recent review of the literature related with the sequencing problem can be found in [7].

Joaquín Bautista is with the Nissan Chair, Technical University of Catalonia (UPC), Barcelona, Spain *(joaquin.bautista@upc.edu)*.

Raúl Suárez is with the Institute of Industrial and Control Engineering (IOC), Technical University of Catalonia (UPC), Barcelona, Spain *(raul.suarez@upc.edu)*

This work was partially supported by the Spanish Government through the projects PROTHIUS-II DPI2007-63026, DPI2007-63665 and DPI2008-02448, including EDRF funding. Thanks also to the UPC Nissan Chair.

II. Existing Models

In order to solve the sequencing problem of mixed products in a production line with multiple workstations minimizing the work overload, we take as reference two models already presented in the literature, namely Model M1, introduced by Yano and Rachamadugu [18], and Model M2, introduced by Scholl, Klein and Domschke [13]. These two models are described in the following subsections.

A. Model M1

The goal of this model is the maximization of the profit of the performed work. The involved parameters are:

- K: number of workstations.
- I: number of different products.
- d_i: demand of product i.
- $p_{i,k}$: processing time required by product i in the workstation k.
- T: number of products to be sequentially processed (i.e. $\sum_{i=1}^{I} d_i = T$).
- b_k: profit per time of the work done in the workstation k.
- c: cycle time (i.e. time assigned to the workstations to process any product).
- $L_k > c$: maximum time that the workstation k is allowed to work on any product.

And the involved variables are:

- $x_{i,t}$: binary variable equal to 1 when a product i is assigned to position t in the sequence, and equal 0 otherwise.
- $s_{k,t}$: start time of the work done in the workstation k on the t-th product (i.e. t-th element in the sequence) .

978-1-4244-4628-5/09/$25.00 ©2009 IEEE

$v_{k,t}$: processing time really used to work in the workstation k on the t-th product.

Using these parameters and variables and calling V to the total profit of performed work, we can write the following mathematical programming problem,

Problem M1:

$$\text{Max } V = \sum_{k=1}^{K} \left(b_k \sum_{t=1}^{T} v_{k,t} \right) \quad (1)$$

subject to:

$$\sum_{t=1}^{T} x_{i,t} = d_i \quad i = 1, ..., I \quad (2)$$

$$\sum_{i=1}^{I} x_{i,t} = 1 \quad t = 1, ..., T \quad (3)$$

$$v_{k,t} \leq \sum_{i=1}^{I} p_{i,k}\, x_{i,t} \quad k = 1, ..., K \quad t = 1, ..., T \quad (4)$$

$$s_{k,t} \geq (t-1)c \quad k = 1, ..., K \quad t = 1, ..., T \quad (5)$$

$$s_{k,t} \geq s_{k,t-1} + v_{k,t-1} \quad k = 1, ..., K \quad t = 2, ..., T \quad (6)$$

$$s_{k,t} + v_{k,t} \leq (t-1)c + L_k \quad k = 1, ..., K \quad t = 1, ..., T \quad (7)$$

$$v_{k,t} \geq 0; s_{k,t} \geq 0 \quad k = 1, ..., K \quad t = 1, ..., T \quad (8)$$

$$x_{i,t} \in \{0, 1\} \quad i = 1, ..., I \quad t = 1, ..., T \quad (9)$$

Taking as start time reference for the operations $s_{k,1} = 0\ \forall k$, the minimum start time for each operation is $s_{k,1}^{min} = (t-1)c\ \forall k$. Constraints (2) impose the satisfaction of the demand, constraints (3) indicate that in each position of the sequence only one product unit can be assigned, constraints (4) indicate that the processing time really used to work on each product unit is constrained by its required processing time, constraints (5), (6) and (7) set the limits for the operation start times and for the processing times for each product, constraints (8) indicate that the processing times and the start times are non negatives, and, finally, constraints (9) indicate the binary condition of the assignation variables.

Note that for $b_k = 1\ \ \forall k$ maximizing V is equivalent to maximize the completed work, and therefore equivalent to minimize the overload.

B. Model M2

The goal of this model is the minimization of the total overload W. Additional variables needed in this model are:

$\hat{s}_{k,t}$: difference between the actual start time $s_{k,t}$ and the minimum start time $s_{k,t}^{min}$ of the t-th operation in the workstation k (i.e. $\hat{s}_{k,t} = s_{k,t} - (t-1)c$).

$w_{k,t}$: overload time generated by the t-th product of the sequence in the workstation k.

$\rho_{k,t}$: processing time of the work on the t-th product in the sequence in the workstation k.

