

Multi-objective optimization of the turning process using a Gravitational Search Algorithm (GSA) and NSGA-II approach

Klančnik, S.^{a,*}, Hrelja, M.^{a,b}, Balic, J.^a, Brezocnik, M.^a

^aProduction Engineering Institute, Faculty of Mechanical Engineering, University of Maribor, Maribor, Slovenia

^bAVL, Piezocryst: Advanced Sensorics GmbH, Graz, Austria

ABSTRACT

In this paper, we proposed a Gravitational Search Algorithm (GSA) and an NSGA-II approach for multi-objective optimization of the CNC turning process. The GSA is a swarm intelligence method exploiting the Newtonian laws on elementary mass objects interaction in the search space. The NSGA-II is an evolutionary algorithm based on non-dominated sorting. On the basis of varying values of the three independent input machining parameters (i.e., cutting speed, depth of cut, and feed rate), the values of the three dependent output variables were measured (i.e., surface roughness, cutting forces, and tool life). The obtained data set was divided further into two subsets, for the training data and the testing data. In the first step of the proposed approach, the GSA and the training data set were applied to modelling a suitable model for each output variable. Then the accuracies of the models were checked by the testing data set. In the second step, the obtained models were used as the objective functions for a multi-objective optimization of the turning process by the NSGA-II. The optimization constraints relating to intervals of dependent and independent variables were set on the theoretical calculations and confirmed with experimental measurements. The goal of the multi-objective optimization was to achieve optimal surface roughness, cutting forces, and increasing of the tool life, which reduces production costs. The research has shown that the proposed integrated GSA and NSGA-II approach can be implemented successfully, not only for modelling and optimization of the CNC turning process, but also for many other manufacturing processes.

© 2016 PEI. University of Maribor. All rights reserved.

ARTICLE INFO

Keywords:

Turning
Multi-objective optimization
Evolutionary algorithms
Particle swarm
Gravitational Search Algorithm
NSGA-II algorithm

*Corresponding author:

simon.klančnik@um.si
(Klančnik, S.)

Article history:

Received 8 July 2016
Revised 19 November 2016
Accepted 21 November 2016

1. Introduction

Advances in existing production are measured in terms of the flexibility, autonomous functioning and extent of automation. Modernisation of production covers the modernisation and integration of the latest technologies into production systems so that the companies remain competitive. Integration of advanced technologies results in shorter manufacturing times, higher capacities and reduction of production costs. Because of the dynamics and increase in the amount of input parameters or data, the optimization of systems has become more difficult than previously. Because of high quantities of data, the optimal functioning of production can often be reached no more by conventional methods; therefore, more advanced methods are applied to acquire new and better solutions in the optimization of production systems.

The research is aimed at conceiving an advanced system for modelling of CNC machining operations and subsequent optimization of machining parameters by the use of the developed machining models. In the past, a lot of researches have been executed in the field of optimization

and modelling of the CNC machining operations [1-3]. Important researches in this field have been carried out for optimization of milling [4], turning [5], laser cutting and water jet cutting [6]. Successful optimization relies on the use of the training and testing databases [7] comprised of the values measured in experiments [8, 9]. The training and testing databases allow the training of algorithms and testing of correctness of the models developed. The development of modern methods of optimizing and modelling of complex production systems requires know-how from the area of intelligent systems, which presupposes incorporation of artificial intelligence into solving complex engineering problems. One of the most widespread nondeterministic approaches for solving complex engineering problems is evolutionary computation, especially genetic algorithms and genetic programming [10]. There are many works where genetic algorithms, as well as other nondeterministic methods (e.g., Artificial Neural Networks – ANN) were implemented for solving various problems in cutting systems, including surface roughness optimization and optimal tool selection (see for example [1, 14-21]). Another artificial intelligence method used frequently is swarm intelligence. Swarm intelligence is only a covering term and is divided into many sub-types [22]. Often, the authors refer to intelligence adopted from biological aspects, e.g. bee swarms, bird flights or ant colonies [23]. A relative novelty in the swarm intelligence methods is the Gravitational Search Algorithm (GSA) working on the principle of the interaction of masses [24]. A review of the literature reveals that, so far, the GSA has not been reported frequently for optimization of manufacturing technologies.

This paper proposed an effective approach for modelling and multi-objective optimization of CNC machining processes by the GSA and NSGA-II. Data gained on the basis of experimental measurements were used for training and testing of the model developed by the GSA. After modelling, the multi-objective optimization of the cutting parameters was done on the basis of CNC cutting models developed by the GSA. The NSGA-II algorithm was implemented for the multi-objective optimization. The system is generally convenient for different CNC machine tools and can be adapted to several different materials' removing processes, e.g. milling and laser cutting. However, our research was restricted to turning as far as training of the system and testing of its effectiveness are concerned.

