Logist. Res. (2016) 9:3
DOI 10.1007/s12159-015-0130-7

ORIGINAL PAPER

A mobile application for knowledge-enriched short-term scheduling of complex products

D. Mourtzis[1] · M. Doukas[1] · E. Vlachou[1]

Received: 8 June 2015 / Accepted: 21 December 2015 / Published online: 8 January 2016

Abstract The ever-increasing product complexity, especially for the case of engineer-to-order products, highly affects the performance of manufacturing systems. Therefore, a high degree of flexibility is needed during daily decision-making activities, such as production scheduling. For addressing this challenge, this research work proposes a knowledge-enriched short-term job-shop scheduling mechanism, which is implemented into a mobile application. More precisely, it focuses on the short-term scheduling of the resources of the machine shop, through an intelligent algorithm that generates and evaluates alternative assignments of resources to tasks. Based on the requirements of a new order, a similarity mechanism retrieves successfully executed past orders together with a dataset that includes the processing times, the job and task sequence, and the suitable resources. In addition to that, the similarity mechanism is used to calculate the due-date assignments of the orders based on the knowledge stored in past cases. Afterwards, it adapts these parameters to the requirements of the new order so as to evaluate the alternative schedules and identify a good alternative in a timely manner. The deriving schedule can be presented on mobile devices, and it can be manipulated by the planner on-the-fly respecting tasks precedence constraints and machine availability. A case study from the mould-making industry is used for validating the proposed method and application.

Keywords Manufacturing systems · Scheduling · Mobile applications

1 Introduction

Modern manufacturing relies on the reuse of previous knowledge that is either contained in data repositories and IT systems or exists in the form of tacit human knowledge. Knowledge constitutes a key factor for improving manufacturing performance, during design, planning, and operational phases [1]. Most contemporary manufacturing industries acknowledge that the exploitation of the existing knowledge is necessary to enhance the performance of manufacturing [2]. Indicatively, the importance of knowledge reuse for a system's design and planning phase is evident, as rough estimations indicate that more than 20 % of an engineer's time is spent on searching and absorbing information for a new project [3].

A particular type of manufacturing system, which essentially relies on the knowledge and expertise of human assets to improve its performance, is a job-shop that produces engineer-to-order (ETO) products. Usually, the incorporation of new orders in the schedules of such systems is performed empirically and using rules of thumb, even when the system operates near its maximum capacity. However, with the rising complexity of production requirements and the increased penetration of IT systems in manufacturing, knowledge reuse is an enabler to reduce the product development cycle and increase manufacturing performance. On the contrary, in current practice, this valuable knowledge generated and associated with

This article is part of a focus collection on "Robust Manufacturing Control: Robustness and Resilience in Global Manufacturing Networks".

✉ D. Mourtzis
mourtzis@lms.mech.upatras.gr

[1] Laboratory for Manufacturing Systems and Automation, Department of Mechanical Engineering and Aeronautics, University of Patras, 26500 Rio, Greece

products and processes in a daily basis remains tacit and its reusability is confined to a specific operator or planner [4].

Further to that, another requirement today is pervasive access to information and decision-making. One enabling technology to achieve ubiquitous access to knowledge and assist decision-making is mobile apps, i.e. applications developed for mobile devices. The general market of apps is expected to reach revenues of $70B by 2017 [5]. More and more companies are starting to base their business on mobility; however, the adoption of apps in the manufacturing domain is yet at a primary stage [6].

Motivated by the above, in this research work, a scheduling method that is enhanced with an integrated knowledge reuse mechanism is proposed. The knowledge reuse mechanism retrieves historical scheduling cases and through the case-based reasoning (CBR) methodology extracts information related to the modelling of the scheduling workload. The deriving workload model includes necessary input for a scheduler, such as the job structure and the task breakdown, the precedence constraints, and the processing and set-up times. Alternative schedules are generated and are evaluated using multiple conflicting criteria, such as flowtime and tardiness. The scheduling is performed using an intelligent search algorithm (ISA) with three tuneable parameters, which are adjusted through a parametric investigation, using a statistical design of experiments method. All functionalities are exposed through a developed mobile app.

The remainder of the paper is structured as follows. Section 2 includes a literature survey on knowledge-enriched scheduling (KES) applications. Section 3 analyses the proposed methodology. Section 4 describes the design of the scheduling app. Section 5 demonstrates a real-life case study in a mould-making industry. Section 6 draws the conclusions and describes the future work directions.

2 State of the art

Following the main topics of the research work, this section discusses knowledge reuse in manufacturing, knowledge-enriched scheduling methods, mobile apps in manufacturing, and lead time estimation approaches.

Throughout the years, several methods have been proposed for knowledge reuse in the manufacturing domain with the aim to support designers and engineers in decisions related to modelling, design, prediction, monitoring, and optimisation. Knowledge reuse is considered to have a major impact on several manufacturing domains, offering productivity gains [7]. There are two main ways to reuse past knowledge: reuse the past case solution and reuse the past method that constructed the solution [8]. A widely used artificial intelligence (AI) method that can effectively enable reuse of past solutions is CBR, which retrieves past experience to reuse for a target problem; of course, the solutions of past cases may need to be revised for applying in the new case. The successful problem-solving experience is then retained for further reuse [9]. CBR utilises similarity mechanisms in order to compare past cases stored in the repository and the target case, i.e. a new product, based on case's features or attributes [10]. The most similar cases to the new one are recalled in order to provide recommendations [11]. Li et al. 2011 presented CBR as a tool for knowledge management during product development and reported that reuse of past product knowledge can be used to improve the problem-solving capabilities [12]. The CBR method is utilised in this research work due to its suitability for complex ill-defined concepts, with unstructured knowledge, and because case generalisation is necessary [9].

The second area of interest in this research work is job-shop scheduling. Scheduling is one of the most critical issues for a manufacturing enterprise. In most SMEs, who cannot afford costly investments in IT software solutions, scheduling is carried out empirically. However, the definition of a performing solution is quite difficult, depending on the job-shop environment, process constrains, and performance indicators [13]. Numerous approaches have been reported for the modelling and solving of the job-shop scheduling problem [14]. Wang et al. [15] proposed the development of an application using a genetic algorithm including a chromosome representation in seven different machines of a job floor that enables a dynamic job-shop scheduling within complex production systems. Chrysolouris et al. [39], considering the limitations of static scheduling, proposed a dynamic scheduling problem to accurately reflect a real job-shop scheduling environment. This dynamic nature of these scheduling problems [16] constitutes KES approaches essential, as knowledge reuse could assist in incorporating and scheduling new jobs in an ever-changing environment. However, literature findings that focus on knowledge reuse as an enabler for improving scheduling performance are scarce. Motivated by empirical knowledge, [17] proposes an efficient search method for the multi-objective flexible job-shop scheduling in order to reach high automation levels towards generating optimal or near-optimal production schedules. Another study exploiting previous knowledge proposed a data mining technique for discovering dispatching rules that improve scheduling performance [18]. The job-shop scheduling problem has been addressed using a knowledge-enriched genetic algorithm in [19]. The idea was to imbue production system knowledge during the formulation of the initial population of the algorithm with the potential of faster and better convergence. The authors in [20] utilise data mining for optimising a basic aspect of production scheduling, i.e.

the assignment of due dates to orders dispatched in a dynamic job-shop. Moreover, a knowledge-based algorithm for flexible job-shop scheduling is presented in [21]. The authors in this approach combine the variable neighbourhood search with a knowledge management module in order to reach the optimum solution more efficiently. In addition to these techniques, group technology could be also utilised in order to minimise the makespan and the sum of the completion times of a generated schedule based on position-dependent learning effects [22]. Concluding, according to [23], research should be shifted more towards knowledge reuse for decision support tools, within safety, reliability, and maintainability. Several of the above-mentioned studies utilise knowledge reuse techniques to improve scheduling performance. However, most of them focus on reusing knowledge to define the tuneable parameters of the algorithm that perform and determine scheduling attributes including dispatching rules or ordering details without offering actual reuse of meaningful scheduling data [22]. There is little existing work on knowledge reuse for determining the processing times of new tasks or the due dates of new jobs. Nevertheless, this can lead to the realisation of accurate and quick dynamic scheduling.

The third area of interest is mobile technology. Mobile technology evolves rapidly; in the last decade, the use of mobile apps has outpaced traditional PC-based web-browsing [24]. The usage of apps doubled on average over the last 2 years, with utility and productivity apps ranking second in growth [25]. The necessary components of apps in order for them to be fully leveraged in manufacturing are presented in [26], where architecture, development, infrastructure, security, portfolio, and privacy issues are investigated. Estimations speak of apps boosting productivity by 5–10 % [27]. The growth of mobility and mobile apps is highly influenced by the growth and the adoption of the cloud technology in manufacturing [28]. Cloud technology acts as an enabler to adopt mobile devices in manufacturing not only for the provision of applications but also for production data management purposes [29]. However, the adoption of apps focused on core manufacturing processes was up to now limited [25]. Cloud and mobile technologies are not sufficiently adopted in manufacturing systems yet, despite the productivity boost that they can offer [28, 30]. Nevertheless, apps are finding their way into activities such as manufacturing network design [31] and other scientific domains [32, 33].

