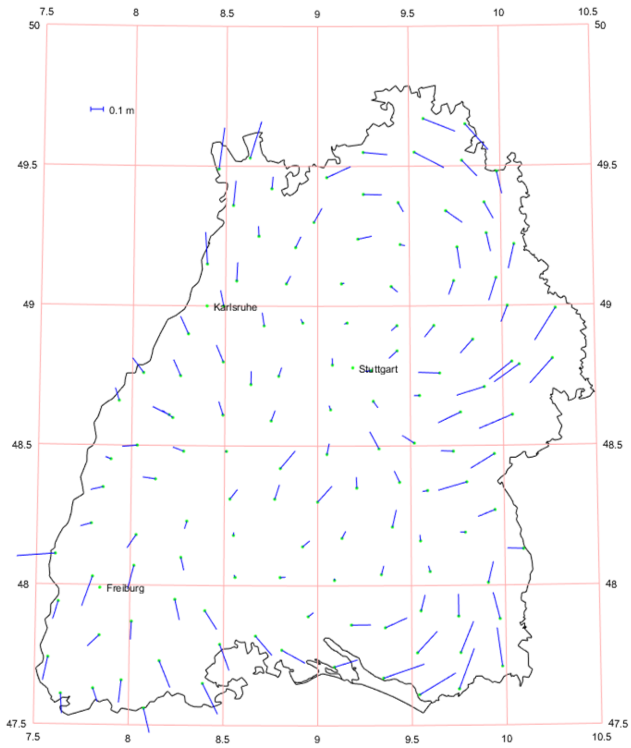
Universität Stuttgart

Geodätisches Institut

Study on the Converted Total Least Squares method and its application in coordinate transformation



Bachelorarbeit im Studiengang

Geodäsie und Geoinformatik

an der Universität Stuttgart

Dalu Dong

Stuttgart, April 2017

Betreuer: Prof. Dr.-Ing. Nico Sneeuw
Universität Stuttgart

Dr.-Ing. Jianqing Cai
Universität Stuttgart

Erklärung der Urheberschaft

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ort, Datum Unterschrift

Abstract

This Thesis gives a brief introduction to Total Least Squares (TLS) comparing with the classical LS, and its common solutions by singular value decomposition (SVD) approaches and the iteration, also following with the advantages and disadvantages of both methods. One method named Converted Total Least Squares (CTLS) dealing with the errors-in-variables (EIV) model can solve the problems of both. The basic idea of it is to take the stochastic design matrix elements as virtual observations, and the TLS problem can be transformed into a LS problem. The significance of CTLS lies not merely in attaining the optimal estimation of parameters and more importantly in completing the theory of TLS with classical LS. As a comparison, another estimation method based on Partial-EIV model will also be presented, which can deal with the TLS problems with iterative algorithm. The coordinate transformation parameter estimation formula of both algorithms are derived. By specifying the accuracy assessment formulas of CTLS, this thesis identifies rigorously the degree of freedom of the EIV model in theory and solves the bottleneck problem of TLS that restricts the application and development of TLS.

Key words Total Least Squares, singular value decomposition, errors-in-variables, virtual observation, Partial-EIV model, accuracy assessment.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the development of measurement means and continuous improvements in precision measuring instruments, there are higher requests for rigor data processing theory. In 1794 Carl Friedrich Gauss pointed out the Least Squares method (LS), and it could relatively solve the problems of random errors by observation vector. After that Markov had systematic discoursed on LS, and reached the famous Gauss-Markov Model. In the classical Least Squares estimation process, for the linear model $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}$, only the errors of observation vector y are considered, and the design matrix A is assumed to be accurate without any errors. In many cases, the sample error, model error, instrument error, and other factors often cause the design matrix A to become incompletely accurate. Which means the classical Least Squares method is no longer reach the optimal solution in these cases. Further study based on Errors-in-Variables(EIV) are wildly discussed. How to obtain the best parameter estimation values and give the statistical information of parameters in the EIV model is not 'perfect' solved. Nevertheless, the EIV model is still becoming increasingly widespread in remote sensing(Felus and Schaffrin 2005) and geodetic datum processing(Schaffrin and Felus 2006,2008; Akyilmaz 2007; Michael K and Mathias A 2007; Cai J and Grafarend EW 2009).

A data processing idea was rediscovered many times in a 2D linear fitting model and subsequently named Total Least Squares problem. The origin of this basic idea can be traced back to the beginning of the last centry. In 1980, the mathematical structure of Total Least Squares(TLS) was completed by Golub and Van Loan(1980), who gave the first numerically stable algorithm based on matrix singular value decomposition. With the rapid development of the numerical method over the last decade, various approach for TLS emerged. These include singular value decompositin (SVD), the completely orthogonal approach, the Cholesky decompositon approach, the iterative approach, and so on(Ding 2007; Qiu 2008; Kong 2010; Golub 1980; Van Huffel 1993,1997,2002; Schaffrin et al.2003; Markovsky et al.2006), the most representative of which are the SVD and iterative solutions. However, there are some problems in the both methods. In the SVD method, some elements of design matrix may be non-stochastic, or some elements containing errors could appear more than once. To perform the minimum norm constraint without this consideration is inappropriate and may result in large deviations. By the Iteration method, since the iteration is the gradual approximation of the true value of parameter, iteration solutions can be a problem if there is a high degree of nonlinearity. And this method also has the problem by the repetition of parameters in design matrix.

Recent years, a further method reformed from EIV model called Partial-EIV has a relative good solution to the problems which mentioned above. However almost all the Partial-EIV models focus on the calculation with iteration. Which makes the mathematical algorithm very complicate. According to the research results by Jianqing Cai, Nico Sneeuw, Yibin Yao and Jian Kong, one method called Converted Total Least Squares (CTLS) was developed since 2010.

This method has been culculated without iteration and at the same time solve the problem by the repetition of elements and the non-stochastic elements containing errors in design matrix. In this thesis the two transformation models (6-parameter affine transformation model and 7-parameter Helmert transformation model) are firstly reviewed. They will be analyzed by using 131 BWREF points in Baden-Württemberg. Then the mathematic foundation of TLS method is reviewed. Two different solutions SVD (Van Huffel, 1991) and iteration approach (Schaffrin, 2005) and the Partial-EIV model are introduced. And the representative experiments will be implemented through the coordinate transformation in Baden-Württemberg. As a comparison, the transformation parameters estimated by LS, TLS (SVD), Partial-EIV model and CTLS will be represented and discussed.

Chapter 2

Transformation models and data preparation

2.1 6-parameter affine transformation models(2D)

In the case of map coordinates, which result from the projection of the reference ellipsoid into plane, a two-dimensional model is more useful. For example, when between the respective reference systems (DHDN, Bessel and ETRS89, GRS80) no direct mathematical relationship exists. Two-dimensional transformation models are used. As a result, the Gauss-Krüger coordinates of the net points in DHDN can be transformed only over collocated points into UTM coordinates related to ETR89. For the two dimensional transformation models, there are three , four, five, or six transformation parameters, whose number depends on the respective requirements. Because the models with three or five parameters can make for some problems, e.g. non-linear equations problem, so they will not be considered here. In most applications of the plane-transformation, the 6-parameter affine transformation model is used and is recommended by the Surveying Authorities of the States of the Federal Republic of Germany. Therefor, the 6-parameter affine model will be reviewed and applied in estimating the parameters of the plane transformation parameters based on 131 collocated points in Baden-Württemberg(Cai, 2006).

With the planar affine transformation, where six parameters are to be determined, both coordinate directions are rotated with two different angles α and β. So that not only the distances and the angles are distorted, but usually also the original orthogonality of the axes of coordinates is lost. An affine transformation preserves collinearity and ratios of distances. While an affine transformation preserves proportions on lines, it does not necessarily preserve angles or lengths(Cai, 2006).

The 6-parameter affine transformation model between any two plane coordinates systems, e.g. from Grauss-Krüger coordinate(H,R) in DHDN (G) directly to the UTM-Coordinate (N,E) in ETRS89 can be written as

$$\begin{bmatrix} \boldsymbol{N} \\ \boldsymbol{E} \end{bmatrix} = \begin{bmatrix} \lambda_H cos\alpha & -\lambda_R sin\beta \\ \lambda_H sin\alpha & \lambda_R cos\beta \end{bmatrix} \begin{bmatrix} \boldsymbol{H} \\ \boldsymbol{R} \end{bmatrix} + \begin{bmatrix} t_N \\ t_E \end{bmatrix} \tag{2.1}$$

Where t_N and t_E are translation parameters; α and β are rotation parameters; λ_H and λ_R are scale corrections.

2.2 7-parameter Helmert transformation models(3D)

In most applications of three-dimensional transformation three seven parameter similarity transformation 3D Helmert model, also called after Bursa-Wolf (Bursa 1962, Wolf 1963), Molodensky-Badekas (Molodensky et al., 1960; Badekas, 1969) and Veis(1960) models are developed and used. Though the three seven parameter models are expressed in different forms with different origin and parameters, their transformation results are completely equialent. The description and the application of the Molodensky-Badekas models are referred to Heck(1995) and Ihde et al.(1995).

• Bursa-Wolf model

$$\begin{bmatrix} \boldsymbol{X}_G \\ \boldsymbol{Y}_G \\ \boldsymbol{Z}_G \end{bmatrix} = (1 + d\lambda) \begin{bmatrix} 1 & \gamma & -\beta \\ -\gamma & 1 & \alpha \\ \beta & -\alpha & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_L \\ \boldsymbol{Y}_L \\ \boldsymbol{Z}_L \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix} \tag{2.2}$$

• Molodensky-Badekas model

$$\begin{bmatrix} \boldsymbol{X}_G \\ \boldsymbol{Y}_G \\ \boldsymbol{Z}_G \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_L \\ \boldsymbol{Y}_L \\ \boldsymbol{Z}_L \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix} + \begin{bmatrix} 0 & \omega & -\psi \\ -\omega & 0 & \varepsilon \\ \psi & -\varepsilon & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_L - \boldsymbol{X}_{L0} \\ \boldsymbol{Y}_L - \boldsymbol{Y}_{L0} \\ \boldsymbol{Z}_L - \boldsymbol{Z}_{L0} \end{bmatrix} + d\lambda \begin{bmatrix} \boldsymbol{X}_L - \boldsymbol{X}_{L0} \\ \boldsymbol{Y}_L - \boldsymbol{Y}_{L0} \\ \boldsymbol{Z}_L - \boldsymbol{Z}_{L0} \end{bmatrix} \tag{2.3}$$

• Veis model

$$\begin{aligned} \begin{bmatrix} \boldsymbol{X}_G \\ \boldsymbol{Y}_G \\ \boldsymbol{Z}_G \end{bmatrix} &= \begin{bmatrix} \boldsymbol{X}_L \\ \boldsymbol{Y}_L \\ \boldsymbol{Z}_L \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix} + d\lambda \begin{bmatrix} \boldsymbol{X}_L - \boldsymbol{X}_{L0} \\ \boldsymbol{Y}_L - \boldsymbol{Y}_{L0} \\ \boldsymbol{Z}_L - \boldsymbol{Z}_{L0} \end{bmatrix} + \\ &\begin{bmatrix} 0 & \boldsymbol{Z}_{L0} - \boldsymbol{Z}_L & \boldsymbol{Y}_L - \boldsymbol{Y}_{L0} \\ \boldsymbol{Z}_L - \boldsymbol{Z}_{L0} & 0 & \boldsymbol{X}_{L0} - \boldsymbol{X}_L \\ \boldsymbol{Y}_{L0} - \boldsymbol{Y}_L & \boldsymbol{X}_L - \boldsymbol{X}_{L0} & 0 \end{bmatrix} \begin{bmatrix} -sinB_{L0}cosL_{L0} & -sinL_{L0} & cosB_{L0}cosL_{L0} \\ -sinB_{L0}sinL_{L0} & cosL_{L0} & cosB_{L0}sinL_{L0} \\ cosB_{L0} & 0 & sinB_{L0} \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \end{aligned} \tag{2.4}$$

The similarity of the transformation is particularly important since the conformal characteristic of the coordinates after the transformation are maintained. They are applied particular for the discrepancies between a local (e.g. DHDN related Bessel ellipsoid) and a global reference system(e.g. ETRS89 related GRS80 Ellipsoid) which are due to the differences in the geodetic datum. However, the three seven parameter models are expressed in different forms with different origin and parameters, their transformation results are completely equivalent. So 7-parameter Helmert model is used most commonly(Cai, 2006).

The further reasons for the choice of 7-parameter Helmert transformation model are:

- It is the only known method which allows a direct interpretation of origin shifts.
- The rotations around the 'Earth-Centered, Earth-Fixed(ECEF)' Cartesian axes can have physical interpretations in global reference frames.
- It perform a conformal transformation, where the ratios of distances and the angles preserve invariantly.(Cai, 2006)

It performs a conformal transformation, where the ratios of distances and the angles preserve invariantly. A 'local' non-geocentric X_L,Y_L,Z_L-system can be transformed into a 'global' geocentric X_G,Y_G,Z_G-system with the help of a 7-parameter Helmert transformation model.

$$\begin{aligned}\begin{bmatrix}\boldsymbol{X}_G\\ \boldsymbol{Y}_G\\ \boldsymbol{Z}_G\end{bmatrix} &= \begin{bmatrix}T_X\\ T_Y\\ T_Z\end{bmatrix} + (1+d\lambda)\begin{bmatrix}1 & \gamma & -\beta\\ -\gamma & 1 & \alpha\\ \beta & -\alpha & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{X}_L\\ \boldsymbol{Y}_L\\ \boldsymbol{Z}_L\end{bmatrix}\\ &\cong \begin{bmatrix}T_X\\ T_Y\\ T_Z\end{bmatrix} + \begin{bmatrix}d\lambda & \gamma & -\beta\\ -\gamma & d\lambda & \alpha\\ \beta & -\alpha & d\lambda\end{bmatrix}\begin{bmatrix}\boldsymbol{X}_L\\ \boldsymbol{Y}_L\\ \boldsymbol{Z}_L\end{bmatrix} + \begin{bmatrix}\boldsymbol{X}_L\\ \boldsymbol{Y}_L\\ \boldsymbol{Z}_L\end{bmatrix}\end{aligned} \tag{2.5}$$

Where T_X,T_Y,T_Z are translate parameters; α , β and γ are differential rotation parameters; $d\lambda$ is scale correction.

2.3 Data preparation and the analysis with collocated points in Baden-Württemberg

In preparation of local coordinate of collocated points the Gauss-Krüger coordinates of DHDN will be transformed to Bessel ellipsoidal coordinates latitude (B_L) and longitude (L_L) through inverse conformal mapping formulas and the conversion of ellipsoidal coordinates (B_L,L_L,H_L) to geodetic Cartesian coordinates (X_L,Y_L,Z_L) and the reverse conversion are accomplished through the general formula. For the global coordinates can be also converted with the same algorithms on the GRS80 ellipsoid. Then we can construct the quasi-observations with the 131 collocated points (131 BWREF points in Baden-Württemberg) and perform the estimation of the transformation parameters of the 7-parameter Helmert transformation and the 6-parameter affine transformation. The transformation parameter solutions using above two models are listed in table I and the residuals are illustrated in figure 2.1 and figure 2.2.

From the distribution of horizontal residuals shown in figure 1 by 3-D 7-parameter Helmert transformation and figure 2 by 2-D 6-Parameter affine transformation we can find that there are two rotational trends of the direction of the horizontal residual vectors which arc clockwise in the northern part and counter-clockwise in the southern part. The cause for it lies despite the homogeneity of the network structure in the remaining distortions of the DHDN, which is highly correlated over larger areas. The horizontal position residuals of these sites bordering (he boundary of the state of Baden-Württemberg arc larger than these inner sites. The latest residual occur in site 130 by 0.43 m. similar results can also be found in figure 2 by 2-D 6-parameter affine transformation (Cai, 2006).

After the transformation of DHDN/Gauss Krüger coordinates into ETRS89/UTM coordinates, the inherent traditional network distortions of the DHDN in Baden- Württemberg (BWREF) can be visually shown through the residuals in 131 collocated points. Since the special characteristic of the main triangulation network in Baden- Württemberg (statewide variable net scales, inhomogeneous point accuracies and network distortions in the decimeter level) a statewide similarity or affine transformation parameter set cannot satisfy the requirement of the transformation accuracy in Baden-Württemberg (Cai, 2006).

