

Extension of the ABC-Procrustes algorithm for 3D affine coordinate transformation

B. Paláncz¹, P. Zaletnyik², J. L. Awange^{3,4}, and B. Heck⁴

¹Department of Photogrammetry and Geoinformatics, Budapest University of Technology and Economics, H-1521, Hungary

²Department of Geodesy and Surveying and Research Group of Physical Geodesy and Geodynamics of the Hungarian Academy of Sciences, Budapest University of Technology and Economics, H-1521, Hungary

³Western Australian Centre for Geodesy and The Institute for Geoscience Research, Curtin University, Australia

⁴Geodetic Institute, Karlsruhe Institute of Technology (KIT), Engler-Strasse 7, D-76131, Karlsruhe

(Received March 20, 2009; Revised July 28, 2010; Accepted October 25, 2010; Online published January 26, 2011)

The Procrustes method is a very effective method for determining the Helmert's datum transformation parameters since it requires neither *initial starting values* nor *iteration*. Due to these attractive attributes, the *ABC-Procrustes* algorithm is extended to solve the 3D affine transformation problem where *scale factors* are different in the 3 principal directions X, Y, Z . In this study, it is shown that such a direct extension is restricted to cases of *mild anisotropy* in scaling. For strong anisotropy, however, the procedure fails. The PZ-method is proposed as an extension of the ABC algorithm for this special case. The procedures are applied to determine transformation parameters for; (i) transforming the Australian Geodetic Datum (AGD 84) to the Geocentric Datum Australia (GDA 94), i.e., mild anisotropy and (ii) synthetic data for strong anisotropy. The results indicate that the PZ-algorithm leads to a local multivariate minimization as opposed to the *ABC-algorithm*, thus requiring slightly longer computational time. However, the ABC-method is found to be useful for computing proper initial values for the PZ-method, thereby increasing its efficiency.

Key words: Procrustes, coordinate transformation, anisotropy scaling, Helmert transformation, singular value decomposition, global minimization.

1. Introduction

Three-dimensional coordinate transformations play a central role in contemporary Euclidean point positioning. In precise positioning using the Global Positioning System (GPS), coordinates given in the World Geodetic System 1984 (WGS84) often have to be transformed into local geodetic coordinate systems. The transformation between the traditional terrestrial coordinate system and the satellite derived network can be a daunting task due to the heterogeneity of the data.

The 3D affine transformation problem arises when the scale is not uniform within two or more configurations of points such that they vary within given dimensions. It means that beside the parameters of the rotation matrix (3) and the translation vector (3), an additional 3 scale parameters should also be computed, thus leading to 9 parameters instead of the usual 7. In order to determine the 9 parameters using the traditional iterative methods, good initial starting values are essential. In practise, however, it is sometimes difficult to have good starting values thereby enhancing the complexity of the problem (e.g., Paláncz *et al.*, 2008). Realizing this difficulty, and in order to circumvent the need for initial starting values and iterations, Awange *et al.* (2008) proposed the extension of the con-

ventional 7-parameter Procrustean algorithm (e.g., Awange and Grafarend, 2005) to solve the 3D affine transformation problem by the application of what we here refer to as the *ABC-Procrustes Algorithm*.

As we demonstrate in this study, however, the ABC-Procrustes algorithm proved to be successful only in cases of very mild *anisotropy* in scaling. For strong anisotropy in scaling, we propose an extension of the ABC-Procrustes algorithm to give a more generally valid solution of the problem. The new extended version of the ABC-algorithm, referred to as the PZ-algorithm, is tested to solve the 3D affine problem under both mild and strong anisotropy conditions.

This study is organized as follows: In Section 2, we provide a brief introduction to the Procrustes problem and illustrate how it is used to solve the Helmert's 3D datum transformation problem in Section 3. Section 4 defines the 3D affine transformation problem while Section 5 revisits the ABC procrustes algorithm that is presented in Awange *et al.* (2008) for solving the 3D affine transformation problem. Section 6 then suggests the PZ-algorithms as an extension of the ABC-algorithm and provides examples of solving both mild and strong anisotropy problems. The study is concluded in Section 7.

