

D.Y. Sha · C.-H. Liu

## Using Data Mining for Due Date Assignment in a Dynamic Job Shop Environment

Received: 11 June 2003 / Accepted: 28 August 2003 / Published online: 16 March 2004  
© Springer-Verlag London Limited 2004

**Abstract** Due date assignment is an important task in shop floor control, affecting both timely delivery and customer satisfaction. Due date related performances are impacted by the quality of the due date assignment methods. Among the simple and easy to implement due date assignment methods, the total work content (TWK) method achieves the best performance for tardiness related performance criteria and is most widely used in practice and in study. The performance of the TWK method can be improved if the due date allowance factor  $k$  could render a more precise and accurate flowtime estimation of each individual job. In this study, in order to improve the performance of the TWK method, we have presented a model that incorporated a data mining tool – Decision Tree – for mining the knowledge of job scheduling about due date assignment in a dynamic job shop environment, which is represented by IF-THEN rules and is able to adjust an appropriate factor  $k$  according to the condition of the shop at the instant of job arrival, thereby reducing the due date prediction errors of the TWK method. Simulation results show that our proposed rule-based TWK due date assignment (RTWK) model is significantly better than its static and dynamic counterparts (i.e., TWK and Dynamic TWK methods). In addition, the RTWK model also extracted comprehensive scheduling knowledge about due date assignment, expressed in the form of IF-THEN rules, allowing production managers to easily understand the principles of due date assignment.

**Keywords** Data mining · Decision tree · Due date assignment

### 1 Introduction

With the current emphasis on the just-in-time (JIT) production philosophy, it is crucial to meet the target job due date. Assigning exact due dates and timely delivering the goods to the customer will enhance customer's satisfaction as well as provide a competitive advantage. Consequently, the due date assignment is an important task in shop floor control.

To date, several due date assignment methods have been developed: TWK, Number of Operations (NOP), Jobs in Queue (JIQ), Jobs in System (JIS), Jobs in Bottleneck Queue (JIBQ), Congestion and Operation Flowtime Sampling (COFS), Operation Flowtime Sampling (OFS), etc. Among these methods, TWK method is most widely used in practice and in prior studies [1–8]. The performance of the TWK method could be improved if the due date allowance factor  $k$  could provide a more accurate and precise flowtime estimation of each individual job. The difficulty of this issue stems from the dynamic and stochastic nature of the job shop environment that precludes accurate estimation. In the TWK method, the due date allowance factor  $k$  is estimated based on historical data by a regression model and is a static coefficient that lacks the means of estimating job flowtime dynamically by taking explicit account of the current shop information [6]. Cheng et al. [6] developed a dynamic TWK (DTWK) method to provide more accurate estimation of flowtime by adjusting an appropriate due date allowance factor  $k$ , on the basis of the feedback information about the job shop status at the time a job arrives at the shop. The results indicated that the DTWK method, which employs on-line information about the job shop status, outperforms the TWK method with respect to both mean absolute lateness (MAL) and mean squared lateness (MSL). The study of [6] inspired us to use the more advance tool to improve the original TWK method.

In this study, we present a model that incorporates a data mining tool (Decision Tree) into the widely practiced and studied static due date assignment method (i.e., TWK method) to assign a suitable due date allowance factor  $k$  based on mined scheduling knowledge regarding due date assignment when the new

---

D.Y. Sha (✉) · C.-H. Liu  
Department of Industrial Engineering and Management,  
National Chiao Tung University,  
1001 Ta Hsueh Road, Hsinchu, Taiwan 30050, ROC  
E-mail: yjsha@cc.nctu.edu.tw,  
Fax: 886-3-5726730

order arrives. The new due date assignment method is capable of dynamically adjusting the due date allowance factor by using feedback information on critical factors available from mined scheduling knowledge.

Objectives of this study are: (1) use a data mining tool – Decision Tree – to mine the knowledge of job scheduling regarding due date assignment in a dynamic job shop, which are expressed in IF-THEN rules, to assign a more accurate and precise factor  $k$  of the TWK method when the job arrives, so as to improve the performance of the TWK method; (2) mining the knowledge of job scheduling about due date assignment to assist production managers in comprehending which factors are most important for predicting the job due date, and how the job due date is affected by various levels of critical factors.

Simulation results show that the proposed RTWK model is significantly better than its static and dynamic counterparts (i.e., TWK and DTWK methods) in reducing mean absolute lateness and mean squared lateness. In addition, the RTWK model also extracted comprehensive scheduling knowledge about the due date assignment for production managers.

The remainder of this paper is organized as follows. In Sect. 2, the relevant literatures on TWK, DTWK methods and data mining are summarized. In Sect. 3, the rule-based TWK due date assignment model is introduced. Simulation experiments conducted to compare performances will be described in Sect. 4. Simulation results will be discussed and analyzed in Sect. 5. And finally, in the last section we will draw conclusions and make suggestions for future study.

## 2 Background

To describe the process of mining scheduling knowledge about due date assignment from a dynamic job shop for constructing the RTWK due date assignment model, it will help to first discuss three areas as background, and they are: the TWK due date assignment method, the dynamic TWK due date assignment method, and data mining.

### 2.1 Total work content due date assignment method (TWK)

The TWK method is a static- and job character-related due date assignment method that issues the same degree of tightness (i.e., due date allowance factor) to all jobs based only on their total processing time information [6].

$$d_i = r_i + kP_i, \quad (1)$$

where  $d_i, r_i, P_i$  denote the due date, the arrival time, the total processing time of job  $i$  respectively; and  $k$  is the due date allowance factor reflecting the expected flowtime that job  $i$  will experience in the system. In general, the  $k$  value is estimated based on the regression model.

Among the simple and easy to implement due date assignment method, the TWK rule achieves the best performance for

tardiness-related performance criteria [1]. The TWK method is widely used in practice and in previous studies [1–8].

### 2.2 Dynamic total work content due date assignment method (DTWK)

Cheng et al. [6] modified the TWK method to provide a more accurate estimation of job flowtime. They used intuition to modify the regular TWK method: when the shop load is heavy, a relatively longer flowtime allowance should be assigned to an arriving job, and in the opposite case, a shorter allowance should be assigned. For the TWK method, Cheng et al. [6] developed a modified form by employing a dynamic due date allowance factor  $k$  based on feedback information regarding the current job shop status at the time a job arrives at the shop.

