Optimization of cutting parameters for machining time in turning process

A R Mavliutov, E G Zlotnikov

Saint-Petersburg Mining University, 2, 21st line, Saint-Petersburg, 199106, Russia

E-mail: guiwter94@gmail.com

Abstract. This paper describes the most effective methods for nonlinear constraint optimization of cutting parameters in the turning process. Among them are Linearization Programming Method with Dual-Simplex algorithm, Interior Point method, and Augmented Lagrangian Genetic Algorithm (ALGA). Every each of them is tested on an actual example – the minimization of production rate in turning process. The computation was conducted in the MATLAB environment. The comparative results obtained from the application of these methods show: The optimal value of the linearized objective and the original function are the same. ALGA gives sufficiently accurate values, however, when the algorithm uses the Hybrid function with Interior Point algorithm, the resulted values have the maximal accuracy.

1. Introduction

In every modern machining operations, it is necessary to select the optimal cutting parameters that will produce the cheapest product with the maximum efficiency and the minimum of production time. To achieve this goal, firstly, the problem of cutting process must be modeled. A typical model of turning appears in these works [1-3]. Such a problem can be solved by traditional methods (linearization programming, gradient methods, penalty methods, quasi-Newton methods etc.) or modern heuristic methods (Artificial Neuron Networks, genetic algorithm, Swarm Intelligence, etc.).

When encountering a nonlinear problem, the first approach in solving it is by means of linearization. However, linearization of nonlinear systems is generally limited in application to the certain class of problems [4]. Some linearized models cannot adequately present the equivalent function with the same global minimum as the original function. A proper research must be conducted in order to determine the validity of such linearization.

The most reliable method for solving a nonlinear constrained problem is interior point method because it is a large-scale method and can handle various problems (including large problems) quickly [5]. Many advantages of interior point methods indicate of its efficiency [6].

A further research of genetic algorithm proposes comparison of genetic algorithm with other methods of optimization [7, 8].

But first, the objective function and all the constraints must be established and then applied to an actual example of turning process.

2. Determination of the optimization criterion and the technological constraints

One of the main goals of an effective manufacturing is an increase of the production rate. The production rate of a single operation can be determined through the machining time.
For simplicity, the part of the machining time—the cutting time that depends on optimizing variables is chosen [9]. The objective function - \( t_m \), optimization parameters - \( n \), \( s \) and \( t \).

\[
t_m = f(n,s,t) = \frac{1}{n \cdot s \cdot t} \left( \frac{L \cdot h}{1-m} \right),
\]

(1)

where \( n \) – rotational speed of the workpiece, rpm; \( s \) – feed rate, mm/turn; \( t \) – depth of cut, mm; \( L \) – the path length of the cutting tool in the feed direction, \( h \) – machining allowance; \( m \) – exponent for tested tool life conditions.

The equation (1) can be transformed into:

\[
t_m = f(n,s,t) = \frac{C_0}{n \cdot s \cdot t},
\]

(2)

where \( C_0 = \frac{L \cdot h}{1-m} \) is a constant parameter.

Optimization of cutting parameters also depends on the right choice of technical constraints – \( i \) which determine the domain of the function. The formulas are taken from [10]:

2.1 Constraint on the cutting properties of the tool

The cutting speed for longitudinal turning must not exceed the maximum limit cutting speed for a given tool life:

\[
n \cdot s^{y_v} \cdot t^{z_p} \leq \frac{1000 \cdot C_v \cdot K_u}{\pi \cdot D \cdot T^m},
\]

(3)

where \( C_v \) – coefficient characterizing processing conditions; \( D \) – machined diameter; \( T \) – tool life; \( K_u \) – correction coefficient that takes into account the change in real processing conditions apart from those for which the coefficient \( C_u \) was tested; \( y_v \), \( x_v \), \( m \) – exponents characterizing the influence of \( s \), \( t \), \( T \) on the cutting speed.

Since the expression on the right does not contain any variables, it can be simplified:

\[
n \cdot s^{y_v} \cdot t^{z_p} \leq C_1,
\]

(4)

where \( C_1 = \frac{1000 \cdot C_v \cdot K_u}{\pi \cdot D \cdot T^m} \).