The last area of interest is the manufacturing lead time estimation. Based on the literature, the most robust methods for lead time estimation are AI methods [34]. Ozturk et al. [34] used data mining as an AI method and attribute tables in order to calculate manufacturing lead time. Among AI methods, CBR, which focuses on solving problems by adapting acceptable solutions and comparing differences and similarities between previous and current products, has been utilised for lead time estimation. An approach based on CBR was applied during product development in [35], and it effectively reduced lead time and improved the problem-solving capabilities. The literature review makes apparent that the CBR and the data mining techniques should be further considered for supporting decision supports tools [22], especially in determining machining sequence and processing times.

Building upon the literature on the field, the combination of knowledge reuse techniques together with intelligent scheduling algorithms under the umbrella of mobile and cloud technology is considered as a necessary step towards the next generation of decision support tools. Addressing these challenges, the proposed research work provides a methodology that utilises a knowledge reuse mechanism for extracting manufacturing information related to machining sequence and orders due dates in order to support a short-term scheduling application. Moreover, the scheduling mechanism is developed into an app, motivated by the fact that the adoption of the mobile devices in manufacturing, and specifically in decision-making activities, can lead to easier access to information, as well as quick and accurate visualisation and interaction with the generated scheduling and planning information [6, 30, 31]. Finally, the proposed methodology is applied in a real manufacturing environment utilising data from a mould-making SME and the scheduling algorithm is compared with others dispatching rules in order to benchmark its performance. This work extends the research presented in [36–38] by enhancing the scheduling algorithm with knowledge reuse capabilities and by verifying the method in a case coming from the domain of ETO products.

3 The knowledge-enriched scheduling method

The knowledge-enriched scheduling (KES) engine consists of two mechanisms, namely: (1) the knowledge extraction and reuse mechanism and (2) the short-term scheduling mechanism (Fig. 1). Regarding the workflow of the first mechanism, once a new order enters the system, a breakdown of the product components into a bill of materials (BoM) structure is performed. The product is characterised by a number of attributes (product features) that are used by the similarity mechanism of CBR for a pairwise attribute comparison. The result of the similarity comparison is an ordered list that contains the past cases ranked from the most to the least similar. By reusing the knowledge stored in these past cases, the expert planner is allowed to extract valuable information that helps introduce the new order into the production system with the needed adaptations.

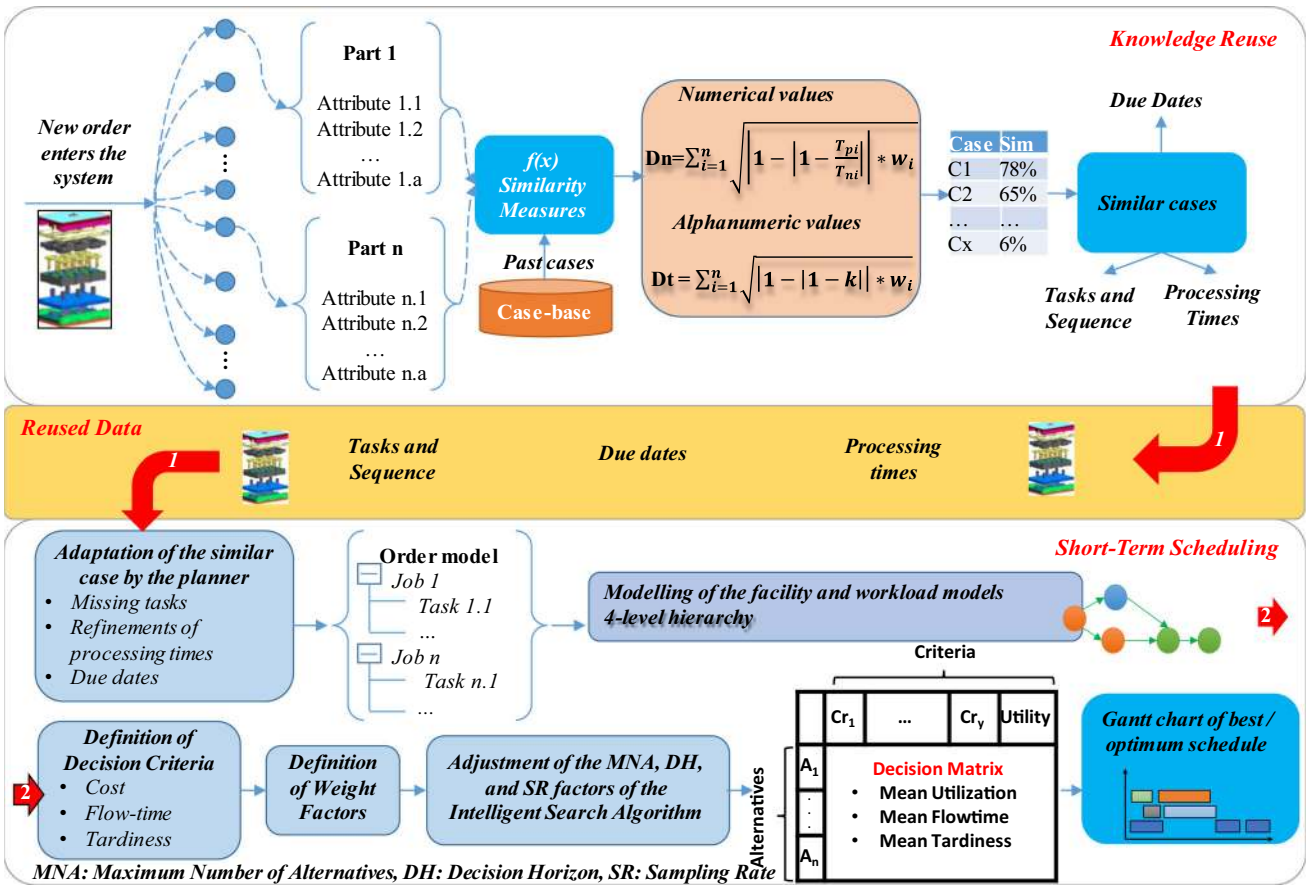

Fig. 1 Workflow of the knowledge-enriched short-term scheduling (KES) method. The method consists of two mechanisms, namely: (1) the knowledge extraction and reuse mechanism and (2) the short-term scheduling mechanism

The reusable information includes the required number and type of jobs, the number of tasks for each job and their precedence constraints, the processing times for each task in specific machines, and finally the due dates of the tasks. The output of this process comprises the necessary input for a scheduling engine. It is noted that specific process planning information, such as cutter selection, process parameters, and fixture specification, is beyond the direct scope of the proposed work.

The latter component of the KES is the short-term scheduling mechanism. After the identification of the most similar cases, the expert planner aggregates information that can be reused in the new case and adapts it. The adaptation is required in order to compensate for missing tasks that were not identified during the similarity measurement, or in order to imbue to the dataset the actual situation of the shop-floor (machine breakdowns and availability). The result of the adaptation is the workload and the facility models. These models are imported into the intelligent scheduling engine. The planner defines the decision-making criteria and their weight factors, which reflect the design and planning objectives of the company. Following on that, the definition of the tuneable parameters of the scheduling algorithm is defined. The tuneable parameters are the maximum number of alternatives (MNA), the decision horizon (DH), and the sampling rate (SR). The description of the function of these parameters is provided in Sect. 3.2. The scheduling algorithm generates alternative schedules, selects a good alternative in a timely manner, and displays it in the form of a Gantt chart.

The two mechanisms are designed in a modular way. The scheduling engine is capable to function without input provided by the knowledge mechanism, if the latter is not available. Similarly, the knowledge mechanism is decoupled from the scheduling engine and can be used for extracting manufacturing information for different purposes, such as for the estimation of the delivery time of an injection mould [38].

3.1 Modelling of the facility and the workload

The production facility is hierarchically divided into job-shops that contain work-centres, which in turn contain a number of resources. The latter are individual processors

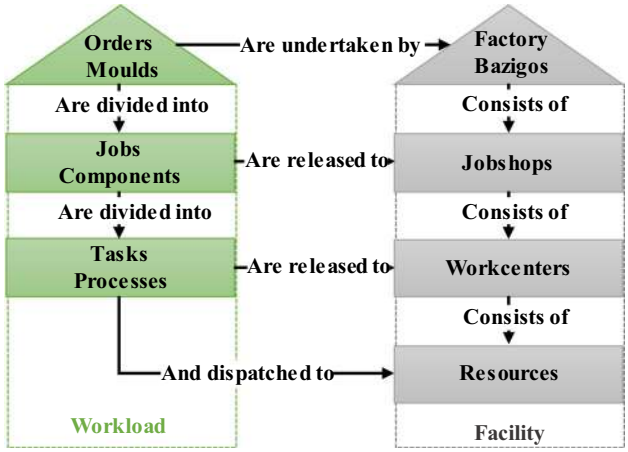

Fig. 2 Four-level hierarchical workload and facility model

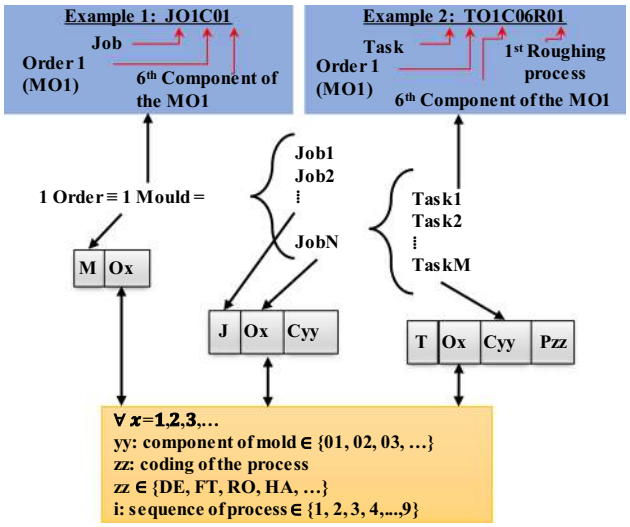

Fig. 3 Modelling of the mould order, jobs, and tasks

(resources with diverse processing capabilities (machining technology, cycle times, investment costs, fixed operating costs, etc.)). Similarly, the workload model includes orders that are broken down into jobs, each containing a number of tasks that are processed by the resources. As Fig. 2 shows, orders are dispatched to the facility, jobs to jobshops, and tasks to work-centres' resources. Resources are not parallel processors, and their availability is subject to the system workload. The release of tasks considers finite capacity, precedence relations, and availability constraints.

