Optimal sequencing of mixed models with sequence-dependent setups and utility workers on an assembly line

May 22, 2009

Abstract

This paper presents an integer programming formulation for the sequencing problem in mixed-model assembly lines where the number of temporarily hired utility workers and the number of sequence-dependent setups are to be optimized simultaneously through a cost function. The resultant model offers an operational way to implement the utility work needed to avoid line stoppages, unlike previous papers addressing the goal of smoothing the workload. The present research has an immediate application to the automotive industry, namely to the car-sequencing problem. Simulation results show that the proposed formulation leads to the optimum in a reasonable time for instances up to 15 items and to satisfactory feasible solutions for some of the larger problems we considered within a moderate time limit.

Keywords: Mixed-model assembly lines; Sequencing; Spacing constraints; Sequence-dependent setups; Utility workers; Economic evaluation.

1 Introduction

Past research on the sequencing problem in a mixed-model assembly line has primarily focused on two goals. Prior to the 1980's, the main goal was (1): to smooth the workload at each workstation through the line. This goal seeks to reduce line stoppages or inefficiencies like work congestion or utility work (see Mitsumori, 1969 or Xiaobo and Ohno, 1997 for optimal models and for heuristics, see Thomopoulos, 1967; Macaskill, 1973; Sumichrast et al., 1992; Smith et al., 1996 and Gottlieb et al., 2003).

The emerging JIT concept by the mid 1980's has raised a second goal that consists in keeping a constant rate of part usage to avoid large inventories. Goal (2) may be achieved by: (2a) synchronizing the production of each model with its demand and (2b) levelling the usage of each part at each level of the manufacturing process. Each goal (2a) and (2b) is usually expressed as an objective of minimization of the variation (squared deviations) of the actual production/usage from the desired amount. Goals (2a) and (2b) are strictly equivalent when all products require the same number and mix of parts (see Miltenburg, 1989). Under this assumption, Goal (2) is optimally addressed in Miltenburg et al., 1990 and heuristically solved in Miltenburg, 1989;

Ding and Cheng, 1993 and Korkmazel and Meral, 2001. Without this hypothesis, Goal (2a) is optimally solved by Kubiak and Sethi, 1991 and by Bautista et al., 2000; heuristic methods to handle Goal (2b) may be found in Bautista et al., 1996; Leu et al., 1996 and Monden, 1998.

Goals (1) and (2) are simultaneously addressed in optimal formulations by Korkmazel and Meral, 2001 and by Drexl and Kimms, 2001 who also provide a heuristic solution approach to the problem by using column-generation techniques to solve the corresponding LP-relaxation. Goals (1) and (2) are heuristically considered through bi-criteria approaches by Choi and Shin, 1997; Tamura et al., 1999 and by Korkmazel and Meral, 2001.

The additional objective of minimizing the set-up cost, namely Goal (3), is rarely examined, possibly because it conflicts with both previous goals: a sequence that smooths the workload neither ensures a constant rate of part usage nor a minimum number of model switches (see Burns and Daganzo, 1997 for a discussion). A heuristic approach handling both Goals (2) and (3) is developed by Mansouri, 2005. Multi-objective models including the three goals are quite scarce (Hyun et al., 1998).

The present paper considers Goals (1) and (3) in an optimal formulation of the problem where the objective function is a cost function associated with hiring temporary/utility workers to avoid line stoppages and with model switches in the sequence. The research presented here differs from earlier studies in the sequencing area in several ways. First, as an economic cost function is associated with the goals, the challenging task of assigning weights to the different objectives—whose measuring units are not identical—is avoided (see Tamura et al., 1999 or Murata et al., 1996 for a discussion on methods to set weights to multi-objective functions). Second, papers that address the goal of smoothing the workload often express it as an objective of minimization of the total utility work (see for instance Hyun et al., 1998). This latter objective consists in minimizing the number of hours that additional workers should handle to avoid line stoppages so no indication on when and how many workers must temporarily be hired is provided. By contrast, our formulation leads to assigning utility workers to stations so as to process some items in the sequence. Spacing constraints are utilized here to account for workstations capacity limitations, like in few papers (Smith et al., 1996; Choi and Shin, 1997; Drexl and Kimms, 2001; Gottlieb et al., 2003) although they are more intuitive and operational than processing time-based rules.

Our work has a direct application to the car sequencing problem that consists in finding a sequence of vehicles that minimizes the set-up costs while satisfying capacity contraints along the assembly line. The set-up costs of switching from one model to another involve the cost of tuning up the machines and —possibly— the cost of cleaning sprayguns if the colour is changed. As solvent purchase constitutes a major expense for automotive body shops, there is a strong incentive to group same-coloured vehicles to minimize the number of purges. In our experimental study, we consider this type of application to generate instances on which we apply the proposed formulation that is solved using the commercial software MIP solver ILOG-Cplex 9.0. Considering the option of hiring utility workers in such a context creates an interesting trade-off as utility workers allow for violations of spacing constraints which result in more colour grouping possibilities.

The next section provides a general description of the sequencing problem as well as the usual constraints relative to the demand satisfaction and the capacity limitations. Section 3 presents the integer programming formulation designed to achieve Goals (1) and (3). Section 4 summarizes the optimization problem and the assumptions we made. The section also includes a useful table of notations the reader may refer to. In Section 5 we describe the experimental framework and we comment the simulation results. We conclude in Section 6.

2 Problem statement

This section is dedicated to a general description of the sequencing problem. Constraints relative to demand satisfaction and capacity limitations are then provided. The notations we will use throughout are summarized in Tables 8 and 9 (see Section 4).

2.1 General description

In the mixed-model assembly line, some models of items have to be assembled over a horizon T (usually a day) for which a total demand of N items must be satisfied. The cycle time $\overline{\theta}$ is defined as $\overline{\theta} = \lfloor T/N \rfloor$ which means that items are launched in fixed $\overline{\theta}$ time intervals to the assembly line.

The line is partitioned into K stations of three types. Subset K_C groups the stations with a fixed capacity and subset K_F is composed of flexible stations whose capacity can be extended with utility workers temporarily hired. Stations that neither belongs to K_C nor K_F are those for which processing times never exceed the cycle time. Operations on such stations involve doing exactly the same thing to all items (installing safety belts on all vehicles is an example).

A model or variant results from a combination of options. Each option may be fully processed on a single station or may undergo several operations on different workstations to be completed. In the following, we will consider each item separately by letting $y_{i,h}$ be a binary variable that takes a value of 1 if item i is in position h in the sequence, and 0 otherwise. Our incentive is to allow any combination of options for the customer so the space of choices is not limited to a predetermined number of variants.

Solving the sequencing problem involves finding an arrangement of the items in the sequence, say a $0-1$ matrix $Y = \{y_{i,h}\}$ of dimension $N \times N$ so as to meet several objectives like minimizing the number of utility workers and the setups, levelling the utilization rate, subject to a set of constraints relative to the demand satisfaction and capacity limitations.

2.2 Usual constraints: demand satisfaction and spacing requirements

We first provide the constraints relative to the demand satisfaction. We then give the general form of the spacing constraints. We finally derive analytically the spacing constraints in the particular case of stations hosting a single permanent worker.

