



Optimization Design of Cutting Parameters Based on the Support Vector Machine and Particle Swarm Algorithm

Chengwen Yang, Hao Jiang, Baolin Liu

The College of Nuclear Technology and Automation Engineering, Chengdu University of Technology, Chengdu, China
Email: mxy_1995@sina.com

How to cite this paper: Yang, C.W., Jiang, H. and Liu, B.L. (2020) Optimization Design of Cutting Parameters Based on the Support Vector Machine and Particle Swarm Algorithm. *Open Access Library Journal*, 7: e6788.

<https://doi.org/10.4236/oalib.1106788>

Received: September 4, 2020

Accepted: September 26, 2020

Published: September 29, 2020

Copyright © 2020 by author(s) and Open Access Library Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In the field of machining, the processing design of the cutting parameters has the characteristics of greater subjectivity and experience. For obtaining the ideal roughness of surface (Ra), the number of machining experiments is carried out. It makes a lot of waste of materials, labor, energy, and so on. In addition, there is a highly non-linear function relationship between the three elements of cutting: cutting speed (v_c), feed (f), cutting depth (a_p) and roughness of the surface. It is hard to use mathematical equations to express the relationship clearly. So, in this paper, the support vector machine (SVM) will be used to establish the model of cutting elements and roughness of the surface. Then, taking roughness of surface as optimization goal and cutting elements as optimization parameters, the particle swarm algorithm (PSO) will be carried out to obtain a group of cutting parameters for the ideal roughness surface. It provides an easy, accurate, and feasible optimization design method for machining cutting parameters optimization design.

Subject Areas

Mechanical Engineering

Keywords

Particle Swarm Algorithm, Support Vector Machine, Cutting Parameters Optimization Design

1. Introduction

Stainless steel is an excellent high-performance alloy steel, which can resist corrosion in the air or in chemical corrosive media, and has a wide range of appli-

cation prospects. Stainless steel has strong toughness, low thermal conductivity, and serious work hardening, which leads to defects such as large cutting force, high cutting temperature, and tools that are prone to adhesion and wear. Therefore, stainless steel is a difficult-to-machine metal material [1]. When turning 304 stainless steel, Support vector machine (SVM) and Particle swarm algorithm (PSO) are used to optimize the cutting parameters [2]. The cutting parameters are mainly the three elements of cutting, including cutting speed (v_c), feed (f), and back-cutting (a_p). The optimization target is the surface roughness of the workpiece in order to achieve the expected surface quality.

2. Support Vector Machine Theory (SVM)

Data-based machine learning is an important aspect of modern intelligent technology. Machine learning is essentially an approximation of a true model of a problem. Research is to start from observation data (samples) to find the rules used to predict unknown data.

Support vector machine (SVM) is a machine learning method developed in the mid-1990s. This method is based on the statistical learning theory. This method is based on statistical learning theory, which improves the generalization ability of learning machine by seeking the structural risk minimization, and completes the minimization of empirical risk and confidence range. Therefore, when the number of statistical samples is small, we can also obtain the purpose of good statistical laws. Because of its outstanding learning performance, this field has become the focus of many scholars. This technology has also become a research hotspot in the machine learning community and has been successfully applied in many fields, such as face recognition, handwritten number recognition, automatic text classification, and machine translation, etc. [3].

The basic idea of SVM is to use the kernel function to map the input sample space to the high-dimensional feature space, find an optimal classification surface in the high-dimensional space, and obtain the nonlinear relationship between input and output variables [4].

