Estimation of an Unknown Cartographic Projection and its Parameters from the Map

Faculty of Sciences, Charles University in Prague , Tel.: ++420-221-951-400

Abstract

This article presents a new off-line method for the detection, analysis and estimation of an unknown cartographic projection and its parameters from a map. Several invariants are used to construct the objective function ϕ that describes the relationship between the 0D, 1D, and 2D entities on the analyzed and reference maps. It is minimized using the Nelder-Mead downhill simplex algorithm. A simplified and computationally cheaper version of the objective function ϕ involving only 0D elements is also presented. The following parameters are estimated: a map projection type, a map projection aspect given by the meta pole K coordinates $[\varphi_k, \lambda_k]$, a true parallel latitude φ_0, central meridian longitude λ_0, a map scale, and a map rotation. Before the analysis, incorrectly drawn elements on the map can be detected and removed using the IRLS. Also introduced is a new method for computing the L_2 distance between the turning functions Θ_1, Θ_2 of the corresponding faces using dynamic programming. Our approach may be used to improve early map georeferencing; it can also be utilized in studies of national cartographic heritage or land use applications. The results are presented both for real cartographic data, representing early maps from the David Rumsay Map Collection, and for the synthetic tests.

Keywords: digital cartography, map projection, analysis, simplex method, optimization, Voronoi diagram, outliers detection, early maps, georeferencing, cartographic heritage, meta data, Marc 21, MapAnalyst.

1 Introduction

The detection and estimation of unknown cartographic projection parameters from a map represents a problem belonging to the category of cartometric analysis. From both cartographic and algorithmic points of view this issue can be considered remarkable; it combines methods from several existing areas, particularly robust statistics, computational geometry, and mathematical cartography. Such an analysis is beneficial and interesting for historic, old, or current maps without information about the map's projection; it could improve georeferencing. This information is also useful for further studies of the national cartographic heritage, land use, or land cover applications.

The appropriate cataloging of maps creates the need for information about the map's projection and geographic extent, which form a part of the cartographic meta data. The bibliographic format Marc 21 contains a detailed description of a map projection in fields 034 (Coded Cartographic Mathematical Data) and 255B (Cartographic Mathematical Data) of the bibliographic record; analogous records are also included in the INSPIRE standard. As there was no exact way to determine a map projection, so far, this feature has been estimated visually, or the record has been left blank. Using the proposed solution, this step can be performed semi-automatically and with a higher degree of relevance using our method.

Some applications can be found in the area of natural gas and oil prospecting, where in less developed countries geological maps in paper form are used.

The vast majority of early maps were not constructed on a solid geometric or geodesic basis. Therefore, it is impossible to consider the existence of a map projection in such cases. Let us ask if that sort of analysis makes sense. These maps have unsuitable geometric properties (locally and nearly randomly changing length, area, and angular distortions not depending directly on the geographic location), making the process of estimating the cartographic parameters more difficult and ambiguous, and causing the results to be somewhat unreliable. In the author's opinion, even in such cases it is more suitable to try to estimate a projection, especially if approximate georeferencing is required. This claim applies in particular to small-scale maps (world maps, maps of hemispheres or continents) created since the 17th century, which were analyzed in the paper.

The current approach, based on 2D transformation, does not estimate the additional parameters of the map projection (φ_k, λ_k, φ_0) and does not remove map elements highly influenced by errors negatively affecting the results. Neither the transverse nor the oblique aspects of the projection are supported, and 1D-2D elements are not involved in the analysis.

This paper brings a new method for the detection and estimation of an unknown map's projection in all aspects (normal, transverse, oblique), eliminating these drawbacks. It is based on the minimization of the objective function ϕ using the downhill simplex method. It is a proper method for solving small dimensional problems. The objective function ϕ, which is discussed in Sec. 5.1.1, is based on a combination of different types of invariants. However, a simplified version of the objective function ϕ, computationally cheaper, is also presented. Meridians and parallels are detected using the RANSAC algorithm, modified so as to provide not only the best solution, but more acceptable solutions. Instead of a single fit meridian or parallel, a set of meridians and parallels will be detected.

Although, the proposed method brings correct results, there is a place for further improvements. The more efficient methods, based on the NLSP approach, available for the on-line analysis, will be presented in the next article.

2 Related Work

Due to the difficulty of estimating unknown projection parameters, especially without deep numeric analysis, this problem has not been studied in detail. However, the development of computer sciences has provided new analytical methods and procedures.

There are software tools focused on georeferencing and analysis of early maps [9], [35], [42], [10], where several transformation models are supported (similarity, affine, spline). So far, they do not allow automated detection of a map's projection. However, they are widely used in various institutions, including the British Library [26], [42], and the New York Public Library [35].

Some simple methods for identifying a projection in an Arc Map software can be found in the Esri Knowledge Base [11], [12]. A built-in Arc Map tool, based on user-entered control points and a simple affine transformation, can also be used. However, these solutions have several limitations and are not exact. The `prjfinder` software[14] represents a more sophisticated tool for Arc Map version 9, which searches for the best matching coordinate system. Unfortunately, its capabilities cannot be tested under version 10.1, where the software did not work.

The detection method, based on 2D transformations, was developed by [23] and the algorithms were implemented in MapAnalyst open-source software (http://mapanalyst.org/). According to author knowledge, this is the only solution that really works.

The proposed method is more robust; it supports the analysis of elements of different spatial dimensions (0D, 1D, 2D), enables the determination of additional parameters of the map projection and supports the normal, transverse, and oblique aspects of the projection.

The detection process consists of different procedures that are frequently used in computer sciences. Let us give a brief overview of similar problems and topics. Given the wide range of the paper, not all relevant articles could be referenced.

RANSAC is the very frequently used robust estimator introduced by [13]. It is applicable to robust regression as M-estimators [47]. In the proposed solution, for fitting meridians and parallels, only the linear regression is used. Algorithms for point patterns matching on the basis of 2D transformations are widely used because of their reliability, efficiency, invariance, and ease of computation. A matching algorithm invariant to translations, rotations, and scale changes using 2D transformation was described by [8]. Another method with uncertainty regions was brought by [36]; a similarity factor was used by [51]. Variable uncertainty regions formed by the Tissot indicatrix, depending both on the geographic position and the map projection properties (e.g., cartographic distortions) represent a novelty, given the current state.

An overview of shape-matching algorithms was brought by [48]; a comparison of their performance and efficiency characteristics was mentioned by [2]. The turning function, a reliable and efficient shape descriptor, was introduced by [1]. Its properties and suitability for different types of element analysis were examined in depth in a series of articles by [29]; [43]; [49]. Statistical methods represent another approach of computing the spatial similarity between two sets of points; the cross nearest neighbor distance [39] measuring the spatial proximity of two normalized data sets seems to be an appropriate similarity descriptor. Measuring similarities between two data sets, using their Voronoi diagrams is not a frequently used method. However, there is a direct approach assessing features of the Voronoi cells. Simple criteria such as area, perimeter [31], number of vertices [7] , and the distribution of cell areas [22]; [4] are considered. Due to the lack of appropriate cells, these criteria are difficult to apply to smaller sets.

There are different strategies for identifying gross errors depending on the number of errors in the data set: M-estimators [21], the Danish method [27], least trimmed squares [45], and many others. An overview of robust methods in heterogeneous linear models was created by [19]. When multiple gross errors exist, an iterative least square method brings accurate results [52]; [18]; [25]. For highly contaminated data, RANSAC is also an efficient robust estimator [20]. Algorithms for Boolean operations on simple polygons have been described in many articles, such as: [32], [30], [33].

The downhill simplex method introduced by [37] is frequently used for non-linear optimization. A detailed description of simplex methods can be found in [50] [24], [41]; a convergence is discussed in [34], [28]. An improved version with adaptive parameters brought [16], a quasi-gradient version [40].

3 Map Projection Parameters

The regular map projection $\mathbb{P}$, $\mathbb{P}: S_1 \rightarrow S_2$, which represents the reference surface S_1 (Earth) or its part on another, typically, flat, surface S_2 (a plane), is given by coordinate functions f, g in the explicit form

$$X = f(\varphi, \lambda), \qquad Y = g(\varphi, \lambda).$$

This formula expresses the relationship of a point P that is transformed between surfaces from geographic coordinates φ, λ to rectangular coordinates x, y. Coordinate functions are continuous with their first and second partial derivatives, and are finite. The correspondence of points on both systems of true and meta pole $K = [\varphi_k, \lambda_k]$ for the oblique aspect express the laws of spherical trigonometry

$$\begin{aligned} \sin\varphi' &= \sin\varphi_k \sin\varphi + \cos\varphi_k \sin\varphi \cos(\lambda - \lambda_k), & (3.1)\\ \sin\lambda' &= \cos\varphi \sin(\lambda - \lambda_k) / \cos\varphi', & (3.2) \end{aligned}$$

where (φ', λ') are geographic coordinates of a point related to the meta pole K. Let us denote e, f, g, h as Gaussian coefficients equal to

$$e = \left(\frac{\partial f}{\partial\varphi}\right)^2 + \left(\frac{\partial g}{\partial\varphi}\right)^2, \qquad f = \frac{\partial f}{\partial\varphi}\frac{\partial f}{\partial\lambda} + \frac{\partial g}{\partial\varphi}\frac{\partial g}{\partial\lambda},$$

$$g = \left(\frac{\partial f}{\partial\lambda}\right)^2 + \left(\frac{\partial g}{\partial\lambda}\right)^2, \qquad h = \frac{\partial f}{\partial\lambda}\frac{\partial g}{\partial\varphi} - \frac{\partial f}{\partial\varphi}\frac{\partial g}{\partial\lambda}.$$

Local linear scale factors m, n in meridian and parallel directions, for a radius of a sphere R, and a given point P, are

$$m^2 = e/R^2, \qquad n^2 = g/(R^2\cos^2\varphi).$$

The angular distortion ω' at a given point P is

$$\omega' = \tan^{-1}(h/f).$$

The ellipse of distortion (i.e., the Tissot indicatrix) $\mathcal{T}$ given by parameters a, b, A expresses the relationship of a local linear scale on the azimuth. An infinitely small circle of the radius r at a given point is projected as an infinitely small ellipse given by semiaxes a, b

$$\begin{aligned} a &= 0.5\,(c + d)\,,\\ b &= 0.5\,(c - d)\,,\\ c &= \sqrt{m^2 + n^2 + 2mn\sin\omega},\\ d &= \sqrt{m^2 + n^2 - 2mn\sin\omega}, \end{aligned}$$

the extreme values of local linear scale α measured from meridian passing P from

$$\alpha = 0.5\tan^{-1}(2mn \cdot \sin\omega/(m^2 - n^2)).$$

The convergence γ_m of a projected meridian or parallel γ_p at a given point P

$$\gamma_m = \tan^{-1}\left(\frac{\partial g}{\partial\varphi}/\frac{\partial f}{\partial\varphi}\right), \qquad \gamma_p = \tan^{-1}\left(\frac{\partial g}{\partial\lambda}/\frac{\partial f}{\partial\lambda}\right) \qquad (3.3)$$

is used to determine the polar angle A of the semi-major axis of the Tissot indicatrix and the y axis (i.e., azimuth), where α' represents the projected extreme value of local linear scale

$$A = \alpha' + \gamma_m, \qquad \alpha' = \tan^{-1}(b/a \cdot \tan\alpha).$$

Parameters a, b, A of the Tissot indicatrix play an important role in the proposed matching algorithm for point features. They enable the construction of uncertainty regions of variable dimensions. We find the percentage ratio μ of point features of the testing set lying within the uncertainty regions drawn on the corresponding point features of the reference set; see Sec. 6.

3.1 Intervals of determined cartographic parameters

To avoid a geometric construct which does not respect the real cartographic parameters for the analyzed area, cartographic principles and rules must be taken into account. Depending on the shape, size, and position of the projected analyzed area, the values of some cartographic parameters will be refined.

Setting new intervals of φ_k, λ_k for an oblique aspect. Let us apply known cartographic assumptions and patterns: both prime meridian and true parallel are passing approximately through the center of an analyzed territory (conic, cylindrical projections) and a meta pole should be placed inside an analyzed territory (azimuthal projections). Other projections will be applied only in the normal aspect. It is also known that cartographic projections used in small-scale maps are commonly defined only in the normal aspect.

For a map projection analyzed in an oblique aspect, both domains apriori given by $\varphi_k \in \langle -\frac{\pi}{2}, \frac{\pi}{2} \rangle$ and $\lambda_k \in \langle -\pi, \pi \rangle$ will be reduced, especially, in the longitudinal direction. New intervals that depend both on the analyzed territory and a map projection can be found in Tab. 1. Extremal values of φ, λ are vertices of the minimum bounding rectangle of Q. This approach brings time improvement, primarily for smaller territories.

Such computed intervals are generally too wide, especially for large territories. Let us make the assumption that a cartographer proposed a map projection, so that

Table 1: Intervals φ_k, λ_k depending on an analyzed territory position and a map projection.

Projection	**Conic, cylindrical**		**Azimuthal**	
Analyzed area	φ_k	λ_k	φ_k	λ_k
$\varphi_{max} \leq 0$	$\left\langle -\frac{\pi}{2}, 0 \right\rangle$	$\left\langle \lambda_{min} - \frac{\pi}{2}, \lambda_{max} + \frac{\pi}{2} \right\rangle$	$\left\langle -\frac{\pi}{2}, 0 \right\rangle$	$\langle \lambda_{min}, \lambda_{max} \rangle$
$\varphi_{min} \geq 0$	$\left\langle 0, \frac{\pi}{2} \right\rangle$	$\left\langle \lambda_{min} - \frac{\pi}{2}, \lambda_{max} + \frac{\pi}{2} \right\rangle$	$\left\langle 0, \frac{\pi}{2} \right\rangle$	$\langle \lambda_{min}, \lambda_{max} \rangle$
$\varphi_{min} \leq 0 \cap \varphi_{max} \geq 0$	$\left\langle 0, \frac{\pi}{2} \right\rangle$	$\left\langle \lambda_{min} - \frac{\pi}{2}, \lambda_{max} + \frac{\pi}{2} \right\rangle$	$\left\langle -\frac{\pi}{2}, \frac{\pi}{2} \right\rangle$	$\langle \lambda_{min}, \lambda_{max} \rangle$

the cartographic distortions on the borders of the analyzed area will be the smallest (or small). Therefore, the optimal meta pole position leads to the minimization of the relative linear distortion measured along all directions and all points. There are many linear distortion criteria given by various formulas (Airy, Airy-Kavrayskiy and Jordan) [46]. We use the complex criterion ε_c [5] of

$$\varepsilon_c = 0.5(|a-1| + |b-1|) + \frac{a}{b} - 1.$$

For a simple projection with straight meridians (cylindrical, conic azimuthal), where $h = a$ and $k = b$, this formula turns into the simpler form of

$$\varepsilon_c = 0.5(|h-1| + |k-1|) + \frac{h}{k} - 1.$$

The criteria of variations within the analyzed area that should be minimized is for the oblique projection aspect given by

$$E = \frac{1}{(\varphi'_{max} - \varphi'_{min})(\lambda'_{max} - \lambda'_{min})} \int_{\varphi'_{min}}^{\varphi'_{max}} \int_{\lambda'_{min}}^{\lambda'_{max}} \varepsilon_c d\varphi' d\lambda'.$$

An exact determination of E is very difficult. Therefore, the analyzed area is divided into k small pieces with midpoints $P_i = [\varphi_i, \lambda_i]$. Then E can be determined as the weighted mean of k pieces

$$E = \frac{\sum_{i=1}^{k} w_i \varepsilon_i}{\sum_{i=1}^{k} w_i},$$

where $w_i = \cos \varphi_i$. We simplify this relationship and compute E values only in two extreme points, $P_{\min} = [\varphi'_{\min}, \lambda'_{\min}]$ and $P_{\max} = [\varphi'_{\max}, \lambda'_{\max}]$ of the analyzed area, so

$$E = \frac{\cos \varphi'_{\min} \varepsilon_c(P_{\min}) + \cos \varphi'_{\max} \varepsilon_c(P_{\max})}{\cos \varphi'_{\min} + \cos' \varphi_{\max}}. \qquad (3.4)$$

For each analyzed P, P' pair, E is computed and compared with the required graphical accuracy G of the map. If $E > 3G$, the proposed projection is inappropriate for the analyzed area, and the objective function is penalized.

Considering the maximum length distortion $m = 0.001\%$ for the large-scale map (1 cm per km for $S = 1000$), $m = 1\%$ for the mid-scale map ($S = 1,000,000$), and analogously $m = 100\%$ for the small-scale map ($S = 100,000,000$), the required graphical accuracy is simply expressed as follows

$$G{=}1.0 \cdot 10^{-6} M.$$

This test can be applied during the heuristic check of determined parameters. Of course, we must assume that the cartographer did not use any improper map projection parameters.

4 Input Parameters of Detection

Taking into account the financial and time constraints, there is no practical need to analyze the entire early map at once. The full vectorization represents a time-consuming process, especially if the map contains rich and extensive details. This may be thousands of features; however, for the analysis, the subset is sufficient. According to the results of tests, 10-20 features are recommended, see Sec. 7.

An essential step of the detection process is represented by finding such geometric characteristics of elements both in the analyzed, and reference maps, to decide which projection has been used. To increase the detection reliability, 1D-2D features or sampled meridians and parallels detected by the RANSAC algorithm (see Sec. 5.4), are involved.

Numerous requirements increasing the analysis efficiency should be put on testing sets. The results of analyses are strongly influenced by the following factors:

Impact of spatial distribution. The uniform distribution of analyzed features on a map plays an important role. The proposed techniques are suitable for sets with approximately the same spatial density of features. However, in many cases where it is impossible to ensure this requirement, the drawing density on the map may be highly variable. This applies especially to early maps. On portolan charts the coastline and ports are drawn, but the interior of the continents is missing. Irregularly spaced clusters of points or wide territories without analyzed points affect the results negatively, see the results in Sec. 8.3.

Impact of the analyzed territory. The size and position of the analyzed territory strongly affect the efficiency

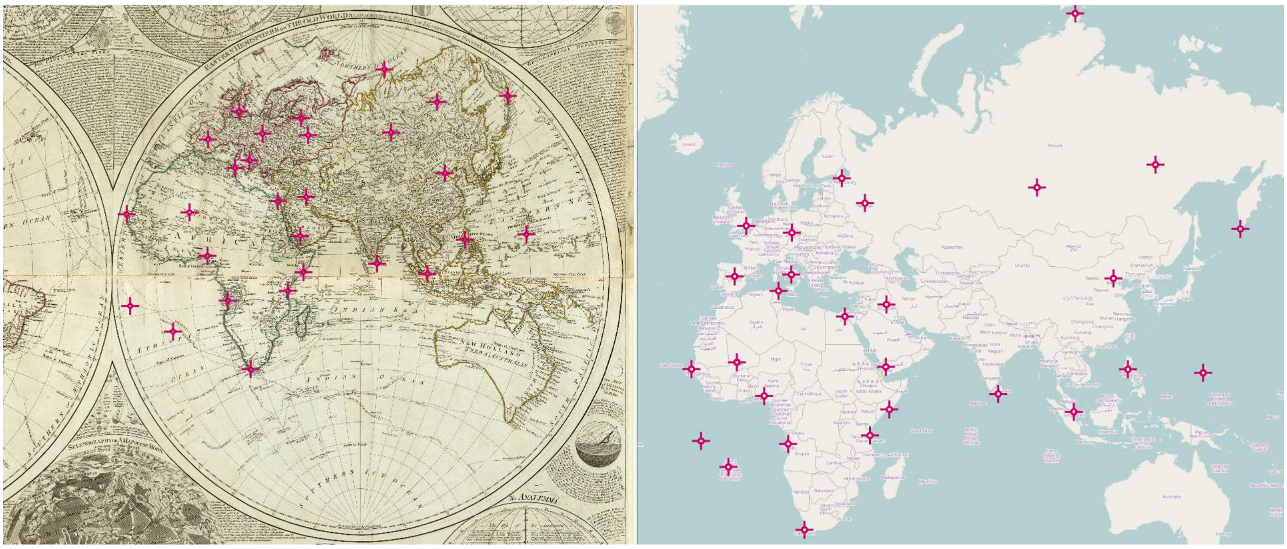

Fig. 4.1: The set of 0D features (identical points) on the analyzed and reference maps acquired in MapAnalyst software.

of the detection algorithm. The analyzed territory should have similar dimensions in the latitudinal and longitudinal directions, and should be large enough (at least $\Delta\varphi = \Delta\lambda = 3°$) to ensure that the positional differences of both sets of features in the map's projection are not less than the graphical accuracy of the map. The locus of an analyzed region near the equator (a similar shape in most cartographic projections) or near the geographic/meta pole (singular points) is also not recommended.

