Journal of Industrial Information Integration 27 (2022) 100281

Contents lists available at ScienceDirect

Journal of Industrial Information Integration

journal homepage: www.sciencedirect.com/journal/journal-of-industrial-information-integration

Full length article

Exploring the benefits of scheduling with advanced and real-time information integration in Industry 4.0: A computational study

Victor Fernandez-Viagas [a,*], Jose M. Framinan [a,b]

[a] Industrial Management, School of Engineering, University of Seville, Camino de los Descubrimientos s/n, Seville 41092, Spain
[b] Laboratory of Engineering for Environmental Sustainability, University of Seville

ARTICLE INFO

Keywords:
Industry 4.0
Smart factory
Information integration
Scheduling
Flow shop

ABSTRACT

The technological advances recently brought to the manufacturing arena (collectively known as Industry 4.0) offer huge possibilities to improve decision-making processes in the shop floor by enabling the integration of information in real-time. Among these processes, scheduling is often cited as one of the main beneficiaries, given its data-intensive and dynamic nature. However, in view of the extremely high implementation costs of Industry 4.0, these potential benefits should be properly assessed, also taking into account that there are different approaches and solution procedures that can be employed in the scheduling decision-making process, as well as several information sources (i.e. not only shop floor status data, but also data from upstream/downstream processes).

In this paper, we model various decision-making scenarios in a shop floor with different degrees of uncertainty and diverse efficiency measures, and carry out a computational experience to assess how real-time and advance information can be advantageously integrated in the Industry 4.0 context. The extensive computational experiments (equivalent to 6.3 years of CPU time) show that the benefits of using real-time, integrated shop floor data and advance information heavily depend on the proper choice of both the scheduling approach and the solution procedures, and that there are scenarios where this usage is even counterproductive. The results of the paper provide some starting points for future research regarding the design of approaches and solution procedures that allow fully exploiting the technological advances of Industry 4.0 for decision-making in scheduling.

1. Introduction

The incorporation of cutting-edge technologies into production systems that is taking place in the last years has given rise to the so-called 'fourth industrial revolution' or Industry 4.0 (see e.g. 49). A main characteristic of Industry 4.0 is the integration of heterogeneous data and knowledge [33], a fact that –at least theoretically– streamlines decision-making [50]. However, although the information integration is an emerging topic that has attracted the attention of many researchers in the last years (4) and has been applied in many industrial areas (see e.g. [30,31,35]), its use in the scheduling sector is very scarce. In this regard, scheduling is often cited as one of the decision-making processes that could benefit most from Industry 4.0 [39] since, in real-world scheduling, the data required for this decision problem are often subject to uncertainty and may change over time [29]. On the one hand, the so-called *predictive* approaches (i.e. executing the initial schedule without changes) may not cope efficiently with the dynamic behaviour that constitutes the trademark of nowadays manufacturing [37], and therefore the intrinsic criticality of scheduling processes becomes even more salient in the context of Industry 4.0 [40]. On the other hand, the integration of updated data for revising the existing schedule –i.e. rescheduling– is not new, since the so-called *predictive-reactive* approaches are well-known to deal with unforeseen events (machine breakdowns, new jobs arrival, etc.), but it is not until recently when the ability of incorporating available real-time information has started to be explored. Indeed, the management of real-time information is considered one of the main research lines in Industry 4.0-based scheduling [17, 36,39].

Despite the potential for the improvement of operations nurtured by Industry 4.0 (see e.g. 9 for a summary of benefits), the capital involved in its implementation is extremely high [7]. Hence, it is critical to assess the potential advantages of the use of this information in providing a more efficient scheduling [19]. Moreover, Industry 4.0 technologies may enable not only accessing to real-time shop floor data, but they can

* Corresponding author.:.
E-mail addresses: vfernandezviagas@us.es (V. Fernandez-Viagas), framinan@us.es (J.M. Framinan).

https://doi.org/10.1016/j.jii.2021.100281
Received 29 July 2020; Received in revised form 2 May 2021; Accepted 30 August 2021
Available online 1 September 2021
2452-414X/

also provide advance information regarding the flow of raw materials entering to the process (either coming from providers or from a process upstream), and with respect to the constraints required by downstream processes or by customers [16]. Clearly, the access to all data sources in information integration is costly and sometimes even impossible (32), therefore it is also crucial to analyse separately the contribution of each one of them to the global scheduling efficiency. More specifically, the questions motivating this research are:

- How much the predictive approaches can be improved by predictive-reactive ones using real-time, integrated information collected not only from the shop floor status, but also from upstream/downstream processes? Note that this question is not trivial, as 1) there are solution procedures within the predictive approach –such as e.g. stochastic scheduling– that can effectively handle shop floor uncertainties and 2) there is evidence that, in certain cases, deterministic predictive approaches may be quite robust to uncertainty (see e.g. 14). Furthermore, it is known that continuous scheduling updates such as in the predictive-reactive approach may introduce nervousness in the system and may not improve its efficiency (see the evidence collected by 25).
- What is the contribution of the different data sources in improving the scheduling process (including downstream/upstream information flows)? This is an aspect that, to the best of our knowledge, has been neglected in the literature so far, which has focused mainly on data regarding the shop floor status. Indeed, recent papers addressing production scheduling in the context of Industry 4.0 point out to the need of conducting further research by investigating additional sources of uncertainty such as changes in the due dates and material shortages [19].

To investigate these questions, we conduct a series of experiments aimed at quantifying the benefits in scheduling for different widely-employed scheduling criteria. We use the flowshop as the testing platform in our experiments, due to its extensive use both in practice and in academia. We employ two different scheduling criteria (total flowtime and total tardiness), the first one as a typical objective aimed at maximizing the internal efficiency of the process, and the second 1 aimed at complying with the requirements of downstream processes/customers. Taking into account these different criteria, we develop several *scenarios*, each one determined by a specific approach to solve the scheduling decision-problem by means of a specific solution procedure. These scenarios range from the case where predictive approaches and deterministic procedures are used to solve a scheduling problem, to the case where predictive-reactive approaches are coupled with stochastic solution procedures.

The extensive experimentation carried out allows extracting a number of conclusions which can be useful for practitioners and academics. Perhaps the bottom line is that it should not be taken for granted that integrated, real-time information (i.e. using real-time shop floor data and advanced information from upstream/downstream processes) will improve the efficiency of the operations *per se* –at least regarding the scheduling process, since such higher efficiency heavily depends on the proper choice of scheduling approaches, on the data sources, and on how the solution procedures use these data. For instance, it is relatively less important to have shop floor status data in real-time than having advance downstream and upstream information. Among the two latter, upstream information can be much more useful in increasing the quality of the schedule. On the other hand, for a predictive scheduling approach, it seems that stochastic methods do not seem to greatly outperform deterministic ones, being much CPU-time consuming than the latter ones. However, this is not the case in predictive-reactive scenarios, where their higher computational requirements seem to pay-off.

The rest of the paper is organised as follows: in Section 2, we introduce the problem background and discuss the related literature. A formal description of the problem and the research methodology are presented in Section 3. The experimental scenarios are detailed in Section 4, while the computational results are shown in Section 5. Finally, the conclusions of our study are discussed in Section 6.

2. Background

The research conducted in this paper refers to three interrelated areas. The first one is the efficiency of the different approaches that can be used for scheduling (more specifically, that of predictive scheduling as compared to predictive-reactive approaches). The second one is related to the effectiveness of stochastic vs deterministic scheduling procedures, both for predictive and for predictive-reactive approaches. Finally, our research analyses the usage of real-time information for scheduling decisions. In these three areas, we will briefly describe the main contributions and point out the open research questions.

Regarding the scheduling approaches, the classical paper by Vieira et al. [48] describes a general framework to classify them. It is usual to distinguish between approaches that use dispatching rules (or reactive approach), and those carrying out a schedule generation [5]. In the reactive approach, a priority rule is employed to decide which job is processed first once a machine becomes idle, so there is no need to explicitly generate a schedule. Although this approach is fast and easy to implement, there is empirical evidence that, for complex systems with high competition for resources, it is usually outperformed by approaches using schedule generation and it will not be discussed further.

Among the approaches using schedule generation, it is usual to distinguish between the two following approaches:

- *Predictive* approaches. In the predictive approach, a schedule is built at the beginning of the decision period, and this schedule is executed regardless the new events that might occur during its execution. Although most of the classical scheduling literature assumes that, in this approach, the schedule is generated assuming that the data are deterministic (or at least that their variance is sufficiently small so their means are statistically representative), this does not have to be the case, as the unforeseen events can be incorporated in the initial schedule in an implicit manner by e.g. generating robust schedules, or schedules that assume some probability distribution in the data so the average value of the scheduling criteria is estimated via simulation (see some examples of this approach in [3,14,23]).
- *Predictive-reactive procedures*. In the predictive-reactive approach, a predictive schedule (*base schedule*) is generated as in the predictive approach. However, a modified schedule can be generated (rescheduling) in view of the incoming information. Again, the procedure to generate the modified schedule can be deterministic, or it can incorporate some stochastic considerations.

  Clearly, the frequency or timing for triggering the rescheduling procedure is critical. According to Church and Uzsoy [5], Sabuncuoglu and Bayiz [41], Vieira et al. [48], three different policies have been considered in the literature:
  - *Continuous rescheduling* (CR). In this policy, rescheduling is performed every time an event that is recognised by the system (e.g. new job arrivals, machine breakdowns, etc) occurs.
  - *Periodic rescheduling* (PR). In this policy, the rescheduling process is trigged in given time intervals (rescheduling points or times).
  - *Event-driven rescheduling* (EDR). In this case, the process is triggered if certain conditions related to the system occurs. Note that both CR and PR can be seen as a particular case of EDR (as it is done e.g. in 48), but most of the literature introduces this differentiation (e.g. 1), and this will also be the case here.

