Proceedings in Manufacturing Systems, Volume 8, Issue 2, 2013 ISSN 2067-9238

SWARM INTELLIGENCE COMBINED WITH NEURAL NETWORK OBJECTIVE FUNCTION MODELLING FOR TURNING PROCESS OPTIMIZATION

Uros ZUPERL

[1)] Assist. Prof., researcher, University of Maribor, Faculty of mechanical engineering, Maribor, Slovenia

Abstract: *This paper discusses the use of non-traditional optimization technique based on combination of artificial neural network and swarm intelligence for optimization of cutting parameters in turning.*
An artificial neural network model (ANN) was used to predict objective function during optimization and particle swarm optimization algorithm (PSO) was used to obtain optimum cutting speed and feed rate. This paper also presents fundamentals of ANN-PSO optimization technique.
The study also incorporates the manufacturer's value system into a combined neuro–swarm system to optimize the cutting parameters. An objective function based on manufacturer's multi-attribute function is used. The optimization process considers the practical constraints, such as maximum machine power, force, allowable speed, feedrate and surface requirement. The objective is to find the best parameter settings to maximize the production rate and surface quality and to minimize the production costs.
The results indicate that the proposed optimization system is efficient and accurate compared to other methods developed by other researchers. This paper compares the results of proposed optimization system with the GA, ACO and simulated annealing (SA). The optimization system should be used for the fast approximate determination of optimum cutting conditions on the machine, when there is not enough time for deep analysis.

Key words: *machining, optimization, swarm intelligence, neural network.*

1. INTRODUCTION

The selection of optimal cutting parameters is a very important task for every machining process. In workshop practice, cutting parameters are selected from handbook recommendations, machining databases or experience. The cutting conditions set by such practices are actually starting values and far from optimal values.

Optimization of machining parameters is complicated where the following knowledge is required: knowledge of machining, numerical optimization techniques, machine tool capabilities and knowledge of effective optimization criterion.

For the optimization of a machining process, either the minimum production time or the maximum profit rate is used as the objective function subject to the practical constraints.

Many optimization algorithms have been introduced in solving machining optimization problems. These algorithms are known as traditional-classical algorithms.

Traditional optimization algorithms are not efficient for multiobjective optimization problems, because they cannot find multiple solutions in a single run. They are not ideal for solving these problems as they tend to obtain a local optimal solution. These methods are also not robust.

[*] Corresponding author: Faculty of Mechanical Engineering, Smetanova ul. 17, 2000 Maribor, Slovenia
Tel.: 386 2 220 7500
Fax: 386 2 220 7990
E-mail addresses: *uros.zuperl@um.si*

To eliminate this difficulty of classical methods evolutionary algorithms have emerged and demonstrated that these methods can be efficient in robust.

Genetic algorithms (GA) [1], simulated annealing (SA) [2] and ant colony optimization (ACO) [3] are some of the non-traditional algorithms used for solving optimization problems in machining.

In this paper, a multi-objective optimization method, based on a combination of artificial neural network (ANN) and particle swarm optimization (PSO), is proposed to find the optimal parameters in turning processes.

2. CUTTING PROCESS MODEL

The objective of this optimization turning model is to determine the optimal machining parameters including cutting speed, feed rate and depth of cut in order to minimize the operation cost (C_p) and to maximize production rate (represented by manufacturing time (T_p) and cutting quality (R_a).

$$C_p = T_p \cdot \left(C_t / T + C_l + \cdot C_0 \right), \quad (1)$$

where C_t, C_1 and C_0 are the tool cost, the labour cost and the overhead cost respectively; T is tool life.

The objectives used in this work are determined according to [2]. In order to ensure the evaluation of mutual influences and the effects between the objectives and to be able to obtain an overall survey of the manufacturer's value system the multi-attribute function of the manufacturer (y) is determined.

The cutting parameter optimization problem is formulated as the following multi-objective optimization problem: min T_p (v, f, a), min C_p (v, f, a), min R_a (v, f, a).

A multiattribute value function (y) is defined as a real-valued function that assigns a real value to each multiattribute alternative, such that more preferable alternative is associated with a larger value index than less preferable alternative.