2.2 Constraint on the machine power

The effective power during turning should not exceed the power supplied to the spindle of the machine:

\[
n^{v_p+1} \cdot s^{y_p} \cdot t^{z_p} \leq \frac{1020 \cdot 60 \cdot N_{mp} \cdot \eta}{10 \cdot C_{p1} \cdot K_{p1} \left( \frac{1000}{\pi \cdot D} \right)^{v_p+1}},
\]

(5)

where \( N_{mp} \) – machine power, kW; \( \eta \) – power efficiency; \( C_{p1} \) – coefficient characterizing a group of processed materials; \( K_{p1} \) – the general correction factor; \( y_p \), \( x_p \) and \( z_p +1 \) are the exponents characterizing the influence of \( s \), \( t \) and \( \nu = \frac{\pi \cdot D \cdot n}{1000} \) – cutting speed on the tangential component of cutting force.

Since the expression on the right does not contain any variables, it can be simplified:

\[
n^{v_p+1} \cdot s^{y_p} \cdot t^{z_p} \leq C_2,
\]

(6)

where \( C_2 = \frac{1020 \cdot 60 \cdot N_{mp} \cdot \eta}{10 \cdot C_{p1} \cdot K_{p1} \left( \frac{1000}{\pi \cdot D} \right)^{y_p+1}} \).
2.3 Constraint on the working accuracy

For surface machining with an error not exceeding the tolerance for the diametrical size of the workpiece, it is necessary to satisfy the following inequality:

\[ n^{y_3} \cdot S^{y_3} \cdot t^{y_3} \leq \frac{k_3 \cdot \delta_P \cdot \left( \frac{1000}{\pi \cdot D} \right)^{y_3} \cdot \alpha_3}{2 \cdot C_{P_3} \cdot \left[ \frac{1}{J_w} + \frac{1}{J_m} + \frac{1}{J_t} \right]} \]  

(7)

where \( \frac{1}{J_w}, \frac{1}{J_m}, \frac{1}{J_t} \) are mechanical compliance of workpiece, machine and tool accordingly, mm/N; \( k_3 \) – a coefficient that indicates the part of the tolerance relating to the error which is caused by the deformation of the workpiece, the machine and the tool; \( \delta_P \) – allowance on the size being processed, mm; \( C_{P_3} \) – coefficient reflecting the influence of the processing conditions on the cutting force component \( P_y \); \( y_p, x_p \) and \( z_p \) are the exponents characterizing the influence of \( s, t \) and \( v = \frac{\pi \cdot D \cdot n}{1000} \).

Since the expression on the right does not contain any variables, it can be simplified:

\[ n^{y_3} \cdot S^{y_3} \cdot t^{y_3} \leq C_3, \]  

(8)

where \( C_3 = \frac{k_3 \cdot \delta_P \cdot \left( \frac{1000}{\pi \cdot D} \right)^{y_3} \cdot \alpha_3}{2 \cdot C_{P_3} \cdot \left[ \frac{1}{J_w} + \frac{1}{J_m} + \frac{1}{J_t} \right]} \).

2.4 Constraint on the surface roughness

Part processing with an allowable surface roughness can be carried out under the condition:

\[ S \cdot t^{y_3} \leq \frac{C_s \cdot R^{y_3} \cdot r^{y_3}}{\phi^{y_3} \cdot \phi_3^{y_3}}, \]  

(9)

where \( \phi, \phi_3 \) are the primary and the secondary angles in the plane, rad; \( r \) is the corner radius of the main cutting edge, mm; \( C_s, x_p, y_p, z_p \) and \( u_p \) are experimental coefficients depending on the material being processed.

Since the expression on the right does not contain any variables, it can be simplified:

\[ S \cdot t^{y_3} \leq C_4, \]  

(10)

where \( C_4 = \frac{C_s \cdot R^{y_3} \cdot r^{y_3}}{\phi^{y_3} \cdot \phi_3^{y_3}} \).