2. Procrustes Problems

The Procrustean approach for solving the least squares problem which appeared nearly 45 years ago (see Schönemann, 1966), is a very fast method that needs neither initial starting values nor iteration. The Procrustes ap-

proach matches one configuration to another and produces a measure of match, while seeking the isotropic dilation and the rigid translation, reflection and rotation needed to best match one configuration to another (Cox and Cox, 1994). In the application of the original Procrustes method, the geodetic pioneer was *F. Crosilla*, while for photogrammetry they were *F. Crosilla* and *A. Beinat* (e.g., Crosilla, 1983a, b, 2003; Crosilla and Beinat, 2002).

The Procrustes problem is concerned with fitting a configuration \mathcal{B} to \mathcal{A} as close as possible. The simplest Procrustes case is one in which both configurations have the same dimensionality and the same number of points, which can be brought into a 1-1 correspondence by substantive considerations (Borg and Groenen, 1997). Let us consider the case where both \mathcal{A} and \mathcal{B} are of the same dimension. The *partial Procrustes* problem is then formulated as

$$\mathcal{A} = \mathcal{B}\mathcal{T} \quad (1)$$

The rotation matrix \mathcal{T} is then solved by measuring the distances between corresponding points in both configurations, squaring these values, and adding them to obtain the root of the sum of squares $\|\mathcal{A} - \mathcal{B}\mathcal{T}\|$, which is then minimized. One proceeds via Frobenius norm as follows

$$\|\mathcal{A} - \mathcal{B}\mathcal{T}\| = \sqrt{\text{tr}((\mathcal{A}^T - \mathcal{T}^T \mathcal{B}^T)(\mathcal{A} - \mathcal{B}\mathcal{T}))} \rightarrow \min_{\mathcal{T}}, \quad (2)$$

where $\mathcal{T}^T \mathcal{T} = \mathbf{I}$. This partial Procrustes problem can be extended for additional translation and scaling, in addition to the rotation (e.g., Gower and Dijksterhuis, 2004)

$$\mathcal{A} = s\mathcal{B}\mathcal{T} + \mathcal{D} \quad (3)$$

Then, the objective function to be minimized is written as

$$\begin{aligned} \|\mathcal{A} - s(\mathcal{B}\mathcal{T} + \mathcal{D})\|_W &= \\ &= \sqrt{\text{tr}((\mathcal{A}^T - s(\mathcal{T}^T \mathcal{B}^T + \mathcal{D})) W (\mathcal{A} - s(\mathcal{B}\mathcal{T} + \mathcal{D})))} \\ &\rightarrow \min_{T,s,\mathcal{D}}, \end{aligned} \quad (4)$$

where W is a weight matrix, s is the scaling parameter and \mathcal{D} is the translation vector. This Procrustes problem is referred to as the *general Procrustes* problem.

These Procrustes problems arise in applications related to, e.g., rigid body movement and psychometrics, factor analysis, multivariate analysis, multidimensional scaling and GPS locations (see for example, Gower (1984) and Meridith (1977)). The solution of the general Procrustes problem provides the solution of the parameter estimation problem of the Helmert transformation with 7 parameters.

3. Procrustean Solution of the Helmert's Transformation Problem

The original Procrustes method with weighting can be applied to the parameter estimation problem of Helmert transformation with 7 parameters (e.g., Awange and Grafarend, 2005). Given coordinates in two systems

$\{x, y, z\}$ and $\{X, Y, Z\}$, the form of the general Procrustes algorithm for the $C_7(3, 3)$ problem with N points is given as

$$\begin{pmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & z_N \end{pmatrix} = \begin{pmatrix} X_1 & Y_1 & Z_1 \\ \vdots & \vdots & \vdots \\ X_N & Y_N & Z_N \end{pmatrix} \mathbf{R}^T s + \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} (X_0, Y_0, Z_0) + \mathbf{E}, \quad (5)$$

where s is a scalar valued unknown for the scale factor, $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ unknown *orthonormal matrix* for the rotation matrix, and $\mathbf{E} = e_{i,j}$ is the error matrix, $i = 1, \dots, N$, $j = 1, \dots, 3$. One may also use a more compact form

