Journal of Structural Geology 31 (2009) 546–560

Contents lists available at ScienceDirect

Journal of Structural Geology

journal homepage: www.elsevier.com/locate/jsg

# Optimized trishear inverse modeling

Nestor Cardozo[a,*], Sigurd Aanonsen[b]

[a] Department of Petroleum Engineering, University of Stavanger, 4036 Stavanger, Norway
[b] StatoilHydro, Postboks 7200, 5020 Bergen, Norway

ARTICLE INFO

*Article history:*
Received 11 June 2008
Received in revised form
13 February 2009
Accepted 2 March 2009
Available online 13 March 2009

*Keywords:*
Trishear
Inverse modeling
Optimization
Uncertainty

ABSTRACT

We describe the application of optimization methods to 2D and 3D trishear inverse modeling. These optimization methods traverse the parameter space in search for the combination of trishear parameters that best restores a fold profile to a straight line in 2D, or a folded surface to a plane in 3D. Optimized trishear inverse modeling is significantly faster than the standard grid-search method. An optimized search of all six parameters of a 2D trishear model can be conducted in seconds. Gradient-based optimization methods converge to the solution much faster than the simplex optimization method, and therefore are the preferred choice for 2D or 3D inversions involving a large number of parameters. The optimization algorithms, however, are affected by local minima. This is particularly an issue in the inversion of actual fold data, where local minima can occur even in a parameter space of low dimensions. Local minima can be assessed by running several optimizations from different regions in the parameter space. Physically meaningless local minima can be avoided by constraining the optimization. Fast, optimized trishear inverse modeling makes possible within hours, heavy tasks such as estimating the uncertainties of the best-fit parameters by the inversion of hundreds of fold realizations. We illustrate this technique using synthetic fold data and an actual fault-propagation fold from the Los Angeles Basin, the Santa Fe Springs anticline.

## 1. Introduction

Trishear is a kinematic model of fault-propagation folding in which the decrease in displacement along the fault is accommodated by heterogeneous shear in a triangular zone radiating from the fault tip (Erslev, 1991; Allmendinger, 1998). Trishear explains features that cannot be explained by the self-similar, parallel kink fold model (Suppe and Medwedeff, 1990), such as (i) Changes in stratigraphic thickness and dip of fold forelimbs, (ii) Footwall synclines, (iii) Rounded and angular fold hinges, and (iv) Fold geometry and strain that change with proximity to the fault.

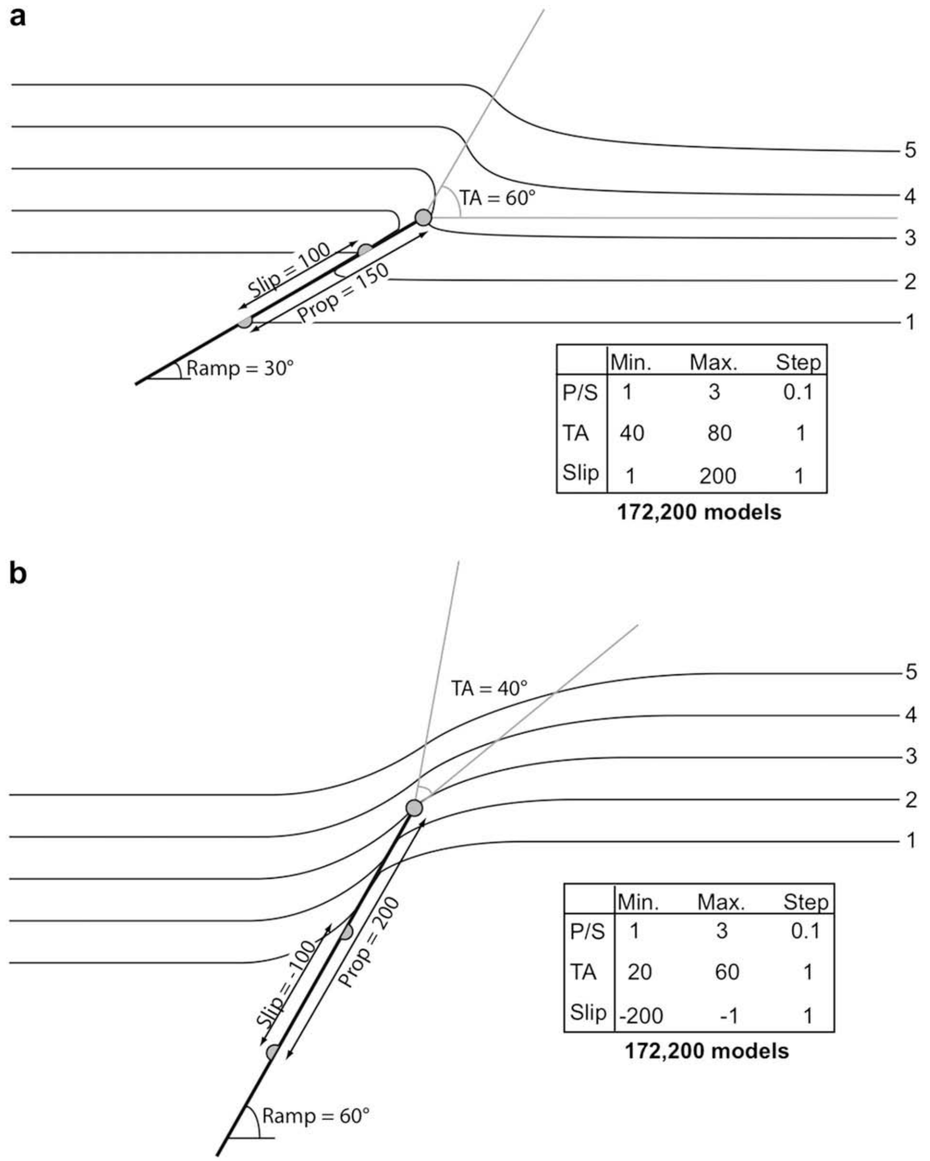
Compared to the kink fold model, however, trishear presents a fundamental problem in the inverse modeling of real fault-propagation folds. Trishear must be applied numerically (i.e. incrementally) rather than graphically. Knowledge of the fold geometry relies on knowledge of its incremental evolution. Allmendinger (1998) devised a solution to this inverse problem. Since trishear can be run backwards, he implemented a grid-search algorithm that finds the combination of trishear parameters that best restores a 2D fold profile to a straight line. Cardozo (2005) extended this strategy to the situation of beds outcropping along a topographic profile. In this case, the grid-search algorithm finds the combination of trishear parameters that best reproduces the attitude and spacing of the beds intersections along the topographic profile.

Allmendinger (1998) and Cardozo (2005) are robust solutions for trishear inverse modeling. Grid searching the parameter space (which can be up to six dimensions in 2D), however, is inefficient. In this paper, we describe the use of several optimization algorithms that speed up trishear inverse modeling. Rather than systematically searching the parameter space for a minimum, the optimization algorithms traverse the parameter space in ideal directions towards a minimum, thus reducing the time of estimation of the best-fit parameters. Using the optimization algorithms, a search of all six parameters of a 2D trishear model can be conducted in seconds, as opposed to the hours it can take using the grid-search algorithm.

An optimized parameter estimation is a prerequisite for 3D trishear inverse modeling, where the parameter space can be up to fourteen dimensions. We show the application of the optimization algorithms to 3D trishear inverse modeling. By analogy with 2D trishear inverse modeling (Allmendinger, 1998), in 3D the optimization algorithms search for the combination of parameters that best restores a folded surface to a plane. This strategy works with either pseudo or true-3D trishear models (Cardozo, 2008), and

* Corresponding author. Tel.: +47 5183 2391.
*E-mail addresses:* nestor.cardozo@uis.no, nfcd@mac.com (N. Cardozo).

0191-8141/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jsg.2009.03.003



|  | Min. | Max. | Step |
|---|---|---|---|
| P/S | 1 | 3 | 0.1 |
| TA | 40 | 80 | 1 |
| Slip | 1 | 200 | 1 |

172,200 models

|  | Min. | Max. | Step |
|---|---|---|---|
| P/S | 1 | 3 | 0.1 |
| TA | 20 | 60 | 1 |
| Slip | -200 | -1 | 1 |

172,200 models

**Fig. 1.** 2D forward trishear models and their key parameters, a. Reverse fault, b. Normal fault. Bed 3 in a and b was fed into the grid-search (Figs. 2 and 3) and optimized search (Figs. 4 and 5 and Table 1) procedures to test the accuracy of these techniques. Tables in a and b show the range of values tested in the grid-search procedure.

allows the rapid estimation of the best-fit parameters and their variation along fault strike.

Fast trishear inverse modeling makes more feasible statistical analyses for estimating the uncertainties of the best-fit parameters. We describe a methodology for generating synthetic fold profiles (i.e. realizations) from a measured fold profile with uncertainties in location. Optimized trishear inverse modeling of these realizations gives a set of simulated, best-fit parameters, from which confidence limits on the best-fit parameters can be estimated.

## 2. The problem

The basic problem in trishear inverse modeling is how to determine the model (i.e. the combination of model parameters) that best fits a folded profile in 2D or surface in 3D. In kink fold models, one can measure dip panels and use a set of rules (Suppe, 1983; Suppe and Medwedeff, 1990) to estimate the appropriate parameters. In trishear, however, there are no rules (i.e. mathematical relations) to estimate model parameters such as fault propagation to slip ratio (P/S) or trishear angle (apical angle of triangular zone) from the fold geometry. Trishear inverse modeling is a trial and error process in which a group of models, systematically or randomly chosen, is compared to the data, and the model with the closest agreement to the data is chosen as the solution. The agreement between the data and the model with a particular choice of parameters is measured by an objective function (or merit function) that by convention is small when the agreement is good (Press et al., 1992).

Since trishear can be run forwards or backwards, the objective function can be designed to measure either how well a model restores folded beds, or how well a model deforms beds from their undeformed configuration. The first approach works well when the geometry of at least one bed is known across the structure. In this case and in 2D, one can identify the best model as the one that best restores a key folded bed to a straight line (Allmendinger, 1998). The second approach is more convenient when the information about the fold is limited to a small window along a topographic profile. In this case, the best model is the one that deforms the beds from their initial configuration, such that their intersections with

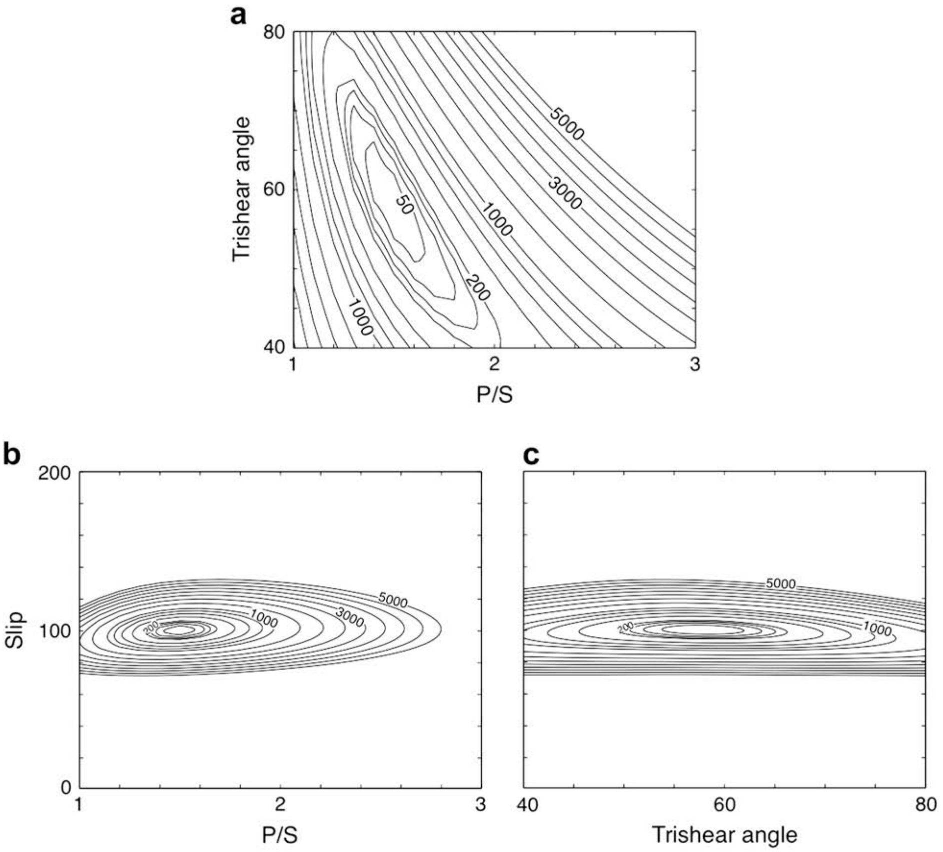


**Fig. 2.** 2D slices through the 3D matrix of objective function values produced by grid searching the forward reverse fault model over the range of values shown in Fig. 1a. Slices are constructed at a. Best fault slip (100 units), b. Best trishear angle (58°), and c. Best P/S (1.5). In each diagram, contours are objective function values. Contours below 200 are in intervals of 50, from 200 to 1000 are in intervals of 200, and above 1000 are in intervals of 500.

the topographic profile are the closest in agreement with the measured beds intersections (Cardozo, 2005).

