

Deep neural networks based order completion time prediction by using real-time job shop RFID data

Chuang Wang¹ · Pingyu Jiang¹

Received: 15 July 2016 / Accepted: 11 April 2017 / Published online: 19 April 2017
© Springer Science+Business Media New York 2017

Abstract In the traditional order completion time (OCT) prediction methods, some mutable and ideal production data (e.g., the arrival time of work in process (WIP), the planned processing time of all operations, and the expected waiting time per operation) are often used. Thus, the prediction time always deviates from the actual completion time dramatically even though the dynamicity of the production capacity and the real-time load conditions of job shop are considered in the OCT prediction method. On account of this, a new prediction method of OCT using the composition of order and real-time job shop RFID data is proposed in this article. It applies accurate RFID data to depict the real-time load conditions of job shop, and attempts to mine the mapping relationship between RFID data and OCT from historical data. Firstly, RFID devices capture the types and waiting list information of all WIPs which are in the in-stocks and out-stocks of machining workstations, and the real-time processing progress of all WIPs which are under machining at machining workstations. Secondly, a description model of real-time job shop load conditions is put forward by using the RFID data. Next, the mapping model based on the composition of order and real-time RFID data is established. Finally, deep belief network, which is one of the major technologies of deep neural networks, is applied to mine the mapping relationship. To illustrate the advantages of the proposed method, a numerical experiment compared with back-propagation (BP) network based prediction method, multi-hidden-layers BP network based prediction method and the principal com-

ponents analysis and BP network based prediction method is conducted at last.

Keywords Order completion time prediction · Deep neural networks · DBN · RFID

Introduction

The evolution of the competitive context in recent decades has led firms to face a more dynamic and uncertain environment where the main feature is the necessity of offering a higher and higher level of customization (Corti et al. 2006). Greater product variety forces the firms to shift from make-to-stock (MTS) to make-to-order (MTO) production. In MTO firms, before an order is released to the job shop for processing, its due-date needs to be assigned (Vinod and Sridharan 2011). Thus, due-date assignment is becoming more and more complex and significant. Providing customers with exact due-date is much better than a competitive price, because the deviations of actual due date (delay or early delivery) would result in increasing the product cost, decreasing the firm competitiveness (Gordon et al. 2002). The actual due date of order primarily depends on the order completion time (OCT) in the job shop. Thus, OCT prediction is the first important task in job shop control, and it is also a difficult decision. More importantly, accurate OCT prediction is also needed for better management of the job shop control activities, such as order review, contracts negotiation, product quote etc. (Liang et al. 2013).

A vast amount of literature exists on scheduling to meet due dates, but little work considers how to predict the OCT before setting these due dates (Moses et al. 2004). In practice, many traditional firms address this problem by establishing fixed lead times based on experience (Ziarnetzky and Mönch

✉ Pingyu Jiang
pjiang@mail.xjtu.edu.cn

¹ State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an, People's Republic of China

2016). And they do not consider the dynamicity of job shop load. However, manufacturing system operates in a dynamic environment where the unpredictable disturbances affecting the production schedule emerge at any time. Thus, the reliability of the assigned due date is relatively low. In theory, the prediction techniques of OCT can be distinguished by the methodology proposed: analytical and heuristic (Sun et al. 2013). The analytical approaches are usually based on the queue theory. They create an optimal schedule, and then set the order due dates equal to their scheduled completion dates (Moses et al. 2004; Hopp and Melanie 2001). However, the optimal completion time clearly cannot be computed in a large-scale systems context, because it is a NP-hard problem. The heuristic approaches, e.g., expert system, data mining, and neural networks, typically try to find an optimal value by heuristic procedures. They need not establish the analytical model of job shop. However, their performance is often dependent on a large number of training data.

Referring to the existing literature, the OCT prediction can be classified into three conceptual categories (Zorzini et al. 2008): (1) no job shop load analysis. The completion time prediction of order is based on average completion time, derived from similar products already realized in the past. (2) Aggregate job shop load based analysis. According to the hierarchical approach, aggregate information (from both the time and resource point of view) is used for the OCT prediction. They mostly consider the medium–long term and to the bottleneck resource. (3) Detailed shop load based analysis. The information about real-time job shop load conditions is used for OCT prediction. Although more and more works concentrate on detailed job shop load based analysis, they still use some mutable and ideal production data (e.g., the arrival time of work in process (WIP), the processing time of all operations, the waiting time per operation) to represent the detailed job shop load (Sabuncuoglu and Comlekci 2002; Vinod and Sridharan 2011). They view the job shop load conditions as static contexts, and predict the OCT based on the assumption that each production disturbance has a known stochastic distribution.

However, the production capacity of job shop is dynamically changed. Actually, there are too many uncertainties in a job shop (Wang and Jiang 2016). It is difficult to present an explicit analytic expression for the impact of those uncertainties to the completion time of order. However, the impact is implied in a large amount of historical information of job shop conditions. Thus, based on the historical information of the job shop load conditions, using intelligent approach to establish the prediction method of OCT is much more feasible. Previous studies clearly indicate that neural networks (NN) method is good at the OCT prediction. Using NN to predict the OCT has following merits (Hsu and Sha 2007): (1) NN can obtain a probable result even if the input data are incomplete or noisy. (2) A well-trained NN model

can provide a real-time forecasting result. (3) Creating a NN does not necessitate understanding the complex relationship among the input variables. Although NN has been introduced into the OCT prediction, it is not well applied in large-scale systems due to its shallow architecture which makes it have admittedly less capability with respect to inference mechanisms (Lopes and Ribeiro 2015). This architecture would lead to a large hidden layer and make the learning time scale poorly as the number of parameters increases in large-scale systems (Hinton et al. 2006).

The information of job shop load conditions is not usually available in traditional job shop. The advent of RFID technology has made obtaining the real-time information easy, but the real-time information captured by RFID is quite huge. Thus, in RFID-driven job shop, a new intelligent approach should be found to replace the NN in the OCT prediction. Theoretical and empirical evidence indicates that deep neural networks (DNN) are more efficient than NN in large-scale and/or high-dimensional systems (Bengio et al. 2007). Thus, it is very necessary to discuss the OCT prediction using DNN for RFID-driven job shop.

In this article, an RFID based descriptive model of real-time job shop load conditions is proposed. It depicts the real-time load conditions through the type and waiting list information of all WIPs which are in the in-stocks and out-stocks of machining workstations, and the real-time processing progress of all WIPs which are under machining at machining workstations. The RFID system can not only track the manufacturing progress of every order, but also provide a large number of historical production data. Furthermore, DNN is employed to learn the mapping relationship between the real-time job shop load conditions and OCT from the historical production data.

This article proposes an intelligent method for OCT prediction in RFID-driven job shop. It attempts to provide a credible OCT for firm managers when they negotiate the contracts with customers or assign the due-date of order. The method does not require an accurate analytical model of the manufacturing system, nor does it require the system to be operating in static state. This article combines the efforts of the following three aspects: (1) the real-time job shop load conditions are depicted by RFID data; (2) the OCT is predicted according to the information of WIPs such as type, waiting sequence, and real-time processing progress; and (3) DNN is introduced into OCT prediction to solve the deficiency of NN when is applied in large-scale systems.

The remainder of this article is organized as follows: “Background and motivation” section indicates three aspects of the background and motivation of our research. “The OCT prediction method of RFID-driven job shop” section describes the real-time job shop load conditions descriptive model of RFID-driven job shop. The DNN calculation method is depicted in “DBN based order completion time

prediction” section. In “Numerical experiment” section, a numerical experiment is taken as an example to illustrate the utility of the proposed method. “Conclusions” section summarizes the principal conclusions of this work and suggests areas of future research.

Background and motivation

Due dates assignment and OCT prediction

Due dates can be assigned either externally by the customer or internally by the firm. The methods of due date assignment for the firm include two ways: the scheduling approach and the completion time prediction approach. The scheduling approach is to create an optimal schedule and then set order due dates equal to their scheduled completion dates (Weng 1996; Hopp and Melanie 2001; Gordon and Strusevich 1999). Since due dates often need to be computed in a few seconds or less, full schedule optimization cannot be performed for an industrial-sized system (Moses et al. 2004). The completion time prediction approach sets the due date for the arriving/upcoming order to its completion date. Since Enns (1995) assigned due dates to orders based on the predicted completion time of orders, a number of articles have addressed the due dates problem from a completion time perspective. Vinod and Sridharan (2011) proposed a simulation modeling to predict the completion time of orders for a typical dynamic job shop production system. Hu et al. (2012) developed a prediction model of the order completion date based on the assumption that each production disturbance has a known stochastic distribution. Brahimi et al. (2014) took into account the load of job shop in the completion time prediction. Then they proposed a model which integrates production planning decisions together with order acceptance decisions. Lawrence (1995) approached the estimation of flow times as a forecasting problem, and used the empirical distribution of forecast errors to set the order due dates. Although some papers considered the load of job shop in the completion time prediction, they predicted the time only based on historical/empirical data, and neglected the current job shop load conditions. There has been little consideration of the order and the current shop load conditions as a whole in the completion time prediction method.