Table 2.1: *Transformation parameters and their standard deviation with 131 BWREF points*

6-parameter affine transformation GK(DHDN)-UTM(ETRS89)							
$t_N(m)$	$t_E(m)$	$\alpha('')$	$\beta('')$	$d\lambda_H(\times 10^{-4})$	$d\lambda_R(\times 10^{-4})$	$RMS^*(m)$	$\hat{\sigma}(m)$
437.1946	119.7567	0.1654	-0.1965	-3.9968	-3.9884	0.1187	0.1199

7-parameter Helmert transformation GK(DHDN)-UTM(ETRS89)								
$T_X(m)$	$T_Y(m)$	$T_Z(m)$	$\alpha('')$	$\beta('')$	$\gamma('')$	$d\lambda(\times 10^{-6})$	$RMS^*(m)$	$\hat{\sigma}(m)$
582.9017	112.1681	405.6031	-2.2550	-0.3350	2.0684	9.1172	0.1241	0.1026

(RMS^*: quadratic means of the horizontal residuals)

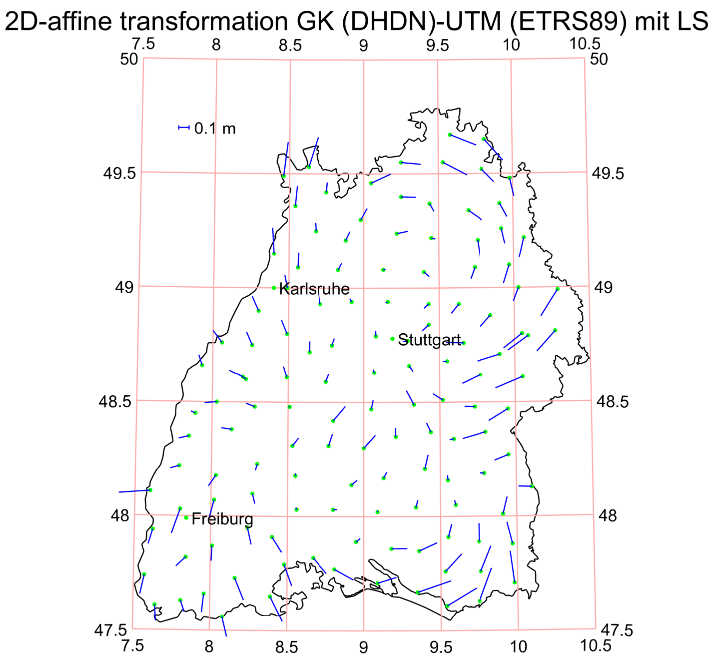


Figure 2.1: *Horizontal residuals after 6-parameter affine transformation in Baden-Württemberg network*

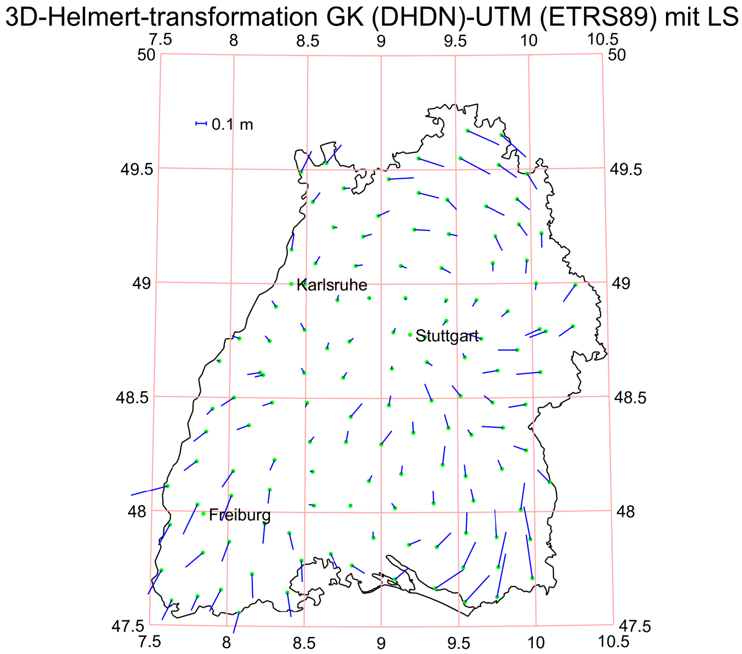


Figure 2.2: *Horizontal residuals after 7-parameter Helmert transformation in Baden-Württemberg network*

Chapter 3

Total Least Squares

3.1 Introduction

The total least squares method is one of several linear parameter estimation techniques that have been devised to compensate for data errors. The basic motivation for total least squares (TLS) is the following: Let a set of multidimensional data points (vectors) be given. How can one obtain a linear model that explains these data? The idea is to modify all data points in such a way that some norm of the modification is minimized subject to the constraint that the modified vectors satisfy a linear relation. (Van Huffel, 1991)
The origin of this basic idea can be traced back to the beginning of last century. It was rediscovered many times, often independently, mainly in the statistical and psychometric literature. However, it is only in the 1980s and 1990s that the technique also found wide use in scientific and engineering applications. One of the main reasons for its popularity is the availability of efficient and numerical robust algorithms, in which the singular value decomposition plays a prominent role. Another reason is the fact the TLS is an application oriented procedure. It is ideally suited for situations in which all data are corrupted by noise, which is almost always the case in engineering applications. In this sense, it is a powerful extension of the classical least squares method, which corresponds only to a partial modification of the data.
The problem of linear parameter estimation arises in a broad class of scientific disciplines such as signal processing, automatic control, system theory and in general engineering, statistics, physics, economics, biology, medicine, etc... It starts from a model described by a linear equation.

$$y = a_1\xi_1 + \cdots + a_m\xi_m \tag{3.1}$$

Where $a_1, \cdots, a_m$ and y denote the variables and $\boldsymbol{\xi} = [\xi_1, \cdots, \xi_m]^T \in R^m$ plays role of a parameter vector that characterizes the specific system. The basic problem is then to determine an estimate of the true but unknown parameters from certain measurements of the variables. This gives rise to an overdetermined set of m linear equations$(m > n)$:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\xi} + \boldsymbol{e_y} \tag{3.2}$$

Where the $i - th$ row of the data matrix $\boldsymbol{A} \in R^m$ and the vector of observations $y \in R^n$ contain the measurements of the variables $a_1, \cdots, a_m$ and y, respectively. In the classical least squares (LS) approach the measurements $\boldsymbol{A}$ of the variables a_i (the right-hand side of (3.2)) are assumed to be free of error and hence, all errors are confined to the observation vector $\boldsymbol{y}$ (the left-hand side of (3.2)). However, this assumption is frequently unrealistic: sampling errors, human errors, modelling errors and instrument errors may imply inaccuracies of the data matrix $\boldsymbol{A}$ as well. TLS is one method of fitting that is appropriate when there are errors in both the observation vector $\boldsymbol{y}$ and the data matrix $\boldsymbol{A}$. It amounts to fitting a 'best' subspace to the measurement data $(\boldsymbol{A}_i^T, y_i)$,$i = 1, \cdots, n$, where $\boldsymbol{A}_i^T$ is the $i - th$ row of $\boldsymbol{A}$. To illustrate the effect of TLS in comparison with LS, we consider here a simple example of parameter estimation, i.e., only one parameter $(m = 1)$ is to be estimated. Hence, equation(3.1) reduces to

$$\boldsymbol{y} = a\boldsymbol{\xi} \tag{3.3}$$

An estimate for parameter $\boldsymbol{\xi}$ is to be found from n measurements of the variables a and $\boldsymbol{y}$:

$$\begin{aligned} a_i &= a_i^0 + \Delta a_i \\ y_i &= y_i^0 + \Delta y_i \qquad i = 1, \cdots, n \end{aligned} \tag{3.4}$$

By solving the linear system (3.2) with $\boldsymbol{A} = [a_1, \cdots, a_n]^T$ and $\boldsymbol{b} = [b_1, \cdots, b_n]^T$. Δa_i and Δy_i represent the random errors added to the true values a_i^0 and y_i^0 of the variables a and y. If a can be observed exactly, i.e., $\Delta a_i = 0$, errors only occur in the measurements of $\boldsymbol{y}$ contained in the right-hand side vector $\boldsymbol{y}$. Hence, the use of LS for solving (3.2) is appropriate. This method perturbs the observation vector y by a minimum amount $\boldsymbol{e}$ so that $(\boldsymbol{y} - \boldsymbol{e})$ can be predicted by $\boldsymbol{A\xi}$. This is done by minimizing the sum of squared and differences

$$\sum_{i=1}^{n} (y_i - a_i\boldsymbol{\xi})^2 \tag{3.5}$$

The best estimate $\hat{\xi}_y$ of ξ follows then immediately:

$$\hat{\boldsymbol{\xi}}_y = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{y} = \frac{\sum_{i=1}^n a_i y_i}{\sum_{i=1}^n a_i^2} \tag{3.6}$$

If $\boldsymbol{y}$ can be measured without error, i.e., $\Delta y_i = 0$, the use of LS is again appropriate. Indeed we can rewrite as

$$\frac{\boldsymbol{y}}{\boldsymbol{\xi}} = a \tag{3.7}$$

And confine all errors to the measurements of α contained in the right-hand side vector $\boldsymbol{A}$ of the corresponding set of equations $\boldsymbol{A} \approx \boldsymbol{y\xi^{-1}}$. By minimizing the sum of squared differences between the measured values a_i and the predicted values $\boldsymbol{y}_i/\boldsymbol{\xi}$, the best estimate $\hat{\boldsymbol{\xi}}_A$ of $\boldsymbol{\xi}$ is given by

$$\hat{\boldsymbol{\xi}}_y = (\boldsymbol{y}^T\boldsymbol{y})^{-1}\boldsymbol{y}^T\boldsymbol{A} = \frac{\sum_{i=1}^n a_i^2}{\sum_{i=1}^n a_i y_i} \tag{3.8}$$

In many application, however, both variables are measured with errors, i.e., $\Delta a_i \neq 0$ and $\Delta y_i \neq 0$. If the errors are independently and identically distributed with zero mean and common variance σ_v^2, the best estimate $\hat{\boldsymbol{\xi}}$ is obtained by minimizing the sum of squared distances of the observed points from the fitted line, i.e.,

$$\sum_{i=1}^{n} (y_i - a_i\boldsymbol{\xi})^2/(1 + \boldsymbol{\xi}^2) \tag{3.9}$$

This is in fact the solution $\hat{\xi}_{TLS}$ we obtain by solving (3.2) with the TLS method for $m = 1$.

3.2 Total Least Squares with singular value decomposition(SVD)

The singular value decomposition(SVD) is of great theoretical and practical importance for the LS and TLS problems. Not only does it provide elegant geometrical and algebraic insights into many numerical linear algebra problems, but also at the same time, a numerically reliable algorithm can be devised.

If $\boldsymbol{C} \in \boldsymbol{R}^{n\times m}$ then there exist orthonormal matrices $\boldsymbol{U} = [u_1, \cdots, u_n] \in \boldsymbol{R}^{n\times m}$ and $\boldsymbol{V} = [v_1, \cdots, v_n] \in \boldsymbol{R}^{n\times m}$ such that (Van Huffel, 1991)

$$\boldsymbol{U}^T\boldsymbol{C}\boldsymbol{V} = \sum = diag(\sigma_1, \cdots, \sigma_p), \sigma_1 \geq \cdots \geq \sigma_p \geq 0 \quad and \quad p = min(n, m) \tag{3.10}$$

The σ_i are are the singular values of $\boldsymbol{C}$ and they are collectively known as the singular value spectrum. The vectors u_i and v_i are the $i-th$ left singular vector and the $i-th$ right singular vector, respectively. The triplet $u_i, sigma_i, v_i$ is called a singular triplet. It is easy to verify by comparing columns in the equations $\boldsymbol{C}\boldsymbol{V} = \boldsymbol{U}\sum$ and $\boldsymbol{C}^T\boldsymbol{U} = \sum^T \boldsymbol{V}$ that

$$\boldsymbol{C}v_i = \sigma_i u_i \quad and \quad \boldsymbol{C}^T u_i = \sigma_i v_i \qquad i = 1, \cdots, p \tag{3.11}$$

The SVD reveals many interesting structures of a matrix. If the SVD of C is given by Theorem 1, and we define r by

$$\sigma_1 \geq \cdots \geq \sigma_r \geq \sigma_{r+1} = \cdots = \sigma_p = 0$$

the number of positive singular values, then

$$\begin{aligned} rank(\boldsymbol{C}) &= r, \\ \boldsymbol{R}(\boldsymbol{C}) &= \boldsymbol{R}([u_1, \cdots, u_r]), \\ \boldsymbol{N}(\boldsymbol{C}) &= \boldsymbol{R}([v_{r+1}, \cdots, v_m]), \\ \boldsymbol{R}_r(\boldsymbol{C}) &= \boldsymbol{R}(\boldsymbol{C}^T) = \boldsymbol{R}([v_1, \cdots, v_r]), \\ \boldsymbol{N}_r(\boldsymbol{C}) &= \boldsymbol{N}(\boldsymbol{C}^T) = \boldsymbol{R}([u_{r+1}, \cdots, u_n]). \end{aligned} \tag{3.12}$$

Moreover, if $\boldsymbol{U}_r = [u_1, \cdots, u_r]$, $\sum_r = diag(\sigma_1, \cdots, \sigma_p)$ and $\boldsymbol{V}_r = [v_1, \cdots, v_r]$, then we have the SVD expansion

$$\boldsymbol{C} = \boldsymbol{U}_r \sum_r \boldsymbol{V}_r^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T \tag{3.13}$$

Above equation, which is also called the dyadic decomposition, decomposes the matrix $\boldsymbol{C}$ of rank r in a sum of r matrices of rank one. Also, the 2-norm an the Frobenius norm are neatly characterized in terms of the SVD:

$$\|\boldsymbol{C}\|_{\boldsymbol{F}}^2 = \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}^2 = \sigma_1^2 + \cdots + \sigma_p^2, p = min(n, m)$$

$$\|\boldsymbol{C}\|_2 = \sup_{y\neq 0} \frac{\|\boldsymbol{C}\boldsymbol{y}\|_2}{\|\boldsymbol{y}\|_2} = \sigma_1$$

With Eckart-Young-Mirsky matrix approximation theorem, let the SVD of $\boldsymbol{C} \in \boldsymbol{R}^n \times m$ be given by $\boldsymbol{C} = \sum_{i=1}^{r} \sigma_i u_i v_i^T$ with $r = rank(\boldsymbol{C})$. If $k < r$ and $\boldsymbol{C}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$, then

$$
\begin{aligned}
\min_{rank(\boldsymbol{D})=k} \|\boldsymbol{C}-\boldsymbol{D}\|_2 &= \|\boldsymbol{C}-\boldsymbol{C}_k\|_2 = \sigma_{k+1} \\
\min_{rank(\boldsymbol{D})=k} \|\boldsymbol{C}-\boldsymbol{D}\|_F &= \|\boldsymbol{C}-\boldsymbol{C}_k\|_F = \sqrt{\sum_{i=k+1}^{p} \sigma_i^2}, p = min(n,m)
\end{aligned}
\tag{3.14}
$$

Eckart and Young originally proved the theorem for the Frobenius norm in 1936. In 1960, Mirsky proved the theorem for the 2-norm. Therefore, Theorem 2 is called the Eckart-Young-Mirsky Theorem (Van Huffel, 1991).

On the basis of the above mathematical theorems, convert model $\boldsymbol{Ax} = \boldsymbol{b}$ to $\begin{bmatrix} \boldsymbol{A} & \boldsymbol{b} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ -1 \end{bmatrix} = 0$. If it performs SVD on matrix $\begin{bmatrix} \boldsymbol{A} & \boldsymbol{b} \end{bmatrix}$, using the above theorem, following formulas can be derived under the criteria $\left\| \left[\hat{\boldsymbol{A}}; \hat{\boldsymbol{b}}\right] - \left[\boldsymbol{A}; \boldsymbol{b}\right] \right\|_2 = min$.

$$
\begin{aligned}
\hat{x} &= -\frac{1}{v_{t+1,t+1}} \left[v_{1,t+1} \cdots v_{t+1,t+1}\right]^T \\
\begin{bmatrix} \Delta\boldsymbol{A} & \Delta\boldsymbol{b} \end{bmatrix} &= \begin{bmatrix} \hat{\boldsymbol{A}} & \hat{\boldsymbol{b}} \end{bmatrix} - \begin{bmatrix} \boldsymbol{A} & \boldsymbol{b} \end{bmatrix} = \sigma_{t+1} u_{t+1} v_{t+1}^T
\end{aligned}
\tag{3.15}
$$

Where t is the number of parameters. Although this algorithm is based on established mathematical theorems, it has a weakness. Golub and Van Loan presumed that A, b are stochastic elements and performed SVD directly on the matrix $\begin{bmatrix} \hat{A} & \hat{b} \end{bmatrix}$. However, some elements of A may be non-stochastic, or some elements containing errors could appear more than once. To perform the minimum norm constraint without this consideration is inappropriate and may result in large deviations.

3.3 Total Least Squares with the Euler-Lagrange approach

Another solution for TLS is the iterative method(Schaffrin, 2006). The most important feature of the iterative method is it's straightforward algorithm. Schaffrin and Felus(2003) have introduced a multivariate version of Total Least Squares (TLS) adjustment in order to determine the optimal parameters of an affine coordinate transformation empirically.