We limit our discussion to the first, “restoration” approach, partly because its statistical description (objective function) is simpler. However, there is nothing that prevents the optimization algorithms discussed here from being applied to the second, “deformation” approach. For the restoration approach, the objective function (*f*) in 2D is expressed as:

$$f = \sum_{i=1}^{N} (y_i - a - bx_i)^2 \tag{1}$$

where $x_i$ and $y_i$ are the cartesian coordinates of the *N* points on the restored fold profile, and **a** and *b* are the coefficients of the linear regression of the restored fold profile (i.e. intercept and slope of the line that best fits the restored profile). Trishear inverse modeling is thus a least squares minimization problem (Press et al., 1992). *f* is referred by Allmendinger (1998) as the chi-square ($\chi^2$). We prefer avoiding this terminology because uncertainties are not included in Eq. (1).

## 3. Grid-search

In the grid-search algorithm, the parameter space is systematically searched within ranges and step sizes specified by the user. Grid-search is a brute force strategy where every parameter combination in the search matrix defined by the user is evaluated, and the combination with the lowest value of the objective function is identified as the solution. The advantage of the grid-search algorithm is that it is not affected by local minima. If the step sizes are small enough, the solution corresponds to the lowest value of the objective function within the searched parameter space. The disadvantage of the grid-search algorithm is that it is time consuming. Grid searching for all six parameters of a 2D trishear model, within realistic ranges and step sizes, involves testing millions of models that take hours to run, even in the fastest computer. Grid searching the parameter space of a 3D trishear model, which can be up to fourteen dimensions, is not practical.

To illustrate the performance of the grid-search algorithm, we use two forward 2D trishear models (Fig. 1): one for a reverse (Fig. 1a), and one for a normal (Fig. 1b) fault. The models were constructed using the simplest, linear velocity field for a symmetric trishear zone (Zehnder and Allmendinger, 2000; their Eq. (6) with $s = 1$). Modeling a synthetic fold geometry generated with known parameters might seem circular, but is the only situation for which we can evaluate with complete confidence the success of an inverse algorithm. This is because synthetics are true representations of the model, and unlike actual folds, they don’t have errors. In both, the reverse (Fig. 1a) and normal (Fig. 1b) fault cases, bed 3 was used for the inversion, the location of the fault tip (*x* and *y*) and ramp angle were fixed, and the P/S, trishear angle (TA) and fault slip were searched over the ranges and step sizes shown in Fig. 1 (172,200 models were evaluated in each case).

Figs. 2 and 3 show the results of the grid-search for the reverse (Fig. 2) and normal (Fig. 3) fault models of Fig. 1. The statistics are illustrated as 2D slices through the 3D matrix of objective function (Eq. (1)) values produced by the grid-search. In both, the reverse (Fig. 2) and normal (Fig. 3) fault models: (i) the best-fit parameters are very close to the parameters of the forward model, (ii) there are no local minima in the parameter space, (iii) there is an inverse correlation between P/S and trishear angle

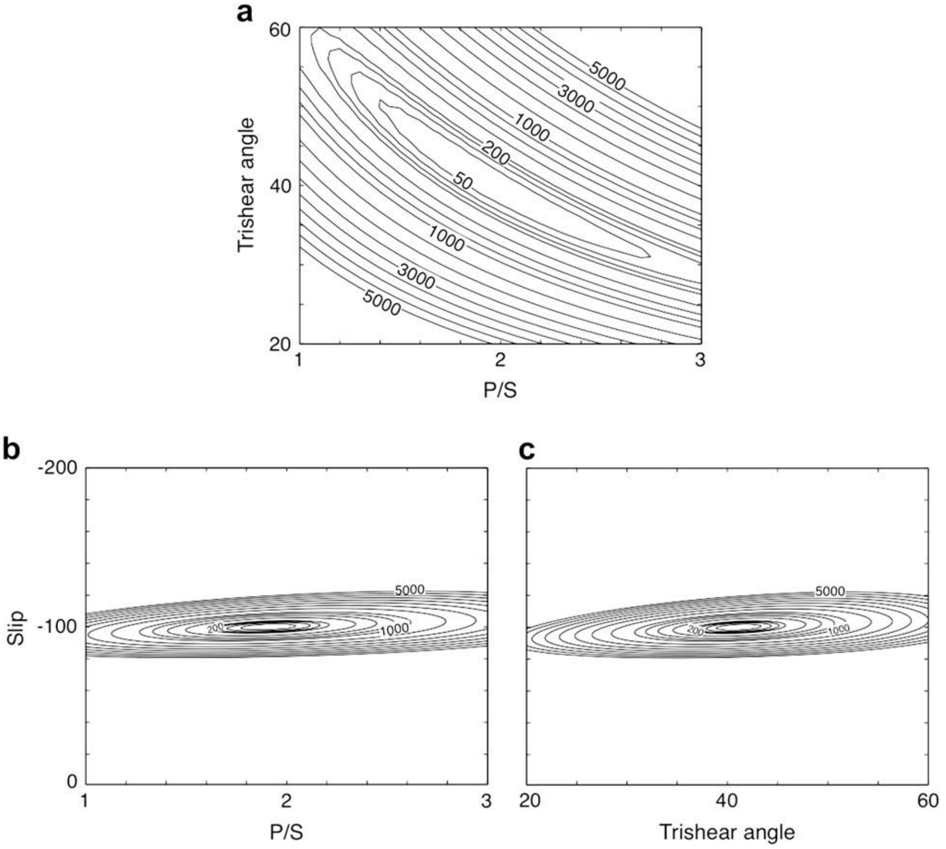


**Fig. 3.** 2D slices through the 3D matrix of objective function values produced by grid searching the forward normal fault model over the range of values shown in Fig. 1b. Slices are constructed at a. Best fault slip (−100 units), b. Best trishear angle (41°), and c. Best P/S (1.9). In each diagram, contours are objective function values. Contours below 200 are in intervals of 50, from 200 to 1000 are in intervals of 200, and above 1000 are in intervals of 500.

(Figs. 2a and 3a; Allmendinger et al., 2004), and (iv) the solution is well constrained in fault slip (Figs. 2b, c and 3b, c). Interestingly, the solution is more constrained in P/S than in trishear angle in the reverse fault model, where a 0.2 change in P/S close to the minimum corresponds to about 15° change in trishear angle (Fig. 2a; Allmendinger et al., 2004). In the normal fault model, on the contrary, a 0.2 change in P/S close to the minimum corresponds to just about 5° change in trishear angle (Fig. 3a). This can be explained by the different orientations of the triangular zone of shear with respect to bedding in the models. In the normal fault model (Fig. 1b) the triangular zone of shear traverses faster the stratigraphy than in the reverse fault model (Fig. 1a).

**Table 1**
Optimized searches for the parameters **a** that best fit the forward reverse fault model of Fig. 1a. $\mathbf{a_0}$ is the initial guess, and $\mathbf{a_f}$ the best-fit parameters. $\mathbf{a_{min}}$ and $\mathbf{a_{max}}$ are lower and upper bounds for fmincon.

| Function | **a** = [RA, P/S, TA, Slip][a] | **a** = [xt, yt, RA, P/S, TA, Slip][b] |
|---|---|---|
| fminsearch | $\mathbf{a_0}$ = [35, 2.5, 75, 175]<br>$\mathbf{a_f}$ = [29.7, 1.51, 58.5, 101]<br>Iterations = 326 | $\mathbf{a_0}$ = [400, 100, 35, 2 5, 75, 125]<br>$\mathbf{a_f}$ = [428, 73, 25.7, 1 5, 57.1, 1, 115]<br>Iterations = 659 |
| fminunc | $\mathbf{a_0}$ = [35, 2.5, 75, 175]<br>$\mathbf{a_f}$ = [29.7, 1.51, 58.5, 101]<br>Iterations = 42 | $\mathbf{a_0}$ = [400, 100, 35, 2.5, 125]<br>$\mathbf{a_f}$ = [452, 87, 27.3, 2.2, 46.4, 111]<br>Iterations = 17<br>Number of function evaluations exceeded |
| fmincon | $\mathbf{a_{min}}$ = [20, 1, 40, 0]<br>$\mathbf{a_{max}}$ = [40, 3, 80, 200]<br>$\mathbf{a_0}$ = [35, 2.5, 75, 175]<br>$\mathbf{a_f}$ = [29.7, 1.51, 58.5, 101]<br>Iterations = 21 | $\mathbf{a_{min}}$ = [300, 0, 20, 1, 40, 0]<br>$\mathbf{a_{max}}$ = [500, 200, 40, 3, 80, 200]<br>$\mathbf{a_0}$ = [400, 100, 35, 2.5, 75, 125]<br>$\mathbf{a_f}$ = [430, 75, 29.9, 1.5, 59.7, 100]<br>Iterations = 51 |

[a] Model parameters are: [30, 1.5, 60, 100].
[b] Model parameters are: [430, 75, 30, 1.5, 60, 100].

## 4. Optimization

Optimization algorithms are iterative. They begin with an initial guess of the values of the model parameters, and generate a sequence of improved estimates until they reach a minimum of the objective function. The strategy to move from one iterate to the next differentiates one algorithm from another (Nocedal and Wright, 1999). Contrary to the grid-search algorithm, optimization algorithms do not systematically search the parameter space for a minimum of the objective function, but rather traverse the parameter space towards a minimum. The advantage of optimization algorithms is that they are significantly faster than the grid-search algorithm. The disadvantage is that they do not always find the lowest minimum (although the probability of being trapped in a local minimum may be lower for global search methods such as simulated annealing and genetic algorithms). In general, gradient-based optimization algorithms find the minimum closest to the initial guess in a descent direction. This is particularly a problem in actual fold data for which the objective function might have several local minima. As we show later, this problem can be overcome by running several local optimization problems distributed over the parameter space, and/or by constraining the optimization (i.e. setting upper and lower parameter bounds).

In this paper, we apply three optimization functions from the MATLAB Optimization Toolbox™ to trishear inverse modeling.

