CHAPTER 8

BALANCE FOR OPERATION

The product development process is concerned,
not only with the design of products,
but with how these products are manufactured,
distributed and serviced.

Barry O Sullivan
Department of Computer Science
National University of Ireland, Cork

Keywords: assembly line, line balancing, line layout, line operation phase, scheduling.

1. Introduction

As seen in Chapter 3, more and more manufacturers are using mixed model (multi-product) assembly line (AL). In their research and development efforts firms can make economies by producing a range of related products as well as in selling a variety of related products. Manufacturing firms have thus, shifted from producing single product line to multiple product line. The products diversification create problems that have occupied (and still occupying) manufacturing innovators for the last decades. Producing multiple products in a single factory yields at least two major problems to be overcome:

1. getting the right input (part or work-in-progress) to a right place at a right time;
2. determining the best production run to match supply with demand.

The pragmatic approach (which is not a solution) to the first problem is to pile inventory of inputs at each station in the production process to ease the timing problem. Operators or machines can find a correct part in the inventory. However, the inventory and work-in-progress is a financial investment that garners no rate of return.

The economics of the second problem are determined by the cost and time to changeover the production of one product to another. The more costly the changeover and the longer the required setup-time, the longer must be run the production system to distribute the fixed costs of changeover. But again, long production runs also imply hidden costs in that the inventory of final products does not garner a rate of return until payment from the sale. Finally, the faster and cheaper the changeover, the greater a variety of products can be produced at a single factory.

In MPAL, the variants are represented by a generic product model which contains the necessary information. These products are characterised by variations in model styles, options, etc. Several technical problems are associated with the design and operation of MPAL, the most important ones consist in generic product modelling, operating modes and assembly techniques, line layout and model launching (ordering variants). The *model launching* (ML) is concerned with the scheduling the different models to be produced during a given work shift.

> *Since one of the main aims of the LB and RP is to balance the workload of the stations, in the following the term assembly line design (ALD) or assembly line balancing (ALB) means without confusion the same thing.*

The ALD and the ML are interrelated because the balancing solution affects the determination of the launching schedule. In this chapter, we will discuss a concurrent strategy for product family and assembly system development. The emphasis is on the modelling of assembled product families, their corresponding line balancing and finally the scheduling process.

Perhaps the confusing aspect of *scheduling* is its interaction with *design*. Design and scheduling are often grouped together in manufacturing domain applications. Product design can be thought of as the process that decides what kind of product to do. Alternatively, scheduling is the process which decides about the order to execute tasks specified by designers. Instead of analysing the scheduling *theoretically*, the problem can be approached from a different point of view. Indeed, we can simulate reality by building a model based on a given production environment. If several different products are manufactured on the same assembly line their sequence has an effect on the *throughput*[1] because the *demands can vary* considerably from time to time (Chow, 1990).

Section 2 presents general features of the mixed-production assembly line design problem. Related works concerning design of multi-product assembly line are presented in section 3. The essentials on the ordering genetic algorithms (OGA) are given in section 4. The general architecture of the balance for operation concept is introduced in section 5. We draw some conclusions in section 6.

---

[1] The amount of production per time period.

2. Multi-product assembly line

The innovations leading to production for final demand in firms producing multiple product create a difficult scheduling problem. Indeed, factories have to shift from one product to another satisfying customer demands, minimising inventory and at the same time fully utilising the production resources. Thus, multi-product assembly has become an important issue in industry because of market pressures for diversity, shorter life and competition. However, diversity is in contradiction with other current goals such as low costs, high productivity and standardisation. Thus company's ultimate success depends on its ability to deal with the complexity of product and process design. It has become clear that assembly and manufacturing questions must be taken into account from the product design. Design for manufacturing (DFM), design for assembly (DFA) have reported significant benefits such as product simplification, lower assembly and manufacturing costs, improved quality and reduced time to market (Delchambre, 1996).

In the automotive industry, a typical example is a family of cars with different options: some cars have a sunroof, some have the ABS brake system and so on. These cars are assembled on the same AL (Falkenauer, 1993) (Shimokawa, 1997).

A family of products is a set of products with common functions and some characteristics differing between them. Variants are products which are members of the same family. Traditionally because of the variability of process time of each station (this is due to the fact that some operations are missing for some variants), the multi-product assembly line is balanced on average. The aim is to minimise the imbalance measured by the difference between the average cycle time and the total duration of operations concerning a variant on each station. A large imbalance of the workload among different variants has to be rejected. Indeed, even when the load of the stations is balanced on average, the production can easily be faced with work *overflow* or *starvation* on individual stations. Thus, work imbalance due to the variants should be distributed among the stations. Local buffers are introduced in order to maintain the pace of the line as even as possible. Each station has a local buffer[2] (upstream). Stations process one unit at time, and are linked to buffers by conveyors.

When a station completes a process, a product is moved to a downstream buffer if possible (sufficient place), in the other case it remains at the station until it can move. Once a station is free, it takes a product from the upstream buffer if it is possible and processes a new job.

The aim of the multi-product line balancing is to provide a unique AL, valid for all the variants of the product family. A decision is required to assign an operation to a station in order to meet production requirements of a set of products at a minimum total cost.

---

[2] Buffers allow stations to operate independently, cushioning against machine failure, worker or part shortage, and production rate difference. A large buffers increase throughput time, space requirements, material handling costs, etc. Thus, the buffers size must be reduced as possible.

The objective is to assign operations to stations in order to balance the workload and minimise the number of stations when more than one product variant has to be assembled. Since assembly process and process times may not be the same for different products, a single line cannot be perfectly balanced for each product. The ordering variants problem occurs when a number of products (variants of product family) are produced on an assembly line at the same time. Each job[3] must be processed by each station exactly once. Furthermore, all jobs have the same routing, i.e. they must visit the stations in the same order. Without loss of generality, we can number the stations so that station 1 is first, station 2 is second, and so forth. A job cannot begin processing on the second station until it has completed processing on the first. The objective is to determine the sequence of variants which maximises the utilisation of the assembly stations.