Assume that the training data set $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, in a given feature space, where $x_i \in R^n$ is the i -th feature vector, also called is an example; $y_i \in \{1, -1\}, i = 1, 2, \dots, N$. N is the class label of x_i , when $y_i = 1$, x_i is called a positive example, when $y_i = -1$, x_i is called Negative case. (x_i, y_i) is the sample point. The key of the algorithm is to establish a classification hyperplane as the decision surface, which maximizes the separation edge of positive and negative examples. The classification hyperplane is to find the function:

$$\varphi(w) = \frac{1}{2} \|w\|^2 \quad (1)$$

$$\text{s.t. } y_i(w \cdot x_i + b) \geq 1, i = 1, 2, \dots, N$$

where: w is the normal vector of the hyperplane, b is the constant term of the hyperplane, x_i is the training sample, and y_i is the type of the sample.

In practice, there may be linear inseparability. At this time, it can be mapped

to a high-dimensional space. When the sample is linear and inseparable, mapping it to a high-dimensional space will have a particularly large dimension, which makes calculation difficult. At this time, the kernel function plays an important role in dealing with the problem. Its value lies in the fact that although the feature is also converted from low-dimensional to high-dimensional, the difference is that the method will calculate in low-dimensional in advance, and then it will essentially. The classification effect is manifested in high dimensions, thus avoiding complicated calculations directly in high-dimensional space.

In practical applications, we often rely on prior domain theoretical knowledge to select an effective kernel function. The widely used kernel functions mainly include:

Polynomial kernel and Gaussian kernel linear kernel function:

$$k(x_1, x_2) = (\langle x_1, x_2 \rangle + R)^d \quad (2)$$

$$k(x_1, x_2) = \exp\left\{-\frac{\|x_1 - x_2\|^2}{2\sigma^2}\right\} \quad (3)$$

$$k(x_1, x_2) = \langle x_1, x_2 \rangle \quad (4)$$

According to the different problems and data, choosing different parameters will actually get different kernel functions. At the same time, different selection of kernel function parameters will directly affect the prediction accuracy and classification performance of the support vector machine.

Particle Swarm Optimization Algorithm (PSO)

Particle swarm optimization (PSO) was first proposed by Kennedy [5] and Eberhart [6] in 1995, and its basic concept originated from the study of bird flock foraging behavior. The main principle of the particle swarm algorithm is: based on the observation of animal group movement behavior, the information sharing of individuals in the group is used for reference to make the movement of the entire group occur in the problem solving space, the evolution process from disorder to order. The optimal solution is finally obtained. This algorithm is a global parallel optimization algorithm. Compared with other optimization algorithms, it has the advantages of short evolution time and high optimization accuracy [7]. Since the particle swarm algorithm was proposed, it has been successfully applied to solve the traveling salesman problem, capacitor allocation problem [8] and machine learning [9] and other related fields.

The PSO algorithm is to simulate the intelligent search of the community generated by the mutual cooperation and competition between bird groups [10]. The algorithm simulates the solution of the optimization problem as particles in the search space, these particles can ignore their own volume and mass, but fly at a definite speed [11]. The particle selects the appropriate calculation function to find its own fitness, and dynamically adjusts its flight speed and corresponding points through the current best fitness value of itself and the community, and

finally finds the global best. The particle adjustment speed V and the point position X are shown in formula (5) respectively.

$$\begin{cases} V_i^{k+1} = wV_i^k + c_1r_1(p_i^k - X_i^k) + c_2r_2(p_g^k - X_i^k) \\ X_i^{k+1} = X_i^k + V_i^{k+1} \end{cases} \quad (5)$$

In the formula, w is the inertia weight, v is the current iteration number, $i = 1, 2, \dots, m$; r_1, r_2 is a random number uniformly distributed in the interval $[0,1]$, c_1, c_2 is the learning factor, p_i^k is the optimal solution for the K -th iteration, and p_g^k is the global optimal so far solution.

3. Optimize the Design Process

3.1. Data Preparation

This article has 49 sets of experimental data, some of which are shown in **Table 1**.

Among them, a_p , v_c and f are used as input variables, and the surface roughness Ra is used as the output variable to randomly divide the training set and the test set. After the 49 groups of samples were randomly shuffled, the first 40 groups of samples were used as the training set, and the last 19 groups were used as the test set. The overall algorithm flow is shown in **Figure 1**.

