



PERGAMON

Computers & Geosciences 28 (2002) 961–969

**COMPUTERS &
GEOSCIENCES**

www.elsevier.com/locate/cageo

FBF: a software package for the construction of balanced cross-sections[☆]

Juan Contreras*

*Departamento de Geología, Centro de Investigación Científica y de Educación Superior de Ensenada,
Km 107 Carretera Tijuana-Ensenada, Ensenada BC, 22860, Mexico*

Received 10 May 2001; received in revised form 23 February 2002; accepted 28 February 2002

Abstract

FBF is a series of modular programs coded in ANSI C++ to simulate thrust and normal faulting in cross-section. The employed deformation function preserves area and therefore balanced cross-sections can be obtained with these programs. The programs run in text mode and the source code can be ported to and compiled on most computer platforms.

The package consists of a preprocessor, a processor, and a postprocessor. The preprocessor generates meshes that represent stratigraphic units in their undeformed state. Units in the initial state can have tabular geometries, thin laterally, or a composite geometry. The processor carries out a forward simulation to obtain the dislocated state of the units induced by a fault of known shape and displacement. Finally, the postprocessor converts the processor output to a format suitable for plotting.

The structural models obtained with the package are in good agreement with structures observed in fold-and-thrust belts and in extensional areas. Moreover, FBF can be used to test activity sequences of faults, to calculate potential fields associated with subsurface structures, and to simulate more complex geological systems and processes such as the stratigraphic response to synsedimentary brittle deformation.

© 2002 Elsevier Science Ltd. All rights reserved.

Keywords: Fault-related folding; Brittle deformation simulation; Cellular automaton

1. Introduction

Contreras and Suter (1990) presented a kinematic model for the simulation of fault-related folding in two dimensions that used a cellular automaton to carry out a forward simulation of the deformation induced by the tectonic transport over a fault surface. The model is useful to construct balanced cross-sections in areas deformed by thin-skinned tectonics since the employed deformation function conserves area. The computational implementation of the model was described in

Contreras (1991). The source code presented there, however, is not portable because the use of the Turbo Pascal compiler v. 3.07 is necessary to generate the executable program and because it contains calls to commercial subroutines (Turbo Pascal Graphix Toolbox) that are specific of the MS-DOS operative system. Moreover, the use of Pascal as a computer language has been replaced by other languages such as C++ and FORTRAN 90/95, which offer the same functionality of Pascal. This article presents a new computational implementation of the cellular automata of Contreras and Suter (1990) called fault-bend folding (FBF) that includes the following improvements: (i) the source code is written in ANSI C++; (ii) the design is modular and uses the object-manipulation capabilities of C++; (iii) calls to commercial subroutines have been eliminated;

[☆]Code available from server at <http://www.iarn.org/CGE-ditor/index.htm>

*Fax: +52-646-175-0559.

E-mail address: juanc@cicse.mx (J. Contreras).

(iv) the programs presented here can now handle several and larger meshes; and (v) more complex initial mesh geometries can be defined.

2. Description of the model

The construction of balanced cross-sections by computer can be addressed in two different ways. The first one consists in designing a CAD environment where simple graphics primitives and interpolation–extrapolation techniques are used to reconstruct the subsurface and eroded parts of structures (Kligfield et al., 1986; Moretti and Larrère, 1989). These may include the geometrical relations derived by Chamberlain (1910), Suter (1981), and Suppe (1983) and thus a bed length or an area-balanced cross-section can be constructed. The second approach consists in performing a forward simulation of the deformation process to find a deformation sequence by trial and error that matches the observed geological structures. FBF is based on the second approach; it uses an empirical deformation function derived from geological observations to calculate the dislocated state produced by faulting. The crux of the model consists in approximating the geometry of a fault by a series of straight segments and moving the hanging wall block parallel to the fault trace, whereas the footwall block remains fixed (Fig. 1). The axial planes between the straight fault segments define regions where the dislocation vectors remain constant. The deformed state is given by the following equation (detailed descriptions can be found in Contreras and

Suter, 1990):

$$\begin{bmatrix} p'_x \\ p'_y \end{bmatrix} = \begin{bmatrix} p_x + \Delta u \cos \theta \\ p_y + \Delta u \sin \theta \end{bmatrix}, \quad (1)$$

where p is the initial position of a material particle, p' its deformed position, u the magnitude of the displacement along the fault, and θ the dip of the underlying fault segment. In other areas of the physical sciences, this approach is known as cellular automaton (Toffoli, 1984). The deformation function has the important attribute that preserves area (Contreras and Suter, 1990, 1997). The cross-sections obtained with the cellular automaton are able to reproduce field observation as will be show in subsequent sections of this article and are similar to cross-sections constructed with geometrical techniques that preserve bed length (Suppe, 1983; Mitra, 1986, Cruickshank et al., 1989). Moreover, this model is able to generate balanced cross-sections without invoking any assumptions about the deformation function (e.g. conservation of bed length or area).