The following limitations are taken into account: Permissible range of cutting conditions: $v_{\min} \le v \le v_{\max}$, $f_{\min} \le f \le f_{\max}$, $a_{\min} \le a \le a_{\max}$;

Implied limitations issuing from the tool characteristics and the machine capacity;

The limitations of the power and cutting force are equal to: $P(v, f, a) \le P_{\max}$, $F(v, f, a) \le F_{\max.}$

3. ANN-PSO BASED MULTI-OBJECTIVE OPTIMIZATION OF CUTTING PARAMETERS IN TURNING

The proposed optimization routine consists of two main steps.

First, experimental data are prepared to train and test artificial neural network (ANN) to represent the objective function (y).

Then, a PSO algorithm is utilized to obtain the optimal objective value.

Fig. 1 shows the flowchart of the proposed approach. In this approach the swarm flays over the objective function surface (y) and searches for the extreme of this function. The coordinate of the particle which is the nearest to mentioned extreme, represent the optimal cutting conditions.

Required steps for optimization of cutting parameters by proposed approach:

1. Generation of initial swarm population. An array of 50 particles with random positions and velocities are generated. Velocity vector has 3 dimensions, cutting speed, feed rate and cutting depth. This constitutes Generation 0.
2. Calculation of other values (P; F; C_p; T; R_a; T_p).
3. Training and testing of ANN.
4. Use of ANN model: The purpose of ANN is to predict the manufacturer's value function (y) for initial swarm population.
5. Optimization process: Evaluation of objective function for each particle. The cutting conditions (coordinate of the particle) where the function (y) has the maximum are the optimum cutting conditions. PSO is searching for the extreme of the function (y). Since