3.2. SVM Modeling

The SVM regression toolbox in Matlab is used to set it, and the input training set samples are automatically standardized by the solver to remove the influence of dimensions. The kernel function adopts the Gaussian kernel function, which automatically adjusts the super error in the optimization solver. The mean square error and coefficient of determination are used as indicators to evaluate the accuracy of the model. The modeling results are shown in **Figure 2**.

The test set is input into the model for testing, and the comparison of model prediction results is shown in **Figure 3**. It can be seen from **Figure 4** that the SVM regression model has high accuracy, its mean square error $mse = 0.059$, and the coefficient of determination $R^2 = 0.908$, which can be used to characterize the mapping relationship between the three elements of cutting and surface roughness.

Table 1. Experimental sample data.

NO.	a_p (mm)	v_c (mm/rev)	f (mm)	Ra (μm)
1	0.40	250	0.125	2.45
2	0.25	350	0.075	1.09
3	0.35	200	0.050	0.99
4	0.20	200	0.075	0.90
5	0.20	300	0.050	0.69

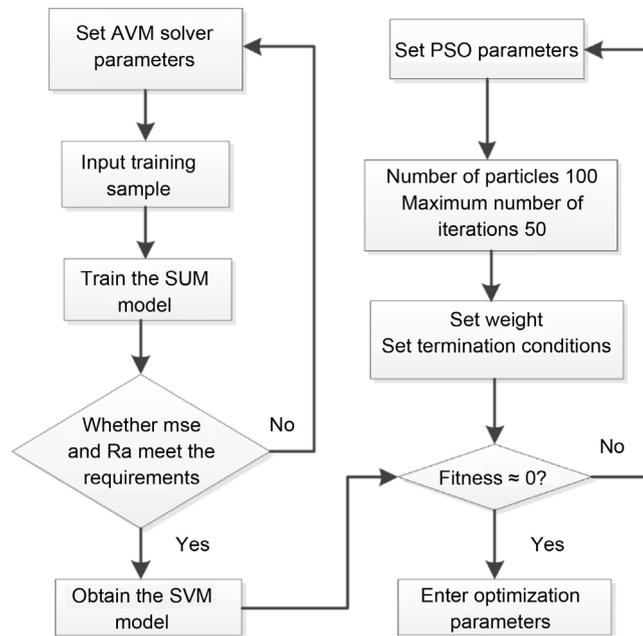


Figure 1. SVM-PSO algorithm flow.

