Available online at www.elixirpublishers.com (Elixir International Journal)

Electrical Engineering

Elixir Elec. Engg. 60C (2013) 16105-16114

ISSN: 2229-712X

Determination of optimum turning condition using genetic algorithm and analytic hierarchy process

Salar Fathi[1], Maghsud Solimanpur[2] and Mohammad Reza Azizi[3]

[1]University of Applied Science and Technology (Industrial Management Institute), Kermanshah Branch, Kermanshah, Iran.

[2]Faculty of Engineering, Urmia University, Urmia, Iran.

[3]Mechanic at the Zanjan University, Zanjan, Iran.

ARTICLE INFO

Article history:

Received: 2 January 2013;

Received in revised form: 17 June 2013;

Accepted: 27 June 2013;

Keywords

Turning parameters,

Machining optimization,

Genetic algorithms,

AHP.

ABSTRACT

Determination of optimum turning parameters in turning operations is one of the important tasks of process planners. The importance of this task is due to the fact that the values of turning parameters affect such objectives as turning time, turning cost and surface roughness. Thus, this problem is a multi-objective decision-making issue meaning that a parameter setting which satisfactorily achieves one of these objectives, it may not be good with respect to other objectives. Therefore, a rational compromise is to be made among these objectives. This paper determines the best cutting parameters considering relevant constraints in turning operations using a genetic algorithm coupled with an Analytic Hierarchy Process approach. The main stages of the proposed approach are two folds: (a) the non-dominated turning parameters being in the Pareto frontier are identified through a genetic algorithm, and (b) the AHP method is used to select the most suitable values for turning parameters from amongst the non-dominated solutions obtained in the first stage. Application of the proposed approach is demonstrated though an illustrative numerical example.

Introduction

Optimally determination of the turning parameters is very important to reduce turning time and cost and improve the quality of the machined surface. The basic turning parameters which affect the performance of machining operations are cutting speed (*v*), feed rate (*f*), and cutting depth (*d*). Optimization of these parameters leads to better utilization of tools and machines, improvement in the productivity of human resources and reduction in the final cost of finished goods.

Traditional studies in the field of optimization of machining operations are limited to single-pass operations (Ermer and Morris, 1969). The subsequent attempts in this field explored multi-pass operations to determine the optimal machining parameters (Shin and Joo, 1992). Classical methods used for optimizing machining parameters include geometric programming (Gopalakrishnan and Faiz, 1991), dynamic programming (Shin and Joo, 1992), linear programming (Gupta et al, 1995), and linear programming and branch-and-bound (Tan and Creese, 1995). To simplify the machining optimization problem, the earlier researches mainly do not consider all the cutting constraints. Thus, these techniques may only be useful for a specific problem and are likely to obtain a local optimal solution (Chen and Tsai, 1996). Consequently, approximation algorithms have been recently applied to solve various types of turning problems. Determination of optimal conditions for machining operations is a combinatorial optimization problem (Cus and Balic, 2003), for which local search and meta-heuristic methods are of potential advantage. Some newly emerged meta-heuristics for solving this problem are simulated annealing (SA), neural networks, genetic algorithms (GA), and tabu search (TS), and ant systems (AS) (Baskar et al. 2005). A neural network-based approach has been developed in (Zuperl and Cus, 2003) which take into account such issues as technological, economic and organizational limitations to optimize cutting parameters. Chen and Tsai (Chen and Tsai, 1996) combined pattern search (PS) technique and simulated annealing to solve the turning optimization problem. They used the pattern search technique to generate an initial solution. The simulated annealing technique is then applied to guide the solution process towards the global optimum solution. They divided the machining operations into roughing and finishing stages. Alberti and Perrone

Tele:

E-mail addresses: emamemostafa@gmail.com

(Alberti and Perrone, 1999) proposed a fuzzy possibilistic formulation of the classical multi-pass machining operations and optimized the resulting model using genetic algorithm.

Optimization of the turning conditions is inherently a multi-objective problem. Three main objectives considered in the earlier studies for optimization of turning parameters are unit production time, unit production cost, and surface roughness. Therefore, the problem of interest is to obtain a turning condition which minimizes these objectives. However, due to the multiple objectives, it may not be possible to determine the values of turning parameters in such a way that all the objectives mentioned above are minimized, simultaneously. In other words, there could be conflict among these objectives meaning that a turning condition with minimum unit production time may result in a higher roughness and vice versa. Therefore, the optimum solution in the optimization of turning parameters is not unique but a frontier. This frontier is called Pareto frontier in the literature. To define this frontier, the concept of *dominated solutions* is briefly discussed in the following.