Impact of the map scale. There is a strong dependance on the previous point; the projected territory size is a function of the map scale. Small-scale maps show a large territory in low detail and vice versa, but they are affected by the generalization of the content. However, small-scale map parameters are better to determine.

Impact of map projection. Map projection properties represent a very important factor. Both oblique and transverse aspects are more difficult to detect. The objective function ϕ has a complicated course, with many local minima (see Fig. 5.3). Therefore, the iteration process can become stuck in a local minimum. Some shapes of the graticule are easily recognizable (cylindrical, azimuthal projections in the normal); others have a similar shape (pseudo-azimuthal, pseudo-cylindrical, globular). Because of the various types of projection constant parameters, the exact values of the parameters are difficult to determine, especially for conic projections (The North Pole projected as a point or a part of the circle).

Impact of the map content. Map sheet distortions caused by a paper aging, or organization by map fields, must also be taken into account [17], [44], [3]. Some cartographic techniques have a strong influence on the geometric accuracy of a map; for example, a generalization. These effects should be corrected before analysis. However, some can be removed completely (a paper distortion), partially (outliers), or remains (a generalization effect). Due to the smaller generalization, topographic maps represent a better source for the analysis.

Impact of types of features. The selection of appropriate point features for analysis is relatively complicated. We assume that their position does not change significantly over time (they are stable in their geographical locations), and they are easy to identify on a map (cities, castles, river confluences, churches). In most situations, this set of elements is sufficient for further analysis. If both the analyzed map and the reference maps contain a graticule, an additional analysis of sampled meridian/parallel points is a promising method, improving the results. An implementation of line features (rivers, roads) into the assessment process reduces the discretion and enables additional analysis, which further improves the results. Polygonal features allow the analysis of continuous and extensive parts of maps in a single step; they represent one of the best materials for the assessment process. Features appropriate for analysis are selected by a cartographer manually, in accordance with all recommendations; see Fig. 4.1.

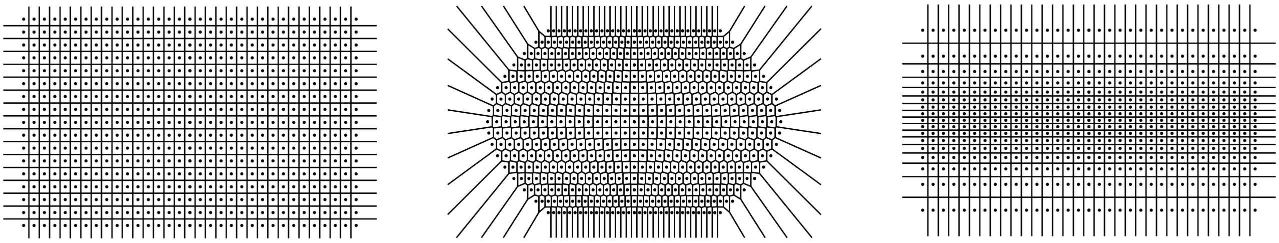

Fig. 5.1: Comparing the Voronoi diagram of the Plate Carree (a) projection to the Voronoi diagrams of Eckert V (b) projection (more similar, if the geometric accuracy approach used) and the Mercator (c) projection (more similar, if the shape preservation approach is used).

5 The Concept of Detections

This section describes, in general, selected features, procedures, and techniques used for the detection of an unknown map's projection and the estimation of its parameters.

Test and reference features. We consider a set of test features $P = \{P_1, ..., P_n\} \subset \mathbb{R}^2, (1 \leq n \leq \infty)$, where P_i represents a point, a line segment, or a closed area on the analyzed map, and analogously, a set of reference features $Q = \{Q_1, ..., Q_n\} \subset \mathbb{R}^2, (1 \leq n \leq \infty)$ on the sphere. We assume that the elements in P and Q do not intersect each other and that the element $P_i = [x_i, y_i]$ represents corresponds to $Q_i = [\varphi_i, \lambda_i]$.

Vector of determined parameters. Let us denote $\mathbf{X}_j = (\varphi_k, \lambda_k, \varphi_0, \lambda_0)$ the vector of unknown parameters representing projection $\mathbb{P}$ constant parameters, and $\widehat{\mathbf{X}} = (\widehat{\varphi}_k, \widehat{\lambda}_k, \widehat{\varphi}_0, \widehat{\lambda}_0)$ their best determined values.

Fundamentals of analysis. During the analysis, we search for a suitable map projection $\mathbb{P}$ and its parameters $\widehat{\mathbf{X}}$

$$\mathbb{P}(Q) : Q \quad \rightarrow \quad P. \tag{5.1}$$

Since the input data are affected by errors, we have an optimization problem

$$\mathbb{P}(Q) : Q \rightarrow P', \qquad P \neq P', \quad \phi(P', P) \neq \emptyset,$$

where the objective function ϕ will be minimized. The aim is to determine $\widehat{\mathbf{X}}$, which minimizes ϕ over $\mathbb{P}$

$$\widehat{\mathbf{X}} = \arg\min_{\mathbb{P}}(\phi(P, Q)). \tag{5.2}$$

This step can be solved separately for every $\mathbb{P} \in \mathbb{L}_{\mathbb{P}}$ projection, where $\mathbb{L}_{\mathbb{P}}$ denotes a list of analyzed projections. Finally, a vector $\overline{\mathbf{X}}$

$$\overline{\mathbf{X}} = \arg\min_{\forall\mathbb{P}}(\phi(\widehat{\mathbf{X}})),$$

representing the best parameters over all projections, is determined.

We do not perform this analysis in a single step, so the problem is decomposed in two phases solved separately. For any $\mathbf{X}_j \in \mathbf{X}$ representing the simplex vertex, the objective function $\phi(\mathbf{X}_j)$ between P, Q, is computed in the following sub-steps:

- project $\mathbb{P}(Q) : Q \rightarrow P'$ using $\mathbf{X}_j$,
- compute the objective function ϕ value between P, P'.

Further details can be found in Sec. 5.1.1.

Eq. 5.2 represents a classical optimization problem that can be solved in many ways. To minimize ϕ, the downhill simplex method based on the global optimizing Nelder-Mead algorithm was used. This detection technique does not need the knowledge of $\nabla\phi$; it is easy to compute, but off-line (the analysis takes time). However, the next article brings more advanced methods for the on-line detection.

The objective function ϕ and its construction are discussed in Sec. 5.1.1, it consists of multiple 0D-2D criteria.

For small territories, the elements of P' can be found directly from the map; the impact of distortions is less than the graphic accuracy of the map. The best choice is to have the spherical coordinates φ, λ. It is recommended to derive both P and Q from maps having an approximately analogous scale, type and content.

Scheme of analysis. We have P, Q sets, $\mathbb{L}_{\mathbb{P}}$ list and a vector $\mathbf{X}_j$, representing the input parameters of analyses. A detection of meridians and parallels using the modified RANSAC algorithm is performed over reference set Q. For this purpose, we need to project Q in a convenient map projection $\mathbb{P}$, transforming meridians and parallels into a set of straight lines.

Then starts the optimization process, which is repeated for all $\mathbb{P} \in \mathbb{L}_{\mathbb{P}}$ map projections. To make the optimizing process more efficient, the two following steps must be taken. An heuristic H realizing a fast (and approximate) matching between selected parts of P, P' for $\mathbf{X}_j$ is computed. A wrong matching pair with $H(P, P') > H_{max}$ is considered as unpromising, and penalized $\phi(\mathbf{X}_j) = \infty$.

Subsequently, for a P, P' pair, a detection of gross errors representing improperly drawn map elements that negatively affect the results is performed. The robust IRLS method, based on the standard Markov model, is used. Based on the analysis of about one hundred early maps, an empirical limit of 20% of incorrectly drawn elements was determined. If P, P' contain more than 25% gross errors, a map is suspected of being unprojected or unsimilar, and $\mathbf{X}_j$ is penalized in the same manner.

Otherwise P, P', form a suitable pair, whose objective function value is determined from ϕ. The selected matching algorithms for the 0D-2D entities are members of the objective function ϕ. An essential role is played by point features algorithms; analyses of other features (not always available) are more complementary and improve the results.

For all $\mathbb{P} \in \mathbb{L}_{\mathbb{P}}$ the vector $\widehat{\mathbf{X}}$ of best parameters is determined. Finally a vector $\overline{\mathbf{X}}$ with the minimum objective function value is found, and its parameters are assigned to the analyzed map.

Geometric accuracy approach vs. shape preservation approach. Map projections are divided into several categories and subcategories according to their common geometric and cartographic properties [46], [6], which are reflected in the shape of meridians and parallels. Therefore, a pair of maps in projections belonging to the same category has the analogous shape of the graticule, but the residuals of corresponding points in these projections may not be small. A classification of the analyzed map's projection into a specific category with the detailed determination of parameters should be one of the preferred outcomes. On the other hand, two maps may be created in projections belonging to different categories. They may have small residuals of corresponding points, but their graticules have a different shape; see Fig. 5.1. The geometric accuracy approach classifies the Eckert projection to be more similar to the plate carree projection. On the contrary, the shape preservation approach finds the Mercator projection to be more similar to the Plate Carree projection.

Which criterion is more important: a similar shape of the graticules, or small residuals of corresponding points? In the author's opinion, both criteria should be taken into account. The standard deviation is a current descriptor for the first case; the suitable parametrization of the shape, or the analysis of auxiliary 2D planar structures (Voronoi diagrams), are appropriate for the second case. This fact is important in the design of the objective function ϕ, which contains both approaches.

Results of the analyses. Depending on the results of the analyses, in descending order of relevance and level of distinction, the projection parameters (case 1), projection type (case 2), or projection category (case 3) might be determined, or the map is unprojected (case 4). Case 1: estimated parameters for all the analyses are identical. Case 2: results show a large consensus in most of the criteria. Case 3: More projections have similar, inconclusive and unclear results. Case 4: The results are discrepant, large residuals are present, and $\mu = 0$ (see Eq. 5.5). However, this situation may occur when none of the parameters can be reliably determined (an unprojected map without a geodesic basis).

5.1 Nelder-Mead simplex method

The Nelder-Mead method minimizes the objective function ϕ without explicit knowledge of its gradient $\nabla\phi$. It belongs to the global optimizing simplex methods using pure heuristics. A simplex is represented by the $m + 1$ dimensional polytope. It is a proper method for solving small dimensional optimizing problems [16]. In general, the minimized function ϕ is not strictly convex. Therefore, the algorithm may fail or become stuck in the local minimum.

Algorithm description. A detailed description of the algorithm can be found in [3], [7]. The Nelder-Mead method uses an heuristic approach. During the optimization, the worst point of the simplex is replaced. The algorithm can be summarized as follows:

1. Select a random simplex $\mathbf{X}(m + 1, m)$ and order its vertices as $\phi(\mathbf{X}_1) < \phi(\mathbf{X}_2) < ... < \phi(\mathbf{X}_{m+1})$

2. Compute the simplex centroid $\bar{\mathbf{X}}$ from m best points
$$\bar{\mathbf{X}} = \frac{1}{m}\sum_{j=1}^{m}\mathbf{X}_j.$$

3. Compute the reflection point $\mathbf{X}_r$
$$\mathbf{X}_r = (1 + \alpha)\bar{\mathbf{X}} - \alpha\mathbf{X}_{m+1}.$$
Test Eq. 5.3 for $\mathbf{X}_r$, perform the flip into the search space S, if necessary. The reflected point $\mathbf{X}_r$ is accepted: $\mathbf{X}_{m+1} = \mathbf{X}_r$, $\phi(\mathbf{X}_{m+1}) = \phi(\mathbf{X}_r)$, if $\phi(\mathbf{X}_1) < \phi(\mathbf{X_r}) < \phi(\mathbf{X}_m)$.

4. If $\phi(\mathbf{X}_r) < \phi(\mathbf{X}_1)$, compute the expansion point $\mathbf{X}_e$
$$\mathbf{X}_e = (1 + \alpha\beta)\bar{\mathbf{X}} - \alpha\beta\mathbf{X}_{m+1}.$$
Test Eq. 5.3 for $\mathbf{X}_e$, perform the flip into the search space S, if necessary. The expansion point $\mathbf{X}_e$ is accepted: $\mathbf{X}_{m+1} = \mathbf{X}_e$, $\phi(\mathbf{X}_{m+1}) = \phi(\mathbf{X}_e)$, if $\phi(\mathbf{X}_e) < \phi(\mathbf{X}_r)$, otherwise: $\mathbf{X}_{m+1} = \mathbf{X}_r$, $\phi(\mathbf{X}_{m+1}) = \phi(\mathbf{X}_r)$.

5. If $\phi(\mathbf{X}_r) < \phi(\mathbf{X}_m)$, compute the outer contraction point $\mathbf{X}_{co}$
$$\mathbf{X}_{co} = (1 + \alpha\gamma)\bar{\mathbf{X}} - \alpha\gamma\mathbf{X}_{m+1}.$$
Test Eq. 5.3 for $\mathbf{X}_{co}$, perform the flip into the search space S, if necessary. The contraction

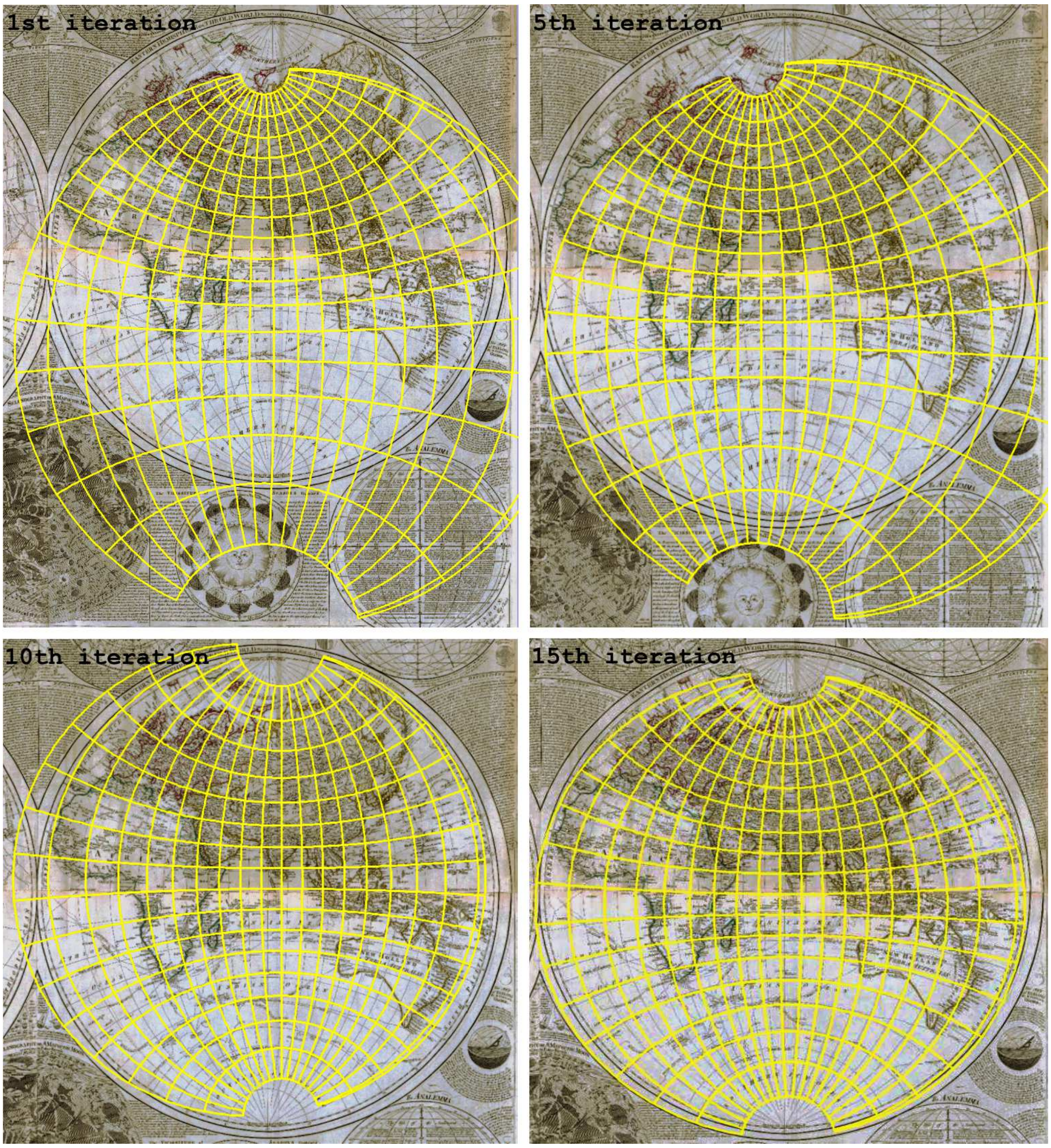

Fig. 5.2: Map 8: Stereographic projection in the oblique aspect, parameters estimated using the Nelder-Mead algorithm; 1st, 5th, 10th, 15th iterations are shown.

point $\mathbf{X}_{co}$ is accepted: $\mathbf{X}_{m+1} = \mathbf{X}_{co}$, $\phi(\mathbf{X}_{m+1}) = \phi(\mathbf{X}_{co})$, if $\phi(\mathbf{X}_m) < \phi(\mathbf{X}_r) < \phi(\mathbf{X}_{m+1})$, otherwise a shrink is performed (go to step 7).

6. Otherwise, compute the inner contraction point $\mathbf{X}_{ci}$
$$\mathbf{X}_{ci} = (1+\gamma)\bar{\mathbf{X}} + \gamma \mathbf{X}_{m+1}.$$
Test Eq. 5.3 for $\mathbf{X}_{ci}$, perform the flip into the search space S, if necessary. The contraction point $\mathbf{X}_{ci}$ is accepted: $\mathbf{X}_{m+1} = \mathbf{X}_{ci}$, $\phi(\mathbf{X}_{m+1}) = \phi(\mathbf{X}_{ci})$, if $\phi(\mathbf{X}_r) > \phi(\mathbf{X}_{m+1})$, otherwise a shrink is performed (Go to step 7).

7. Perform a shrink of the simplex; only the best point does not change
$$\mathbf{X}_j = \mathbf{X}_1 + \alpha(\mathbf{X}_j - \mathbf{X}_1),$$
where $j = 2, ..., m+1$. Test Eq. 5.3 for $\mathbf{X}_j$, perform the flip into the search space S, if necessary.

8. Order $\mathbf{X}$ vertices so as $\phi(\mathbf{X}_1) < \phi(\mathbf{X}_2) < ... < \phi(\mathbf{X}_{n+1})$.

9. The terminal condition: if
$$|\phi(\mathbf{X}_1) - \phi(\mathbf{X}_{m+1})| < \varepsilon,$$
stop the iteration process; otherwise, go to step 2.

The following values of the reflection, expansion, contraction, and shrink coefficients have been set:

$$\alpha = 1, \qquad \beta = 2, \qquad \gamma = 0.5, \qquad \delta = 0.5.$$

Based on the deep numerical tests, for our problem $\varepsilon = 1.0^{-6}$ was set. The implementation is easy, but a convergence is slower compared to NLSP methods. Therefore, the Nelder-Mead method can not be used for on-line analysis.

Vertex reflection. For all determined constant parameters of the projection, the maximum and minimum values form an edge of search space S. If any simplex vertex $\mathbf{X}_j \in \mathbf{X}$ falls outside the search space S, i.e., $\mathbf{X}_j \notin S$, its simple flip must be done

$$\mathbf{X}_j = \begin{cases} 2\mathbf{X}_j^{min} - \mathbf{X}_j, & \mathbf{X}_j < \mathbf{X}_j^{min}, \\ 2\mathbf{X}_j^{max} - \mathbf{X}_j & \mathbf{X}_j > \mathbf{X}_j^{max}. \end{cases} \tag{5.3}$$

The extreme values of parameters $\varphi_k, \lambda_k, \varphi_0, \lambda_0$ can be found in Tab. 1, the R' values in Sec. 5.1.1.

5.1.1 Objective function ϕ

The objective function ϕ is to be proposed to involve the 0D-2D elements of the analyzed map. Taking into account Eq. 5.2, its global minimum is found, and the determined parameters $\widehat{\mathbf{X}}$ are assigned to the analyzed map.