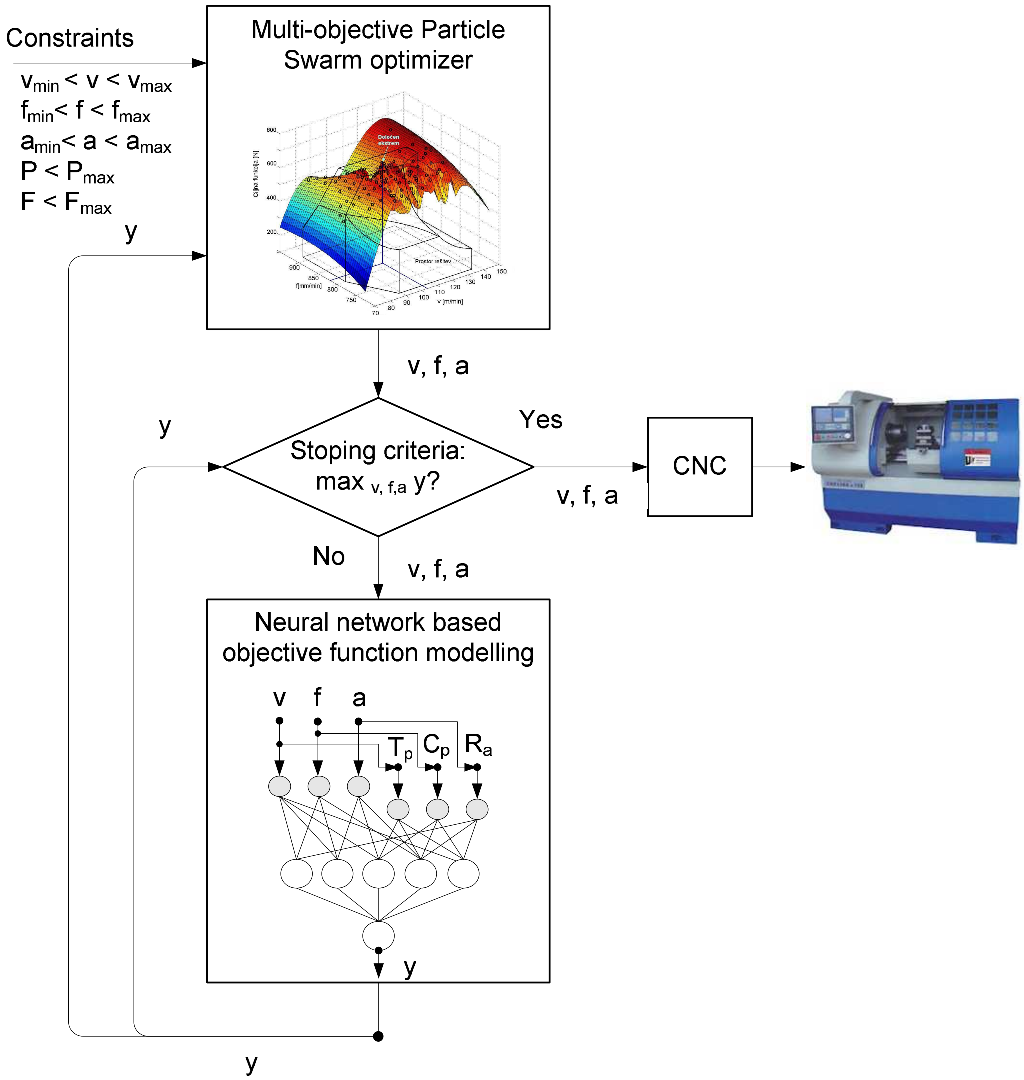

Fig. 1. The proposed ANN-PSO optimization system.

the function (y) is expressed with ANN, it means that the extreme of ANN is searched for. The objective function surface is limited with planes which represent the constraints of cutting process. Five constraints, which arise from technological specifications, are considered during the optimization process.

6. Survey of optimum cutting conditions and the variables relevant to them.

4. MANUFACTURER'S IMPLICIT MULTIATTRIBUTE FUNCTION MODELING

First step uses artificial multylayer feedforward neural network (ANN) to model the response (manufacturer's implicit multiattribute) function (y). The variables of this problem are velocity, feed rate and depth of cut, which can have any continuous value subject to the limits available.

The ANN system needs three inputs for three parameters: cutting speed (v), feedrate (f) and depth of cutting (a). Then T_p, C_p and R_a are calculated based on v, f and a.

The output from the system is a real value (y), therefore only one output neuron is necessary.

During training of ANN, 120 sets of experimental data were used to conduct 250 cycles of training. Network training involves the process of interactively adjusting the interconnection weights in such a way that the prediction errors on the training set are minimized.

The back- propagation algorithm is applied to each pattern set, input and target, for all pattern sets in the training set. Since the learning process is iterative, the entire training set will have to be presented to the network over and over again, until the global error reaches a minimum acceptable value.

An additional 80 examples were used to test the trained network.

ANN has proved to be an excellent universal approximator of non-linear functions. It is capable to represent the manufacturer's implicit multiattribute function.

After training the model, its performance was tested under various cutting conditions. Test data sets collected from a wide range of cutting conditions in turning were applied to the estimator for evaluating objective function (y). The performance of this method turned out to be satisfactory for estimating of objective function (y), within a 2% mean percentage error.

Once a multi-attribute value function is assessed and validated the ANN is used to decipher the manufacturer's overall preference and the multi-objective optimization problem is reduced to a single objective maximization problem as follows:

$$\max_{v,f,a} y\left[T_p(v,f,a), C_p(v,f,a), R_a(v,f,a)\right]. \quad (2)$$

5. PARTICLE SWARM OPTIMIZER

Particle swarm optimization is a non-traditional optimization technique which is based on swarm intelligence.

Researches [4, 5] developed swarm models with simple rules and generated complicated swarm behavior. These models imitate graceful but unpredictable movement of a bird swarm. They are called "Swarm Intelligence".

Special swarms like birds, fishes, and bees that live in a big colony are capable of solving their daily complex life problems. These behaviours which are seen in a special group of animals are called swarm intelligence.

Swarm intelligence techniques focus on the group's behaviour and study the decartelized reactions of group agents with each other and with the environment.

The swarm intelligence system includes a mixture of simple local behaviours for creating a complicated general behaviour and there is no central control in it.

Swarm behaviour of birds inspired the new computational paradigm for optimizing real life systems and it is suited for solving large scale optimization problem.

Swarm behaviour can be modelled with a few simple rules. Even if the behaviour rules of each individual (particle) are simple, the behaviour of the swarm can be very complex.

The behaviour of each agent inside the swarm can be modelled with simple vectors. This characteristic is the basic concept of PSO.

The first PSO algorithm has been applied to the travelling salesman problem (TSP) [5], proposed an ant PSO methodology for milling parameters optimization.

6. PSO ALGORITHM

The general flow chart of PSO strategy for optimization of cutting parameters in multi-pass turning is shown in Fig. 2.

