Constrained Ellipse Fitting with Center on a Line

Patrick Waibel[1] · **Jörg Matthes**[1] · **Lutz Gröll**[1]

Abstract Fitting an ellipse to given data points is a common optimization task in computer vision problems. However, the possibility of incorporating the prior constraint "the ellipse's center is located on a given line" into the optimization algorithm has not been examined so far. This problem arises, for example, by fitting an ellipse to data points representing the path of the image positions of an adhesion inside a rotating vessel whose position of the rotational axis in the image is known. Our new method makes use of a constrained algebraic cost function with the incorporated "ellipse center on given line"-prior condition in a global convergent one-dimensional optimization approach. Further advantages of the algorithm are computational efficiency and numerical stability.

Keywords Ellipse fitting · Constrained cost function · Eigenvalue problem

1 Introduction

The fitting of ellipses to measured data points arises in many computer vision and pattern recognition tasks, where it is used to gain a high-level interpretation of the image data. Often, only the measured data points and no other information are available (e. g. eye tracking [23], medical imaging [27,28]). In this case, an unconstrained (free) ellipse fitting method has to be applied. In applications where the measured data points result from the path of the image positions of points from rotating objects (e. g. camera/laser-based 3D scanning and structure from motion [19,26] or the image positions of an adhesion inside a rotating vessel such as a rotary kiln [31]) additional information about the location of the rotational center is available. Here, the rotational center must be located on the rotational axis, which is usually known beforehand.

The existing ellipse fitting approaches lack the possibility to integrate such prior knowledge about the ellipse's center location into the optimization process. If there are only few noisy data points or the data points are disadvantageously positioned (e. g. due to occlusion), these unconstrained approaches fail to find a good ellipse fit. Our new constrained ellipse fitting approach allows to integrate the prior knowledge on the possible location of the ellipse's center.

As the applicability of the constrained ellipse fitting method might not seem obvious, a possible scenario is presented in Fig. 1a. Here, the inside of a rotating vessel is observed by a camera whose mounting position is shifted from the rotational axis of the vessel. Then, the peak of an adhesion inside the vessel describes a circular movement in the real world. In the acquired camera image sequence, this leads to an elliptic movement (adhesion path) due to the oblique view (Fig. 1b). In this case, the position of the rotational axis in the image where the ellipse's center has to lie on is known beforehand. The height and depth of the adhesion inside the vessel and thus the angle of the oblique view on the rotational plane are unknowns and so are the ellipse's parameters beside the center location on the rotational axis. Therefore, it is not possible to geometrically transform the

✉ Patrick Waibel
patrick.waibel@kit.edu

Jörg Matthes
joerg.matthes@kit.edu

Lutz Gröll
lutz.groell@kit.edu

[1] Institute for Applied Computer Science, Karlsruhe Institute of Technology (KIT), Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

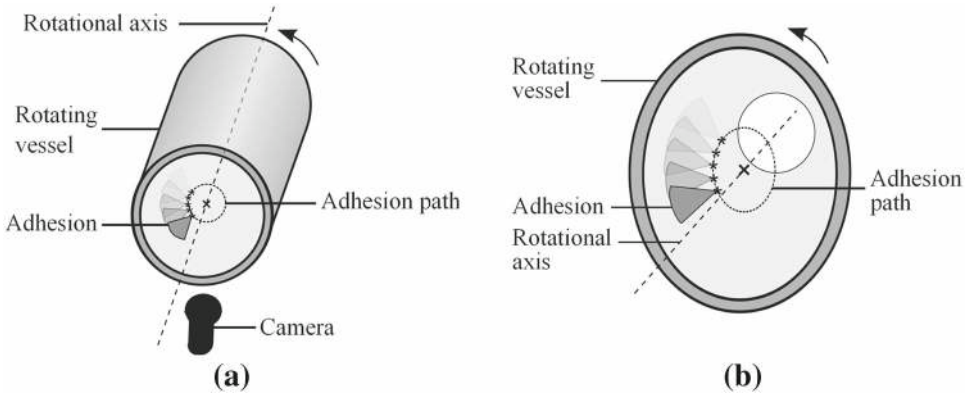


Fig. 1 Example application of the constrained ellipse fitting method; **a** overview of scenario with rotating vessel, adhesion, and camera system; **b** outline of an acquired image of scenario

constrained ellipse fitting problem into a constrained circle fitting problem.

Figure 2 shows the advantage of our constrained ellipse fitting method over two unconstrained approaches. The data points are created from an original ellipse by restricting it to a small angle range (reflecting occlusion) and adding some zero-mean Gaussian noise. Additionally, the line where the center has to be located on is given. If this prior knowledge is not included, the best fit for the noisy measurements using the method from Fitzgibbon et al. [9] leads to a small ellipse with high offset. Employing the method from Szpak et al. [30], the ellipse fit gets too large and is also far from the original ellipse. Exploiting the information, that the ellipse's center must lie on the given line, a much more appropriate ellipse fit is found.

This paper presents a new method which enables the fitting of ellipses whose centers are restricted to a given line. Furthermore, the special case of a given line segment is covered. In order to avoid a complicating case-by-case analysis, the line segment treatment is not considered in the theoretical part of this paper. A detailed discussion of the line segment restriction is given in Sect. 3.4.

Task: Let $N \geq 4$ disjoint data points $\boldsymbol{x}_i \in \mathbb{R}^2$ be given. Additionally, a prior known line is defined in the plane. Then an ellipse fitting has to be performed such that the ellipse's center is located on this line.

In order to solve this task, the following topics need to be resolved:

- proper description of ellipse
- proper description of line
- formulation of cost function
- formulation of necessary parameter constraints
- formulation of requirements on the data
- complete treatment of the mathematical problem
- design of a robust and efficient algorithm.

1.1 Related Work

Several existing methods provide the possibility to find ellipses by a given set of data points in an unconstrained

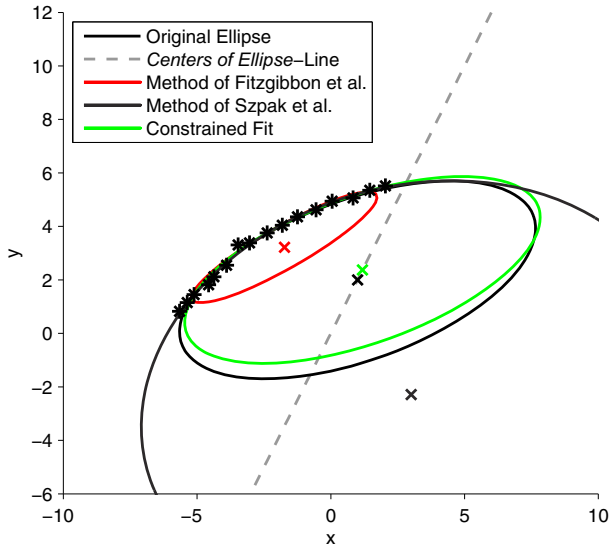


Fig. 2 Comparison of two unconstrained ellipse fitting methods (Fitzgibbon et al. [9] and Szpak et al. [30]) and the new constrained ellipse fitting method; data points and line are given

case. Coarsely, they can be divided into two basic approaches: clustering methods and optimization methods.

Amongst the clustering methods, the Hough Transformation is a very popular method [12,33]. Besides, the Randomized Hough Transformation [22], the RANSAC method [6] and fuzzy clustering approaches [8] are available.

Optimization methods try to minimize a cost function (or loss function) that values the accuracy of the ellipse fit according to the given data points. The cost function is either based on the algebraic distance or the geometric distance [2,7]. The geometric distance can be identified with the maximum likelihood estimate under the assumption of homogeneous Gaussian noise in the data. The algebraic distance on the other hand has advantages in manageability of the problem [11,32]. Fitzgibbon et al. presented the first approach that allows for a direct specific least square fitting of ellipses based on the algebraic distance [9]. Halif et al. [13] provided improvements towards numerical stability. Newer

ellipse fitting methods proved to achieve higher accuracies [3,16,17,20,29,34]. Nevertheless, most of these approaches do not guarantee that the final result is an ellipse, rather hyperbolas or parabolas might occur. In [21], a method is presented that tries to avoid non-ellipses by proper re-initializations. A method that ensures an ellipse is proposed by [30]. Nevertheless, the computational effort is high which can be a problem for real-time applications. A large summary of algorithms for ellipsoidals is given in [24].

So far, none of the existing ellipse fitting methods allow for the direct incorporation of prior knowledge such as the ellipse's center is located on a given line. Our constrained ellipse fitting approach is based on the Fitzgibbon method because it is the only method that always guarantees an ellipse fit in a non-iterative and therefore fast way. Other methods which proved to be superior in accuracy in free conic fitting require a post hoc procedure to ensure an ellipse fit [3,17].

1.2 Paper Outline

The remainder of the paper is organized as follows. Section 2 covers the problem formulation including the derivation of the cost function, the required constraints, and conditions on the data. On this basis, the minimizers of the cost function are characterized and a method to reach the global minimum is presented. In Sect. 3, the numerical algorithm is outlined. Here, details of our MATLAB implementation are shown, followed by two methods that provide the possibility for obtaining good initial values. The special case of a given line segment instead of an unbounded line is discussed. Additionally, some hints for a microcontroller implementation and alternative algorithms for solving the cost function are presented. Section 4 shows the results of our algorithm in a comparison with unconstrained ellipse fitting methods.

1.3 Notation

In this paper, we use the following notation. Matrices are written with bold capital letters (e. g. $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$). Vectors are denoted by lowercase bold letters such as $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$. Scalars are denoted by lowercase Roman or Greek letters (e. g. a, b, c, α, β, γ). All vectors are assumed to be column vectors, if not stated otherwise. A superscript T denotes a transposition. Zero vectors and unit vectors of length N are given by $\mathbf{0}_N$ and $\mathbf{1}_N$, respectively. The identity matrix of size $N \times N$ is denoted by $\boldsymbol{I}_N$. $\lambda(\boldsymbol{A}, \boldsymbol{B})$ denotes a generalized eigenvalue of the problem $\boldsymbol{A}\boldsymbol{x} = \lambda \boldsymbol{B}\boldsymbol{x}; \boldsymbol{x} \neq \mathbf{0}$. Further, we employ the block matrix notation, especially that of bordered matrices in $\mathbb{R}^{3\times3}$ of the form $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_{11} & \boldsymbol{a}_{12} \\ \boldsymbol{a}_{21}^T & a_{22} \end{bmatrix}$ with $\boldsymbol{A}_{11} \in \mathbb{R}^{2\times2}$.

2 Problem Formulation

An appropriate cost function which allows the incorporation of the ellipse's center restriction has to be established. For that purpose, the notation of the center restriction has to be chosen in such a way, that it does not appear as an explicit restriction in the objective form (Sect. 2.1). Furthermore, constraints of the cost function have to be formulated that assure the ellipse form, the translation and rotation invariance, and the uniqueness (Sect. 2.2). In Sect. 2.3, conditions and tests on the data are formulated, which guarantee a correct and unique solution of the ellipse fitting problem. In Sect. 2.4, the reformulation of the cost function into an one-dimensional minimum search is presented. A new method which allows to find the global minimum in the reformulated cost function is given in Sect. 2.5.

2.1 Establishing the Cost Function

For our purpose, ellipses in center form are convenient. They are defined by

$$f_1(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}_c)^T \boldsymbol{A} (\boldsymbol{x} - \boldsymbol{x}_c) + g = 0 \tag{1}$$

with $\boldsymbol{A} = \begin{bmatrix} a & \frac{b}{2} \\ \frac{b}{2} & c \end{bmatrix}$ being positive definite and g being negative and finite (ensures proper ellipse area). $\boldsymbol{A}$ describes the orientation and eccentricity, $\boldsymbol{x}_c = \begin{bmatrix} x_c & y_c \end{bmatrix}^T$ the center, and g is related with the ellipse's area

$$\text{area} = \frac{\pi}{\sqrt{\det(\boldsymbol{A})}} \cdot (-g). \tag{2}$$

A suitable line description for the possible ellipse's center position is

$$\boldsymbol{x_c} = \boldsymbol{p} + \gamma \boldsymbol{q} \tag{3}$$

with the point vector $\boldsymbol{p} = \begin{bmatrix} p_1 & p_2 \end{bmatrix}^T$, the directional vector $\boldsymbol{q} = \begin{bmatrix} q_1 & q_2 \end{bmatrix}^T$, and the parameter γ.

