APEM journal

Advances in Production Engineering & Management
Volume 9 | Number 1 | March 2014 | pp 21–30
http://dx.doi.org/10.14743/apem2014.1.173

ISSN 1854-6250
Journal home: apem-journal.org
Original scientific paper

Particle swarm optimization approach for modelling a turning process

Hrelja, M.[a,*], **Klancnik, S.**[a], **Irgolic, T.**[a], **Paulic, M.**[a], **Jurkovic, Z.**[b], **Balic, J.**[a], **Brezocnik, M.**[a]

[a]Faculty of Mechanical Engineering, University of Maribor, 2000 Maribor, Slovenia
[b]Faculty of Engineering, University of Rijeka, 51000 Rijeka, Croatia

ABSTRACT

This paper proposes the modelling of a turning process using particle swarm optimization (PSO). The independent input machining parameters for the modelling were cutting speed, feed rate, and cutting depth. The input parameters affected three dependent output parameters that were the main cutting force, surface roughness, and tool life. The values of the independent and dependent parameters were acquired by experimental work and served as knowledge base for the PSO process. By utilizing the knowledge base and the PSO approach, various models could be acquired for describing the cutting process. In our case, three different polynomial models were obtained: models a) for the main cutting force, b) for surface roughness, and c) for tool life. All the models had exactly the same basic polynomial form which was chosen similarly to that in the conventional regression analysis method. The PSO approach was used for optimization of the polynomials' coefficients. Several different randomly-selected data sets were used for the learning and testing phases. The accuracies of the developed models were analysed. It was discovered that the accuracies of the models for different learning and testing data sets were very good, having almost the same deviations. The least deviation was noted for the cutting force, whilst the most deviation, as expected was for tool life. The obtained models could then be used for later optimization of the turning process.

ARTICLE INFO

Keywords:
Machining
CNC turning
Modelling
Optimization
Particle swarm optimization

**Corresponding author:*
marko.hrelja@um.si
(Hrelja, M.)

Article history:
Received 9 July 2013
Revised 26 November 2013
Accepted 15 January 2014

1. Introduction

Since the advent of modern manufacturing technologies and up-to-date machine tool CNC systems, shorter manufacturing times and higher manufacturing capabilities have been achieved that have led to reductions in final production costs, thus increasing profit margins. As the modern production technologies were significantly improving, this directly affected the optimizing of machining parameters. Machining experts can usually work with a design team so that machining can be optimized in order to obtain the best combination of cutting force, surface roughness, and minimal tool wear. Should the machining experts be eliminated from the process for any reason (i.e., employment issues, no experts available), intelligent methods could be used instead. Naturally the results can have some deviation from the true optimal values, however even experts cannot always provide the most optimal parameters for various situations.

In general, the concept behind all optimization algorithm variants is the same, namely that optimal cutting conditions are desired in order to reduce manufacturing costs. This is the easiest to achieve by combining basic cutting parameters. During the turning process the definable pa-

rameters are typically cutting speed, feed rate, and cutting depth. As the cutting diameter becomes progressively smaller, the revolutions should increase in order to obtain the same cutting speed, which usually is higher the lower the roughness is. In regard to feed rate it is exactly the opposite. Should lower roughness be preferred a reduction of the feed rate is needed, exactly the same as with cutting depth, which provides lower surface roughness if it is smaller. These combinations are crucial especially for finish turning, which usually consists of only one fine cut finish.

