Self-learning Optimization of Turning Process Parameters Based on NSGA-II and ANNs

Shenghao Shi, Hui Zhang, and Peng Mou
Department of Mechanical Engineering, Tsinghua University, Beijing, China
Email: shish17@mails.tsinghua.edu.cn, {wwjj, moupeng }@ mail.tsinghua.edu.cn

***Abstract*—This paper proposes an optimization method of turning process parameters based on nondominated sorting genetic algorithm II (NSGA-II) and artificial neural networks (ANNs). In the period of computer-aided process planning (CAPP), each machining operation with its process parameters should be given in sequence for the final output of workshop documentation and machining program. NSGA-II algorithm is used in the optimization of process parameters, including spindle speed, feed rate, depth of cut, etc. ANNs are used to predict the performance parameters, including cut force and surface roughness as constraints or objective of the optimization process. This process is a self-learning process because in the period of machining, data and signal from CNC is collected as training sample to iterate the artificial neural networks.**

***Index Terms*—CAM, CAPP, optimization, machine learning**

I. Introduction

Machining is one among the four popular manufacturing processes, the other three being forming, casting, and joining [1]. As the performance and environmental requirements of the product increase, the requirements for machining are also increased. These requirements include higher machining efficiency, machining accuracy and surface quality, lower economic and ecological cost. The process parameters are important factors affecting machining efficiency, machining accuracy and surface quality.

Process parameters is traditionally determined by cut manuals, production practice data, or cut tests. The data in the cut manual is widely sourced and organized, but it is less targeted and accurate. The production practice data is highly targeted to the enterprise, but the data is too scattered and lacks regularity. The data obtained through the cut test is the most targeted, but the data is extremely limited due to various limitations of the test conditions. In addition, test conditions and production site conditions often differ greatly.

The general process of turning involves rotating a part while a single-point cut tool is moved parallel to the axis of rotation [2]. Turning can be done on the external surface of the part as well as the internal surface (the process known as boring). The peripheral speed of the work piece called cut speed, movement of the tool along the axis of job for one revolution of job called feed, and radial depth of cut of the tool are the process parameters[1]. Different from traditional ways of parameters determination, These parameters can be optimized to obtain the minimum cost of machining and minimum production time. However, for optimization, the effect of process parameters on machining performance has to be studied to predict the machining performance.

Surface finish is defined as the degree of smoothness of a part's surface after it has been manufactured. Surface finish is the result of the surface roughness, waviness, and flaws remaining on the part [1]. Researchers studied the effect of factors such as feed rate, cut speed, depth of cut, on surface finish. Artificial neural networks (ANNs)are widely used in the prediction of surface finish. Pal and Chakraborty [3] predicted surface roughness by taking main cut force ,feed force, cut speed, feed, and depth of cut as input parameters of the network. Ozel and Karpat [4] predicted surface roughness by developing two different network models. One model was offline with process parameters, tool and job information as input, while the other was an online model with cut forces as an additional input. It was found that the model with cut forces as input yielded better results.

Cut force is an important characteristic variable to be monitored and predicted. To predict and monitor cutting forces, various models were proposed. Ezugwu et al. [5] used an ANN approach to model the correlation between five process parameters, viz., speed, feed rate, depth of cut, cutting time, and coolant pressure, with seven performance parameters, viz., tangential force, feed force, spindle motor power consumption, surface roughness, average flank wear, maximum flank wear, and nose wear. The developed model agrees well with experimental data and can be used to analyze and predict the relationship between process and performance parameters. Li et al. [6] used neurofuzzy techniques to estimate feed cutting force by measuring motor current using current sensor in CNC turning center. Motor current and feed rate were used as input parameters. The authors found that the estimated force was within an error of 5%.

Normally, a turning process involves a number of rough turnings and a finish turning. In rough turnings, highest possible metal removal is of most concern. Surface roughness is not an important consideration. In finish turning, surface finish is the most important consideration. In rough turnings and finish turning, cut force should be taken into consideration to avoid tool breakage and reduce tool wear.

Manuscript received July 7, 2019; revised April 1, 2020.

