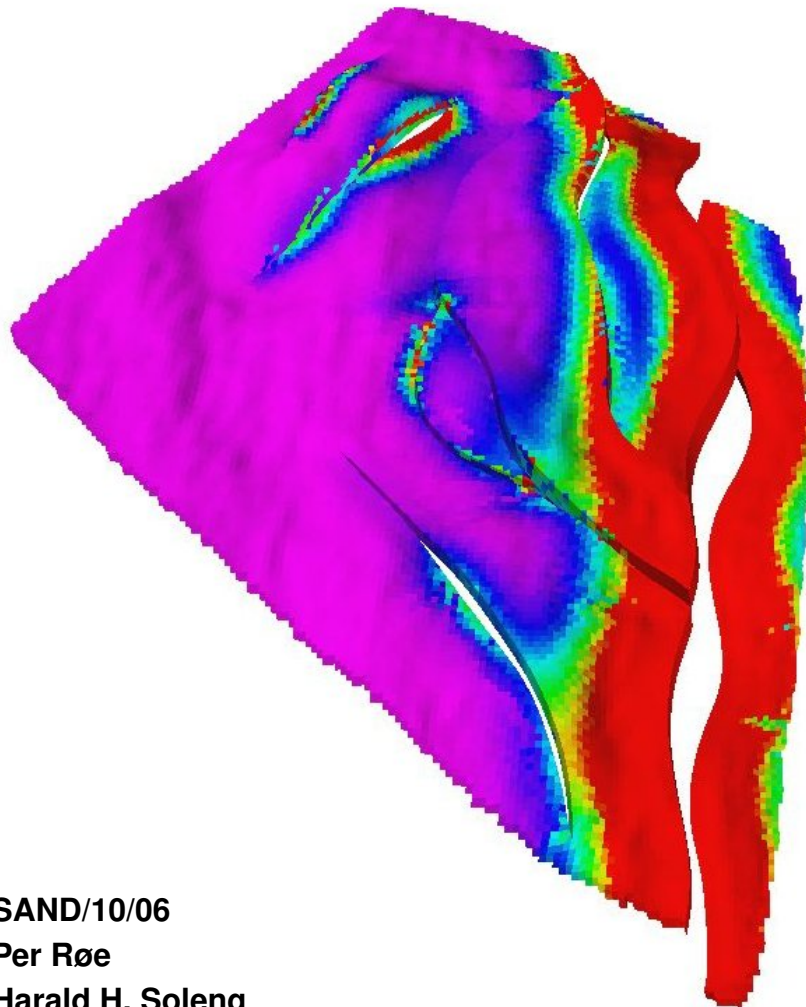


Strain calculations in Havana

Algorithm and implementation



Note no
Authors

Date

SAND/10/06
Per Røe
Harald H. Soleng
1st December 2006

The authors

Per Røe is research scientist at the Norwegian Computing Center. Harald H. Soleng is chief research scientist at Norwegian Computing Center. He holds a Dr. Philos. degree in theoretical physics.

Norwegian Computing Center

Norsk Regnesentral (Norwegian Computing Center, NR) is a private, independent, non-profit foundation established in 1952. NR carries out contract research and development projects in the areas of information and communication technology and applied statistical modeling. The clients are a broad range of industrial, commercial and public service organizations in the national as well as the international market. Our scientific and technical capabilities are further developed in co-operation with The Research Council of Norway and key customers. The results of our projects may take the form of reports, software, prototypes, and short courses. A proof of the confidence and appreciation our clients have for us is given by the fact that most of our new contracts are signed with previous customers.

Title	Strain calculations in Havana
Authors	Per Røe , Harald H. Soleng
Date	1st December 2006
Publication number	SAND/10/06

Abstract

In the FaultFacies project a new method for modeling of fault zones is developed. As part of this effort it might be feasible to use strain as a conditional parameter for the modeling of facies inside the fault zone. Havana already can handle grids and faults, and also has a displacement model for faults, and hence had what was needed for calculating strain. This document describes the strain calculation algorithm implemented in Havana. The strain was calculated by taking the deformation gradient of the displacement field defined for a set of faults. The strain was thereafter mapped back into the grid, and can be used in other parts of the FaultFacies workflow.