It is of the essence to take into consideration essential equations for machining that serve for understanding the concept of turning process modelling using particle swarm optimization (PSO). A study of cutting basics is required for this purpose, which would include descriptions from turning, milling, drilling, and grinding. The literature is mainly oriented towards high speed cutting, and this is a good starting point for optimal and fast manufacturing processes. It is also wise to check experimental results using the integrated approach for machining parameters, which was done by Liang et. al. [1], and Jafarian et al. by applying neural networks to the same process [2]. After all the equations and variables are known, input and output information is needed based on experimental work [3]. Bharati and Baskar introduced particle swarm optimization into manufacturing systems, as did Chan and Tiwari, however their work was based mainly on optimizing a single parameter per cutting operation, and for optimization purposes these individual parameters were not linked together with other cutting parameters (i.e., roughness, cutting force, tool life) [4, 5]. Cus and Balic presented the optimization of a machining process via GA algorithms [6]. El-Mounayri et al. composed an optimization algorithm for predicting surface roughness [7], whilst Senveter et al. used the neural network approach for the same problem [8]. Zuperl and Cus used neural networks as well for the machining optimization purposes [9]. Bushan conducted similar parameters' optimization, however solely for minimizing power consumption during machining and also for maximizing the tool life [10]. Byrne et al. implemented tool condition monitoring within the system [11]. A similar procedure was also introduced by Choudhury and Appa [12], however it was done solely for minimizing tool wear. The importance of proper cutting parameters selection has also been pointed out by Lee and Tarng [13]. Billatos and Tseng paved the way for knowledge-based optimization for intelligent machining, which is essential for proper particle swarm optimization procedure if we wish to optimize using more than one input parameter [14]. Brezocnik et al. proposed and developed a genetic programming system [15], as well as a very efficient and highly integrated genetic programming and genetic algorithm system for the modelling of surface roughness for different machining processes [16]. Quiza et al. upgraded a whole procedure to multi-objective optimization in order to increase the versatility of an algorithm [17].

This paper proposes a modelling of the machining process using particle swarm optimization by which models for specific materials can be prepared by successfully combining independent and dependent variables. Such polynomial models would serve for the later optimizations of manufacturing processes. It is vital to use as much input information as possible at the same time, as only in this way is it assumable that the polynomial will be accurate, as this affects the quality of optimization.

2. Experimental work

2.1 Equipment, tools, and materials

The experimental work presented in this paper was based on the work of Jurkovic Z. [3], and was carried out at the Production Engineering Institute, Faculty of Mechanical Engineering, at the University of Maribor. The aim of this experiment was to obtain suitable dependent output values regarding machining parameters from independent input machining parameters' values.

CNC machine tool:

A CNC lathe Georg Fischer NDM-16 was used for our experiment. The machine characteristics are briefly as follows.

- main electric motor power / safety limited: P = 30 kW, maximum P = 40 kW,
- feed rate motor power: P = 1.8 kW,
- maximal feed rate: f = 5000 mm/min,
- maximal workpiece size: Ø 160 mm × 500 mm,
- revolution area stage I: P = 27 kW; T = 625 Nm at 410 min^{-1}, 15-1140 min^{-1},
- revolution area stage II: P = 30 kW; T = 220 Nm at 1320 min^{-1}, 40-4000 min^{-1},
- tool system: Block tool system (BTS) – BT32.

Tool holder and insert:

- tool holder 0-3225P15,
- insert Sandvik Coromant DNMG 150608-PM4025: manufactured by CVD technology, middle layer Al_2O_3, top layer TiN covered.

Manufacturers recommended cutting conditions:

- v_c = 265-405 m/min,
- f = 0.15-0.50 mm/rev,
- a_p = 0.5-6 mm.

Tested material:

Workpiece material was carbonised steel with standard markings C45E (EN 10083/1996). The material was hot-rolled into a 6 m long cylinder with diameter of Ø 100 mm, and mass of 61.7 kg/m. After the essential forming into cylinders, it was tempered. The material was later cut into cylinder lengths with dimensions of Ø 100 mm × 380 mm.

Measuring tools:

In order to obtain the measuring results, the measuring tools had to successfully acquire the following measurements as required: main cutting force F_C, surface roughness R_a, and maximal tool life T. The measurement equipment was:

- cutting force: Kistler 9257A dyanometer, which had a measuring area covering three axes $F_{x,y,z}$ = 5 kN, which sent the measured signal to the computer by utilizing LabVIEW™,
- surface roughness: SJ-201P Mitutoyo measuring unit with reference values 2.5 mm,
- tool wear: Carl Zeiss microscope with magnification of 30× and resolution of 0.0001 mm.