For a multi-objective minimization problem with *n*-objectives, solution *u* is said to be dominated by a solution *v* if:

1. $f_i(u) \geq f_i(v) \quad \forall i = 1, 2, \ldots, n$
2. $f_j(u) > f_j(v) \quad \exists j = 1, 2, \ldots, n$

Those solutions that are not dominated by any other solution in the solution space make the Pareto frontier. Thus, in the case of the problem of concern with three objectives, namely unit production time, unit production cost, and surface roughness, we want to find the Pareto frontier in a three-dimensional space.

In this paper, a genetic algorithm (GA) is used to obtain the Pareto frontier. The basic feature of genetic algorithms is the multiple directional and global searches, in which a population of potential solutions is maintained from generation to generation (Solimanpur et al, 2004). This feature empowers the GA approach in leading the optimization process towards the Pareto frontier.

Due to the multiple objectives, a challenging issue in the optimization of turning parameters is that the optimization process finally yields to a frontier instead of a unique solution. On the other hand, importance of the objectives considered in the optimization of turning conditions is different from a work piece to one another. For example, in some work-pieces surface smoothness is of higher interest than the machining time. On contrary, for example, in mass production systems where production volume is large, it would be desirable to process work-pieces in shorter times. Therefore, a process planner has to select the most appropriate turning condition amongst the solutions being in the Pareto frontier. Our studies in the current literature indicate that available methods leave this task to the process planner. An Analytic Hierarchy Process (AHP) approach is developed in this paper to systematically help the process planner to select the solution which suits his/her interests at most. It is notable that the AHP approach is applied for optimization of turning condition for the first time in this paper.

This paper is organized as follows. The mathematical model of the attempted problem is presented in Section 2. In Section 3, the principles of methods VEGA and AHP are explored. Section 4 includes an illustrative example demonstrating the application of the proposed method. Conclusions and scope for future works are discussed in Section 5.

Mathematical model

Optimization of turning operations typically requires optimum determination of three parameters, namely cutting speed, feed rate and depth of cut. These parameters affect the time, cost and roughness of the turning operations. It is attempted in this paper to determine the optimal turning parameters to optimize unit turning time, unit turning cost, and surface roughness in overall. The mathematical formulation of the problem is presented in the following.

▪ ***Unit turning time**:* Usually, unit turning time, T_P , is considered as the entire time necessary for performing turning operations. It is a function of metal removal rate (*MRR*) and tool life *T* (Zuperl and Cus, 2003):

$$T_P = V(1 + T_C / T) / MRR, \qquad (1)$$

Where T_C, and *V* are the tool change time, and the volume of the metal to be removed, respectively. The parameters T_C and *V* are constant in the optimization of turning parameters, as are not affected by the turning parameters (Dereli et al, 2001). Hence, T_P is a function of *MRR* and *T*.

▪ ***Metal removal rate (MRR):*** MRR can be expressed as the product of the cutting speed, feed rate and cutting depth (Cus and Balic, 2003):

$$MRR=1000vfd. \quad (2)$$

Where *v*, *f*, and *d* denote the cutting speed, feed rate and cutting depth, respectively.

▪ ***Tool life (T)*****:** This parameter is measured as the average time between the tool changes. The relation between the tool life and the cutting parameters is expressed with the following formula (Zuperl and Cus, 2003):

$$T = K_T / v^{\alpha_1} f^{\alpha_2} d^{\alpha_3} \cdot \quad (3)$$

Where $K_T, \alpha_1, \alpha_2, \alpha_3$ are positive parameters determined statistically ((Cus and Balic, 2003), (Zuperl and Cus, 2003)).

▪ ***Unit turning cost:*** The unit turning cost can be expressed as follows (Cus and Balic, 2003), (Zuperl and Cus, 2003).

$$C_P = T_P\left(C_t / T + C_l + C_o\right) \cdot \quad (4)$$

Where C_t, C_l and C_o are the tool cost, the labor cost and the overhead cost, respectively. These parameters are independent of the cutting parameters.