If we let  $N_S, \lambda, F$  denote the number of jobs in the system, the average job arrival rate and the job flowtime respectively, then Little's law for a shop in steady state can be expressed by Eq. 2, where  $E(\cdot)$  is the expected value operator.

$$E(N_S) = \lambda E(F) \quad (2)$$

If the shop load is relatively steady for a short period of time then, at any given time  $t$ , the approximate average flowtime of a job  $F_t$  in the shop with  $N_{S_t}$  number of jobs in this period is given by Eq. 3.

$$F_t = \frac{N_{S_t}}{\lambda} \quad (3)$$

Thus, the dynamic allowance factor for a newly arrived job at time  $t$  would be determined by the current average flowtime as Eq. 4,

$$k_t = \frac{F_t}{\mu_p \mu_q}, \quad (4)$$

where  $k_t, \mu_p, \mu_q$  denote the real level of tightness at the time  $t$  when a new job arrives, mean operation processing time and the average number of operations per job, respectively. To prevent allowance factors less than one, we use  $\max[1, k_t]$  instead of  $k_t$ , and define the corresponding dynamic due date as Eq. 5,

$$d_i = r_i + \max[1, k_t] P_i, \quad (5)$$

where  $d_i, r_i, P_i$  denote the due date assigned to job  $i$ , the arrival time, and total processing time of job  $i$  respectively. Simulation results show that the DTWK method outperforms the TWK method with respect to mean absolute lateness and mean squared lateness [6]. This illustrates that the dynamic feedback of shop load information is very useful in due date setting [6].

### 2.3 Data mining

Data mining is an automated process of extracting structured knowledge from databases, which is often referred to as a particular step in the overall process of discovering useful knowledge from data, called knowledge discovery from database

(KDD) [9]. Data mining is an interdisciplinary field, whose core is at the intersection of machine learning, statistics, and databases [10]. There are several data mining tasks, including classification, regression, clustering, dependence modeling, etc. [11]. Among these tasks, the goal of classification is to assign each case (object, record, or instance) to one class, out of a set of predefined classes, based on the values of some attributes (called predictor attributes) for the case [10]. In the classification task, the discovered knowledge is often expressed in the form of IF-THEN rules, which have the advantage of being a high-level and symbolic knowledge representation contributing towards the comprehensibility of the discovered knowledge [10, 12], as follows:

Rule: *If  $X_1$  and  $X_2$  and ...  $X_n$  then  $Y$ ,*

where  $X_i$ ,  $Y$  denote the rule antecedent and the rule consequent, respectively. The rule antecedent contains a set of conditions (terms), usually connected by a logical conjunction operator (AND); The rule consequent specifies the predefined class predicted for cases whose predictor attributes satisfy all the terms specified in the rule antecedent [10].

In this study, comprehensibility is very important whenever discovered knowledge will be used to support a decision made by production managers. If discovered knowledge is not comprehensible to the user, the user may not be able to interpret it properly, and may not have sufficient trust in the discovered knowledge to use it for decision making. Therefore, we employ a data mining method for discovering classification rules, for developing a new due date assignment model that is comprehensible and effective for production managers. There are several paradigms of classification, including decision trees, neural networks, evolutionary algorithms, etc. Among these paradigms, the output from neural networks is not easily interpretable by humans [13], and evolutionary algorithms are time-consuming. Therefore, decision trees were chosen for this study.

### 2.3.1 Decision trees

Decision trees are popular tools in machine learning. They are also well suited for the classification task in that they create symbolic rules that are interpretable by humans, they are fast, and they are reasonably good at a variety of classification tasks [13, 14]. The strengths of decision trees are as follow [15].

- Decision trees are able to generate understandable rules.
- Decision trees perform classification without requiring much computation.
- Decision trees are able to handle both continuous and categorical variables.
- Decision trees provide a clear indication of which fields are most important for prediction or classification.

The strengths of decision trees will support us in the development of a rule-based TWK due date assignment model for production managers: a model that would improve the due

date related performances and allows for easy comprehension of which factors are most critical for job due date setting.

A decision tree is a tree-shaped structure that represents a series of decisions. A decision tree can be used to classify a particular case (object, record or instance) by starting at the root node of the tree and moving through it until the case arrives at a leaf node (terminal). When a leaf node is reached, a decision is made. Each non-terminal node represents a test to be carried out on a single attribute value (i.e., input variable value) of the considered case, which leads to the root of the sub-tree corresponding to the test's outcome. Iterative Dichotomiser 3 (ID3) algorithm and its successor C4.5 algorithm are the two most widely used and practical decision tree algorithms for inductive inference, and have been successfully applied to a wide range of learning tasks [16, 17]. The basic strategy of both algorithms is as follows [14].

- The tree starts as a single node representing the training samples.
- If the samples are all of the same class, then the node becomes a leaf and is labelled with that class.
- Otherwise, the algorithm uses an entropy-based measure, known as information gain as a heuristic for selecting the attribute that will best separate the samples into individual classes. The attribute with the highest information gain becomes the test or decision attribute at the node.
- A branch is created for each known value of the test attribute, and the samples are partitioned accordingly.
- The algorithm uses the same process recursively to form a decision tree for the samples at each partition. Once an attribute has occurred at a node, it need not be considered in any of the node's descendants.
- The recursive partitioning stops only when any one of the following conditions is true.
  - a. All samples for a given node belong to the same class.
  - b. There are no remaining attributes on which the samples may be further partitioned. In this case, majority voting is employed. This involves converting the given node into a leaf and labelling it with the class that is in a majority among the samples. Alternatively, the class distribution of the node samples may be stored.
  - c. There are no samples for branch *test-attribute* =  $a_i$ . In this case, a leaf is created with the majority class in the samples.

The C4.5 algorithm is an extended form of ID3 algorithm with some additional specific enhancements such as the ability to handle continuous attribute values and missing attribute values, alternative measures for selecting attributes, and pruning decision trees. Therefore, the C4.5 algorithm was used in this study for constructing decision tree and rules.

When a decision tree is built, many branches will reflect anomalies in the training data due to noise or outliers that are caused by overfitting the data. Overfitting fails to accurately classify new unseen instances. Experiments have shown that overfitting decreases the accuracy of the learned decision trees by 10%–25% in most problems [18]. The C4.5 algorithm uses *rule post-pruning* to overcome the problem of overfitting [16, 17].