The job and task modelling is shown in Fig. 3. Specifically, each mould consists of a number of components. Each mould is represented as an order, and each component is represented as a job. Each job is composed of a number of tasks that need to be performed in order to manufacture each component of an order. Finally, the generation, evaluation, and selection of task to resource assignment are performed by an intelligent multi-criteria search algorithm (ISA) as described in [39]. ISA evaluates the alternatives in a decision matrix based on set-up cost and processing time criteria. A utility function is used for ranking the alternatives and for selecting the highest performing one.

3.2 Description of the knowledge reuse mechanism

The first step in the workflow of the knowledge reuse mechanism is the comparison of the new order against past cases in order to identify similar cases and reuse their data. This similarity measurement emphasises on the differences exhibited between the basic attributes that characterise old and new orders alike. CBR is functioning on the premise that a comparison between cases is feasible. To accomplish that, a set of attributes of the product that enters the system as a new order is selected, to characterise the case. For every new ETO product, the engineers together with the customer complete a web-based form with predefined fields that comprise the product's specifications. The attributes used to characterise the ETO product cases include: type of product, geometry stacks, slides, type of hardening, core cap, stacks material, profile rings, data provided, polishing, side of injection, tamper evident, gating type, way of ejection, ejector rings type, no of cavities, and wall thickness, depth, width, diameter, and length.

The past cases are retrieved using the CBR methodology and are compared with similarity mechanisms. The type of attributes considered takes both numeric and alphanumeric values. The alphanumeric attributes are mapped into discrete values represented by numbers in [0–1] for normalisation reasons. Moreover, both attribute types are multiplied with weight factors, considering their influence on the actual similarity between cases. Equations (1) and (2) are used for measuring the Euclidean distance through a pairwise comparison between the attributes of past and new cases. Equation (3) aggregates the results of the two distance metrics.

$$D_n = \sum_{i=1}^{n} \sqrt{\left|1 - \left|1 - \frac{T_{pi}}{T_{ni}}\right| \times w_i\right|}, \quad \text{numerical values} \qquad (1)$$

$$D_t = \sum_{i=1}^{n} \sqrt{|1 - |1 - k|| \times w_i}, \quad \text{alphanumeric values} \qquad (2)$$

$$S = (D_n + D_t)^2 \qquad (3)$$

where D_n numerical distance, D_t text distance, n number of attributes, T_{ni} ith attribute of the new case n, T_{pi} ith attribute of the past case p, k mapping for alphanumeric attributes, and w_i the weight of attributes.

The past case with the highest similarity index is analysed first (Fig. 4). The planner may retrieve the process sequence, precedence constraints, the components, and the resources used in the past case, as well as processing and set-up times per task and resource. Moreover, based on their experience, expert planners have the capability to recognise whether the retrieved data are adequate to describe the new order. In case they are insufficient, the planner adapts the dataset to the requirements of the new case. Furthermore, in cases when the new product requires a different amount of components/processes than the retrieved most similar past case, then the second most similar case can be consulted, then the third, and so on. Either way, the similarity index between two cases must always remain above the threshold of 60 %, which is calculated based on historical observations; otherwise, the retrieved information would be misleading. Indicatively, if the best match in terms of similarity index is fairly aged in comparison with the new case, it is probable that adaptations would be required to compensate for changes in the shop-floor, such as the addition of new manufacturing resources and technologies. In this case, engineers are aware of the current state of the shop-floor and can replace the old resources with the new similar ones in the new process plan. Having decided the matching past similar cases, the task sequences are retrieved, the availability of the machines is confirmed, and then, the final combination of the new sequence of processes and components is settled.

3.3 Description of the short-term scheduling mechanism

Schedules are constructed on the basis of events occurring sequentially through time. Thus, the next scheduling decision is identified by moving along the time horizon until an event (release of a new order in the system or the completion of a task) is scheduled to occur that will initiate a change in the status of the system [40]. The set of pending tasks becomes eligible for release at the time a resource becomes available. The operational policy behind the assignment of a task to a specific resource can be either

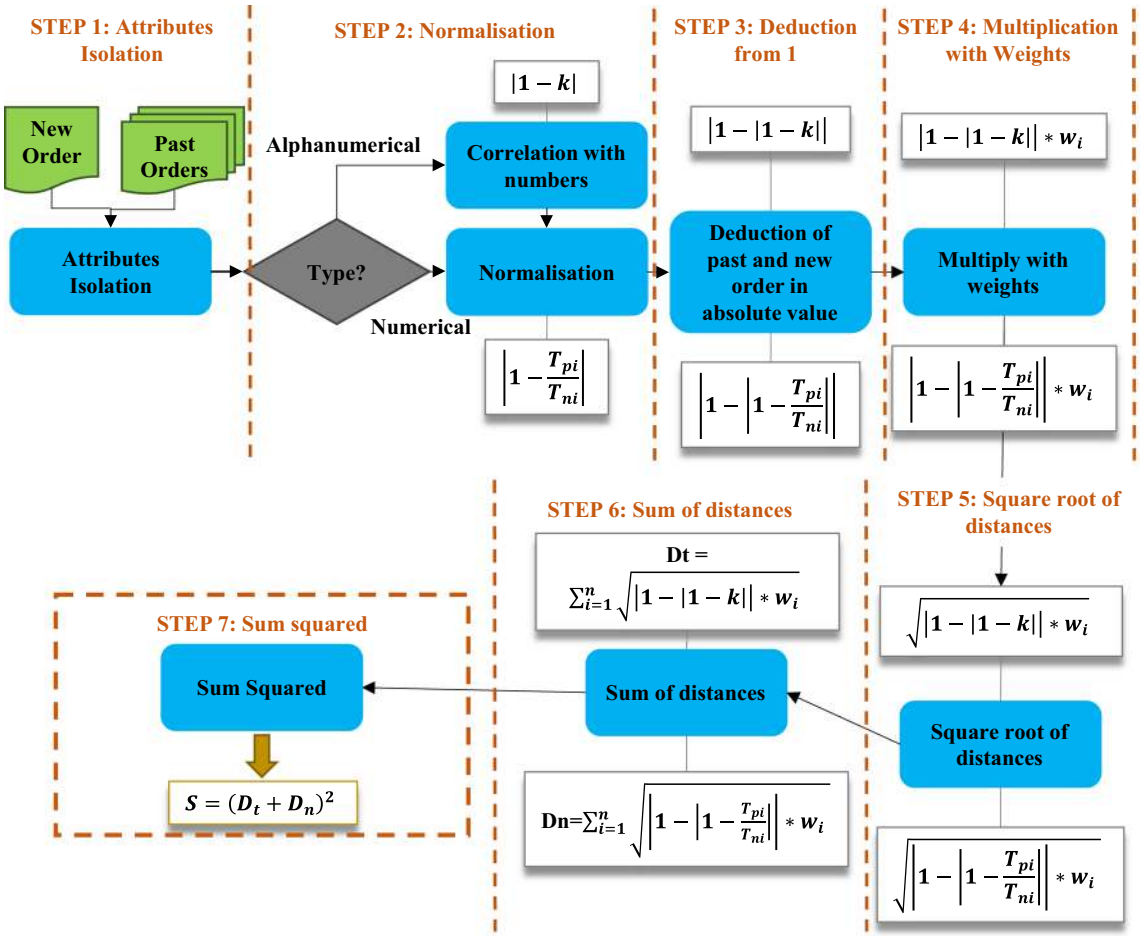

Fig. 4 Main steps of the case-based reasoning (CBR) mechanism

a simple dispatching rule or a multiple-criteria decision-making technique as described below. The advantages of dispatching rules derive from their simplicity, since they do not attempt to predict the future, but rather make decisions based on the present. Thus, these rules are very useful in factories that are extremely unpredictable, such as job-shops. Also, they are spatially local, requiring only the information available at the location where the decision will be implemented. Finally, they are easily understood by human operators and are easy to implement [41]. On the other hand, the multiple-criteria decision-making technique involves the formation of several alternatives and their evaluation before assigning the available resources to pending production tasks.

Since the method considers a finite capacity problem, in case multiple jobs are competing for a resource, ISA, and the decision matrix with the criteria and their weighting factors are used to determine which task will be dispatched to which resource, considering the planning objectives. The weighting factors that are used in the scheduling algorithm are determined based on the planner's knowledge of the business in order to satisfy business's production requirements and goals. Moreover, in case of a tie between two identical resources that are both suitable and available for a task assignment, the intelligent search algorithm selects randomly one of them and evaluates the generated path. Nevertheless, other tie breaking policies can be used such as the one proposed in [42].

