Designing Reconfigurable Manufacturing Systems to Minimize Power Peak

Xavier Delorme, Paolo Gianessi

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 55-10 (2022) 1296–1301

Designing Reconfigurable Manufacturing Systems to Minimize Power Peak

Xavier Delorme [*], **Paolo Gianessi** [*]

[*] *Mines Saint-Étienne, Univ. Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F-42023, Saint-Étienne, France (email: [delorme, paolo.gianessi]@emse.fr)*

Abstract: Industry is more and more urged towards energy efficiency by the increasing societal and environmental concern about energy. This is also true for more recent production system types, like Reconfigurable Manufacturing Systems (RMS), which are gaining momentum due to the advent of Industry 4.0 and the growing uncertainty of markets. In this work, we consider energy efficiency in the design stage of an RMS. The tasks of a production process must be assigned to the machines of a Parallel-Serial manufacturing line, and scheduled, so that the related power consumption peak is minimized. An Integer Linear Programming formulation is proposed and tested on a set of benchmark instances. Numerical results are discussed.

Keywords: Reconfigurable Manufacturing System; Assembly Line Balancing; Operations Scheduling; Power Peak; Integer Linear Programming

1. INTRODUCTION

The activity of most production systems is based on scarce and finite energy resources and generate greenhouse gas (GHG) emissions. This explains the growing general societal and environmental concern about energy, and the focus on major objectives as carbon footprint reduction and energy usage optimization, even more so since energy consumption worldwide will expectedly rise by nearly 50% by 2050 (U.S. EIA, 2019). Moreover, in 2014, the 36% of the global total final energy consumption (TFEC) was due to the industrial sector, whose energy consumption has since increased by 1.5% per year (International Energy Agency, 2017). It is therefore not surprising that in the future the most consistent reduction of GHG emissions is expected to come from an improved energy efficiency of industries (Lawrence et al., 2019), and that energy-efficient manufacturing systems (MS) will increasingly consider renewable energy sources (Battaïa et al., 2020).

The growing attention paid in the last decades to energy-aware MSs has led to an increased scientific effort to design decision support methods capable of achieving an optimized energy management. As shown e.g. in Masmoudi et al. (2019), three measures are usually referred to for evaluating energy efficiency in MS: total energy consumption, total energy cost w.r.t. to a given pricing policy, and power peak limitation. However, it seems that most of the scientific output that deals with energy efficiency in production system focus on planification and scheduling problems: few seem to be the works that consider it in the design or reconfiguration phases of the system. Among the aforementioned measures, power limitation increases energy efficiency by smoothening energy consumption and reducing the effects of its volatility. This is of great importance when considering a more and more diversified range of used energy sources. However, few works also seem to exist that minimize the production-related power peak.

The last decade also represents a major transition phase for the industrial sector. Production systems are more subject to market uncertainty, mostly as a consequence of shorter product life cycle and mass customization, and need to be more agile and capable to quickly adapt their throughput to the volatility of the demand, but also to technological evolutions and the constantly changing regulatory framework, especially for what concerns energy. However, the advent of Industry 4.0 allows to rise to these economical, technological, organizational, societal and environmental challenges. One of the possibilities offered by Industry 4.0 is improved energy efficiency, thanks to the augmented possibilities to track or control energy consumption (Mohamed et al., 2019).

Moreover, Reconfigurable MS (RMS), introduced in Koren et al. (1999), can help achieve such reactivity to demand volatility by reconfiguring the production system, and are gaining momentum in the Industry 4.0 context. A typical RMS is composed of a set of workstations organized in a serial line, each with multiple parallel identical machines. Resources on workstation can be computer numerical control machines, reconfigurable machine tools, or other types of resources, for instance workers with cobot. Reconfigurations consist in either adding or removing resources, or changing the assignment of production tasks to workstations. Parts are moved from a workstation to the next by a conveyor and a gantry. To this end, each configuration of an RMS is to all intents and purposes a paced *Parallel-Serial manufacturing line with Crossover* (see e.g. Freiheit et al. (2004)), as shown in Figure 1.

Motivated by the growing interest in RMS, in this work we study the problem of balancing a Parallel-Serial line with Crossover, i.e. assigning its production tasks to its workstations, so as to attain a given production pace. Such decision process occurs in the design phase of the line and can only affect power peak, while energy consumption or energy economic cost are more related to, respectively,

10.1016/j.ifacol.2022.09.569

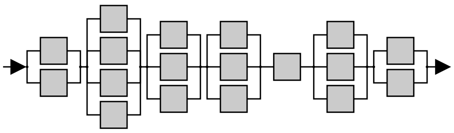

Fig. 1. A Parallel-Serial line with 7 workstations, each with 1 to 4 parallel resources, in which crossover is possible between any two resources of two consecutive stages.

equipment selection and production planning. Hence, the problem at study aims at energy efficiency by minimizing the peak of production power consumption.