After pruning the decision tree, we are able to turn low-level data into high-level knowledge, which is represented as a set of IF-THEN rules that are created for each path from the root node to a leaf node. Each attribute-value pair along a given path forms a conjunction in the rule antecedent; the leaf node holds the class prediction, forming the rule consequent [14]. A rule can also be pruned by removing any condition in its antecedent that does not improve the estimated accuracy of the rule [14]. The IF-THEN rules are generally easier to understand for humans than other representations (e.g., neural network) [18].

To the best of our knowledge, the use of decision trees for mining the scheduling knowledge to develop due date assignment method in a dynamic job shop, in the context of shop floor control, is a research area that has so far remained unexplored. We believe that the development of data mining applied to shop floor control is a promising area of research.

### 3 Rule-based TWK due date assignment model

In this section, we discuss in detail our proposed decision tree for developing a new due date assignment model by modifying the due date allowance factor setting in the regular TWK method, called rule-based TWK due date assignment (RTWK) model. The section is divided into four subsections, namely, general description of the RTWK model, data preparation, model construction, and model application.

#### 3.1 General description of the RTWK model

The TWK method is static- and job character-related due date assignment method [6]. It seems that for due date related performances, which emphasize the importance of meeting job due date as closely as possible, the performance can be improved if

the due date allowance factor could render a more accurate and precise flowtime estimation of each individual job. The TWK method lacks the means of estimating the due date allowance factor  $k$  dynamically, which explicitly takes into account some of the factors that are critically affected in due date setting.

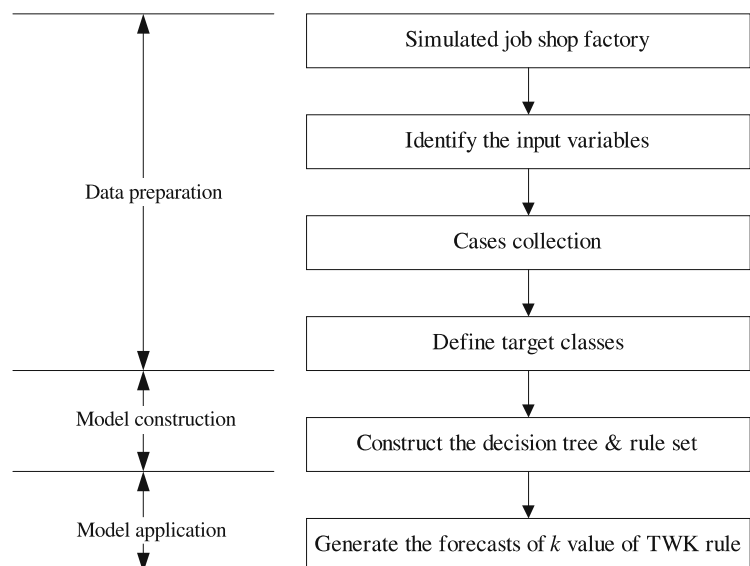
In this study, we have presented a model that incorporates a data mining tool (Decision Tree) for mining the scheduling knowledge about due date assignment in a dynamic job shop, which is expressed in IF-THEN rules, which modifies the due date allowance factor setting of the TWK method. We expect this new rule-based TWK due date assignment (RTWK) model will provide a more accurate and precise flowtime estimation than the TWK method.

The overall flow of the RTWK model is shown in Fig. 1. After modelling the virtual job shop, the case set must be collected for constructing the decision tree and the rule set. Job due date may be affected by many factors related to job characteristic and shop conditions. We must identify some of the influential factors as input variables for the C4.5 algorithm. Once the input variables are identified, the case set will be a list of such input variables and due date allowance factor  $k$ , including all orders received in a simulated period of data collection which are stored in the database. After collecting the case set, the target classes must also be defined by segmenting the distribution of the due date allowance factor  $k$  of the case set. Finally, the C4.5 algorithm mines the scheduling knowledge regarding the due date assignment from the case set to construct the rules for estimating the due date allowance factor  $k$  of the arriving job.

#### 3.2 Data preparation

In order to use the decision tree to mine the scheduling knowledge about due date assignment in a dynamic job shop, we need to generate sufficient data as a case set for the training. This subsection is further divided into four parts, namely, the job shop

Fig. 1. Overall flow of the rule-based due date assignment model



model, identifying of the input variables, cases collection, and defining of the target classes.

### 3.2.1 The job shop model

To mine the scheduling knowledge from a dynamic job shop, a suitable case needs to be defined. This research uses a  $10 \times 10$  benchmark problem from Lawrence [19]. This test model has ten jobs, each with ten operations and ten machines. Table 1 provides the data for the problem using the following structure: machine, processing time. In this study, the probability of each product being chosen to be released into the shop is equal. Jobs arrive continually with inter-arrival times generated from a negative exponential distribution, which has a mean value chosen to create a certain expected shop utilization rate.

The virtual job shop was built on personal computer with a Pentium III 700 processor using the eM-Plant 4.6, a simulation package developed by Tecnomatix Technologies Corporation.

### 3.2.2 Identify the input variables

To assign a due date for job  $i$  in a dynamic job shop, most methods include one or more influential factors into their prediction models. Among these factors, those related to job characteristics and shop conditions are listed as follows [3].

1. Total processing time of job
2. Number of operations of the job
3. Number of jobs in work-centre queues on job  $i$ 's routing when it is released to the shop
4. Number of jobs in the system when job  $i$  is released to the shop
5. Total processing time of all jobs in work-centre queues on job  $i$ 's routing when it is released to the shop

Among these factors, factors 1 and 2 are related to job characteristics, the remaining ones are related to shop conditions. Due to the configuration of our virtual job shop, the above factors should be revised. The value of factor 1 is dependent upon the product type of job, so the product type of job will be used in place of it. Because each job must be processed through all machines (10 machines) in our shop, the values of factor 2 for each job are equal and the value of factor 3 is also equal to factor 4.