2.5 Bound constraints related to the kinematic capabilities of the lathe

The rotational speed of the workpiece and the feed of the tool must be limited according to the values set in the machine's passport:

\[ n_{\min} \leq n \leq n_{\max}; \]  

(11)

\[ s_{\min} \leq s \leq s_{\max}. \]  

(12)

To ensure the specified accuracy of the detail, cutting depth \( t \) cannot be less than the value determined for each tool and each processed material. On the other hand, the cutting depth cannot be greater than the allowance for processing:

\[ t_{\min} \leq t \leq t_{\max}. \]  

(13)
3. Optimization of cutting parameters with Linearization Programming Method

The method and the solving algorithm for optimization of machining time in the turning process first appeared in the monograph of S. V. Grubyi [11]. By getting logarithm of the inequalities (3) – (13) and the objective function (2), the system of linear inequalities is obtained. The objective function takes the form:

\[ f^0 = C_0 - x_1 - x_2 - x_3, \]  

(14)

where \( x_1 = \ln n \); \( x_2 = \ln s \); \( x_3 = \ln t \).

Linear constraints:

\[
\begin{align*}
&x_1 + y_p \cdot x_3 + x_4 \cdot x_3 \leq C_1; \\
&(1 + z_p) \cdot x_1 + y_p \cdot x_2 + x_p \cdot x_3 \leq C_2; \\
&z_p \cdot x_1 + y_p \cdot x_2 + x_p \cdot x_3 \leq C_3; \\
&x_2 + x_R \cdot x_3 \leq C_4,
\end{align*}
\]

(15)

where \( C_1 = \ln C_1 \); \( C_2 = \ln C_2 \); \( C_3 = \ln C_3 \).

Bound constraints:

\[
\begin{align*}
x_{1_{\text{max}}} & \leq x_1 \leq x_{1_{\text{min}}}; \\
x_{2_{\text{max}}} & \leq x_2 \leq x_{2_{\text{min}}}; \\
x_{3_{\text{max}}} & \leq x_3 \leq x_{3_{\text{min}}},
\end{align*}
\]

(16)

where \( x_{1_{\text{max}}} = \ln x_{1_{\text{max}}} \); \( x_{1_{\text{min}}} = \ln x_{1_{\text{min}}} \); \( x_{2_{\text{max}}} = \ln x_{2_{\text{max}}} \); \( x_{2_{\text{min}}} = \ln x_{2_{\text{min}}} \); \( x_{3_{\text{max}}} = \ln x_{3_{\text{max}}} \); \( x_{3_{\text{min}}} = \ln x_{3_{\text{min}}} \).

For the reason that constants change only the absolute value of a function and disappear with the formation of an extremum, the objective function (14) converts to:

\[ F = -x_1 - x_2 - x_3. \]

(17)

One of the possible ways to solve this problem by means of linearization programming is the dual simplex method [12]. In the dual simplex method, the dual problem is being solved instead of the initial problem with the simplex method, since the solution to the primal problem is the same as the solution to the dual problem in the optimal point. In the simplex method, the system of linear inequalities determines a polytope as a region with feasible solution. At first, a starting extreme point has to be found. If this point is the optimal solution, then the problem is solved, otherwise, the solving process goes to the next point, where the value of the objective function is better. The algorithm uses a finite number of steps because there are a finite number of extreme points.

The left side of the system (15) can be specified as \( i \)-by-\( j \) matrix of coefficients, where \( i \) is the number of inequalities, \( j \) is the number of variables: \( A = \begin{bmatrix} 1 & y_p & x_p \\ (1 + z_p) & y_p & x_p \\ z_p & y_p & x_p \\ 0 & 1 & x \end{bmatrix} \); the right side of the system (15) can be specified as vector: \( \mathbf{C} = [C_1; C_2; C_3; C_4] \); the objective function (17) as \( \mathbf{F} = [-1; -1; -1] \); bound constraints (16): \( \mathbf{lb} = [x_{1_{\text{min}}}; x_{2_{\text{min}}}; x_{3_{\text{min}}}]; \) and \( \mathbf{ub} = [x_{1_{\text{max}}}; x_{2_{\text{max}}}; x_{3_{\text{max}}}]. \)

To find the minimum of this problem, MATLAB function “linprog” is used which returns the vector of the objective function with optimal coordinates – \( x \):

\[ x = \text{linprog}(\mathbf{F}, \mathbf{A}, \mathbf{C}, \mathbf{lb}, \mathbf{ub}, \text{options}), \]

(18)

where the brackets[ ] are intentionally left empty because the equality constraints are absent. In options must be specified the algorithm being used, which is dual-simplex.

