BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 69(1), 2021, Article number: e135840
DOI: 10.24425/bpasts.2021.135840

Direct least squares and derivative-free optimisation techniques for determining mine-induced horizontal ground displacement

Janusz RUSEK [1] and Krzysztof TAJDUŚ [2*]

[1] AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Krakow, Poland
[2] Strata Mechanics Research Institute, Polish Academy of Sciences, Reymonta 27, 30-059 Krakow, Poland

Abstract. The paper presents the results of analyses concerning a new approach to approximating trajectory of mining-induced horizontal displacements. Analyses aimed at finding the most effective method of fitting data to the trajectory of mining-induced horizontal displacements. Two variants were made. In the first, the direct least square fitting (DLSF) method was applied based on the minimization of the objective function defined in the form of an algebraic distance. In the second, the effectiveness of differential-free optimization methods (DFO) was verified. As part of this study, the following methods were tested: genetic algorithms (GA), differential evolution (DE) and particle swarm optimization (PSO). The data for the analysis were measurements of on the ground surface caused by the mining progressive work at face no. 698 of the German Prospel-Haniel mine. The results obtained were compared in terms of the fitting quality, the stability of the results and the time needed to carry out the calculations. Finally, it was found that the direct least square fitting (DLSF) approach is the most effective for the analyzed registration data base. In the authors' opinion, this is dictated by the angular range in which the measurements within a given measuring point oscillated.

Key words: horizontal ground displacement; mining; direct least squares; derivative-free Optimisation; genetic algorithms; differential evolution; particle swarm optimization.

1. Introduction

Mining operations carried out underground disturb the original state of stress and strain, which may lead to negative effects on the ground surface in the form of deformations. These can include extensive continuous subsidence trough, local discontinuities in the form of faults, sinkholes, cracks, etc. and ground vibrations resulting from the release of energy during the relaxation of the rock mass layers. The first group of effects is of a continuous and static nature and is described with the use of *deformation indices* (the index of horizontal ground strain ε, the tilt index T, and the curvature radius R). On the other hand, *mining seismic events*, which generate additional inertial forces, are dynamic phenomena and are most commonly described by means of ground vibrations parameters: the *peak ground acceleration* (PGA) and *peak ground velocity* (PGV) [1].

From the standpoint of safety analysis of buildings and structures, these effects, in each of the forms listed, are manifested as kinematic forces on the supports of a structure, causing an additional state of effort in its structural elements and generating deformations of the system as a whole [2–4]. They may thus pose a potential danger with regard to the safety or the usability of building structures in the neighbourhood of mines. For this reason, European mines conduct permanent monitoring of areas subject to the negative impact of mining operations, with regard both to anthropogenic seismic phenomena (mining seismic events – geophysical monitoring) and to continuous and discontinuous ground deformations (*survey monitoring*). The measurement data serve as a basis for the creation, calibration or updating of predictive models of these phenomena, and are used by mines to evaluate the potential impacts of their operations.

The focus of this study is the analysis of a variation in horizontal ground displacements $u(x,y)$ arising as a result of underground mining work.

2. Variation in horizontal ground deformations – a case study

Horizontal displacements are a very important element in the model description of ground deformation. They are particularly significant for the evaluation of dangers to building structures. As a parameter describing a static phenomenon progressing over time and being dependent on the degree of mining extraction, this effect has come to be treated as a stochastic process. Unfortunately, the methods used globally for predicting surface deformations caused by underground mining operations, in both qualitative and quantitative terms, provide a good description of the degree of subsidence only, and are inadequate for the parameter of horizontal displacement [5–7]. This leads to erroneous determination of the index of horizontal deformations of the ground surface. This is an extremely significant problem, since most of the negative effects of mining operations on building structures are caused by horizontal strain of the ground (ε),

*e-mail: tajdus@imgpan.pl

Manuscript submitted 2020-07-27, revised 2020-10-09, initially accepted for publication 2020-10-29, published in February 2021

being partial derivatives of the horizontal displacement function $u(x,y)$. For this reason, the authors consider it to be of great importance to address the problems relating to the real values of the field of horizontal displacements and to attempt to give an individual description of that field.

2.1. Case study. Observations of changes in the pattern of horizontal displacements and their values for sample underground mines [5,9] have revealed a need for further study of the formation of the field of horizontal displacements in the region of such mines. The authors performed analyses for a sample German underground mine – *Prospel–Haniel* [8]. In the region analysed, mining operations were carried out at wall *no*. 698 in the seam O/N at the average depth of $H = 960$ m. The length of the production face was 270 m, panel length 970 m and the height – from 3.6 m to 4.3 m [7]. The seam was dipping towards the south-west at the angle of 7°. On the surface, the competent services made measurements of both subsidence and horizontal displacements for a database covering 48 measurement points using the GPS satellite technology . The locations of the individual measurement points and the longwall panel face advance in relation to measurements time (from *no*.0 to *no*.17) are presented in Fig. 1. Exploitation in longwall panel started in early May 1999 and was completed by the end of November 1999, while the measurements started in early April 1999 and end in January 2000.

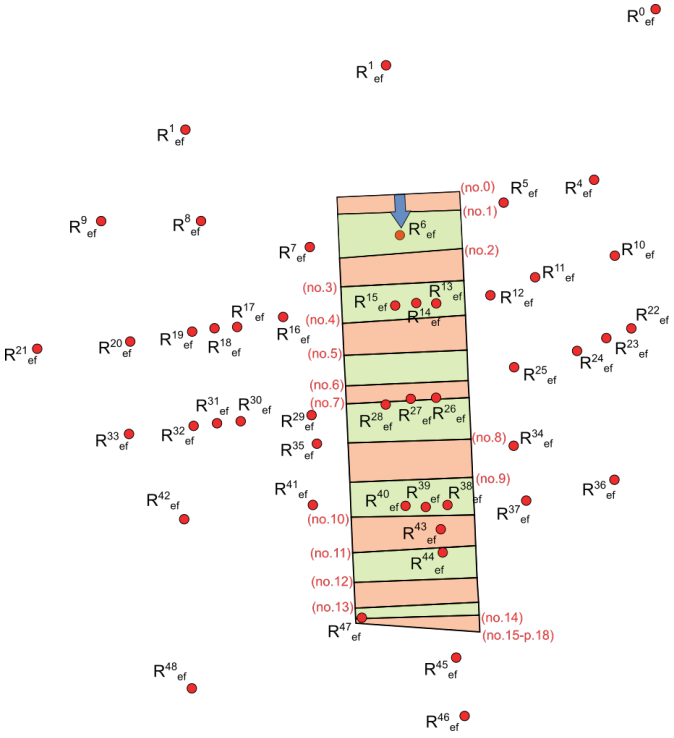


Fig. 1. Location of observation points with respect to wall no. 698 at the Prospel–Haniel mine [8]

Initial analysis showed a certain systematic regularity in the distribution of the resultant vector of horizontal ground displacements $u(x,y)$ registered at all measurement points. Regularity was detected in the trajectory traced by successive measured vectors of horizontal displacements relative to the stabilised reference points $R_{ef}^{(j)}$ as the worked face (wall *no*. 698) progressed (Figs. 2 and 3).

