Hindawi
Complexity
Volume 2019, Article ID 1891365, 12 pages
https://doi.org/10.1155/2019/1891365

Research Article

Model Predictive Control of Robotic Grinding Based on Deep Belief Network

Shouyan Chen[1], **Tie Zhang**[2], **Yanbiao Zou**[2] **and Meng Xiao**[2]

[1]*Guangzhou University, China*
[2]*South China University of Technology, China*

Correspondence should be addressed to Shouyan Chen; 15989285002@sina.cn

Received 10 November 2018; Revised 10 February 2019; Accepted 25 February 2019; Published 27 March 2019

Academic Editor: Sing Kiong Nguang

Considering the influence of rigid-flexible dynamics on robotic grinding process, a model predictive control approach based on deep belief network (DBN) is proposed to control robotic grinding deformation. The rigid-flexible coupling dynamics of robotic grinding is first established, on the basis of which a robotic grinding prediction model is constructed to predict the change of robotic grinding status and perform feed-forward control. A rolling optimization formula derived from the energy function is also established to optimize control output in real time and perform feedback control. As the accurately model parameters are hard to obtain, a deep belief network is constructed to obtain the parameters of robotic grinding predictive model. Simulation and experimental results indicate that the proposed model predictive control approach can predict abrupt change of robotic grinding status caused by deformation and perform a feed-forward and feedback based combination control, reducing control overflow and system oscillation caused by inaccurate feedback control.

1. Introduction

The deformation occurs during robotic grinding process has significant impact on robotic grinding dynamic and robotic grinding performance [1, 2]. Two ways are mainly presented in current studies to solve this problem. One way is to optimize mechanical structure of robotic machining system or increase stiffness and stability of robot machining system. The other is to adjust machining trajectory by off-line planning or real-time force control according to the robot dynamic model [3].

The real-time force control approaches presented in current studies include adaptive control, fuzzy control, and control based on neural network [4]. Mendes et al. [5] proposed an adaptive fuzzy control approach, which is based on Hybrid force/motion control system, to cope with contact issues between robot and a given surface. Fu et al. [6] proposed an adaptive fuzzy force control model, which includes a speed control loop and a position control loop to control both feed rate and position of robot, to achieve stable robotic deburring control. Yen [7] proposed an adaptive control method based on recursive fuzzy wavelet neural network to optimize motion control parameters of three-axis robot in real time.

The above feedback control approaches implement only when trajectory deviations appear, which may result in overshoot, control overspill, and system oscillations. In view of this, some scholars attempt to implement model predictive control approach to achieve feed-forward compensation control [8, 9]. Many nonlinear model predictive control approaches are then proposed. Wilson [10] discussed the performances of three model predictive control approaches applied to robot system control. The three approaches are nonlinear model predictive control (nMPC) approach, PID-based nonlinear model predictive control (PID nMPC) approach, and simplified nonlinear model predictive control (SnMPC) approach. The results of discussion indicated that the performance of nonlinear model predictive control approach is susceptible to system model errors. Some scholars try to improve nonlinear model predictive control approach by using intelligent algorithms such as neural network [11–13]. Li [14] proposed a nonlinear model control method based on neural dynamic network, where the neural dynamic network is used to obtain optimal values of the formulated

constrained quadratic programming (QP) problem derived from the cost function of nonlinear model predictive control model. Zeng [15] used Gaussian radial basis function (RBF) neural networks to improve the nonlinear model predictive control approach applied in the control of nonlinear multivariable systems. Dalamagkidis [16] proposed a nonlinear model predictive control approach based on recurrent neural network to achieve the predictive control of propeller self-rotation process while unmanned aerial vehicle engine is damaged.

In this paper, a model predictive control approach based on a deep belief network (DBN) is proposed to control robotic deformation and reduce rigid-flexible effect on robotic grinding dynamics. The following parts are arranged as follows: Firstly, the dynamic model of robotic grinding is established with the consideration of rigid-flexible coupling effect. Based on this, the model predictive controller of robotic grinding is designed. Since the accurate parameters of robotic grinding dynamics model and model predictive controller are hard to acquire, a deep belief network is designed to access nonlinear predictive model of robotic grinding. Simulation and experiments are finally carried out to verify performance of the proposed approach.

2. Rigid-Flexible Coupling Dynamics of Grinding Robot

Traditional grinding dynamics model can be expressed as [17]

$$M\ddot{x}_p + C\dot{x}_p + K_d x_p = F_p \tag{1}$$

where F_p is the cutting force; M is the system mass matrix; C is the system damping; K_d is the system dynamic stiffness; x_p is the grinding tool position; and $\dot{x}_p$, $\ddot{x}_p$ are its first and second derivatives. Since the stiffness of CNC is large and the deformation is small, the grinding tool position is approximate to the planned position. However, the stiffness of robot is not sufficient which may lead to large deformation and large deviation between grinding position and planned position. Therefore, the relationship of the grinding position x_p, the planned position x_a, and the deformation σ can be expressed as

$$x_p = x_a - \sigma \tag{2}$$

Similarly, the relationship of the force acts to robot end-effector F, the cutting force F_p, and the force caused by robotic grinding deformation F_d is

$$F = F_d + F_p \tag{3}$$

2.1. Robotic Grinding Deformation. Robotic grinding deformation consists of extrusion deformation and periodic deformation. The extrusion deformation is caused by relative motion between grinding tool and workpiece, while the periodic deformation is caused by relative motion between blades and workpiece. Therefore the grinding deformation can be expressed as

$$\sigma = \sigma_j(t_i) + \sigma_c(t_i) \tag{4}$$

where $\sigma_j(t_i)$ is the extrusion deformation; $\sigma_c(t_i)$ is the periodic deformation. The generation of extrusion deformation is shown in Figure 1. The grinding tool is driven toward the workpiece at a feed rate of V_w to perform cutting. According to traditional grinding theory, there are sliding and extrusion state before the actual grinding is conducted. Based on this, an assumption is made that the deformation generated when the grinding tool cut-in workpiece is mostly derived from extrusion deformation, written as

$$F_j(t_i) = K_d \sigma_j(t_i) \tag{5}$$

where K_d is the dynamic stiffness matrix of robot grinding system; $\sigma_j(t_i)$ is the extrusion deformation at time t_i. The value of extrusion deformation can be regarded as the accumulation of the difference between the feed rate $V_w(t_j)$ and the removal speed $V_e(t_j)$ from time t_1 to time t_i, written as

$$\sigma_j(t_i) = \sum_{h=1}^{i} \left(V_w(t_h) - V_e(t_h)\right) \tag{6}$$

Similarly, the relationship between periodic deformation force and the periodic deformation $\sigma_c(t_i)$ can be expressed as

$$F_c(t_i) = K_d \sigma_c(t_i) \tag{7}$$

$$\sigma_c(t_i) = \sigma_{co}(t_i) \sin\left(\omega_c t_i + \varphi_c\right) \tag{8}$$

where $\omega_c = 2\pi f_c = 2\pi n V_c/60$ is the circular frequency of grinding tool; n is the number of blades,; V_c is the tool rotating speed; φ_c is the corresponding phase; $\sigma_{co}(t_i)$ is the corresponding maximum cutting deformation. Therefore, the actual grinding position can be obtained by (6) and(8):

$$x_p(t_i) = x_a(t_i) - \sigma_j(t_i) - \sigma_c(t_i) \tag{9}$$

Substituting (9) and (1) into (3), the dynamic model of robot grinding can be expressed as

$$\begin{aligned} F(t_i) = {} & M\ddot{x}_p(t_i) + C\dot{x}_p(t_i) + K_d x_p(t_i) \\ & + K_d\left(\sigma_c(t_i) + \sigma_j(t_i)\right) \end{aligned} \tag{10}$$

For the convenience of discussion, $F(t_i)$ is named as robot grinding force in the following text.