In the ISA algorithm, the search of the solution space is guided by three adjustable control parameters, namely the maximum number of alternatives (MNA), the decision horizon (DH), and the sampling rate (SR). MNA controls the breadth of the search, DH controls the depth, and SR directs the search towards branches of high-quality solutions [43]. The proper selection of MNA, DH, and SR allows the identification of a good solution by examining a limited portion of the search space, thus effectively reducing computational time. For that purpose, a statistical design of experiments [44] has been carried out to reduce the number of experiments and to identify the preferable values of these factors in order to obtain the results of the highest possible quality [37, 45]. The workflow of the algorithm follows (Fig. 5):

Step 1: Start at the root and generate alternatives by random assignments for DH layers until MNA.
Step 2: For each branch (Step 1), create SR random samples until all the branch nodes are searched.
Step 3: Calculate the criteria scores for all the samples belonging to the same alternative of Step 1.
Step 4: Calculate the score of the branch as the average of the scores achieved by its samples.
Step 5: Calculate the utility values of each alternative/branch.
Step 6: Select the alternative with the highest utility value.
Step 7: Repeat Steps 1–6 until an assignment has been done for all the nodes of the selected branch.

The nodes mentioned in the workflow steps above represent decision points where a task is assigned to a resource. Once a task to resource assignment is made in each one of the nodes, an alternative production scheme is generated as shown in Fig. 5. More specifically, the ISA follows consecutive steps during the decision-making phase. The first step is the determination of the alternatives. An alternative is defined as a set of possible assignments of tasks to resources. The second step is the determination of the attributes, which are the criteria used to evaluate the alternatives. The multi-criteria ISA take into consideration a number of conflicting criteria including flowtime (4), cost (5), quality (6), and tardiness (7):

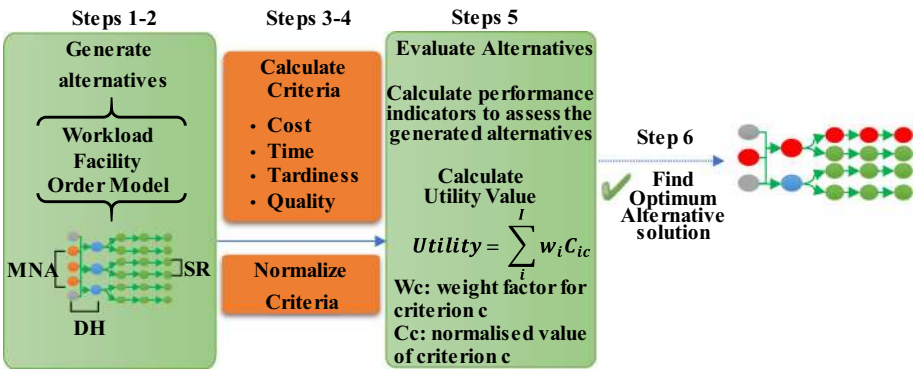

Fig. 5 Six main steps of the intelligent search algorithm (ISA). *Step 1* generate alternatives for DH until MNA, *Step 2* for each branch (*Step 1*) create SR random samples, *Step 3* calculate the criteria scores of each alternative, *Step 4* calculate the score of the branch based on its samples scores, *Step 5* calculate the utility value of each alternative, and *Step 6* select the alternative with the highest utility value

$$\text{FLOW}\,(\text{alt}_q) = \frac{\sum_{i=1}^{L} T_i^{\text{comp}}(alt_q) - T_i^{\text{arr}}}{L} \quad (4)$$

$$\text{TARD}\,(\text{alt}_q) = \frac{\sum_{i=1}^{L} T_i^{\text{comp}}(alt_q) - T_i^{\text{dd}}}{L} \quad (5)$$

$$\text{COST}\,(\text{alt}_q) = \frac{\sum_{i=1}^{L} t_i^{\text{proc}}(alt_q) \times R_i^{\text{proc}}}{L} \quad (6)$$

$$\text{QUALITY}\,(\text{alt}_q) = \frac{\sum_{i=1}^{L} (QL_i)}{L} \quad (7)$$

where R_i ith resource, alt_q qth alternative formed at the decision point, L number of completed tasks in the work-centre/job-shop at a decision point, T_i^{comp} (alt_q) completion time of the ith pending task if alt_q is implemented, T_i^{arr} time at which the ith pending task arrived at the work-centre, $t_i^{\text{proc}}(\text{alt}_q)$ estimated time required to process ith pending task if alt_q is implemented, R_i^{proc} cost of resource i to process the pending task if alt_q is implemented, QL_i quality index of the resource R_i to perform a ith pending task if alt_q is implemented, and T_i^{dd} due date of the ith pending task.

Once the criteria are determined, the consequences need to be defined. Consequences are the values of the attributes at the time the decisions and are performed in order to evaluate the selected alternatives. The set of alternatives selected during the decision-making phase is assessed using a set of performance indicators including production flowtime, resource utilisation, and mean tardiness.

A main benefit offered by the proposed scheduling mechanism is its ability to adapt to new order arrivals and quickly reschedule the job-shop. More specifically, when a new schedule needs to be generated due to the arrival of a new order, the running tasks of the existing jobs are fixed in their current positions and the rest of the tasks that are planned ahead of them are rescheduled together with the new ones. Through this functionality, the job-shop can adapt to unforeseen demand, rush orders, and other disruption in production such as machine breakdowns.

3.4 Order due-date assignment

Scheduling that incorporates due dates is of permanent interest [46]. The problem has been investigated since 1965 in the work of Jackson [47]. In academic research, the incorporation of due dates in scheduling problem acts in a twofold manner: it increases the constraints of the problem and thus it increases computational complexity, and it serves as a milestone against which important indicators are calculated, such as tardiness, earliness, and slack time. In a similar manner, in an industrial context, where the satisfaction of customer needs is the primary concern, the assignment and adherence of due dates determine the efficiency of a factory. Therefore, the assignment of realistic due dates is utterly important. In the literature, few works treat scheduling problems with the due-date assignment decision being of primary focus. Among the most significant contributions, Cheng and Gupta [48] review due-date assignment approaches up to 1990, whereas Gordon et al. [46] review more recent publications (up to 2002). In most works, the due-date assignment problem is treated using benchmarking instances [49], single-resource problems [50, 51] or is focused on assembly shops [52]. Moreover, the calculation of due dates in ETO environments is relatively mistreated. Most studies focus on static problem instances where jobs do not arrive continuously in the system [46], or require significant modelling efforts [53] and simulation experiments [54], which are rarely feasible in actual daily practice for due-date calculation.

In ETO industries, and particularly in mould-making, the actual practice implies for delivery dates to be negotiated with the customer during quotation. This agreement, later on during the job dispatching phase, will dictate how the due dates for the order will be set. The customer of a mould-making SME cannot tolerate delays in mould deliveries from the latter since this may delay the entire production of the former. Deviations in the expected delivery date of a mould can cause perturbations across the value chain and lead to supplying bottlenecks in the subsequent value-added phases of the customer, in the interlinked economy [55]. Therefore, it is utterly important for an SME to provide a customer with a solid estimation of when the mould will be available. However, the planner must have an estimation of the manufacturing lead time in the first place to use it in customer negotiations. The due date that will be defined by the lead time estimation of an order will be considered together with the customer's due date in order to satisfy the customer's demand according to the estimated due date calculated by the mould-making industry.

Therefore, the mechanism described in Sect. 3.2 is also utilised for the accurate calculation of the due date of a new order. It is reasonable that the manufacturing of two similar ETO products in a similarly configured manufacturing system will require the use of similar resources and will be completed in approximately the same time. Utilising the similarity mechanism, accurate estimations about the delivery dates can be produced for ETO products manufacturing. Once the complete set of attributes is submitted as described above, a sales agreement is achieved, the order is considered active, and the calculation of lead time initiates. The proposed due date estimation method reuses previous knowledge from executed orders, which are stored as cases in a case base. First, previous cases are retrieved and a pairwise comparison between the new order and each of the stored ones is performed. Based on the procedure described previously, the most similar case is obtained together with the calculated similarity index. The obtained case is revised in order to fit the new case

requirements. Then, the manufacturing lead time of the past case is multiplied with the similarity index to derive the estimated lead time for the new case. The estimated manufacturing lead time is essentially the due date of the order. The estimation is finally checked by the planner and if accepted is inserted in the scheduling module as the due date of the order. If rejected, the procedure described in Sect. 3.2 is followed. Finally, after production is completed, multiple scheduling adherence methods (integration with an MES system, machine monitoring techniques, manual reporting, etc.) can be utilised in order to validate the accuracy of this estimation. If the accuracy is acceptable, the case is retained and stored in the case base. The mechanism which is shown to yield high-quality results is a similar industrial case as reported by Mourtzis et al. [38].

4 Development of the knowledge-enriched scheduling app

The two sections below present the KES app and describe its architecture and the software tools used for its development.

4.1 Description of the knowledge-enriched scheduling app

The scheduling and the similarity mechanisms have been implemented in Java for validation purposes. The integrated KES engine has been designed for implementation into a mobile app for the Android OS. The designed app allows data entry, selection of decision-making criteria, definition of weight factors and tuneable parameters of the ISA, and visualisation of results, as shown in the screenshot of Fig. 6.