The geometric mean is less sensitive to (possible) discrepant values of criteria than the arithmetic mean. In accordance with Sec. 6, the objective function ϕ, which combines several invariants (e.g., distance functions), is written as follows

$$\phi = \left[\frac{(1+|\alpha|)\,\sigma_0}{1+\mu[\%]} \cdot C_{nnd} \cdot \overline{d}_2(P,P') \cdot \overline{d}_2(F(P),F(P'))\right]^{1/4}.$$

The geometric accuracy approach is represented by the apriori variance σ_0, the cross nearest distance C_{nnd}, and the similarity ratio μ. The turning function distance $\overline{d}_2(F(P),F(P'))$ of auxiliary planar structures generated from 0D features, as well as the distance $\overline{d}_2(P,P')$ of 1D-2D features, belong to the shape preserving approach. All invariants are widely discussed in Sec. 6

Determining ϕ is computationally expensive; therefore, it can be replaced by the heuristic H, which is cheaper. The idea is very simple: If one criterion gives bad results, the remaining criteria probably will not provide appropriate results.

Simplex initialization. The simplex vertices $\mathbf{X}(m+1,m)$, $m = 5$,

$$\mathbf{X} = \begin{pmatrix} R'_1 & \varphi_{k,1} & \lambda_{k,1} & \varphi_{0,1} & \lambda_{0,1} \\ ... & ... & ... & ... & ... \\ R'_6 & \varphi_{k,6} & \lambda_{k,6} & \varphi_{0,6} & \lambda_{0,6} \end{pmatrix}$$

are randomly initialized, φ_k, λ_k intervals can be found in Tab. 1. Analogously $\varphi_0 \in \langle 0, \varphi_{max}\rangle$ and for the normal aspect $\lambda_0 \in \langle \min(0, \lambda_{min}), \lambda_{max}\rangle$, otherwise $\lambda_0 = 0$. A different scale of both analyzed sets must be taken into account. The simplex contains R', representing the reduced radius of the Earth as a determined parameter. The initial value R'_0 is computed from the scale factor

$$s = \sqrt{\lambda_1^2 + \lambda_2^2}$$

of a 2D Helmert transformation $T(P,P')$

$$R'_0 = R/s,$$

where λ_1, λ_2 are transformation coefficients and $R' \in \langle 0.1R'_0, 10R'_0\rangle$. It is apparent that R' represents the radius of the sphere reduced to the scale of the analyzed map.

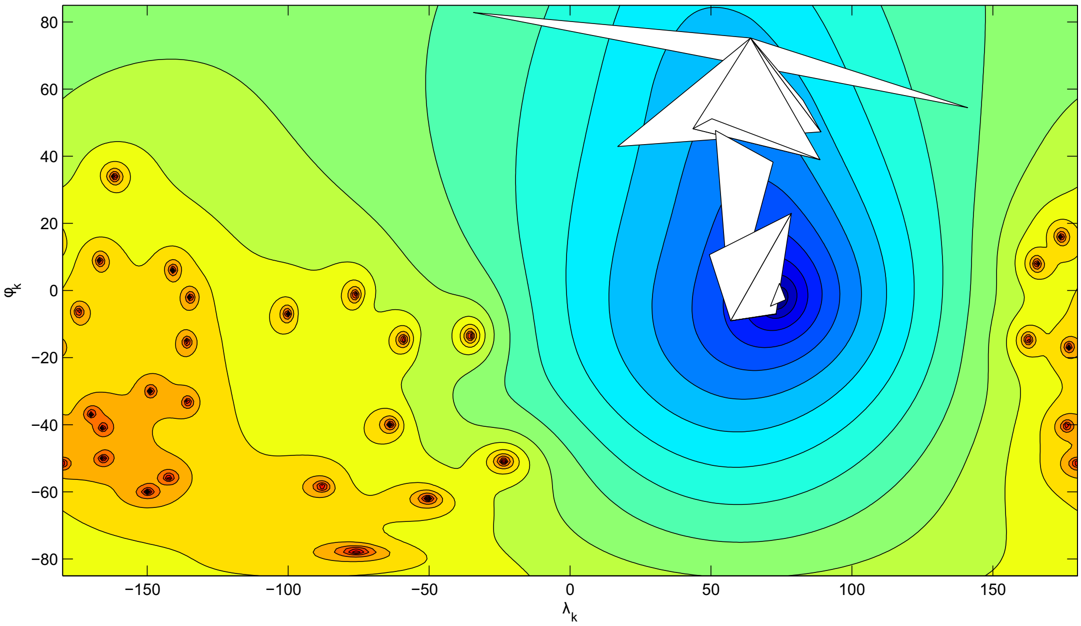

Fig. 5.3: A convergence of the simplex $\mathbf{X}$ to the global minimum $\varphi_k = 1.0°S$, $\lambda_k = 73.1°E$ (Map 8), selected steps from the first 30 iterations are visualized.

Simplified objective function ϕ. Instead of the "full" objective function ϕ, the simplified version involving only 0D elements can be used. Both sets P, P' are aligned in the centers of mass $C = [X_c, Y_c]$, $c = [x_c, x_c]$

$$\begin{aligned} x_c &= \frac{1}{n}\sum_{i=1}^{n} x_i \qquad y_c, = \frac{1}{n}\sum_{i=1}^{n} y_i, \\ X_c &= \frac{1}{n}\sum_{i=1}^{n} X_i, \qquad Y_c = \frac{1}{n}\sum_{i=1}^{n} Y_i. \end{aligned}$$

The vector of residuals $\mathbf{r}$ is written as follows

$$\mathbf{r} = \begin{pmatrix} (X_1 - X_c) - (x_1 - x_c) \\ \dots \\ (X_m - X_c) - (x_m - x_c) \\ (Y_1 - Y_c) - (y_1 - y_c) \\ \dots \\ (Y_m - Y_c) - (y_m - y_c) \end{pmatrix},$$

and the objective function ϕ

$$\phi = \mathbf{r}^T \mathbf{W} \mathbf{r}, \qquad (5.4)$$

where $\mathbf{W}$ represents the weight matrix of elements. This method minimizes $\|\mathbb{P}(Q) - P\|_2^2$ and gives results analogous to 2D homothetic transformation (no rotation is allowed); it is computationally cheap. In general, the results of both methods are in accordance.

Computation of the objective function. A practical computation of objective function $\phi(\mathbf{X}_j)$, $\mathbf{X}_j(1, m)$, for actual P, Q, $\mathbb{P}$ in one Nelder-Mead iteration consists of several sub-steps. Let $\mathbf{X}_j$ represent the j-th vertex of the simplex with stored cartographic parameters:

1. *Conversion to the transverse/oblique aspect*

   Convert all elements $Q_i = [\varphi, \lambda_i] \rightarrow [\varphi_i', \lambda_i']$ using Eq. 3.1, where φ_k, λ_k are stored in the actual simplex $\mathbf{X}_j$, if transverse or oblique aspects are needed. For the normal aspect of $\mathbb{P}$, reduce $\lambda = \lambda - \lambda_0$ and do not perform a conversion.

2. *Applying projection equation*

   Project all Q elements to P' using Eq. 5.1 of $\mathbb{P}$.

3. *Heuristic check*

   Compute heuristic $H(\mathbf{X}_j)$ between P, P'; see Sec. 5.2. For a successful case go to step 4, otherwise set $\phi(\mathbf{X}_j) = \infty$ and skip steps 4-5;

4. *Outlier detection*

   As an optional step, outliers between P, P' using IRLS are analyzed; see Sec. 5.3. Create a weight matrix $\mathbf{W}$ storing the weights of all analyzed P, P' elements; otherwise, $\mathbf{W} = \mathbf{I}$.

5. *Objective function value computation*

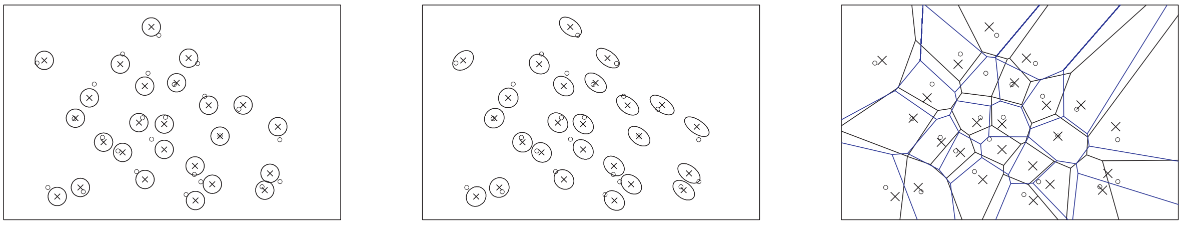

Fig. 5.4: Points with drawn uncertainty regions represented by circles $r = \varepsilon$ (a) and Tissot indicatrices (b) in the sinusoidal projection, Voronoi diagrams of both data sets P, P' (c).

> Compute the objective function ϕ for P, P'; full or simplified version is used, the weights of elements $\mathbf{W}$ are involved.

A sequence of iterations for Map 8 created in a stereographic projection is shown in Fig. 5.2. The course of objective function ϕ is interesting; the global minimum close to the transverse aspect, and many local maxima caused by points with $\varphi' \doteq -\pi/2$ projected to infinity

$$\lim_{\varphi' \to -\pi/2} (\rho) = \infty,$$

have been found. It is apparently that the problem is not convex.

Time complexity. The most time-consuming steps are connected with the $d_2(F(P_i), F(P_i'))$ invariant based on the Voronoi diagram analysis. The Voronoi diagram construction takes $O(N \log_2(N))$ time. It is known, that the average number of Voronoi vertices per Voronoi cell does not exceed six. The merging algorithm uses $O(kN \log_2(N))$ time, where $k = 6$ represents the amount of shared vertices. Creating corresponding merged faces $F(P), F(P')$ from Voronoi cells takes $O(N^2 \log_2^2(N))$, and their analysis $O(2N^2)$ time. Therefore, this step has the overall complexity of $O(N^4 \log_2^2(N))$.

Because $N < 100$, this step can be considered as acceptable (but probably off-line).

5.2 The heuristic strategy

Before the $\phi(\mathbf{X}_j)$ computation, $\mathbf{X}_j$ is verified to see whether it will be a good candidate for deeper analysis. To reduce the computational time, instead of $\phi(\mathbf{X}_j)$, an heuristic $H(\mathbf{X}_j)$ is to be calculated. Omitting this step may significantly increase the processing time (by a factor of approximately 10).

If $H(\mathbf{X}_j) > H_{max}$, the objective function $\phi(\mathbf{X}_j)$ is penalized

$$\phi(\mathbf{X}_j) = \infty.$$

Keep in mind that an inappropriate heuristic parameter H_{max} setting may also exclude part of perspective $\mathbf{X}_j$, (too strict) or the heuristic becomes inefficient (too weak).

The presented heuristic is based on two types of preliminary analyses: an analysis of point features using similarity transformation (standard deviation σ), and an analysis of line features using the turning function (turning function $d_2(P, P')$ distance). The second invariant has been used repeatedly and successfully for finding corresponding elements in the maps [1], [15].

Coordinate limits. The simplest form of heuristic brings the maximum coordinate limit MAX_COORD. This procedure is used primarily for conformal projections, and excludes projections of the inappropriate meta pole position φ_k, λ_k, where P' coordinates go to infinity. This step takes $O(n)$ time.

Heuristic based on the similarity transformation. Let us suppose a similarity between P and P' given by the key $t_s(m, \alpha, s_x, s_y)$ with one scale m, two shifts s_x, s_y and rotation α. Large values of α can cause an acceptance of the undesirable set of features (see Fig. 5.6). Instead of the projection in the normal aspect, the transverse or oblique aspects may be detected. Thus, the real similarity is not the best indicator, if rotation is allowed. Instead, the homothetic transformation can be used.

The similarity ratio μ, $\mu \in \langle 0, 1 \rangle$ is defined as the fraction of norms from both sets P and S

$$\mu = \frac{\|S\|}{\|P\|}, \tag{5.5}$$

where $S \subset P$ represents a subset $S = \{S_1, ..., S_m\}, m \leqq n,$ of transformed features P given by

$$S = |T(P) - P'| < \varepsilon. \tag{5.6}$$

The Eq. 5.5 formally expresses the relationship between the amount of transformed points $T_S(P)$ located within a distance ε from P' points and the amount of projected points P.

Let us briefly comment on ε representing the radius of the *uncertainty region*, which is discussed in Sec. 6.1. Considering dimension d (a longer edge of the P' min-max box), n as the amount of analyzed points and

$\beta, \beta \in (0,1)$ as the sensitivity factor, the radius ε is computed as follows

$$\varepsilon = \beta \frac{d}{\sqrt{n}}, \qquad d = \max(x_{max} - x_{min}, y_{max} - y_{min}),$$

It is noticeable that ε represents an approximate root of point density

$$d/\sqrt{n} \doteq \sqrt{A_{\mathcal{H}}/n}$$

in relation to the area $A_{\mathcal{H}}$ of the convex hull $\mathcal{H}(P')$ (or min-max box) multiplied by sensitivity factor β. This geometrically represents the radius of a circle centered at P'_i. In our case, $\beta = 0.2$ was chosen empirically to prevent the elimination of an unnecessary large amount of samples. In the different strategy, the dependance on the map scale, can be used

$$\varepsilon = \beta \cdot G \cdot 1000 \qquad [km],$$

where $\beta = 10 - 30$.

Let us set $H_\mu(\mathbf{X}_j) = \mu$. If $H_\mu(\mathbf{X}_j) < 0.75$, the $\phi(\mathbf{X}_j)$ is penalized. This heuristic strategy is set as primary and takes $O(n)$ time. Another approach, based on the expected distance to the nearest neighbor using the radius r_c circle circumscribed to P'

$$\varepsilon = \beta \cdot r_c/(2\sqrt{n}),$$

was used by [51],

Heuristic based on the turning function. This complementary heuristic strategy is focused on 1D features that may be present on the map. If there are no input line features, the heuristic attempts to use detected meridians and parallels. The turning function Θ seems to be an appropriate fast shape descriptor [1]. First, both data sets P and P' are rescaled.

Let us denote the turning functions of corresponding line features P_i, P'_i as $\Theta(P_i)$ and $\Theta(P'_i)$ and their L_2 distance $d_2(P_i, P'_i) = \|\Theta(P_i) - \Theta(P'_i)\|$. The average dissimilarity $\overline{d_2}(P, P')$ between P and P', which contain n features, is given by

$$\overline{d_2}(P, P') = \frac{1}{n} \sum_{i=1}^{n} d_2(P_i, P'_i) \tag{5.7}$$

Consider an heuristic criterion $H_d = \overline{d_2}(P, P')$ and the amount of points in the analyzed line feature m. If

$$H_d > \beta m \overline{d_2}(P, P')_{max},$$

the $\phi(\mathbf{X}_j)$ is penalized. Currently, we set $\overline{d_2}(P, P')_{max} = 1$. The algorithm for the $d_2(P_i, P'_i)$ computation is described in Sec. 7. This step takes $O(n^2)$ time.

Heuristic based on the Voronoi diagram. If the Voronoi diagram $\mathcal{V}(P)$ is constructed under P and does not contain at least three bounded Voronoi cells, due to a lack of suitable cells, the Voronoi analyses are disabled. Cells of inappropriate shapes have been removed from further analysis.

The values of heuristic parameters $\beta, \mu_{min}, \overline{d_2}(P, P')_{max}$ must be set so that both heuristics have a *similar efficacy*. If all $\mathbf{X}_j$ are marked as inappropriate, the analyzed map may be classified as unprojected.

5.3 Removing gross errors

Due to the fact that many old maps are not constructed on a solid geometric or geodetic basis, some of the drawn map elements are highly influenced by errors. This issue has been mentioned in several publications: [17], [44], [3].

Flaws are likely distributed according to the normal law and have both a random and a systematic character. Some map elements can be affected by gross errors; therefore, it is important to locate, find, and exclude these large mistakes automatically from further analysis. The impact of gross errors on the reconstructed graticule is illustrated in Fig. 5.7. An important role is also played by the generalization, but its influence is usually below the level of gross errors.

However, other factors, such as paper distortion or the aging effect, may also contribute to the geometric inaccuracy. Primarily, they do not represent serious mistakes and have a systematic character. In most cases, their effect is significantly less and cannot be detected by this technique.

We assume a similarity between P, P'. Therefore, the problem can be solved by repeated 2D transformation of $P \to P'$ with outlier detection. There are different strategies to identify gross errors depending on the amount of errors in the data set. When multiple gross errors exist (up to 20%), the IRLS method based on the standard Markov model, resistant to the influence of the outliers, is appropriate [52].

Mathematical model. Let us denote $\mathbf{v}$ to be a vector of residuals, $\mathbf{A}$ a matrix of the model, $\mathbf{x}$ a vector of estimated model parameters, and $\mathbf{l}$ a vector of measurements. We solve the linear model

$$\mathbf{v} = \mathbf{A}\mathbf{x} - \mathbf{l},$$

where the weighted least squares condition $\sum \mathbf{W}\mathbf{v}\mathbf{v} = \min$ is applied. Then

$$\mathbf{x} = (\mathbf{A}^T \mathbf{Q}_l^{-1} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{Q}_l^{-1} \mathbf{l}.$$

The weight matrix $\mathbf{W} = \mathbf{Q}_l^{-1}$ is defined as the inverse of the measurement co-factor matrix $\mathbf{Q}_l$ ($\mathbf{Q}_l = \mathbf{E}$ for non-weighted transformation); the cofactor matrix $\mathbf{Q}_v$ of the

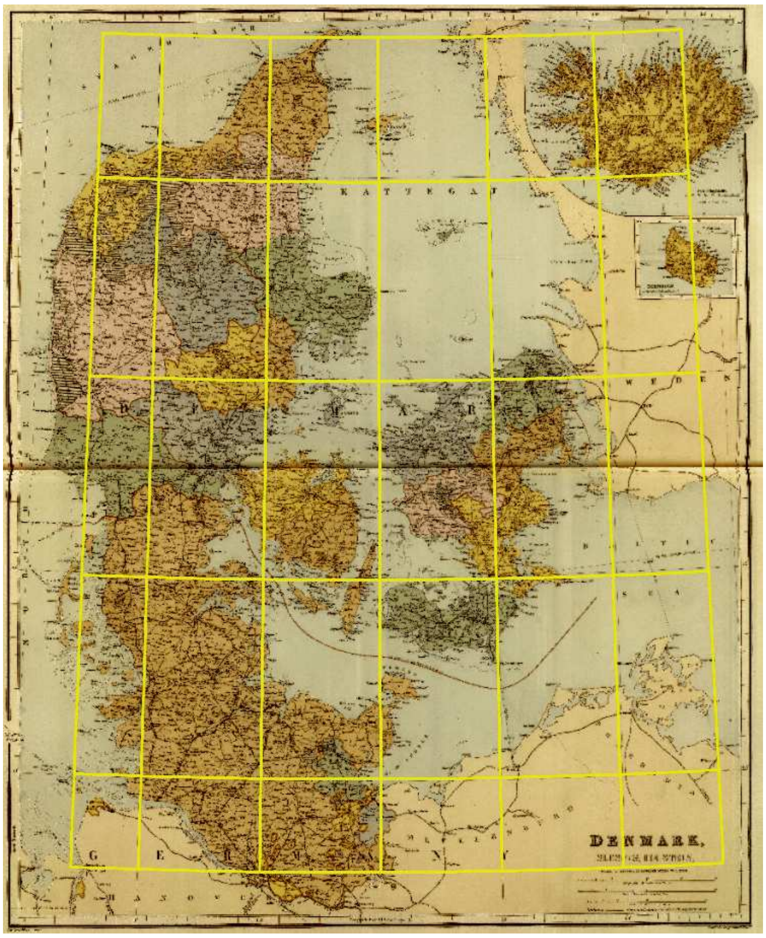

Fig. 5.5: Map 6: a graticule of the equidistant conic projection, reconstruction from the estimated parameters.

estimated residuals is

$$\mathbf{Q}_v = \mathbf{Q}_l - \mathbf{A}(\mathbf{A}^T\mathbf{Q}_l^{-1}\mathbf{A})^{-1}\mathbf{A}^T,$$

The standard deviation σ_v of the estimated residuals

$$\sigma_{v_i} = \sigma_0\sqrt{\mathbf{Q}_v(i,i)}$$

is computed from the diagonal items of $\mathbf{Q}_v$ and apriori variance factor σ_0.

The Danish method [27] iteratively decreases weights of measurements suspected to be outliers. Weights of "proper" measurements are not changed. For every measurement in iteration step k, we consider a normalized residual $\overline{v_i} = v_i/\sigma_{v_i}$, and compare it to the specific threshold

$$\mathbf{f}_i^{(k)} = \begin{cases} \mathbf{f}_i^{(k-1)} \cdot e^{-0.5|\overline{v_i}|} & \text{if } |\overline{v_i}| \geq 2.0, \\ \mathbf{f}_i^{(k-1)} & \text{else} \end{cases} \tag{5.8}$$

and compute a new weight factor $\mathbf{f}_i^{(k)}$ from $\mathbf{f}_i^{(k-1)}$. The new co-factor matrix $Q_l^{(k)}$ is

$$Q_l^k(i,i) = \begin{cases} \mathbf{E} & k = 1, \\ 1/\mathbf{f}_i^{(k)} & k > 1. \end{cases}$$

The iteration steps are repeated until the given condition

$$\left|\sigma_0^{(k)} - \sigma_0^{(k-1)}\right| < \varepsilon$$

is true.