Keywords	Structural modelling, fault core, strain
Target group	
Availability	Open
Project	Fault facies
Project number	808005
Research field	Structural modelling
Number of pages	16
© Copyright	Norwegian Computing Center

Contents

1	Introduction	7
2	The parametric fault model	7
3	Calculation of strain.	8
4	Algorithm.	9
	4.1 Calculation of the deformation gradient	11
5	Examples.	11
6	Future work.	14
A	Havana parameter file	15

1 Introduction

Strain is a geometrical expression of deformation caused by stress on a physical body. It is an expression of the relative distortion of the body. Strain can be split into two components, longitudinal strain measuring stretching or compression, and shear strain measuring transvectional and twisting deformations.

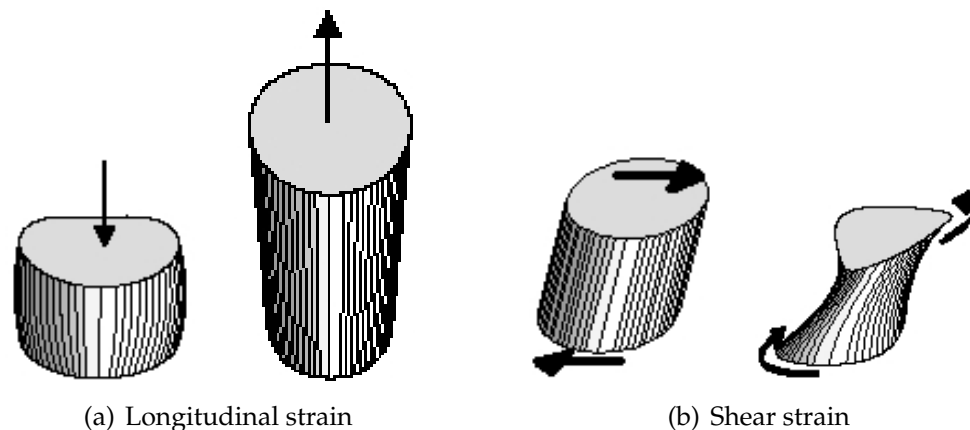


Figure 1. Forms of strain

In geology strain can be used as a measurement of how much a rock is deformed, and the density of fractures.

The FaultFacies project examines new methods for modeling of fault zones. Traditionally faults have been modeled as planar objects, but in reality it is a fault consist of a zone that has been deformed. The original facies can be fractured in various ways giving radically different petrophysical properties, and new fault breccias can be generated and cementation can occur inside the fault zone. It is possible to use strain as a conditional parameter for the modeling of the metamorphosis of the facies in the fault zones and the generation of fault facies.

Havana is a software tool for describing faults and their impact on fluid flow that is developed at Norwegian Computing Center. Havana can be used for simulation of subseismic faults, studying flow across faults, and for studies of uncertainty of faults on volumes. Havana can represent faults with a corresponding displacement field, and can also handle corner point grids.

2 The parametric fault model

Havana supports fault in several formats, but the main format for large faults is the parametric fault model (PFM). In this model the faults are represented as

a fault plane, and a corresponding volume that has been displaced. A displacement vector can be calculated for each point in the faulted volume, and it is also possible to calculate the inverse displacement vector.

