J Intell Manuf (2016) 27:1209–1220
DOI 10.1007/s10845-014-0945-0

A metaheuristic for fast machining error compensation

Roman Stryczek

Received: 9 March 2014 / Accepted: 16 July 2014 / Published online: 17 August 2014

Abstract In case of complex parts machining or multi-directional machining in multi-part fixtures the error compensation in multi-dimensional decision space poses a difficult problem. The article focuses on the limitation of defective products by means of systematic increase of the remaining error budget due to correction of the setup data. A vectorial equation for machine tool space description is presented. The development of geometric dimensioning and tolerancing scheme to the levels connected with the setup data is proposed. The optimization algorithm used here is based on the paradigm particle swarm optimization (PSO), but it includes a few significant modifications inspired by the growth of the coral reef thus the name of the method—coral reefs inspired particle swarm optimization (CRIPSO). CRIPSO has been compared with three other popular metaheuristics: classic PSO, genetic algorithm, and cuckoo optimization algorithm. There is a practical example in this article.

Keywords Workpiece accuracy · Error budget · Machining error compensation · Metaheuristic optimization algorithm

List of symbols

Indices

f	Index of features ($f = 1, 2, \ldots, \|F\|$)
i	Index of decision variable ($i = 1, 2, \ldots, \|W\| + \|T\|$)
j	Index of WKS ($j = 1, 2, \ldots, \|W\|$)
m	Index of MKS ($m = 1, 2, \ldots, \|M\|$)
q	Index of dimension ($q = 1, 2, \ldots, \|D\|$)
p	Index of particle (polyp) (p $= 1, 2, \ldots, \|P\|$)
t	Index of tool ($t = 1, 2, \ldots, \|T\|$)

Sets

M	Nonempty, finite set of machine points
W	Nonempty, finite set of WKS points
T	Nonempty, finite set of code points of tools
F	Nonempty, finite set of features
D	Nonempty, finite set of dimensions
P	Current set of solutions (particles, polyps)
P_1	Set of solutions fulfilling criterion f_1
R	$R_{MW} \cup R_{WF} \cup R_{TF} \cup R_{FD}$
R_{MW}	Relation between the elements of sets M and W
R_{WF}	Relation between the elements of sets W and F
R_{TF}	Relation between the elements of sets T and F
R_{FD}	Relation between the elements of sets F and D

Parameters

b	Error budget bonus
c_1	Cognition coefficient
c_2	Social coefficient
$\overset{\leftrightarrow}{D}_q$	The modification of qth dimension
dZ^-	Lower error budget
dZ^+	Top error budget
E_i^{sd}	Error of ith setup data
ε_q	Performance error of qth dimension
E_q	Random error of qth dimension
$\overset{\leftrightarrow}{F}_f$	The modification of fth surface position
$gbest_i$	ith coordinate of the best positioned polyp

R. Stryczek (✉)
Department of Manufacturing Technology and Automation, University of Bielsko-Biala,ul. Willowa 2, 43-309 Bielsko-Biała, Poland
e-mail: rstryczek@ath.bielsko.pl

$plast_i$	ith coordinate of the last planula
$pbest_i$	ith coordinate of the best planula
r, r_1, r_2	Uniformly distributed random numbers among 0 to 1
T_q	Tolerance of qth dimension
w, w_1, w_2	Weighting factors
δ	Unilateral error budget
ΔZ	Maximum error budget

Decision variables

$\overleftrightarrow{W}$	Correction of nth WKS
$\overleftrightarrow{T}$	Correction of tth tool
x_i	Position of the polyp in ith dimension
x'_i	The position of setting a new planula in ith dimension
$\overleftrightarrow{x}$	$[\overleftrightarrow{W}_1, \overleftrightarrow{W}_2, \ldots, \overleftrightarrow{W}_{\lvert W\rvert}] \cup [\overleftrightarrow{T}_1, \overleftrightarrow{T}_2, \ldots, \overleftrightarrow{T}_{\lvert T\rvert}]$

Introduction

Error budget serves to predict and/or control errors in a system. Every machine tool has its own error budget serving to predict its accuracy and repeatability (Slocum 1992). The error budget helps identify where to focus resources to improve the accuracy of an existing machine or one under development (Hale 1999). Every workpiece can be also given certain error budget which stands for an acceptable error in machining it. The amount of error budget is related to the production cost (Bohez 2002). Therefore on every stage of the production process we need to maximize the remaining error budget.

In the machining process many various sources of errors can be identified (Fig. 1) as well as the manners of their compensation (Yuan and Ni 1998; Liu 1999; Ramesh et al. 2000a, b; Mehrabi et al. 2002; Wang et al. 2006; Loose et al. 2007; Rahou et al. 2009; Mekid and Ogedengbe 2010). Identification and compensation for particular errors is difficult time-consuming and costly, moreover their total eradication is in most cases impossible. The amount of errors can be limited even while designing the machine tool (Hale 1999) or they can be measured and compensated for (Sartori and Zhang 1995). Compensation can relate to the chosen errors or the total error. Typical approach to the error compensation includes building the mathematical models taking into account the data from the sensors placed on the machine tool in on-machine-measurement (OMM) system (Ziegert and Kalle 1994; Ouafi et al. 2000; Cho et al. 2006; Marinescu et al. 2011), by the reconstruction of the control program (Mäkelä et al. 2011; Cui et al. 2012) and more.

In case of mass production of complex parts in the machining centers we often deal with a summary compensation of the system errors based on the measurement reports from coordinate measuring machines (CMM). A major problem with CMMs in a modern manufacturing environment is the difficulty associated with taking advantage of the measurement results (Davis et al. 2006). The CMM reports rarely present direct values of the correction necessary to modify setup data of CNC (Schwenke et al. 2008). In case of mass production we have to deal with multi-tool, multi-directional machining, in addition, during one machining process many features are machined in a few workpieces fixed in multi-part fixtures. In such case the measurement reports are thick documents, which analysis is difficult and time-consuming and they can be easily misinterpreted. Errors on this stage usually result in the defective production. Taking into account the variety of error sources and their dynamic change in every cycle machining, the process of compensating should be applied automatically and quite often.