▪ ***Cutting quality:*** The most important criterion for the assessment of surface quality is the roughness of finished surface calculated as follows ((Cus and Balic, 2003), (Zuperl and Cus, 2003), (Fang and Safi, 1997)):

$$Ra = x_1 \times v + x_2 \times f + x_3 \times d, \quad (5)$$

Where x_1, x_2 and x_3 are constants corresponding to the tool and work-piece combination.

▪ ***Constraints:*** There are several technical factors limiting the cutting parameters. These limitations can be expressed through the following inequalities:

$$v_{\min} \le v \le v_{\max} \quad ; \quad f_{\min} \le f \le f_{\max} \quad ; \quad d_{\min} \le d \le d_{\max} \quad (6)$$

▪ ***Cutting power and force*****:** The consumption of the power can be expressed as a function of the cutting force and cutting speed ((Shin and Joo, 1992), (Cus and Balic, 2003)):

$$P = \frac{F.v}{6122.45\eta}, \quad (7)$$

Where η is the mechanical efficiency of machine, since *F* and *v* are in terms of *Kgf* and *m*/*min*, the numeric value in the denominator of Equation (7) converts the dimension of *P* to *KW*. The cutting force *F* is given as follows ((Shin and Joo, 1992), (Cus and Balic, 2003)):

$$F = K_F f^{\beta_2} d^{\beta_3} \cdot \quad (8)$$

Where K_F, β_2, β_3 are constants related to cutting tool and work piece. The constraints on power and cutting force can now be expressed as

$$P(v, f, d) \le P_{\max} \qquad F(v, f, d) \le F_{\max} \cdot \quad (9)$$

Where $P_{\max}$ and $F_{\max}$ are the maximum power and cutting force of the machine tool respectively. As seen above, the three objectives T_p, C_p, and *Ra* are functions of cutting parameters *v*, *f* and *d*.

Proposed method

The method proposed in this paper solves the problem in two sequential stages. In the first stage, the solutions aligned in the Pareto frontier is obtained using Vector Evaluated Genetic Algorithm (VEGA). Of solutions identified in the Pareto frontier, the overall optimum solution is selected by AHP in the second stage. In sections 3.1 and 3.2, a general description is provided for genetic algorithms and VEGA, respectively. Section 3.3 presents the proposed genetic algorithm. The AHP approach used for selecting the final solution is then discussed in Section 3.4.

Genetic Algorithms

A Genetic Algorithm is an evolutionary optimization method used to solve, in theory "any" possible optimization problem. A GA (Man et. al., 1999) is based on the idea that a solution to a particular optimization problem can be viewed as an *individual* and that these individual characteristics can be coded into a finite set of parameters. These parameters are the *genes* or the *genetic information* that makes up the *chromosome* that represents the real world structure of the individual, which in this case is a solution to a particular optimization problem. Because the GA is an evolutionary method, this means that a repetitive loop or a series of *generations* are used in order to evolve a *population S* of *p* individuals to find the *fittest* individual to solve a particular problem. The *fitness* of each *individual* is determined by a given *fitness function* that evaluates the level of aptitude that a particular *individual* has to solve the given optimization problem. Each *generation* in the genetic search process produces a new set of individuals through *genetic operations* or *genetic operators: Crossover* and *Mutation,* operations that are governed by the *crossover rate* p_c and the *mutation rate* p_m, respectively. These operators produce new *child chromosomes* with the intention of improving the overall fitness of the population while maintaining a global search space. Individuals are selected for *genetic operations* using a *Selection method* that is intended to select the fittest individuals for the role of *parent chromosomes* in the *Crossover* and *Mutation* operations. Finally these newly generated child chromosomes are reinserted into the population using a *Replacement method.* This process is repeated a *k* number of *generations*.

In general, a genetic algorithm has five basic components, as summarized bellow (Gen and Cheng; 2000):

1. A genetic representation of solutions to the problem
2. A way to create an initial population of solutions
3. An evaluation function rating solutions in terms of their fitness
4. Genetic operators that alert the genetic composition of children during reproduction
5. Values for the parameters of genetic algorithm

In Step 3 mentioned above, mainly the objective function of problem is used to establish the fitness of each solution. However, this is applicable when the optimization problem has only one objective function. In the case of multiple objectives, a crucial issue in the evaluation of solutions in step 3 is the way through which the fitness of each solution is calculated. One of the approaches available in the literature for tackling this issue is the vector evaluated genetic algorithm (VEGA).