In the following, after a short review of the related literature in Section 2, Section 3 introduces the problem and an Integer Linear Programming model for it, while Section 4 delves into some of its features. In Section 5, the computational experience conducted to assess the performance of the model is described, along with the results analysis. Finally, Section 6 gives some conclusions and perspectives.

2. LITERATURE REVIEW

The most investigated optimization problems concerning the design of production systems are Assembly Line Balancing (ALB) problems. In them, the tasks of an assembly process must be assigned to a set of workstations so as to comply with the precedence constraints among them. In spite of the name due to historical reasons, ALB problems can as well describe other types of industrial environments, such as machining or disassembly systems (see e.g. the literature review of Battaïa and Dolgui (2013)).

The most studied case is the Simple ALB Problem (SALBP) which studies a paced, synchronous, mono-product line, whose task execution times are deterministic and workstation-independent. Either the line takt time c is known, and the number of workstations m has to be minimized (type I SALBP or SALBP-1), or the inverse (SALBP-2), or the total idle time must be minimized (SALBP-E), or the feasibility of a given pair (c, m) must be assessed (SALBP-F). SALBP is NP-hard (Scholl, 1999) and best known solution methods are relatively recent (Cerqueus and Delorme (2019); Pape (2015); Sewell and Jacobson (2012)).

Among the most known ALB variants, in the Robotic ALB problem (RALBP) (Borba et al., 2018) special equipment must be assigned to workstations to perform tasks. RALBPs are relevant in that they are, to the best of our knowledge, one of the few production system design problems in the literature for which energy-efficiency is considered, as e.g. in Li et al. (2016), where a two-sided RALBP is proposed, and Pareto-optimal solution w.r.t. energy consumption and cycle time are sought for. Some other work exist, however, that consider energy efficiency in balancing problems, for instance Liang et al. (2021) seeks for energy consumption minimization in the balancing of a disassembly line.

As far as the authors are aware of, the only problem dealing with minimization of the power peak in the design phase of an MS is the Simple Assembly Line Balancing Problem with Power Peak Minimization (SALB3PM), introduced by Gianessi et al. (2019). In it, both the takt time c and the maximum number of workstations m are given, and the production tasks feature constant power consumption values, other than precedence relations and processing times as in the SALBP. The aim is to assign tasks to workstations and schedule them so as to minimize the peak of the overall power consumption profile, due to the overlapping processing of tasks along each cycle of the line. The SALBP3PM is harder than the SALBP, since scheduling decision about the starting date of tasks add to the assignment decisions, and intermediate idle time between two consecutive tasks on a workstation can be considered. A particular case is studied in Lamy et al. (2020), in which tasks are triggered at the earliest available starting date to better fit the case of manual or semi-automated production systems, and scheduling decisions actually become sequencing decisions.

To conclude this short review on line balancing problems, we cite some more variants which are nearer to the problem studied here. The balancing of assembly lines with parallel resources has also received attention (Buxey, 1974; Pinto et al., 1981). More recently, Essafi et al. (2010) and Borisovsky et al. (2014) have studied the balancing of RMSs with sequence-dependent task setup times, which require to take into account task sequencing decisions.

RMS have been introduced in Koren et al. (1999) to combine the productivity of dedicated lines and the flexibility of flexible MSs. One of the most prominent features of RMS is scalability (Koren et al., 2017), which allows to adapt productivity to face large uncertainty of demand (Koren, 2020) or fluctuations of energy prices, and can improve the optimization of MS design and management and the development of new paradigms for sustainability (Putnik et al., 2013). Research works on planning and sustainability of RMS are still scarce though (Yelles-Chaouche et al., 2021), and the consideration of both scalability and energy expenditure is even scarcer. In Wang and Koren (2012), a scalability planning methodology for RMS is explored that consists in changing the capacity of an existing system by successive reconfigurations, so as to minimize the number of resource changes required to reconfigure. In Moghaddam et al. (2020), design and reconfiguration costs are minimized in multi-product and scalable RMS, while fulfilling a given demand over multiple production periods.

As for energy consideration in RMSs, the survey of Battaïa et al. (2020) shows that RMS have great potential to improve energy efficiency in production, but the amount of research work on the topic is still scarce. Zhang et al. (2015) introduces the concept of *energy efficient RMS* and investigates a discrete event simulation model to assess the energy performances of such a system; Choi and Xirouchakis (2015) studies the problem of determining the production planning of an RMS and assess its performance based on energy consumption, throughput, and inventory holding. Recently, Cerqueus et al. (2020) and Gianessi et al. (2021) have tackled the Bilevel Optimization problem of designing a set of configuration set for an RMS that allows fulfilling a demand over a known time horizon with minimum energy-related costs w.r.t. a given Time-Of-Use pricing scheme.