Now we can write the following mathematical programming problem,

Problem M2:

$$\text{Min } W = \sum_{k=1}^{K} \sum_{t=1}^{T} w_{k,t} \quad (10)$$

subject to:

$$\sum_{t=1}^{T} x_{i,t} = d_i \quad i = 1, ..., I \quad (11)$$

$$\sum_{i=1}^{I} x_{i,t} = 1 \quad t = 1, ..., T \quad (12)$$

$$\rho_{k,t} = \sum_{i=1}^{I} p_{i,k}\, x_{i,t} \quad k = 1, ..., K \quad t = 1, ..., T \quad (13)$$

$$\hat{s}_{k,t+1} \geq \hat{s}_{k,t} + \rho_{k,t} - w_{k,t} - c \quad k = 1, ..., K \quad t = 1, ..., T \quad (14)$$

$$\hat{s}_{k,t} + \rho_{k,t} - w_{k,t} \leq L_k \quad k = 1, ..., K \quad t = 1, ..., T \quad (15)$$

$$\hat{s}_{k,t} \geq 0; w_{k,t} \geq 0 \quad k = 1, ..., K \quad t = 1, ..., T \quad (16)$$

$$\hat{s}_{k,1} = 0; \hat{s}_{k,T+1} = 0 \quad k = 1, ..., K \quad (17)$$

$$x_{i,t} \in \{0, 1\} \quad i = 1, ..., I \quad t = 1, ..., T \quad (18)$$

The objective function (10) represents the minimization of the total work overload, constraints (13) link the processing time required by each product type with the processing time required by the units in the sequence, and, finally, constraints (14), (15), (16) and (17) set the lower and upper limits of $\hat{s}_{k,t}$ as well as their relation with required processing times and the overloads.

III. Example

The following example illustrates the performance of the models presented above and the potential points for improvements.

Consider that there are 6 products to be manufactured ($T = 6$), 3 of them of the type A, 1 of type B and 2 of type C. All the products are manufactured in 3 workstations m_1, m_2 and m_3 (i.e. $K = 3$), with the processing times $p_{i,k}$ for each one given in Table I. Note that, according to the demand of each product and the total processing time per product, the total amount of work is equivalent to 77 units of time. With this information the cycle time was set to $c = 4$ and the maximum working window set to $L_k = 6, \forall k$.

The optimal solution obtained using the model M1 and considering $b_k = 1, \forall k$ is shown in Figure 1. The sequence of products with maximum completed work is: A-C-B-A-C-A. It can be seen that the total amount of performed work is $V = 76$, and the total overload is $W = 1$ (the overload is produced in the workstation m_2 when the second product, of type C, is being manufactured).

The optimal solution obtained using the model M2 is shown in Figure 2. The sequence of products with maximum completed work is: A-A-A-C-C-B. It can be seen that the

TABLE I

PROCESSING TIMES $p_{i,k}$ FOR EACH PRODUCT TYPE IN THE WORKSTATION m_i, $i = 1, ..., 3$

	A	B	C
m_1	5	4	3
m_2	5	4	4
m_3	4	3	5
total time	14	11	12

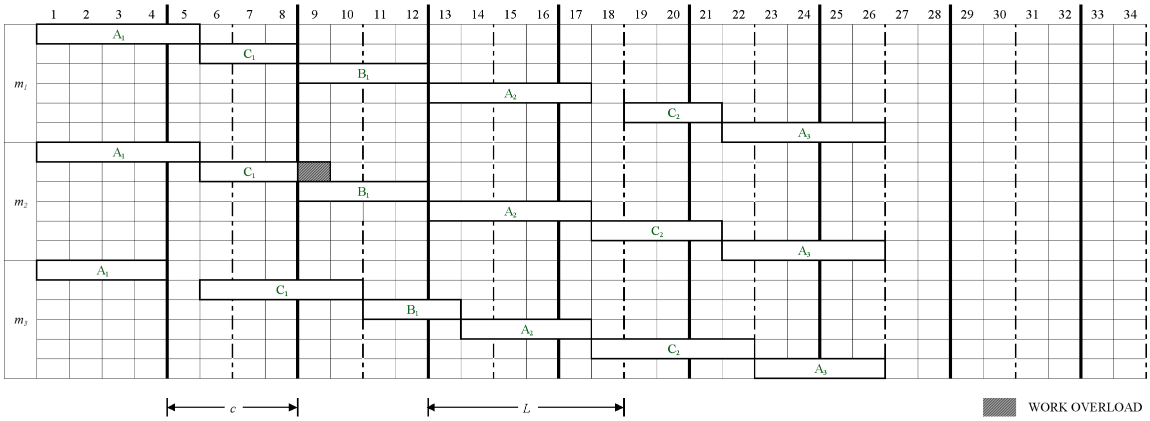

Fig. 1. Optimal solution for the example using model M1.

