Optimal Sequencing and Lot Sizing in a Multi-model Synchronous Assembly Line

Diego C. Cafaro[1], Nicolás Recaman[2], Emmanuel Almada[2], Noel Conesa[2], Hernán Luna Beckley[2], Juan Pablo García[2]

[1] INTEC (UNL-CONICET), Güemes 3450, 3000, Santa Fe, ARGENTINA
dcafaro@fiq.unl.edu.ar
[2] Sola y Brusa S.A., Ruta Prov. 6, km 57, 3009, Franck, Santa Fe, ARGENTINA
{nicolas.recaman,emmanuel.almada,noel.conesa,hernan.luna,
juan.garcia}@solaybrusa.com

Abstract. This work presents a mixed-integer nonlinear mathematical programming (MINLP) model aiming at optimizing the sum of transition and inventory carrying costs in a multi-model synchronous assembly line. Different products are assembled in the same line in runs or campaigns, whose sequence and length should be optimally determined. As products show different cycle times, transition periods are of particular interest due to the productivity reduction. The proposed model precisely accounts for transient periods, at the time it incorporates further details such as semi-elaborate stocks. It is successfully applied to a real-world case study of the argentine truck trailer industry.

Keywords. Multi-model, Synchronous Assembly Line, Lot Sizing, Sequencing.

1 Introduction

Modern assembly lines are highly efficient mass production systems comprising a number of work stations that are usually linked together by an automatic material handling mechanism. In general, flow lines can be classified according to the number of products or models produced into single model (or dedicated) and multiple model lines. Moreover, flow lines producing multiple models can be subdivided into mixed-model and multi-model lines. In mixed-model lines, the production lot size is equal to one, whereas multi-model lines cyclically launch campaigns or runs of different models, each one having a certain length [1]. As a result, the multi-model line balancing problem is particularly focused on changeover (or run launching) costs, as well as inventory carrying costs, whose simultaneous minimization yields the optimal lot size for every production run.

Assembly line balancing research has traditionally focused on single model assembly line balancing problems, which have attracted attention from operations research community for almost half of a century [2]. A considerable amount of work has been done with the aim of finding an efficient solution technique for this kind of problems, including exact methods [3,4,5], heuristics [6,7,8] and meta-heuristic procedures [9,10]. More recently, assembly line balancing research evolved towards more realistic problems by relaxing the restricting assumptions of the single-model line. Among them, mixed-model and multi-model flow lines are of particular interest of current studies due to the increasing flexibility required in modern production systems. Most of recent work has been focused on the solution of mixed-model assembly line balancing problems with specific constraints, using both exact methods [11] and heuristic approaches [12,13], while heuristic procedures have been proposed to the optimal lot sizing problem in multi-model, multi-stage flow shops [13]. However, to the best of our knowledge, no rigorous formulations have been proposed for solving the multi-model lot-sizing problem in synchronous assembly lines. Synchronous assembly lines are coordinated in such a way that product transfers take place all together, immediately after all operations are completed in all the work stations. This kind of problems is typically met in the automotive industry.

In this work we propose three different models for solving the multi-model assembly line problem, presented in order of increasing complexity. The first one is an approximated non-linear programming (NLP) approach, whose global optimal solution can be explicitly found. It can be regarded as the optimal lot-sizing problem for a single stage, multi-model production system. The second model is a non-linear mixed-integer programming (MINLP) formulation accounting for the discrete nature of some problem variables, particularly the number of units being produced at every run, also taking into consideration the number of work stations comprising the line. Finally, an extended MINLP formulation additionally deals with the part supply problem associated to the assembly line. All the models have been successfully tested in a real-world assembly line producing truck trailers for the transportation industry in Argentina.

2 Formulations of the Multi-model Lot-Sizing Problem

2.1 NLP Model for the Single-Stage, Multi-Model Production System

The single-stage, multi-model production system operation is organized in runs, each one producing a single model $i \in I$. During the production cycle (T), every model is produced through a unique run of length L_i, as stated by eq. 1.

$$\sum_{i \in I} L_i = T \tag{1}$$

Production and demand rates for all the models (p_i and r_i) are known data usually given in units per hour. As depicted in Fig. 1, the stock of models of type i at the end of every production run should be high enough to cover the demand for i while other models $i' \neq i$ are being produced. Such a relationship is mathematically expressed through eq. 2. Note that product stocks are treated as continuous variables.