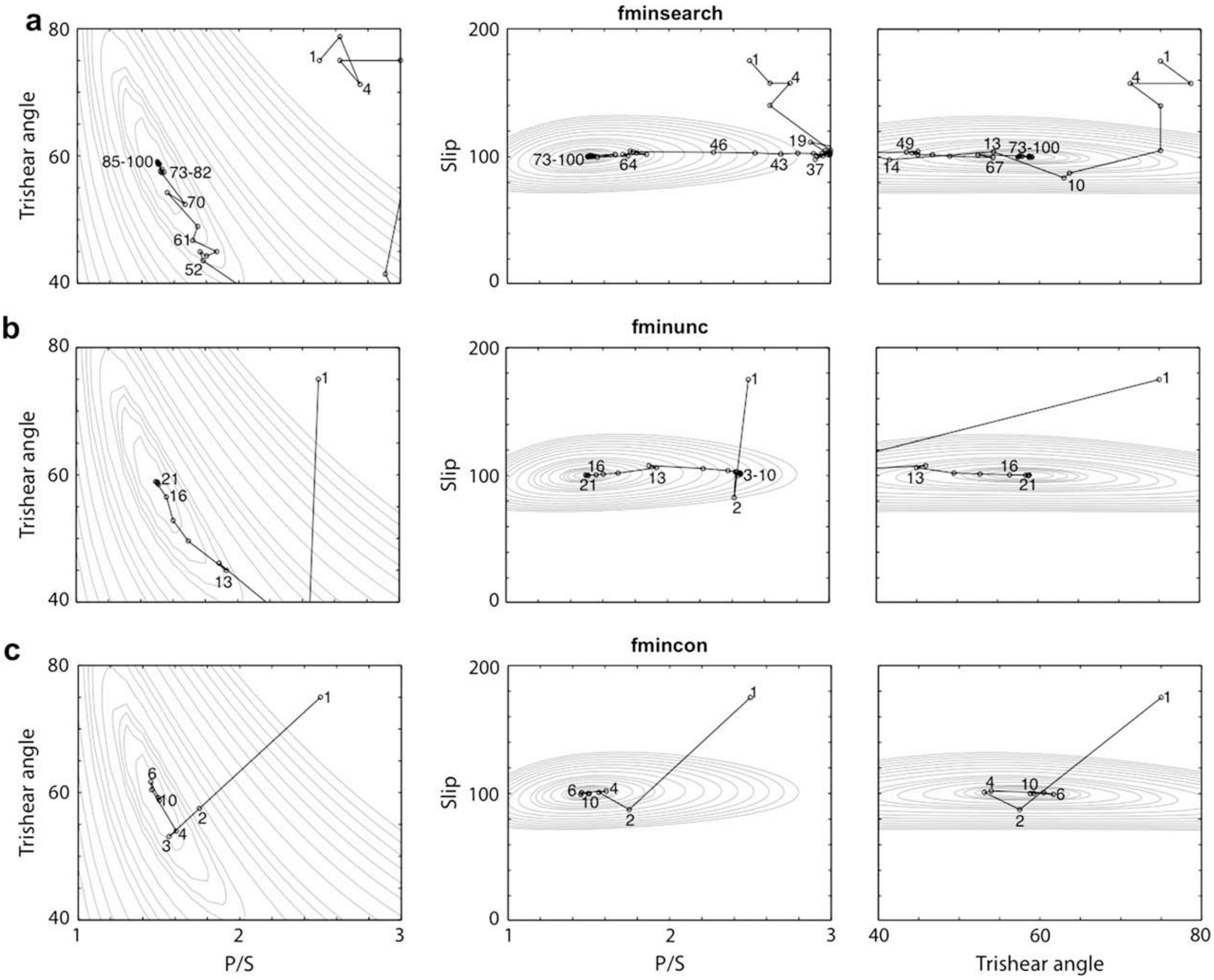


**Fig. 4.** a. fminsearch, b. fminunc, and c. fmincon searches for the P/S, trishear angle and fault slip that best fit the forward reverse fault model of Fig. 1a. The paths of the optimization algorithms (black circles and lines) are shown on: P/S versus trishear angle (left), P/S versus fault slip (middle), and trishear angle versus fault slip (right) diagrams. Iteration numbers on the paths indicate the progression of the search. Gray contours are objective function values on the 2D slices of Fig. 2. Contours are as in Fig. 2.

These functions are fminsearch, fminunc, and fmincon (MATLAB Optimization Toolbox™ 4 User's Guide). fminsearch and fminunc perform unconstrained nonlinear optimization (i.e. no constraints on the parameters), while fmincon performs constrained nonlinear optimization (i.e. explicit constraints on the parameters). fminsearch uses the Nelder-Mead simplex method (Lagarias et al., 1998). In $n$ dimensions, this method evaluates the objective function over a polytope (a simplex) of $n + 1$ points in the parameter space (in 2D, the simplex is a triangle). At each iteration, fminsearch computes the objective function at the points of the simplex, deletes the point with the highest (worst) objective function, and replaces it by a new point giving a new simplex. Where appropriate, the simplex can shrink or grow in size. This is analogous to flopping a triangle around the parameter space until it finds a minimum. fminsearch stops when the objective function is the same (within some tolerance) in all points of the simplex, or when the size of the simplex is less than the specified tolerance. fminsearch requires no gradient information and can handle function discontinuities. The disadvantage of fminsearch is that it converges very slowly to the solution, especially for searches of three or more parameters.

Without information about the gradient of the objective function, fminunc uses the Broyden–Fletcher–Goldfarb–Shanno (BFGS) Quasi-Newton method with a mixed quadratic and cubic line search procedure (MATLAB Optimization Toolbox™ 4 User's Guide). At each iteration, fminunc determines a direction of search by updating the approximation of the Hessian matrix (second order partial derivative) of the objective function, using a BFGS formula (Nocedal and Wright, 1999). After choosing the direction of search, fminunc uses a mixed quadratic and cubic polynomial interpolation method to determine how far to move in the search direction. This strategy determines three points along the search direction that bracket a minimum of the objective function, and uses cubic interpolation to estimate the minimum at each line search. The advantage of fminunc is that it converges rapidly to the solution. As the optimization proceeds, fminunc "learns" about the shape of the objective function and "adapts" next iterates to this shape. The disadvantage of fminunc is that it requires more function evaluations to approximate the Hessian and do the line search, fminunc cannot handle discontinuities.

Without information about the gradient of the objective function, fmincon uses a sequential quadratic programming (SQP) method, which generates iterates by solving quadratic subproblems (MATLAB Optimization Toolbox™ 4 User's Guide; Nocedal and Wright, 1999). At each iteration, fmincon approximates the Hessian of the Lagrangian for the constrained optimization problem using a Quasi-Newton method and the BFSG formula (Nocedal and Wright, 1999). fmincon then uses this approximation to generate a quadratic programming (QP; Nocedal and Wright, 1999) subproblem, which fmincon solves using an active set strategy (Gill et al., 1991). The solution to the QP problem produces a search direction vector, which fmincon uses to form a new iterate. The step size is determined in

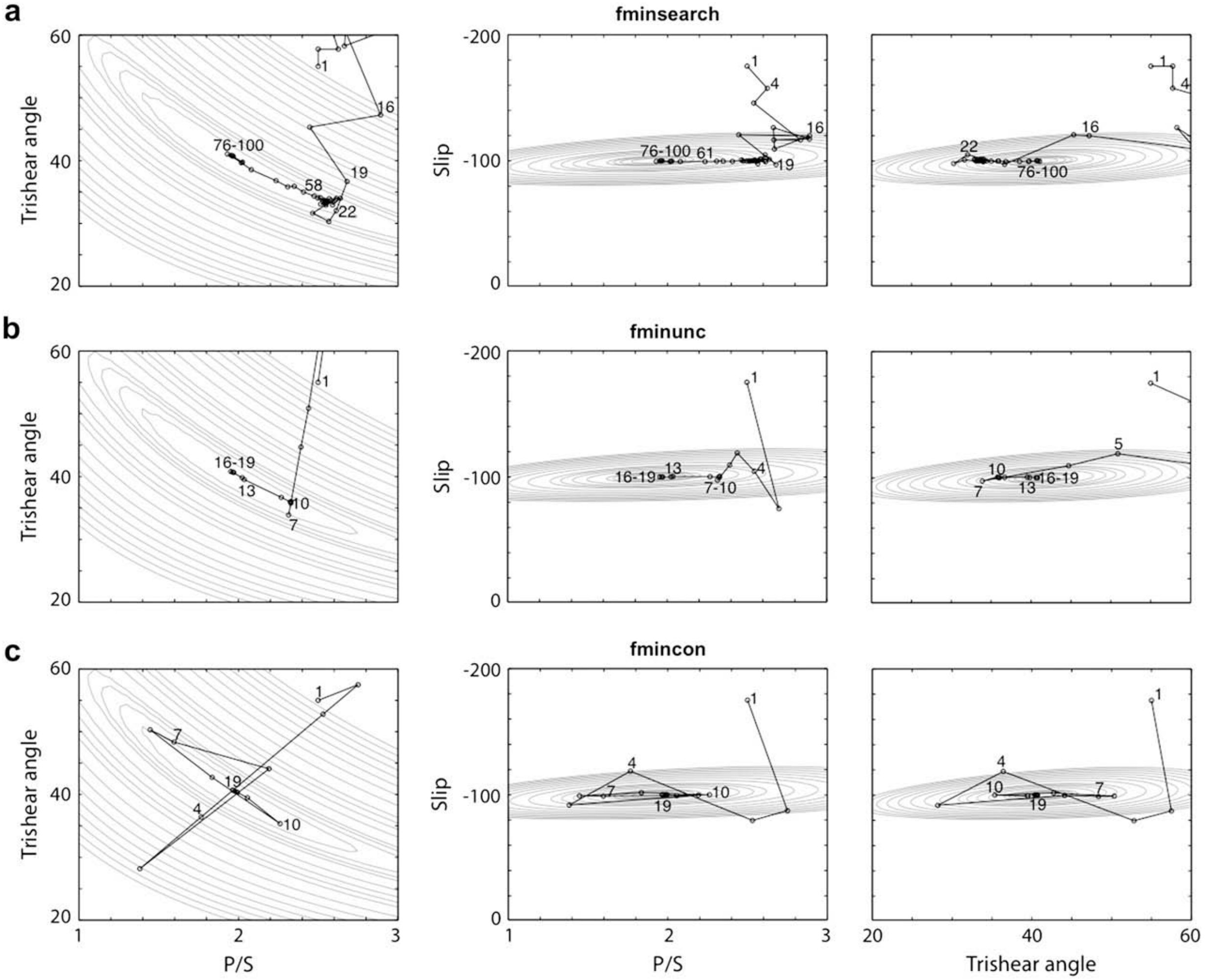


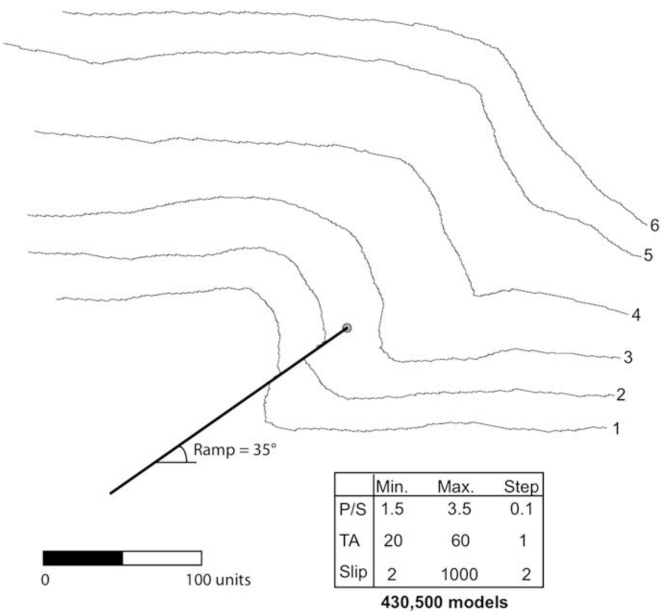
**Fig. 5.** a. fminsearch, b. fminunc, and c. fmincon searches for the P/S, trishear angle and fault slip that best fit the forward normal fault model of Fig. 1b. The paths of the optimization algorithms (black circles and lines) are shown on: P/S versus trishear angle (left), P/S versus fault slip (middle), and trishear angle versus fault slip (right) diagrams. Iteration numbers on the paths indicate the progression of the search. Gray contours are objective function values on the 2D slices of Fig. 3. Contours are as in Fig. 3.

order to produce a sufficient decrease in the objective function (MATLAB Optimization Toolbox™ 4 User's Guide). fmincon often converges to the solution in fewer iterations than fminunc. One of the reasons for this is that, because of limits on the parameter space, fmincon can make informed decisions regarding directions of search and step size. The disadvantage of fmincon is that it requires more function evaluations than fminunc to form a new iterate. Also the algorithm requires a feasible point to start.

In all, fminsearch, fminunc, and fmincon, scaling of the parameters is an important issue. Before inverse modeling, we scale the fault slip and location of the fault tip ($x$ and $y$ in 2D) so that their values are within about an order of magnitude of 1. Trishear and ramp angles are converted to radians. This guarantees that all parameters (including the P/S) are within about an order of magnitude of 1. We then solve the optimization problem in terms of the scaled parameters. This makes the solution more balanced (Nocedal and Wright, 1999).

Figs. 4 and 5 show the results of optimized searches for the reverse (Fig. 4) and normal (Fig. 5) fault models of Fig. 1. As in the grid-search of Figs. 2 and 3, bed 3 was used for the inversion, the location of the fault tip and ramp angle were fixed, and the algorithms searched for the best-fit P/S, trishear angle (TA), and fault slip. For the reverse fault model (Fig. 4), the initial guess ($\mathbf{a_0}$) was [2.5, 75°, 175] (P/S, trishear angle, and fault slip in units). For the normal fault model (Fig. 5), $\mathbf{a_0}$ was [2.5, 55°, −175]. fmincon lower ($\mathbf{a_{min}}$) and upper ($\mathbf{a_{max}}$) bounds for the reverse fault model (Fig. 4c) were [1, 40°, 0] and [3, 80°, 200], and for the normal fault model (Fig. 5c) they were [1, 20°, −200] and [3, 60°, 0]. In Figs. 4 and 5, the paths of the optimization algorithms are shown over the two-dimensional slices and objective function contours of Figs. 2 and 3, and some of the iterations are labeled to show the progression of the search.