RFID and real-time information in OCT prediction

The OCT prediction heavily relies on the real-time statuses of various manufacturing resources. Moses et al. (2004) analyzed the difficulties of the real-time completion time prediction. They considered dynamic time-phased availability of resources required for each operation of the order when computing the completion time. Based on the platform of

ExSpec, a high-level Petri net simulation model for production process, Zhu et al. (2009) used the real-time state of workshop to predict the OCT. Sabuncuoglu and Comlekci (2002) utilized the detailed job shop and route information for operations of WIPs as well as the machine imbalance information to estimate the completion time by virtue of computer-integrated manufacturing systems (CIMS). From the above mentioned literatures, it can be seen that the real-time information of manufacturing resources is hardly captured. Capturing real-time information of manufacturing resources mainly depends on CIMS or simulation in the past. With the advent of the RFID, this information is now available. There is, however, a lack of systematic study of the OCT prediction for RFID-driven Job shop. The rigorous relevant researches have appeared as follows. Li et al. (2015) observed the real-time status of a one-of-a-kind production process accurately with the help of RFID. They studied due date assignment when the processing time is uncertain and normally distributed. Zhong et al. (2013) proposed a data mining approach to analysis the completion time prediction from the historical data of a RFID-enabled real-time job shop environment. They quantitatively examined the impact factors such as processing routine, batching strategy, scheduling rules and critical parameters of specification. Although these literatures discussed the OCT prediction in RFID-driven manufacturing environments, they did not point out the information acquisition method of job shop load conditions.

NN and DNN in completion time prediction

NN, as a heuristic approach, have been applied in the OCT prediction since it doesn't need the accurate analytical model of job shop. Hsu and Sha (2007) presented an artificial neural networks (ANN)-based due date assignment model. From the simulation and statistical results, the model performed better in due date prediction. Chen (2007, 2008) proposed a hybrid fuzzy c-means and back propagation network (BP) approach to enhance the effectiveness/accuracy of OCT prediction in a semiconductor fabrication factory. According to experimental results, the prediction accuracy of the proposed approach was significantly better than those of some existing approaches. Asadzadeh et al. (2011) presented a flexible algorithm for estimation and forecasting lead time based on ANN. The algorithm was used to estimate the weekly lead times of an actual assembly shop. The experiment showed that ANN is superior to other algorithm. Although there have been some literatures to predict the OCT based on NN, some problems in the application of NN are still not well solved: (1) the accuracy of the completion time prediction using NN loses when the load of job shop or the composition of orders changes. This results from that the input parameters do not reflect the real-time job shop conditions (Okubo et al. 2000);

(2) as mentioned previously, it is not well applied in large-scale systems due to their inherent drawback.

DNN can be used to deal with large data (Hinton and Salakhutdinov 2006). It has been successfully applied in full-sized, high-dimensional images inference, automatic speech recognition, human pose estimation, (Mohamed et al. 2012; Toshev and Christian 2014; Lee et al. 2009). In manufacturing domain, there have been some researches that use DNN. Tamilselvan and Wang (2013) presented a novel multi-sensor health diagnosis method using DNN for operation and maintenance of complex engineered systems. DNN were employed using vibration signal obtained from end milling to build feature space for cutting states monitoring (Fu et al. 2015). Keshmiri et al. (2015) presented a deep learning approach to estimation of the bead parameters in welding tasks. At present, although DNN has been introduced into manufacturing domain, there are few reports in RFID-driven job shop production information processing.

The OCT prediction method of RFID-driven job shop

The OCT prediction

The OCT prediction is always done at contracts negotiation stage without changing any of the existing production planning. Meanwhile, the production of the job shop has four principles: (1) machine tools select WIPs to process in a first come and first served (FCFS) manner; (2) each WIP requires a specific set of operations that need to be performed in a specified sequence (routing) on the machines; (3) each WIP type is allowed to have unique routing through the manufacturing system; (4) set-up times and transportation times are included in the processing times. Principles 1, 2 and 3 are satisfied in many, if not most, practical applications. Principle 4 is in line with production schedule. It's important to note that the dominant disturbances which occur after the OCT prediction (e.g., machine tools breakdown, arrival of urgent job) are not considered in the article.

For explanatory purposes, the OCT is defined as follows:

Definition 1 OCT is the last WIP completion time of the order in the job shop. It is affected by two qualitative factors, namely, the real-time load conditions of the job shop, and the composition of the order.

To realize the OCT, the two qualitative factors should be quantified. Here, OI is used to represent the quantified composition of the order, JS is used to represent the quantified real-time load conditions of the job shop. Therefore, based on OI and JS , the OCT can be formulated as follows:

$$ptv = f(OI, JS) \quad (1)$$

where ptv is the prediction value of OCT, f stands for the mapping function from OI and JS to ptv .

The quantization approaches of OI and JS are discussed as following sections.

The order composition quantization

To formulize a job shop, assuming the job shop consists of M machine tools and can produce N types of parts, any machine tool i can process an operation for n types of parts ($n \leq N$). The prediction method should consider individual order requirements and characteristics on a detailed level so that the accurate completion time can be calculated. Because the only difference between two orders is the number that the order demands for different part types, the composition of an order can be described as follows:

$$OI = \{NK_1, NK_2, \dots, NK_n, \dots, NK_N\} \quad (2)$$

where NK_n represents the order's demands for the n th type part number.

It should be noted that if the order does not demand for some part types, the corresponding numbers can be marked as zero.

The job shop real-time load conditions quantization

The real-time load conditions of job shop include machine tools and WIPs related information. This information can be captured by RFID devices which are deployed in the job shop.

The routing of a WIP in discrete systems has multiple operations. Each WIP should be processed on machine tools in FCFS manner. When an operation completed, the WIP would continually be conveyed to the next machine tool for processing until the last designated operation finished and it is qualified.

As a general rule, a machine tool would be configured with an operator, in-stock, out-stock and other tools (e.g., measuring devices, fixtures) in job shop. Thus, the machine tool with its auxiliaries can form a machining workstation (MW). For explanatory purposes, the MW is defined as follow:

Definition 2 MW is a manufacturing unit which can process a complete operation independently. It consists of an in-stock (IB), an out-stock (OB), and a machine tool (MT). Based on the set theory, the m th machining workstation can be written as $MW_m = \{IB_m, MT_m, OB_m\}$.

Note that the impact of machine operators to OCT is not considered in this article.

A WIP in a MW would go through three steps: (1) entering the in-stock waiting for processing; (2) being machined on the machine tool; and (3) entering the out-stock waiting

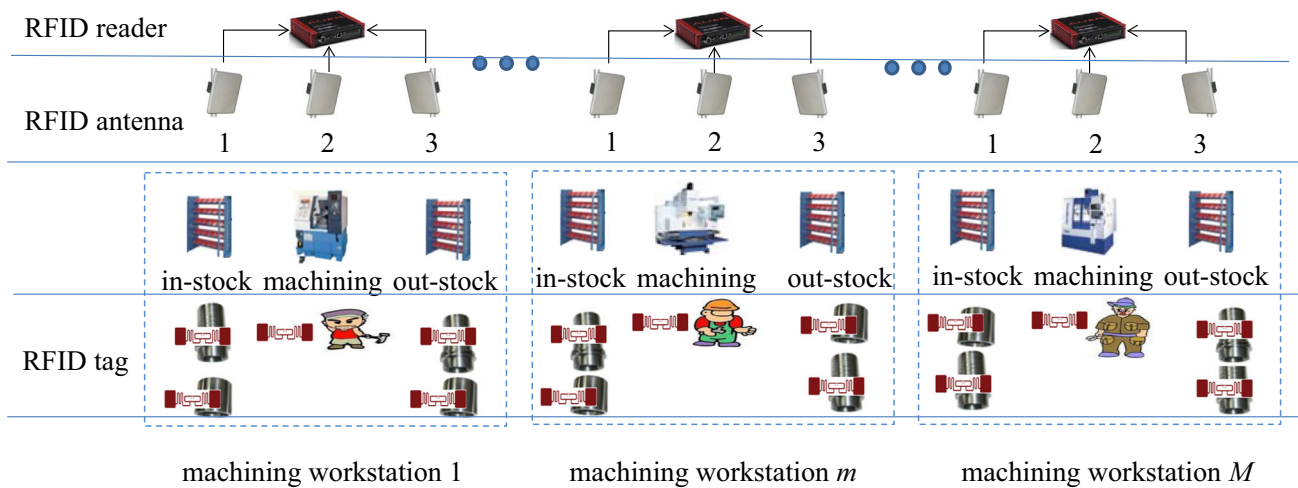


Fig. 1 The configuration of RFID devices in job shop

for transportation. Since a large portion of process time can be attributed to queuing for machining. Thus, the detailed information about each in-stock and out-stock such as WIPs types, WIPs numbers and the waiting list is very important for the OCT prediction (Yang et al. 2013). Although RFID can easily capture the identification code of WIPs, the RFID devices should be reasonably deployed in the job shop in order to get the location information of WIPs.