3. Model Predictive Control Based on Deep Belief Network

3.1. Problem Description. During robotic grinding, the actual grinding position $x_p(t_i)$ is expected to be approximate to the planned position $x_a(t_i)$

$$\lim_{i \longrightarrow \infty} \left(x_p(t_i) - x_a(t_i)\right) = 0 \tag{11}$$

However, according to (10), as the grinding robot rigidity is insufficient and the robotic grinding deformation is considerable, (11) is hard to realize. Therefore, an ideal deformation is

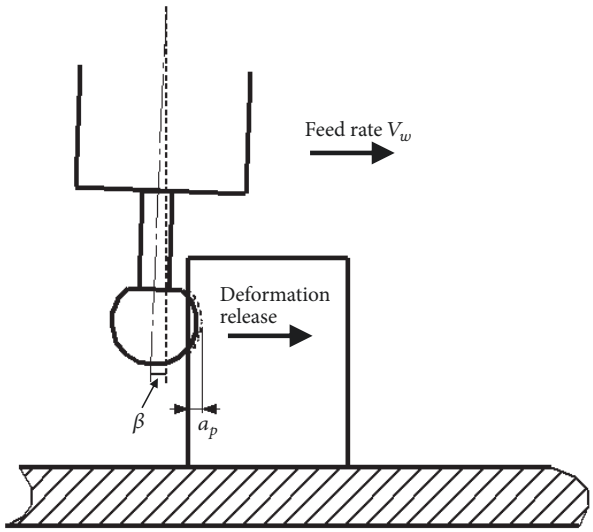

Figure 1: Generation of extrusion deformation.

defined as well as the corresponding ideal trajectory of robot grinding $x_p*(t_i)$. Equation (11) can be written as

$$\lim_{i\longrightarrow\infty}\left(x_p(t_i)-x_p*(t_i)\right)=0 \tag{12}$$

Assuming that there is a corresponding relation between the grinding position $x_p(t_i)$ and the robotic grinding force $F(t_i)$, when the system parameters (such as feed rate, grinding tool rotation speed, and material of workpiece) are constant. The control target in this paper can be converted into

$$\lim_{i\longrightarrow\infty}\left(F(t_i)-F*(t_i)\right)=0 \tag{13}$$

where $F*(t_i)$ is the ideal grinding force corresponding to the ideal grinding position $x_p*(t_i)$.

3.2. Model Predictive Control for Robotic Grinding. Define robot grinding force deviation as

$$e_d(t_i)=F(t_i)-F*(t_i) \tag{14}$$

Assume that robot grinding force deviation at time t_{i+1} is affected by robot grinding force deviation $e_d(t_i)$ and system transfer function $G(V_w(t_i))$ at time t_i.

$$e_d(t_{i+1})=A_e e_d(t_i)+G(V_w(t_i)) \tag{15}$$

where $G(V_w(t_i))$ is the control output which takes feed rate $V_w(t_i)$ as variables; A_e is the weighting factor of the robot grinding force deviation. Assume $A_e = 1$. Rewrite (15) according to (14)

$$\begin{aligned}F(t_{i+1})-F*(t_{i+1})&=F(t_i)-F*(t_i)+G(V_w(t_i))\\ F(t_{i+1})-F(t_i)&=F*(t_{i+1})-F*(t_i)\\&\quad+G(V_w(t_i))\\ \Delta F(t_i)&=\Delta F*(t_i)+G(V_w(t_i))\end{aligned} \tag{16}$$

where

$$\begin{aligned}\Delta F(t_i)&=F(t_{i+1})-F(t_i)\\ \Delta F(t_i)&=M\Delta\ddot{x}_a(t_i)+C\Delta\dot{x}_a(t_i)+K_d\Delta x_a(t_i)\\&\quad+K_d\left(\Delta\sigma_c(t_i)+\Delta\sigma_j(t_i)\right)\end{aligned} \tag{17}$$

$$\Delta F*(t_i)=F*(t_{i+1})-F*(t_i) \tag{18}$$

Define $\Delta V_w(t_i)=V_w(t_{i+1})-V_w(t_i)$, $\dot{x}_a(t_i)=V_w(t_i)$, $\Delta x_a(t_i)=V_w(t_i)dt$. According to (6) there is $\Delta\sigma_j(t_i)=\sigma_j(t_{i+1})-\sigma_j(t_i)=V_w(t_{i+1})-V_e(t_{i+1})$, and (17) can be rewritten as

$$\begin{aligned}\Delta F(t_i)&=M\Delta\ddot{x}_a(t_i)+(C+K_d)\Delta V_w(t_i)\\&\quad+K_dV_w(t_i)(dt+1)\\&\quad+K_d\left(\Delta\sigma_c(t_i)-V_e(t_{i+1})\right)\end{aligned} \tag{19}$$

3.2.1. Robotic Grinding Trajectory Prediction. Substitute (19) and (15) into (14) to obtain the relationship between robotic grinding force deviation and feed rate

$$\begin{aligned}e_d(t_{i+1})&=A_e e_d(t_i)+G'(t_i)+(C+K_d)\Delta V_w(t_i)\\&\quad+K_dV_w(t_i)(dt+1)\end{aligned} \tag{20}$$

where $G'(t_i)=M\Delta\ddot{x}_a(t_i)-\Delta F*(t_i)+K_d(\Delta\sigma_c(t_i)-V_e(t_{i+1}))$. Therefore, the robotic grinding force deviation matrix $E_d(t_i)$ from time t_{i+1} to time t_{i+p} can be acquired by

$$\begin{aligned}&e_d(t_{i+1}\mid t_i)\\&\quad=A_e e_d(t_i)+G'(t_i)+(C+K_d)\Delta V_w(t_i\mid t_i)\\&\qquad+K_dV_w(t_i)(dt+1)\\&e_d(t_{i+2}\mid t_i)\\&\quad=A_e e_d(t_{i+1}\mid t_i)+G'(t_{i+1})\\&\qquad+(C+K_d)\Delta V_w(t_{i+1}\mid t_i)\\&\qquad+K_dV_w(t_{i+1}\mid t_i)(dt+1)\\&E_d(t_i)\\&\quad=\left[e_d(t_{i+1}\mid t_i),e_d(t_{i+2}\mid t_i),\ldots,e_d\left(t_{i+p}\mid t_i\right)\right]\end{aligned} \tag{21}$$

$$\begin{aligned}&e_d(t_{i+1}\mid t_i)\\&\quad=A_e e_d(t_i)+G'(t_i)+(C+K_d)\Delta V_w(t_i\mid t_i)\\&\qquad+K_dV_w(t_i)(dt+1)\\&e_d(t_{i+2}\mid t_i)\\&\quad=A_e e_d(t_{i+1}\mid t_i)+G'(t_{i+1})\\&\qquad+(C+K_d)\Delta V_w(t_{i+1}\mid t_i)\\&\qquad+K_dV_w(t_{i+1}\mid t_i)(dt+1)\\&E_d(t_i)\\&\quad=\left[e_d(t_{i+1}\mid t_i),e_d(t_{i+2}\mid t_i),\ldots,e_d\left(t_{i+p}\mid t_i\right)\right]\end{aligned} \tag{22}$$

$e_d(t_{i+p}\mid t_i)$ represents the predictive of robotic grinding force deviation at time t_{i+p} according to the system information at time t_i. Assume that there is a relationship between change of feed rate and input voltage:

$$\Delta V_w(t_i)=A\Delta V_w(t_{i-1})+B\Delta u_p(t_i) \tag{23}$$

where A and B are weight factors. The prediction of feed rate increment from time t_{i+1} to time t_{i+p} can be acquired by

$$\begin{aligned}
&\Delta V_w\left(t_{i+1} \mid t_i\right) = A\Delta V_w\left(t_i\right) + B\Delta u_p\left(t_{i+1} \mid t_i\right)\\
&\Delta V_w\left(t_{i+1} \mid t_i\right) = A^2\Delta V_w\left(t_{i-1}\right) + AB\Delta u_p\left(t_i \mid t_i\right)\\
&\quad + B\Delta u_p\left(t_{i+1} \mid t_i\right)\\
&\Delta V_{wy}\left(t_{i+p} \mid t_i\right) = \left(\Delta V_w\left(t_{i+1} \mid t_i\right), \Delta V_w\left(t_{i+2} \mid t_i\right), \ldots,\right.\\
&\quad \left.\Delta V_w\left(t_{i+p} \mid t_i\right)\right)
\end{aligned} \tag{24}$$

where $\Delta V_w(t_{i+1} \mid t_i)$, $\Delta u_p(t_{i+1} \mid t_i)$, respectively, represent the changes of feed rate and control voltage at time t_i predicted according to the system information at time t_{i+1}.