The literature is abundant in contributions comparing the approaches to generate a schedule in different layouts. Sabuncuoglu and Bayiz [41] compare two specific predictive and reactive approaches in a dynamic deterministic job shop scheduling problem with and without machine breakdowns. More specifically, they use a deterministic procedure as the predictive procedure (a beam-search-based constructive heuristic) which is compared against a dispatching rule applied as the reactive procedure. In their study, the predictive procedure clearly improves the reactive one, although is more affected by the disturbances. More recently, Larsen and Pranzo [29] propose a new framework which using a solver addresses a dynamic job shop scheduling problem.Framinan et al. [12] compare different rescheduling strategies against the predictive approach in a flowshop scheduling problem with stochastic processing times. These two recent contributions find that, under certain conditions, the rescheduling procedure does not improve the solutions for makespan minimisation. Particularly, Larsen and Pranzo [29] found no benefit in rescheduling when the range of processing times is narrow, while in Framinan et al. [12] the same result is observed when the variability is high.

As it can be seen, despite these contributions, there are questions that remain open regarding the scheduling approaches. The first one refers to the fact that most *works* consider the makespan as objective, which is known to be very robust with respect to the variability of the solutions, so it is interesting to know whether these results also hold for other objectives. Furthermore, the solution procedures employed for rescheduling are –to the best of our knowledge– based on either deterministic scheduling or on dispatching rules, whereas the use of stochastic rescheduling methods such as the one discussed below has not been employed.

Regarding the second topic (the effectiveness of stochastic / deterministic procedures), in the context of predictive approaches, Framinan and Perez-Gonzalez [14] compare both approaches for a flowshop layout with variable processing times and makespan minimisation as objective. They find that the deterministic procedures work quite well as compared to their stochastic counterpart, particularly when the variability of the processing times is high. However, to the best of our knowledge, this comparison has not been discussed in the context of predictive-reactive approaches.

Finally, with respect to the use of real-time information integration for rescheduling, the literature is very scarce. In the single machine setting, the work by [6] employs experimentation to show that there are different strategies in which the real-time information regarding the processing times can be advantageously employed. The paper by Framinan et al. [12] investigates the problem of incorporating real-time information in a flowshop to reschedule the jobs with the objective of minimizing the makespan. They find that a careful choice of base schedule, rescheduling policy and rescheduling procedures is required to take advantage of the additional real-time information. Finally, the recent paper by [19] studies a flexible job shop with uncertainties regarding the arrival of new jobs and machine breakdowns. In this setting, the authors find that the usage of real-time information can substantially improve the performance of the system. As it can be seen, in none of these studies external sources of variability (i.e. upstream and downstream processes) are considered.

As a summary of the state-of-the-art described in the section, further research is needed to assess the contribution of different sources of real-time data, as well as on the scheduling approaches and solution procedures that can benefit from the integration of this data in the decision-making process. In the next section, we present the methodology employed to carry out this research.

3. Research methodology

In this section we present the research methodology by first introducing the manufacturing process modeled in the experiments. Since in some cases several process variables and data are unknown at the time of the decision-making process, these are denoted using capital letters in the following.

As discussed in Section 1, the experiments to be carried out refer to the problem of scheduling jobs in a flowshop layout. In this problem, there are n jobs, each one composed of m operations that have to be carried out in a set of m machines, o_{ij} denoting the operation of job j that has to be performed on machine i and P_{ij} denoting its processing time. In addition, each job has a release date R_j, which is typically imposed by the availability of raw materials for the process, either coming from upstream processes or from suppliers. Similarly, each job j is characterized by a due date D_j, which may represent a customer requirement or an expected completion time from downstream processes in the facility.

In most real-life settings, processing times are not known in advance, although some estimation can be provided based on historical data. Similarly, there is uncertainty in the release dates and in the due dates of the jobs: upstream processes/suppliers are not, in general, fully reliable, while customers or downstream processes are subject to uncertainty themselves and therefore may change their requirements or expectations. However, some estimate can be provided on both R_j and D_j based either on the initial commitments (requirements) from suppliers (customers) or from historical/estimate data from upstream (downstream) processes. Let us denote by $\widehat{p}_{ij}$, $\widehat{r}_j$ and $\widehat{d}_j$ such estimated values. Whenever the actual realizations of P_{ij}, R_j and D_j are known for a given instance, the corresponding lowercase letter is employed (i.e. p_{ij}, r_j and d_j)

In this setting, the goal of the decision-making problem is to find the sequence of jobs $\Pi := (\pi_1, \ldots, \pi_n)$ on the machines that minimises the expected value of the flow time ($\sum_{\forall j} E[F_j]$) or the expected value of the total tardiness ($\sum_{\forall j} E[T_j]$). The flowtime of job j (F_j) and the tardiness of that job (T_j) are defined by Eqs. 1 and 2, respectively:

$$F_j = C_{mj} - R_j \tag{1}$$

$$T_j = \max\{0, C_{mj} - D_j\} \tag{2}$$

where the completion times, C_{ij}, are defined by Eq. 3.

$$C_{i,\pi_j} = \max\{C_{i-1,\pi_j}, C_{i,\pi_{j-1}}\} + P_{i,\pi_j}, \forall i = 2, \ldots, m, j = 1, \ldots, n \tag{3}$$

In addition, $C_{i,\pi_0} = 0$ and C_{1,π_j} are defined by the following expression:

$$C_{1,\pi_j} = \max\{R_{\pi_j}, C_{1,\pi_{j-1}}\} + P_{i,\pi_j}, \forall j = 1, \ldots, n \tag{4}$$

The choice of these two objectives for the experiments is motivated by the fact that they are extremely good indicators of both the *internal* and *external* performance of a schedule: On the one hand, flowtime minimization is known to be well-aligned with work-in-process and average cycle time minimization (which are related to inventory and lead time costs) while performing reasonably well with respect to throughput maximization (which is related to equipment utilization) [13]. On the other hand, tardiness minimization is a critical indicator for downstream processes or customers, as it is also widely employed to assess the effectiveness of the operations (see e.g. 46). Finally, note that we assume that the same sequence of jobs is adopted for all machines (permutation constraint), a hypothesis that is widely used.

As discussed in Section 2, one option is to solve the scheduling decision problem using a predictive approach. In this case, since the scheduling decision is made before, at the beginning of the decision interval, P_{ij}, R_j, and D_j are uncertain and some estimates of them have to be produced to be employed in the scheduling procedure. However, in a predictive-reactive approach, this schedule may be reviewed at a later time. At this point in time, different data sources might be available. More specifically, these are:

- Shop floor status data ($\mathscr{S}_F$). In this case, the actual processing status of each job at the time of the rescheduling is known, so it is possible to know what jobs have been already processed (or are being processed) by some machine. Thus, the rescheduling process could incorporate the actual realizations of the completion times.
- Advance downstream data ($\mathscr{S}_C$). In this case, at the beginning of the decision period, for each job j a due date $\widehat{d}_j$ is given by the customers

or it is set by some procedure as the expected time to initiate the downstream process. This initial due date can be changed in the future, as the internal variability of the customers' (downstream) processes may delay or expedite the need of this job. However, as the time advances there is a point where a final due date (the same or different than the initial one) is confirmed and *frozen* (i.e. it cannot be further modified). This information could eventually be passed in real-time to the rescheduling process, so the decision-making procedure might use it.

- Advance upstream data ($\mathscr{S}_S$). In this case, the initial release date $\hat{r}_j$ quoted for each job j can be subject to changes due to suppliers or upstream processes variability. Similar to the previous case, there is a point where a final, frozen release date is set. Again, this information could eventually be passed in real-time to the rescheduling process, so the decision-making procedure might use it.

For a given combination of a scheduling approach, a solution procedure to solve the subsequent scheduling/rescheduling problems, and the type of data sources available at the time that the procedure is invoked – which is denoted as *scenario* in the following–, schedules of different quality can be obtained. In the next section, we define in detail the scenarios to be considered in our research, while the computational results obtained are discussed in Section 5.

4. Experimental scenarios

Four types of scenarios are considered in this research depending on the approach and the type of scheduling/rescheduling procedure employed to solve the decision problem:

- Scenarios (P,D). A deterministic procedure (using the estimates of the input data, i.e. $\hat{p}_{ij}$, $\hat{r}_j$ and $\hat{d}_j$) is employed to obtain an initial schedule which is not modified subsequently. These scenarios are described in detail in Section 4.1.
- Scenarios (P,S). A stochastic procedure is employed to obtain an initial schedule which is not modified subsequently. This procedure assumes that the input data P_{ij}, R_j and D_j follow some known distribution. These scenarios are described in detail in Section 4.2.
- Scenarios (PR,D). A deterministic procedure (using the estimates of the input data, i.e. $\hat{p}_{ij}$, $\hat{r}_j$ and $\hat{d}_j$) is employed to obtain an initial schedule. Then, using a periodic rescheduling policy, each γ time units, a deterministic rescheduling procedure is used to provide a new schedule. This deterministic rescheduling procedure may use data from all/some of the sources described in Section 3, i.e. $\mathscr{S}_F$, $\mathscr{S}_S$ and/or $\mathscr{S}_F$. These scenarios are described in detail in Section 4.3.
- Scenarios (PR,S). A stochastic procedure (assuming some distribution of the input data P_{ij}, R_j and D_j) is employed to obtain an initial schedule. Then, using a periodic rescheduling policy, each γ time units, a stochastic rescheduling procedure is used to provide a new schedule. This rescheduling procedure may use data from all/some of the sources described in Section 3, i.e. $\mathscr{S}_F$, $\mathscr{S}_S$ and/or $\mathscr{S}_F$. These scenarios are described in detail in Section 4.4.