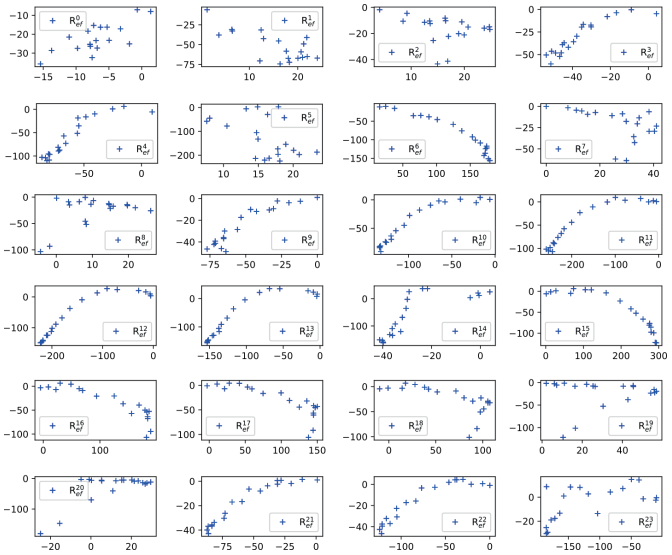


Fig. 2. Graphical presentation of registered values of component horizontal displacements [mm] at all measurement points (from R_{ef}^{0} to R_{ef}^{22})

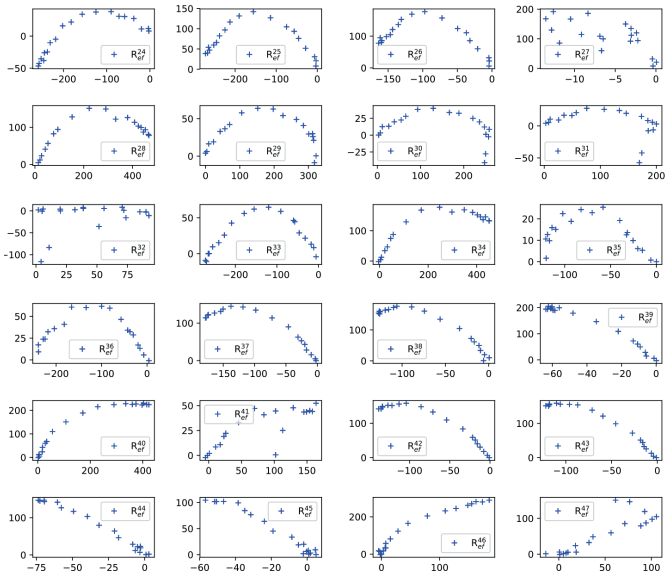


Fig. 3. Graphical presentation of registered values of component horizontal displacements [mm] at all measurement points (from R_{ef}^{23} to R_{ef}^{47})

In the great majority of cases, this trajectory was close to an elliptical shape. This fact suggested to the authors the possibility of approximating the registered displacements $u(x,y)$ with elliptical functions. This facilitated making a precise statement of an initial thesis for further research, namely that the field of horizontal displacements $u(x,y)$ may be described in a domain of parameters corresponding to the distance of longwall panel

run, the depth of mining, the thickness of the seam, and the relative location of the point on the ground surface with respect to the outline of the excavations.

3. Research methodology

Algorithms for the fitting of elliptical functions are used in an extremely wide range of fields. Currently, they are predominantly used for pattern recognition. This includes both static and dynamic problems in terms of detection of objects in real time. In static problems the main criterion is the accuracy of fitting and the effectiveness in terms of the number of correctly detected objects based on analysis of photographs [10]. In dynamic problems, there is an additional important requirement relating to the algorithm's speed. This is a key parameter that determines the effectiveness of a given approach for permanent monitoring of a group of objects in real time [11]. A broader description of applications may be found in, e.g. [12, 13].

The addressed problem relating to the fitting of elliptical functions to measured component values of the field of horizontal ground displacements (u_x, u_y) was considered to be a static problem. This view is clearly supported by the rate at which these effects appear at the ground surface. Hence, in seeking a suitable approximation algorithm, the sole fundamental criterion considered was the accuracy of fit of an elliptical function to the registered data, not the real-time speed of the algorithm (Fig. 4).

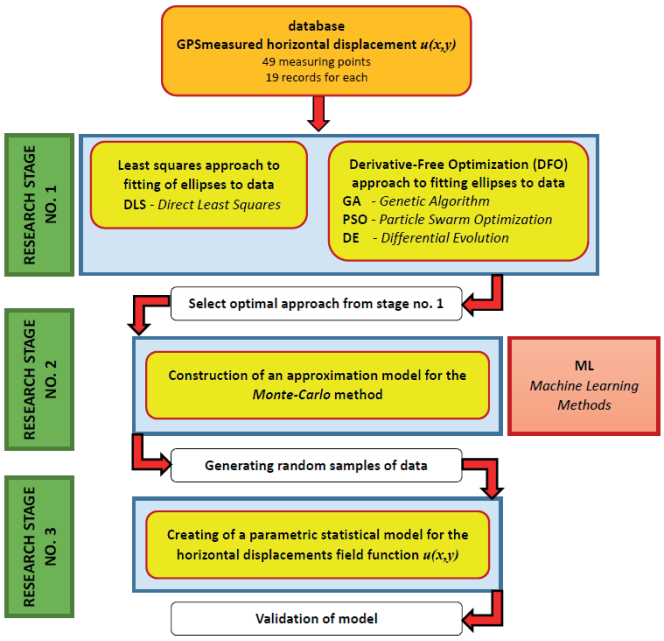


Fig. 4. Schematic diagram showing the scope of the analyses described in this article (stage 1) and further planned research (stages 2 and 3)

Based on the survey of available algorithms given in [14], two main approaches were selected for comparison. The first of these is based on the use of the *algebraic error* in the minimisation process. Here the *direct least squares fitting* (DLSF) approach was applied, as described in [15]. In the second case, inspired by [16], the objective function used for the minimisation process was a measure of the *geometric error*. For this approach, apart from the *genetic algorithm* (GA) proposed in [16], the tests also included the method known as *particle swarm optimisation* (PSO) [17–20] and the *differential evolution* (DE) algorithm [21]. All of these approaches are classified as *derivative-free optimisation* (DFO) methods [22], which avoid stacking the optimisation process at a local minimum, thereby increasing the chance of attaining a global minimum. The choice of the aforementioned optimisation methods was partly motivated by the authors' experiences with the use of *genetic algorithms* (GA) for the problem of optimum selection of hyperparameters for the *support vector machine* (SVM) method from a regression standpoint [23, 24].

All analyses were performed in a *Python* language environment. The following libraries were used in the computations: *distributed evolutionary algorithms in Python* (DEAP) [25, 26], *Mystic* [27, 28], and *least squares fitting of ellipses – Python* routine [29] based on [15].

3.1. Direct least squares fitting approach to approximating ellipses to data. The study used the *direct least squares fitting* (DLSF) procedure described in [15]. This is a modification of the procedure determined by *Fitzgibbon* et al. [30].

Consideration is first given to a general quadratic form representing an arbitrary conic:

$$F(x,y) = x^2 + bxy + cy^2 + dx + ey + f = 0, \tag{1}$$

where a, b, c, d, e, f are the coefficients of the quadratic form representing a conic, and (x, y) are the coordinates of a point lying on the conic.

To ensure that Eq. (1) describes an ellipse, the following condition must be imposed on the coefficients b, a and c:

$$b^2 - 4ac < 0\,. \tag{2}$$

In this formulation, the function $F(x,y)$ given by Eq. (1) subject to condition (2) is an ellipse. Moreover, the function $F(x,y)$ represents what is called an algebraic distance, such that $F(x,y) = 0$ for points with coordinates lying on the ellipse, and $F(x,y) \neq 0$ for other points. Hence, in the procedure described, this function is treated as an error, and its square value serves as the objective function for minimisation by the *Least Squares* method.

Following [15], it is possible to group the coefficients and coordinates appearing in Eq. (5) and to present them in the following disjoint vector form:

$$\boldsymbol{a} = \left[a, b, c, d, e, f\right]^T, \tag{3}$$

$$\boldsymbol{x} = \left[x^2, xy, y^2, x, y, 1\right]. \tag{4}$$

Thus, Eq. (1) may be written in vector form as:

$$F_{\boldsymbol{a}}(\boldsymbol{x}) = \boldsymbol{x} \cdot \boldsymbol{a} = 0\,. \tag{5}$$

Next, having the set of data (x_i, y_i), $i = 1, \ldots, N$, the general criterion for the minimisation procedure may be formulated as:

$$\min_a \sum_{i=1}^{n} F(x_i, y_i)^2 = \min_a \sum_{i=1}^{n} F_{\boldsymbol{a}}(\boldsymbol{x}_i)^2 = \\ = \min_a \sum_{i=1}^{n} (\boldsymbol{x}_i \cdot \boldsymbol{a})^2. \tag{6}$$

To ensure that the parameters (3) determined by the minimisation process satisfy the requirement of (2) – that is, to define a special case of a conic section strictly for an ellipse – the domain of considered parameters must be extended to include *Lagrange multipliers*, $\lambda \neq 0$, using a typical approach as for minimisation problems with constraints. We thus obtain a more specific form of the criterion function, in the form of a *Lagrangian*:

$$L(\boldsymbol{x}_i, \boldsymbol{a}, \boldsymbol{\lambda}) = \sum_{i=1}^{n} (\boldsymbol{x}_i \cdot \boldsymbol{a})^2 - \boldsymbol{\lambda} \left(\boldsymbol{a}^T \boldsymbol{C} \boldsymbol{a} - 1 \right), \tag{7}$$

where:

$$\boldsymbol{C} = \begin{pmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

is the constraint matrix enforcing condition (6) for the vector representation of coefficients a (2) of the quadratic form (1).