Traditionally assembly line design and ordering variants have been considered as two separate but related problems (Thomopoulos, 1967). Most research on assembly line sequencing considers the scheduling problem the ALB problem. By separating the two problems, sub-optimal solutions are often obtained. Even if the balancing and the ordering can be solved optimally, the optimality of the overall solution may not be obtained by solving the two problems separately. Optimal solution must be found by treating the two problems simultaneously. Chevalier (Chevalier, 1999) pointed out that there is a strong link between how a line is designed and how it can be operated. The author investigated how models from the mechanical engineering literature can be combined with models originating from the management science literature. A concurrent approach can be the only way to address the full complexity of the problem.

In its most general form, the model launching problem is defined by:

- a set of products to be produced;
- a set of tasks that must be executed on the different products;
- a set of stations on which a set of tasks have to be performed;
- a set of constraints which must be satisfied;
- a set of measures (objectives) to judge the schedule performances.

What is the best way to order products at the entry of the assembly line so that all the constraints are satisfied and the best objective measures is reached? The general problem encapsulates many variations such as flow-shop and job-shop, production scheduling problems.

Most companies forecast what they expect to produce each year, month, week, day or period. Typically this is part of the *strategic plan* of the company. The quantity to be produced is often accompanied by *probabilistic* information about the likelihood of meeting these *forecasts*. In designing facilities it is important to take into consideration such probabilistic information. A good facility plan allows for capacity *expansion* if

[3] A job is defined as an activity that transforms inputs (a set of requirements) to outputs (products to meet those requirements).

sales are stronger than expected and a *reduction* plan in the case that actual sales are lower than the forecasts.

There exist two choices concerning 'when to schedule': *on-line* and *off-line* methods. Off-line scheduling (called static scheduling) refers to the formation of a complete schedule before launching production. An off-line schedule suffers from the lack of information on the real behaviour of the system. To combat this lack of information, several schedules called *contingency* schedules are constructed to represent the situations that may occur during run-time (Romanowicz, 1997). The alternative to off-line scheduling is on-line scheduling (called dynamic scheduling). Scheduling is performed incrementally while the facility is operating. As each decision problem arises, the results are immediately evaluated and a choice is made. The dynamic scheduling is under a severe real-time constraint. Since the proposed concept belongs to the design phase, for the rest of this chapter, discussion will be limited to off-line scheduling issues.

In general, in the case of flow-shop installations the order of products is deterministically known from the current schedule. Unfortunately, two philosophies of thought confuse the issue. The first claims that the schedule should contain precise event times. This is usually used in on-line scheduling. The second school of thought places the importance on the relative order of the events and not on their exact times. Due to unavoidable variance in processing times, the research presented in this book will operate under the second philosophy. That is, the exact order of operations should be determined by the schedule, but not the times of those events. This philosophy is more adapted to off-line scheduling.

The proposed scheduling (ordering) algorithm is part of the 'BFO' concept and it is used as test method in the line layout design module. It can be used as it to schedule variants in real time.

3. State of the art

It is not unusual to find MPAL in industry in which the sequencing of models (variants) is done without applying any of the heuristics introduced below. Instead, simple rules-of-thumb and experience factors influence selection of the model sequence. For example, if the total daily production is 100 units (vehicles), and includes 20 units of variant V1, 10 units of variant V2, and so forth, then every fifth unit launched will be a vehicle of type V1, every tenth unit will be a car of type V2 and so on.

Although the roots of the scheduling problem can be traced back in time, active research in this field began with the creation of computers. Linear programming became the first formulation of scheduling problems and especially after the invention of the simplex algorithm. Many techniques appeared after, the most famous are Monte Carlo simulation techniques, stochastic optimisation, queuing theory, integer programming, etc.

Various objectives have been considered in MPAL problems, the most cited are: minimise total utility work (makespan), keep a constant rate of part usage, minimise total setup cost, minimise the risk of stopping a conveyor, minimise the overall line length, and so on. Many line balancing methods used for MPAL are adaptations of those used to solve SPAL problems. The first research addressing the MPAL sequencing problem was apparently presented in (Wester, 1964), although the problem obviously existed in industry prior to their work and attempts must have been made to deal with it before.

Most scheduling problems belong to the class of NP-hard problems. Nobody knows about the existence of a polynomial bounded algorithm for these problems. Exact solution methods are thus of limited practical relevance in obtaining better performances. As a result, most research has been focussed on either simplifying the scheduling problem (mostly by making some assumptions) to the point that it is solvable by some algorithms within reasonable time limits or devising efficient heuristics for finding acceptable (not necessarily optimal) solutions. In the sequel are presented approaches that deal, without any separation, with ALB and RP and are subdivided into exact methods and heuristic approaches.

3.1. Classical methods

Monden (Monden, 1983) introduced a goal chasing (GC) method which is based on the part usage goal. The problem is formulated as a mixed integer programming model. The objective of the method is to keep a constant speed in the consumption of each part on the MPAL. The main idea behind the method is the fact that if products with relatively longer processing times are successively fed into the AL, a delay in model completion will eventually occur, which may stop the line. Thus, in order to avoid this problem, the processing time at each station must be smoothed by sequencing models so that a model with relatively short processing time at a station follows soon after a model with relatively long processing time at this station.

Berger et al. (Berger, 1992) developed a branch-and-bound algorithm for solving a *tree*[4] assembly line balancing problem (TALB), which is special case of the multi-product assembly line balancing problem. The precedence graph is not necessarily a *forest*[5], but may be any directed acyclic graph. The aim is to minimise the number of stations necessary to manufacture all products. The approach exploits a lower-bound procedure as well as a partitioning scheme. The algorithm can be used either as a heuristic (in its truncated version) or as an exact method (in its full version). The method gives good results for TALB problems and behaves poorly in the case of the general problem.

Bard et al. (Bard, 1992) formulated the sequencing problem as an integer programming in order to establish a common mathematical framework that might be

---

[4] Trees are the quintessential nontrivial recursively defined objects: a tree is either empty or a root node connected to a sequence (or a multi-set) of trees.

[5] A forest is a number of disjoint trees.

applicable to various MPAL configurations. The model involved two objectives: minimising the overall line length and keeping a constant rate of part usage. Analysis of several test problems revealed that solutions that minimised line length were not significantly different from those that minimised throughput time.