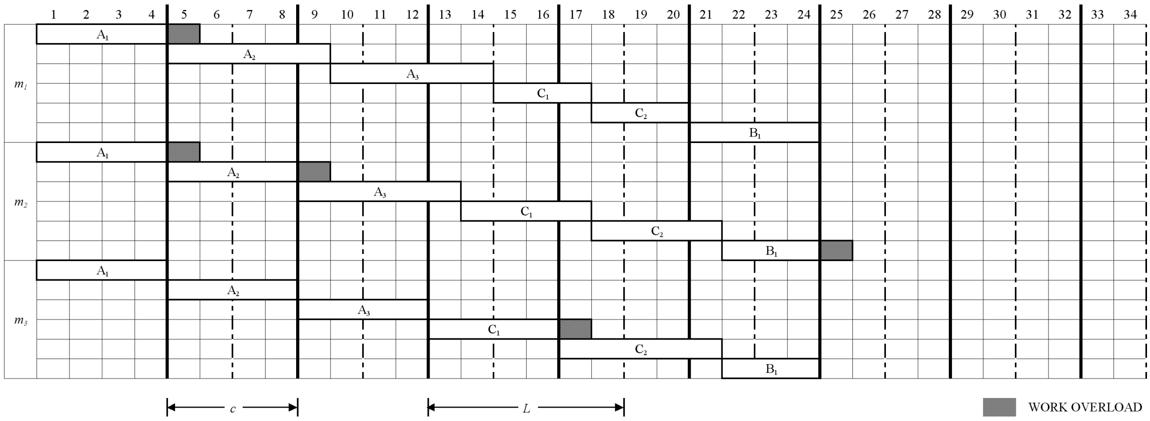

Fig. 2. Optimal solution for the example using model M2.

total amount of performed work is $V = 72$, and the total overload is $W = 5$.

In the Model M1, the start time of each operation in each workstation is referred to an absolute time (for each station), while in the model M2, each new cycle means a reset and thus a reference $t = 0$. In order to simplify the comparison of both approaches, Figures 1 and 2 represent the results using the time definition of the model M1.

Although both models look for an equivalent objective, there is an evident difference in the overload. This is justified because M1 considers the possibility of using for the last operation of each workstation the whole window (L_k), while M2 gives to the last operation of each workstation just the cycle time c. It is easy to verify that without this constraint the model M2 also arrives to an overload $W = 1$.

IV. Proposed Model with Bonds between Serial Workstations

A. Motivation

The models M1 and M2 do not consider any type of bond between workstations, then those models are valid only if a set of parallel lines are considered, such that they have to process a sequence of components that are going to be assembled in a common final workstation into a single product, and the goal is finding a unique sequence for the components (i.e. the components of each final product are in the same sequence in all the parallel lines) that minimizes the whole process overload. Despite the validity of these models for this type of problem, when the production line includes only serial workstations linked by a transfer system with constant velocity and without intermediate buffers, it is necessary to include some additional constraints to these models. In this type of lines, any workstation (with exception of the first one) can start working on a product (with exception of the first one) only when the previous operation in the workstation is considered finished and a new product comes from the previous workstation (even when the current product in the workstation is not already finished). In these conditions, considering that the start time of the operations on the same product are delayed one cycle between any two contiguous workstations, the solutions obtained with the models M1 and M2 may be incoherent. This effect is illustrated in Figures 3 and 4 that show, respectively, the application of the models M1 and M2. For instance, in Figure 3 there are incoherences related with the products A1 and A2 in their transitions between workstations m_1 and m_2, and in Figure 4 there is an incoherences related with the product A2 in its transition between workstations m_1

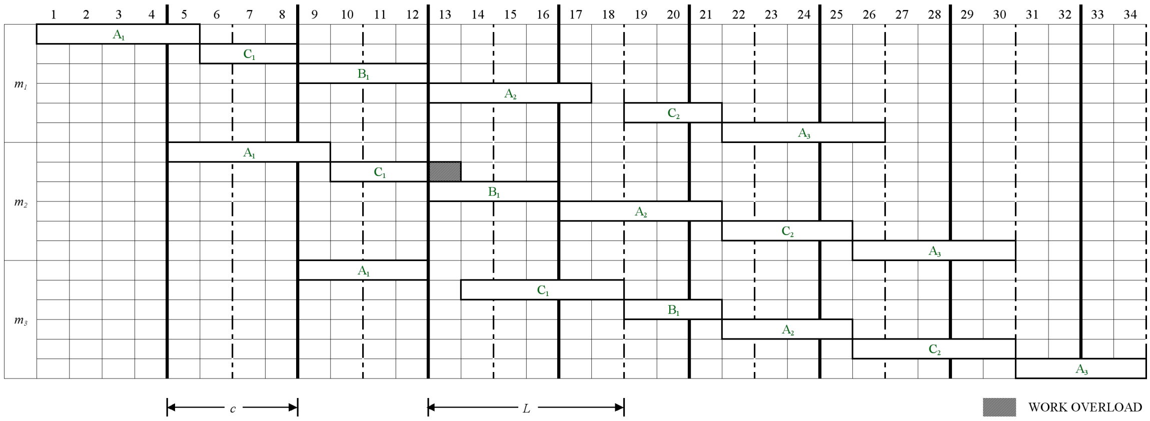

Fig. 3. Optimal solution for the example using model M1 with respect to absolute time.