$$(p_i - r_i)\, L_i = r_i\, (T - L_i) \qquad \forall i \in I \tag{2}$$

Combining eqs. 1 and 2 yields eq. 3, meaning that although demand and production rates are given, the production system presents no idle time only if eq. 3 is fulfilled.

$$\sum_{i \in I} r_i / p_i = 1 \tag{3}$$

In this simplified model, transition costs are independent of the product sequence, and are incurred every time a new run is launched. Relaxing this assumption is studied in further sections. Moreover, transient times are neglected, i.e. no transition times between subsequent production campaigns or runs are required. If ic_i is the inventory carrying cost of a single unit of the final product *i* per unit time, while ch_i is the fixed cost incurred every time a new run of product *i* is launched, the sum of inventory carrying and production launching costs per unit time is given by eq. 4.

$$z = \sum_{i \in I} ic_i \, (p_i - r_i) \, L_i \, / 2 + \sum_{i \in I} ch_i \, / T \tag{4}$$

Minimizing eq. 4 subject to eq. 2, under the non-negativity condition of variables L_i and T, yields a convex nonlinear programming problem whose exact solution can be readily obtained. The explicit solution for such an NLP formulation is given in eq. 5.

$$T^* = \sqrt{\frac{2 \sum_{i \in I} ch_i}{\sum_{i \in I} ic_i \, r_i \, (p_i - r_i) / p_i}} \quad ; \quad L_i^* = \frac{r_i}{p_i} T^* \tag{5}$$

Although being a simplified model, it constitutes a very useful tool for estimating the optimal length of production runs in multi-model assembly lines. As will be shown later, it permits to obtain tighter bounds on the variables of more rigorous models, yielding significant savings in the computational effort required by global solvers.

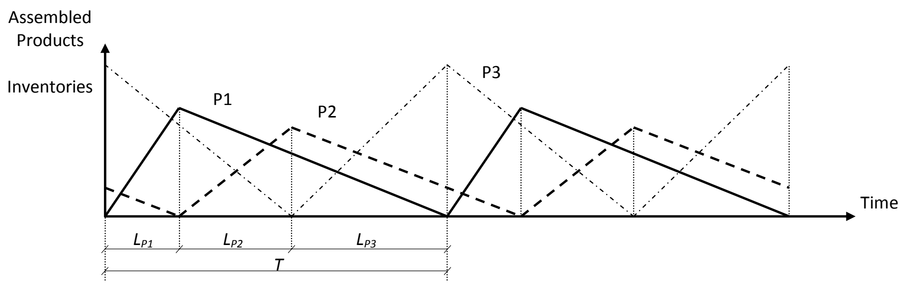

Fig. 1. Stocks of final products in a single-resource, multi-model production system

2.2 MINLP Formulation for a Multi-Model Synchronous Assembly Line

Consider a multi-model synchronous assembly line with m workstations in series. Run lengths are no longer assumed to be floating numbers L_i but integer multiples of the cycle times ct_i. In fact, the run length can be roughly estimated as the product between the cycle time and the number of models produced in the corresponding run (N_i). However, further corrections should be incorporated to the formulation to account for the transition times between two consecutive production runs i and i', with models featuring different cycle times $ct_i \neq ct_{i'}$. For instance, if a production run of model i is succeeded by a run of model i', and $ct_i < ct_{i'}$, the last (m-1) units of the model i will be produced at the cycle time $ct_{i'}$, since the synchronous transfers between adjacent stations should wait until the upstream stations finish their work on the first i'-models. As a result, idle times arise at the downstream work stations, until the last unit of i exits the assembly line. On the contrary, if $ct_i > ct_{i'}$, the first (m-1) units of i' are processed at the cycle time ct_i, and the idle time arises at the upstream stations, until the last unit of i exits the assembly line.

Production sequence. Besides the model index i, we introduce the chronologically ordered set of runs $k \in K$ to determine the most convenient production sequence. The binary variable $y_{k,i}$ will be equal to one if the model i is produced in the k-th run. Given that we are dealing with a cyclic planning problem, the first and the last elements in the set K are two parts of the same production run. Moreover, it is assumed that all the workstations are initially occupied with the same model type, and at time t = 0 the first element of a different model enters the line. In other words, run k_1 completes the last m elements of the last run in the cycle, initially filling the assembly line. Eq. 6a states that every run should be assigned to the production of a single model type, while eq. 6b stands for the relationship between the first and the last elements in the set K. To avoid symmetric solutions, the first run is arbitrarily assigned to model i_1.

$$\sum_{i \in I} y_{k,i} = 1 \qquad \forall k \in K \qquad \text{(6a)} \qquad\qquad y_{k_1,i_1} = y_{K,i_1} = 1 \qquad \text{(6b)}$$

It can be easily proved that if $|K| = |I|+1$, all the models are produced in a single run during one production cycle, and if $|K| > |I|+1$, one or more models can be produced through more than one run in the same cycle. For simplicity, we assume that $|K| = |I|+1$.