Agnetis et al. (Agnetis, 1995) presented a dynamic programming polynomial algorithm for solving the problem for both single-type and multi-type production. The method deals with flow management problems in flexible assembly systems. The system consists of a set of machines which perform the assembly of a number of units, possibly different from each other. Each unit requires a set of operations. Each machine can be tooled in order to perform any of the required operations. The aim is to assign operations to machines and to synchronise the sub-assembly lines. The objective functions considered are the makespan and the total tooling cost.

Sarker and Pan (Sarker, 1998) presented two models for open-station and closed-station[6] systems, with an objective of minimising the utility and idle time cost for MPAL. Their study showed that the optimal line parameters, such as launching interval, station length, and operator's moves have dominant effects on the throughput of the line. Due to the complexity of the model which is based on the mixed-integer linear programming, the computational time tended to be very long when the number of stations and the number of models increased.

Miltenburg (Miltenburg, 1998) proposed an amelioration of the GC method (Monden, 1983). The objectives look like Monden's but it also tries to minimise the gap between the ideal consumption and the real one. Miltenburg assumed that all products required the same number of mixs. The GC method is a local optimisation. Thus, since the scheduling problems are very constrained, quickly obtained solutions are not always good enough and can create bad drifts. These methods also concentrate on balancing the workload on each station taken lonely and not among all the stations[7]. The difference of assembly times between jobs has not been considered in the goal chasing method. Recently, a time-based GC (TBGC) method that considers the different assembly times between jobs has been proposed by Kurashige et al. (Kurashige, 1999). The method seems very interesting, unfortunately it is limited to a single station.

3.2. Heuristics

Driscoll (Driscoll, 1985) presented an integrated line balancing and simulation based evaluation technique to address the line balancing problem tacking into account stochastic task durations, mixed-model processing, task times greater than cycle time, and zoning requirements. The technique first performs a line balance using the

[6] A station may be open or closed depending on whether or not the operator working in it is allowed to cross its boundaries.

[7] The average process time of stations is not balanced along the line, but each station has approximately the same process time on all its variants,

ranked positional weight (RPW) technique (Helgeson, 1961), then it performs simulations to assess the performance of the layout.

Wang and Wilson (Wang, 1986) compared several assembly line designs in terms of station idle time, incomplete units, and production rate. Station process time were assumed to be variable. They proposed a sequencing heuristic while a simulation was used to evaluate the performance of the different solutions. They conclude that an accumulation conveyor with non fixed products on pallets would improve throughput and reduce worker idle time and utility time in comparison to a continuously moving conveyor with fixed product on pallets.

Fernandez (Fernandez, 1995) proposed a mixed-model launching algorithm. The objective is to minimise the sum of squares of the deviations from a perfect model sequence in which idle time and work congestion are both zero. The author also surveyed the research literature on mixed-model assembly line. The principal problems in the design and operation of assembly line (balancing, model launching, process time variability, etc.) are discussed, as well as the methods for their solutions.

In case of human workers, systematic reductions of the process time of tasks are possible due to learning effects or successive improvements of the production process. Thus, manual lines have to be balanced on average, the accent must not be highly put on the process time. Bartholdi (Bartholdi, 1996) introduced the concept of ‘operation of bucket brigades’ for manual ALs. The concept is well suited for the automotive industry. It works as follows: each worker carries a task towards completion; when the last worker finishes his product he sends it off and then walks back upstream to take over the work of his predecessor, who walks back and takes over the work of his predecessor and so on. In such a configuration the workers play the role of the buffers, by supporting the variability of the process times (due to stochastic phenomena or differences between the operations according to the different variants). As in many types of work cells, there are fewer workers than stations; but the distinctive feature of the bucket-brigade is that the workers maintain their sequence. The authors ignored some issues that are important to the effectiveness of their implementations, such as the assignment of work content to stations, the detailed choreography of worker movement, the strategic training and motivation of workers, etc.

Wall (Wall, 1996) proposed a genetic algorithm to deal with the resource-constrained scheduling using a direct, time-based representation (most methods are typically sequence-based). They defined objective measures such as minimisation of makespan or minimisation of average tardiness. The proposed GA is very promising, however, it needs more reflection on the encoding. The method did not perform well on some basic job-shop problems. Indeed, the method was designed to deal with a more complex problem which is a resource-constrained scheduling problem.

Romanowicz (Romanowicz, 1997) developed an expert system called scheduling method choice tool (SMCT) that proposes scheduling methods to the user on the basis of information he has supplied on his system. The author also gave a good survey of existing methods addressing scheduling problems.

McMullen and Frazier (McMullen, 1997) presented an approach for solving a mixed-model assembly line balancing problem with stochastic task times and paralleling of tasks within workcenters. The authors presented and compared ten task-selection rules using the output performance measures and requirements of workers and equipments. The resulting layouts were simulated and the performance results were analysed. The primary goal of the authors was to provide a methodology that addresses the mixed-production problem. The method yields many questions like (1) the effect of the rules on the output performances measures, (2) which is the task-selection rule which provides the best layout, (3) how the task-selection rules influence the number of workers.

(Minzu, 1997) proposed a 'Kangaroo' algorithm (a stochastic descent method) to treat multi-product assembly line with a fixed number of stations. The precedence graph of the family of products is obtained by merging the precedence graph of the variant products. The stochastic descent method aims to minimise the maximum work content of the stations, which leads to a well balanced line.

He and Kusiak (He, 1997) proposed a special method to deal with assembly system for modular products (a set of sub-assemblies). The assembly line is decomposed into two parts: a subassembly line for basic tasks and a subassembly line for variant tasks. The basic subassembly line is designed as a paced single product assembly line with a fixed cycle time. The variant subassembly line is designed as flow-shop line and balanced by a flow-shop scheduling method. In order to absorb the unbalanced flow of products, a buffer between the two subassembly lines is used. The approach is very interesting, however, one have to find an easy and an economic way to cluster tasks into basic and variants ones.

Hyun et al. (Hyun, 1998) presented a multiple objective GA to obtain diverse near-pareto optimal solutions to multiple objective sequencing problems. They considered three objectives: minimise total utility work, level the part usage, and minimise the total setup cost. Since the objectives may be in conflict, there may not exist a solution that optimise all the objectives at the same time. Thus, the authors used the Pareto optimality technique, that is, the method seeks a diverse non-dominated solutions.