Based on the above analysis, the deployment of RFID devices in a MW can be described as Fig. 1. All WIPs should be labeled with RFID tags. A MW is configured with one RFID reader which has three RFID antennas. The RFID antenna 1 monitors whether the WIPs are in in-stock. The RFID antenna 2 is responsible for capturing the machining start time of the WIP. The RFID antenna 3 monitors whether the WIPs are in out-stock.

Since the time that a WIP waiting for processing is heavily affected by the waiting list in-stock, the prediction method should consider the detailed real-time condition of in-stock. Suppose there are p WIPs in the in-stock at current time t , the real-time condition of in-stock can be formulated as follows:

$$IB_m^t = \{PI_{m,1}^t, PI_{m,2}^t, \dots, PI_{m,p}^t\} \tag{3}$$

where $PI_{m,p}^t$ represents the p th WIP in in-stock at the m th MW at current time t .

When a WIP is in machining, it means that the machine tool has started to process the current operation of the WIP. Thus, the real-time condition of the machine tool can be described as follows:

$$MT_m^t = \{PM_m^t, RP_m^t\} \tag{4}$$

where PM_m^t represents the WIP in machining at the m th MW at current time t , RP_m^t stands for the real-time processing

progress of the current operation. Because RFID can capture the start time of current operation ST_m , the RP_m^t can be calculated as follows:

$$RP_m^t = t - ST_m \tag{5}$$

Now that the actual wait time of $PI_{m,p}^t$ can be formulated as follows:

$$WT_{m,p}^t = \sum_{a=1}^{p-1} PT_{m,a} + \sum_{a=1}^{p-1} (AT_{m,a}^t - PT_{m,a}) + (MPT_m - RP_m^t) \tag{6}$$

where $WT_{m,p}^t$ is the actual wait time of $PI_{m,p}^t$, $PT_{m,a}$ represents the planned processing time of WIP a at the m th MW. $AT_{m,a}^t$ stands for the actual processing time of WIP a at the m th MW, MPT_m represents the planned processing time of the WIP in machining at the m th MW.

If the type of WIP a is the n th part type, its routing is described as follows

$$MR_n^a = \{PT_{n,1}^1, PT_{n,2}^2, \dots, PT_{n,x}^y\} \tag{7}$$

where MR_n^a is the routing of WIP a which belongs to the n th type part, $PT_{n,x}^y$ represents the planned processing time of the x th operation of the n th type part at MW y , $y \in [1, M]$.

From Eqs. (6) and (7), it can be found that the planned processing time of a WIP at a MW is decided by the WIP type. Thus, Eq. (6) can be modified as follows

$$WT_{m,p}^t = \sum_{a=1}^{p-1} \sum_{n=1}^N \prod_{y=m} (MR_n^a)$$

$$\begin{aligned}
 & + \sum_{a=1}^{p-1} \sum_{n=1}^N \left(AT_{m,a}^t - \prod_{y=m} (MR_n^a) \right) \\
 & + \left(\prod_{y=m} (MR_n^{PM_m^t}) - RP_m^t \right). \tag{8}
 \end{aligned}$$

Because $AT_{m,a}^t$ cannot be obtained at the moment of the OCT prediction, $WT_{m,p}^t$ cannot be calculated accurately. Additionally, according to the principles mentioned above, there is a one-to-one correspondence between routings and WIP types. Thus, the wait time of $WT_{m,p}^t$ can be estimated according to the types of WIPs before it and the real-time processing progress of PM_m^t as follows

$$\begin{aligned}
 & WT_{m,p}^{t'} \\
 & = f_1 (PIT_{m,1}, PIT_{m,2}, \dots, PIT_{m,p-1}, PMT_m, RP_m^t) \tag{9}
 \end{aligned}$$

where $WT_{m,p}^{t'}$ is the estimate of $WT_{m,p}^t$. $PIT_{m,p-1}$ stands for the type of $PI_{m,p-1}^t$. PMT_m represents the type of PM_m^t . f_1 is the estimation function.

From Eq. (9), it can be seen that the wait time of a WIP is decided by the types of WIPs in in-stock, in machining, and the real-time processing progress of the current operation. Thus, the real-time condition of in-stock can be described by the types of WIPs and waiting list in in-stock, Eq. (3) can be transferred as follow

$$IB_m^t = \{PIT_{m,1}, PIT_{m,2}, \dots, PIT_{m,p}\}. \tag{10}$$

Similarly, Eq. (4) can be transferred as follow

$$MT_m^t = \{PMT_m, RP_m^t\}. \tag{11}$$

Since the types and waiting list of WIPs in out-stock would also affect the wait time of their next operation, similarly, the real-time condition of out-stock can be described as follow

$$OB_m^t = \{POT_{m,1}, POT_{m,2}, \dots, POT_{m,q}\} \tag{12}$$

where $POT_{m,q}$ represents the type of the q th WIP in out-stock at the m th MW.

Since a WIP can be in only one MW at any time, it is either in in-stock or in machining or in out-stock. And the real-time conditions of all MWs actually represent the current load conditions of job shop. Thus, the real-time conditions of the job shop at current time t can be quantified as follow

$$\begin{aligned}
 JS^t & = \{MW_1^t, MW_2^t, \dots, MW_m^t, \dots, MW_M^t\} \\
 & = \{ \{ \{ PIT_{1,1}, PIT_{1,2}, \dots, PIT_{1,p1} \}, \{ PMT_1, RP_1^t \}, \\
 & \quad \{ POT_{1,1}, POT_{1,2}, \dots, POT_{1,q1} \} \},
 \end{aligned}$$

$$\begin{aligned}
 & \dots, \\
 & \{ \{ PIT_{m,1}, PIT_{m,2}, \dots, PIT_{m,pm} \}, \{ PMT_m, RP_m^t \}, \\
 & \quad \{ POT_{m,1}, POT_{m,2}, \dots, POT_{m,qm} \} \}, \\
 & \dots, \\
 & \{ \{ PIT_{M,1}, PIT_{M,2}, \dots, PIT_{M,pM} \}, \{ PMT_M, RP_M^t \}, \\
 & \quad \{ POT_{M,1}, POT_{M,2}, \dots, POT_{M,qM} \} \}. \tag{13}
 \end{aligned}$$

Based on the discussion above, some information can be found from Eqs. (1), (2) and (13). (1) OCT can be predicted using the types and waiting list information of all WIPs in the job shop with the composition of the order. (2) Because the function f is unknown, NN or DNN can be used to establish the mapping function. (3) But Eq. (13) actually includes all WIPs in the job shop. The job shop real-time load conditions quantization JS , coupled with the order composition quantization OI , inevitably leads to high-dimensional input data of the mapping function. According to previous analysis about NN and DNN, DNN is more suitable for establishing the function.

DBN based order completion time prediction

Assuming that a job shop has accumulated a wealth of historical production data through RFID devices, which includes the information of orders, the corresponding job shop load conditions, $HI = \{hi_1, hi_2, \dots, hi_r\}$, and the actual completion time of the orders, $AT = \{at_1, at_2, \dots, at_r\}$. According to the previous discussion, the historical production data should include the following details

$$\begin{aligned}
 hi_r & = \{ \{ \{ NK_1^r, NK_2^r, \dots, NK_n^r, \dots, NK_N^r \}, \\
 & \quad \{ \{ \{ PIT_{1,1}^r, PIT_{1,2}^r, \dots, PIT_{1,p1}^r \}, \\
 & \quad \quad \{ PMT_1^r, RP_1^r \}, \{ POT_{1,1}^r, POT_{1,2}^r, \dots, POT_{1,q1}^r \} \} \\
 & \quad \dots, \\
 & \quad \{ \{ \{ PIT_{m,1}^r, PIT_{m,2}^r, \dots, PIT_{m,pm}^r \}, \\
 & \quad \quad \{ PMT_m^r, RP_m^r \}, \\
 & \quad \quad \{ POT_{m,1}^r, POT_{m,2}^r, \dots, POT_{m,qm}^r \} \} \\
 & \quad \dots, \\
 & \quad \{ \{ \{ PIT_{M,1}^r, PIT_{M,2}^r, \dots, PIT_{M,pM}^r \}, \\
 & \quad \quad \{ PMT_M^r, RP_M^r \}, \\
 & \quad \quad \{ POT_{M,1}^r, POT_{M,2}^r, \dots, POT_{M,qM}^r \} \} \} \tag{14}
 \end{aligned}$$

where hi_r is the r th historical production data.