PSO is developed through simulation of bird flocking on objective function represented by ANN.

The position of each agent is represented by *XYZ* axis position and also the velocity is expressed by *vx* (the velocity of X axis), *vz* (the velocity of Z axis) and *vy* (the velocity of Y axis).

Modification of the agent position is realized by the position and velocity information. With a search, the N birds select N new regions and move in search of better fitness. The variables of this problem are cutting speed, feedrate, depth of cut, all of which can have any continuous value subject to the limits imposed. Bird flocking optimizes an objective function.

The objective functions are calculated for each solution. New solutions will be obtained after the global search.

The solutions will also have the new position values. The solutions are sorted in ascending order of the objective values and the best objective value is stored.

The process is repeated for a specified number of iterations.

Each agent knows its best value so far (*pbest*) and its *XYZ* position. Further, each agent knows the best value so far in the group (*gbest*) among (*pbests*). This information is analogy of knowledge of how the other agents around them have performed.

Each agent tries to modify its position using the following information: the current positions (x, y, z), current velocities (*vx*, *vy*, *vz*), distance between the current position and (*pbest*); distance between the current position and (*gbest*).

Performing a PSO, birds are repeatedly sent to trail solutions in order to optimize the objective value.

This modification can be represented by the concept of velocity. Velocity of each agent can be modified by the following equation:

$$max_{v,f,a}\ y\left[T_p(v,f,a), C_p(v,f,a), R_a(v,f,a)\right], \qquad (3)$$

where, v_i^k – velocity of agent i at iteration k, w – weighting function, c_j – weighting factor, *rand* – random number between 0 and 1, s_i^k – current position of agent i at iteration k, $pbest_i$ – *pbest* of agent i, *gbest* – *gbest* of the group.

The following weighting function is usually utilized (1):

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \times iter\ , \qquad (4)$$

where, w_{max} – initial weight, w_{min} – final weight, $iter_{max}$ – maximum iteration number, *iter* : current iteration number.

Using the above equation, a velocity, which gradually gets close to *pbest* and *gbest* can be calculated. The current position (searching point in the solution space) can be modified by the following equation:

$$s_i^{k+1} = s_i^k + v_i^{k+1}. \qquad (5)$$

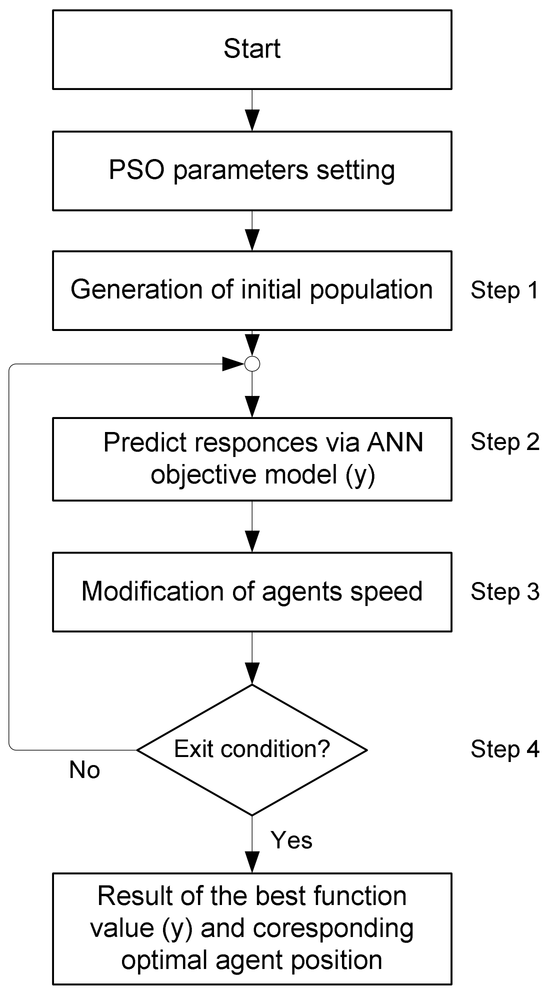

Fig. 2. Flowchart of the PSO algorithm.