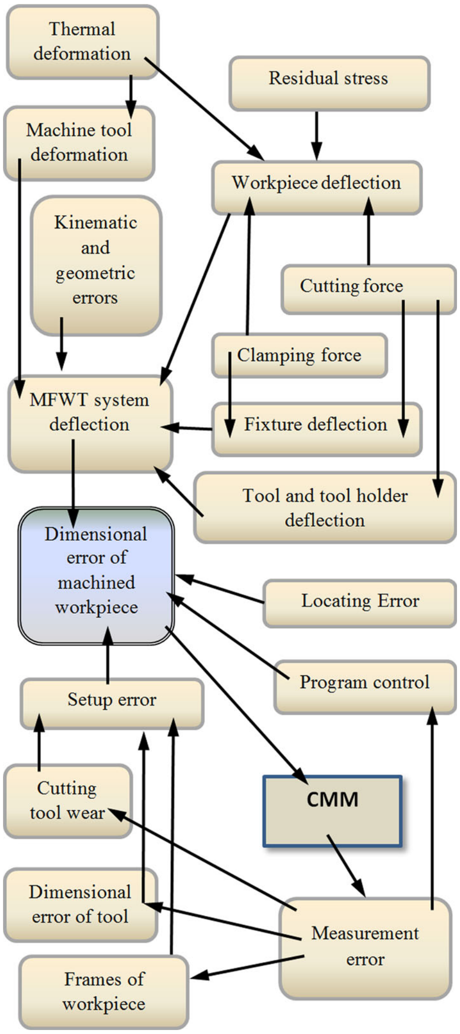
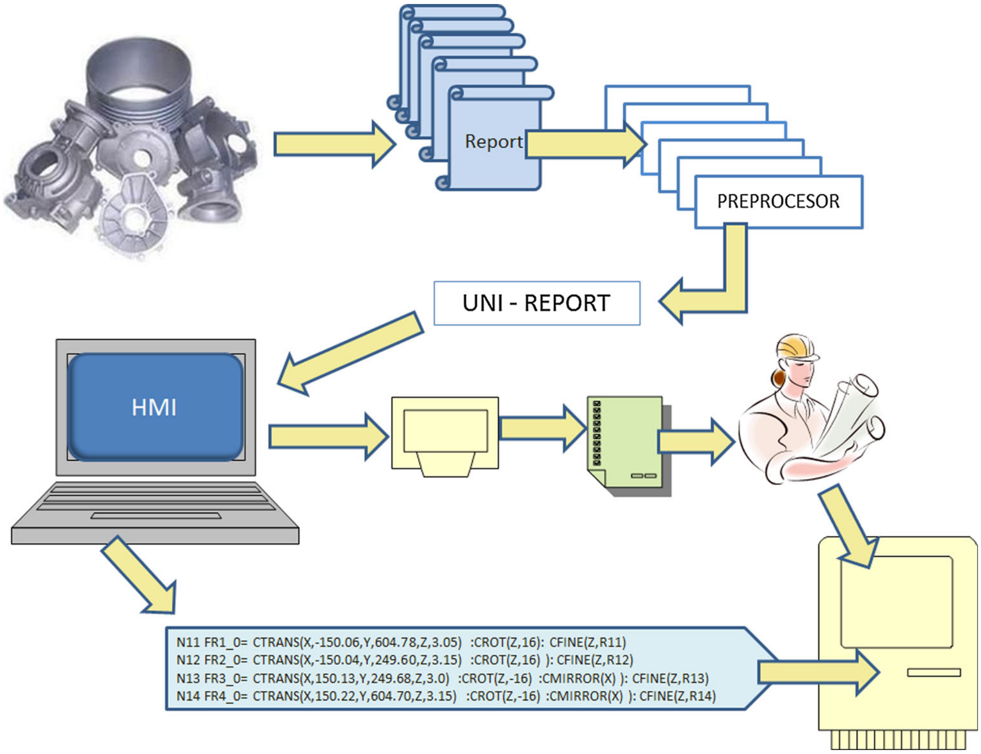


Fig. 1 Source of machining errors



```
N11 FR1_0= CTRANS(X,-150.06,Y,604.78,Z,3.05)  :CROT(Z,16): CFINE(Z,R11)
N12 FR2_0= CTRANS(X,-150.04,Y,249.60,Z,3.15)  :CROT(Z,16) ): CFINE(Z,R12)
N13 FR3_0= CTRANS(X,150.13,Y,249.68,Z,3.0)  :CROT(Z,-16)  :CMIRROR(X) ): CFINE(Z,R13)
N14 FR4_0= CTRANS(X,150.22,Y,604.70,Z,3.15)  :CROT(Z,-16) :CMIRROR(X)  ): CFINE(Z,R14)
```

Fig. 2 The process chain of taking into account the measurement results

The solution to this problem is creating the process chain enabling us to apply the measurement results automatically in the form of compensating corrections for CNC setup data (Fig. 2). Since the early 1990s, new product data standards such as STEP and STEP-NC have been developed to provide standardized and comprehensive data models for machining and inspections (Kumar et al. 2007; Zhao et al. 2009). In a plant or factory they usually check many produced parts with one or a few measuring machines. Thus the first element of the process chain should be a set of postprocessors which standardizes the form of the reports. In such form the reports can be processed by the right program which generates the corrections for the tools and workpiece coordinate systems (WKS) used in the machining process. The correction in the form of changing the tool trajectory should be considered only as a last resort, as it limits—in case of multi-part fixtures—the possibility to apply the subprogram technique. The corrections should rather be automatically conveyed to right registers directly or indirectly by use of the array of the Real-parameters.

This article discusses automatic determination of the corrections on the axle Z of the machine tool by changing the setup data of the tools and the position of the local coordinate systems connected with the workpiece. In case of a complex geometric dimensioning and tolerancing (GD&T) model finding good solution requires optimization procedure.

Metaheuristic techniques in process planning and manufacturing

The problem of optimization in the discussed issue comes down to searching the optimal solution in multi-dimensional, discrete state space. It is a classical artificial intelligence problem. Computational intelligence (CI) is a successor of good old-fashioned artificial intelligence (GOFAI). GOFAI developed as the project of empirical research, implements a weak model of semantic networks. CI rather relies on metaheuristic algorithms such as fuzzy systems, artificial neural networks (ANNs), evolutionary computation, artificial immune systems, etc. CI combines elements of learning, adaptation, evolution and fuzzy logic (rough sets) to create programs that are, in a sense, intelligent. So far, many metaheuristic search algorithms have been invented. Timeline of main metaheuristic algorithms devised up to 2010 has been presented in (Gandomi et al. 2013). Recently, there's been rapid increase in popularity of newly defined approaches, such as: cuckoo search algorithm (Mohamad et al. 2013), glowworm swarm optimization (Zainal et al. 2013), Levy flight algorithm (Kamaruzaman et al. 2013), firefly algorithm (Johari et al. 2013).

In year 2000, Kusiak published a trailblazing book (Kusiak 2000), which, for the first time, included state-of-the-art CI techniques to all phases of manufacturing system design and operations. The most important areas of manufacturing engineering are process planning and process control. The process plan constitutes an ordered sequence of tasks which can transform raw material into the final part economically and competitively. The major process planning activities are: features extraction, part classification for group technology, machining volume decomposition, tool path generation, machining condition optimization, operation sequencing, machine, setup and tool selection, and others. Because process planning is a NP-hard problem, some global search techniques must be applied. A brief review of CI applications in machining process planning and related methods and problems will be presented in work (Stryczek 2007). Fuzzy logic (FL) can by effectively used in modeling of machining processes such as predicting the surface roughness and controlling the cutting force. Adnan et al. (2013) present FL techniques used in machining process. FL was considered for prediction, selection, monitoring, control and optimization of machining process.

In the field of machining process planning, CI methods have been used most often for optimizing machining process parameters. The gradient-based non-linear optimization techniques have difficulty with solving those optimization problems; one must resort to alternative, conventional, non-systematic optimization techniques, i.e., evolutionary algorithms (Zhang et al. 2006). An overview and the comparison of the researches from 2007 to 2012 that used evolutionary optimization techniques to optimize machining process parameters were presented in Yusup et al. (2012a, b). Genetic algorithm (GA) stands out among CI methods as it is currently the full-grown method of single and multi-objective optimization, tested in many variants. Zain et al. (2008) discussed on how GA system operates in order to optimize the surface roughness performance measure in milling process. Yusoff et al. (2011) reviews the application of non dominated sorting genetic algorithm II (NSGA-II), classified as one of multi-objective GA techniques, for optimizing process parameters in various machining operations.

Coral reefs inspired particle swarm optimization

The popular metaheuristic, Particle Swarm Optimization (PSO), has been developed originally by Kennedy and Eberhart (1995). At the same time deep modifications have been introduced to PSO (Poli et al. 2007). They were inspired by the rebuilding processes in the coral reef, so the method has been called coral reefs inspired particle swarm optimization (CRIPSO).