Deep belief network (DBN) is one of the major technologies of DNN. Since proposed by Hinton et al. (2006), DBN has excelled in visual recognition and AI areas with notable

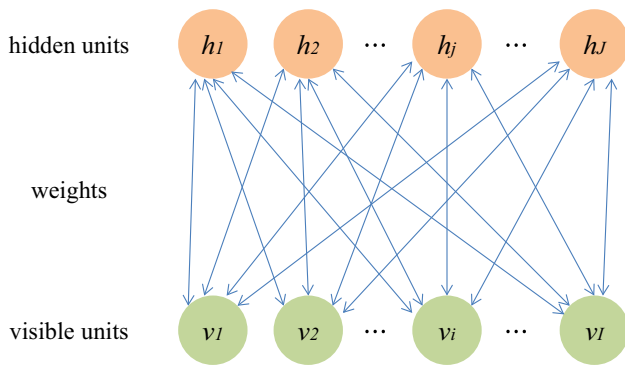


Fig. 2 Schematic representation of an RBM

success. A DBN is a feedforward neural network with a deep architecture, i.e., with many hidden layers. It consists of an input layer, a number of hidden layers and finally an output layer. The input layer accepts the input data and transfers the data to the hidden layers in order to complete the learning process (Tamilselvan and Wang 2013). The hidden layers are created by several layers of restricted Boltzmann machine (RBM) that are stacked on top of each other, thus forming a network that is able to capture the underlying regularities and invariances directly from the input data (Lopes and Ribeiro 2015).

To establish the function of OCT prediction, the historical production information HI can be used as the input data to train DBN.

Restricted Boltzmann machines

An RBM is a two-layer undirected bipartite graphical model where the first layer corresponds to inputs variables (visible units $\mathbf{v} = \{v_1, v_2, \dots, v_i, \dots, v_I\}$, $v_i \in \{0, 1\}$), and the second layer corresponds to the latent variables (hidden units $\mathbf{h} = \{h_1, h_2, \dots, h_j, \dots, h_J\}$, $h_j \in \{0, 1\}$). The visible and hidden layers are fully inter-connected via connections with symmetric undirected weights, but there are no intra-layer connections within either the visible or the hidden layer. A typical RBM model topology is shown in Fig. 2.

An RBM is an energy-based generative model. The weights and biases of the RBM determine the energy of a joint configuration of the hidden and visible units $E(\mathbf{v}, \mathbf{h})$ (Lopes and Ribeiro 2015; Sarikaya et al. 2011; Bengio et al. 2007),

$$\begin{aligned}
 E(\mathbf{v}, \mathbf{h}) &= -\mathbf{c}\mathbf{v}^T - \mathbf{b}\mathbf{h}^T - \mathbf{h}\mathbf{w}\mathbf{v}^T \\
 &= -\sum_{i=1}^I c_i v_i \\
 &\quad - \sum_{j=1}^J b_j h_j - \sum_{j=1}^J \sum_{i=1}^I w_{ji} v_i h_j
 \end{aligned}
 \tag{15}$$

where $\mathbf{w} \in \mathbb{R}^{J \times I}$ is a matrix containing the RBM connection weights, $\mathbf{c} = \{c_1, c_2, \dots, c_i, \dots, c_I\} \in \mathbb{R}^I$ is the bias of the visible units and $\mathbf{b} = \{b_1, b_2, \dots, b_j, \dots, b_J\} \in \mathbb{R}^J$ the bias of the hidden units.

The RBM assigns a probability to every possible visible-hidden vector pair via the energy function as follow:

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v}, \mathbf{h})}
 \tag{16}$$

where Z is a normalization constant called partition function by analogy with physical systems, which is obtained by summing over all possible pairs of visible and hidden vectors:

$$Z = \sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}
 \tag{17}$$

Since there are no connections between any two units within the same layer, given a particular random input configuration \mathbf{v} , all the hidden units are independent of each other and the probability of \mathbf{h} given \mathbf{v} becomes:

$$\begin{cases}
 p(\mathbf{h}|\mathbf{v}) = \prod_j p(h_j = 1|\mathbf{v}) \\
 p(h_j = 1|\mathbf{v}) = \sigma\left(b_j + \sum_{i=1}^I v_i w_{ji}\right)
 \end{cases}
 \tag{18}$$

where σ is the activation function, $\sigma = 1/1 + e^{-x}$.

Similarly given a specific hidden state \mathbf{h} , the probability of \mathbf{v} given \mathbf{h} is obtained as follows:

$$\begin{cases}
 p(\mathbf{v}|\mathbf{h}) = \prod_i p(v_i = 1|\mathbf{h}) \\
 p(v_i = 1|\mathbf{h}) = \sigma\left(c_i + \sum_{j=1}^J h_j w_{ji}\right)
 \end{cases}
 \tag{19}$$

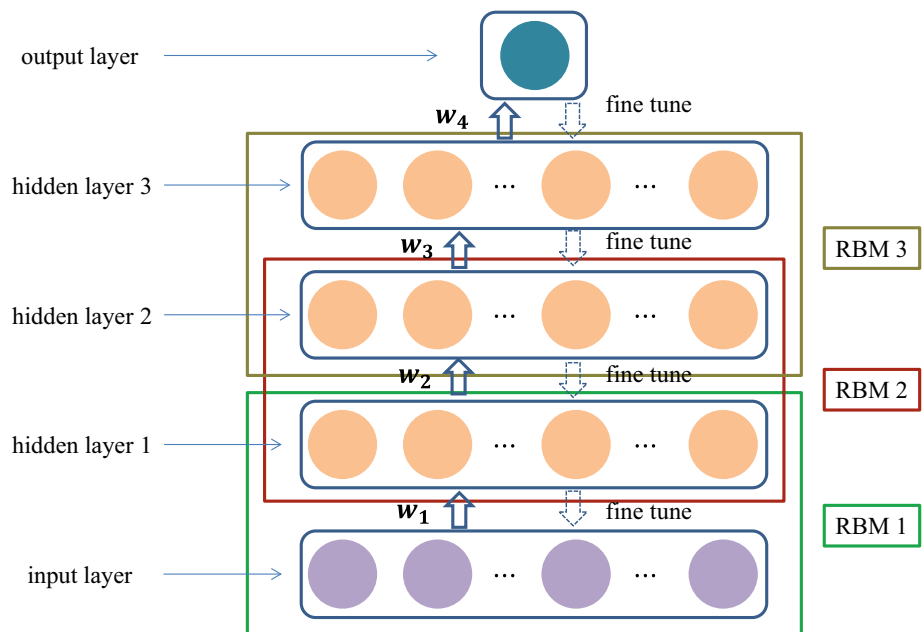
RBM's are usually trained by using the contrastive divergence (CD) learning procedure, which is described by Hinton (2002).

DBN architecture

DBN is produced by stacking RBMs. An example of DBN architecture is shown in Fig. 3, which consists of three stacked RBMs, as input layer and hidden layer 1 forms the first RBM, hidden layer 1 and hidden layer 2 forms the second RBM, and hidden layer 2 and hidden layer 3 forms the third RBM.

The training process of DBN is in a greedy manner (Hinton et al. 2006). Firstly, the first RBM that receives the DBN inputs is trained. Secondly, the second RBM that receives the first RBM outputs is trained. Thirdly, the third RBM that receives the second RBM outputs is trained, and so on (Lopes and Ribeiro 2015). Figure 4 demonstrates a DBN training

Fig. 3 Deep belief network architecture



process for OCT prediction based on the DBN architecture in Fig. 3. Considering the training process for the RBM 1 unit as shown in Fig. 4a, the input data HI is first given to the visible layer of this RBM unit. The next step is to transform the input data from the RBM visible layer to the hidden layer using visible layer parameters. While the training epoch reaches its maximum number and the training of the RBM 1 is accomplished, the hidden layer of this RBM unit becomes the visible layer of the RBM 2. The training process is continued for the second and the third RBM units as shown in Fig. 4b, c respectively. The training of the DBN is accomplished through the successive training of each individual RBM unit, as shown in Fig. 4d. This training approach represents an efficient way of learning by combining multiple and simpler RBM models, learned sequentially (Lopes and Ribeiro 2015). From the training process, also called pre-training, it can be seen that the layer-by-layer learning algorithm is unsupervised (Hinton et al. 2006).

After training the DBN, the trained weights of the RBM layers can be used to initialize the weights of a multi-layer feedforward neural network. However, learning the weight matrices one layer at a time is efficient but not optimal (Hinton et al. 2006).

Back-propagation learning

To improve the prediction accuracy of the DBN model, the back-propagation (BP) training algorithm is used to train the DBN after the pre-training. BP is the supervised learning process as Fig. 3 dotted arrow shows, which fine-tunes the weights initialized by pre-training (Tamilselvan and Wang 2013). In pre-training process, the corresponding actual completion time AT of the historical information HI

are useless. However, they would be used during the succeeding supervised learning process. Unlike pre-training process that considers one RBM at a time, BP considers all DBN layers simultaneously. BP uses the actual completion time AT for the training of the DBN model. The training error is calculated using DBN model outputs PTV and the actual completion time AT . Note that PTV is a set of OCT prediction values, i.e., $PTV = \{ptv_1, ptv_2, \dots, ptv_r\}$.

The weights of the DBN model are updated in order to minimize the training error (Bengio et al. 2007). The supervised learning process is continued until the network output reaches the maximum number of epochs.

Numerical experiment

The introduction of experiment environment and sample data

In order to show the applicability and superiority of the OCT prediction method, an RFID-driven job shop of a famous equipment manufacturing enterprise in China is chosen as an example. There are 12 MWs in the job shop. The configuration of each MW is listed in Table 1. The number under the length of IB_m (OB_m) in Table 1 represents the maximum number of WIPs in the in-stock (out-stock), which is decided by the production experience.