The article is composed as follows: Section 2 presents detailed descriptions of the experimental setup, Gravitational Search Algorithm, and elitist Non-dominated Sorting Genetic Algorithm's (NSGA-II) background. The results and discussion are presented in Section 3. The article ends with a short Conclusion.

2. Materials and methods

2.1 Experimental setup and results

Models of the turning process were created by means of experimental measurements of the output parameters of machining operations depending on the input parameters, in accordance with the work of Jurkovic [25]. The experiment was carried out during a fine turning operation on the CNC lathe Georg Fischer NDM-16. Samples from the material C45E were used for execution of the experiment. This is a hot rolled structural steel. The workpiece diameter for machining was $\varnothing 10 \text{ mm} \times 380 \text{ mm}$. The tool used for the test samples was the cutting insert Sandvik Coromant DNMG 150608-PM4025. The following equipment was used for proper measuring of tool wear, machining forces, and surface roughness:

- Measuring of cutting forces:
Force meter Kistler 9257A. Measuring range $F_{x,y,z} = 5 \text{ kN}$ allowing capturing, displaying and processing of resulting cutting forces with the aid of the programme package Labview™.
- Surface roughness:
Roughness meter R_a produced by Mitutoyo SJ-201P.
- Tool wear:
Microscope Carl Zeiss with magnification $30\times$ and 0.0001 mm resolution.

The input data for execution of the experiment were as follows:

- Cutting speed V_c (m/min)
- Feed rate f (mm/rev) and
- Cut depth a_p (mm).

The output data were gained from the input data, as the latter have direct impact on their values. The output data and/or the experiment results were:

- Main cutting force F_c (N)
- Surface roughness R_a (μm) and
- Tool resistance time T (min).

The experimental results obtained during the fine turning are presented in Table 1.

Table 1 The results of the experiment

Trial No.	Input data			Output data		
	V_c (m/min)	f (mm/rev)	a_p (mm)	F_c (N)	R_a (μm)	T (min)
1	400	0.100	0.40	128.893	0.77	32.66
2	500	0.100	0.40	130.755	0.80	11.15
3	400	0.200	0.40	201.899	1.70	25.89
4	500	0.200	0.40	202.200	1.67	7.450
5	400	0.100	1.20	337.859	1.11	28.43
6	500	0.100	1.20	330.745	1.19	9.230
7	400	0.200	1.20	492.945	2.14	20.74
8	500	0.200	1.20	550.848	1.77	5.610
9	450	0.150	0.80	299.005	1.26	14.44
10	450	0.150	0.80	301.647	1.30	14.38
11	450	0.150	0.80	304.772	1.29	14.39
12	450	0.150	0.80	299.519	1.28	14.48
13	450	0.150	0.80	299.875	1.27	14.43
14	450	0.150	0.80	303.832	1.28	14.46
15	366	0.150	0.80	313.225	1.37	34.46
16	534	0.150	0.80	307.622	1.31	6.120
17	450	0.066	0.80	174.024	1.21	20.25
18	450	0.234	0.80	406.719	2.32	10.93
19	450	0.150	0.13	61.2230	1.17	12.18
20	450	0.150	1.47	497.895	1.13	10.05

Training data

Testing data

2.2 Gravitational Search Algorithm background

In the Gravitational Search Algorithm, a set of objects with relevant atomic masses is concerned, the objects representing the search mechanism [24]. Those objects find the proper solution during the searching process. The search mechanism works on the principle of Newtonian gravitation laws and laws of motion and, on the basis of gravitational attraction, it searches for the optimum solution in the multitude of potential solutions, where each of the solutions has a specific mass. All particles within the population are attracted by mutual gravitational forces according to Eq. 1 and 2, which causes global motion of objects in the system, heavier objects attracting lighter objects to a larger extent. The objects communicate mutually by means of the gravitational force. Larger masses result faster in a better solution, but they move more slowly than lower masses with worse solutions. In the GSA, each particle has four basic properties: Location, inertial mass, active gravitational mass and passive gravitational mass. The location of mass coincides with the location of the result of the problem. With the aid of the active, passive and inertia masses, it is also possible to write the mathematical function. In that way, the algorithm is guided through the space of solutions by means of mutual effects of gravitational and inertia masses. After expiration of the search time, the solutions are expected to start tending to the

largest mass in the system, that mass increasing during the search process and representing the optimum solution in the search space. The search space represented by the algorithm is not influenced by the external masses so that only two physical laws, i.e. the gravitation law and the law of motion act on it. In a system of two or more bodies, it is the active gravitational mass M_A that affects most the remaining bodies inside the system with its gravitational field, which means that it has the greatest mass. The passive gravitational mass M_P represents the gravitational field of the body with less mass and affects the gravitational field of the more massive object passively only and has a lesser mass. The system of several bodies must be calculated in pairs, two by two bodies simultaneously. The inertia gravitational mass M_I is the mass with which the object in the system resists the position change, if acted upon by a specific mass. The body having larger inertia mass changes the direction of motion slower than the body with smaller mass, as the force must act on the body longer in order to direct the body into a new direction. If the Newtonian equations are written by using newly named masses in combination with two bodies, the result is as follows:

$$F_{12} = G \frac{M_{A2} \cdot M_{P1}}{R^2} \quad (1)$$

where:

- F_{12} – Attraction with which the second body acts on the first body
- G – Gravitational constant
- M_{A2} – Active mass of second object
- M_{P1} – Passive mass of first object
- R – Distance between the two objects

Consequently, the gravitational force F_{12} acting on the first object with the mass of the second object is proportional to the product of the active part of gravitational mass in the system with the passive part of gravitational mass, and inversely proportional to the distance between the two objects. The acceleration with which the body moves and/or the effect of the second on the first body can be written with the equation:

$$a_1 = \frac{F_{12}}{m_{I1}} \quad (2)$$

- a_1 – Acceleration with which the body moves; effect of the second on the first body
- F_{12} – Attraction with which the second body acts on the first body
- m_{I1} – Inertia mass of the first body

Transformation of the system by means of a random component ensures random motion of mass bodies.

$$F_a^n(t) = \sum_{b=1, b \neq i}^N rand_b \cdot F_{ab}^n(t) \quad (3)$$

where:

- $F_a^n(t)$ – Force with stochastic characteristic in given time t
- $rand_b$ – Random number at interval $[0,1]$ ensures randomization
- $F_{ab}^n(t)$ – Force which is caused by the b -th mass particle and acts on the a -th mass particle in time t

As the stochastic mode of operation of gravitational force has been given, the same must be effected also on the gravitational acceleration and, thus, Eq. 2 is transformed to an n -dimensional equation in specific time t :

$$a_a^n(t) = \frac{F_a^n(t)}{M_{ia}(t)} \quad (4)$$

where:

- $a_a^n(t)$ – Gravitational acceleration of body a in the n -dimension space
 $F_a^n(t)$ – Force with stochastic characteristic in specific time t
 $M_{ia}(t)$ – Inertia mass of body a in specific time t

On the basis of the updated equation for the force with stochastic characteristic in specific time t , and on the basis of the gravitational acceleration of body a in the d -dimension space, the new velocity of particle or body in the space with n -dimensions – Eq. 5, can be expressed; the same also applies to the newly calculated position of the body in the solution space – Eq. 6:

$$v_a^n(t+1) = rand_a \cdot v_a^n(t) + a_a^n(t) \quad (5)$$

$$x_a^n(t+1) = x_a^n(t) + v_a^n(t+1) \quad (6)$$

Terms in Eq. 5 and 6:

- $v_a^n(t+1)$ – New velocity of body a in newly calculated time $t+1$
 $x_a^n(t+1)$ – New position of body a in newly calculated time $t+1$
 $rand_a$ – Random number at interval $[0,1]$ ensures randomization
 $v_a^n(t)$ – Velocity of motion of body a in specific time t
 $a_a^n(t)$ – Acceleration of body a in specific time t
 $x_a^n(t)$ – Position of body a in specific time t
 n – Index n stands for the n -dimension space with solutions

Gravitational and inertia masses of bodies in the system can also be written as a fitness function to define the success of the algorithm in searching for the solution. It must be borne in mind that larger masses are better indicators of solutions and are more effective; however, as those bodies have larger mass, they move slower in the space with solutions. That phenomenon is compensated for by a random number of lighter bodies representing worse solutions but, in combination with heavier bodies, they take part in converging to proper solutions much faster. If it is assumed that gravitational and inertia masses are identical, they can be written in the form of an equation. Updated equations of the mass system are:

$$M_{an} = M_{pn} = M_{in} = M_n \cdot n = 1, 2, \dots, N \quad (7)$$

$$m_a(t) = \frac{fit_a(t) - worst(t)}{best(t) - worst(t)} \quad (8)$$

$$M_a(t) = \frac{m_a(t)}{\sum_{b=1}^N m_b(t)} \quad (9)$$

where $fit_a(t)$ stands for the fitness value of body a in specific time t , while $best(t)$ and $worst(t)$ are values representing the best and worst current values of results and can be written even in more detail, particularly, when searching for minimums and maximums.