**Table 1.**  $10 \times 10$  job shop problem [19]

Job	Operation									
	1	2	3	4	5	6	7	8	9	10
1	5,18	8,21	10,41	3,45	4,38	9,50	6,84	7,29	2,23	1,82
2	9,57	6,16	2,52	8,74	3,38	4,54	7,62	10,37	5,54	1,52
3	3,30	5,79	4,68	2,61	9,11	7,89	8,89	1,81	10,81	6,57
4	1,91	9,8	4,33	8,55	6,20	3,20	5,32	7,84	2,66	10,24
5	10,40	1,7	5,19	9,7	7,83	3,64	6,56	4,54	8,8	2,39
6	4,91	3,64	6,40	1,63	8,98	5,74	9,61	2,6	7,42	10,15
7	2,80	8,39	9,24	4,75	5,75	6,6	7,44	1,26	3,87	10,22
8	2,15	8,43	3,20	1,12	9,26	7,61	4,79	10,22	6,8	5,80
9	3,62	4,96	5,22	10,5	1,63	7,33	8,10	9,18	2,36	6,40
10	2,96	1,89	6,64	4,95	10,23	8,18	9,15	3,64	7,38	5,8

**Table 2.** The case for each job  $i$

Variable	Information
<i>Job Characteristic</i>	
Job type	The product type for job $i$
<i>Shop Conditions</i>	
M0	Sum of the remaining processing time on the 3 <sup>rd</sup> bottleneck machine for all the jobs in the shop
M3	Sum of the remaining processing time on the 1 <sup>st</sup> bottleneck machine for all the jobs in the shop
M6	Sum of the remaining processing time on the 2 <sup>nd</sup> bottleneck machine for all the jobs in the shop
WIP	Number of jobs in the system when job $i$ is released to the shop
SRT	Sum of the remaining processing time for all jobs in the shop
<i>Target Value</i>	
$k$	Due date allowance factor that reflects the flowtime that job $i$ experienced in the system (actual flowtime in the system of the job divided by the total processing time of the job)

Hence, the factors 2 and 3 are omitted from our study. In addition, we also choose to put an extra three factors regarding shop conditions into our prediction model: the sum of the remaining processing time on 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> bottleneck machines for all the jobs in the shop. In this study, the case will indicate six input variables and a due date allowance factor  $k$ , the value of factor  $k$  equals the actual flowtime in the system of the job divided by the total processing time of the job. The content of a single case is shown in Table 2. A case is a contextualized piece of knowledge representing an experience.

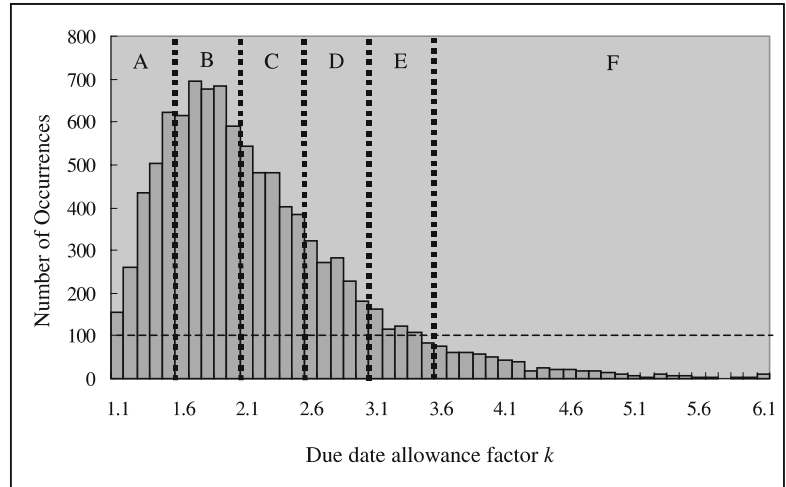
### 3.2.3 Cases collection

In this study, the warm-up period for the shop is the time interval from the start of the simulation to the completion of the first 1000 jobs. The cases are then collected on the input variables and due date allowance factor  $k$  for the next 10 000 jobs as a case set.

### 3.2.4 Define target classes

The decision tree method that we used in our study required the target classes to be a categorical variable. Clearly, the value of factor  $k$  is required to be discrete, as it is has to be a categorical value. After collecting the case set, an easy and intuitive method is used to generate the target categorical classes in this study. Figure 2 is a histogram of number of occurrences of factor  $k$  in the case set, and is used to generate the categorical  $k$  value. First, if the number of occurrences of the  $k$  value in the case set is less than 100 (1% of the total cases), the  $k$  value is an outlier. Then, each five  $k$  values are grouped into a target class from left to right of the histogram until the outliers are filled in a class; and further, all outliers are also identified as the outlier class. In Fig. 2, there are six target classes to be identified as A, B, C, D, E, and F. Among these six target classes, the class F is an outlier class in this example, representing the extreme conditions in due date prediction.

**Fig. 2.** Target classes generalization



The representative value of class  $i$  ( $CV_i$ ) must be assigned and calculated by the weighted averages method as follow.

$$CV_i = \frac{\sum_{j=1}^5 n_{ij} * k_{ij}}{\sum_{j=1}^5 n_{ij}}, \quad i \in \text{target class set}, \quad (6)$$

where  $n_{ij}$  and  $k_{ij}$  denote the number of occurrences of  $j$ th factor  $k$  in class  $i$  and the value of  $j$ th factor  $k$  in class  $i$ , respectively. In this example, the value of the target classes are calculated: class A as 1.4, class B as 1.8, class C as 2.3, class D as 2.8, class E as 3.3, and class F as 4.2. After the target classes are generated, the factor  $k$  for the 10000 jobs in the case set is transformed into a target categorical target class according to the above principles of generalization. For example, if the value of factor  $k$  of the sample job in the case set is greater than 1.1 and less than 1.5, class A and class value (= 1.4) are identified to represent it.

### 3.3 Model construction

After the case set is collected and target classes are defined, we use the See5 package (C4.5 commercial windows version) as the decision tree learning tool to construct a decision tree and rule set from the case set, which means the scheduling knowledge about due date assignment, for developing our proposed rule-based TWK due date assignment model.

### 3.4 Model application

In order to assign a more accurate and precise due date for an arriving job, the discovered rules are applied. The rule set is generally easier to understand than trees since each rule describes a specific context associated with a class. Furthermore, a rule set generated from a tree usually has fewer rules than the tree has leaves, another plus for comprehensibility.

When a customer asks for a due date quotation for an order, a new input case consisting of the input variables that are specified in the rule antecedents will be provided to the RTWK model for assigning an appropriate rule consequent (i.e., due date allowance factor  $k$ ). Sometimes, it may happen that several of the rules are applicable. Therefore, the discovered rules must be sorted by confidence in our study. Under this option, the rule that most reduces the error rate appears first, and the rule that has the lowest confidence appears last. The first rule that covers the new order is applied, i.e., the due date of the order is assigned to the class value predicted by that rule's consequent. There is also a default class that is used when none of the rules apply. In summary, the structure of the rule-based TWK is shown in Fig. 3.

## 4 Simulation experiments

The main objective of this work is to evaluate the effectiveness and robustness of the RTWK model in improving the due date performance. A three-factor full factorial design is employed to do a comprehensive study of the effects of the decision factors on the selected performance measures. Factors to be evaluated are due date assignment method, dispatching rule, and shop utilization. Table 3 lists the parameters.