Bukchin (Bukchin, 1998) used five performance measures for throughput using simulation. An experimental study was developed to evaluate the validity of each measure in a wide range of problem environments. The performance measures were: (1) smoothness[8] of a station, (2) idle time, (3) station process time variation, (4) bottleneck, (5) model variability. The study showed that the bottleneck measure performed better than other measures and that it is fairly robust to changes of the line configuration and operating environment. The results also indicated that the

---

[8] The smoothness index of a station measures the standard deviation of the distribution of work among the variants.

absolute quality of all these measures decreased if line length increase. This is due to the fact that the *starvation* and *blockage* phenomenon have a great impact on long ALs. The rest of the measures seem to be less valid and hardly correlated with simulations results.

Choi and Lee (Choi, 1998) proposed a heuristic method for mixed-model assembly line. This heuristic takes in consideration the product mix and the production sequence of the products (the product mix is assumed to be fixed). The total line length necessary for a given AL depends on the production sequence. The authors proposed a method which achieve the distribution of work-element of each model.

Distributing material flows among stations of a plant (designed for large-mix productions) is necessary so as to minimise production costs. The general tendency is to reduce setup-times and to assign operations according to the available production resources. Most of the models available in the literature tend to decompose the problem into a routing problem and a scheduling problem. Arbib et al. (Arbib, 1999) presented a method that first solves the routing problem in order to minimise setup times, and then face the problem of deciding a job sequence meeting suitable requirements (due-dates, production costs, scheduling, etc.).

In MPAL the production efficiency decreases when the variant models differ significantly. In order to reduce this production inefficiency, Tamura (Tamura, 1999) proposed to install a bypass sub-line adjacent to the main line. The sub-line will execute a portion of assembly tasks for product models with relatively longer assembly times. The author developed a heuristic algorithm based upon tabu search method (TS) and dynamic programming (DP). The DP technique is used to assign tasks to stations, while the TS technique is used as a local improvement method. In his formulation, models processed at the sub-line are assumed to be known.

For a comprehensive review, see the overview of the different MPAL methods presented by Scholl (Scholl, 1999). We believe that more studies are needed to better understand the working principles of MPAL.

Once the product to be made is determined, we must determine the quantities as well as the frequency to made it (scheduling problem). In the next section is introduced the ordering genetic algorithm, used to schedule variants on multi-product assembly line. The balancing methods were introduced in Chapter 6.

4. Ordering genetic algorithm

The ordering variants problem occurs when a number of products (variants) are produced at the same time. Previous research on MPAL sequencing suggests that analytical methods become inefficient for large problems. In these problems, the number of jobs to be scheduled is such that methods such as B&B cannot efficiently solve them. Finding optimal makespan schedules for more than three stations is difficult. During the last decades, the problem has captured the interest of a significant number of researchers, and many solutions methods have been proposed and specially metaheuristics.

4.1. Algorithm

In this section, we first introduce the ordering genetic algorithm (OGA), used to schedule variants on multi-product assembly line. We then present the different heuristics used as initialisation techniques.

4.1.1. Encoding Scheme

Because the ordering variants problem is essentially a permutation problem, we can use the permutation of jobs (variants) as representation scheme of solutions, which is the natural representation[9] for sequencing problems. For example, let the genes of the given chromosome be: $C = [1\,2\,1\,3\,1\,2]$. This mean that the job sequence is v1, v2, v1, v3, v1, v2 (vi = variant *i*) (see Figure 8.1).

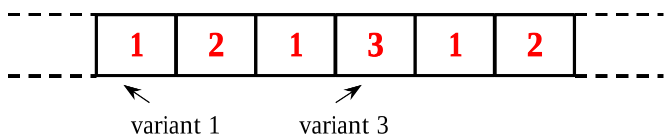

Figure 8.1. An ordering of variants and the corresponding OGA chromosome.

4.1.2. Crossover

A crossover's job consists of producing offspring out of two parents in such a way that the children inherit as much as possible of the meaningful information from both parents. Various techniques like PMX (Partially Mapped Crossover) (Goldberg, 1989), OX (Order crossover) (Davis, 1985) and PBX (Position Based Crossover) (Gen, 1997) are known from literature. We have used the PMX crossover and the PBX heavily modified to suit constraints of our problem.

4.1.3. Mutation

The mutation operator proceeds by changing the order of the job making idle time. According to the nature of the ordering problem, one or more of the following operators can be applied.

- Shift the place of a randomly selected job (variant).

[9] The choice of representation controls the size of the search space. If one chooses a very general representation, more types of problems may be solved at the expense of searching a larger space. Conversely, one may choose a very specific representation that significantly reduces the size of the search, but will work on only a single problem instance (Culberson, 1998).

- Invert a place of two selected jobs.
- Shift a selected job from pos1 to pos2.
- Select a job and find a place to insert it so as to minimise the makespan.

4.1.4. Inversion

The inversion serves to shorten promising schemata made of coadapted genes. In OGA, the mechanism is the same as the operator of (Holland, 1975), i.e. a segment on the chromosome is selected at random and the order of genes in that segment is inverted.

4.1.5. Evaluation

Since in our case the aim is to minimise the total production time, a simple way to determine the fitness for each chromosome is to use the makespan (total assembly time),

$$Eval(chromosome) = makespan$$

A fast *simulation* procedure (algorithm) is used to estimate the total assembly time (makespan) of a given mix (permutation of variants). Variants are introduced in the order given by their corresponding chromosome (first in first out). We take only the beginning and the end of production time of variants to estimate the total production. Setup-time is taken into account while estimating production time of each variant, i.e. if *variant 1* is followed by *variant 2*, setup time is added to operating time of *variant 2*. The total production time is reached when the last operator (last station) finishes his job on the last variant of the mix. Idle time is the time lost by operators (machines or robots) waiting for jobs, and it is calculated as follows:

$$IdleTime_{i,j} = Begin_{i,j} - End_{i,j-1}$$

where:

$IdleTime_{i,j}$	is time lost by operators waiting for job *j* at station *i*,
$Begin_{i,j}$	is the beginning production time of variant *j* on station *i*,
$End_{i,j}$	is the end production time of variant *j* on station *i*,

The makespan is given by

$$Makespan = End_{lw,lj} - Begin_{1,1}$$

where

Makespan	is total production time of the mix,
$End_{lw,lj}$	is the end production time of the last station on the last job,
lw	is last station of the assembly line,
lj	is last job (variant) of the mix of products.