For example, a typical case of the turning process is taken. The part is processed on CNC lathe
EMCO CONCEPT TURN 250. The data for the workpiece: material – steel 40X, workpiece surface condition – pre-machined, type of process – finishing and semi-finishing, Ra value – 5 µm, diameter – 60 mm, length – 180 mm. The compelling interest presents the optimization of cutting parameters in machining non-ferrous alloys [13] with the usage of a ceramic cutting tool because of the accuracy of the machined surface, low parameter Ra and cheap cost of the material [14, 15], as well as optimization of cutting isotropic materials [16].

In the example the workpiece is being machined to the diameter 55 mm on the length of 150 mm with Ra = 2.5 µm, tolerance grade – 8.

The tool has the following characteristics: cutting plate material – T15K6, shank tool material – steel 15, angles in the plane ϕ, φ₁ are equal to 45°, shank size – 12×12 mm, overhang – 22 mm.

The lathe used is CONCEPT TURN 250 with spindle torque 35 Nm and main spindle speed up to 6000 rpm, main spindle speed range 60 – 6300 rpm, and feed rate range 0.05-2.8 mm/turn.

Constants and exponents for equestions (2) – (10) are taken from the reference literature [17], the parameters for (11) – (13) are taken from the technical data. Their values are presented in the Table 1:

| Parameter | Value |
|-----------|-------|
| C₀ | 280 |
| K₁ | 1.09 |
| x₀ | 0.9 |
| z₀ | 0.15 |
| T | 45 min |
| x₀ | 1 |
| y₀ | 0.6 |
| u₀ | 0.3 |
| K₁ | 1.04 |
| y₀ | 0.75 |
| z₀ | -0.3 |
| r | 0.4 |
| x₀ | 0.15 |
| y₀ | -0.15 |
| k₁ | 0.75 |
| nₘₐₙ | 60 rpm |
| y₀ | 0.45 |
| δₚ | 0.054 mm/N |
| φ | 45° |
| nₘₐₓ | 6300 rpm |
| m | 0.2 |
| jₙ | 0.034·10⁻⁵ mm/N |
| φ₁ | 45° |
| sₘᵢₙ | 0.05 mm/turn |
| Nₘₚ | 11 kW |
| jₘ | -1.842·10⁻³ mm/N |
| C₁ | 0.17 |
| Cₚ | 300 |
| Cₚ | 243 |
| y₀ | 0.6 |
| tₘₐₓ | 2.5 mm |

Coordinates of the optimal point: \( x = [0.6726; -0.7092; 0.91] \), which corresponds to the cutting parameters: \( n = 864.888 \text{ rpm} \), \( s = 0.492 \text{ mm/turn} \), \( t = 2.4843 \text{ mm} \). Function value approximately is 0.45 min or 27 sec.

4. Interior Point Method

The idea of all penalty methods is to transform the objective function with constraints into the corresponding approximate function without constraints that can be solved by iterative process:

\[
z = f(x) \rightarrow Z = f(x) + P(x),
\]

where \( P(x) \) is a barrier function (or penalty function).

The penalty function increases the approximate function \( Z \) when the constraints are not met, in this case the minimum of \( Z \) would be inside the constraint area.

The problem can be formulated as:

\[
\min f(x);
\]

\[
s.t. h(x) = 0;
\]

\[
g(x) \leq b,
\]

where \( f : \mathbb{R}^n \rightarrow \mathbb{R}, h : \mathbb{R}^n \rightarrow \mathbb{R}^l, g : \mathbb{R}^n \rightarrow \mathbb{R}^m \) are twice continuously differentiable functions.

Corresponding approximate problem:
\[
\begin{align*}
\min_{x,s} Z(x,s) & \equiv f(x) - \mu \sum_{i=1}^{m} \ln s_i; \\
\text{s.t.} \quad & h(x) = 0; \\
& g(x) + s = 0,
\end{align*}
\] (21)

where the barrier parameter \( \mu \) and the slack variable \( s \) are positive.

The whole process is described in [18]. The number of inequality constraints is the same as the number of slack variables. At each iteration from starting point with decreasing of \( \mu \), the algorithm converges to the optimal solution point of the function \( Z(x,s) \). The minimum of approximate problem will obviously satisfy the constraints of the original problem.

The whole process consists of two steps. In the first step (direct step), the barrier function is set up to solve for a search direction in \( x, s \).