3. PROBLEM DESCRIPTION

In this section, after defining suitable notations, we formally define the Parallel-Serial-with-Crossover Assembly Line Balancing Problem with Power Peak Minimization (PSCALB3PM), which generalizes the SALB3PM, and provide a time-indexed Integer Linear Programming model, inspired from that of Gianessi et al. (2019) for the latter. Let $\mathcal{O} = \{0...n-1\}$ denote the set of the n production tasks of a production process, and $\mathcal{M} = \{0...m-1\}$ the set of the m available workstations. Each task $j \in \mathcal{O}$ features integer, deterministic, workstation-independent processing time t_j and power consumption values w_j, and is involved in some precedence constraints, i.e. there exists a subset of tasks, i, s.t. the processing of each of them must be over before j can begin, which we denote by the notation $i \prec j$. Each workstation $k \in \mathcal{M}$ can be assigned a number r_k of identical resources. For economic reasons, both r_k and the maximum total number of resources for the line are bounded, i.e. $r_k \in \rho = \{1...r_{\max}\}$ and $\sum_{k\in\mathcal{M}} r_k \leq R_{\max}$, with $R_{\max} \leq |\mathcal{M}| \cdot r_{\max}$. Finally, let us note c the known value of the takt time, which represents the line pace.

The objective is to decide the number of resources r_k of each workstation $k \in \mathcal{M}$, assign tasks to workstations and decide the starting date of tasks, so that precedence constraints are complied with, the workload of workstations is s.t. the target takt time of the line is not exceed, and the peak of the overall power consumption profile, due to the concurrent processing of tasks on the resources of the different workstations, is minimized. Line structure decisions add to the assignment and scheduling decision of the SALB3PM: indeed, the PSCALB3PM generalizes the SALB3PM, which is the particular case with $r_{\max} = 1$. Given workstations k' s.t. $r_{k'} > 1$, its resources are assigned the same tasks, which they process according to the same schedule, but shifted by multiples of c. Therefore, they can be assigned tasks for a total workload of up to $r_{k'} \cdot c$, and still k can output a produced item each c time units, i.e. without compromising the target line pace.

In order to represent the scheduling decisions in the PSCALB3PM, we resort to time-indexed modeling, a very common choice in the study of scheduling problems (Bowman, 1959). To do so, and since the time horizon of workstation k *virtually* expands to $r_k \cdot c$, it is suitable to define set $\mathcal{T} = \{0...r_{\max} \cdot c - 1\}$, containing all the time slots of the augmented timespan $r_{\max} \cdot c$, as the index set of time-indexed variables. To represent the candidate dates of $\mathcal{T}$ to trigger task j, the set $\mathcal{T}^j = \{0...r_{\max} \cdot c - t_j\}$ is further defined. Finally, and more generally, a task on a workstation with r_k resources can be virtually triggered on any date of set $\mathcal{T}^j_r = \{0...r \cdot c - t_j\}$, where $\mathcal{T}^j_{r_{\max}} = \mathcal{T}^j$. The decisions in the PSCALB3PM can be modeled by the following binary decisions variables:

- *assign* variables $X_{j,k}$, $j \in \mathcal{O}$, $k \in \mathcal{M}$, $X_{j,k} = 1 \Leftrightarrow$ task j assigned to workstation k,
- *trigger* variables $S_{j,t}$, $j \in \mathcal{O}$, $t \in \mathcal{T}^j$, $S_{j,t} = 1 \Leftrightarrow$ task j starts at time slot t,
- *resource* variables $R_{k,r}$, $k \in \mathcal{M}$, $r \in \rho$, $R_{k,r} = 1 \Leftrightarrow$ workstation k uses r-th resource,

to which we add an integer non-negative *power-peak* variable $W_{_M}$, an upper bound on the power consumption peak all along the takt $\mathcal{T}$. For each task j, only one $S_{j,t}$ variable can take value 1, corresponding to the trigger date of j. Based on this, we also define the binary decision expression $F(j,t)$, which evaluates to 1 if task j is *running* at date t, having been triggered at a date $\tau \in \{t-t_j+1...t\}$:

$$F(j,t) = \sum_{\tau=t-t_j+1}^{t} S_{j,\tau}$$

The proposed model for PSCALB3PM is then as follows:

$$\min W_{_M} \tag{1}$$
$$\text{s.t.} \sum_{k\in\mathcal{M}} X_{j,k} = 1 \qquad \forall j \in \mathcal{O} \tag{2}$$
$$\sum_{j\in\mathcal{O}} t_j \cdot X_{j,k} \leq c \cdot \sum_{r\in\rho} R_{k,r} \qquad \forall k \in \mathcal{M} \tag{3}$$
$$X_{j,k} \leq \sum_{h\in\mathcal{M}: h\leq k} X_{i,h} \qquad \forall i,j \in \mathcal{O}: i \prec j,\ k \in \mathcal{M} \tag{4}$$
$$\sum_{t\in\mathcal{T}^j} S_{j,t} = 1 \qquad \forall j \in \mathcal{O} \tag{5}$$
$$X_{j,k} - R_{k,r} \leq 1 - \sum_{t\in\mathcal{T}^j\setminus\mathcal{T}^j_{r-1}} S_{j,t} \qquad \forall j \in \mathcal{O},\ k \in \mathcal{M}, r \in \rho \tag{6}$$
$$S_{j,t} \leq \sum_{\tau=0}^{t-t_i} S_{i,\tau} + 2 - X_{i,k} - X_{j,k} \qquad \forall i,j \in \mathcal{O}: i \prec j,\ k \in \mathcal{M}, t \in \mathcal{T}^j \tag{7}$$
$$X_{i,k} + X_{j,k} + F(i,t) + F(j,t) \leq 3 \qquad \forall i,j \in \mathcal{O}: i < j,\ k \in \mathcal{M}, t \in \mathcal{T} \tag{8}$$
$$\sum_{j\in\mathcal{O}, r\in\rho} w_j \cdot F(j,(r-1)c+t) \leq W_{_M} \qquad \forall t \in \{0...c-1\} \tag{9}$$
$$R_{k,r+1} \leq R_{k,r} \qquad \forall k \in \mathcal{M},\ r \in \rho: r < r_{\max} \tag{10}$$
$$\sum_{k\in\mathcal{M}, r\in\rho} R_{k,r} \leq R_{\max} \tag{11}$$
$$X_{j,k}, S_{j,t}, R_{k,r} \in \{0,1\}, W_{_M} \in \mathbb{Z}_+$$

Constraints (2) assert the assignment of each task to exactly one workstation, while (3) concern the workload on station k: each task j assigned to it ($X_{j,k} = 1$) occupies t_j time slots of its workload, the workload limit being c times the number of resources assigned to k, $\sum_{r\in\rho} R_{k,r}$. Precedence relations (4) state that given two task i, j s.t. $i \prec j$, i must be assigned either to the same workstation of j, or to an upstream one; in the former case, inequalities (7) (which are redundant otherwise) further state that i must be triggered at least t_i slots before j. Constraints (5) assert that task j must be triggered at exactly one of its candidate dates $\mathcal{T}^j$: more precisely, if j is assigned to workstation k ($X_{j,k} = 1$) with r_k resources, the trigger must take place at one of the dates $\{0...r_k \cdot c - t_j\}$, as (10) ensures that $R_{k,1} = ... = R_{k,r_k} = 1$, and consequently (6) forbids starting j at a date $t \geq r_k \cdot c - t_j + 1$. Relations (8) prevents two task i and j from being executed on the same workstation at the same time, as either their processing overlap, i.e. $(\exists t \in \mathcal{T})\ F(i,t) = F(j,t) = 1$ but they are assigned to different workstations, or they are on the same one and at most one among them can be running at date t. Constraints (9) consider, for each date t of the *actual* timespan $\{0...c-1\}$ of the line, all the tasks currently running at t simultaneously on some resource of the (k,r) of the line, i.e. which are running at some date $(r-1)c+t$ of the schedule of the workstation they are assigned to. The overall power consumption of all the tasks running at each date t is bounded by (9) by $W_{_M}$, which we minimize in (1). Finally, relation (11) bounds the economic cost by bounding at $R_{\max}$ the overall number of resources.

4. A MORE IN-DEPTH VIEW OF THE PROBLEM

In Section 3, we formally stated that the PSCALB3PM generalizes the SALB3PM. More precisely, the former reduces to the latter when $r_{\max} = 1$, while greater values of $r_{\max}$ allow non-serial configurations. To better understand the problem studied here, it is useful to have a closer