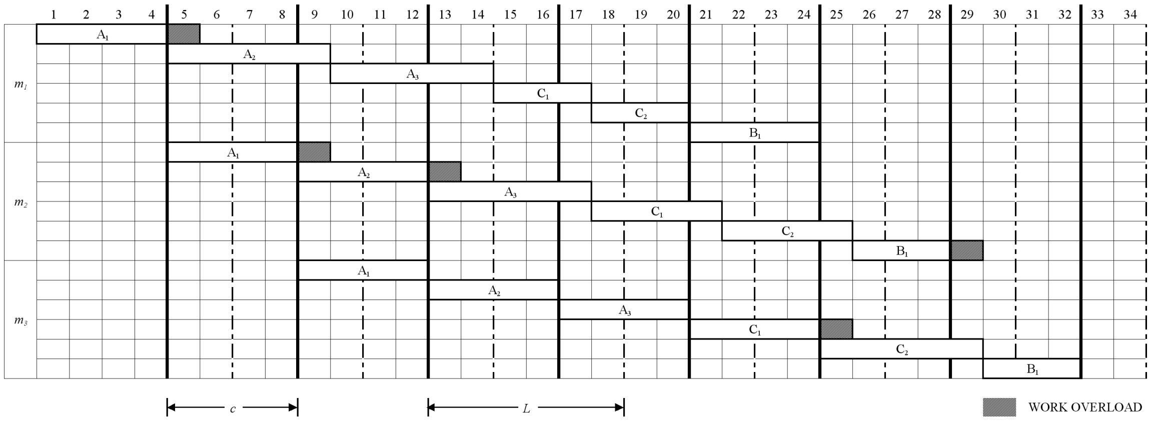

Fig. 4. Optimal solution for the example using model M2 with respect to absolute time.

and m_2, and another one related with the product C1 in its transition between workstations m_2 and m_3.

In order to solve the problem illustrated in the example above, we propose a new model that considers the time constraints generated by consecutive workstations. This new model, described in next subsection, is a variation of the model M1 described in Subsection II-A.

B. Model M3

Using the same parameters and variables introduced for the model M1, we can write the following mathematical programming problem,

Problem M3:

$$\text{Max } V = \sum_{k=1}^{K} \left(b_k \sum_{t=1}^{T} v_{k,t} \right) \tag{19}$$

which is equivalent to (1) but, now, subject to:

$$\sum_{t=1}^{T} x_{i,t} = d_i \quad i = 1, ..., I \tag{20}$$

$$\sum_{i=1}^{I} x_{i,t} = 1 \quad t = 1, ..., T \tag{21}$$

$$v_{k,t} \leq \sum_{i=1}^{I} p_{i,k}\, x_{i,t} \quad k = 1, ..., K \quad t = 1, ..., T \tag{22}$$

$$s_{k,t} \geq (t + k - 2)c \quad k = 1, ..., K \quad t = 1, ..., T \tag{23}$$

$$s_{k,t} \geq s_{k,t-1} + v_{k,t-1} \quad k = 1, ..., K \quad t = 2, ..., T \tag{24}$$

$$s_{k,t} \geq s_{k-1,t} + v_{k-1,t} \quad k = 2, ..., K \quad t = 1, ..., T \tag{25}$$

$$s_{k,t} + v_{k,t} \leq (t + k - 2)c + L_k \quad k = 1, ..., K \quad t = 1, ..., T \tag{26}$$

$$v_{k,t} \geq 0; s_{k,t} \geq 0 \quad k = 1, ..., K \quad t = 1, ..., T \tag{27}$$

$$x_{i,t} \in \{0, 1\} \quad i = 1, ..., I \quad t = 1, ..., T \tag{28}$$

Taking as start time reference for the operations $s_{1,1} = 0$, the minimum start time for each operation in the model M3 is $s_{k,t}^{min} = (t + k - 2)c \ \forall k, \forall t$.

Note that in M3, the constraints (23) and (26), replace the original constraints (5) and (7), respectively. These new constraints include the proper delays in the start working times in each workstation according to their positions in the manufacturing line. On the other hand, constraints (25) force that no workstation can start the work before the product has really left the previous workstation.

Figure 5 illustrates the result of applying model M3 to the example in Section III with the same unitary profit for the work done in each the workstations (i.e. $b_k = 1, \forall k$). The sequence of products with maximum completed work

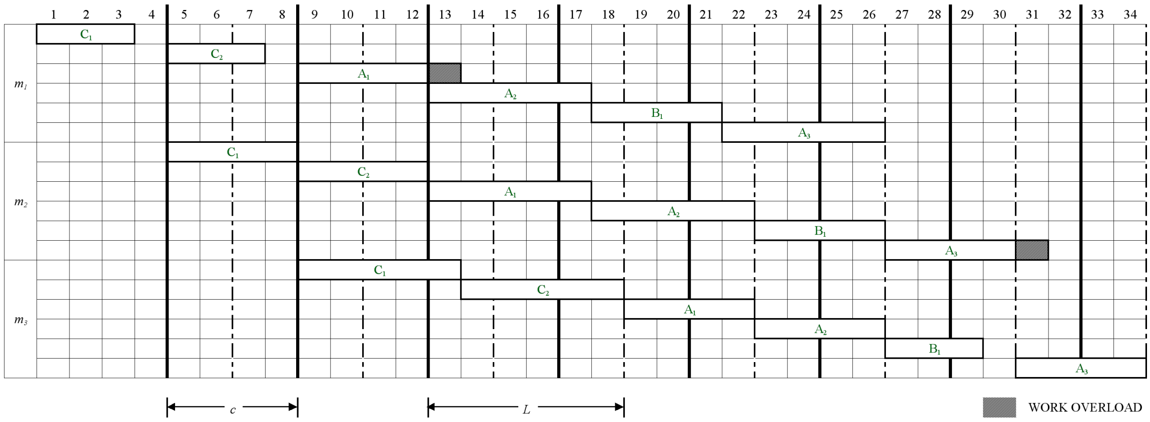

Fig. 5. Optimal solution for the example using model M3 with respect to absolute time.