2.2.1 Demand satisfaction

The following constraints ensure that the demand is satisfied over horizon T (each item is sequenced) and that exactly one product is assigned to each position in the sequence (see for instance Drexl and Kimms, 2001)

$$\begin{aligned}\sum_{h=1}^{N} y_{i,h} &= 1,\ i=1,\ldots,N, \quad (1)\\ \sum_{i=1}^{N} y_{i,h} &= 1,\ h=1,\ldots,N.\end{aligned}$$

2.2.2 General form of the spacing constraints

Spacing constraints are specified to avoid line stoppages. They dictate the spacing (i.e. the number of items without option) between two consecutive items with option. The spacing constraint $\nu_o : \eta_o$ stipulates that 'there must be at most ν_o items with option o in any sequence of η_o consecutive items'.

A spacing constraint for option o is required as soon as its processing time on one station k, denoted by $\theta_{o,k}$ is such that $\theta_{o,k} > \lambda_k \cdot \overline{\theta}$, where λ_k is the number of permanent workers dedicated to station k. When the previous inequality holds for several stations k, the spacing constraint for option o is dictated by the most capacitated station k_o processing this option o, with $k_o \in \arg\max_{k\in K_C\cup K_F}\{\theta_{o,k}\}$, and we usually have $\nu_o = \lambda_{k_o}$ and $\eta_o = \nu_o/d_o$, where d_o is the average rate of demand for option o.

Consider for instance one option $o = 1$ and 3 stations $k = 1, 2, 3$. The cycle time is $\overline{\theta} = 60$ sec. The number of permanent workers on each station is given by $\lambda_1 = 2$; $\lambda_2 = 4$; $\lambda_3 = 3$. We assume the demand rate for option 1 equal to 20%. Table 1 provides for each station k the processing time $\theta_{1,k}$ and the value of $\lambda_k \cdot \overline{\theta}$.

Station k	Process. time $\theta_{1,k}$	$\lambda_k \cdot \overline{\theta}$	Spacing constraint $\nu_1 : \eta_1$
1	127	120	
2	200	240	
3	190	180	3 : 15

Table 1: Spacing constraints for an option o - illustration

Two stations $k = 1$ and $k = 3$ are such that $\theta_{1,k} > \lambda_k \cdot \overline{\theta}$, indeed for $k = 1$ we have $127 > 120$ and for $k = 2$,we have $190 > 180$. Among these two stations, station $k_1 = 3$ is the station for which $\theta_{1,k}$ is maximum. With $\lambda_3 = 3$ we have $\nu_1 = 3$. Since $d_1 = 1/5$ then $\eta_1 = 15$. For option 1, the most capacitated station is station 3 and the spacing constraint is $\nu_1 : \eta_1 = 3 : 15$ meaning that there must be at most 3 items with option 1 in any sequence of 15 items.

When a station k provides the tightest spacing constraint for more than one option, these options are grouped in a subset and the tightest of the tightest spacing constraints applies for any option in this subset. This reasoning substantiates the common assumption that each option is fully processed on a single station only dedicated to the installation of this option. Thus, in the following, we will make no difference between option o and the station k on which the option is treated and we will keep the notation k to designate the option processed on station k.

For instance, following the previous example, if a second option $o = 2$ is also such that station 3 is the most capacitated station with a spacing constraint $\nu_2 : \eta_2 = 3 : 4$, we thus will consider option 1 or 2 as a single 'artificial' option o processed on station $k = 3$ with a spacing constraint of $\nu_o : \eta_o = 3 : 15$ (since this is tighter than $\nu_2 : \eta_2 = 3 : 4$).

Letting $\delta_{i,k}$ be a parameter that takes a value of 1 if option k is required by item i and 0 otherwise, the spacing constraint is written as

$$\sum_{l=h-\eta_k+1}^{h} \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,l} \leq \nu_k, \;\; h = 1, \ldots, N \;;\; k \in K_C \cup K_F. \tag{2}$$

The spacing constraint (2) applies for any flexible station as long as no utility worker operates on this station. This constraint can also iteratively be defined as

$$\begin{aligned} & m_{k,h} = m_{k,h-1} + \textstyle\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h} - \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h-\eta_k} \leq \nu_k, \\ & h = 1, \ldots, N \;;\; k \in K_C \cup K_F, \end{aligned} \tag{3}$$

where $m_{k,h}$ is the total number of items with option recorded in the set of positions (or window) $\{h - \eta_k + 1, \ldots, h\}$. For the sake of clarity, we will use expression (3) when we adapt the spacing constraint to the case of flexible stations hosting utility workers (see paragraph 3.1.1, p. 8).

As an illustration, consider $N = 6$ items to be produced during horizon T. Four models are produced: A, B, C, D and models B and D require option $k = 1$. The spacing constraint is $\nu_1 : \eta_1 = 1 : 2$. Table 2 provides a sequence of models, where $h = -1, 0$ are negative indices chosen for the previous horizon $T - 1$. Indices $h = 1, ..., 6$ represents items'position during horizon T. For each position h, the value of $\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h}$ is simply equal to 1 if item in position h requires option k (with $k = 1$ in our example) and zero otherwise. Constraint (2) consists in counting the number of options k in any window of η_{k} items and checking if this number is inferior to ν_k. In constraint (3), the number of items with option in the window $\{h - \eta_k + 1, \ldots, h\}$ equals the number of items with option in the previous window to which we add the last element of the new window and from which we subtract the first element of the former window, as windows roll. Considering $h = 1$, we have item D in the window $\{0, 1\}$ thus $m_{1,1} = 1$. For $h = 2$, we consider window $\{1, 2\}$, we add $\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,2} = 1$ (last element of the new window) to $m_{1,1}$ and sustract $\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,0} = 0$ (first element of the former window) to obtain $m_{1,2} = 2$. Note the spacing constraint is not satisfied.

Horizon		$T-1$		T					
h	...	-1	0	1	2	3	4	5	6
Item		D	C	D	B	A	B	D	C
$\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h}$		1	0	1	1	0	1	1	0

Table 2: Checking the spacing constraints

2.2.3 Deriving the spacing constraint for stations with a single dedicated worker

For the stations designed to host a single permanent worker ($\lambda_k = 1$), spacing constraints can be derived from capacity limitations defined in terms of operation time. We note $\theta_{i,k}$ the processing time of item i on station k. The operation time of the h^{th} item on station k is $\sum_{i=1}^{N} \theta_{i,k} \cdot y_{i,h}$. We can reasonably assume that there are only two possible values for the processing time on station k: $\theta_k^{\max}$ and $\theta_k^{\min}$ which respectively designates the operation time when option k is installed on any item i and when it is not, with $\theta_k^{\min} < \overline{\theta} < \theta_k^{\max}$. Th e processing time can therefore be written as $\sum_{i=1}^{N} \theta_{i,k} \cdot y_{i,h} = \theta_k^{\max} \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h} + \theta_k^{\min} \sum_{i=1}^{N} (1 - \delta_{i,k}) \cdot y_{i,h}$, recalling $\delta_{i,k} = 1$ if item i has option k and zero otherwise. Utilizing constraint (1), we have $\sum_{i=1}^{N} \theta_{i,k} \cdot y_{i,h} = (\theta_k^{\max} - \theta_k^{\min}) \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h} + \theta_k^{\min}$. Thus, if item i has option k, its processing time on station k equals $\theta_k^{\max}$; it equals $\theta_k^{\min}$ otherwise.