This job shop mainly produces 10 different kinds of parts. The information of the parts can be found in Table 2, where part number is the index of part type and the number under machine model represents the planned processing time of the part.

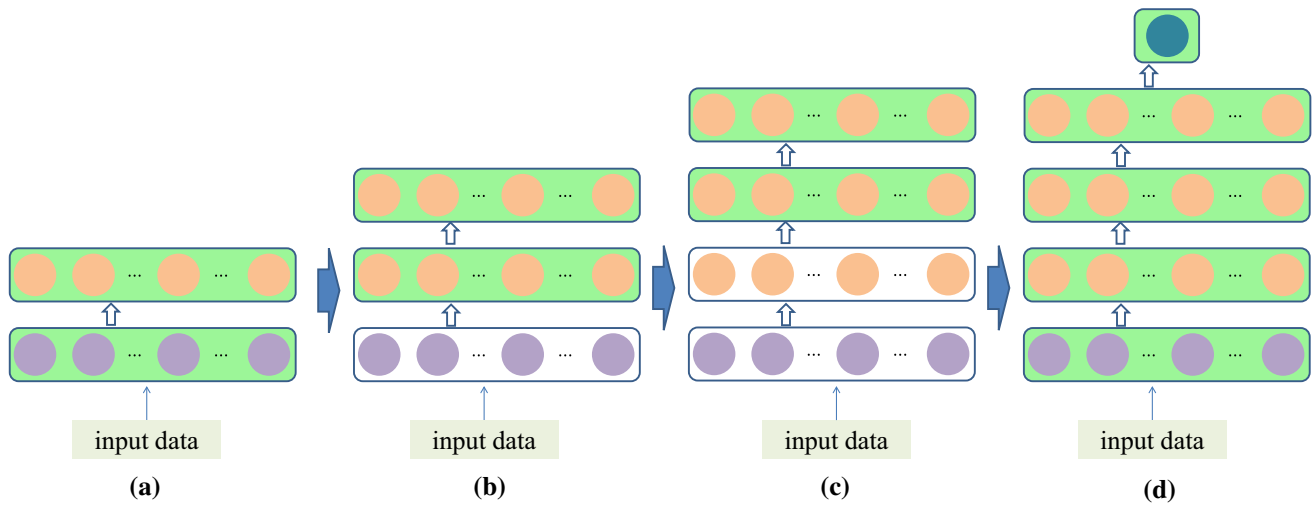


Fig. 4 Training process of a DBN

Table 1 The configuration information of each machining workstation in the job shop

MW_1			MW_2			MW_3		
Length (IB_1)	CNC lathe	Length (OB_1)	Length (IB_2)	CNC lathe	Length (OB_2)	Length (IB_3)	CNC lathe	Length (OB_3)
70	CKA6163L	10	70	CKA6163L	10	70	CKA6163L	10
MW_4			MW_5			MW_6		
Length (IB_4)	Boring	Length (OB_4)	Length (IB_5)	Boring	Length (OB_5)	Length (IB_6)	CNC milling	Length (OB_6)
10	TPX6113	10	10	TPX6113	10	10	XD-40A	20
MW_7			MW_8			MW_9		
Length (IB_7)	CNC milling	Length (OB_7)	Length (IB_8)	Processing center	Length (OB_8)	Length (IB_9)	Processing center	Length (OB_9)
10	XD-40A	20	10	DGMA1320	30	10	DGMA1320	30
MW_{10}			MW_{11}			MW_{12}		
Length (IB_{10})	Machining center	Length (OB_{10})	Length (IB_{11})	NC boring and milling	Length (OB_{11})	Length (IB_{12})	CNC milling	Length (OB_{12})
15	VDWA50	10	15	TK6516	10	15	XW2416	10

Table 2 The planned processing time of each kind of parts

Part number	Part type	Planned processing time (h)						
		CKA 6163L	TPX 6113	XD-40A	DGMA 1320	VDWA 50	TK 6516	XW 2416
1	Cylinder block	5.5	6.5	0	4	5.2	2.5	3
2	Cylinder head	3	0	2	1.5	0	4	5.1
3	Shell	6.1	3.5	5.2	4.1	0	2.5	1.5
4	Crankcase	4	5.5	2	0	3.5	0	1.5
5	Crankshaft	2	0	1	3	3.1	0	0
6	Piston	3	0	1.5	0	4	1	2
7	Flange disk	4	1	3	7	1	0	0
8	Shaft	7.5	3	0	0	6.1	2	5
9	Cylinder liner	5	0	0	4.2	1	0	1
10	Bearing ring	4.5	0	0	0	1.5	2	0

Table 3 The training and testing samples

Sample no.	$NK_{1, \dots, NK_{10}}$	$PIT_{1,1, \dots, PIT_{1,70}}$	PMT_{1, RP_1}	$POT_{1,1, \dots, POT_{1,10}}$...	$PIT_{6,1, \dots, PIT_{6,10}}$	PMT_{6, RP_6}
1	36,17,38,33,33, 14,10,34,34,30	1,1,6,6,7,7,7,7,7,8,8,8,8, 9,9,9,9,2,2,5,5,10,10,4,4	3,48	10	...	2,2,2,3,3	0
2	24,40,26,1,33, 9,30,16,40,36	4,7,8,8,9,9,2,5,10,4	4,42	10,10	...	4,3,3,3	4,114
3	30,4,9,6,32,37, 4,15,20,20	4,1,3,3,6,6,6,6,7,7,7,7,7,7,7,8,8, 8,8,9,9,9,9,9,9,2,2,5,5,10,10,4,4,4,4	4,204	0	...	5,3,3,3,5,5	5,18
4	27,7,8,35,5, 38,32,9,20,36	1,1,7,8,8,8,9,9,2,2,5,10,10,4,4	7,72	2	...	4,4,4,4	6,31
5	19,11,28,6,23, 12,9,23,18,19	1,1,6,6,6,6,6,7,7,7,7,7,7,7,7,8,8, 8,8,9,9,9,9,9,9,2,2,5,5,10,10,10,10, 4,4,4,4,3,3,3	1,120	6	...	6,6,6,6	0
...
1996	10,5,12,9,34, 19,22,14,34,14	1,7,8,8,2,5,10,4,9,9	4,35	8,5	...	4,6,6,4,4,4	4,121
1997	15,7,18,20,13, 20,10,33,16,40	1,1,8,8,8,8,9,9,9,9,2,2,5,5,10,10,4,4,6,6, 7,7,7,7,7,7	3,51	7	...	7,7,7,5,5	0
1998	37,32,13,27,34, 14,12,35,21,24	7,7,7,7,8,8,8,8,9,9,2,2,5,5,10,10,4,4,1,1,6,	9,162	4	...	2,2,2,5,5,5	0
1999	2,18,17,31,21, 5,25,10,33,15	1,1,5,5,10,10,10,4,4,4,4,4,1,1,6,6,6,6,6,7,7,7, 7,7,7,7,7,7,7,8,8,8,8,9,9,9,9,9,9,3,3,3	1,113	9	...	3,3,3,5,5	7,60
2000	13,18,14,35,10, 18,17,25,11,38	7,7,7,9,9,9,2,2,5, 10,10,4,4,8,8,8	7,69	5	...	5,5,5,6,6	6,24

Sample no.	$POT_{6,1, \dots, POT_{6,20}}$...	$PIT_{12,1, \dots, PIT_{12,15}}$	$PMT_{12, RP_{12}}$	$POT_{12,1, \dots, POT_{12,10}}$	<i>at</i>
1	6,6,6,4,4,4	...	6,6,8,8,2,2,2	0	1,1	1183,1
2	4,4,4,4,4,6,6,6,6,6,6,6	...	1,8,8,2,4	1,48	9,9,9	843,2
3	3,3,3	...	3,3,3,3,2	3,18	3,3,3	1043,4
4	6,6,6,6,6,4,4,4,4,4	...	6,1,1,3,8,8,2	6,84	6,6	969
5	2,2,2,2	...	3,3,3,6,2	0	4,4,1,1	939,5
...
1996	5,5,5,4,4,4,4,4,4, 4	...	1,8,8,2,4	4,42	4,4,4	716,7
1997	6,6,6,4,4,4,7,7	...	8,8,6,6,2,2,2	2,222	8,8,8	623,8
1998	3,3,3,3,4,4,4,4	...	4,3,6,6,8,8, 8,8,8,2,2	0	1,1,1	522,1
1999	3,3,3	...	3,3,6,6,2	4,54	3,3,3	633,6
2000	7,7,7,4,4,4,4,5,5,5	...	6,8,8,2,1,1,1	6,75	3,3	909,4

According to Tables 1 and 2; Eq. (14) can be detailed in this job shop as $M = 12, N = 10, p1 = 70, q1 = 10, p2 = 70, q2 = 10, p3 = 70, q3 = 10, p4 = 10, q4 = 10, p5 = 10, q5 = 10, p6 = 10, q6 = 20, p7 = 10, q7 = 20, p8 = 10, q8 = 30, p9 = 10, q9 = 30, p10 = 15, q10 = 10, p11 = 15, q11 = 10, p12 = 15, q12 = 10$. Thus, the dimensionality of hi_r is 529.