The following model, with full-rank matrix $\boldsymbol{A}$, is assumed (Schaffrin, 2003):

$$
\begin{aligned}
(\boldsymbol{A}-\boldsymbol{E}_A)\boldsymbol{\xi} - (\boldsymbol{y}-\boldsymbol{e}) &= 0 \\
\boldsymbol{E}\left\{[\boldsymbol{E}_A, \boldsymbol{e}]\right\} &= 0 \\
\boldsymbol{C}\left\{[\boldsymbol{E}_A, \boldsymbol{e}]\right\} &= 0 \\
\boldsymbol{D}\left\{vec\,[\boldsymbol{E}_A, \boldsymbol{e}]\right\} &= \boldsymbol{\Sigma_0} \otimes \boldsymbol{I}_n
\end{aligned}
\tag{3.16}
$$

Where $\boldsymbol{e}$ and $\boldsymbol{E_A}$ denote a random error vector, resp. matrix. $\boldsymbol{\Sigma_0} = \sigma_0^2 \boldsymbol{I}_{m+1}$ is a $(m+1) \times (m+1)$ matrix with an unknown variance component σ_0^2 and given identity matrix $\boldsymbol{I}_{m+1}$. The symbol $\otimes$ denotes the 'Kronecker-Zehfuss product' of matrices, defined by:

$$\boldsymbol{M} \otimes \boldsymbol{N} := \left[\boldsymbol{m}_{ij} \cdot \boldsymbol{N}\right]$$

For $\boldsymbol{M} = \boldsymbol{m}_{ij}$ and $\boldsymbol{N}$ arbitrary.

The 'vec' operator stacks one column of a matrix under the other, moving from left to right. In contrast to the Least-Squares(LS) method, this is based on the minimization of

$$\boldsymbol{e}^T \boldsymbol{e} = (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi})^T (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}) \tag{3.17}$$

Under the condition $\boldsymbol{E_A} = 0$, the(equally weighted) TLS principle is based on minimizing the objective function (Schaffrin, 2003):

$$\boldsymbol{e}^T \boldsymbol{e} + (vec\boldsymbol{E_A})^T (vec\boldsymbol{E_A}) = min(\boldsymbol{\xi}) \tag{3.18}$$

When performing an adjustment, it is sometimes necessary to fix some parameters to specific values. Here, a total least squares solution will be presented, along with an iteration scheme(Schaffrin, 2003).

In order to solve the TLS problem as presented in (3.18) and minimize the respective objective function in view of the model (3.16), we define the Lagrange target function as follows where

$$\begin{aligned} \boldsymbol{e^A} &: = vec(\boldsymbol{E_A}) \sim (0, \sigma_0^2 \boldsymbol{I_m} \otimes \boldsymbol{I_n}) \\ \boldsymbol{\Phi}(\boldsymbol{e}, \boldsymbol{e_A}, \boldsymbol{\lambda}, \boldsymbol{\xi}) &= \boldsymbol{e}^T \boldsymbol{e} + \boldsymbol{e}_A^T \boldsymbol{e_A} + 2\lambda^T \left[\boldsymbol{y} - \boldsymbol{e} - \boldsymbol{A}\boldsymbol{\xi} + \boldsymbol{E_A}\boldsymbol{\xi}\right] \end{aligned} \tag{3.19}$$

Where λ denotes the $n \times 1$ vector of Lagrange multipliers; note that

$$\boldsymbol{E_A}\boldsymbol{\xi} = (\boldsymbol{\xi}^T \otimes \boldsymbol{I_n})\boldsymbol{e_A} \tag{3.20}$$

thus the Euler-Lagrange necessary conditions are (Schaffrin, 2003):

$$\begin{aligned} \frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{e}} &= \hat{\boldsymbol{e}} - \hat{\boldsymbol{\lambda}} = 0 \\ \frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{e_A}} &= \hat{\boldsymbol{e}}_A - (\hat{\boldsymbol{\xi}} \otimes \boldsymbol{I_n})\hat{\boldsymbol{\lambda}} = 0 \\ \frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{\lambda}} &= \boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\xi}} - \hat{\boldsymbol{e}} + \hat{\boldsymbol{E}}_A\hat{\boldsymbol{\xi}} = 0 \\ \frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{\xi}} &= -\boldsymbol{A}^T\hat{\boldsymbol{\lambda}} + \hat{\boldsymbol{E}}_A^T\hat{\boldsymbol{\lambda}} = 0 \end{aligned}$$

This system is simplified into:

$$\begin{aligned} (\boldsymbol{A}^T\boldsymbol{A})\hat{\boldsymbol{\xi}} &= \boldsymbol{A}^T y + \hat{\boldsymbol{\xi}}(\hat{\boldsymbol{\lambda}}^T\hat{\boldsymbol{\lambda}})(1 + \hat{\boldsymbol{\xi}}^T\hat{\boldsymbol{\xi}}) \\ \hat{\boldsymbol{\lambda}} &= (\hat{e} - \hat{E}_A\hat{\boldsymbol{\xi}})(1 + \hat{\boldsymbol{\xi}}^T\hat{\boldsymbol{\xi}})^{-1} = (\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\xi}})(1 + \hat{\boldsymbol{\xi}}^T\hat{\boldsymbol{\xi}})^{-1} \end{aligned} \tag{3.21}$$

Note that

$$\hat{\boldsymbol{v}} = \frac{(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\xi}})^T(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\xi}})}{(1 + \hat{\boldsymbol{\xi}}^T\hat{\boldsymbol{\xi}})} = (\hat{\boldsymbol{\lambda}}^T\hat{\boldsymbol{\lambda}})(1 + \hat{\boldsymbol{\xi}}^T\hat{\boldsymbol{\xi}}) = \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}} + \hat{E}_A^T\hat{E}_A\tilde{\boldsymbol{\xi}} = min(\boldsymbol{\xi}) \tag{3.22}$$

is Rayleigh's quotient for the matrix

$$\begin{bmatrix} \boldsymbol{A}^T\boldsymbol{A} & \boldsymbol{A}^T\boldsymbol{y} \\ \boldsymbol{y}^T\boldsymbol{A} & \boldsymbol{y}^T\boldsymbol{y} \end{bmatrix} \tag{3.23}$$

with $\left[\hat{\boldsymbol{\xi}}^T, -1\right]^T$ as the vector argument. Rayleigh's quotient defines the minimum eigenvalue of the augmented matrix, based on the corresponding eigenvector(see, e.g.,G. Strang, 1988)(Schaffrin, 2003).
Using these equations the following algorithm had been developed by Schaffrin (2003) to solve the TLS problem (Schaffrin, 2003):

1) Compute the LS solution:

$$\hat{\boldsymbol{\xi}}^1 = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{y} \qquad (for \quad \hat{v}^0 := 0)$$

2) Insert the solution of step 1) as the innitial value for the following iterative process:

$$\hat{\boldsymbol{\xi}}^{i+1} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}[\boldsymbol{A}^T\boldsymbol{y} + \frac{(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\xi}}^i)^T(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\xi}}^i)}{(1 + (\hat{\boldsymbol{\xi}}^i)^T\hat{\boldsymbol{\xi}}^i)}]$$

3) End when $\left\|\hat{\boldsymbol{\xi}}^{i+1} - \hat{\boldsymbol{\xi}}^i\right\| < \varepsilon$. Then

$$\hat{\sigma}_0^2 = \frac{\hat{v}}{(n-m)}$$

The algorithm seems to converge to the TLS solution in most cases although it's efficiency (convergence rate, convergence radius, etc.) still needs to be further investigated. However, since the iterative method is the gradual approximation of the true value of parameter, iterative solutions can be a problem if there is a high degree of nonlinearity.

3.4 Total Least Squares based on Partial-EIV model

Total Least Squares has attracted a widely spread attention since Golub and van Loan (1980) coined the terminology of Total Least Squares and demonstrated that the TLS solution can be readily obtained algorithmically by singular value decomposition about 30 years ago. The Total Least squares method has been developed to deal with observation equations, which are functions of both unknown parameters of interest and other measured data contaminated with random errors. Such an observation model is well known as an errors-in-variables (EIV) model and almost always solved as a nonlinear equality-constrained adjustment problem. Xu, Liu and Shi (2012) reformulate it as a nonlinear adjustment model without constraints and further extend it to a Partial-EIV model, in which not all the elements of the design matrix are random. As a result, the total number of unknowns in the normal equations has been significantly reduced.

The EIV observation model is defined as:

$$\boldsymbol{y} - \boldsymbol{e_y} = (\boldsymbol{A} - \boldsymbol{E_A})\boldsymbol{\xi} \tag{3.24}$$

Where $\boldsymbol{y}$ denotes the $m \times 1$ observation vector, $\boldsymbol{A}$ represents the the $m \times n$ coefficient matrix with rank$(\boldsymbol{A}) = n < m$. $\boldsymbol{\xi}$ represents the $n \times 1$ unknown parameter vector. Moreover, $\boldsymbol{e_y}$ denotes the random error vector of $\boldsymbol{y}$, and $\boldsymbol{E_A}$ denotes the random error matrix of $\boldsymbol{A}$. $\boldsymbol{e_y}$ is often supposed to be of zero mean and a variance-covariance matrix $\sigma_0^2 \boldsymbol{Q_y}$, with $\boldsymbol{Q_y}$ being a given positive definite cofactor matrix and σ_0^2 an variance of unit weight. $\boldsymbol{e_A} = vec\boldsymbol{E_A}$, in which 'vec' denotes the operator which stacks one column of the matrix underneath the previous one, $\boldsymbol{e_A}$ is also assumed to be zero mean and the variance-covariance matrix is defined as $\sigma_0^2 \boldsymbol{Q_y}$, with $\boldsymbol{Q_A}$ being the cofactor matrix of $\boldsymbol{e_A}$, $\boldsymbol{Q_A}$ is singular when the matrix $\boldsymbol{A}$ contains non-random elements. $\boldsymbol{e_y}$ and $\boldsymbol{e_A}$ is uncorrelated.

Xu *etal*.(2012) transformed the EIV model into a partial-EIV model by extracting functionally independent random variables within the coefficient matrix:

$$\boldsymbol{y} - \boldsymbol{e_y} = (\boldsymbol{\xi}^T \otimes \boldsymbol{I_m}) \left[\boldsymbol{h} + \boldsymbol{B}(\boldsymbol{a} - \boldsymbol{e_a})\right] \tag{3.25}$$

where $\boldsymbol{I_m}$ is the $m - th$ order identity matrix, $\boldsymbol{a}$ is a $t \times 1$ vector of functionally independent variables within $\boldsymbol{A}$, $\boldsymbol{e_a}$ denotes the random error vector of $\boldsymbol{a}$, $\boldsymbol{h}$ is a deterministic constant vector whose elements correspond to the non-random elements of $\boldsymbol{A}$, $\boldsymbol{B}$ is a given deterministic matrix with a dimension of $mn \times t$. Obviously, $\boldsymbol{A}$ can be expressed as:

$$\boldsymbol{A} = ivec(\boldsymbol{h} + \boldsymbol{Ba}) \tag{3.26}$$

where 'ivec' represents the inverse operator of 'vec' , which recovers the $mn \times 1$ vector to the original matrix with a dimension of $m \times n$.
The corresponding stochastic model of the partial EIV model is:

$$\begin{bmatrix} \boldsymbol{e_y} \\ \boldsymbol{e_a} \end{bmatrix} \sim \left(\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \quad \sigma_0^2 \begin{bmatrix} \boldsymbol{Q_y} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q_a} \end{bmatrix} \right) \tag{3.27}$$

where $\boldsymbol{Q_a}$ denotes a given positive definite cofactor matrix of $\boldsymbol{e_a}$.
We assume that the obtained parameter estimator vector after $i - th$ iteration is $\boldsymbol{\xi}_{(i)}$, and the predictive residual vector of $\boldsymbol{a}$ is $\boldsymbol{e}_{a(i)}$. The right-hand member of (3.25) is expressed through Taylor series expansion at $\left(\boldsymbol{X}_{(i)}, \boldsymbol{e}_{a(i)}\right)$:

$$\begin{aligned} \boldsymbol{y} - \boldsymbol{e_y} &= \left(\boldsymbol{X}_{(i)}^T \otimes \boldsymbol{I_m}\right)(\boldsymbol{h} + \boldsymbol{Ba}) - \left(\boldsymbol{X}_{(i)}^T \otimes \boldsymbol{I_m}\right) \boldsymbol{Be_a} + ivec(\boldsymbol{h} + \boldsymbol{B}(\boldsymbol{a} - \boldsymbol{e}_{a(i)}))\delta\boldsymbol{\xi} \\ &= \boldsymbol{A}\boldsymbol{\xi}_{(i)} + \boldsymbol{A}_{(i)}\delta\boldsymbol{\xi} - (\boldsymbol{\xi}_{(i)}^T \otimes \boldsymbol{I_m})\boldsymbol{B}\boldsymbol{e}_a \end{aligned} \tag{3.28}$$

where

$$\boldsymbol{A}_{(i)} = ivec(\boldsymbol{h} + \boldsymbol{B}(\boldsymbol{a} - \boldsymbol{e}_{a(i)})) = \boldsymbol{A} - ivec(\boldsymbol{B}\boldsymbol{e}_{a(i)}) \tag{3.29}$$

In (3.28), the terms of the second and higher orders are omitted, whereas only the first order terms are maintained. $\delta\boldsymbol{\xi}$ denotes the small corrected values of $\boldsymbol{\xi}$.
The Lagrange objective function of TLS is constructed as:

$$\begin{aligned} &\boldsymbol{\Phi}(\boldsymbol{e_y}, \boldsymbol{e_a}, \boldsymbol{\xi}, \boldsymbol{\lambda}) = \\ &\boldsymbol{e}_y^T \boldsymbol{Q}_L^{-1} \boldsymbol{e_y} + \boldsymbol{e}_a^T \boldsymbol{Q}_a^{-1} \boldsymbol{e}_a + 2\lambda^T (\boldsymbol{y} - \boldsymbol{e}_y - \boldsymbol{A}\boldsymbol{\xi}_{(i)} - \boldsymbol{A}_{(i)}\delta\boldsymbol{\xi} + (\boldsymbol{\xi}_{(i)}^T \otimes \boldsymbol{I_m})\boldsymbol{B}\boldsymbol{e}_a) \end{aligned} \tag{3.30}$$

where λ is the $m \times 1$ vector of 'Lagrange multipliers', the solution of this target function can be derived via the Euler-Lagrange necessary conditions:

$$\begin{aligned}
\frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{e}_y} &= \boldsymbol{Q}_L^{-1}\hat{\boldsymbol{e}} - \hat{\boldsymbol{\lambda}} = 0 \\
\frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{e}_a} &= \boldsymbol{Q}_a^{-1}\hat{\boldsymbol{e}}_a + \boldsymbol{B}^T(\xi_{(i)} \otimes \boldsymbol{I_m})\hat{\boldsymbol{\lambda}} = 0 \\
\frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \xi} &= -\boldsymbol{A}_{(i)}^T\hat{\boldsymbol{\lambda}} = 0 \\
\frac{1}{2}\frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{\lambda}} &= \boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}_{(i)} - \hat{\boldsymbol{e}}_y + (\boldsymbol{\xi}_{(i)}^T \otimes \boldsymbol{I_m})\boldsymbol{B}\hat{\boldsymbol{e}}_a = 0
\end{aligned} \tag{3.31}$$

The following equations are derived though solving (3.31):

$$\delta\hat{\boldsymbol{\xi}}_{(i+1)} = (\boldsymbol{A}_{(i)}^T\boldsymbol{Q}_{c(i)}^{-1}\boldsymbol{A}_{(i)})^{-1}\boldsymbol{A}_{(i)}^T\boldsymbol{Q}_{c(i)}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}_{(i)}) \tag{3.32}$$

with

$$\hat{\boldsymbol{\xi}}_{(i+1)} = \delta\hat{\boldsymbol{\xi}}_{(i+1)} + \boldsymbol{\xi}_{(i)}$$

then we get the new $\hat{\boldsymbol{\xi}}_{(i+1)}$

$$\delta\hat{\boldsymbol{\xi}}_{(i+1)} = (\boldsymbol{A}_{(i)}^T\boldsymbol{Q}_{c(i)}^{-1}\boldsymbol{A}_{(i)})^{-1}\boldsymbol{A}_{(i)}^T\boldsymbol{Q}_{c(i)}^{-1}(\boldsymbol{y} - (\boldsymbol{\xi}_{(i)}^T \otimes \boldsymbol{I_m})\boldsymbol{B}\boldsymbol{e}_{a(i)}) \tag{3.33}$$

$$\hat{\boldsymbol{e}}_{\boldsymbol{y}(i+1)} = \boldsymbol{Q}_y\boldsymbol{Q}_{c(i)}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}_{(i)} - \boldsymbol{A}_{(i)}\delta\hat{\boldsymbol{\xi}}_{(i+1)}) \tag{3.34}$$

$$\hat{\boldsymbol{e}}_{a(i+1)} = \boldsymbol{Q}_a\boldsymbol{B}^T(\boldsymbol{\xi}_{(i)} \otimes \boldsymbol{I_m})\boldsymbol{Q}_{c(i)}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}_{(i)} - \boldsymbol{A}_{(i)}\delta\hat{\boldsymbol{\xi}}_{(i+1)}) \tag{3.35}$$

In equations (3.35-3.38),

$$\boldsymbol{Q}_{c(i)} = \boldsymbol{Q}_y + (\boldsymbol{\xi}_{(i)}^T \otimes \boldsymbol{I_m})\boldsymbol{B}\boldsymbol{Q}_a\boldsymbol{B}^T(\boldsymbol{\xi}_{(i)} \otimes \boldsymbol{I_m}) \tag{3.36}$$

The iterative process described in equations (3.32-3.35) can be implemented from LS solution. A small positive number threshold ε should be primarily presented to terminate iteration until $\left\|\delta\boldsymbol{\xi}_{\hat{(i+1)}}\right\| < \varepsilon$, where $\|\cdot\|$ represents l_2-norm of a vector. This algorithm reduced the number of unknowns and directly deal with the positive cofactor matrix $\boldsymbol{Q}_a$ instead of $\boldsymbol{Q}_A$. However, we can see from the equations above that, the mathematical algorithm is very complicate. Which makes it quite difficult to grasp.