The planner, through the data entry menu, is capable to insert information related to the facility, jobs, and workload and model their interrelationships in the form of precedence constraints (pre- and post-conditions). The planner can also provide information related to the working calendar not only of the resources but also of the entire factory. Moreover, the data related to the workload are defined according to the order due-date assignment. The due date of each order is specified by the knowledge reuse mechanism and is taken into consideration during the generation of the alternative schedules.

The app also allows the operator to interact with the proposed schedule. This rescheduling is necessary in cases

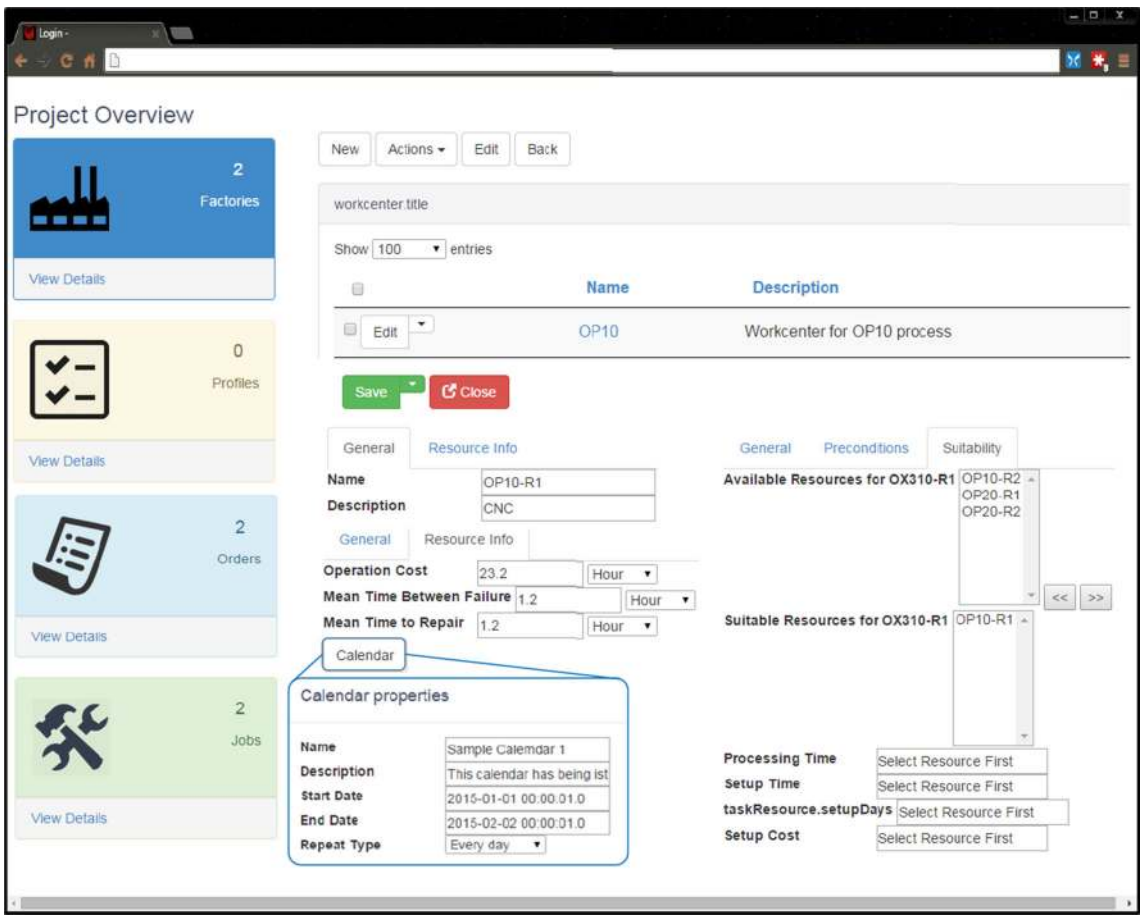

Fig. 6 Screenshot of the data entry fields of the scheduling app

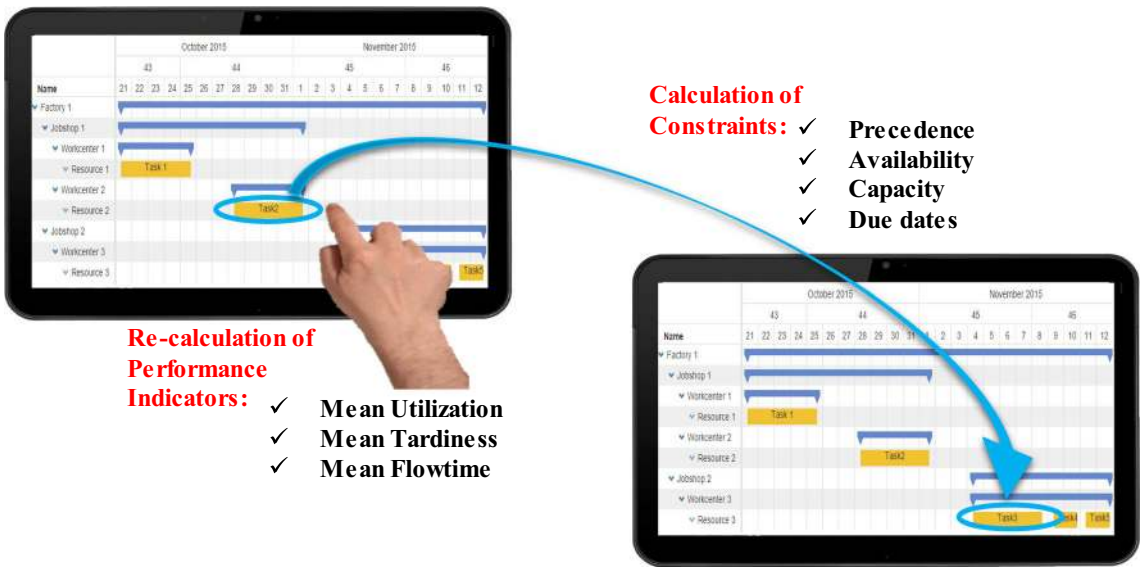

Fig. 7 Rescheduling performed by the operator through a tablet using drag and drop commands

that the derived schedule needs refinements due to order prioritisation and machine breakdowns among other reasons (Fig. 7). The precedence constrains, machine availability and capacity, and due dates are respected during this human-triggered rescheduling action. Moreover, performance indicators are recalculated each time a rescheduling occurs. Finally, the alternative with the highest utility value is displayed together with the scheduling Gantt chart and the mean values of the performance indicators (utilisation, flowtime, and tardiness).

4.2 Software architecture and development tools

Mobile apps deployed on the Android OS are based on a three-tier architecture that consists of three layers (data, business, and presentation) following the rules of the Model–view–controller architectural pattern. The presentation layer includes the graphical user interfaces of the app, and the data layer retrieves data from the back end. Finally, the business layer handles the data exchange between these two layers. For the programming of the platform, the Android software development kit (SDK) was used, which provides the developers the API libraries and tools necessary to build, test, and debug apps for Android. The back end is implemented with the Apache Tomcat version 7.0.19, since it is fully compliant with the latest advances in web programming and servlet specifications. The supporting data model of the app is based on requirements' collection from a mould manufacturer. The application runs on devices with ARM-based processors, 512 MB minimum memory, 300 MB free minimum storage space, and OS Android 4.0™ or later.

5 Industrial case study: experiments and results

The case study uses real data from a high-precision mould-making machine shop. The mould-shop best fits to the engineer-to-order (ETO) business model, where custom moulds and dies are designed and manufactured based on particular customer needs. Injection moulds produced by this mould-shop are one of a kind, first-time-right products that vary greatly in terms of quality needs, tolerances, and mainly functionality. Evidently, the mould-making industry is highly specialised and knowledge dependant. Once a new production order is released, a scheduling of its tasks must follow. Work is delegated among engineers, based on their expertise, who are usually in charge of a project from start to end. The resources needed are determined by the project's particularities. The manufacturing lead time is identified as a major competitive factor of mould-making industries [56]. Machinists together with designers usually perform a first estimation of manufacturing lead time, but the accuracy is empirical [38]. Similarly, in the current business model of the company, unofficial oral meetings take place in order to schedule resources, and, if the situation demands it, the management department is involved in the decision-making and work prioritisation. However, no software tools are used to support short-term scheduling or to document the decisions made. Therefore, by using the developed knowledge-enriched scheduling mechanism, not only will the scheduling of the tasks be more accurate, flexible, and easily reconfigurable to handle unpredictable events, but also the employees will be able to receive notifications related to schedule changes and other job-shop-related information through mobiles devices.