Let $R_k^{\max}$ be the difference between the total time dedicated to operations performed on station k and $\overline{\theta}$. This represents the extra time that can be spent on station k, compared to the average $\overline{\theta}$. To process one item with option on station k, this extra time $R_k^{\max}$ must be such that $R_k^{\max} \geq \theta_k^{\max} - \overline{\theta}$. The excess working time compared to the cycle time, $R_{k,h}$, after completion on station k of operations on the h^{th} item is given by

$$R_{k,h} = \max \left\{ R_{k,h-1} + (\theta_k^{\max} - \theta_k^{\min}) \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h} + \theta_k^{\min} - \overline{\theta}, 0 \right\}. \tag{4}$$

The capacity of station k is violated by the item in position h as soon as $R_{k,h} > R_k^{\max}$ and $R_{k,h-1} \leq R_k^{\max}$, $\forall\, k \in K_C \cup K_F |\, \lambda_k = 1$ and $h = 1, \ldots, N$.

To illustrate, consider a station k hosting a single dedicated worker with $\theta_k^{\max} = 80$ sec., $\theta_k^{\min} = 50$ sec., $\overline{\theta} = 60$ sec. and $R_k^{\max}$, the extra time that can be spent on station k is $R_k^{\max} = 40$ sec. As option k consumes an additional time of $\theta_k^{\max} - \overline{\theta} = 20$ sec. then 2 items with option k may be produced consecutively. To catch up these 40 extra seconds, it is then necesary to produce 4 items without option, since the underuse of such items on station k is $\overline{\theta} - \theta_k^{\min} = 10$ sec.

More formally, let us consider the situation in which option k is required by the first item in the sequence ($\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,1} = 1$) and $R_{k,0} = 0$. From (4) we have $R_{k,1} = \theta_k^{\max} - \overline{\theta}$. A second item with option may follow if $R_{k,2} = 2(\theta_k^{\max} - \overline{\theta}) \leq R_k^{\max}$. Finally, the maximum number ν_k of consecutive items with option k is such that $(\theta_k^{\max} - \overline{\theta})\nu_k \leq R_k^{\max}$, which gives

$$\nu_k = \left\lfloor \frac{R_k^{\max}}{\theta_k^{\max} - \overline{\theta}} \right\rfloor . \tag{5}$$

After ν_k consecutive items with option, we have $R_{k,\nu_k} = R_k^{\max}$ and a number μ_k of items without option must follow before sequencing once again ν_k consecutive items requiring the option. This number μ_k is such that $(\overline{\theta} - \theta_k^{\min})\mu_k \geq (\theta_k^{\max} - \overline{\theta})\nu_k$ which finally leads to

$$\mu_k = \left\lceil \nu_k \cdot \frac{\theta_k^{\max} - \overline{\theta}}{\overline{\theta} - \theta_k^{\min}} \right\rceil . \tag{6}$$

The spacing constraint (2) or (3) still applies with ν_k as determined by (5) and $\eta_k = \nu_k + \mu_k$ with μ_k defined by (6) for all stations k such that $\lambda_k = 1$. It should be noted that this spacing constraint is a necessary and sufficient condition not to exceed the capacity only when $\nu_k = 1$ or equivalently when $R_k^{\max} < 2 \cdot (\theta_k^{\max} - \overline{\theta})$. Otherwise, the spacing constraint is only a sufficient condition: a window $\{h - \eta_k + 1, \dots h\}$ may contain more than ν_k items with the option without violating the capacity constraint of the station, $R_{k,l} \leq R_k^{\max}$ for all $l \in \{h - \eta_k + 1, \dots h\}$, with $R_{k,l}$ defined by (4).

Let us consider again the example of a station k hosting a single dedicated worker with $\theta_k^{\max} = 80$ sec., $\theta_k^{\min} = 50$ sec., $\overline{\theta} = 60$ sec. and $R_k^{\max} = 40$ sec. From (5) and (6), we get $\nu_k = 2$ and $\mu_k = 4$ leading to $\eta_k = 6$. The spacing constraint states in a strict sense that after 2 consecutive items with option, 4 items without option must be sequenced so as to get a zero exceeding processing time which allows again for the sequencing of 2 consecutive items with option and so on. The sufficient condition not to exceed the capacity limitation is to sequence at most 2 vehicles with option out of 6 vehicles. Table 3 displays a sequence of items for which the spacing constraint can be checked for $h = 6$ and $h = 7$. For $h = 6$, constraint (2) gives $\sum_{l=1}^{6} \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,l} = 2 \leq 2$. For $h = 7$, we get $\sum_{l=2}^{7} \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,l} = 2 \leq 2$. Note the values of $R_{k,l}$ never exceed $R_k^{\max} = 40$, meaning that the capacity limitation is not violated. However, placing an item with option in position 7 would have led to $R_{k,7} = 40 \leq R_k^{\max}$, with a number of 3 items with option in the window $\{2, \dots, 7\}$. This illustrates the fact that the spacing constraint is a sufficient condition to meet the capacity constraint but it is not a necessary condition when $\nu_k > 1$.

Position l	1	2	3	4	5	6	7
$\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,l}$	0	1	0	0	0	1	0
$R_{k,l}$	10	30	20	10	0	20	10

Table 3: Spacing constraint with no utility worker

3 Optimal formulation of the sequencing problem

The objective function considered here is a cost function involving two elements: the cost associated with additional utility workers and the setup cost. There is a trade-off between these two costs as utility workers allow for violations of spacing constraints without entailing line stoppages, so more grouping possibilities are available, hence reducing the number of setups.

We first provide the analytical description of the problem of optimizing the number of utility workers to be temporarily hired. We then turn to the formal description of the problem relative to the minimization of the number of setups in the sequence.

3.1 Objective and constraints relative to utility workers

Past studies have considered the objective of minimizing total utility work expressed as a global working time and minimized as such, without evaluating its economic impact as no wage costs are introduced (see for instance Hyun et al. 1998). We handle the objective differently: our incentive is to minimize the number of utility workers required to implement sequencing decisions taken for horizon T. The resultant formulation gives the optimal assignment of utility workers on stations and items. We first provide an adaptation of constraints (3) to the case of flexible stations (from the set K_F). We then derive the optimal number of utility workers.

3.1.1 Spacing constraints for stations with utility workers

We assume that each flexible station hosts a single dedicated worker ($\lambda_k = 1$) so the spacing constraint $\nu_k : \eta_k$ is defined by using Eq. (5) to determine ν_k and Eq. (6) to get the value of $\eta_k = \nu_k + \mu_k$. Each flexible station $k \in K_F$ is conceived to host a single utility worker by cycle time.

Let $w_{k,h}$ be a binary variable that takes a value of 1 if a utility worker is required on station k for the item in position h and 0 otherwise. We assume that workers do not collaborate on the same task: the item in position h is fully processed by the utility worker while the regular worker finishes the operations on the $(h-1)^{th}$ item.

A utility worker is required on station k to process the item in position h in order to avoid a violation of capacity which occurs when $R_{k,h-1} + \theta_k^{\max} - \overline{\theta} > R_k^{\max}$. The item in position h is necessarily an item with option (see Eq. (4)). We assume the regular worker catches up on the total backlog $R_{k,h-1}$ (while the utility worker fully handles the item in position h). This assumption makes possible an adaptation of the spacing constraints otherwise a reasoning in terms of excess working time must be adopted and spacing constraints must be abandoned. Formally, we assume $R_{k,h-1} \leq \overline{\theta}$ (the excess working time can not be superior to the 'average' time for producing an item). As a straightforward upper bound for $R_{k,h-1}$ equals $(\theta_k^{\max} - \overline{\theta})\nu_k$, we finally assume $(\theta_k^{\max} - \overline{\theta})\nu_k \leq \overline{\theta}$. This implies that the presence of a utility worker on station k for item in position h leads to a reduction of the excess working time by an amount of $(\theta_k^{\max} - \overline{\theta})(\nu_k + 1)$ where $(\theta_k^{\max} - \overline{\theta})\nu_k$ is handled by the regular worker and $\theta_k^{\max} - \overline{\theta}$ is absorbed by the utility worker himself. This is equivalent to 'cancelling' $\nu_k + 1$ items with option in the window $\{h - \eta_k + 1, \ldots, h\}$ since the next item in position $h+1$ may be chosen as if there was no item with option in positions $h - \eta_k + 1$ to h.