### 4.1 Due date assignment method

For comparison with the proposed RTWK model, we choose its corresponding static and dynamic counterparts, i.e., the TWK and DTWK due date assignment methods.

### 4.2 Dispatching rule

Three typical dispatching rules are listed in order of increasing sophistication in the use of information for determining the next job to be processed on an available machine: first come

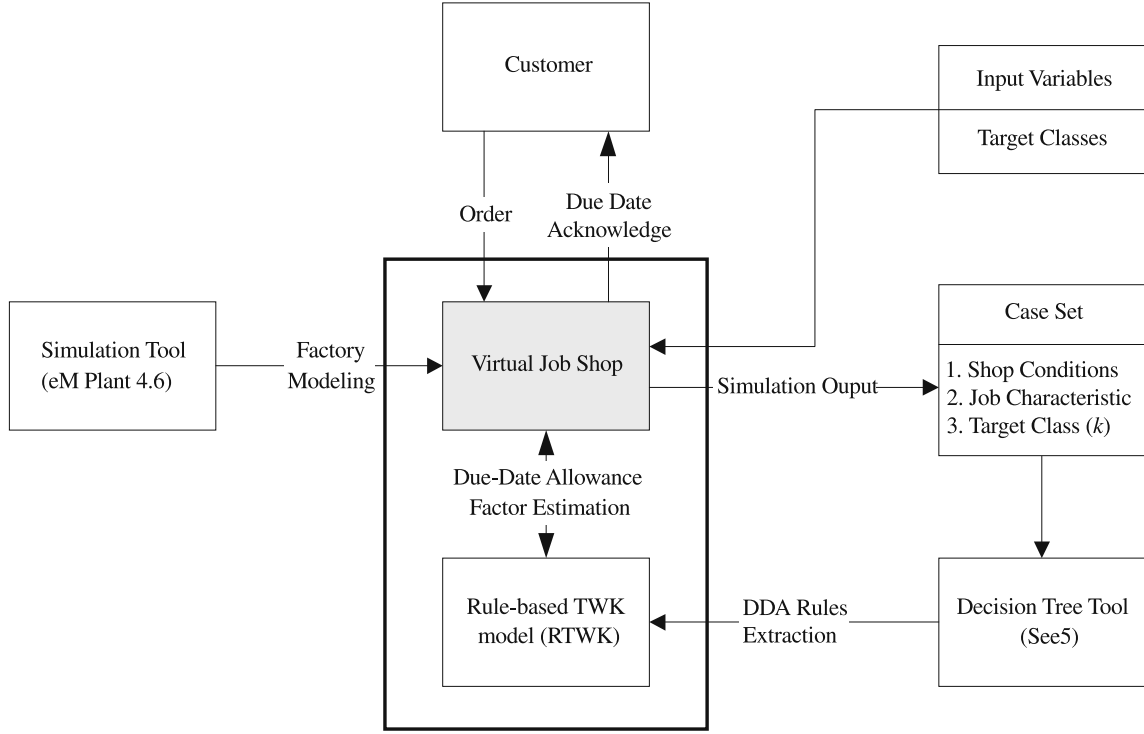


Fig. 3. The structure of the RTWK model

Table 3. Decision factor setting

Factor	Level	Number of levels
Due date assignment method	RTWK, DTWK, TWK	3
Dispatching rule	EDD, SPT, FCFS	3
Shop utilization	90%, 80%	2

first served (FCFS), earliest due date (EDD), and shortest processing time (SPT). We choose those rules because they don't need parameter estimation, were most frequently used in previous studies, and each has different characteristics. Among these rules, EDD is a due date oriented rule, SPT is a process time oriented rule, and FCFS is a random rule that will be used as the base line rule in the experiments.

#### 4.3 Shop utilization

Two levels of shop utilization were employed in this study: the 80% level represents a moderately heavy shop load, while the 90% represents a heavy shop load. They are also typical shop load ratios used in many previous studies on dynamic job shop scheduling [6].

#### 4.4 Performance measures

The quality of the flowtime estimator can be determined in terms of *accuracy* and *precision*. Vig et al. [20] defined *accuracy* of an

estimate as the closeness of the individual estimates to their true values and, *precision* as the variability of the prediction errors. In this study, we use mean absolute lateness (MAL) to measure the accuracy; and mean squared lateness (MSL) to measure the precision. The formulas used for the performance measure are as follows.

1. Mean absolute lateness (MAL), which measures the average absolute difference between the actual completion dates and the promised due dates for orders. A smaller MAL value implies a better due date prediction capability. MAL is always equal to the sum of the mean earliness (ME) and mean tardiness (MT).

$$MAL = \left\{ \sum_{i=1}^n [\max(0, d_i - f_i) + \max(0, f_i - d_i)] \right\} / n \quad (7)$$

2. Mean squared lateness (MSL), which measures the average squared difference between the actual completion dates and the promised due dates for orders. A smaller MSL value implies a smaller deviation from a designated due date occurred.

$$MSL = \left\{ \sum_{i=1}^n [\max(0, d_i - f_i) + \max(0, f_i - d_i)]^2 \right\} / n, \quad (8)$$

where  $f_i$ ,  $d_i$ ,  $n$  denote the completion time, promised due date of order  $i$ , and sample sizes respectively.

## 5 Results and discussions

For each of the 6 case sets (3 dispatching rules, 2 shop utilizations), the decision tree tool (See5) induced the specific characteristics rules for each combination of dispatching rule and shop utilization. This section presents the rules and their numerical comparison with the TWK and DTWK methods.

### 5.1 Rule set

The rules induced are all of the same IF-THEN form. The inferred information by See5 for all combinations of dispatching rule and shop utilization in this job shop are listed in Table 4. The RTWK model uses fewer rules for assigning the due date allowance factor  $k$  to a new order, except for the combinations of the SPT rule and 90% utilization, and of the FCFS rule and 90% utilization, which process more randomly, causing an increase in job flowtime as well as in its variability. The number of target class and the value of target class are dependent upon the variations of due date allowance factor  $k$  in the case set.