$$\mathbf{xyz} = \mathbf{XYZ} * \mathbf{R}^T * s + \mathbf{1} * \mathbf{XYZ}_0^T + \mathbf{E}, \quad (6)$$

where

$$\begin{aligned} \mathbf{1} &= \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}, \quad \mathbf{xyz} = \begin{pmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_N & y_N & z_N \end{pmatrix}, \\ \mathbf{XYZ} &= \begin{pmatrix} X_1 & Y_1 & Z_1 \\ \vdots & \vdots & \vdots \\ X_N & Y_N & Z_N \end{pmatrix}, \quad \mathbf{XYZ}_0 = \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}, \end{aligned}$$

with length of N . Let us express the error matrix explicitly,

$$\mathbf{E} = \mathbf{xyz} - \mathbf{XYZ} * \mathbf{R}^T * s - \mathbf{1} * \mathbf{XYZ}_0^T, \quad (7)$$

then, introducing \mathbf{W} , a positive definite weight matrix,

$$\mathbf{W} = \text{diag}(w_1, \dots, w_N), \quad (8)$$

the square of the Frobenius norm of the weighted error matrix becomes

$$\|\mathbf{E}\|_w^2 = \text{tr}(\mathbf{E}^T \mathbf{W} \mathbf{E}). \quad (9)$$

The problem therefore is to minimize,

$$\|\mathbf{E}(s, \mathbf{XYZ}_0, \mathbf{R})\|_w^2 \rightarrow \min \quad (10)$$

under the condition

$$\mathbf{R}^T \mathbf{R} = \mathbf{I}_3 \text{ and } \|\mathbf{R}\| = +1 \quad (11)$$

This means, that, in case of $C_7(3, 3)$, one has to solve a constrained minimization problem, where the scale factor s corresponds to $s \in \mathbb{R}$. The translation vector (X_0, Y_0, Z_0) corresponds to $\{X, Y, Z\}_0 \in \mathbb{R}^3$, and the rotation matrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$.

The algorithm for solving this problem can be found in Gower and Dijksterhuis (2004), and is summarized as follows:

- 1) Compute the centering matrix \mathbf{C}

$$\mathbf{C} = \mathbf{I}_N - \frac{1}{N} * \mathbf{1} * \mathbf{1}^T, \quad (12)$$

where (\mathbf{I}_N) is the N dimensional unit matrix.

2) Compute the \mathbf{A} matrix

$$\mathbf{A} = \mathbf{xyz}^T * \mathbf{C}^T * \mathbf{W} * \mathbf{C} * \mathbf{XYZ} \quad (13)$$

3) Employing singular value decomposition (SVD) of \mathbf{A} , we obtain the matrices \mathbf{U} , \mathbf{V} and $\text{diag}(\lambda_1, \lambda_2, \lambda_3)$, where

$$\mathbf{A} = \mathbf{U} * \text{diag}(\lambda_1, \lambda_2, \lambda_3) * \mathbf{V}^T \quad (14)$$

4) The rotation matrix can then be computed as (15)

$$\mathbf{R} = \mathbf{U} * \mathbf{V}^T \quad (15)$$

5) The rotation matrix in general is given by using 3 axial rotation angles (α, β, γ) , but can also be expressed by the skew matrix as

$$\mathbf{R} = (\mathbf{I}_3 - \mathbf{\Xi})^{-1}(\mathbf{I}_3 + \mathbf{\Xi}) \quad (16)$$

Consequently, the skew matrix is

$$\mathbf{\Xi} = (\mathbf{R} - \mathbf{I}_3)(\mathbf{R} + \mathbf{I}_3) \quad (17)$$

and then the parameters a, b and c can be computed from

$$\mathbf{\Xi} = \begin{pmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{pmatrix} \quad (18)$$

6) The scale parameter can then be expressed as

$$s = \frac{\text{tr}(\mathbf{xyz}^T * \mathbf{C}^T * \mathbf{W} * \mathbf{C} * \mathbf{XYZ} * \mathbf{R}^T)}{\text{tr}(\mathbf{XYZ}^T * \mathbf{C}^T * \mathbf{W} * \mathbf{C} * \mathbf{XYZ})}, \quad (19)$$

where {tr} is the trace of a matrix.