2.2 Experimental results

The measured values during the experiment were of the cutting force, surface roughness and tool life, whilst the given parameters were surface speed, feed rate, and cutting depth. Suitable equipment was used for obtaining correct parameters, and monitoring those tools that gave us proper results.

Input parameters:

- cutting speed – v_c [m/min],
- feed rate – f [mm/rev],
- cutting depth – a_p [mm].

Output parameters:

- main cutting force – F_C [N],
- surface roughness – R_a [μm],
- maximal tool life – T [min].

Using these parameters, including polynomial equations optimization, successful multiple regression analysis implementation can be achieved. However, the basis of this paper is a non-deterministic approach, so regression analysis will not be analytical but a stochastic method

based on acquiring a particle swarm algorithm that does the computing of the coefficients of the prescribed mathematical model. The measured values essential for rough turning are presented in Table 1.

Cutting speed is a tangential component of the spindle speed, which is measured in min^{-1}. In general, for finish cutting it is of the essence that the cutting speed is noticeably higher than the one used for rough machining. In contrast the feed rate requires the finish machining to be lower than for the roughing. The same also applies for the cutting depth, which is also much smaller with the finish cutting. The input and output parameters are shown graphically in Fig. 1.

Table 1 Input and output values for rough turning

	Input values			Output values		
Nr.	V_c [m/min]	f [mm/rev]	a_p [mm]	F_c [N]	R_a [μm]	T [min]
1	300	0.30	1.50	879.2240	4.300	17.6
2	400	0.30	1.50	894.3270	3.880	4.73
3	300	0.50	1.50	1436.299	11.11	6.68
4	400	0.50	1.50	1408.114	11.48	1.88
5	300	0.30	3.00	1754.215	4.210	13.8
6	400	0.30	3.00	1726.937	4.500	3.80
7	300	0.50	3.00	2896.122	14.29	4.10
8	400	0.50	3.00	2860.663	13.71	1.16
9	350	0.40	2.25	1677.149	8.100	5.38
10	350	0.40	2.25	1672.771	8.130	5.10
11	350	0.40	2.25	1679.359	8.120	5.44
12	350	0.40	2.25	1678.825	8.120	5.28
13	350	0.40	2.25	1675.829	8.110	5.50
14	350	0.40	2.25	1678.223	8.100	5.22
15	266	0.40	2.25	1697.504	7.820	12.9
16	434	0.40	2.25	1683.361	8.150	1.81
17	350	0.23	2.25	1002.763	2.460	10.5
18	350	0.57	2.25	2609.254	17.95	0.75
19	350	0.40	1.00	765.9210	6.360	6.65
20	350	0.40	3.50	2746.389	9.070	3.58

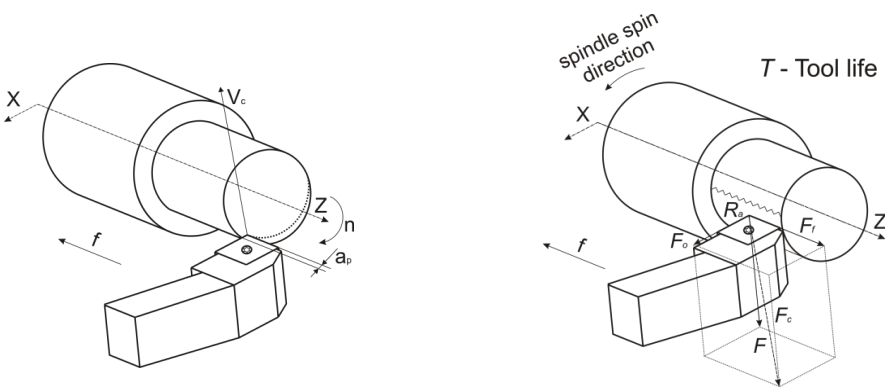

Fig. 1 Input values (left) and output values (right)

3. Used methods – PSO algorithm

Particle swarm optimization algorithm uses stochastic operations and is designed around a population of organisms/particles. This algorithm is based upon a living organism model, such as flocks of birds. These organisms then interact upon social-psychological correlations, the very same way as living organisms and have the possibility of adapting to various problems.