All three, fminsearch (Figs. 4a and 5a), fminunc (Figs. 4b and 5b), and fmincon (Figs. 4c and 5c) find best-fit parameters ($\mathbf{a_f}$) that are very close to the parameters of the forward models. The algorithms find the best fault slip very fast, and move along the best fault slip in search for the best P/S and trishear angle (columns 2 and 3 in Figs. 4 and 5). In the reverse fault model (Fig. 4), the algorithms find the best P/S faster than the best trishear angle, since the model is well constrained in P/S. In the normal fault model (Fig. 5), the algorithms wondered as much to find the best P/S as to find the best trishear angle. Both fminunc (Figs. 4b and 5b) and fmincon (Figs. 4c and 5c) converge faster to the solution than fminsearch (Figs. 4a and 5a). Once the best fault slip is found, fminunc moves systematically towards the solution (Figs. 4b and 5b), while fmincon takes larger, bolder steps (Figs. 4c and 5c). This allows fmincon to converge to the solution in fewer iterations. For example in the reverse fault model, fmincon finds the best-fit parameters in just 10 iterations



| | Min. | Max. | Step |
|---|---|---|---|
| P/S | 1.5 | 3.5 | 0.1 |
| TA | 20 | 60 | 1 |
| Slip | 2 | 1000 | 2 |

**430,500 models**

**Fig. 6.** Fault-propagation fold from the Hudson Valley fold and thrust belt (New York, USA; Mitra, 1990). Simplified tracing of the photograph by Allmendinger (1998). Scale is in pixel units. Table below shows the range of values tested in the grid-search procedure (gray contours, Fig. 7).

(Fig. 4c). Because there are no local minima in the parameter space, the optimization algorithms converge to the minimum from any $\mathbf{a_0}$. The time of solution is less than 5 s. All times reported in this note were measured in a 2.4 GHz PC with 3.0 GB of RAM.

Table 1 shows the results of more demanding searches for bed 3 of the reverse fault model of Fig. 1a. Searching in addition for the best ramp angle (RA) from an $\mathbf{a_0}$ of [35°, 2.5, 75°, 175] (ramp angle, P/S, trishear angle, and fault slip in units) produces best-fit parameters ($\mathbf{a_f}$) that are very close to the parameters of the forward model (column 2, Table 1). In this case, there are no local minima and all three optimization algorithms converge to the minimum from any $\mathbf{a_0}$. fminunc and fmincon converge much faster to the solution than fminsearch. Searching for all six parameters of the 2D trishear model, however, is more complicated. This is because, contrary to the previous examples, the six-dimensional parameter space has local minima. Searching with the unconstrained, fminsearch and fminunc, optimization algorithms can give erroneous results if $\mathbf{a_0}$ is far from the correct parameters. In Table 1, the selected $\mathbf{a_0}$ of [400, 100, 35°, 2.5, 75°, 175] ($x$ and $y$ location of the fault tip in units, ramp angle, P/S, trishear angle, and fault slip in units) gives moderately good results ($\mathbf{a_f}$) for fminsearch, and not so good for fminunc (column 3, Table 1). Searching with the constrained fmincon, optimization algorithm (within the $\mathbf{a_{min}}$ and $\mathbf{a_{max}}$ limits shown in Table 1) gives much better results. This is because erroneous values such as negative or very large P/S, very low or large ramp angle, trishear angle, or fault slip, can be avoided. For the selected $\mathbf{a_0}$, fmincon converges to the global minimum, and the best-fit parameters ($\mathbf{a_f}$) are very close to the parameters of the

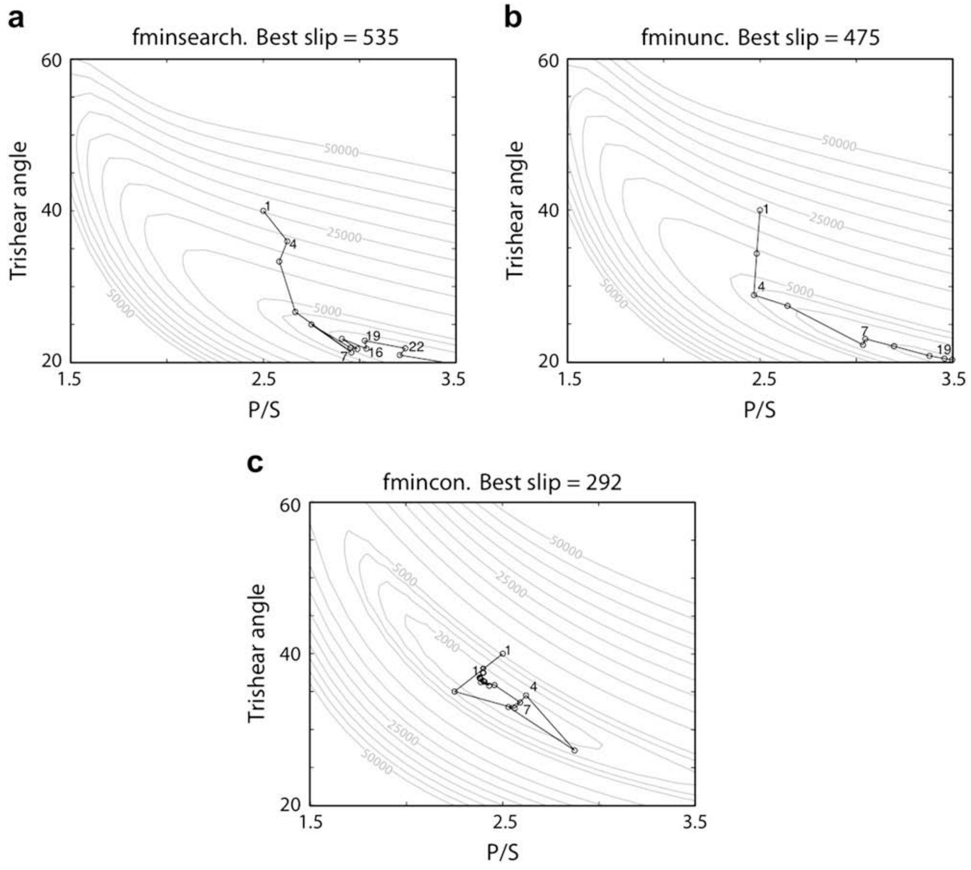


**Fig. 7.** a. fminsearch, b. fminunc, and c. fmincon searches for the P/S, trishear angle and fault slip that best fit bed 3 of Fig. 6. The paths of the optimization algorithms (black circles and lines) are shown on P/S versus trishear angle diagrams. Iteration numbers on the paths indicate the progression of the search. Gray contours show the variation of the objective function on 2D slices through the 3D matrix of objective function values produced by grid searching the section over the ranges shown in Fig. 6. In a–c, slices are constructed at the best fault slip estimated by the optimization algorithm. Contours below 5000 are in intervals of 1000, and above 5000 are in intervals of 5000.

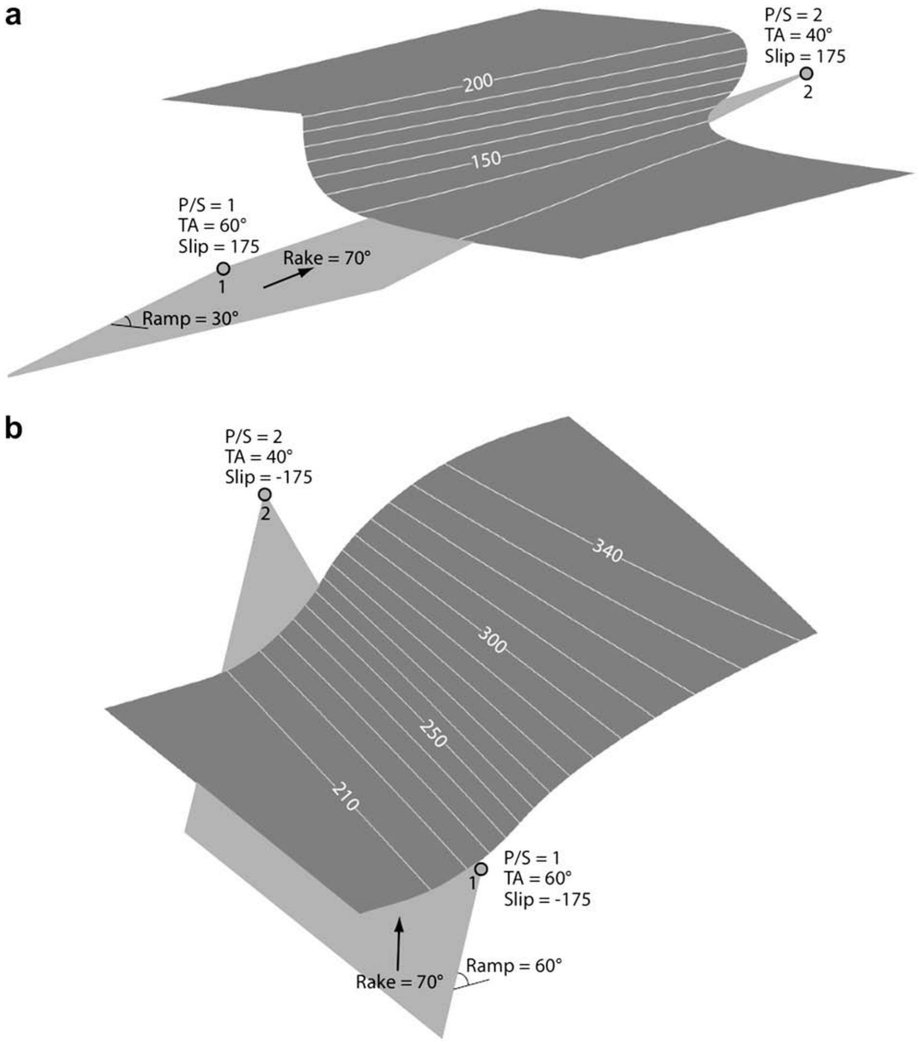


**Fig. 8.** True-3D forward trishear models and their key parameters, a. Reverse fault, b. Normal fault. Contours on bed are elevation. Models a and b were fed into the optimized search (Figs. 9 and 10 and Table 2) procedure to test the accuracy of this technique.

forward model (column 3, Table 1). fmincon finds a good solution in 12 s, as opposed to the hours this would take with the grid-search method. However with fmincon, not all $\mathbf{a_0}$ guesses converge to the lowest minimum. Some $\mathbf{a_0}$, especially close to the lower ($\mathbf{a_{min}}$) and upper ($\mathbf{a_{max}}$) bounds, are not feasible points to start the search. In this case, it helps to run several (five or so) optimized searches with different $\mathbf{a_0}$ spanning the constrained parameter space along one key parameter, for example fault slip. One of these searches will surely find the lowest minimum.

Local minima are especially challenging in the inversion of actual fold data for which the trishear model might not be a perfect representation. To illustrate this, we conduct grid and optimized searches for the trishear model that best fits a meter size fault-propagation fold from the Hudson Valley fold and thrust belt (Fig. 6. Mitra, 1990; his Fig. 21). The same structure was modeled by Allmendinger (1998, his Figs. 11 and 12) using the grid-search method. In this structure, the location of the fault tip and ramp angle are known; thus, the search is for three parameters: P/S, trishear angle (TA), and fault slip (Fig. 6). Grid searching the parameters that best fit bed 3 in Fig. 6, over the range of parameter values shown in Fig. 6 (testing 430,500 models), produces an $\mathbf{a_f}$ of [2.4, 36°, 290] (P/S, trishear angle, and fault slip in pixel units). Doing the same search with the unconstrained optimization algorithms fminsearch and fmincon from an $\mathbf{a_0}$ of [2.5, 40°, 500], however, result in $\mathbf{a_f}$ estimates of [3.8, 18°, 535] and [3.5, 20°, 475], respectively (Fig. 7a, b). Repeating the search with the constrained optimization algorithm fmincon from the same $\mathbf{a_0}$, and with $\mathbf{a_{min}}$ and $\mathbf{a_{max}}$ limits of [1.5, 20°, 0] and [3.5, 60°, 1000], results in an $\mathbf{a_f}$ of [2.38, 37°, 292] (Fig. 7c), which is close to the $\mathbf{a_f}$ of the grid-search.