Chapter 4

Converted Total Leat Squares

4.1 Introduction and mathematical foundation

The Converted Total Least Squares (CTLS) is proposed for dealing with the errors-in-variables (EIV) model. Firstly take classic Gauss-Markov model of LS as basis equation.

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\xi} + \boldsymbol{e}_y \tag{4.1}$$

Taking into account the design matrix's errors in model (4.1) will lead to difficulties for parameter estimation and accuracy assessment. Particularly, one cannot apply the traditional error propagation law directly, since die law is established on the basis of linear relations.
The basic idea of CTLS is to take the stochastic design matrix elements as virtual observation. On the basis of the original error equation, the number of observation equation is increased by taking the design matrix elements as the observation vector, and some of the design matrix elements are estimated as parameters in the new algorithm. The advantage of such strategy is the ability to obtain the adjusted value of required parameters, the design matrix is formed by the initial value of design matrix parameters, which no longer has random properties. The parameters obtained are the linear functions of the observation vector. After this treatment, (4.1) is combined with the classical LS adjustment theory.
Augmenting the observation equations that take design matrix elements as virtual observation on the basis of the original error equation.

$$\boldsymbol{y_a} = \boldsymbol{\xi_a} + \boldsymbol{e_a} \tag{4.2}$$

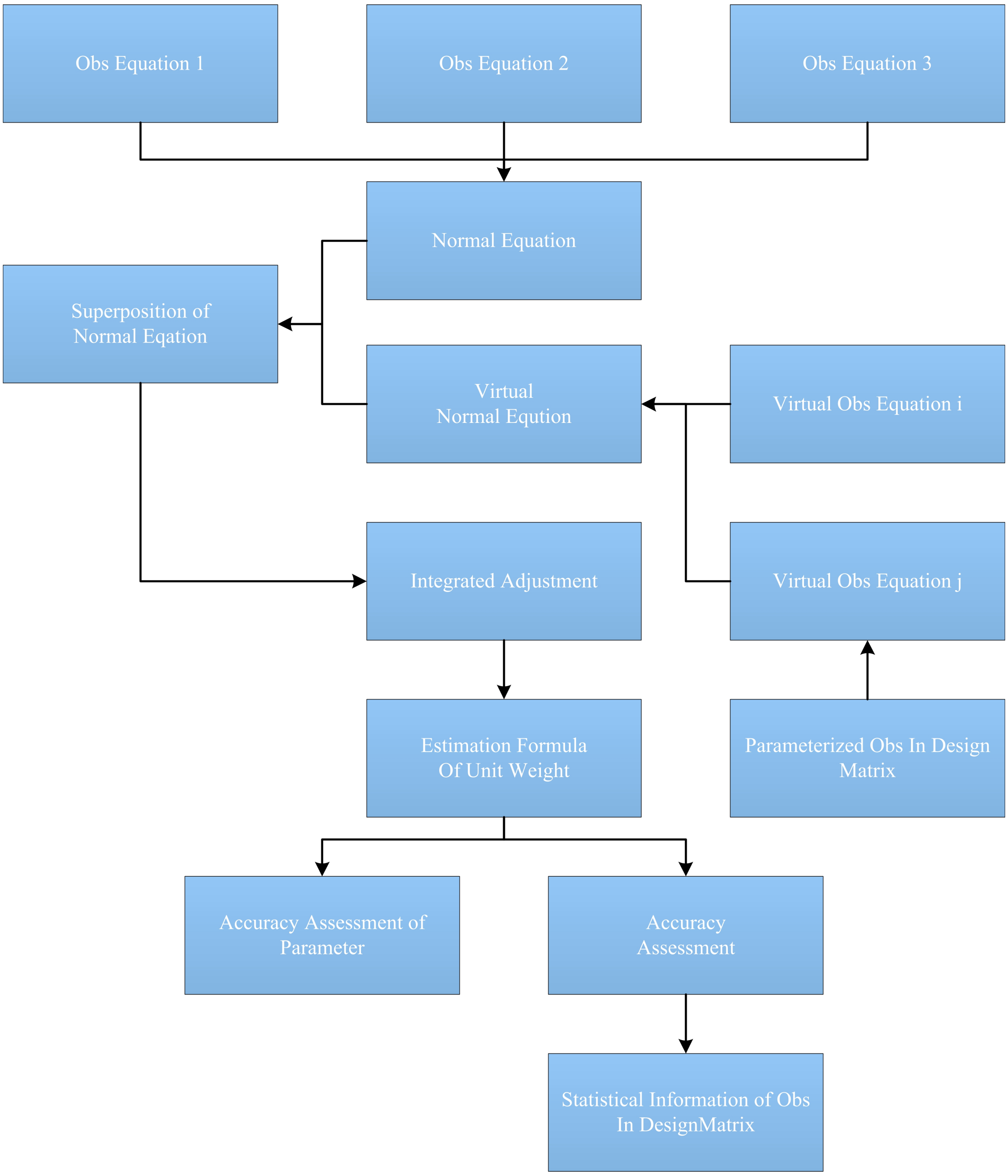


Figure 4.1: *Illustrate the basic idea of the algorithm*

Where y_a is comprised of the design matrix elements that contain errors, and ξ_a is comprised of the new parameters. If (4.1)is combine with (4.2), a mathematical model under the new algorithm can be obtained.

$$\begin{aligned} y &= A\xi + e_y \\ y_a &= \xi_a + e_a \end{aligned} \tag{4.3}$$

It should be clear that y_a contains only the observations of design matrix. To distinguish the design matrix in the original model, the symbol A_ξ is used to denote the design matrix in (4.1), which is formed by the initial value of parameters ξ_a and some elements without errors.

Based on the above model, we can get the following error equations

$$\begin{aligned} e_y &= (A^0_\xi + E_A)(\xi^0 + \Delta\xi) - y \\ &= A^0_\xi \Delta\xi + E_A \xi^0 + A^0_\xi \xi^0 - y + \Delta A \Delta\xi \qquad \rightarrow E_A \Delta\xi \approx 0 \\ &= A^0_\xi \Delta\xi + B\Delta a + A^0_\xi \xi^0 - y \\ e_a &= a - y_a \end{aligned} \tag{4.4}$$

Where E_A is composed of Δa, the corrections to the new parameters, and $B\Delta a$ is the rewritten form of $E_A\xi^0$. In converting $E_A\xi^0$ to $B\Delta a$, which is the key step for the approach. A^0_ξ is composed of non-stochastic elements in the design matrix and the initial value a.

Define $\eta = \begin{bmatrix} y - A^0_\xi \xi^0 \\ a - y_a \end{bmatrix}$, $A_\eta = \begin{bmatrix} A^0_\xi & B \\ 0 & E \end{bmatrix}$, $\Delta\eta = \begin{bmatrix} \Delta\xi \\ \Delta a \end{bmatrix}$, $e_\eta = \begin{bmatrix} e_y \\ e_a \end{bmatrix}$, (4.5) can be reduced to:

$$\eta = A_\eta \Delta\eta + e_\eta \tag{4.5}$$

Where e_η is the residual vector of all observations, A_η is formed by the initial values of the parameters, and $\Delta\eta$ is comprised of the corrections to all parameters. The estimation criterion is still $e^T_\eta e_\eta \rightarrow min$, which is the same as $e^T_y e_y + e^T_\eta e_\eta \rightarrow min$. Since the TLS problem is transformed into the classical LS problem, the adjustment can be completed by following the classical LS principle. The new weight matrix is $P_\eta = \begin{bmatrix} P_y & 0 \\ 0 & P_a \end{bmatrix}$. And the TLS problem can be solved considering the weight of observations and stochastic design matrix by:

$$\Delta\hat{\eta} = (A^T_\eta P_\eta A_\eta)^{-1} A^T_\eta P_\eta \eta \tag{4.6}$$

4.2 Estimation formula of unit weight variance

The estimation formula of unit weight variance of the TLS is difficult to determine. In considering the design matrix errors, the question of whether or not the degree of freedom for adjustment model changes arises. The TLS problem is converted into a classical LS problem using the new algorithm. With model (4.6), after adjustment by the LS principle, and V as the correct ion of observations, accuracy assessment is straightforward from the adjustment theory. (4.8) is the resulting formula of unit weight variance for TLS.

$$\sigma_0^2 = \frac{A^T_\eta P_\eta A_\eta}{tr(P_\eta Q_{VV})} = \frac{A^T_\eta P_\eta A_\eta}{(n+u)-(t+u)} = \frac{A^T_\eta P_\eta A_\eta}{n-t} \tag{4.7}$$

where n is the number of observations that the observation vector y contains, u is the number of stochastic elements in the design matrix A. t is the number of parameters, here only the number of original parameters in model (4.1). $u \leq n \times t$ and $tr(P_\eta Q_{VV}) = (n+u) - (t+u)$ is the conclusion of the Gauss-Markov theorem.

Compared with LS, the degree of freedom for TLS does not change, that is, TLS and LS have the same degrees of freedom.

4.3 Co-factor matrix of parameters

Giving the error description of parameters is a basic object of adjustment. The TLS accuracy assessment problem is difficult to resolve in TLS adjustment theory. For decades, many scholars have proposed different methods to do so. However, these methods vary in terms of degrees of approximation, such that real statistical information of parameters cannot be obtained. CTLS allows TLS accuracy assessment through the theory of classical LS accuracy assessment.
After the adjustment based on model (4.6), the design matrix is formed by the initial value of design matrix parameters since it no longer has random properties, and accuracy assessment can continue based on the principle of error propagation The co-factor of parameters may be taken as an example below:

$$\boldsymbol{Q}_{\Delta\hat{\eta}'\Delta\hat{\eta}'} = (\boldsymbol{A}_{\eta}^{T}\boldsymbol{P}_{\eta}\boldsymbol{A}_{\eta})^{-1}\boldsymbol{A}_{\eta}^{T}\boldsymbol{P}_{\eta}\boldsymbol{Q}_{\eta}\boldsymbol{P}_{\eta}\boldsymbol{B}(\boldsymbol{B}^{T}\boldsymbol{P}_{\eta}\boldsymbol{B})^{-1} = (\boldsymbol{A}_{\eta}^{T}\boldsymbol{P}_{\eta}\boldsymbol{A}_{\eta})^{-1} \tag{4.8}$$

Solving TLS problems by CTLS does not only solve the TLS accuracy assessment problem, which limit the expanded use of TLS, but also achieves integration of the TLS theory with the classical LS approach.

Appendix Vectorization of the matrix product equation

$$\boldsymbol{Vec}(\boldsymbol{ABC}) = (\boldsymbol{C}^{T} \otimes \boldsymbol{A})\boldsymbol{Vec}(\boldsymbol{B})$$

$$\boldsymbol{ABX} = \boldsymbol{Vec}(\boldsymbol{ABX}) = \boldsymbol{Vec}(\boldsymbol{X}^{T}\boldsymbol{B}^{T}\boldsymbol{A}^{T})$$

$$\boldsymbol{ABX} = (\boldsymbol{X}^{T} \otimes \boldsymbol{A})\boldsymbol{Vec}(\boldsymbol{B})$$

If $\underset{n\cdot n}{\boldsymbol{A}} = \boldsymbol{I}$, $\underset{n\cdot t}{\boldsymbol{B}}$, then

$$\underset{n\cdot n}{\boldsymbol{I}}\,\underset{n\cdot t}{\boldsymbol{B}}\boldsymbol{X} = (\underset{t\cdot 1}{\boldsymbol{X}^{T}} \otimes \boldsymbol{I})\boldsymbol{Vec}(\boldsymbol{B})$$

And

$$\underset{n\cdot t}{\boldsymbol{B}}\,\underset{t\cdot 1}{\boldsymbol{X}} = \underset{n\otimes np}{(\boldsymbol{X}^{T} \otimes \boldsymbol{I})}\,\underset{np\cdot 1}{\boldsymbol{Vec}(\boldsymbol{B})}$$

Chapter 5

Application of the coordinate transformation in Baden-Württenberg with different Estimation methods

5.1 Transformation in two-dimensional(2D)

Before we transform the coordinate, we need to centralize the 6-parameter affine transformation model in order to vanish the translation parameters.

$$\begin{aligned}\begin{bmatrix}\boldsymbol{N}\\\boldsymbol{E}\end{bmatrix} &= \begin{bmatrix}\lambda_H cos\alpha & -\lambda_R sin\beta\\\lambda_H sin\alpha & \lambda_R cos\beta\end{bmatrix}\begin{bmatrix}\boldsymbol{H}\\\boldsymbol{R}\end{bmatrix} + \begin{bmatrix}t_N\\t_E\end{bmatrix}\\ &=: \begin{bmatrix}\xi_{11} & \xi_{21}\\\xi_{12} & \xi_{22}\end{bmatrix}\begin{bmatrix}\boldsymbol{H}\\\boldsymbol{R}\end{bmatrix} + \begin{bmatrix}\xi_{31}\\\xi_{32}\end{bmatrix} = \begin{bmatrix}\boldsymbol{H} & \boldsymbol{R} & 0 & 0 & 1 & 0\\0 & 0 & \boldsymbol{H} & \boldsymbol{R} & 0 & 1\end{bmatrix}\begin{bmatrix}\xi_{11}\\\xi_{21}\\\xi_{12}\\\xi_{22}\\\xi_{31}\\\xi_{32}\end{bmatrix}\end{aligned} \tag{5.1}$$

Because the element '1' and '0' have no error, the translation parameters shall disappear by centering this equation. Thus, after the centering the coordinates in the mid point, the translation parameters t_N and t_E will be automatically vanished. Then the observation and old coordinates are centered on their average values in the form:

$$\begin{bmatrix}\underline{\boldsymbol{N}}\\\underline{\boldsymbol{E}}\end{bmatrix} =: \begin{bmatrix}\xi_{11} & \xi_{21}\\\xi_{12} & \xi_{22}\end{bmatrix}\begin{bmatrix}\underline{\boldsymbol{H}}\\\underline{\boldsymbol{R}}\end{bmatrix} \tag{5.2}$$

with

$$\underline{\boldsymbol{N}} = \boldsymbol{N} - mean(\boldsymbol{N}), \underline{\boldsymbol{E}} = \boldsymbol{E} - mean(\boldsymbol{E})$$
$$\underline{\boldsymbol{H}} = \boldsymbol{H} - mean(\boldsymbol{H}), \underline{\boldsymbol{R}} = \boldsymbol{R} - mean(\boldsymbol{R})$$

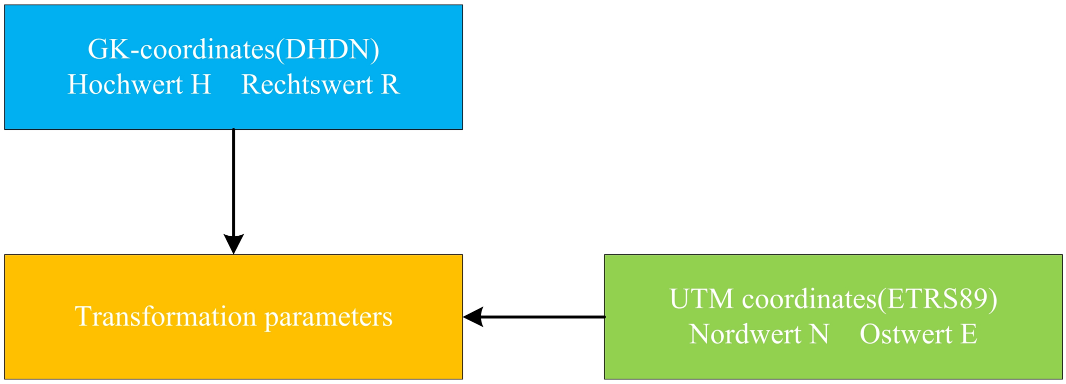


Figure 5.1: *Illustrate the process for 6p-affine transformation*

5.1.1 Transformation with Total Least Squares(SVD)

For the n couple of coordinates we have the transformation model, which is suited for the application of TLS solution.