7) Since the translation vector satisfies the following normal equation

$$\mathbf{1}^T * \mathbf{W} * \mathbf{1} * \mathbf{XYZ}_0 = (\mathbf{xyz} - \mathbf{XYZ} * \mathbf{R}^T s)^T * \mathbf{W} * \mathbf{1}, \quad (20)$$

we therefore obtain

$$\mathbf{XYZ}_0 = (\mathbf{1}^T \mathbf{W} * \mathbf{1})^{-1} (\mathbf{xyz} - \mathbf{XYZ} * \mathbf{R}^T s)^T \mathbf{W} * \mathbf{1} \quad (21)$$

For the purpose of this study, the algorithm has been implemented in *Mathematica*.

4. The 3D Affine Transformation Problem

The 3D affine transformation is one possible generalization of the $C_7(3, 3)$ Helmert transformation, using three different scale (s_1, s_2, s_3) parameters instead of a single one. In this case, the scale factors can be modeled by a diagonal matrix (\mathbf{S}).

$$\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} = \mathbf{S} * \mathbf{R} \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} + \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} \quad (22)$$

where

$$\mathbf{S} = \begin{pmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & s_3 \end{pmatrix} \quad (23)$$

is the scale matrix, X_0, Y_0, Z_0 are the 3 translation parameters and \mathbf{R} is the rotation matrix. The rotation matrix can be expressed by the skew-symmetric matrix ($\mathbf{\Xi}$) as in (18) which together with (16) leads to:

$$\mathbf{R} = \begin{pmatrix} \frac{1+a^2-b^2-c^2}{1+a^2+b^2+c^2} & \frac{2ab-2c}{1-a^2+b^2-c^2} & \frac{2(b+ac)}{1+a^2+b^2+c^2} \\ \frac{2(ab+c)}{1+a^2+b^2+c^2} & \frac{1+a^2+b^2+c^2}{1-a^2+b^2-c^2} & \frac{2(a-bc)}{1+a^2+b^2+c^2} \\ \frac{1+a^2+b^2+c^2}{2(-b+ac)} & \frac{1+a^2+b^2+c^2}{2(a+bc)} & \frac{1+a^2+b^2+c^2}{1-a^2-b^2+c^2} \end{pmatrix}, \quad (24)$$

for which the restriction satisfy

$$\mathbf{R}^T \mathbf{R} = \mathbf{I}_3. \quad (25)$$

5. ABC Procrustean Algorithm

In order to solve the 3D affine transformation problem outlined in (22), the ABC (Awange, Bae and Claessens) algorithm is implemented as follows (Awange *et al.*, 2008):

1) Compute the center of gravity of the two systems

$$\mathbf{a}_0 = \frac{\sum_{i=1}^N \mathbf{xyz}_i}{N} \quad (26)$$

$$\mathbf{b}_0 = \frac{\sum_{i=1}^N \mathbf{XYZ}_i}{N} \quad (27)$$

2) Translate the systems into the center of the coordinate system

$$\mathbf{xyz}^T * \mathbf{C} = (\mathbf{xyz} - \mathbf{1} * \mathbf{a}_0)^T \quad (28)$$

$$\mathbf{XYZ}^T * \mathbf{C} = (\mathbf{XYZ} - \mathbf{1} * \mathbf{b}_0)^T \quad (29)$$

3) Compute the matrix \mathbf{A}

$$\mathbf{A} = (\mathbf{XYZ}^T * \mathbf{C})^T * \mathbf{xyz}^T * \mathbf{C} \quad (30)$$

4) Compute the rotation matrix \mathbf{R} via (15)