Equation (7) may be reduced, with respect to the data (x_i, y_i), $i = 1, \ldots, N$, to the form:

$$L(\boldsymbol{x}_i, \boldsymbol{a}, \boldsymbol{\lambda}) = \|\boldsymbol{D} \cdot \boldsymbol{a}\|^2 - \boldsymbol{\lambda} \left(\boldsymbol{a}^T \boldsymbol{C} \boldsymbol{a} - 1 \right), \tag{8}$$

where

$$\boldsymbol{D} = \begin{pmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i^2 & x_i y_i & y_i^2 & x_i & y_i & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^2 & x_N y_N & y_N^2 & x_N & y_N & 1 \end{pmatrix}$$

is a *design matrix* with dimensions $N \times 6$.

Determining the components of the gradient for the *Lagrangian* (8) with respect to the sought parameters of vector a and the *Lagrange coefficients* λ, we obtain the following set of equations:

$$\nabla_{\boldsymbol{a},\boldsymbol{\lambda}} L(\boldsymbol{x}_i, \boldsymbol{a}, \boldsymbol{\lambda}) = \left\{ \frac{\partial L}{\partial \boldsymbol{a}}, \frac{\partial L}{\partial \boldsymbol{\lambda}} \right\} = \left\{ \begin{array}{ll} \boldsymbol{S}\boldsymbol{a} = \boldsymbol{\lambda}\boldsymbol{C}\boldsymbol{a} & a) \\ \boldsymbol{a}^T \boldsymbol{C} \boldsymbol{a} = 1 & b) \end{array} \right\}, \tag{9}$$

where

$$S = \boldsymbol{D}^T \boldsymbol{D} = \begin{pmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i^2 & x_i y_i & y_i^2 & x_i & y_i & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^2 & x_N y_N & y_N^2 & x_N & y_N & 1 \end{pmatrix}$$

is a 6×6 *scatter matrix*.

In the original approach proposed in [30], the next stage is to solve the generalised eigenproblem (9a), leading to a set of six solutions $\{\lambda_j,\ a_j\}$. The decision on the final solution is made based on the obtained value of λ_j. From the equations of the originally formulated problem (6), using the transformations appearing in the formulae (8) and (9a, b), one obtains a reduced form supporting the choice of a minimum value of λ from the set of all values obtained when solving the generalised eigenproblem (9a). These transformations take the following form:

$$\min_{\boldsymbol{a}} \sum_{i=1}^{n} (\boldsymbol{x}_i \cdot \boldsymbol{a})^2 = \min_{\boldsymbol{a}} \|\boldsymbol{D} \cdot \boldsymbol{a}\|^2 = \min_{\boldsymbol{a}} \boldsymbol{a}^T \boldsymbol{D}^T \boldsymbol{D} \boldsymbol{a} = \\ = \min_{\boldsymbol{a}} \boldsymbol{a}^T \boldsymbol{S} \boldsymbol{a} = \min_{\boldsymbol{a},\boldsymbol{\lambda}} \boldsymbol{\lambda} \boldsymbol{a}^T \boldsymbol{C} \boldsymbol{a} = \min_{\boldsymbol{\lambda}} \boldsymbol{\lambda}. \tag{10}$$

However, in [15] an analysis was made of the original procedure and a modification was proposed. The reasoning given referred to the instability of the solution proposed in [30]: in view of the singularity of the matrix $\boldsymbol{C}$ and the near-singularity of the matrix $\boldsymbol{S}$, inappropriate results are generated (infinite values or complex numbers). It was also shown that the claim supported by Eq. (14), indicating the existence of only one solution corresponding to the minimum eigenvalue λ, is erroneous.

To eliminate from the original formulation the singularity of the matrices $\boldsymbol{C}$ and $\boldsymbol{S}$, which results from the introduction of the matrix $\boldsymbol{D}$ (9), it is proposed in [15] to divide the matrix $\boldsymbol{D}$ into two parts: the first containing all linear elements, and the other all second-order elements. That matrix finally takes the form:

$$\boldsymbol{D} = (\boldsymbol{D}_1 | \boldsymbol{D}_2), \tag{11}$$

where

$$\boldsymbol{D}_1 = \begin{pmatrix} x_1^2 & x_1 y_1 & y_1^2 \\ \vdots & \vdots & \vdots \\ x_i^2 & x_i y_i & y_i^2 \\ \vdots & \vdots & \vdots \\ x_N^2 & x_N y_N & y_N^2 \end{pmatrix}$$

is the component matrix following decomposition, with second-order elements,

$$\boldsymbol{D}_2 = \begin{pmatrix} x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots \\ x_i & y_i & 1 \\ \vdots & \vdots & \vdots \\ x_N^2 & y_N & 1 \end{pmatrix}$$

is the component matrix following decomposition, with linear elements.

This operation results in a division of the matrix $\boldsymbol{S}$, in correspondence with the separated parts of the matrix $\boldsymbol{D}$ (12). A similar procedure is performed for the matrix $\boldsymbol{C}$ (13) and the vector of parameters a (14).

$$\boldsymbol{S} = \begin{pmatrix} \boldsymbol{S}_1 & \boldsymbol{S}_2 \\ \boldsymbol{S}_2^T & \boldsymbol{S}_3 \end{pmatrix}, \tag{12}$$

where

$$\boldsymbol{S}_1 = \boldsymbol{D}_1^T \boldsymbol{D}_1,$$
$$\boldsymbol{S}_2 = \boldsymbol{D}_1^T \boldsymbol{D}_2,$$
$$\boldsymbol{S}_3 = \boldsymbol{D}_2^T \boldsymbol{D}_2,$$

$$\boldsymbol{C} = \begin{pmatrix} \boldsymbol{C}_1 & 0 \\ 0 & 0 \end{pmatrix}, \qquad \boldsymbol{C}_1 = \begin{pmatrix} 1 & 2 \\ -2 & 0 \end{pmatrix}, \tag{13}$$

$$\boldsymbol{a} = \begin{pmatrix} \boldsymbol{a}_1 \\ \boldsymbol{a}_2 \end{pmatrix}, \qquad \boldsymbol{a}_1 = \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \qquad \boldsymbol{a}_1 = \begin{pmatrix} d \\ e \\ f \end{pmatrix}. \tag{14}$$

By making the above modifications, the final form of Eq. (9a) is obtained in the following block representation:

$$\begin{pmatrix} \boldsymbol{S}_1 & \boldsymbol{S}_2 \\ \boldsymbol{S}_2^T & \boldsymbol{S}_3 \end{pmatrix} \begin{pmatrix} \boldsymbol{a}_1 \\ \boldsymbol{a}_2 \end{pmatrix} = \lambda \begin{pmatrix} \boldsymbol{C}_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{a}_1 \\ \boldsymbol{a}_2 \end{pmatrix}. \tag{15}$$

The matrix system in the form (15) may be presented as:

$$\begin{cases} \boldsymbol{S}_1 \boldsymbol{a}_1 + \boldsymbol{S}_2 \boldsymbol{a}_2 = \lambda \boldsymbol{C}_1 \boldsymbol{a}_1, \\ \boldsymbol{S}_2^T \boldsymbol{a}_1 + \boldsymbol{S}_3 \boldsymbol{a}_2 = 0. \end{cases} \tag{16}$$

By making further transformations, following [15], one may obtain the following set of equations:

$$\begin{cases} \boldsymbol{M}\boldsymbol{a}_1 = \lambda \boldsymbol{a}_1 \\ \boldsymbol{a}_1^T \boldsymbol{C}_1 \boldsymbol{a}_1 = 1 \\ \boldsymbol{a}_2 = -\boldsymbol{S}_3^{-1} \boldsymbol{S}_2^T \boldsymbol{a}_1 \end{cases}, \tag{17}$$

where after the transformations $\boldsymbol{M} = \boldsymbol{C}_1^{-1} \left(\boldsymbol{S}_1 - \boldsymbol{S}_2 \boldsymbol{S}_3^{-1} \boldsymbol{S}_2^T \right)$ is the reduced scatter matrix.