Vector evaluated genetic algorithm

As mentioned in Section 3.1, one of the crucial difficulties in the use of genetic algorithms for solving multi-objective problems is the calculation of fitness value for an individual. The difficulty is due to the fact that each solution has different degree of optimality with respect to each objective function. Schaffer (1985) proposed a vector evaluated genetic algorithm (VEGA) for treating multiple objectives. The genetic algorithm proposed in this paper uses the VEGA approach in the *selection* stage.

In the VEGA approach, the selection step in each generation becomes a loop. Each time an appropriate fraction of the next generation is selected on the basis of each objective. Specifically, suppose the population size in the genetic algorithm is *pop_size* and the number of objective functions is *K*. In VEGA, when selecting the solutions for the next generation, a number of *pop_size/K* solutions is selected with respect to each objective function.

Proposed genetic algorithm

- *Coding*

The optimization problem modeled in Section 2 has three objective functions (T_p, C_p and Ra) and three independent variables (v, f, d). These variables are represented with N_v, N_f and N_d genes in a binary mode. Hence, in the proposed algorithm the length of each chromosome representing a complete solution is $N= N_v + N_f + N_d$ (Figure 1). In this figure, N_v is the number of required genes for the representation of cutting speed in binary mode, N_f is the number of required genes for the representation of feed rate in binary mode, and N_d is the number of required genes for the representation of cutting depth in binary mode.

- *Calculation of N_v , N_f and N_d:*

We only discuss *calculation* of N_v, as N_f and N_d can be calculated in the same way. Let p denote the required precision (number of places after the decimal point) for parameter v. The parameter N_v equals to (Gen and Cheng, 2000):

$$N_v = [Log_2^L] + 1 , \qquad (10)$$

$$L = (v_{max} - v_{min}) \times 10^p . \qquad (11)$$

- *Fitness functions*

As mentioned in Section 3.2, the VEGA considers each objective function at a time and selects *Pop-size/K* solutions based on this function. Since there are three objectives in the formulated problem, number of fitness functions in the proposed genetic algorithm is three. Thus,

$$fit_1(S) = T_p, \quad fit_2(S) = C_p, \quad fit_3(S) = R_a, \qquad (12)$$

in which $fit_k(S)$ is the kth fitness value of solution S.

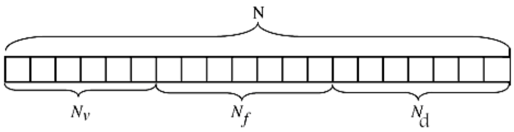

Figure 1: display the answer as a chromosome with N gene

- *Crossover*

The proposed genetic algorithm uses a simple crossover operator in which a random crossover point is determined and the second parts of the chromosomes are exchanged. The probability of selecting chromosome S for crossover is denoted by p_c(S) and calculated by equation (13).

$$p_c(S) = \frac{\max\ fit_k - fit_k(S)}{\sum_u \left(\max\ fit_k - fit_k(u)\right)} . \qquad (13)$$

Where max fit_k denotes maximum value of the fitness function k in the current population and u stands for the solutions in the current population.

- *Mutation*

Mutation brings unexpected features to the children that do not exist in parents. Each gene is chosen for mutation with a probability of p_m which is a GA parameter.

- *Selection*

The proposed algorithm uses the mechanism of VEGA to select solutions for the next generation. The probability of selection of solution S for the next generation with respect to the objective function k is:

$$p_k(S) = \frac{max\ fit_k - fit_k(S)}{\sum_u \left(max\ fit_k - fit_k(u)\right)} , \qquad (14)$$

in which *max* fit_k denotes maximum value of the fitness function k in the current population.

Decoding

After crossover and mutation operations, the values obtained for v, f, d may not be necessarily in the favorable ranges. Let v_{min} and v_{max} denote the minimum and maximum cutting speed, respectively. Let v denote the cutting speed obtained after genetic operations. The following equation is used in this paper to make the obtained value feasible.

$$v = v_{min} + (v_{max} - v_{min}) \cdot \frac{v}{2^{N_v} - 1} . \qquad (15)$$

Similar equations can be derived for variables *f* and *d*. This transformation guarantees that the solutions obtained by the genetic algorithm will satisfy the constraints (6).