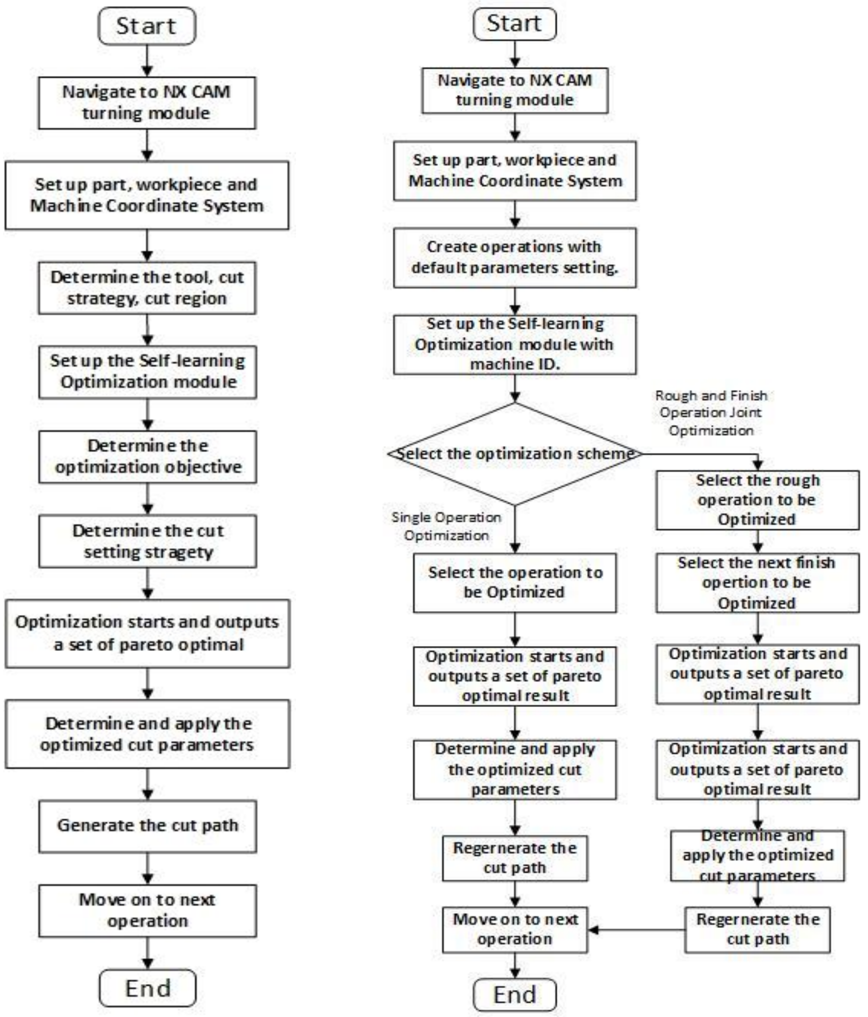
doi: 10.18178/ijmerr.9.6.841-846

In multipass turning optimization, the distribution of total depth of cut among different rough turns and finish turn is an important task. In single turning optimization, the determination of spindle speed and feed rate is the main task.

This paper developed an process optimization model for turning process. The effect of process parameters on machining performance was predicted by ANNs prediction model. Different prediction models were developed for different turning operations. Cut force prediction models are developed for outer diameter turning, inner diameter turning, facing and grooving operation. Surface roughness prediction models were developed for finish turning operation, including outer diameter finishing, inner diameter finishing, and facing. Due to the space limitation of groove, surface roughness prediction model was not developed because surface roughness detection could not be executed.

Generally, each optimization model is a NSGA-II optimization model based on artificial neural networks prediction model to optimize turning process parameters, obtaining the minimum cost of machining or minimum production time. The optimization model is called self-learning optimization because the neural networks prediction model iterates itself by training on the data collected from CNC and NX CAM. The optimization models and prediction models are deployed in Siemens NX CAM software. A scheme to collect data from plant devices CNC and to generate training samples is also designed and discussed.

II. Concept



(a) Optimization scheme when creating operations (b) Optimization scheme after creating operations

Figure 1. Workflows of cut parameters optimization module

For key industries such as machinery, aerospace, automotive, and other industrial product suppliers, it is common to use CAM software such to help with turned parts manufacture.

Taking NX CAM as an example, the turning module utilizes the Operation Navigator to manage operations and parameters. They enable programmer s to create roughing, finishing, centerline drilling, and threading operations. Parameters such as spindle definition, workpiece geometry, machining methods, and tools are specified as groups with parameters shared among operations. Other parameters are defined within the individual operations. when creating a CAM program, a programmer usually follow the following workflow. Some details are omitted to make the workflow brief and clear.