A common approach to fit an ellipse to N given data points $\boldsymbol{x}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^T$ is minimizing the sum of squared algebraic errors caused by noisy data

$$Q_1(\boldsymbol{x}_c, \boldsymbol{A}, g) = \sum_{i=1}^{N} f_1^2(\boldsymbol{x}_i) \tag{4}$$

with respect to the unknowns. Substituting the parameter vector $\boldsymbol{x}_c$ yields a modified ellipse representation with the new parameter γ.

$$f_1(\boldsymbol{x})|_{\boldsymbol{x}_c = \boldsymbol{p}+\gamma\boldsymbol{q}} = (\boldsymbol{x} - \boldsymbol{p} - \gamma\boldsymbol{q})^T \boldsymbol{A} (\boldsymbol{x} - \boldsymbol{p} - \gamma\boldsymbol{q}) + g = 0 \tag{5}$$

It fulfills the constraint that the center is located on a line. This new function is reduced by one parameter. Using the substitution $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \boldsymbol{p}$, (5) can be written as

$$f_2(\tilde{\boldsymbol{x}}) = (\tilde{\boldsymbol{x}} - \gamma \boldsymbol{q})^T \boldsymbol{A}(\tilde{\boldsymbol{x}} - \gamma \boldsymbol{q}) + g = 0 \tag{6}$$

The cost function is then given by

$$Q_2(\gamma, \boldsymbol{A}, g) = \sum_{i=1}^{N} f_2^2(\tilde{\boldsymbol{x}}_i) \tag{7}$$

By introducing the new parameter

$$h = g + \gamma^2 \boldsymbol{q}^T \boldsymbol{A} \boldsymbol{q} \tag{8}$$

the ellipse representation $f_2(\tilde{\boldsymbol{x}})$ can be written in a modified form

$$\begin{aligned} f_2(\tilde{\boldsymbol{x}}) &= \tilde{\boldsymbol{x}}^T \boldsymbol{A} \tilde{\boldsymbol{x}} - 2\gamma \boldsymbol{q}^T \boldsymbol{A} \tilde{\boldsymbol{x}} + h \\ &= a(\tilde{x}^2 - \gamma 2 q_1 \tilde{x}) + b(\tilde{x}\tilde{y} - \gamma\,(q_1 \tilde{y} + q_2 \tilde{x})) \\ &\quad + c(\tilde{y}^2 - \gamma 2 q_2 \tilde{y}) + h \\ &= f_3(\tilde{\boldsymbol{x}}). \end{aligned} \tag{9}$$

Separating the data terms regarding their independence from γ leads to a matrix notation for the cost function Q_3.

$$\begin{aligned} Q_3(.) &= \left\| \begin{bmatrix} f_3(\tilde{\boldsymbol{x}}_1) \\ \vdots \\ f_3(\tilde{\boldsymbol{x}}_N) \end{bmatrix} \right\|_2^2 \\ &= \left\| \left(\begin{bmatrix} \tilde{x}_1^2 & \tilde{x}_1\tilde{y}_1 & \tilde{y}_1^2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{x}_N^2 & \tilde{x}_N\tilde{y}_N & \tilde{y}_N^2 & 1 \end{bmatrix} \right. \right. \\ &\quad \left. \left. - \gamma \begin{bmatrix} 2q_1\tilde{x}_1 & q_1\tilde{y}_1 + q_2\tilde{x}_1 & 2q_2\tilde{y}_1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 2q_1\tilde{x}_N & q_1\tilde{y}_N + q_2\tilde{x}_N & 2q_2\tilde{y}_N & 0 \end{bmatrix} \right) \begin{bmatrix} a \\ b \\ c \\ h \end{bmatrix} \right\|_2^2 \end{aligned} \tag{10}$$

By introducing the data matrices,

$$\boldsymbol{D}_0 = \begin{bmatrix} \tilde{x}_1^2 & \tilde{x}_1\tilde{y}_1 & \tilde{y}_1^2 \\ \vdots & \vdots & \vdots \\ \tilde{x}_N^2 & \tilde{x}_N\tilde{y}_N & \tilde{y}_N^2 \end{bmatrix} \tag{11}$$

and

$$\boldsymbol{D}_1 = \begin{bmatrix} 2q_1\tilde{x}_1 & q_1\tilde{y}_1 + q_2\tilde{x}_1 & 2q_2\tilde{y}_1 \\ \vdots & \vdots & \vdots \\ 2q_1\tilde{x}_N & q_1\tilde{y}_N + q_2\tilde{x}_N & 2q_2\tilde{y}_N \end{bmatrix} \tag{12}$$

as well as the parameter vector

$$\boldsymbol{\theta} = \begin{bmatrix} a & b & c \end{bmatrix}^T \tag{13}$$

a simplified representation of the cost function is given by

$$Q_3(\gamma, \boldsymbol{\theta}, h) = \left\| \left(\begin{bmatrix} \boldsymbol{D}_0 & \mathbf{1}_N \end{bmatrix} - \gamma \begin{bmatrix} \boldsymbol{D}_1 & \mathbf{0}_N \end{bmatrix} \right) \begin{bmatrix} \boldsymbol{\theta} \\ h \end{bmatrix} \right\|_2^2. \tag{14}$$

2.2 Establishing Additional Constraints

Additional constraints are necessary to guarantee the following properties of the solution:

- ellipse form
- rotation and translation invariance
- uniqueness.

The positive definiteness of $\boldsymbol{A}$ in (1), which is equivalent to $a > 0$ and $4ac - b^2 > 0$, ensures the ellipse form. The sign constraint $a > 0$ helps ignore the sign-equivalent solutions $-\boldsymbol{A}, -g$. Rotation and translation invariance can be achieved, if the constraints are symmetric with respect to the eigenvalues $\lambda_i(\boldsymbol{A})$ [9]. From $4ac - b^2 = 4 \det \boldsymbol{A} = 4\lambda_1(\boldsymbol{A})\lambda_2(\boldsymbol{A})$, it can be seen that $4ac - b^2 = 1$ possesses such a symmetry. In addition, the equality $4ac - b^2 = 1$ ensures the positiveness and reduces the number of degrees of freedom. Furthermore, from (2), it follows that solely g represents the ellipse's area $= -2\pi g$. Using the parameter vector from (13), this constraint can now be written as [9]

$$\boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta} = 1 \tag{15}$$

with

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & 2 \\ 0 & -1 & 0 \\ 2 & 0 & 0 \end{bmatrix}. \tag{16}$$

Using this notation, the optimization problem can be stated as

$$\underset{\gamma, \boldsymbol{\theta}, h}{\text{minimize}} \quad Q_3(\gamma, \boldsymbol{\theta}, h) \quad \text{s. t.} \quad \boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta} = 1 \text{ and } \theta_1 > 0. \tag{17}$$

The feasible set

$$\mathcal{F} = \left\{ \begin{bmatrix} \boldsymbol{\theta} \\ h \\ \gamma \end{bmatrix} \in \mathbb{R}^5 : \boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta} = 1,\ \theta_1 > 0 \right\} \tag{18}$$

is a non-convex set, since $\boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta} = 1$ is a nonlinear equality. The cost function considered without constraints is a non-convex function, because of the bi-linear dependency

between γ and the parameters $\boldsymbol{\theta}$, h. Therefore, different local minimizers are to be expected. Another difficulty arises as the feasible set is not closed ($\theta_1 > 0$). Additionally, it is not bounded, because the matrix $\boldsymbol{B}$ in $\boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta} = 1$ is indefinite.

Hence an infimum instead of a minimum is possible. In order to exclude such situations, we formulate some conditions on the data in the next section.

2.3 Conditions on the Data

In order to get an unique free ellipse fit, five distinct points are necessary. This results from the five parameters together with the condition for an ellipse. In our situation, we have an additional constraint, namely the line where the ellipse's center has to lie on. Thus, only four distinct points are required. It should be remarked that in the three point case, an infinite number of perfect ellipses with center on the line exist.

Let us now consider the infimum topic that arises from the strict inequalities $0 < -g < \infty$. From a geometric point of view, the two boundaries correspond to degenerated ellipses whose areas tend to zero or to infinity. We denote an ellipse as degenerated if it has a zero or an infinite area.

Surely, an ellipse possesses a zero area if it is a line, i.e., all data points lie on a single line. The case of an ellipse with infinite area arises if the data points are located on two parallel lines. In the theory of conics, two other cases of degenerated ellipses, the single point, and intersecting lines are known. The single point case is out of focus because $N \geq 4$ and the intersecting lines are coming from the degeneration of a hyperbola.

In contrast to the free ellipse fitting problem where no real center is defined for a degenerated ellipse, in the constrained case a "center candidate" can be found. In the single line case, the "center candidate" of the degenerated ellipse is the intersection point of the single line with the given line for the centers (Fig. 3a). In the parallel line case it is the intersection point of the parallel's midline with the given line (Fig. 3b). In both cases, an exception is built if the line or the parallels are parallel to the given line (Fig. 3c, d). Then an ellipse fitting could be done, which results in a non-zero value of the cost function, except that the line or the midline is identical to the given line (Fig. 3e, f). In addition, it should be mentioned that the single line case (Fig. 3c) possesses also an infimum for all γ, since this case can be interpreted as the parallel case, where all points lie on one side of the degenerated ellipse. Since such cases are non-generic in standard applications, we exclude the line and the parallel cases completely in order to avoid numerical troubles and the distinction of numerous cases. All derived results can be summarized as follows:

Data Condition: If $N \geq 4$ points are available and if all points are not located on a single line or on a pair of parallels, then the ellipse fit problem possesses neither a non-isolated solution nor a degenerated solution.

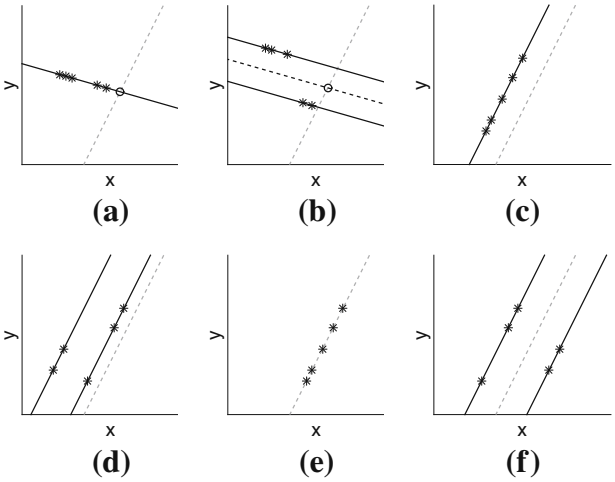


Fig. 3 Special cases in constrained ellipse fitting (*gray dashed line* given line for ellipse center, *circle* "center candidate"): **a** all data points on single line; **b** data points on two parallels, the parallels' *midline is black dashed*; **c** all data points on a line which are parallel to the given line; **d** data points on two parallels which are parallel to the given line; **e** all data points on the given line; **f** data points on two parallels, midline of these parallels is the given line

A proper test for this condition is based on the analysis of

$$\underbrace{\begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & & & & & \vdots \\ x_N^2 & x_N y_N & y_N^2 & x_N & y_N & 1 \end{bmatrix}}_{\boldsymbol{T}} \cdot \underbrace{\begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \\ \eta_5 \\ \eta_6 \end{bmatrix}}_{\boldsymbol{\eta}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (19)$$

Lemma 1 (Data condition test)

- *If all points are on a single line then* $\operatorname{rank}(\boldsymbol{T}) = 3$, *($N \geq 4$),*
- *If all points are on two different parallels then*
  - $\operatorname{rank}(\boldsymbol{T}) = 5$, *($N \geq 5$) and*

$$\det \begin{bmatrix} \eta_1 & \frac{\eta_2}{2} & \frac{\eta_4}{2} \\ \frac{\eta_2}{2} & \eta_3 & \frac{\eta_5}{2} \\ \frac{\eta_4}{2} & \frac{\eta_5}{2} & \eta_6 \end{bmatrix} = 0 \quad \text{and} \quad \det \begin{bmatrix} \eta_1 & \frac{\eta_2}{2} \\ \frac{\eta_2}{2} & \eta_3 \end{bmatrix} = 0 \quad (20)$$

    where η_i are the components of an arbitrary vector in the null space $\mathcal{N}(\boldsymbol{T})$ or
  - $\operatorname{rank}(\boldsymbol{T}) = 4$, *($N = 4$) and* $\det[\boldsymbol{x}_1 - \boldsymbol{x}_2,\ \boldsymbol{x}_3 - \boldsymbol{x}_4] \cdot \det[\boldsymbol{x}_1 - \boldsymbol{x}_3,\ \boldsymbol{x}_2 - \boldsymbol{x}_4] \cdot \det[\boldsymbol{x}_1 - \boldsymbol{x}_4,\ \boldsymbol{x}_2 - \boldsymbol{x}_3] = 0$.

Proof Let $\alpha_1 x + \alpha_2 y + \alpha_3 = 0$ be an arbitrary line where all data points are located on, then

$$
\boldsymbol{T} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \alpha_1 \\ 0 & 0 & 0 & 0 & 1 & \alpha_2 \\ 0 & 0 & 0 & 0 & 0 & \alpha_3 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & \alpha_1 & 0 & 0 \\ 0 & 1 & 0 & \alpha_2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
\cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \alpha_1 & 0 \\ 0 & 0 & 1 & 0 & \alpha_2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} x_1^2 & x_1 y_1 & y_1^2 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N^2 & x_N y_N & y_N^2 & 0 & 0 & 0 \end{bmatrix}
$$

from which $\operatorname{rank}(\boldsymbol{T}) = 3$ can be concluded.

$\mathcal{N}(\boldsymbol{T})$ obtains all solutions of $\boldsymbol{T}\boldsymbol{\eta} = \boldsymbol{0}_N$ or equivalent, all parameters for perfect conics. Since two parallels are nontrivial perfect conics, $\dim \mathcal{N}(\boldsymbol{T}) \geq 1$, respectively, $\operatorname{rank}(\boldsymbol{T}) \leq 5$ has to hold. Further, the parallels are assumed as different which excludes $\operatorname{rank}(\boldsymbol{T}) = 3$. From the theory of degenerated conics, it is known that in the case of two parallels (20) is necessary [18]. Here, the first determinant condition characterizes all degenerated conics and the second is simply the area infinity condition. In the case $\operatorname{rank}(\boldsymbol{T}) = 5$ or equivalent $\dim \mathcal{N}(\boldsymbol{T}) = 1$, an arbitrary element of $\mathcal{N}(\boldsymbol{T})$ can be used for the test because all others are multiples. But if $\operatorname{rank}(\boldsymbol{T}) = 4$ or equivalent $\dim \mathcal{N}(\boldsymbol{T}) = 2$, the test is more involved. That is why we work for $N = 4$ directly with the points. □

2.4 Characterization of the Minimizers

The goal of a solution is to find the parameters that minimize the cost function given in (17). This will be done by a successive reduction of the cost function's unknown variables

$$
\min_{\substack{\gamma,\boldsymbol{\theta},h \\ \boldsymbol{\theta}^T\boldsymbol{B}\boldsymbol{\theta}=1 \\ \theta_1>0}} Q_3(\gamma,\boldsymbol{\theta},h) = \min_{\gamma} \min_{\substack{\boldsymbol{\theta} \\ \boldsymbol{\theta}^T\boldsymbol{B}\boldsymbol{\theta}=1,\theta_1>0}} \min_{h} Q_3(\gamma,\boldsymbol{\theta},h) \tag{21a}
$$

$$
= \min_{\gamma} \min_{\substack{\boldsymbol{\theta} \\ \boldsymbol{\theta}^T\boldsymbol{B}\boldsymbol{\theta}=1,\theta_1>0}} Q_4(\gamma,\boldsymbol{\theta}) \tag{21b}
$$

$$
= \min_{\gamma} Q_5(\gamma). \tag{21c}
$$

The following Lemmas 2–3 show the intermediate results.