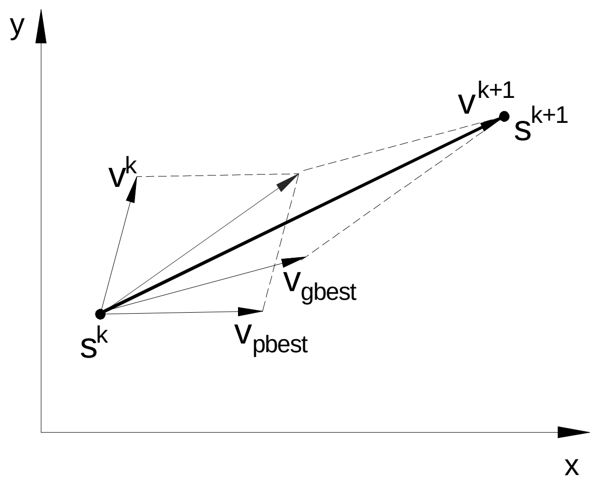

Fig. 3. Flowchart of the PSO algorithm.

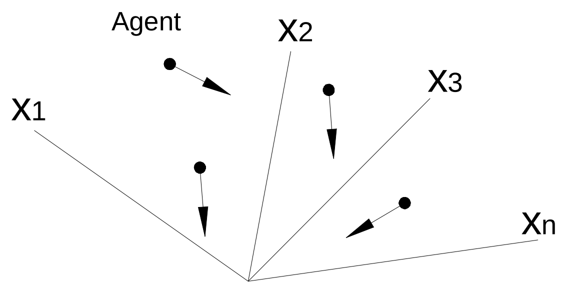

Fig. 4. Flowchart of the PSO algorithm.

Fig. 3 shows a concept of modification of a searching point by PSO algorithm.

Fig. 4 shows a searching concept with agents in a solution space. Each agent changes its current position using the integration of vectors as shown in Fig. 3.

The general flow chart of PSO method can be described as follows:

Step 1: Generation of initial condition of each agent Initial searching points (s_i^0) and velocities (v_i^0) of each agent are generated randomly within the allowable range. Initial solution of N will consist of 50 randomly generated solutions, with values that lie in the range of allowable cutting speed, depth of cut and feedrate. The current searching point is set to *pbest* for each agent. The best-evaluated value of *pbest* is set to *gbest* and the agent number with the best value is stored.

Step 2: Evaluation of searching point of each agent. The objective function value is calculated for each agent. If the fitness is improved, the new solutions are updated to the current location. Correspondingly the location position vector is updated. The positions pertaining to minimum production cost are referred to as superior solutions, while positions pertaining to the maximum production cost are referred to as inferior solutions. If the value is better than the current *pbest* of the agent, the *pbest* value is replaced by the current value. If the best value of *pbest* is better than the current *gbest*, *gbest* is replaced by the best value and the agent number with the best value is stored.

Step 3: Modification of each searching point The current searching point of each agent is changed using (1), (2) and (3).

Step 4: Checking the exit condition The current iteration number reaches the predetermined maximum iteration number, then exit. Otherwise, go to step 2.

Fig. 5 shows the PSO flowchart of optimization of turning process.

The optimization process of turning is depicted by the following steps:

1. Generation and initialization of an array of 50 particles with random positions and velocities. Velocity vector has 2 dimensions, feed rate and spindle speed. This constitutes Generation 0.
2. Evaluation of objective (cutting force surface) function for each particle.
3. The cutting force values are calculated for new positions of each particle. If a better position is achieved by particle, the *pbest* value is replaced by the current value.
4. Determination if the particle has found the maximal force in the population. If the new *gbest* value is better than previous *gbest* value, the *gbest* value is replaced by the current *gbest* value and stored. The result of optimization is vector *gbest* (feedrate, spindle speed).
5. Computation of particles' new velocity.
6. Update particle's position by moving towards maximal cutting force.
7. Steps 2 to 6 are repeated until the iteration number reaches a predetermined iteration.

7. RESULTS AND DISCUSSION

The PSO optimization method combined with ANN prediction system was tested on the CNC lathe GF02. The work piece material is mild steel (Ck45) and the tool material has a carbide tip.

The task is to find optimum cutting conditions for the process of turning with minimal costs.

Proposed ANN-PSO approach was compared with three non-traditional techniques (GA, SA and ACO.

The task is to find optimum cutting conditions for the process of turning with minimal costs.