In this study, eight rules were induced by See5 in the combination of EDD rule and 90% utilization as shown in Table 5. The induced rules are sorted by level of confidence. The first rule that covered the new job was applied, i.e., the job due date allowance factor is assigned the class value predicted by that rule's consequent. The result of the first rule indicates that if the sum of the remaining processing time on the 1<sup>st</sup> bottleneck machine for all jobs in the shop is greater than 1082, which means the shop is very congested, then the new order should have the greatest due date allowance (i.e.,  $k = 4.2$ ). The result of the second rule indicates that when the value of SRT is less than 2185, it means that the shop load is relative low, and the smallest due date allowance is given (i.e.,  $k = 1.4$ ), and so on. If none of the rules can apply to the new order, then the default class will be assigned to it (i.e.,  $k = 1.8$ ). At the same time, the Table 5 shows that the information of the remaining load of 1<sup>st</sup> and 2<sup>nd</sup> bottleneck machines, SRT, and WIP have critical effects on the due date setting in the combination of EDD rule and the 90% shop utilization, but the remaining input variables do not.

### 5.2 Numerical comparison of RTWK, DTWK and TWK

For each treatment (there are 18 treatments, each with a specific combination of due date assignment method, dispatching rule,

**Table 5.** Rules inferred from the case sets with EDD rule at 90% utilization

Rule	Rule antecedent	Rule consequent ( $k$ )	Confidence
1	M3 > 1082	F ( $k = 4.2$ )	0.703
2	SRT <= 2185	A ( $k = 1.4$ )	0.591
3	SRT > 2185 SRT <= 3193	B ( $k = 1.8$ )	0.570
4	M3 <= 709 WIP <= 13	B ( $k = 1.8$ )	0.490
5	M3 <= 709 WIP > 13 SRT > 3193	C ( $k = 2.3$ )	0.472
6	M3 > 709 M3 <= 862	D ( $k = 2.8$ )	0.452
7	M3 > 709 M3 <= 1082 M6 <= 933	D ( $k = 2.8$ )	0.427
8	M3 > 862 M3 <= 1082 M6 > 933	E ( $k = 3.3$ )	0.381
Default class: 1.8			

and shop utilization), ten replications are conducted so as to minimize the variation of the results. For each simulation run (there are 180 runs, 10 runs for each treatment), the warm-up period for the shop is the time interval from the start of the simulation to the completion of the first 1000 jobs, the information is then collected on the two performance measures (MAL, MSL) for the next 10000 jobs. The results of the factorial experiment are summarized in Tables 6–7 and Figs. 4–7. Each item in Tables 6–7 is an average of the ten replications of the experiment.

The results of the due date related performance in Tables 6 and 7 deteriorate as shop utilization increases. To explain this phenomenon we noted that, the more congested the shop is, the less stable the scheduling system becomes, thus causing difficulties in predicting the job flowtime accurately and precisely.

For the performance of the dispatching rules, the phenomenon of reduced MAL and MSL values with the EDD rule demonstrates that due date information is very useful in controlling and improving the due date assignment process. The combinations with the SPT rule yield the poorest results in comparison with the other dispatching rules in respect to both performance measures. This is due to the fact that the SPT rule creates a more dynamic and stochastic environment that makes

**Table 4.** The inferred information of all configuration of shop

Combination of dispatching rule and shop utilization	Number of rules in RTWK	Number of target classes	The value of target classes
EDD-90%	8	6	1.4, 1.8, 2.3, 2.8, 3.3, 4.2
EDD-80%	6	5	1.3, 1.8, 2.3, 2.8, 3.5
SPT-90%	27	5	1.4, 1.8, 2.3, 2.8, 4.5
SPT-80%	4	5	1.3, 1.8, 2.3, 2.8, 4.2
FCFS-90%	23	6	1.4, 1.8, 2.3, 2.8, 3.3, 4.5
FCFS-80%	8	5	1.3, 1.8, 2.3, 2.8, 3.6

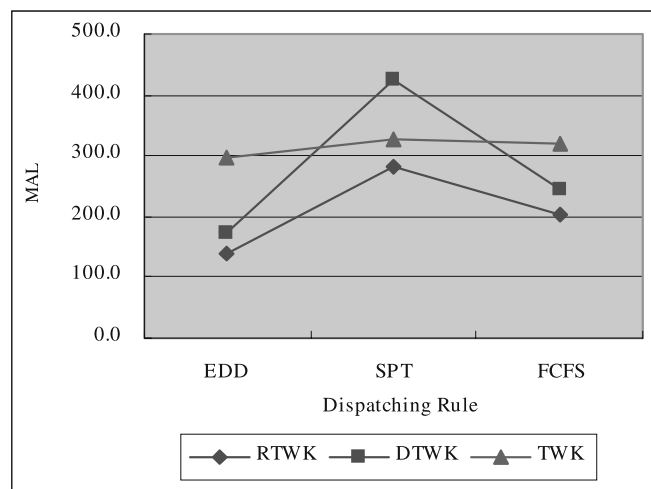


**Table 6.** Experimental results for mean absolute lateness

Shop utilization	Due date assignment method	Dispatching rule		
		EDD	SPT	FCFS
90%	RTWK	<b>138.5</b>	<b>280.8</b>	<b>201.5</b>
	DTWK	171.3	423.6	243.2
	TWK	295.7	327.3	319.3
80%	RTWK	<b>121.0</b>	<b>180.8</b>	<b>149.4</b>
	DTWK	166.0	263.6	210.8
	TWK	188.1	205.8	214.6