Production lot sizing. We introduce the integer variable N_k to account for the total number of units of the same kind produced during run k, while $NP_{k,i}$ (a floating, nonnegative variable) equals N_k if the model produced through run k is i, and is zero otherwise, as stated by eq.7. Note that by convention, $N_{k1} = m$.

$$NP_{k,i} \leq n_{\max} \, y_{k,i} \quad \forall k \in K, i \in I \quad ; \qquad \sum_{i \in I} NP_{k,i} = N_k \qquad \forall k \in K \qquad (7)$$

To limit changeover costs, we assume that every production run comprises a number of entities (N_k) equal or greater than the number of stations (m). In other words, only two different model types can at most be in-process at the same time in the line. Furthermore, as the line works with synchronous transfers, transition periods (i.e., the time when the line is occupied with different model types) are determined by the larger cycle time. Fig. 2 shows the transition period between models A, B and C, whose cycle times are 3, 5 and 4 h, respectively, produced by a 5-stations assembly line. Dotted arrows represent the movements of models of type B. Transfer times between subsequent workstations are included in the cycle times. At t = 0 h, station 1 starts to process the first unit of B in a run of 6 elements. Not before 5 h later, all the models in the line can be transferred to the following station, even though models of type A in stations 2 to 5 are finished 2 h earlier. This effect is critical in synchronous assembly lines.

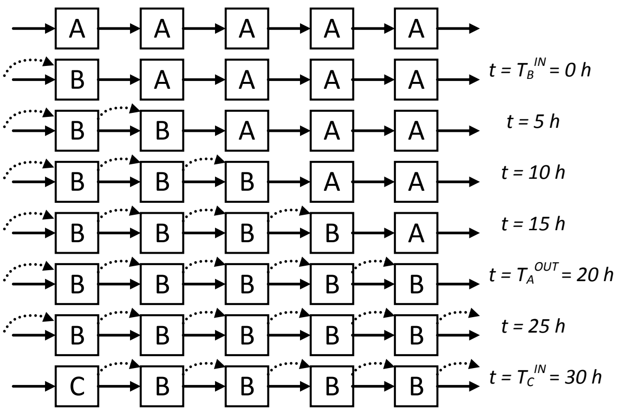

Fig. 2. Transition of models A, B and C in a 5-stations assembly line

Time events. Since the time scale is managed in a continuous manner, we are particularly interested in two time events: (a) T_k^{IN}, the time instant when the first unit in run k enters the line, and (b) T_k^{OUT}, the time point at which the last entity in run k exits the assembly line. The relationship between these variables is controlled by the block of constraints 8. As explained before, we assume $T_{k2}^{IN} = 0$.

$$
\begin{aligned}
&T_{k-1}^{OUT} \geq T_k^{IN} + (m-1)\sum_{i\in I} ct_i \; y_{k,i} \qquad \forall k > k_1 \\
&T_{k-1}^{OUT} \geq T_k^{IN} + (m-1)\sum_{i\in I} ct_i \; y_{k-1,i} \qquad \forall k > k_1 \\
&T_{k+1}^{IN} = T_{k-1}^{OUT} + \sum_{i\in I} ct_i \, [NP_{k,i} - (m-1)\, y_{k,i}] \qquad \forall k_1 < k < K
\end{aligned}
\tag{8}
$$

Finally, the production cycle length (T) is obtained by adding the production time of the [N_K - (m-1)] elements of the last run K to the completion time of the last element in the run (K-1).

$$T = T_{K-1}^{OUT} + \sum_{i \in I} ct_i \; [NP_{K,i} - (m-1) \; y_{K,i}] \tag{9}$$