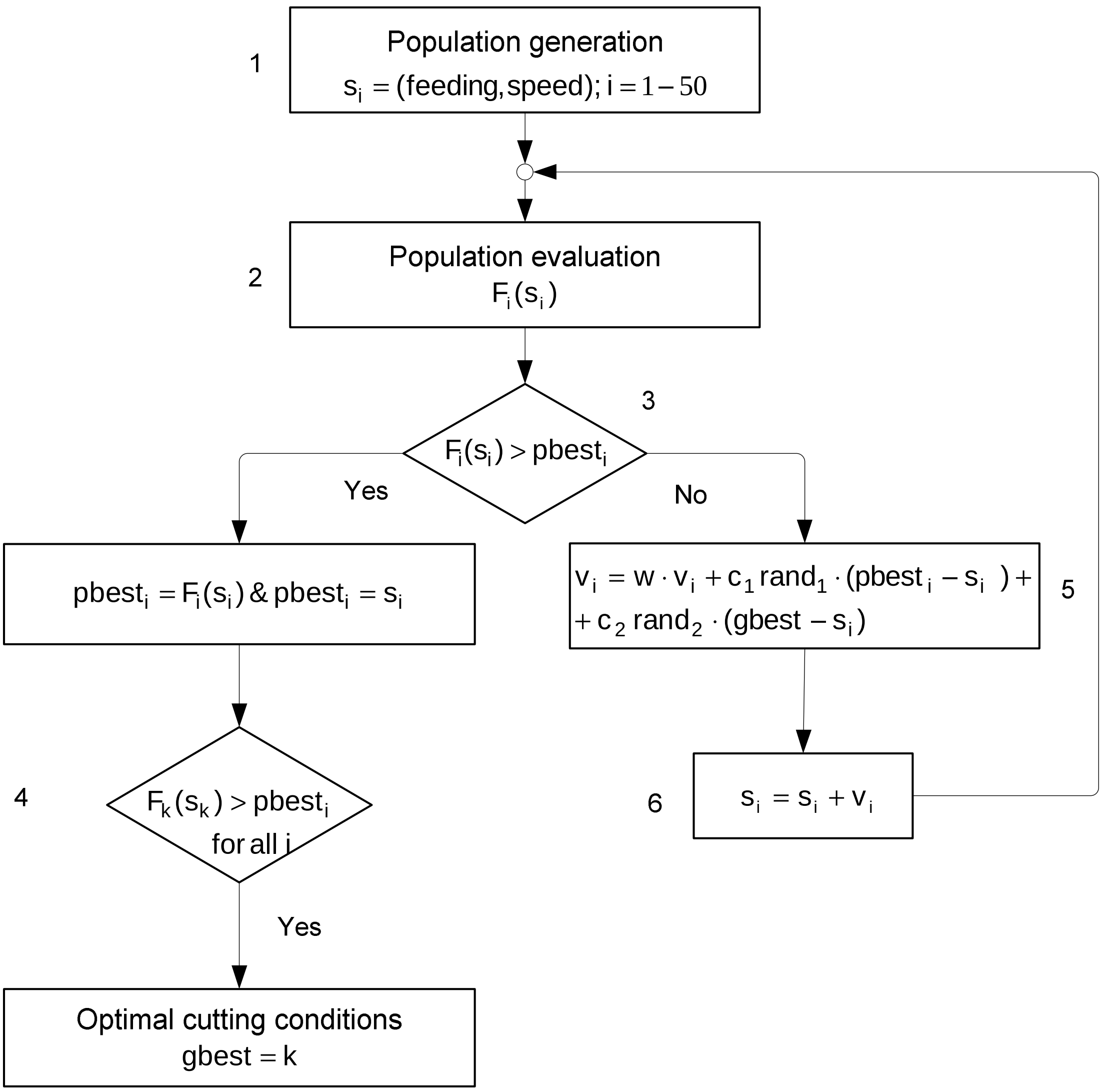

Fig. 5. PSO algorithm for optimization of turning parameters.