The coral reef is a complicated ecosystem bound by the evolution rules, which undergoes constant reconstruction. It is a live organism consisting mostly of polyps which cluster into colonies. During the reproduction coral reef grows upwards and sideward. Polyps produce small larva better known as planula. After swimming for a few hours or even days planula settles down and transforms into a new polyp competing for food and space. Comparing polyp colonies with the particle swarm we can see the following differences. Polyp is a stationary particle undergoing reproduction and dying many times. In the classic PSO, particle is in constant movement and there is no generational change. Potential advantage of the modification is that the particles with the lower mark are eliminated from the population and the best ones can generate their descendants many times. CRIPSO is based on the principle of elitism

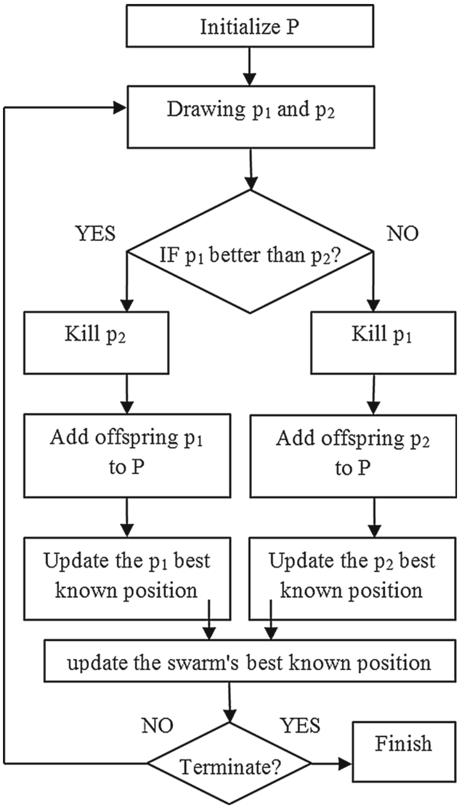

Initialize P
Drawing p_1 and p_2
IF p_1 better than p_2?
YES
NO
Kill p_2
Kill p_1
Add offspring p_1 to P
Add offspring p_2 to P
Update the p_1 best known position
Update the p_2 best known position
update the swarm's best known position
NO
YES
Terminate?
Finish

Fig. 3 Basic CRIPSO flow chart

Figure 3 represents the course of the best solution search cycle. Each particle (polyp) stands for one solution in decision space. Algorithm was based on a contest between the two solutions chosen at random. The winning solution remains in set P thus the elitist character of the procedure while the worse solution is removed from the set P. In its place the winner generates its own planula.

In real world planula moves in the ocean in the chaotic manner. Most of them die. However we can assume that those who manage to settle down and change into polyp obeyed the rules of particle swarm movement. Therefore the procedure of defining the position of a descendant has been based on PSO method. The position of the planula settlement in ith dimension is determined according to the formula 1.

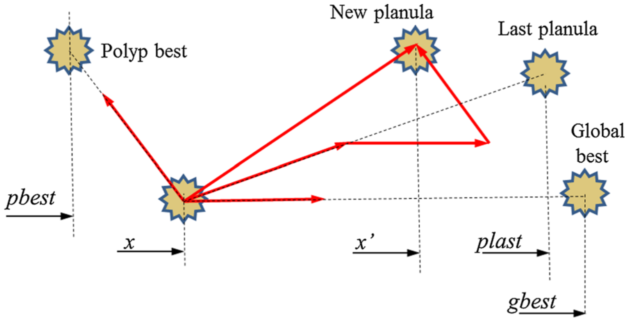


Fig. 4 Determining new position of particle

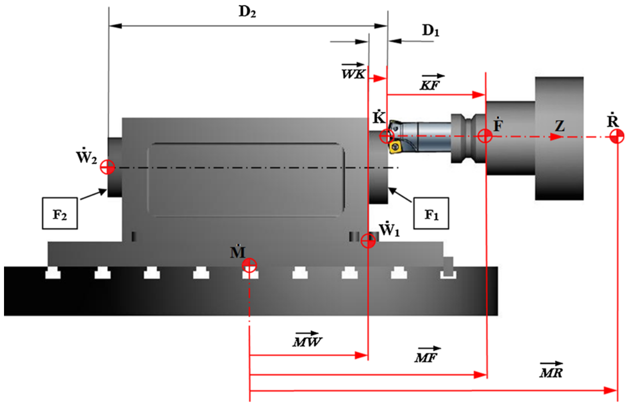


Fig. 5 Workspace of horizontal-spindle machining center

According to this, the particle shift vector's components are determined as the sum of three components (Fig. 4) in the directions consistent with:

I. The shift of the last planula generated by the given polyp. In case of a new polyp the direction of the inertia vector is agrees with the movement direction between the planula's parent's position and the point of planula's settlement.
II. The shift towards the descendant of the given polyp which has the best position so far. In case of a new polyp, position "polyp best" is inherited from the parent.
III. The shift towards the best situated polyp in the whole colony.

$$x\prime_i = x_i + w\left(plast_i - x_i\right) + c_1 r_1 \left(pbest_i - x_i\right) + c_2 r_2 \left(gbest_i - x_i\right) \tag{1}$$

The proposed compensation system

Let us focus on a simple example stemming from the analysis of the workspace of the 4-Axis (X′YZB′) horizontal-spindle machining center (Fig. 5). In the workspace we distinguish the following characteristic points: $\dot{R}$—reference point, $\dot{M}$—machine point, $\dot{W}_j$—the points of the origin of the coordinate system connected with the workpiece, $\dot{K}_t$—the code point of the tth tool and $\dot{F}$—the toolholder reference point.

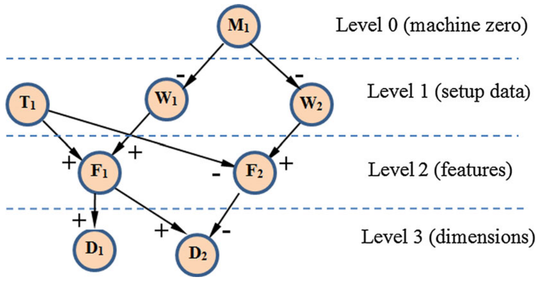


Fig. 6 Four-level weighted relationships graph

Those points are connected by the vectorial equation for machine tool space:

$$\overrightarrow{MF} = \overrightarrow{MW} + \overrightarrow{WK} + \overrightarrow{KF} \tag{2}$$

This equation is the basis for analyzing the influence of the setup data presented here by the components $\overrightarrow{MW}$ i $\overrightarrow{KF}$, on the position of machined features, and consequently on the obtained dimensions. The relationships occurring here can be displayed as four-level weighted relationship graph (Fig. 6).

On the basis of this model we can generate formulas which show the influence of the setup data change on fth features $\overset{\leftrightarrow}{F}$ and qth dimension $\overset{\leftrightarrow}{D}$ shift.

$$\begin{aligned}
&\overset{\leftrightarrow}{\underset{1}{F}} = -\overset{\leftrightarrow}{M}_1 + \overset{\leftrightarrow}{W}_1 + \overset{\leftrightarrow}{T}_1,\\
&\overset{\leftrightarrow}{F}_2 = -\overset{\leftrightarrow}{M}_1 + \overset{\leftrightarrow}{W}_2 - \overset{\leftrightarrow}{T}_1,\\
&\overset{\leftrightarrow}{D}_1 = \overset{\leftrightarrow}{F}_1 = -\overset{\leftrightarrow}{M}_1 + \overset{\leftrightarrow}{W}_1 + \overset{\leftrightarrow}{T}_1,\\
&\overset{\leftrightarrow}{D}_2 = \overset{\leftrightarrow}{F}_1 - \overset{\leftrightarrow}{F}_2 = -\overset{\leftrightarrow}{M}_1 + \overset{\leftrightarrow}{W}_1 + \overset{\leftrightarrow}{T}_1 - \left(-\overset{\leftrightarrow}{M}_1 + \overset{\leftrightarrow}{W}_2 - \overset{\leftrightarrow}{T}_1\right)\\
&\quad = \overset{\leftrightarrow}{W}_1 + \overset{\leftrightarrow}{W}_2 + 2\overset{\leftrightarrow}{T}_1.
\end{aligned}$$

On the whole the influence of setup data on resultant dimensions in Z axis can be presented in the form of orderly six-tuple PM = $(M, W, T, F, \mathrm{D}, R)$. In the graphical interpretation the elements of M, W, T, F and D sets are the graph vertices whereas the elements of R set are weighted directed arcs. Weight for arc $a \to z$ take the value:

$$w_{az} = \begin{cases} 1 & \text{if increase } a \text{ causes increase } z, \\ -1 & \text{if increase a causes decrease } z, \\ 0 & \text{otherwise.} \end{cases}$$