To setup, assemble the solid models that represent the part and workpiece (blank) to be machined. Set the program zero(s) - The Machine Coordinate System. Identify PART and BLANK geometry to the CAM system. To create a machining program, a programmer needs to create a series of turning operation in sequence, such as creating a Facing opertion, a Centerline Drilling Operation, a Roughing Operation, a Finishing Operation, a Teach Mode Operation, a Grooving Operation, a Threading Operation etc.

To create a functional operation, the programmer needs to define or select a tool to be used, select the proper cut strategy which defines the cut pattern such as parallel cuts, and determine the cut region. Then, the programmer needs to set the path settings, which include feeds, speeds and cut depths setting. And this is point where Self-learning Optimization of turning process parameters strike in to make the process easier and better. Basically, the Self-learning Optimization module detects the defined cut region, cut strategy and obtains the geometry features of cut region and determines the feeds, speeds and cut depths based on its prediction and optimization result. The Self-learning Optimization module calls different prediction networks corresponding to different devices. Thus, when calling the Self-learning Optimization module, the device ID should be selected from the device lists.

The workpiece progresses through the program, In process workpiece tracking computes and graphically displays the total remaining material to be removed. The turning module graphically display the in process workpiece after each operation generated. The in process workpiece is defined by the total material removed for all operations in sequence up to the currently selected operation.

When programming is finished and checked, post processing needs to be executed, which creates Shop documentation and machining program.

With the Self-learning Optimization module embedded in NX CAM, programmers can optimize the cut parameters when they create a new operation. Also, thay can optimize the parameters after they create the operations. A rough turning process and the following finish turning process can be optimized together by using joint optimization. The workflows can be shown as Fig. 1.

III. Optimization Method

A. Determination of Optimization Algorithm

Commonly used multi-objective optimization algorithms mainly include non-dominated sorting genetic algorithm (NSGA-II), multi-objective particle swarm optimization (MOPSO), and multi-objective evolutionary algorithm based on decomposition (MOEA/D). The NSGA-II algorithm was proposed by Kalyanmoy Deb et al. in 2002. The algorithm firstly performs an evolutionary operation on the individuals in the parent population P through genetic operators, and then generates the progeny population Q. The parent and child populations are then merged to create a new population R. The individuals in the population R are then subjected to fast non-dominated sorting to calculate their Pareto rank. Among them, the individual with a Pareto rank of 1 is the Pareto optimal solution. To ensure the uniformity of the population, the NSGA-II algorithm proposes the concept of congestion and ranks the individual in each Pareto rank. In order to preserve the elite individuals in the population, the NSGA-II algorithm adopts the elite retention strategy, and selects the individuals with lower Pareto rank and greater congestion from the population R as the new generation population P', and repeats the above operations until the number of algorithm iterations reaches The maximum number of iterations.

The performance of the optimization algorithm is evaluated by the convergence and uniformity of the Pareto optimal solution set obtained by the multi-objective optimization algorithm. The convergence of the solution set refers to the distance between the Pareto optimal solution set calculated by the multi-objective optimization algorithm and the real Pareto optimal solution set. The smaller the convergence is, the closer the Pareto optimal solution set of the multi-objective optimization algorithm is to the real Pareto solution set, and the better the convergence of the algorithm is.

This paper selects the convergence metric, the inverse generation distance (IGD) and the Spacing indicator to evaluate the performance of each algorithm.

The convergence metric is the distance from each point in the Pareto optimal solution set P to the standard Pareto solution set Q. The convergence metric can evaluate the convergence of the solution set. The smaller the value is, the better the convergence is. Its definition is as follow:

$$Y = \frac{\sum_{x \in P} distance(x, Q)}{|P|} \tag{1}$$

Where |P| is the number of solutions in the solution set P, and distance(x, Q) is the minimum distance of the solution x of set P to set Q, which is defined as follow:

$$distance(x, Q) = min|x - y|\ ,\ \ y \in Q \tag{2}$$

The IGD indicator is the distance from each point in the standard Pareto solution set Q to the Pareto optimal solution set P. The definition of IGD is as follow:

$$IGD = \frac{\sum_{x \in Q} distance(x, P)}{|Q|} \tag{3}$$

Where |Q| is the number of solutions in the solution set P. The smaller the indicator is, the better the convergence is.