Inventory profiles. Through the block of equations 10, stocks of final products are monitored over the production cycle at the start and the ending times of every production run ($t = T_k^{IN}$, $t = T_k^{OUT}$). These stock levels are represented by the variables $S_{i,k}^{IN}$ and $S_{i,k}^{OUT}$. For simplicity, we assume that inventory profiles of final products are piecewise linear functions (see Fig. 3) while model demand rates are constant values represented by the parameter r_i.

$$\begin{aligned} S_{i,k+1}^{IN} &= S_{i,k-1}^{OUT} + [NP_{k,i} - (m-1)\; y_{k,i}] - r_i \; (T_{k+1}^{IN} - T_{k-1}^{OUT}) && \forall k_1 < k < K, i \in I \\ S_{i,k}^{OUT} &= S_{i,k+1}^{IN} + (m-1)\; y_{k,i} - r_i \; (T_k^{OUT} - T_{k+1}^{IN}) && \forall k < K, i \in I \end{aligned} \tag{10}$$

In addition, the initial and the final inventory levels (So_i, Sf_i) should match, given the cyclic nature of the problem.

$$\begin{aligned} So_i &= S_{i,k_2}^{IN} = Sf_i && \forall i \in I \\ Sf_i &= S_{i,K-1}^{OUT} + [NP_{K,i} - (m-1)\; y_{K,i}] - r_i \; (T - T_{K-1}^{OUT}) && \forall i \in I \end{aligned} \tag{11}$$

Fig. 3 illustrates the typical behavior of final product stocks assembled in a multi-model synchronous line presenting neither idle times nor safety stock levels. As can be deducted, the selected cyclic production sequence is P1(k1)-P2(k2)-P3(k3)-P1(k4=k1), while cycle times verify: $ct_{P1} < ct_{P3} < ct_{P2}$. As the first unit of P2 enters the line at $t = 0$, the initial (m-1) units of P1 are produced at the rate of $1/ct_{P2}$ units per hour. No further production rate alteration is observed in other product transitions.

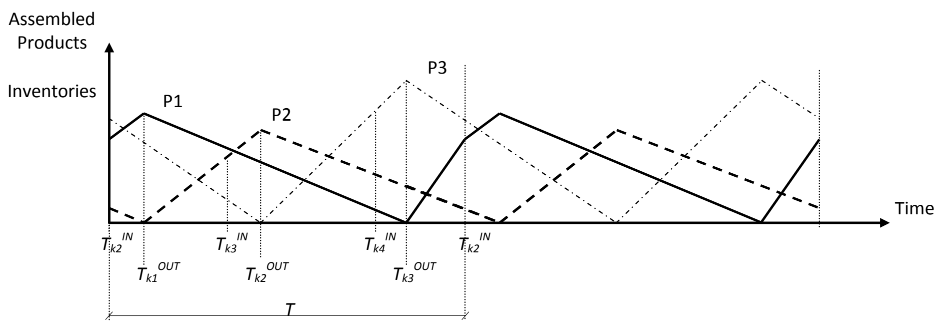

Fig. 3. Final product inventory profiles in a multi-model assembly line

Inventory carrying costs. Inventory carrying costs due to stocks of model *i* over a single production cycle ($ICost_i$) can be obtained by calculating the area below the inventory profile, multiplied by the unit individual inventory carrying cost (ic_i), as stated by eq. 12.

$$\begin{aligned} ICost_i = ic_i [& \sum_{k<K} (S_{i,k+1}^{IN} + S_{i,k}^{OUT})(T_k^{OUT} - T_{k+1}^{IN})/2 + \\ & \sum_{k_1<k<K} (S_{i,k-1}^{OUT} + S_{i,k+1}^{IN})(T_{k+1}^{IN} - T_{k-1}^{OUT})/2 + \\ & (S_{i,K-1}^{OUT} + Sf_i)(T - T_{K-1}^{OUT})/2] \qquad \forall i \in I \end{aligned} \tag{12}$$

Transition costs. Transition costs are due to changeover operations, converting the line from running one model to another. The transition cost when changing from model *i* to *i'* is a given datum: $tc_{i,i'}$. By knowing the model production sequence (through the binaries $y_{k,i}$) the transition cost when launching run *k* ($TCost_k$) is lower-bounded by constraint 13.

$$TCost_k \geq tc_{i,i'} (y_{k-1,i} + y_{k,i'} - 1) \qquad \forall k > k_1, i \neq i' \tag{13}$$

Transition costs are very important components of the total operation cost, which can be deducted from to the productivity reduction produced by variations in the cycle time. Other changeover times (e.g., machine adaptations) are assumed to be negligible compared to cycle times.