5) The first approximation of the scale matrix (\mathbf{S}_0) only considers the diagonal elements of,

$$\mathbf{S} = \left(\mathbf{R}^T * (\mathbf{XYZ}^T * \mathbf{C})^T * \mathbf{XYZ}^T * \mathbf{C} * \mathbf{R} \right)^{-1} * \mathbf{R}^T * (\mathbf{XYZ}^T * \mathbf{C})^T * \mathbf{xyz}^T * \mathbf{C} \quad (31)$$

i.e.,

$$\mathbf{S}_0 = \text{diag}(\mathbf{S}) \quad (32)$$

- 6) Iterate to improve the scale matrix

$$\mathbf{w} = \mathbf{xyz} - \mathbf{XYZ} * \mathbf{R} * \mathbf{S}_0 - \mathbf{1} * (\mathbf{a}_0 - \mathbf{b}_0 * \mathbf{R} * \mathbf{S}_0) \quad (33)$$

$$\mathbf{dS} = \text{diag} \left((\mathbf{R}^T * \mathbf{XYZ}^T * \mathbf{XYZ} * \mathbf{R})^{-1} * \mathbf{R}^T * \mathbf{XYZ}^T * \mathbf{w} \right) \quad (34)$$

then after the first iteration step

$$\mathbf{S}_1 = \mathbf{S}_0 + \mathbf{dS} \quad (35)$$

- 7) After the iteration, the scale matrix is

$$\mathbf{S} = \text{diag} (\mathbf{S}_0) \quad (36)$$

- 8) Then the translational vector

$$\mathbf{XYZ}_0 = \mathbf{a}_0 - \mathbf{b}_0 * \mathbf{R} * \mathbf{S} \quad (37)$$

6. Extension of the ABC Method via the PZ Algorithm

The ABC method is very fast, and the computation effort for iteration to improve the \mathbf{S} scale matrix is negligible. The main problem with the ABC method, however, is that after shifting the two systems onto the origin of the coordinate system, the rotation matrix is computed via SVD. This means that the physical rotation, as well as the deformation (distortion) caused by the different scaling in the different directions of the 3 different principal axes, will also be involved in the computation of the rotation matrix. This can be approximately allowed only when these scale factors do not differ from each other considerably. In addition, the necessary condition for obtaining the optimal scale matrix is restricted to the diagonal matrix (e.g., Eq. (36)), where the off-diagonal elements are simply deleted in every iteration step thereby leading to solutions that do not ensure a true global minimum for the trace of the error matrix.

The PZ (Paláncz-Zaletnyik) method can cure these problems, but there is a price for it! This method starts with an initially diagonal \mathbf{S} scale matrix, and uses the inverse of this matrix to eliminate the deformation caused by scaling. Only after the elimination of this “distortion” will SVD be applied to compute the rotation matrix itself. It goes without saying that in this way, the computation of the rotation matrix should be repeated in an iterative way in order to decrease the error. That is why this method is precise and correct for any ratio of scale parameters, but takes longer to achieve the results than in the case of the ABC method.

The algorithm can be summarized as it follows:

- 1) Use the elements of the diagonal scale matrix \mathbf{S} in (23). The function for computing the error for the case of a given scale matrix \mathbf{S} is the following:
- 2) Using Eqs. (26) and (27), compute the center of gravity of the two systems.
- 3) Translate the systems using (28) and (29).
- 4) Eliminate rotation (distortion) caused by scaling with different values in different directions in the computation of matrix \mathbf{A} using

$$\mathbf{A} = (\mathbf{XYZ}^T * \mathbf{C})^T * \mathbf{xyz}^T * \mathbf{C} * \mathbf{S}^{-1} \quad (38)$$

- 5) Compute the rotation matrix using (15).
- 6) Compute the translation vector using (37).
- 7) Using the error matrix

$$\mathbf{E} = \mathbf{xyz} - \mathbf{XYZ} * \mathbf{R}^T * \mathbf{S} - \mathbf{1} * \mathbf{XYZ}_0^T, \quad (39)$$

compute the error matrix using (9). If this error is too high, then modify the scale matrix \mathbf{S} , and repeat the computation. In the case of the 7-parameter similarity transformation (Helmert), the scale value (s) can be estimated by dividing the sum of length in both systems from the centre of gravity (see, Albertz and Kreiling, 1975).