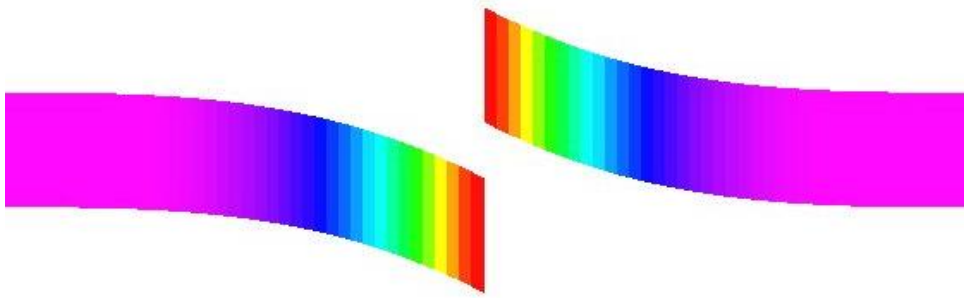


Figure 2. Cross section of a PFM fault

The displacement away from the fault goes as

$$\left(1 - \frac{r}{R}\right)^2 D, \quad \text{for } r \in [0, R], \quad (1)$$

where R is the reverse drag for the fault, D is the fault displacement, and r is the distance away from the fault surface. The reverse drag R specifies how far away from the fault surface the volume is displaced.

The fault is defined by a set of pillars at the fault surface, and the extent of the faulted volume is defined by another set of pillars on each side of the fault surface at the distance away from the fault specified by the reverse drag.

3 Calculation of strain

We have an original grid configuration \mathbf{x} and a displacement field \mathbf{u} defined from the displacement given by the PFM model. The displaced grid \mathbf{x}' is then given by

$$\mathbf{x}' = \mathbf{x} + \mathbf{u}. \quad (2)$$

The *deformation gradient* \mathbf{F} measures how much the displacement varies inside the grid and is defined as

$$\mathbf{F} \equiv \nabla \mathbf{x}'. \quad (3)$$

In component notation, we have

$$F^i_j \equiv \delta^i_j + \frac{\partial u^i}{\partial x^j}, \quad (4)$$

where δ^i_j is the Kronecker delta

$$\delta^i_j \equiv \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases} \quad (5)$$

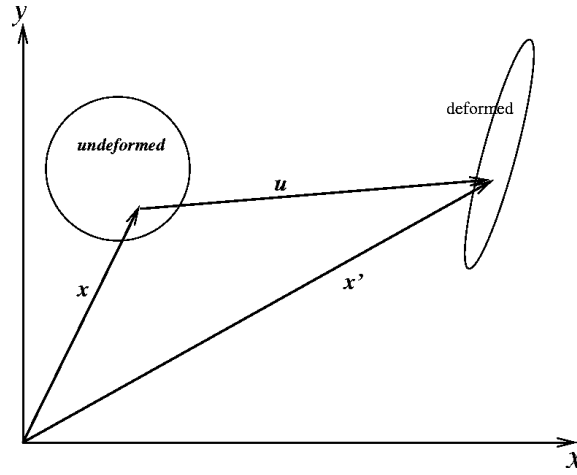


Figure 3. The displacement field

The Cauchy deformation tensor is then defined as

$$B^i_j \equiv \sum_k F^{ki} F_{kj}. \quad (6)$$

Due to symmetry, this tensor has six independent components.

The strain metric that we calculate is the maximal principal stretch of deformation S defined as

$$S \equiv \sqrt{\frac{1}{\lambda_{min}}}, \quad (7)$$

where λ_{min} is the smallest eigenvalue to the Cauchy deformation tensor.

4 Algorithm

The input data for the algorithm is the undeformed corner point grid in ECLIPSE format, and a set of faults in a PFM format. In the reverse case Havana can first use the inverse displacement operator on the deformed grid to generate the undeformed grid.

The PFM data can be converted from fault data exported from RMS in RPF format.