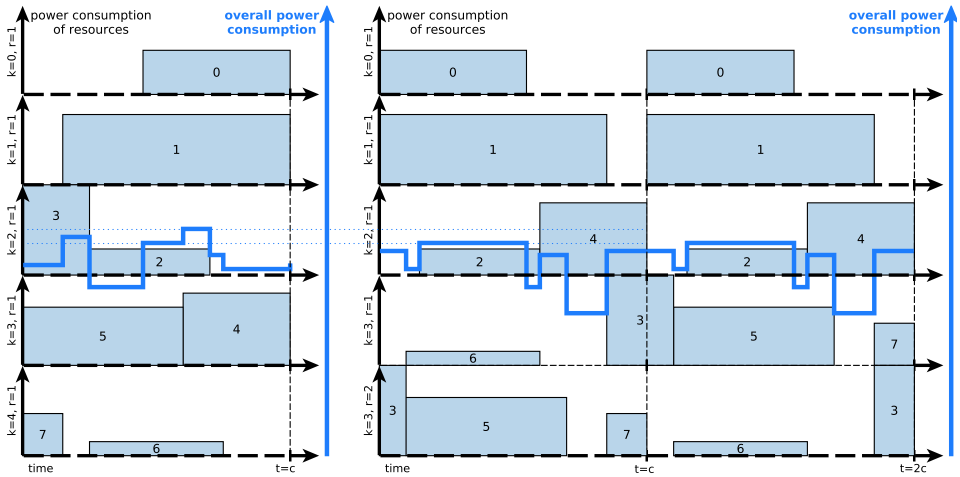

Fig. 2. Effect on power peak of allowing non-serial configurations, bowman-1 instance.

look to what this implies. To this end, we consider two SALB3PM instances taken from Gianessi et al. (2019), namely bowman-1 and jaeschke-2. Both are derived from original SALBP-1 instances and the number of workstations, m, is the minimum required w.r.t. the given takt time c, therefore $R_{\max} \geq m$ must hold.

First, Figure 2 shows on instance bowman-1 the potential benefits of allowing non-serial configurations. Both its subfigures represent optimal solutions when $R_{\max} = m = 5$: the leftmost one has $r_{\max} = 1$, i.e. multiple parallel resources in a workstation are forbidden – the depicted solution is actually the SALB3PM optimum; the rightmost one has $r_{\max} = 2$, i.e. non-serial configurations are allowed and up to 2 resources per workstations can be used. The two subfigures show in detail, via Gantt diagrams, how tasks are assigned to resources and scheduled in the optimal solutions: tasks are represented by boxes whose width, horizontal position and height represent the processing time, trigger date and power consumption. Thin and thick dotted lines separate the schedule of resources of the same workstation and that of resources of different ones, respectively; a thick continuous line shows the overall power consumption profile. When $r_{\max} = 1$, the peak occurs at the overlap of tasks 0, 1, 2, 4 and 6, thus $W_M = w_0 + w_1 + w_2 + w_4 + w_6 = 22 + 35 + 13 + 36 + 7 = 113$. When $r_{\max} = 2$, workstations $k = 0...2$ have $r_k = 1$, while two resources are used for workstation $k = 3$, i.e. $r_3 = 2$. It is worth noting that for $k = 0...2$, the cycle has a period c, while for $k = 3$ resources have a period $2c$ and process the same tasks but with schedules shifted by c in the same workstation. In this solution with parallel resources allowed, the previously described overlap can be avoided, and the peak occurs when tasks 0, 1, 2, 5 and 6 overlap: this decreases W_M by $w_4 - w_5 = 7$ to 106 power units.

After showing the effect of allowing non-serial configurations, it is interesting to show, in such a setting, the impact of adding additional resources to an existing parallel-serial configuration, which Figure 3 shows on instance jaeschke-2. The curve represents the minimum power peak for $r_{\max} = 2$ and $R_{\max}$ ranging from m, i.e. 3, to $r_{\max} \cdot m = 6$. When augmenting $R_{\max}$ from 3 to 4, W_M decreases from 73 to 70, a value that does not change for further values of the overall number of resources. Figure 4 shows more in detail how tasks rearrange, in the optimal solution, when

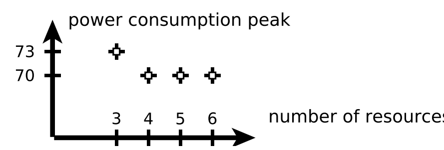

Fig. 3. Effect on power peak of augmenting $R_{\max}$ by one unit, jaeschke-2 instance, $r_{\max} = 2$.

the line resources increase from 3 (leftmost subfigure) to 4 (rightmost subfigure). The overall power consumption profile periodically repeats twice, since in both cases at least one workstation ($k = 0$ for $R_{\max} = 3$, $k = 1$ for $R_{\max} = 4$) uses all the $r_{\max} = 2$ resources available, which process the same tasks and have the same schedule shifted by c, therefore the Gantt diagrams cover a time extent of $2c$. With $R_{\max} = 3$, the power peak is due to the overlap of tasks 1, 4 and 8, whose total power consumption is $w_1 + w_4 + w_8 = 27 + 26 + 20 = 73$. Adding a resource allows to avoid this overlap, and the peak occurs when tasks 3 and 4 overlap, with a consumption of $w_3 + w_4 = 44 + 26 = 70$.