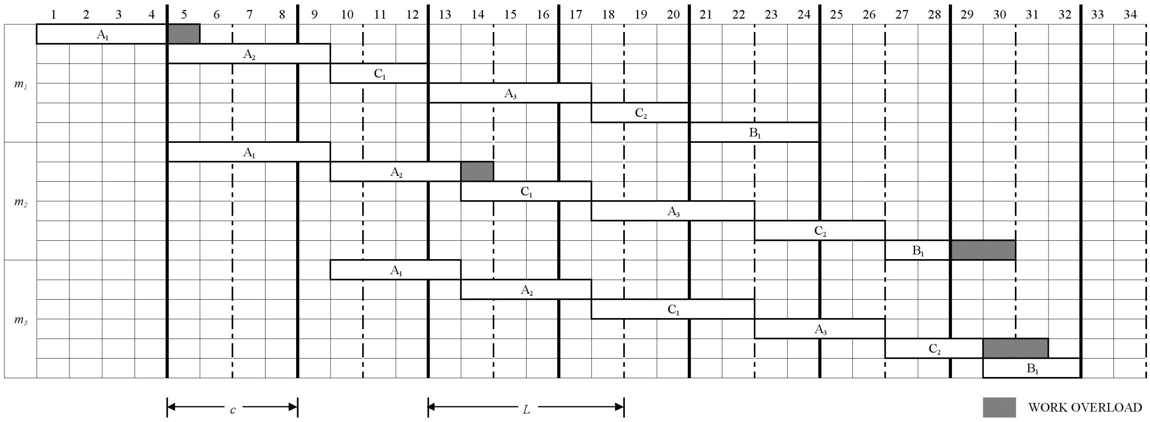

Fig. 6. Optimal solution for the example using model M3 and the constraints in the last operation of each workstation (with respect to absolute time).

is: C-C-A-A-B-A, which is different to that obtained using models M1 or M2.

If a constraint to the window working time of the last operation of all the workstations is added in the model M3 imposing that the work of each workstations has to be finished within the nominal cycle time (as it is also considered in the model M2), i.e.

$$s_{k,T} + v_{k,T} \leq (T + k - 1)c \quad k = 1, ..., K \tag{29}$$

then the optimal solution changes, as it is illustrated in Figure 6 for the considered example.

V. Computational Results

In order to validate the proposed model M3, we use 225 problem instances (similar to those presented in [2]) built from the 45 production programs and the 5 structures of processing times shown in Tables II and III. The 45 production programs has been grouped into 5 blocks with the following properties: B1) one product with significant higher demand; B2) half of the products with significant higher demand; B3) all the products with similar demand; B4) one product with significant smaller demand; and B5) all the products with different demands (regularly distributed) . The 5 structures of processing times are characterized with the following properties: S1) similar processing times and close to the cycle time; S2) half of the processing times are close to the allowed window time; S3) half of the products have high processing times in the first workstations and the other half in the last workstations, S4) half of the products have high processing times and the other half have short processing times in all the workstations; and S5) each product has a high processing time in a different workstations.

The optimal solution for each of the 225 instances, according to the model M3, was found using SOLVER CPLEX V11.0 with a single processor license running in a computer MacPro with CPU Intel Xeon 3 GHz and 2 Gb RAM under Windows XP. Table IV shows the results.

The following comments regarding the CPU time can be derived from the results in Table IV: 1) the average time of the 225 instances was 32.92 s; 2) the actual range of time goes from 0.03 to 1336.69 s; 3) the block B1 has the minimal average time, 0.31 s, and the block B3 the highest one, 111.13 s, showing that the instances with large demand of one product are solved easier, while the instances with similar demand of different product are more difficult to be solved; 4) the structure S4 (half of the products have high processing times and the other half have short processing times in all the workstations) has the highest

TABLE II

DATA FOR EXPERIMENTAL VALIDATION: SET OF PRODUCTION PROGRAMS GROUPED INTO 5 BLOCKS

i	Block 1				Block 2						Block 3							Block 4			
	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}	P_{15}	P_{16}	P_{17}	P_{18}	P_{19}	P_{20}	P_{21}
d_1	13	1	1	1	7	7	7	1	1	1	5	5	5	3	3	3	4	5	5	5	1
d_2	1	13	1	1	7	1	1	7	7	1	5	3	3	5	3	3	4	5	5	1	5
d_3	1	1	13	1	1	7	1	7	1	7	3	5	3	5	5	4	4	5	1	5	5
d_4	1	1	1	13	1	1	7	1	7	7	3	3	5	3	5	4	4	1	5	5	5

i	Block 5																							
	P_{22}	P_{23}	P_{24}	P_{25}	P_{26}	P_{27}	P_{28}	P_{29}	P_{30}	P_{31}	P_{32}	P_{33}	P_{34}	P_{35}	P_{36}	P_{37}	P_{38}	P_{39}	P_{40}	P_{41}	P_{42}	P_{43}	P_{44}	P_{45}
d_1	1	1	1	1	1	1	3	3	3	3	3	3	5	5	5	5	5	5	7	7	7	7	7	7
d_2	3	3	5	5	7	7	1	1	5	5	7	7	1	1	3	3	7	7	1	1	3	3	5	5
d_3	5	7	3	7	3	5	5	7	1	7	1	5	3	7	1	7	1	3	3	5	1	5	1	3
d_4	7	5	7	3	5	3	7	5	7	1	5	1	7	3	7	1	3	1	5	3	5	1	3	1