Based on the historical data of the job shop captured by RFID devices, the production information of 2000 orders is chosen as sample data, as listed in Table 3. Eighteen hundred data points are used as training samples and two hundred data points as test samples, i.e., the training samples: $HI = \{hi_1, hi_2, \dots, hi_{1800}\}$, $AT = \{at_1, at_2, \dots, at_{1800}\}$ and test samples: $HI' = \{hi'_1, hi'_2, \dots, hi'_{200}\}$, $AT' = \{at'_1, at'_2, \dots, at'_{200}\}$.

The experiment of DBN based OCT prediction

The appropriate number of hidden layers and units of each layer need to be determined before using DBN. Considering

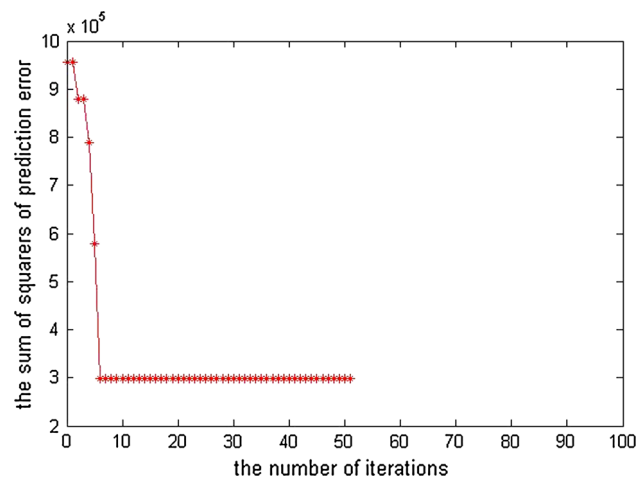


Fig. 5 The optimization process of the unit number for three hidden layers

genetic algorithm (GA) has been well studied, and its outstanding performance in parameter optimization has been proved by large amounts of practices, here, GA is used to find the appropriate parameters of DBN. According to the previous work and experience, the range of hidden layers number is set to [2, 4], and the range of the units number for each layer is set to [0, 200]. The sum of the squared errors (SSE) between the model outputs PTV and the actual completion time AT is used to as the GA fitness criteria. A simulation implementation is constructed based on GA toolbox in MATLAB R2013a. The simulation parameters are stochastic uniform selection, scattered crossover, constraint dependent mutation, a population size of 20. The stopping criteria are that the generations are 100 or the function tolerance is equal to 1×10^{-6} . The optimization process is shown in Fig. 5. From the optimization process, it can be found that the parameters of DBN reach the best after 52 iterations. The number of hidden layers is 3. And the corresponding unit number of three hidden layers is 106, 80, and 88, respectively.

According to the results of GA, the DBN model of the OCT prediction is designed as $529 \rightarrow 106 \rightarrow 80 \rightarrow 88 \rightarrow 1$. The training samples $HI = \{hi_1, hi_2, \dots, hi_{1800}\}$ are divided into 100 mini-batches to train the DBN. They are given as an input to the DBN for pre-training. Then, $HI = \{hi_1, hi_2, \dots, hi_{1800}\}$ and $AT = \{at_1, at_2, \dots, at_{1800}\}$ are used for BP learning. In the RBM learning process and back-propagation learning process, the number of training epochs is set to 1 in this experiment.

When the DBN model finishes the learning process, it can be used to predict the OCT according to the real-time job shop load conditions and the order information. Here, the test samples $HI' = \{hi'_1, hi'_2, \dots, hi'_{200}\}$ are used to demonstrate the performance of the DBN model. The prediction results of the DBN model for the test samples are depicted in Fig. 6 using black markers. It can be seen from the results that the prediction values of OCT is very close to the actual

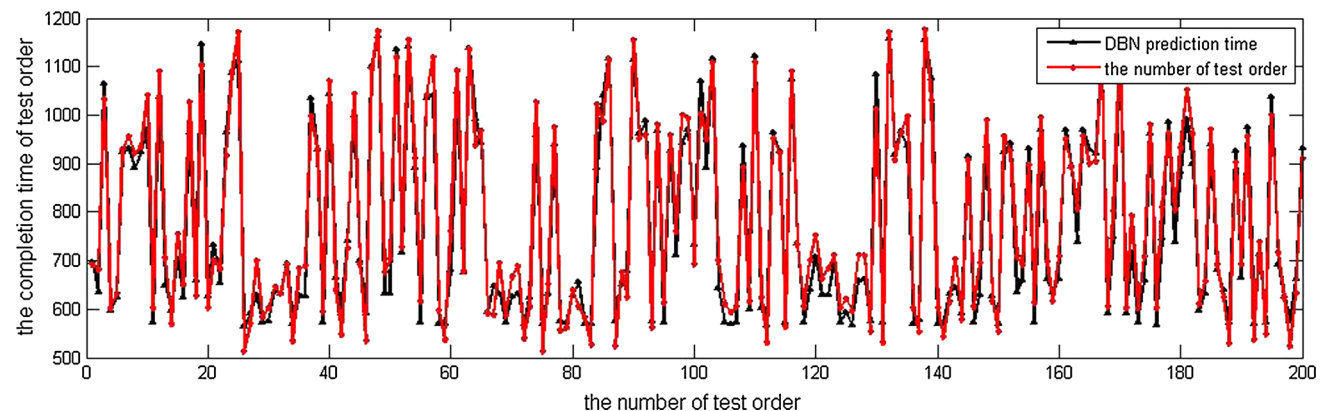


Fig. 6 The prediction results of the DBN model for the test samples

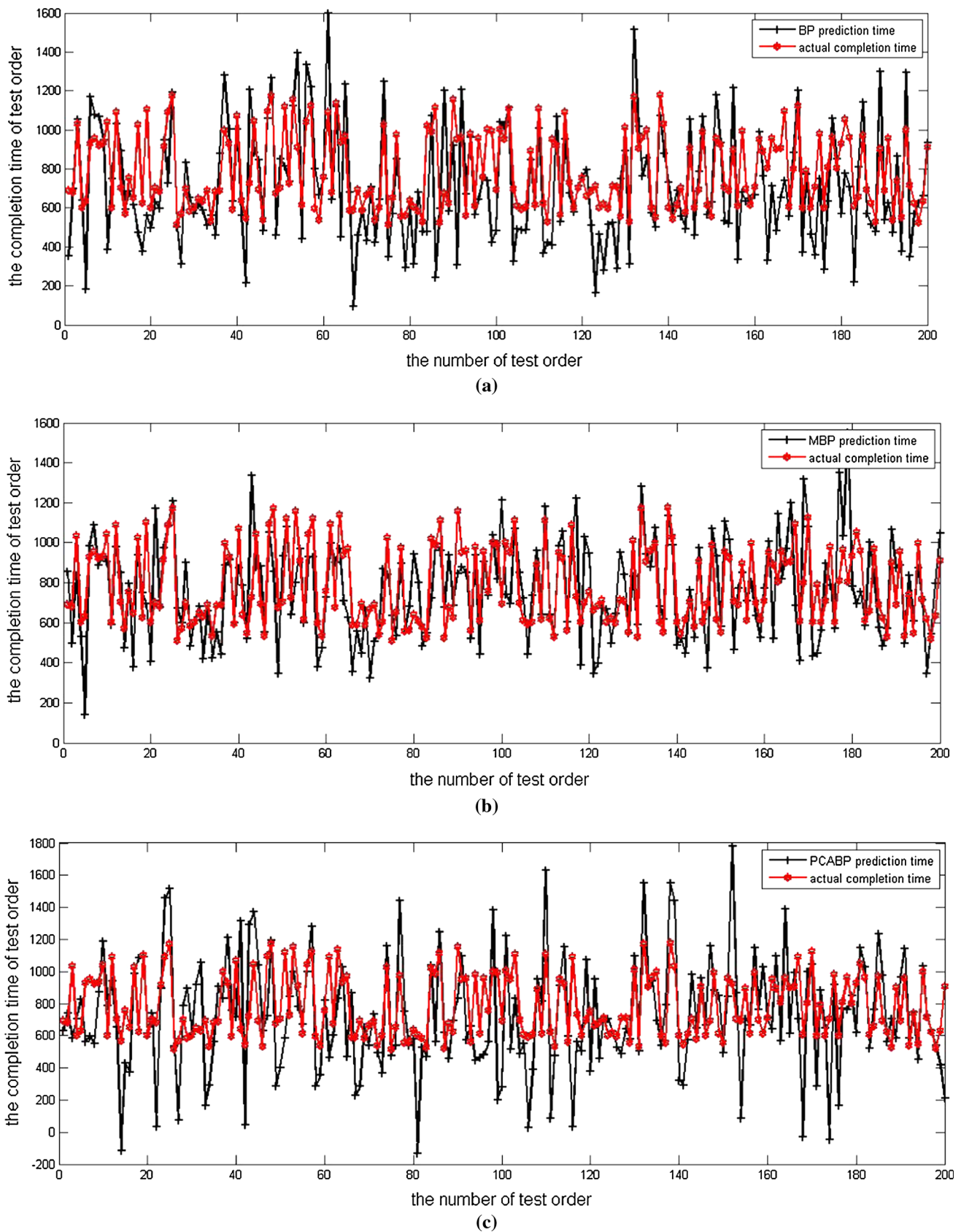


Fig. 7 The prediction results of three comparison methods. **a** The prediction results of the BP method for the test samples. **b** The prediction results of the MBP method for the test samples. **c** The prediction results of the PCABP method for the test samples

Table 4 The statistical analysis of different prediction results

Prediction method	DBN (Fig. 6)	BP (Fig. 7a)	MBP (Fig. 7b)	PCABP (Fig. 7c)
RMSE	38.611	229.658	195.762	279.344
MV (%)	4.39	22.47	19.47	31.48
SSE	2.982×10^5	1.054×10^7	7.665×10^6	1.561×10^7

completion time of the orders which is marked in red color, and the change direction and amplitude of black line is nearly the same as red line.