3.1 Basis information

Randomly generated initial organisms/particles are needed in order to determine the optimal solution. The algorithm works within a basic route, which is determined using solution particle position and particle velocity vectors as a guide, and where we can determine that a certain solution, within a certain optimization time, is currently determined by the velocity vector, which defines our best solution. This is determined as a fitness function for each organism which is also commonly known as the capability of finding a better solution. Such a vector marks the personal best values for each single organism within the system and is called the personal best solution – pBest. In contrast, each particle swarm within every singular moment has its best global position, which is called gBest. At each cycle the repetition values for pBest and gBest are updated.

3.2 Computational model

Initial locations for each organism within the search space are created randomly. After that the algorithm conducts optimizing cycles, where with each repetition the current personal best solution (pBest) and global best solution (gBest) are searched for. Eq. 1 shows the core of the optimization algorithm, whilst Eq. 2 stands for updating the particle location after each optimization cycle.

$$v_i = v_i + c_1 rand(\,)(p_i - x_i) + c_2 Rand(\,)(p_g - x_i) \tag{1}$$

Particle location update:

$$x_i = x_i + v_i \tag{2}$$

The variables in Eq. 1 and Eq. 2 represent:

- c_1 in c_2 – acceleration coefficients (acceleration coefficients),
- $rand(\,)$ and $Rand(\,)$ – random values within interval (0 1),
- x_i – i^{th} particle,
- p_i – pBest for i^{th} particle,
- p_g – gBest of all particles (global best particle),
- v_i – velocity update value for particle i.

The particle swarm optimization equation (Eq. 1) consists of three terms. The first term allows initialisation and it is not changed, however it does get us to the current velocity and initial solution location. The second term allows that a particle learns from its own experiences, and in the third term the particles interact with each other, exchanging valuable expertise for solving the problems. Therefore, the pseudocode of the PSO algorithm can be written as shown in Fig. 2.

```
Start PSO
    For each particle
        Initialize particle
    END
    Do
            For each particle
                Calculate particle fitness
                If fitness function > best particle fitness (pBest) then value becomes new pBest
                Choose particle with best fitness value (gBest)
            END
            For each particle
                Calculate particle velocity
                Update particle location
            END
    While maximal iterations N are reached, or maximal tolerated error is reached
END PSO
```

Fig. 2 Particle swarm optimization pseudocode

4. Modelling results and discussion of the prediction model

As previously stated, the optimization algorithm starts with the initialisation of particles with random velocity values. Manually adjustable acceleration coefficients c_1 and c_2 are required, which are directing the algorithm searching abilities in search space. The rest of the parameters (i.e., pBest, gBest) are manipulated and updated directly through the optimization algorithm. If a finish machining optimization model is desired, despite the similarity, two separate procedures have to be initiated in order to obtain results for both rough machining and finishing machining, regardless of the machining process type. According to Fig. 2, a knowledge-based table has to be included into the initial procedure. Basic prediction polynomial model has to be created at this point. On the basis of the preliminary results, the following polynomial model was chosen for the modelling of the turning process:

$$f(x_1, x_2, x_3) = k_1 + k_2 \cdot x_1 + k_3 \cdot x_2 + k_4 \cdot x_3 + k_5 \cdot x_1 \cdot x_2 + k_6 \cdot x_1 \cdot x_3 + k_7 \cdot x_2 \cdot x_3 + \\ + k_8 \cdot x_1 \cdot x_2 \cdot x_3 \tag{3}$$

Here $f(x_1, x_2, x_3)$ stand for any of the three resulting output machining parameters:

- main cutting force – F_c,
- surface roughness – R_a,
- maximal tool life – T.