Lemma 2 *The problem*

$$
\underset{h}{\text{minimize}}\ Q_3(\gamma,\boldsymbol{\theta},h) \tag{22}
$$

possesses the minimum

$$
Q_4(\gamma,\boldsymbol{\theta}) = \gamma^2\boldsymbol{\theta}^T\boldsymbol{C}_2\boldsymbol{\theta} + \gamma\boldsymbol{\theta}^T\boldsymbol{C}_1\boldsymbol{\theta} + \boldsymbol{\theta}^T\boldsymbol{C}_0\boldsymbol{\theta} \tag{23}
$$

with the corresponding minimizer

$$
h_{\text{opt}}(\gamma,\boldsymbol{\theta}) = -\frac{1}{N}\boldsymbol{1}_N^T\left(\boldsymbol{D}_0 - \gamma\boldsymbol{D}_1\right)\boldsymbol{\theta}, \tag{24}
$$

where the abbreviations

$$
\boldsymbol{C}_0 = \boldsymbol{D}_0^T\boldsymbol{Z}\boldsymbol{D}_0 \tag{25a}
$$

$$
\boldsymbol{C}_1 = -\boldsymbol{D}_0^T\boldsymbol{Z}\boldsymbol{D}_1 - \boldsymbol{D}_1^T\boldsymbol{Z}\boldsymbol{D}_0 \tag{25b}
$$

$$
\boldsymbol{C}_2 = \boldsymbol{D}_1^T\boldsymbol{Z}\boldsymbol{D}_1 \tag{25c}
$$

are used. The centering matrix $\boldsymbol{Z}$ is defined by

$$
\boldsymbol{Z} = \boldsymbol{I}_N - \frac{1}{N}\boldsymbol{1}_{N\times N}. \tag{26}
$$

Proof If h_{opt} is a minimizer of problem (17) then the Karush-Kuhn-Tucker (KKT) condition with respect to h is given by

$$
\left.\frac{\partial Q_3}{\partial h}\right|_{\text{opt}} = 2\cdot\boldsymbol{1}_N^T\left(\begin{bmatrix}\boldsymbol{D}_0 & \boldsymbol{1}_N\end{bmatrix} - \gamma\begin{bmatrix}\boldsymbol{D}_1 & \boldsymbol{0}_N\end{bmatrix}\right)\left.\begin{bmatrix}\boldsymbol{\theta} \\ h\end{bmatrix}\right|_{\text{opt}} = 0 \tag{27}
$$

from which (24) follows. Reformulation of Q_3 yields

$$
Q_3(\gamma,\boldsymbol{\theta},h) = \left\|\left(\boldsymbol{D}_0 - \gamma\boldsymbol{D}_1\right)\boldsymbol{\theta} + \boldsymbol{1}_N h\right\|_2^2 \tag{28}
$$

and substituting h using (24) simplifies the cost function

$$
Q_4(\gamma,\boldsymbol{\theta}) = \left\|\boldsymbol{Z}\left(\boldsymbol{D}_0 - \gamma\boldsymbol{D}_1\right)\boldsymbol{\theta}\right\|_2^2 \tag{29a}
$$

$$
= \gamma^2\boldsymbol{\theta}^T\boldsymbol{C}_2\boldsymbol{\theta} + \gamma\boldsymbol{\theta}^T\boldsymbol{C}_1\boldsymbol{\theta} + \boldsymbol{\theta}^T\boldsymbol{C}_0\boldsymbol{\theta}. \tag{29b}
$$

□

Lemma 3 *If the data condition from Sect. 2.3 holds, the problem*

$$
\underset{\boldsymbol{\theta}^T\boldsymbol{B}\boldsymbol{\theta}=1,\theta_1>0}{\text{minimize}}\ Q_4(\gamma,\boldsymbol{\theta}) \tag{30}
$$

possesses the minimum

$$
Q_5(\gamma) = \lambda_{\max}\left(\gamma^2\boldsymbol{C}_2 + \gamma\boldsymbol{C}_1 + \boldsymbol{C}_0, \boldsymbol{B}\right) \tag{31}
$$

with the corresponding minimizer

$$
\boldsymbol{\theta}_{\text{opt}}(\gamma) = \frac{\operatorname{sign}(\theta_1^*)}{\sqrt{\boldsymbol{\theta}^{*T}\boldsymbol{B}\boldsymbol{\theta}^*}}\boldsymbol{\theta}^*. \tag{32}
$$

Here $\boldsymbol{\theta}^*$ denotes an arbitrary eigenvector of the generalized eigenvalue problem

$$\left(\gamma^2 \boldsymbol{C}_2 + \gamma \boldsymbol{C}_1 + \boldsymbol{C}_0\right) \boldsymbol{\theta} = \lambda \boldsymbol{B} \boldsymbol{\theta} \tag{33}$$

corresponding to $\lambda_{\max}$.

Proof Using the method of Lagrange multipliers,

$$L(\boldsymbol{\theta}, \lambda; \gamma) = Q_4(\gamma, \boldsymbol{\theta}) - \lambda(\boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta} - 1) \tag{34}$$

the first-order KKT condition states

$$\left.\frac{\partial L}{\partial \boldsymbol{\theta}}\right|_{\text{opt}} = 2\left(\gamma^2 \boldsymbol{C}_2 + \gamma \boldsymbol{C}_1 + \boldsymbol{C}_0 - \lambda \boldsymbol{B}\right) \boldsymbol{\theta} = \boldsymbol{0}_3 \tag{35}$$

$$\boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta}|_{\text{opt}} = 1 \tag{36}$$

$$\theta_1|_{\text{opt}} > 0. \tag{37}$$

(35) states the eigenvalue problem (33). The constraints (36) and (37) are satisfied by the scaling in (32). Multiplying (33) with $\boldsymbol{\theta}^T$ from the left gives

$$\underbrace{\boldsymbol{\theta}^T \left(\gamma^2 \boldsymbol{C}_2 + \gamma \boldsymbol{C}_1 + \boldsymbol{C}_0\right) \boldsymbol{\theta}}_{=Q_4(\gamma, \boldsymbol{\theta})}|_{\text{opt}} = \lambda \underbrace{\boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta}}_{=1}|_{\text{opt}}. \tag{38}$$

Hence, $Q_4(\gamma, \boldsymbol{\theta}_{\text{opt}}) = \lambda$ holds. Since the cost function is non-negative, only the non-negative values of λ are of interest. Specifically, among all non-negative values $\lambda_{\max}$ is the correct value, which is shown next. Using

$$Q_4(\gamma, \boldsymbol{\theta}) = \boldsymbol{\theta}^T \underbrace{\left(\gamma^2 \boldsymbol{C}_2 + \gamma \boldsymbol{C}_1 + \boldsymbol{C}_0\right)}_{\boldsymbol{M}(\gamma)} \boldsymbol{\theta} \geq 0 \; \forall \, \boldsymbol{\theta} \tag{39}$$

from (29b), it follows that $\boldsymbol{M}(\gamma)$ has to be positive definite or positive semi-definite.

At first, if $\boldsymbol{M}(\gamma)$ is positive definite for γ, then Sylvester's Law of Inertia can be applied in (33) in order to conclude on the sign configuration of $(\lambda_1, \lambda_2, \lambda_3)$. The eigenvalues of $\boldsymbol{B}$ are $(-2, -1, 2)$, therefore $\text{sign}(\lambda_1, \lambda_2, \lambda_3) = (-, -, +)$. That is why there is only one positive eigenvalue denoted by $\lambda_{\max}$. The positiveness of $\lambda_{\max}$ and the positive definiteness of $\boldsymbol{M}(\gamma)$ imply $\boldsymbol{\theta}^{*T} \boldsymbol{B} \boldsymbol{\theta}^* > 0$ which guarantees a real square root in (32).

At second, if $\boldsymbol{M}(\gamma)$ is positive semi-definite, then Sylvester's Law of Inertia cannot be applied. But it can be concluded that at least one λ is zero. Therefore, the sign configurations $(-, -, 0)$, $(-, 0, 0)$, $(0, 0, 0)$, $(-, 0, +)$, $(0, 0, +)$, and $(0, +, +)$ have to be examined. In $(-, -, 0)$, $\lambda_{\max} = 0$ is the only non-negative λ and herewith the desired solution. $(-, 0, 0)$ and $(0, 0, 0)$ correspond to multiple perfect conic fits which are excluded by the data condition from Sect. 2.3. In the cases $(-, 0, +)$ and $(0, 0, +)$, also $\lambda_{\max}$ is the correct solution. This can be clarified by regarding an infinitesimal variation of the data, which would lead to a positive definite $\boldsymbol{M}(\gamma)$ and thus to the sign configuration $(-, -, +)$. It is obvious that the positive λ in the semi-definite case hast to correspond with the positive λ after the infinitesimal variation. An infinitesimal variation of the case $(0, +, +)$ cannot yield the sign configuration $(-, -, +)$, and therefore this sign configuration does not appear in our problem. □

Theorem 1 *The local minimizers of the problem*

$$\underset{\boldsymbol{x}_c, \boldsymbol{A}, g}{\text{minimize}} \; Q_1(\boldsymbol{x}_c, \boldsymbol{A}, g) \tag{40}$$

can be calculated from the local minimizers of

$$\underset{\gamma}{\text{minimize}} \; Q_5(\gamma) \tag{41}$$

using the following equations:

$$\gamma_{\text{loc}} = \underset{\gamma}{\text{argmin}} \; \lambda_{\max}\left(\gamma^2 \boldsymbol{C}_2 + \gamma \boldsymbol{C}_1 + \boldsymbol{C}_0, \boldsymbol{B}\right) \tag{42a}$$

$$\boldsymbol{\theta}_{\text{loc}} = \boldsymbol{\theta}_{\text{opt}}(\gamma_{\text{loc}}) \qquad \textit{by (32)} \tag{42b}$$

$$h_{\text{loc}} = h_{\text{opt}}(\gamma_{\text{loc}}, \boldsymbol{\theta}_{loc}) \qquad \textit{by (24)} \tag{42c}$$

$$\boldsymbol{A}_{\text{loc}} = \begin{bmatrix} \theta_{1,\text{loc}} & \frac{1}{2}\theta_{2,\text{loc}} \\ \frac{1}{2}\theta_{2,\text{loc}} & \theta_{3,\text{loc}} \end{bmatrix} \tag{42d}$$

$$\boldsymbol{x}_{c,\text{loc}} = \boldsymbol{p} + \gamma_{\text{loc}} \boldsymbol{q} \tag{42e}$$

$$g_{\text{loc}} = h_{\text{loc}} - \gamma_{\text{loc}}^2 \boldsymbol{q}^T \boldsymbol{A}_{\text{loc}} \boldsymbol{q}. \tag{42f}$$

In addition $Q_1(\boldsymbol{x}_{c,\text{loc}}, \boldsymbol{A}_{\text{loc}}, g_{\text{loc}}) = Q_5(\gamma_{\text{loc}})$.

Proof The result is a consequence of the reformulation of (4) using the steps (7) and (14), the optimization splitting (21) as well as the Lemmas 2–3. □

The essential benefit of Theorem 1 is the reduction of the ellipse fitting problem into an one-dimensional minimum search, considering that the ellipse's center is located on a given line. Further, this theorem presents a relation between all solutions of the one-dimensional optimization problem and those of the higher-dimensional optimization problem.

Due to the difficult nonlinear interdependences in (33), a closed form solution of the optimization problem would be very tedious. Either it would lead to a higher-order polynomial solution (approach by Groebner bases [1]) or to an eigenvalue problem of expanded order (approach by linearization techniques via dimension expansion [15]). For this reason, we prefer a simple iterative search algorithm for γ which solves the generalized eigenvalue problem (33) in each iteration step.

2.5 From Local to Global Minimum

Since one-dimensional search methods usually get stuck in local minima at non-convex problems, in the following section a new search method is proposed. It provides the

possibility to reach the global minimum target orientated, i.e., it is no trial-and-error approach with arbitrary initial values. Rather, in our method, new initial values always ensure a lower value of the cost function which guarantees global convergence to the global minimum. Furthermore, a statement on the maximum number of minima of our problem is given.

2.5.1 Tunneling Idea

A typical cost function $Q_5(\gamma)$ is illustrated in Fig. 4a. Here, one local minimum and one global minimum are observable. Fig. 4b shows the corresponding ellipses. In most cases, the goal of the optimization is to find the global minimum. Nevertheless, many optimization methods get stuck in a local minimum.

The main idea to circumvent this is to apply the Tunneling Theorem which is proposed at the end of this section. Figuratively speaking, a tunnel is dug through the adjacent little hill to get into the deeper valley. Before the Theorem is given, three preparative Lemmas are required.

The following Lemma 4 shows that the characteristic polynomial of (33) is contrary to the expectation not of sixth power in γ but of fourth power.

Lemma 4 *The characteristic polynomial*

$$r(\gamma,\lambda) = \det \underbrace{\left(\gamma^2 \boldsymbol{C}_2 + \gamma \boldsymbol{C}_1 + \boldsymbol{C}_0 - \lambda \boldsymbol{B}\right)}_{=\boldsymbol{R}} \tag{43}$$

defined by the generalized eigenvalue problem (33) is of fourth degree if the data condition in Sect. 2.3 holds. Specifically, it is a polynomial of fourth degree with respect to γ and of third degree with respect to λ.

Proof The four point condition prevents a singular quadratic pencil, i.e., $\det \boldsymbol{R}(\gamma,\lambda) \equiv 0$ corresponding to a total degeneration of the polynomial degree. The line as well as the parallel exclusion for the data points prevents a partial degeneration of the polynomial degree.

Since each equation that equals zero can be extended by any non-zero factor, problem (43) is equivalent to

$$\underbrace{\det \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ q_2^2 & -2q_1q_2 & q_1^2 \end{bmatrix}}_{\neq 0, \text{since } [q_1,q_2]\neq[0,0]} \cdot \det(\boldsymbol{R}) \cdot \det \begin{bmatrix} 1 & 0 & q_2^2 \\ 0 & 1 & -2q_1q_2 \\ 0 & 0 & q_1^2 \end{bmatrix}$$
$$= \det\left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ q_2^2 & -2q_1q_2 & q_1^2 \end{bmatrix} \boldsymbol{R} \begin{bmatrix} 1 & 0 & q_2^2 \\ 0 & 1 & -2q_1q_2 \\ 0 & 0 & q_1^2 \end{bmatrix} \right) = 0.$$

Calculating all terms of $\boldsymbol{R}$ and denoting arbitrary matrix elements by $\times$ results in

$$\det\left(\gamma^2 \begin{bmatrix} \times & \times & 0 \\ \times & \times & 0 \\ 0 & 0 & 0 \end{bmatrix} + \gamma \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & 0 \end{bmatrix} + \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} - \lambda \begin{bmatrix} 0 & 0 & \times \\ 0 & \times & \times \\ \times & \times & 0 \end{bmatrix} \right) = 0. \tag{44}$$

The zeros follow from the linear dependency of the columns in $\boldsymbol{D}_1$, compare

$$\begin{bmatrix} 2q_1\tilde{x}_1 & q_1\tilde{y}_1 + q_2\tilde{x}_1 & 2q_2\tilde{y}_1 \\ \vdots & \vdots & \vdots \\ 2q_1\tilde{x}_N & q_1\tilde{y}_N + q_2\tilde{x}_N & 2q_2\tilde{y}_N \end{bmatrix} \cdot \begin{bmatrix} q_2^2 \\ -2q_1q_2 \\ q_1^2 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \tag{45}$$

and from the anti-diagonal structure of $\boldsymbol{B}$ as well as

$$\begin{bmatrix} q_2^2 & -2q_1q_2 & q_1^2 \end{bmatrix} \cdot \boldsymbol{B} \cdot \begin{bmatrix} q_2^2 \\ -2q_1q_2 \\ q_1^2 \end{bmatrix} = 0. \tag{46}$$

Applying Sarrus' rule in (44), it is easy to see that γ^4 is the highest order term for γ, and λ^3 is the highest order term for λ. □

For the set of all minimizer–minimum pairs $(\gamma_{\text{loc}}, \lambda_{\text{loc}})$, the following Lemma holds.