Analytic Hierarchy Process (AHP)

AHP was first introduced in the middle of the 1970s by T.L. Saaty. AHP divides a sophisticated problem into elements, groups them in the light of a dominating relationship to form an orderly hierarchical structure, determines relative importance among elements in the structure through paired comparison, and then synthesizes human judgments to give a total order of decision elements (Liu et al., 1999). Generally, three steps are done in AHP.

1. Establish the hierarchical decision structure of the problem. This hierarchy breaks the total goal down to sublevels like criterion level, sub criterion level and scheme level.

2. Construct pairwise comparison judgment matrices to decide subordinate relationships between upper levels and lower levels and the relative importance among elements at the same level. Judgment matrices are critical to hierarchical analysis. Suppose the following factors $B1$, $B2$, . . . , B_n influence the general goal Z. By pair wise comparison, a_{ij} is used to demonstrate the influential ratio of B_i vs. B_j. The judgment matrix is expressed as follows:

$$A = (a_{ij}), \quad a_{ij} > 0$$

$$a_{ij} = 1, \quad i=j \tag{16}$$

$$a_{ji} = 1/a_{ij}, \quad i, j = 1, 2, \ldots, n \ (i \neq j)$$

The following five grades are usually used to set the value of a_{ij} (Liu, et al, 1999).

B_i *vs.* B_j	*Value of* a_{ij}
Equivalent	1
A little strong	3
Strong	5
Very strong	7
Absolutely strong	9

3. Obtain weight of each scheme with respect to the main goal. This is computed by the inner product of the weights of criteria in the weights of schemes. The detailed description of the computational process in AHP is shown through an illustrative example in Section 4. The scheme with a higher weight is finally selected as the optimum solution.

Application of the method discussed above for the case of optimization of turning parameters will be demonstrated in Section 4 through a numerical example.

Illustrative example

An example has been adopted from (Cus and Balic, 2003) to show the application of the proposed method. In this problem, cast steel has to be machined on a CNC lathe through a tool made from HSS. Values of different factors are as follows.

$$T_c = 0.26 \min$$

$$C_t = 13.55\$, \ C_l = 0.31\$/min$$

$$C_o = 0.08\$/min$$

K=1.001, K_T = 1575134.21, K_F =1.38

x_1 = 0.0088, x_2 = 0.3232, x_3 = 0.3144

$$\alpha_1 = 1.7, \ \alpha_2 = 1.55, \ \alpha_3 = 1.22,$$

$$\beta_1 = 0, \ \beta_2 = 1.18, \ \beta_3 = 1.26$$

$$V = 231376mm^3, \ \eta = 36\%$$

$v_{min} = 70\, m/\min$, $v_{max} = 90\, m/\min$ $f_{min} = 0.1 mm/rev$, $f_{max} = 2 mm/rev$

$d_{min} = 0.1mm$, $d_{max} = 5mm$, $F_{max} = 230N$, $P_{max} = 5KW$

By substituting these values, the mathematical model is summarized as follows:

$$MinT_P = 231376(1+0.26/T)/MRR$$

$$MinC_P = (13.55/T + 0.39)T_P$$

$$MinRa = 0.0088v + 0.3232f + 0.3144d$$

Subject to:

$$T = 1575134.21(v^{-1.7} f^{-1.55} d^{-1.22})$$

MRR=1000vfd

$$70 < v < 90$$

$$0.1 < f < 2$$

$$0.1 < d < 5$$

$$0.000626(vf^{1.18}d^{1.26}) \le 5$$

$$1.38(f^{1.18}d^{1.26}) \le 230$$

Based on the computational experiences, the values of parameters *pop_size* and p_m are considered as 105 and 0.005, respectively.

The cutting conditions are generated at random inside the specified limits in the first generation. Application of the proposed genetic algorithm resulted in 17 non-dominated solutions along the Pareto frontier as shown in Table 1. The AHP method is used in the following to determine the optimum solution among the 17 solutions reported in Table 1. Since the problem of interest in this paper is the determination of turning parameters, each non-dominated solution in Table 1 is called a setting.