In order to implement these types of scenarios, the different approaches, solution procedures –including different criteria– and data availability have to be carefully designed. This design is described in the next subsections. Finally, note that, although the periodic rescheduling policy is assumed in the scenarios, the experiments have been conducted using also other policies and will be briefly discussed in Section 5, even if the full results are not presented due to the lack of space.

4.1. Predictive approach, deterministic procedures (P,D)

As discussed before, in this type of scenarios a decision is taken by solving the corresponding deterministic scheduling problem, and this decision is not modified subsequently (predictive approach). The data required to solve the scheduling problem (i.e. P_{ij}, R_j and D_j) are not known at this time and their mean (i.e. $E[P_{ij}]$, $E[R_j]$, and $E[D_j]$, respectively) are used instead. Note that this is equivalent to assume a MMSE (Minimal Mean Square Error) estimate of these data if they are independent and identically distributed.

Two solution procedures are employed to assess the effect of the quality of the solution in this scenario, denoted as $\mathscr{P}_D$ and $\mathscr{P}_D^{LS}$. $\mathscr{P}_D$ is a proxy of a high-quality, time-consuming solution procedure that would eventually yield a near-optimal solution to the problem. $\mathscr{P}_D^{LS}$ is a proxy of a fast solution procedure (typically a constructive or composite heuristic) that, however, would probably yield a solution of lower quality. Obviously, the procedures selected are objective-dependent, so we have selected the following:

- For the total flowtime objective ($Fm|prmu, r_j|\sum F_j$ problem), we adapt for $\mathscr{P}_D$ the *MRSILS* algorithm proposed by Dong et al. [8] due to their excellent performance in the $Fm|prmu|\sum C_j$ problem (see 10). Basically, this metaheuristic iteratively inserts a job into the best position of the iteration sequence. Once n jobs have been tested, a sequence is randomly selected from a pool and then, is perturbed by inserting a random job into another random position. We refer to Dong et al. [8] for a fully description of the algorithm. The NEH algorithm (15) is used as the initial solution. Regarding $\mathscr{P}_D^{LS}$, it consists in a simple insertion-based local search (denoted as LS_D) on the solution obtained by the previous NEH algorithm, i.e. starting with the NEH solution, each job is re-inserted in all positions and the one yielding the lowest value of the objective function is retained.
- For the total tardiness objective ($Fm|prmu, r_j|\sum T_j$ problem), we adapt for $\mathscr{P}_D$ the iterated local search (IA_{RAS}) metaheuristic by [11], which is the state-of-the-art metaheuristic for the problem. Basically, this algorithm iteratively perturbs an iteration solution and search its local optimum using an insertion local search method. After that, the acceptance criterion proposed by [26] is adopted. This procedure is repeated until the stopping criterion is reached. For the perturbation phase, the algorithm randomly performs $d = 4$ adjacent interchanges. We refer to [11] for a fully description of this algorithm. As in the previous case, in order to adapt the algorithm to the $Fm|prmu, r_j|\sum T_j$ problem, the NEH algorithm (27) replaces the initial solution of the algorithm. Furthermore, $\mathscr{P}_D^{LS}$ is the same as the previous case but replacing the initial solution from [15] by the previous NEH variant.

4.2. Predictive approach, stochastic procedures (P,S)

In this scenario, a schedule is given at the beginning of the decision interval using procedures assuming that the relevant data are random variables with a known distribution. This schedule is not modified subsequently (predictive approach).

Regarding the scheduling procedure adopted, we use a stochastic version of the procedure employed in the (P,D) scenario. Since the evaluation of the objective function is carried out by means of an extremely high number of simulations, it does not make sense to distinguish among different procedures (fast, slow) as it is done for the (P,D) scenario. More specifically, in this scenario an initial deterministic solution is obtained by applying the NEH heuristic (15 for the $Fm|prmu, r_j|\sum F_j$ problem, and 27 for the $Fm|prmu, r_j|\sum T_j$ problem). Then, a stochastic variant of the LS_D local search (denoted as LS_S in the following) is applied. This variant is identical in its steps to the deterministic local search being the only difference that the value of the objective function is estimated by running simulations according to the procedure proposed by Framinan and Perez-Gonzalez [14]. We refer the reader to this reference for the details of the procedure, and simply recall that the parameters of this procedure employed for the experiments are $\alpha = 0.001$, $p = 0.01$ and a maximum number of 15,000,000 simulations).

This procedure is denoted by P_S in the following.

4.3. Predictive-reactive approach, deterministic procedures (PR,D)

In this scenario it is assumed that, although an initial schedule (obtained from the application of some deterministic procedure such as the ones discussed in Section 4.1) is executed, some data from the shop floor status or from the downstream/upstream processes are captured in real-time and are employed to reschedule the jobs.

More specifically, a periodic rescheduling (PR) policy is adopted, so a rescheduling procedure is performed each γ time units. In our experiments, γ is set to 50 time units. The rationale is that, since the processing times of the jobs in the testbeds are generated from a random [1,99] distribution (see Section 5.1), their average is 50 time units. Therefore, using lowest value for γ may render the rescheduling useless, as the status of the jobs may be identical to the one since the last reschedule. On the contrary, using much higher values for γ can fail to adequately respond to the changes in the shop floor status.

In the PR policy, whenever the rescheduling procedure is triggered at time ρ, the set of jobs $\mathscr{J}$ considered in the initial schedule is in one of the three following sets:

- Finished jobs (set $\mathscr{J}_F$). This set is composed of jobs in $\mathscr{J}$ that are completed by the time ρ at which the rescheduling procedure is invoked, i.e. $\mathscr{J}_F := \{j \in \mathscr{J} : C_{mj} \le \rho\}$), see Fig. 1, where all jobs until position k are finished.
- In-process jobs (set $\mathscr{J}_P$). This set is composed of jobs in $\mathscr{J}$ that have started to be processed, but are not finished. An example of these jobs with at least one operation not finished is shown in Fig. 1 (jobs in position $k+1$ to j).
- Remaining jobs (set $\mathscr{J}_R$). This set is composed of jobs in $\mathscr{J}$ that have not been started to be processed in any machine.

Clearly, there is no rescheduling decision affecting the set $\mathscr{J}_F$, as these jobs are already completed. Furthermore, although the remaining operations of the jobs in $\mathscr{J}_P$ could be eventually changed, if we assume the permutation constraint, their sequence in the remaining machines cannot be altered either. Therefore, the jobs in $\mathscr{J}_R$ are the only ones altered by the rescheduling procedure. Note, however, that such rescheduling procedure must take into account a_i the availability time of each machine i, as the machine would become available after processing all the corresponding operations from the jobs in $\mathscr{J}_P$. Such availability has to be estimated, as the processing times are not known in advance. To perform this estimation, we use the procedure in Framinan et al. [12], can be summarised as follows. First, a *reference* completion time $\widetilde{c}_{i\pi_j}$ for each job $\pi_j \in \mathscr{J}_F \cup \mathscr{J}_P$ on each machine i is defined. $\widetilde{c}_{i\pi_j}$ indicates the actual completion time if the operation in machine i is completed by time ρ. Otherwise, $\widetilde{c}_{i\pi_j}$ is estimated by assuming that the operation would take their average processing time and using the actual or average (estimated) release date for this job (see Eq. 5). Then, the availability of the first machine can be estimated using the reference completion times of the in-process and completed jobs, i.e.: $a_1 = \max_{j\in\{\mathscr{J}_F\cup\mathscr{J}_P\}}\{\widetilde{c}_{1\pi_j}\}$. Similarly, the reference completion times can be recursively computed for the rest of the machines using Eq. 6, and the availability of machine $i>1$ is computed, i.e. $a_i = \max_{j\in\{\mathscr{J}_F\cup\mathscr{J}_P\}}\{\widetilde{c}_{i\pi_j}\}$.

$$\widetilde{c}_{1\pi_j} = \begin{cases} c_{1\pi_j}, & \text{if } c_{1\pi_j} \le \rho \\ \max\left\{\rho, \max\left\{\widetilde{c}_{1\pi_{j-1}}, r_{\pi_j}\right\} + \widehat{p}_{1\pi_j}\right\}, & \text{if } c_{1\pi_{j-1}} \le \rho \,\&\, c_{1\pi_j} > \rho \,\&\, r_{\pi_j} \le \rho \\ \max\left\{\rho, \widehat{r}_{\pi_j}\right\} + \widehat{p}_{1\pi_j}, & \text{if } c_{1\pi_{j-1}} \le \rho \,\&\, c_{1\pi_j} > \rho; \& \, r_{\pi_j} > \rho \end{cases} \tag{5}$$

$$\widetilde{c}_{i\pi_j}(\forall i > 1) = \begin{cases} c_{i\pi_j}, & \text{if } c_{i\pi_j} \le \rho \\ \max\left\{\rho, \max\left\{c_{i-1,\pi_j}, c_{i,\pi_{j-1}}\right\} + \widehat{p}_{i\pi_j}\right\}, & \text{if } c_{i\pi_j} > \rho \end{cases} \tag{6}$$