4.2. Heuristics

Several heuristics are used to construct valid solutions (permutation of variants).

– **Random lots** Jobs are inserted in random manner in chromosome. It permits to avoid local optima.
– **Lot by lot (batch by batch)** If the total daily production is 100 units, that includes 20 units of variant V1, 10 units of variant V2, and so forth, the 20 units of type V1 will be launched, then the 10 units of type V2 and so on.
– **Mix percent** If the total daily production is 100 units, that includes 20 units of variant V1, 10 units of variant V2, and so forth, every fifth unit launched will be a job of type V1, every tenth unit will be a job of type V2 and so on.
– **Slope order index** The idea is to give higher priority to jobs with processing times that tend to increase from station to station, while jobs with processing times that tend to decrease from station to station will receive lower priority (Gen, 1997). The slope index s_i for job i is calculated as:

$$s_i = \sum_{j=1}^{m} (2j - m - 1) t_{ij}$$

where t_{ij} is the process time of station j on task i, m the number of stations and n number of jobs. Then a permutation schedule is contructed by sequencing the jobs in a nonincreasing order of s_i such as $s_{i1} \geq s_{i2} \geq \ldots \geq s_{in}$.
– **Gupta's heuristic** Is similar to the slope order index heuristic, except that it takes into account some interesting facts about optimality of Johnson's (Johnson, 1954) rule for the three-station problem (Gen, 1997). The slope index s_i for job i is calculated as

$$s_i = \frac{e_i}{\min\limits_{1 \leq k \leq m-1} \{t_{i,k} + t_{i,k+1}\}} \quad \text{where } e_i = \begin{cases} 1 & if\ t_{i1} < t_{im} \\ -1 & if\ t_{i1} < t_{im} \end{cases}$$

Thereafter the jobs are sequenced according to the slope index s_i.

These heuristics are used each time we are about to construct new solutions or to improve the quality of existing ones.

The multi-product assembly line has the following features.

– Each task is assigned to one station.
– Production is composed by a mix of variants. The quantity of each variant is known at the beginning, i.e. we have to produce 20 products of variant type 1 and 50 products of variant 2, and so forth.
– Process time of variants can exceed cycle time, but the average on all variants cannot (max peak time constraint).

- The job sequence at the entry of stations is the same (first in first out). When workers perform their tasks on manual ALs, times to perform each task vary from cycle to cycle. Experience indicates that task times can be approximated by a normal distribution when operators work under paced conditions.
- To evaluate a makespan of a given solution, a fast simulation algorithm is used.
- The size of buffers is finite.

The schedule (state of each station at all times) is represented using a 'Gantt chart' (Figure 8.2). The *x* axis corresponds to the time and each horizontal bar corresponds to a station. When a job is processed on a station, a rectangle is placed on the horizontal bar, which begins at the job's start time and ends at its completion time.

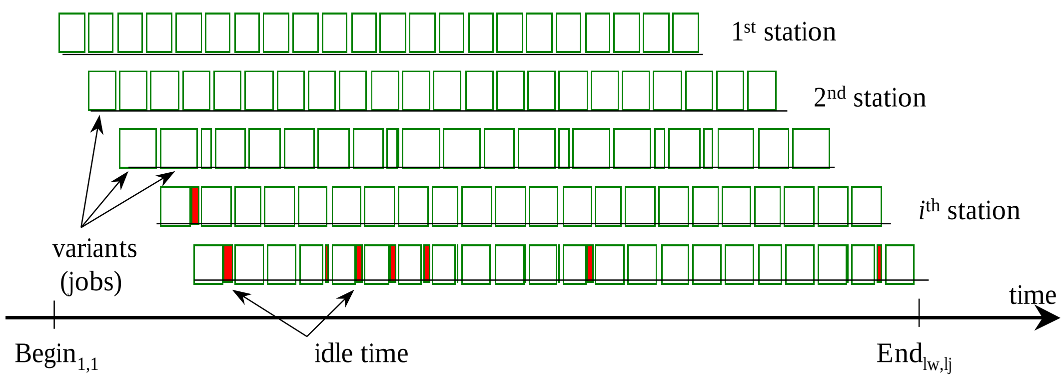

Figure 8.2. Gantt chart for schedule.

Aside from the sheer volume of data and management of information required to make a schedule, there are some inherent difficulties for solving even simplified scheduling problems. Indeed, the size of a scheduling problem can be approximated by the number of stations and the size of the mix. Scheduling problems consist of asking *what* must be done and *when*. More, practically speaking, finding an *optimal* schedule is *less important* than coping with *uncertainties* during planning and *unpredictable* disturbances during the schedule execution. In only few cases, plans are based upon well known processes in which task requirements are well known and can be accurately predicted. However, in many other cases, predictions are less accurate due to the lack of data or predictive models. The expected amount of time during which a schedule is applicable is called the *horizon*. Sometimes a long horizon is chosen for management or economic reasons. Alternatively, if the horizon is too short, poor solutions may result due to the short-sightedness of the scheduler. For this reason, the appropriate choice of the horizon is critical to the performance of a system.

Let us just recall that a scheduling problem with reasonable number of tasks is usually so complex that the most sophisticated scheduling heuristics and fastest machines do not give the *optimal* solutions–they only give good solutions.

5. Balance for operation concept

> *"The interdependency of planning and scheduling can be seen immediately. One cannot plan which tasks to perform in a day unless he has some idea of how much can be accomplished. This implies the need for a schedule. Yet a schedule cannot be constructed unless the tasks to perform are known". (Rickel, 1988)*

The operation stage of mixed assembly line ask for ordering the variants at the entry of the line. The objective is to determine the sequence of variants which maximises the utilisation of the assembly stations. The *balance for operation* (BFO) concept is introduced to tackle the operation phase problems (ordering) at the design stage.