The system of algebraic equations in the form (17) is equivalent to that of (9). The solution is thus a vector $_1$ resulting from the solution of the eigenproblem and satisfying the algebraic equality constraints in (17). The procedure proposed in [15] leads to a unique solution in a single stage, and does not require an iterative approach based on the minimisation of a specified objective function.

The results of a fitting process with the use of the proposed procedure are also given in [15], serving to prove the correctness of the adopted solution. These results, illustrating the fit of an ellipse to the generated points, were key in the choice of this methodology for further study. This concerns particularly those cases where there was shown to be an agreement in ellipse determination for points distributed on the edge of ellipses contained in the range $0^\circ \leq \varphi \leq 60^\circ$. This was highly significant because the registered data relating to the components of the field of displacements $u(x,y)$, corresponding to successive phases of ground deformation related to the progress of mine working, often indicated only a section of the edge of a potential ellipse, and did not cover an angle greater than $\varphi = 180^\circ$ (Figs. 2 and 3). Hence, in the authors' view, this method may be an effective tool for analysing the field of horizontal displacements $u(x,y)$ at an early stage of the appearance of effects on the ground surface resulting from underground mining operations.

3.2. Derivative-free optimisation (DFO) approach to fitting ellipses to data. Apart from the fitting of ellipses based on the least squares method, an approach in which the minimisation of the criterion function was based on *derivative-free optimisation* (DFO) was also tested. One of the reasons for the choice of this approach was the experience of the authors in applying DFO as a *meta-heuristics* for the calibration of hyperparameters in SVM models, in connection with the protection of mining areas development [3,24]. A second reason was an apparent analogy between the problem at hand and the approach described in [16]. In that work, a comparison was made between the results of applying the least squares method and optimisation using genetic algorithms in a problem involving the fitting of ellipses to data. However, it was noted that there was a dissimilarity regarding the data. The analyses in [16] were based on the data whose radial spread covered a full angle of $\varphi = 360^\circ$. On the other hand, the problem addressed in the present work concerns the fitting of ellipses to data spread over an angular range that frequently does not exceed a value of $\varphi = 180^\circ$.

Hence, with a view to expanding the research methodology, it was decided to investigate whether the fitting of ellipses using *derivative-free optimisation* (DFO) at an early stage of the development of the field of horizontal ground displacements (u_x, u_y) caused by underground mining operations is more effective than the least squares approach.

Finally, having available literature data relating to the use of optimisation algorithms classified as *DFO* methods in problems involving the fitting of ellipses to data [31,32], it was decided to further investigate the use of *genetic algorithms* (GA), *differential evolution* (DE) and *particle swarm optimisation* (PSO).

In contrast to the method of least squares, where the function subject to minimisation was an *algebraic distance* [15], in analyses using DFO methods the criterion function is the *geometric distance* (F_i) between the fitted ellipse and the recorded data (Fig. 5).

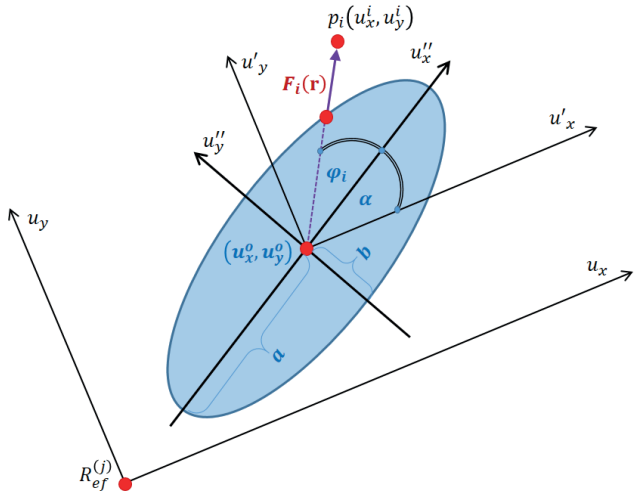


Fig. 5. Graphical interpretation of the criterion function for an optimisation process, in the form of a geometric distance between the point p_i and the fitted ellipse in the system related to the location of the j-th measurement position $R_{ef}^{(j)}$

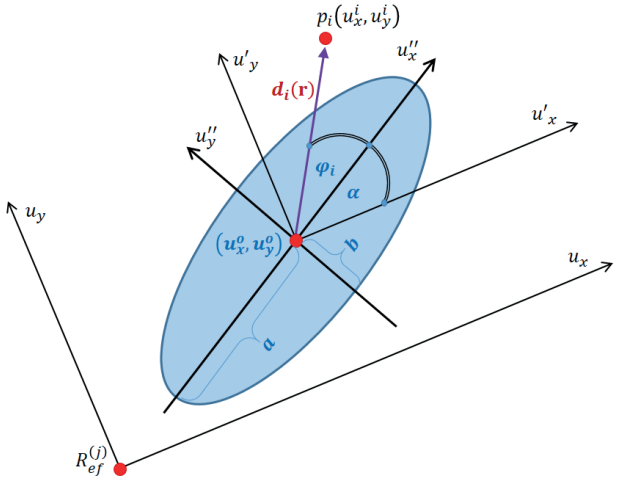


Fig. 6. Graphical interpretation of the function $d_i(r)$

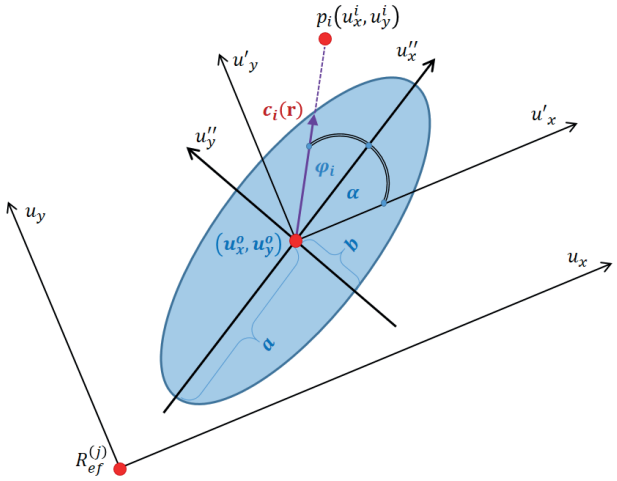


Fig. 7. Graphical interpretation of the function $c_i(r)$

The geometric distance (F_i) is determined beginning with the parametric description of an ellipse in the form [16]:

$$\begin{pmatrix} u_x \\ u_y \end{pmatrix} = \begin{pmatrix} u_x^o \\ u_y^o \end{pmatrix} + \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} a\cos\alpha \\ b\sin\alpha \end{pmatrix}, \quad (18)$$

where u_x is the x axis of the system determined by the location of the measurement point $R_{ef}^{(j)}$; u_y is the y axis of the system determined by the location of the measurement point $R_{ef}^{(j)}$; u_x^o is the x coordinate of the centre of the ellipse in the system (u_x, u_y); and u_y^o is the y coordinate of the centre of the ellipse in the system (u_x, u_y).

Taking the vector r of target parameters of the ellipse in the form (19), as in [16], the geometric distance may be presented as a function of the vector r and written in the form (20):

$$\mathbf{r} = \left(a, b, u_x^o, u_y^o, \alpha\right)^T, \quad (19)$$

$$F_i(\mathbf{r}) = \|d_i(\mathbf{r}) - c_i(\mathbf{r})\|, \quad (20)$$

where

$d_i(\mathbf{r}) = \left\| \begin{pmatrix} u_x^i - u_x^o \\ u_y^i - u_y^o \end{pmatrix} \right\|$ – for interpretation see Fig. 6,

$c_i(\mathbf{r}) = \left\| \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} a\cos\varphi_i \\ b\sin\varphi_i \end{pmatrix} \right\|$ – for interpretation see Fig. 7.

