

Optimization of tool wear in turning using genetic algorithm

C Felix Prasad, S Jayabal & U Natarajan

Department of Mechanical Engineering, A C College of Engineering and Technology, Karaikudi 630 004, India

Received 25 January 2007; accepted 11 December 2007

Optimization has significant practical importance particularly for operating in machines. In order to increase the accuracy of finishing product the tool must be in good condition always as much as possible. To achieve good condition of tool the machining parameters like speed, feed and depth of cut should be optimized²⁻⁴. This paper aims to increase the condition of tool, i.e., minimization of tool wear by applying the optimized input parameters using Genetic Algorithm technique. For solving the machining problems various traditional techniques such as Integer programming, Quadratic programming and Simplex method have been used so far. But results of these various methods vary to some extent. Genetic Algorithm helps in obtaining better results than of traditional methods available.

Keywords: Optimization, Tool wear, Genetic Algorithm, Input parameters

The growing demand for product quality and economy necessity have forced incorporation of monitoring the process parameters in automated manufacturing system. The greatest limitation of automation in the machining operation is tool wear. If the tool wear increases then the tool life will be minimum. So the tool wear has been optimized by selecting the optimal cutting parameters such as speed and depth of cut. The main machining parameters which are to be considered as variables of the optimization are speed, feed and depth of cut the required output is minimum tool wear³.

The optimum set of these three parameters are determined for a particular job-tool combination of High Carbon Steel-Tungsten Carbide during turning which optimizes the tool life and tool wear. Many graphical and analytical techniques were initially used for optimization⁴. Some of them are overcome by numerical optimization methods which may be either Simplex or Gradual based methods. These and other similar numerical optimization techniques have been found to give reasonably accurate results only for simple problems¹.

But in case of highly non-linear optimization problems, the numbers of iterations are required to get optimum point, thus consuming significant computation time²⁻⁴. So an alternative method for optimization is probabilistic methods. Genetic Algorithm is one of the example of probabilistic methods. Genetic Algorithm searches from a population and gives a set of reasonably good solution to the user.

Prediction Techniques

The mathematical modeling is done using simple probabilistic considerations and design of experiments^{5,6}.

Design of experiments

The statistical design of experiments is the process of planning the experiments so that appropriate data could be collected which may be analyzed by statistical methods resulting in valid objective conditions³⁻⁶. Here central composite method 3³ factorial design is used to develop a model for the tool wear. In this work three factors are used, each factor is at three levels, arranged in a factorial experiment, i.e., 3³ factorial designs. The three levels of factors are referred to as low (-1), medium (0) and high (+1). In the present work cutting speed, feed and depth of cut have been used as input variables.

Genetic algorithm

Genetic algorithm is a global population search technique based on the operations of natural genetics and mimics the natural biological process^{8,9}. In this work for a given optimization problem Genetic Algorithm first encodes all variables into a finite bit binary string called as chromosomes. Chromosomes represent a possible solution to the optimization problem. A population of chromosomes is formed. Each chromosome is decoded and is evaluated according to the fitness function.

Fitness function

The fitness function plays very important roll in Genetic algorithm. Genetic algorithms are naturally

suitable for solving maximization problems. But minimization problems are usually transformed into maximization problems by using suitable transformation^{5,6}. So generally, the fitness function $f(x)$ is derived from the objective function. For maximization problems the fitness function can be considered to be same as that of objective function.

$$\text{i.e., } F(x)=f(x)$$

For minimization problems the fitness function is an equivalent maximization problem chosen such that the optimum point remains unchanged^{4,7}. So for minimization problem the fitness function is,

$$F(x)=1/(1+f(x))$$

This fitness transformation will not alter the location of the minimum. This fitness function value of a string is known as the string's fitness.

Genetic algorithms are ideally suited for unconstrained optimization problems. As the present problem is a constrained optimization one, it is necessary to transform it into an unconstrained problem to solve it using genetic algorithms⁴⁻⁷. To handle constraints, penalty function methods have been mostly used. Traditional transformations using penalty functions are not appropriate for genetic algorithm. A formulation based on the violations of normalized constrained is proposed in this paper.

Coding

To solve any problem in order to use Genetic Algorithm variables must be coded in some string structures. Binary coded strings having 1's and 0's are mostly used. Usually, the length of the string is determined with respect to the solution accuracy derived.