**Table 7.** Experimental results for mean squared lateness

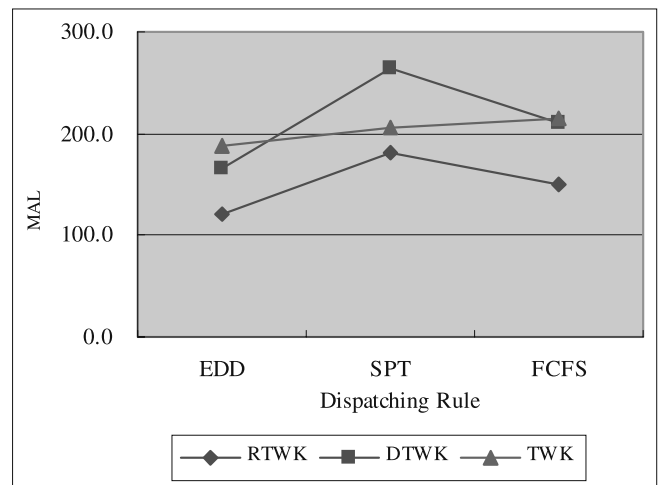
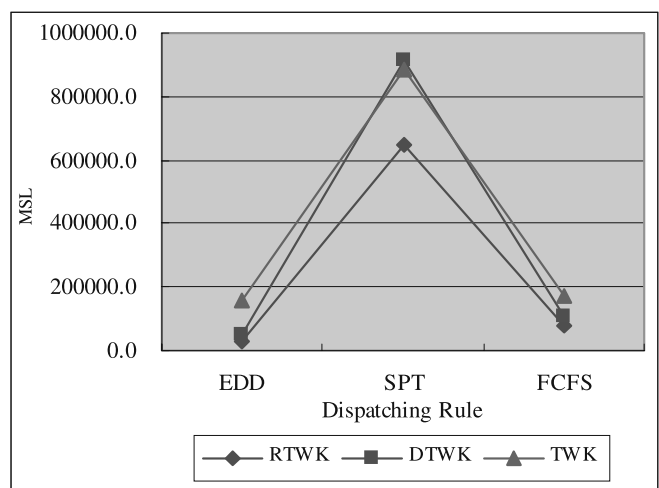
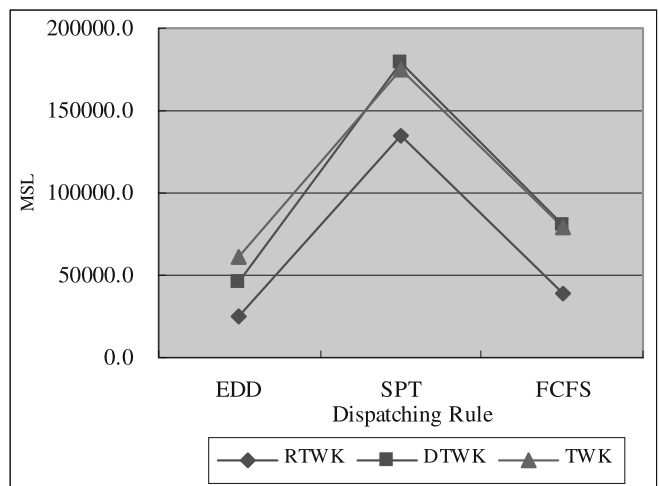
Shop utilization	Due date assignment method	Dispatching rule		
		EDD	SPT	FCFS
90%	RTWK	<b>32262.2</b>	<b>650191.6</b>	<b>77919.8</b>
	DTWK	47872.6	913627.6	109195.5
	TWK	157968.0	882118.5	170753.2
80%	RTWK	<b>24314.7</b>	<b>134806.8</b>	<b>38758.3</b>
	DTWK	46019.6	179010.3	80785.5
	TWK	60790.0	175376.1	79745.7

**Fig. 4.** Mean absolute lateness, shop utilization = 90%

it difficult for due date assignment methods to predict the job flowtime accurately and precisely.

Among the three due date assignment methods, the RTWK model appears to be the best with respect to MAL and MSL in all combinations of dispatching rule and shop utilization. That is to say, the performance of the RTWK model is not affected by the dispatching rules and the shop utilizations used. For MAL and MSL, the overall best performance is obtained with the RTWK model in conjunction with the EDD rule. Thus, it can be seen that the RTWK model in conjunction with the EDD rule provides the greatest improvement in respect to due date related performances.

A three factor ANOVA method with fixed factor levels is then applied to test the effects of the due date assignment method,

**Fig. 5.** Mean absolute lateness, shop utilization = 80%**Fig. 6.** Mean squared lateness, shop utilization = 90%**Fig. 7.** Mean squared lateness, shop utilization = 80%

dispatching rule, and shop utilization on the performance measures. As indicated by the statistics shown in Tables 8 and 9, all three main factors and their interactions have significant effects on both performance measures, except for the interaction of the due date assignment method and the dispatching rule on MSL, and the interaction of three factors on MSL beyond the prescribed  $\alpha$  ( $= 0.05$ ) level.

Because the main objective of this study was to study the relative effects of various due date assignment methods, and due date assignment methods are found to significantly affect both performance measures, a multiple range test was performed for comparing all due date assignment method means. Owing to the interaction of three factors being significant on MAL, results of Tukey multiple comparisons of the due date assignment methods in all combination of dispatching rule and shop utilization are shown in Table 10. The due date assignment rules in Table 10 are listed in descending order of performance, and they are grouped into homogeneous subsets that are indicated by underline as to where the difference between the means of MAL of two methods in the subset is not significant beyond the prescribed  $\alpha$  level. Based on the results in Table 10, the following observations can be made.

1. The RTWK model clearly outperforms both the TWK and DTWK methods in all combinations of dispatching rule and shop utilization. This illustrates that the RTWK model renders a more accurate job due date prediction.
2. Relative performances of the DTWK and TWK methods depend on the dispatching rule used. When the SPT rule is used,

**Table 10.** Tukey multiple range test for two performance measures of due date assignment methods ( $\alpha = 0.05$ )

Performance measure	Shop utilization	Dispatching rule	Due date assignment method		
MAL	90%	EDD	RTWK	DTWK	TWK
		SPT	RTWK	TWK	DTWK
		FCFS	RTWK	DTWK	TWK
	80%	EDD	RTWK	DTWK	TWK
		SPT	RTWK	TWK	DTWK
		FCFS	RTWK	DTWK	TWK

the TWK method is better than the DTWK method; in the opposite case, the DTWK method is better than the TWK method, except for the combination of the FCFS rule and 80% shop utilization.

Also, the interaction of the due date assignment method and the shop utilization significantly affects the performance of MSL. Tukey multiple comparisons of the due date assignment methods in each shop utilization rate are shown in Table 11. Based on the results in Table 11, the following observations can be made.

1. The RTWK model clearly provides the best performance under moderately heavy and heavy shop loads. This illustrates that the performance of the RTWK model is not affected by changes in shop utilization. The RTWK model renders more precise job due date prediction.