To illustrate the reasoning, consider a station k with $\theta_k^{\max} = 80$ sec., $\theta_k^{\min} = 40$ sec., $\overline{\theta} = 60$ sec. and $R_k^{\max} = 20$, so the spacing constraint is $\nu_k : \eta_k = 1 : 2$. Let us assume a first item with option is produced and no excess working time is to be caught up. The regular worker handles this first item and needs to work 20 seconds more than the cycle time to complete the item. If a second item with option is to be processed, the cumulated excess working time would be 40 seconds, which is twice the extra time that is allowed. A utility worker is therefore necesary to handle this second item, so we have $w_{k,2} = 1$. The utility worker fully processes this item, thus absorbing an extra time of 20 sec. while the regular worker can finish his work on the first item, therefore absorbing also an extra time of 20 seconds. A third item with option can therefore be processed, as the presence of the utility worker finally allows for absorbing the equivalent of the extra time needed to process 2 items with option (one extra time is done by the utility worker himself and the other one by the regular worker). The presence of the utility worker for $h = 2$ (second item) is thus equivalent to cancelling $\nu_k + 1 = 2$ items with option in the window $\{h - \eta_k + 1, \dots, h\} = \{1, 2\}$.

Thus, to adapt constraint (3), we must decrease $m_{k,h}$ by $(\nu_k + 1)\, w_{k,h}$ and replace $\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h-\eta_k}$ with a proper binary variable, namely $q_{k,h-\eta_k}$, that takes into account the fact that a possible utility worker has led to the 'cancellation' of the option, if any, on the $(h - \eta_k)^{th}$ item. Letting $q_{k,h-\eta_k}$ be such a binary variable, the spacing constraints for flexible stations can now be written as

$$\begin{aligned} &m_{k,h} = m_{k,h-1} + \textstyle\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h} - q_{k,h-\eta_k} - (\nu_k + 1)\, w_{k,h} \le \nu_k, \\ &h = 1, \dots, N;\ k \in K_F, \end{aligned} \tag{7}$$

where $q_{k,h-\eta_k}$ takes a value of 1 if item in position $h - \eta_k$ has the option and none of the items in positions $\{h - \eta_k, \dots, h - 1\}$ was handled by a utility worker, and 0 otherwise. Formally, the variable $q_{k,h-\eta_k}$ corresponds to the following definition

$$q_{k,h-\eta_k} = \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h-\eta_k} \prod_{l=h-\eta_k}^{h-1} (1 - w_{k,l}). \tag{8}$$

This expression is linearized using the three inequalities

$$\begin{aligned} &q_{k,h-\eta_k} \le \textstyle\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h-\eta_k}, \\ &q_{k,h-\eta_k} \ge \textstyle\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h-\eta_k} - \sum_{l=h-\eta_k}^{h-1} w_{k,l}, \\ &2q_{k,h-\eta_k} \le 1 + \textstyle\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h-\eta_k} - \sum_{l=h-\eta_k}^{h-1} w_{k,l}. \end{aligned} \tag{9}$$

To illustrate, consider the example in Table 4 for a station $k \in K_F$ and $\nu_k = \mu_k = 1$ so the spacing constraint is $\nu_k : \eta_k = 1 : 2$. Table 4 provides the value of the variables included in Eq. (7) for a sequence of 3 items each requiring the option.

For $h = 1$ constraint (3) gives $m_{k,1} = \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,1} - 2w_{k,1}$, since $m_{k,0}$ and $q_{k,-3}$ have negative indices, they are ignored. Assuming there is no initial condition, we set to zero all variables with negative indices in Eq. (7). However, initial conditions (i.e. past decisions) will be taken into account in the experiment so this point will be discussed further in Section 5. We have $m_{k,1} = 1 \le \nu_k = 1$ with $w_{k,1} = 0$. Note that $w_{k,1}$ could have been equal to 1 without violating the spacing constraint. However, the objective is to minimize the cost associated with temporary workers so

Position h	1	2	3
$\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,h}$	1	1	1
$m_{k,h}$	1	0	1
$w_{k,h}$	0	1	0
$q_{k,h-\eta_k}$	/	/	0

Table 4: Spacing constraint for a flexible station $k \in K_F$ hosting a utility worker

variables $\{w_{k,h}\}$ will always take their minimum value while satisfying the spacing constraints. For $h = 2$, we have $m_{k,2} = m_{k,1} + \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,2} - 2w_{k,2} = 0$ with $m_{k,1} = 1$ and $w_{k,1} = 1$. A utility worker is required to process the second item as the spacing constraint is violated since items in position 1 and 2 both require the option. For $h = 3$, we get $m_{k,3} = m_{k,2} + \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,3} - q_{k,1} - 2w_{k,3}$ with $q_{k,1} = \sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,1} \prod_{l=1}^{2}(1 - w_{k,l})$ as defined by Eq. (8). We have $q_{k,1} = 0$ since the first item has the option but a utility worker was present for the second item and this leads to 'cancelling' the option on item 1 and 2. We have $m_{k,3} = 1 \leq 1$ with $w_{k,3} = 0$.

3.1.2 Optimal number of utility workers

In this paragraph we first provide a formulation of the objective function relative to the cost of hiring temporary workers for horizon T which is usually a day of production. This objective function includes variables representing the total number of utility workers hired per half-days. We then derive suitable constraints to link these variables to the binary variables $\{w_{k,h}\}$ reflecting the presence (or the absence) of utility workers on some station and item.

Let $W_1(T)$ be the number of active temporary workers in the first half of horizon T. Analogously, $W_2(T)$ designates the number of utility workers required to process items in the second half of horizon T. We finally define $W_3(T)$ as the number of utility workers that are needed to implement sequencing decisions made for horizon T but physically present only in the next horizon $T+1$. The objective is to minimize

$$\omega \cdot [W_1(T) - W_3(T-1) + W_2(T) + W_3(T)], \tag{10}$$

where ω is the cost of hiring a utility worker for one half-day, and the difference $W_1(T) - W_3(T-1)$ represents the number of utility workers resulting from hiring decisions made for T and actually present during the first half of horizon T. The economic function (10) therefore refers to hiring decisions aimed at implementing the sequencing of items for horizon T only. We shall now specify the relationships between variables $\{w_{k,h}\}$ and variables $W_1(T)$, $W_2(T)$, $W_3(T)$.

Any temporary worker required on station k has a working time of $\theta_k^{\max}$ on each item requiring the option. This implies that he is assigned to station k for a number of cycle times equals to $\beta_k = \lceil \theta_k^{\max} / \overline{\theta} \, \rceil$. Thus a utility worker that handles on station k the h^{th} item in the sequence becomes anew available for performing operations on the $(h + \beta_k)^{th}$ item, if need be, either on the same station or on any other station, assuming workers moving time between any pair of stations is ignored. Obviously,

$\beta_k \geq 2$ since $\theta_k^{\max} > \overline{\theta}$. From the assumption $(\theta_k^{\max} - \overline{\theta})\nu_k \leq \overline{\theta}$ we made in the previous paragraph, we get $\theta_k^{\max}/\overline{\theta} \leq 1 + 1/\nu_k$ with $1 + 1/\nu_k$ taking a maximum value of 2 when $\nu_k = 1$. We therefore have $\beta_k = 2$, $\forall k$. When $\nu_k > 1$, it should be noted that the utility worker spends less than 2 cycle times on station k so the remaining time can be considered as a free time for moving from one station to another.