In case of elements belonging to M, W, T and F, sets the increase means shifting towards the positive values of the Z axis of MKS meanwhile in case of the elements of D set it means the increase of the dimension value. These relationships can be written down synthetically in the matrix form:

$$R_{MW} = \begin{bmatrix} w_{11} & \dots & w_{1|W|} \\ \vdots & \ddots & \vdots \\ w_{|M|1} & \dots & w_{|M||W|} \end{bmatrix},$$

$$R_{WF} = \begin{bmatrix} w_{11} & \dots & w_{1|F|} \\ \vdots & \ddots & \vdots \\ w_{|W|1} & \dots & w_{|W||F|} \end{bmatrix},$$

$$R_{TF} = \begin{bmatrix} w_{11} & \dots & w_{1|F|} \\ \vdots & \ddots & \vdots \\ w_{|T|1} & \dots & w_{|T||F|} \end{bmatrix},$$

$$R_{FD} = \begin{bmatrix} w_{11} & \dots & w_{1|D|} \\ \vdots & \ddots & \vdots \\ w_{|F|1} & \dots & w_{|F||D|} \end{bmatrix}.$$

Predicted shifts of the features position and the changes of the dimensions can be, therefore, expresses as the vectors' components in form of the following formula:

$$\left[\overset{\leftrightarrow}{F}_1 \dots \overset{\leftrightarrow}{F}_{|F|}\right] = ([\overset{\leftrightarrow}{M}_1 \dots \overset{\leftrightarrow}{M}_{|M|}] \cdot R_{MW} + [\overset{\leftrightarrow}{W}_1 \dots \overset{\leftrightarrow}{W}_{|W|}]) \cdot R_{WF} + [\overset{\leftrightarrow}{T}_1 \dots \overset{\leftrightarrow}{T}_{|T|}] \cdot R_{TF} \quad (3)$$

$$[\overset{\leftrightarrow}{D}_1 \dots \overset{\leftrightarrow}{D}_{|D|}] = [\overset{\leftrightarrow}{F}_1 \dots \overset{\leftrightarrow}{F}_{|F|}] \cdot R_{FD} \quad (4)$$

In the practical applications the 0 level of the model, i.e. components $[\overset{\leftrightarrow}{M}_1 \dots \overset{\leftrightarrow}{M}_{|M|}$ in Eq. 4 can be omitted. This is so, because the errors connected with the zero point of the machine are usually comparatively insignificant and there may be some problems with their automatic compensation.

Formulating the optimization task

The aim for the engineer of the technological process is to provide the maximum unilateral error budget δ, which is understood as the distance between the zero deviation line and a closer of the two borderlines (top and lower) of the tolerance zone (Fig. 7b). Value δ should be bigger than the performance error ε. In the best possible case value δ matches a half of the admissible error budget ΔZ_{max}, that is a half of the minimal tolerance zone of all considered dimensions (Fig. 7a). The ideal case occurs very seldom or never. In practice the amount of the remaining error budget ΔZ is always smaller than the smallest tolerance zone (Fig. 7b). On of the manners to increase δ is to correctly modify setup data so that to acquire additional error stock, so called bonus error budget b. The bonus error budget reduces the number of rejected parts by increasing the tolerance zone. In order to determine b, first we have to find the value of the tolerance zone center shift off the zero line ε_q (Fig. 7b) resulting from the correction of the setup data. The objective of the optimization f_1 is to find the proper values of the setup data so as to maximize δ for the least advantageous position of the tolerance zone (8). The bonus error budget is the difference between the presently registered value δ and the predicted value obtained after the optimization.

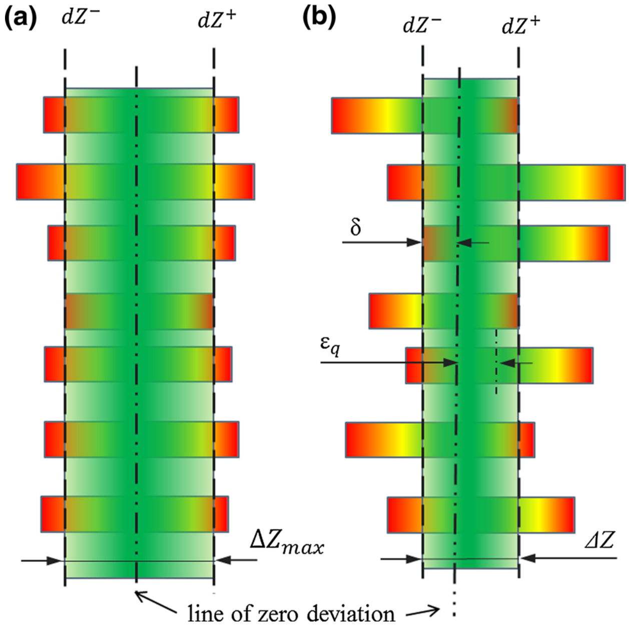


Fig. 7 The distributions of tolerance zone: **a** in best case, **b** in real case

$$dZ^- = \max_q\left(\varepsilon_q - \frac{T_q}{2}\right) \quad (5)$$

$$dZ^+ = min_q\left(\varepsilon_q + \frac{T_q}{2}\right) \quad (6)$$

$$\Delta Z = dZ^+ - dZ^- \quad (7)$$

The following objective functions have been taken into account:

$$f_1 = \delta_{max} = \max_p\left(\min_q\left\{dZ_q^-, dZ_q^+\right\}\right) \quad (8)$$

$$f_2 = \min_{p \in P_1}\sqrt{\sum_i \overset{\leftrightarrow}{x}{}_i^2} \quad (9)$$

$$f = 10 f_1 - f_2 \quad (10)$$

During the first step we seek the solutions for which f_1 has maximum value. The decision space should be limited on both sides in each of dimensions, e.g. to the range [−0.1, 0.1]. Furthermore it is a discrete space which results from the precision of determining the setup data; mostly 0.001 mm or 0.0001 of an inch. However the number of solutions, even with the above limitations, is huge, e.g. 1.024×10^{23} for 10-dimensional decision space. Therefore the set of P_1 solutions matching f_1 is also quite numerous. Among P_1 elements we need to find the one for which the resultant vector of the change for all the setup data is the smallest (f_2). To make calculations simpler the substitute objective function f has