This step tries to solve the Karush-Kuhn-Tucker (KKT) conditions:

\[
\begin{align*}
\frac{\partial Z}{\partial x_k} &= 0, \quad k = 1, 2, \ldots, n; \\
\lambda_i g_i(x) &= 0, \quad i = 1, \ldots, m; \\
\lambda_i &\geq 0, \quad i = 1, \ldots, m; \\
\lambda_j h_j(x_1, x_2, \ldots, x_n) &= 0, \quad j = 1, \ldots, p; \\
\lambda_j &\geq 0, \quad j = 1, \ldots, p,
\end{align*}
\] (22)

where \( \lambda_i, \lambda_j \) are Lagrange multipliers.

The Hessian \( H \) of the merit function defines the step:

\[
H = \nabla^2 f(x) + \sum_i \lambda_i \nabla^2 g_i(x) + \sum_j \lambda_j \nabla^2 h_j(x).
\] (23)

The Hessian then transforms into a matrix form and can be solved by any quasi-Newton methods. If the computation fails, the algorithm uses conjugate gradient step and again attempts to solve KKT conditions. The step this time is determined to minimize a quadratic approximation to the problem \( Z(x,s) \) keeping the solution in the trust region.

After finding the search direction, the appropriate step size needs to be found. IPM (Interior Point Method) uses a decrease in merit function approach, where the resulted function is the combination of the objective function with the absolute value of the constraint violation times \( \nu \) to determine if it is a better step or not.

\[
Z(x,s) + \nu \|h(x), g(x) + s\|,
\] (24)

where parameter \( \nu \) forces the solution towards feasibility.

The convergence criteria are KKT conditions that are satisfied with a certain tolerance.

In the example, MATLAB function “fmincon” is used which returns the solution vector of the problem (21) and the value of the objective function:

\[
[x, fval] = \text{fmincon}(\text{fun}, x0, A, b, Aeq, beq, lb, ub, nonlcon, options),
\] (25)

where “fun” is the objective function; \( x0 \) – starting point for the minimization algorithm; \( A, b \) – linear inequalities, which is set as empty because the linear inequalities are absent in the optimization problem; \( \text{Aeq, beq} \) – linear equalities which are also stay empty; \( \text{lb, ub} \) – bound constraints: (11) – (13); “nonlcon” – nonlinear constraints, the user function “@nonlcon” is created which contains the nonlinear inequalities: (3) – (10); in options, the current algorithm must be chosen.

The uncertainty of this method may lie in coordinates for starting point. Here, the lower bounds may be used as the vector for starting point. However, the starting point may even be not in the bound constraints, the only difference will be the number of iterations.
The returned values are $x = [864.7348; 0.4927; 2.5]$ where $n = 864.7348$ rpm; $s = 0.4927$ mm/turn; $t = 2.5$ mm; $fval = 0.4459$ min. The constrained tolerance $-1e-6$.

Lambda for inequality constraints $\lambda = \begin{bmatrix} 0.0006 \\ 0 \\ 0 \\ 0.4459 \end{bmatrix}$ satisfies the conditions (22). The Hessian $H(x) = \begin{bmatrix} 0.0000 & 0.0015 & 0.0002 \\ 0.0015 & 3.6614 & 0.3815 \\ 0.0002 & 0.3815 & 0.4555 \end{bmatrix}$ meaning that the sufficient conditions for the minimum are satisfied.

5. Augmented Lagrangian Genetic Algorithm

Among heuristics Augmented Lagrangian Genetic Algorithm (ALGA) is considered for constrained optimization [5].

The main idea of the method lies in the modulation of the evolutionary process. The population is created which consists of individuals with unique genotype (a set of vectors). Every set of genes is used to evaluate the fitness function (the objective function). As a result, every genotype is associated with a certain phenotype which determines their fitness. This population is affected by mutation, selection, and crossover. However, there are other mechanisms like migration or cataclysm, but they are unnecessary for solving the problem.

After obtaining a necessary amount of individuals in the population follows the selection of the most fitted to the next generation. In order to diversify the population, a mechanism of mutation is added which consists of random changes in genes of some individuals. The crossover function is used to produce individuals with better genes. In the simplest case, two individuals are selected, thus, producing a child with half the genes from the first parent and the other half from the second one. After application of these mechanisms, the next generation is formed. The process continues until the search algorithm meets the stopping criteria (maximum number of generations, time limit, fitness limit, function tolerance, etc.).