Table 1

Comparison of results for ANN-PSO, GA, LP and ACO approach

No.	Algorithm	Constraint set	Runs	Optimum solution v_{opt} [m/min]	f_{opt} [mm/rev]	a_{opt} [mm]	C_p [$]	Average optimiz. time [s]
1	Proposed ANN-PSO	tool-life; cutting force- power; surface roughness;	1 - 25 1 - 150	95.1926 98.132	0.3793 0.2883	0.84 0.91	12.423 **12.213**	2 8
2	ACO [13,14]	tool-life; cutting force- power; surface roughness;	1 - 25 1 - 150	101.211 103.377	0.231 0.217	0.44 0.51	12.461 **12.235**	3 7
3	SA [15]	tool-life; cutting force- power; surface roughness;	1 - 1000 1 - 1400	112.852 108.464	0.194 0.221	0.46 0.41	**16.152** 16.171	12 11
4	GA [16, 17, 18]	tool-life; cutting force- power; surface roughness;	1-150 1-500	102.165 98.122	0.039 0.313	1.268 0.612	18.394 **14.661**	7 9

Proposed ANN-PSO approach was compared with three non-traditional techniques (GA, SA and ACO.

The results obtained from four techniques are given below in Table 1.

All the parameters and constraint sets are the same in all four cases. There is a total of 4 constraints.

Cutting forces and their influence on the economics of machining is summarized according to investigation of Kopac [6].

The proposed model is run on a PC compatible computer using the Matlab language.

The results revealed that the proposed method significantly outperforms the GA and SA approach.

The proposed approach found an optimal solution of 12.213 for as low as 1–115 runs the genetic-based approach require as much as 1–500 runs to find an solution of 14.661.

This means that the proposed approach has 20.04% improvement over the solution found by GA approach and 23.08% over SA [7] approach. Moreover, the simulated annealing approach (SA/PS) of [2] also generated an inferior solution of 16.152 for as much as 901–1000 runs which means that the optimal solution of PSO algorithm has an improvement of 32.2

It is observed that ANN-PSO has outperformed all other algorithms [4, 8]. Next ANN-PSO, SA and GA are ranked according to costs obtained from algorithms.

The costs obtained and optimum machining conditions are shown in Table 1. From the results, it is clear that the proposed ANN-PSO approach significantly outperforms the other two methods, such as GA and SA.

Clearly, the ANN-PSO approach provides a sufficiently approximation to the true optimal solution.

8. SUMMARY

Although several non-traditional optimization techniques have been applied to solve turning problems, their application is often limited due to lack of robustness and of getting stuck in local–optimum.

Neural network assisted particle swarm optimization is one of the recently developed optimization technique. The results indicate that it can be a very useful for optimization of machining conditions with the ability to escape local optimums.

It is also observed that the ANN-PSO system can obtain a near optimal solution when compared to GA and SA in large solution space.

This optimization system can be extended to optimize the parameters of other machining processes, such as drilling, turning, cylindrical grinding and unconventional machining processes

REFERENCES

[1] F. Cus, J. Balic, *Optimization of cutting process by GA approach,* Robot. comput. integr. manuf., Vol. 19, 2003, pp. 113−121.

[2] F. Cus, J. Balic, *Selection of cutting conditions and tool flow in flexible manufacturing system*, The international journal for manufacturing science & technology, Vol. 2, 2000, pp. 101−106.

[3] E. Dorigo, *The ant system: Optimization by a colony of cooperating agents*, IEEE Transaction on Systems, Man and Cybernetics, Vol. 26, 2006, pp. 1−13.

[4] R.C. Eberhart, Y. Shi, *Comparison Between Genetic Algorithm and Particle Swarm Optimization*, Proceedings of the 7th ICEC, pp. 611−616, 1998.

[5] U. Zuperl, F. Cus, V. Gecevska, *Optimization of the characteristic parameters in milling using the PSO evolution technique*, Stroj. vestn., Vol. 53, No. 6, 2007, pp. 354–368.

[6] J. Kopac, *Cutting forces and their influence on the economics of machining*, Stroj. vestn., Vol. 48, No. 3, 2002, 121−132.

[7] P.J Angeline, *Evolutionary Optimization Versus Particle Swarm Optimization: Philosophy and Performance Differences*, Proceedings of the 7th ICEC, pp. 601−610, 1998.

[8] U. Zuperl, F. Cus, *Optimization of cutting conditions during cutting by using neural networks*, Robot. comput. integr. manuf., Vol. 19, No. 1/2, 2003, pp. 189−199.