For minimum:

$$\begin{aligned} best(t) &= \min_{j \in (1, 2, \dots, N)} fit_b(t) \\ worst(t) &= \max_{j \in (1, 2, \dots, N)} fit_b(t) \end{aligned} \quad (10)$$

For maximum:

$$\begin{aligned} best(t) &= \max_{j \in (1, \dots, N)} fit_b(t) \\ worst(t) &= \min_{j \in (1, \dots, N)} fit_b(t) \end{aligned} \quad (11)$$

To avoid efficiently the local minimums at the beginning of the algorithm start-up the research principle must be introduced, whereas, after n -iterations, the research limit is dimmed by means of elimination and the system of algorithm functioning passes into the search mode. That is reached by gradual elimination and each subsequent algorithm iteration assures better convergence of solutions. Each subsequent potential optimal solution can be designated N -best, where each iteration changes the action of the virtual gravitational force with which it affects all the remaining bodies in the space of solutions. Consequently, N -best is the time function having

the value No at the beginning of time t . In that way, at the beginning of the algorithm, all elements act with gravitational force on all remaining bodies, while the force decreases linearly with the reduction of the number of objects in the system and/or convergence of solutions to one point, which can also be described by means of gravitational collision of two bodies. The algorithm finalises the optimization, when only one body is still available with the greatest possible mass with which it acted on the remaining bodies and attracted them gravitationally. In that way, the Eq. 3 can be transformed and the result is:

$$F_a^n(t) = \sum_{b \in N - best, b \neq a} rand_b F_{ab}^n(t) \quad (12)$$

$F_a^n(t)$ – Force with stochastic characteristic in specific time t

$rand_b$ – Random number at interval $[0,1]$, it assures randomization

$F_{ab}^n(t)$ – Force which is caused by the b -th mass particle and acts on the a -th mass particle in time t

N -best – Set of first bodies with largest mass and highest fitness value

The pseudocode of the GSA algorithm is presented in Fig. 1.

```

1: Start GSA
2: Set-up search space
3: For each particle
4:   Initialize mass particle
5:   Start regression analysis module/with measured experimental data
6:   END
7:   Do
8:     For each particle/import measured experimental data
9:       Calculate capacity/fitness function of particle
10:      Capacity of particle > best capacity of particle (best (t)) → new best
11:      Evaluate worst values as worst (t)
12:     END
13:   For each particle
14:     Calculate velocity, masses and accelerations of mass particle
15:     Update particle position and velocity
16:   END
17: While → number N of iterations is reached
18: END GSA

```

Fig. 1 Pseudocode of the GSA algorithm

2.3 NSGA-II algorithm background

The NSGA-II (elitist Non-dominated Sorting Genetic Algorithm) is a genetic algorithm designed for multi-objective optimization [26]. The algorithm uses the crowding distance metric and non-dominated sorting as its main features. Fig. 2 presents the algorithm's pseudocode.

Roughly speaking, an algorithm consists of initialization, selection and re-combination. Selection and re-combination are the main algorithm steps (see Fig. 2). In the first step, the old population of parents and descendants is combined into a joint population of size $2n$. In the next step, the subjects from that combined population R_{t-1} are sorted front by front using non-dominated sorting. Subjects not dominated by any other subject go into the first front. The first i fronts still going whole into the new parent population P_t are written into that population. The first next front (front $i+1$) not going whole into the newly created population is sorted by using the crowding metric. The least crowded subjects from front $i+1$ are added to the new population P_t . The solutions have the largest range if the extreme subjects in the front are evaluated as best when sorting with the crowding metric. Evaluation of the remaining subjects in the front is performed by calculating the distance to its nearest neighbours.

-
- 1: Randomly generate and evaluate initial population of parents P_0 .
 - 2: Prepare empty initial population of descendants Q_0 .
 - 3: Setup $t = 0$.
 - 4: Until stopping criterion has been fulfilled, repeat:
 - 5: Setup $t = t + 1$.
 - 6: Unite two old populations of parents and descendants: $R_{t-1} = P_{t-1} \chi Q_{t-1}$.
 - 7: Perform non-dominated sorting on R_{t-1} and determine fronts $F_i, i = 1, 2, \dots$
 - 8: Prepare new empty population of parents P_t .
 - 9: Put into population P_t the first i front fronts still fitting whole into it.
 - 10: With the use of crowding distance metric sort front F_{i+1} no more fitting whole into population P_t
 - 11: Add the least crowded subjects from F_{i+1} to population P_t
 - 12: Generate population of descendants Q_t from parent population P_t by using tournament selection, crossover and mutations.
 - 13: Evaluate subjects from population of descendants Q_t .
-

Fig. 2 Pseudocode of the NSGA-II algorithm

When optimising k criteria, k subjects are first sorted according to growing evaluating values f_j for each criterion $j = 1, 2, \dots, k$. In the next step for each subject i , the distance between its two neighbours u and v is calculated according to the following equation:

$$d_j(i) = \frac{f_j(u) - f_j(v)}{f_j^{max} - f_j^{min}} \quad (13)$$

where f_j^{max} and f_j^{min} are the maximum and the minimum values of the j -th criterion, respectively.