Finally, considering all the recorded data (m) for the measurement points $R_{ef}^{(j)}$, the criterion function for minimisation may be written in the form:

$$\mathbf{G}(\mathbf{r}) = \sum_{i=1}^{m} F_i(\mathbf{r})^2 \to \min. \quad (21)$$

For each of the implemented algorithms, an upper bound was determined for the ellipse radii (a and b), using information contained in the data on the registered values of components of the field of displacements (u_x, u_y). For each set of data corresponding to a given registering location, the radii of curvature R were estimated according to Eq. (22) and the scheme shown in Fig. 8 For this purpose, the equations of the *finite differential method* were adapted to the second derivative according to [33, 34]. Considering n registrations of components of the field of horizontal displacements (u_x, u_y) determined for a given measurement position, a fit of the radius of curvature R was performed for each triple of consecutive points (i, $i+1$, $i+2$). In each case this gave $n-2$ estimated values of R. In view of the different values of the ellipse radii (a and b), from the resulting set of estimated radii of curvature $\{R_1, R_2, \ldots, R_{n-2}\}$ the largest value was selected as the upper bound for the fitted ellipse radii (a and b) in the optimisation process.

$$\left.\frac{\mathrm{d}^2 u_y^t}{\mathrm{d} u_x^{t\,2}}\right|_{p_{i+1}} \approx \frac{-2\,u_y^t\big|_{p_{i+1}}}{L_1 L_2} \approx \frac{1}{R_{i+1}}, \quad (22)$$

where after the transformations:

u_x^t – temporal x coordinate according to Fig. 8;

u_y^t – temporal y coordinate according to Fig. 8;

p_{i+1} – $(i+1)$-th point from the set of records of components of the field of displacements $(u_x,\ u_y)$ for the measurement point $R_{ef}^{(j)}$;

L_1, L_2 – lengths of segments determined according to the adopted differential scheme (cf. Fig. 8);

R_{i+1} – the estimated radius of curvature at p_{i+1}.

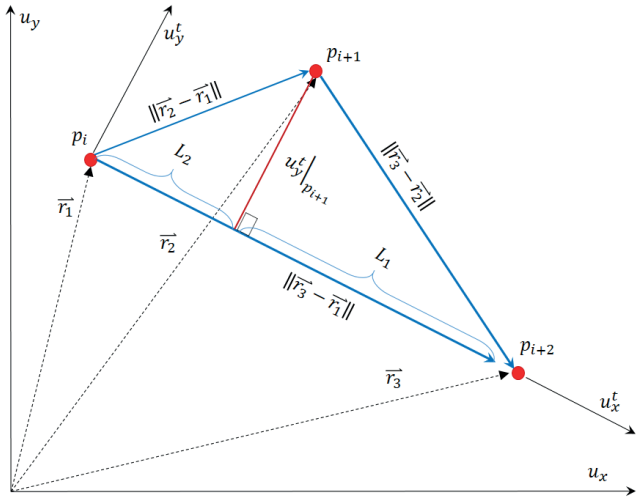


Fig. 8. Graphical interpretation of the function $d_i(r)$

Genetic algorithms (**GA**) belong to the class of *derivative-free optimisation* (DFO) methods. In general, they involve iterative searching of a space of potential solutions with the aim of satisfying a set optimisation criterion. The number of iterations is predetermined and is referred to as the number of *generations* (n_{gen}). The values of the parameters subject to optimisation are coded and represented in zero-one notation. The numerical form coded in this way is called a *chromosome*, and the individual components of such a sequence are treated as *genes*. In this way, *crossover* and *mutation* operations can be performed. These operations are the basis for the action of the genetic algorithm and are performed on elements of a pre-generated set of potential solutions, known as the *population*. Within each generation, the products of *crossover* and *mutation* operations are selected according to a preselected optimisation criterion [35]. In the subsequent generation, this leads to the selection of a new *population* to which analogous transformations are applied until the value of the criterion function reaches an acceptable level or the predefined limit on the number of generations.

In addition, depending on the formulation of the optimisation problem, genetic algorithms may be applied to problems which contain one criterion function (*single-objective optimisation problems*) or many equivalent functional criteria (*multi-criterial optimisation problems* or *multi-objective optimisation problems*).

The analysis was carried out in a *Python* language environment using the *distributed evolutionary algorithms in Python* (DEAP) library [25, 26].

The parameters being found were the ellipse radii (a, b), the coordinates of the centre of the ellipse $\left(u_x^o, u_y^o\right)$ in a local system with respect to the coordinates of a given measurement position, and the angle α of rotation of the ellipse about its centre (Figs. 5–7).

As noted at the beginning of Section 4, the values of the ellipse radii (a and b) were bounded above by the value of the maximum radius of curvature R resulting from the analysis of the recorded component displacements (u_x, u_y) at a given measurement position. The remaining parameters – the coordinates of the centre of the fitted ellipse $\left(u_x^o, u_y^o\right)$ and the angle α – were not subject to any constraining conditions.

The fact of the bounding of the ellipse radii (a and b) was a criterion determining the choice of appropriate *crossover* and *mutation* operations for the genetic algorithm because it was in the operation of these processes, these were considered during the optimisation. For the crossover operation the method of *simulated binary crossover* with constraints was selected, and for the mutation procedure the method of *polynomial mutation* with constraints. Both methods are implemented in the *DEAP* library [25,26] and are analogous to methods of the widely used *NSGA-II* (*non-dominated sorted genetic algorithm*) [35].

Differential evolution (**DE**) method is classified as an *evolutionary optimisation algorithm* (EOA) [36]. From a procedural standpoint, this approach is very similar to the optimisation typical of genetic algorithms (GA). Nonetheless, there are cases reported in the literature which indicate the advantage of DE over other *gradient-free optimisation* methods, including GA [21, 36–38].

The most important difference between DE and GA is the way in which the values of the optimised variables (parameters) are represented. In the case of genetic algorithms, the values of the parameters are usually coded as sequences of zeros and ones. In the DE method, however, all operations are performed on real values. This fundamental difference means that DE is well suited to the solution of problems in which optimisation takes place with respect to parameters belonging to the domain of real numbers.

At the first stage, as in the case of genetic algorithms, the initial *population* is determined as a set of N vectors, representing the list of parameters subject to optimisation. The particular values of these parameters for each vector in the initial population are initialised in a random manner, with the possibility of simultaneously taking account of lower and upper bounds on the permissible values of each of them [21].

$$\begin{cases} \boldsymbol{P}_0 = \boldsymbol{X}_{i,0} \\ \boldsymbol{X}_{i,0} = x_{j,i,0} \\ i = 1,\ldots,NP \quad j = 1,\ldots,D \end{cases}, \tag{23}$$

where $\boldsymbol{P}_0$ is the initial population; $\boldsymbol{X}_{i,0}$ is the i-th vector of parameters in the initial population; $x_{j,i,0}$ is the j-th component of the i-th vector of parameters in the initial population.

The next stage involves a simultaneous operation of *mutation* and *crossover*, based on the previously selected set of the initial population P_0. This is a self-referential operation, and applies

to each of the vectors $X_{i,0}$. It entails the modification of the particular components of the vectors of the initial population P_0 according to Eq. (24). A change is made to the component $x_{j,i,0}$ of vector $X_{i,0}$, where information may be exchanged with randomly selected corresponding components $\{x_{j,r1,0}, x_{j,r2,0}, x_{j,r3,0}\}$ of three other vectors $X_{r1,0}$ $X_{r2,0}$ $X_{r3,0}$ from the initial population P_0. The occurrence of an event leading to the exchange of information, and in consequence an operation of *mutation* and *crossover*, is determined by arbitrarily chosen values of the control parameters F and $\boldsymbol{CR}$ [21]. In effect, after such a procedure has been carried out for all vectors of the initial *population*, a new set is obtained, constituting a collection of candidates for the *population* in the next *generation*: $P_0' = U_{i,0} = u_{j,i,0}$. The effectiveness of the modified initial *population*, and thus the decision concerning its use in a subsequent iteration is verified at the stage of *selection*. For this purpose, the value of the adopted criterion function $f(X)$ is checked, using Eq. (25).