Comparison with other different methods

To illustrate the advantages of the proposed method, an experiment compared with other methods is conducted. According to the previous discussion, NN is the most widely acceptable method in the OCT prediction. Here, a BP networks with only one hidden layer is used as the first comparison method. The unit number of hidden layer is also optimized by GA, and the range is set to [0, 300] according to experience. And the number is finally set to 126 based on the optimization result. In order to compare the NN with DBN which is based on multi hidden layers, a BP networks with same network architecture as DBN, namely MBP, is used as the second comparison method. Hence, the network architecture of the BP and MBP can be given as $529 \rightarrow 126 \rightarrow 1$ and $529 \rightarrow 106 \rightarrow 80 \rightarrow 88 \rightarrow 1$ respectively.

Since the historical production information of the job shop is high-dimensional data, one possible method is to reduce its dimensionality first, and then use NN to predict the OCT. The principal components analysis (PCA), which is more popular in dimensionality reduction, is combined with BP to be used as the third comparison method, namely PCABP. Firstly, PCABP reduces the dimensionality of the training samples $HI = \{hi_1, hi_2, \dots, hi_{1800}\}$ and test samples $HI' = \{hi'_1, hi'_2, \dots, hi'_{200}\}$ together into lower dimensionality. Then, the low-dimensional data of the training samples and the corresponding actual completion time $AT = \{at_1, at_2, \dots, at_{1800}\}$ are used for BP learning. Finally, the low-dimensional data of the test samples is used by BP to predict the OCT.

All the methods proposed above have been implemented in MATLAB R2013a and run on Intel(R) Xeon(R) CPU E5-2630 v2 @2.60GHz, 192GB RAM, on Windows Server 2008 R2 datacenter. The training parameter of the BP accepted in three comparison methods is set as the goal 0.01, the epochs 500 and learning rate 0.01. The unit number of BP hidden layer in PCABP is changed into 50 according to the dimensionality reduction results of PCA and experiments. The prediction results of three comparison methods can be found in Fig. 7a–c, respectively.

The error between the prediction time and the actual completion time is the key factor, and it determines whether the prediction method of OCT can succeed. Here, the root mean square error (RMSE) of the prediction results is used to reflect the accuracy. The mean value (MV) of the relative errors and SSE are to represent their precision. The statistical analysis of different prediction results (see Figs. 6, 7) is shown in Table 4.

To minimize the variation of the results, ten replications are conducted for each prediction method. The experiment results are shown in Fig. 8.

Comparisons of ten experiment results are given in Table 5. The statistical analysis of ten experiments results (see Fig. 8), such as average RMSE, average MV, and average standard deviation (SD) of the relative errors, are used to judge the performance of different prediction methods. Additionally, the average training time and average decision time are used to reflect the responsiveness of different prediction methods.

A few conclusions can be seen from the experiment results:

1. DBN based prediction method reaches the highest accuracy and precision in all prediction methods. Besides, its training time and decision time is the shortest one. This means that the DBN based OCT prediction method can respond quickly to the change of a job shop load conditions.
2. Facing a huge amount of real-time production data, the training time of BP based prediction method is very long, and the accuracy and precision of prediction results are very low. Although the MBP based prediction method can improve the performance by adding hidden layers, the prediction results are still not good enough.
3. Using PCA to reduce the dimensionality of real-time production data can effectively shorten the training time and decision time of the BP based prediction method. However, the accuracy and precision of prediction results become lower since some useful information may be lost during the dimensionality reduction.
4. There exists the overfitting problem in BP based prediction method and PCABP based prediction method, as Fig. 8 shows. While DBN based prediction method can solve this problem successfully.

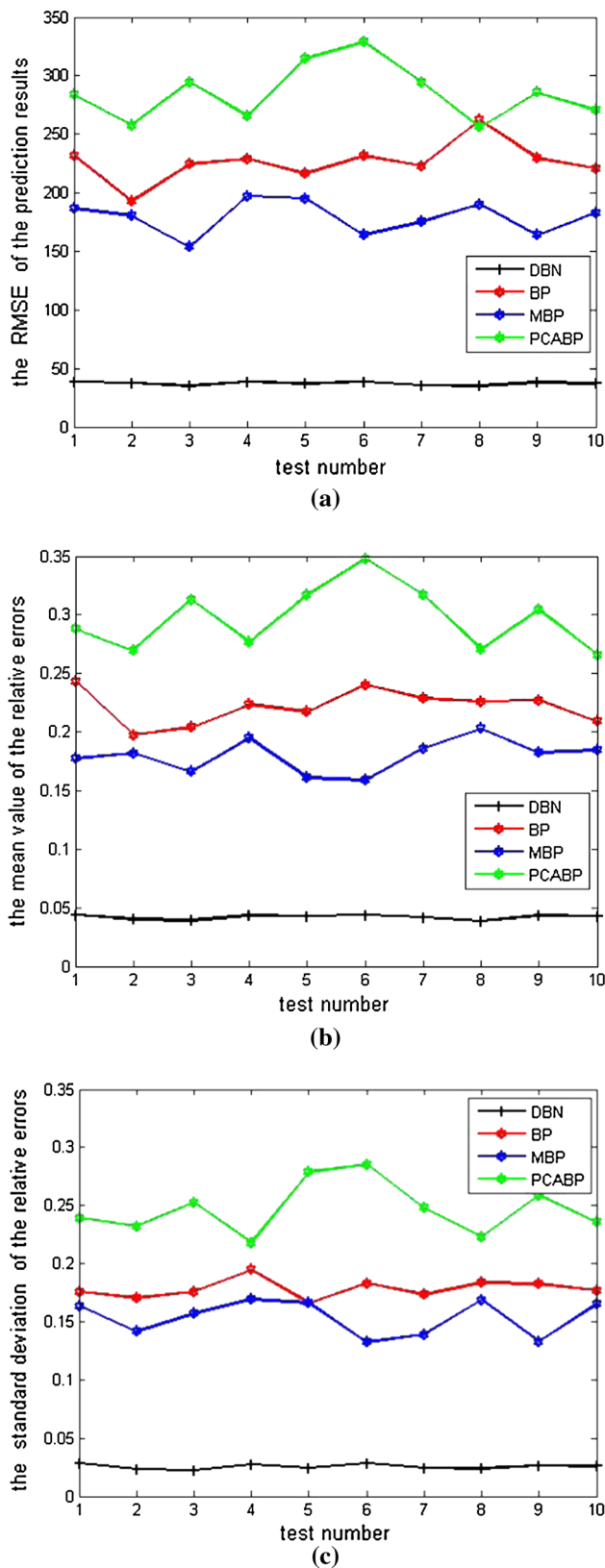


Fig. 8 The results of all prediction methods in 10 experiments. **a** The RMSE of the prediction results in 10 experiments. **b** The MV of the relative errors in 10 experiments. **c** The SD of the relative errors in 10 experiments

Table 5 The performance of different prediction methods in 10 experiments

Prediction method	DBN	BP	MBP	PCABP
Average training time CPU time (s)	0.7312	35.0129	23.5476	2.7153
Average decision time CPU time (s)	0.0183	17.3725	21.627	0.0704
Average RMSE	37.103	225.804	178.922	285.071
Average MV (%)	4.19	22.16	18.0	29.69
Average SD	0.025	0.178	0.153	0.247

Conclusions

This article focuses on the OCT prediction using real-time RFID data of job shop load conditions. A DBN based prediction method of OCT is proposed. Firstly, RFID devices capture the types and waiting list of all WIPs which are in in-stocks and out-stocks and the real-time processing progress of all WIPs which are in machining at all MWs, and the real-time job shop load conditions is presented by those RFID data. Next, the prediction model is established by using the composition of orders and the real-time RFID data. Finally, based on historical production data of the job shop, a DBN based prediction method is trained to predict the OCT.