Table 1: non-dominated solutions obtained by the proposed genetic algorithm

Setting	V	F	D	T	C	R
1	79.6774	2.000	1.5000	0.9693	0.4464	1.8192
2	82.9032	0.7333	0.8000	4.7579	1.8908	1.2181
3	88.7079	2.000	3.6000	0.3640	0.2316	2.5589
4	72.5806	1.3667	1.5000	1.5560	0.6588	1.5520
5	72.5806	2.000	3.6000	0.4442	0.2510	2.4169
6	75.1613	1.3667	1.5000	1.0250	0.4576	1.7948
7	88.0645	0.7333	3.6000	2.3893	0.9740	1.4836
8	86.1290	2.000	2.2000	0.6120	0.3173	2.0960
9	74.5161	1.3667	1.5000	1.5157	0.6440	1.5690
10	70.0000	2.0000	3.6000	0.4605	0.2554	2.3942
11	77.0968	1.3667	2.2000	0.9993	0.4486	1.8118
12	73.2258	1.3667	2.9000	0.7984	0.3718	1.9979
13	71.2903	1.3667	2.2000	1.0805	0.4772	1.7607
14	85.4893	1.3667	0.8000	2.4766	1.0165	1.4455
15	71.2903	2.000	3.6000	0.4522	0.2532	2.4056
16	74.5161	1.3667	2.2000	1.0338	0.4607	1.7891
17	89.3548	0.7333	0.8000	4.4145	1.7587	1.2745

With respect to the stepwise procedure of the AHP method described in Section 3.4, this method is applied for the illustrative example as follows.

(1) Figure 2 shows the decision hierarchy of the problem in which S_1, S_2, …, S_{17} denote the non-dominated settings shown in Table 1.

(2) The weight (importance) of each setting with respect to another setting is calculated considering each objective function. These weights are computed using the equations (17). Therefore, three matrices, namely *A*, *B* and *C* associated with objectives T_p, C_p and *Ra* are calculated. The entry a_{ij} indicates the relative importance of setting *i* over setting *j* with respect to T_p. Similarly, the entry b_{ij} and c_{ij} in matrices *B* and *C* indicate the importance of setting *i* over setting *j* with respect to objectives C_p and *Ra*, respectively. In the illustrative example, the size of matrices *A*, *B* and *C* is 17×17.

$$a_{ij} = \frac{T_{p\,j}}{T_{p\,i}} \qquad b_{ij} = \frac{C_{p\,j}}{C_{p\,i}} \qquad c_{ij} = \frac{Ra_j}{Ra_i} \,. \tag{17}$$

Then these matrices should be normalized. To do so, each entry is divided into the summation of the entries being in the corresponding column. Mathematically, the following formulas are used to normalize the entries a_{ij}, b_{ij} and c_{ij} in matrices *A*, *B* and *C*, respectively. Let A', B' and C' denote the normalized matrices of *A*, *B* and *C*, respectively.

$$a'_{ij} = \frac{a_{ij}}{\sum_{j=1}^{17} a_{ij}} \qquad b'_{ij} = \frac{b_{ij}}{\sum_{j=1}^{17} b_{ij}} \qquad c'_{ij} = \frac{c_{ij}}{\sum_{j=1}^{17} c_{ij}} \,. \tag{18}$$

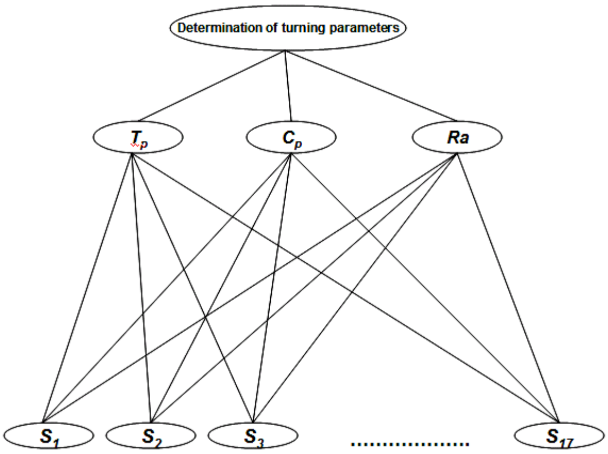

Figure 2: Decision hierarchy of the illustrative example

Due to the consistency of matrix *A*, all the columns of matrix A' will be the same. Figure 4 shows one of the columns of matrix A' for instance. This would be the case in matrices B' and C' as well. The columns of matrices A', B' and C' are put in a 17×3 matrix denoted by *D*. In other words, the first, second and third columns of matrix *D* are the column of matrices A', B' and C', respectively. Consequently, the values in the first row of matrix *D* indicate the weight of setting 1 with respect to objectives T_p, C_p and *Ra*, respectively.