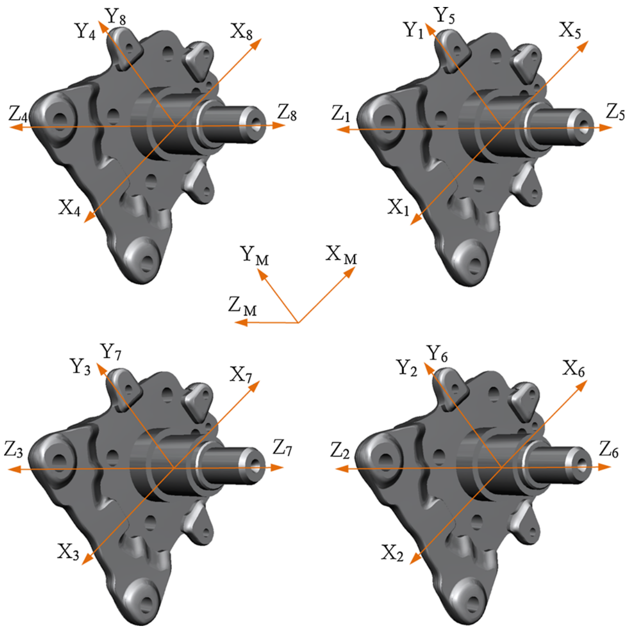


Fig. 8 Machine and workpiece coordinate systems

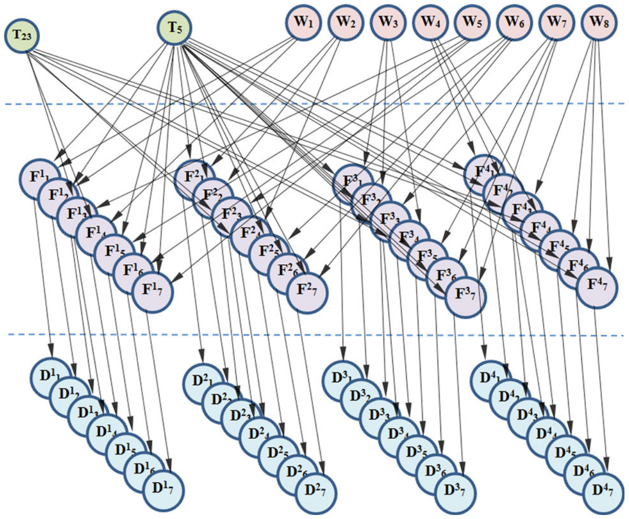

T23
T5
W1
W2
W3
W4
W5
W6
W7
W8
F11 F12 F13 F14 F15 F16 F17
F21 F22 F23 F24 F25 F26 F27
F31 F32 F33 F34 F35 F36 F37
F41 F42 F43 F44 F45 F46 F47
D11 D12 D13 D14 D15 D16 D17
D21 D22 D23 D24 D25 D26 D27
D31 D32 D33 D34 D35 D36 D37
D41 D42 D43 D44 D45 D46 D47

Fig. 9 Model of machining process of the given parts

been introduced. It converts the original problem with two objectives into a single-objective optimization problem.

Practical example

Let us take into consideration a machining process in a 4-part fixture (Fig. 8). Rear axle guide pins are being machined. The machining process is carried out in two angular positions of the rotary table, 0° and 180°, so taking 8 WKS is advantageous. Figure 9 presents the model of the relationship between the structure elements of the machining process. Figure 10 presents the GD&T model for one machined part.

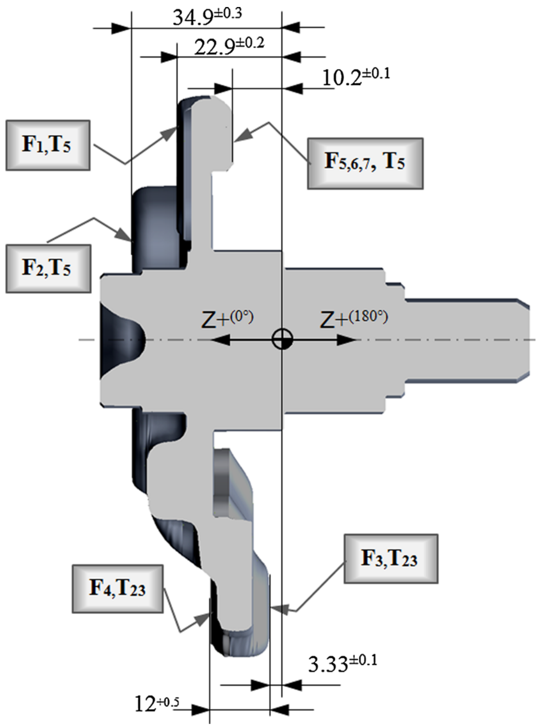


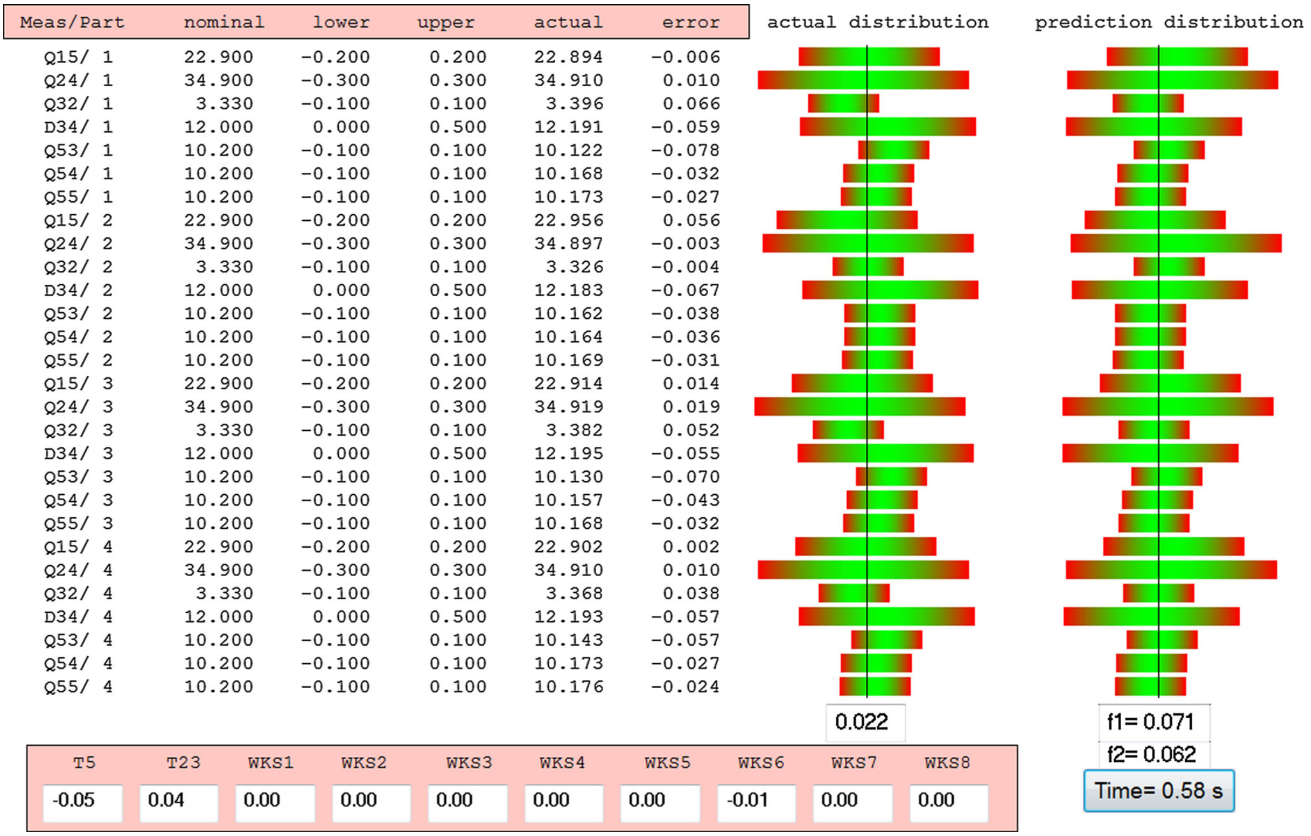
Fig. 10 Model GD&T of the given part

Relationships R_{TF}, R_{WF}, R_{TF}, for one workpiece have been presented below in the matrix form.

$$R_{TF} = \begin{array}{c|ccccccc} & F_1 & F_2 & F_3 & F_4 & F_5 & F_6 & F_7 \\ \hline T_5 & 1 & 1 & 0 & 0 & -1 & -1 & -1 \\ T_{23} & 0 & 0 & -1 & 1 & 0 & 0 & 0 \end{array}$$

$$R_{WF} = \begin{array}{c|ccccccc} & F_1 & F_2 & F_3 & F_4 & F_5 & F_6 & F_7 \\ \hline W_1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ W_5 & 0 & 0 & -1 & 0 & -1 & -1 & -1 \end{array}$$

$$R_{FD} = \begin{array}{c|ccccccc} & D_1 & D_2 & D_3 & D_4 & D_5 & D_6 & D_7 \\ \hline F_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ F_2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ F_3 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ F_4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ F_5 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ F_6 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ F_7 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}$$