5. COMPUTATIONAL STUDY

We describe here the computational experiments conducted to evaluate the performance of the proposed ILP model for the PSCALB3PM. The model is implemented in CPLEX 12.6 and solved by Branch&Bound. Tests are run on a Intel Xeon E5-2660 v3 2.6 Ghz machine with 62.64Gb RAM. All instances are given a time limit of one hour.

We considered 10 of the 19 SALB3PM instances of Gianessi et al. (2019), inspired from as many benchmark SALBP-1 datasets[1], in which task power consumption values w_j are uniformly generated between 5 and 50. The number of tasks varies between 7 and 21, c is given and m is the computed optimal number of workstations for the SALBP-1 original instance. Hence, for each instance $R_{\max} \geq m$ must hold, i.e. the total number of resources cannot be less than m, and if $r_{\max} = 1$, the PSCALB3PM optimal solution will use all available workstations.

Table 1 presents the results on the chosen instances and for values of $r_{\max}$ from 1 to 3; $R_{\max}$ is given the highest

[1] assembly-line-balancing.de/salbp/benchmark-data-sets-1993/

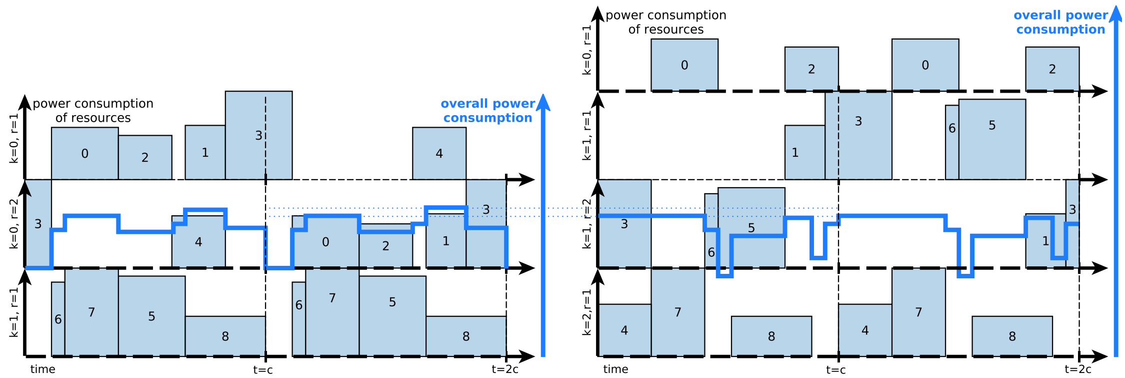

Fig. 4. Assignment and schedule of tasks to the resources when $R_{\max}$ increases from 3 to 4, jaschke-2 instance, $r_{\max} = 2$.

Table 1. Computational results for values of $r_{\max}$ from 1 to 3.

instance	n	c	m	$r_{\max}=1$				$r_{\max}=2$				$r_{\max}=3$			
				$\overline{\%}$	W_M	ur	$T/(\%)$	$\overline{\%}$	W_M	ur	$T/(\%)$	$\overline{\%}$	W_M	ur	$T/(\%)$
mertens-1	7	6	6	0.00%	171	6	0.03	8.78%	161	10	0.40	8.78%	161	10	1.00
mertens-2	7	18	2	9.62%	57	2	0.21	14.00%	57	4	1.63	14.00%	57	6	2.00
bowman-1	8	20	5	9.13%	113	5	0.15	8.16%	106	7	104.62	8.16%	106	11	2204.66
jaeschke-1	9	6	8	0.00%	249	8	0.03	2.59%	198	13	1.78	2.59%	198	14	23.29
jaeschke-2	9	18	3	12.31%	73	3	0.96	7.69%	70	6	382.62	7.69%	70	9	1649.44
jackson-1	11	7	8	0.00%	188	8	0.07	0.00%	162	12	1.17	0.00%	162	21	10.01
jackson-2	11	21	3	7.41%	58	3	3.39	7.41%	58	6	512.05	7.41%	58	9	(5.45%)
mansoor-1	11	48	4	3.10%	133	4	0.55	8.40%	129	7	(8.40%)	8.40%	129	10	(8.40%)
mansoor-2	11	94	2	19.60%	78	2	2.91	22.11%	77	4	260.31	22.07%	77	6	(16.67%)
mitchell-1	21	14	8	6.03%	211	8	9.94	1.01%	201	15	(1.01%)	0.50%	200	20	(0.50%)