The process thus formulated is repeated until an acceptable value of the minimised criterion function $f(X)$ is attained or the predetermined number of *generations* is reached.

$$u_{j,i,0} = \begin{cases} x_{j,r3,0} + \boldsymbol{F}\left(x_{j,r1,0} - x_{j,r2,0}\right) & \\ \qquad\qquad \text{rand}_j\langle 0,1) \leq \boldsymbol{CR} \cup j = \boldsymbol{k} & , \\ x_{j,i,0} \qquad\quad \text{otherwise} & \end{cases} \tag{24}$$

where $\boldsymbol{F}$ is an arbitrarily chosen control parameter $F \in (0,1\rangle$; $\boldsymbol{CR}$ is an arbitrarily chosen control parameter $\boldsymbol{CR} \in \langle 0,1\rangle$; k is a random parameter from the set $\{1,...,D\}$ selected randomly at each iteration; r_1, r_2, r_3 are random values from the set $\{1,\ldots,NP\}$ excluding the currently considered vector $X_{i,0}$: $r_1 \neq r_2 \neq r_3 \neq i$.

$$\boldsymbol{X}_{i,0+1} = \begin{cases} \boldsymbol{U}_{i,0} & \text{if} \quad f(\boldsymbol{U}_{i,0}) \leq f(\boldsymbol{X}_{i,0}) \\ \boldsymbol{X}_{i,0} & \text{else} \end{cases}, \tag{25}$$

where $U_{i,0} = u_{j,i,0}$ is the i-th vector modified at the mutation and crossover stage according to (23); $X_{i,0} = x_{j,i,0}$ is the i-th vector in the initial population P_0; $X_{i,0+1}$ is the i-th vector of the population for the subsequent iteration.

Particle swarm optimisation (**PSO**) is very widely applied in various branches of science and engineering, including electronics, biomedicine, transport network design, data classification and clustering problems, the design of fuzzy and neuro-fuzzy controllers, and many others [17].

Optimisation based on a *swarm of particles* was defined in [17]. The inspiration behind it were the processes taking place among herd animals in a natural environment. In essence, the proposed method provides a procedural implementation of the emergent phenomenon of collective intelligence displayed in the behaviours of shoals of fish or flocks of birds [39]. Scientists also use the term *swarm intelligence* [40].

Procedurally, the operation of the PSO method entails the definition of a certain objective function in the space of the parameters subject to optimisation. In that space a certain number of *particles* (also called *agents* [39]) is placed with random coordinates. The value of the objective function is then computed at the *particles'* locations. After being initiated in this way, the procedure moves to the next step – the characteristic feature of the PSO method – in which an adjustment is made to the initial coordinates for each *particle* in the *swarm*. Following [41], this is done according to Eq. (26):

$$x_i(t+1) = x_i(t) + v_i(t+1), \tag{26}$$

where $x_i(t)$ is the i-th component of the position of the particle at time $t, i = 1,\ldots,N$; $x_i(t+1)$ is the adjusted i-th component of the position of the particle at time $t+1$; $i = 1,\ldots,N$; $v_i(t+1)$ is the i-th component of the particle velocity at time $t+1$, $i = 1,\ldots,N$.

$$\begin{aligned} v_i(t+1) = \omega \cdot v_i(t) + \psi_1 R_1 \left[x_{si} - x_i(t)\right] + \\ + \psi_2 R_2 \left[x_{pi} - x_i(t)\right], \end{aligned} \tag{27}$$

where x_{si} is the i-th component of the position of the particle from the neighbourhood of the point $x(t)$ for which the best value of the criterion function $f(x)$ was obtained according to the adopted optimisation condition (*min*/*max*) in all previous iterations; x_{pi} is the i-th component of the position of the analysed *particle* $x(t)$ for which the best value of the criterion function $f(x)$ was obtained according to the adopted optimisation condition (*min*/*max*) in all previous iterations; R_1, R_2 are independent parameters with random values in the interval $[0, 1]$; ω is the inertia weight constant; Ψ_1, Ψ_2 are constants known as *acceleration coefficients*, regulating the level of individual cognition and interaction between surrounding *swarm particles*; $v_i(t+1)$ is the i-th component of the particle velocity at time $t+1$, $i = 1,\ldots,N$.

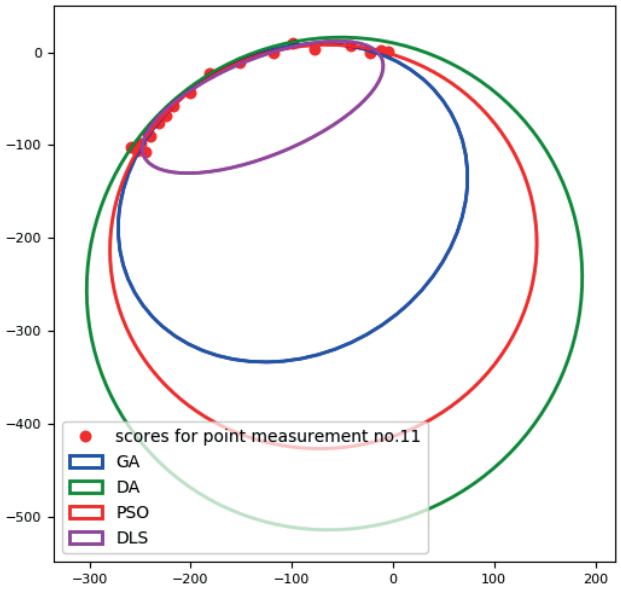
4. Results of analysis

The analyses were based on a database of registered GPS values of horizontal ground displacements $u(x,y)$ for 48 points $R_{ef}^{(j)}$. The number of records in the components of horizontal displacements $u(x,y)$ at each reference point $R_{ef}^{(j)}$ was $n_R = 19$. Finally, for each reference point, ellipses were fitted to the recorded data in the manner described in Section 4. The fitting process involved determining the optimum parameters describing the ellipse, namely $\{a, b, u_x^o, u_x^o, \alpha\}$ – cf. Fig. 5.

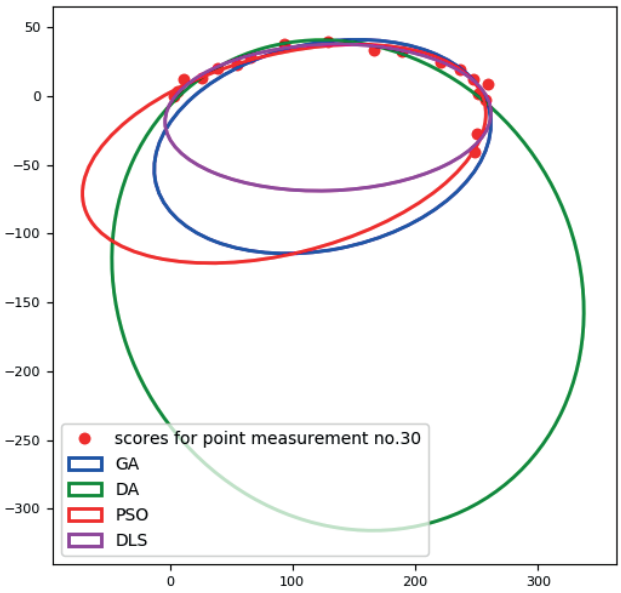
In the *direct least squares fitting* (DLSF) approach, in accordance with [16], optimisation without constraints was applied, with the objective function taken to be the sum of the squares of geometric distances, given by Eq. (21). Preliminary analysis showed that when ellipses were fitted to data spread over an angle no greater than $\varphi = 180^\circ$, the optimisation process was unbounded, and thus lacked uniqueness. All DFO algorithms led to unreasonably large values of the parameters describing the ellipse semi-axes (a, b) (the smaller the error tolerance used in the optimisation, the larger were these values). To stabilise the optimisation for DFO algorithms, the information contained in the data was used, and on that basis, in accordance with Eq. (22) and Fig. 8, the upper limits were estimated for the values of

the parameters $\{a,b\}$. These values resulted from the analysis of the curvatures in the database of registrations, performed for each reference point $R_{ef}^{(j)}$. The final value to be used in the calculations was taken to be the maximum radius of the fitted curve for the measurement data. As a result of this operation, an optimisation with constraints was determined for each applied algorithm from the DFO family. This made it possible to obtain results that could be interpreted in accordance with the predictions resulting from the physics of the analysed process.

Fig. 10 shows overall results in the form of fitted ellipses for all of the algorithms analysed. Fig. 9 presents some of them so as to provide more detail (these are results for the measurement positions: $R_{ef}^{(11)}$ and $R_{ef}^{(30)}$.