3.2.2. Control Output Optimizing. In order to optimize control output voltage, an energy function is defined as

$$\begin{aligned}
E\left(t_i\right) = &\sum_{i=1}^{i+n} \left\|\Gamma_1\left(e_d\left(t_{i+1} \mid t_i\right) - e_d * \left(t_{i+1}\right)\right)\right\|^2\\
&+ \left\|\Gamma_2\left(e_d\left(t_{i+1} \mid t_i\right)\right)\right\|^2
\end{aligned} \tag{25}$$

where

$$\begin{aligned}
&e_d\left(t_{i+1} \mid t_i\right) - e_d * \left(t_{i+1}\right)\\
&\quad = A_e e_d\left(t_i\right) + G'\left(t_i\right) + \left(C + K_d\right)\Delta V_w\left(t_i \mid t_i\right)\\
&\qquad + K_d V_w\left(t_i\right)\left(dt + 1\right) - e_d * \left(t_{i+1}\right)
\end{aligned} \tag{26}$$

Substitute (23) into (26):

$$\begin{aligned}
&e_d\left(t_{i+1} \mid t_i\right) - e_d * \left(t_{i+1}\right)\\
&\quad = A_e e_d\left(t_i\right) + G'\left(t_i\right)\\
&\qquad + \left(CA + K_d\left(A + dt + 1\right)\right)\Delta V_w\left(t_{i-1}\right)\\
&\qquad + \left(C + K_d\right)B\Delta u_p\left(t_i \mid t_i\right)\\
&\qquad + K_d V_w\left(t_{i-1}\right)\left(dt + 1\right) - e_d * \left(t_{i+1}\right)
\end{aligned} \tag{27}$$

Define

$$\begin{aligned}
E\left(t_i\right) &= \rho^{\mathrm{T}}\rho\\
\rho &= A_1 Z - D
\end{aligned} \tag{28}$$

where

$$\begin{aligned}
D &= \begin{bmatrix} \Gamma_1\left(A_e e_d\left(t_i\right) + G'\left(t_i\right) + \left(CA + K_d\left(A + dt + 1\right)\right)\Delta V_w\left(t_{i-1}\right) + K_d V_w\left(t_{i-1}\right)\left(dt + 1\right) - e_d * \left(t_{i+1}\right)\right)\\ \Gamma_2\left(A_e e_d\left(t_i\right) + G'\left(t_i\right) + \left(CA + K_d\left(A + dt + 1\right)\right)\Delta V_w\left(t_{i-1}\right) + K_d V_w\left(t_{i-1}\right)\left(dt + 1\right)\right) \end{bmatrix},\\
A_1 &= \begin{bmatrix} \Gamma_1\left(C + K_d\right)B \\ \Gamma_2\left(C + K_d\right)B \end{bmatrix},\\
Z &= \Delta u_p\left(t_i \mid t_i\right).
\end{aligned} \tag{29}$$

According to (26), the extreme value of energy function can be obtained by $\min \rho^T\rho$. Therefore, there is extreme value when:

$$\Delta u_p\left(t_i \mid t_i\right) = Z = \left(A_1{}^{\mathrm{T}}A_1\right)^{-1}A_1{}^{\mathrm{T}}D \tag{30}$$

Substitute (23) into (30):

$$\begin{aligned}
\Delta u_p\left(t_i \mid t_i\right) = &W_1 e_d\left(t_i\right) + W_2\Delta V_w\left(t_{i-2}\right)\\
&+ W_3\Delta u_p\left(t_{i-1}\right) + W_4 V_w\left(t_{i-1}\right) + b_1
\end{aligned} \tag{31}$$

where

$$\begin{aligned}
W_1 &= \left(A_1{}^{\mathrm{T}}A_1\right)^{-1}\left(\Gamma_1{}^2 + \Gamma_2{}^2\right)\left(C + K_d\right)BA_e;\\
W_2 &= \left(A_1{}^{\mathrm{T}}A_1\right)^{-1}\left(\Gamma_1{}^2 + \Gamma_2{}^2\right)\left(C + K_d\right)\\
&\quad \cdot B\left(CA + K_d\left(A + dt + 1\right)\right)A;\\
W_3 &= \left(A_1{}^{\mathrm{T}}A_1\right)^{-1}\left(\Gamma_1{}^2 + \Gamma_2{}^2\right)\left(C + K_d\right)\\
&\quad \cdot B\left(CA + K_d\left(A + dt + 1\right)\right)B;\\
W_4 &= \left(A_1{}^{\mathrm{T}}A_1\right)^{-1}\left(\Gamma_1{}^2 + \Gamma_2{}^2\right)\left(C + K_d\right)BK_d\left(dt + 1\right);\\
b_1 &= \left(A_1{}^{\mathrm{T}}A_1\right)^{-1}\left(\Gamma_1{}^2 + \Gamma_2{}^2\right)\left(C + K_d\right)\\
&\quad \cdot B\left(2G'\left(t_i\right) - e_d * \left(t_{i+1}\right)\right)
\end{aligned} \tag{32}$$

3.2.3. Stability Analysis. According to (22), (23), and (30), there is

$$\begin{aligned}
e_d\left(t_{i+1} \mid t_i\right) = &W_5 e_d\left(t_i\right) + W_6\Delta V_w\left(t_{i-2}\right)\\
&+ W_7\Delta u_p\left(t_{i-1}\right) + W_8 V_w\left(t_{i-1}\right) + b_2
\end{aligned} \tag{33}$$

where

$$
\begin{aligned}
W_5 &= A_e + (C + K_d) BW_1,\\
W_6 &= \left((CA + K_d (A + dt + 1)) A + (C + K_d) BW_2\right);\\
W_7 &= \left((CA + K_d (A + dt + 1)) B + (C + K_d) BW_3\right);\\
W_8 &= \left((C + K_d) BW_4 + K_d (dt + 1)\right);\\
b_2 &= G'(t_i) + (C + K_d) Bb_1
\end{aligned}
\tag{34}
$$

According to the model predictive control principle [18], when the eigenvalues of the matrix W_5 are all within the unit circle, the control system is asymptotically stable.

3.2.4. Robot Grinding State Prediction Model. Merging (33) and (31), there is

$$
O = IW_n^{\mathrm{T}} + B_b \tag{35}
$$

where

$$
\begin{aligned}
O &= \left[\Delta u_p (t_i \mid t_i), e_d (t_{i+1} \mid t_i)\right];\\
W_n &= \left[W_1, W_2, W_3, W_4; W_5, W_6, W_7, W_8\right];\\
B_b &= \left[b_1, b_2\right];\\
I &= \left[e_d (t_i), \Delta V_w (t_{i-2}), \Delta u_p (t_{i-1}), V_w (t_{i-1})\right].
\end{aligned}
\tag{36}
$$

W_n and B_b represent predictive model parameters. Then, the deep belief network is introduced to obtain predictive model parameters.