It is important to quantify the balance efficiency of a MPAL. This balance is impacted by the number of stations, the *idle time* on the stations (due to starving), the *waiting time* of the variants (due to blocking), and the *makespan* (total production time for a given mix). Normally, the fewer the number of stations and the less the idle time, the more efficient the line is. We conclude that there is an interaction between the balancing and the scheduling problems.

The balancing of the line is realised using the *desired cycle time* C (production forecast). The *effective cycle time* C_E of a multi-product assembly line is defined as:

$$C_E = average(\frac{makespan}{size\,of\,mix})$$

where the size of the mix is the total number of variants produced in a given period.

This balancing phase is iterative, since it is difficult to find a good maximum peak time for a given assembly line due to problem's constraints. The ordering algorithm aims to minimise the makespan, and consequently the effective cycle time C_E.

Since during the operation phase of the line an important task is the scheduling of the mixed production, it must be taken into account at the design phase. By separating the balancing from this scheduling (most of the time studied after the design stage), a seemingly good balancing according to the desired cycle time may lead to serious starving and blocking problems during the operation phase. The effective cycle time is then much greater than the desired one. Thus, the optimal solution must be found by treating the two problems in a while.

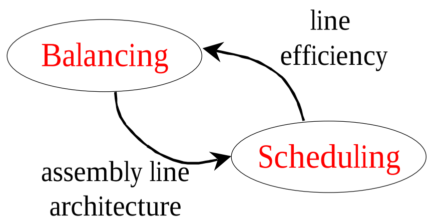

Figure 8.3. General architecture of the BFO concept.

The balance for operation is an iterative and interactive procedure used to balance the multi-product assembly line, taking the scheduling of the variants into account. Its philosophy is illustrated at Figure 8.3.

As pointed out in Chapters 3 and 6, there exist two approaches to assembly line balancing problems. The first tendency is to minimise the number of stations for a given fixed cycle time, while the second tend to balance the workload for a given fixed number of stations. The results of the ordering module permits to find a desired cycle time as close as possible to the effective one. The goal of the two balancing approaches (in the case of multi-product assembly line) is to minimise the imbalance between variants defined as the difference between the process time of a variant and the desired cycle time. The whole procedure of the BFO for the two approaches is described in the next section.

5.1. Non fixed number of stations

The main steps of the concept can be summarised in the following points.

1. *Set a desired cycle time C.*
2. *Set the maximum peak time to (cycle time * var), where var Î [1, 2].*
3. *Balance the line (see Chapter 6).*
4. *If satisfying balancing, then continue else return to 2.*
5. *Test the corresponding AL using the OGA. Evaluate the efficiency of the corresponding line (makespan and idle time). Is the effective cycle time C_E close to the desired one?*
6. *If satisfying solution then assembly line architecture for the family of products found else return to 1 to try another desired cycle time C.*

The algorithm developed to solve the MPAL has been tested on problems randomly generated. The operation durations were generated randomly according to a continuous distribution in the range [5..100]. The number of variant operations is generated uniformly in the range [10 .. (number of operations)/3]. An operation is called variant if its duration is null for at least one variant of the product family. For each operation, the number of precedence constraints is generated randomly in the range [0..8]. The percentage of production requirements is the same for all the variants. The percentage of production requirements is the same for all variants.

The program was executed for a number of operations varying from 50 to 500 and a number of variants varying from 1 (mono product line balancing) to 50. For each instance, the optimal number of stations N_{opt} is known. The stop criterion for the balancing is attained when the number of stations N is equal to N_{opt}. The program was executed more than 25 times (for each instance of the problem) and the optimum solution was found every time. It takes less than a minute for small size instances and less than two minutes for large size instances (tests are done on a PENTIUM II 333 MHz).

The BFO method presented above was tested on randomly generated instances of the problem. The method use a population of 50 individuals (for both balancing and ordering). As the optimal solution of the ordering problem resulting from the balancing is unknown we fixed the stop criterion for the ordering as: less than 5 minutes are needed on average to obtain optimal solutions for the biggest instances of the problem. The results are presented in Table 8.1. A set of instances where the max peak time is less than the cycle time were allowed to explore the search space. The corresponding solutions are characterised by a high number of stations and reduced makespan.

Note that as the mix will change with the consumer's demand, it is important to simulate several mix for a given assembly line[10]. The designer will choose a line yielding similar results for different mixes.

CT	MPT	MS	SN
7	7	149	8
7	8	167	7
7	9	172	6
7	10	172	6
7	11	172	6
7	12	172	6
7	14	192	6
7	16	192	6
7	20	169	6

CT	MPT	MS	SN
8	7	149	8
8	8	180	7
8	9	179	6
8	10	191	5
8	11	191	5
8	12	191	5
8	14	191	5
8	16	191	5
8	20	208	5

CT	MPT	MS	SN
9	7	149	8
9	8	180	7
9	9	203	5
9	10	199	5
9	11	199	5
9	12	199	5
9	14	227	4
9	16	227	4
9	20	227	4

CT	MPT	MS	SN
10	7	149	8
10	8	180	7
10	9	203	5
10	10	218	5
10	11	216	5
10	12	218	4
10	14	218	4
10	20	218	4

CT	MPT	MS	SN
11	7	149	8
11	8	180	7
11	9	203	5
11	10	218	5
11	11	241	4
11	12	243	4
11	14	248	4
11	20	248	4

CT	MPT	MS	SN
12	7	149	8
12	8	180	7
12	9	203	5
12	10	218	5
12	11	241	4
12	12	243	4
12	14	264	4
12	16	264	4

CT	MPT	MS	SN
13	7	149	8
13	8	180	7
13	9	203	5
13	10	218	5
13	11	241	4
13	12	243	4
13	13	273	3
13	14	281	3
13	16	281	3

CT	MPT	MS	SN
14	7	149	8
14	8	180	7
14	9	203	5
14	10	218	5
14	11	241	4
14	12	243	4
14	14	298	3
14	16	291	3
14	20	303	3

CT : Cycle time
MPT : Max peak time
MS : Makespan
SN : station number

Table 8.1. Results of multi-product assembly line (BFO).