A numerical experiment based on real production data of an RFID-driven job shop is used to verify the performance of the DBN based prediction method. Additionally, the advantages of the proposed method are fully demonstrated by comparing with BP based prediction method, MBP based prediction method, and PCABP based prediction method.

From the experiments conducted, the following important managerial implications have been drawn.

1. This study provides a credible prediction value of OCT by using real-time RFID data of job shop load conditions. It can help firm managers to choose suitable order based on time and/or cost of production.
2. From the experiment results, it can be observed that DNN can be successfully applied into the OCT prediction. DNN can not only solve the problems that NN applied in large-scale job shop would arise, but also decrease the information loss that PCA would result in during the dimensionality reduction.

3. Historical production data is very important and critical to improve management and decision-making in job shop. The decision-making which depends on the firm managers' experience and knowledge can also be realized through historical production data analysis.

However, the method is in an early stage of implementation and has limitations for some applications:

1. The prediction value of OCT in this article is a reference for firm managers when they negotiate with the customers about business. The actual due date of the order should be greater than the prediction value in order to cope with unforeseen problems in production.
2. RFID in this article is only used to realize the automatic acquisition of the job shop real-time load conditions. For other acquisition methods, such as barcode, smart sensors, the model of OCT need to be modified correspondently.
3. The proposed method can do well in a stable manufacturing system. And frequent disturbances would debase its effect.

Future work in this area includes three aspects: (1) the impact of different operators to OCT should be taken into account; (2) the production disturbances captured by RFID can be used to improve the robustness of the proposed method; (3) the experience of the firm managers can be used in the prediction of OCT.

Acknowledgements The research work presented in this article is under the support of National Natural Science Foundation of China with Grant No. 51275396.

References

- Asadzadeh, S. M., Azadeh, A., & Ziaefar, A. (2011). A neuro-fuzzy-regression algorithm for improved prediction of manufacturing lead time with machine breakdowns. *Concurrent Engineering*, 19(4), 269–281. doi:10.1177/1063293X11424512.
- Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2007). Greedy layer-wise training of deep networks. *Advances in Neural Information Processing Systems*, 19, 153–160.
- Brahimi, N., Aouam, T., & Aghezzaf, E. (2014). Integrating order acceptance decisions with flexible due dates in a production planning model with load-dependent lead times. *International Journal of Production Research*, 53(12), 3810–3822.
- Chen, T. (2007). Incorporating fuzzy c-means and a back-propagation network ensemble to job completion time prediction in a semiconductor fabrication factory. *Fuzzy Sets and Systems*, 158(19), 2153–2168. doi:10.1016/j.fss.2007.04.013.
- Chen, T. (2008). A hybrid fuzzy-neural approach to job completion time prediction in a semiconductor fabrication factory. *Neurocomputing*, 71(16–18), 3193–3201. doi:10.1016/j.neucom.2008.04.040.
- Corti, D., Pozzetti, A., & Zorzini, M. (2006). A capacity-driven approach to establish reliable due dates in a MTO environment. *International Journal of Production Economics*, 104(2), 536–554. doi:10.1016/j.ijpe.2005.03.003.
- Enns, S. T. (1995). A dynamic forecasting model for job shop flowtime prediction and tardiness control. *International Journal of Production Research*, 33(5), 1295–1312.
- Fu, Y., Zhang, Y., Qiao, H., Li, D., Zhou, H., & Leopold, J. (2015). Analysis of feature extracting ability for cutting state monitoring using deep belief networks. *Procedia CIRP*, 31, 29–34. doi:10.1016/j.procir.2015.03.016.
- Gordon, V. S., & Strusevich, V. A. (1999). Earliness penalties on a single machine subject to precedence constraints: SLK due date assignment. *Computers and Operations Research*, 26(2), 157–177.
- Gordon, V., Proth, J., & Chu, C. (2002). A survey of the state-of-the-art of common due date assignment and scheduling research. *European Journal of Operational Research*, 139(1), 1–25. doi:10.1016/S0377-2217(01)00181-3.
- Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8), 1771–1800.
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 504–507.
- Hinton, G. E., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7), 1527–1554.
- Hopp, W. J., & Melanie, R. S. (2001). A simple, robust leadtime-quoting policy. *Manufacturing and Service Operations Management*, 3(4), 321–336.
- Hsu, S. Y., & Sha, D. Y. (2007). Due date assignment using artificial neural networks under different shop floor control strategies. *International Journal of Production Research*, 42(9), 1727–1745. doi:10.1080/00207540310001624375.
- Hu, S., Zhang, B., & Zhang, X. (2012). Order completion date estimation and due date decision under make-to-order mode. *Industrial Engineering Journal*, 15(3), 122–129.
- Keshmiri, S., Zheng, X., Chew, C. M., & Pang, C. K. (2015). Application of deep neural network in estimation of the weld bead parameters. [arXiv:1502.4187](https://arxiv.org/abs/1502.4187).
- Lawrence, R. S. (1995). Estimating flowtimes and setting due-dates in complex production systems. *IIE Transactions*, 27(5), 657–668.
- Lee, H., Grosse, R., Ranganath, R., & Ng, A. Y. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. *Paper presented at the proceedings of the 26th annual international conference on machine learning*.
- Li, M., Yao, L., Yang, J., & Wang, Z. (2015). Due date assignment and dynamic scheduling of one-of-a-kind assembly production with uncertain processing time. *International Journal of Computer Integrated Manufacturing*, 28(6), 616–627.
- Liang, F., Fung, R. Y., & Jiang, Z. (2013). A comined approach of cycle time estimation in mass customization enterprise. *International Journal of Industrial Engineering*, 20(9), 574–588.
- Lopes, N., & Ribeiro, B. (2015). Deep belief networks (DBNs). In J. Kacprzyk (Ed.), *Machine learning for adaptive many-core machines—a practical approach* (pp. 155–186). Switzerland: Springer.
- Mohamed, A., Dahl, G. E., & Hinton, G. (2012). Acoustic modeling using deep belief networks. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1), 14–22. doi:10.1109/TASL.2011.2109382.
- Moses, S., Grant, H., Gruenwald, L., & Pulat, S. (2004). Real-time due-date promising by build-to-order environments. *International Journal of Production Research*, 42(20), 4353–4375.
- Okubo, H., Weng, J., Kaneko, R., & Simizu, T. (2000). *Production lead-time estimation system based on neural network*. Research paper.
- Sabuncuoglu, I., & Comlekci, A. (2002). Operation-based owtime estimation in a dynamic job shop. *Omega*, 30(6), 423–442.
- Sarikaya, R., Hinton, G. E., & Ramabhadran, B. (2011). Deep belief nets for natural language call-routing. *Paper presented at the 2011*

- IEEE international conference on acoustics, speech and signal processing.*
- Sun, D., Shi, H., & Chang, L. (2013). Application of support vector regression in prediction of application of support vector regression in prediction of due date under uncertain assemble-to-order environment. *Journal of Computer Applications*, 8, 2362–2365.
- Tamilselvan, P., & Wang, P. (2013). Failure diagnosis using deep belief learning based health state classification. *Reliability Engineering and System Safety*, 115, 124–135. doi:10.1016/j.res.2013.02.022.
- Toshev, A., & Christian, S. (2014). DeepPose: Human pose estimation via deep neural networks. *Paper presented at the 2014 IEEE conference on computer vision and pattern recognition (CVPR).*
- 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. *International Journal of Production Economics*, 129(1), 127–146. doi:10.1016/j.ijpe.2010.08.017.
- Wang, C., & Jiang, P. (2016). Manifold learning based rescheduling decision mechanism for recessive disturbances in RFID-driven job shops. *Journal of Intelligent Manufacturing*. doi:10.1007/s10845-016-1194-1.
- Weng, Z. K. (1996). Manufacturing lead times, system utilization rates and lead-time-related demand. *European Journal of Operational Research*, 89(2), 259–268.
- Yang, S., Lee, H., & Guo, J. (2013). Multiple common due dates assignment and scheduling problems with resource allocation and general position-dependent deterioration effect. *The International Journal of Advanced Manufacturing Technology*, 67(1–4), 181–188. doi:10.1007/s00170-013-4763-x.
- Zhong, R. Y., Huang, G. Q., Dai, Q., & Zhang, T. (2013). Estimation of lead time in the RFID-enabled real-time shopfloor production with a data mining model. *Paper presented at the The 19th international conference on industrial engineering and engineering management.*
- Zhu, H., Liu, F., Liu, Q., & Shao, X. U. (2009). A predictive method for order due date based on real-time state of workshop. *China Mechanical Engineering*, 3, 300–304.
- Ziarnetzky, T., & Mönch, L. (2016). Incorporating engineering process improvement activities into production planning formulations using a large-scale wafer fab model. *International Journal of Production Research*, 54(21), 6416–6435.
- Zorzini, M., Corti, D., & Pozzetti, A. (2008). Due date (DD) quotation and capacity planning in make-to-order companies: Results from an empirical analysis. *International Journal of Production Economics*, 112(2), 919–933. doi:10.1016/j.ijpe.2007.08.005.