3.3. Predictive Model Based on Deep Belief Network. According to the deep belief network principle,

$$
O = \left(\left(\left(IW_{d,1}^{\mathrm{T}} + B_{d,1}\right) W_{d,2}^{\mathrm{T}} + B_{d,2}\right)\ldots\right) W_{d,i}^{\mathrm{T}} + B_{d,i} \quad (i = 1, 2, \ldots, m) \tag{37}
$$

Comparing (37) and (35), it can be found that predictive model parameter W_n can be obtained by weights product $\prod_{i=1}^{m} W_{d,i}$:

$$
\begin{aligned}
W_n^{\mathrm{T}} &= \prod_{i=1}^{m} W_{d,i}^{\mathrm{T}},\\
B_b &= \sum_{j=1}^{m-1} B_{d,j} \prod_{j=1}^{m-1} W_{d,j+1}^{\mathrm{T}} + B_{d,m}
\end{aligned}
\tag{38}
$$

Therefore, an assumption is made that $\prod_{i=1}^{m} W_{d,i} = [W_1, W_2, W_3, W_4; W_5, W_6, W_7, W_8]$ can be obtained by the product of $\Gamma_1, \Gamma_2, A_e, C, K_d, B$. And subassumptions are made that $W_{d,i} = [B, B^2, A, A^2, Bdt, Adt, \ldots]$; $W_{d,i+1} = [C, C^2, K_d, {K_d}^2, \ldots]$; $W_{d,i+2} = [({A_1}^{\mathrm{T}} A_1)^{-1} {\Gamma_1}^2, ({A_1}^{\mathrm{T}} A_1)^{-1} {\Gamma_2}^2, \ldots]$. Based on these assumptions, the deep belief network is constructed, which is comprised of three layers, including two layers of Restricted Boltzmann Machine (RBM) and one Back-Propagation (BP) layer. Two layers of RBM are used to obtain weights $W_{d,i} = [B, B^2, A, A^2, Bdt, Adt, \ldots]$ and $W_{d,i+1} = [C, C^2, K_d, {K_d}^2, \ldots]$, while the BP layer is used to obtain $W_{d,i+2} = [({A_1}^{\mathrm{T}} A_1)^{-1} {\Gamma_1}^2, ({A_1}^{\mathrm{T}} A_1)^{-1} {\Gamma_2}^2, \ldots]$. The inputs of DBN are grinding force deviation $e_d(t_i)$, feed rate $V_w(t_{i-1})$, increment of feed rate $\Delta V_w(t_{i-2})$, and increment of control output voltage $\Delta u_p(t_{i-1})$, while the outputs are $\Delta u_p(t_i \mid t_i)$ and $e_d(t_{i+1} \mid t_i)$. The energy function of general RBM is [19]

$$
E(v_i, h_i) = -{v_i}^T w_i h_i - {b_i}^T v_i - {d_i}^T h_i \quad (i = 1, 2) \tag{39}
$$

where v_i, h_i are explicit layer and hidden layer; w_i, b_i, d_i are, respectively, network weight, explicit layer bias, and hidden layer bias. Consider O as the first visible layer of first RBM and h_1 as the hidden layer first RBM, and (39) can be rewritten as

$$
E(O, h_i) = -O^T w_i h_i - {b_i}^T O - {d_i}^T h_i \tag{40}
$$

The probability that the RBM assigns to a visible vector is

$$
P(O) = \frac{1}{Z} \sum_{h_i} e^{-E(O, h_i)} \quad \text{where } Z = \sum_{O, h_i} e^{-E(O, h_i)} \tag{41}
$$

The contrastive divergence method is then used to obtain the update formula of DBN parameters [19]:

$$
\begin{aligned}
\frac{\partial \log P(O)}{\partial w_{ij}} &= P\left(h_{1,i} = 1 \mid O^{(0)}\right) O_j^{(0)}\\
&\quad - P\left(h_{1,i} = 1 \mid O^{(k)}\right) O_j^{(k)}\\
\frac{\partial \log P(O)}{\partial b_1} &= O^{(0)} - O^{(k)}\\
\frac{\partial \log P(O)}{\partial d_1} &= P\left(h_{1,i} = 1 \mid O^{(0)}\right) - P\left(h_{1,i} = 1 \mid O^{(k)}\right)
\end{aligned}
\tag{42}
$$

For the construction of BP network, a classical model is used. The relevant functions, including activation function, energy function, and update formulas, are then set based on the classical model of BP network. The structure diagram of DBN is presented in Figure 2.

The overall control flowchart is shown in Figure 3. Firstly, the function of control output voltage is obtained by minimizing energy function, which is the sufficient condition of optimal control, and the MPC model is constructed based on it. The DBN is designed according to MPC model to obtain the parameters of MPC. The training of DBN is achieved with the use of robotic grinding historical data. After identification of MPC parameters, the MPC is used to perform online close-loop control with the use of robotic grinding force deviation $e_d(t_i)$ and system information $\Delta V_w(t_{i-2}), \Delta u_p(t_{i-1}), V_w(t_{i-1})$."

4. DBN Training and Simulation

Simulations are conducted to evaluate feasibility and performance of the DBN used to obtain the model predictive control parameters of robotic grinding. A BP network is constructed, and a comparison between the BP network and the proposed DBN is conducted.

Table 1: Performance comparison between BP and DBN (mean square error).

Time	BP network		DBN	
	e_d	Δu_p	e_d	Δu_p
t_{i+0}	$9.86 * 10^{-7}$	$6.86 * 10^{-7}$	$6.04 * 10^{-7}$	$3.01 * 10^{-7}$
t_{i+1}	$9.76 * 10^{-7}$	$8.96 * 10^{-7}$	$8.36 * 10^{-7}$	$9.76 * 10^{-7}$
t_{i+2}	0.156	0.167	0.152	0.160
t_{i+3}	0.165	0.170	0.160	0.163
t_{i+4}	0.177	0.171	0.171	0.165
t_{i+5}	0.187	0.183	0.185	0.175
t_{i+6}	0.194	0.185	0.191	0.181
t_{i+7}	0.216	0.201	0.205	0.194

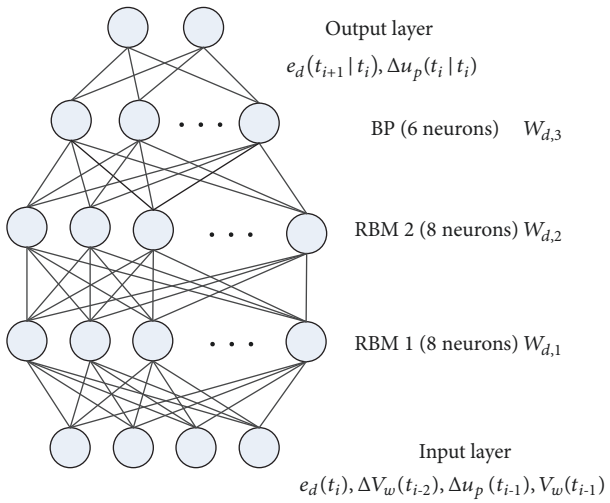

Figure 2: DBN structure diagram.

According to (35) and (37), the MPC is used to obtain $O = [\Delta u_p(t_i \mid t_i), e_d(t_{i+1} \mid t_i)]$ with the use of $I = [e_d(t_i), \Delta V_w(t_{i-2}), \Delta u_p(t_{i-1}), V_w(t_{i-1})]$, which means that the MPC represents the relationship between $I = [e_d(t_i), \Delta V_w(t_{i-2}), \Delta u_p(t_{i-1}), V_w(t_{i-1})]$ and $O = [\Delta u_p(t_i \mid t_i), e_d(t_{i+1} \mid t_i)]$. Therefore, the training data is required to contain the relationship between $I = [e_d(t_i), \Delta V_w(t_{i-2}), \Delta u_p(t_{i-1}), V_w(t_{i-1})]$ and $O = [\Delta u_p(t_i \mid t_i), e_d(t_{i+1} \mid t_i)]$.

The training data of DBN is determined according to the following aspects.

(1) The parameters of training data are determined according to influence factors of robotic grinding process, including grinding depth, grinding tool feed rate, tool rotation speed, and control output voltage. As there is a functional relationship between feed rate and control output voltage when a controller is applied, this paper uses feed rate change to represent change of control voltage. The ranges of the parameters are grinding depth is from 0.6 mm to 1.2 mm; initial feed rate is from 0.5 mm/s to 1.2mm/s; feed rate change is from 0.1 mm^2/s to 1 mm^2/s; tool rotation speed is 4000 r/min. (2) Since grinding is a material removal process, it is difficult and costly to perform dozens of experiments and obtain a large amount of training data; the DBN model used in this paper is only designed for specific case (robotic grinding) as Oh and Jung did in their research [19], which can reduce the quantity demand of data for DBN training but also reduce the applicability of DBN applying in other robotic operations. Based on this, the scale and the size of learning sample are determined. The scale of learning sample batch is 40 packets and each packet has about 5000∗4 data. 34 packets are randomly selected for training and the remaining 6 are selected for model validation. The maximum epoch is 300.