The elimination of outliers decreases the values of the objective function ϕ by 1-2 orders of magnitude.

5.4 Detection of Meridians and Parallels Using RANSAC

Another possible data source, reliable for analysis and easily available in most maps, are meridians and parallels. These map elements keep the typical properties of the graticule more efficient than point features. They help prevent shape discretization and improve the reliability of the analyses. The proposed algorithm detects all types of meridians and parallels forming more than 4 points, which may occur in both data sets P, Q.

Because of the complex shape, meridians and parallels cannot be detected directly from P, Q. At first, Q must be projected to P' in a convenient projection where the meridians and parallels are formed by the straight

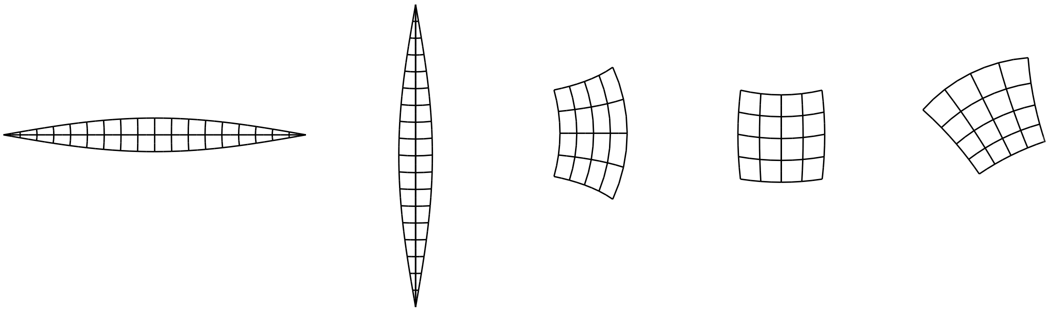

Fig. 5.6: Similar shapes of territories along the central meridian in an equidistant cylindrical projection, transverse aspect (a), and Bonne projection, normal aspect (b), and territories along the intersection of the prime meridian and equator in an equidistant cylindrical projection, transverse aspect (c), Bonne projection, normal aspect (d) and Lambert conformal projection, transverse aspect (e).

lines. Projected meridians and parallels are easily detectable by RANSAC, where the model function is represented by the linear fit function. For this purpose, the Plate Carree projection given by equations $x = R\lambda$, $y = R\varphi$ is used.

Projected meridian and parallel. Let us denote the projected sampled meridian as a set of points $\mathcal{M} = \{\mathcal{M}_j\}_{j=1}^{m} \subset \mathbb{R}^2$ and analogously, a projected sampled parallel $\mathcal{P} = \{\mathcal{P}_j\}_{j=1}^{m} \subset \mathbb{R}^2$, where the angle between any triplet of points is π. Taking into account Eq. 3.3, the partial derivatives are

$$\frac{\partial f}{\partial \varphi} = 0, \frac{\partial f}{\partial \lambda} = R, \frac{\partial \varrho}{\partial \varphi} = R, \frac{\partial \varrho}{\partial \lambda} = 0$$

and $\gamma_m = 0, \gamma_p = \frac{\pi}{2}$.

Basic formulas. The detection method based on the RANSAC algorithm will be modified to provide, not only the best solution, but more acceptable solutions, detecting lines fulfilling the condition of $\gamma = 0 \pm k\frac{\pi}{2}$ (i.e., projected meridians and parallels). Instead of the one best fit projected meridian or parallel, a set of projected meridians and parallels will be detected. Their vertices will be used in further analysis.

We consider $D = P', |D| = n$ to be an input data set, $S, S \subset D, |S| = m$ a sample (containing detected meridian or parallel points), $p^\star$ input parameters computed with a model function F (linear fit function)

$$F(S) : S \to p^*,$$

p^{**} output parameters minimizing the cost C of the function ρ

$$C \quad = \quad \rho(p^*, D).$$

The number of iterations k for probability $p = 0.99$ is given by

$$k \geqq \frac{\log(1-p)}{\log(1-w^n)}, \qquad (5.9)$$

where w is the number of inliers contained in D.

Linear fit function The model function $F(S) = \{X_c, Y_c, \xi\}$ represents a non-weighted linear fit function with residuals to X, Y axes, where

$$X_c = \sum_{i=1}^{n} \frac{X_i}{n}, \qquad X_i' = X_i - X_c,$$
$$Y_c = \sum_{i=1}^{n} \frac{Y_i}{n}, \qquad Y_i' = Y_i - Y_c,$$

and

$$\xi \quad = \quad 0.5 \tan^{-1}\left(\frac{2\sum_{i=1}^{n} X_i' Y_i'}{\sum_{i=1}^{n} X_i' X_i' - \sum_{i=1}^{n} Y_i' Y_i'}\right),$$
$$\rho \quad = \frac{1}{n} \quad \sum_{i=1}^{n} d_i^2,$$
$$d \quad = \quad (X_i' - X_c)\sin\xi - (Y_i' - Y_c)\cos\xi.$$

X_c, Y_c are centroids of P', ξ is the polar angle of the fitted line expressing the slope. All 3 parameters could be transformed easily into the slope form

$$c \cdot y = m \cdot x + b, \quad m = \tan(\xi), \quad b = Y_c - m \cdot X_c.$$

For a meridian projected with Plate Carree, the output parameters $p^{**} = \{\xi, b, \}$ are simply given as

$$\xi = \frac{\pi}{2}, \quad b = Y_c, \quad c = 0, \quad m = 1$$

and analogously for a projected parallel

$$\xi = 0, \quad b = X_c, \quad c = 1, \quad m = 0.$$

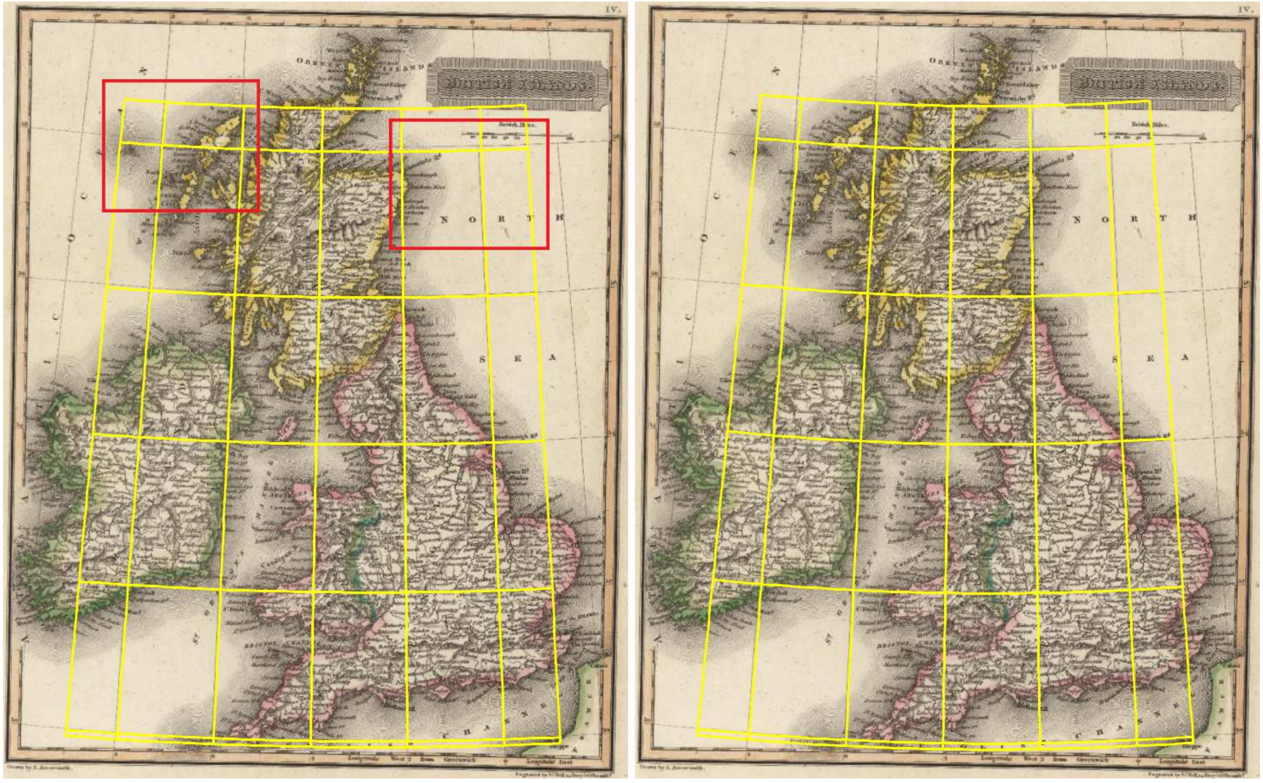

Fig. 5.7: The impact of outliers on Map 7: reconstructed graticule with outliers (a) and without outliers (b). The discrepancies are noticeable on the northern part of the map.

Storing detected samples. Each detected RANSAC sample S represented by a line with $p^{**} = \{\xi, b, \}$ is stored into a hash table, according to its address $\mathcal{A}$ given by the hash function $h(\xi, b)$

$$\begin{aligned}\mathcal{A} \;=\; & [((int)(m_\xi \cdot |\xi|))\&0xFFFF] \\ & \ll 16|\,[((int)(m_b \cdot |b|))\&0xFFFF]\end{aligned}$$

Detected samples representing one meridian or one parallel have the same values of $\mathcal{A}$; they differ only in the cost C. We keep only a sample with the minimum cost C representing the best-fit line. Values m_ξ, m_b are multiplication factors. These affect the hashing sensitivity, the minimum of the acceptable slope, and the vertical shift differences of best-fit lines.

A modified detection algorithm based on RANSAC. Consider a sample S storing meridian or parallel points, C its cost given by ρ, p^{**} parameters of the fitting line, and h its hash. The detection procedure can be summarized as follows:

1. *Initialization*

   Initialize variables $l_{max} = 0$, $l = 0$, $k = 0$.

2. *Main loop*

   While $l < l_{max}$ repeat steps 3-6, where a suitable meridian or parallel are found and added to the list of meridians $\mathbb{L}_\mathcal{M}$ or parallels $\mathbb{L}_\mathcal{P}$.

3. *Perform a RANSAC procedure*

   Call a RANSAC procedure for the following parameters: $P', k, m, \varepsilon_{acc}$, and find any acceptable solution S_{acc} described with parameters $C_{acc}, p_{acc}, h_{acc}$.

4. *Test the RANSAC solution: meridian*

   If any acceptable sample S_{acc} is found and its angle $|p_{acc}.\xi| = \frac{\pi}{2}$, it represents a meridian. Perform the following steps:

   (a) Test, if any meridian with the same hash h_{acc} is in $\mathbb{L}_\mathcal{M}$. If such a meridian $\mathcal{M} \in \mathbb{L}_\mathcal{M}$ with worse cost $\mathcal{M}.C_{acc} > C_{acc}$ exists, update its properties in $\mathbb{L}_\mathcal{M}$:

   $$\mathcal{M}.S_{acc} = S_{acc}, \mathcal{M}.C_{acc} = C_{acc}, \mathcal{M}.p^{**}_{acc} = p^{**}_{acc}.$$

   (b) Otherwise, create a new meridian $\mathcal{M}(S_{acc}, C_{acc}, p_{acc}, h_{acc})$, which is inserted in the list: $\mathbb{L}_\mathcal{M} \leftarrow \mathcal{M}$.

5. *Test the RANSAC solution: parallel*

If any acceptable sample S_{acc} is found and its angle $|p_{acc}.\xi| = 0$, it represents a parallel. Perform the following steps:

(a) Test, if any parallel with the same hash h_{acc} existing in $\mathbb{L}_{\mathcal{P}}$. If such a parallel $\mathcal{P} \in \mathbb{L}_{\mathcal{P}}$ with worse cost $\mathcal{P}.C_{acc} > C_{acc}$ exists, update its properties in $\mathbb{L}_{\mathcal{P}}$:

$$\mathcal{P}.S_{acc} = S_{acc}, \mathcal{P}.C_{acc} = C_{acc}, \mathcal{P}.p^{**}_{acc} = p^{**}_{acc}.$$

(b) Otherwise, create a new parallel $\mathcal{P}(S_{acc}, C_{acc}, p_{acc}, h_{acc})$ which is inserted in the list: $\mathbb{L}_{\mathcal{P}} \leftarrow \mathcal{P}$.

6. Increment $l = l + 1$.

The repeatedly called RANSAC procedure with input parameters $P', k, m, \varepsilon_{acc}$ returns any acceptable sample S_{acc}, which satisfies both the conditions of cost C_{acc} and length m. It can be summarized as follows:

1. *Main loop*

   While $k < k_{max}$, repeat steps 2-5.

2. *Initialize random sample*

   Add 2 random P' items to S and compute the initial fit $p^* = F(S)$ and its cost $C = \rho(p^*)$.

3. *Create a sample adding P' elements*

   Add random P'_i elements to a sample S. Repeat sub-steps a-b for all P'_i, $i = 1, ..., n$ not presented in S.

   (a) Store properties of the actual sample S

   $$S_{old} = S; C_{old} = C; p^*_{old} = p^*.$$

   (b) Add P'_i temporary to S and compute a new fit $p^* = F(S)$ and its cost $C = \rho(p^*)$ for updated S. If a cost difference ΔC satisfies

   $$\Delta C = |C - C_{old}| > \frac{|\varepsilon_{acc}|}{\sqrt{m_i}},$$

   the new sample S is unacceptable and we return to the previous one, S_{old}

   $$S = S_{old}; C = C_{old}; p^* = p^*_{old}.$$

4. *Test for acceptability of the sample*

   If $|S| \geq m$ and its cost $C < \varepsilon_{acc}$ are acceptable, the suitable sample S has been found. Update the properties of the acceptable solution

   $$S_{acc} = S; C_{acc} = C; p^{**}_{acc} = p^*,$$

   compute its hash $h_{acc} = H(p^{**}_{acc}.\xi, p_{acc}.b)$ and return a sample S.

5. Increment $k = k + 1$.

Both algorithms are computationally expensive and partially inappropriate for on-line analysis. Saving meridians and parallel points into separate input files seems to be a better option.

6 Detection Methods for Point Features

Most of the analyzed early maps are available in raster formats. For purposes of analysis, the complete vectorization of the map is somewhat redundant. Therefore, a set of 0D features (identical points) is a primary and typical source material for analysis.

This feature applies particularly to libraries with large printed map collections, where the complete vectorization is impossible due to financial or time constraints. The majority of their collection is stored in the raster formats.

Hence, the partial vectorization is done by the user. It depends on his knowledge and experience, which are reflected in the choice of the identical points. Overall, the detection process appears more laborious. The below mentioned invariants will be used to construct the objective function ϕ.

6.1 Detection Methods Based on the Geometric Accuracy

Let us bring a short overview of methods describing the geometric accuracy. In the simplest case, the objective function ϕ may express the least squares condition.

6.1.1 Detection Methods based on 2D Transformations

These techniques are based on the assumption of the linear relationship between sets P, P', primarily the similarity. They represent a standard descriptor taking into account the P, P' residuals. Affinity and co-linearity are non-conformal, non-linear models causing a twist, and both are inconvenient.

Uncertainty regions. Uncertainty regions of fixed sizes formed by squares [36], or circles [51] are using as auxiliary descriptors. Matching with uncertainty regions of variable dimensions formed by the Tissot indicatrix depending both on the geographical position (and thus on distortions) and the analyzed projection represents a new approach, given the current state.

As mentioned above, we are trying to determine the subset $S \subset P$ lying inside the uncertainty regions and to compute the similarity ratio μ. The results of the analysis are given by the following parameters: the similarity

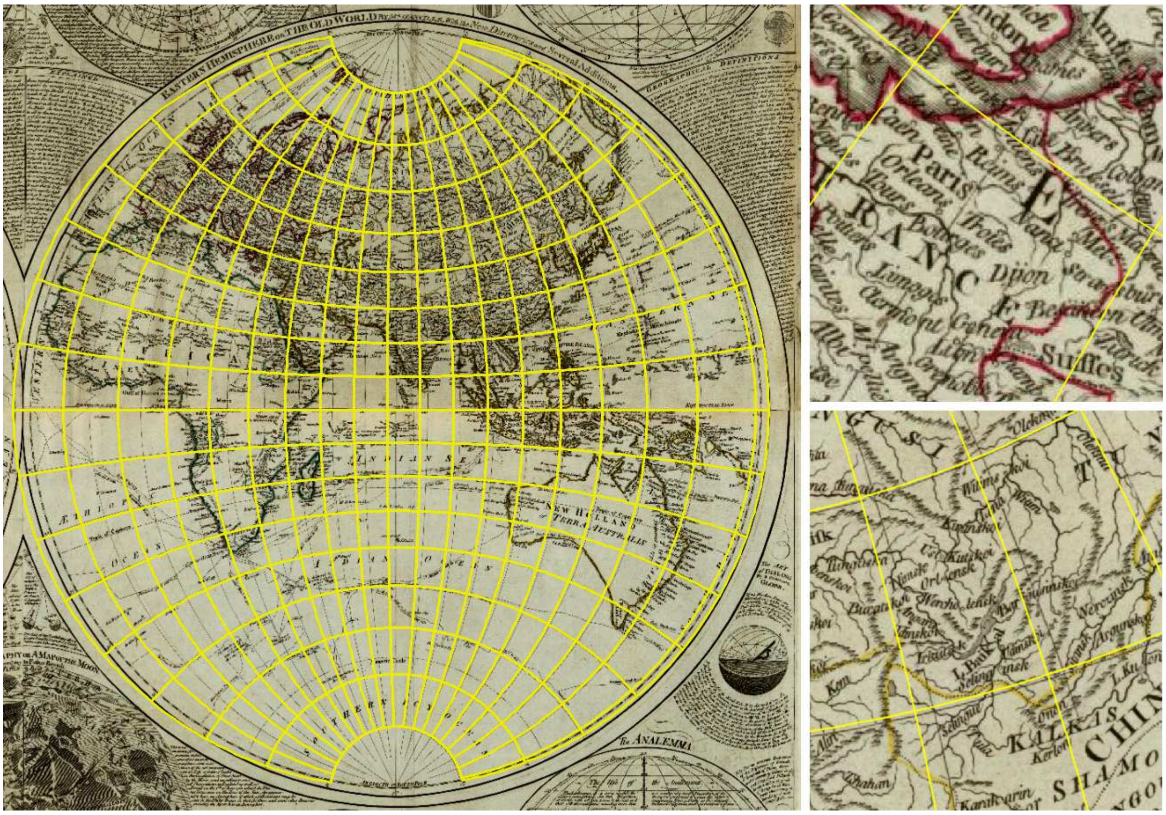

Fig. 6.1: Map 8: a graticule of the stereographic projection (close to the transverse aspect); a reconstruction from the estimated parameters, detail of France and China.

ratio μ and the apriori variance σ_0^2. Unlike heuristics, there is a generalized condition for S

$$S = |T(P) - P'| < \mathbb{P}(\varepsilon).$$

It revisits S as a subset of the transformed points P, which lay inside the modified uncertainty regions represented by ellipses of distortion centered on the P' points projected by $\mathbb{P}$. Hence, a circle of radius ε is transformed into an ellipse $\mathcal{T}$ with parameters a, b, A. This fact is important especially for boundary territories (northern and southern), where extreme local linear scales a, b and maximum angular distortion $\Delta\omega$

$$\Delta\omega = 2\sin^{-1}(|a - b| / (a + b))$$

have large values. For conformal projections, ellipses of distortion are replaced with circles having a rapidly increasing radius $r = a$. A circle with a fixed radius $r = \varepsilon$ is not a good estimator in such territories; see Fig. 5.4.

Apriori variance σ_0^2. An apriori variance σ_0^2, the basic accuracy characteristic computed from the roots of residuals v_i used in the objective function ϕ, is written as follows:

$$\phi(P, P') = \sigma_0^2 = \sum_{i=1}^{n} \frac{v_i v_i}{2n - 4}, \tag{6.1}$$

$$v_i = \sqrt{(X_i - X_i')^2 + (Y_i - Y_i')^2},$$

where X_i', Y_i' are transformed coordinates x_i, y_i. Minimizing this function represents a solution of the least squares problem. A detailed description of these techniques will be presented in the next article. In accordance with our requirements of the rotation invariance or dependency, Helmert or homothetic transformation is used. Let us set P' to be a global system and P to be a local system.