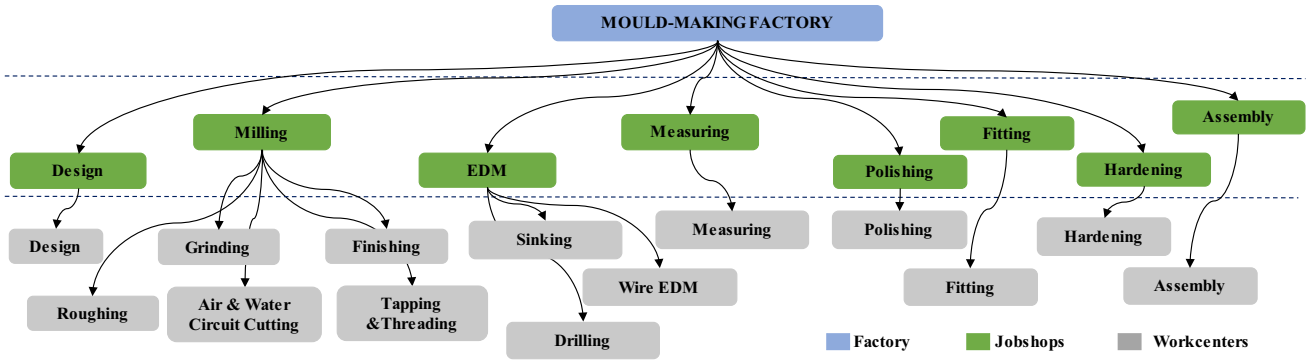

Fig. 8 Facility model of the mould-making job-shop

Table 1 Hierarchical model of the mould-making job-shop

Job-shop	Work-centre	No. of resources
Design	Design	2
Milling	Roughing	14
	Grinding	5
	Air and water circuit cutting	3
	Tapping and threading	4
	Finishing	6
EDM	Sinking	3
	Wire EDM	5
	Drilling	1
Measuring	Measuring	1
Polishing	Polishing	1
Fitting	Fitting	1
Hardening	Hardening	2
Assembly	Assembly	1

The shop-floor of the case study consists of eight job-shops, which include 14 work-centres that are formed by 40 individual resources in total (Fig. 8). The resources include on the one hand high-precision CNC machines that are capable to perform milling, drilling, turning, electro-discharge wire cutting, sinking, grinding, tapping, roughing, polishing, and hardening operations and on the other human operators that perform manually operations of design, fitting, assembly, measuring, and polishing (Table 1). The hierarchical model of the production facility is shown in Fig. 8. The utilised historical dataset includes the processing times, tasks, sequences, and resources used for the manufacturing of thirty (30) moulds. The dataset was collected within a time span of approximately 3 years.

In actual production terms, five orders (moulds) are simultaneously executed in the shop-floor on average, as the analysis of the historical data depicted. Therefore, in the experiments below, it is considered that four orders are already under processing and a new order enters the system eight calendar days later. The schedules for these orders have already been generated previously and are currently being executed. The system must then be rescheduled in order to accommodate the new order. The new mould order carries the identification number "13.23". The basic attributes of "13.23" are given in Table 2, and the different components and tasks required for this mould are shown in Fig. 9.

As described in Sect. 3.2, the new order that triggers the scheduling mechanism is first compared against documented past cases for the reuse of knowledge related to processes and product structure. In the presented work, the

Table 2 Attributes of the compared mould cases

Attributes	Mould 13.23	Mould 12.20	Mould 11.38
Number of cavities	6	2	4
Type of hardening	Very good	Very good	Very good
Side of injection	Moving side	Moving side	Moving side
Mould size	Medium	Large	Large
Core cap	No	Yes	Yes
Ejector rings	6	2	4
Temper evident	No	No	No
Type of data	Idea	Idea	Idea
Surface's quality	Mirrors	Mirrors	Mirrors
Number of basic components	9	12	11

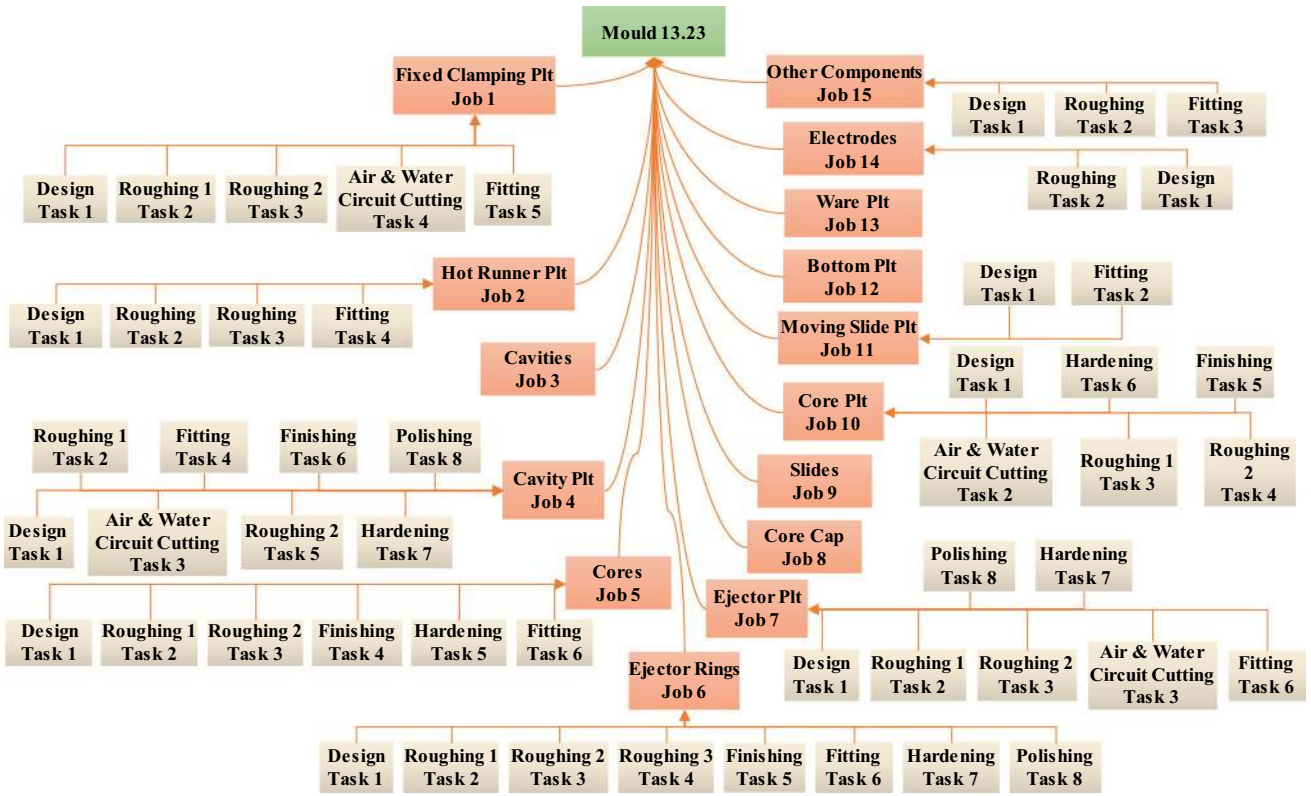

Fig. 9 Components, jobs, and tasks of the "13.23" order

past case is the complete dataset of a previously executed mould-making order, including the mould specifications/attributes, scheduling parameters used, the schedule followed (policy, sequencing, etc.), and the documented processing times and sequencing of tasks. In the case study, the new order is compared against all 30 documented past cases. Ten attributes are considered by CBR for the pairwise comparison of cases, namely the: number of cavities, type of hardening, side of injection, mould size, core cap, ejector rings, tamper evident, type of data, surface's quality, and number of basic components, with the following weight factors: 0.15, 0.05, 0.1, 0.1, 0.1, 0.1, 0.1, 0.05, 0.1, and 0.15. The weight factors were defined and fine-tuned together with the experts of the case study. The resulting values denote the potential of an attribute to reveal actual production-related similarities between any two cases. For instance, the number of basic components that need to be manufactured directly affects the required number of manufacturing processes, and therefore, the attribute "number of basic components" is attributed with a large weight value, since the more the number of required process, the longer the flowtime of production. Another fundamental parameter that is taken into consideration is the shape of the stacks. There are two options, namely moulds with cylindrical or rectangular stacks. Since the mould "13.23" has cylindrical stacks, the attribute "length" is not considered during similarity. After a similarity calculation, the results indicate a similarity index of 83 % between moulds "13.23" and "12.20". The planner then adapts the process plan of the latter in order to prepare the dataset for scheduling the former. The next most similar mould is "11.38", which is 75 % similar to "13.23". As given in Table 2, mould "13.23" differs from "12.20" in the attributes: "type of hardening", "number of basic components", and "width". The components that are needed for manufacturing the "13.23" mould are less than the components of "12.20", so the planner should reuse the sequence of processes of "11.38" mould and observe that components, such as the bottom plates, are missing. Based on his expertise, the extra components are removed and the process sequence is successfully customised for the new mould.

In a second step, the similarity mechanism is utilised in order to estimate the due date of the new mould case. The pairwise comparison of the past cases with the new one results in a ranked list of the most similar cases according to the defined attribute weights. Then, the manufacturing lead time of the past case is multiplied with the similarity index to derive the estimated lead time for the new case.