For the problem, MATLAB Optimization Tool is used.

As fitness function, the objective function (2) is assigned; the nonlinear constraints (3) – (10) lie in the @nlcon file that was previously used in IPM; bounds are (11) – (13).

The specifications in the options:

1) **Population size** – the number of individuals in every generation. Here, 60 individuals are taken.
2) **Creation function** is **Nonlinear feasible population** since there are nonlinear constraints.
3) **Fitness scaling** is **Ranking**. This option converts the raw fitness score which is obtained after evaluation of each individual in each generation to the scaled score. The chosen option is ranking: the most fitted individual has the highest rank – 1, the second – 2, the last - n. The scaled values are allocated proportionally to $n^{-0.5}$.
4) **Selection function** is **Stochastic uniform**. Here the algorithm creates a line on which it puts parents where the length of the line corresponds to its expectation. Then, the algorithm moves along the line with equal step and determines the parents that are in the section of a step.
5) **Reproduction option** specifies the number of elite children – ones with the best fitness value – and the crossover fraction – the fraction of individuals in new generation which is produced by crossover. **Elite count** is set to 3 and **Crossover fraction** – 0.8.
6) **Mutation function** is **Adaptive feasible**. The function generates directions by chance that are in accordance to the successfulness of the last generation.
7) **The stopping criteria** are **Fraction tolerance** which is $1e-10$ and **Constraint tolerance** – $1e-6$.
8) **Augmented Lagrangian** is specified as a nonlinear constraint algorithm.
The Augmented Lagrangian Genetic Algorithm (ALGA) [19] tries to solve a nonlinear optimization problem with constraints. The algorithm formulates a subproblem that consists of the fitness function and nonlinear constraints using Lagrangian and penalty parameters. For each generation, the subproblem is formulated such that the bound constraints are satisfied. The ALGA minimizes a sequence of subproblems that approximate the original problem in the first generation. Since there aren’t any linear constraints a subproblem is defined as

\[ \Theta(x, \lambda, s) = f(x) - \sum_{i=1}^{n} \lambda_i \cdot s_i \cdot \log(s_i - c_i(x)), \]  

where \( \lambda_i \) is Lagrange multiplier; \( s_i \) – element of shifts.

The core of this method is simple imposing penalty function on the objective function which was described in IPM. When the problem is minimized to required tolerance, the solution is updated, or else penalty function increases to satisfy the constraints. The process continues until stopping criteria are met.

The results from solving the example by the using ALGA method are presented in the Table 2.

**Table 2.** Data results obtained from running Augmented Lagrangian Genetic Algorithm.

| №  | \( n \), rpm | \( s \), mm/turn | \( t \), mm | \( fval \), min |
|----|--------------|-----------------|----------|----------------|
| 1  | 864.0294     | 0.4041          | 2.4979   | 0.5446         |
| 2  | 864.0334     | 0.4936          | 2.5000   | 0.4455         |
| 3  | 864.0294     | 0.4936          | 2.5000   | 0.4455         |
| 4  | 864.0297     | 0.4936          | 2.5000   | 0.4455         |
| 5  | 864.0294     | 0.4041          | 2.4979   | 0.5446         |
| 6  | 864.0325     | 0.4936          | 2.5000   | 0.4455         |
| 7  | 864.0952     | 0.4935          | 2.5000   | 0.4455         |
| 8  | 864.0294     | 0.4936          | 2.5000   | 0.4455         |
| 9  | 864.0294     | 0.4936          | 2.5000   | 0.4455         |
| 10 | 864.0294     | 0.4041          | 2.4979   | 0.5446         |
| 11 | 864.0294     | 0.4936          | 2.5000   | 0.4455         |
| 12 | 864.0294     | 0.4936          | 2.5000   | 0.4455         |
| 13 | 864.0374     | 0.4936          | 2.5000   | 0.4455         |
| 14 | 864.0627     | 0.4936          | 2.5000   | 0.4455         |
| 15 | 864.0366     | 0.4936          | 2.5000   | 0.4455         |
| 16 | 871.5634     | 0.4842          | 2.5000   | 0.4502         |
| 17 | 864.1366     | 0.4935          | 2.5000   | 0.4455         |
| 18 | 864.0301     | 0.4936          | 2.5000   | 0.4455         |
| 19 | 864.0296     | 0.4936          | 2.5000   | 0.4455         |
| 20 | 864.0294     | 0.4936          | 2.5000   | 0.4455         |

6. Conclusion
The application of Linearization Programming Method for optimization of cutting parameters is justified. The results closely match those obtained using IPM. The method has fast convergence, ease in application, and precise accuracy.