Parameters x_1, x_2, x_3 in Eq. 3 are independent input parameters: cutting speed, feed rate, and cutting depth. Optimization polynomials for dependent machining parameters will be obtained by applying the PSO algorithm to the learning data set. These dependent values' polynomials (cutting force, surface roughness, and tool life) are then useable for processing and optimizing the turning process by means of multi-objective optimization, in order to determine the most optimal input data set for machining the surface to optimal roughness, with minimal cutting force and maximal tool life duration.

4.1 PSO parameters

The following results are representative values only for the material C45E (EN 10083/1996). Under different machining circumstances, the developed system still remains completely the same; the user only has to prepare the new knowledge-base.

In order to properly set a PSO algorithm, choosing certain essential additional parameters is in order, which in our case will be:

- number of iterations 500000,
- correction factor c_1 = 1.2,
- correction factor c_2 = 2.4,
- swarm size 35,
- particle size 8.

4.2 Modelling results

Modelling of the machining process was done by PSO, therefore the optimizing procedure was controlled by PSO parameters. On the basis of the particle swarm optimization algorithm's architecture, the display of the results had to be done in the form of coefficients k_1, k_2,..., k_8, which determined the specific combination and weight factor per independent machining parameter. The results of four PSO algorithm runs are shown in Tables 2, 3 and 4. Those polynomial coefficients, that were acquired using the PSO approach for cutting force, surface roughness, and tool life were representative and would serve for preparing the computing models according to Eq. 3.

Table 2 Coefficients for F_c

Coefficients	Run No. 1	2	3	4
k_1	-484.575	-494.739	-513.83	-551.681
k_2	1.76549	1.794344	1.848204	1.955626
k_3	1049.84	1074.501	1120.634	1212.698
k_4	190.7856	195.0709	203.0669	219.0319
k_5	-3.95561	-4.02566	-4.15588	-4.41687
k_6	-0.63984	-0.65201	-0.67457	-0.7199
k_7	1592.993	1582.59	1563.301	1524.522
k_8	1.185408	1.214965	1.26942	1.379419

Table 3 Coefficients for R_a

Coefficients	Run No. 1	2	3	4
$k1$	12.81179	4.868139	4.148009	5.390292
$k2$	-0.04611	-0.02361	-0.02159	-0.0251
$k3$	-17.3847	1.716672	3.550831	0.593602
$k4$	-9.44103	-6.05252	-5.78467	-6.3452
$k5$	0.117782	0.063638	0.058538	0.066865
$k6$	0.020514	0.010912	0.010171	0.011751
$k7$	26.79597	18.64592	17.96812	19.2813
$k8$	-0.05334	-0.03023	-0.02837	-0.03207

Table 4 Coefficients for T

Coefficients	Run No. 1	2	3	4
k_1	75.93721	47.00029	56.27326	58.90766
k_2	-0.14739	-0.06632	-0.09197	-0.09925
k_3	-71.3179	-2.12526	-23.9187	-29.7181
k_4	9.408307	21.41499	17.60706	16.57052
k_5	0.099249	-0.09507	-0.03459	-0.01866
k_6	-0.03268	-0.0664	-0.05579	-0.05294
k_7	-38.9424	-67.6777	-58.818	-56.505
k_8	0.116313	0.197149	0.172374	0.166003

4.3 The best models

Several different combinations of learning and testing data sets were applied during modelling of prediction models. Eight of the experimental results were applied for the learning phase and the remaining 12 for testing the prediction model. Different combinations of learning input data sets provided similar results in terms of accuracy. Note, it was possible to encounter slight differences between the initial polynomial (Eq. 3) and the final polynomials, due to certain coefficients' eliminations, hence the partial result was insignificant for the final result. The best models obtained by particle swarm optimization for the rough turning were:

$$F_c = -484.575 + 1.76549 \cdot x_1 + 1049.84 \cdot x_2 + 190.7856 \cdot x_3 - 3.95561 \cdot x_1 \cdot x_2 - 0.63984 \cdot x_1 \cdot x_3 + 1592.993 \cdot x_2 \cdot x_3 + 1.185408 \cdot x_1 \cdot x_2 \cdot x_3 \quad (4)$$