The training performances of both networks are shown in Figure 4. For the model training of robotic grinding status at time t_{i+0}, the DBN realizes fitting at 31st epoch and BP network realizes fitting at 52nd epoch. The model training of robotic grinding status from time t_{i+0} to t_{i+7} is also conducted, and the training performance of both BP network and DBN is presented in Table 1. In overall, the training performances of the proposed DBN are approximate to the training performances of the classical BP network. The training results are then used to construct.

A simulation is conducted to evaluate the predictive performance of models obtained, respectively, by the proposed DBN and the BP network. The steps of the simulation are as follows. Firstly, a sample is selected randomly, and the first 10 data of the sample are taken as inputs. Then, predictions of robotic grinding state are made according to these inputs. As shown in Figure 5, the prediction results of two models are basically consistent with the actual grinding force. The average prediction deviation of grinding force is about 0.5N. The maximum prediction deviation, about 2 N, appears at the peak of grinding force during cut-into state. At 2nd second, the grinding force predicted by the BP based model changes dramatically, while the one obtained by the DBN based model is smoother. Based on the above simulations, a preconclusion can be made that the performance of predictive model obtained by DBN network is better than the one obtained by BP network.

5. Robotic Grinding Control Experiments

5.1. Robotic Grinding System. The robotic machining system consists of YASKAWA industrial robot MH24, six-axis force sensor ME-FK6D40, and grinding tool. As shown in Figure 6, the force sensor and grinding tool are installed on the end-effector of robot, while the workpiece is installed on a fixed

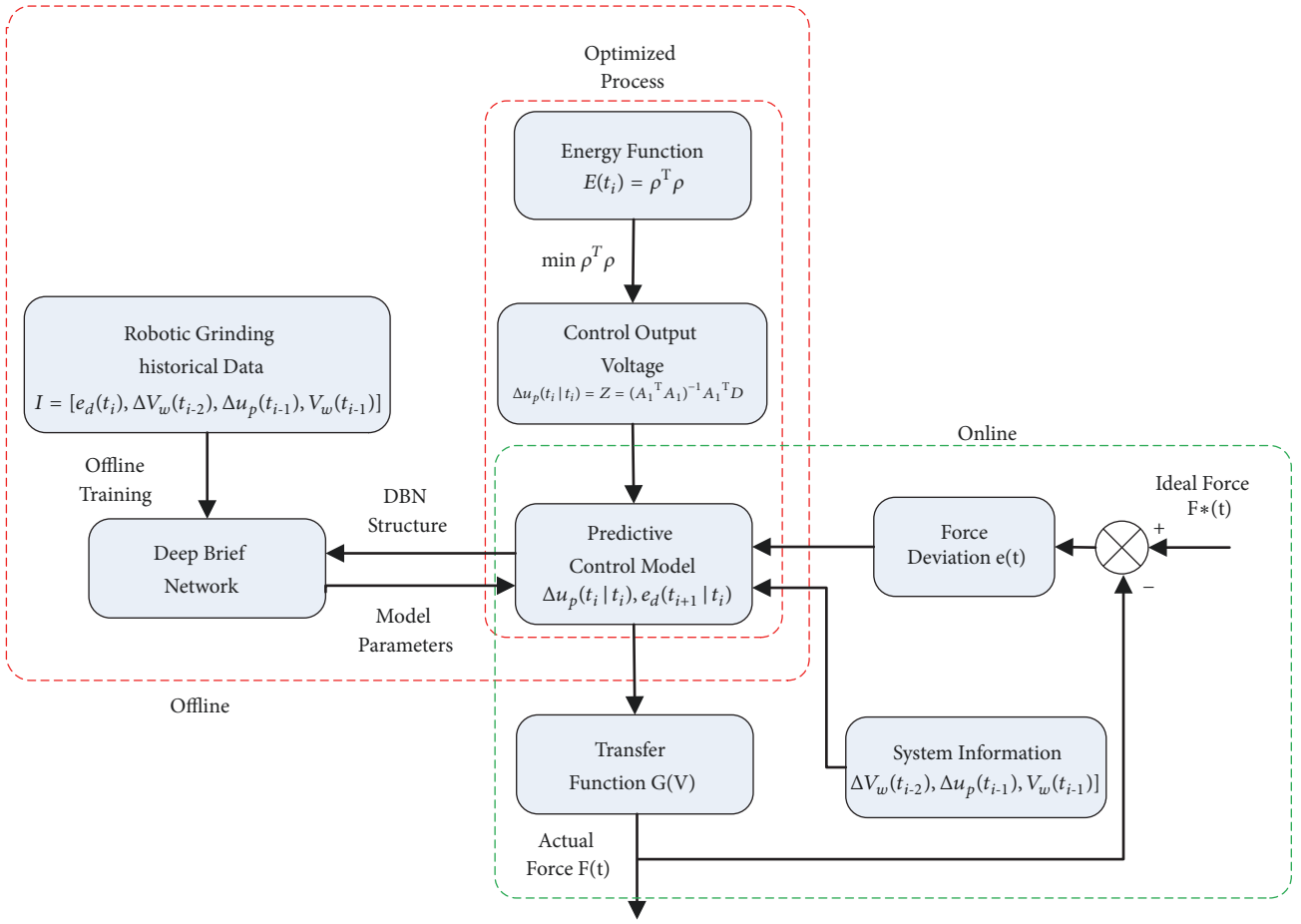

Figure 3: Schematic diagram of robot model predictive control process based on DBN.

platform. The spindle of grinding tool is perpendicular to the Z direction of industrial robot sixth axis. The world coordinate system is set according to the robot Cartesian coordinate. The relevant parameters of equipment are the six-axis force sensor is ME-FK6d40, Germany; the workpiece material is Q235 steel; the grinding tool is composed of motor, handle, and grinding head.

The force signals are collected by the force sensor and delivered to the embedded real-time control system in PC with the use of Ethercat protocol. The analogue filter frequency of force sensor is 2500 Hz, and the sampling frequency of embedded real-time control system is 1 ms. The control output is then calculated by embedded real-time control system and sent to YASKAWA robot control cabinet to modify output pulse and adjust the feed rate of end-effector. The frequency of output voltage of control system is about 100 ms.

5.2. Robotic Grinding Control Experiments. To evaluate the performance of proposed control approach, robotic grinding control experiments are conducted. The open loop robotic grinding experiment, robotic grinding experiment with fuzzy-PD control, and robotic grinding experiment with model predictive control are conducted, and comparisons between these experiments are made. The robotic grinding control experiments are carried out on the steel plate plane (Q235). The rotation speed of grinding tool is 4000 r/min, and the grinding depth is 1.2 mm; the initial feed rate is 1 mm/s; the material of cutter is D1614M06.

The grinding path of the experiment is shown in Figure 7. The grinding tool moves from point A to D via point B and C. The grinding path AB is perpendicular to workpiece surface, while the path BC is tangential to workpiece surface. For the convenience of discussion, in subsequent articles, AB is regarded as cut-into state, while BC is regarded as stable grinding state.