Contours of the objective function (gray lines in Fig. 7) at the best slip estimated by the optimization algorithms show that there are several local minima in the parameter space. In all three optimized searches, $\mathbf{a_f}$ coincides with a minimum, and the value of the objective function at this minimum is low. It is about 2100 units$^2$ in fminsearch and fminunc (Fig. 7a, b), and 1600 units$^2$ in fmincon (Fig. 7c). Thus, any of the solutions $\mathbf{a_f}$ provided by the optimization algorithms is statistically correct. The best-fit values ($\mathbf{a_f}$) are strongly dependent on the initial guess ($\mathbf{a_0}$). For example, running an unconstrained search with fminunc from an $\mathbf{a_0}$ of [2.5, 40°, 400] (lower fault slip than in the previous fminunc search), results in an $\mathbf{a_f}$ of [2.38, 37°, 292], which is close to the $\mathbf{a_f}$ of the grid-search. Running a constrained search with fmincon from an $\mathbf{a_0}$ of [3.5, 40°, 500] (higher P/S than in the previous fmincon search), results in an $\mathbf{a_f}$ of [3.5, 20°, 475]. Performing searches for the parameters that best fit other beds in the section further constrain the solution. fmincon searches for beds 1 and 6 of the section (Fig. 6), with the same $\mathbf{a_0}$ and $\mathbf{a_{min}}$ and $\mathbf{a_{max}}$ limits, give $\mathbf{a_f}$ values of [3.5, 24°, 242] and [2.5, 33°, 280], respectively. fminsearch and fminunc searches for the same

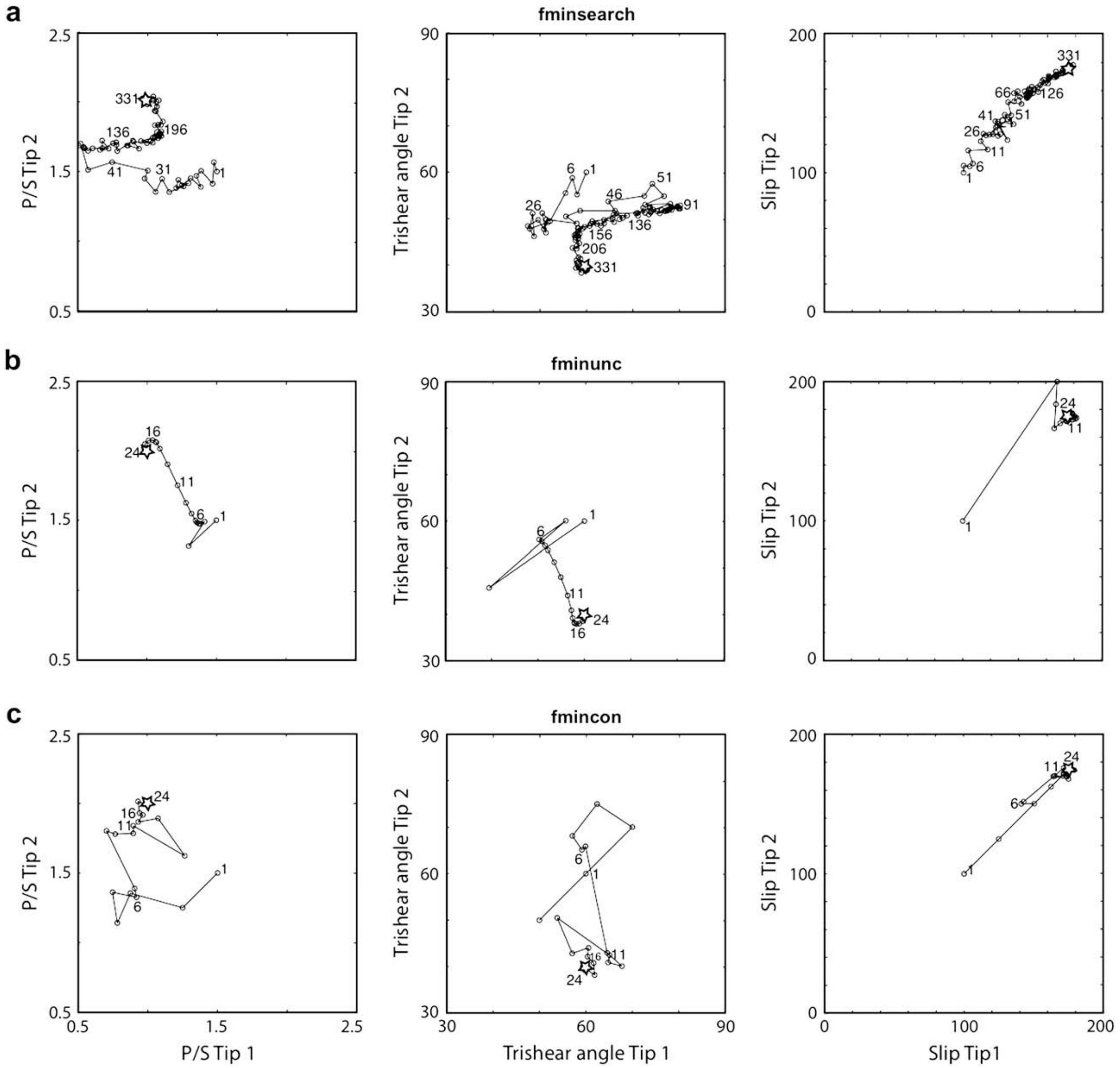


**Fig. 9.** a. fminsearch, b. fminunc, and c. fmincon searches for the P/S, trishear angle and fault slip that best fit the 3D forward reverse fault model of Fig. 8a. The paths of the optimization algorithms are shown on: P/S at fault tip 1 versus P/S at fault tip 2 (left), trishear angle at fault tip 1 versus trishear angle at fault tip 2 (middle), and fault slip at fault tip 1 versus fault slip at fault tip 2 (right) diagrams. Iteration numbers on the paths indicate the progression of the search. In all diagrams, star indicates the parameters of the forward model.

beds, give erroneous results (very large fault slip, and very low P/S and trishear angle). This suggests that a P/S of 2.4–2.5, a trishear angle of 34–37°, and a fault slip of 280–290 units, are appropriate to fit beds 3 and 6. An $\mathbf{a_f}$ of [2.38, 37°, 292] fits moderately well the structure (see Allmendinger, 1998, his Fig. 11c).

The above example shows that local minima exist in the inversion of real fault-propagation folds, even if the parameter space has low dimensions. These local minima highly influence the result of the optimization algorithms. In comparison to the grid-search method, this is a natural disadvantage of the optimization methods. At the same time, the speed of the optimization methods makes it easy to sample the parameter space for local minima. When inverting real structures, one should not be content with the first solution, but rather run several searches with different $\mathbf{a_0}$ to prove the consistency (or inconsistency) of the solution. For example, one can feed the optimization routines with a coarse grid of reasonable $\mathbf{a_0}$ values to ensure that local minima are avoided. This is almost like combining the grid-search and the optimization methods. Local minima may render the unconstrained optimization methods (fminsearch and fminunc) unusable. The unconstrained optimization methods just deliver physically erroneous $\mathbf{a_f}$ estimates. In this case, the only solution is to use fmincon and constrain the parameter space within physically meaningful limits.

## 5. 3D inverse modeling

By analogy with the restoration, 2D inverse modeling approach (Allmendinger, 1998), in 3D we seek the combination of trishear parameters that best restores a folded surface to a plane. In 3D, the objective function ($f$) is expressed as:

$$f = \sum_{i=1}^{N} (z_i - a - bx_i - cy_i)^2 \tag{2}$$

where $x_i$, $y_i$ and $z_i$ are the cartesian coordinates ($z$ being elevation) of the $N$ points on the restored fold surface, and $\mathbf{a}$, $b$ and $c$ are the

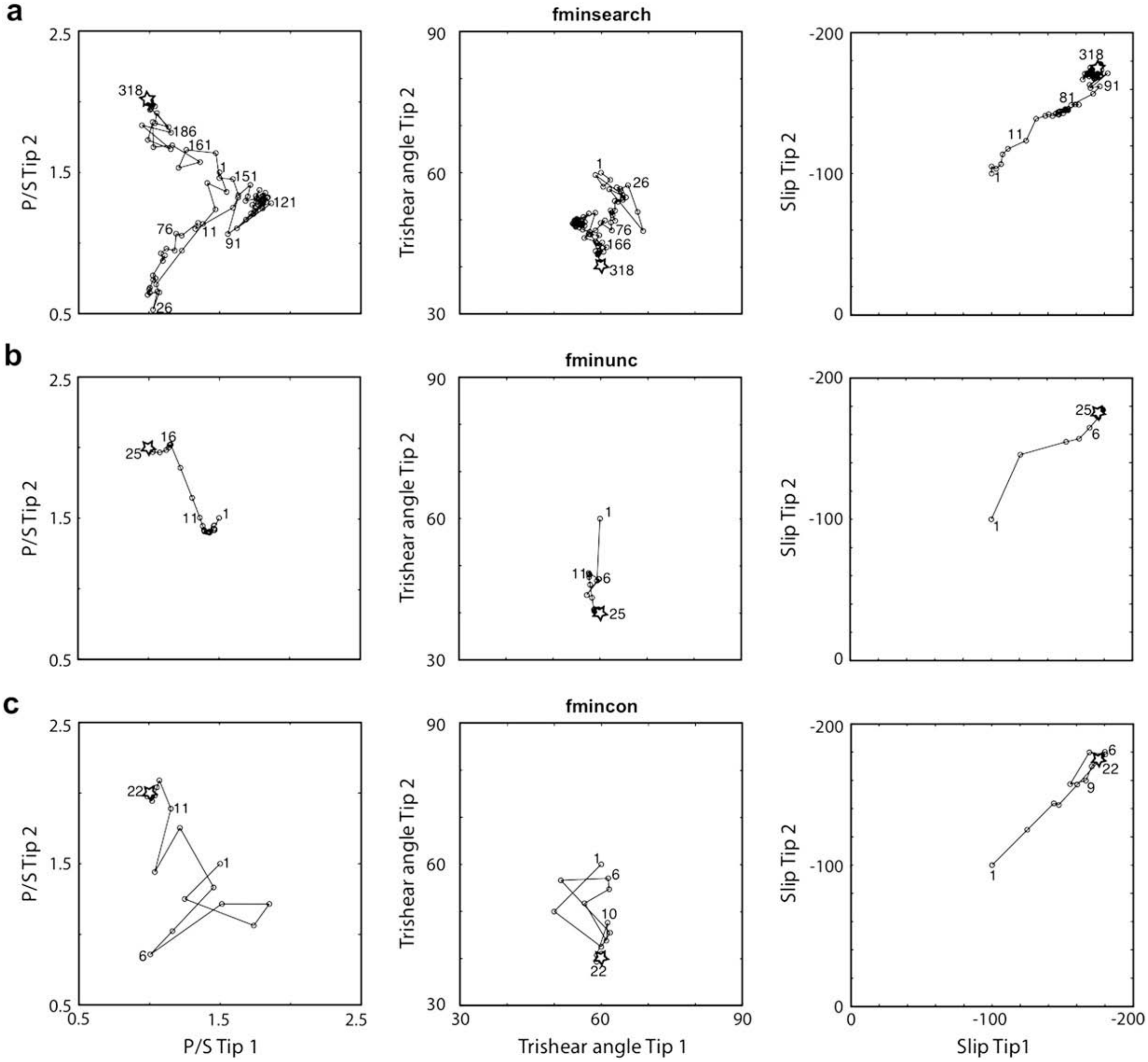


**Fig. 10.** a. fminsearch, b. fminunc, and c. fmincon on a search for the P/S, trishear angle and fault slip that best fit the 3D forward normal fault model of Fig. 8b. The paths of the optimization algorithms are shown on: P/S at fault tip 1 versus P/S at fault tip 2 (left), trishear angle at fault tip 1 versus trishear angle at fault tip 2 (middle), and fault slip at fault tip 1 versus fault slip at fault tip 2 (right) diagrams. Iteration numbers on the paths indicate the progression of the search. In all diagrams, star indicates the parameters of the forward model.

coefficients of the planar regression of the restored fold surface. 3D trishear inverse modeling is a least squares minimization problem on which optimization methods can be applied.