$$D = \begin{bmatrix} 0.0523 & 0.0573 & 0.0569 \\ 0.0107 & 0.0135 & 0.0850 \\ 0.1393 & 0.1104 & 0.0405 \\ 0.0326 & 0.0388 & 0.0667 \\ 0.1141 & 0.1019 & 0.0429 \\ 0.0495 & 0.0559 & 0.0577 \\ 0.0212 & 0.0263 & 0.0698 \\ 0.0828 & 0.0806 & 0.0494 \\ 0.0334 & 0.0397 & 0.0660 \\ 0.1101 & 0.1001 & 0.0433 \\ 0.0507 & 0.0570 & 0.0572 \\ 0.0635 & 0.0688 & 0.0518 \\ 0.0469 & 0.0536 & 0.0588 \\ 0.0205 & 0.0252 & 0.0717 \\ 0.1121 & 0.1010 & 0.0431 \\ 0.0490 & 0.0555 & 0.0579 \\ 0.0115 & 0.0145 & 0.0813 \end{bmatrix}$$

A pairwise judgment is to be made among the objectives T_p, C_p and *Ra* to capture the relative importance of each objective with respect to the main goal. Here, the judgment of one process planner may be different to that of one another. Even pairwise importance of objectives depends upon the work-piece to be machined. This means that a process planner may judge the importance of one objective function over another objective different from one work-piece to another.

A typical pairwise matrix indicating the importance of objectives for the illustrative example is shown in Table 2. For example, the number 3 in the cell (T,C) indicates that the turning time is three times important than turning cost for in the illustrative example.

Table 2. Pairwise comparison matrix for T, C and R

	T	C	R
T	1	3	2
C	1/3	1	2/3
R	0.5	1.5	1

A similar normalization procedure is to be done for the matrix shown in Table 2. As mentioned so far, all the columns of the normalized matrix will be the same as shown by vector *E*.

$$E = \begin{bmatrix} 0.5455 \\ 0.1818 \\ 0.2727 \end{bmatrix}$$

For example, the number 0.5455 in vector *E* indicates the weight of T_p with respect to the main goal.

(3) To obtain the weight of each setting with respect to main goal, we compute vector *F* as the inner product of matrix *D* and vector *E* as follows.

$$F = D \cdot E = \begin{bmatrix} 0.0545 \\ 0.0315 \\ 0.1071 \\ 0.0430 \\ 0.0924 \\ 0.0529 \\ 0.0354 \\ 0.0733 \\ 0.0435 \\ 0.0900 \\ 0.0536 \\ 0.0613 \\ 0.0514 \\ 0.0353 \\ 0.0912 \\ 0.0526 \\ 0.0311 \end{bmatrix}$$

Each value in the row *i* of vector *F* indicates the total weight of setting *i*. According to the weights computed in vector *F*, setting 3 with *v*=88.7079 *m*/*min*, *f*=2.000 *mm*/*rev* and *d*=3.6000 *mm* has the highest weight and thus serves the objectives of interest at most. The corresponding turning time, cost and roughness for this setting is $T_p=0.3640\ min$, $C_p=0.2316\ \$$ and $Ra=2.5589\ \mu m$.

Conclusion and discussion

Advantages of the proposed method can be outlined as follows.

(1) The proposed method explores the solution space and finds non-dominated settings being in the Pareto frontier.

(2) Although the AHP approach is a well known multi-criteria decision-making technique, it is applied to the determination of cutting parameters for the first time in this paper. The AHP method helps a process planner to systematically select a solution from Pareto frontier which fits to his/her interests at most.

(3) Although the AHP method discussed in this paper considers three objectives to select the optimum cutting condition, it is still open to take other criteria into account. If a process planner intends to consider other criteria in addition to T_p, C_p and *Ra* in determining turning conditions, he/she can simply add these criteria to the first level of the hierarchy shown in Figure 2.