Therefore, each γ time units, the availability of each machine is determined by the procedure above, and a new schedule for the jobs in the $\mathscr{J}_R$ set is constructed by solving the $Fm|prmu, r_j, a_i|\sum F_j$ ($Fm|prmu, r_j, a_i|\sum T_j$) problems. The procedures to generate these schedules are the same than in the predictive approach, i.e. MRSILS and IA_{RAS} (note that, from a decision problem viewpoint, the only difference between this scenario and the predictive one is the set of jobs to be scheduled). However, if some real-time data is available at time ρ (either from the shop floor status or from the downstream/upstream processes), then this data can be incorporated into the decision problem. More specifically, the following scenarios depending on the sources of real-time data can be considered:

- Shop floor status data (scenario $\mathscr{PR}_D^{\mathscr{J}_F}$). In this scenario, the shop floor status data available at time ρ (i.e. the actual completion times

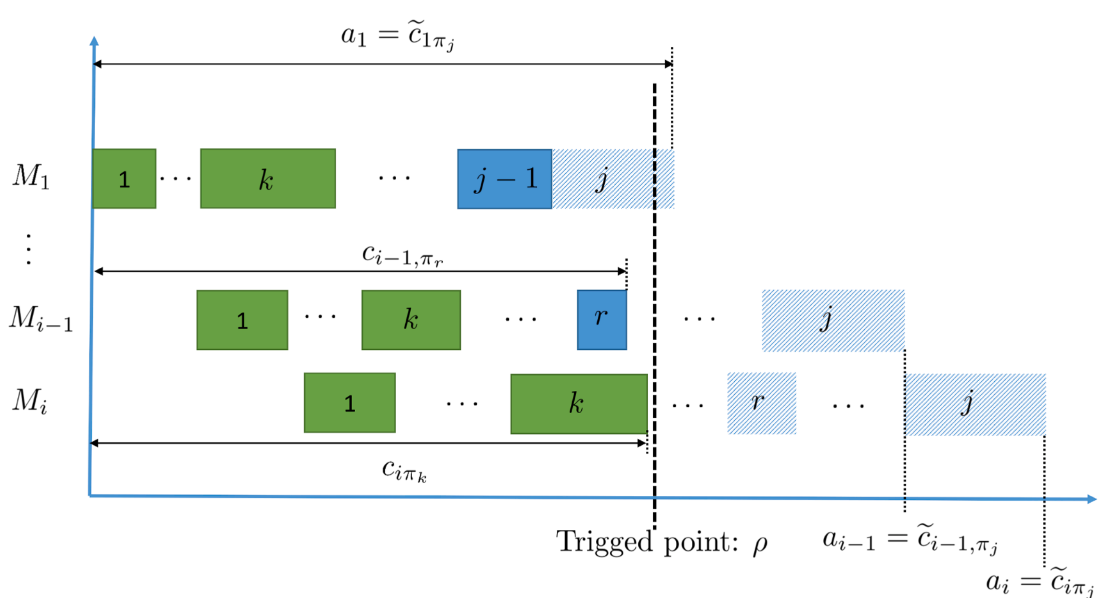

Fig. 1. Estimation of initial availabilities.

of the jobs in $\mathscr{J}_P$) is incorporated into the scheduling procedures. No additional data from upstream/downstream processes is assumed to be available in this scenario.

- Advance downstream data (scenario $\mathscr{PR}_D^{\mathscr{J}^{FC}}$). In this scenario, at time ρ, the data related to the downstream process (i.e. the due date of the jobs in $\mathscr{J}_R$) is known δ time units before its realization. More specifically, for each job j, the 'true' value of its due date d_j is known δ time units before its realisation (i.e., if $d_j \leq \rho + \delta$ then d_j is used as input for the decision problem). Furthermore, shop floor status data are also available at time ρ. The inclusion of δ as a parameter would serve to assess the relevance of timely downstream information in this scenario. Clearly, this scenario is only relevant for the total tardiness criterion, as the same problem with total flowtime as objective does not require downstream data.
- Advance upstream data (scenario $\mathscr{PR}_D^{\mathscr{J}^{FS}}$). In an analogous manner to the previous one, in this scenario, at time ρ, the data related to the upstream process (i.e. the release date of the jobs in $\mathscr{J}_R$) is known δ time units before its realization, i.e. if $r_j \leq \rho + \delta$ then its realization r_j is used as input for the decision problem, which now is relevant both for total tardiness and for total flowtime criteria. Furthermore, shop floor status data are also available at time ρ.
- Integrated information (scenario $\mathscr{PR}_D^{\mathscr{J}^{FCS}}$). In this scenario, the data related to both the upstream and downstream processes (i.e. the release and due dates of the jobs in $\mathscr{J}_R$) are known δ time units before their realization, in addition to shop floor status data.

4.4. Predictive-reactive approach, stochastic procedures

This scenario differs from the previous one in the procedure employed both to generate the base schedule and the rescheduling. In this case, the base schedule is provided by the same stochastic procedure than in Section 4.2 and, every γ time units, this procedure is applied to reschedule the jobs that at this time are in the set $\mathscr{J}_R$. For the rescheduling procedure, the same three sub-scenarios regarding the available data are considered, i.e.: Shop floor status data (scenario $\mathscr{PR}_S^{\mathscr{J}^F}$), Advance downstream data (scenario $\mathscr{PR}_S^{\mathscr{J}^{FC}}$), Advance upstream data (scenario $\mathscr{PR}_S^{\mathscr{J}^{FS}}$), and Integrated information (scenario $\mathscr{PR}_S^{\mathscr{J}^{FCS}}$).

5. Computational results

In this section we present the computational results of the experimentation in the different scenarios. The experiments have been run on a cluster of computers Intel Core i7-3770 PC with 3.4 GHz and 16 GB RAM and using C# under Visual Studio 2019. All the extensive experimentation included in the paper has taken 6.3 years of combined CPU time. Due to the high computational effort required by the methods $\mathscr{P}_S$ and $\mathscr{PR}_S$ methods, two computational evaluation are carried out. Firstly, we compare all procedures detailed in Section 4 in a set of small instances. After that, we extend some results by comparing the set of non-stochastic procedures in medium-big size instances. To do so, we explain the generation of the sets of instances in Section 5.1, and we analyse the results obtained in the sets of instances in Section 5.2. The computational evaluations included in this study are performed for both total flowtime and total tardiness minimisation criteria. In addition, all comparisons are carried out using the Average Relative Percentage Deviation (ARPD, see Eq. 7) as indicators of the quality of the solutions for the total flowtime, and the Average Relative Deviation Index (ARDI, see Eq. 8) for the total tardiness. I is the total number of instances, OF_{ip} is the objective function value found by procedure p in instance i, and $Best_i$ ($Worst_i$) is the best (worst) value found in the instance among all procedure tested.

$$ARPD = \frac{100}{I} \sum_{i=1}^{I} \frac{OF_{ip} - Best_i}{Best_i} \tag{7}$$

$$ARDI = \frac{100}{I} \sum_{i=1}^{I} \frac{OF_{ip} - Best_i}{Worst_i - Best_i} \tag{8}$$

5.1. Testbed generation

In this section, we detail the procedure adopted to generate two sets of instances, denoted as β_1 and β_2. β_1 is a benchmark composed of 1,728 small-sized instances, while β_2 is composed of 3456 medium/big instances. In both sets, the following parameters have to be defined: processing times P_{ij}, due dates D_j, release dates R_j, number of jobs n, and number of machines m. Regarding the generation of the due dates, several approaches have been used in the literature, see e.g. Gelders and Sambandam [18], Hasija and Rajendran [22], Potts and Van Wassenhove [38]. In this paper, we apply the most widely-employed procedure, developed by [38] (see 47), to generate the mean of the due date distributions. This procedure generates the mean of each job according to a uniform distribution between $P\cdot(1 - T - R/2)$ and $P\cdot(1 - T + R/2)$, where T and R are parameters to control the variability in the mean, and P is a lower bound of the makespan, taken from Taillard [45]. Regarding the release dates, also some approaches have been used in the related scheduling problem literature so far. Several authors (see e.g. [28,43,44]) use release times generated by uniform distributions between two constants values (e.g. $r_j \in [0, 100]$ or by $r_j \in [0, 200]$), while others authors (e.g. [2,21,51]) use release dates following a uniform distribution between 0 and a multiple of the number of jobs (e.g. $[0, 5n]$). Both approaches have been found by Hall and Posner [20] inadequate to generate a wide range of instances. In view of this, we generate four types of the means for the release times following a similar procedure as Hall and Posner [20] and Mrad et al. [34]:

1. $\alpha = 1$: $E[R_j] = E[R_{j-1}] + X_{10}$ where X_{10} is a random number generated from an exponential distribution with mean equals to 10.
2. $\alpha = 2$: $E[R_j] = E[R_{j-1}] + X_{10}$ where X_{50} is a random number generated from an exponential distribution with mean equals to 50.
3. $\alpha = 3$: $E[R_j] = E[R_{j-1}] + E[P_{1,j}] + X_{10}$.
4. $\alpha = 4$: $E[R_j] = E[R_{j-1}] + E[P_{1,j}] + X_{50}$.