The spacing indicator is defined as follow:

$$Spacing(P) = \sqrt{\frac{1}{|P|-1} \sum_{i=1}^{|P|} \left(\bar{d} - d_i\right)^2} \tag{4}$$

Where d_i represents the shortest distance from the ith solution to the other solutions. The smaller the indicator is ,the better the uniformity is.

Zitzler E, Deb K, Thiele L et al. proposed a standard ZDT test function. The ZDT test function is mainly used to test multi-objective optimization algorithms with a small number of optimization variables. The performance of NSGA-II, MOPSO and MOEA/D algorithms is tested by ZDT1, ZDT2, ZDT4 and ZDT6 test functions. The standard Pareto reference of each function is used. In this paper, the cut parameters are optimized, and the number of optimization variables is three. Therefore, the variable dimensions of the ZDT function is three. Comparison results are shown in TABLE I-IV.

TABLE I. PERFORMANCE COMPARISON ON ZDT1

Algorithms	ϒ (*10^-4)	IGD(*10^-3)	Spacing(*10^-3)
NSGA-II	9.308	2.402	2.769
MOPSO	24.11	15.59	21.87
MOEA/D	9.117	3.672	4.941

TABLE II. PERFORMANCE COMPARISON ON ZDT2

Algorithms	ϒ (*10^-4)	IGD(*10^-3)	Spacing(*10^-3)
NSGA-II	8.146	2.641	2.672
MOPSO	21.20	12.33	13.99
MOEA/D	8.138	3.816	3.551

TABLE III. PERFORMANCE COMPARISON ON ZDT4

Algorithms	ϒ (*10^-3)	IGD(*10^-3)	Spacing(*10^-3)
NSGA-II	2.434	2.206	2.533
MOPSO	1494	186.0	216.7
MOEA/D	131.2	128.7	10.11

TABLE IV. PERFORMANCE COMPARISON ON ZDT6

Algorithms	ϒ (*10^-4)	IGD(*10^-3)	Spacing(*10^-3)
NSGA-II	8.301	1.556	1.978
MOPSO	1.358	7.076	42.13
MOEA/D	3.242	1.565	4.919

From the comparison results, we can clearly see the advantage of NSGA-II algorithms on three dimensional multi-objective optimization problems. Thus, we select NSGA-II algorithm in the multi-objective optimization of cut parameters.

B. Objective and Constrains

Commonly, the cut parameters optimization model takes machining time and machining cost as objective. In some cases, surface roughness and cut force can also be the optimization objective. This paper establishes a multi-objective optimization model taking machining time, machining cost, cut force and surface roughness as the optimization objective, taking spindle speed, feed rate, cut speed and surface roughness as constraints. Programmers can set the objective and constrains of the optimization model, as they need.

In single pass turning or finish turning operation, the machining time can be calculated as follow:

$$t = \frac{L}{Sf} \tag{5}$$

Where L is the cut length, S is the spindle speed, and f is the feed rate.

For multi-pass machining, the distribution of cut depth should be optimized. To deal with multi-pass machining optimization, the optimization model first optimizes the finish pass to fulfill the surface roughness requirement. The rough turnings are optimized by equally dividing the remaining cut depth. Assuming that the remaining cut depth is H, cut depth of each pass is a_p, the number of passes is H/a_p. Then the machining time of multi-pass machining is as follow:

$$t = \left\lceil \frac{H}{a_p} \right\rceil \cdot \frac{L}{Sf} + t_{fin} \tag{6}$$

In the turning process, the machining cost mainly comes from time cost and tool cost. Among them, the time cost includes the cost of the machine, the wages of the workers, etc. The tool cost is the cost of replacing the tool after the tool wears or breakage. The objective function of the machining cost is as follow:

$$C = C_m \frac{L}{Sf} + C_T \frac{t_m}{T} \tag{7}$$

Where C_m is the time cost of each hour, C_T is the cost of the tool, T is the service life of the tool.

Cut force and surface roughness are also used as constrains or objective in the optimization model. The cut force and surface roughness prediction model are established by using BP-ANNs, which are described in chapter IV.