To illustrate the performance of the optimization algorithms, we use two forward 3D trishear models (Fig. 8): one for a reverse (Fig. 8a), and one for a normal (Fig. 8b) fault. The models were constructed using the true-3D trishear algorithm of Cristallini et al. (2004). In both models, the bed is made of 10,000 points (twenty times as many as in a bed of the synthetic 2D models of Fig. 1), the P/S and trishear angle (TA) vary linearly along fault strike, and the fault slip is constant and oblique to fault strike (slip rake = 70°, Fig. 8). For inverse modeling of both, the reverse (Fig. 8a) and normal (Fig. 8b) fault cases, the location of the fault tips (*x*, *y*, and *z*), ramp angle, and slip rake were fixed, and the P/S, trishear angle and fault slip at the fault tips were searched (a total of six unknown parameters).

Figs. 9 and 10 show the results of optimized searches for the reverse (Fig. 9) and normal (Fig. 10) fault models of Fig. 8. For the reverse fault model (Fig. 9), the initial guess ($\mathbf{a_0}$) was [1.5, 1.5, 60°, 60°, 100, 100] ($\mathrm{P/S_{tip1}}$, $\mathrm{P/S_{tip2}}$, trishear $\mathrm{angle_{tip1}}$, trishear $\mathrm{angle_{tip2}}$, fault $\mathrm{slip_{tip1}}$, and fault $\mathrm{slip_{tip2}}$ in units). For the normal fault model (Fig. 10), $\mathbf{a_0}$ was [1.5, 1.5, 60°, 60°, −100, −100]. fmincon lower ($\mathbf{a_{min}}$) and upper ($\mathbf{a_{max}}$) bounds for the reverse fault model (Fig. 9c) were [0.5, 0.5, 20°, 20°, 0, 0] and [3.5, 3.5, 90°, 90°, 200, 200], and for the normal fault model (Fig. 10c) they were [0.5, 0.5, 20°, 20°, −200, −200] and [3.5, 3.5, 90°, 90°, 0, 0]. In Figs. 9 and 10, the paths of the optimization algorithms are shown on $\mathrm{parameter_{tip1}}$ versus $\mathrm{parameter_{tip2}}$, diagrams, and some of the iterations are labeled to show the progression of the search. The stars in Figs. 9 and 10 indicate the parameters of the forward models.

All three, fminsearch (Figs. 9a and 10a), fminunc (Figs. 9b and 10b), and fmincon (Figs. 9c and 10c) find best-fit parameters ($\mathbf{a_f}$) that are very close to the parameters of the forward models. The algorithms find the best fault slip at the fault tips faster (third column in Figs. 9 and 10), and spend more iterations finding the best P/S and trishear angle (first and second columns in Figs. 9 and 10). Gradient-based fminunc (Figs. 9b and 10b) and fmincon (Figs. 9c and 10c) converge faster to the solution than fminsearch (Figs. 9a and 10a). The time of solution for fminsearch is 6 min, while for fminunc and fmincon is 2 and 2.5 min, respectively. All three optimization

**Table 2**
Optimized searches for the parameters **a** that best fit the 3D forward reverse fault model of Fig. 8a. $\mathbf{a_0}$ is the initial guess, and $\mathbf{a_f}$ the best-fit parameters. $\mathbf{a_{min}}$ and $\mathbf{a_{max}}$ are lower and upper bounds for fmincon.

| Function | **a** = [P/S$_1$, P/S$_2$, TA$_1$, TA$_2$, Slip$_1$, Slip$_2$, Rake][a] | **a** = [RA, P/S$_1$ P/S$_2$, TA$_1$ TA$_2$, Slip$_1$ Slip$_2$, Rake][b] |
|---|---|---|
| fminsearch | $\mathbf{a_0}$ = [1.5, 1.5, 60, 60, 100, 100, 90]<br>$\mathbf{a_f}$ = [0.99, 2.01, 59.7, 38.7, 172, 172, 72.4]<br>Iterations = 714 | $\mathbf{a_0}$ = [25, 1.5, 1.5, 60, 60, 100, 100, 90]<br>$\mathbf{a_f}$ = [29.8, 1.0, 2.0, 59.5, 35.8, 158, 164, 91.8]<br>Iterations = 1000<br>Maximum number of iterations exceeded |
| fiminunc | $\mathbf{a_0}$ = [1.5, 1.5, 60, 60, 100, 100, 90]<br>$\mathbf{a_f}$ = [0.99, 2.01, 59.7, 38.7, 172, 172, 72.4]<br>Iterations = 41 | $\mathbf{a_0}$ = [25, 1.5, 1.5, 60, 60, 100, 100, 90]<br>$\mathbf{a_f}$ = [29.9, 1.0, 2.0, 59.7, 39.1, 172, 172, 72.6]<br>Iterations = 78 |
| fmincon | $\mathbf{a_{min}}$ = [0.5, 0.5, 20, 20, 0, 0, 60]<br>$\mathbf{a_{max}}$ = [3.5, 3.5, 90, 90, 200, 200, 120]<br>$\mathbf{a_0}$ = [1.5, 1.5, 60, 60, 100, 100, 90]<br>$\mathbf{a_f}$ = [0.99, 2.01, 59.7, 38.7, 172, 172, 72.4]<br>Iterations = 39 | $\mathbf{a_{min}}$ = [20.9, 0.5, 0.5, 20, 20, 0, 0, 60]<br>$\mathbf{a_{max}}$ = [40, 3.5, 3.5, 90, 90, 200, 200, 120]<br>$\mathbf{a_0}$ = [25, 1.5, 1.5, 60, 60, 100, 100, 90]<br>$\mathbf{a_f}$ = [29.9, 1.0, 2.0, 59.7, 39.1, 172, 172, 72.6]<br>Iterations = 49 |

[a] Model parameters are: [1.0, 2.0, 60, 40, 175, 175, 70].
[b] Model parameters are: [30, 1.0, 2.0, 60, 40, 175, 175, 70].

algorithms converge to the same solution from any $\mathbf{a_0}$. This suggests that the six-dimensional parameter space has no local minima.

Table 2 shows the results of more demanding searches for the 3D reverse fault model of Fig. 8a. Searching in addition for the slip rake with any of the optimization algorithms, produces best-fit parameters ($\mathbf{a_f}$) that are very close to the parameters of the forward model (column 2, Table 2). This happens regardless of the value of $\mathbf{a_0}$. The seven-dimensional parameter space has no local minima. The same result is obtained when searching in addition for ramp angle (Column 3, Table 2). The eight-dimensional parameter space has no local minima. The rate of convergence of the optimization algorithms is different. For the eight parameters search (column 3, Table 2), fminsearch still does not converge to the solution after 17 min, while gradient-based fminunc and fmincon converge to the solution in 9 and 6 min, respectively. 3D trishear inverse modeling using the simplex method (fminsearch) is not efficient. Searching for all fourteen parameters of the 3D trishear model is complicated. This is because the parameter space has several local minima. In this case, it is necessary to constrain the parameter space as much as possible, and to use an $\mathbf{a_0}$ as close as possible to the suspected solution.

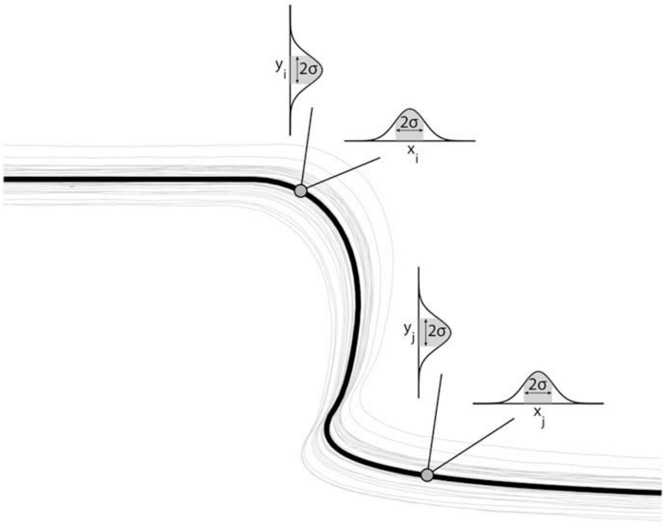


**Fig. 11.** Strategy to generate realizations from a fold profile (dark line). Each point on the profile has uncertainties in *x* and *y* that obey a normal probability distribution, with a mean (μ) equal to the measured *x* or *y* and a standard deviation (σ). Uncertainties within a distance along the bed lower than the correlation length are correlated. Gray lines are realizations.

As in 2D trishear inverse modeling, local minima should be expected when inverting actual 3D fold data, even if the parameter space has low dimensions. Running several searches with different $\mathbf{a_0}$ to sample the local minima, and constraining the optimized search, should help to overcome this problem. 3D inverse modeling works with either the pseudo-3D (Cristallini and Allmendinger, 2001) or true-3D (Cristallini et al., 2004) trishear algorithms. However, to restrict the number of searched parameters to those at the end fault tips, a simple mathematical function should describe the variation of the parameters along the fault. For the true-3D trishear algorithm, this function should be linear. For the pseudo-3D trishear algorithm, this function can be linear or elliptical (Cardozo, 2008). A simple variation of trishear parameters along the fault might not be the case in a real structure. If more complex variations of parameters such as fault slip are observed along fault strike, and if the structure is well exposed in several cross sections perpendicular to fault strike; it is more appropriate to run 2D trishear inverse modeling of the cross sections and to integrate the results of the 2D inversions in a forward pseudo-3D trishear model (Cristallini and Allmendinger, 2001; Cardozo, 2008).

## 6. Uncertainties of best-fit parameters

Fold data are not exact. They are subject to errors of various kinds, including imaging and interpretation errors. These errors introduce some uncertainty in the estimated best-fit parameters ($\mathbf{a_f}$). In this section, we describe a methodology to estimate the uncertainties of $\mathbf{a_f}$. This methodology uses the fold data (with estimated parameters $\mathbf{a_{f0}}$) and an assumed error distribution at the data points to generate several synthetic datasets (i.e. realizations). Inverse modeling of these realizations gives a set of simulated best-fit parameters ($\mathbf{a_{f1}}$, $\mathbf{a_{f2}}$,...) that is distributed around $\mathbf{a_{f0}}$, and from which we can determine the probability distribution and uncertainties of $\mathbf{a_f}$. This technique is known as the randomized maximum likelihood (RML) method (Oliver et al., 1996). An accurate representation of the probability distribution of $\mathbf{a_f}$ may require the inverse modeling of hundreds or thousands of fold realizations; a task that can be significantly expedited with the optimization algorithms.

Fig. 11 shows the strategy we use to generate realizations from the fold data. For simplicity, we only discuss 2D fold profiles, although the methods described here can be extended to 3D fold surfaces. Each of the *N* points on the fold profile (black line in Fig. 11) has errors in location in *x* and y. These errors follow a normal probability distribution, with a mean (μ) equal to the measured *x* or *y* and a standard deviation (σ, Fig. 11). The errors in location are correlated along the profile up to a maximum distance or correlation