5.2.1. Experiment with Open Loop Control and Experiment with Fuzzy-PD Control

(A) Experiment with Open Loop Control. The desired grinding force for the experiment is 14 N. The experiment result is shown in Figure 8. The grinding force increases dramatically to 29N during cut-into state and then decreases around 15N. The dramatically increase of grinding force, which is labeled as abrupt change, during cut-into state can be illustrated by (10) and (16). When grinding tool cuts into the workpiece, the gap between feed rate and removal rate

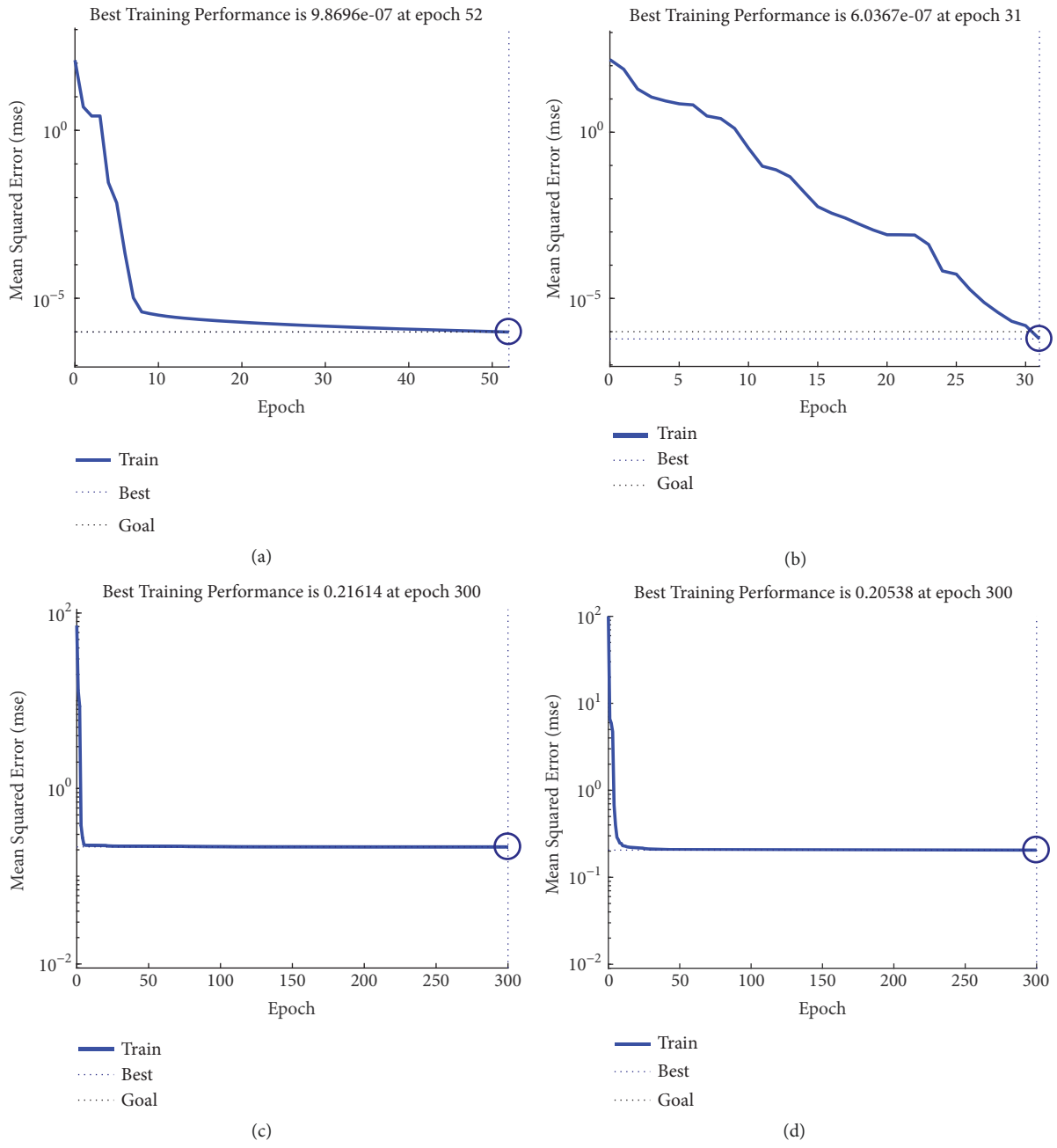

FIGURE 4: Training performance comparison between BP and DBM (Mean Square Error). ((a) BP (time t_{i+0}); (b) DBN (time t_{i+0}); (c) BP (time t_{i+7}); (d) DBN (time t_{i+7})).

results in a deformation and leads to an increase grinding force. After perpendicular move, the grinding tool starts to move in tangential direction and the value of removal rate in perpendicular direction is larger than the one of feed rate resulting in a decrease of grinding force.

(B) Experiment with Fuzzy-PD Control. To verify the effectiveness of proposed control approach, an experiment based on fuzzy-PID control is conducted. The inputs of fuzzy-PD controller are force deviation e and its first-order derivative $\dot{e}$ and control parameters T_p and T_d. Their domains are $e = \{-6, -3, 0, 3, 6\}$, $\dot{e} = \{-6, -3, 0, 3, 6\}$, $T_p = \{-0.6, -0.3, -0.1, -0.05, -0.01\}$, and $T_d = \{-0.4, -0.2, -0.1, -0.05, -0.01\}$. The fuzzy reasoning is "IF A1 and B1 Then C1 and D1", while the Gaussian function is used as fuzzy membership function.

The result of grinding experiment with PD control is shown in Figure 9. The grinding force increases considerably and reaches a peak value of 25 N, which is 4 N lower than the one of open loop control. The corresponding feed rate increases to 1.4mm/s at the first time and then decrease to 0.7mm/s with a valley of 0.4mm/s. During this process,

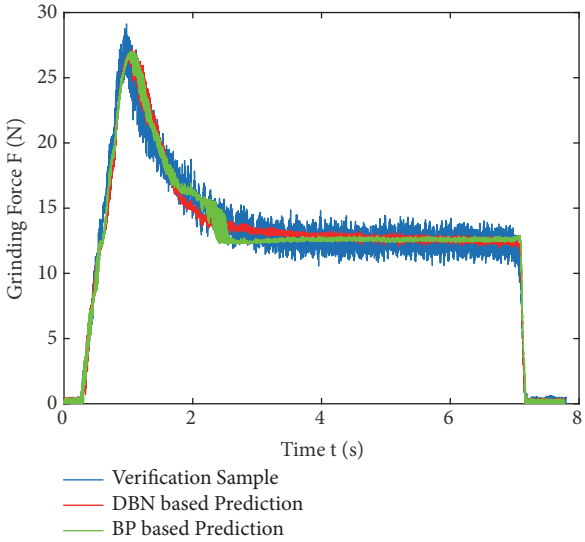

Figure 5: Prediction performance.

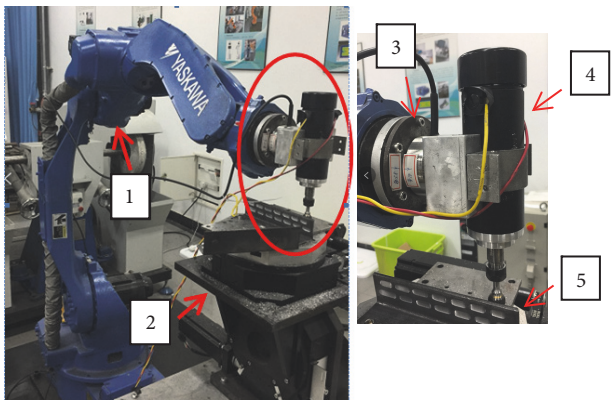

Figure 6: Architecture of robotic cutting system ((1) YASKAWA 6-DOF industrial robot; (2) rotating platform; (3) six-axis force sensor; (4) cutting tool; (5) workpiece).

a fluctuation of feed rate appears at 1st second and the fluctuation amplitude of feed rate is around 0.1mm/s. This is because when the grinding force is close to the target force, the control parameters swift according to the fuzzy rules which results in a fluctuation of control output. This is a common problem of fuzzy control as fuzzy rules are often defined according to the artificial experience which is easy to result in control oscillations caused by switching of control rate. In view of this, a model predictive control approach is applied to robotic grinding control.

5.2.2. Robotic Grinding Experiment with Model Predictive Control. The robotic grinding experiment with model predictive control is then conducted to reduce the grinding force deviation and feed rate fluctuation that occurs in cut-into state. The prediction time of model predictive control is set as t_{i+10}, and the experiment result is shown in Figure 10. The peak value of robotic grinding force is 18N, which is 7N lower and the corresponding feed rate curve changes from 1.4mm/s to 0.7mm/s with a valley of 0.3mm/s. The changes of robotic

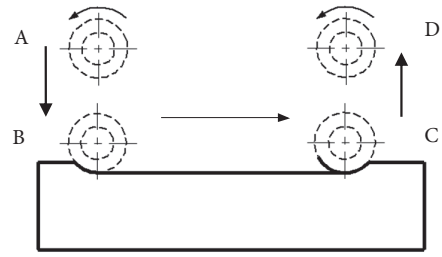

Figure 7: Grinding path.