IV. PREDICTION MODEL

This paper established main cut force and surface roughness prediction model based on BP-ANNs. Since BP neural network can approximate arbitrary functions, it is common to use BP neural network to fit various nonlinear models. In addition, factors such as tool wear will change with time during the machining process, which will affect the main cut force and surface roughness. The neural network can be iterated continuously to reflect the actual situation. Main cut force is an important characteristic variable to be monitored and predicted. To monitor and predict and cutting forces, this paper collected the torque current signal from the spindle motor controller. The controller of the experimental machine tool uses the vector control algorithm to control the spindle motor. For a three-phase AC asynchronous motor, the spindle torque output T_e is as follow:

$$T_e = n_p \frac{L_m}{L_r} i_{st} \Psi_r \tag{8}$$

Where n_p is the pole pair number of the motor, L_m is the magnetizing inductance, which is the mutual inductance between the stator and the rotor, L_r is the rotor self-inductance, i_{st} is the torque current; Ψ_r is the flux linkage of the rotor. n_p, L_m, L_r are constants. The flux linkage Ψ_r usually changes when the spindle speed changes, and is stable when the speed is stable. The cutting process usually takes place at a stable speed. So we can approximate that the spindle torque T_e is proportional to the torque current i_{st} in the cutting process. That is:

$$T_e = K \cdot i_{st} \tag{9}$$

Transmission system of the machine tool is subjected to cutting force and friction. The equation of motion for the system is:

$$T_e = J_s\left(\frac{d\omega_s}{dt}\right) + B_s\omega_s + T_{fs} + T_c \tag{10}$$

Where T_{fs} is the friction torque, J_s is the rotational inertia of the machine drive system, ω_s is the angular velocity of the spindle, and B_s is the viscous damping coefficient. The friction torque T_{fs} is composed of the idling friction torque T_{fs0}, the coulomb friction torque δT_{fs} and the viscous friction torque δT_{bs}. The equation can be expressed as:

$$T_{fs} = T_{fs0} + \delta T_{fs} + \delta T_{bs} \tag{11}$$

When the machine tool is working in a pass, Spindle speed can be considered constant, thus,

$$T_{ew} = K \cdot i_w = B_s\omega_s + T_{fs} + F_c \cdot \frac{D}{2} \tag{12}$$

Where F_c is the main cut force, I_w is the torque current when machine tool is working in a pass.

We ignore the influence of cut force on friction torque. Thus, when the machine tool is idling,

$$T_{ei} = K \cdot i_i = B_s\omega_s + T_{fs} \tag{13}$$

Where I_i is the torque current when machine tool is idling.

Taking $\Delta I = I_w - I_i$, we have:

$$K \cdot \Delta I = T_c = F_c \cdot \frac{D}{2} \tag{14}$$

Considering above equation, we can estimate the main cut force using the difference between the machining and the idle current of the motor.

So, we use the spindle rate, feed rate, cut depth and turning diameter as input parameters of the ANNs, to predict the torque current of the motor, which indicates the main cut force. The prediction model of main cut force or torque current is used as objective or constrains of the optimization model. The model can be expressed as follow:

$$I_w = BPN_I(v_c, a_p, f) \tag{15}$$

$$F_c = \frac{2K \cdot (I_w - I_i)}{D} \tag{16}$$

As for the prediction model of surface roughness, This paper takes tool nose arc radius, main cut force, cut speed, feed rate, and depth of cut as input parameters of the network. The model can be expressed as follow:

$$Ra = BPN_{Ra}(r, F_c, v_c, a_p, f) \tag{17}$$

If the accurate numeric relationship between F_c and ΔI is not clear, ΔI can be used as input parameters of surface roughness prediction model. The errors show no significant difference. It is also reasonable and appropriate to use ΔI as the objective or constrains of the optimization model in light of the proportional relationship between F_c and ΔI.

This paper uses the output of BPN_I as one of the input parameters of BPN_{Ra}. The input/output dataset of the model is illustrated schematically in Fig. 2.