TABLE III

DATA FOR EXPERIMENTAL VALIDATION: 5 STRUCTURES OF PROCESSING TIMES PER PRODUCT AND WORKSTATION

i	Processing times for products in stations																			
	Structure 1				Structure 2				Structure 3				Structure 4				Structure 5			
	m_1	m_2	m_3	m_4	m_1	m_2	m_3	m_4	m_1	m_2	m_3	m_4	m_1	m_2	m_3	m_4	m_1	m_2	m_3	m_4
p_1	92	103	101	95	91	120	90	100	111	120	85	82	113	119	115	116	115	99	104	96
p_2	97	98	105	104	80	105	113	107	114	113	100	94	114	113	112	118	104	119	100	102
p_3	103	104	99	96	107	88	117	86	83	85	115	119	82	85	84	87	89	98	114	87
p_4	108	95	95	105	114	87	100	114	98	87	110	115	95	87	94	81	95	87	85	118
L_k	108	105	106	106	115	120	120	115	115	120	115	120	115	120	115	120	115	120	115	118
b_k	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

average time, 88.39 s, while the other structures have similar and significantly smaller average times, 19.05 s.

VI. CONCLUSIONS

By means of an example, it was shown that existing models to minimize the total overload in a manufacturing line are valid for synchronized parallel lines, but when serial workstations are considered they present some incoherences and the solutions cannot be used in practice. In order to solve this problem a new model was presented as a extension of that introduced in [18]. The performance of the new model was validated through a computational experience, using a set of 225 instances and using the solver CPLEX. The computational times required to find the optimal solutions are acceptable for very small dimension problems.

REFERENCES

[1] J. Bautista, R. Companys and A. Corominas (1996). Heuristics and exact algorithms for solving the Monden problem. European Journal of Operational Research, vol. 88/1, 1996, pp. 101-113.

[2] J. Bautista and J. Cano (2008). Minimizing work overload in mixed-model assembly lines. International Journal of Production Economics, vol. 112, no. 1, 2008, pp. 177-191.

[3] J. Bautista, J. Pereira, and B. Adenso-Díaz (2008). A Beam Search approach for the optimization version of the Car Sequencing Problem. Annals of Operations Research, vol. 159/1, Springer, 233-244.

[4] A. Bolat (1997). Efficient methods for sequencing minimum job sets on mixed model assembly lines. Naval Research Logistics, vol. 44, no. 5, pp. 419-437, 1997.

[5] A. Bolat and C. A. Yano (1992). Scheduling algorithms to minimize utility work at a single station on paced assembly line. Production Planning and Control, vol. 3, no. 4, pp. 393-405, 1992.

[6] A. Bolat (2003). A mathematical model for sequencing mixed models with due dates. International Journal of Production Research, vol. 41, no. 5, pp. 897-918, 2003.

[7] N. Boysen, M. Fliedner and A. Scholl (2009). Sequencing mixed-model assembly lines: Survey, classification and model critique. European Journal of Operational Research, vol. 192, no. 2, pp. 349-373, 2009.

[8] E. Erel, Y. Gocgunz and I. Sabuncuoglu (2007). Mixed-model assembly line sequencing using beam search. International Journal of Production Research, vol. 45, no. 22, pp. 5265-5284, 2007.

[9] S. Kotani, T. Ito and K. Ohno (2004). Sequencing problem for a mixed-model assembly line in the Toyota production system. International Journal of Production Research, vol. 42, no. 23, 2004, pp. 4955-4974.

[10] Y. Monden (1983). Toyota Production System, Norcross, GA, 1983.

[11] K. Okamura and H. Yamashina (1979). A heuristic algorithm for the assembly line model-mix sequencing problem to minimize the risk of stopping the conveyor. International Journal of Production Research, vol. 17, no. 3, 1979,pp. 233-247.

[12] A. Rahimi-Vahed and A. H. Mirzaei (2007). A hybrid multi-objective shuffled frog-leaping algorithms for a mixed-model assembly line sequencing problem. Computers & Industrial Engineering, vol. 53, no. 4, pp. 642-666, 2007.

[13] A. Scholl, R. Klein, and W. Domschke (1998). Pattern based vocabulary building for effectively sequencing mixed-model assembly lines. Journal of Heuristics, vol. 4, no. 4, pp. 359-381, 1998.

[14] R. Suárez and J. Rosell (2005). Feeding sequence selection in a manufacturing cell with four parallel machines. Robotics and Computer-Integrated Manufacturing (ISSN: 0736-5845). Vol. 21, Issue 3, June 2005, pp. 185-195.