Lemma 5 *Any local minimizer γ_{loc} of $Q_5(\gamma)$ is a double or a fourfold root of $r(\gamma, \lambda_{\text{loc}})$.*

Proof A double root γ^* of $r(\gamma, \lambda_{\text{loc}})$ is characterized by

$$r(\gamma^*, \lambda_{\text{loc}}) = 0 \quad \text{and} \quad \frac{\mathrm{d}}{\mathrm{d}\gamma} r(\gamma^*, \lambda_{\text{loc}}) = 0. \tag{47}$$

Condition (47) is also part of the characterization of a fourfold root. In a local minimum the left statement is fulfilled for $\gamma^* = \gamma_{\text{loc}}$ by (33) and (43). Let λ_{loc} be a simple root of $r(\gamma, \lambda)$. Then in a local minimum $\frac{\mathrm{d}Q_5}{\mathrm{d}\gamma} = \frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}|_{\text{loc}} = 0$ holds. If this term is inserted in the implicit differentiation of $r(\gamma, \lambda)$

$$r_\gamma(\gamma,\lambda) = \frac{\mathrm{d}}{\mathrm{d}\gamma} r(\gamma,\lambda) = \frac{\partial r}{\partial \gamma} + \frac{\partial r}{\partial \lambda} \cdot \frac{\mathrm{d}\lambda}{\mathrm{d}\gamma} = 0 \tag{48}$$

one gets

$$r_\gamma(\gamma,\lambda_{\text{loc}})\Big|_{\gamma_{\text{loc}}} = \frac{\mathrm{d}}{\mathrm{d}\gamma} r(\gamma,\lambda_{\text{loc}})\Big|_{\gamma_{\text{loc}}} = 0 \tag{49}$$

$$\underset{\gamma^*=\gamma_{\text{loc}}}{\Rightarrow} \frac{\partial r}{\partial \gamma}\Big|_{\gamma^*} + \frac{\partial r}{\partial \lambda} \cdot \underbrace{\frac{\mathrm{d}\lambda_{\text{loc}}}{\mathrm{d}\gamma}}_{=0} = \frac{\mathrm{d}}{\mathrm{d}\gamma} r(\gamma^*, \lambda_{\text{loc}}) = 0. \tag{50}$$

Let λ_{loc} now be a double root of $r_\gamma(\gamma, \lambda)$. In this case, the term $\frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}$ is not defined because the derivative of a defective double

eigenvalue does not exist. In such cases, some subgradient calculus has to be applied. But the assumption of a double root ensures now that $\frac{\partial r}{\partial \lambda} = 0$ holds. Hence, a reduction to the same form like before shows the validity of the right statement in (47). If two local minimizers merge in one point, a fourfold root occurs. n-fold roots with $n > 4$ are impossible due to the fourth degree of $r(\gamma, \lambda_{\text{loc}})$. □

The following Lemma shows that the cost function $Q_5(\gamma)$ is a coercive function.

Lemma 6 *Assuming that the ellipse constraint* (15) *holds, then $Q_5(\gamma)$ is coercive, i.e.,* $\lim_{|\gamma|\to\infty} Q_5(\gamma) = \infty$.

Proof From the equivalence of $Q_5(\gamma)$ with $Q_4(\gamma, \boldsymbol{\theta})$ in (29a) according to (33), the following generalized eigenvalue problem can be inferred

$$(\boldsymbol{D}_0 - \gamma \boldsymbol{D}_1)^T \underbrace{\boldsymbol{Z}^T \boldsymbol{Z}}_{=\boldsymbol{Z}} (\boldsymbol{D}_0 - \gamma \boldsymbol{D}_1)\boldsymbol{\theta} = \lambda \boldsymbol{B}\boldsymbol{\theta}. \tag{51}$$

Dividing both sides by γ^2 leads to

$$\left(\tfrac{1}{\gamma}\boldsymbol{D}_0 - \boldsymbol{D}_1\right)^T \boldsymbol{Z} \left(\tfrac{1}{\gamma}\boldsymbol{D}_0 - \gamma \boldsymbol{D}_1\right) \boldsymbol{\theta} = \tfrac{\lambda}{\gamma^2} \boldsymbol{B}\boldsymbol{\theta}. \tag{52}$$

The proof will be done by contradiction. Assuming $\lambda_{\max}(\gamma)$ is bounded by $0 \leq \lambda_{\max}(\gamma) \leq \lambda_\infty < \infty$, then the limit $|\gamma| \to \infty$ in (52) yields

$$\boldsymbol{D}_1^T \boldsymbol{Z} \boldsymbol{D}_1 \boldsymbol{\theta}_\infty = \boldsymbol{0}. \tag{53}$$

The limit solution $\boldsymbol{\theta}_\infty$ is an element of the null space $\mathcal{N}(\boldsymbol{D}_1^T \boldsymbol{Z} \boldsymbol{D}_1)$. Since $\mathcal{N}(\boldsymbol{D}_1^T \boldsymbol{Z} \boldsymbol{D}_1) \supseteq \mathcal{N}(\boldsymbol{D}_1)$ and the ranks of both matrices are equal to two, we have from (45)

$$\boldsymbol{\theta}_\infty \in \text{span}\left(\begin{bmatrix} q_2^2 \\ -2q_1q_2 \\ q_1^2 \end{bmatrix}\right) \overset{(46)}{\Rightarrow} \boldsymbol{\theta}_\infty^T \boldsymbol{B} \boldsymbol{\theta}_\infty = 0. \tag{54}$$

This contradicts the constraint $\boldsymbol{\theta}_\infty^T \boldsymbol{B} \boldsymbol{\theta}_\infty = 1$. Thus $\lambda_{\max}(\gamma)$ cannot be bounded when the ellipse constraint has to be fulfilled. Consequently, $\lim_{|\gamma|\to\infty} Q_5(\gamma) = \infty$. □

With Lemmas 4, 5 and 6 the basis for stating the following Tunneling Theorem is given:

- A variation of the parameter γ leads to different values λ for the cost function Q_5.
- For a fixed λ, a polynomial of fourth degree with respect to γ is given i.e., multiple parameter values γ can be associated with the same value of λ, and these γ are the roots of (43).
- In order for a particular γ to characterize a local minimum λ_{loc}, γ has to be a double or a fourfold root of (43) with λ_{loc} substituted for λ in the equation.
- The cost function $Q_5(\gamma)$ is a coercive function.

Theorem 2 (Tunneling Theorem) *A local minimizer γ_{loc} is a global one if $r(\gamma, \lambda_{\text{loc}}) = 0$ has no other real root beside γ_{loc}. Additionally, if $r(\gamma, \lambda_{\text{loc}}) = 0$ possesses two distinct real roots beside γ_{loc}, there exists a γ_{new} in the interval given by these two roots with the property $Q_5(\gamma_{\text{new}}) < Q_5(\gamma_{\text{loc}})$. If there exists a double root $\gamma_{3,4}$ distinct from γ_{loc}, then this root marks a second global isolated minimum provided that $Q_5(\gamma_{3,4}) = \lambda_{\text{loc}}$.*

Proof The polynomial $r(\gamma, \lambda_{\text{loc}})$ is quartic by Lemma 4 and γ_{loc} is a double root by Lemma 5. In consequence, three cases for the other two roots are possible.

If the other two roots are a pair of conjugate complex values or $\gamma_{3,4} = \gamma_{\text{loc}}$, then there is no other real γ having the same or a smaller value of Q_5. That means $\gamma_{\text{loc}} = \gamma_{\text{glob}}$ (Fig. 5).

In the case of a distinct double root $\gamma_{3,4}$, then $\gamma_{\text{loc}} = \gamma_{\text{glob},1}$ and $\gamma_{3,4} = \gamma_{\text{glob},2}$ are isolated global minimizers with $Q_5(\gamma_{\text{glob},1}) = Q_5(\gamma_{\text{glob},2})$ (Fig. 6). A special case arises when γ_{loc} represents a perfect ellipse fit with $Q_5(\gamma_{\text{loc}}) = 0$ and simultaneously $\gamma_{3,4}$ represents a perfect fit of some other conic. Then the best ellipse at $\gamma_{3,4}$ yields a $Q_5(\gamma_{3,4}) > 0$ and consequently only γ_{loc} is the global minimizer.

If there are two distinct real roots (γ_3, γ_4), i.e., $Q_5(\gamma_3) = Q_5(\gamma_4) = Q_5(\gamma_{\text{loc}}) = \lambda_{\text{loc}}$, then between these roots smaller cost function values $Q_5(\gamma_{\text{new}})$ with $\gamma_3 < \gamma_{\text{new}} < \gamma_4$ exist (Fig. 4). In order to clarify this statement, a deeper analysis of the topology of $Q_5(\gamma)$ is carried out.

Let λ be a differentiable cost function on the real line $\mathbb{R}$ (λ is effectively the same as Q_5). Given $r \in \mathbb{R}$, we denote by $\lambda^{-1}(r)$ the preimage of r by λ, that is, $\lambda^{-1}(r) = \{x \in \mathbb{R} \mid f(x) = r\}$. Let γ_{loc} be a local minimizer of λ. Because the polynomial $r(\gamma; \lambda_{\text{loc}})$ is quartic by Lemma 4 and γ_{loc} is a double root by Lemma 5, the cost function λ intersects the horizontal line $\lambda(\gamma) = \lambda_{\text{loc}}$ at most three times. This fact greatly constrains the topology of the graph of the cost function and can be used to determine whether a local minimizer is a global one.

Suppose that $\lambda^{-1}(\lambda(\gamma_{\text{loc}}))$ consists of γ_{loc} and two other numbers γ_3 and γ_4 which are non-stationary points for λ. This in particular means that $\lambda(\gamma_{\text{loc}}) = \lambda(\gamma_3) = \lambda(\gamma_4)$ and $\frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}\big|_{\gamma_3} \neq 0$ and $\frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}\big|_{\gamma_4} \neq 0$. We assume that $\gamma_3 < \gamma_4$. Let γ_{glob} be a global minimizer of λ. We investigate under what conditions the inequality $\gamma_3 < \gamma_{\text{glob}} < \gamma_4$ holds. We consider three cases:

Case 1 $\gamma_{\text{loc}} < \gamma_3 < \gamma_4$ (Fig. 7a).
We first show that

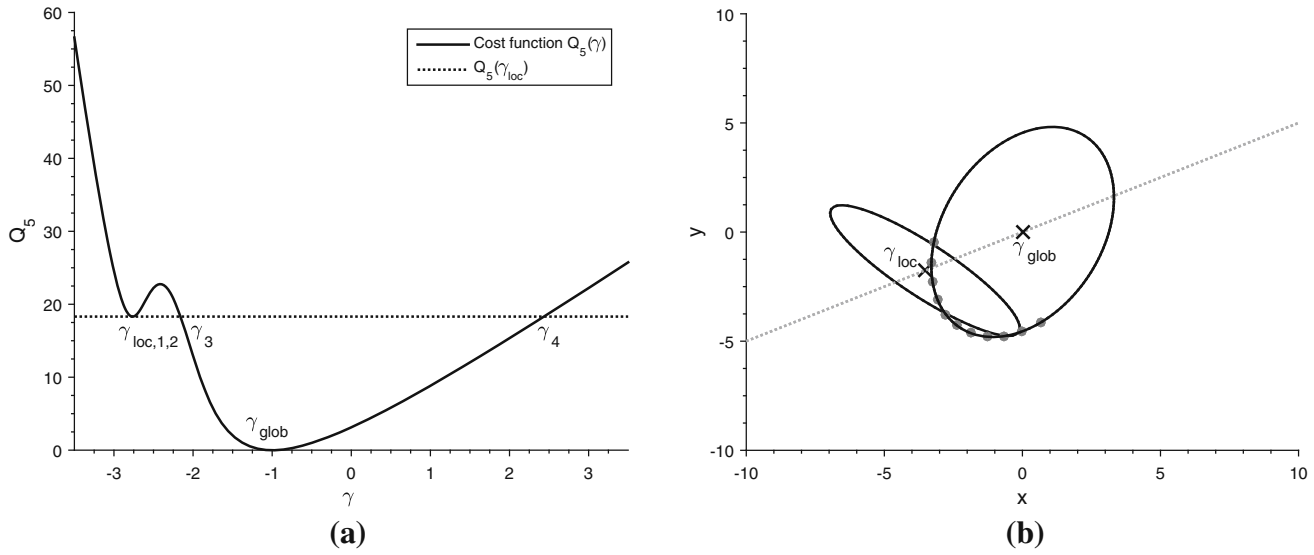


Fig. 4 Cost function $Q_5(\gamma)$ with two minima in γ_{loc} and γ_{glob}; the roots of the quartic polynomial $r(\gamma, \lambda_{\text{loc}})$ are the double root $\gamma_{1,2}$, γ_3 and γ_4

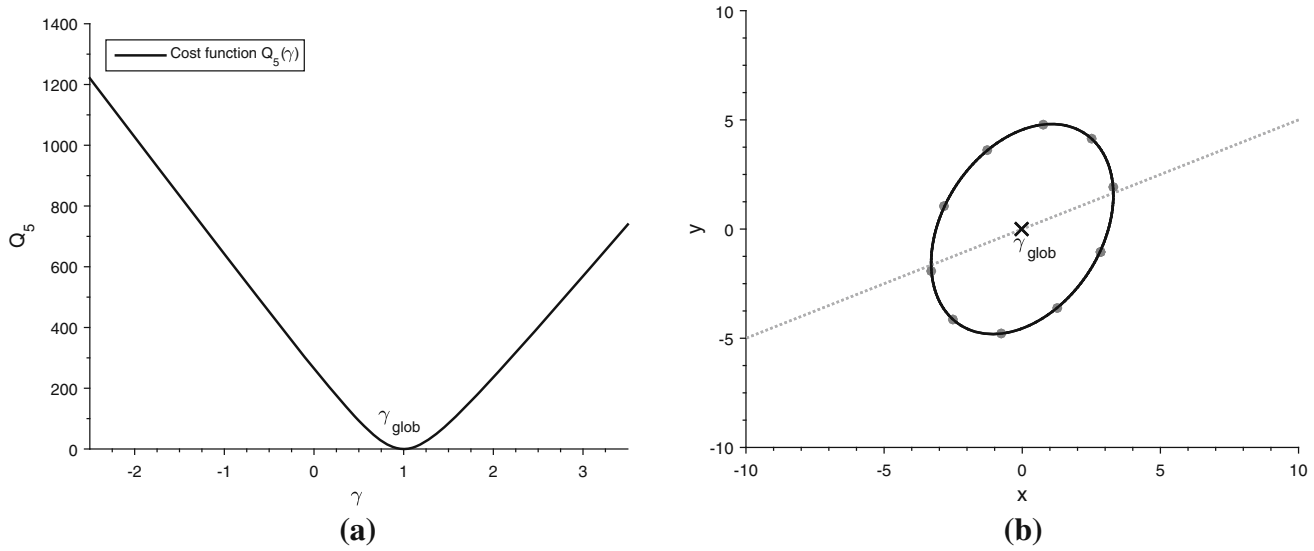


Fig. 5 Cost function $Q_5(\gamma)$ with only one minimum in γ_{glob}

(a) $\lambda(\gamma) \geq \lambda(\gamma_{\text{loc}})$ for $\gamma < \gamma_{\text{loc}}$,
(b) $\lambda(\gamma) \geq \lambda(\gamma_{\text{loc}})$ for $\gamma \in (\gamma_{\text{loc}}, \gamma_3)$,
(c) $\lambda(\gamma) \geq \lambda(\gamma_{\text{loc}})$ for $\gamma > \gamma_4$.