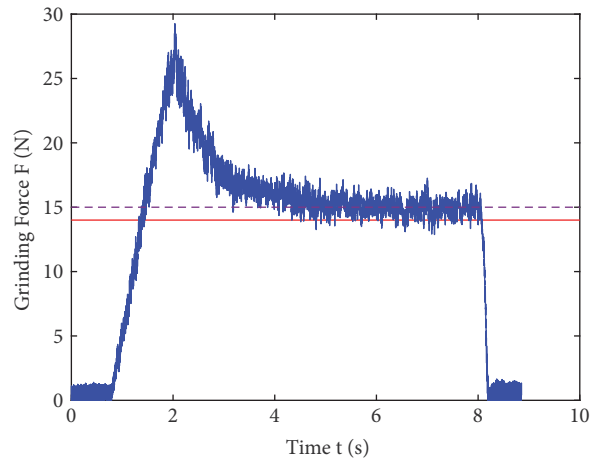

Figure 8: Robotic grinding experiment with open loop control.

grinding force and feed rate shown in Figure 10 are smoother compared with the ones shown in Figure 9. This is because that the model predictive control approach can predict the future grinding deviation based on acquired information and result in a control compensation to reduce the coming up force deviation and feed rate fluctuation.

To further verify the performance of proposed control approach, another robotic grinding experiment based on model predictive control is conducted of which the prediction time is set as t_{i+100}. As shown in Figure 11, the grinding force increases smoothly from 0N to 14N and the fluctuation of grinding force is around 2N. The abrupt change of grinding force, which occurs during cut-into state as shown in Figures 10 and 9, is completely eliminated. The corresponding feed rate changes from 1 mm/s to 0.75 mm/s without an abrupt increase as shown in Figures 9 and 10. This is because when the prediction time is adjusted to t_{i+100}, the control system can have sufficient time to adjust feed rate smoothly, which can lead to sufficient control compensation.

To explore the flexibility of the proposed control approach, the grinding depth is changed to 1.4mm, while the grinding depth range of study sample is from 0.6mm to 1.2mm. The experiment result with open loop control is shown in Figure 12(a), while the experiment results with model predictive control are shown in Figures 12(b) and 12(c). As shown in Figure 12(a), the grinding force increases dramatically to about 33 N during cut-into state and then

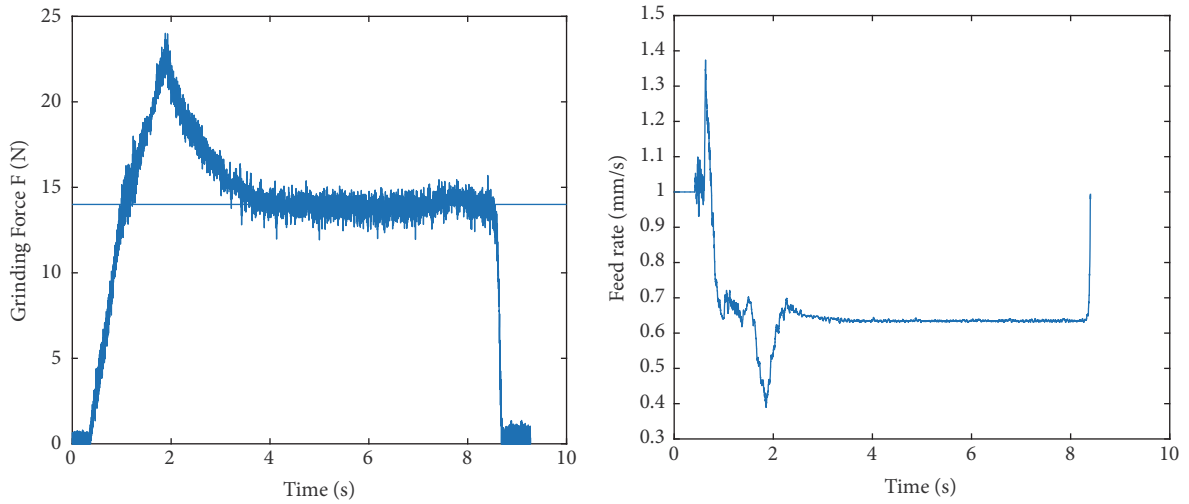

Figure 9: Robotic grinding experiment with fuzzy-PD control.

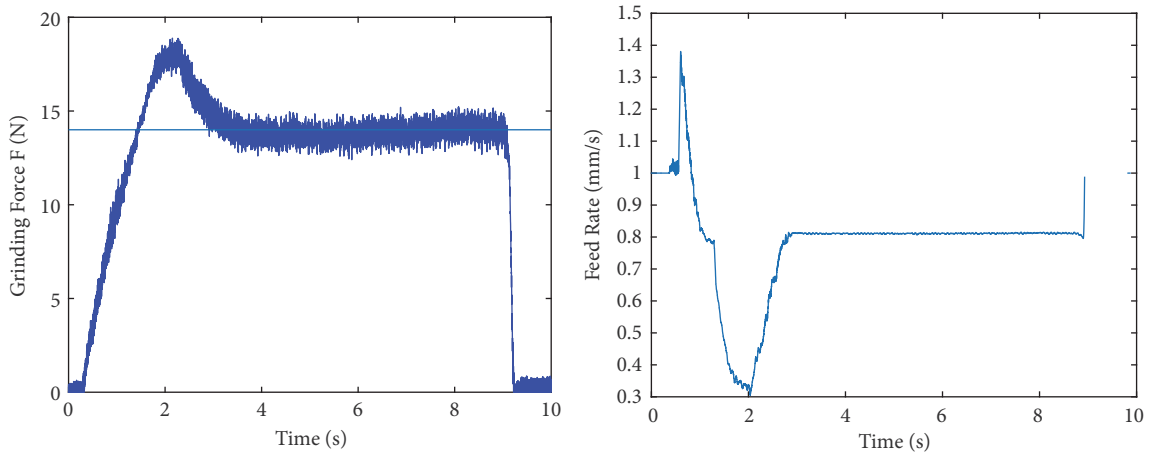

Figure 10: Robotic grinding experiment with model predictive control (t_{i+10}).

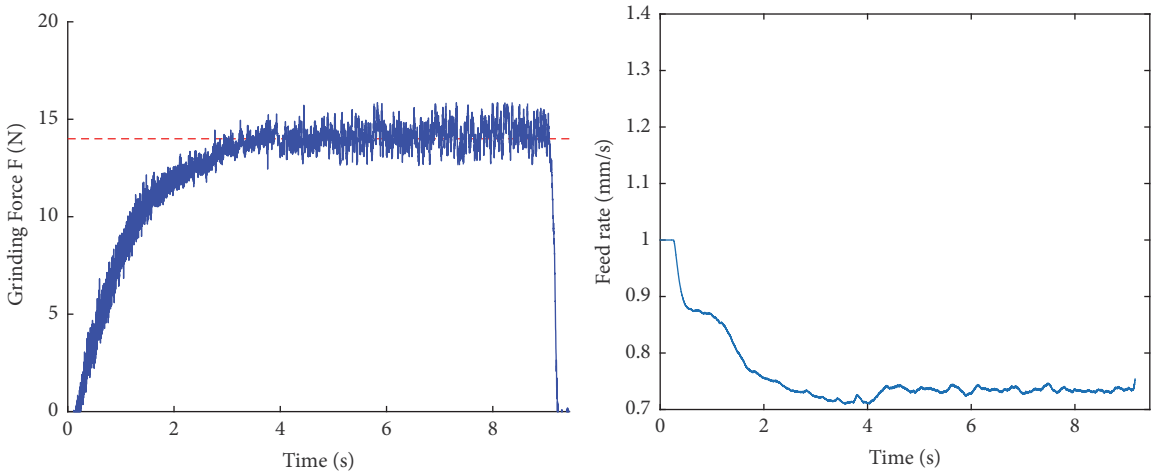

Figure 11: Robotic grinding experiment with model predictive control (t_{i+100}; depth 1.2mm).