[15] L.H. Tsai (1995). Mixed-model sequencing to minimize utility work and the risk of conveyor stoppage. Management Science, vol. 41, no. 3, pp. 485-495, 1995.

[16] Z. Xiaobo and K. Ohno (1997). Algorithms for sequencing mixed models on assembly line in a JIT production system. Computers & Industrial Engineering, vol. 31, no. 1, pp. 47-56, 1997.

[17] C. A. Yano and A. Bolat (1989). Survey, development, and application of algorithms for sequencing paced assembly lines, Journal of Manufacturing and Operations Management, vol. 2, no. 3, pp. 172-198, 1989.

[18] C. A. Yano and R. Rachamadugu (1991). Sequencing to minimize work overload in assembly lines with product options. Management Science, V.37, N.5, May 1991, pp.572-586.

TABLE IV

OPTIMAL RESULTS OF APPLYING THE MODEL M3 TO THE 225 PROBLEM INSTANCES

Instance	V_0	*V*	*W*	*CPU*	*Instance*	V_0	*V*	*W*	*CPU*	*Instance*	V_0	*V*	*W*	*CPU*
1/1	6292	6252	40	0.08	16/1	5605	5573	32	16.63	31/1	6410	6358	52	85.00
1/2	6431	6201	230	0.27	16/2	5670	5553	117	17.84	31/2	6429	6276	153	6.42
1/3	6407	6038	369	0.11	16/3	5705	5509	196	5.63	31/3	6523	6278	245	27.77
1/4	7171	6461	710	0.03	16/4	5540	5300	240	21.19	31/4	6397	6083	314	258.34
1/5	6580	6369	211	0.31	16/5	5609	5544	65	4.61	31/5	6468	6320	148	9.64
2/1	6448	6348	100	0.14	17/1	6400	6363	37	123.22	32/1	6418	6365	53	9.72
2/2	6479	6197	282	0.08	17/2	6476	6343	133	38.36	32/2	6511	6324	187	14.06
2/3	6683	6344	339	0.94	17/3	6524	6305	219	39.86	32/3	6593	6358	235	23.64
2/4	7099	6461	638	0.03	17/4	6460	6140	320	544.50	32/4	6711	6308	403	27.31
2/5	6712	6436	276	1.17	17/5	6448	6364	84	7.13	32/5	6530	6354	176	27.84
3/1	6424	6343	81	0.95	18/1	6388	6343	45	42.02	33/1	6414	6361	53	60.55
3/2	6395	6125	270	0.06	18/2	6435	6299	136	5.14	33/2	6443	6269	174	7.81
3/3	6455	6034	421	0.09	18/3	6515	6286	229	7.45	33/3	6561	6318	243	10.45
3/4	5671	5591	80	0.14	18/4	6647	6232	415	21.27	33/4	6635	6232	403	33.97
3/5	6268	6102	166	0.17	18/5	6520	6362	158	10.39	33/5	6542	6371	171	8.98
4/1	6436	6293	143	0.03	19/1	6392	6359	33	21.81	34/1	6386	6361	25	3.44
4/2	6599	6275	324	1.09	19/2	6503	6353	150	5.56	34/2	6509	6368	141	6.83
4/3	6551	6238	313	0.16	19/3	6547	6330	217	8.64	34/3	6487	6279	208	5.59
4/4	5899	5819	80	0.19	19/4	6723	6308	415	23.50	34/4	6285	6033	252	41.33
4/5	6232	6020	212	0.19	19/5	6508	6357	151	49.49	34/5	6354	6263	91	28.67
5/1	6370	6324	46	6.02	20/1	6384	6353	31	6.94	35/1	6382	6341	41	19.16
5/2	6455	6255	200	0.41	20/2	6475	6361	114	7.06	35/2	6441	6324	117	17.25
5/3	6545	6200	345	0.16	20/3	6471	6250	221	4.47	35/3	6455	6220	235	5.36
5/4	7135	6461	674	0.03	20/4	6247	5995	252	11.48	35/4	6209	5957	252	13.75
5/5	6646	6424	222	4.11	20/5	6360	6292	68	11.64	35/5	6366	6280	86	7.98
6/1	6358	6312	46	3.84	21/1	6436	6380	56	55.42	36/1	6390	6361	29	7.80
6/2	6413	6313	100	4.23	21/2	6491	6298	193	44.19	36/2	6523	6371	152	8.11
6/3	6431	6204	227	6.83	21/3	6563	6286	277	3.47	36/3	6525	6329	196	38.83
6/4	6421	6083	338	64.73	21/4	6223	5995	228	8.94	36/4	6523	6197	326	65.83
6/5	6424	6288	136	6.30	21/5	6404	6298	106	28.69	36/5	6428	6282	146	37.73
7/1	6364	6350	14	2.22	22/1	6434	6371	63	0.14	37/1	6384	6336	48	32.99
7/2	6515	6364	151	3.05	22/2	6511	6298	213	15.75	37/2	6421	6305	116	8.02
7/3	6479	6294	185	8.77	22/3	6541	6230	311	0.52	37/3	6477	6242	235	40.92