To prove (a), suppose a contrario that there exists a $\gamma' < \gamma_{\text{loc}}$ with $\lambda(\gamma') < \lambda(\gamma_{\text{loc}})$. Since γ_{loc} is a local minimizer, there exists an open interval I around γ_{loc} such that $\lambda(\mu) > \lambda(\gamma_{\text{loc}})$ for every $\mu \in I \backslash \{\gamma_{\text{loc}}\}$. Choose μ from I so that $\gamma' < \mu < \gamma_{\text{loc}}$. Then, of course, $\lambda(\mu) > \lambda(\gamma_{\text{loc}})$. Now this inequality and the inequality $\lambda(\gamma') < \lambda(\gamma_{\text{loc}})$ imply that there exists a $\gamma'' \in (\gamma', \mu)$ such that $\lambda(\gamma'') = \lambda(\gamma_{\text{loc}})$. Being smaller than μ, γ'' is smaller than γ_{loc}. It follows that γ'' is different from γ_3, γ_4, and γ_{loc}. But this is incompatible with the fact that $\lambda^{-1}(\lambda(\gamma_{\text{loc}})) = \{\gamma_{\text{loc}}, \gamma_3, \gamma_4\}$. The assertion is proved.

To prove (b), suppose that there exists a $\gamma' \in (\gamma_{\text{loc}}, \gamma_3)$ with $\lambda(\gamma') < \lambda(\gamma_{\text{loc}})$. Since γ_{loc} is a local minimizer, we see, reasoning as before, that there exists a $\mu \in (\gamma_{\text{loc}}, \gamma')$ such that $\lambda(\mu) > \lambda(\gamma_{\text{loc}})$. The inequalities $\lambda(\mu) > \lambda(\gamma_{\text{loc}})$ and $\lambda(\gamma') < \lambda(\gamma_{\text{loc}})$ imply that there exists a $\gamma'' \in (\mu, \gamma')$ such that $\lambda(\gamma'') = \lambda(\gamma_{\text{loc}})$. Located between μ and γ', γ'' lies between γ_{loc} and γ_3. In particular, γ'' is different from γ_3, γ_4, and γ_{loc}. But again this contradicts the fact that $\lambda^{-1}(\lambda(\gamma_{\text{loc}})) = \{\gamma_{\text{loc}}, \gamma_3, \gamma_4\}$. The assertion is proved.

To prove (c), suppose that there exists a $\gamma' > \gamma_4$ with $\lambda(\gamma') < \lambda(\gamma_{\text{loc}})$. Since γ_4 is a non-stationary point, there exists $\varepsilon > 0$ such that one of the following two possibilities hold:

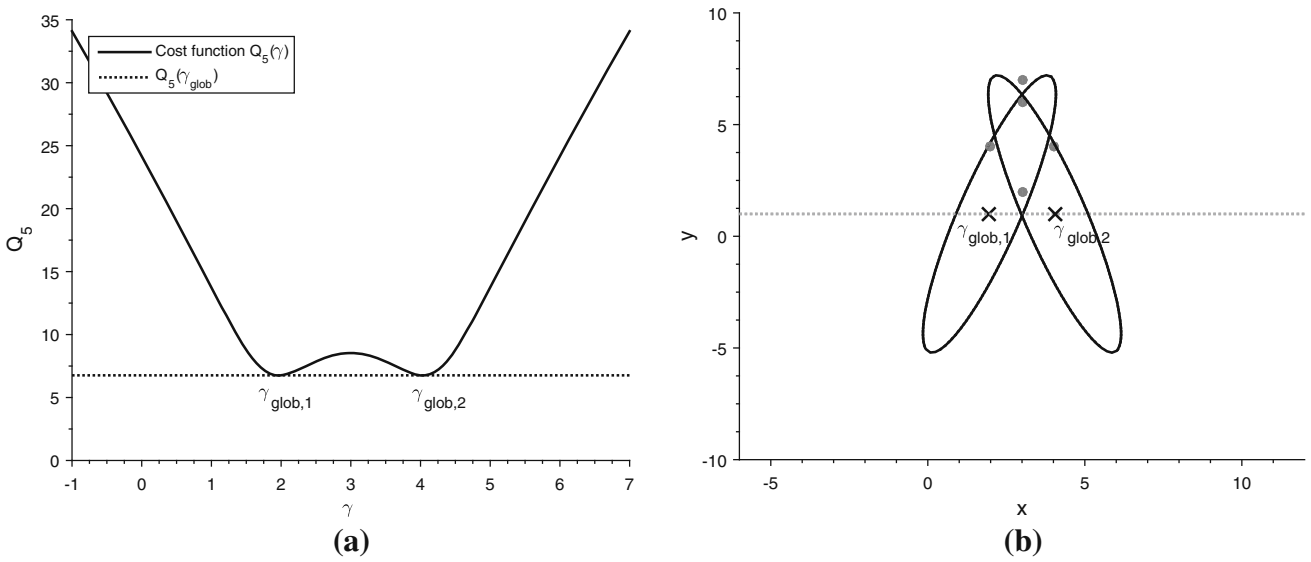


Fig. 6 Cost function $Q_5(\gamma)$ with two global minima in $\gamma_{\text{glob},1}$ and $\gamma_{\text{glob},2}$

(i) $\lambda(\mu) < \lambda(\gamma_4) = \lambda(\gamma_{\text{loc}})$ for $\mu \in (\gamma_4 - \varepsilon, \gamma_4)$ and $\lambda(\mu) > \lambda(\gamma_4) = \lambda(\gamma_{\text{loc}})$ for $\mu \in (\gamma_4, \gamma_4 + \varepsilon)$; this case corresponds to the eventuality that $\frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}\big|_{\gamma_4} > 0$.

(ii) $\lambda(\mu) > \lambda(\gamma_4) = \lambda(\gamma_{\text{loc}})$ for $\mu \in (\gamma_4 - \varepsilon, \gamma_4)$ and $\lambda(\mu) < \lambda(\gamma_4) = \lambda(\gamma_{\text{loc}})$ for $\mu \in (\gamma_4, \gamma_4 + \varepsilon)$; this case corresponds to the eventuality that $\frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}\big|_{\gamma_4} < 0$.

Suppose that case (i) holds. Without loss of generality, we may assume that ε is so small that $\gamma_4 + \varepsilon < \gamma'$. Since $\gamma_4 + \varepsilon/2$ lies between γ_4 and $\gamma_4 + \varepsilon$, we have $\lambda(\gamma_4 + \varepsilon/2) > \lambda(\gamma_{\text{loc}})$. On the other hand, as $\gamma_2 + \varepsilon < \gamma'$, we also have $\gamma_4 + \varepsilon/2 < \gamma'$. Now the inequalities $\lambda(\gamma_4 + \varepsilon/2) > \lambda(\gamma_{\text{loc}})$ and $\lambda(\gamma') < \lambda(\gamma_{\text{loc}})$ imply that there exists a $\gamma'' \in (\gamma_2 + \varepsilon/2, \gamma')$ such that $\lambda(\gamma'') = \lambda(\gamma_{\text{loc}})$. Being larger than γ_4, γ'' is different from γ_3, γ_4, and γ_{loc}. But this contradicts the fact that $\lambda^{-1}(\lambda(\gamma_{\text{loc}})) = \{\gamma_{\text{loc}}, \gamma_3, \gamma_4\}$. This proves the assertion when (i) holds.

Suppose now that case (ii) holds. With no loss of generality, we may assume that ε is so small that $\gamma_3 < \gamma_4 - \varepsilon$. Since $\gamma_4 - \varepsilon/2$ lies between $\gamma_4 - \varepsilon$ and γ_4, we have $\lambda(\gamma_4 - \varepsilon/2) > \lambda(\gamma_{\text{loc}})$. Taking into account that γ_3 is a non-stationary point and that assertion (b) holds, we infer that there exists a $\varepsilon' > 0$ such that $\lambda(\varsigma) < \lambda(\gamma_3) = \lambda(\gamma_{\text{loc}})$ for $\varsigma \in (\gamma_3, \gamma_3 + \varepsilon')$. Without loss of generality, we may assume that ε' is so small that $\gamma_3 + \varepsilon' < \gamma_4 - \varepsilon/2$. Since $\gamma_3 + \varepsilon'/2$ lies between γ_3 and $\gamma_3 + \varepsilon'$, we have $\lambda(\gamma_3 + \varepsilon'/2) < \lambda(\gamma_{\text{loc}})$. Clearly, $\gamma_3 + \varepsilon'/2 < \gamma_4 - \varepsilon/2$ and the inequalities $\lambda(\gamma_3 + \varepsilon'/2) < \lambda(\gamma_{\text{loc}})$ and $\lambda(\gamma_4 - \varepsilon/2) > \lambda(\gamma_{\text{loc}})$ hold. This implies the existence of a $\gamma'' \in (\gamma_3 + \varepsilon'/2, \gamma_4 - \varepsilon/2)$ such that $\lambda(\gamma'') = \lambda(\gamma_{\text{loc}})$. Of course, γ'' satisfies $\gamma_3 < \gamma'' < \gamma_4$. But again this contradicts the fact that $\lambda^{-1}(\lambda(\gamma_{\text{loc}})) = \{\gamma_{\text{loc}}, \gamma_3, \gamma_4\}$. This shows that case (ii) does not occur and finishes the proof of assertion (c).

Now, it follows from (a), (b), and (c) that if γ is such that $\lambda(\gamma) < \lambda(\gamma_{\text{loc}})$, then necessarily $\gamma_3 < \gamma < \gamma_4$. That such γ exists follows from the fact in assertion (c), that (ii) never holds, or equivalently, that (i) always holds. This implies that γ_{glob} falls into (γ_3, γ_4).

Case 2 $\gamma_3 < \gamma_4 < \gamma_{\text{loc}}$.

The analysis in this case is analogous to that in case 1. The end result is that the inequality $\gamma_3 < \gamma_{\text{glob}} < \gamma_4$ continues to hold.

Case 3 $\gamma_3 < \gamma_{\text{loc}} < \gamma_4$ (Fig. 7b).

In this case, $\frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}\big|_{\gamma_3} > 0$ and $\frac{\mathrm{d}\lambda}{\mathrm{d}\gamma}\big|_{\gamma_4} < 0$ hold. This means there exists a $\gamma' \in (-\infty, \gamma_3)$ with $\lambda(\gamma') < \lambda(\gamma_{\text{loc}})$. From Lemma 6 $\lim_{\gamma \to -\infty} \lambda(\gamma) = \infty$ holds. This implies the existence of a $\gamma'' \in (-\infty, \gamma')$ such that $\lambda(\gamma'') = \lambda(\gamma_{\text{loc}})$. Analogously, a $\gamma''' \in (\gamma_4, \infty)$ with $\lambda(\gamma''') = \lambda(\gamma_{\text{loc}})$ can be found. This contradicts the fact that $\lambda^{-1}(\lambda(\gamma_{\text{loc}})) = \{\gamma_{\text{loc}}, \gamma_3, \gamma_4\}$ from which follows that this case does not occur. □

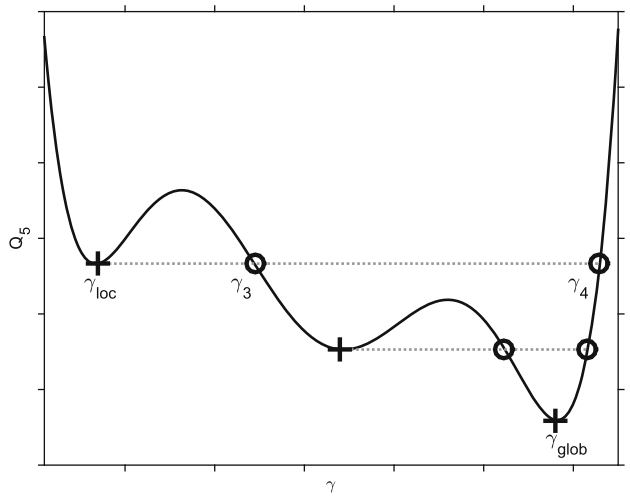
2.5.2 Tunneling Step

When the one-dimensional iterative algorithm finds a minimizer γ_{loc}, then the Tunneling Theorem can be used to decide whether γ_{loc} is the global minimizer or not. If not, then new (better) initial values γ_{new} for the iteration can be found using the theorem.