Meas/Part	nominal	lower	upper	actual	error	actual distribution	prediction distribution
Q15/ 1	22.900	-0.200	0.200	22.894	-0.006		
Q24/ 1	34.900	-0.300	0.300	34.910	0.010		
Q32/ 1	3.330	-0.100	0.100	3.396	0.066		
D34/ 1	12.000	0.000	0.500	12.191	-0.059		
Q53/ 1	10.200	-0.100	0.100	10.122	-0.078		
Q54/ 1	10.200	-0.100	0.100	10.168	-0.032		
Q55/ 1	10.200	-0.100	0.100	10.173	-0.027		
Q15/ 2	22.900	-0.200	0.200	22.956	0.056		
Q24/ 2	34.900	-0.300	0.300	34.897	-0.003		
Q32/ 2	3.330	-0.100	0.100	3.326	-0.004		
D34/ 2	12.000	0.000	0.500	12.183	-0.067		
Q53/ 2	10.200	-0.100	0.100	10.162	-0.038		
Q54/ 2	10.200	-0.100	0.100	10.164	-0.036		
Q55/ 2	10.200	-0.100	0.100	10.169	-0.031		
Q15/ 3	22.900	-0.200	0.200	22.914	0.014		
Q24/ 3	34.900	-0.300	0.300	34.919	0.019		
Q32/ 3	3.330	-0.100	0.100	3.382	0.052		
D34/ 3	12.000	0.000	0.500	12.195	-0.055		
Q53/ 3	10.200	-0.100	0.100	10.130	-0.070		
Q54/ 3	10.200	-0.100	0.100	10.157	-0.043		
Q55/ 3	10.200	-0.100	0.100	10.168	-0.032		
Q15/ 4	22.900	-0.200	0.200	22.902	0.002		
Q24/ 4	34.900	-0.300	0.300	34.910	0.010		
Q32/ 4	3.330	-0.100	0.100	3.368	0.038		
D34/ 4	12.000	0.000	0.500	12.193	-0.057		
Q53/ 4	10.200	-0.100	0.100	10.143	-0.057		
Q54/ 4	10.200	-0.100	0.100	10.173	-0.027		
Q55/ 4	10.200	-0.100	0.100	10.176	-0.024		
						0.022	f1= 0.071 f2= 0.062

T5	T23	WKS1	WKS2	WKS3	WKS4	WKS5	WKS6	WKS7	WKS8
-0.05	0.04	0.00	0.00	0.00	0.00	0.00	-0.01	0.00	0.00

Time= 0.58 s

Fig. 11 User interface

The changes of the features position $\overleftrightarrow{F}$ and the dimensions values $\overleftrightarrow{Q}$, depending on the setup data corrections are determined for each part by means of the following formulas:

$$\overleftrightarrow{F}_1 = \overleftrightarrow{W}_1 + \overleftrightarrow{T}_5 \qquad \overleftrightarrow{D}_1 = \overleftrightarrow{W}_1 + \overleftrightarrow{T}_5$$

$$\overleftrightarrow{F}_2 = \overleftrightarrow{W}_1 + \overleftrightarrow{T}_5 \qquad \overleftrightarrow{D}_2 = \overleftrightarrow{W}_1 + \overleftrightarrow{T}_5$$

$$\overleftrightarrow{F}_3 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_{23} \qquad \overleftrightarrow{D}_3 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_{23}$$

$$\overleftrightarrow{F}_4 = \overleftrightarrow{W}_1 + \overleftrightarrow{T}_{23} \qquad \overleftrightarrow{D}_4 = \overleftrightarrow{W}_1 + \overleftrightarrow{W}_5 + 2\overleftrightarrow{T}_{23}$$

$$\overleftrightarrow{F}_5 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_5 \qquad \overleftrightarrow{D}_5 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_5$$

$$\overleftrightarrow{F}_6 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_5 \qquad \overleftrightarrow{D}_6 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_5$$

$$\overleftrightarrow{F}_7 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_5 \qquad \overleftrightarrow{D}_7 = -\overleftrightarrow{W}_5 - \overleftrightarrow{T}_5$$

In Fig. 11 the user interface of the setup data correction program has been presented. Input data are read automatically after indicating the file with the measurement results. Figure 12 shows how the values of benchmark f_1, f_2, and f as function of iteration are determined. In this example the bonus error budget amounted to b = 0.071–0.22 = 0.049 mm. Increasing this value only by means of modifying the setup data is impossible. Further increase of error budget would require interfering with the control program. In case of multi-part machining process, in which the subprogram technique is applied, such interference is unadvisable as it complicates the structure and makes the control over the program more difficult.

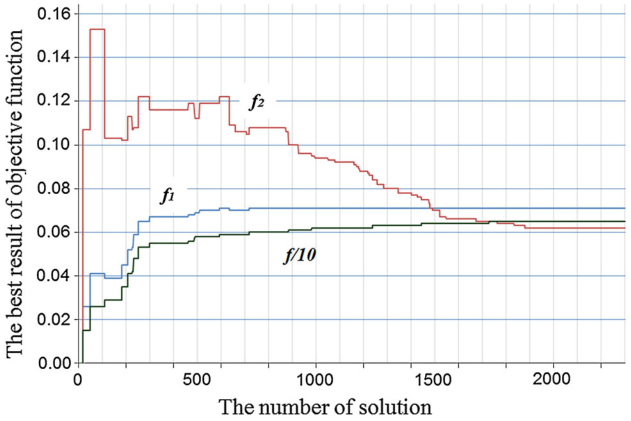


Fig. 12 Values of benchmark f_1, f_2 and f as function of number of solutions in CRIPSO

Comparison between the proposed CRIPSO and other algorithms

CRIPSO has been compared with three other popular meta-heuristics: classic PSO, GA, and cuckoo optimization algo-

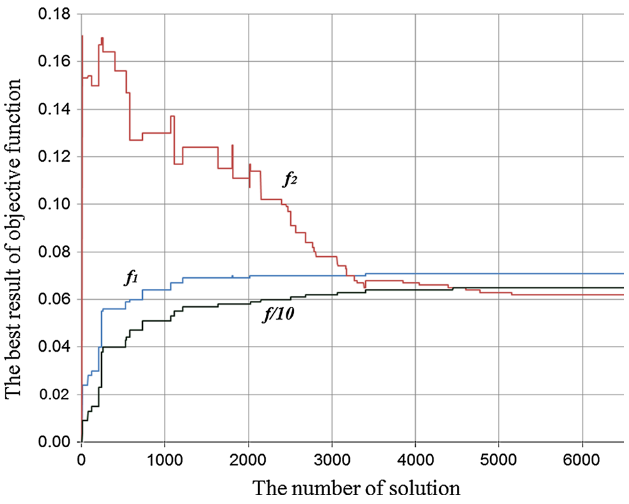


Fig. 13 Values of benchmark f_1, f_2 and f as function of number of solutions in PSO

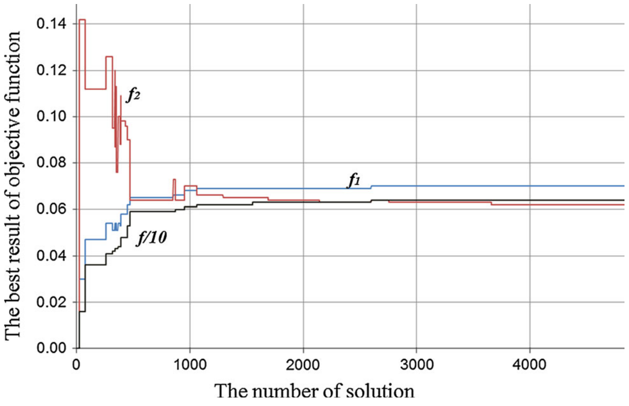


Fig. 14 Values of benchmark f_1, f_2 and f as function of number of solutions in GA

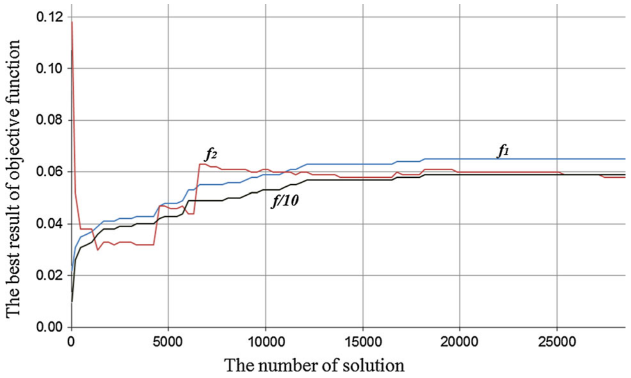


Fig. 15 Values of benchmark f_1, f_2 and f as function of number of solutions in COA