One of the limitations of the proposed AHP is that the pairwise comparison between T_p, C_p and *Ra* are done in a crisp way. However, the relative importance of an objective over one another can be better captured by fuzzy logic. This logic provides process planner with a flexible tool to judge the relative importance of objectives. This issue can be attempted in subsequent researches.

References

Chen M.C., Tsai D.M., (1996) "A simulated annealing approach for optimization of multi-pass turning operations", *International Journal of Production Research,* Vol. 34(10), pp. 2803-2825.

Cus F., Balic J., (2003) "Optimization of cutting process by GA approach", *Robotics and Computer Integrated Manufacturing,* Vol. 19, pp. 113-121.

Liu D., Duan G., Lei N., and Wang J.S., (1999) "Analytic Hierarchy Process Based Decision Modeling in CAPP Development Tools", *International journal of advanced manufacturing technology,* Vol. 15, pp. 26–31.

Dereli T., Filiz I.H., Baykasoglu A., (2001) "Optimizing cutting parameters in process planning of prismatic parts by using genetic algorithms". *International Journal of Production Research,* Vol. 39(15), pp. 3303-3328.

Ermer D.S., and Morris S.M.., (1969) "A treatment of error of estimation in determining optimum machining conditions", *International Journal of Machine Tool Design and Research,* Vol. 9, pp.357-362.

Fang X.D., Safi J.H., (1997), "A new algorithm for developing a reference-based model for predicting surface roughness in finish machining of steels*". International Journal of Production Research,* Vol. 35(1), pp. 179-199.

Gen M., Cheng R., (2000) "Genetic Algorithms and Engineering Optimization", New York: *John Wiley*.

Gopalakrishnan B., Faiz A.K., (1991) "Machining parameter selection for turning with constraints: an analytical approached based on geometric programming", *International Journal of Production Research*, Vol. 29(9), pp.1897-1908.

Gupta R, Batra JL, Lal GK, (1995), "Determination of optimal subdivision of depth of cut in multi-pass turning with constraints", *International Journal of Production Research*, Vol. 33, pp. 2555-2565.

Man, K. F., Tang, K. S. and Kwong, S. (1999) "Genetic Algorithms", *Ed. Springer,* 1st Edition, London, UK.

Nathasit Gerdsri, Dundar F. Kocaoglu, (2007) " Applying the Analytic Hierarchy Process (AHP) to build a strategic framework for technology road mapping", *Mathematical and Computer Modeling*, Vol. 46, pp. 1071–1080

Baskar, N., Asokan, P., Saravanan, R., and Prabhaharan G., (2005) "Optimization of Machining Parameters for Milling Operations Using Non-conventional Methods", *International Journal of advanced manufacturing technology*, Vol. 25, pp. 1078–1088.

Castillo O., and Trujillo L., (2005) "MULTIPLE OBJECTIVE OPTIMIZATION GENETIC ALGORITHMS FOR PATH PLANNING IN AUTONOMOUS MOBILE ROBOTS", *International Journal of Computers, Systems and Signals,* Vol. 6, No. 1., pp. 48-63.

Shankar Chakraborty and Sammilan Dey, (2006) "Design of an analytic-hierarchy-process-based expert system for non-traditional machining process selection", *International Journal of advanced manufacturing technology,* Vol. 31, pp. 490–500.

Shin Y.C., and Joo Y.S., (1992) "Optimization of machining conditions with practical constraints", *International Journal of Production Research,* Vol. 30(12), pp. 2907-2919.

Solimanpur, M., Vrat, P., and Shankar, R., (2004) "A multi-objective genetic algorithm approach to the design of cellular manufacturing systems", *International journal of production research,* Vol. 42, No. 7, 1419–1441

Tan F.P., Creese R.C., (*1995*) "A generalized multi-pass machining model for machining parameter selection in turning". *International Journal of Production Research,* Vol. 33, pp. 1467-1487.

Thomas L. Saaty, Luis G. Vargas, and Klaus Dellmann, (2003), "The allocation of intangible resources: the analytic hierarchy process and linear programming" *Socio-Economic Planning Sciences,* Vol. 37, pp. 169–184.

Zuperl U., Cus F., (2003) "Optimization of cutting conditions during cutting by using neural networks", *Robotics and Computer Integrated Manufacturing,* Vol. 19, pp.189-199.