**Table 8.** ANOVA results for mean absolute lateness (MAL)

Source	Sum of Squares	df	Mean Square	F-Value	P-value
Due date assignment method (A)	221852.56	2	110926.28	848.93	0.000
Dispatching rule (B)	303336.18	2	151668.09	1160.72	0.000
Shop utilization (C)	273164.15	1	273164.15	2090.54	0.000
A * B	124798.62	4	31199.65	238.77	0.000
A * C	25724.88	2	12862.44	98.44	0.000
B * C	57482.50	2	28741.25	219.96	0.000
A * B * C	28761.04	4	7190.26	55.03	0.000
Error	21168.02	162	130.67		
Total	1056287.93	179			

**Table 9.** ANOVA results for mean squared lateness (MSL)

Source	Sum of Squares	df	Mean Square	F-Value	P-value
Due date assignment method (A)	2.89279E+11	2	1.44639E+11	12.35	0.000
Dispatching rule (B)	6.81885E+12	2	3.40943E+12	291.15	0.000
Shop utilization (C)	2.74368E+12	1	2.74368E+12	234.30	0.000
A * B	1.11324E+11	4	2.78310E+10	2.38	0.054
A * C	9.35473E+10	2	4.67736E+10	3.99	0.020
B * C	3.69872E+12	2	1.84936E+12	157.93	0.000
A * B * C	8.85623E+10	4	2.21406E+10	1.89	0.115
Error	1.89704E+12	162	1.17101E+10		
Total	1.57410E+13	179			

**Table 11.** Tukey multiple range test for MSL of due date assignment methods ( $\alpha = 0.05$ )

Performance measure	Shop utilization	Due date assignment method		
		RTWK	DTWK	TWK
MSL	90%	RTWK	DTWK	TWK
	80%	RTWK	DTWK	TWK

- The differences between the DTWK and the TWK methods are not significant at the two shop utilizations.

## 6 Conclusions

In this study, in order to improve the performance of TWK method, we presented a model that incorporated the data mining tool – Decision Tree – for mining the knowledge of job scheduling in a dynamic job shop environment, which was expressed in IF-THEN form, to estimate a more accurate and precise due date allowance factor  $k$  of the TWK method when the job arrives. The RTWK model is able to adjust the factor  $k$  of TWK rule for quoting job due date according to the condition of shop at the instant of job entry, to reduce the prediction errors.

We conducted a full factorial design with three factors (due date assignment method, dispatching rule, and shop utilization) to test the performance of our proposed rule-based TWK due date assignment model. The RTWK model was compared with its static and dynamic counterparts, i.e., the TWK and DTWK methods. The results indicated that the RTWK model outperforms the TWK and the DTWK methods with respect to the mean absolute lateness and the mean squared lateness in all combinations of dispatching rule and shop utilization. The Tukey multiple range test also reinforced the fact that the RTWK model is the overall best. The RTWK model clearly renders less lateness, and lesser deviation of lateness in due date setting. This shows that the RTWK model provides more accurate and precise job due date prediction than its static and dynamic counterparts. This also illustrates that the dynamic feedback of job characteristic and shop conditions information is useful in due date allowance factor estimating. The best due date related performance is obtained by a combination of the RTWK model and the EDD rule. Moreover, the RTWK model is quite robust against changing dispatching rule and shop utilization.

The findings of this study indicate that production managers should develop a due date assignment method that depends upon the characteristics of their own production system by using a data mining tool (i.e., Decision Tree). By analysing historical production data, the scheduling knowledge about due date assignment can be extracted and then be expressed in IF-THEN rules. This form of representing knowledge provides clear indications as to which factors are most influential in predicting the job due date,

and of how the job due date is affected by various levels of critical factors. At the same time, the scheduling knowledge can significantly improve the performance of the due date assignment process.

One of the future studies might want to focus on designing and then using an efficient method to segment the due date allowance factor  $k$  for enhancing the performance of the RTWK model. Moreover, the integration of other shop floor control strategies, such as order review/release being a decision factor, would be another worthwhile research topic.

## References

- Baker KR (1984) Sequencing rules and due-date assignment in a job shop. *Manage Sci* 30(9):1093–1104
- Baker KR, Bertrand JWM (1981) A comparison of due-date selection rules. *AIIE Trans* 13(2):123–131
- Chang F-CR (1994) A study of factors affecting due-date predictability in a simulated dynamic job shop. *J Manuf Syst* 13(6):393–400
- Cheng TCE (1984) Optimal due-date determination and sequencing of  $n$  jobs on a single machine. *J Oper Res Soc* 35(5):433–437
- Cheng TCE (1988) Optimal total-work-content-power due-date determination and sequencing. *Comput Math Appl* 14(8):579–582
- Cheng TCE, Jiang J (1998) Job shop scheduling for missed due-date performance. *Comput Ind Eng* 34(2):297–307
- Kanet JJ (1982) On anomalies in dynamic ratio type scheduling rules: a clarifying analysis. *Manage Sci* 28(11):1337–1341
- Sabuncuoglu I, Comlekci A (2002) Operation-based flowtime estimation in a dynamic job shop. *Omega – Int J Manage Sci* 30(6):423–442
- Fayyad UM (1997) Data mining and knowledge discovery in databases: implications for scientific databases. In: *Proceedings of the ninth International Conference on Scientific and Statistical Database Management*, pp 2–11
- Parpinelli RS, Lopes HS, Freitas AA (2002) Data mining with an ant colony optimization algorithm. *IEEE Trans Evol Comput* 6(4):321–332
- Fayyad UM, Piatetsky-Shapiro G, Smyth P (1996) From data mining to knowledge discovery: an overview. In: *Advances in Knowledge Discovery & Data Mining*, MIT Press, Cambridge, pp 1–34
- Tan KC, Tay A, Lee TH, Heng CM (2002) Mining multiple comprehensible classification rules using genetic programming. In: *Proceedings of the 2002 Congress on Evolutionary Computation*, Piscataway, NJ, USA, pp 1302–1307
- Congdon CB (2000) Classification of epidemiological data: a comparison of genetic algorithm and decision tree approaches. In: *Proceedings of the 2000 Congress on Evolutionary Computation*, Piscataway, NJ, USA, pp 442–449
- Han J, Kamber M (2001) *Data mining: Concepts and Techniques*. Morgan Kaufmann Publishers, San Francisco
- Michael JAB, Gordon L (1997) *Data mining techniques: for marketing, sales, and customer support*. John Wiley & Sons, New York
- Michael J (1997) *Machine Learning*. McGraw-Hill, New York
- Quinlan JR (1986) Induction of decision trees. *Mach Learn* 1(1):81–106
- Braha D, Shmilovici A (2002) Data mining for improving a cleaning process in the semiconductor industry. *IEEE Trans Semiconductor Manuf* 15(1):91–101
- Lawrence S (1984) Resource constrained project scheduling: an experimental investigation of heuristics scheduling techniques. Graduate School of Industrial Administration, Pittsburgh, Carnegie Mellon University
- Vig MM, Dooley KJ (1993) Mixing static and dynamic estimates for due date assignment. *J Oper Manage* 11(1):67–79