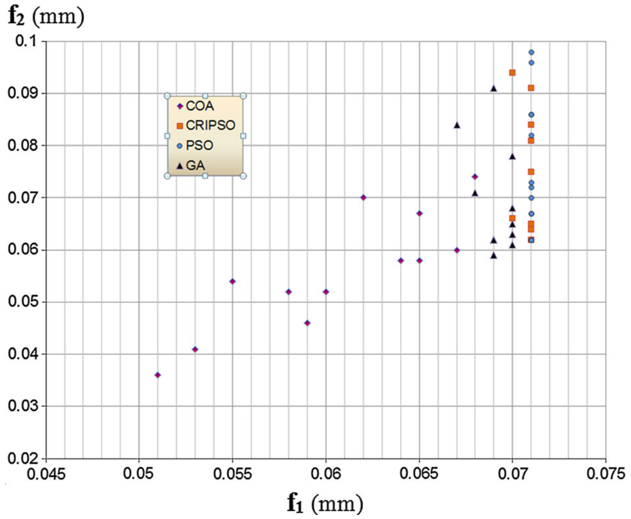


Fig. 16 Comparison of test results for COA, CRIPSO, PSO and GA

rithm (COA). For each of these methods the tests consisting of twelve trials have been carried out. Figure 12 presents a typical the process of searching for the best solution by means of CRIPSO. The current set of polyps (solutions) was constant and amounted to 100. Coefficients c_1 and c_2 were calculated each time according to the formula 11. Inertia weight (w) was established on the constant level of 0.75. The total number of iterations was established as 5,000. The optimal solution was reached after less than 2,000 iterations. The solutions were repeatable (Figs. 16, 17). The time needed to generate 5,000 solutions was 0.58 s.

$$c_1 = c_2 = \cos\left(\frac{\pi}{2} \cdot \frac{L_i}{L}\right) \tag{11}$$

where L_i: successive number of a solution, L: the total number of solutions.

Figure 13 shows a typical process of searching for the best solution by means of PSO. The basic pattern of PSO has been applied. Globally, the best position of a particle was updated each time after generating a new solution. The current set of particles (solutions) was constant and amounted to 100. Coefficients c_1 and c_2 were calculated identically as in CRIPSO. The time needed to generate 10,000 solutions was 0.96 s. Solutions were obtained after about 5,000 iterations on average. The spread of results was similar to CRIPSO (Figs. 16, 17).

Figure 14 shows a typical process of searching for the best solution by means of GA. The applied flowchart of GA was presented by Stryczek (2009). It is the variation of GA including the principle of elitism, as the winners of the competition qualify to the next population. The quantity of the population was 50 and the probability of the gene mutation 0.1. The time needed to generate 5,000 solutions was 0.72 s. The obtained results were only slightly worse than those of PSO and CRIPSO. GA finds the right solution quickly, after about 500 trials, but it has a problem with improving them afterwards.

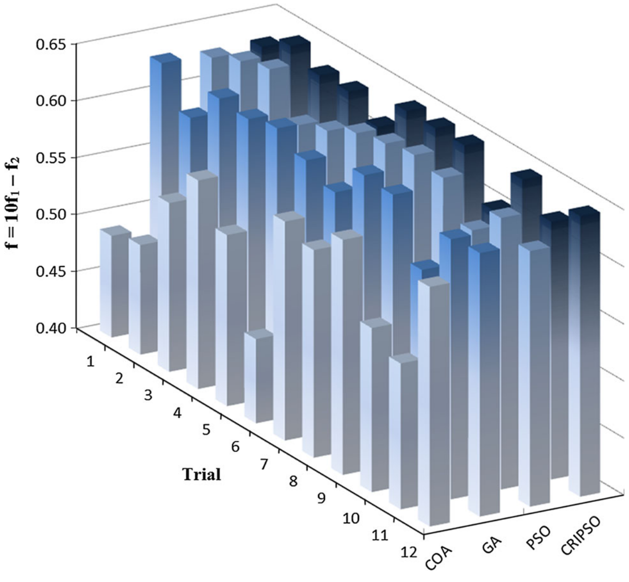


Fig. 17 Comparison of test results for COA, CRIPSO, PSO and GA

The fourth method of optimization is a metaheuristic known for several years which is called COA (Rajabioun 2011). The method is interesting, yet it takes into account many parameters which affect it's effectiveness. The best solutions were obtained for the maximum number of habitats which amounted to 40 and for 5–10 laid eggs (Fig. 15). The trials of 100 iterations which corresponds with checking approximately 30,000 solutions in 2.11 s. The results diverged from other methods in terms of quality. The best of them are to be considered satisfactory, though (Figs. 16, 17).

Validate the results of the proposed scheme

For the purpose of statistic validation of the proposed method a number of tests have been carried out for the model presented in "Practical example" section, for 100 measurement result sets. For each set, 10 tests have been conducted. On the whole, 1,000 trials have been carried out and their results are presented in Fig. 18. measurement results have been generated at random, but with maintaining the weight proportions considering the influence of setup data errors (w_1) and the remaining sources of performance errors (w_2). Firstly, setup data errors have been generated randomly (12) then their influence on analyzed dimensions. After the additional, random deviations (13) have been taken into account, the error budget for each case has been calculated. For the conducted trials it was within: −0.011 to 0.063. The negative value of the error budget means that the part has been made beyond the boundaries of tolerance field. Those boundaries can be controlled by establishing the proper weights w_1 and w_2. During the tests they were established on the set level accordingly 0.03 and 0.02. Choosing the proper weight has crucial influence on the possibility of increasing the error budget by correcting the setup data. The important conclusion drawn from the conducted tests should point out the possibility of using the proposed methodology to estimate the share of the setup data error in the total machining error. For instance, for 60 % share of setup data in the total machining error and the initial error budget value (f_1) of 0.022 the foreseen error budget amounted to 0.085 (Fig. 18, series 26). Meanwhile, for the same value of the initial error budget, for the example form Fig. 16, none of the tested optimization methods succeeded in reaching the value higher than 0.071. Therefore the conclusion that the share of setup data error was less that 60 % in this case. This subject is so extensive that it is not discussed here further. CRIPSO always generated good solutions regardless of the initial error budget value as shown in the results of the conducted tests. Only a few per cent of the results deviated for the average (Fig. 18).

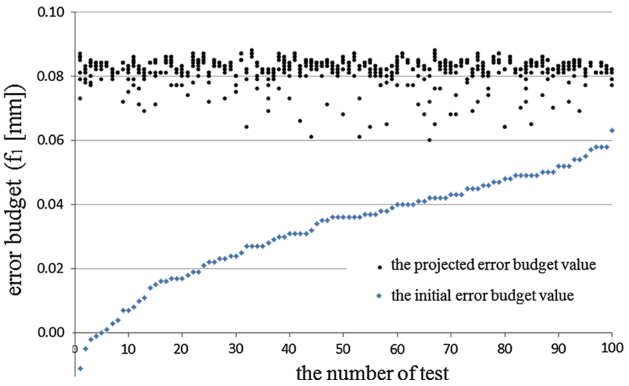


Fig. 18 The sorted test results

$$E_i^{sd} = w_1(1 - 2r) \tag{12}$$

$$E_q = w_2(1 - 2r) \tag{13}$$

Conclusion

The article presented the practical method of the automatic correction of setup data of the horizontal-spindle machining center in axle Z. Applying this method to the remaining axles is aimless as the errors in *XY* surface and in axle *B* have different character. This method is suitable for multi-directional and multi-tool machining using multi-part fixtures. The approach presented has been compared with the classic PSO, GA and COA methods. The effectiveness of PSO, GA and CRIPSO has been on a similar level and the results have been within the range of the random spread characteristic for these methods. The above method of the setup data correction can be applied automatically without the worker's interference. It should be the essential element of the computer-integrated manufacturing (CIM). The time needed to apply this method in practice depends mainly on the link between GD&T and the setup data in the production process. Therefore the further research should focus on automation of this stage of planning.