possible value, i.e. $m \cdot r_{\max}$, corresponding to $r_{\max}$ resources for each of the available workstations. For each instance and value of $r_{\max}$, the table reports: $\overline{\%}$, the gap at the root node (whose computation is enhanced by the lower bound improvement routines of CPLEX); W_M, the value of the best found solution within time limit; $T/(\%)$, the total computation time (in seconds) to compute it, or the optimality gap still to close at time limit; and finally ur, the number of resources used in the best found solution. The case $r_{\max} = 1$ is equivalent to the SALB3PM. The same results of Gianessi et al. (2019) are found and the performances in terms of time to find the optimal solution are equivalent. For $r_{\max} = 2$, for which 2 resources per workstation can be used, a reduction of the minimum power peak value occurs in 8 instances out of 10, ranging from -1.3% to -20.5%, and hence the benefit of allowing non-serial configurations. Computation times are higher, and in 2 instances out of 10 the optimality gap is not closed within time limit. This can be explained by the augmented size of the model: the number of trigger variables doubles with $r_{\max}$, and –to a minor extent– some of the constraints increase in number. Also, we can notice that in 6 cases out of 10 the number of resources used by the best found solution is strictly less than the maximum allowed $R_{\max}$; this means that for each instance the problem is less constrained and the solving is more prone to symmetry issues. By looking at results with $r_{\max} = 3$, we notice that no substantial further improvement is achieved in terms of power peak, even in cases for which $2 \cdot m$ resources where used for $r_{\max} = 2$ and hence an improvement could be expected. This is probably due to the size and the features of the processing tasks of the chosen instances: an increase of one unit in the maximum number of parallel resources per workstation do not produce any benefit, as c and the precedence constraints do not allow tasks to be arranged any better. This probably means that there exist, for each instance, threshold values for $R_{\max}$ that allow an actual reduction of power peak, and between two values the overall power peak do not change, which we could foresee from Figure 3.

6. CONCLUSION

In this work, we tackled the problem of balancing a Reconfigurable Manufacturing System so as to minimize the power consumption peak, and introduced the Parallel-Serial-with-Crossover Assembly Line Balancing Problem with Power Peak Minimization (PSCALB3PM). We proposed an Integer Linear Programming model and tested it on a set of benchmark instances. The results show that the approach is effective and that allowing non-serial configurations can have actual benefits on the power peak. The PSCALB3PM opens a promising research path, and it would also be interesting to study the case of production tasks with non-constant power consumption profiles, or the bi-objective problem of finding the best trade-off between takt time and power consumption peak. Also of interest would be to enhance the proposed method with more advanced techniques (e.g. preprocessing, problem-specific valid inequalities) so that instances inspired from real-world case studies could be dealt with.

ACKNOWLEDGEMENTS

This work is supported by the project ANR-21-CE10-0019 “RECONFIDURABLE”.

REFERENCES

Battaïa, O. and Dolgui, A. (2013). A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics*, 142, 259–277.

Battaïa, O., Benyoucef, L., Delorme, X., Dolgui, A., and Thevenin, S. (2020). Sustainable and energy efficient reconfigurable manufacturing systems. In *Reconfigurable Manufacturing Systems: From Design to Implementation*, 179–191. Springer.

Borba, L., Ritt, M., and Miralles, C. (2018). Exact and heuristic methods for solving the Robotic Assembly Line Balancing Problem. 270, 146–156.

Borisovsky, P.A., Delorme, X., and Dolgui, A. (2014). Balancing reconfigurable machining lines via a set partitioning model. *International Journal of Production Research*, 52(13), 4026–4036.

Bowman, E.H. (1959). The schedule-sequencing problem. *Operations Research*, 7(5), 621–624.

Buxey, G. (1974). Assembly line balancing with multiple stations. *Management science*, 20(6), 1010–1021.

Cerqueus, A. and Delorme, X. (2019). A branch-and-bound method for the bi-objective simple line assembly balancing problem. *International Journal of Production Research*, 57(18), 5640–5659.

Cerqueus, A., Gianessi, P., Lamy, D., and Delorme, X. (2020). Balancing and configuration planning of RMS to minimize energy cost. In *Advances in Production Management Systems: Towards Smart and Digital Manufacturing*, volume 592 of *IFIP Advances in Information and Communication Technology*, 518–526. Springer.

Choi, Y.C. and Xirouchakis, P. (2015). A holistic production planning approach in a reconfigurable manufacturing system with energy consumption and environmental effects. *International Journal of Computer Integrated Manufacturing*, 28(4), 379–394.

Essafi, M., Delorme, X., Dolgui, A., and Guschinskaya, O. (2010). A MIP approach for balancing transfer line with complex industrial constraints. *Computers & Industrial Engineering*, 58(3), 393–400.