$$\boldsymbol{E}\left\{\begin{bmatrix}\underline{N}_1\\ \vdots\\ \underline{N}_n\\ \underline{E}_1\\ \vdots\\ \underline{E}_n\end{bmatrix}\right\} = \boldsymbol{E}\left\{\begin{bmatrix}\underline{H}_1 & \underline{R}_1 & 0 & 0\\ \vdots & \vdots & \vdots & \vdots\\ \underline{H}_n & \underline{R}_n & 0 & 0\\ 0 & 0 & \underline{H}_1 & \underline{R}_1\\ \vdots & \vdots & \vdots & \vdots\\ 0 & 0 & \underline{H}_n & \underline{R}_n\end{bmatrix}\right\}\begin{bmatrix}\xi_{11}\\ \xi_{21}\\ \xi_{12}\\ \xi_{22}\end{bmatrix}$$

$$\boldsymbol{y} - \boldsymbol{e} = (\boldsymbol{A} - \boldsymbol{E_A})\boldsymbol{\xi} \tag{5.3}$$

$$\boldsymbol{e^T e} + \boldsymbol{E_A^T E_A} = \boldsymbol{min}(\boldsymbol{e}, \boldsymbol{E_A}, \boldsymbol{\xi}) \tag{5.4}$$

Solution of the TLS problem by using the singular value decomposition(SVD).

$$\boldsymbol{\xi_{TLS}} = (\boldsymbol{A^T A} - \boldsymbol{\sigma_{m+1}^2 I})^{-1}\boldsymbol{A^T y} \tag{5.5}$$

with σ_{m+1} the smallest singular value of the augmented design matrix $[\boldsymbol{A}; \boldsymbol{y}]$:

$$[\boldsymbol{A}; \boldsymbol{y}] = \boldsymbol{U\Sigma V^T} = \sum_{i=0}^{m+1} \boldsymbol{\sigma_i u_i v_i^T}, \quad \boldsymbol{\sigma_1} \geq \cdots \geq \boldsymbol{\sigma_{m+1}} \geq \boldsymbol{0}$$

The best TLS approximation $[\hat{\boldsymbol{A}}; \hat{\boldsymbol{y}}]$ of $[\boldsymbol{A}; \boldsymbol{y}]$ is give by

$$[\hat{\boldsymbol{A}}; \hat{\boldsymbol{y}}] = \boldsymbol{U\Sigma V^T}, \boldsymbol{with} \quad \hat{\boldsymbol{\Sigma}} = \boldsymbol{diag}(\boldsymbol{\sigma_1}, \cdots, \boldsymbol{\sigma_m}, \boldsymbol{0})$$

and with corresponding TLS correction matrix

$$[\hat{\boldsymbol{E}}_{\boldsymbol{A}}; \hat{\boldsymbol{e}}] = [\boldsymbol{A}; \boldsymbol{y}] - [\hat{\boldsymbol{A}}; \hat{\boldsymbol{y}}] = \boldsymbol{\sigma_{m+1} u_{m+1} v_{m+1}^T}$$

With MATLAB function [U,S,V]=svd(X) these procedures can be implemented easily.
We can see from the design matrix $\boldsymbol{A}$ that, the elements $\underline{\boldsymbol{H}}_i$ and $\underline{\boldsymbol{R}}_i$ appeared twice, which means

their corrections have been calculated twice. Additionally, the element 0 also have correction when we perform SVD method. 0 is a non-stochastic element and theoretically has no correction. This means that SVD has a theoretical weakness in that it can not be applied directly when only part of the design matrix contains errors.

5.1.2 Transformation with Total Least Squares based on Partial-EIV model

For the n couple of coordinates we have the same transformation model, which is suited for the application of TLS solution.

$$\boldsymbol{E}\left\{\begin{bmatrix} \underline{N}_1 \\ \vdots \\ \underline{N}_n \\ \underline{E}_1 \\ \vdots \\ \underline{E}_n \end{bmatrix}\right\} = \boldsymbol{E}\left\{\begin{bmatrix} \underline{H}_1 & \underline{R}_1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \underline{H}_n & \underline{R}_n & 0 & 0 \\ 0 & 0 & \underline{H}_1 & \underline{R}_1 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \underline{H}_n & \underline{R}_n \end{bmatrix}\right\} \begin{bmatrix} \xi_{11} \\ \xi_{21} \\ \xi_{12} \\ \xi_{22} \end{bmatrix}$$

Reform the EIV observation model from

$$\boldsymbol{y} - \boldsymbol{e_y} = (\boldsymbol{A} - \boldsymbol{E_A})\boldsymbol{\xi}$$

to

$$\boldsymbol{y} - \boldsymbol{e_y} = (\boldsymbol{\xi}^T \otimes \boldsymbol{I_m})\,[\boldsymbol{h} + \boldsymbol{B}(\boldsymbol{a} - \boldsymbol{e_a})] \tag{5.6}$$

Create the vector a, h and the deterministic matrix $\boldsymbol{B}$

$$\underset{(2n\times 1)}{\boldsymbol{a}} = \begin{bmatrix} \underline{H}_1 \\ \vdots \\ \underline{H}_n \\ \underline{R}_1 \\ \vdots \\ \underline{R}_n \end{bmatrix} \qquad \boldsymbol{h} = \boldsymbol{0}$$

$$\underset{(8n\times 2n)}{\boldsymbol{B}} = \begin{bmatrix} I_n & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & I_n \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \\ I_n & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & I_n \end{bmatrix} \quad with \quad I_n = \left.\begin{bmatrix} 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix}\right\} = n, \quad \boldsymbol{0} = \left.\begin{bmatrix} 0 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 0 \end{bmatrix}\right\} = n$$

The iterative process will implemented with the following steps.

1) The initial values of parameters $\boldsymbol{\xi}$ can be taken from the LS solution.

$$\boldsymbol{\xi}_{(1)} = (\boldsymbol{A}^T\boldsymbol{P_y}\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{P_y}\boldsymbol{y}$$

2) Get the correspond cofactor matrix of y

$$Q_{c(i)} = Q_y + (\xi_{(i)}^T \otimes I_m) B Q_a B^T (\xi_{(i)} \otimes I_m)$$

3) Calculate the value differences $\delta\hat{\xi}$ and get the new value ξ

$$\begin{aligned} \delta\hat{\xi}_{(i+1)} &= (A_{(i)}^T Q_{c(i)}^{-1} A_{(i)})^{-1} A_{(i)}^T Q_{c(i)}^{-1} (y - A\xi_{(i)}) \\ \hat{\xi}_{(i+1)} &= \delta\hat{\xi}_{(i+1)} + \xi_{(i)} \end{aligned}$$

4) Calculate the correction of y and a

$$\begin{aligned} \hat{e}_{y(i+1)} &= Q_y Q_{c(i)}^{-1} (y - A\xi_{(i)} - A_{(i)} \delta\hat{\xi}_{(i+1)}) \\ \hat{e}_{a(i+1)} &= Q_a B^T (\xi_{(i)} \otimes I_m) Q_{c(i)}^{-1} (y - A\xi_{(i)} - A_{(i)} \delta\hat{\xi}_{(i+1)}) \end{aligned}$$

5) Repeat steps 2)-4), until $\left\| \delta\xi_{(\hat{i+1})} \right\| < \varepsilon$ for a given $\varepsilon > 0$. The detail calculations are written as MATLAB code, which can be found in the Appendix.

5.1.3 Transformation with Converted Total Least Squares

In Converted Total Least Squares for the n couple of coordinates with the same transformation model, which is suited for the application of TLS solution.

$$E\left\{\begin{bmatrix} \underline{N}_1 \\ \vdots \\ \underline{N}_n \\ \underline{E}_1 \\ \vdots \\ \underline{E}_n \end{bmatrix}\right\} = E\left\{\begin{bmatrix} \underline{H}_1 & \underline{R}_1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \underline{H}_n & \underline{R}_n & 0 & 0 \\ 0 & 0 & \underline{H}_1 & \underline{R}_1 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \underline{H}_n & \underline{R}_n \end{bmatrix}\right\} \begin{bmatrix} \xi_{11} \\ \xi_{21} \\ \xi_{12} \\ \xi_{22} \end{bmatrix}$$

Reform the EIV observation model from

$$y - e_y = (A - E_A)\xi$$

to

$$\eta = A_\eta \Delta\eta + e_\eta \tag{5.7}$$

Where $\eta = \begin{bmatrix} y - A_\xi^0 \xi^0 \\ a - y_a \end{bmatrix}$, $A_\eta = \begin{bmatrix} A_\xi^0 & B \\ \mathbf{0} & E \end{bmatrix}$, $\Delta\eta = \begin{bmatrix} \Delta\xi \\ \Delta a \end{bmatrix}$, $e_\eta = \begin{bmatrix} e_y \\ e_a \end{bmatrix}$

$$\begin{aligned} e_y &= (A_\xi^0 + E_A)(\xi^0 + \Delta\xi) - y \\ &= A_\xi^0 \Delta\xi + E_A \xi^0 + A_\xi^0 \xi^0 - y + \Delta A \Delta\xi \qquad \rightarrow E_A \Delta\xi \approx \mathbf{0} \\ &= A_\xi^0 \Delta\xi + B\Delta a + A_\xi^0 \xi^0 - y \\ e_a &= a - y_a \end{aligned}$$

The key step here is converting $\boldsymbol{E_A}\boldsymbol{\xi^0}$ to $\boldsymbol{B\Delta a}$.
Create the correspond matrixs

$$\underset{(2n\times4)}{\boldsymbol{E_A}} = \begin{bmatrix} \Delta\underline{H}_1 & \Delta\underline{R}_1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \Delta\underline{H}_n & \Delta\underline{R}_n & 0 & 0 \\ 0 & 0 & \Delta\underline{H}_1 & \Delta\underline{R}_1 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \Delta\underline{H}_n & \Delta\underline{R}_n \end{bmatrix}, \quad \underset{(4\times1)}{\boldsymbol{\xi^0}} = \begin{bmatrix} \xi_{11}^0 \\ \xi_{21}^0 \\ \xi_{12}^0 \\ \xi_{22}^0 \end{bmatrix}, \quad \underset{(2n\times1)}{\boldsymbol{\Delta a}} = \begin{bmatrix} \Delta\underline{H}_1 \\ \vdots \\ \Delta\underline{H}_n \\ \Delta\underline{R}_1 \\ \vdots \\ \Delta\underline{R}_n \end{bmatrix}$$

$$\boldsymbol{E_A}\boldsymbol{\xi^0} = \boldsymbol{B\Delta a} = (\begin{bmatrix} \xi_{11}^0 & \xi_{21}^0 \\ \xi_{12}^0 & \xi_{22}^0 \end{bmatrix} \otimes \boldsymbol{I_n})\boldsymbol{\Delta a}$$

$$\underset{(2n\times2n)}{\boldsymbol{B}} = \begin{bmatrix} \xi_{11}^0 & \xi_{21}^0 \\ \xi_{12}^0 & \xi_{22}^0 \end{bmatrix} \otimes \boldsymbol{I_n} = \begin{bmatrix} \xi_{11}^0 & 0 & 0 & \xi_{21}^0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & \xi_{11}^0 & 0 & 0 & \xi_{21}^0 \\ \xi_{12}^0 & 0 & 0 & \xi_{22}^0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & \xi_{12}^0 & 0 & 0 & \xi_{22}^0 \end{bmatrix}$$

The solution of CTLS is

$$\boldsymbol{\Delta\hat{\eta}} = (\boldsymbol{A_\eta^T P_\eta A_\eta})^{-1}\boldsymbol{A_\eta^T P_\eta \eta}$$

The solution $\boldsymbol{\Delta\hat{\eta}}$ is a $(2n+4)\times 1$ vector. The first 4 elements of $\boldsymbol{\Delta\eta}$ are the corrections of $\boldsymbol{\xi}$ and the following $2n$ elements are the corrections of $\boldsymbol{y_a}$, which are the corrections for the initial design matrix $\boldsymbol{A}$. The final transformation parameters are $\hat{\boldsymbol{\xi}} = \boldsymbol{\Delta\xi} + \boldsymbol{\xi_0}$, with $\boldsymbol{\xi_0}$ calculated from the LS solution.
The detail calculations are written as MATLAB code, which can be found in the Appendix.

5.1.4 Presentation and Comparison of the results

Statistical data by the quadratics sums of the residuals for 4 estimators. The $\boldsymbol{\hat{e}}$ is the residuals of observation and $\boldsymbol{\hat{E}}$ is the residuals of design matrix.
LS:

$$\boldsymbol{\hat{e}}_{LS}^T\boldsymbol{\hat{e}}_{LS} = 3.678308 \quad (m^2)$$

TLS:

$$\boldsymbol{\hat{e}}_{TLS}^T\boldsymbol{\hat{e}}_{TLS} = 0.409136 \quad (m^2)$$
$$\boldsymbol{\hat{E}}_{TLS}^T\boldsymbol{\hat{E}}_{TLS} = 0.817619 \quad (m^2)$$
$$\boldsymbol{\hat{e}}_{TLS}^T\boldsymbol{\hat{e}}_{TLS} + \boldsymbol{\hat{E}}_{TLS}^T\boldsymbol{\hat{E}}_{TLS} = 1.226756 \quad (m^2)$$

TLSP:

$$\hat{\boldsymbol{e}}_{TLSP}^T \hat{\boldsymbol{e}}_{TLSP} = 0.920311 \quad (m^2)$$
$$\hat{\boldsymbol{E}}_{TLSP}^T \hat{\boldsymbol{E}}_{TLSP} = 0.919577 \quad (m^2)$$
$$\hat{\boldsymbol{e}}_{TLSP}^T \hat{\boldsymbol{e}}_{TLSP} + \hat{\boldsymbol{E}}_{TLSP}^T \hat{\boldsymbol{E}}_{TLSP} = 1.839889 \quad (m^2)$$

CTLS:

$$\hat{\boldsymbol{e}}_{CTLS}^T \hat{\boldsymbol{e}}_{CTLS} = 0.920311 \quad (m^2)$$
$$\hat{\boldsymbol{E}}_{CTLS}^T \hat{\boldsymbol{E}}_{CTLS} = 0.919577 \quad (m^2)$$
$$\hat{\boldsymbol{e}}_{CTLS}^T \hat{\boldsymbol{e}}_{CTLS} + \hat{\boldsymbol{E}}_{CTLS}^T \hat{\boldsymbol{E}}_{CTLS} = 1.839889 \quad (m^2)$$

Table 5.1: *Comparison of 6-parameter affine transformation parameters with 4 estimators*

Transformation models	6-parameter affine transformation GK(DHDN)-UTM(ETRS89) $t_N(m)$	$t_E(m)$	$\alpha('')$	$\beta('')$	$d\lambda_H(\times 10^{-4})$	$d\lambda_R(\times 10^{-4})$
LS	437.194567	119.756709	0.165368	-0.196455	-3.996797	-3.988430
TLS	437.194554	119.756712	0.165368	-0.196455	-3.996797	-3.988430
Partial-EIV	437.194556	119.756709	0.165375	-0.196445	-3.996797	-3.988430
CTLS	437.194556	119.756709	0.165375	-0.196445	-3.996797	-3.988430

Table 5.2: *Numerical deviation of 6-parameter affine transformation with 4 estimators*

Transformation model	Collocated sites	Absolute mean of Residuals (m) $[V_N]$	$[V_E]$	Max.absolute mean of Residuals(m) $[V_N]$	$[V_E]$	RMS (m)	Standard deviation of unit weight (m)
LS	B-W 131	0.1049	0.0804	0.3288	0.3226	0.1187	0.1199
TLS	B-W 131	0.0350	0.0268	0.1097	0.1076	0.0396	0.0400
Partial-EIV	B-W 131	0.0525	0.0402	0.1645	0.1614	0.0594	0.0848
CTLS	B-W 131	0.0525	0.0402	0.1645	0.1614	0.0594	0.0848

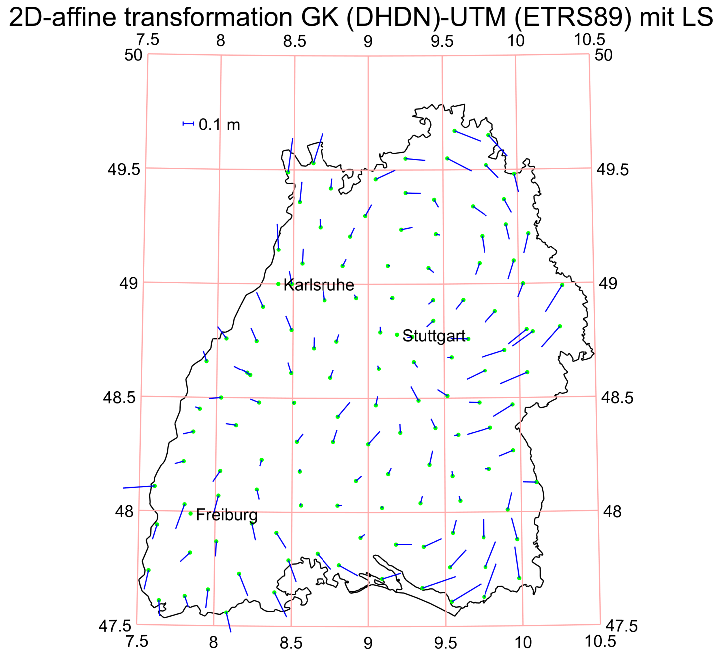


Figure 5.2: *Horizontal residuals after 6-parameter affine transformation in Baden-Württemberg network*