The center of gravity of the systems is therefore,

$$\mathbf{a}_0 = \begin{pmatrix} x_s \\ y_s \\ z_s \end{pmatrix} \text{ and } \mathbf{b}_0 = \begin{pmatrix} X_s \\ Y_s \\ Z_s \end{pmatrix} \quad (40)$$

The estimated scale parameter according to Albertz and Kreiling (1975) in the Helmert similarity transformation is thus given by

$$s_{\text{priori}} = \frac{\sum_{i=1}^N \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2 + (z_i - z_s)^2}}{\sum_{i=1}^N \sqrt{(X_i - X_s)^2 + (Y_i - Y_s)^2 + (Z_i - Z_s)^2}} \quad (41)$$

For the case of the 9 parameter affine transformation where 3 different scale values (s_1, s_2, s_3) are applied to the 3 coordinate axes, a good approach for the scale parameters can be given by modifying the Albertz and Kreiling (1975) expression. Instead of the quotient of the two lengths in the centre of gravity system, we can use the quotients of the sum of the lengths in the corresponding coordinate axes directions.

The estimated scale parameters according to the modified Albertz and Kreiling (1975) expression for the 9 parameter transformation (see also Zaletnyik, 2008) is

$$s_{01} = \frac{\sum_{i=1}^N \sqrt{(x_i - x_s)^2}}{\sum_{i=1}^N \sqrt{(X_i - X_s)^2}}, s_{02} = \frac{\sum_{i=1}^N \sqrt{(y_i - y_s)^2}}{\sum_{i=1}^N \sqrt{(Y_i - Y_s)^2}},$$

$$s_{03} = \frac{\sum_{i=1}^N \sqrt{(z_i - z_s)^2}}{\sum_{i=1}^N \sqrt{(Z_i - Z_s)^2}}, \quad (42)$$

where (s_{01}, s_{02}, s_{03}) are the initial elements in (23).

Example 1: Application to a network with mild anisotropy

For the determination of the 9 transformation parameters ($a, b, c, X_0, Y_0, Z_0, s_1, s_2, s_3$), we need $N \geq 3$ non-collinear points with known positions in both coordinate systems. In the case of N stations, we have $3N$ nonlinear equations, a system of multivariable polynomial to solve.

First, we consider the transformation problem of the Australian Geodetic Datum (AGD 84) to Geocentric Datum Australia (GDA 94) with 82 points (e.g., Awange *et al.*, 2008). This problem is solved using (see results in Table 1):

- 1) Direct Numerical Solution via Global Minimization (GM). This approach solves a system of 246 equations for the unknown parameters $a, b, c, s_1, s_2, s_3, X_0, Y_0,$

Table 1. Results of the different methods for the case of a network with mild anisotropy.

Method	Time (sec)	Error (m)	Scale s_1, s_2, s_3	Translation (m)	Rotation angles α, β, γ (seconds)
Procrustes	0.062	6.867	1.00000368981	-115.838 -48.373 144.760	-0.120 -0.384 -0.370
ABC	0.282	6.788	1.00000396085 1.00000354605 1.00000345982	-115.062 -47.676 144.096	-0.120 -0.384 -0.370
PZ	0.687	6.642	1.00000416838 1.00000310067 1.00000346938	-112.169 -44.047 144.311	-0.081 -0.436 -0.437
GM	3.562	6.642	1.00000416842 1.00000310066 1.00000346943	-112.169 -44.046 144.312	-0.081 -0.436 -0.437

Table 2. Results of the different methods for the case of a network with strong anisotropy.

Method	Time (sec)	Error (m)	Scale s_1, s_2, s_3	Translation (m)	Rotation angles α, β, γ (degrees)
Original	—	$\sigma_x = 0.07$ $\sigma_y = 0.08$ $\sigma_z = 0.10$	0.620 1.300 1.870	1345.34 -233.23 121.11	6.0 11.1 16.3
Procrustes	0.047	117285.16	1.234	$-4.826 * 10^6$ 253120.162 $6.356 * 10^6$	9.990 -18.655 17.900
ABC	0.281	93431.630	1.099 1.287 1.136	$-4.074 * 10^6$ 219870.673 $6.642 * 10^6$	15.678 -14.267 21.520
PZ	0.906	1.3581	0.620 1.300 1.870	1339.036 -236.061 152.711	2.733 12.304 15.422
GM	4.625	1.23799	0.620 1.300 1.870	1339.072 -236.047 152.461	6.000 11.100 16.300

Table 3. Running times of the PZ method with initial values (s_1, s_2, s_3) estimated by different methods for the case of a network with strong anisotropy.