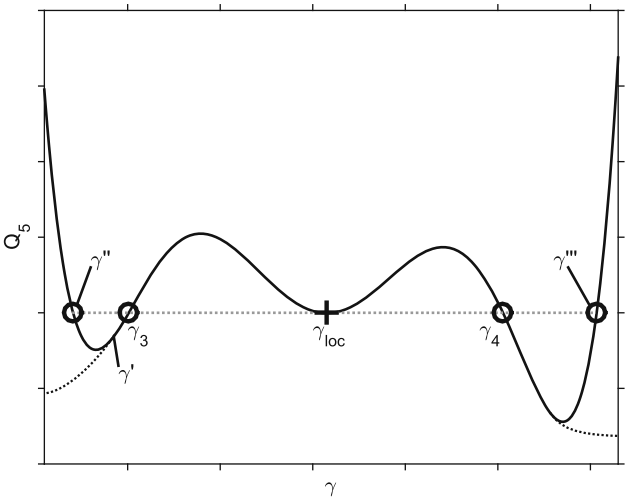
According to the theorem, the roots of the quartic polynomial $r(\gamma, \lambda_{\text{loc}})$ have to be examined. These roots are eigenvalues of the quadratic eigenvalue problem (QEP)

$$\left(\gamma^2 \boldsymbol{C}_2 + \gamma \boldsymbol{C}_1 + \boldsymbol{C}_0 - \lambda_{\text{loc}} \boldsymbol{B}\right) \boldsymbol{v} = \boldsymbol{0}_3. \tag{55}$$

A standard way to solve the QEP is to construct a linearization into a generalized eigenvalue problem (GEP). In the first step,



(a) CASE 1



(b) CASE 3

Fig. 7 CASE 1 and CASE 3 of the Tunneling Theorem proof for the scenario when two distinct real roots appear. Both functions are just for illustration purposes and do not coincide with real cost functions. CASE 1 depicts the scenario $\gamma_{\text{loc}} < \gamma_3 < \gamma_4$. For all γ between γ_3 and γ_4, $Q_5(\gamma) < Q_5(\gamma_{\text{loc}})$ holds. Theoretically, the tunneling step could be repeatedly done to reach the global minimum. However, in Theorem 3 it will be shown that the number of possible minima is limited. CASE 3 depicts the scenario $\gamma_3 < \gamma_{\text{loc}} < \gamma_4$. Tunneling would be impossible here. Fortunately, this case cannot occur in real cost functions due to the impossible number of real roots. The dotted function progression for which the number of real roots would be valid is not possible since $Q_5(\gamma)$ is a coercive function

the substitution $\boldsymbol{w} = \gamma \boldsymbol{v}$ rewrites (55) in

$$\gamma \boldsymbol{C}_2 \boldsymbol{w} + \boldsymbol{C}_1 \boldsymbol{w} + (\boldsymbol{C}_0 - \lambda_{\text{loc}} \boldsymbol{B})\, \boldsymbol{v} = \boldsymbol{0}_3. \tag{56}$$

Merging $\boldsymbol{w} = \gamma \boldsymbol{v}$ and (56) in a block matrix notation yields

$$\begin{bmatrix} \boldsymbol{0}_{3\times3} & \boldsymbol{I}_3 \\ \boldsymbol{C}_0 - \lambda_{\text{loc}} \boldsymbol{B} & \boldsymbol{C}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{w} \end{bmatrix} = \gamma \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & -\boldsymbol{C}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{w} \end{bmatrix}. \tag{57}$$

This GEP has six eigenvalues, from which two are at infinity, since $r(\gamma, \lambda_{\text{loc}})$ is only a polynomial of fourth degree by Lemma 4. Two of the remaining four eigenvalues are identically with γ_{loc} because of Lemma 5. Following the Tunneling Theorem, the global minimum ($\gamma_{\text{glob}} = \gamma_{\text{loc}}$) has been achieved when the last two eigenvalues are complex or define another double root (the latter denotes another global minimizer with the same minimum). If there are two distinct real eigenvalues (γ_3, γ_4), they create a new interval for a search algorithm or they can be used for a new proper initial value

$$\gamma_{\text{new}} = \frac{\gamma_3 + \gamma_4}{2} \tag{58}$$

which yields a lower value of the cost function.

Since in contrast to $r(\gamma, \lambda_{\text{loc}})$, the original cost function $Q_5(\gamma)$ is not a quartic polynomial, a statement on the number of local minima of the cost function cannot be given so far. Therefore, the question how often the tunneling step has to be applied in our problem will be addressed in the next section.

2.5.3 Number of Local Minima

An explicit representation $\lambda(\gamma)$ is not available. That is why we have to analyze the bivariate polynomial deg $r(\lambda, \gamma)$ which possesses at most twelve double roots. This can be deduced using the Bézout-Theorem about the multi-degree of deg $r(\lambda, \gamma) = 4$ and deg $r_\gamma(\lambda, \gamma) = 3$. However, in doing so a limitation of the number of minima is not possible. Instead, another strategy is presented here.

Theorem 3 *If the data condition from Sect. 2.3 holds, then the problem*

$$\underset{\gamma}{\text{minimize}}\ Q_5(\gamma) \tag{59}$$

possesses at most two local minimizers.

Proof Assume without loss of generality that the position vector $\boldsymbol{p}$ is the zero vector and that $\gamma_{\text{glob}} = 0$ holds. This assumption is feasible due to the translation invariance of our geometric problem. The proof will be done by contradiction, in particular it will be shown that it is impossible to have three or more minimizers. Assume we have three distinct minimizers which are located at $\gamma_{\text{loc},1} = \gamma_{\text{glob}} = 0$ as well as $\gamma_{\text{loc},2}$ and $\gamma_{\text{loc},3}$. Any of these local minimizers fulfills the equation

$$\left(\gamma^2 \boldsymbol{\theta}^T \boldsymbol{C}_2 \boldsymbol{\theta} + \gamma \boldsymbol{\theta}^T \boldsymbol{C}_1 \boldsymbol{\theta} + \boldsymbol{\theta}^T \boldsymbol{C}_0 \boldsymbol{\theta}\right)\Big|_{\text{loc},i} = \lambda_{\text{loc},i}. \tag{60}$$

If the data are scaled with a factor α, then $\boldsymbol{D}_{0,\text{new}} = \alpha^2 \boldsymbol{D}_0$ and $\boldsymbol{D}_{1,\text{new}} = \alpha \boldsymbol{D}_1$ imply $\boldsymbol{C}_{0,\text{new}} = \alpha^4 \boldsymbol{C}_0$, $\boldsymbol{C}_{1,\text{new}} = \alpha^3 \boldsymbol{C}_1$, and $\boldsymbol{C}_{2,\text{new}} = \alpha^2 \boldsymbol{C}_2$. Inserting the $\boldsymbol{C}_{i,\text{new}}$ in (60) shows

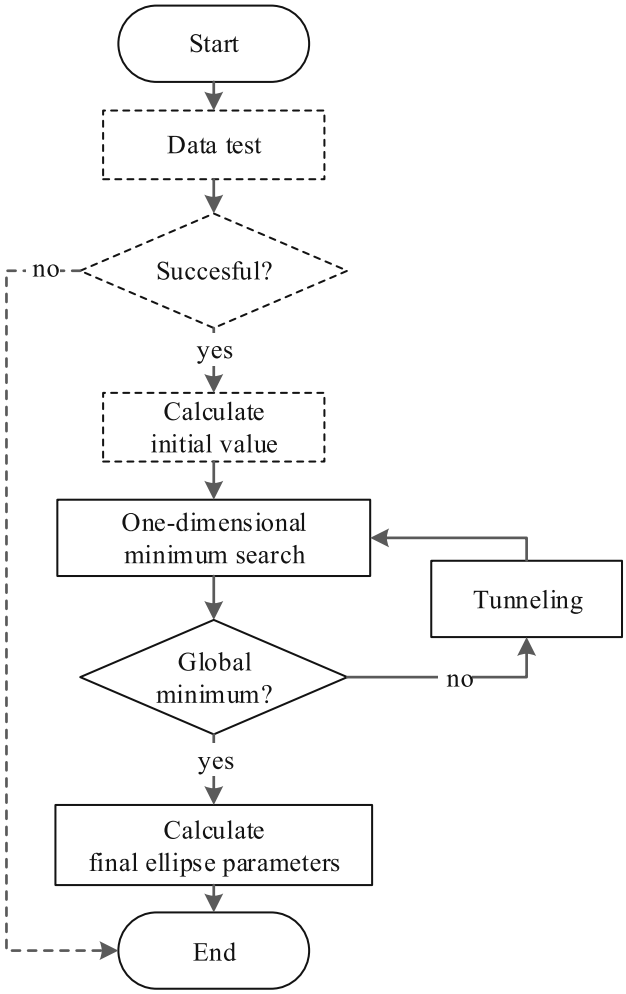


Fig. 8 Flow chart of algorithm (optional steps are marked with *dashed lines*)

that $\gamma_{\mathrm{loc},i,\mathrm{new}} = \alpha\gamma_{\mathrm{loc},i}$ and $\lambda_{\mathrm{loc},i,\mathrm{new}} = \alpha^4\lambda_{\mathrm{loc},i}$ fulfill (60) for the scaled data. If now $\alpha \to 0$ then $\gamma_{\mathrm{loc},2} \to 0$ and $\gamma_{\mathrm{loc},3} \to 0$ while $\gamma_{\mathrm{loc},1} = 0$ holds for all α. Because $Q_5(\gamma)$ is a continuous function all three minimizers fall together in zero. By Lemma 5, any of the minimizers is a double root of $r(\gamma, \lambda_{\mathrm{loc},i,\mathrm{new}})$. Because $\lambda_{\mathrm{loc},i,\mathrm{new}} \to 0$ for $\alpha \to 0$ we have $r(\gamma, 0)$ is a polynomial of sixth degree. This contradicts Lemma 4 after what $r(\gamma, \lambda)$ is a polynomial of fourth degree for any fixed λ. □

3 Numerical Algorithm

In Fig. 8, the principle of the algorithm is outlined. Details on the implementation of the algorithm such as required modifications are given in Sect. 3.1. Depending on the one-dimensional search algorithm, the step for calculating the initial values can be omitted. In our code, we use the MATLAB function `fminbnd` which is based on a golden section search and a parabolic interpolation [5,10]. Here, an interval for γ has to be specified. For other applications that cannot use MATLAB code, the determination of good initial values is required. For this see Sect. 3.2. Hints for a microcontroller implementation of the algorithm are given in Sect. 3.3. The special case of using a line segment instead of a line as prior knowledge and thereby the required treatment of boundaries are discussed in Sect. 3.4. Section 3.5 provides two alternative algorithms that allow for the solution of the constrained ellipse fitting problem.

3.1 Implementation Details

A MATLAB implementation of our constrained ellipse fitting method is given in Fig. 9. In order to focus on the essential part of the implementation, shortenings of the code have been conducted. The checking of the data conditions (see Sect. 2.3) is not considered in this presentation. Additionally, in the case of two global minima, only one out of them is returned and the treatment of the boundaries of the given line segment is omitted. The code can easily be modified to perform a data conditions test, to handle multiple minima (by storing the minima inside the `while`-loop) as well as to handle the additional cases of the line segment restriction (see Sect. 3.4). A download of the complete implementation is possible at http://sourceforge.net/projects/constrainedellipsefitting/.

3.1.1 Modified Evaluation of the Cost Function

For the calculation of $Q_5(\lambda)$, the GEP (31) can be transformed into an ordinary EVP. For this purpose, the data matrices $\boldsymbol{C}_0, \boldsymbol{C}_1, \boldsymbol{C}_2$ are pre-multiplied once only by the inverse of the regular matrix $\boldsymbol{B}$ [13].

3.1.2 Modified Tunneling

For the tunneling step, all roots γ_{loc} of (55) have to be calculated. In (57), we presented the companion form linearization for solving this quadratic eigenvalue problem. In our application, the leading coefficient matrix is singular, which produces eigenvalues at infinity. In addition, the condition numbers of the eigenvalues may be much larger than in the quadratic matrix polynomial. Both numerical difficulties, the infinity issue as well as the increase of the condition number, could be observed in some rare cases. In order to circumvent this issue, we eliminate the eigenvalues at infinity in our implementation before the linearization takes place.

Therefore, the special structure of the matrices in (44) is utilized. With the transformation matrix

```
function [ A, x_c, g ] = fitEllipseWithLineRestriction( x, y, p, q, gammaBound )
epsFmin = 1e-8; epsRootsImag = 1e-6; epsRootsMin = 1e-3;             %Threshold Values
B = [0 0 2; 0 -1 0; 2 0 0];                                          %Constraint Matrix
n = length(x);
Z = eye(n)-ones(n,n)/n;                                              %Centering Matrix
D0 = [(x-p(1)).^2, (x-p(1)).*(y-p(2)), (y-p(2)).^2];
D1 = [2*q(1)*(x-p(1)), q(1)*(y-p(2))+q(2)*(x-p(1)), 2*q(2)*(y-p(2))];
C0 = D0'*Z*D0;
C1 = -D0'*Z*D1 - D1'*Z*D0;
C2 = D1'*Z*D1;
BC0 = B\C0; BC1 = B\C1; BC2 = B\C2;                                  %Premultiply by inv(B)
isGlobalMinimumFound = false;
while ~isGlobalMinimumFound
  [gamma,lambda] = fminbnd(@(gamma) getLambdaOfGamma(gamma,BC0, BC1, BC2),...
      gammaBound(1),gammaBound(2),optimset('TolX',epsFmin));
  P = [1 0 q(2) ^2; 0 1 -2*q(1)*q(2); 0 0 q(1)^2];                   %Transformation Matrix
  C2t = P'*C2*P; C1t = P'*C1*P; C0t = P'*C0*P; Bt = P'*B*P;
  C0t = C0t - lambda*Bt;
  C2b = C2t(1:2,1:2)-C1t(1:2,3)*C1t(3,1:2)/C0t(3,3);
  C1b = C1t(1:2,1:2)-C1t(1:2,3)*C0t(3,1:2)/C0t(3,3)-C0t(1:2,3)*C1t(3,1:2)/C0t(3,3);
  C0b = C0t(1:2,1:2)-C0t(1:2,3)*C0t(3,1:2)/C0t(3,3);
  rootsVec = eig([zeros(2) eye(2); C0b C1b], [eye(2) zeros(2);zeros(2) -C2b]);
  realRootsIdx = find(abs(imag(rootsVec)) < epsRootsImag &...
      abs(rootsVec-gamma)>epsRootsMin);
  if isempty(realRootsIdx)
    isGlobalMinimumFound = true;
  else
    gammaBound(1) = min(rootsVec(realRootsIdx));
    gammaBound(2) = max(rootsVec(realRootsIdx));
  end
  M = BC0+gamma*BC1+gamma^2*BC2;
  [V,D] = eig(M);                                                    %Solve Eigensystem
  idx = real(diag(D))==max(real(diag(D)));                           %Get correct Eigenvalue
  theta = real(V(:,idx));                                            %Get theta
  theta = sign(theta(1))*theta/sqrt(theta'*B*theta);                 %Normalization
  h_opt = -1/n*ones(1,n)*(D0-gamma*D1)*theta;
  A = [theta(1) theta(2)/2; theta(2)/2 theta(3)];                    %Calculate results
  x_c = p+gamma*q;
  g = h_opt - gamma^2*q'*[theta(1) theta(2)/2; theta(2)/2 theta(3)]*q;
end
end

function lambda = getLambdaOfGamma(gamma,BC0, BC1, BC2)
M = BC0+gamma*BC1+gamma^2*BC2;
D = eig(M);
lambda = abs(max(real(D)));
return;
end
```