After the coding has been done, the

$$\text{Accuracy} = \frac{(x_i^{(U)} - x_i^{(L)})}{2^{l_i}}$$

where l_i is the length of substring, $x_i^{(L)}$ is upper bound of parameter, and, $x_i^{(U)}$ is lower bound of parameter corresponding point $x = (x_1, x_2, \dots, x_N)^T$ can be found using the following equation

$$x_i = x_i^{(L)} + \frac{x_i^{(U)} - x_i^{(L)}}{2^{l_i} - 1} \times \text{decoded value}$$

finally $f(x)$ values can also be calculated

Genetic operators

Reproduction

First of all the reproduction operator is applied on a population. Which selects best strings in a population and forms a mating pool. So it is also names as selection operator. In reproduction good strings are probabilistically assigned a larger number of copies in a population and a mating pool is formed. In this reproduction phase, there is no new string formed.

Crossover

In crossover operator, randomly two strings are picked from the mating pool after that some portions of the strings are exchanged between the strings. In which uniform crossover and single point crossover are there. A single point crossover operator is carried out by randomly choosing a crossing site along the string after that all bits on the right side of the crossing site are exchanged.

For example

$$\begin{array}{cc|cc} 00010 & 10001 & \Rightarrow & 00010 & 01110 \\ 10100 & 01110 & & 10100 & 10001 \end{array}$$

In the above example two participated strings are known as parent strings and the new resulting strings are known as children strings. In which the crossover is carried out at fifth site. The effect of this process may be beneficial or detrimental. This method is only responsible for creating reasonably new better strings.

Mutation

Mutation is used to make a point neighborhood of the current point so that there is a chance of achieving a local search around current solution⁴⁻⁷. This operator changes from 0 to 1 and vice versa with a small mutation probability, which is also needed to maintain diversity in the population. Consider one example

```
1001000110
1110010101
1010101010
1000100001
```

In the above all four strings have '1' in the leftmost bit position. But the required optimum solution is '0' in that all position. In that situation both the reproduction and crossover operators will not be able to create '0' in that position. Only if we introduce a mutation operator then it will create probability of turning '1' into '0'.

Violation parameter

In this paper, the present problem is a constrained optimization but using genetic algorithm it is possible if the problem is unconstrained optimization. In order to nullify the effect of constrained a parameter “ a ” is used called violation parameter.

$$\text{Obj}(x) = f(x) + a \text{ (constraints)}$$

where $\text{Obj}(x)$ is a penalty function

Before evaluating the fitness function the values are checked for constraint violation. If the designed variable set violated the constraint then a higher value say 10000 will be assigned and if not a lower value say 1.0 is assigned for violation parameter. By doing so, the design variables set that violates the constraints. If evaluated for objective function will give a very high value and thus results in a very low fitness function [$f(x) = \{1/(1+\text{obj}(x))\}$]. This will reduce the probability of selection of this particular set in the next generation and so.

Reproduction operator selects only good strings. Crossover operator recombines good strings and mutation operator changes a string to create a better string locally. The new created string either beneficial or detrimental. If it is worst string then it will be eliminated by the reproduction operator in the next generation. After that the string will get improved in coming generations.

Development of mathematical model

A 3^3 factorial design of experiment had been used to develop a model for tool wear. For that the following relationship could be considered as

$$y = f(v, f, d) + \epsilon$$

where v , f and d are speed, feed and depth of cut of machining parameters respectively. ϵ represents the error which is normally distributed with zero mean with respect to observed response y .

$$f(v, f, d) = \eta$$

where η is the response of tool wear.

$$t_w = c_1 v^{k_1} f^{k_2} d^{k_3}$$

The above equation is representing the relationship between tool wear and independent variables. c_1 is constant. By using SPSS software the constant values of c_1 is found as 0.0512. The values of parameter as

follows : $k_1=0.588$, $k_2=0.034$, $k_3=0.667$; where k_1 , k_2 and k_3 are parameters.

Objective function

Tool wear equation minimization

$$t_w = 0.0512 v^{0.588} f^{0.034} d^{0.667}$$

In this work the minimum tool wear had been found out at optimized set of parameters.

Constraints

The constraints represent some functional relationship among variables and other parameters satisfying some physical phenomenon and certain resource limitations^{5,6}. The above objective functions are subjected to the following constraints.

Cutting force

The cutting force calculated should be kept smaller than the 1100 N.

$$f_c = 844 v^{-0.1013} f^{0.725} d^{0.75} \leq 1100$$

Temperature

The temperature should be kept less than 500°C

$$t = 75.0238 v^{0.527} f^{0.836} d^{0.156} \leq 500$$

Optimization

Optimization is the act of obtaining the best solution under given circumstances⁷. It can be defined as the process finding the condition (parameter values) that gives the maximum and minimum value of a function.