7/4	6535	6197	338	66.95	22/4	6023	5869	154	2.23	37/4	6409	6083	326	33.53
7/5	6406	6263	143	23.36	22/5	6324	6242	82	30.28	37/5	6446	6314	132	18.66
8/1	6436	6375	61	39.42	23/1	6432	6377	55	12.61	38/1	6394	6351	43	23.20
8/2	6437	6238	199	3.56	23/2	6477	6280	197	20.13	38/2	6483	6304	179	1.30
8/3	6569	6290	279	5.69	23/3	6525	6196	329	0.73	38/3	6569	6306	263	1.70
8/4	6385	6083	302	90.17	23/4	5985	5831	154	1.70	38/4	6923	6413	510	2.89
8/5	6490	6313	177	2.91	23/5	6330	6263	67	11.77	38/5	6588	6403	185	9.53
9/1	6442	6361	81	3.63	24/1	6438	6371	67	0.25	39/1	6392	6345	47	15.98
9/2	6539	6321	218	3.09	24/2	6525	6320	205	10.67	39/2	6449	6281	168	1.94
9/3	6617	6385	232	17.33	24/3	6579	6320	259	5.97	39/3	6553	6296	257	4.11
9/4	6499	6197	302	92.13	24/4	6261	6033	228	9.47	39/4	6885	6376	509	3.89
9/5	6472	6297	175	6.56	24/5	6398	6270	128	33.69	39/5	6594	6400	194	10.67
10/1	6430	6362	68	2.64	25/1	6434	6378	56	64.47	40/1	6362	6340	22	8.91
10/2	6497	6272	225	1.67	25/2	6457	6267	190	23.81	40/2	6481	6377	104	4.48
10/3	6503	6136	367	0.14	25/3	6547	6252	295	5.13	40/3	6463	6264	199	7.70
10/4	5785	5705	80	0.16	25/4	6185	5957	228	12.48	40/4	6497	6159	338	80.24
10/5	6250	6208	42	1.78	25/5	6410	6307	103	5.80	40/5	6412	6309	103	27.33
11/1	6390	6353	37	85.75	26/1	6440	6377	63	16.74	41/1	6360	6326	34	10.31
11/2	6469	6337	132	7.50	26/2	6505	6299	206	38.72	41/2	6447	6355	92	9.89
11/3	6531	6316	215	11.70	26/3	6601	6358	243	58.59	41/3	6447	6234	213	23.17
11/4	6685	6270	415	85.19	26/4	6461	6159	302	61.86	41/4	6459	6121	338	37.09
11/5	6514	6393	121	3.94	26/5	6478	6326	152	108.53	41/5	6418	6311	107	20.88
12/1	6386	6349	37	89.03	27/1	6438	6380	58	39.36	42/1	6366	6344	22	26.09
12/2	6455	6343	112	21.14	27/2	6471	6271	200	52.39	42/2	6495	6358	137	3.02
12/3	6493	6272	221	73.55	27/3	6585	6324	261	24.05	42/3	6501	6290	211	3.36
12/4	6447	6121	326	123.20	27/4	6423	6121	302	48.27	42/4	6735	6308	427	25.84
12/5	6440	6345	95	15.48	27/5	6484	6347	137	17.11	42/5	6486	6344	142	47.58
13/1	6388	6359	29	54.67	28/1	6408	6362	46	6.97	43/1	6362	6318	44	21.05
13/2	6489	6367	122	22.11	28/2	6503	6328	175	10.89	43/2	6427	6311	116	2.89
13/3	6509	6302	207	91.34	28/3	6495	6230	265	1.02	43/3	6469	6237	232	6.88
13/4	6485	6159	326	1336.69	28/4	6035	5869	166	0.91	43/4	6659	6232	427	24.16
13/5	6434	6339	95	31.81	28/5	6302	6238	64	13.94	43/5	6498	6347	151	15.66
14/1	6412	6367	45	159.69	29/1	6406	6358	48	11.03	44/1	6368	6334	34	23.86
14/2	6463	6310	153	91.99	29/2	6469	6311	158	9.25	44/2	6475	6312	163	2.03
14/3	6539	6308	231	34.59	29/3	6479	6196	283	0.95	44/3	6523	6252	271	0.80
14/4	6435	6121	314	504.38	29/4	5997	5831	166	1.05	44/4	6935	6413	522	2.47
14/5	6462	6359	103	9.09	29/5	6308	6252	56	6.97	44/5	6566	6397	169	8.42
15/1	6410	6370	40	63.41	30/1	6416	6361	55	9.78	45/1	6366	6323	43	6.32
15/2	6483	6336	147	63.14	30/2	6531	6353	178	12.13	45/2	6441	6292	149	2.27
15/3	6517	6278	239	15.45	30/3	6571	6357	214	68.31	45/3	6507	6242	265	3.92
15/4	6235	5995	240	56.06	30/4	6511	6197	314	135.06	45/4	6897	6376	521	2.86
15/5	6382	6312	70	19.70	30/5	6450	6295	155	44.69	45/5	6572	6390	182	7.42

V : Completed work; V_0 : Total work to complete; W : Work overload; CPU : computer time to obtain the optimal solution
The instances are identified by the program number and structure number separated by "/"