$$R_a = 12.81179 - 0.04611 \cdot x_1 - 17.3847 \cdot x_2 - 9.44103 \cdot x_3 + 0.117782 \cdot x_1 \cdot x_2 + 0.020514 \cdot x_1 \cdot x_3 + 26.79597 \cdot x_2 \cdot x_3 - 0.05334 \cdot x_1 \cdot x_2 \cdot x_3 \quad (5)$$

$$T = 75.93721 - 0.14739 \cdot x_1 - 71.3179 \cdot x_2 + 9.408307 \cdot x_3 + 0.099249 \cdot x_1 \cdot x_2 - 0.03268 \cdot x_1 \cdot x_3 - 38.9424 \cdot x_2 \cdot x_3 + 0.116313 \cdot x_1 \cdot x_2 \cdot x_3 \quad (6)$$

Although analytical multiple regression analysis seemed to be easier to calculate, however, per the results the PSO algorithm was superior to it in terms of the amount of work required to obtain the same, similar or even better results. The following results are representative results for the PSO algorithm for rough tuning with material C45E (EN 10083/1996.) The data set determined for main cutting force F_C (Eq. 4), surface roughness R_a (Eq. 5), and tool life T (Eq. 6) provide a full set of information for an elementary model of the turning process, where the predictions are presented in paragraph 4.4.

4.4 Testing phase and deviation analysis

The developed models had to be proved during testing phase. The results presentation has been simplified due to the extensive amount of data included within the analysis for all three output values: cutting force F_c, surface roughness R_a, and tool life T. Only results for surface roughness are presented here. Experimental and predicted values for surface roughness are presented in Table 5. In the same way as the results for surface roughness, identical tables have been prepared for cutting force and tool life. These results are described in detail in the next paragraph.

Table 5 Calculated values for surface roughness R_a, with the inclusion of deviation analysis

Experimental value R_a [μm]	Prediction 1 [μm]	Prediction 2 [μm]	Prediction 3 [μm]	Prediction 4 [μm]	Max. deviation [%]	Min. deviation [%]
4.30	4.289	4.169	4.159	4.173	3.264	0.232
3.88	3.888	3.994	4.005	3.988	3.235	0.231
11.11	11.118	11.204	11.219	11.202	0.986	0.075
11.48	11.472	11.395	11.385	11.392	0.825	0.062
4.21	4.217	4.310	4.315	4.291	2.514	0.174
4.50	4.493	4.412	4.410	4.425	1.978	0.150
14.29	14.284	14.218	14.213	14.218	0.536	0.041
13.71	13.715	13.779	13.777	13.766	0.504	0.038
8.10	8.435	8.435	8.435	8.432	4.147	4.101
8.13	8.435	8.435	8.435	8.432	3.763	3.716
8.12	8.435	8.435	8.435	8.432	3.890	3.844
8.12	8.435	8.435	8.435	8.432	3.890	3.844
8.11	8.435	8.435	8.435	8.432	4.019	3.972
8.10	8.435	8.435	8.435	8.432	4.147	4.101
7.82	8.506	8.503	8.505	8.497	8.776	8.663
8.15	8.363	8.367	8.366	8.366	2.673	2.622
2.46	1.273	1.272	1.273	1.270	48.345	48.213
17.95	15.59	15.599	15.597	15.593	13.127	13.097
6.36	7.197	7.194	7.196	7.193	13.169	13.106
9.07	9.672	9.676	9.675	9.671	6.690	6.624

The model for cutting force F_c, was the most accurate prediction model as the percentage deviation reached a minimum of 0.001 %, however, in certain cases the value of 6.3 % was exceeded. The reason for such a high percentage error is probably the single cutting force optimization procedure (i.e., only the main cutting force was taken into consideration), therefore error difference might be derived from incomplete model. The average deviation of the cutting force was marked at around 1.75 %, solely due to the fact of few higher percentage deviation values.