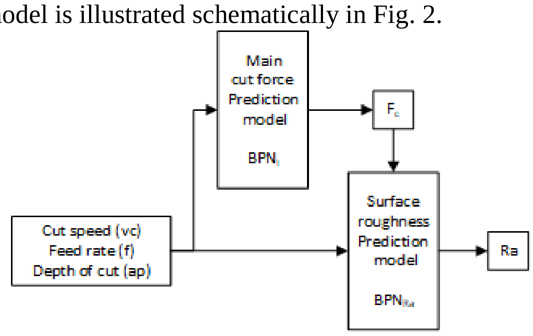


Figure 2. The input/output dataset of the model

The performance capability of each network was examined based on the correlation coefficient between the network predictions and the experimental values. The experiment was carried standalone and its data was not used in the training of prediction model.

Correlation between the predicted outputs of the neural network model and the experimental data for main cut force prediction model is shown in Fig. 3.

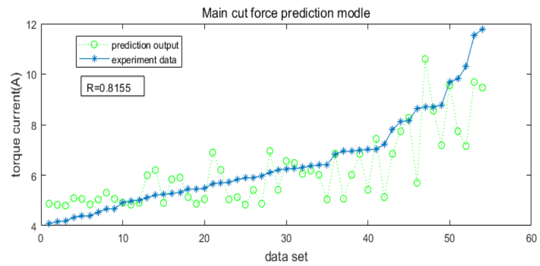


Figure 3. Correlation between the predicted outputs and the experimental data of main cut force

Correlation between the predicted outputs of the neural network model and the experimental data for surface roughness prediction model is shown in Fig. 4.

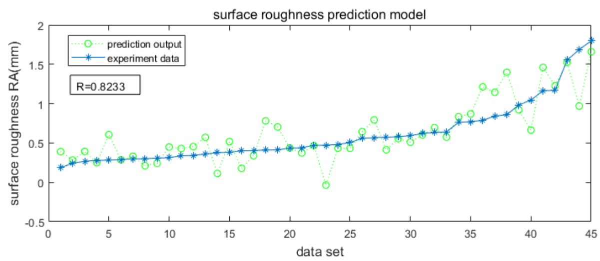


Figure 4. Correlation between the predicted outputs and the experimental data of surface roughness

V. Conclusion

This paper developed a self-learning turning optimization module and deployed it in the NX CAM software. The turning optimization module can optimize the machining time, machining cost etc. with the constraints of main cut force and surface roughness.

The experiments on certain aluminum alloy workpieces show improvement on optimization objectives and roughness and cutting force requirements are met.

Conflict of Inter est

The submitted work was carried out with the support of Siemens China Research Institute.

Author Contributions

Shenghao. Shi was the main accomplisher of this work, completing theoretical derivation, algorithm design, software development and so on. Prof. Hui. Zhang gave important instruction in the research direction and gave support to the experimental. Prof. Peng. Mou put forward a lot of suggestions on the writing of the paper.

Acknowledgment

The authors wish to thank Xuli. Liu. Xuli. Liu. made great effort in the developing of the optimization module of this work.

References

[1] M. Chandrasekaran, M. Muralidhar, C. M. Krishna, and U. S. Dixit, "Application of soft computing techniques in machining performance prediction and optimization: A literature review," *International Journal of Advanced Manufacturing Technology*, pp. 445-464, 2010.

[2] R. H. Todd, and D. K. Allen, *Manufacturing Processes Reference Guide*.

[3] S. K. Pal and D. Chakraborty, "Surface roughness prediction in turning using artificial neural network," *Neural Computing & Applications*, pp. 319-324, 2005.

[4] Ö. Tugrul, and K. Yigit, "Predictive modeling of surface roughness and tool wear in hard turning using regression and neural networks," *International Journal of Machine Tools & Manufacture*, pp. 467-479, 2005.

[5] E. O. Ezugwu, D. A. Fadare, J. Bonney, R. B. D. Silva, and W. F. Sales, "Modelling the correlation between cutting and process parameters in high-speed machining of Inconel 718 alloy using an artificial neural network," *International Journal Of Machine Tools & Manufacture*, pp. 1375-1385, 2005.

[6] X. Li, P. K. Venuvinod, and M. K. Chen, "Feed cutting force estimation from the current measurement with hybrid learning," *International Journal of Advanced Manufacturing Technology*, pp. 859-862, 2000.

Shenghao. Shi. Completed undergraduate study at Tsinghua University with a bachelor's degree in mechanical engineering. I am studying for a master's degree of mechanical engineering at Tsinghua University. Research interests include CAM, CNC and sustainable manufacturing.