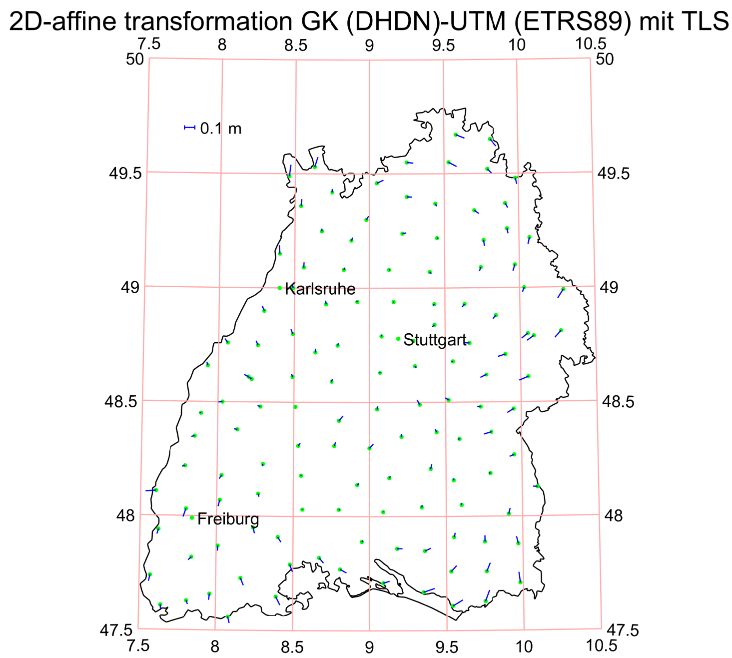


Figure 5.3: *Horizontal residuals after 6-parameter affine transformation in Baden-Württemberg network*

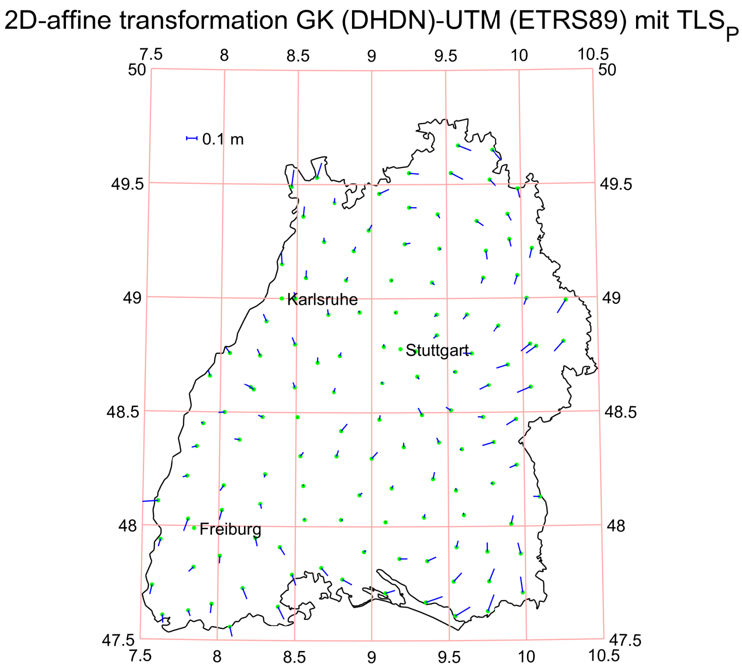


Figure 5.4: Horizontal residuals after 6-parameter affine transformation in Baden-Württemberg network

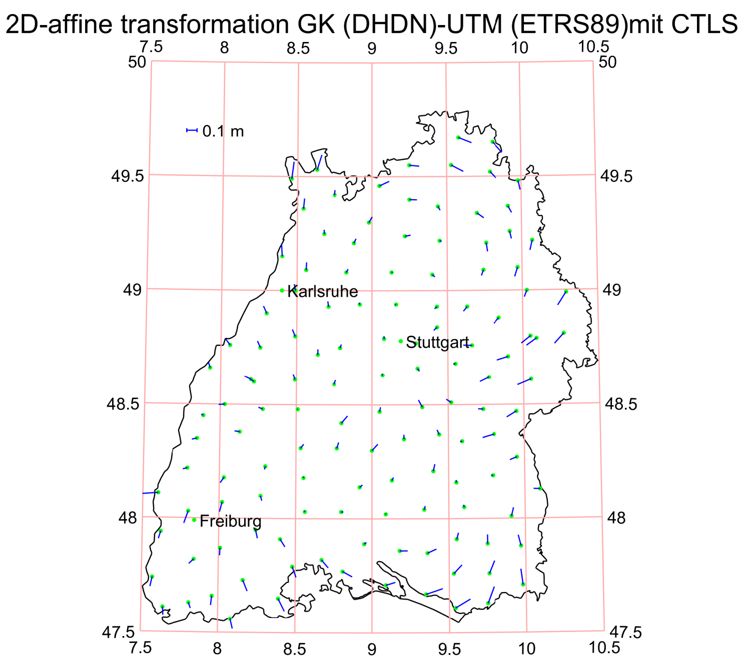


Figure 5.5: Horizontal residuals after 6-parameter affine transformation in Baden-Württemberg network

From the results above, we can see that the SVD, Partial-EIV and CTLS all have better estimations than the LS, which is obvious and reasonable. It should be pointed out that, it seems that SVD has a better results than Partial-EIV and CTLS. Which is not correct, because the SVD is a method taking the whole elements in design matrix into consideration. Under the same weight conditions, the deviation is systematically distributed to every element in the design matrix, even those non-stochastic elements included. As a consequence, the residuals calculated with SVD are not that dispersion. For the Partial-EIV and CTLS, they have almost the same results in every part of estimation data. That means CTLS can calculate the 2 dimensional transformation parameters without iteration and has the same level of accuracy with Partial-EIV.

5.2 Transformation in three-dimensional(3D)

The following formula has been used for the estimation of the parameters in seven-parameter Helmert transformation.

$$\begin{bmatrix} \boldsymbol{X}_G \\ \boldsymbol{Y}_G \\ \boldsymbol{Z}_G \end{bmatrix} = (1+d\lambda) \begin{bmatrix} 1 & \gamma & -\beta \\ -\gamma & 1 & \alpha \\ \beta & -\alpha & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_L \\ \boldsymbol{Y}_L \\ \boldsymbol{Z}_L \end{bmatrix} + \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$
$$\boldsymbol{X}_G = \lambda \begin{bmatrix} 1 & \gamma & -\beta \\ -\gamma & 1 & \alpha \\ \beta & -\alpha & 1 \end{bmatrix} \boldsymbol{X}_L + \boldsymbol{T}_L \tag{5.8}$$

Where λ is scale factor, α, β, γ are rotation angles. The translation terms T_X, T_Y, T_Z are the coordinates of the origin of the 3-D network.
After the linearization, the formula is rewritten:

$$\begin{bmatrix} \boldsymbol{X}_G \\ \boldsymbol{Y}_G \\ \boldsymbol{Z}_G \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & -\boldsymbol{Z}_L & \boldsymbol{Y}_L & \boldsymbol{X}_L \\ 0 & 1 & 0 & \boldsymbol{Z}_L & 0 & -\boldsymbol{X}_L & \boldsymbol{Y}_L \\ 0 & 0 & 1 & -\boldsymbol{Y}_L & \boldsymbol{X}_L & 0 & \boldsymbol{Z}_L \end{bmatrix} \begin{bmatrix} T_X \\ T_Y \\ T_Z \\ \delta\alpha \\ \delta\beta \\ \delta\gamma \\ \lambda \end{bmatrix} \tag{5.9}$$

After centering the coordinates in the midpoints, the translation parameter T_X, T_Y, T_Z will disappear, and then the observations and old coordinates are centered on their average values. This will be assumed in the following:

$$\begin{bmatrix} \boldsymbol{x}_g \\ \boldsymbol{y}_g \\ \boldsymbol{z}_g \end{bmatrix} = \begin{bmatrix} 0 & -\boldsymbol{z}_l & \boldsymbol{y}_l & \boldsymbol{x}_l \\ \boldsymbol{z}_l & 0 & -\boldsymbol{x}_l & \boldsymbol{y}_l \\ -\boldsymbol{y}_l & \boldsymbol{x}_l & 0 & \boldsymbol{z}_l \end{bmatrix} \begin{bmatrix} \delta\alpha \\ \delta\beta \\ \delta\gamma \\ \lambda \end{bmatrix} \tag{5.10}$$

with

$$\begin{bmatrix} \boldsymbol{x}_g \\ \boldsymbol{y}_g \\ \boldsymbol{z}_g \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_G \\ \boldsymbol{Y}_G \\ \boldsymbol{Z}_G \end{bmatrix} - mean \begin{bmatrix} \boldsymbol{X}_G \\ \boldsymbol{Y}_G \\ \boldsymbol{Z}_G \end{bmatrix}, \quad \begin{bmatrix} \boldsymbol{x}_l \\ \boldsymbol{y}_l \\ \boldsymbol{z}_l \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_L \\ \boldsymbol{Y}_L \\ \boldsymbol{Z}_L \end{bmatrix} - mean \begin{bmatrix} \boldsymbol{X}_L \\ \boldsymbol{Y}_L \\ \boldsymbol{Z}_L \end{bmatrix} \tag{5.11}$$

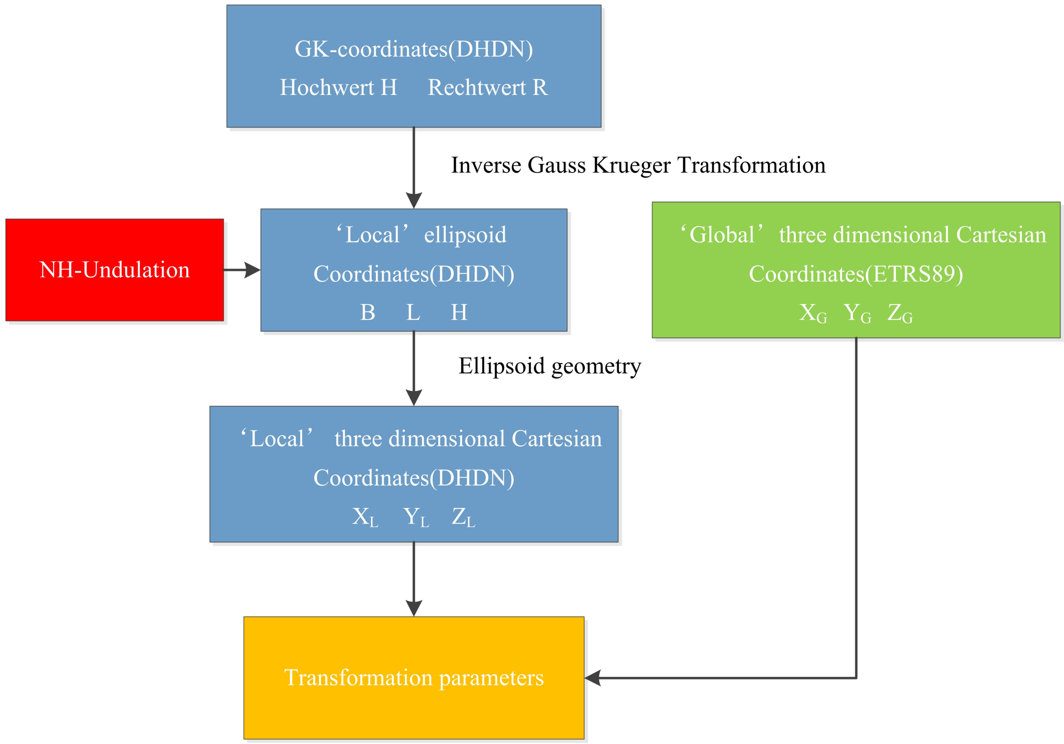


Figure 5.6: *Illustrate the process for 7p-Helmert transformation*

5.2.1 Transformation with Total Least Squares(SVD)

In 3D coordinate transformation we have the similar solution with 2D in the SVD method.

$$e\left\{\begin{bmatrix} x_{g1} \\ \vdots \\ x_{gn} \\ y_{g1} \\ \vdots \\ y_{gn} \\ z_{g1} \\ \vdots \\ z_{gn} \end{bmatrix}\right\} =: E\left\{\begin{bmatrix} 0 & -z_{l1} & y_{l1} & x_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & -z_{ln} & y_{ln} & x_{ln} \\ z_{l1} & 0 & -x_{l1} & y_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ z_{ln} & 0 & -x_{ln} & y_{ln} \\ -y_{l1} & x_{l1} & 0 & z_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ -y_{ln} & x_{ln} & 0 & z_{ln} \end{bmatrix}\right\} \begin{bmatrix} \delta\alpha \\ \delta\beta \\ \delta\gamma \\ \lambda \end{bmatrix}$$

$$y - e = (A - E_A)\xi \tag{5.12}$$

$$e^T e + E_A^T E_A = min(e, E_A, \xi) \tag{5.13}$$

Solution of the TLS problem by using the singular value decomposition(SVD).

$$\xi_{TLS} = (A^T A - \sigma_{m+1}^2 I)^{-1} A^T y \tag{5.14}$$

with σ_{m+1} the smallest singular value of the augmented design matrix $[A; y]$:

$$[\boldsymbol{A}; \boldsymbol{y}] = \boldsymbol{U\Sigma V^T} = \sum_{i=0}^{m+1} \sigma_i u_i v_i^T, \quad \sigma_1 \geq \cdots \geq \sigma_{m+1} \geq 0$$

The best TLS approximation $[\hat{A}; \hat{y}]$ of $[A; y]$ is give by

$$[\hat{A}; \hat{y} = U\Sigma V^T, with \quad \hat{\Sigma} = diag(\sigma_1, \cdots, \sigma_m, 0)$$

and with corresponding TLS correction matrix

$$[\hat{E}_A; \hat{e}] = [A; y] - [\hat{A}; \hat{y}] = \sigma_{m+1} u_{m+1} v_{m+1}^T$$

5.2.2 Transformation with Total Least Squares based on Partial-EIV model

For the n couple of coordinates we have the transformation model, which is suited for the application of TLS solution.

$$\boldsymbol{E}\left\{\begin{bmatrix} x_{g1} \\ \vdots \\ x_{gn} \\ y_{g1} \\ \vdots \\ y_{gn} \\ z_{g1} \\ \vdots \\ z_{gn} \end{bmatrix}\right\} =: \boldsymbol{E}\left\{\begin{bmatrix} 0 & -z_{l1} & y_{l1} & x_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & -z_{ln} & y_{ln} & x_{ln} \\ z_{l1} & 0 & -x_{l1} & y_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ z_{ln} & 0 & -x_{ln} & y_{ln} \\ -y_{l1} & x_{l1} & 0 & z_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ -y_{ln} & x_{ln} & 0 & z_{ln} \end{bmatrix}\right\} \begin{bmatrix} \delta\alpha \\ \delta\beta \\ \delta\gamma \\ \lambda \end{bmatrix}$$

Reform the EIV observation model from

$$\boldsymbol{y} - \boldsymbol{e_y} = (\boldsymbol{A} - \boldsymbol{E_A})\boldsymbol{\xi}$$

to

$$\boldsymbol{y} - \boldsymbol{e_y} = (\boldsymbol{\xi^T} \otimes \boldsymbol{I_m})\,[\boldsymbol{h} + \boldsymbol{B}(\boldsymbol{a} - \boldsymbol{e_a})] \tag{5.15}$$

Create the vector a, h and the deterministic matrix B

$$\underset{(3n\times 1)}{\boldsymbol{a}} = \begin{bmatrix} x_{l1} \\ \vdots \\ x_{ln} \\ y_{l1} \\ \vdots \\ y_{ln} \\ z_{l1} \\ \vdots \\ z_{ln} \end{bmatrix} \qquad \boldsymbol{h} = \boldsymbol{0}$$

$$\underset{(12n\times 3n)}{\boldsymbol{B}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_n \\ \mathbf{0} & -I_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -I_n \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ I_n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & I_n & \mathbf{0} \\ -I_n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ I_n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & I_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_n \end{bmatrix} \quad with \quad I_n = \left.\begin{bmatrix} 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix}\right\} = n, \quad \mathbf{0} = \left.\begin{bmatrix} 0 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 0 \end{bmatrix}\right\} = n$$

The iterative process will implemented with the following steps.