Time (sec)	s_1	s_2	s_3	Method used for initial values
1.11	1	1	1	—
0.86	0.509	0.712	2.150	modified Albertz-Kreiling
0.70	0.538	1.291	1.627	ABC without iteration
1.00	1.100	1.287	1.136	ABC solution

Z_0 . In order to use directly the least squares method, we can define the objective function as the sum of the square of the errors, which should be minimized (e.g., Paláncz *et al.*, 2008). To carry out the global minimization via a genetic algorithm, the built-in function **NMinimize** in Mathematica is employed. As this is the direct least squares solution of the problem, we accepted this result as reference solution.

- 2) Application of the Helmert transformation using the original Procrustes approach.
- 3) Application of the ABC method.
- 4) Application of the PZ method.

The error in Table 1 and Table 2 is defined as the sum of the square of the residuals of the coordinates after the transformation.

Example 2: Application to a network with strong anisotropy

For this test, a synthetic dataset was generated from 81 Hungarian Datum points. In this case, in order to test the efficiency of the different methods, the angles of rotations are not small and the values of the 3 scale parameters are considerably different from each other. A 9-parameter transformation was applied on the original dataset, with $\alpha = 6^\circ$, $\beta = 11.1^\circ$, $\gamma = 16.3^\circ$, $s_1 = 0.62$, $s_2 = 1.30$, $s_3 = 1.87$, and $X_0 = 1345.34$ m, $Y_0 = -233.23$ m, $Z_0 = 121.11$ m. After the transformation, a normal distributed noise was added to the generated coordinate set with zero means and standard deviation of $\sigma_x = 0.07$ m, $\sigma_y = 0.08$ m, $\sigma_z = 0.10$ m. This synthetic data was then used to estimate the original transformation parameters.

Using the four approaches defined above leads to the results of Table 2. For comparison, the second row of Table 2 contains the original data used for creating the syn-

thetic dataset. Comparing the results, it can be seen that the Procrustes solution of the 7-parameter transformation, and the ABC method differ totally from the correct ones, whereas the PZ method has a good solution according to the residual error, since it returned the same scale parameters as the original ones, although both the translation and rotation parameters differ somewhat. The best results are achieved with the global minimization, but this calculation took the longest time. In this case, both the scale and rotation parameters are equal to the original values, but the translation parameters differ. This is because of the added normal distributed noise. The translation parameters are almost the same in case of PZ method and global minimization, and the residual error has the same magnitude. The most sensitive parameter according to the chosen method are the rotation parameters.

Example 3: Testing different initial values for the PZ algorithm

In this example, we are interested in testing the PZ-algorithm with different starting values. We consider four scenarios: (i) using an identity matrix \mathbf{I}_3 as the starting values in which case ($s_{01} = 1, s_{02} = 1, s_{03} = 1$); (ii) the modified Albertz-Kreiling expression (see Eq. (42)); (iii) the ABC-method without iteration (i.e., the results of the first run); and (iv) The ABC-method with the solutions after iterations. The results are presented in Table 3. The different initial values give the same solutions, but with different processing times, the ABC method without iteration provided the best initial value for S.

7. Conclusions

For the case of mild anisotropy in the network, the ABC method using 3D affine transformation gives a better approximation than the general Procrustes method employing Helmert transformation model, while, the PZ method provides a precise, geometrically correct solution. The ABC method is about 2 times faster than the PZ method, and the later is roughly 5 times faster than the global optimization method when applied to the 3D affine model (see Table 1).

For the case of strong anisotropy, use of the ABC method to solve the 3D affine transformation completely fails, as does the general Procrustes method for Helmert transformation. The PZ method provides a good solution, and it is about 5 times faster than the global minimization. However, even in the case of strong anisotropy, the ABC method is useful for computing proper initial values for the PZ method in order to increase its efficiency (see Table 3). The different initial values give the same solutions, but with different processing times. The fastest version was that using the results of the first run of the ABC method as the initial values.