References

Adnan, M. R. H. M., Sarkheyli, A., Zain, A. M., & Haron, H. (2013). Fuzzy logic for modeling machining process: A review. *Artificial Intelligence Review*. doi:10.1007/s10462-012-9381-8.

Bohez, E. L. J. (2002). Compensating for systematic errors in 5-axis NC machining. *Computer-Aided Design*, *34*, 391–403.

Cho, M. W., Kim, G. H., Seo, T. I., Hong, Y. C., & Cheng, H. H. (2006). Integrated machining error compensation method using OMM data and modified PNN algorithm. *International Journal of Machine Tools & Manufacturing*, *46*, 1417–1427.

Cui, G., Lu, Y., Li, J., Gao, D., & Yao, Y. (2012). Geometric error compensation software system for CNC machine tools based on NC program reconstructing. *International Journal of Advanced Manufacturing Technolology*, *63*, 169–180.

Davis, T. A., Carlson, S., Red, W. E., Jensen, C. G., & Sipfle, K. (2006). Flexible in-process inspection through direct control. *Measurement*, *39*, 57–72.

El Ouafi, A., Guillot, M., & Bedruni, A. (2000). Accuracy enhancement of multi-axis CNC machines through on-line neurocompensation. *Journal of Intelligent Manufacturing*, *11*, 535–545.

Gandomi, A. H., Yang, X.-S., & Alavi, A. H. (2013). Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems. *Engineering with Computers*, *29*, 17–35.

Hale, L. C. (1999). *Principles and techniques for designing precision machines*. Ph.D. thesis. Livermore: University of California.

Johari, N. F., Zain, A. M., Mustafa, N. H., & Udin, A. (2013). Firefly algorithm for optimization problem. *Applied Mechanics and Materials*, *421*, 496–501.

Kamaruzaman, A. F., Zain, A. M., Yusuf, S. M., & Udin, A. (2013). Levy llight algorithm for optimization problems—A literature review. *Applied Mechanics and Materials*, *421*, 496–501.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of IEEE international conference on neural networks* (Vol. 4, pp. 1942–1948).

Kumar, S., Nassehi, A., Newman, S. T., Allen, R. D., & Tiwari, M. K. (2007). Process control in CNC manufacturing for discrete components: A STEP-NC compliant framework. *Robotics and Computer-Integrated Manufacturing*, *23*, 667–676.

Kusiak, A. (2000). *Computational Intelligence in Design and Manufacturing*. New York: Wiley.

Liu, Z. Q. (1999). Repetitive measurement and compensation to improve workpiece machining accuracy. *International Journal of Advanced Manufacturing Technology*, *15*, 85–89.

Loose, J. P., Zhou, S., & Ceglarek, D. (2007). Kinematic analysis of dimensional variation propagation for multistage machining processes with general fixture layouts. *IEEE Transactions on Automation Science and Engineering*, *4*(2), 141–152.

Mäkelä, K. K., Huapana, J., Kananen, M., Karkalainen, J. A. (2011). Improving accuracy of aging CNC machines without physical changes. In *IEEE international symposium on ISAPT* (pp. 1–5).

Marinescu, V., Constantin, I., Apostu, C., Marin, F. B., Banu, M., & Epureanu, A. (2011). Adaptive dimensional control based on in-cycle geometry monitoring and programming for CNC turning center. *International Journal of Advanced Manufacturing Technolology*, *55*, 1079–1097.

Mehrabi, M. G., O'Neal, G., Min, B. K., Pasek, Z., Koren, Y., & Szuba, P. (2002). Improving machining accuracy in precision line boring. *Journal of Intelligent Manufacturing*, *13*, 379–389.

Mekid, S., & Ogedengbe, T. (2010). A review of machine tool accuracy enhancement through error compensation in serial and parallel kinematic machines. *International Journal of Precision Technology*, *1*(3/4), 251–286.

Mohamad, A., Zain, A. M., Bazin, N. E. N., & Udin, A. (2013). Cuckoo search algorithm for optimization problems—A literature review. *Applied Mechanics and Materials*, *421*, 502–506.

Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm Intelligence*, *1*, 33–57.

Rahou, M., Cheikh, A., & Sebaa, F. (2009). Real time compensation of machining errors for machine tools NC based on systematic dispersion. *Engineering and Technology*, *32*, 10–16.

Rajabioun, R. (2011). Cuckoo optimization algorithm. *Applied Soft Computing*, *11*, 5508–5518.

Ramesh, R., Mannan, M. A., & Poo, A. N. (2000a). Error compensation in machine tools—A review. Part I: Geometric, cutting-force induced and fixture dependent errors. *International Journal of Machine Tools & Manufacture*, *40*, 1257–1284.

Ramesh, R., Mannan, M. A., & Poo, A. N. (2000b). Error compensation in machine tools—A review. Part II: Thermal errors. *International Journal of Machine Tools & Manufacture*, *40*, 1257–1284.

Sartori, S., & Zhang, G. X. (1995). Geometric error measurement and compensation of machines. *Annals of the CIRP*, *44*(2), 599–609.

Schwenke, H., Knapp, W., Haitjema, H., Weckenmann, A., Schmitt, R., & Delbressine, F. (2008). Geometric error measurement and compensation of machines—An update. *CIRP Annals—Manufacturing Technology*, *57*, 660–675.

Slocum, A. H. (1992). *Precision machine design*. Dearborn: Society of Manufacturing Engineers.

Stryczek, R. (2007). Computational intelligence in computer aided process planning—A review. *Advances in Manufacturing Science and Technology*, *31*(4), 77–92.

Stryczek, R. (2009). A meta-heuristics for manufacturing systems optimization. *Advances in Manufacturing Science and Technology*, *33*(2), 23–32.

Wang, S. M., Yu, H. J., & Liao, H. W. (2006). A new high-efficiency error compensation system for CNC multi-axis machine tools. *International Journal of Advanced Manufacturing Technology*, *28*, 518–526.

Yusoff, Y., Ngadiman, M. S., & Zain, A. M. (2011). Overview of NSGA-II for optimizing machining process parameters. *Procedia Engineering*, *15*, 3978–3983.

Yuan, J., & Ni, J. (1998). The real-time error compensation technique for CNC machining systems. *Mechatronics*, *8*(4), 352–380.

Yusup, N., Zain, A. M., & Hashim, S. Z. M. (2012a). Evolutionary techniques in optimizing machining parameters: Review and recent applications (2007-1011). *Expert Systems with Applications*, *39*, 9909–9927.

Yusup, N., Zain, A. M., & Hashim, S. Z. M. (2012b). Overview of PSO for optimizing process parameters of machining. *Procedia Engineering*, *29*, 914–923.

Zain, A. M., Haron, H., & Sharif, S. (2008). An overview of GA technique for surface roughness optimization in milling process. In *Proceedings—International symposium on information technology* (Vol. 4, pp. 1–6).

Zainal, N., Zain, A. M., Radzi, N. H. M., & Udin, A. (2013). Glowworm swarm optimization (GSO) algorithm for optimization problems:

A state-of-the-art. Review. *Applied Mechanics and Materials*, *421*, 507–511.

Zhang, J. Y., Liang, S. Y., Yao, J., Chen, J. M., & Huang, J. L. (2006). Evolutionary optimization of machining process. *Journal of Intelligent Manufacturing*, *17*, 203–2015.

Zhao, F., Xu, X., & Xiev, S. Q. (2009). Computer-aided inspection planning—The state of the art. *Computers in Industry*, *60*, 453–466.

Ziegert, J. C., & Kalle, P. (1994). Error compensation in machine tools: A neural network approach. *Journal of Intelligent Manufacturing*, *5*, 143–151.