1) The initial values of parameters $\boldsymbol{\xi}$ can be taken from the LS solution.

$$\boldsymbol{\xi}_{(1)} = (\boldsymbol{A}^T \boldsymbol{P}_y \boldsymbol{A})^{-1} \boldsymbol{A}^T \boldsymbol{P}_y \boldsymbol{y}$$

2) Get the correspond cofactor matrix of y

$$\boldsymbol{Q}_{c(i)} = \boldsymbol{Q}_y + (\boldsymbol{\xi}_{(i)}^T \otimes \boldsymbol{I}_m) \boldsymbol{B} \boldsymbol{Q}_a \boldsymbol{B}^T (\boldsymbol{\xi}_{(i)} \otimes \boldsymbol{I}_m)$$

3) Calculate the value differences $\delta\hat{\xi}$ and get the new value ξ

$$\begin{aligned} \delta\hat{\boldsymbol{\xi}}_{(i+1)} &= (\boldsymbol{A}_{(i)}^T \boldsymbol{Q}_{c(i)}^{-1} \boldsymbol{A}_{(i)})^{-1} \boldsymbol{A}_{(i)}^T \boldsymbol{Q}_{c(i)}^{-1} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}_{(i)}) \\ \hat{\boldsymbol{\xi}}_{(i+1)} &= \delta\hat{\boldsymbol{\xi}}_{(i+1)} + \boldsymbol{\xi}_{(i)} \end{aligned}$$

4) Calculate the correction of $\boldsymbol{y}$ and $\boldsymbol{a}$

$$\begin{aligned} \hat{\boldsymbol{e}}_{y(i+1)} &= \boldsymbol{Q}_y \boldsymbol{Q}_{c(i)}^{-1} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}_{(i)} - \boldsymbol{A}_{(i)} \delta\hat{\boldsymbol{\xi}}_{(i+1)}) \\ \hat{\boldsymbol{e}}_{a(i+1)} &= \boldsymbol{Q}_a \boldsymbol{B}^T (\boldsymbol{\xi}_{(i)} \otimes \boldsymbol{I}_m) \boldsymbol{Q}_{c(i)}^{-1} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\xi}_{(i)} - \boldsymbol{A}_{(i)} \delta\hat{\boldsymbol{\xi}}_{(i+1)}) \end{aligned}$$

5) Repeat steps 2)-4), until $\left\| \delta\boldsymbol{\xi}_{(\hat{i+1})} \right\| < \varepsilon$ for a given $\varepsilon > 0$. The detail calculations are written as MATLAB code, which can be found in the Appendix.

5.2.3 Transformation with Converted Total Least Squares

In Converted Total Least Squares for the n couple of coordinates with the same transformation model, which is suited for the application of TLS solution.

$$\boldsymbol{E}\left\{\begin{bmatrix} x_{g1} \\ \vdots \\ x_{gn} \\ y_{g1} \\ \vdots \\ y_{gn} \\ z_{g1} \\ \vdots \\ z_{gn} \end{bmatrix}\right\} =: \boldsymbol{E}\left\{\begin{bmatrix} 0 & -z_{l1} & y_{l1} & x_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & -z_{ln} & y_{ln} & x_{ln} \\ z_{l1} & 0 & -x_{l1} & y_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ z_{ln} & 0 & -x_{ln} & y_{ln} \\ -y_{l1} & x_{l1} & 0 & z_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ -y_{ln} & x_{ln} & 0 & z_{ln} \end{bmatrix}\right\} \begin{bmatrix} \delta\alpha \\ \delta\beta \\ \delta\gamma \\ \lambda \end{bmatrix}$$

Reform the EIV observation model from

$$\boldsymbol{y} - \boldsymbol{e_y} = (\boldsymbol{A} - \boldsymbol{E_A})\boldsymbol{\xi}$$

to

$$\boldsymbol{\eta} = \boldsymbol{A_\eta}\boldsymbol{\Delta\eta} + \boldsymbol{e_\eta} \tag{5.16}$$

Where $\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{y} - \boldsymbol{A^0_\xi}\boldsymbol{\xi^0} \\ \boldsymbol{a} - \boldsymbol{y_a} \end{bmatrix}, \boldsymbol{A_\eta} = \begin{bmatrix} \boldsymbol{A^0_\xi} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{E} \end{bmatrix}, \boldsymbol{\Delta\eta} = \begin{bmatrix} \boldsymbol{\Delta\xi} \\ \boldsymbol{\Delta a} \end{bmatrix}, \boldsymbol{e_\eta} = \begin{bmatrix} \boldsymbol{e_y} \\ \boldsymbol{e_a} \end{bmatrix}$

$$\begin{aligned} \boldsymbol{e_y} &= (\boldsymbol{A^0_\xi} + \boldsymbol{E_A})(\boldsymbol{\xi^0} + \boldsymbol{\Delta\xi}) - \boldsymbol{y} \\ &= \boldsymbol{A^0_\xi}\boldsymbol{\Delta\xi} + \boldsymbol{E_A}\boldsymbol{\xi^0} + \boldsymbol{A^0_\xi}\boldsymbol{\xi^0} - \boldsymbol{y} + \boldsymbol{\Delta A}\boldsymbol{\Delta\xi} \qquad \rightarrow \boldsymbol{E_A}\boldsymbol{\Delta\xi} \approx \boldsymbol{0} \\ &= \boldsymbol{A^0_\xi}\boldsymbol{\Delta\xi} + \boldsymbol{B}\boldsymbol{\Delta a} + \boldsymbol{A^0_\xi}\boldsymbol{\xi^0} - \boldsymbol{y} \\ \boldsymbol{e_a} &= \boldsymbol{a} - \boldsymbol{y_a} \end{aligned}$$

The key step here is converting $\boldsymbol{E_A}\boldsymbol{\xi^0}$ to $\boldsymbol{B}\boldsymbol{\Delta a}$.
Create the correspond matrixs

$$\underset{(3n\times 4)}{\boldsymbol{E_A}} = \begin{bmatrix} 0 & -\Delta z_{l1} & \Delta y_{l1} & \Delta x_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & -\Delta z_{ln} & \Delta y_{ln} & \Delta x_{ln} \\ \Delta z_{l1} & 0 & -\Delta x_{l1} & \Delta y_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ \Delta z_{ln} & 0 & -\Delta x_{ln} & \Delta y_{ln} \\ -\Delta y_{l1} & \Delta x_{l1} & 0 & \Delta z_{l1} \\ \vdots & \vdots & \vdots & \vdots \\ -\Delta y_{ln} & \Delta x_{ln} & 0 & \Delta z_{ln} \end{bmatrix}, \quad \underset{(4\times 1)}{\boldsymbol{\xi^0}} = \begin{bmatrix} \xi^0_{11} \\ \xi^0_{21} \\ \xi^0_{12} \\ \xi^0_{22} \end{bmatrix}, \quad \underset{(3n\times 1)}{\boldsymbol{\Delta a}} = \begin{bmatrix} \Delta x_{l1} \\ \vdots \\ \Delta x_{ln} \\ \Delta y_{l1} \\ \vdots \\ \Delta y_{ln} \\ \Delta z_{l1} \\ \vdots \\ \Delta z_{ln} \end{bmatrix}$$

$$\boldsymbol{E_A}\boldsymbol{\xi^0} = \boldsymbol{B}\boldsymbol{\Delta a} = (\begin{bmatrix} \xi_{22}^0 & \xi_{12}^0 & -\xi_{21}^0 \\ -\xi_{12}^0 & \xi_{22}^0 & \xi_{11}^0 \\ \xi_{21}^0 & -\xi_{11}^0 & \xi_{22}^0 \end{bmatrix} \otimes \boldsymbol{I_n})\boldsymbol{\Delta a}$$

$$\underset{(3n\times 3n)}{\boldsymbol{B}} = \begin{bmatrix} \xi_{22}^0 & \xi_{12}^0 & -\xi_{21}^0 \\ -\xi_{12}^0 & \xi_{22}^0 & \xi_{11}^0 \\ \xi_{21}^0 & -\xi_{11}^0 & \xi_{22}^0 \end{bmatrix} \otimes \boldsymbol{I_n}$$

$$= \begin{bmatrix} \xi_{22}^0 & 0 & 0 & \xi_{12}^0 & 0 & 0 & -\xi_{21}^0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & \xi_{22}^0 & 0 & 0 & \xi_{12}^0 & 0 & 0 & -\xi_{21}^0 \\ -\xi_{12}^0 & 0 & 0 & \xi_{22}^0 & 0 & 0 & \xi_{11}^0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & -\xi_{12}^0 & 0 & 0 & \xi_{22}^0 & 0 & 0 & \xi_{11}^0 \\ \xi_{21}^0 & 0 & 0 & -\xi_{11}^0 & 0 & 0 & \xi_{22}^0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & \xi_{21}^0 & 0 & 0 & -\xi_{11}^0 & 0 & 0 & \xi_{22}^0 \end{bmatrix}$$

The solution of CTLS is

$$\boldsymbol{\Delta\hat{\eta}} = (\boldsymbol{A_\eta^T P_\eta A_\eta})^{-1}\boldsymbol{A_\eta^T P_\eta \eta}$$

The solution $\boldsymbol{\Delta\hat{\eta}}$ is a $(3n+4)\times 1$ vector. The first 4 elements of $\boldsymbol{\Delta\eta}$ are the corrections of $\boldsymbol{\xi}$ and the following $3n$ elements are the corrections of $\boldsymbol{y_a}$, which are the corrections for the initial design matrix $\boldsymbol{A}$. The final transformation parameters are $\boldsymbol{\hat{\xi}} = \boldsymbol{\Delta\xi} + \boldsymbol{\xi_0}$, with $\boldsymbol{\xi_0}$ calculated from the LS solution.
The detail calculations are written as MATLAB code, which can be found in the Appendix.

5.2.4 Presentation and Comparison of the results

Statistical data by the quadratics sums of the residuals for 4 estimators. The $\hat{e}$ is the residual of observation and $\hat{E}$ is the residual of design matrix.
LS:

$$\boldsymbol{\hat{e}}_{LS}^T\boldsymbol{\hat{e}}_{LS} = 4.063234 \quad (m^2)$$

TLS:

$$\begin{aligned} \boldsymbol{\hat{e}}_{TLS}^T\boldsymbol{\hat{e}}_{TLS} &= 1.015790 \quad (m^2) \\ \boldsymbol{\hat{E}}_{TLS}^T\boldsymbol{\hat{E}}_{TLS} &= 1.015808 \quad (m^2) \\ \boldsymbol{\hat{e}}_{TLS}^T\boldsymbol{\hat{e}}_{TLS} + \boldsymbol{\hat{E}}_{TLS}^T\boldsymbol{\hat{E}}_{TLS} &= 2.031598 \quad (m^2) \end{aligned}$$

TLSP:

$$\hat{\boldsymbol{e}}_{TLSP}^T\hat{\boldsymbol{e}}_{TLSP} = 1.015790 \quad (m^2)$$
$$\hat{\boldsymbol{E}}_{TLSP}^T\hat{\boldsymbol{E}}_{TLSP} = 1.015808 \quad (m^2)$$
$$\hat{\boldsymbol{e}}_{TLSP}^T\hat{\boldsymbol{e}}_{TLSP} + \hat{\boldsymbol{E}}_{TLSP}^T\hat{\boldsymbol{E}}_{TLSP} = 2.031598 \quad (m^2)$$

CTLS:

$$\hat{\boldsymbol{e}}_{CTLS}^T\hat{\boldsymbol{e}}_{CTLS} = 1.015790 \quad (m^2)$$
$$\hat{\boldsymbol{E}}_{CTLS}^T\hat{\boldsymbol{E}}_{CTLS} = 1.015808 \quad (m^2)$$
$$\hat{\boldsymbol{e}}_{CTLS}^T\hat{\boldsymbol{e}}_{CTLS} + \hat{\boldsymbol{E}}_{CTLS}^T\hat{\boldsymbol{E}}_{CTLS} = 2.031598 \quad (m^2)$$

Table 5.3: *Comparison of 7-parameter Helmert transformation parameters with 4 estimators*

Transformation models	7-parameter Helmert transformation GK(DHDN)-UTM(ETRS89)						
	$T_X(m)$	$T_Y(m)$	$T_Z(m)$	$\alpha('')$	$\beta('')$	$\gamma('')$	$d\lambda(\times 10^{-6})$
LS	582.901711	112.168080	405.603061	-2.255032	-0.335003	2.068369	9.117208
TLS	582.901702	112.168078	405.603051	-2.255032	-0.335003	2.068369	9.117210
Partial-EIV	582.901701	112.168078	405.603051	-2.255032	-0.335003	2.068369	9.117210
CTLS	582.901711	112.168080	405.603061	-2.255032	-0.335003	2.068369	9.117208

Table 5.4: *Numerical deviation of 7-parameter Helmert transformation with 4 estimators*

Transformation model	Collocated sites	Absolute mean of Residuals (m)		Max.absolute mean of Residuals(m)		RMS (m)	Standard deviation of unit weight (m)
		$[V_N]$	$[V_E]$	$[V_N]$	$[V_E]$		
LS	B-W 131	0.1051	0.0843	0.4212	0.3112	0.1240	0.1026
TLS	B-W 131	0.0526	0.0421	0.2106	0.1556	0.0620	0.0513
Partial-EIV	B-W 131	0.0526	0.0421	0.2106	0.1556	0.0620	0.0513
CTLS	B-W 131	0.0526	0.0421	0.2106	0.1556	0.0620	0.0513

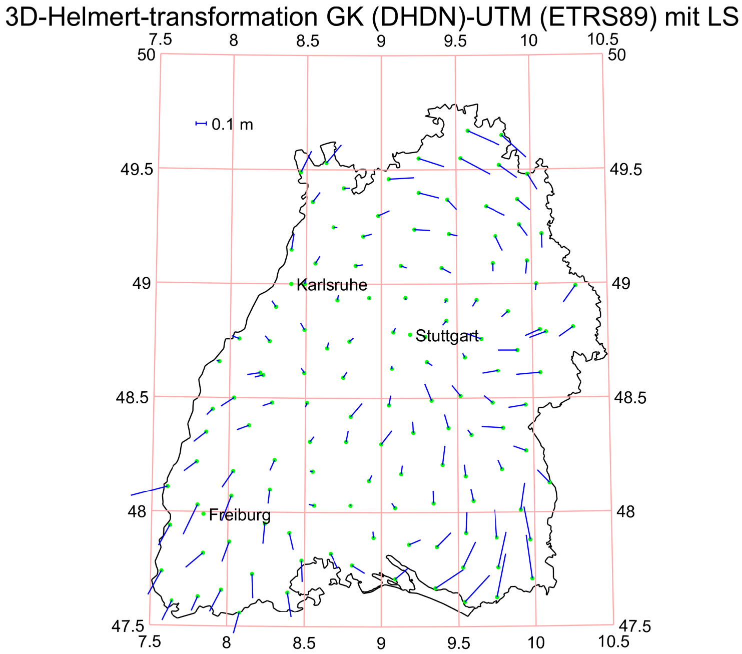


Figure 5.7: *Horizontal residuals after 7-parameter Helmert transformation in Baden-Württemberg network*

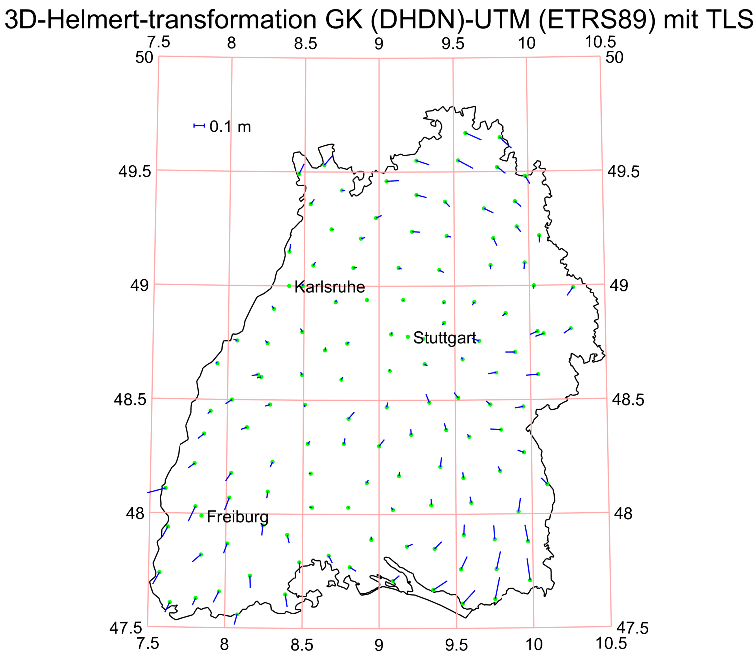


Figure 5.8: *Horizontal residuals after 7-parameter Helmert transformation in Baden-Württemberg network*