Taking these aspects into account, the parameters used for generating both benchmarks are detailed as follows:

- Benchmark β_1. 1,728 instances are generated with the following levels of the parameters $n \in \{10,15,20\}$, $m \in \{5,10\}$, $\alpha \in \{1,2,3,4\}$, $T \in \{0.2,0.4,0.6\}$, and $R \in \{0.2,0.6,1.0\}$. In addition, we include CV as the coefficient of variation ($CV = \frac{\sigma}{\mu}$) to control the standard deviation (σ) of the stochastic due dates, release dates and processing times. More specifically, four different levels for the coefficient are selected ($CV \in \{0.1, 0.5, 1.0, 1.5\}$) to represent very low, low, medium, and large variability in the manufacturing shop floors, respectively (see e.g. 24). For each combination of the previous parameters, two instances are generated. Processing times, due dates, and release dates are assumed to follow log normal distributions. The mean μ of the due dates depends on R and T and on α for the release dates, as explained above. Regarding the processing times, we use a uniform distribution [1,99] for the mean μ. The standard deviation of each previous data is computed by the previous means μ and CV according to $\sigma = \mu CV$.
- Benchmark β_2. In this benchmark, we consider the following levels of the parameters to reproduce medium-big size instances: $n \in \{25, 50, 75\}$, $m \in \{2, 5, 10, 15\}$, $\alpha \in \{1, 2, 3, 4\}$, $T \in \{0.2, 0.4, 0.6\}$, and $R \in \{0.2, 0.6, 1.0\}$. Regarding D_j, R_j, P_{ij}, we use the same procedure to determine the mean and standard deviation of these stochastic distributions.

5.2. Computational results

Computational results are shown in Tables 1 and 2 for total flowtime and total tardiness minimisation, respectively. The most extended procedure to solve the stochastic problem under consideration, i.e. the deterministic approach, yields an ARPD value of 24.89 and 25.25 for total flowtime and total tardiness respectively. The variation of each procedure with respect to the coefficient of variation (CV) is shown in Tables 3 and 4.

The following conclusions can be derived from the results:

- Overall, when a predictive approach is adopted, there are no great differences between taking into account the uncertainties in the processing times by using a stochastic approach ($\mathscr{P}_S$), and addressing the problem in a deterministic manner employing the average values of the uncertain variables ($\mathscr{P}_D^{LS}$). This trend is observed regardless the degree of uncertainty in the processing times. In view of these results and since the former approach is much more time-consuming, it is worth questioning if such higher requirements are justified by the differences in the quality of the solutions. On the other hand, the results are better if the deterministic procedure employed is of high quality ($\mathscr{P}_D$), which speaks for the importance of having efficient deterministic solution procedures for predictive scheduling. Note that this conclusion holds both for flowtime and tardiness minimisation, and it is in line with prior results for different objectives (see e.g. the work by 14 for makespan minimisation).
- However, if a predictive-reactive approach (i.e. periodic rescheduling) is adopted, using a stochastic procedure ($\mathscr{PR}_S$) results in substantially better results as compared to the deterministic one ($\mathscr{PR}_D$). This happens regardless the objective considered, and whether upstream or downstream data is available, or not. Thereby, e.g. when only internal (shop floor status) real-time data are considered, the ARPD (ARDI) value is reduced from 31.06 (using $\mathscr{PR}_D^{\mathscr{I}_F}$) to 23.75 (using $\mathscr{PR}_S^{\mathscr{I}_F}$) in the $Fm|prmu, r_j|\sum F_j$ problem, and from 29.90 ($\mathscr{PR}_D^{\mathscr{I}_F}$) to 24.22 ($\mathscr{PR}_S^{\mathscr{I}_F}$) in the $Fm|prmu, r_j|\sum T_j$ problem (the hypothesis that $\mathscr{PR}_D^{\mathscr{I}_F} = \mathscr{PR}_S^{\mathscr{I}_F}$ is rejected in both cases, using a non-parametric Mann-Whitney test, with p-values of 0.003 and 0.032, respectively). Furthermore, the difference between both approaches increases with the degree of uncertainty in the processing times, which seems foreseeable. Only for the flowtime objective (where downstream data are not required) and for the scenario with lowest variability of the processing times (i.e. $CV = 0.1$) both approaches yield similar results.
- Regarding the usage of external real-time data (i.e. upstream and downstream), several conclusions can be obtained:
  - The predictive-reactive approach including data from suppliers ($\mathscr{PR}_D^{\mathscr{I}_{FS}}$) clearly improves the total flowtime of the solutions regardless the value of the applied parameter δ (see results in Table 5 for $\delta \in \{50, 100, 200, 300\}$). Thereby, e.g. $\mathscr{PR}_D^{\mathscr{I}_{FS}}$ with $\delta = 200$ finds an ARPD value of 16.02, which outperforms a rescheduling policy with only shop floor status data ($\mathscr{I}_F$) with an ARPD value of 31.06 (which means a close to 50% reduction). This is confirmed by a p-value of 0.000 using a non-parametric Mann-Whitney test (with the hypothesis $\mathscr{PR}_D^{\mathscr{I}_{FS}} = \mathscr{PR}_D^{\mathscr{I}_F}$). The improvement is even higher using a stochastic rescheduling approach with $\mathscr{I}_{FS}$ (i.e. $\mathscr{PR}_S^{\mathscr{I}_{FS}}$) which found an ARPD value of 7.88 (for $\delta = 200$) versus 23.75 of $\mathscr{PR}_S^{\mathscr{I}_F}$.
  - Incorporating advance downstream data in the predictive-reactive approach with the objective of total tardiness minimization has a much lower incidence than incorporating the advanced upstream data when the same time δ is used (we assume $\delta = 200$ in the experimentation). Thereby, the ARDI found in scenario $\mathscr{PR}_D^{\mathscr{I}_{FS}}$ ($\mathscr{PR}_S^{\mathscr{I}_{FS}}$) is 29.21 (23.83), while in $\mathscr{PR}_D^{\mathscr{I}_{FC}}$ ($\mathscr{PR}_S^{\mathscr{I}_{FC}}$) is 16.84 (9.93). This conclusion is also confirmed by a Mann-Whitney test finding a p-value equal to 0.000 (for the both hypotheses $\mathscr{PR}_D^{\mathscr{I}_{FC}} = \mathscr{PR}_D^{\mathscr{I}_{FS}}$ or $\mathscr{PR}_S^{\mathscr{I}_{FC}} = \mathscr{PR}_S^{\mathscr{I}_{FS}}$). In fact, advance downstream information has almost no influence in the quality solution. Thereby, the hypothesis that $\mathscr{PR}_D^{\mathscr{I}_F} = \mathscr{PR}_D^{\mathscr{I}_{FC}}$ ($\mathscr{PR}_S^{\mathscr{I}_F} = \mathscr{PR}_S^{\mathscr{I}_{FC}}$) cannot be rejected finding a p-value equal to 0.625 (0.331).
  - Finally, the advantages of an integrated information scenario are worth to be highlighted for both criteria. For instance, for the total tardiness, the ARDI is reduced from 29.90 (obtained by $\mathscr{PR}_D^{\mathscr{I}_F}$) to 16.58 (obtained by $\mathscr{PR}_D^{\mathscr{I}_{FCS}}$) or to 9.64 (obtained by $\mathscr{PR}_S^{\mathscr{I}_{FCS}}$). It is also interesting to note that the advantages of these additional data sources increase with the variability of the scenario: the difference between $\mathscr{P}_D^{LS}$ and $\mathscr{PR}_S^{\mathscr{I}_{FCS}}$ is 2.25 for $CV = 0.1$ and 33.74 for $CV =$

Table 1
Empirical results in small size instances for the $Fm|prmu, r_j|\sum F_j$ problem .

CV	n	m	$\mathscr{P}_D$	$\mathscr{P}_D^{LS}$	$\mathscr{P}_S$	$\mathscr{PR}_D^{\mathscr{I}_F}$	$\mathscr{PR}_D^{\mathscr{I}_{FS}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{I}_F}$	$\mathscr{PR}_S^{\mathscr{I}_{FS}}(\delta = 200)$
0.1	10	5	3.05	3.52	2.87	3.26	0.42	3.37	1.01
0.1	10	10	1.86	2.21	2.11	1.55	0.28	2.04	1.54
0.1	15	5	5.15	5.64	6.35	4.91	0.77	5.01	3.48
0.1	15	10	2.54	3.59	3.56	2.62	0.56	3.65	2.18
0.1	20	5	8.24	10.17	9.70	9.14	0.80	8.94	4.75
0.1	20	10	4.29	5.37	5.63	4.65	0.63	5.70	4.00
0.5	10	5	18.26	18.46	18.93	21.72	6.76	18.38	8.71
0.5	10	10	9.78	9.82	8.27	10.62	5.97	7.56	4.11
0.5	15	5	28.05	28.17	27.02	32.45	11.53	24.62	7.01
0.5	15	10	12.76	13.18	11.60	19.19	11.12	11.22	4.87
0.5	20	5	25.29	27.17	24.32	35.66	13.93	24.24	7.66
0.5	20	10	17.32	19.09	17.29	30.32	13.24	16.77	4.85
1	10	5	39.07	35.56	37.03	44.94	18.06	36.09	8.93
1	10	10	19.22	19.88	18.85	21.76	9.74	18.83	6.81
1	15	5	38.59	40.21	35.26	48.95	27.49	37.58	9.23
1	15	10	21.77	25.28	23.85	32.63	16.31	21.23	9.22
1	20	5	44.22	42.90	45.27	64.27	37.45	43.20	9.82
1	20	10	29.75	30.22	28.00	36.54	21.57	29.10	12.65
1.5	10	5	46.15	47.17	44.18	50.39	26.05	43.96	12.89
1.5	10	10	26.98	28.20	25.25	30.87	16.29	22.69	11.22
1.5	15	5	54.88	56.15	54.94	65.83	29.72	52.01	11.45
1.5	15	10	36.88	37.86	39.25	47.56	25.63	34.21	11.77
1.5	20	5	59.40	61.07	54.66	68.03	31.68	49.55	15.05
1.5	20	10	43.89	49.36	47.85	57.65	30.78	50.17	15.03
	ARPD		24.89	25.84	24.67	31.06	16.02	23.75	7.84

Table 2
Empirical results in small size instances for the $Fm|prmu, r_j|\sum T_j$ problem .