Recall the horizon is divided into N cycle times. During the h^{th} cycle time, item in position $h-k+1$ is processed on station k. For instance, with no initial condition, $N = 6$ and $K_F = \{1, 2\}$, in the first cycle time, the first item in the sequence is processed on station 1 and no item is processed on station 2. In the second cycle time, the second item is processed on station 1 while item 1 crosses station 2 etc. As a utility worker stays $\beta_k = 2$ cycle times on any station station k each time he is required, the sum $w_{k,h-k} + w_{k,h-k+1}$ gives the number of utility workers operating on station k in the h^{th} cycle time. The sum $\sum_{k \in K_F}(w_{k,h-k} + w_{k,h-k+1})$ therefore represents the total number of active utility workers on all stations in the h^{th} cycle. It follows that the number of utility workers needed in the first half of horizon T is $W_1(T) = \max_{h=1,\ldots,[N/2]} \left\{\sum_{k \in K_F}(w_{k,h-k} + w_{k,h-k+1})\right\}$. A similar reasoning holds for $W_2(T)$ and $W_3(T)$. The linear version of the constraints associated with $W_1(T)$, $W_2(T)$, $W_3(T)$ is

$$\begin{array}{l} \sum_{k \in K_F} (w_{k,h-k} + w_{k,h-k+1}) \leq W_1(T),\ h = 1, \ldots, \left[\frac{N}{2}\right], \\ \sum_{k \in K_F} (w_{k,h-k} + w_{k,h-k+1}) \leq W_2(T),\ h = \left[\frac{N}{2}\right] + 1, \ldots, N, \\ \sum_{k \in K_F} \sum_{\substack{l=h-k \\ l \leq N}}^{h-k+1} w_{k,l} \leq W_3(T),\ h = N+1, \ldots, N+K-1. \end{array} \tag{11}$$

Let us note that at the end of horizon T, part of the sequence that has been decided in T is only processed and completed in $T + 1$. The third inequality in (11) reflects the hiring decisions referred to the sequencing of these items although the corresponding utility workers will only be present at the beginning of horizon $T + 1$.

3.1.3 Illustration

Let us consider a simple example with $K = 3$ stations, where stations $k = 1$ and $k = 3$ are flexible and $k = 2$ is neither a fixed-capacity station nor a flexible one. We assume that $N = 6$ items are sequenced during each horizon $T - 1$, T and $T + 1$. Three horizons are examined to clarify the interdependence between past, present and future decisions. With two options ($k = 1$, $k = 3$), 4 models of items are available: item A has no option, item B requires option $k = 1$ only; option $k = 3$ is exclusively installed on item C and item D includes both options. We assume $\nu_1 = \nu_3 = 1$ and $\mu_1 = 1$, $\mu_3 = 2$.

Table 5 illustrates the satisfaction of constraints (7) for stations $k = 1$ and $k = 3$, for each position h where $h = -5, \ldots, 0$ are the positions of items for which sequencing decisions are made during horizon $T - 1$, decisions for $h = 1, \ldots, 6$ are made for horizon T, etc. We indeed adopt the convention that negative positions $h \leq 0$ refer to past sequencing decisions whereas positions $h > N$ are relative to future sequencing. Table 5 also displays the values of $\{w_{k,h}\}$. For instance, we have

$w_{1,2} = 1$ meaning that a temporary worker must process on station 1 the item in the second position in the sequence which is decided during horizon T. This item—a model B—includes option $k = 1$ just like the previous one (a model D) which would have led to a violation of the spacing constraint $\nu_1 : \eta_1$ (i. e. $1 : 2$) if no worker could have temporarily been hired.

			$\sum_{i=1}^{N} \delta_{i,k} y_{i,h}$		$m_{k,h} \leq \nu_k = 1$		$w_{k,h}$		$q_{k,h-\eta_k}$	
Hor.	h	Model	$k=1$	$k=3$	$k=1$	$k=3$	$k=1$	$k=3$	$k=1$	$k=3$
$T-1$	-5	D	1	1	/	/	0	0	/	/
	-4	A	0	0	1	/	0	0	/	/
	-3	B	1	0	1	1	0	0	1	/
	-2	D	1	1	0	1	1	0	0	1
	-1	C	0	1	0	0	0	1	0	0
	0	C	0	1	0	1	0	0	0	0
T	1	D	1	1	1	0	0	1	0	0
	2	B	1	0	0	0	1	0	0	0
	3	A	0	0	0	0	0	0	0	0
	4	B	1	0	1	0	0	0	0	0
	5	D	1	1	0	1	1	0	0	0
	6	C	0	1	0	0	0	1	0	0
$T+1$	7	D	1	1	1	1	0	0	0	0
	8	B	1	0	0	1	1	0	0	0
	9	A	0	0	0	1	0	0	0	0
	10	D	1	1	1	1	0	0	0	1
	11	B	1	0	0	1	1	0	0	0
	12	A	0	0	0	1	0	0	0	0

Table 5: Spacing constraints for two flexible stations

Table 6 shows for each cycle time the items positions that are processed on each station. For instance, item in position $h = -5$ is processed on station $k = 1$ in the first cycle time of horizon $T - 1$. This item leads over station $k = 2$ in the second cycle time. It undergoes operations on station $k = 3$ in the third cycle time and is finally completed in the fourth cycle time, as indicated in the penultimate column of Table 6. Recall that a utility worker required on station k to perform operations on the h^{th} item is busy for 2 cycle times, $\forall k = 1, 3$. The activity of every utility worker is represented by a rectangle in Table 6. As an example, $w_{1,-2} = 1$ is symbolized by a rectangle around positions -2 and -1 indicating that the corresponding utility worker is busy on station 1 in the fourth and fifth cycle time of horizon $T - 1$. We also report in each rectangle the horizon in which the hiring decision has been taken. With such a table, we can visualize the activity of utility workers in the time so the number of necessary workers can easily be derived. The last column provides the number of active workers per cycle time. Considering horizon T, one utility worker is active on workstation $k = 3$ in the first cycle time and he is engaged for 2 cycle times. This means that this worker can not be allocated to station 1 in the second cycle time to perform operations on the item in position 2, thus 2 utility workers must be active in the second cycle time.