Table 3 presents the actual data of the processing times of each mould based on the company's database. Moreover, based on the data retrieved from the mould-making

Table 3 Lead time estimation produced by the CBR mechanism

Process time (h)	Mould 11.38	Mould 12.20	Mould 13.23
Roughing	406.5	333.5	212.5
Finishing	87.0	296.5	221.8
Air and water circuit	81.0	80.0	89.5
Fitting	124.5	46.5	43.5
Polishing	69.5	33.0	41.5
Hardening	504.0	504.0	504.0
EDM	34.0	20.0	30.0
Electrodes	51.0	7.5	11.5
Other processes	61.5	23.5	10.5
Assembly	72.0	86.5	72.0
Design	40.0	12.25	12.3
Lead time in hours (days)	1531 (64)	1443.25 (60)	1249 (52)

Table 4 Similarity measurements between the mould cases "13.23", "12.20", and "11.38"

Compared mould attributes	Distance 11.38 → 13.23	Distance 12.20 → 13.23
Number of cavities	0.273861279	0.273861279
Type of hardening	0.223606798	0.223606798
Side of injection	0.316227766	0.316227766
Mould size	0.223606798	0.316227766
Core cap	0	0.316227766
Ejector rings	0.223606798	0.223606798
Temper evident	0.316227766	0.316227766
Type of data	0.223606798	0.223606798
Surface's quality	0.316227766	0.316227766
Number of basic components	0.350324525	0.369274473
Sum	2.467296293	2.895094975
Similarity measure	6.087550996	8.381574914

industry and after their analysis, the average set-up time required to perform a task is 30 min. This set-up time is incorporated in the processing times in order to decrease the complexity of the model.

The results of the calculated similarity index are included in Table 4. As previously mentioned, the two stored cases "12.20" and "11.38" are the most similar to the new case "13.23". The similarity measure between "12.20" and "13.23" is calculated at $S_1 = 8.381574914$. The measure between mould "11.38" and "13.23" is $S_2 = 6.087550996$.

Afterwards, the adaptation of the case is performed in order to estimate its manufacturing lead time. The lead time of case "13.23" mould is multiplied with the similarity measure between "12.20" and "13.23", and the result is divided by 10. The resulting value for the estimation of the lead time is Lead $\text{time}_{13.23} = 1209.6708$ h (Table 5). Translating this lead time into calendar days obeying at all times the working shifts of each resource per day, the due date of the order is calculated at 51 days. It is highlighted that this estimation deviated only 3.15 % compared to the actual historical values, which depicts the accuracy of the lead time estimation method. It is also noted that by utilising a larger pool of past cases, the

Table 5 Lead time and processing times' estimation according to similarity comparison results

Processes	Processing times of 13.23
Roughing	279.52
Finishing	248.51
Air and water circuits	67.05
Fitting	38.97
Polishing	27.66
Hardening	422.43
EDM	16.76
Electrodes	6.29
Assembly	72.50
Design	10.27
Other processes	19.70
Sum (lead time)	1209.67

Fig. 10 Visualisation of schedule for two orders that are executed in the job-shop

method yields results of even higher accuracy. Once the adaptation of the new case is complete, the scheduling algorithm generates and evaluates scheduling alternatives and their respective performance indicators.

The tuneable parameters of the ISA are defined using a statistical design of experiments [44], which reduced the required number of experiments for determining the impact of tuneable parameters on the cardinal preference of the decision-making process. The number of experiments was 25, and each tuneable factor had five levels. The analysis of means (ANOM) diagrams are created and depicted the impact of the values of the tuneable factors to the utility value. According to ANOM diagrams, the preferable values to be used in the particular scheduling experiment are MNA = 100, DH = 15, and SR = 20. Each schedule is assessed with the mean values of the performance indicators of utilisation, flowtime, and tardiness, which are given by the following formulas (8, 9, and 10):

$$\text{Tardiness} \quad \mathrm{MT}(t_n) = \frac{1}{N^{\mathrm{comp}}} \sum_{i=1}^{N^{\mathrm{comp}}} \max\left(0, t_i^{\mathrm{comp}} - t_i^{\mathrm{dd}}\right) \tag{8}$$

$$\text{Flowtime} \quad \mathrm{MF}(t_n) = \frac{1}{N^{\mathrm{comp}}} \sum_{i=1}^{N^{\mathrm{comp}}} \left(t_i^{\mathrm{comp}} - t_i^{\mathrm{arr}}\right) \tag{9}$$

$$\text{Utilisation} \quad \mathrm{MU}(t_n) = \frac{1}{N^{\mathrm{comp}}} \sum_{i=1}^{N^{\mathrm{comp}}} \left(\frac{t_i^{\mathrm{comp}} - t_i^{\mathrm{start}}}{t^{\mathrm{tot}}}\right) \tag{10}$$

where N^{comp} the number of completed jobs up to time t_n, t_i^{comp} the completion time of job i, t_i^{dd} the due date of job i, t_i^{arr} the arrival time of job i, t_i^{start} the start time of job i, t^{tot} the total operating time of the facility, and t_n the time point at which all performance measures are calculated.

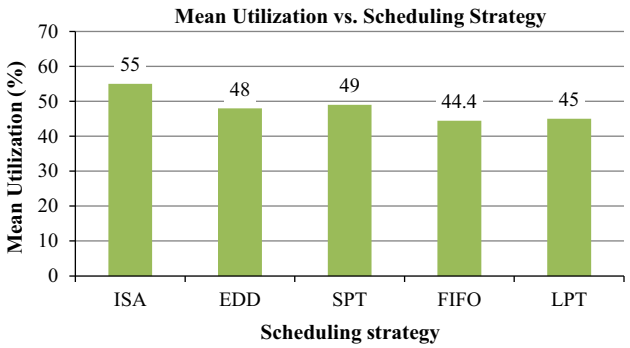

Fig. 11 Mean utilisation versus scheduling strategy

Figure 10 visualises in slip-view the schedules of an old order that was executed in the job-shop (left-hand side) (right) and the new "13.23" order that has a start date 8 days earlier (right-hand side). The figure displays the tasks for specific jobs, their duration, and the start and end time in calendar form.

In order to benchmark the performance of the ISA, a comparison against widely used dispatch rules is also performed. The rules are: first in first out (FIFO), shortest processing time (SPT), earliest due date (EDD), and least process time (LPT) [45]. The diagrams of Figs. 11, 12 and 13 reveal the superiority of the ISA in terms of the calculated performance indicator values. Still, in cases when a specific production target must be achieved, dispatch rules yielded high-quality results. For instance, EDD identified schedules with lowest flowtime and near zero tardiness compared to the other dispatch rules and ISA.

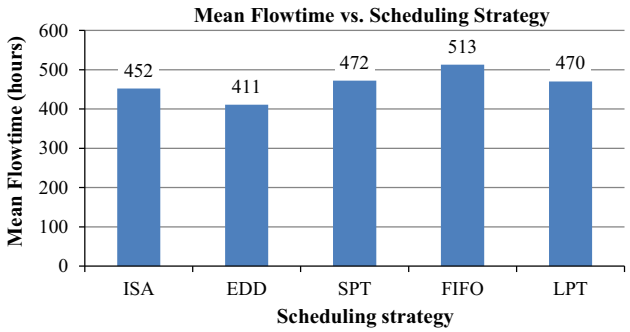

Fig. 12 Mean flowtime versus scheduling strategy

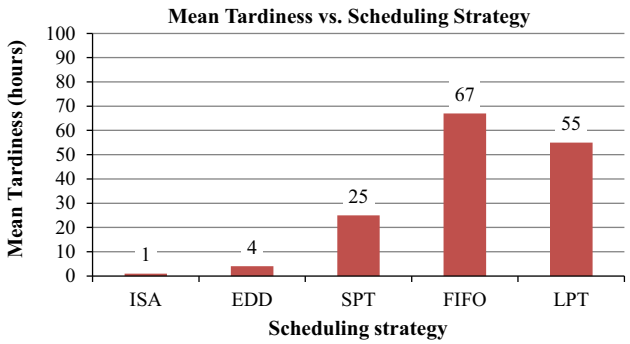

Fig. 13 Mean tardiness versus scheduling strategy

6 Conclusions and future work

The presented work focused on the enhancement of the short-term scheduling of manufacturing resources through the exploitation of historical design and planning knowledge. The scheduling of tasks for the realisation of ETO products is supported by a knowledge retrieval mechanism that is based on the CBR method and similarity measurements. The similarity index between past and new cases is measured using the Euclidean distance, and both numerical and alphanumeric attributes are considered. The scheduling is performed by the ISA that uses adjustable parameters, which configure the depth and breadth of the search, while guiding it through the solution space to identify high performing alternatives in a timely manner. The results of the application of the methodology into a real-life pilot case with data obtained from a mould-making industry verified that the short-term scheduling algorithm provides solutions of high quality in comparison to the historical values. Moreover, the deployment of the scheduling engine on mobile devices offers a certain degree of mobility, which is desired for compensating for the dynamic nature of today's turbulent manufacturing environment.

A limitation of the proposed knowledge reuse approach is the necessity for pre-existing and sufficiently documented cases. The repository of past cases in the examined case study included 30 cases with ten attributes each and provided good results. The performance of the method, in case fewer cases with partial documentation were stored in the case base, is expected to be decreased. Yet, the gathering of this amount of information about previous cases is relatively easy, since these ten attributes comprise basic characteristics of a mould, well known to the planner, and a repository with 30 products can be built in a fairly short amount of time.

Future work will focus on the quantitative evaluation of the knowledge reuse and scheduling mechanisms. The company of the case study is currently testing the developments in real-life situations. Moreover, a series of interviews with the engineers will be organised to assess the quality of the produced schedules and the accuracy of the similarity measurement results. A long-term vision is the total integration of these mechanisms in the everyday practice of the company and their utilisation through the developed app.

Acknowledgments The work reported in this paper has been partially supported by the EC FP7 Project "Applications for Advanced Manufacturing Engineering—Apps4aME" (GA No. 314156).