Data analysis for surface roughness R_a, is shown in detail in Table 5, however a few important facts are still in order for properly displaying the optimization model. The minimum deviation was at 0.04 %, whilst on the other hand the maximum remained at 48 %. Interestingly this value

was for the same cut as the maximal deviation for the cutting force. By considering the maximal value to be correct, this provides us with an average error of 5.85 %, however, if we were to eliminate the possible incorrect measurement, this value would decrease significantly.

However, in regard to tool life values *T*, as the experimental values became drastically lower, the optimization analysis became harder. As with the preliminary results, as the minimal deviation approached 4.31 % and the maximal value remained at 60 %, the average for the tool-life values increased up to 24.5 %. One way to decrease such a high value of error is to increase the knowledge-base. As previously mentioned, we took only eight cuts (i.e., measurements) during the learning phase, and such a low amount of information in combination with the low output values, combine to create a higher error possibility. On the other hand, with the inclusion of all available twenty cuts within the knowledge-base for the learning phase, the average error decreased to a value of 17.54 %. Should we have had even more available experimental results, the error would have decreased even more.

5. Conclusion and future research

This article proposed a particle swarm optimization approach for predicting (i.e., modelling) of cutting force, surface roughness, and tool-life. The predictions are based on independent input parameters (cutting speed, feed rate, and cutting depth). Conclusions from the research are:

- The particle swarm optimization approach can be successfully used for the modelling of machining processes such as turning and similar cutting processes.
- The proposed approach provides comparable results to other well-known approaches, such as conventional regression analysis.
- If the dependent output values are of higher value (i.e., cutting force), a smaller knowledge base can be used but in contrast, if the dependent values are lower (i.e., tool life), the number of independent values (i.e., number of measurements) will at least be doubled.
- The obtained models have relatively simple polynomial forms and may be further optimized by various approaches, such as genetic algorithms. They may also serve as inputs to special system based on multi-objective optimization (e.g., by using NSGA-II algorithm).

During the research we decided to develop and implement a relatively new gravitational search algorithm (GSA), which is based on physical gravitational laws [18, 19]. Preliminary tests showed slight deviations of the results, however the data processing was very fast [20]. In addition, the models obtained by the PSO will be further optimized by multi-objective optimization approaches, such as NSGA-II, SPEA2, and DEMO.

Acknowledgment

The authors would like to acknowledge support and funding to Ministry of Education, Science and Sport, and Slovenian Research Agency in association with University of Maribor, Slovenia for providing the resources for postgraduate studies.

References

[1] Liang, M., Mgwatu, M., Zuo, M. (2001). Integration of cutting parameter selection and tool adjustment decisions for multipass turning, *The International Journal of Advanced Manufacturing Technology*, Vol. 17, No. 12, 861-869, doi: 10.1007/s001700170097.

[2] Jafarian, F., Taghipour, M., Amirabadi, H. (2013). Application of artificial neural network and optimization algorithms for optimizing surface roughness, tool life and cutting forces in turning operation, *Journal of Mechanical Science and Technology*, Vol. 27, No. 5, 1469-1477, doi: 10.1007/s12206-013-0327-0.

[3] Jurkovic, Z. (2007). *Modelling and optimization of cutting parameters using evolutionaty algorithms in intelligent machining systems*, (original Croatian title: *Modeliranje i optimizacija parametara obrade primjenom evolucijskih algoritama kod inteligentnih obradnih sustava*), doctoral dissertation, Rijeka, Croatia.

[4] Bharathi Raja, S., Baskar, N. (2011). Particle swarm optimization technique for determining optimal machining parameters of different work piece materials in turning operation, *International Journal of Advanced Manufacturing Technology*, Vol. 54, No. 5-8, 445-463, doi: 10.1007/s00170-010-2958-y.