CV	n	m	$\mathscr{P}_D$	$\mathscr{P}_D^{LS}$	$\mathscr{P}_S$	$\mathscr{PR}_D^{\mathscr{S}^F}$	$\mathscr{PR}_D^{\mathscr{S}^{FC}}(\delta = 200)$	$\mathscr{PR}_D^{\mathscr{S}^{FS}}(\delta = 200)$	$\mathscr{PR}_D^{\mathscr{S}^{FCS}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{S}^F}$	$\mathscr{PR}_S^{\mathscr{S}^{FC}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{S}^{FS}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{S}^{FCS}}(\delta = 200)$
0.1	10	5	3.09	3.82	4.49	3.89	3.57	1.45	2.10	4.50	5.06	3.02	3.08
0.1	10	10	2.52	3.93	3.39	2.26	2.09	1.01	1.14	3.86	3.61	3.05	2.75
0.1	15	5	6.67	9.46	7.20	6.91	4.48	2.56	4.73	8.97	8.11	5.81	5.79
0.1	15	10	3.39	6.01	5.50	4.20	4.12	2.14	1.76	6.94	7.95	6.88	5.70
0.1	20	5	9.42	21.02	15.81	10.78	10.72	5.57	5.32	15.83	19.91	8.67	9.80
0.1	20	10	6.65	11.49	9.53	5.91	6.06	2.25	4.82	11.00	12.48	8.49	9.52
0.5	10	5	17.15	16.70	17.01	21.10	22.83	13.10	11.77	17.02	17.91	8.24	9.13
0.5	10	10	13.09	10.43	10.93	14.11	14.19	10.13	9.89	10.55	9.67	6.57	5.93
0.5	15	5	29.00	26.92	24.46	30.18	29.35	15.30	11.07	25.67	25.18	10.36	7.98
0.5	15	10	16.69	15.56	13.13	23.08	22.32	14.60	14.91	13.33	13.66	6.12	6.29
0.5	20	5	26.86	27.81	26.42	35.79	33.97	19.60	19.08	25.85	24.80	9.25	9.67
0.5	20	10	20.05	23.57	21.38	30.92	30.34	17.11	16.96	19.01	18.03	7.61	7.54
1	10	5	36.78	35.21	32.35	38.62	37.60	18.07	16.66	30.56	29.94	10.59	10.90
1	10	10	23.30	22.40	22.09	26.63	25.80	14.54	14.44	21.80	20.84	9.26	10.06
1	15	5	40.91	40.75	37.19	50.45	47.92	28.85	28.42	39.37	36.45	14.72	15.34
1	15	10	22.92	25.27	20.58	29.83	29.82	15.20	14.15	23.10	22.95	9.78	8.24
1	20	5	44.36	46.47	45.82	56.30	56.14	33.64	37.20	44.03	43.17	15.87	17.12
1	20	10	30.78	30.89	28.38	35.88	34.52	23.02	22.28	25.07	23.60	11.97	8.18
1.5	10	5	42.77	41.18	39.13	42.48	41.25	25.23	23.30	40.06	40.05	14.78	14.64
1.5	10	10	24.21	26.34	23.47	29.10	29.68	17.26	16.85	20.94	20.62	9.36	8.45
1.5	15	5	52.30	53.15	48.57	63.96	61.30	34.35	32.31	48.03	48.30	15.73	13.80
1.5	15	10	38.63	37.26	32.71	44.90	43.86	22.56	21.53	32.87	32.91	13.95	13.42
1.5	20	5	50.50	53.67	51.98	56.79	59.50	35.32	40.22	46.75	42.69	13.20	13.67
1.5	20	10	43.89	46.48	45.99	53.58	49.64	31.22	27.06	46.16	43.90	15.07	14.39
	ARDI		25.25	26.49	24.48	29.90	29.21	16.84	16.58	24.22	23.83	9.93	9.64

Table 3
Grouped computational results in small size instances for the $Fm|prmu, r_j|\sum F_j$ problem .

CV	$\mathscr{P}_D$	$\mathscr{P}_D^{LS}$	$\mathscr{P}_S$	$\mathscr{PR}_D^{\mathscr{S}^F}$	$\mathscr{PR}_D^{\mathscr{S}^{FS}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{S}^F}$	$\mathscr{PR}_S^{\mathscr{S}^{FS}}(\delta = 200)$
0.10	4.19	5.08	5.04	4.35	0.58	4.79	2.83
0.50	18.58	19.31	17.90	24.99	10.42	17.13	6.20
1.00	32.10	32.34	31.38	41.52	21.77	31.00	9.44
1.50	44.70	46.64	44.36	53.39	26.69	42.10	12.90
ARPD	24.89	25.84	24.67	31.06	14.86	23.75	7.84

Table 4
Grouped computational results in small size instances for the $Fm|prmu, r_j|\sum T_j$ problem .

CV	$\mathscr{P}_D$	$\mathscr{P}_D^{LS}$	$\mathscr{P}_S$	$\mathscr{PR}_D^{\mathscr{S}^F}$	$\mathscr{PR}_D^{\mathscr{S}^{FC}}(\delta = 200)$	$\mathscr{PR}_D^{\mathscr{S}^{FS}}(\delta = 200)$	$\mathscr{PR}_D^{\mathscr{S}^{FCS}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{S}^F}$	$\mathscr{PR}_S^{\mathscr{S}^{FC}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{S}^{FS}}(\delta = 200)$	$\mathscr{PR}_S^{\mathscr{S}^{FCS}}(\delta = 200)$
0.10	5.29	9.29	7.65	5.66	2.50	3.31	5.17	8.52	6.10	5.99	9.52
0.50	20.47	20.17	18.89	25.86	14.97	13.95	25.50	18.57	7.76	8.03	18.21
1.00	33.18	33.50	31.07	39.62	22.22	22.19	38.63	30.65	11.64	12.03	29.49
1.50	42.05	43.01	40.31	48.47	27.66	26.88	47.54	39.13	13.06	13.68	38.08
ARPD	25.25	26.49	24.48	29.90	29.21	16.84	16.58	24.22	23.83	9.93	9.64

1.5 for the total tardiness case (3.19 and 29.95 for total flowtime, respectively).

- Regarding shop floor status ($\mathscr{S}_F$), the use of real-time data from the shop to reschedule has almost no improvement in the solutions. In fact, the solution is globally worsened due to the high degree of nervousness provoked in the system. Thereby, the deterministic approach $\mathscr{P}_D$ is worsened from 24.89 to 31.06 by applying real-time rescheduling with only shop floor status $\mathscr{PR}_D^{\mathscr{S}^F}$ in the $Fm|prmu, r_j|\sum F_j$ problem and analogously from 25.25 to 29.90 in the $Fm|prmu, r_j|\sum F_j$ problem. In case of stochastic approaches, a rescheduling approach slightly improves the solution from an ARDI of 24.67 (using $\mathscr{P}_S$) to 23.75 (using $\mathscr{PR}_S^{\mathscr{S}^F}$) for the total tardiness minimisation (and from 24.48 to 24.22 for total flowtime). However this improvement is far to be statistical significance (a p-value of 0.729 is found testing the hypothesis $ARPD_{\mathscr{P}_S} = ARPD_{\mathscr{PR}_S^{\mathscr{S}^F}}$ using a non-parametric Mann-Whitney test for the total flowtime case). Thereby, these computational results show how the PR policy does not necessarily improve the solutions. Note that a similar conclusion is found by Schuh et al. [42] asking to 1300 participants, most of them related to mechanical and plant engineering, automotive industries or electrical equipment companies. Regarding previous computational studies, this trend has been found by Framinan et al. [12] for the $\mathscr{PR}_D^{\mathscr{S}^F}$ scenario and makespan objective, and our results show that for flowtime and tardiness the results are much worse. This behaviour can be explained using a simple example on a shop floor with only two machines. Originally, Job 1 should be finished in machine 1 at time 5 and in machine 2 at time 10. After that job, it was scheduled job 2 with processing times 5 and 6 in the first and second machines, respectively. However, job 1 is finished at time 3 in the first machine and time 13 in the second one. Using real-time data, it should be more interesting (for a deterministic rescheduling procedure minimising the total flowtime) to insert first job 3 (with processing times 10 and 5 in the first and second machines, respectively). Therefore, the flowtime for sequence (1,3,2) would be 53

Table 5
Procedure $\mathscr{PR}_D^{\mathscr{I}^{FC}}$ using different values of δ to solve the $Fm|prmu, r_j|\sum F_j$ problem .