1. First we generate a regular grid that encloses the original undeformed grid. The regular grid is needed since we calculate deformation gradient by numerical derivation.
2. We locate the fault surfaces by looking at where the z-component of the displacement changes sign.
3. For each point in the regular grid we calculate the displacement using the displacement field that is defined cumulative displacement from a set of PFM

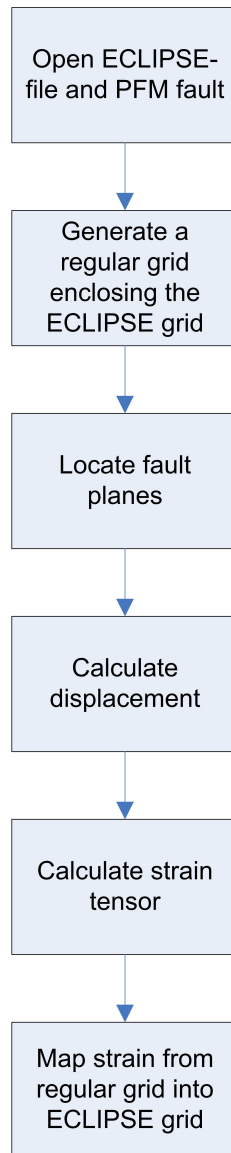


Figure 4. The workflow

- faults. The order of the faults is specified in the `.faultlist` file.
4. We calculate the deformation gradient by derivation of the displacement field numerically.
 5. The strain tensor is calculated for each point in the regular grid.
 6. We map the invariant of the strain tensor back into the undeformed ECLIPSE grid. The strain value is mapped into the STRAIN parameter.
 7. The undeformed grid is deformed using the IntoEclipse Havana action.

4.1 Calculation of the deformation gradient

The two-way approximation to the derivate is used to calculate derivate.

For a point with indices i, j , and k and with position $\mathbf{x} = (x_{i,j,k}^1, x_{i,j,k}^2, x_{i,j,k}^3)$ in the regular grid and the corresponding displacement $\mathbf{u} = (u_{i,j,k}^1, u_{i,j,k}^2, u_{i,j,k}^3)$, we calculate the element of the deformation gradient:

$$\mathbf{F} = \begin{bmatrix} 1 + \frac{u_{i+1,j,k}^1 - u_{i-1,j,k}^1}{x_{i+1,j,k}^1 - x_{i-1,j,k}^1} & \frac{u_{i,j+1,k}^1 - u_{i,j-1,k}^1}{x_{i,j+1,k}^2 - x_{i,j-1,k}^2} & \frac{u_{i,j,k+1}^1 - u_{i,j,k-1}^1}{x_{i,j,k+1}^3 - x_{i,j,k-1}^3} \\ \frac{u_{i+1,j,k}^2 - u_{i-1,j,k}^2}{x_{i+1,j,k}^1 - x_{i-1,j,k}^1} & 1 + \frac{u_{i,j+1,k}^2 - u_{i,j-1,k}^2}{x_{i,j+1,k}^2 - x_{i,j-1,k}^2} & \frac{u_{i,j,k+1}^2 - u_{i,j,k-1}^2}{x_{i,j,k+1}^3 - x_{i,j,k-1}^3} \\ \frac{u_{i+1,j,k}^3 - u_{i-1,j,k}^3}{x_{i+1,j,k}^1 - x_{i-1,j,k}^1} & \frac{u_{i,j+1,k}^3 - u_{i,j-1,k}^3}{x_{i,j+1,k}^2 - x_{i,j-1,k}^2} & 1 + \frac{u_{i,j,k+1}^3 - u_{i,j,k-1}^3}{x_{i,j,k+1}^3 - x_{i,j,k-1}^3} \end{bmatrix} \quad (8)$$

The deformation gradient can only be calculated in continuous volumes. At discontinuities, e.g. across faults this breaks down. This means that the deformation gradient gets higher for the same model when the grid resolution gets smaller. If for example \mathbf{x}_{ijk} and \mathbf{x}_{i-1jk} is on one side of the fault while \mathbf{x}_{i+1jk} is on the other, the formula above would give unreasonably high strain value. To mend this we only use the one-way approximate to calculate the derivate, for example:

$$F_{11}^1 = 1 + \frac{u_{i,j,k}^1 - u_{i-1,j,k}^1}{x_{i,j,k}^1 - x_{i-1,j,k}^1}. \quad (9)$$

5 Examples

Figures 5, 6 and 7 shows strain computed in three grids. The first grid is a single straight vertical fault. The next example shows three intersecting faults. Figure 7 shows strain computed in a complex case with six normal faults of which 4 faults intersect.