Objective function. The aim of the MINLP formulation is to minimize the sum of inventory carrying and transition costs per unit time, which can be calculated by dividing the costs incurred every cycle by the cycle length *T*, as shown in eq. 14.

$$Min \quad z = 1/T\, [\sum_{i \in I} ICost_i + \sum_{k > k_1} TCost_k] \tag{14}$$

Overall, we seek to minimize the nonlinear function 14 subject to constraints 6 to 13. Note that the only nonlinear constraint in the model is the calculation of the inventory carrying costs (12), which includes bilinear terms. In the next section we propose an approximation of the inventory carrying costs which leads an important reduction in the computational effort incurred for solving the model.

Approximation of inventory carrying costs. If all the models had a common cycle time, with the assembly line showing neither idle times nor safety stock levels, the inventory profile of every product at every production cycle would appear as an exact triangle. The highest inventory level of model *i* (height of the corresponding triangle) occurs at the time the last unit of that model exits the assembly line ($t = T_k^{OUT}$, if *k* is the run producing model i). Hence, we can estimate the inventory carrying costs per cycle as in constraint 15. Note that constraint 15 is relaxed in case run k is not dedicated to the production of model i.

$$ICost_i \geq ic_i[\sum_{k<K} S_{i,k}^{OUT}\ T/2 - n_{\max}h_{\max}(1-y_{k,i})/2] \tag{15}$$

As a result, inventory carrying costs per unit time can be estimated as in constraint 16.

$$ICost'_i \geq ic_i[\sum_{k<K} S_{i,k}^{OUT}/2 - n_{\max}(1-y_{k,i})/2] \tag{16}$$

This approximation yields a new objective function (17), for which the inventory carrying cost calculation is totally linear. The expression $S_{i,k}^{OUT}$ / 2 can be regarded as the average inventory level of model i, in case run k is the one producing i. Note that approximation 16 is also valid if more than one run during the same cycle is devoted to the production of the same model. In such a case, the average inventory level is calculated from S_{i,k^*}^{OUT} / 2, with k^* being the largest run producing models of type i. Finally, the minimization of function 17 is subject to constraints 6-11,13 and 16.

$$Min \quad z = \sum_{i \in I} ICost'_i + 1/T \sum_{k>k_1} TCost_k \tag{17}$$

Through simple analysis we conclude that the closer the cycle times of the various models and the larger the production runs (meaning that transition periods have a lesser influence), the more precise the approximation. In later sections we show through an illustrating example that the approximated model yields exactly the optimal solution with evaluation errors of the objective function below 2%, taking less than 1% of the CPU time needed by the full model to converge to the global optimum.

2.3 MINLP Formulation for a Multi-Model Synchronous Assembly Line with Part Supply

Most assembly line systems require anticipated part supplies from different sectors of the same factory before a production run can be launched. In this section we extend the MINLP model already presented to account for part supplies. In particular, we are interested in including part inventory carrying costs to the objective function, since different models may significantly differ regarding the part supply strategy. For instance, customized models usually require a larger stock of parts at the time of launching the corresponding production run. With that purpose, we introduce the set P of part types. Moreover, we are given the number of parts of type p that have to be available in the assembly line at the time the run of product i is launched. Such an initial stock level ($ISP_{p,i,k}$) is assumed to be proportional to the number of units produced through the corresponding run. In other words, $IPS_{p,i,k} = is_{p,i}\ NP_{k,i}$. Note that if run k is not associated to the production of model i, then $NP_{k,i} = 0$ and the variable $IPS_{p,i,k}$ is also null.

Assuming that: (a) the parts needed to assemble a certain model are uniformly consumed throughout the production run, (b) the parts start to be produced $\tau_{i,p}$ hours before the run launching time, and (c) stocks of parts destined to assemble a certain product are depleted at the time the last entity of the run exits the line, we can obtain an approximation of the average stock level of parts of type *p* required to assemble model *i*, and its corresponding carrying cost per production cycle, as shown in eq. 18.

$$PCost_p \geq ic_p \sum_{i \in I} is_{p,i} [\sum_{k_1 < k < K} NP_{k,i} (T_k^{OUT} - T_k^{IN} + \tau_{i,p}) + NP_{K,i} (T - T_K^{IN} + \tau_{i,p} + T_{k_1}^{OUT})] / 2 \tag{18}$$

The model objective function is finally expressed by eq. 19.

$$Min \quad z = 1/T \, [\sum_{i \in I} ICost_i + \sum_{p \in P} PCost_p + \sum_{k > k_1} TCost_k] \tag{19}$$

3 Results and Discussion

In this section we present two examples with the aim of evaluating the performance of the optimization models. The first example is an illustrative case involving the production of four product types in a five-stations synchronous assembly line. Afterwards, we analyze the effects of increasing the number of model types. The second example is a real-world case study from a trailers production industry in Argentina.