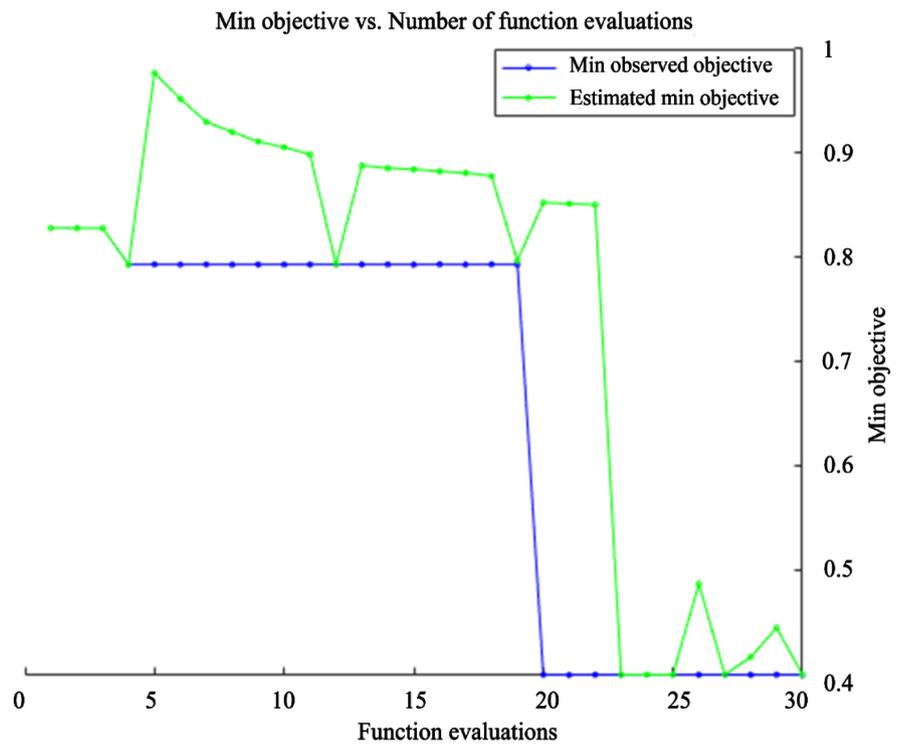


Figure 2. SVM modeling process.

$$Fitness = \left| 0.5 - SVM(a_p, v_c, f) \right| \tag{6}$$

4. Conclusions

In the current cutting process, the process design of cutting parameters has

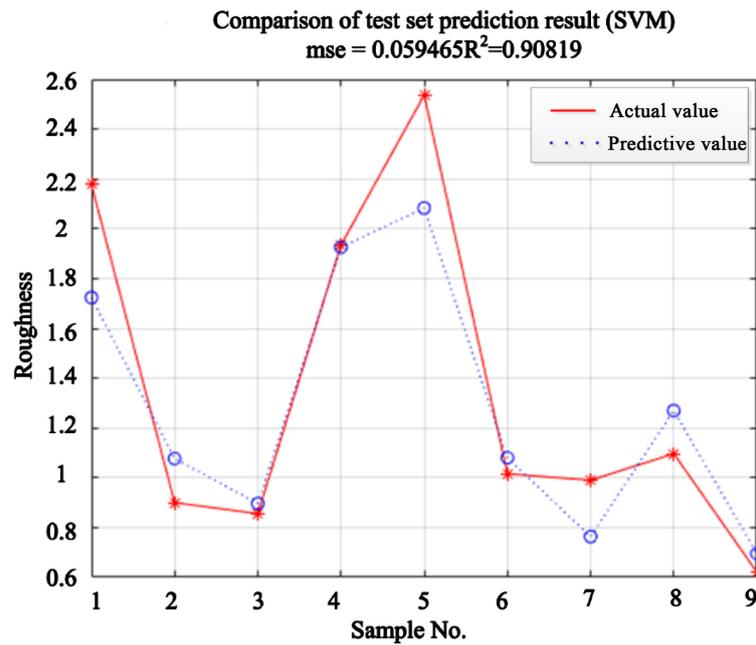


Figure 3. SVM modeling results.

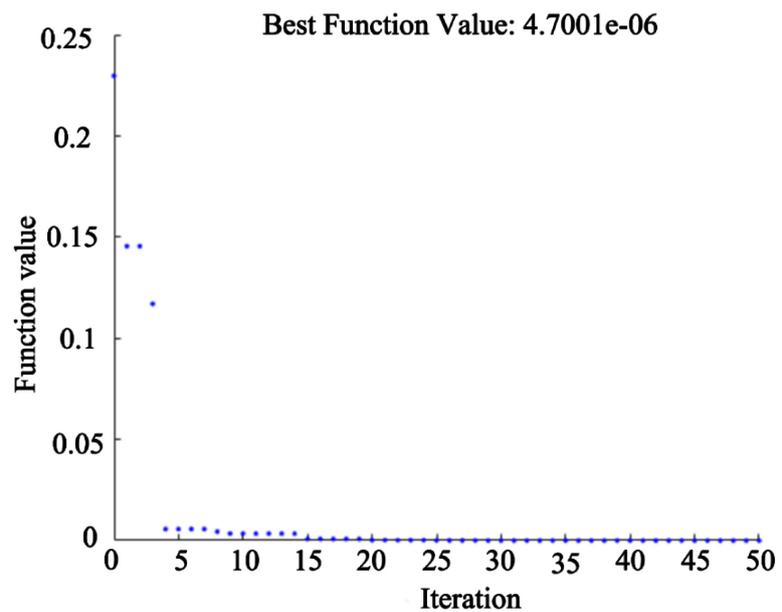


Figure 4. PSO optimization process.