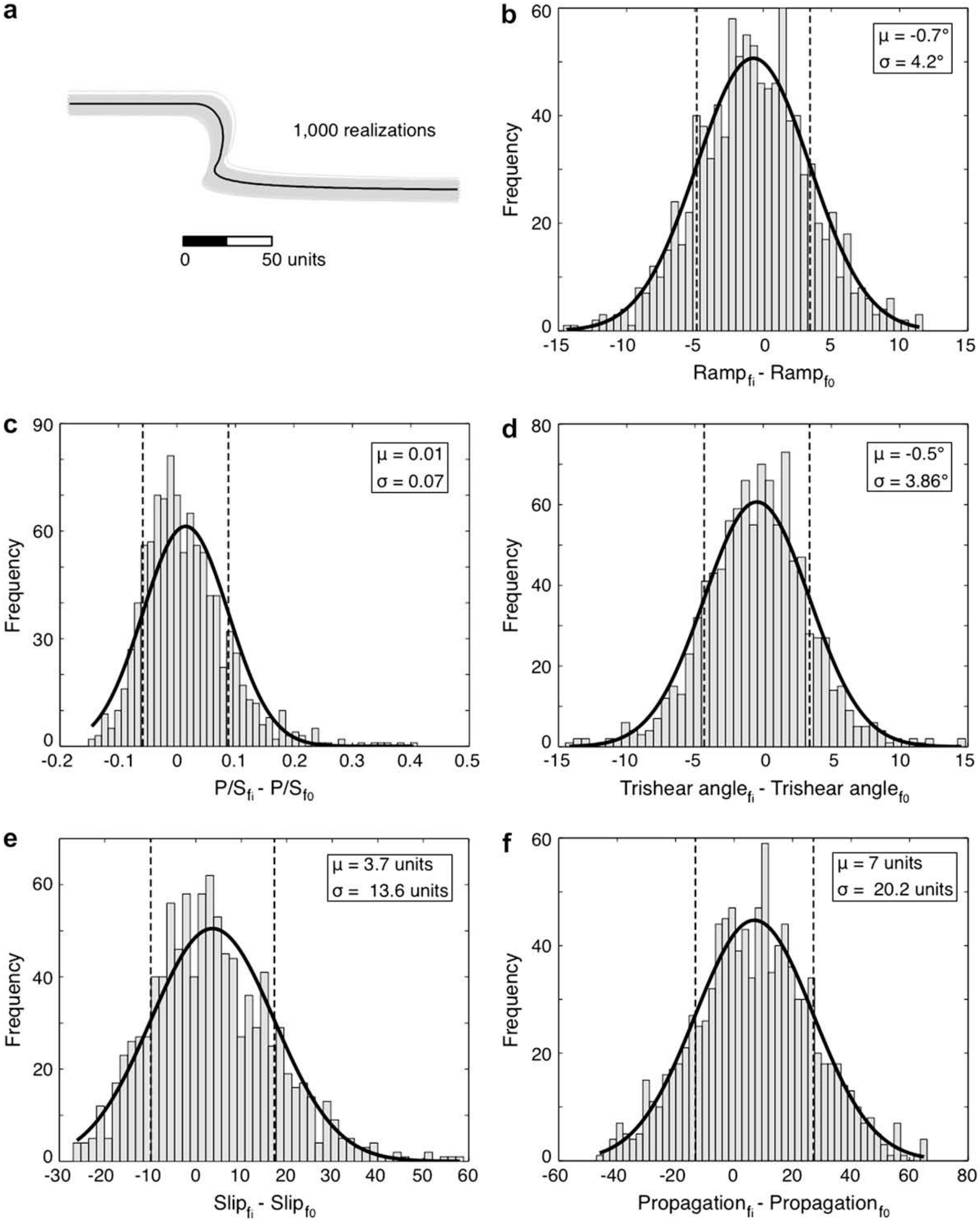


**Fig. 12.** Application of the RML method to the forward reverse fault model of Fig. 1a. a. Bed 3 in Fig. 1a (black line) and its realizations (gray lines). Realizations are based on a standard deviation of 2.236 units and a correlation length of 10,000 units, b–f. Histograms for uncertainties in b. Ramp angle, c. P/S, d. Trishear angle, e. Fault slip, and f. Fault propagation. In b–f, black thick line is a normal distribution (with mean |μ and standard deviation σ) fit to the histogram. Dash lines delimit the 1σ range of permissible values (i.e. 68% confidence interval).

length ($l_c$). For a spherical variogram model (Davis, 2002), the $N \times N$ matrix **R** that describes the correlation of the errors in location is:

$$R_{ij} = \begin{cases} 1 + 0.5\left(h^3 - 3h\right), & h < 1 \\ 0, & h \geq 1 \end{cases} \tag{3}$$

where $h$ is $l_{ij}/l_c$ and $l_{ij}$ is the distance along the profile between points $i$ and $j$ (Fig. 11). The $N \times N$ covariance matrix (**C**) is:

$$\mathbf{C} = \mathbf{S}^*\mathbf{R}^*\mathbf{S} \tag{4}$$

where **S** is an $N \times N$ diagonal matrix with the diagonal elements equal to the standard deviation (σ), and the asterisk denotes matrix multiplication. We can generate realizations (gray lines in Fig. 11) from the fold profile with measured locations in $x$ and $y$ ($\boldsymbol{\mu}_\mathbf{x}, \boldsymbol{\mu}_\mathbf{y}$) and covariance matrix (**C**) by using the following process (Aster et al., 2005): (i) Find the Cholesky factorization $\mathbf{C} = \mathbf{L}^*\mathbf{L}^\mathrm{T}$, (ii) Let **z** be

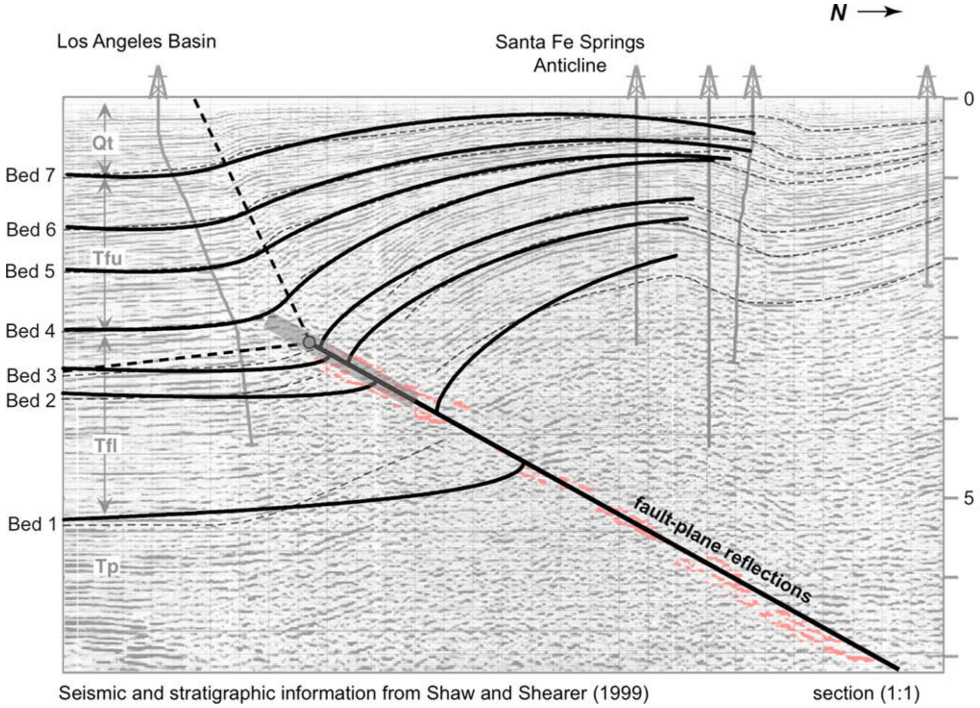


**Fig. 13.** Best-fit trishear model (black thick lines) superimposed on depth-converted seismic section of the Santa Fe Springs anticline. Seismic reflection data and marked stratigraphic units (thin, dashed lines) are from Shaw and Shearer (1999, their Fig. 1). Tfu and Tfl are the upper and lower members of the Fernando Formation.

a vector of $N$ independent (0,1) random numbers, (iii) Let $x$ and $y$ locations of points on realization be $\mathbf{x} = \mu_\mathbf{x} + \mathbf{L}^*\mathbf{z}$ and $\mathbf{y} = \mu_\mathbf{x} + \mathbf{L}^*\mathbf{z}$.

Fig. 12 shows the application of the RML method to bed 3 of the forward trishear model of Fig. 1a (reverse fault). In this case, the trishear model is a perfect representation of the fold profile and the estimated best-fit parameters for bed 3 ($\mathbf{a_{f0}}$) are very close to the parameters of the forward model. The purpose of the RML method is to determine the sensitivity of $\mathbf{a_f}$ to random errors (i.e. uncertainties) in the location of the beds. Fig. 12a shows the profile of bed 3 (black line) and its realizations (gray lines). 1000 realizations were created using a σ of 2.236 units (variance = 5 units$^2$) and an $l_c$ of 10,000 units. $l_c$ must be large (ten times the length of the bed) in order to obtain smooth profiles. Optimized searches for the ramp angle, P/S, trishear angle, and fault slip that best fit each realization provide a set of simulated best-fit parameters ($\mathbf{a_{fi}}$). Specifically, we used fminunc with an initial guess ($\mathbf{a_0}$) of [35°, 2.5, 75°, 175] (ramp angle, P/S, trishear angle, and fault slip in units), and only included in the statistics those realizations for which fminunc converged to the solution. The inversion of the 1000 realizations took about 2 h. The results are shown as probability distributions of the difference between $\mathbf{a_{fi}}$ and $\mathbf{a_{f0}}$ (Fig. 12b–f). In these diagrams, black, thick lines are normal distribution fits (with mean μ and standard deviation σ) to the histograms, and dashed vertical lines delimit 68% (±σ) confidence intervals. For σ errors in $x$ and $y$ locations of 2.236 units, the σ errors in ramp angle, P/S, trishear angle, fault slip, and fault propagation are 4.2°, 0.07, 3.86°, 13.6 units, and 20.2 units, respectively (Fig. 12b–f). 68% of the $\mathbf{a_{fi}}$ around the correct $\mathbf{a_f}$ fall within intervals in ramp angle, P/S, trishear angle, fault slip, and fault propagation of 24.9–33.2°, 1.45–1.6, 54.2–61.9°, 91.1–118.2 units, and 139–179.4 units, respectively (Fig. 12b–f). Similar results are obtained using other beds in the section (for example bed 4 in Fig. 1a). Best-fit P/S and trishear angle values are sensitive to variations of fold shape. For example, not including errors in $x$ locations in the realizations (narrower width of realizations in the forelimb area), results in larger σ errors for the best-fit P/S and trishear angle (σ = 0.2 for best-fit P/S, and 9.8° for best-fit trishear angle). Best-fit ramp angle and fault slip values are not sensitive to variations of fold shape, but rather to variations in depth in the flat hanging wall and footwall areas, which determine the fault throw.

As an example of the application of the RML method to actual fold data, we analyze a buried fold-fault pair in the Los Angeles basin, the Santa Fe Springs anticline and the underlying Puente Hills thrust fault (Shaw and Shearer, 1999; Fig. 13). This structure was studied by Allmendinger and Shaw (2000), who showed that the tip of the Puente Hills thrust initiated at the same position as the 1987 M6.0 Whittier Narrows earthquake. Because of the possible implications of this study for earthquake hazard assessment, an estimate of the uncertainties of $\mathbf{a_f}$ is highly desirable. The thrust fault is well-defined in the seismic reflection data (Fig. 13). The ramp angle is 29°. The location of the fault tip is not exactly known and is assumed to be along the fault projection within the gray, translucent rectangle of Fig. 13. The distance along the fault between the estimated fault tip and the lower end of the gray rectangle in Fig. 13 is defined here as lft. Thus the search is for the lft, P/S, trishear angle, and fault slip that best fit one of the pregrowth beds of Fig. 13. For this analysis, we choose the top of the lower Fernando Formation (bed 4 in Fig. 13). With an $\mathbf{a_0}$ of [1.0 km, 2.5, 60°, 7.5 km], $\mathbf{a_{min}}$ of [0, 1.5, 40°, 0], and $\mathbf{a_{max}}$ of [2 km, 3.5, 80°, 15 km], the constrained optimization algorithm fmincon gives an $\mathbf{a_{f0}}$ for bed 4 of [1.44 km, 2.52, 71°, 6.7 km] (lft, P/S, trishear angle, and fault slip). A forward trishear model with this $\mathbf{a_{f0}}$ (black, thick lines in Fig. 13) fits beds 4–7 well, and beds 1–3 not so well (Fig. 13). The thrust nucleates 17 km down dip from its final (current) location according to this model. This estimate is close to Allmendinger and Shaw (2000) who determined a propagation distance of 18.7 km for the Puente Hills thrust. Differences between our $\mathbf{a_{f0}}$ estimate and Allmendinger and Shaw (2000) $\mathbf{a_{f0}}$ estimate can be attributed to differences in the interpretation and digitization of the beds, differences in fault slip increment, and differences in lft. Reducing lft such that the fault tip is below bed 2 for example, produces an $\mathbf{a_{f0}}$ that fits beds 2 and 3 better, fits beds 4–7 quite well (Fig. 13), and is closer to Allmendinger and Shaw (2000) estimate.