In the case of strong anisotropy, a synthetic dataset was used with known transformation parameters with added normally distributed noise. The results show that the scale factors are quite robust and insensitive to the added noise, with both the PZ method and global minimization giving back exactly the original values of the scale. The cal-

culated translation parameters were more sensitive to the noise and differ from the original values in both cases (with 0.5%, 0.3% and 20.4%). The rotation angles were the same for global minimization, but slightly greater differences are noted with the PZ method. However, this method had the same error magnitude as the global minimization. The PZ method can be extended for a 12 parameter model, as well as for the general affine model.

The results of the computations have also been verified with the original MATLAB code of the ABC method.

Acknowledgments. The authors thank Dr. Kevin Fleming for proof reading the manuscript, but take the responsibility of any errors. JLA acknowledges the financial support of Alexander von Humboldt (Ludwig Leichhardt's Memorial Fellowship) and Curtin Research Fellowship. He is grateful for the warm welcome and the conducive working atmosphere provided by his host Prof. Bernhard Heck at the Geodetic Institute, Karlsruhe Institute of Technology (KIT). TIGer Publication No. 253.

References

- Albertz, J. and W. Kreiling, *Photogrammetric Guide*, pp. 58–60, Herbert Wichmann Verlag, Karlsruhe, 1975.
- Awange, J. L. and E. W. Grafarend, *Solving Algebraic Computational Problems in Geodesy and Geoinformatics*, Springer, 2005.
- Awange, J. L., K.-H. Bae, and S. Claessens, Procrustean solution of the 9-parameter transformation problem, *Earth Planets Space*, **60**, 529–537, 2008.
- Borg, I. and P. Groenen, *Modern Multidimensional Scaling*, Springer, New York, 1997.
- Cox, T. F. and M. A. Cox, *Multidimensional Scaling*, St. Edmundsbury Press, St. Edmunds, Suffolk, 1994.
- Crosilla, F., A criterion matrix for a second order design of control networks, *Bull. Geod.*, **57**, 226–239, 1983a.
- Crosilla, F., Procrustean transformation as a tool for the construction of a criterion matrix for control networks, *Manuscripta Geodetica*, **8**, 343–370, 1983b.
- Crosilla, F., Procrustes analysis and geodetic sciences, in *Geodesy—The Challenge of the 3rd Millennium*, edited by Grafarend, E. W., F. W. Krumm, and V. S. Schwarze, pp. 287–292, Springer, Heidelberg, 2003.
- Crosilla, F. and A. Beinat, Use of generalised Procrustes analysis for the photogrammetric block adjustment by independent models, *J. Photogramm. Remote Sens.*, **56**, 195–209, 2002.
- Gower, J. C., Multivariate Analysis. Ordination, Multidimensional Scaling and Allied Topics, *Handbook of Applicable Mathematics, VI.: Statistics (B)*, 1984.
- Gower, J. C. and G. B. Dijksterhuis, *Procrustes Problems*, New York, Oxford University Press, 2004.
- Meridith, W., On weighted Procrustes and Hyperplane fitting in factor analytic rotation, *Psychometrika*, **42**, 491–522, 1977.
- Paláncz, B., R. Lewis, P. Zaletnyik, and J. Awange, Computational study of 3D affine coordinate transformation: Part I. 3-point problem, *MathSource /7090/* and Part II. N-point problem, *MathSource /7121/*. : <http://library.wolfram.com/infocenter/MathSource/7090/>, <http://library.wolfram.com/infocenter/MathSource/7121/>, 2008.
- Schönemann, P. H., A generalized solution of the orthogonal procrustes problem, *Psychometrika*, **31**(1), 1–10, 1966.
- Zaletnyik, P., Coordinate transformation using Computer Algebra Systems (CAS) and Artificial Neural Networks (ANN), Ph.D. Dissertation, Budapest University of Technology and Economy, http://www.agt.bme.hu/staff_h/zaletnyik_pub.html, 2008.