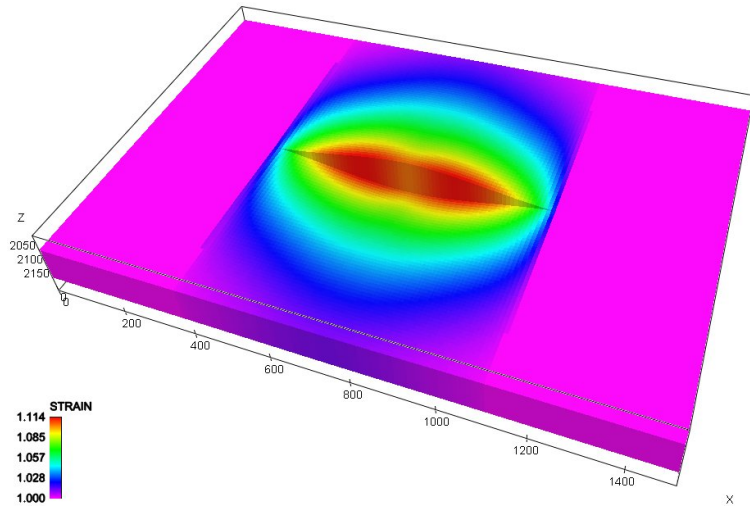


Figure 5. Single fault

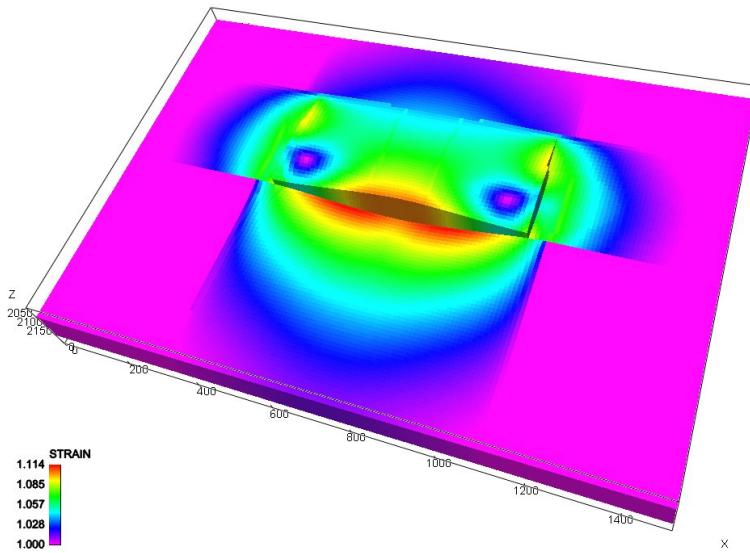


Figure 6. Three faults

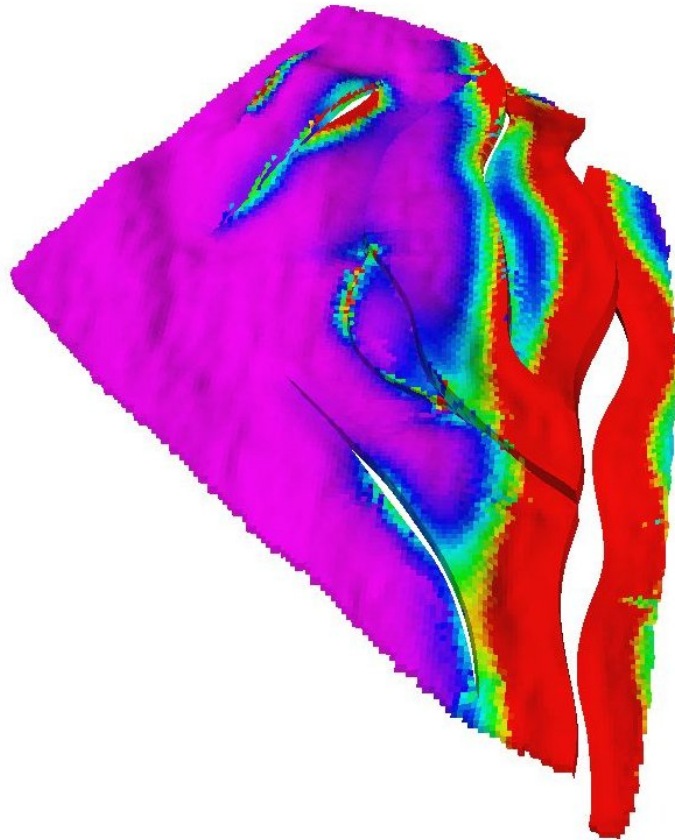


Figure 7. Complex pattern

6 Future work

The following future work tasks are suggested:

- Better interpolation of strain values back into the original grid. Instead of just taking the value of the cell in the regular grid closest to the center of the cell in the undeformed grid, a weighted average of the strain values for all cells in the regular grid encompassed by the cell in the undeformed grid could be used.
- Modify fault displacement field representation of faults that are truncated by the edge of the grid. In the present implementation we get large strain values just at the end of the grid, at the edge of the area where the grid displacement operator is defined.

- The PFM fault does not always define the fault plane properly, resulting in that some cells are placed on the wrong side of the fault plane.

To prevent this the side of fault should be detected by looking at the grid geometry instead of side of the PFM fault plane. The side of fault should then be taken into account when calculating the deformation vectors.

- Presently the deformation gradient of the total displacement field for all the faults is calculated. In stead it is probably more correct to calculate the strain for each fault individually, and then aggregate these strains to get the total strain field.

A Havana parameter file