Freiheit, T., Shpitalni, M., and Hu, S.J. (2004). Productivity of paced parallel-serial manufacturing lines with and without crossover. *Journal of Manufacturing Science and Engineering*, 126(2), 361–367.

Gianessi, P., Cerqueus, A., Lamy, D., and Delorme, X. (2021). Using reconfigurable manufacturing systems to minimize energy cost: a two-phase algorithm. *IFAC-PapersOnLine*, 54(1), 379–384.

Gianessi, P., Delorme, X., and Masmoudi, O. (2019). Simple Assembly Line Balancing Problem with Power Peak Minimization. In F. Ameri, K.E. Stecke, G. von Cieminski, and D. Kiritsis (eds.), *Advances in Production Management Systems*, volume 566, 239–247. Springer Int. Publishing, Cham.

International Energy Agency (2017). Tracking Clean Energy Progress 2017: Industry. https://www.iea.org/etp/tracking2017/industry/.

Koren, Y. (2020). The emergence of reconfigurable manufacturing systems (RMSs). In *Reconfigurable Manufacturing Systems: From Design to Implementation*, 1–9. Springer.

Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G., and H, V.B. (1999). Reconfigurable manufacturing systems. *CIRP Annals*, 48, 2.

Koren, Y., Wang, W., and Gu, X. (2017). Value creation through design for scalability of reconfigurable manufacturing systems. *International Journal of Production Research*, 55(5), 1227–1242.

Lamy, D., Delorme, X., and Gianessi, P. (2020). Line balancing and sequencing for peak power minimization. *IFAC-PapersOnLine*, 53(2), 10411–10416.

Lawrence, A., Thollander, P., Andrei, M., and Karlsson, M. (2019). Specific Energy Consumption/Use (SEC) in Energy Management for Improving Energy Efficiency in Industry: Meaning, Usage and Differences. *Energies*, 12(2), 247.

Li, Z., Tang, Q., and Zhang, L. (2016). Minimizing energy consumption and cycle time in two-sided robotic assembly line systems using restarted simulated annealing algorithm. 135, 508–522.

Liang, J., Guo, S., Du, B., Li, Y., Guo, J., Yang, Z., and Pang, S. (2021). Minimizing energy consumption in multi-objective two-sided disassembly line balancing problem with complex execution constraints using dual-individual simulated annealing algorithm. *Journal of Cleaner Production*, 284, 125418.

Masmoudi, O., Delorme, X., and Gianessi, P. (2019). Job-shop scheduling problem with energy consideration. *International Journal of Production Economics*, 216, 12–22.

Moghaddam, S.K., Houshmand, M., Saitou, K., and Fatahi Valilai, O. (2020). Configuration design of scalable reconfigurable manufacturing systems for part family. *International Journal of Production Research*, 58(10), 2974–2996.

Mohamed, N., Al-Jaroodi, J., and Lazarova-Molnar, S. (2019). Leveraging the capabilities of industry 4.0 for improving energy efficiency in smart factories. *IEEE Access*, 7, 18008–18020.

Pape, T. (2015). Heuristics and lower bounds for the simple assembly line balancing problem type 1: Overview, computational tests and improvements. *European Journal of Operational Research*, 240, 32–42.

Pinto, P.A., Dannenbring, D.G., and Khumawala, B.M. (1981). Branch and bound and heuristic procedures for assembly line balancing with paralleling of stations. *The International Journal of Production Research*, 19(5), 565–576.

Putnik, G., Sluga, A., ElMaraghy, H., Teti, R., Koren, Y., Tolio, T., and Hon, B. (2013). Scalability in manufacturing systems design and operation: State-of-the-art and future developments roadmap. *CIRP Annals*, 62(2), 751–774.

Scholl, A. (1999). Balancing and sequencing of assembly lines. Technical report, Darmstadt Technical University.

Sewell, E.C. and Jacobson, S.H. (2012). A branch, bound, and remember algorithm for the simple assembly line balancing problem. *INFORMS Journal on Computing*, 24(3), 433–442.

U.S. EIA (2019). International energy outlook 2019. https://tinyurl.com/yc43apkm.

Wang, W. and Koren, Y. (2012). Scalability planning for reconfigurable manufacturing systems. *Journal of Manufacturing Systems*, 31(2), 83–91.

Yelles-Chaouche, A.R., Gurevsky, E., Brahimi, N., and Dolgui, A. (2021). Reconfigurable manufacturing systems from an optimisation perspective: a focused review of literature. *International Journal of Production Research*, 59(21), 6400–6418.

Zhang, J., Khalgui, M., Boussahel, W.M., Frey, G., Hon, C., Wu, N., and Li, Z. (2015). Modeling and verification of reconfigurable and energy-efficient manufacturing systems. *Discrete Dynamics in Nature and Society*.