Helmert transformation. The Helmert transformation is given by the key $t(m, \alpha, s_x, s_y)$. It is the first and probably the most common and obvious estimator. A similar approach, used by [23], is available in MapAnalyst software.

In most cases, the Helmert transformation brings very accurate results. Due to the rotation invariance, for some types of input sets, it is not an appropriate descriptor. An incorrect matching of samples with a large value of rotation between P, P' may lead to the detection of a different type of projection. This relatively rare case may occur for specific types of sets, when several negative factors acting simultaneously. Sets located along the equator or prime meridian projected both in normal and transverse aspects have similar shapes in most map projections; however, they are rotated one another.

Examples of such situations are shown in Fig. 5.6. Although, the graticule shapes of the strip, located along the central meridian in the Bonne and equidistant cylindrical projection are analogous, they are mutually rotated. Notice the similar cases relating to the spherical quadrangle, close to the intersection of the equator and the central meridian. Its shape is almost similar across all categories of map projections. Without taking into account the angle of rotation, several projections may be incorrectly detected and interchanged.

Homothetic Transformation. The homothetic transformation is given by the key $t(m, s_x, s_y)$ with one scale m and two shifts s_x, s_y (no rotation is allowed). The transformation model matrix $\mathbf{A}(2n, 3)$ and vector of measurements $\mathbf{l}(2n, 1)$ are

$$\mathbf{A} = \begin{pmatrix} X_1 & 1 & 0 \\ ... & ... & ... \\ X_n & 1 & 0 \\ Y_1 & 0 & 1 \\ .. & .. & .. \\ Y_n & 0 & 1 \end{pmatrix}, \qquad \mathbf{l} = \begin{pmatrix} x_1 \\ ... \\ x_n \\ y_1 \\ ... \\ y_n \end{pmatrix}.$$

It has all the benefits of the Helmert transformation; moreover, it is not invariant to rotation. However, there are some situations when rotation independence is required. These include a secondary rotation of the analyzed map caused by inappropriate insertion into the scanner or by a switched map orientation on the page (portrait vs. landscape). This negatively affects the apriori variance σ_0^2, and a projection is not detectable.

This estimator is very close to the simplified version of the objective function ϕ, which minimizes the residuals between P and P'.

6.1.2 Detection Based on the Nearest Neighbor Distance Methods

The nearest neighbor distance methods are based on an examination of objects located in the close region of the analyzed object. This uses the assumption that there is an influence from an object which depends on its distance to the analyzed object.

Cross nearest neighbor distance. The cross nearest neighbor distance C_{nnd} of data sets P, P' containing n_P and $n_{P'}$ elements, aligned in the centers of mass, is defined as

$$\begin{aligned}\phi(P,P') = C_{nnd}(P,P') &= \frac{1}{n_P + n_{P'}}(\sum_{i=1}^{n_P} d^{(1)}(P_i, P'_j) + \\ &\qquad \sum_{i=1}^{n_{P'}} d^{(1)}(P'_i, P_j)),\end{aligned}$$

where $d^{(1)}$ represents the distance between a point and the point nearest to it. C_{nnd} is dependent on the rotation; its advantages are similar to the homothetic transformation. The cross nearest distance measures a spatial similarity between two sets of points, which is expressed by spatial proximity. It is computed between the nearest points $P_i \to P'_j$ and vice versa $P'_i \to P_j$ (not between a corresponding pair of points $P_i \to P'_i$ for both data sets.

Because of $d^{(1)}(P_i, P'_j) \leq v_i$, $v_i = \|P_i - P'_i\|$, the cross distance of the two elements is never greater than residual v_i. Therefore, the C_{nnd} estimator is slightly more sensitive than apriori variance σ_0.

6.2 Detection Methods Based on the Shape Preservation

They represent a descriptor taking into account the spatial distribution of points, and better reflect the shape of the graticule. Residuals between P, P' do not play such a significant role. Compared to the geometric accuracy approach, the objective function ϕ is more complex, and it is more difficult to minimize. In general, it is possible to use some advantageous properties of the Voronoi diagram.

There are many ways to measure the similarity between two sets using their Voronoi diagrams. These include a direct approach, assessing the features of the Voronoi cells using the simple criteria (area, perimeter), or an indirect approach (entropy statistic); further details can be found in [38]. Due to the lack of appropriate cells, the Voronoi criteria are not suitable for smaller sets.

The proposed indirect approach is based on the computation of the objective function value between auxiliary planar geometric structures constructed over sets P, P'. Sets of which the planar structures are matching, are similar.

In comparison with the transformation methods, the solution has some advantages (better ability to assess the real spatial distribution of points) and disadvantages (lower sensitivity, noise effect, time complexity, degenerative cases).

Voronoi diagrams of the analyzed sets. Let us state only the short and the formal definition of the Voronoi diagram. We call the set of regions $\mathcal{V}(P)$=$\{\mathcal{V}(P_1), ..., \mathcal{V}(P_n)\}$ and $\mathcal{V}(P')$=$\{\mathcal{V}(P'_1), ..., \mathcal{V}(P'_n)\}$

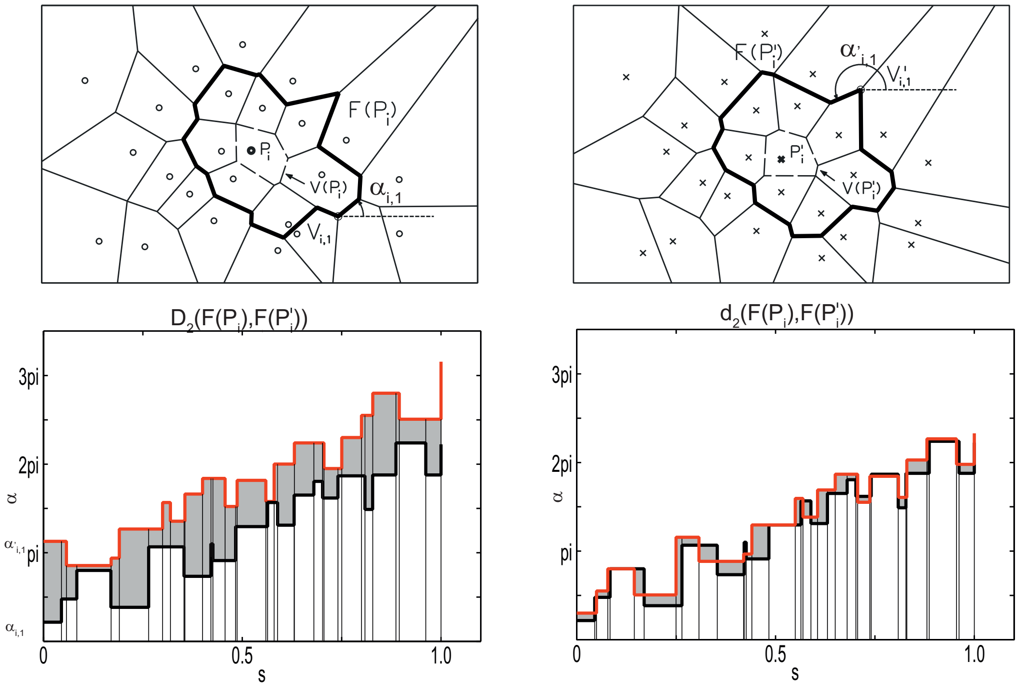

Fig. 6.2: Corresponding merged Voronoi faces $F(P_i)$ (a), $F(P'_i)$ (b) belonging to points P_i and P'_i and their turning functions with broken rectangular strips. The highlighted areas represent $D_2(F(P_i), F(P'_i))$ (c) and $d_2(F(P_i), F(P'_i))$ (d).

the planar Voronoi diagrams generated by P, P'; regions $\mathcal{V}(P_i) = \{X \,\|X - P_i\| \le \|X - P_j\|$ for $j \neq i\}$ and $\mathcal{V}(P'_i) = \{X \,\|X - P'_i\| \le \left\|X - P'_j\right\|$ for $j \neq i\}$ are Voronoi polygons (i.e., Voronoi cells) associated with P_i, P'_i, and $\partial\mathcal{V}(P_i)$, $\partial\mathcal{V}(P'_i)$ are boundaries of Voronoi polygons.

Analyzed structures. The similarity assessment between $\mathcal{V}(P_i)$ and $\mathcal{V}(P'_i)$ represents a key step in the detection process. Due to the fact that a Voronoi diagram is sensitive to the locus of generators, the change of positions of a few of them can cause significant changes in its shape. Presented detection algorithm is based on the following assumption: $\mathcal{V}(P)$ is similar to $\mathcal{V}(P')$ if, and only if, P is similar to P', and vice versa. As mentioned above, only bounded pairs $\mathcal{V}(P_i)$ and $\mathcal{V}(P'_i)$ are subject to further analysis, and bounded cells of an inappropriate shape have been removed from the analysis. If there are not at least three appropriate bounded pairs of Voronoi cells in P, P', $\phi(\mathbf{X}_j)$ is penalized $\phi(\mathbf{X}_j) = \infty$. It is obvious that both P, P' must contain a sufficient amount of points, and analysis should be performed for sets above 15 points. The following planar data structures could be analyzed further:

- *Voronoi diagrams* $\mathcal{V}(P), \mathcal{V}(P')$

  Direct analysis of the corresponding Voronoi cells $\mathcal{V}(P_i) \in \mathcal{V}(P)$, $\mathcal{V}(P'_i) \in \mathcal{V}(P')$ is less sensitive than the second method, but has a better time complexity $O(N^3 \log(N))$ (including $O(N^2)$ analysis).

- *Merged faces* $F(P), F(P')$

  The direct analysis of the corresponding merged Voronoi faces $F(P_i)$, $F(P'_i)$. We consider

$$F(P_i) = \bigcup_{j=1}^{k} [[\mathcal{V}(P_i) \cup \mathcal{V}(P_j)] \cap [\mathcal{V}(P_i) \cap \mathcal{V}(P_j) \neq \emptyset]]$$

  as a face with m_1 vertices formed by all k merged Voronoi faces $\mathcal{V}(P_j)$ adjacent to $\mathcal{V}(P_i)$ merged with $\mathcal{V}(P_i)$, and analogously for $F(P'_i)$. Let $V_i = \{V_{i,1}\}_{i=1}^{m_1}$ and $V'_i = \{V'_{i,1}\}_{i=1}^{m_2}$, be sets of $F(P_i)$ and $F(P'_i)$ vertices. This approach, set as the primary analysis tool, is based on the extended assumption: $F(P)$ is similar to $F(P')$, if and only if, P is similar to P', and vice versa. It is more sensitive and has higher time complexity $O(kN^4 \log^2(N))$ and noise resistance. For the Boolean operation on polygons, a modified algorithm described by [33] was implemented.

Before the analysis, the faces $F(P_i), F(P'_i)$ are rescaled, where each segment is normalized by the face perimeter.

6.3 Measuring similarity using the Turning Function

Without a loss of generality, we can say that a turning function Θ transforms the $j-th$ vertex $V_{i,j} \in F(P_i)$, $j = 1, ..., m$ to the pair of parameters

$$\Theta(V_{i,j}) : V_{i,j} \to \theta_{i,j} = [s_{i,j}, \alpha_{i,j}],$$

where $s_{i,j}$ represents the cumulative length of $V_{i,j}$ measured along the boundary of $\partial F(P_i)$ from the start vertex O, and $\alpha_{i,j}$

$$\alpha_{i,j} = \alpha_{i,j-1} + \pi - (\sigma_{i,j} - \sigma_{i,j-1}),$$

a cumulative angle of $V_{i,j}$, $\sigma_{i,j}$ is the polar angle of the segment $V_{i,j}, V_{i,j+1}$ and the x axis; see Fig. 6.2. According to the requirements, the turning function can be defined as rotation-dependent or rotation-invariant. Unlike the original method described in [1], where no distinction between two states has been realized, and the angle of rotation of both analyzed features was determined as the unknown parameter, we simply change the definition of the descriptor for the first polar angle. For a rotation-dependent case

$$\alpha_{i,1} = \sigma_{i,1},$$

for a rotation-invariant case,

$$\alpha_{i,1} = \pi - (\sigma_{i,1} - \sigma_{i,m}),$$

If the polygon was rescaled and its perimeter length is 1, then $s_{i,1} = 0$, $s_{i,m_1} = 1$, and

$$\alpha_{i,m} = \alpha_{i,1} + 2\pi.$$

Let $\Theta_{F(P_i)}$ and $\Theta_{F(P'_i)}$ be turning functions and θ_i, θ'_i are ordered sets of m_1, m_2 pairs of corresponding faces $F(P_i), F(P'_i)$.

Distance of turning functions. In accordance with [1], let us denote $t, t \in \langle 0, 1 \rangle$ as a shift of the reference point O along the face boundary $F(P'_i)$, written as $\Theta_{F(P'_i)}(s + t)$.

A distance d_2 between $F(P_i), F(P'_i)$ could be determined as the minimum over all possible shifts t

$$d_2(F(P_i), F(P'_i)) = (\min_{t \in \langle 0,1 \rangle}(D_2(F(P_i), F(P'_i))))^{0.5} \quad (6.2)$$

where

$$D_2(F(P_i), F(P'_i)) \quad = \quad \int_0^1 \left|\Theta_{F(P_i)}(s+t) - \ \Theta_{F(P'_i)}(s)\right|^2$$

It is apparent that d_2 represents the sum of the highlighted areas between turning functions; see Fig. 6.2). Hence, in the new algorithm based on dynamic programming, we find t minimizing this equation.

Algorithm description. The proposed algorithm with $O(m_1(m_2 + m_2)$ time complexity solves both rotation-dependent and invariant cases by using dynamic programming. First, the turning functions $\Theta_{F(P_i)}(s)$ and $\Theta_{F(P'_i)}(s)$ for the analyzed faces $F(P_i), F(P'_i)$ are computed. A shift t in $\Theta_{F(P'_i)}(s + t)$ is realized by $m_2 - 1$ cyclic rotations of $\theta'_{i,j}$ elements.

During the cyclic rotation, each vertex becomes a new start vertex; its cumulative length must be set to zero and the turning angle must be corrected. For the rotation dependent case, where the polar angles relate to the x axis, they needn't to be recalculated. The actual shift t is realized by updating $s'_{i,k}, \alpha'_{i,k}$ values. The following steps are to be performed:

1. The actual distance D_2 is computed and compared to D_2^{min}. Then a new minimum is actualized, if necessary.

2. First, the turning angle $\alpha'_1 = \alpha'_{i,1}$ and the distance of the vertex $V_{i,2}$ (next to the start vertex) $s'_2 = s'_{i,2}$ are stored.

3. To achieve the new cumulative length for a vertex $V_{i,j}$, the cumulative length of the second vertex is subtracted from $V_{i,j}$, the successor. All new $s'_{i,j}$, are given by subtracting
$$s'_{i,j} = s'_{i,j+1} - s'_2.$$

4. Analogously, a turning angle is replaced by its successor, $\alpha'_{i,j} = \alpha'_{i+1,j}$.

5. For a rotation-invariant case, all turning angles must be recalculated. For a new starting vertex and its successors, the turning angle α'_1 must be subtracted
$$\alpha'_{i,j} = \alpha'_{i,j} - \alpha'_1.$$

6. Finally, the values of the last pair: $\alpha'_{i,m_2} = \alpha'_{i,1} + 2\pi$ and $s'_{i,m_2} = 1$ are updated.

It is noticeable that D_2 represents the sum of the unsigned area differences between strips (or their parts) belonging to both turning functions. Each turning function has a set of rectangular strips (m_1 strips for $\Theta_{F(P_i)}$ and m_2 strips for $\Theta_{F(P'_i)}$) of various widths and heights. The vertical edges of the strips are formed by the ordinates of their turning function adjacent vertices; the horizontal edges are formed by the x axis and the "graph". It is obvious that only intersecting strips of both turning functions should be processed. It is not necessary to compute the area of each strip. Rather, the area differences of intersected strip parts, are sufficient.

Breaking rectangular strips. Let us do a simple trick and break the strips of both turning functions. All vertical edges (i.e., ordinates) of the first turning function are extended to the graph of the second turning function and vice versa. Old strips are broken into new $m \leq m_1 + m_2$ pairs of strips; see Fig. 6.2. One pair of broken strips shares a lower horizontal edge on the $x-$axis. Broken strips are processed pair by pair in the ascending order according to the s values.

This step is represented by a procedure analogous to two-way merging (see below). The required rectangle representing the result of the difference, is formed by the horizontal edge $s_{i,l}, s_{i,l}^{old}$, vertical edge $\alpha_{i,l}, \alpha'_{i,l}$, and area $A_{i,l}$. The L_2 distance $D_2(F(P_i), F(P'_i))$ corresponding with the sum of areas of $m_1 + m_2$ strip differences, is given by

$$\begin{aligned} D_2(F(P_i), F(P'_i)) &= \sum_{l=1}^{m_1+m_2} A_{i,l}(F(P_i), F(P'_i)) \\ &= \sum_{l=1}^{m_1+m_2} \left(s_{i,l} - s_{i,l}^{old}\right) \left|\alpha_{i,l} - \alpha'_{i,l}\right| \end{aligned} \tag{6.3}$$

The algorithm is very easy, we initialize

$$s_{i,1} = s_{i,1}^{old} = 0,$$

If

$$s'_{i,k} < s_{i,j},$$

then

$$s_{i,l}^{old} = s_{i,l}, \quad s_{i,l} = s'_{i,k}, \quad \alpha_{i,l} = \alpha_{i,j-1} \quad \alpha'_{i,l} = \alpha'_{i,k-1}.$$

Analogously, if

$$s_{i,j} < s'_{i,k},$$

then

$$s_{i,l}^{old} = s_{i,l}, \quad s_{i,l} = s_{i,j}, \quad \alpha_{i,l} = \alpha_{i,j-1} \quad \alpha'_{i,l} = \alpha'_{i,k-1}.$$

Inappropriately rotated faces. How can one avoid the situation in which both faces are lightly rotated in different directions, so their polar angles α, α_{old} belong to the first and fourth quadrants, and vice versa? In such cases, the $\Delta\alpha$ values are greater than if both values α, α_{old} belonged to the first-second, second-third or third-fourth quadrants (compare the $\Delta\alpha$ values for $\alpha = 23/24\pi$, $\alpha_{old} = 1/24\pi$ to $\alpha = 5/24\pi$, $\alpha_{old} = 7/24\pi$). This is important for the rotation-dependent variant, when both turning functions will be classified, incorrectly, as dissimilar. There is a simple heuristic avoidance of this problem, by minimizing the angle difference $\Delta\alpha$ and adding or subtracting the period of 2π

$$\Delta\alpha = \min(\min(|\Delta\alpha|, |\Delta\alpha - 2\pi|), |\Delta\alpha + 2\pi|).$$

Those steps (with reindexing starting from 0) are formally rewritten in Alg. 1, which has $O(m_1 + m_2)$ time complexity. The dynamic approach is analogous to the simplified **two-way merging** procedure used in the merge sort algorithm. The simplification is given by the fact that the last elements of both turning functions are equal

$$s_{i,m_1} = s'_{i,m_2} = 1.$$

The resulting criterion based on the average dissimilarity $\overline{d}_2(F(P_i), F(P'_i))$, measured between P and P', is given by

$$\phi(P, P') = \overline{d}_2(F(P), F(P')) = \frac{1}{n} \sum_{i=1}^{n_f} d_2(F(P_i), F(P'_i)),$$

where n_f represent the amount of the analyzed faces.

Algorithm 1 Compute $D_2(F(P_i), F(P'_i))$ using the merging procedure.

```
function $D_2(\Theta_{F(P)}, \Theta_{F(P')})$
  $s \leftarrow 0; A \leftarrow 0;$
  for $i \leftarrow 2$, $j \leftarrow 1$, $k \leftarrow 0$; $k < m_1 + m_2$; $k \leftarrow k + 1$
        $s_{old} \leftarrow s$; $\alpha \leftarrow \alpha_{i-1}; \alpha' \leftarrow \alpha'_{j-1}$
        if $s_i \leq s_j$ //Find smaller element
            $s \leftarrow s_i$
            $i \leftarrow i + 1$
        else
            $s \leftarrow s'_j$
            $j \leftarrow j + 1;$
        $\Delta\alpha \leftarrow$
$\min(|\alpha - \alpha'|, |\alpha - \alpha' - 2\pi|, |\alpha - \alpha' + 2\pi|)$
        $A \leftarrow A + (s - s_{old})\,\Delta\alpha;$
  return $A$;
```

7 Detection Methods for Line and Polygon Features

As mentioned above, both line and polygonal features allow for the continuous analysis of maps. They represent the best source material for the assessment process. Because the acquisition of 1D and 2D features represents a time-consuming process, such a complex data structure is available only in rare cases. Therefore, the vast majority of analyses is based on point sets. If input data sets contain no line segments, the detected meridians and parallels are analyzed.