			Station		Completed	Nb of active
Horizon	Cycle	$k = 1$	$k = 2$	$k = 3$	item (pos.)	u. workers
	1	-5				0
	2	-4	-5			0
$T-1$	3	-3	-4	-5		0
	4	-2 $(T-1)$	-3	-4	-5	1
	5	-1	-2	-3	-4	1
	6	0	-1	-2	-3	0
	1	1	0	-1 $(T-1)$	-2	1
	2	2 (T)	1	0	-1	2
T	3	3	2	1 (T)	0	2
	4	4	3	2	1	1
	5	5 (T)	4	3	2	1
	6	6	5	4	3	1
	1	7	6	5	4	0
	2	8 $(T+1)$	7	6 (T)	5	2
$T+1$	3	9	8	7	6	2
	4	10	9	8	7	0
	5	11 $(T+1)$	10	9	8	1
	6	12	11	10	9	1
			12	11	10	
				12	11	
					12	

Table 6: Items positions on each station and activity of utility workers

Applying constraints (11) to our example yields $w_{1,0} + w_{1,1} + w_{3,-2} + w_{3,-1} \leq W_1(T)$ for $h = 1$, where the value of the sum $w_{1,0}+w_{1,1}+w_{3,-2}+w_{3,-1}$ can be read in the last column of Table 6 in the row relative to the first cycle time of horizon T. We get $W_1(T) \geq 1$, $W_1(T) \geq 2$, $W_1(T) \geq 2$ for $h = 1, 2, 3$ respectively. This leads to an optimal value $W_1^*(T) = 2$, assuming the sequence of items given for T in Table 5 is an optimal one given the cost parameters. As $W_1(T)$ is to be minimized, it takes its minimum possible value. Similarly we obtain $W_2^*(T) = 1$. The third inequality in (11) applies for $h = 7, 8$. We have $w_{1,6} + w_{3,4} + w_{3,5} \leq W_3(T)$ for $h = 7$ and $w_{3,5} + w_{3,6} \leq W_3(T)$ for $h = 8$. Note that $W_3(T)$ is only linked to variables $\{w_{k,h}\}$ associated with items positions that do not exceed $N = 6$. We have $W_3^*(T) = 1$. The optimal number of utility workers whose economic consequence is attributable to the sequencing decisions made for horizon T equals $W_1^*(T) - W_3^*(T-1) + W_2^*(T) + W_3^*(T) = 3$. In the first half of T, 2 operators are required but one of them has been decided in $T - 1$ so its wage is included in the economic function related to decisions made for horizon $T - 1$. A single operator is required in the second half-day and a utility worker that will be present only in the next day is actually paid in horizon T.

3.2 Minimizing sequence-dependent setups

Many operations necessitate sequence-dependent setups. For instance, in the automotive industry, vehicles of a same colour are grouped in an attempt to minimize the cost of cleaning sprayguns. We will focus here on such setup costs and we will consider a single paint station. The formulation below may easily be generalized to any number of stations and any type of setup costs (see Giard, 2003). A typical setup cost involves the cost associated with machine tune ups to switch from a model to another, e.g. a two-door vehicle to a four-door one. Vehicles of a same colour are grouped in batches but the batch size has an upper bound for two reasons. First, after some vehicles, sprayguns get paint-encrusted and need a purge. Second, a colour change is required to avoid eye tiredness responsible for a less effective paint flaw detection. We let L be the maximum number of cars in any identical-colour group. The binary variable u_h takes a value of 1 if the colour of the car in position h is different from the colour of the previous one in the sequence (position $h-1$) and 0 otherwise. The sum of variables u_h over the whole sequence expresses the number of purges. Letting γ be the unit cost of cleaning paint from sprayguns (solvent, labour and station downtime), the objective is to minimize

$$\gamma \cdot \sum_{h=1}^{N} u_h. \tag{12}$$

The constraint on the maximum group size is equivalent to enforce at least one colour switch between positions $h-L+1$ and h, $\forall h$. This is written as

$$\sum_{l=h-L+1}^{h} u_h \geq 1,\ h = 1, \ldots, N, \tag{13}$$

which indicates the number of colour changes any subsequence of L vehicles must be at least equal to one.

We must impose additional constraints on variables u_h so they take a value of 1 each time there is a colour switch in the sequence, and zero otherwise. The colour index of vehicle i is denoted by ρ_i so the colour index which is applied to the vehicle in position h is $\sum_{i=1}^{N} \rho_i \cdot y_{i,h}$. Variable u_h will take a value of 1 if a colour change occurs between position $h-1$ and position h, which is translated by a non zero difference of colour indices between position h and position $h-1$, that is $\sum_{i=1}^{N} \rho_i \cdot y_{i,h} - \sum_{i=1}^{N} \rho_i \cdot y_{i,h-1} \neq 0$. This condition is fulfilled when the two following inequalities hold:

$$\begin{array}{l} \sum_{i=1}^{N} \rho_i \cdot y_{i,h} - \sum_{i=1}^{N} \rho_i \cdot y_{i,h-1} \leq P \cdot u_h,\ h = 1, \ldots, N, \\ \sum_{i=1}^{N} \rho_i \cdot y_{i,h} - \sum_{i=1}^{N} \rho_i \cdot y_{i,h-1} \geq -P \cdot u_h,\ h = 1, \ldots, N. \end{array} \tag{14}$$

To illustrate, consider $N = 6$ vehicles and $P = 2$ colours (an index value of 1 is chosen for the blue colour and a value of 2 corresponds to the red hue). The maximum number of cars with same colours in a sequence is $L = 3$. In table 7, we give the values of relevant variables for a given sequence of vehicles for which colours are indicated. For each position h, the value of $\sum_{i=1}^{N} \rho_i \cdot y_{i,h}$ simply gives the colour index of the vehicle in position h (this is given in the 4th row of Table 7). The difference $\sum_{i=1}^{N} \rho_i \cdot y_{i,h} - \sum_{i=1}^{N} \rho_i \cdot y_{i,h-1}$ between the colour indices in position h and $h-1$ is non zero when the colour has changed between the current position and the previous one. This difference is given for each position h in the 5th row of the table (colour difference). During horizon T, this happens in positions $h = 1$ and $h = 3$ (and also for $h = 0$ which is the last position in horizon $T - 1$). Appropriate values for u_h are then obtained utilizing constraint (14). For instance, in position $h = 3$, we have $\sum_{i=1}^{6} \rho_i \cdot y_{i,3} - \sum_{i=1}^{6} \rho_i \cdot y_{i,2} = -1$ and we want u_3 such that $-2u_3 \leq -1 \leq 2u_3$ which implies $u_3 = 1$. Thus, a colour difference necesarily implies the implementation of a colour change. The last row of Table 7 checks constraint (13). This constraint is violated for $h = 6$ as no colour change has been done for the last 3 vehicles so there is more than 3 consecutive vehicles with same colour. Letting $\gamma = 10$ euros be the unit cost of a purge, the total cost of cleaning guns during horizon T amounts to 20 euros as $\sum_{h=1}^{6} u_h = 2$.

Horizon			$T-1$				T			
Position h	...	-2	-1	0	1	2	3	4	5	6
Colour name		red	red	blue	red	red	blue	blue	blue	blue
Colour index		2	2	1	2	2	1	1	1	1
Colour difference			0	-1	1	0	-1	0	0	0
Colour change (u_h)			0	1	1	0	1	0	0	0
$\sum_{l=h-2}^{h} u_h$					2	2	2	1	1	0

Table 7: Constraints on colour changes - illustration

4 Summary: notations, assumptions and the optimization problem

Tables 8 and 9 provide the notations we used so far. The optimization problem finally consists in minimizing the sum of the objective functions in (10) and (12) subject to constraints (1), (3) for all fixed-capacity stations in the set K_C, (7) for all flexible stations in K_F, (8), (9), (11), (13), (14). We made the following assumptions.

Assumption 1. There are only two possible values for the processing time on station k, $\theta_k^{\max}$ and $\theta_k^{\min}$ where $\theta_k^{\max} = \max_{i=1...,N} \{\theta_{i,k}\}$ (with at least one item i requiring option k) and $\theta_k^{\min} = \max_{i=1...,N} \left\{ \theta_{i,k} | \, \theta_{i,k} < \overline{\theta} \right\}$.