CV	n	m	$\mathscr{PR}_D^{\mathscr{I}^{FS}}(\delta = 50)$	$\mathscr{PR}_D^{\mathscr{I}^{FS}}(\delta = 100)$	$\mathscr{PR}_D^{\mathscr{I}^{FS}}(\delta = 200)$	$\mathscr{PR}_D^{\mathscr{I}^{FS}}(\delta = 300)$
0.1	10	5	2.25	1.78	0.42	0.65
0.1	10	10	1.16	1.09	0.28	0.73
0.1	15	5	4.65	3.89	0.77	2.42
0.1	15	10	2.29	2.05	0.56	1.13
0.1	20	5	7.29	6.50	0.80	1.80
0.1	20	10	3.52	2.97	0.63	1.93
0.5	10	5	15.02	11.65	6.76	9.82
0.5	10	10	9.71	8.12	5.97	7.22
0.5	15	5	17.44	13.86	11.53	10.19
0.5	15	10	15.40	12.10	11.12	12.42
0.5	20	5	18.86	20.24	13.93	16.39
0.5	20	10	19.33	16.82	13.24	14.97
1	10	5	24.82	19.98	18.06	18.34
1	10	10	15.74	13.85	9.74	9.85
1	15	5	30.48	26.10	27.49	27.97
1	15	10	16.29	16.65	16.31	15.30
1	20	5	48.77	48.77	37.45	44.52
1	20	10	21.19	19.39	21.57	21.40
1.5	10	5	31.80	28.71	26.05	27.50
1.5	10	10	17.80	19.36	16.29	17.72
1.5	15	5	38.00	42.92	29.72	31.21
1.5	15	10	25.72	25.58	25.63	23.90
1.5	20	5	38.52	34.69	31.68	37.98
1.5	20	10	33.73	28.23	30.78	29.13
	ARPD		19.16	17.72	16.02	14.86

against 55 of sequence (1,2,3). However, sequence (1,3,2) would be much more sensible to any changes in processing times of job 3. First, any delay in job 3 would increase the completion time of the subsequent jobs by this amount. Secondly, since the standard deviation of the processing time of job 3 in machine 1 is higher than job 2 (the mean is higher and the coefficient of variation is constant), although sequence (1,2,3) worsens the solution, the latter is more robust as there are waiting times between jobs 1 and 2 to avoid forced idle times. Obviously, this nervousness in the system highly increases when release date are considered (following a similar reasoning) and when there are more jobs to be scheduled (as the probability of worsening the solutions increases). In this regard, we present in Table 6 a comparison of the non-stochastic procedures (i.e. $\mathscr{P}_D$, $\mathscr{PR}_D^{\mathscr{I}^F}$, $\mathscr{PR}_D^{\mathscr{I}^{FC}}(\delta = 200)$, $\mathscr{PR}_D^{\mathscr{I}^{FS}}(\delta = 200)$, and $\mathscr{PR}_D^{\mathscr{I}^{FCS}}(\delta = 200)$) for higher sizes of the instances (these procedures have been tested on Benchmark β_2). We can observe how an increase in the number of jobs or in the release data values worsens the deterministic predective-reactive scheduling, which cannot be avoided even for the integrated information scenario.

For the sake of brevity, preliminary analyses with the other rescheduling policies are not included in this paper. Nevertheless, it is worth mentioning that no statistically difference has been found, in both β_1 and β_2, comparing the considered periodic rescheduling strategy and an event-driven strategy triggered each time that a job is finished in the first machine (12) for total flowtime minimisation, which achieves e.g. in β_1 an ARPD of 30.49 versus 31.06 finding by $\mathscr{PR}_S^{\mathscr{I}^F}$ (the hypothesis that both rescheduling procedures obtain the same ARPD cannot be rejected finding a p-value of 0.888 using a Mann-Whitney test).

6. Conclusions

In this paper, we explore the potential advantages of information integration (both real-time shop floor status and advanced upstream/downstream processes data) that the technological advances of Industry 4.0 make available. More specifically, we want to assess how such integration can improve shop floor performance when it is used as input for the scheduling decision-making process, depending on the data sources available, and on the approach and solution procedure adopted for scheduling. To do so, we simulate a common shop floor layout (i.e. the flowshop) and define different scenarios formed by a combination of (re)scheduling approaches, solution procedures and objective functions, and different data sources.

Regarding the solution procedures, the best performance has been obtained by the stochastic procedures, both in the predictive and in the predictive-reactive approaches, improving the deterministic ones. However, these improvements in the quality of the solutions must be balanced with the extremely high computational effort required by the stochastic procedures as compared to the deterministic ones. In this regard, the deterministic procedures are very efficient for the predictive approach if a balance between the quality of the solutions and the computational effort is sought. However, when embedded in a predictive-reactive approach (without considering advanced information from upstream/downstream processes), it has been found that, in most cases, these deterministic procedures induce nervousness in the system and that this translates in a poor quality of the solutions. Only for instances with a very low coefficient of variation, some solution improvement can be found. Regarding the importance of the different data sources, the experimentation highlights the importance of advance upstream information rather than using shop floor data, as the quality of the solutions for both objective functions (ARPD and ARDI) has been reduced up to 50% as compared to the same scenario without upstream

Table 6
Computational results of deterministic procedures on Benchmark β_2.

		$Fm\|prmu, r_j\|\sum F_j$			$Fm\|prmu, r_j\|\sum T_j$				
		$\mathscr{P}_D$	$\mathscr{PR}_D^{\mathscr{I}^F}$	$\mathscr{PR}_D^{\mathscr{I}^{FCS}}(\delta = 200)$	$\mathscr{P}_D$	$\mathscr{PR}_D^{\mathscr{I}^F}$	$\mathscr{PR}_D^{\mathscr{I}^{FCS}}(\delta = 200)$	$\mathscr{PR}_D^{\mathscr{I}^{FC}}(\delta = 200)$	$\mathscr{PR}_D^{\mathscr{I}^{FS}}(\delta = 200)$
CV	0.1	9.82	22.26	10.10	38.86	60.27	29.06	58.99	31.73
	0.5	13.52	42.17	26.24	36.13	71.67	43.06	66.81	48.11
	1	21.56	46.42	24.78	41.28	69.33	38.65	66.80	42.58
	1.5	27.29	47.92	24.95	44.89	68.78	38.71	63.97	42.14
n	25	25.37	34.67	13.69	53.42	66.26	30.96	62.22	34.67
	50	16.96	39.45	21.73	38.77	68.76	37.39	63.28	41.71
	75	11.81	44.96	29.12	28.66	67.51	43.76	66.92	47.05
m	2	25.01	56.36	26.01	41.49	76.19	37.46	68.84	44.73
	5	19.23	40.59	23.08	41.83	67.62	36.40	65.11	41.98
	10	14.87	32.87	18.50	40.45	64.11	37.00	63.12	37.19
	15	12.60	27.93	18.13	37.23	61.61	38.66	59.17	40.41
α	1	7.93	8.73	6.78	51.22	45.17	28.05	34.61	39.28
	2	18.12	41.00	23.04	38.41	70.52	38.32	69.65	40.38
	3	18.65	49.39	27.86	32.38	73.16	44.08	72.81	43.55
	4	27.49	59.67	28.38	39.13	81.20	39.04	79.51	41.36
ARPD		18.05	39.69	21.51	40.29	67.51	37.37	64.14	41.14

and downstream information. In contrast, the effect of downstream information hardly has some incidence in the quality of the solutions.

As future research lines, the results highlight the importance of making proper choices regarding rescheduling approaches and solution procedures to take advantage of the additional data provided by Industry 4.0. Perhaps one way to do so (at least for the deterministic procedures) is to use additional criteria to provide robust solutions. In this regard, e.g. the maximisation of waiting time could provide some time to reduce the nervousness of the system. This provides an interesting research avenue within the domain of deterministic solution procedures, which seems to cope quite well with the system uncertainty in the predictive approach. Finally, it has to be noted that the best overall solutions have been found by integrating and making available all data sources for a stochastic solution procedure embedded in a predictive-reactive approach. Despite the procedure is heavily time-consuming and therefore cannot be applicable with nowadays computers, it points out towards the need of speeding up these procedures to make them feasible in the future.

CRediT authorship contribution statement

Victor Fernandez-Viagas: Investigation, Software, Formal analysis, Writing – original draft, Visualization, Methodology, Data curation, Conceptualization, Validation. **Jose M. Framinan:** Validation, Writing – original draft, Writing – review & editing, Supervision, Visualization, Resources, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors wish to thank the referees for their comments on the earlier versions of the manuscript. This research has been funded by the Spanish Ministry of Science and Innovation, under the project “ASSORT” with reference PID2019-108756RB-I00, and by the Junta de Andalucia under the projects “DEMAND”, “IBSOS and “EFECTOS”, with references P18-FR-1149, 5835 and US-1264511, respectively.