Due to its versatility, the turning function was selected as a shape descriptor. An appropriate parametrization for 1D-2D elements is available. The final criterion for assessing detected meridians, parallels

Table 2: *Analyzed maps from David Rumsay Map Collection and estimated parameters for both programs and decreasing amount of analyzed features.*

Map		**detectproj**							**MapAnalyst**					
#	n	**Proj**	φ_q	λ_q	φ_0	λ_0	RES	N_IT	**Proj**	φ_q	λ_q	φ_0	λ_0	RES
1	28	sinu	90.0	0.0	0.0	16.4	1.74^{-3}	85	sinu	90.0	0.0	0.0	16.3	1.74^{-3}
	14	sinu	90.0	0.0	0.0	17.1	6.69^{-4}	83	sinu	90.0	0.0	0.0	17.4	6.71^{-4}
	8	sinu	90.0	0.0	0.0	16.5	4.96^{-4}	102	sinu	90.0	0.0	0.0	17.8	5.18^{-4}
	5	bonne	90.0	0.0	5.0	15.9	2.13^{-4}	97	sinu	90.0	0.0	0.0	20.0	4.89^{-4}
2	28	bonne	90.0	0.0	54.7	20.2	1.67^{-3}	100	lcc	90.0	0.0	40.0	-2.6	7.76^{-1}
	14	bonne	90.0	0.0	58.2	20.1	4.67^{-4}	91	eqdc	90.0	0.0	40.0	42.1	3.79^{-1}
	8	lcc	40.0	152.1	13.4	0.0	5.90^{-5}	263	eqdc	90.0	0.0	40.0	41.0	1.76^{-1}
	5	bonne	90.0	0.0	57.7	20.3	1.91^{-5}	98	lcc	90.0	0.0	40.0	26.7	1.00^{-2}
3	29	nicol	90.0	0.0	0.0	-110.7	2.23^{-3}	122	nicol	90.0	0.0	0.0	-110.5	2.24^{-3}
	15	nicol	90.0	0.0	0.0	-110.3	9.45^{-4}	157	nicol	90.0	0.0	0.0	-110.7	9.52^{-4}
	8	nicol	90.0	0.0	0.0	-110.3	3.74^{-4}	124	nicol	90.0	0.0	0.0	-99.1	4.53^{-4}
	5	nicol	90.0	0.0	0.0	-110.1	1.80^{-4}	123	four	90.0	0.0	0.0	-77.0	not. impl.
4	31	nicol	90.0	0.0	0.0	72.4	3.05^{-3}	310	nicol	90.0	0.0	0.0	70.1	3.83^{-3}
	15	nicol	90.0	0.0	0.0	71.5	9.19^{-4}	294	nicol	90.0	0.0	0.0	70.3	1.21^{-4}
	8	nicol	90.0	0.0	0.0	71.5	6.11^{-4}	500	nicol	90.0	0.0	0.0	69.5	7.81^{-4}
	5	nicol	90.0	0.0	0.0	71.4	4.95^{-4}	255	nicol	90.0	0.0	0.0	69.5	6.16^{-4}
5	28	lcc	52.2	2.9	80.0	0.0	9.95^{-4}	231	lcc	90.0	0.0	40.0	7.7	1.08^{-1}
	14	lcc	90.0	0.0	39.2	2.1	6.04^{-4}	132	lcc	90.0	0.0	40.0	1.2	2.29^{-3}
	8	lcc	90.0	0.0	40.2	2.0	1.19^{-4}	154	lcc	90.0	0.0	40.0	5.6	1.06^{-2}
	5	lcc	90.0	0.0	36.4	1.7	6.95^{-5}	107	lcc	90.0	0.0	40.0	3.2	9.36^{-4}
6	34	eqdc2	90.0	0.0	60.9	10.8	1.89^{-3}	102	lcc	90.0	0.0	40.0	9.4	9.4^{-3}
	17	eqdc2	90.0	0.0	61.3	10.8	8.34^{-4}	106	eqdc	90.0	0.0	40.0	13.2	1.03^{-2}
	9	eqdc2	90.0	0.0	61.1	10.8	5.94^{-4}	94	eqdc	90.0	0.0	40.0	10.2	2.56^{-3}
	5	eqdc2	90.0	0.0	60.7	10.7	3.76^{-4}	89	eqdc	90.0	0.0	40.0	9.7	9.43^{-3}
7	30	eqdc2	90.0	0.0	57.0	-2.9	8.73^{-4}	112	eqdc	90.0	0.0	40.0	-9.2	2.72^{-2}
	15	eqdc2	90.0	0.0	55.3	-3.0	4.06^{-4}	98	sinu	90.0	0.0	0.0	-3.1	5.90^{-4}
	8	eqdc2	90.0	0.0	53.0	-3.2	1.02^{-4}	90	sinu	90.0	0.0	0.0	-3.6	2.51^{-4}
	5	leac	30.0	4.5	69.0	0.0	5.88^{-5}	170	sinu	90.0	0.0	0.0	-3.1	1.07^{-4}
8	29	stereo	-1.0	73.1	0.0	0.0	8.19^{-3}	119	nicol	90.0	0.0	0.0	70.3	9.61^{-3}
	14	stereo	-1.9	73.4	0.0	0.0	1.38^{-3}	109	nicol	90.0	0.0	0.0	70.1	2.69^{-3}
	8	stereo	-0.9	72.2	0.0	0.0	1.30^{-4}	144	nicol	90.0	0.0	0.0	71.8	1.71^{-3}
	5	stereo	-1.4	72.6	0.0	0.0	8.02^{-6}	89	nicol	90.0	0.0	0.0	9.0	1.08^{-2}

and all line segments

$$\begin{aligned}\phi(P,P') = \overline{d_2}(P,P') \quad = \quad & \frac{\sum_{i=1}^{n_{mer}} d_2^{mer}(P,P')}{n_{mer}} + \\ & \frac{\sum_{i=1}^{n_{par}} d_2^{par}(P,P')}{n_{par}} + \\ & \frac{\sum_{i=1}^{n_{lin}} d_2^{lin}(P,P')}{n_{lin}}\end{aligned}$$

corresponds to Eq. 5.7, where n represents the number of analyzed polygons.

8 Experiments and Results

The proposed method was tested extensively; the results were compared to MapAnalyst software, version 1.4 pre-release, which has a similar functionality. We bring results for 8 early maps created since the 18th century and published in the David Rumsay Map Collection. Maps 1-3, 4, and 8 are world atlas maps, and maps 2, 5, 6, and 7 are national maps. Map projections were apriori unknown. Recall that it is not recommended to select the analyzed area around the prime meridian, true parallel, equator, or poles, where most map projections have similar properties. For all the tests, the same values of parameters have been set. Analyses were performed for all aspects (normal, transverse, oblique) with the disabled heuristics.

8.1 Test of early maps

The following characteristics were measured: RES (sum of residuals), N_IT (number of iterations). However, the second criterion cannot be acquired for MapAnalyst. The analyzed features, whose amount decreased ap-

Table 3: *Results of 300 tests with a random initialization for maps 1-8 and three sizes of intervals ψ.*

Map	ψ	EFF	N_IT	RES_C	RES_A	TIME
1	1	100	29765	4.91^{-1}	4.91^{-1}	425
	0.1	100	26020	4.91^{-1}	4.91^{-1}	432
	0.01	98	31419	4.80^{-2}	9.05^{2}	535
2	1	99	46728	3.42^{-1}	4.67^{-1}	628
	0.1	97	50837	3.39^{-1}	1.34^{1}	554
	0.01	97	48089	3.40^{-1}	1.60^{1}	539
3	1	98	43914	6.78^{-1}	1.07^{3}	1869
	0.1	82	49509	5.67^{-1}	4.11^{3}	1325
	0.01	74	50934	5.01^{-1}	4.17^{2}	1648
4	1	100	39828	6.75^{-1}	6.75^{-1}	2682
	0.1	100	34896	6.75^{-1}	6.75^{-1}	2316
	0.01	96	36415	6.48^{-1}	9.54^{2}	2309
5	1	97	55362	4.30^{-1}	5.24^{-1}	916
	0.1	94	59931	4.22^{-1}	5.99^{-1}	619
	0.01	87	7353	4.60^{-1}	6.79^{1}	280
6	1	100	52092	6.13^{-1}	6.13^{-1}	1009
	0.1	96	51145	6.53^{-1}	9.09^{0}	592
	0.01	88	53621	8.70^{-1}	1.07^{2}	683
7	1	100	49071	2.97^{-1}	3.88^{-1}	570
	0.1	97	47555	3.13^{-1}	1.22^{0}	421
	0.01	87	52330	3.17^{-1}	1.33^{0}	501
8	1	100	43647	6.04^{-1}	6.04^{-1}	576
	0.1	96	39826	5.76^{-1}	2.19^{2}	121
	0.01	93	42719	5.59^{-1}	3.00^{1}	445

proximately 50 percent in each step, were chosen in accordance with the above-mentioned principles.

The identical points both on the analyzed and Open Street maps (OSM) have been collected in MapAnalyst. First, the Cartesian coordinates x, y of identical points in OSM were reprojected from the Mercator projection (WGS-84, $\varphi_0 = 0°$) to the geographical coordinates φ, λ using the inverse formulas. For λ, the following formula was used

$$\lambda = x/a,$$

where a represents the semi-major axis of the WGS-84 ellipsoid (analogously b the semi-minor axis). Then

$$\begin{aligned} e2 &= 1 - b^2/a^2, \\ t &= e^{-y/a}, \\ \chi &= \pi/2 - 2\arctan t \end{aligned}$$

and

$$\varphi \doteq \chi + J_1 \sin 2\chi + J_2 \sin 4\chi + J_3 \sin 6\chi + J_4 \sin 8\chi,$$

where

$$\begin{aligned} J_1 &= 2e2 + 5e2^2/24 + e2^3/12 + 13e2^4/360, \\ J_2 &= 7e2^2/48 + 29e2^3/240 + 811e2^4/11520, \\ J_3 &= 7e2^3/120 + 81e2^4/1120, \\ J_4 &= 4279e2^4/161280. \end{aligned}$$

Further details can be found in [46]. All algorithms have been implemented in C++ (VS 2010 compiler); our testing PC had the following hardware specification: Intel Core E4500 processor, 2.2 GHz with 2GB RAM running on Win7 32-bit Professional.

The following 8 maps from David Rumsey Map Collection were analyzed, see Tab. 2:

Map 1. "Carte generale de l'Afrique, Adrien Hubert, Atlas universel de geographie physique, politique, ancienne & moderne", 1828, 28 identical points. Estimated projection: sinusoidal, $\varphi_k = 90°N$, $\lambda_k = 0°$E, $\varphi_0 = 0°N$, $\lambda_0 = 16.4°$E. The estimated results are almost independent of the amount of analyzed points. Both programs achieved similar results, but `detectproj` has slightly smaller residuals.

Map 2. "Europe Politique", Atlas St. Cyr. Furne, Jouvet et Cie, Paris, 1885, 28 identical points. Estimated projection: Bonne, $\varphi_k = 90°N$, $\lambda_k = 0°E$, $\varphi_0 = 57.7°N$, $\lambda_0 = 20.2°E$. The presented solution brings much better results (more reliable parameters); the residuals are significantly lower. Because the area covered by the map in the longitude direction is $\lambda \in \langle 15°W, 45°E\rangle$, the central meridian $\lambda_0 = 42.14°E$ determined by MapAnalyst looks like a mistake.

Map 3. "Western Hemisphere", Faden W., General Atlas, 1811, 29 identical points. Estimated projection: Nicolosi, $\varphi_k = 90°N$, $\lambda_k = 0°E$, $\varphi_0 = 0°N$, $\lambda_0 = 110.7°W$. Somewhat more difficult test due to the confusion with an azimuthal projection in the transverse aspect. Both programs were successful; the results are comparable.

Map 4. "Eastern Hemisphere", Faden W., General Atlas, 1811, 31 identical points. Estimated projection: $\varphi_k = 90°N$, $\lambda_k = 0°E$, $\varphi_0 = 0°N$, $\lambda_0 = 72.4°$. The same situation and similar results.

Map 5. "Composite of Carte de France, Levee par ordre du Roy", Cassini, Cesar-Francois, 1750, 28 identical points. Estimated projection: Lambert conformal conic, $\varphi_k = 90°N$, $\lambda_k = 0°N$, $\varphi_0 = 39.2°N$, $\lambda_0 = 2.1°E$. The large-scale map (one sheet) projection, which covers a small territory $\Delta\varphi = 4°$, $\Delta\lambda = 7°$, was surprisingly not detected as the Cassini-Soldner. The presented solution has slightly smaller residuals compared to MapAnalyst.

Map 6. "Denmark, Sleswig, Holstein, with Iceland, with Bornholm". London atlas series, Stanford's Geographical, 1901, 34 identical points. Estimated projection: equidistant conic, North Pole projected as point, $\varphi_k = 90°N$, $\lambda_k = 0°E$, $\varphi_0 = 61.3°N$, $\lambda_0 = 10.8°E$. Both results are comparable. The graticule reconstructed from the determined parameters can be found in Fig. 5.5.

Map 7 "British Islands", World Atlas, A. Arrowsmith, 1817, David Rumsay Map Collection, 30 identical points. Estimated projection: equidistant conic, North Pole projected as point, $\varphi_k = 90°N$, $\lambda_k = 0°E$, $\varphi_0 = 57.0°N$, $\lambda_0 = 2.9°E$. The presented solution brings lower residuals and better resistance to the amount of analyzed points.

Map 8 "A general map of the world or terraqueous globe", S. Dunn, 1787. David Rumsay Map Collection, 29 identical points. Estimated projection: stereographic projection, $\varphi_k = 1.0°S$, $\lambda_k = 73.1°E$, $\varphi_0 = 0°N$, $\lambda_0 = 0°E$. A difficult test; the analyzed map was probably created in a stereographic projection (transverse aspect). Its graticule is similar to the globular projections. Unlike MapAnalyst, the developed software detects the oblique aspect very close to the transverse aspect. Here, the lack of ability of MapAnalyst to determine anything other than the normal aspect appeared. However, the solution found by MapAnalyst is not strictly imprecise, but has a larger residuals. The graticule reconstructed from determined parameters can be found in Fig. 6.1. While the parallels fit well, the meridians are shifted (prime meridian centered on Paris).

The proposed algorithm achieved better results in all cases; the estimated parameters are more natural and have better objective function value, which decreases with the amount analyzed features. In all test, the global minimum of the objective function ϕ have been found.

However, only the normal aspect of the projection is detected by MapAnalyst. Nonetheless, it supports more map projections (compromise, unclassified projections), and the analysis is much faster (about 10% of `detectproj` time). Recall that both programs are still in development.

The results also indicate that for a reliable analysis, a set of approximately 10 features is required. The analysis of 5 features is vague and we can determine a category, rather than a specific projection. Instead of an oblique projection aspect (used primarily for larger scale maps), for atlas maps, both normal and transverse aspects are more common.

8.2 Test of robustness

The second test measures both the robustness and efficiency of the Nelder-Mead algorithm, depending on the simplex size and position under all maps. The new simplex intervals with the size $\psi = 0.1$ and $\psi = 0.01$ of the original interval were set randomly as follows:

$$\begin{aligned}(\varphi_k, \lambda_k, \varphi_0, \lambda_0)'_{min} &= (\varphi_k, \lambda_k, \varphi_0, \lambda_0)_{min} + rnd(0,1) \cdot \\ &\quad (1-\psi)(\Delta\varphi_k, \Delta\lambda_k, \Delta\varphi_0, \Delta\lambda_0), \\ (\varphi_k, \lambda_k, \varphi_0, \lambda_0)'_{max} &= (\varphi_k, \lambda_k, \varphi_0, \lambda_0)'_{min} + rnd(0,1) \cdot \\ &\quad \psi(\Delta\varphi_k, \Delta\lambda_k, \Delta\varphi_0, \Delta\lambda_0).\end{aligned}$$

The test was repeated 300 times for $n = 15$, where the following characteristics have have been measured: global minimum efficiency (EFF) in $\%$, number of iterations (N_IT), the objective function value ϕ of correctly detected samples given by residuals (RES_C), the objective function value ϕ of all tests (RES_A) and the time (TIME) in sec.; see Tab. *3*. A test is successful if $\phi < 3\overline{\phi}$, where $\overline{\phi}$ represents the objective function value determined with the Mathematica optimizer (differential evolution algorithm). The arithmetic expression parser was enabled; the analyzed projection equations were loaded from a txt file. This allows the user to enlarge the amount of analyzed map projections, but also increases the computational times significantly (approximately 10x).

The Nelder-Mead algorithm efficiency depends on the simplex size, but not significantly. Minimum efficiency occurs, when the estimated $\widehat{\mathbf{X}}$ is far outside the initial simplex and is unable to expand (map 3:λ_0, map 7: φ_0). In such cases, the iteration process may become stuck in a local minimum. We do not recommend reducing the simplex size and initializing the simplex over the whole domain φ, λ.

The average amount of iterations per sample is approximately 145; and the average time per sample 3s

Table 4: *A test of the detection efficiency on the geographical position, additional errors, and the map scale; the early map set is contaminated by additional errors (test T1).*