3.1 Example 1

We deal with an assembly line comprising 5 stations, producing 4 model types (P1, P2, P3, P4). Product demand rates are 0.300, 0.400, 0.100, 0.300 units per hour, cycle times are 0.792, 0.950, 1.187, 0.712 hours, and inventory carrying costs are \$1.00, \$1.10, \$1.20, \$1.30 per unit per hour, for models P1, P2, P3 and P4, respectively. Transition costs are presented in Table 1.

Table 1. Transition costs (in \$) for every pair of models subsequently produced in the line

	Successor			
Predecessor	P1	P2	P3	P4
P1	-	200	100	200
P2	240	-	240	160
P3	200	300	-	300
P4	240	120	240	-
Average	227	207	193	220

In the first step, we obtain an approximate lot size for every product using the model described in section 2.1. Run launching costs are calculated by averaging transition costs, yielding: $227, $207, $193 and $220 for products P1, P2, P3, and P4, while production rates are simply obtained from raising the cycle times to the power of (-1). From eq. 5, a reasonable choice for the production cycle time is 43.05 h, and the approximate lot sizes are 12.9, 17.2, 4.3, and 12.9 units respectively.

Given these results, we arbitrarily fix the upper bounds of the lot sizes ($n_{\max}$) to 30 units, i.e. almost double the maximum approximate size, and we solve the MINLP model presented in section 2.2, with the exact objective function (14). The model comprises 193 equations and 96 variables, 24 of which are integer. Using GAMS/BARON 11.3.0 [15] on an Intel Xeon 2.67 GHz we find the global optimal solution in 1083 CPU s, amounting to 37.93 US$/h, with a 10^{-4} of optimality gap. The optimal product sequence is $P1_{15}$-$P4_{15}$-$P2_{20}$-$P3_{5}$, with subscripts indicating the lot sizes. Note that the lot sizes are slightly larger than the ones obtained using the model of section 2.1 because the minimum lot size in the MINLP is $m = 5$. Besides, the transition P1-P4 shows no idle time as both products have the same production rate. Moreover, the production cycle time is 50 h, with the following run launching times, given as subscripts in h: $P4_{0}$-$P2_{11}$-$P3_{30}$-$P1_{35.9}$. Transition costs average 15.2 US$/h, while inventory costs amount to 22.73 US$/h.

Afterwards, to reduce the computational effort, we solve the model using the approximate objective function (17) and we exactly obtain the same product sequence and lot sizes, in only 6.62 CPU s. This behavior can be attributed to the elimination of bilinear terms in the objective function. The optimal value of the approximate function is 37.52 US$/h, i.e. 1% below the actual value.

Finally, when we increase the number of products to 8, the model comprises 781 equations and 332 variables, 80 of which are integer. In this case, using the approximate objective function (17) and after 3 h of computational time, the global optimizer cannot find the optimal solution and reports a relative gap of 4.88. In fact, using the rigorous objective function (14), the solver cannot find even a feasible solution after 3 h of CPU time. This proves the NP-hard complexity of this combinatorial problem, requiring advanced solution strategies when the product variety is wider.

3.2 Example 2

This example deals with a real-world synchronous assembly line producing two families of truck trailers (A and B) for the transportation industry in Argentina. The line comprises 5 stations with 2 workers in each of them. Both models majorly differ in their length (9 m vs. 13 m), resulting in a much larger cycle time for the long trailer (B). Traditionally, this fact has lead planners to adopt relatively extensive production campaigns ($80_A - 32_B$) so as to reduce transition times and changeover costs. However, the company finance area has remarked that this strategy results in extremely large inventory carrying costs due to high stocks of final products and intermediate (semi-elaborate) material. Further information on this case study is not given due to confidentiality reasons.