[10] The simulated mix have to be as close as possible to the future mix generated by the consumer's demand.

To obtain a feasible MPAL, the variants (models) must have some similarity. If the models are significantly different, it is difficult for the individual station to cope with the differences during assembly. Also if the setup-time is not negligible, the strategy of alternating variants must be discarded. In this case, either a batch-model line or a single assembly line for each product are more appropriate.

A set of tests were done on many assembly lines and on many products having a set of variants. These tests let us to make some conclusions on the design of multi-product assembly line. The main factors influencing the successful design and efficient operation (scheduling) of a mixed-model assembly line are the following.

- The line length: it increases as balancing efficiency decreases.
- The more the number of stations the more it easy to find a good scheduling.
- A few number of stations yield a high ratio of reliability and a high idle time.
- The complexity of sequencing: it increases with number of variants.
- Sensibility to production demand deviation for each variant.
- Process time deviation due to variants of each station (use of max peak time parameter).
- Operator task time variation (use of stochastic duration time).
- Design of multi-product assembly line is an iterative procedure, designer has to solve simultaneously balancing and sequencing problems. Figure 8.4 shows that the optimum depends on the cycle time and max peak time. It corresponds to the solution having the minimum number of stations and the minimum makspan.
- The use of simulation algorithms to validate results.
- The use of buffers to solve starving and blocking problems of the line.
- Take into account setup-time and operator moves to evaluate process time of stations.

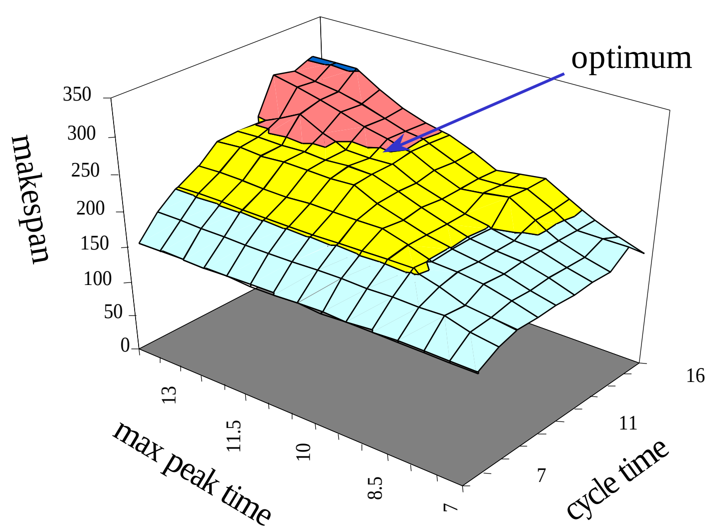

Figure 8.4. Distribution of makespan versus cycle time and max peak time.

5.2. Fixed number of stations

As shown in Chapter 6, the second variation of the ALBP is the EPAL where the number of stations is set as input data of the method. The approach is quite similar to the one used in the case of non fixed number of stations. The main difference is the parameters used to find a well balanced assembly line. The main features of the approach are presented below.

1. *Set preferences of the 'variant process time standard deviation' and of the 'stations misbalance' (see Chapter 6).*
2. *Balance the line (see Chapter 6).*
3. *If satisfying balancing, then continue else return to 1.*
4. *Test the corresponding AL using an (OGA). Evaluate efficiency of the corresponding line.*
5. *If satisfying solution then assembly line architecture for the family of products found else return to 1 and try other preferences.*

The method has as objective to balance the *average* workload among the stations and the process among the variants. For more details on the approach the reader is suggested to refer to Chapter 6. In this case the scheduling module serves as a validation technique of the balancing module. Some results of the proposed approach will be detailed in Chapter 10.

6. Conclusions and further works

If the line is optimally balanced but the launching schedule is incompatible, then the line will operate at less than maximum efficiency and if the schedule is optimised for a poorly balanced line, operating efficiency will also be low. The appropriate order in which to solve the two problems is to first balance the line and then determine the best sequence (test many production rates and mixes) for that balancing solution. The procedure must be iterative, and the designer will choose the best balancing given good results at the sequencing phase.

Since buffers are used in MPAL to resolve problems of starving and blocking, an important design issue is to develop an automatic method to estimate the optimal size of buffers. The importance of human behaviour is often disregarded in research on assembly line performance, but factors such as job motivation and training are critical to product quality and line efficiency. Operations complexity and reliability must be taken into account in design and operation of MPAL.

Since the aim is production and not simply the design of the line, more attention must be done to the operation phase. Indeed, these considerations naturally lead us to abandon the number of stations as indicator of the price of the line. Design (resources, operators, etc.) and operation phase (deadlines, idle time, delivery time, blocking, lateness, etc.) must be taken into account. This, can help the designer avoiding some remaining pitfalls of the concurrent engineering.

7. References

(Agnetis, 1995) Agnetis A., Nicolò F., Arbib C. and Lucertini M., 'Task assignment and subassembly scheduling in flexible assembly lines', IEEE Transactions on Robotics and Automation, Vol. 11(1), pp. 1-20, 1995.

(Arbib, 1999) Arbib C., Ciaschetti G. and Rossi F., 'Distributing material flows in a manufacturing system with large product mix: two models based on column generation', Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, Vol. 480, pp. 235-249, 1999.

(Bard, 1992) Bard J.F., Dar-El E. and Shtub A., 'An analytic framework for sequencing mixed model assembly lines', Int. J. of Prod. Res., Vol. 30(1), pp. 35-48, 1992.

(Bartholdi, 1996) Bartholdi J.J., Eisenstein D.D., 'A production line that balances itself', Operations Research, Vol. 44(1), pp. 21-34, 1996.

(Berger, 1992) Berger I., Bourjoully J-M and Laporte G., 'Branch and bound algorithms for the multi-product assembly line balancing problem', European Journal of Operational Research, Vol. 58, pp. 215-222, 1992.

(Bukchin, 1998) Bukchin J., 'A comparative study of performance measures for throughput of a mixed model assembly line in a JIT environment', Int. J. of Prod. Res., Vol. 36(10), pp. 2269-2685, 1998.

(Chevalier, 1999) Chevalier P., Raucent B. and Semal P., 'Optimizing the design of an assembly line', Proceeding of FAIM'99, Tilburg, Netherlands, pp. 415-424, 1999.