Thus, installing option k takes always the same processing time whatever the item type. All items without option k are assumed to have the same processing time $\theta_k^{\min}$ on station k although some of them may require a zero time which actually leads to a greater decrease in the excess working time than the one we considered. Without this assumption however the logic of spacing requirements on flexible stations must be abandoned in favor of a reasoning in terms of processing times.

Assumption 2. For any option o processed on several stations K_o with $K_o = \left\{ k | \, \theta_{o,k} > \overline{\theta} \right\}$ we associate option o with the most capacitated station $k \in K_o$.

PARAMETERS	
T	Sequencing horizon, usually a day
N	Number of items to be sequenced over T
$\overline{\theta} = \lfloor T/N \rfloor$	Cycle time
K	Set of all stations in the line
K_C	Subset of fixed-capacity stations
K_F	Subset of flexible stations
$O \equiv K_C \cup K_F$	Set of options
λ_k	Number of dedicated operators on station k
$\nu_k : \eta_k$	Spacing constraint: 'at most ν_k items with option k out of η_k consecutive items'
$\delta_{i,k}$	Boolean taking a value of 1 if item i requires option k and 0 otherwise
$\theta_{i,k}$	Processing time of item i on station k
$\theta_k^{\max}$	Processing time on station k of any item requiring the option k
$\theta_k^{\min}$	Processing time on station k of any item without option k
$R_k^{\max}$	Maximum excess working time allowed on station k
μ_k	Number of items without option to be sequenced after ν_k consecutive items with option k to avoid exceeding capacity, when $\lambda_k = 1$
β_k	Number of cycle times a utility worker spends on station k to process one item with option
ω	Cost of hiring a utility worker per half-day
γ	Set-up cost
L	Maximum number of consecutive items with same colour
P	Number of colours
ρ_i	Colour index chosen for item i, $1 \leq \rho_i \leq P$

Table 8: Notations for parameters

It should be noted that when a utility worker is allowed to work on such station k, this relaxes the spacing constraint so all stations in K_o must be considered separately. A simplifying assumption consists in considering that any option is fully processed by a single specialized workstation.

Assumption 3. Each flexible station has a single dedicated worker and may host a single utility worker per cycle time. *Assumption 4.* The maximum excess working time, $(\theta_k^{\max} - \overline{\theta})\nu_k$, without violating the capacity of any flexible station k is such that $(\theta_k^{\max} - \overline{\theta})\nu_k \leq \overline{\theta}$ so the regular worker fully absorbs the excess working time while the utility worker solely processes an item with option.

VARIABLES	
$y_{i,h} \in \{0,1\}$	Binary that takes a value of 1 if item i is placed in position h
$m_{k,h} \in \mathbb{N}$	Number of items with option k in the window $\{h - \eta_k + 1, \ldots, h\}$
$w_{k,h} \in \{0,1\}$	Binary that takes a value of 1 if a utility worker is required on station k to process the h^{th} item
$q_{k,h} \in \{0,1\}$	Binary that takes a value of 1 if item in position h has the option and none of the items in positions $\{h - \eta_k, \ldots, h-1\}$ is handled by a utility worker
$W_1(T) \in \mathbb{N}$	Number of active temporary workers in the first half of horizon T
$W_2(T) \in \mathbb{N}$	Number of utility workers required to process items in the second half of horizon T
$W_3(T) \in \mathbb{N}$	Number of utility workers needed to implement sequencing decisions made for T but physically present only in the next horizon $T+1$
$u_h \in \{0,1\}$	Binary that takes a value of 1 if a set-up is incurred to process the h^{th} item (e.g. a colour swith)

Table 9: Notations

5 Computational study

We first discuss the initial conditions. We then describe the experimental design. We finally turn to the comment of the results.

5.1 Initial conditions

In practice, sequencing decisions are made on a rolling horizon basis: the sequence implemented within horizon $T-1$ provides initial conditions to the sequencing problem for horizon T. In the experiment, we have generated a random sequencing of items for horizon $T-1$ and we have optimized the sequencing for horizon T. In the random sequencing, we have incorporated some rules in order to obtain a feasible sequence that satisfies the spacing constraints for fixed-capacity stations only. We then have computed the values of all variables corresponding to this feasible random sequence, some of them being included as initial parameters in the optimization program relative to horizon T.

5.2 Experimental design

The number N of vehicles to be sequenced during the horizon took the values in the set $\{10, 15, 20, 25, 30, 35\}$. For $N = 10$ and 15, the number P of colours was set to 3 with a maximum of $L = 3$ consecutive cars with same colour. For $N = 20$ and 25, we chose $P = 4$ and $L = 5$. Finally, for $N = 30$ and 35, we set $P = 5$ and $L = 6$. Note that for each value of N, parameters P and L are such that a reasonable number of subset of consecutive identical-colour of vehicles may be obtained.

The number K_C of fixed-capacity stations was set to 1 and 2 and the number

K_F of flexible stations took the values of 2 and 4, i.e. $(K_C, K_F) = \{(1,2),(2,4)\}$ leading to a total number of either 3 or 6 options. We set $\nu_k = 1$ for all k, so the spacing constraint for flexible stations is a necessary and sufficient condition not to exceed the capacity. We let $\eta_k \rightsquigarrow U\{2,3,4,5\}$ for all k so we get reasonable values for the spacing constraints with respect to the number of items to be sequenced.

Letting N_k be the number of vehicles with option k, we set $N_k = 0.75N\nu_k/\eta_k$ for all $k \in K_C$ and $N_k = 2N\nu_k/\eta_k$ for all $k \in K_F$. This makes easier the search for a feasible sequence satisfying the spacing constraints for fixed-capacity stations while making necessary the appointment of some utility workers.

Each item is uniformly picked at random to be endowed with option k until the number of items with option k reaches the value of N_k. The colours of cars are identically independently uniformly distributed over the set of possible colours. Thus there is on average a proportion $1/P$ of cars of each colour. In accordance with the french wage levels we set $\omega = 70$ euros. In the french automotive industry, the cost of cleaning the sprayguns approximates 10 euros so we set $\gamma = 10$ throughout.

For each problem defined in terms of values for N and for the pair (K_C, K_F), we made 3 replications and we obtained 30 instances since we only considered $(K_C, K_F) = (1,2)$ for $N \geq 30$. In the optimization programs, the spacing constraints for flexible stations were finally written as

$$\sum_{l=1}^{h} \left(\sum_{i=1}^{N} \delta_{i,k} \cdot y_{i,l} - (1+\nu_k) \cdot w_{k,l} \right) - \sum_{l=1}^{h-\eta_k} q_{k,l}, \;\; h = 1, \ldots, N; k \in K_F, \qquad (15)$$

instead of using (7) which would lead to a higher number of variables with the introduction of the 'intermediary' variables $\{m_{k,h}\}$. Analogously, we used the spacing constraints for capacitated stations as given by (2).

5.3 Results

Table 10 reports the results for the small instances (up to 15 vehicles) for which we get an optimal solution in a reasonable time through the use of the commercial software MIP solver ILOG-Cplex 9.0. The first three columns give the values of N, K_C and K_F which define a type of instance. The next two columns provide the number of variables and of constraints. For each type of problem and each of the three replications, we report the quality of the first feasible solution measured as the relative deviation (gap) to the optimal value. The penultimate column displays the execution time to reach the first feasible solution whereas the last column exhibits the time to obtain the optimum. Computing times are expressed in CPU seconds and refer to a Bi-Xeon 3.4 GHz with 4Go of main memory.