The following must apply:

$$f_j(u) \leq f_j(i) \leq f_j(v) \quad (14)$$

As already mentioned, the highest possible distance is assigned to the two extreme subjects (with respect to the j criterion). For all remaining subjects, the crowding metric for subject i is equal to the sum of those distances according to all criteria:

$$c(i) = \sum_{j=1}^k d_j(i) \quad (15)$$

The parent population P_t is obtained in the manner described above. In the next steps, the descendant population is generated from that population by using tournament selection, crossover and mutation. Out of two random subjects, the subject classified into the front with lower consecutive value, wins in the selection. If two subjects taking part in selection come from an equal front, then the subject better evaluated by the crowding metric is chosen. The descendant population is designated Q_t . Each subject from the descendant population is evaluated in the last step of the NSGA-II algorithm. That estimate is used in the next generation for non-dominated sorting of subjects. The algorithm is repeated until the maximum number of generations has been reached or until another stopping criterion has been reached.

3. Results and discussion

3.1 Modelling of turning process by the Gravitational Search Algorithm

For processing experimental data the GSA method was used for the purposes of turning system modelling. For the modelling process the following control parameters of the GSA optimization algorithm were used:

- Number of mass bodies: 210
- Number of iterations: 1000
- Size of mass body (dimension): 10
- Power of Euclidean distance between mass bodies: 1

- Power factor α : 20
- Gravitational constant G_0 : 100

In particular, the two random factors α and G_0 must be pointed out. Both of them are inter-related and form the refreshment of iteration within the scope of the gravitational constant, where the G_0 value is the initial value of the gravitational constant and is changed exponentially on the basis of factor α which, in combination with the ratio of consecutive iteration and total number of iterations, affects the contingency of gravitational constant values additionally during each iteration. The mass particle size and/or the properties of solutions written in the mass particle are represented with a dimension which also defines the number of coefficients in the polynomial representing the model of the individual machining operation. A prerequisite for successful execution of modelling was correct selection of the polynomial optimized by means of GSA. Eq. 16 shows the form of a ten-coefficient polynomial which proved to be a good choice for prediction of output variables.

$$f(x_1, x_2, x_3) = k_1 + k_2 \cdot x_1 + k_3 \cdot x_2 + k_4 \cdot x_3 + k_5 \cdot x_1 \cdot x_2 + k_6 \cdot x_1 \cdot x_3 + k_7 \cdot x_2 \cdot x_3 + k_8 \cdot x_1 \cdot x_1 + k_9 \cdot x_2 \cdot x_2 + k_{10} \cdot x_3 \cdot x_3 \quad (16)$$

where x_1 is cutting speed v_c , x_2 is feed rate f , and x_3 is depth of cut a_p .

The polynomial represents the core of the system round which the optimization process is designed. Input and output experimental values are inserted into the polynomial, while the optimization process optimises the coefficients in such a manner that the predictions given by the polynomial model are as near the experimental values as possible. Out of all the experimental measurements performed, fifteen measurements were chosen at random for the process of optimising of the polynomial coefficients, while the remaining 5 measurements were used for result verification. The results of optimization are polynomial models for the calculation of cutting force, surface roughness and tool wear (Eq. 17-19).

$$F_c = 523.66 - 1.89 \cdot x_1 - 813.18 \cdot x_2 + 48.88 \cdot x_3 + 3.05 \cdot x_1 \cdot x_2 + 0.29 \cdot x_1 \cdot x_3 + 1439.13 \cdot x_2 \cdot x_3 + 0.0014 \cdot x_1 \cdot x_1 - 1257.73 \cdot x_2 \cdot x_2 - 43.70 \cdot x_3 \cdot x_3 \quad (17)$$

$$R_a = 3.74 - 0.01 \cdot x_1 - 4.25 \cdot x_2 + 1.53 \cdot x_3 - 0.01 \cdot x_1 \cdot x_2 - 0.001 \cdot x_1 \cdot x_3 - 1.03 \cdot x_2 \cdot x_3 + 0.00002 \cdot x_1 \cdot x_1 + 68.81 \cdot x_2 \cdot x_2 - 0.31 \cdot x_3 \cdot x_3 \quad (18)$$

$$T = 294.77 - 0.95 \cdot x_1 - 242.17 \cdot x_2 - 4.60 \cdot x_3 + 0.24 \cdot x_1 \cdot x_2 + 0.02 \cdot x_1 \cdot x_3 - 9.20 \cdot x_2 \cdot x_3 + 0.0007 \cdot x_1 \cdot x_1 + 287.75 \cdot x_2 \cdot x_2 - 5.37 \cdot x_3 \cdot x_3 \quad (19)$$