[5] Chan, F.T.S., Tiwari, M.K. (2007). *Swarm intelligence, focus on ant and particle swarm optimization*, I-Tech Education and Publishing, Austria.
[6] Cus, F., Balic, J. (2003). Optimization of cutting process by GA approach. *Robotics and Computer-Integrated Manufacturing,* Vol. 19, No. 1-2, 113-121, doi: 10.1016/S0736-5845(02)00068-6.
[7] El-Mounayri, H., Dugla, Z., Haiyan, D. (2013). Prediction of surface roughness in end milling using swarm intelligence, In: *Swarm Intelligence Symposium SIS '03. Proceedings of the 2003 IEEE*, doi: 10.1109/SIS.2003.1202272.
[8] Senveter, J., Klancnik, S., Balic, J., Cus, F. (2010). Prediction of surface roughness using a feed-forward neural network, *Management and Production Engineering Review*, Vol. 1, No. 2, 47-55.
[9] Zuperl, U., Cus, F. (2003). Optimization of cutting conditions during cutting by using neural networks, *Robotics and Computer-Integrated Manufacturing*, Vol. 19, No. 1-2, 189-199, doi: 10.1016/S0736-5845(02)00079-0.
[10] Bhushan, R.K. (2013). Optimization of cutting parameters for minimizing power consumption and maximizing tool life during machining of Al alloy SiC particle composites, *Journal of Cleaner Production,* Vol. 39, 242–254, doi: 10.1016/j.jclepro.2012.08.008.
[11] Byrne, G., Dornfeld, D., Inasaki, I., Ketteler, G., König, W., Teti, R. (1995). Tool condition monitoring (TCM) – the status of research and industrial application, *CIRP Annals – Manufacturing Technology*, Vol. 44, No. 2, 541-567, doi: 10.1016/S0007-8506(07)60503-4.
[12] Choudhury, S.K., Appa Rao, I.V.K. (1999). Optimization of cutting parameters for maximizing tool life, *International Journal of Machine Tools and Manufacture*, Vol. 39, No. 2, 343-353, doi: 10.1016/S0890-6955(98)00028-5.
[13] Lee, B.Y., Tarng, Y.S. (2000). Cutting-parameter selection for maximizing production rate or minimizing production cost in multistage turning operations, *Journal of Materials Processing Technology*, Vol. 105, No. 1-2, 61-66, doi: 10.1016/S0924-0136(00)00582-3.
[14] Billatos, S.B., Tseng, P.-C. (1991). Knowledge-based optimization for intelligent machining, *Journal of Manufacturing Systems*, Vol. 10, No. 6, 464-475, doi: 10.1016/0278-6125(91)90004-L.
[15] Brezocnik, M., Kovacic, M., Ficko, M. (2004). Prediction of surface roughness with genetic programming, *Journal of Materials Processing Technology*, Vol. 157/158, 28-36, doi: 10.1016/j.jmatprotec.2004.09.004.
[16] Brezocnik, M., Kovacic, M. (2003). Integrated genetic programming and genetic algorithm approach to predict surface roughness, *Materials and Manufacturing Processes*, Vol. 18, No. 3, 475-491, doi: 10.1081/AMP-120022023.
[17] Sardiñas, R.Q., Santana, M.R., Brindis, E.A. (2006). Genetic algorithm-based multi-objective optimization of cutting parameters in turning processes, *Engineering Applications of Artificial Intelligence*, Vol. 19, No. 2, 127-133, doi: 10.1016/j.engappai.2005.06.007.
[18] Rashedi, E., Nezamabadi-pour, H., Saryazdi, S. (2009). GSA: a gravitational search algorithm, *Information Sciences*, Vol. 179, No. 13, 2232-2248, doi: 10.1016/j.ins.2009.03.004.
[19] Balachandar, S.R., Kannan, K. (2010). A meta-heuristic algorithm for set covering problem based on gravity, *International Journal of Computational & Mathematical Sciences*, Vol. 4, No. 5, 223-228.
[20] Hrelja, M., Klancnik, S., Balic, J., Brezocnik, M. (2014). Modelling of a turning process using the gravitational search algorithm, *International Journal of Simulation Modelling*, Vol. 13, No. 1, 30-41, doi: 10.2507/IJSIMM13(1)3.248.