Fig. 9 MATLAB implementation of constrained ellipse fitting method

$$\boldsymbol{P} = \begin{bmatrix} 1 & 0 & q_2^2 \\ 0 & 1 & -2q_1q_2 \\ 0 & 0 & q_1^2 \end{bmatrix} \tag{61}$$

and

$$\tilde{\boldsymbol{C}}_i = \boldsymbol{P}^T \boldsymbol{C}_i \boldsymbol{P}; i = 0, 1, 2 \tag{62a}$$

$$\tilde{\boldsymbol{B}} = \boldsymbol{P}^T \boldsymbol{B} \boldsymbol{P} \tag{62b}$$

(55) can be written in block matrix notation as

$$\left(\gamma^2 \begin{bmatrix} \tilde{\boldsymbol{C}}_{2,11} & \boldsymbol{0}_2 \\ \boldsymbol{0}_2^T & 0 \end{bmatrix} + \gamma \begin{bmatrix} \tilde{\boldsymbol{C}}_{1,11} & \tilde{\boldsymbol{c}}_{1,12} \\ \tilde{\boldsymbol{c}}_{1,21}^T & 0 \end{bmatrix} + \begin{bmatrix} \tilde{\boldsymbol{C}}_{0,11} & \tilde{\boldsymbol{c}}_{0,12} \\ \tilde{\boldsymbol{c}}_{0,21}^T & c_{0,22} \end{bmatrix} - \lambda_{\text{loc}} \begin{bmatrix} \tilde{\boldsymbol{B}}_{11} & \tilde{\boldsymbol{b}}_{12} \\ \tilde{\boldsymbol{b}}_{21}^T & 0 \end{bmatrix}\right) \begin{bmatrix} \boldsymbol{v}_{1:2} \\ v_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{0}_2 \\ 0 \end{bmatrix}. \tag{63}$$

Here, the first index refers to the underlying matrix according to (62a). The second index denotes the position in the block matrix (see Sect. 1.3).

From the third row of (63), v_3 is given by

$$v_3 = -\gamma \frac{\tilde{\boldsymbol{c}}_{1,21}^T \boldsymbol{v}_{1:2}}{\tilde{c}_{0,22}} - \frac{(\tilde{\boldsymbol{c}}_{0,21}^T - \lambda_{\mathrm{loc}} \tilde{\boldsymbol{b}}_{21}^T)\boldsymbol{v}_{1:2}}{\tilde{c}_{0,22}}. \tag{64}$$

Inserting (64) into (63) yields

$$\left(\gamma^2 \bar{\boldsymbol{C}}_2 + \gamma \bar{\boldsymbol{C}}_1 + \left(\bar{\boldsymbol{C}}_0 - \lambda_{\mathrm{loc}} \bar{\boldsymbol{B}}_{11}\right)\right) \bar{\boldsymbol{v}} = \boldsymbol{0}_2 \tag{65}$$

with

$$\bar{\boldsymbol{v}} = \boldsymbol{v}_{1:2} \tag{66a}$$

$$\bar{\boldsymbol{B}}_{11} = \tilde{\boldsymbol{B}}_{11} \tag{66b}$$

$$\bar{\boldsymbol{C}}_2 = \tilde{\boldsymbol{C}}_{2,11} - \frac{\tilde{\boldsymbol{c}}_{1,12}\tilde{\boldsymbol{c}}_{1,21}^T}{\tilde{c}_{0,22}} \tag{66c}$$

$$\bar{\boldsymbol{C}}_1 = \tilde{\boldsymbol{C}}_{1,11} - \frac{\tilde{\boldsymbol{c}}_{1,12}(\tilde{\boldsymbol{c}}_{0,21}^T - \lambda_{\mathrm{loc}}\tilde{\boldsymbol{b}}_{21}^T)}{\tilde{c}_{0,22}} - \frac{(\tilde{\boldsymbol{c}}_{0,12} - \lambda_{\mathrm{loc}}\tilde{\boldsymbol{b}}_{12})\tilde{\boldsymbol{c}}_{1,21}^T}{\tilde{c}_{0,22}} \tag{66d}$$

$$\bar{\boldsymbol{C}}_0 = \tilde{\boldsymbol{C}}_{0,11} - \frac{\tilde{\boldsymbol{c}}_{0,12}(\tilde{\boldsymbol{c}}_{0,21}^T - \lambda_{\mathrm{loc}}\tilde{\boldsymbol{b}}_{21}^T)}{\tilde{c}_{0,22}} \tag{66e}$$

Along the lines of (56) and (57), from (65) a modified GEP for the calculation of $\gamma_{\mathrm{loc},1} \ldots \gamma_{\mathrm{loc},4}$ can be formulated

$$\begin{bmatrix} \boldsymbol{0}_{2\times2} & \boldsymbol{I}_2 \\ \bar{\boldsymbol{C}}_0 - \lambda_{\mathrm{loc}}\bar{\boldsymbol{B}}_{11} & \bar{\boldsymbol{C}}_1 \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{v}} \\ \bar{\boldsymbol{w}} \end{bmatrix} = \gamma \begin{bmatrix} \boldsymbol{I}_2 & \boldsymbol{0}_{2\times2} \\ \boldsymbol{0}_{2\times2} & -\bar{\boldsymbol{C}}_2 \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{v}} \\ \bar{\boldsymbol{w}} \end{bmatrix}. \tag{67}$$

In contrast to the sixth-order GEP in (57), this modified GEP is only of fourth order and has no infinite solutions.

3.2 Initial Value

Since the optimization problem (17) is non-convex, depending on the applied search method, the selection of a good initial value for γ is essential in order to reduce the number of iterations required to reach the global minimum. In practical applications, prior knowledge can be used to define a reasonable initial value. Two generally valid methods that are solely data driven are presented in the following.

3.2.1 Orthogonal Projection

A simple and fast way to obtain an appropriate initial value γ_0 is averaging all given data points, followed by an orthogonal projection of the mean onto the given line

$$\gamma_0 = \frac{(\overline{\boldsymbol{x}} - \boldsymbol{p})^T \boldsymbol{q}}{\boldsymbol{q}^T \boldsymbol{q}} \quad \text{with} \quad \overline{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i. \tag{68}$$

The advantage of this approach is its fast computation. Nevertheless, if the data points are not distributed along the entire ellipse, this method could yield an initial value, that is located far from the global minimum.

3.2.2 Unconstrained Ellipse Fitting

A more robust variant is given by the solution of the unconstrained ellipse optimization. Here, arbitrary unconstrained ellipse fitting methods e.g., the method presented by Fitzgibbon et al. [9] and Halif et al. [13] can be used. Usually, the resulting unconstrained ellipse is defined by the six parameters a, b, c, d, e, f of a function

$$f_{uc}(\boldsymbol{x}) = ax^2 + bxy + cy^2 + dx + ey + f \tag{69}$$

Using the relations

$$x_c = \frac{2cd - be}{b^2 - 4ac} \quad \text{and} \quad y_c = \frac{2ae - bd}{b^2 - 4ac} \tag{70}$$

the center coordinates $\boldsymbol{x}_c$ of the ellipse can be obtained. As the center of the best unconstrained ellipse is not necessarily located on the given line, an orthogonal projection referred to (68) (using $\boldsymbol{x}_c$ instead of $\bar{\boldsymbol{x}}$) has to be applied. This method also works well, when the data points are disadvantageously located.

3.3 Hints for Microcontroller Implementation

In the proposed method, the calculation of the eigenvalue $\lambda = Q_5(\gamma)$ requires the biggest part of the computation time. In order to reduce this effort, a special Rayleigh quotient iteration for non-symmetric matrices can be applied [4]. This method converges towards the nearest eigenvalue. Since we look for the only positive eigenvalue, a sufficiently large initial value will work.

The four-dimensional GEP for the modified tunneling can be circumvented by calculating the characteristic polynomial of fourth degree with the Faddejew-Leverrier-Algorithm. Applying this method, a quadratic polynomial can be obtained by dividing the original polynomial of fourth degree by the known quadratic polynomial $(\gamma - \gamma_{\mathrm{loc}})^2$. In the resulting polynomial, the two solutions can be analyzed according to Theorem 2.

3.4 Line Segment Case

For the application of the presented one-dimensional minimum search function, a search interval for γ has to be defined

beforehand. This property corresponds to the case of a line segment restriction instead of a line restriction. The boundaries of the line segment need to be considered.

The one-dimensional search algorithm yields a minimizer $\gamma_{\min}$ within the given interval. This minimizer either represents a true local minimum within the given interval or it is located on one of the boundaries. In the first case, the tunneling step could lead to new boundaries outside the given interval. In the latter case, a true local minimum with a lower value of the cost function within the given interval is possible. Both cases require a special treatment.

All potential cases of this interval search problem can be distinguished by regarding the real roots resulting from the tunneling step.

Number of real roots distinct from $\gamma_{\min}$

- No roots
  The search algorithm has converged into the global minimum.
- One root
  The search algorithm reached a boundary and the global minimum is located outside the given interval. (No better solution within given interval exists)
- Two roots (both outside the interval)
  The search algorithm has converged into a local minimum γ_{loc}, which is also the minimum of the cost function inside the given interval. The global minimum is outside the given interval.
- Two roots (one outside and one inside the interval)
  The search algorithm has converged into a local minimum γ_{loc} of the cost function. The new search interval is given by the root inside and the nearest given boundary to the outside root.
- Two roots (both inside the interval)
  The search algorithm has converged into a local minimum γ_{loc} of the cost function. The global minimum is located inside the given search interval and both roots can be deployed as new interval boundaries.
- Three roots
  The search algorithm has converged into one of the boundaries. Here, only two or no other roots inside the given interval are possible. If no other roots are inside the given interval, the boundary is the solution of the interval search problem. Otherwise, the new search interval is given by the two roots that are located inside the given interval.

3.5 Alternative Algorithms

3.5.1 Golden Section Search

In our implementation, we use the `fminbnd` function of MATLAB for the one-dimensional minimum search. If this function is not available, we propose to employ the easily implementable golden section search algorithm [25].

3.5.2 Alternating Least Squares

In contrast to the proposed minimum search algorithm, the following algorithm does not need boundaries for γ. Alternating least squares (ALS) is an iterative algorithm in which the cost function is minimized by alternately keeping one variable fixed in any current step. In our problem (21), the two variables are $\boldsymbol{\theta}$ and γ and the ALS reads as

$$\boldsymbol{\theta}_{k+1} = \underset{\boldsymbol{\theta}^T \boldsymbol{B} \boldsymbol{\theta} = 1}{\operatorname{argmin}} \; Q_4(\gamma_k, \boldsymbol{\theta}) \tag{71a}$$

$$\gamma_{k+1} = \underset{\gamma}{\operatorname{argmin}} \; Q_4(\gamma, \boldsymbol{\theta}_{k+1}) \tag{71b}$$

or using the results in the manner of the proof of Theorem 1 the basic structure of our algorithm writes as follows:

$$\boldsymbol{\theta}^*_{k+1} = \text{EigVec of } \lambda_{\max}(\gamma_k^2 \boldsymbol{C}_2 + \gamma_k \boldsymbol{C}_1 + \boldsymbol{C}_0, \boldsymbol{B}) \tag{72a}$$

$$\gamma_{k+1} = -\frac{\boldsymbol{\theta}^{*T}_{k+1} \boldsymbol{C}_1 \boldsymbol{\theta}^*_{k+1}}{2\boldsymbol{\theta}^{*T}_{k+1} \boldsymbol{C}_2 \boldsymbol{\theta}^*_{k+1}} \tag{72b}$$

(72b) is simply a result of minimizing the quadratic function (30) in γ and the fact that inside the iteration loop a normalization of $\boldsymbol{\theta}$ is not necessary. An advantage of this method is the unboundedness of γ which allows for lines instead of line segments. Additionally, no one-dimensional search algorithm has to be implemented which possibly leads to more compact code. The separation in two variable problems also makes it possible to incorporate auxiliary information. Nevertheless, the computation effort using ALS in MATLAB is four times higher than using the `fminbnd` algorithm presented above.

4 Results and Discussion

In order to evaluate the results of our constrained ellipse fitting approach compared to existing methods, we determine an accuracy measure which is based on the joint (overlap) area of the original underlying ellipse and the fitted ellipse [14]. The accuracy measure m depicts the similarity of the fitted ellipse with the original ellipse from a practical point of view. It can be determined with

$$m = \frac{\text{area(Overlap)}}{\text{area(El}_{\text{fit}})} \cdot \frac{\text{area(Overlap)}}{\text{area(El}_{\text{Orig}})} \tag{73}$$

where area(Overlap) is the area of the overlap, area ($\text{Ellipse}_{\text{fit}}$) represents the area of the fitted ellipse, and area($\text{Ellipse}_{\text{Orig}}$) is the area of the original ellipse. A measure

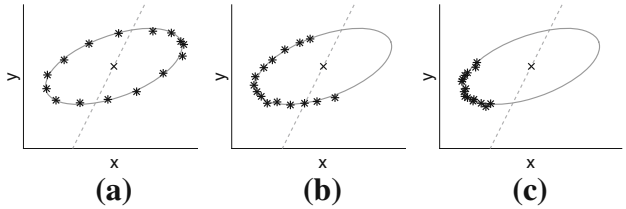


Fig. 10 Different angle ranges of data points: **a** $[0\pi\ \ 2\pi]$, **b** $[0.5\pi\ 1.5\pi]$, $[0.75\pi\ 1.25\pi]$

of 0 arises when both ellipses have no overlapping area and a measure of 1 means a perfect match of the ellipses.