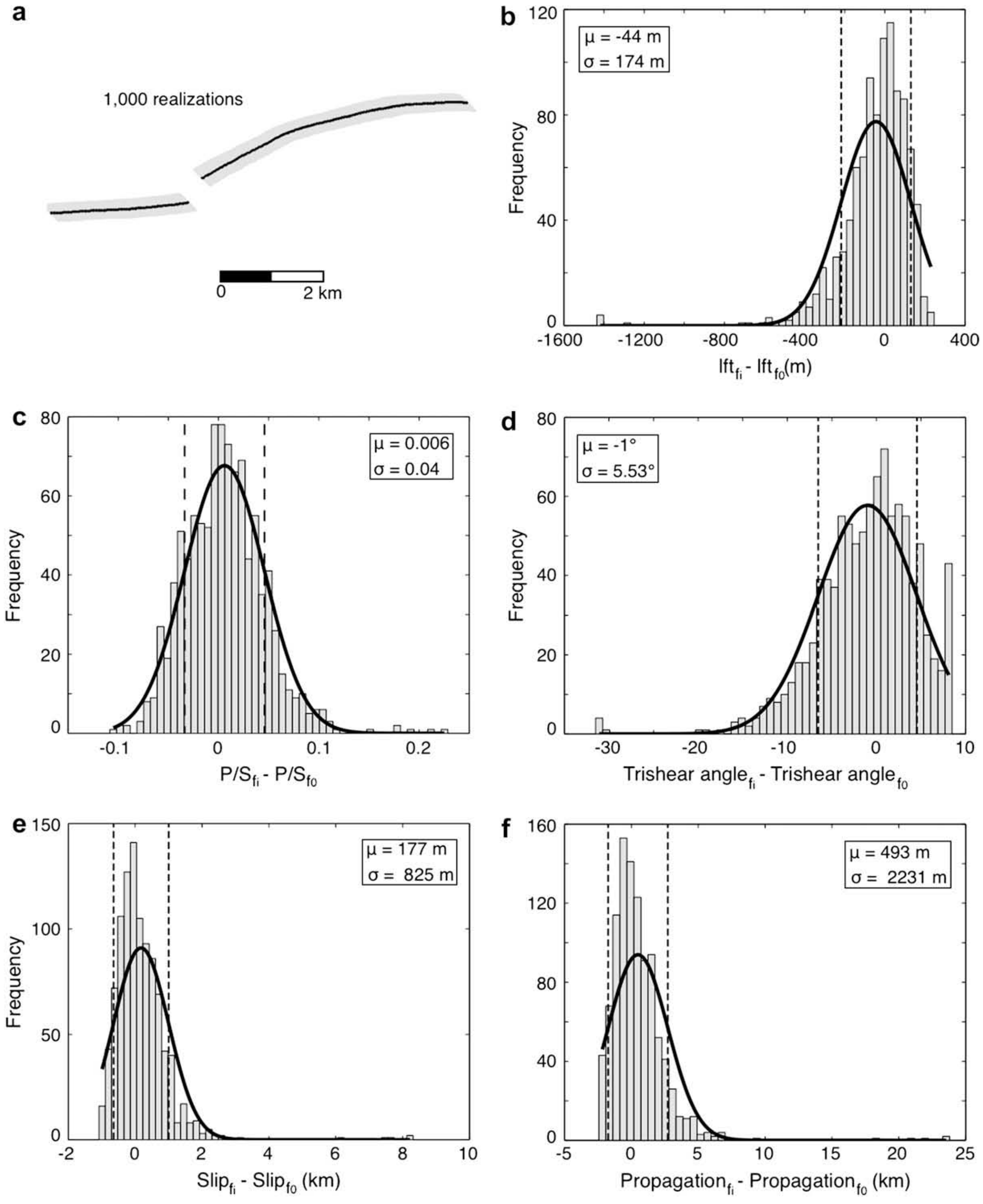


**Fig. 14.** Application of the RML method to the Santa Fe Springs anticline, a. Top of the lower Fernando Formation (black line) and its realizations (gray lines). Realizations are based on a standard deviation of 50 m and a correlation length of 100 km. b–f. Histograms for uncertainties in b. Location of fault tip along fault projection (lft), c. P/S, d. Trishear angle, e. Fault slip, and f. Fault propagation. In b–f, black thick line is a normal distribution (with mean μ and standard deviation σ) fit to the histogram. Dash lines delimit the 1σ range of permissible values (i.e. 68% confidence interval).

However, for the RML analysis, we prefer not to constrain the location of the fault tip so much.

Fig. 14 shows the application of the RML method to the Santa Fe Springs anticline. Fig. 14a shows the top of the lower Fernando Formation (black line) and its realizations (gray lines). 1000 realizations were created using a σ of 50 m and an $l_c$ of 100 km. This makes the realizations fall within a distance ±200 m from the interpreted bed in a direction perpendicular to the bed. This uncertainty is comparable with the possible uncertainty due to the time-depth conversion of the seismic data, where a 10% error in a seismic velocity of 4 km/s (a reasonable estimate for the Tertiary sediments in the basin) at 1 s two-way travel time would yield an uncertainty in the depth to the bed of ±200 m. Inverse modeling of the realizations with fmincon using the same $\mathbf{a_0}$, $\mathbf{a_{min}}$, and $\mathbf{a_{max}}$ as before, produces a set of simulated best-fit parameters ($\mathbf{a_{fi}}$). As in the synthetic case, we only included in the statistics those realizations for which fmincon converged to the solution. The inversion of the 1000 realizations took about 2 h. For σ errors in $x$ and $y$ locations of 50 m, normal distribution fits to the probability distribution of $\mathbf{a_{fi}}-\mathbf{a_{f0}}$ indicate that the σ errors in lft, P/S, trishear angle, fault slip, and fault propagation are 174 m, 0.04, 5.5°, 825 m, and 2231 m, respectively (Fig. 14b–f). There is 68% chance that the true parameter values fall within intervals in lft, P/S, trishear angle, fault slip, and fault propagation of 1218–1565 m, 2.48–2.56, 65–76°, 6.04–7.7 km, and 15.1–19.6 km, respectively (Fig. 14b–f). The position of the 1987 M6.0 Whittier Narrows earthquake (Shaw and Shearer, 1999) is within these 68% lft and fault propagation confidence intervals. The uncertainties in $\mathbf{a_f}$ come mainly from uncertainties in trishear angle and fault slip. The structure can be fit with relatively high trishear angle and low fault slip, or vice versa (Fig. 14d, e). These results are similar to Hardy and Allmendinger (in press), who conducted inverse modeling of 200 realizations of the anticline using the grid-search method. The computing time for the inversion of those 200 realizations is not reported in their publication.

## 7. Discussion

2D or 3D trishear inverse modeling is a minimization problem on which optimization algorithms can be applied. In comparison to the grid-search algorithm, the optimization algorithms discussed here are significantly faster but less robust. The optimization algorithms rapidly traverse the parameter space towards a minimum of the objective function (especially gradient-based algorithms), but this minimum is not necessarily the lowest minimum in the parameter space. The performance of the optimization algorithms is affected by local minima. This is a problem in the inversion of actual fold data, where several local minima may exist in the parameter space, even if the space has low dimensions (i.e. few parameters searched). These local minima may cause non-constrained optimization algorithms to give nonsensical, physically meaningless results. We find that when modeling actual fold data, the best is to use a constrained optimization algorithm, to set physically meaningful limits on the parameters, and to conduct several searches from different initial guesses ($\mathbf{a_0}$). The optimization algorithms are fast, and therefore one should aim for several runs that involve different optimization paths and sample different regions in the parameter space. The application of the optimization algorithms to trishear inverse modeling requires critical thinking by the modeler. The optimization algorithms should not be used as black boxes that provide a unique solution, but rather like efficient means to test the robustness of a possible solution (i.e. a best-fit model). In this sense, the optimization algorithms are complementary to the grid-search method and provide an additional source of information that cannot easily be established with the grid-search method (see for example Figs. 6 and 7).

Fast inverse modeling routines are a prerequisite for demanding tasks such as 3D trishear inverse modeling, and estimating the uncertainties of the best-fit parameters ($\mathbf{a_f}$). Gradient-based optimization algorithms that rapidly converge to the solution are necessary for these tasks. 3D trishear inverse modeling is limited by the complexity of the spatial variation of the model parameters. Simple linear or elliptical variation of trishear parameters along fault strike can be managed with optimized 3D inversion routines. More complex variations of model parameters in 3D are better accounted for a combination of 2D trishear inverse modeling in cross sections perpendicular to fault strike, and pseudo-3D trishear forward modeling that integrates the results of the 2D inversions. Estimating the uncertainties of $\mathbf{a_f}$ by the RML method involves the inverse modeling of hundreds of fold realizations, which can be very computing intensive if using a grid-search algorithm. Optimized trishear inverse modeling renders this task possible within hours. A drawback of the application of the optimization techniques to the RML method, however, is that the optimization can deliver erroneous, not statistically significant results for some realizations. Including only those realizations for which the optimization converges to a solution and/or for which the objective function is below a maximum value ensures the correctness of the statistics data.

## Acknowledgements

We would like to thank our colleagues Jan Tveranger, Alvar Braathen, Haakon Fossen, and Tao Feng for discussions on fault related folding and inverse problems. Thanks to the inverse problem group at CIPR for opening an arena where geologists and mathematicians could share ideas. Richard Allmendinger and John Shaw kindly provided us with data from the Hudson Valley fold and thrust belt and the Santa Fe Springs anticline. The optimized trishear inverse modeling methods described in this paper were implemented as scripts in MATLAB® using the MATLAB Optimization Toolbox™. The scripts are available from the corresponding author upon request. This work was supported by the Center for Integrated Petroleum Research (CIPR) and the University of Stavanger, Norway.

## References

Allmendinger, R.W., 1998. Inverse and forward numerical modeling of trishear fault-propagation folds. Tectonics 17, 640–656.

Allmendinger, R.W., Shaw, J.H., 2000. Estimation of fault propagation distance from fold shape: implications for earthquake hazard assessment. Geology 28, 1099–1102.

Allmendinger, R.W., Zapata, T.R., Manceda, R., Dzelalija, F., 2004. Trishear kinematic modeling of structures, with examples from the Neuquen Basin, Argentina. In: McClay, K.R. (Ed.), Thrust Tectonics and Hydrocarbon Systems. American Association of Petroleum Geologists, Memoir, v, vol. 82, pp. 356–371.

Aster, R.C., Borchers, B., Thurber, C.H., 2005. Parameter Estimation and Inverse Problems. Elsevier Academic Press, 296 pp.

Cardozo, N., 2005. Trishear modeling of fold bedding data along a topographic profile. Journal of Structural Geology 27, 495–502.

Cardozo, N., 2008. Trishear in 3D. Algorithms, implementation, and limitations. Journal of Structural Geology 30, 327–340.

Cristallini, E.O., Allmendinger, R.W., 2001. Pseudo-3D modeling of trishear fault-propagation folding. Journal of Structural Geology 23, 1883–1899.

Cristallini, E.O., Giambiagi, L., Allmendinger, R.W., 2004. True three-dimensional trishear: a kinematic model for strike-slip and oblique-slip deformation. Geological Society of America Bulletin 116, 938–952.

Davis, J.C., 2002. Statistics and Data Analysis in Geology, third ed. John Wiley & Sons, New York, 638 pp.

Erslev, E.A., 1991. Trishear fault-propagation folding. Geology 19, 617–620.

Gill, P.E., Murray, W., Wright, M.H., 1991. Numerical Linear Algebra and Optimization, vol. 1. Addison Wesley, Redwood City, California, 426 pp.

Hardy, S., Allmendinger, R.W. Trishear: a review of kinematics, mechanics and applications. In: McClay, K., Shaw, J.H., Suppe, J. (Eds.), Thrust Fault Related Folding. American Association of Petroleum Geologists, Memoir, in press.

Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, R.E., 1998. Convergence properties of the Nelder-Mead simplex method in low dimensions. SIAM Journal of Optimization 9, 112–147.

MATLAB Optimization Toolbox™ 4 User's Guide. Available at: http://www.mathworks.com/access/helpdesk/help/pdf_doc/optim/optim_tb.pdf.

Mitra, S., 1990. Fault-propagation folds: geometry, kinematic evolution, and hydrocarbon traps. American Association of Petroleum Geologists Bulletin 74, 921–945.

Nocedal, J., Wright, S.J., 1999. Numerical optimization. In: Glynn, P., Robinson, S.M. (Eds.), Springer Series in Operations Research. Springer-Verlag, New York, pp. 636.

Oliver, D.S., Nanqun, H., Reynolds, A.C. 1996. Conditioning permeability fields to pressure data. In: Proceedings of the Fifth European Conference for the Mathematics of Oil Recovery, pp. 1–11.

Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P., 1992. Numerical Recipes in C: the Art of Scientific Computing, second ed. Cambridge University Press, Cambridge, UK, 994 pp.

Shaw, J.H., Shearer, P.M., 1999. An elusive blind-thrust fault beneath metropolitan Los Angeles. Science 283, 1516–1518.

Suppe, J., 1983. Geometry and kinematics of fault-bend folding. American Journal of Science 283, 684–721.

Suppe, J., Medwedeff, D.A., 1990. Geometry and kinematics of fault propagation folding. Eclogae Geologicae Helvetiae 83, 409–454.

Zehnder, A.T., Allmendinger, R.W., 2000. Velocity field for the trishear model. Journal of Structural Geology 22, 1009–1014.