Below is an example of a model file computing the strain in a forward modeling example.

We run two actions in Havana, CalculateStrain and IntoEclipse. See the Havana manual for details about the commands and parameters.

First we run CalculateStrain. This action takes four commands; INPUT_ECLIPSE gives the input eclipse file, the undeformed grid. OUTPUT_ECLIPSE gives the name of the output ECLIPSE file. INPUT_FILES gives the name of the directory containing the faults in PFM format. GRID_SIZE gives the size of the regular grid in number of nodes.

Thereafter the IntoEclipse action is run. This action deforms the ECLIPSE grid according to the displacement field defined for the faults.

```
ACTION CalculateStrain \  
  
! Reading the Eclipse grid and permeability data:  
INPUT_ECLIPSE grids/Input/regular.GRDECL \  
  
! The output grid, with the strain keyword.  
OUTPUT_ECLIPSE grids/Output/StrainGrid.GRDECL \  
  
! Faults input directory.  
INPUT_FAULTS faults/Simple_PFM \  
  
! Size of grid used for calculations of the strain.  
GRID_SIZE 150 100 20 \  
  
ACTION IntoEclipse \  
  
! The input PFM faults.  
INPUT_FAULTS faults/Simple_PFM \  
  
! No fault sealing.  
NO_FAULTSURFACE_EFFECT \  
  
! The input Eclipse grid.  
INPUT_ECLIPSE  
grids/Output/StrainGrid.GRDECL \  
  
! The output Eclipse grid and Eclipse faults.
```

```
OUTPUT_ECLIPSE
SPECGRID COORD ZCORN ACTNUM STRAIN
grids/Output/StrainGrid_Faulted.GRDECL ;
FAULTS \
faults/EclipseFaults/eclipseFaults.dat \
```