φ_c	S	ε[mm]				
		1.0	**2.0**	**3.0**	**4.0**	**5.0**
0	1	**100**/**100**/22/1/0/**69**	**100**/**88**/26/0/0/37	**95**/**68**/18/0/0/18	**89**/**54**/2/0/0/11	**70**/36/6/0/0/6
	2	27/5/0/0/0/0	20/2/0/0/0/0	13/0/0/0/0/0	10/1/0/0/0/0	4/0/0/0/0/0
	3	15/0/0/0/0/0	10/0/0/0/0/0	5/0/0/0/0/2	7/3/0/0/0/1	3/3/0/0/0/3
	4	2/0/0/0/0/1	3/1/0/1/0/1	1/1/0/4/0/0	1/1/4/1/0/1	2/2/2/1/0/0
	5	1/1/2/1/0/0	3/0/4/3/0/0	2/1/0/2/0/1	2/0/0/0/0/1	1/2/0/1/0/0
	6	0/1/2/1/0/0	0/1/0/0/0/1	0/0/0/0/0/0	0/1/2/0/0/1	1/0/0/1/0/0
10	1	**100**/**100**/26/3/0/62	**100**/**91**/18/3/0/36	**96**/**81**/14/1/0/24	**90**/**63**/6/2/0/18	**86**/44/6/1/0/15
	2	**93**/**64**/6/22/0/14	**61**/30/0/15/0/6	46/16/0/11/0/4	25/13/0/10/0/3	28/9/0/7/0/3
	3	**71**/33/14/13/0/9	32/16/0/7/0/8	20/12/0/9/0/5	19/6/0/7/0/2	16/4/0/4/0/0
	4	24/9/2/0/0/5	14/7/0/1/0/3	11/4/2/4/0/5	8/2/0/1/0/4	6/6/0/1/0/6
	5	20/6/2/1/0/5	15/6/2/0/0/7	13/4/2/0/0/5	11/2/0/1/0/2	8/5/0/0/0/2
	6	13/5/0/0/0/3	7/5/6/0/0/4	5/1/0/0/0/3	11/3/0/0/0/2	11/6/0/0/0/3
20	1	**100**/**100**/32/6/0/**83**	**100**/**100**/26/4/0/**55**	**100**/**79**/10/1/0/31	**93**/**69**/10/1/0/18	**84**/**55**/6/3/0/24
	2	**100**/**94**/40/14/0/**52**	**79**/**63**/8/17/0/23	**63**/43/2/11/0/19	**53**/28/2/17/0/10	44/29/4/9/0/10
	3	**89**/**76**/16/17/0/25	**67**/35/12/7/0/13	44/21/8/12/0/7	38/26/8/10/0/8	30/15/0/9/0/5
	4	46/29/10/6/0/6	26/20/6/1/0/6	12/11/4/4/0/3	15/11/8/1/0/1	14/12/0/1/0/5
	5	27/17/4/2/0/8	21/10/6/1/0/10	11/7/0/1/0/3	6/5/6/3/0/5	16/6/2/2/0/4
	6	13/7/0/0/0/6	10/9/2/0/0/4	10/8/2/0/0/2	14/8/0/0/0/4	7/7/4/0/0/0
30	1	**100**/**100**/**50**/19/1/**77**	**100**/**99**/34/6/0/**64**	**98**/**92**/24/9/0/39	**93**/**67**/18/0/0/21	**83**/**60**/8/6/0/24
	2	**100**/**95**/**58**/2/0/**55**	**82**/**61**/28/0/0/37	**62**/39/8/0/0/15	**63**/28/2/0/0/8	43/22/2/0/0/7
	3	**85**/**75**/34/0/0/39	**60**/44/20/0/0/17	47/33/6/0/0/10	35/16/6/0/0/6	40/29/6/22/0/8
	4	**54**/29/8/6/0/6	39/19/2/7/0/6	26/11/8/13/0/4	21/11/8/7/0/5	19/13/2/3/0/6
	5	36/20/20/3/0/7	25/10/4/3/0/7	16/10/2/6/0/6	13/11/6/1/0/5	10/9/0/5/0/6
	6	13/10/2/1/0/2	11/9/2/0/0/6	18/6/2/1/0/4	17/5/8/1/0/6	10/1/2/0/0/4
40	1	**100**/**100**/**61**/29/4/**78**	**100**/**99**/45/24/3/**67**	**97**/**93**/32/18/1/**58**	**97**/**84**/24/20/0/44	**75**/**71**/24/21/0/34
	2	**99**/**84**/42/2/0/50	**82**/**58**/20/0/0/21	**62**/42/14/0/0/19	**52**/35/16/0/0/8	**51**/31/8/0/0/8
	3	**87**/**59**/20/9/0/38	**66**/**52**/24/16/0/19	**59**/40/10/14/0/13	**50**/30/6/15/0/12	**51**/22/4/13/0/9
	4	**52**/40/18/15/0/12	42/22/10/13/0/7	29/20/6/15/0/6	21/7/0/8/0/8	16/13/0/5/0/4
	5	43/19/2/4/0/6	27/7/12/8/0/5	21/15/2/6/0/6	20/13/6/6/0/3	15/10/8/8/0/9
	6	10/7/6/2/0/3	17/3/0/0/0/4	12/4/2/2/0/5	8/4/4/1/0/1	17/4/6/1/0/4
50	1	-/99/65/27/2/75	-/98/53/26/2/63	-/92/46/17/1/56	-/81/28/14/0/48	-/69/26/8/0/39
	2	95/83/58/1/0/53	78/64/30/0/0/38	76/54/20/0/0/18	72/37/18/0/0/14	64/35/10/0/0/14
	3	77/64/28/0/0/32	74/56/18/0/0/27	68/32/12/0/0/16	59/22/16/0/0/9	42/17/4/0/0/7
	4	60/34/12/0/0/13	28/18/10/0/0/1	28/16/10/0/0/4	27/10/6/0/0/5	18/14/6/12/0/5
	5	36/18/10/8/0/5	24/15/10/5/0/8	12/12/2/9/0/11	16/4/12/8/0/7	11/14/4/13/0/6
	6	7/13/4/1/0/5	14/4/6/2/0/2	20/5/8/3/0/9	17/6/2/2/0/2	12/10/4/1/0/5
60	1	-/-/52/2/0/10	-/-/54/0/0/1	-/-/47/1/0/4	-/-/39/1/0/2	-/-/35/2/0/5
	2	**83**/**69**/**50**/0/0/41	**71**/**58**/27/0/0/30	**66**/49/12/0/0/18	**67**/**54**/15/0/0/16	**62**/33/17/0/0/11
	3	**72**/**62**/41/12/0/38	**70**/**54**/24/17/0/23	**68**/30/18/12/0/19	**61**/31/4/14/0/13	**57**/36/11/18/0/13
	4	**55**/41/17/13/0/15	49/17/13/15/0/5	43/17/11/13/0/6	28/14/11/9/0/6	20/14/8/11/0/12
	5	38/23/10/9/0/7	32/16/13/15/0/3	22/10/10/8/0/8	15/12/4/8/0/10	17/6/9/4/0/4
	6	25/9/5/0/0/6	10/9/9/0/0/2	11/7/10/0/0/2	15/7/6/3/0/3	13/11/4/4/0/6
70	1	-/-/36/0/0/0	-/-/32/0/0/0	-/-/32/0/0/0	-/-/28/0/0/0	-/-/23/0/0/0
	2	**72**/**69**/**50**/0/0/39	**71**/**53**/36/0/0/21	**61**/40/36/0/0/14	**65**/**54**/12/0/0/15	**65**/34/14/0/0/11
	3	**65**/**64**/**52**/12/0/23	**63**/**72**/36/11/0/24	**61**/**53**/20/14/0/10	**66**/**51**/26/13/0/16	**63**/37/6/14/0/17
	4	**64**/**51**/18/16/0/8	48/24/18/19/0/10	25/20/12/21/0/9	28/25/2/14/0/11	17/22/6/17/0/10
	5	38/18/22/14/0/8	26/18/14/17/0/11	28/21/10/18/0/10	10/17/2/12/0/7	15/10/6/8/0/9
	6	16/10/12/4/0/12	20/8/6/1/0/8	10/16/0/1/0/2	15/13/0/3/0/3	11/7/0/2/0/6

Table 5: *A test of the detection efficiency on the geographical position, additional errors, and the map scale; both maps are contaminated by additional errors (test T2).*

φ_c	S	ε[mm]				
		1.0	**2.0**	**3.0**	**4.0**	**5.0**
0	1	**100**/**99**/36/3/0/67	**99**/**90**/18/2/0/30	**90**/**67**/15/0/0/12	**69**/**56**/10/0/0/8	**60**/46/6/0/0/10
	2	31/4/0/0/0/1	18/3/0/0/0/0	6/0/0/0/0/0	13/8/1/0/0/2	6/4/2/0/0/2
	3	6/5/4/0/0/0	5/2/4/0/0/1	6/4/2/0/0/3	5/3/2/0/0/1	2/1/1/1/0/2
	4	4/3/0/0/0/1	2/1/1/1/0/0	3/0/3/2/0/1	0/1/1/1/0/1	0/0/2/1/0/1
	5	3/4/2/3/0/4	0/2/2/0/0/1	3/1/1/1/0/0	1/2/1/0/0/1	1/0/0/0/0/1
	6	1/1/0/1/0/0	0/0/2/0/0/0	1/0/1/0/0/0	0/0/1/0/0/0	0/0/0/0/0/0
10	1	**100**/**100**/35/6/0/**84**	**98**/90/22/4/0/45	**87**/**76**/15/1/0/23	**70**/**50**/10/0/0/10	**58**/**50**/6/2/0/10
	2	**78**/**72**/16/13/0/15	48/35/9/13/0/6	36/23/4/9/0/3	21/10/2/17/0/5	22/11/5/14/0/5
	3	**50**/38/8/14/0/11	26/19/11/17/0/8	20/14/3/11/0/3	16/14/4/8/0/6	9/3/5/8/0/1
	4	15/16/10/4/0/7	12/13/5/3/0/2	7/4/5/1/0/7	11/9/7/2/0/5	8/3/4/0/0/3
	5	13/11/10/0/0/3	13/7/5/0/0/4	12/6/6/1/0/4	5/6/8/1/0/5	9/5/6/1/0/7
	6	12/4/7/0/0/6	11/5/7/0/0/6	11/8/8/1/0/6	12/6/11/0/0/4	8/7/8/0/0/2
20	1	**100**/**99**/41/4/0/**79**	**96**/**97**/26/4/1/48	**83**/**83**/18/6/0/39	**74**/**73**/9/4/0/19	**64**/**51**/3/2/0/16
	2	**96**/**93**/46/15/0/40	**72**/**72**/17/12/0/25	**54**/41/16/15/0/10	39/25/15/15/0/11	29/21/5/8/0/4
	3	**73**/**79**/25/11/0/24	**52**/39/18/9/0/14	32/32/13/7/0/7	26/20/16/6/0/8	17/26/12/8/0/8
	4	35/27/20/4/0/13	19/19/8/4/0/9	20/7/5/2/0/7	7/11/11/1/0/2	16/11/10/2/0/5
	5	22/14/16/1/0/5	17/20/7/0/0/4	13/7/9/2/0/7	13/16/8/3/0/8	9/3/5/2/0/11
	6	18/14/6/2/0/8	9/6/8/1/0/0	11/13/10/0/0/10	8/10/14/0/0/4	9/8/7/0/0/4
30	1	**100**/**100**/**59**/13/0/88	**98**/**94**/29/9/0/61	**85**/**83**/20/8/0/41	**77**/**72**/10/3/0/24	**62**/**54**/11/0/0/22
	2	**95**/**94**/**58**/0/0/**54**	**65**/**66**/17/0/0/24	46/39/12/0/0/16	47/28/8/0/0/14	32/38/11/17/0/13
	3	**77**/**81**/40/12/0/28	**60**/**59**/26/11/0/19	42/43/16/16/0/12	37/29/16/16/0/5	28/19/13/17/0/12
	4	48/32/20/8/0/7	29/21/10/9/0/5	18/9/9/4/0/8	15/12/8/4/0/7	13/9/9/6/0/11
	5	26/19/11/1/0/12	23/13/9/3/0/12	12/13/11/2/0/8	9/11/6/6/0/13	5/18/9/2/0/13
	6	14/8/9/1/0/6	11/10/11/0/0/8	9/14/6/1/0/5	11/14/6/0/0/7	9/11/5/1/0/9
40	1	**100**/**100**/**55**/25/0/81	**97**/**97**/48/24/1/**66**	**95**/**91**/32/23/1/**52**	**78**/**84**/29/15/0/47	**65**/**66**/21/13/0/32
	2	**89**/**80**/**51**/1/0/42	**66**/**61**/23/0/0/22	**51**/44/18/0/0/23	46/41/10/0/0/13	35/20/9/0/0/14
	3	**70**/**65**/41/13/0/40	**62/52**/21/10/0/23	49/43/19/12/0/16	43/42/12/17/0/11	39/30/9/14/0/12
	4	**51**/34/22/8/0/14	29/23/16/10/0/8	29/11/17/11/0/6	19/15/11/15/0/11	16/10/5/5/0/12
	5	33/19/15/8/0/8	19/9/15/8/0/10	11/11/13/7/0/9	14/10/6/7/0/3	9/20/10/8/0/14
	6	10/16/13/1/0/6	15/13/12/0/0/6	13/13/14/1/0/11	12/13/8/3/0/2	15/10/12/1/0/6
50	1	-/**100**/**57**/38/0/71	-/**100**/**56**/20/0/68	-/**93**/36/24/0/**61**	-/**87**/27/23/0/**55**	-/**59**/30/14/0/46
	2	**84**/**85**/51/1/0/44	**72**/**67**/26/0/0/36	**73**/**53**/28/0/0/20	**52**/37/14/0/0/21	**56**/34/6/0/0/17
	3	**69**/**69**/**53**/0/0/33	**65**/**58**/23/0/0/19	**53**/42/10/0/0/13	45/31/5/0/0/10	38/28/15/13/0/9
	4	**58**/35/17/17/0/9	31/25/16/11/0/11	19/15/7/12/0/8	21/15/15/11/0/7	19/8/11/16/0/5
	5	36/31/19/9/0/13	13/10/10/8/0/12	18/12/12/11/0/12	13/6/10/5/0/12	16/10/16/7/0/9
	6	13/13/10/0/0/8	12/11/10/0/0/5	8/13/10/1/0/9	14/9/13/0/0/3	10/13/9/1/0/8
60	1	-/-/**58**/2/1/9	-/-/**61**/1/1/5	-/-/**39**/1/0/4	-/-/38/0/0/7	-/-/20/0/0/6
	2	**87**/**72**/45/0/0/42	**76**/**64**/25/0/0/32	**60**/**55**/19/0/0/28	**70**/**57**/14/0/0/18	**68**/28/8/0/0/13
	3	**69**/**65**/45/6/0/29	**57**/**53**/19/9/0/25	**65**/**50**/16/13/0/19	**66**/38/24/14/0/13	48/38/24/14/0/10
	4	49/25/25/12/0/16	39/22/13/9/0/4	29/17/15/13/0/9	21/17/10/17/0/8	20/18/12/14/0/6
	5	34/16/16/11/0/7	24/20/8/11/0/9	19/19/16/12/0/16	15/17/11/7/0/9	16/11/8/9/0/10
	6	11/2/5/2/0/2	14/14/16/0/0/2	15/13/13/1/0/9	11/12/11/0/0/6	6/11/11/2/0/12
70	1	-/-/37/0/0/1	-/-/31/0/0/0	-/-/27/0/0/0	-/-/25/0/0/0	-/-/27/0/0/0
	2	**66**/**73**/43/0/0/33	**63**/**58**/30/0/0/28	**60**/46/22/0/0/16	**59**/44/20/0/0/12	**56**/42/15/0/0/12
	3	**62**/**64**/43/9/0/31	**64**/**61**/21/9/0/18	**65**/48/32/19/0/19	**54**/**50**/18/17/0/18	42/45/18/20/0/19
	4	**65**/**50**/22/18/0/19	34/35/19/20/0/10	35/32/17/10/0/17	25/21/14/17/0/20	25/14/22/8/0/12
	5	42/28/20/15/0/12	15/23/18/6/0/7	27/16/15/12/0/10	19/24/8/7/0/7	10/13/15/7/0/12
	6	13/16/13/1/0/11	16/10/13/1/0/3	11/16/13/1/0/8	14/13/16/4/0/3	8/12/10/6/0/5

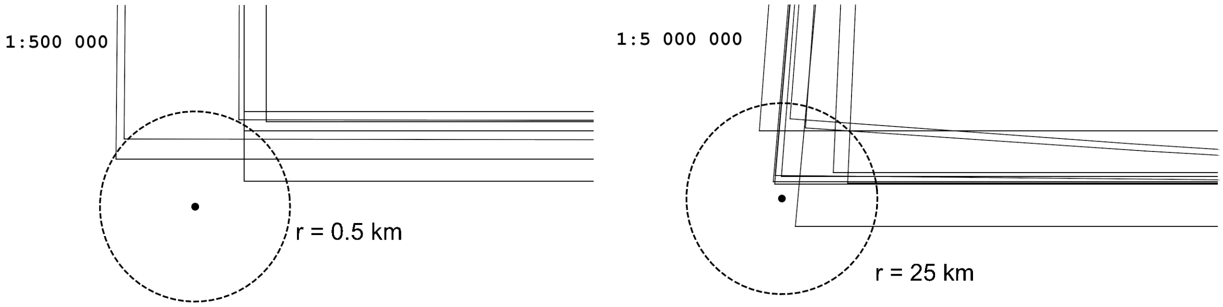

Fig. 8.1: The spatial variability of the map sheets corners for the 10 best determined projections; maps of scales 1:500,000 (a) and 1:5,000,000 (b), $\varphi_c = 40°$, $\varepsilon = 1$ mm.

(with a parser), 0.4s (without a parser), the overall efficiency for the largest simplex is higher than 97%. The proposed algorithm can be considered as efficient and robust to the initial simplex guess.

8.3 Test of the variability of features

To illustrate the properties and behavior of detection algorithms over of the variability of the user entered point features, the following characteristics

- impact of errors on one set of the input features P (test T1),
- impact of errors on both sets of the input features P, Q (test T2),

have been measured for six different sets of points: *grid, random distribution, cluster, random meridian, random parallel, circle*. The T1 test simulates the error-free reference map, the T2 test simulates both maps with errors. All the testing sets have been proportionally decreased from the map covering the whole world to the large-scale map, and concurrently moved around the central meridian of the map. Therefore, the dependence of the efficiency on several factors from Sec. 4, was investigated. Let us take a closer look at the methodology of both *synthetic tests*.

For simplicity, only one projection (sinusoidal) has been involved in the testing. Due to the similarity of the projection graticule to other pseudo-cylindrical projections, it seems an appropriate choice. A set of 0D features has been generated, where $n = 20$.

The distribution of input features. The six types of sets, representing the typical input data acquired from different maps, were involved in the tests. Hence, their simple geometric simulation has been proposed. Analyzed graticule points are expressed by the grid. The random set implies a case, when a cartographer may choose any element on the map. Due to the lack of the analyzed features, when not all parts of the map sheet are available for the analysis, the clustered data is used (4 clusters by 5 points). The radius of the cluster was set to

$$r = \frac{1}{10}\max(\varphi_{max} - \varphi_{min}, \lambda_{max} - \lambda_{min}).$$

A typical example are territories covered by the sea, where numerous islands are present. Portolan charts, where the coastline and ports are drawn, but the interior of the continents is missing, are simulated by points placed on the circle of the radius

$$r = \frac{2}{5}\max(\varphi_{max} - \varphi_{min}, \lambda_{max} - \lambda_{min}).$$

Wide or narrow areas are represented by points on the meridian or parallel, randomly generated.

Dimensions of territory and map scale. Our testing starts from the optimistic case, when all testing features are distributed over the hemisphere, and continues with a proportionally decreasing set, both latitudinally and longitudinally. We set a simplified dependence of the projected territory size and the map scale S, given by the condition that the Earth of the perimeter $d \doteq 40,000$ km fits on the A3 format paper. This requirement corresponds to the approximate scale $S = 1 : 100,000,000$ for the world map, and half of the value for the hemisphere. The following scale series

$$S = \{50,000,000; 40,000,000; 30,000,000;\\ 20,000,000; 10,000,000; 5,000,000;\\ 1,000,000; 500,000; 100,000\}$$

involving the range from the small-scale to the large-scale maps, was tested. In Tabs. 4 and 5, only the scales 50,000,000; 10,000,000; 5,000,000; 1,000,000; 500,000; 100,000, labeled S1-6, are listed. In accordance with the above-mentioned condition, the initial size of the territory for the map scale $S = 1 :$

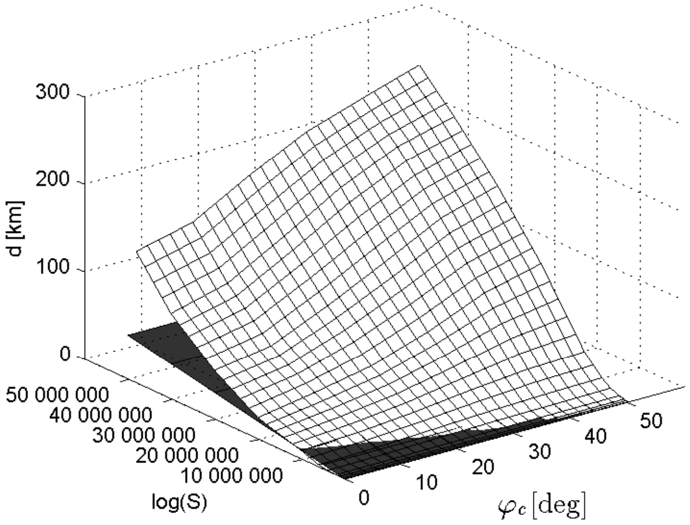

Fig. 8.2: A spatial diversity d, depending on the map scale S and the latitude φ_c, together with the graphical accuracy G of the map (the gray plane).

$100,000,000$, was set to the whole domain $\langle -90°, 90°\rangle \times \langle -180°, 180°\rangle$, and decreases linearly, depending on the map scale.

Position of the territory. Of the many options, the difficult, but the most commonly used variant - when the analyzed territory of the rectangular shape is located along the central meridian - was chosen. The central meridian was set to pass through the middle part of the map, which is obvious. Both the negligible values of distortions, and the analogous shape of the graticule, hindering the projection detection, are typical for these territories. The testing territory moves north, along the central meridian of the map, with the latitude increment $\Delta\varphi = 10°$ so the latitude of its center φ_c changes in the range of $\langle 0°, 70°\rangle$. Tests for scales S1, S2, for some sets, and all values of φ_c could not be performed; they are crossed out in the tables.

Adding errors. For the T1 test, all P sets have been intentionally contaminated by additional errors of a random character simulating the lack of a solid geometric basis or the imprecision input from the user. The error ε lies on a circle of the fixed radius r_ε for all points, taking values $\langle 1, 2, 3, 4, 5\rangle$ mm on the map. Its components can be expressed as

$$\varepsilon_x = r_\varepsilon \cos\alpha,$$
$$\varepsilon_y = r_\varepsilon \sin\alpha,$$

where $\alpha = rand(0, 2\pi)$, are added to both coordinates of the testing set P. The seconds test, T2, brings the uncertainty ε also to the reference set Q

$$\varepsilon_\varphi = \frac{r_\varepsilon S}{R}\cos\alpha,$$
$$\varepsilon_\lambda = \frac{r_\varepsilon S}{R}\sin\alpha,$$

in a similar manner.

Results of the tests. Both tests (see Tabs. 4, 5) were carried out for six different distributions of the analyzed features, variable map scale, variable position on the central parallel, and increasing error. The sinusoidal projection was compared to other 60 map projections, the simplified objective function was given by Eq. 5.4. Overall, 100 random samples have been generated for each variant, so the total amount of testing sets was 313,320; the computational time was 348 hours.