(a)



(b)

Fig. 9. Selected sample results for positions: (a) $R_{ef}^{(11)}$; (b) $R_{ef}^{(30)}$

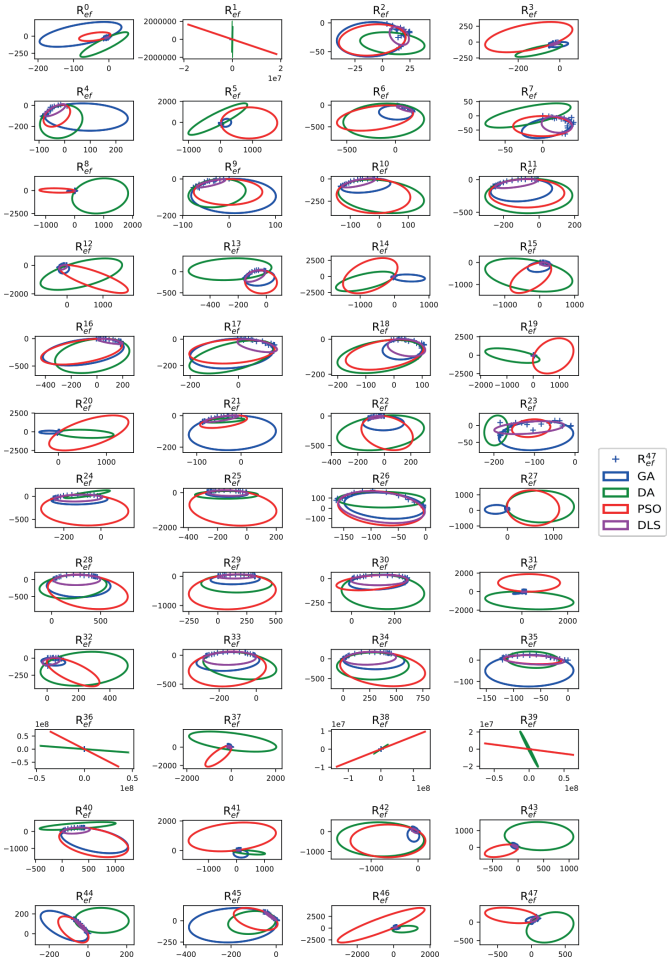


Fig. 10. Graphical presentation of the results of fitting by the methods GA, DE, PSO and DLSF for data from all measurement points $R_{ef}^{(j)}$

As may be observed, the DLSF algorithm leads to the ellipses with the smallest semi-axis values a and b. Moreover, the centre coordinates for particular ellipses fitted by means of this algorithm are located closest to the data registered at all positions $R_{ef}^{(j)}$. The algorithm itself is numerically stable and unique. It also has the shortest running time, which in the case analysed amounted to approximately 5 minutes (the total computation time for the fitting of ellipses for all measurement positions $R_{ef}^{(j)}$.

The results obtained with the use of genetic algorithms (GA), as with the other DFO algorithms analysed, required significantly greater computational resources, as is indicated by the average time of 95 minutes required to fit ellipses for all measurement positions. In the course of multiple simulations it was found that this approach is less stable than DLSF, as shown by the different results obtained depending on the starting population, the parameters determining the mutation and crossover options, the set number of generations (computation cycles), and the acceptable threshold of the defined objective function. Moreover, as noted above, for a spread of data that does not cover a full angle of $\varphi = 360^\circ$, but only some part of that range – in the cases analysed, generally not exceeding $\varphi = 180^\circ$ – it

is necessary to place an additional upper bound on the values of the semi-axes a and b. It may thus be concluded that all of the analysed approaches using DFO methods require two stages. In the absence of bounds, the values of a and b become absurdly high, and the centre coordinates of the fitted ellipses lie at a great distance from the data. Nonetheless, the approach using a genetic algorithm (GA), as a method of optimum fitting of ellipses to data, generates the results closest to those of DLSF – cf. Figs. 9 and 10.

Similar tendencies to those of the GA case were observed in the numerical computations using the PSO method. In this case also, the values of the approximated semi-axes a and b were greater than those obtained by the DLSF method.

The greatest divergences were obtained for the *differential evolution* (DE) algorithm. In spite of multiple simulations for different parameters determining the initial population, the number of generations and the procedure for crossover and mutation, no results better than those shown in Fig. 10 were obtained.

The above findings are confirmed by the results obtained for fitting errors. It may be noted that the average values of the *RMSE* (*root mean square error*) for the DLSF and GA methods are similar (Table 1). A slightly worse result, though close to that of GA and DLSF, was obtained for the PSO algorithm. By far the largest average *RMSE* was obtained for the DE method, which confirms the results of the study reported in [42], which analysed the fitting of ellipses to data lying only within a small angular range.

Table 1
Average values of *RMSE* (*[mm]*) for the fitted ellipses with respect to data from all measurement positions $R_{ef}^{(j)}$

Method of fitting (algorithm)			
Genetic algorithm (GA)	Differential evolution (DE)	Particle swarm optimisation (PSO)	Direct least squares fitting (DLSF)
25.62	150.32	51.31	31.11

5. Conclusions

In this study, an attempt was made to fit ellipses to measurement data, namely registrations of components of the field of horizontal ground displacements resulting from progressive work at the *Prospel–Haniel* mine.

Three different algorithms were used, divided into two groups:

- The *direct least squares fitting* (DLSF) algorithm, using algebraic distance as the error measure.
- Optimisation algorithms belonging to the class of DFO methods, the objective function being a measure of geometric distance (*genetic algorithm* – GA, *differential evolution* – DE, *particle swarm optimisation* – PSO).

The results indicate that when it is necessary to fit ellipses to data spread over an angle no greater than $\varphi = 180^\circ$, the best method is the DLSF approach.

The remaining algorithms in the DFO class require in this case a two-stage approach, consisting of:

- Preliminary determination of limiting values for the parameters a and b based on the estimation of curvature in the original data set.
- Optimisation by the selected procedure taking account of the constraints resulting from the preliminary analysis of the measurement data.

The initial data constructed in this way for the optimisation process are nonetheless subject to error, resulting from the non-uniform distances between the measurement points used to estimate the curvature by the *finite differential method*, among other things.

In the case of failure to apply an initial constraint to the parameters a and b, the values obtained for those parameters via optimisation are unreasonably large. The results cannot then be subjected to further interpretation in accordance with the physics of the analysed phenomenon.

A further defect that came to light in the course of multiple simulations is the high sensitivity of the DFO methods to the initial sets of data (*population*) and to the parameters determining the operation of particular components of the optimisation process (such as the *crossover* and *mutation* operations in the GA method). As a result, the operation of these methods is characterised by a high degree of inertia and by certain chaotic features.

Nonetheless, in cases where the data are spread over an angle of $\varphi = 360^\circ$, the DFO algorithms may constitute an alternative to the DLSF method in ellipse fitting problems.

As regards the required computational resources, expressed in terms of computation time, the DLSF method is again the most effective. The average ratio between the time required for fitting using the DLSF method and the time required by the other analysed methods was approximately t(DLSF) : t(DFO) $\approx$ 5 min : 95 min.

To conclude, the *direct least squares fitting* method was selected for further analysis, having been found to be the most effective tool for fitting ellipses to measurement data. This method will be used in further work together with *machine learning methods* (ML). The authors plan to construct a model which will be used in an implicit manner as a random case generator for the *Monte Carlo* method [43].

References

[1] T. Chmielewski and Z Zembaty Z, *Podstawy dynamiki budowli*, Warsaw: Arkady, 2006 [in Polish].

[2] J. Rusek, "Influence of the Seismic Intensity of the Area on the Assessment of Dynamic Resistance of Bridge Structures", in *IOP Conf. Ser.: Mater. Sci. Eng.* 2017, pp. 245-252, DOI: 10.1088/1757-899X/245/3/032019.

[3] J. Rusek and W Kocot, "Proposed Assessment of Dynamic Resistance of the Existing Industrial Portal Frame Building Structures to the Impact of Mining Tremors" in *IOP Conf. Ser.: Mater. Sci. Eng.* 2017, pp.162–245, DOI: 10.1088/1757-899X/245/3/032020.

[4] J. Rusek, "A proposal for an assessment method of the dynamic resistance of concrete slab viaducts subjected to impact loads caused by mining tremors", in *JCEEA*. 64(1), 469–486 (2018), DOI: 10.7862/rb.2017.43.

[5] K. Tajduś, "Analysis of Horizontal Displacements Measured over the Mining Operations in Longwall No. 537 at the Girondelle 5 Seam of the Bw Friedrich Heinrich-Rheinland Coal Mine", *Arch. Min. Sci.* 61(1), 157–168 (2016), DOI: 10.1515/amsc-2016-0012.

[6] K. Tajdus, "The nature of mining-induced horizontal displacement of surface on the example of several coal mines", *Arch. Min. Sci.* 59(4), 971–986 (2014), DOI: 10.2478/amsc-2014-0067.