(Choi, 1998) Choi W. and Lee Y., 'Line balancing of mixed-model assembly lines', Proceedings of EDA'98 (CD-ROM), Hawaii, 1998.

(Chow, 1990) Chow W.-M., 'Assembly line design – methodology and applications', Marcel Dekker, New York, 1990.

(Culberson, 1998) Culberson J.C., 'On the futility of blind search: an algorithmic view of "No free lunch"', Evolutionary Computation, Vol. 6(2), 109-127, 1998.

(Davis, 1985) Davis L., 'Applying adaptive algorithms to domains', Proceedings of the International Joint Conference on Artificial Intelligence, pp. 162-164, 1985.

(Delchambre, 1996) Delchambre A., 'CAD method for industrial assembly concurrent design of products equipment and control systems', John Wiley & Sons Inc., Chichester, England, 1996.

(De Lit, 2000) De Lit P., Danloy J., Rekiek B., L'Eglise T., Spineux A. and Delchambre A., 'A new approach to product structures for product families', Proceedings of IDMME'2000, Montreal, Canada, JA-28.3, 10 pages, 2000.

(Driscoll, 1985) Driscoll J. and Abdel-Shaffi A., 'A simulation approach to evaluating assembly line balancing solutions', Int. J. of Prod. Res., Vol. 23(5), pp. 975-985, 1985.

(Falkenauer, 1993) Falkenauer E. and Delchambre A., 'Resource planning in the SCOPES project', Proceedings of the 26th International symposium on Automotive Technology and Automation, Aachen Germany, pp. 295-302, 1993.

(Fernandez, 1995) Fernandez C.J.L. and Groover M.P., 'Mixed-model assembly line balancing and sequencing : a survey engineering', Design and Automation, 1(1), pp. 33-42, 1995.

(Gen, 1997) Gen M. and Cheng R., 'Genetic algorithms & engineering design', John Wiley & Sons Inc., Canada, First Edition, 1997.

(Goldberg, 1989) Goldberg D. E., 'Genetic algorithms in search, optimisation and machine learning', Addison-Wessley Publishing Company. Inc, 1989.

(He, 1997) He D.W. and Kusiak A., 'Design of assembly systems for modular products', IEEE Transactions on Robotic and Automation, Vol. 13(5), pp. 646-655, 1997.

(Helgeson, 1961) Helgeson W.B. and Birnie D.P., 'Assembly line balancing using the ranked positional weight technique', J. Ind. Engng, Vol. 12(6), pp. 394-398, 1961.

(Holland, 1975) Holland J.H., 'Adaptation in natural and artificial systems', University of Michigan Press, Ann Arbor, 1975.

(Hyun, 1998) Hyun C.J., Kim Y. and Kim Y.K., 'A genetic algorithm for multiple objective sequencing problems in mixed assembly lines', Computers and Operations Research, Vol. 25(7-8), pp. 675-690, 1998.

(Johnson, 1954) Johnson S., 'Optimal two- and three-stage production schedules with setup times included', Naval Research Logistics Quarterly, Vol. 1, pp. 61-68, 1954.

(Kurashige, 1999) Kurashige K., Yoshinari Y., Miyazaki S. and Kameyama Y., 'Sequencing method for products in consideration of assembly time', Int. J. Prod. Eco., Vol. 60-61, pp. 565-573, 1999.

(McMullen, 1997) McMullen P.R. and Frazier G.V., 'A heuristic for solving mixed-model line balancing problems with stochastic task durations and parallel stations', Int. J. Prod. Eco., Vol. 51, pp. 177-190, 1997.

(Minzu, 1997) Minzu V. and Henrioud J.-M., 'Assignment stochastic algorithm in multi-product assembly lines', Proceedings of ISATP'97, Marina del Rey, California, pp. 109-114, 1997.

(Miltenburg, 1998) Miltenburg J., 'Level schedules for mixed-model assembly lines in JIT production systems', Management Science, Vol. 35(2), pp. 192-207, 1998.

(Monden, 1983) Monden Y., 'Toyota production system: practical approach to production management', Industrial Engineering and Management Press, Atlanta, 1983.

(Rickel, 1988) Rickel J., 'Issues in the design of scheduling systems', Expert Systems and Intelligent Manufacturing, Michael D. Oliff (editor), pp. 70-89, 1988.

(Romanowicz, 1997) Romanowicz R., 'A tool for an efficient comparison of scheduling methods', Ph.D. Thesis, Swiss Federal Institute of Technology of Lausanne (EPFL), 1997.

(Sarker, 1998) Sarker B.R. and Pan H., 'Designing a mixed-model assembly line to minimise the costs of idle and utility times', Computers Ind. Engng., Vol. 34(3), pp. 609-628, 1998.

(Scholl, 1999) Scholl A., 'Balancing and sequencing of assembly lines'. Heidelberg Physica, 1999.

(Shimokawa, 1997) Shimokawa K., Jürgens U. and Fujimoto T., 'Transforming automobile assembly: experience in automation and work organization', (Eds.), Springer-Verlag, Berlin Heidelberg, Germany, 1997.

(Tamura, 1999) Tamura T., Wang W., Fujita S. and Ohno K., 'Line balancing for mixed-model assembly line with bypass sub-line', Proceedings of ICED'99, Vol. 2, Munich, Germany, pp. 965-968, 1999.

(Thomopoulos, 1967) Thomopoulos N. T., 'Line balancing-sequencing for the mixed-model assembly', Management Science, Vol. 14(2), pp. 59-75, 1967.

(Wall, 1996) Wall M.B., 'A Genetic algorithm for resource-constrained scheduling', Ph.D. Thesis, Massachusetts Institute of Technology, 1996.

(Wang, 1986) Wang F.Z. and Wilson R.C., 'Comparative analyses of fixed and removable item mixed-model assembly lines', IIE Transactions, Vol. 18(3), pp. 313-317, 1986.

(Wester, 1964) Wester L. and Kilbridge L., 'The assembly line model-mix sequencing problem', Proceedings of the 3rd International Conference on Operations Research, Oslo, pp. 247-260, 1964.