Due to a high number of parameters and influenced by our application scenario presented in Fig. 1, we focused in our set of experiments on the variation of the angle range and the number of data points. From both parameters, we expected the highest influence on the results. We compared the unconstrained approaches from Fitzgibbon et al./Halif et al. [9,13] and Szpak et al. [30] with our constrained ellipse fitting method. The experimental conditions are orientated on the comparison provided in [29]. The data points are generated from an underlying original ellipse and the addition of independent zero-mean Gaussian noise with a standard deviation of 1.5 to the data points. The original ellipse which is the same as in Fig. 2 is described by its center $\boldsymbol{x}_c = \begin{bmatrix}1\ 2\end{bmatrix}^T$, its semi-axis lengths of 7 and 3, and its rotation of $\pi/9$. The line is given by $\boldsymbol{p} = \begin{bmatrix}2\ 4\end{bmatrix}^T$ and $\boldsymbol{q} = \begin{bmatrix}-1\ -2\end{bmatrix}^T$. The examined angle ranges of the data points are $[0\pi\ 2\pi]$, $[0.5\pi\ 1.5\pi]$, and $[0.75\pi\ 1.25\pi]$ (see Fig. 10).

We took 1000 samples of each parameter configuration and determined the accuracy measure m for each sample. The results are shown in Fig. 11.

In all analyzed cases, the constrained ellipse fitting method yields the highest accuracy. This effect is low when data points taken from the entire ellipse's angle range ($[0\ 2\pi]$) are analyzed but it considerably increases when the observed angle range gets narrower.

Besides the accuracy, the calculation time plays an important role for the overall efficiency of an algorithm in practical applications. In Table 1, the mean computation time of a single sample is given for the datasets with 15 data points and for the datasets with 250 data points. We employed an Intel Core i5-2500 3.3 GHz with 12 GB RAM, Windows 7 64 bit, and MATLAB R2013b for the calculations.

The fastest method is the direct ellipse fit by Fitzgibbon et al./Halif et al. [9,13]. The ellipse fitting method from Szpak et al. [30] required the longest computation time and with increasing number of data points this difference to the other

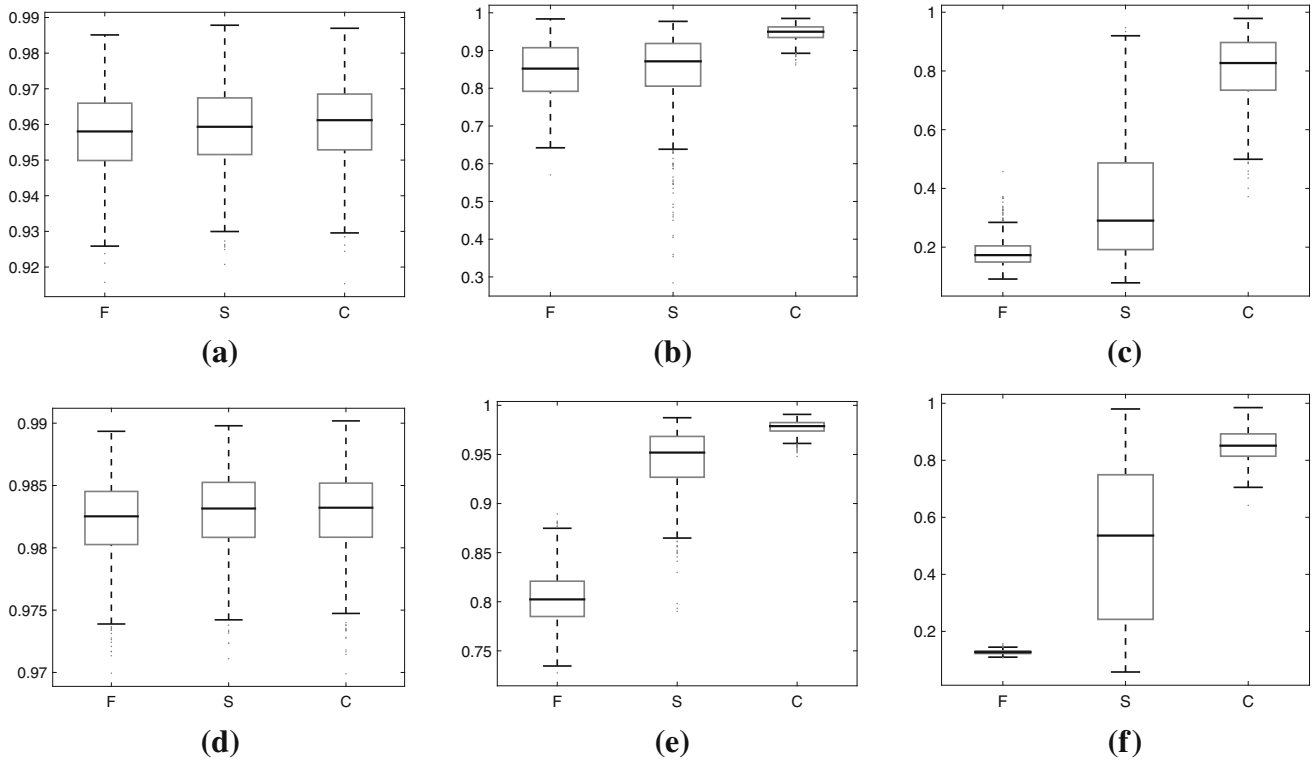


Fig. 11 Comparison of the accuracy m of two unconstrained ellipse fitting methods from Fitzgibbon et al. [9] (F) and Szpak et al. [30] (S) and our new constrained ellipse fitting (C); the boxplots are each based on 1000 samples; for the top three figures **a–c** 15 data points and for the bottom three figures **d–f** 250 data points are used; from left to right the observed angle range decreases from $[0\ 2\pi]$ over $[0.5\ 1.5\pi]$ to $[0.75\ 1.25\pi]$

Table 1 Comparison of computation time

No. of data points	F	S	C
15	0.1 ms	6.7 ms	2.0 ms
250	0.1 ms	85.0 ms	2.8 ms

F Fitzgibbon, *S* Szpak, *C* constrained ellipse fitting

methods increases. The constrained ellipse fitting lies in the middle; however, with increasing number of data points, its computation time increases only slightly.

5 Conclusion

The new constrained ellipse fitting method we presented in this paper allows the incorporation of the prior knowledge, that the ellipse's center is located on a given line into a constrained algebraic cost function. Existing methods are not capable of using this prior knowledge and therefore produce less accurate results when the data points only represent a small part of the underlying ellipse. By deducing and applying several theorems, we reduced the underlying optimization problem into a fast and global convergent one-dimensional minimum search. In addition, the special case of a line segment instead of a line restriction has been treated. Numerical issues and implementation details were discussed and the essential part of our new algorithm was provided as a MATLAB implementation. A download of the complete implementation of the constrained ellipse fitting (including data condition test and boundary treatment) is possible at http://sourceforge.net/projects/constrainedellipsefitting/. Future works comprise the consideration of other distance measures in the constrained ellipse fitting framework.

Acknowledgments The authors thank the anonymous reviewers for their valuable feedback. Especially, we would like to thank for providing parts of the proof for Theorem 2.

References

1. Adams, W., Loustaunau, P.: An introduction to Gröbner bases. Graduate studies in mathematics. Am. Math Soc (1994)
2. Ahn, S.J., Rauh, W., Warnecke, H.J.: Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola. Pattern Recognit. **34**(12), 2283–2303 (2001)
3. Al-Sharadqah, A., Chernov, N.: A doubly optimal ellipse fit. Comput. Stat. Data Anal. **56**(9), 2771–2781 (2012)
4. Batterson, S., Smillie, J.: Rayleigh quotient iteration for nonsymmetric matrices. Math. Comput. **55**(191), 169–178 (1990)
5. Brent, R.P.: Algorithms for Minimization without Derivatives. Prentice-Hall, Englewood Cliffs, NJ (1973)
6. Cheng, Y.C., Lee, S.C.: A new method for quadratic curve detection using k-ransac with acceleration techniques. Pattern Recognit. **28**(5), 663–682 (1995)
7. Chojnacki, W., Brooks, M., van den Hengel, A., Gawley, D.: On the fitting of surfaces to data with covariances. IEEE Trans. Pattern Anal. Mach. Intell. **22**, 1294–1303 (2000)
8. Dave, R.N.: Generalized fuzzy c-shells clustering and detection of circular and elliptical boundaries. Pattern Recognit. **25**(7), 713–721 (1992)
9. Fitzgibbon, A., Pilu, M., Fisher, R.B.: Direct least-squares fitting of ellipses. IEEE Trans. Pattern Anal. Mach. Intell. **21**, 476–480 (1999)
10. Forsythe, G., Malcolm, M., Moler, C.: Computer Methods for Mathematical Computations. Prentice-Hall, Englewood Cliffs, NJ (1976)
11. Gander, W., Golub, G.H., Strebel, R.: Least squares fitting of circles and ellipses. BIT Numer. Math. **34**(4), 558–578 (1994)
12. Guil, N., Zapata, E.: Lower order circle and ellipse hough transform. Pattern Recognit. **30**, 1729–1744 (1997)
13. Halíř, R., Flusser, J.: Numerically stable direct least squares fitting of ellipses. In: Proceedings of the 6th International Conference Computer Graphics and Visualization, vol. 1, pp. 125–132 (1998)
14. Hughes, G.B., Chraibi, M.: Calculating ellipse overlap areas. Comput. Vis. Sci. **15**(5), 291–301 (2012)
15. Jarlebring, E., Kvaal, S., Michiels, W.: Computing all pairs (λ, μ) such that λ is a double eigenvalue of $A + \mu B$. SIAM J. Matrix Anal. Appl. **32**(3), 902–927 (2011)
16. Kanatani, K., Al-Sharadqah, A., Chernov, N., Sugaya, Y.: Renormalization returns: Hyper-renormalization and its applications. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) Computer Vision ECCV 2012. Lecture Notes in Computer Science, vol. 7574, pp. 384–397. Springer, Berlin (2012)
17. Kanatani, K., Rangarajan, P.: Hyper least squares fitting of circles and ellipses. Comput. Stat. Data Anal. **55**(6), 2197–2208 (2011)
18. Lasley, J.W.: On degenerate conics. Am. Math. Mon. **64**(5), 362–364 (1957)
19. Lukacs, G., Martin, R., Marshall, D.: Faithful least-squares fitting of spheres, cylinders, cones and tori for reliable segmentation. In: H. Burkhardt, B. Neumann (eds.) Proceedings of the 5th European Conference on Computer Vision—Volume I, Lecture Notes in Computer Science, vol. 1406, pp. 671–686. Springer, Berlin (1998)
20. Markovsky, I., Kukush, A., Huffel, S.V.: Consistent least squares fitting of ellipsoids. Numerische Mathematik **98**(1), 177–194 (2004)
21. Masuzaki, T., Sugaya, Y., Kanatani, K.: High accuracy ellipse-specific fitting. In: Klette, R., Rivera, M., Satoh, S. (eds.) Image and Video Technology. Lecture Notes in Computer Science, vol. 8333, pp. 314–324. Springer, Berlin (2014)
22. McLaughlin, R.A.: Randomized hough transform: improved ellipse detection with comparison. Pattern Recognit. Lett. **19**(3–4), 299–305 (1998)
23. Morimoto, C.H., Mimica, M.R.: Eye gaze tracking techniques for interactive applications. Comp. Vis. Image Underst. **98**(1), 4–24 (2005)
24. Pope, S.B.: Algorithms for ellipsoids. Cornell University, Ithaca, NY (2008)
25. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes 3rd Edition: The Art of Scientific Computing. Cambridge University Press, Cambridge, MA (2007)
26. Rahayem, M., Werghi, N., Kjellander, J.: Best ellipse and cylinder parameters estimation from laser profile scan sections. Opt. Lasers Eng. **50**(9), 1242–1259 (2012)
27. Ray, N., Acton, S., Ley, K.: Tracking leukocytes in vivo with shape and size constrained active contours. IEEE Trans. Med. Imag. **21**, 1222–1235 (2002)
28. Sage, D., Neumann, F., Hediger, F., Gasser, S., Unser, M.: Automatic tracking of individual fluorescence particles: application to

the study of chromosome dynamics. IEEE Trans. Med. Imag. **14**, 1372–1383 (2005)
29. Szpak, Z., Chojnacki, W., van den Hengel, A.: A comparison of ellipse fitting methods and implications for multiple-view geometry estimation. In: 2012 International Conference on Digital Image Computing Techniques and Applications (DICTA), pp. 1–8 (2012)
30. Szpak, Z., Chojnacki, W., van den Hengel, A.: Guaranteed ellipse fitting with the sampson distance. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) Computer Vision ECCV 2012. Lecture Notes in Computer Science, vol. 7576, pp. 87–100. Springer, Berlin (2012)
31. Waibel, P., Matthes, J., Gröll, L., Keller, H.: A structure from motion approach for the analysis of adhesions in rotating vessels. In: 2014 International Conference on 3D Vision—3DV 2014, p. to appear (2014)
32. Yu, J., Kulkarni, S.R., Poor, H.V.: Robust ellipse and spheroid fitting. Pattern Recognit. Lett. **33**(5), 492–499 (2012)
33. Zhang, S.C., Liu, Z.Q.: A robust, real-time ellipse detector. Pattern Recognit. **38**(2), 273–287 (2005)
34. Zhang, Z.: Parameter estimation techniques: a tutorial with application to conic fitting. Image Vis. Comput. **15**(1), 59–76 (1997)

Patrick Waibel received the Dipl.-Ing. degree from the Technical University of Karlsruhe, Germany in electrical engineering in 2007. In 2014 he received the PhD from the Karlsruhe Institute of Technology (KIT), Germany where he works since 2008 as a research staff member at the Institute for Applied Computer Science. His research interests include computer vision, signal processing and process analysis.

Jörg Matthes received the Dipl.-Ing. degree in mechatronics from the Technical University of Freiberg, Germany in 1999 and the Dr.-Ing. degree from the Technical University of Karlsruhe, Germany in 2004. Since 2000 he is a research staff member at the Institute for Applied Computer Science, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany. His research interests are in computer vision and camera based process control.

Lutz Gröll received the Dipl.-Ing. and Dr.-Ing. degrees from the Technical University Dresden, Dresden, Germany, in 1985 and 1995, both in electrical engineering. Since 2001 he has been the head of the research group "Process Modelling and Control Engineering" at the Institute for Applied Computer Science, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany. His research interests are in parameter identification, nonlinear control, and optimization theory.