Testing of the developed models was recorded as a comparison of experimental values, predictions and percent deviation; experimental value \rightarrow prediction \rightarrow percent deviation:

- Principal cutting force (Eq. 17)
Lowest deviation with measured value: 550.848 N \rightarrow 557.877 N \rightarrow 1.28 %
Highest deviation with measured value: 174.024 N \rightarrow 189.579 N \rightarrow 8.94 %
Average deviation of all twenty measurements with ten predictions: 3.57 %
- Surface roughness (Eq. 18)
Lowest deviation with measured value: 2.32 μm \rightarrow 2.31 μm \rightarrow 0.43 %
Highest deviation with measured value: 1.31 μm \rightarrow 1.48 μm \rightarrow 13.30 %
Average deviation of all twenty measurements with ten predictions: 4.02 %
- Tool life (Eq. 19)
Lowest deviation with measured value: 10.93 min \rightarrow 10.77 min \rightarrow 1.49 %
Highest deviation with measured value: 28.43 min \rightarrow 26.45 min \rightarrow 6.97 %
Average deviation of all twenty measurements with ten predictions: 4.50 %

In this case, the deviation was within the acceptable ten percent. In any case, the lowest possible deviations are desirable, but the size of deviations depends primarily on the number of

measurements. The experiment assured only 20 results of finishing machining, therefore, the final average of results can be assumed to be acceptable.

3.2 Multi-objective optimization by the NSGA-II algorithm

On the basis of models created by means of the GSA algorithm and outlined in the previous subsection, in this paragraph, a set of optimal solutions for the chosen machining operation will be searched for. In our case, the resulting models for force, roughness and tool resistance represent three objective (criteria) functions having to satisfy also the limitations stated below. In this research, the aim of multi-objective optimization by the NSGA-II method was to determine such a combination of independent input variables (i.e. cutting force, feed rate and cut depth) at chosen intervals, that optimal values of criteria functions will be reached and the limitations prescribed satisfied. On the basis of theoretical calculations of limitations, and on the basis of experimental data, the optimization limitations of cutting force F_c , roughness R_a and tool resistance T were determined:

- $F_c \leq 450.0$ N
- $1.0 \leq R_a \leq 1.6$ μm
- $15.0 \leq T \leq 20.0$ min

According to the theoretical calculations and experimental data, the intervals of independent variables of cutting speed v_c , feed rate f and cut depth a_p were also prepared:

- $366.0 < v_c < 540.0$ $\frac{\text{m}}{\text{min}}$
- $0.10 < f < 0.18$ $\frac{\text{mm}}{\text{rev}}$
- $0.2 < a_p < 1.2$ mm

The results of the fine turning operation are presented by means of a Pareto optimal front. Minimal surface roughness has priority as fine machining is aimed at securing final shapes and dimensions. Fig. 3 shows the results of optimization developed by using the GSA algorithm.

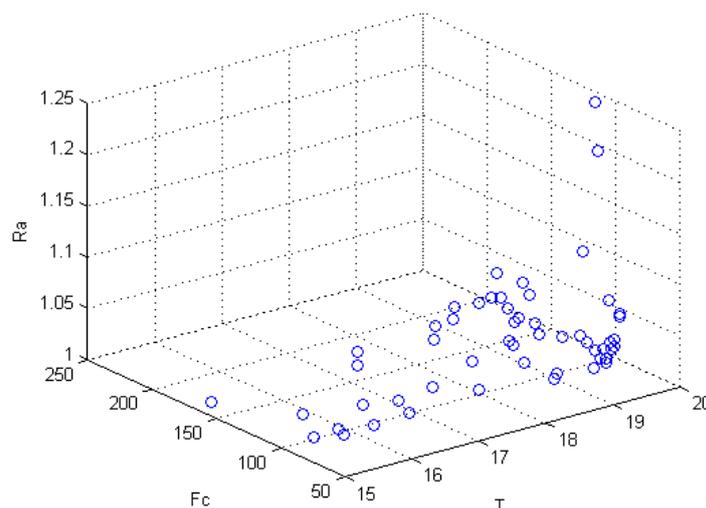


Fig. 3 Pareto optimal front

The Pareto optimal front contains 53 equivalent solutions. The most successful optimal combination is proposed, where:

Independent parameters:

$$x_1 = v_c = 439.9 \text{ m/min}$$

$$x_2 = f = 0.11 \text{ mm/rev}$$

$$x_3 = a_p = 0.62 \text{ mm}$$

Dependent parameters:

$$F_c = 201.36 \text{ N}$$

$$R_a = 1 \mu\text{m}$$

$$T = 19.53 \text{ min}$$

4. Conclusion

In the first step of the research an intelligent system was devised, ensuring effective modelling of the CNC machining process by the use of a GSA. The relevant GSA was developed, with an added regression analysis module allowing wider algorithm functioning for modelling machining operations with a training base. The developed system is useful in all standard machining processes, e.g. turning, milling, drilling, but, for the purpose of this research, the models of the CNC turning process have been developed with the use of experimental measurements relating to the turning process, and their accuracy tested. Results have confirmed that the CNC machining process can be modelled effectively by using a GSA.

In the second step, the models of CNC turning were used for multi-objective optimization of machining parameters by the NSGA-II algorithm. The result of multi-objective optimization of machining parameters was the Pareto front of optimal solutions, out of which the solution most suitable in practice can be selected for the chosen machining operation. For example, if fine machining parameters are looked for, the solution, giving priority to minimum surface roughness, will be selected, as the aim of fine machining is securing final shapes and dimensions.

Prospective further researches could focus on the enhancement of the training and testing base with new measurements, and on repetition of the whole experiment with other materials and tools. The use of appropriate machine tools and sensors could also ensure current measuring of cutting force which would be transmitted via computer communication into the base on an intermediate server or computer.

Acknowledgement

We would like to thank the Slovenian Research Agency (ARRS) for the financial support for execution of the presented research.

References

- [1] Hrelja, M., Klančnik 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](https://doi.org/10.2507/IJSIMM13(1)3.248).
- [2] Lestan, Z., Klančnik, S., Balic, J., Brezocnik, M. (2015). Modeling and design of experiments of laser cladding process by genetic programming and nondominated sorting, *Materials and Manufacturing Processes*, Vol. 30, No. 4, 458-463, [doi: 10.1080/10426914.2014.973586](https://doi.org/10.1080/10426914.2014.973586).
- [3] Ficko, M., Palcic, I. (2013). Designing a layout using the modified triangle method, and genetic algorithms, *International Journal of Simulation Modelling*, Vol. 12, No. 3, 237-251, [doi: 10.2507/IJSIMM12\(4\)3.244](https://doi.org/10.2507/IJSIMM12(4)3.244).
- [4] Alberti, M., Ciurana, J., Casadesús, M. (2005). A system for optimising cutting parameters when planning milling operations in high-speed machining, *Journal of Materials Processing Technology*, Vol. 168, No. 1, 25-35, [doi: 10.1016/j.jmatprotec.2004.09.092](https://doi.org/10.1016/j.jmatprotec.2004.09.092).
- [5] Bharathi Raja, S., Baskar, N. (2011). Particle swarm optimization technique for determining optimal machining parameters of different work piece materials in turning operation, *The International Journal of Advanced Manufacturing Technology*, Vol. 54, No. 5, 445-463, [doi: 10.1007/s00170-010-2958-y](https://doi.org/10.1007/s00170-010-2958-y).
- [6] Zain, A.M., Haron, H., Sharif, S. (2011). Optimization of process parameters in the abrasive waterjet machining using integrated SA-GA, *Applied Soft Computing*, Vol. 11, No. 8, 5350-5359, <https://doi.org/10.1016/j.asoc.2011.05.024>.
- [7] 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](https://doi.org/10.1016/0278-6125(91)90004-L).
- [8] 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](https://doi.org/10.1016/S0007-8506(07)60503-4).
- [9] Chua, M.S., Rahman, M., Wong, Y.S., Loh, H.T. (1993). Determination of optimal cutting conditions using design of experiments and optimization techniques, *International Journal of Machine Tools and Manufacture*, Vol. 33, No. 2, 297-305, [doi: 10.1016/0890-6955\(93\)90081-5](https://doi.org/10.1016/0890-6955(93)90081-5).
- [10] Yusup, N., Zain, A.M., Hashim, S.Z.M. (2012). Evolutionary techniques in optimizing machining parameters: Review and recent applications (2007-2011), *Expert Systems with Applications*, Vol. 39, No. 10, 9909-9927, [doi: 10.1016/j.eswa.2012.02.109](https://doi.org/10.1016/j.eswa.2012.02.109).
- [11] Zain, A.M., Haron, H., Sharif, S. (2010). Application of GA to optimize cutting conditions for minimizing surface roughness in end milling machining process, *Expert Systems with Applications*, Vol. 37, No. 6, 4650-4659, [doi: 10.1016/j.eswa.2009.12.043](https://doi.org/10.1016/j.eswa.2009.12.043).