References

[1] H. Aytug, M.A. Lawley, K. McKay, S. Mohan, R. Uzsoy, Executing production schedules in the face of uncertainties: a review and some future directions, Eur J Oper Res 161 (1) (2005) 86–110.
[2] D. Bai, J. Liang, B. Liu, M. Tang, Z.H. Zhang, Permutation flow shop scheduling problem to minimize nonlinear objective function with release dates, Computers and Industrial Engineering 112 (2017) 336–347, https://doi.org/10.1016/j.cie.2017.08.031.
[3] K.R. Baker, D. Altheimer, Heuristic solution methods for the stochastic flow shop problem, Eur J Oper Res 216 (1) (2012) 172–177, https://doi.org/10.1016/j.ejor.2011.07.021.
[4] Y. Chen, Industrial information integration - a literature review 2006 - 2015, Journal of Industrial Information Integration 2 (2016) 30–64.
[5] L.K. Church, R. Uzsoy, Analysis of periodic and event-driven rescheduling policies in dynamic shops, Int. J. Computer Integr. Manuf. 5 (3) (1992) 153–163.
[6] P. Cowling, M. Johansson, Using real time information for effective dynamic scheduling, Eur J Oper Res 139 (2) (2002) 230–244.
[7] N.K. Dev, R. Shankar, S. Swami, Diffusion of green products in industry 4.0: reverse logistics issues during design of inventory and production planning system, Int. J. Prod. Econ. 223 (2020), 107519.
[8] X. Dong, P. Chen, H. Huang, M. Nowak, A multi-restart iterated local search algorithm for the permutation flow shop problem minimizing total flow time, Computers & Operations Research 40 (2) (2013) 627–632.
[9] S. Echchakoui, N. Barka, Industry 4.0 and its impact in plastics industry: aliterature review, Journal of Industrial Information Integration 20 (2020), 100172, https://doi.org/10.1016/j.jii.2020.100172.
[10] V. Fernandez-Viagas, J.M. Framinan, A beam-search-based constructive heuristic for the PFSP to minimise total flowtime, Computers and Operations Research 81 (2017) 167–177, https://doi.org/10.1016/j.cor.2016.12.020.
[11] V. Fernandez-Viagas, J.M.S. Valente, J.M. Framinan, Iterated-greedy-based algorithms with beam search initialization for the permutation flowshop to minimise total tardiness, Expert Syst Appl 94 (2018) 58–69, https://doi.org/10.1016/j.eswa.2017.10.050.
[12] J.M. Framinan, V. Fernandez-Viagas, P. Perez-Gonzalez, Using real-time information to reschedule jobs in a flowshop with variable processing times, Computers and Industrial Engineering 129 (2019) 113–125, https://doi.org/10.1016/j.cie.2019.01.036.
[13] J.M. Framinan, R. Leisten, Linking scheduling criteria to shop floor performance in permutation flowshops, Algorithms 12 (2) (2019), 263, https://doi.org/10.3390/A12120263.
[14] J.M. Framinan, P. Perez-Gonzalez, On heuristic solutions for the stochastic flowshop scheduling problem, Eur J Oper Res 246 (2) (2015) 413–420, https://doi.org/10.1016/j.ejor.2015.05.006.
[15] J.M. Framinan, R. Leisten, C. Rajendran, Different initial sequences for the heuristic of Nawaz, Enscore and Ham to minimize makespan, idletime or flowtime in the static permutation flowshop sequencing problem, Int. J. Prod. Res. 41 (1) (2003) 121–148.
[16] A. Frankó, G. Vida, P. Varga, Reliable identification schemes for asset and production tracking in industry 4.0, Sensors (Switzerland) 20 (13) (2020) 1–24, https://doi.org/10.3390/s20133709.3709
[17] Y. Fu, J. Ding, H. Wang, J. Wang, Two-objective stochastic flow-shop scheduling with deteriorating and learning effect in industry 4.0-based manufacturing system, Applied Soft Computing Journal 68 (2018) 847–855.
[18] L.F. Gelders, N. Sambandam, Four simple heuristics for scheduling a flow-shop, Int. J. Prod. Res. 16 (3) (1978) 221–231.
[19] M. Ghaleb, H. Zolfagharinia, S. Taghipour, Real-time production scheduling in the industry-4.0 context: addressing uncertainties in job arrivals and machine breakdowns, Computers and Operations Research 123 (2020), 105031.
[20] N.G. Hall, M.E. Posner, Generating experimental data for computational testing with machine scheduling applications, Oper Res 49 (6) (2001) 854–865, https://doi.org/10.1287/opre.49.6.854.10014.
[21] A.M.A. Hariri, C.N. Potts, An algorithm for single machine sequencing with release dates to minimize total weighted completion time, Discrete Appl. Math. 5 (1) (1983) 99–109, https://doi.org/10.1016/0166-218X(83)90019-7.
[22] S. Hasija, C. Rajendran, Scheduling in flowshops to minimize total tardiness of jobs, Int. J. Prod. Res. 42 (11) (2004) 2289–2301.
[23] S. Hatami, L. Calvet, V. Fernandez-Viagas, J.M. Framinan, A.A. Juan, A simheuristic algorithm to set up starting times in the stochastic parallel flowshop problem, Simul. Modell. Pract. Theory 86 (2018) 55–71.
[24] W.J. Hopp, M.L. Spearman, Factory physics, McGraw-Hill/Irwin, Boston, Massachusetts, 2000.
[25] K. Hozak, J.A. Hill, Issues and opportunities regarding replanning and rescheduling frequencies, Int. J. Prod. Res. 47 (18) (2009) 4955–4970.
[26] K. Karabulut, A hybrid iterated greedy algorithm for total tardiness minimization in permutation flowshops, Computers and Industrial Engineering 98 (2016) 300–307, https://doi.org/10.1016/j.cie.2016.06.012.
[27] Y.D. Kim, Heuristics for flowshop scheduling problems minimizing mean tardiness, Journal of the Operational Research Society 44 (1) (1993) 19–28.
[28] G.M. Komaki, V. Kayvanfar, Grey wolf optimizer algorithm for the two-stage assembly flow shop scheduling problem with release time, J Comput Sci 8 (2015) 109–120, https://doi.org/10.1016/j.jocs.2015.03.011.
[29] R. Larsen, M. Pranzo, A framework for dynamic rescheduling problems, Int. J. Prod. Res. 57 (1) (2019) 16–33.
[30] N. Li, L. Zhao, C. Bao, G. Gong, X. Song, C. Tian, A real-time information integration framework for multidisciplinary coupling of complex aircrafts: an application of IIIE, Journal of Industrial Information Integration 22 (2021), 100203.
[31] X. Li, W. Zhang, X. Zhao, W. Pu, P. Chen, F. Liu, Wartime industrial logistics information integration: framework and application in optimizing deployment and formation of military logistics platforms, Journal of Industrial Information Integration 22 (2021), 100201.
[32] Y. Lin, H. Wang, J. Li, H. Gao, Data source selection for information integration in big data era, Inf Sci (Ny) 479 (2019) 197–213.
[33] Y. Lu, Industry 4.0: a survey on technologies, applications and open research issues, Journal of Industrial Information Integration 6 (2017) 1–10.
[34] M. Mrad, S. Chalghoumi, T. Ladhari, A. Gharbi, Enhanced lower bounds and exact procedures for total completion time minimization in a two-machine permutation flowshop with release dates, International Transactions in Operational Research 26 (6) (2019) 2432–2449, https://doi.org/10.1111/itor.12421.
[35] F. Niu, Y. Chen, Industrial information integration in track allocation optimization in high-speed train stations, Journal of Industrial Information Integration 21 (2021), 100193.
[36] M. Parente, G. Figueira, P. Amorim, A. Marques, Production scheduling in the context of industry 4.0: review and trends, Int. J. Prod. Res. (2020).
[37] C. Pascal, D. Panescu, On rescheduling in holonic manufacturing systems, Comput. Ind. 104 (2019) 34–46.
[38] C.N. Potts, L.N. Van Wassenhove, A decomposition algorithm for the single machine total tardiness problem, Operations Research Letters 1 (5) (1982) 177–181.
[39] D.A. Rossit, F. Tohmé, M. Frutos, Production planning and scheduling in cyber-physical production systems: a review, Int. J. Computer Integr. Manuf. 32 (4–5) (2019) 385–395.
[40] D.A. Rossit, F. Tohmé, M. Frutos, A data-driven scheduling approach to smart manufacturing, Journal of Industrial Information Integration 15 (2019) 69–79, https://doi.org/10.1016/j.jii.2019.04.003.

[41] I. Sabuncuoglu, M. Bayiz, Analysis of reactive scheduling problems in a job shop environment, Eur J Oper Res 126 (3) (2000) 567–586.
[42] G. Schuh, J.-P. Prote, M. Luckert, P. Hünnekes, M. Schmidhuber, Effects of the update frequency of production plans on the logistical performance of production planning and control, Procedia CIRP 79 (2019) 421–426, https://doi.org/10.1016/j.procir.2019.02.115.
[43] S. Sheikh, G.M. Komaki, V. Kayvanfar, Multi objective two-stage assembly flow shop with release time, Computers and Industrial Engineering 124 (2018) 276–292, https://doi.org/10.1016/j.cie.2018.07.023.
[44] S. Sheikh, G.M. Komaki, V. Kayvanfar, E. Teymourian, Multi-stage assembly flow shop with setup time and release time, Oper. Res. Perspect. 6 (2019), 100111, https://doi.org/10.1016/j.orp.2019.100111.
[45] E. Taillard, Benchmarks for basic scheduling problems, Eur J Oper Res 64 (2) (1993) 278–285.
[46] M. Thürer, M. Stevenson, M.J. Land, L.D. Fredendall, On the combined effect of due date setting, order release, and output control: an assessment by simulation 57 (6) (2019) 1741–1755.
[47] E. Vallada, R. Ruiz, G. Minella, Minimising total tardiness in the m-machine flowshop problem: a review and evaluation of heuristics and metaheuristics, Computers & Operations Research 35 (4) (2008) 1350–1373.
[48] G.E. Vieira, J.W. Herrmann, E. Lin, Rescheduling manufacturing systems: a framework of strategies, policies, and methods, Journal of Scheduling 6 (1) (2003) 39–62.
[49] B. Waschneck, T. Altenmüller, T. Bauernhansl, A. Kyek, Production scheduling in complex job shops from an industrie 4.0 perspective: A review and challenges in the semiconductor industry. CEUR Workshop Proceedings, 2017.
[50] L.D. Xu, E.L. Xu, L. Li, Industry 4.0: state of the art and future trends, Int. J. Prod. Res. 56 (8) (2018) 2941–2962, https://doi.org/10.1080/00207543.2018.1444806.
[51] S. Zdrzaka, J. Grabowski, An algorithm for single machine sequencing with release dates to minimize maximum cost, Discrete Appl. Math. 23 (1) (1989) 73–89, https://doi.org/10.1016/0166-218X(89)90036-X.