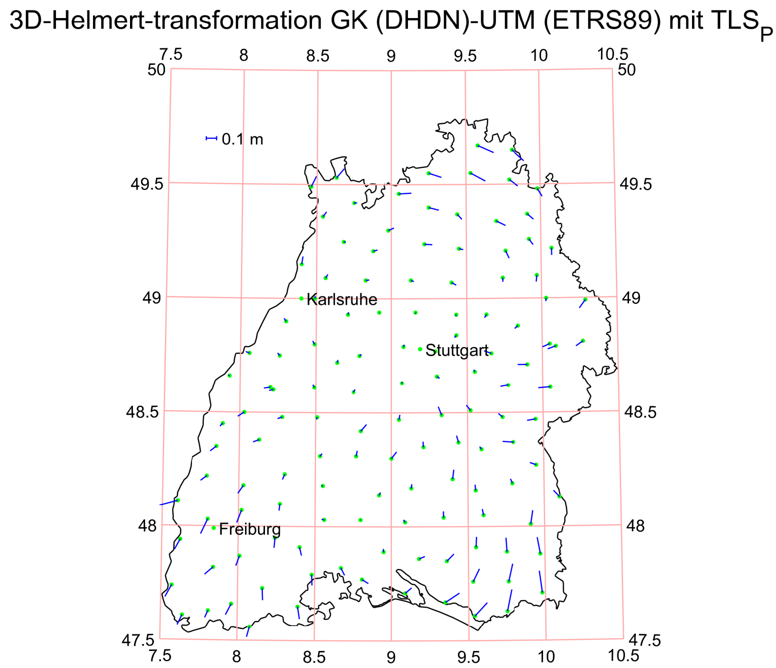


Figure 5.9: *Horizontal residuals after 7-parameter Helmert transformation in Baden-Württemberg network*

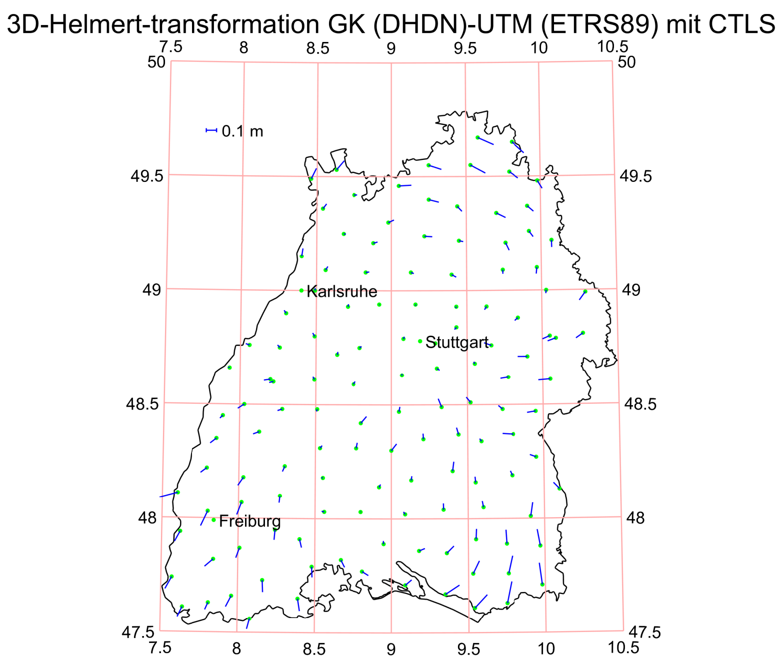


Figure 5.10: *Horizontal residuals after 7-parameter Helmert transformation in Baden-Württemberg network*

From the results above, we can see that, the same with 2D coordinate transformation, the SVD ,Partial-EIV and CTLS all have better estimations than the LS. However,it is important to note here, that the quadratics sums of the residuals of SVD, Partial-EIV and CTLS are the same with each other. As a matter of fact the differences of them exist after 7 decimal places, which is no meaning to discuss about it. So why they have almost the same residuals.
After study into the transformation models, it is found that, unlike the design matrix in affine transformation model, the elements by Helmert transformation in design matrix are calculated 3 times. However, the weights and influences of these elements are different with that in affine transformation. To be more specific we can see from the base transformation models and part of the residuals of the design matrix.

$$E\left\{\begin{bmatrix}\underline{N}_1\\ \vdots\\ \underline{N}_n\\ \underline{E}_1\\ \vdots\\ \underline{E}_n\end{bmatrix}\right\} = E\left\{\begin{bmatrix}\underline{H}_1 & \underline{R}_1 & 0 & 0\\ \vdots & \vdots & \vdots & \vdots\\ \underline{H}_n & \underline{R}_n & 0 & 0\\ 0 & 0 & \underline{H}_1 & \underline{R}_1\\ \vdots & \vdots & \vdots & \vdots\\ 0 & 0 & \underline{H}_n & \underline{R}_n\end{bmatrix}\right\}\begin{bmatrix}\xi_{11}\\ \xi_{21}\\ \xi_{12}\\ \xi_{22}\end{bmatrix}$$

$$E\left\{\begin{bmatrix}x_{g1}\\ \vdots\\ x_{gn}\\ y_{g1}\\ \vdots\\ y_{gn}\\ z_{g1}\\ \vdots\\ z_{gn}\end{bmatrix}\right\} =: E\left\{\begin{bmatrix}0 & -z_{l1} & y_{l1} & x_{l1}\\ \vdots & \vdots & \vdots & \vdots\\ 0 & -z_{ln} & y_{ln} & x_{ln}\\ z_{l1} & 0 & -x_{l1} & y_{l1}\\ \vdots & \vdots & \vdots & \vdots\\ z_{ln} & 0 & -x_{ln} & y_{ln}\\ -y_{l1} & x_{l1} & 0 & z_{l1}\\ \vdots & \vdots & \vdots & \vdots\\ -y_{ln} & x_{ln} & 0 & z_{ln}\end{bmatrix}\right\}\begin{bmatrix}\delta\alpha\\ \delta\beta\\ \delta\gamma\\ \lambda\end{bmatrix}$$

In the affine transformation, the transformation parameters after centering are ξ_{11}, ξ_{12}, ξ_{21} and ξ_{22}. The correspond data are $\lambda_H cos\alpha$,$-\lambda_R sin\beta$,$\lambda_H sin\alpha$ and $\lambda_R cos\beta$, in which the rotation parameters α and β are really small. So $\lambda_H cos\alpha$ and $\lambda_R cos\beta$ (ξ_{11},ξ_{22})have much bigger influences on the design matrix. And after the calculate by SVD method, the absolute value of the first and fourth column of the residuals by design matrix are much bigger than second and third column.
In the Helmert transformation, the transformation parameters after centering are $\delta\alpha$,$\delta\beta$,$\delta\gamma$ and λ. The rotation parameters are very small too and the scale has relative bigger influence. So after the calculate by SVD method, the absolute value of the fourth column of the residuals by design matrix are much bigger than others.

Table 5.5: *First 10 rows of the residuals by design matrix with 2D affine transformation*

0.033378	3.179106×10^{-8}	2.6760356×10^{-8}	0.033378
0.068186	6.494339×10^{-8}	5.466658×10^{-8}	0.068186
-0.096281	-9.170245×10^{-8}	-7.719122×10^{-8}	-0.096281
0.005882	5.601823×10^{-9}	4.715376×10^{-9}	0.005882
0.040717	3.878038×10^{-8}	3.264367×10^{-8}	0.040717
0.041302	3.933744×10^{-8}	3.311258×10^{-8}	0.041302
-0.109623	-1.044095×10^{-7}	-8.788746×10^{-8}	-0.109623
-0.031937	-3.041849×10^{-8}	-2.560499×10^{-8}	-0.031937
-0.029446	-2.804580×10^{-8}	-2.360776×10^{-8}	-0.029446
0.001235	1.175897×10^{-9}	9.898197×10^{-10}	0.001235

Table 5.6: *First 10 rows of the residuals by design matrix with 3D Helmert transformation*

2.261038×10^{-7}	3.358953×10^{-8}	-2.073878×10^{-7}	-0.020682
5.867438×10^{-7}	8.716551×10^{-8}	-5.381755×10^{-8}	-0.053669
-9.169869×10^{-7}	-1.362258×10^{-7}	8.410823×10^{-7}	0.083876
1.253344×10^{-7}	1.861943×10^{-8}	-1.149597×10^{-7}	-0.011464
4.069860×10^{-7}	6.046104×10^{-8}	-3.732973×10^{-7}	-0.037227
3.329597×10^{-7}	4.946384×10^{-7}	-3.053987×10^{-7}	-0.030456
-8.796520×10^{-7}	-1.306794×10^{-7}	8.068378×10^{-7}	0.080461
-1.221220×10^{-7}	-1.814221×10^{-8}	1.120133×10^{-7}	0.011170
-2.906775×10^{-7}	-4.318247×10^{-8}	2.666163×10^{-7}	0.026588
3.301255×10^{-7}	4.904279×10^{-9}	-3.027990×10^{-8}	-0.003020

From the table we can see that, in the 3D Helmert transformation, there is only one column in the residuals of design matrix has effective data. Which means in this case, although the coordinates in design matrix have been calculated 3 times, 2 of them have very small residuals. Only one, the fourth column has effectively calculated. As a result, the final quadratics sums of the residuals of SVD are almost the same with Partial-EIV and CTLS. But it can not prove that, SVD is as good as Partial-EIV and CTLS. Because it still has some systematical problems, like it has the residuals for 0. We could see that, SVD might be acceptable in 3D Helmert transformation.

Chapter 6

Conclusion

The tradition techniques used for solving the linear estimation problems are based on classical LS. However, only the errors of observation vector are considered, and the design matrix is assumed to be accurate without any errors. Which makes LS not valid for most cases. Further study based on Errors-in-Variables(EIV), Total Least Squares method considers the errors in design matrix as well. The problem of which is, the repetition of parameters in design matrix has a deviation influence on the minimum norm constraint. Reform from EIV-model to Partial-EIV model and the Converted Total Least Squares could solve the Problem. Compared with Partial-EIV model, the solution of Converted Total Least Squares does not need the iteration.

For the concrete 'Introduction of ETRS89 into Baden-Württemberg' has an alternative transformation procedure with the different estimation methods been applied in the transformation with the models of the 7-parameter Helmert transformation and 6-parameter affine transformation using the 131 collocated points, and the results have been tested and discussed here. Based on these analyses and comparisons with different estimation methods the following points can be concluded (Cai, 2006):

- The traditional SVD method of TLS has a theoretical weakness in that it can not be applied directly when only part of the design matrix contains errors.
- The Converted Total Least Squares may be used to to deal with stochastic design matrix in LS solutions. In this approach, the TLS problem can be transformed into a LS problem, and the non-linear problem can be transformed into a linear problem.
- The Converted Total Least Squares can be easily used to consider the weight of observations and stochastic design matrix.
- Although the estimated transformation parameters of Partial-EIV model and Converted Total Least Squares are almost identical, the CTLS has its advantage without complicated iteration processing.
- This thesis completes the theory of TLS accuracy assessment, gives statistical information of parameters under the TLS method, improves the TLS algorithm, and solves the bottleneck restricting the application of TLS.

Generally, it is hoped that this thesis can help the reader to have a better understanding about the TLS theory and the application.

Bibliography

Acar, M., Oezluedemir, M. T., Akzilmaz, O., Celik, R. N. and Ayan, T.(2006): Deformation analysis with Total Least Squares, Istanbul Technical University, Division of Geodesy, Istanbul, Turkey.

Adcock, R. J. (1878): A problem in least squares. The Analyst, 5, 53-54.

Akyilmaz O (2007): Total Least-Squares Solution of Coordinate Transformation. Survey Review.39(303):68-80

Badekas, J. (1969): Investigations related to the establishment of a world geodetic system, Report 124, Department of Geodetic Science, Ohio State University, Columbus.

Bursa, M. (1962): The theory of the determination of the non parallelism of the minor axis of the reference ellipsoid, polar axis of the Earth, and initial astronomical and geodetic meridians from observations of artificial Earth satellites. Studia Geophysica et Geodaetica 6, 209-214.

Cai, J. (2000): The systematic analysis of the transformation between the German Geodetic Reference System (DHDN, DHHN) and the ETRF System (DREF91). Earth Planets Space 52, 947-952.

Cai, J. (2009): Systematical nalysis of the transformation between Gauss-Krüger-Coordinate/ DHDN and UTM-Coordinate/ETRS89 in Baden-Württemberg with different estimation methods. IAG Symposium VOLUME 134, Hermann Drewes(ED), Geodetic Reference Frame 2009.

Cai, J. (2003): Gutachten zur 'Einführung von ETRS89 in Baden-Württemberg'. Geodätisches Institut der Universität Stuttgart, Stuttgart 2003.

Ding K., Ou J. and Zhao C. (2007): Methods of the least-suqares orthogonal distance fitting. Science of Surveying and Maping.32(3)17-19(in Chinese)

Gleser, L. J. (1981): Estimation in a multivariate 'errors in variables' regression model: Large sample results. Ann. Statist. 9, 24-44.

Golub, G. H. (1973): Some modified matrix eigenvalue problems. SIAM Rev. 15, 318-344.

Golub, G. H. and Van Loan, C. F. (1980): An Analysis of the total least squares problem. Siam J. Humer Anal. Vol 17, No.6., December 1980.

Grafarend, E. (2006): Linear and Nonlinear Models: Fixed Effects, Random Effects, and Mixed Models. de Gruyter, Berlin; New York.

Guo, R. (2007): Systematical analysis of the transformation procedures in Baden-Württemberg with Least Squares and Total Least Squares methods. Geodätisches Institut der Universität Stuttgart, Stuttgart 2007.

Kong, J., Yao Y., and Wu, H (2010): Iterative Method For Total Least-Squares. Geomatics and information science of wuhan university, 35(6):711-714(in Chinese)

Koopmans, T. C. (1937): Linear Regression Analysis of Economic Time Series. De Erven F. Bohn, Haarlem, The Netherlands.

Levin, M. J. (1964): Estimation of a system pulse transfer function in the presence of noise. IEEE Trans. Automat. Control 9, 229-235

Madansky, A. (1959): The fitting of straight lines when both variables are subject to error. J. Amer. Statist. Assoc. 54, 173-205.

Micheal, K., Mathias, A. (2007) A weighted total least-squares algorithm for fitting a straight line. Meas. Sci Technol.No.18:3438-3442

Molodensky, M., Eremeev, S. and Yurkina, V. F. (1960): Metody izucheniya vnesnego gravitatsion nogopolya I figure Zemli. Tr. CNIIGAIK, Moscow, 131.

Mühlich, M. and Mester, R. (1999): Subspace methods and equilibration in computer vision. Submitted to Citeseer 2000. Institute for Applied Physics, J. W. Goethe-Universität Frankfurt, Frankfurt, Germany.

Pearson, K. (1901): On lines and planes of closest fit to points in space, Philos. Mag., 2, pp. 559-572.

Qiu W, Tao B, YaoY, Wu Y, Huang H (2008): The Theory and Method of Surveying Data Processing. Wuhan: Wuhan University Press.(in Chiness)

Schaffrin, B. and Felus, Y. (2003): On Total Least-Squares adjustment with constraints, In Proceedings to IUGG General Ass. Department of Civil and Environmental Engineering and geodetic Science, The Ohio State Universtity, Columbus.

Schaffrin, B. and Felus, Y. (2008):Multivariate total least squares adjustment for empirical affine transformations. In: Xu, P., Liu, J: Sermanis A(eds) International association of geodesy symposia: VI Hotine-Marussi symposium on the theoretical and computational Geodesy, 29 May-2 June 2006, Wuhan,China, vol 132. Springer, Berlin, pp 238-242

Schaffrin, B. (2003): A Note on Constrained Total Least-Squares Estimation, In Computaional Statistics and Data Analysis, eingereicht Okt 2003.

Schaffin, B. (2005): Generalizing the Total Least-Squares Estimation for Empirical Coordinate Transformation, lecture in Geodetic Inst. Stuttgart.

Strang, G. (1988): Linear Algebra and its Applications, 3rd edition, Harcourt Brace Jovanovich, San Diego, pp. 505.

Van Huffel, S. and Zha, H. (1993): The Total Least-Squares problem, in Handbook of Statistics: Computational Statistics, (Rao, C. R., ed.), vol. 9, North Holland Publishing Company, Amsterdam, The Netherlands, 1993, pp. 377-408.

Van Huffel, S. and Lemmerling, P. (2002): Total Least Squares and Errors-in-Variables Modeling, Kluwer Academic Publishers, Netherlands.

Van Huffel, S. and Vandewalle, J. (1991): The Total Least Squares Problem Computational Aspects and Analysis, Society for Industrial and Applied Mathematics, Philadelphia.

Veis, G. (1960): Geodetic uses of artifical satellites, Smithsonian Contributions to Astrophysics 3, 95-161.

Wang, B., Jiancheng, Li. and Liu, Chao (2015):A robust weighted total least squares algorithm and its geodetic applications.

Wei, M. (1992): Algebraic relations between the total least-squares and least square problems with more than one solution. Numer. Math. 62:123-148(in Chinese)

Wolf, H. (1963): Geometric connection and re-orientation of three-dimensional triangulation nets, Bulletin Geodesique, 68, 165-169.

Wolf, R. P. and Ghilani, D. C. (1997): Adjustment computations: statistics and least squares in surveying and GIS, Wiley, New York, 347.

York, D. (1966): Least squares fitting of a straight line. Canad. J. Phys. 44, 1079-1086.

Yu, J (1996): On the Solvability of the Total Least-Squares Problem. Journal of Nanjing Normal University (Nature Science).19(1)13-16(in Chinese)

Zhang Q., Bao Z. and Jiao L. (1994): A Neural Network Approach to Total Least-Squanres Problem. Journal of China Institute of Communications.15(4): 79-85(in Chinese)