[7] K. Tajduś "Analysis of horizontal displacement distribution caused by single advancing longwall panel excavation", *J. Rock Mech. Geotech. Eng.* 7(4), 395–403 (2015), DOI: 10.1016/j.jrmge.2015.03.012.

[8] Deutsche Montan Technologie GmbH (DMT). BW Prosper Haniel measurements point – Schwarze Heide, 2001 (not published) [in German].

[9] K. Tajduś, R. Misa, and A. Sroka, "Analysis of the surface horizontal displacement changes due to longwall panel advance", *Int. J. Rock Mech. Min. Sci.* 104, 119–125 (2018), DOI: 10.1016/j.ijrmms.2018.02.005.

[10] Z.L. Szpak, W. Chojnacki, and A. van den Hengel, "Guaranteed Ellipse Fitting with a Confidence Region and an Uncertainty Measure for Centre, Axes, and Orientation", *J. Math. Imaging Vision*, 52(2), 173–199 (2015), DOI: 10.1007/s10851-014-0536-x.

[11] M.A. Kashiha, C. Bahr, S. Ott, C.P.H. Moons, T.A. Niewold, F.O. Ödberg, and D. Berckmans, "Automatic identification of marked pigs in a pen using image pattern recognition", *Comput. Electron. Agric.* 93, 111–120 (2013), DOI: 10.1007/978-3-642-38628-2_24.

[12] L. Li, Y. Wang, X. Liu, Z. Tang, and Z. He, "A fast and robust ellipse detector based on top-down least-square fitting", in *BMVC*, 2015, DOI: 10.5244/c.29.156.

[13] A. Xu, Z. Wang, D. Kong, Z. Fu, and Q. Lin, "A new ellipse fitting method of the minimum differential-mode noise in the atom interference gravimeter", *Chin. Phys. B – IOPscience.* 27(7), 070203 (2018), DOI: 10.1088/1674-1056/27/7/070203.

[14] K. Kanatani, Y. Sugaya, and Y. Kanazawa, "Ellipse Fitting" in: *Guide to 3D Vision Computation. Advances in Computer Vision and Pattern Recognition*, pp. 11–32, Springer, Cham, 2016, DOI: 10.1007/978-3-319-48493-8_2.

[15] R Halır and J Flusser, "Numerically stable direct least squares fitting of ellipses", *in Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization*, vol. 98, pp. 125–132, WSCG, Citeseer.

[16] A. Ray and D.C. Srivastava, "Non-linear least squares ellipse fitting using the genetic algorithm with applications to strain analysis". *J. Struct. Geol.* 30(12), 1593–1602 (2008), DOI: 10.1016/j.jsg.2008.09.003.

[17] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization. An overview", *Swarm Intell.* 1(1), 33–57, (2007), DOI: 10.1007/s11721-007-0002-0.

[18] F. Ye, "Particle swarm optimization-based automatic parameter selection for deep neural networks and its applications in large-scale and high-dimensional data", *PLos one.* 12(12), e0188746 2017, DOI: 10.1371/journal.pone.0188746.

[19] A.J. Mantau, A. Bowolaksono, B. Wiweko, and W. Jatmiko, "Detecting ellipses in embryo images using arc detection method with particle swarm for Blastomere-quality measurement system", *JACIII*. 20(7), 1170–1180 (2016), DOI: 10.20965/ jaciii.2016.p1170.

[20] M. Szczepanik and T Burczyński, "Swarm optimization of stiffeners locations in 2-D structures", *Bull. Pol. Ac.: Tech.* 60(2), 241–246 (2012), DOI: 10.2478/v10175-012-0032-7.

[21] J. Lampinen and R. Storn, *Differential evolution. New optimization techniques in engineering*, pp. 123–166, Springer, 2004.

[22] L.M. Rios and N.V. Sahinidis, "Derivative-free optimization: A review of algorithms and comparison of software implementations". *J. Global Optim. Springer.* 56(3), 1247–1293 (2013), DOI: 10.1007/s10898-012-9951-y.

[23] J. Rusek, "Application of support vector machine in the analysis of the technical state of development in the LGOM mining area", *Maint. Reliab.* 19, 54–61, 2017, DOI: 10.17531/ein.2017.1.8.

[24] J. Rusek, "Creating a model of technical wear of building in mining area, with utilization of regressive SVM approach", *Arch. Min. Sci.* 54(3), 455–466, (2009).

[25] D. Rainville, F.-A. Fortin, M.-A. Gardner, M. Parizeau, and C. Gagné, "Deap: A python framework for evolutionary algorithms" in *GECCO '12*, pp. 85–92, 2012.

[26] F.A. Fortin, F.M.D. Rainville, M.A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy", *J. Mach. Learn. Res.* 13(1), 2171–2175 (2012).

[27] M.M. McKerns, P. Hung, and M.A.G. Aivazis, "Mystic: a simple model-independent inversion framework", 2009, [Online] Available: http://dev.danse.us/trac/mystic.

[28] M.M. McKerns, L. Strand, T. Sullivan, A. Fang, and M.A.G. Aivazis, "Building a framework for predictive science" arXiv preprint arXiv:1202.1056, 2012.

[29] B. Hammel and N. Sullivan-Molina, "Bdhammel/least-squares-ellipse-fitting: Initial release (Version v1.0)", Zenodo, DOI: 10.5281/zenodo.2578663.

[30] A.W. Fitzgibbon, M. Pilu, and R.B. Fisher, "Direct least squares fitting of ellipses", *IEEE Xplore* 1, 253–257 (1996), DOI: 10.1109/ICPR.1996.546029.

[31] E. Cuevas, D. Zaldivar, M. Pérez-Cisneros, and M. Ramírez-Ortegón, "Circle detection using discrete differential evolution optimization", *Pattern Anal. Appl. Springer*. 14, 93–107 (2011), DOI: 10.1007/s10044-010-0183-9.

[32] E. Cuevas, M. González, D. Zaldívar, and M. Pérez-Cisneros, "Multi-ellipses detection on images inspired by collective animal behavior", *Neural. Comput. Appl.* 24, 1019–1033 (2014), DOI: 10.1007/s00521-012-1332-4.

[33] T. Witkowski, P. Antczak, and A. Antczak, "Multi-objective decision making and search space for the evaluation of production process scheduling", *Bull. Pol. Ac.: Tech.* 3(57), 195–208 (2012), DOI: 10.2478/v10175-010-0121-4.

[34] J.C. Strikwerda, *Finite difference schemes and partial differential equations*, SIAM, 2004.

[35] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II", *IEEE Trans. Evol. Comput*. 6(2), 182–197 (2002), DOI: 10.1109/4235.996017.

[36] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces", *J. Global Optim.* 11(4), 341–359 (1997).

[37] K. Price, R.M. Storn, and J.A Lampinen, *Differential evolution: a practical approach to global optimization*, Springer-Verlag Berlin Heidelberg, 2006.

[38] M.M. Ali and A. Törn, "Population set-based global optimization algorithms: some modifications and numerical studies", *Comput Oper Res*. 31(10), 1703–1725 (2004), DOI: 10.1016/S0305-0548(03)00116-3.

[39] Y. Fukuyama, *Fundamentals of particle swarm optimization techniques. Modern Heuristic Optimization Techniques: Theory and applications to power systems*, pp. 71–87, John Wiley & Sons, 2008.

[40] C. Blum and X. Li,"Swarm Intelligence in Optimization" in *Swarm Intell*, pp. 43–85, ed. Blum C. Merkle D. Natural Computing Series: Springer, Berlin, Heidelberg, 2008, DOI: 10.1007/978-3-540-74089-6_2.

[41] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory", in *MHS'95. Proc. Sixth Int. Symp. Micro Mach. Hum. Sci*, 1995, pp. 39–43, DOI: 10.1109/MHS.1995.494215.

[42] L.G. de la Fraga, I.V. Silva, and N. Cruz-Cortés, "Euclidean Distance Fit of Conics Using Differential Evolution" in: *Evolutionary Image Analysis and Signal Processing,* pp. 171–184, Springer, Berlin, Heidelberg, 2009, DOI: 10.1007/978-3-642-01636-3_10.

[43] C. Robert and G. Casella. *Monte Carlo statistical methods*, Springer Science and Business Media, 2013.