References

1. Chryssolouris G, Mavrikios D, Papakostas N, Mourtzis D, Michalos G, Georgoulias K (2009) Digital manufacturing: history, perspectives, and outlook. Proc Inst Mech Eng Part B J Eng Manuf 223:451–462
2. Karmarkar US (1993) Manufacturing lead times, order release and capacity loading. Handb Oper Res Manag Sci 4:287–329
3. Lowe A, McMahon C, Culley S (2004) Information access, storage and use by engineering designers. J Eng Des 30(2):30–32
4. Efthymiou K, Sipsas K, Mourtzis D, Chryssolouris G (2013) On an integrated knowledge based framework for manufacturing systems early design phase. Proc CIRP 9:121–126
5. Takahashi D (2014) Mobile apps could hit $70B in revenues by 2017. http://venturebeat.com/2014/04/29/mobile-apps-could-hit-70b-in-revenues-by-2017-as-non-game-categories-take-off/
6. Kubach U (2012) Manufacturing business web: enabling flexible and mobile business processes for the manufacturing industry. Industrial technologies 2012. http://www.industrialtechnologies2012.eu/sites/default/files/presentations_session/4_Uwe_Kubach.pdf
7. Akgun AE, Byrne JC, Keskin H, Lynn GS, Imamoglu SZ (2005) Knowledge networks in new product development projects: a transactive memory perspective. Inf Manag 42:1105–1120
8. Aamodt A, Plaza E (1994) Case-based reasoning: foundational issues, methodological variations, and system approaches. AI Commun 7:39–59

9. Kolodner JL (1992) An introduction to CBR. AI Rev 6:3–34
10. Li BM, Xie SQ, Xu X (2011) Recent development of knowledge-based systems, methods and tools for one-of-a-kind production. Knowl Based Syst 24:1108–1119
11. Lee LH, Chew EP, Frazier PI, Jia QS, Chen CH (2013) Advances in simulation optimisation and its applications. IEE Trans 45:683–684
12. Watson GI (1997) Repulsine particles on two-dimensional lattice. Phys Stat Mech Appl 246:253–274
13. Michalos G, Makris S, Mourtzis D (2011) A web based tool for dynamic job rotation scheduling using multiple criteria. CIRP Ann Manuf Tech 60(1):453–456
14. Jain AS, Meeran S (1999) Deterministic job-shop scheduling: past, present and future. Eur J Oper Res 113(2):390–434
15. Wang JF, Du BQ, Ding HM (2011) A genetic algorithm for the flexible job-shop scheduling problem. Adv Res Comput Sci Inf Eng 152:332–339
16. Ouelhadj D, Petrovic S (2009) Asurvey of dynamic scheduling in manufacturing systems. J Sched 12(4):417–431
17. Xing LN, Chen YW, Yang KW (2009) An efficient search method for multi-objective flexible job shop scheduling problems. J Int Manuf 20:283–293
18. Li X, Olafsson S (2005) Discovering dispatching rules using data mining. J Sched 8:515–527
19. Varela R, Vela CR, Puente J, Gomez A (2003) A knowledge-based evolutionary strategy for scheduling problems with bottlenecks. Eur J Oper Res 145:57–71
20. Sha DH, Liu CH (2005) Using data mining for due date assignment in a dynamic job shop environment. Int J Adv Manuf Technol 25:1164–1174
21. Karimia H, Seyed H, Rahmatia A, Zandiehb M (2012) An efficient knowledge-based algorithm for the flexible job shop scheduling problem. Knowl Based Syst 36:236–244
22. Yong H, Li S (2015) One-machine scheduling problems with deteriorating jobs and position-dependent learning effects under group technology considerations. Int J Syst Sci 26(7):1319–1326
23. Reed N, Scanlan J, Wills G, Halliday ST (2011) Knowledge use in an advanced manufacturing environment. Des Stud 32(3): 1345–1351
24. Accenture. Mobile Web Watch Survey; 2012
25. Enterprise mobile application Report, Kony; 2015. http://forms.kony.com/rs/656-WNA-414/images/Kony-Enterprise-Mobile-App-Report.pdf
26. Gröger C, Silcher S, Westkämper E, Mitschang B (2013) Leveraging apps in manufacturing. A framework for app technology in the enterprise. Proc CIRP 7:664–669
27. Lunani M Role-based mobile applications can help manufacturing companies dramatically boost productivity and growth. http://www.cognizant.com/insights/perspectives/role-based-mobile-apps-transform-manufacturing. Accessed 08 Aug 2013
28. Xu X (2012) From cloud computing to cloud manufacturing. Robot Comput Integr Manuf 28(1):75–86. doi:10.1016/j.rcim.2011.07.002
29. Hu SJ, Ko J, Weyand L, Elmaraghy HA, Kien TK, Koren Y, Bley H, Chryssolouris G, Nasr N, Shpitalni M (2011) Assembly system design and operations for product variety. CIRP Ann Manuf Technol 60(2):715–733
30. Chung S, Lee KY, Choi J (2015) Exploring digital creativity in the workspace: the role of enterprise mobile applications on perceived job performance and creativity. Comput Hum Behav 42:93–109
31. Mourtzis D, Doukas M, Vandera C (2014) Mobile applications for product customisation and design of manufacturing networks. Manuf Lett. doi:10.1016/j.mfglet.2014.01.002
32. Young HA (2012) Scientific apps are here (and more will be coming). Cytokine 59:1–2
33. (2012) Mobile ''apps'' provide parts data and calculate O-ring dimensions. Seal Technol 2012(2):3. doi:10.1016/S1350-4789(12)70037-5
34. Ozturk A, Kaya S, Ozdemirel NE (2006) Manufacturing lead time estimation using data mining. Eur J Oper Res 173(2): 683–700
35. Li BM, Xie SQ, Xu X (2011) Recent development of knowledge-based systems, methods and tools for one-of-a-kind production. Knowl Based Syst 24(7):1108–1119
36. Chryssolouris G, Subramaniam V (2001) Dynamic scheduling of manufacturing job shops using genetic algorithms. J Int Manuf 12:281–293
37. Mourtzis D, Doukas M, Psarommatis F (2014) Design of manufacturing networks for mass customisation using an intelligent search method. Int J Comput Int Manuf. doi:10.1080/0951192X.2014.900867
38. Mourtzis D, Doukas M, Fragou K, Efthymiou K, Matzorou V (2014) Knowledge-based estimation of manufacturing lead time for complex engineered-to-order products. Proc CIRP 17:499–504
39. Chryssolouris G, Dicke K, Lee M (1992) On the resources allocation problem. Int J Prod Res 30(12):2773–2795
40. Lalas C, Mourtzis D, Papakostas N, Chryssolouris G (2006) A simulation-based hybrid backwards scheduling framework for manufacturing systems. Int J Comput Int Manuf 19(8): 762–774
41. Baker A (1998) A survey of factory control algorithms which can be implemented in a multi-agent hierarchy: dispatching, scheduling, and pull. J Manuf Syst 17(4):297–320
42. Viagas F-V, Framinan JM (2014) On insertion tie-breaking rules in heuristics for the permutation flowshop scheduling problem. Comput Oper Res 45:60–67. doi:10.1016/j.cor.2013.12.012
43. Mourtzis D, Doukas M, Psarommatis F (2012) A multi-criteria evaluation of centralised and decentralised production networks in a highly customer-driven environment. CIRP Ann Manuf Technol 61(1):427–430
44. Phadke MS (1989) Quality engineering using robust design, 1st edn. Prentice Hall, Englewood Cliffs
45. Chryssolouris G (2006) Manufacturing systems: theory and practice, 2nd edn. Springer, New York
46. Gordon V, Proth JM, Chu C (2002) A survey of the state-of-the-art of common due date assignment and scheduling research. Eur J Oper Res 139:1–25
47. Jackson JR (1955) Scheduling a production line to minimize maximum tardiness. Management Science Research Project, UCLA
48. Cheng TCE, Gupta MC (1989) Survey of scheduling research involving due date determination decisions. Eur J Oper Res 38:156–166
49. Vig MM, Dooley KJ (1993) Mixing static and dynamic flowtime estimates for due-date assignment. J Oper Manag 11(1):67–79
50. Seidmann A, Panwalkar SS, Smith ML (1981) Optimal assignment of due-dates for a single processor scheduling problem. Int J Prod Res 19(4):393–399
51. Panwalkar SS, Smith ML, Seidmann A (1982) Common due date assignment to minimize total penalty for the one machine scheduling problem. Oper Res 30(2):391–399
52. Bock DB, Patterson JH (1990) A comparison of due date setting, resource assignment, and job pre-emption heuristics for the multiproject scheduling problem. Decis Sci 21(2):387–402
53. Gordon V, Strusevich V, Dolgui A (2012) Scheduling with due date assignment under special conditions on job processing. J Sched 15(4):447–456
54. Vinod V, Sridharan R (2011) Simulation modeling and analysis of due-date assignment methods and scheduling decision rules in

a dynamic job shop production system. Int J Prod Econ 129:127–146
55. Denkena B, Bluemel P, Kroening S, Roebbing J (2012) Condition based maintenance planning of highly productive machine tools. Prod Eng Res Dev 6:277–285
56. Pecas P, Henriques E (2003) The need for agile manufacturing implementation in mould making business. In: Proceedings of the business excellence I—performance measures, benchmarking and best practices in new economy. pp 321–332