After applying the model presented in the section 2.3, we find out that by reducing the run length to less than half of the current value (from $80_A - 32_B$: 400 h, to $35_A - 14_B$: 175 h) important savings are achieved, amounting to 182,000 AR$/y (23.6% cost reduction). Although transition costs increase by 159,000 AR$/y, inventory carrying costs reduce by 341,000 AR$/y (257,000 AR$/y from final product stock reduction and 84,000 AR$/y due to semi-elaborate stock reduction). Using GAMS/BARON 11.3.0 solver in the same processor, the optimal solution is found in 2.46 CPUs, amounting to 251.98 AR$/h (590,000 AR$/y). In this case, the MINLP model comprises only 36 variables (9 integer) and 43 constraints, with bilinear terms accounting for the inventory carrying cost in the objective function.

4 Conclusions

We present a novel tool for the optimal planning of multi-model synchronous assembly lines, in which different products are cyclically produced through runs or campaigns. The proposed MINLP model permits to find the optimal sequence and length of production runs in order to minimize the sum of transition and inventory carrying costs. Moreover, the model rigorously accounts for transient periods occurring in the line due to cycle time variations. Although the MINLP model presents non-convex terms in the objective function, results show that it can be solved to global optimality in a reasonable CPU time, for examples involving up to 4 product types in a line with 5 stations. We also develop a simple approximation of the average inventory levels yielding the actual optimal solution with a two order of magnitude reduction in the CPU time. The optimization framework finally incorporates the cost of semi-elaborate stocks. When applied to a real-world case study of the argentine truck trailer industry, savings over 23% (180,000 AR$/y ~ 22,000 US$/y) are achieved. Future work will be focused on extending the model application to production lines with larger numbers of stations and product types, at the time new solution strategies are developed so as to obtain efficient results with reasonable computational effort.

Acknowledgements. Financial support received from Universidad Nacional del Litoral under CAI+D 2011- PI 256 grant is fully appreciated. We are also most grateful to Sola y Brusa S.A. for the case study provided to us.

References

1. Becker, C., Scholl, A.: A survey on problems and methods in generalized assembly line balancing. European Journal of Operational Research. 168, 694-715 (2006).
2. Scholl, A., Becker, C.: State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. European Journal of Operational Research. 168, 666-693 (2006).

3. Hoffmann, T.R.: EUREKA: A hybrid system for assembly line balancing. Management Science. 38, 39-47 (1999).
4. Scholl, A., Klein, R.: SALOME: A bidirectional branch and bound procedure for assembly line balancing. INFORMS Journal of Computing. 9, 319-334 (1997).
5. Sprecher, A. Dynamic search tree decomposition for balancing assembly lines with parallel search. International Journal of Production Research. 41, 1413-1430 (2003).
6. Helgeson, W.B., Birnie, D.P.: Assembly line balancing using the ranked positional weight technique. Journal of Industrial Engineering. 12, 394-398 (1961).
7. Kildbridge, M.D., Wester, L.: A heuristic method for assembly line balancing. Journal of Industrial Engineering. 12, 292-298 (1961).
8. Hackman, S.T., Magazine, M.J., Wee., T.S.: Fast, effective algorithms for simple assembly line balancing problems. Operations Research. 37, 916-924 (1989).
9. Kim, Y.K., Kim, Y., Kim, Y.J.: Two-sided assembly line balancing: A genetic algorithm approach. Production Planning and Control. 11, 44-53 (2000).
10. Lapierre, S.D., Ruiz, A., Soriano, P.: Balancing assembly lines with tabu search. European Journal of Operational Research. 168, 826-837 (2006).
11. Seker, S., Ozgurler, M., Tanyas, M.: A weighted multiobjective method for mixed-model assembly line problem. Journal of Applied Mathematichs. doi: 10.1155/2013/531056 (2013).
12. Karabati, S., Sayin, S.: Assembly line balancing in mixed-model sequencing environment with synchronous transfers. European Journal of Operational Research. 149, 417-429 (2003).
13. Akpinar, S., Bayhan, G.M.: A hybrid genetic algorithm for mixed model assembly line balancing problem with parallel workstations and zoning constraints. Engineering Applications of Artificial Intelligence. 24, 449-457 (2011).
14. Ouenniche, J., Boctor., F.F.: The multi-product, economic lot-sizing problem in flow shops: the powers-of-two heuristic. Computers & Operations Research. 28, 1165-1182 (2001).
15. Tawarmalani, M., Sahinidis N.V.,: A polyhedral branch-and-cut approach to global optimization. Mathematical Programming. 103, 225-249 (2005).