Except for one of the largest problems, it takes less than 2 sec. to get a first integer (feasible) solution. The quality of these first feasible solutions is rather heterogeneous, with a deviation to the optimal value ranging from about 15% to 75% with no obvious relationship between the deviation and the problem size. The execution time to reach the optimum seems to strongly depend on the structure of each problem. However, a relationship between the time and the problem size exists as

N	K_C	K_F	Var.	Const.	Gap (%)	Time	Time, Opt.
10	1	2	151	166	34.38	0.01	0.67
					32.26	0.01	0.27
					72.73	0.03	0.65
10	2	4	189	273	19.15	0.98	14.33
					17.02	1.99	79.99
					35.90	1.26	240.65
15	1	2	301	251	54.55	0.04	3.83
					75.00	1.41	648.98
					57.58	0.03	0.23
15	2	4	359	413	18.03	0.24	66.17
					46.00	15.87	1026.38
					14.52	0.12	47.19

Table 10: Feasible and optimal solution to small instances

the largest time is associated with the biggest problem and the smallest instances are also the fastest problems to be solved to optimality.

Table 11 displays the results for larger instances (up to 35 vehicles) for which Cplex was not able to find the optimal solution within the time limit of 7200 CPU seconds. Like in the previous table, the first five columns describe the instances. The next two columns report the quality of the first feasible solution and the time required by Cplex to reach it. The quality is measured by the relative gap between the first feasible solution and the best bound. The penultimate column displays the gap between the best integer solution delivered by Cplex after 7200 sec. and the best bound. The last column provides the improvement of the objective value after 7200 sec. compared to the first feasible solution (for instance, the best objective value is 23.40% lower than the first feasible solution to the first problem).

For instances up to 20 vehicles Cplex delivers solutions of good quality within the 7200-second time limit. For 13 cases out of 18, Cplex is able to provide a first feasible solution in less than 3 min. Even if the solution quality after 7200 sec. rapidly degrades as larger instances are considered, the solution to 5 instances over 18 exhibits a gap below 5%. In all cases, the first feasible solution is significantly improved during subsequent iterations with a decrease in the objective value of 47.75% on average.

6 Conclusion

In this paper we have developed an optimal formulation for the sequencing problem where simultaneous minimization of the number of utility workers and setups is desired. The resultant optimization model is more realistic than those of earlier papers dealing with such goals since it provides an operational way to implement the utility work needed to avoid line stoppages. Although the proposed approach here can hardly deal with real-size problems, it may lay down the basis for developing heuristic methods.

					First feas. sol.		After 7200 sec.	
N	K_C	K_F	Var.	Const.	Gap (%)	Time	Gap (%)	Imp. (%)
20	1	2	501	336	26.09	0.06	2.17	23.40
					40.63	0.20	3.13	36.36
					65.63	0.15	3.13	60.61
20	2	4	579	553	60.38	468.59	5.00	34.92
					49.06	4.06	14.46	29.51
					50.94	14.60	16.67	14.29
25	1	2	751	420	275.00	11.09	150.00	50.00
					291.67	93.34	95.61	95.83
					66.67	0.63	3.03	61.76
25	2	4	849	693	139.39	1703.85	77.67	33.90
					80.00	424.21	70.00	5.88
					78.79	23.45	66.67	7.27
30	1	2	1051	505	308.33	12.64	158.33	58.06
					308.33	16.37	108.33	96.00
					308.33	163.80	108.33	96.00
35	1	2	1401	590	225.00	735.13	158.33	25.81
					140.74	16.55	39.06	58.54
					361.54	422.38	169.23	71.43

Table 11: Feasible and best integer solution to larger instances

References

[1] J. Bautista, R. Companys, and A. Corominas. Heuristics and exact algorithms to solve the monden problem. *European Journal of Operational Research*, 88:101–113, 1996.

[2] J. Bautista, R. Companys, and A. Corominas. Note on cyclic sequences in the product rate variation problem. *European Journal of Operational Research*, pages 468–477, 2000.

[3] L. D. Burns and C. F. Daganzo. Assembly line job sequencing principles. *International Journal of Production Research*, 25:71–99, 1987.

[4] W. Choi and H Shin. A real-time sequence control system for the level production of the automobile assembly line. *Computers ind. Engng*, pages 769–772, 1997.

[5] F.-Y. Ding and L. Cheng. An effective mixed model assembly line sequencing heuristic for just-in-time production systems. *Journal of Operations Management*, pages 45–50, 1993.

[6] A. Drexl and A. Kimms. Sequencing jit mixed-model assembly lines under station-load and part-usage constraints. *Management Science*, 47:480–491, 2001.

[7] V. Giard. *Gestion de la Production et des Flux.* Economica, 2003.

[8] J. Gottlieb, M. Puchta, and C. Solnon. *A Study of Greedy, Local Search and Ant Colony Optimisation Approaches for Car Sequencing Problems.* Lecture Notes in Computer Science, Springer, 2003.

[9] C. J. Hyun, Y. Kim, and Y. K. Kim. A genetic algorithm for multiple objective sequencing problems in mixed model assembly lines. *Computers and Operations Research*, pages 675–690, 1998.

[10] T. Korkmazel and S. Meral. bicriteria sequencing methods for the mixed-model assembly line in just-in-time production systems. *european journal of operational research*, pages 188–207, 2001.

[11] W. Kubiak and S. Sethi. A note on level schedules for mixed-model assembly lines in just-in-time production systems. *Management Science*, 37:121–122, 1991.

[12] Y. Y. Leu, A. L. Matheson, and L. P. Rees. Sequencing mixed-model assembly lines with genetic algorithms. *Computers and Industrial Engineering*, 30:1027–1036, 1996.

[13] J. L. Macaskill. Computer simulation for mixed-model production lines. *Management Science*, 20:341–348, 1973.

[14] S. A. Mansouri. A multi-objective genetic algorithm for mixed-model sequencing on jit assembly lines. *European Journal of Operational Research*, 167:696–716, 2005.

[15] J. Miltenburg. Level schedules for mixed-model assembly lines in just-in-time production systems. *Management Science*, 35:192–207, 1989.

[16] S. Mitsumori. Optimal schedule control of conveyor line. *IEEE Transactions on Automatic Control*, 10:633–639, 1969.

[17] Y. Monden. *Toyota Production System, an Integrated Approach to Just-in-Time.* Engineering and Management Press, Norcross, Georgia, 1998.

[18] T. Murata, H. Ishibuchi, and H. Tanaka. Multi-objective algorithm and its applications to flowshop scheduling. *Computers and Industrial Engineering*, 30:957–968, 1996.

[19] K. Smith, M. Palaniswami, and M. Krishnamoorthy. Traditional heuristic versus hopfield neural network approaches to a car sequencing problem. *European Journal of Operational Research*, pages 300–316, 1996.

[20] R. T. Sumichrast, R. S. Russell, and B. W. Taylor. A comparative analysis of sequencing procedures for mixed model assembly lines in just-in-time production system. *International Journal of Production Research*, pages 199–214, 1992.

[21] T. Tamura, H. Long, and K. Ohno. A sequencing problem to level part usage rates and work loads for a mixed model assembly line with a bypass subline. *International Journal of Production Economics*, pages 557–564, 1999.

[22] N. T. Thomopoulos. Line balancing-sequencing for mixed-model assembly. *Management Science*, 14:59–75, 1967.

[23] Z. Xiaobo and K. Ohno. algorithms for sequencing mixed models on an assembly line in a jit production system. *Computers ind. Engng*, pages 47–56, 1997.