greater subjectivity and experience. In order to achieve the expected surface roughness of the workpiece processing, this paper uses a support vector machine (SVM) to establish a model between the three elements of cutting and the surface roughness. Using the particle swarm algorithm, taking the surface roughness as the optimization goal, through parameter analysis and optimization, the following conclusions are obtained:

1) For difficult-to-machine materials such as 304 stainless steel, Support vector machine (SVM) is used for modeling. The mapping model between cutting

parameters and surface roughness has higher accuracy and smaller error, which can provide reference for parameter optimization or theoretical derivation.

2) The particle swarm algorithm PSO is used to optimize the cutting parameters, and a set of feasible optimized cutting parameters can be obtained according to the expected surface roughness value, which provides a rationale for the process personnel when designing processing parameters and saves experimental costs.

3) Compared with other intelligent modeling algorithms and optimization algorithms, SVM-PSO is more convenient, concise and efficient, and provides a new idea for process parameters optimization design.

Acknowledgements

I would like to express my gratitude to all those who have helped me during the writing of this thesis. I gratefully acknowledge the help of my coworker Liu. I do appreciate his patient encouragement, and professional suggestions during my thesis writing.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Luo, Y.S., Qin, M.Y. and Li, Y.Z. (2014) Experimental Study on Residual Stress in Stainless Steel Cutting. *Modern Manufacturing Engineering*, No. 3, 79-81+104.
- [2] Zhang, G.X., Ji, W.X. and Wang, Y. (2016) Research on Optimization of CNC Machining Parameters for Green and Efficient Manufacturing. *Machine Design and Research*, No. 32, 127.
- [3] Wang, L.J. and Ding, S.F. (2019) A Particle Swarm Optimization SVM-ELM Model. *Journal of Computer Science and Research*, **13**, 657-665.
- [4] Li, S., Yuan, Z.G., Wang, C., et al. (2018) Overview of Swarm Intelligence Algorithm Optimization Support Vector Machine Parameters. *Journal of Intelligent Systems*, **13**, 70-84.
- [5] Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks*, Perth, 27 November-1 December 1995, 1942-1948. <https://doi.org/10.1109/ICNN.1995.488968>
- [6] Eberhart, R. and Kennedy, J. (1995) A New Optimizer Usingparticle Swarm Theory. *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, Nagoya, 4-6 October 1995, 39-43. <https://doi.org/10.1109/MHS.1995.494215>
- [7] Zhang, Y. and Liu, Y.C. (2009) Traffic Forecasting Using Least Squares Support Vector Machines. *Transportmetrica*, **5**, 193-213. <https://doi.org/10.1080/18128600902823216>
- [8] Shen, J.H. and Wang, K. (2012) Hybrid Particle Swarm Optimization Algorithm for Traveling Salesman Problem. *Journal of Intelligent Systems*, **7**, 174-182.
- [9] Wang, X.B., Luo, F.J., Sang, C.Y., et al. (2017) Personalized Movie Recommendation

System Based on Supportvector Machine and Improved Particle Swarm Optimization. *IEICE Transactions on Information and Systems*, **E100-D**, 285-293.

<https://doi.org/10.1587/transinf.2016EDP7054>

- [10] Wei, J.J. (2015) Improvement and Application of Particle Swarm Optimization Algorithm. Taiyuan University of Technology, Taiyuan.
- [11] Li, N. (2006) Theoretical Analysis and Application Research of Particle Swarm Optimization Algorithm. Huazhong University of Science and Technology, Wuhan.