- [12] D'Addona, D.M., Teti, R. (2013). Genetic algorithm-based optimization of cutting parameters in turning processes, *Procedia CIRP*, Vol. 7, 323-328, doi: [10.1016/j.procir.2013.05.055](https://doi.org/10.1016/j.procir.2013.05.055).
- [13] Khaider, B., Asma, T. (2016). Hard turning behavior improvement using NSGA-II and PSO-NN hybrid model, *The International Journal of Advanced Manufacturing Technology*, Vol. 86, No. 9, 3527-3546, doi: [10.1007/s00170-016-8479-6](https://doi.org/10.1007/s00170-016-8479-6).
- [14] Krimpenis, A., Vosniakos, G.-C. (2009). Rough milling optimisation for parts with sculptured surfaces using genetic algorithms in a Stackelberg game, *Journal of Intelligent Manufacturing*, Vol. 20, No. 4, 447-461, doi: [10.1007/s10845-008-0147-8](https://doi.org/10.1007/s10845-008-0147-8).
- [15] Suresh, P.V.S., Venkateswara Rao, P., Deshmukh, S.G. (2002). A genetic algorithmic approach for optimization of surface roughness prediction model, *International Journal of Machine Tools and Manufacture*, Vol. 42, No. 6, 675-680, doi: [10.1016/S0890-6955\(02\)00005-6](https://doi.org/10.1016/S0890-6955(02)00005-6).
- [16] Mizugaki, Y., Hao, M., Sakamoto, M., Makino, H. (1994). Optimal tool selection based on genetic algorithm in a geometric cutting simulation, *CIRP Annals – Manufacturing Technology*, Vol. 43, No. 1, 433-436, doi: [doi:10.1016/S0007-8506\(07\)62247-1](https://doi.org/10.1016/S0007-8506(07)62247-1).
- [17] Jurkovic, Z., Cukor, G., Brezocnik, M., Brajkovic, T. (2016). A comparison of machine learning methods for cutting parameters prediction in high speed turning process, *Journal of Intelligent Manufacturing*, (in Press), doi: [10.1007/s10845-016-1206-1](https://doi.org/10.1007/s10845-016-1206-1).
- [18] Cukor, G., Jurkovic, Z. (2010). Optimization of turning using evolutionary algorithms, *Engineering Review*, Vol. 30, No. 2, 1-10.
- [19] Mocnik, D., Paulic, M., Klancnik, S., Balic, J. (2014). Prediction of dimensional deviation of workpiece using regression, ANN and PSO models in turning operation, *Tehnički vjesnik – Technical Gazette*, Vol. 21, No. 1, 55-62.
- [20] El-Mounayri, H., Kishawy, H., Briceno, J. (2005). Optimization of CNC ball end milling: A neural network-based model, *Journal of Materials Processing Technology*, Vol. 166, No. 1, 50-62, doi: [10.1016/j.jmatprotec.2004.07.097](https://doi.org/10.1016/j.jmatprotec.2004.07.097).
- [21] Muñoz-Escalona, P., Maropoulos, P.G. (2010). Artificial neural networks for surface roughness prediction when face milling Al 7075-T7351, *Journal of Materials Engineering and Performance*, Vol. 19, No. 2, 185-193, doi: [10.1007/s11665-009-9452-4](https://doi.org/10.1007/s11665-009-9452-4).
- [22] Zhang, Y., Gong, D.-W., Ding, Z.-H. (2011). Handling multi-objective optimization problems with a multi-swarm cooperative particle swarm optimizer, *Expert Systems with Applications*, Vol. 38, No. 11, 13933-13941, doi: [10.1016/j.eswa.2011.04.200](https://doi.org/10.1016/j.eswa.2011.04.200).
- [23] Chiang, F., Braun, R., Agbinya, J.I. (2007). Self-configuration of network services with biologically inspired learning and adaptation, *Journal of Network and Systems Management*, Vol. 15, No. 1, 87-87, doi: [10.1007/s10922-006-9056-3](https://doi.org/10.1007/s10922-006-9056-3).
- [24] 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](https://doi.org/10.1016/j.ins.2009.03.004).
- [25] Jurkovic, Z. (2007). *Modeliranje i optimizacija parametara obrade primjenom evolucijskih algoritama kod inteligentnih obradnih sustava*, PhD thesis, University of Rijeka, Rijeka, Croatia.
- [26] Wang, L., Wang, T.-G., Luo, Y. (2011). Improved non-dominated sorting genetic algorithm (NSGA)-II in multi-objective optimization studies of wind turbine blades, *Applied Mathematics and Mechanics*, Vol. 32, No. 6, 739-748, doi: [10.1007/s10483-011-1453-x](https://doi.org/10.1007/s10483-011-1453-x).