Optimized variables

The variables have to be optimized are speed, feed and depth of cut. Higher speed affects tool life where a lower speed value increases the tool life and decreases the tool wear.

Similarly the feed rate and depth of cut also affect the tool life. So the speed, feed rate and depth of cut have to be optimized in order to increase tool life by decreasing tool wear.

Variables range

Speed $30 \leq v \leq 190$

Feed $0.01 \leq f \leq 2.5$

Depth of cut $0.5 \leq d \leq 4$

Genetic algorithm

The flow-chart of genetic algorithm is given in Fig. 1.

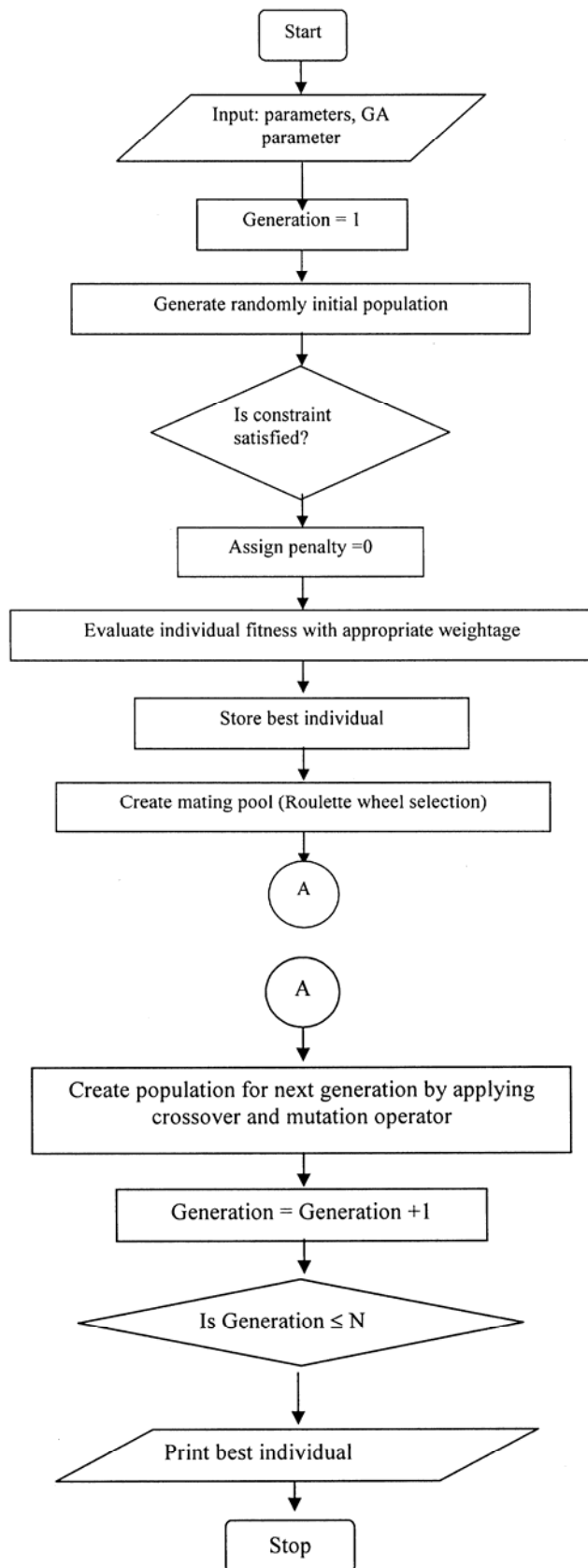


Fig. 1 — Flow-chart of genetic algorithm

Step 1: Choose a coding to represent parameters, operators. Select mutation data. Initialize the random population of string size as l_i . Select a maximum allowable generation number t_{max} .

Let $t = 0$.

Step 2: Evaluate every string

Step 3: If $t > t_{max}$ Terminate.

Step 4: Perform the reproduction, crossover and mutation on population, pairs of strings and every string respectively.

Step 5: Evaluate strings in next population.

Let $t = t + 1$. Go to **Step 3**

Results and Discussion

The effects of genetic algorithm on tool wear are shown in Table 1.