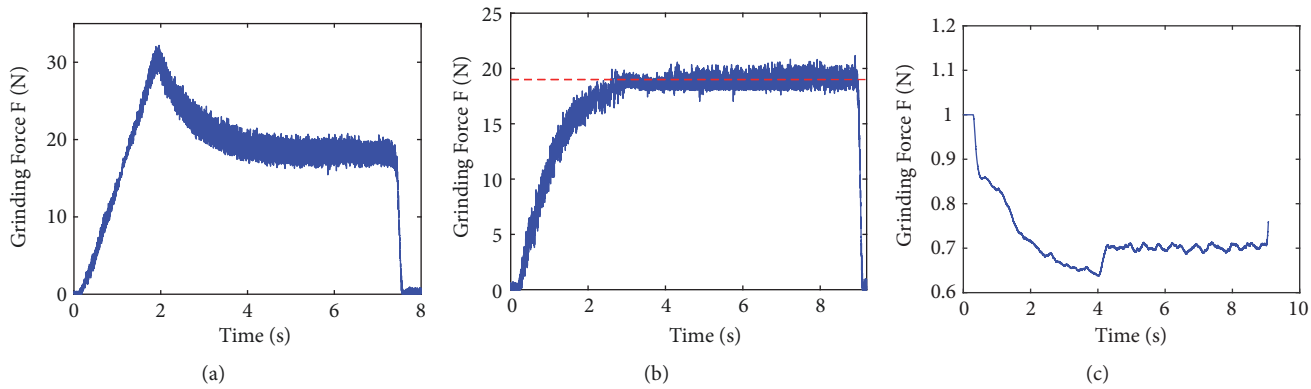

FIGURE 12: Robotic grinding experiment with model predictive control (t_{i+100}; depth 1.4mm).

decreases to around 19N. When model predictive control approach is implemented, the grinding force increases smoothly from 0N to 19N without an abrupt change during cut-into state. The corresponding feed rate curve shown in Figure 12(c) is also smooth, which decreases from 1 mm/s to 0.7mm/s with a small valley of 0.65 mm/s at 4^{th} second.

Compared with fuzzy-PD control approach, the proposed MPC approach can avoid mutation of grinding force and reduce the grinding force deviation when prediction interval is large enough, as shown in Figure 12. Also the feed rate change of robotic grinding based on MPC is smoother compared with the one based on fuzzy-PD control. The conclusion is the same while the comparison is made with the experiments based on adaptive PID approach which presented in a previous research of writers [20]. The comparison shows that adaptive PID control can reduce the mutation of grinding force but not able to avoid it, as the adaptive PID control reacts only when the mutation comes out.

6. Conclusion

In this paper, a model predictive control approach based on deep belief network is proposed to control the deviation of grinding force caused by robotic deformation. The robotic grinding dynamic impacted by rigid-flexible coupling effect is analyzed and the deep belief network is used to study the robotic grinding dynamic to acquire predict model. Based on this, the model predictive control approach can predict the change of grinding force deviation and achieve a combination of feed-forward and feedback control.

The experimental results show that the prediction model of robot grinding based on the deep belief network can accurately predict the change of robot grinding state in time t_{i+10} and in time t_{i+100}. Compared with open loop control and fuzzy-PD control, the model predictive control approach can perform a compensate control to beforehand reduce the force deviation. The elimination of the abrupt change of grinding force caused by deformation as shown in Figures 11 and 12 verifies the performance of proposed control approach.

Since current researches are mostly focusing on the feedback control of robot machining, while feed-forward compensation control is rarely applied to robotic machining, the contributions of this paper are as follows: (1) the model predictive control approach is adjusted according to robotic grinding model within the consideration of coupling effect between rigid-flexible system and control system. The MPC is used to predict the mutation of grinding force and perform forward control to avoid grinding force mutation. The proposed approach has both the advantages of feed-forward control and feedback control, which can reduce the control oscillations and overflow caused by system delay and feedback control. (2) A deep belief network is designed according to the MPC model of robotic grinding control system. Since the MPC model is nonlinear and the parameters are difficult to obtain, the use of deep belief network can benefit the obtainment of MPC model parameters. Therefore, this study provides model reference and data support for the research of nonlinear control method of robot machining process based on intelligent model.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] S. Mousavi, V. Gagnol, B. C. Bouzgarrou, and P. Ray, "Stability optimization in robotic milling through the control of functional redundancies," *Robotics and Computer-Integrated Manufacturing*, vol. 50, pp. 181–192, 2018.

[2] M. Cordes and W. Hintze, "Offline simulation of path deviation due to joint compliance and hysteresis for robot machining," *The International Journal of Advanced Manufacturing Technology*, vol. 90, no. 1-4, pp. 1075–1083, 2017.

[3] A. Brunete, E. Gambao, J. Koskinen et al., "Hard material small-batch industrial machining robot," *Robotics and Computer-Integrated Manufacturing*, vol. 54, pp. 185–199, 2018.

[4] R. Mei and C. Yu, "Adaptive neural output feedback control for uncertain robot manipulators with input saturation," *Complexity*, vol. 2017, Article ID 7413642, 12 pages, 2017.

[5] N. Mendes and P. Neto, "Indirect adaptive fuzzy control for industrial robots: a solution for contact applications," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8929–8935, 2015.

[6] F.-Y. Hsu and L.-C. Fu, "Intelligent robot deburring using adaptive fuzzy hybrid position/force control," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 4, pp. 325–335, 2000.

[7] V. T. Yen, W. Y. Nan, and P. van Cuong, "Recurrent fuzzy wavelet neural networks based on robust adaptive sliding mode control for industrial robot manipulators," *Neural Computing and Applications*, vol. 18, pp. 1–14, 2018.

[8] T. Faulwasser, T. Weber, P. Zometa, and R. Findeisen, "Implementation of nonlinear model predictive path-following control for an industrial robot," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 4, pp. 1505–1511, 2017.

[9] Y. G. Xi, D. L. Li, and S. Lin, "Model predictive control — status and challenges," *Acta Automatica Sinica*, vol. 39, no. 3, pp. 222–236, 2013.

[10] J. Wilson, M. Charest, and R. Dubay, "Non-linear model predictive control schemes with application on a 2 link vertical robot manipulator," *Robotics and Computer-Integrated Manufacturing*, vol. 41, no. C, pp. 23–30, 2016.

[11] H. Peng, J. Wu, G. Inoussa, Q. Deng, and K. Nakano, "Nonlinear system modeling and predictive control using the RBF nets-based quasi-linear ARX model," *Control Engineering Practice*, vol. 17, no. 1, pp. 59–66, 2009.

[12] H. Peng, K. Nakano, and H. Shioya, "Nonlinear predictive control using neural nets-based local linearization arx model—stability and industrial application," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 1, pp. 130–143, 2007.

[13] N. Nikdel, P. Nikdel, M. A. Badamchizadeh, and I. Hassanzadeh, "Using neural network model predictive control for controlling shape memory alloy-based manipulator," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 3, pp. 1394–1401, 2014.

[14] Z. Li, W. Yuan, Y. Chen, F. Ke, X. Chu, and C. L. P. Chen, "Neural-dynamic optimization-based model predictive control for tracking and formation of nonholonomic multirobot systems," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–10, 2018.

[15] X. Zeng, H. Peng, and J. Wu, "An improved multivariable RBF-ARX model-based nonlinear model predictive control approach and application," *Journal of Central South University (Science and Technology)*, vol. 46, no. 10, pp. 3710–3717, 2015.

[16] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl, "Nonlinear model predictive control with neural network optimization for autonomous autorotation of small unmanned helicopters," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 4, pp. 818–831, 2011.

[17] C. Zhang, *Machinery Dynamics*, Higher Education Press, Beijing, China, 2nd edition, 2007.

[18] H. Chen, *Model_Predictive_Control*, Science Press, Beijing, China, 1st edition, 2013.

[19] H. Oh, J. H. Jung, B. C. Jeon, and B. D. Youn, "Scalable and unsupervised feature engineering using vibration-imaging and deep learning for rotor system diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3539–3549, 2018.

[20] S. Chen and T. Zhang, "Force control approaches research for robotic machining based on particle swarm optimization and adaptive iteration algorithms," *Industrial Robot: An International Journal*, vol. 45, no. 1, pp. 141–151, 2018.