The results from the ALGA are satisfactory, even though there are some discrepancies like the results from the tested example number 1, 5, 12, and 16 where the search direction of the algorithm deviated from the global minimum to the local minimum due to the stochastic engine of the algorithm. One blatant drawback of the genetic algorithm is convergence speed owing to all the calculations of generations. But still, for the solved problem, the convergence rate is more than enough. The future application of the algorithm is promising because it can be used for multi-objective optimization of cutting parameters (e. g. cost production and production rate). Also, whereas in the presence of several
local minima it is difficult for gradient methods to find global minimum, ALGA can easily manage such problem. The accuracy of ALGA may be increased with the usage of Hybrid function for example “fmincon” using IPM. When the genetic algorithm terminates, the Hybrid function uses ALGA’s final point as its initial. The application of Hybrid function gives exactly the same result as “fmincon” IPM.

References

[1] Maksarov V V and Leonidov P V 2014 Modeling and control of dynamical properties of the technological systems *Proceedings of the Mining Institute* **209** 71–77

[2] Olt J, Madiissoo M and Maksarov V V 2011 IEEE Int. Symp. on Assembly and Manufacturing ISAM 2011 pp 12–17

[3] Olt J J, Tärgla T T, Liivapuu A A, Maksarov V V and Liivapuu O O 2016 Mathematical modelling of cutting process system *Springer Proceedings in Mathematics & Statistics* **178** 173–86

[4] Hotz M and Vogel C 2014 *IEEE Trans. Signal Process* **62** 2566–79

[5] The MathWorks, Inc 1990–2017 *Optimization toolbox user’s guide*

[6] Forsgren A, Gill P E and Wright M H 2002 Interior methods for nonlinear optimization *SIAM J Appl Math.* **44** 525–97

[7] Cus F and Balic J 2003 Optimization of cutting process by ga approach *Robot Comput Integr Manuf* **19** 113–21

[8] D’Addona D M and Teti R 2013 Genetic algorithm-based optimization of cutting parameters in turning processes *46th CIRP Conf. on Manufacturing Systems* (vol 7) ed Cunha P F (Elsevier Procedia) pp 323–28

[9] Ryzhov E V and Averchenkov V I 1989 *Optimization of technological processes in machining* (Kiev: Naukova Dumka)

[10] Skuratov D L, Trusov V N and Lastochkin D A 2006 *Optimization of technological processes in mechanical engineering* (Samara: Samara State Aerospace University)

[11] Grubiši S V 2014 *Optimization of machining and control of the operating parameters* (Moscow: The Bauman University Publishing House)

[12] Nocedal J and S J Wright 2006 *Numerical Optimization* ed Glynn P and Robinson S M (New York: Springer–Verlag)

[13] Martyushev N V 2014 *Advanced Materials Research* **880** 163–7

[14] Maksarov V, Khalimonenko A 2017 *Key Engineering Materials* **736** 86–90

[15] Maksarov V, Khalimonenko A, Timofeev D 2017 *Agronomy Research* **12** 269–78

[16] Gabov V V, Zadkov D A 2017 *IOP Conf. Ser.: Earth Environ. Sci.* **87** 022007

[17] Dalsky A M, Kosilova A G, Meshcheryakov R K, Suslov A G, et al 2003 *Guide for an industrial engineer* vol 2 (Moscow: Mashinostroenie)

[18] Waltz R A, Morales J L, Nocedal J and Orban D 2006 An interior algorithm for nonlinear optimization that combines line search and trust region steps *Math. Program.* **107** 391–408

[19] Conn A R, Gould N and Toint Ph L 1997 A globally convergent augmented lagrangian barrier algorithm for optimization with general inequality constraints and simple bounds *Math. Comput.* **66** 261–88