Table 1 — Effects of genetic algorithm on tool wear

Number of generation	Minimum objective	Variables (v,f,d)
0	0.328	40,1.99,0.606
1	0.318	40,0.749,0.606
2	0.326	34.06,0.168,0.783
3	0.3	34.222,2.008,0.606
4	0.271	33.754,0.032,0.582
5	0.283	32.971,0.723,0.606
6	0.263	32.971,0.302,0.568
7	0.269	33.284,0.302,0.582
8	0.255	30.469,0.302,0.582
9	0.254	30.156,0.302,0.552
10	0.254	30.156,0.302,0.552
11	0.267	32.659,0.302,0.586
12	0.254	31.72,0.302,0.558
13	0.264	37.664,0.139,0.527
14	0.265	32.346,2.473,0.524
15	0.265	32.346,2.473,0.524
16	0.256	31.095,0.915,0.541
17	0.263	37.351,0.139,0.527
18	0.245	30.782,1.451,0.5
19	0.245	30.782,1.453,0.5
20	0.245	30.782,1.453,0.5
21	0.246	32.033,0.302,0.527
22	0.257	31.408,1.558,0.527
23	0.254	30.938,1.414,0.527
24	0.25	33.441,0.149,0.538
25	0.255	30.938,1.334,0.531
26	0.245	34.848,0.168,0.5
27	0.318	40,0.749,0.606
28	0.269	33.284,0.302,0.582

Contd —

Table 1 — Effects of genetic algorithm on tool wear — *Contd*

Number of generation	Minimum objective	Variables (v, f, d)
29	0.326	34.06, 0.168, 0.783
30	0.328	40, 1.99, 0.606
31	0.249	30.782, 0.852, 0.527
32	0.248	30.782, 1.562, 0.507
33	0.244	31.564, 0.287, 0.527
34	0.244	30.313, 1.405, 0.503
35	0.246	30.313, 1.714, 0.507
36	0.246	30.312, 1.713, 0.506
37	0.247	30.312, 2.152, 0.503
38	0.247	30.312, 2.154, 0.503
39	0.254	30.938, 1.414, 0.527
40	0.263	37.351, 0.139, 0.527

The obtained results using genetic algorithm for the problem considered are given below:

The minimum tool wear $t_w = 0.244$ (mm)

The optimum set of parameters

$v = 31.5$ (m/min)

$f = 0.3$ (mm/rev)

$d = 0.5$ (mm)

are obtained at 33th generation. In this generation, 34 mutation and 271 crossover have been carried out. The following parameters are used in genetic algorithm

Population size	: 20
Total number of generation	: 40
Crossover probability	: 0.8
Mutation probability	: 0.05
String length	: 30

In the present work, an attempt has been made to develop a methodology for tool wear optimization using genetic algorithm with a single objective function of minimizing the output with satisfying the

constraints used. Here, we have obtained optimized parameters. If we apply optimized parameters then we will get minimum tool wear.

Conclusions

In this study, a mathematical model has been developed for the constrained optimization of cutting parameters that are used in turning operations. It is based on the "minimum tool wear" criterion. A non-traditional optimization algorithm, genetic algorithm has been employed to find the optimal cutting parameters such as cutting speed, feed and depth of cut⁵. In order to increase the accuracy and also to reduce time, a program is implemented on a PC by using C Language. The result obtained by genetic algorithm is better than the results of traditional techniques like dynamic programming.

Nomenclature

t_w = tool wear, mm
 f_c = cutting force, N
 t = temperature, °C
 v = cutting Speed, m/min
 f = feed, mm/rev
 d = depth of cut, mm

References

- 1 Ermer D S, *J Eng Ind*, 93 (1971) 1067.
- 2 Deb K, *Optimization for engineering design algorithms and examples*. (Prentice hall of India private ltd, Delhi), 199).
- 3 Gopalakrishnan.B & Faiz Al Khayyal, *Int J Prod Resources*, 30 (1972) 1897-1907.
- 4 Kalyanmoy Deb, *Optimization for Engineering Design*, June 2000.
- 5 Asokan P Saravanan R & Vijayakumar K, *Int J Adv Manuf Technol*, 2002.
- 6 Sabarinath P, Ramesh Babu R & Pitchayyapillai P, *Machining parameters optimization in milling process using, GA*, 6-11ETAM, 2005.
- 7 Rao S.S, *Optimization theory and application*, (New Age International Pvt. Ltd, New Delhi), 2002.
- 8 Sugavel S & Krishnaveni A, *Optimal Design of Worm Gear using GA*, , 12-17 ETAM, 2005..
- 9 Saravanan R, Ramakrishanan K, Surrender S, & Dwivedi A, *IE (I) J-MC*, July (2001), 82,74-79.