Let us take a closer look at the tables. The first column contains information about the central latitude φ_c of the set Q. For this position, six sets of different scales S have been generated (column two). The remaining column headings represent the additional errors belonging to r_ε, in millimeters. Each cell of the table contains the percentage efficiency of the detection for six different sets, in the aforementioned order, delimited with a slash. If the efficiency is higher than 50%, the detection process is classified as successful.

From the perspective of the **sets**, one meridian or one parallel is entirely inappropriate. They cover a small territory (long and narrow), where many projections may have the analogous shape of the meridian/parallel. Worse results were also obtained for the cluster. This is due to the set not covering the whole territory uniformly;

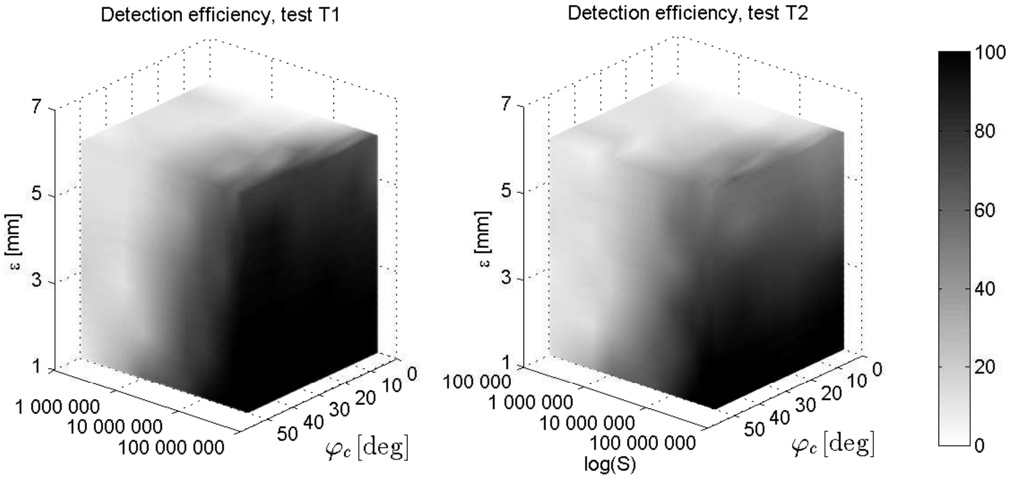

Fig. 8.3: Efficiency of the detection algorithm for tests T1 (a) and T2 (b), depending on the map scale, geographical position, and the error.

the spatial distribution of the features is variable. A circle is more reliable, but the empty part without points decreases the efficiency. Finally, a grid, or randomly spaced data, gave the best results; they represents the most reliable input sets.

Taking into account the geographic position of the set, the increasing **latitude** φ_c brings better efficiency, but in the northern and southern parts it decreases slightly. This is obvious, because the shapes of the graticule differ more at mid-latitudes; along the equator and the North/South poles they are similar. The interval $\langle 20^\circ, 60^\circ \rangle$ can be considered as suitable for analysis.

The decreasing **map scale** S, which is the second most important factor, significantly affects the detection process. Up to the scale of 1:1,000,000, corresponding with the territory of dimensions $\Delta\varphi = \Delta\lambda \doteq$ 3°, the results of the analysis are somewhat unreliable. This does not imply that the wrong projection is determined. However, projections having similar properties, may be found; see Fig. 8.1. For maps up to the scale of 1:500, 000, there are "small" differences (hundreds of meters), between the determined graticule shapes; see the auxiliary circle of the radius of 0.5 km centered at bottom left, point P_i. However, these discrepancies are less than the graphical accuracy of the map (1 mm on the map scale). Concurrently, for the large-scale maps, the requirement for the precision of input features is difficult to fulfill; it is less than 1 mm on the map. In general, it does not make sense to put emphasis on the best-determined projection, any of the top 5 determined solutions is, overall, acceptable. The problem with mid-scale and small-scale maps does not occur; there is a larger spatial diversity between the determined projections. See the auxiliary circle with a radius of 25 km (5 mm on the map scale).

Take advantage of these observations for the next criterion. Let us denote d to be an average spatial diversity of first k samples given by the formula

$$d = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k}\sqrt{\frac{\left\|P_i - P'_{i,j}\right\|_2^2}{k}},$$

where $P'_{i,j}$ represents $i-$th point of the $j-$th sample. If the spatial diversity d is less than the graphical accuracy of the map G, the projection may not be clearly detectable. Depending on the maps scale S and and the latitude φ_c, these areas are shown in Fig. 8.2. As expected, they relate particularly to the large-scale maps.

The most important factor is represented by the **geometrical accuracy** of the input features. It is obvious that the generally acceptable inaccuracy is between about 3 mm on the map; see Tabs. 4, 5. For world maps, the criterion is less strict; it is around 4 mm. Keep in mind, the value represents a radius of the circle (e.g. the uncertainty region) drawn at each point of the set. The operator should avoid placing the points outside the "error circles". The results showed that the contamination of both files by errors decreases the detection ability by about 20%, compared to a single file flaw. Therefore, the proposed method faster loses its ability to clearly determine the projection. In both cases, similar behavior of the detection algorithm is noted. The bold printed values belong to the acceptable combinations of parameters.

For the dependance of the percentage efficiency on the additional errors, the geographical position, and the map scale, generated for tests T1 and T2, see Fig. 8.3. As the source data, the randomly generated set of features (probably the most common set), has been generated. The areas where the efficiency is over 50%, are noticeable on both graphs; take into account the logarithmic scale of the x axis. They preferably cover the small-scale and mid-scale maps, located at mid-latitudes, up to 4 mm of the precision (test T1). For the second test, T2, the efficiency is significantly less, especially at a higher noise level (> 3 mm).

For maps of the world, and maps of continents, up to the scale 1:5,000,000, the moderately trained end-user may work well with the tool. For mid-scale maps up to the scale 1:500,000, the requirements of the precision of the input points are stronger ($\doteq 1$ mm).

Although it is not about accurate and exact simulation illustrating the distribution of errors in map, perhaps it sufficiently illustrates the capability of the algorithm.

The proposed methods described in the next article have similar precision requirements, which are insensitive to the choice of the optimizing method.

9 Conclusion

This article has introduced a new method estimating the unknown map's projection, and its parameters, based on the Nelder-Mead downhill simplex algorithm. Several versions of the objective function ϕ were presented. The complex variant takes into account 0D-2D elements, the simplified version, which is computationally cheaper, only 0D elements. Our proposed solution finding arguments minimizing the objective function ϕ is off-line and seems to be efficient and robust. It also achieved more accurate results than MapAnalyst.

The selected early maps, created since the 18th century, have been analyzed. For the vast majority of them, a map projection was successfully recognized. There is an uncertainty in the estimated parameters for Map 5, which used to be assigned with the Cassini-Soldner projection. However, due to the large scale of the map, our results did not confirm this assumption.

The above-mentioned techniques depend on many parameters; in particular, the analyzed territory size, its geographical location, shape, spatial distribution, precision of selecting points, or existence of the solid geometric basis of the analyzed map, play important roles. Small territories, up to size $\Delta\varphi = \Delta\lambda = 3^\circ$ (e.g. maps up to the scale 1:1 000 000), long and narrow territories placed around the prime meridian, equator, or poles, imprecisely selected matching points (> 4 mm error on the map), are almost undetectable; the impact of a map projection should not be less than the graphical accuracy of a map. Based on the analysis, the ideal amount 10-20 of the analyzed features was found.

All algorithms were implemented in new `detectproj` software available from

web.natur.cuni.cz/∼bayertom/detectproj/det_sw.html.

The C++ source code contains approximately 15,000 lines, supports 60 map projections, several detection methods and operating systems. Its user interface is designed as Proj.4 library, but the cartographic computations are based on newly developed kernel.

Our software is available free of charge. It may also be applicable as a tool for improving early map georeferencing, or may be used for semi-automatic acquisition of bibliographic meta data related to the map's projection.

In the next article, methods for on-line detection based on the NLSP approach will be presented.

As mentioned above, such techniques may be applicable and helpful for many libraries with large printed map collections. There is a need to determine the projection and its parameters for early maps, where the information is unknown. The next step is integration of the selected detection algorithms into the software for on-line georeferencing of scanned maps

http://www.georeferencer.org/.

The tool, used by many famous libraries (e.g., The British Library, The National Library of Scotland), enriched with new capabilities, may bring interesting results when studying the national cartographic heritage.

Acknowledgement

This article was supported by a grant from the Ministry of Culture of the Czech Republic, No. DF11P01OVV003 "TEMAP - Technology for Access to Czech Map Collections: Methodology and Software for the Protection and Re-use of the National Cartographic Heritage".

References

[1] Arkin, E.M., Chew, L.P., Huttenlocher, D.P., Kedem, K., Mitchell, J.S.B.: An efficiently computable metric for comparing polygonal shapes. IEEE J PAMI **13**(3), 209–216 (1991) 2, 5.2, 5.2, 6.3, 6.3

[2] Bai, X., Yang, X., Latecki, L.J., Liu, W., Tu, Z.: Learning context-sensitive shape similarity by graph transduction. IEEE J PAMI **32**(5), 861–874 (2010) 2

[3] Balletti, C., Guerra, F., Monti, C.: Venice: New life in an old map: Geometrical analysis and georeferenced visualisation of historic maps. Geoinformatica **3**, 40–43 (2000) 4, 5.3

[4] Bertram, M., Wendrock, H.: Characterization of planar local arrangement by means of the delaunay neighbourhood. Journal of Microscopy **181**(1), 45–53 (1996). DOI 10.1046/j.1365-2818.1996.93374.x. URL http://dx.doi.org/10.1046/j.1365-2818.1996.93374.x 2

[5] Buchar: Assessment of the map projections for world maps. (18th International Cartographic Conference, Olomouc, 2009) 3.1

[6] Bugayevskiy, L.M., Snyder, J.: Map Projections: A Reference Manual. CRC Press (1995) 5

[7] Burger, G., Embury, J., Wilkinson, D.: The characterization of microstructures using tessellations and their application to deformation processes. Simulation and Theory of Evolving Microstructures pp. 199–209 (1990) 2

[8] Chang, S.H., Cheng, F.H., Hsu, W.H., Wu, G.Z.: Fast algorithm for point pattern matching: Invariant to translations, rotations and scale changes. Pattern Recognition **30**(2), 311–320 (1997) 2

[9] Craciunescu, V., Constantinescu, S.: Eharta (2006). URL http://earth.unibuc.ro/articole/eHarta?lang=en 2

[10] Erle, S., Krishnan, S., Waters, T.: World map warp (2009). URL http://warp.worldmap.harvard.edu/ 2

[11] Esri: Identify an unknown projected coordinate system using arcmap (2003) 2

[12] Esri: Identify an unknown projected coordinate system using arcmap (2005) 2

[13] Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981). DOI 10.1145/358669.358692. URL http://doi.acm.org/10.1145/358669.358692 2

[14] Flacke, W., Kraus, B., Warcup, C.: Working with projections and datum transformations in ArcGIS: theory and practical examples. Points Verlag (2005). URL http://books.google.cz/books?id=PfEsAQAAMAAJ 2

[15] Frank, R., Ester, M.: A quantitative similarity measure for maps. In: A. Riedl, W. Kainz, G.A. Elmes (eds.) Progress in Spatial Data Handling, pp. 435–450. Springer Berlin Heidelberg (2006). URL http://dx.doi.org/10.1007/3-540-35589-8_28 5.2

[16] Gao, F., Han, L.: Implementing the nelder-mead simplex algorithm with adaptive parameters. Comput. Optim. Appl. **51**(1), 259–277 (2012). DOI 10.1007/s10589-010-9329-3. URL http://dx.doi.org/10.1007/s10589-010-9329-3 2, 5.1

[17] Geography, Division, M., Hébert, J.: Panoramic maps of Anglo-American cities: a checklist of maps in the collections of the Library of Congress, Geography and Map Division. Library of Congress (1974). URL http://books.google.cz/books?id=eZ7oGwAACAAJ 4, 5.3

[18] Gkalp, E., Gngr, O., Boz, Y.: Evaluation of different outlier detection methods for gps networks. Sensors (Peterboroug **8**(11), 7344–7358 (2008). URL http://www.mdpi.com/1424-8220/8/11/7344/ 2

[19] Hekimoglu, S., Berber, M.: Effectiveness of robust methods in heterogeneous linear models. Journal of Geodesy **76**(11-12), 706–713 (2003). DOI 10.1007/s00190-002-0289-y 2

[20] Huang, J.F., Lai, S.H., Cheng, C.M.: Robust fundamental matrix estimation with accurate outlier detection. J. Inf. Sci. Eng. **23**(4), 1213–1225 (2007) 2

[21] Huber, P.: Robust Statistics. Wiley series in probability and mathematical statistics. Probability and mathematical statistics. John Wiley & Sons (1981). URL http://books.google.cz/books?id=HQp2BKN-qWoC 2

[22] Itoh, R., Horizoe, M., Gotoh, K.: A method for measuring two-dimensional dispersed state of particles. Advanced Powder Technology **6**(2), 81 – 89 (1995). DOI 10.1163/156855295X00086. URL http://www.sciencedirect.com/science/article/pii/S0921883108605348 2

[23] Jenny, B., Hurni, L.: Cultural heritage: Studying cartographic heritage: Analysis and visualization of geometric distortions. Comput. Graph. **35**(2), 402–411 (2011). DOI 10.1016/j.cag.2011.01.005. URL http://dx.doi.org/10.1016/j.cag.2011.01.005 2, 6.1.1

[24] Kelley, C.T.: Iterative Methods for Linear and Nonlinear Equations. No. 16 in Frontiers in Applied Mathematics. SIAM (1995). URL http://www.siam.org/books/textbooks/fr16_book.pdf 2

[25] Knight, N.L., Wang, J.: A comparison of outlier detection procedures and robust estimation methods in gps positioning. Journal of Navigation **62**(04), 699 (2009). URL http://www.journals.cambridge.org/abstract_S0373463309990142 2

[26] Kowal, K.C., P[rbreve]idal, P.: Online georeferencing for libraries: The british library implementation of georeferencer for spatial metadata enhancement and public engagement. Journal of Map & Geography Libraries **8**(3), 276–289 (2012). DOI 10.1080/15420353.2012.700914. URL http://www.tandfonline.com/doi/abs/10.1080/15420353.2012.700914 2

[27] Krarup, T., Juhl, J., K., K.: Gtterdmmerung over least squares. Proceedings of International Society for Photogrammetry 14th Congress. ISPRS Commission III, Hamburg, Germany (1980) 2, 5.3

[28] Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E.: Convergence properties of the nelder–mead simplex method in low dimensions. SIAM J. on Optimization **9**(1), 112–147 (1998). DOI 10.1137/S1052623496303470. URL http://dx.doi.org/10.1137/S1052623496303470 2

[29] Latecki, L.J., Lakamper, R.: Shape similarity measure based on correspondence of visual parts. IEEE J PAMI **22**(10), 1185–1190 (2000) 2

[30] Liu, Y.K., Wang, X.Q., Bao, S.Z., Gombosi, M., Zalik, B.: An algorithm for polygon clipping, and for determining polygon intersections and unions. Computers &amp; Geosciences **33**(5), 589 – 598 (2007). DOI 10.1016/j.cageo.2006.08.008. URL http://www.sciencedirect.com/science/article/pii/S0098300406001841 2

[31] Marcelpoil, R., Usson, Y.: Methods for the study of cellular sociology: Voronoi diagrams and parametrization of the spatial relationships. Journal of Theoretical Biology **154**(3), 359 – 369 (1992). DOI 10.1016/S0022-5193(05)80176-6. URL http://www.sciencedirect.com/science/article/pii/S0022519305801766 2

[32] Margalit, A., Knott, G.D.: An algorithm for computing the union, intersection or difference of two polygons. Computers &amp; Graphics **13**(2), 167 – 183 (1989). DOI 10.1016/0097-8493(89)90059-9. URL http://www.sciencedirect.com/science/article/pii/0097849389900599 2

[33] Martnez, F., Rueda, A.J., Feito, F.R.: A new algorithm for computing boolean operations on polygons. Computers &amp; Geosciences **35**(6), 1177 – 1185 (2009). DOI 10.1016/j.cageo.2008.08.009. URL http://www.sciencedirect.com/science/article/pii/S0098300408002793 2, 6.2

[34] Mckinnon, K.I.M.: Convergence of the nelder-mead simplex method to a non-stationary point. Tech. rep., SIAM J. Optim (1996) 2

[35] MetaCarta, Labs: Map rectifier (2009). URL http://labs.metacarta.com/rectifier/ 2

[36] Mount, D.M., Netanyahu, N.S., Le Moigne, J.: Improved algorithms for robust point pattern matching and applications to image registration. In: Proceedings of the fourteenth annual symposium on Computational geometry, SCG '98, pp. 155–164. ACM, New York, NY, USA (1998). DOI 10.1145/276884.276902. URL http://doi.acm.org/10.1145/276884.276902 2, 6.1.1

[37] Nelder, J.A., Mead, R.: A simplex method for function minimization. The Computer Journal **7**(4), 308–313 (1965). DOI 10.1093/comjnl/7.4.308. URL http://comjnl.oxfordjournals.org/content/7/4/308.abstract 2

[38] Okabe, A., Boots, B., Sugihara, K., Chiu, D.S.N., Chiu, S.N.: Spatial Tessellations: Concepts and Applications of Voronoi Diagrams (Wiley Series in Probability and Statistics). Wiley (2000) 6.2

[39] Okabe, A., Sugihara, K.: Spatial Analysis Along Networks: Statistical and Computational Methods. Statistics in Practice. John Wiley & Sons (2012). URL http://books.google.cz/books?id=k738tgAACAAJ 2

[40] Pham, N., Wilamowski, B.M.: Improved nelder meads simplex method and applications. Electrical and Computer Engineering, Auburn University, Alabama, US (2011) 2

[41] Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes 3rd Edition: The Art of Scientific Computing, 3 edn. Cambridge University Press, New York, NY, USA (2007) 2

[42] Pridal, P.: Georeferencer (2011). URL http://www.georeferencer.org 2

[43] Pun, C.M., Li, C.: Shape classification using simplification and tangent function. In: Proceedings of the 8th WSEAS International Conference on Circuits, systems, electronics, control & signal processing, CSECS'09, pp. 261–266. World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA (2009). URL http://dl.acm.org/citation.cfm?id=1736282.1736330 2

[44] Reps, J.: Bird's Eye Views: Historic Lithographs of North American Cities. Princeton Architectural Press (1998). URL http://books.google.cz/books?id=bPTFQgAACAAJ 4, 5.3

[45] Rousseeuw, P.J., Leroy, A.M.: Robust regression and outlier detection. John Wiley & Sons, Inc., New York, NY, USA (1987) 2

[46] Snyder, J.P.: Map projections – a working manual. Tech. Rep. 1395, U. S. Geological Survey (1987). URL http://pubs.er.usgs.gov/usgspubs/pp/pp1395 3.1, 5, 8.1

[47] Subbarao, R., Meer, P.: Beyond ransac: User independent robust regression. In: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop, CVPRW '06, pp. 101–. IEEE Computer Society, Washington, DC, USA (2006). DOI 10.1109/CVPRW.2006.43. URL http://dx.doi.org/10.1109/CVPRW.2006.43 2

[48] Veltkamp, R.C., Hagedoorn, M.: State-of-the-art in shape matching. Tech. rep., PRINCIPLES OF VISUAL INFORMATION RETRIEVAL (1999) 2

[49] Volotao, C.F.d.S., Santos, R.D.C.d., Erthal, G.J., Dutra, L.V.: Shape characterization with turning functions. In: Proceedings..., vol. 1, pp. 554–557. 17th International Conference on Systems, Signals and Image Processing, Editora da Universidade Federal Fluminense (2010). URL http://urlib.net/dpi.inpe.br/plutao/2010/11.11.17.36.41 2

[50] Walters, F.H., Jr, L.R., Morgan, S.L., Deming, S.N.: Sequential Simplex Optimization: A Technique for Improving Quality and Productivity in Research, Development, and Manufacturing (Chemometrics series). CRC (1991). URL http://www.worldcat.org/isbn/0849358949 2

[51] Wamelen, P.B.V., Li, Z., Iyengar, S.S.: A fast algorithm for the point pattern matching problem (1999) 2, 5.2, 6.1.1

[52] Wieser, A., Brunner, F.K.: An extended weight model for GPS phase observations. Earth, Planets, and Space **52**, 777–782 (2000) 2, 5.3