3. Description of the programs

The current stable version of FBF, labeled 0.2.01, is distributed under the terms of the General Public License (GNU)¹ and can be freely downloaded and redistributed.^{2,3} The FBF source code as well as binary packages are available for Linux and MS-Windows systems (running in a DOS window). The source code is written in ANSI C++ and consists of approximately 2000 lines of instructions. The source code package contains a makefile that facilitates its compilation. Since the FBF programs run in text mode only the standard C and C++ libraries are needed to compile them. Though FBF can be compiled for MS Windows systems, it was developed in Linux, and therefore this and other UNIX environments (including Mac OS X) are more suitable to run these programs. FBF consists of a preprocessor (fbfmesh), a processor (fbfault), and a postprocessor (fbfplot). The design of these programs partially follows those of Generic Mapping Tools (Wessel and Smith, 1991),⁴ Graphing Language,⁵ and finite element programs. The FBF programs are not interactive and do not use the mouse, rather they parse a series of

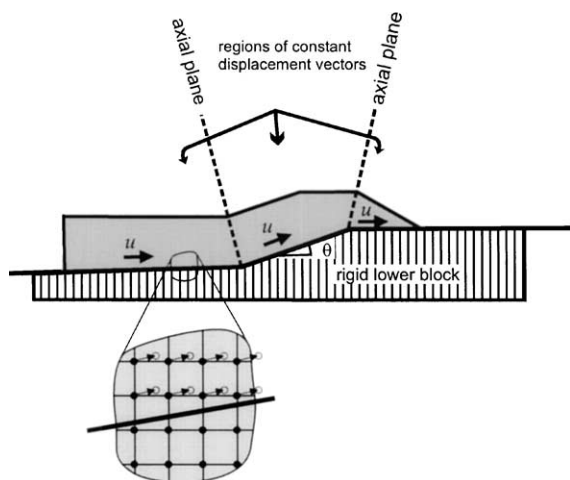


Fig. 1. Sketch showing kinematic model used to simulate deformation induced by faulting. Model consists of cellular automaton that displaces nodes of grid of material points parallel to fault surface. Axial planes define regions of constant displacement vectors. Cellular automaton can model thrust faults as well as normal faults.

¹GNU's Not Unix!. <http://www.gnu.org/licenses/licenses.html>.

²Computers & Geosciences searchable database. <http://www.iamg.org/CGEditor/index.htm>.

³FBF, fault bend folding. <http://www.cicese.mx/~juanc/fbf/fbf.html>.

⁴GMT—The Generic Mapping Tools. <http://gmt.soest.hawaii.edu>.

⁵Gri: a language for scientific illustration. <http://gri.sourceforge.net>.

parameters from input data files and from the command line, which control the execution of the programs. Once the calculations are completed, they are stored in a series of output files. The input files are text files with the parameters defining the geometry of the initial undeformed stratigraphic units or geometry of the faults. These parameters are specified in the following way:

$\langle \text{parameter} \rangle = \langle \text{value} \rangle$,

where *value* is an integer, a float, or a string of characters without blank spaces. The following conventions are used to parse the parameters: values in *x* (horizontal axis) increase to the right, values in *y* (vertical axis) increase upward, and angles are positive in the counter-clockwise direction measured from the *x*-axis. Comments can be introduced in the input files to clarify their content; they must start with the # symbol and must not exceed 80 characters per line (see Appendixes A and B for examples of input files).

3.1. Program *fbfmesh*

Since FBF carries out a forward simulation of the deformation process one must specify the initial (undeformed) state of the involved stratigraphic units. The preprocessor *fbfmesh* assists in creating such a state by generating grids of equally spaced material points or grids that gradually taper in the *x* direction (Fig. 2). Tabular or wedge-like (trapezoidal) domain regions are acceptable approximations of the shape of stratigraphic units. The processor then evaluates the rules of the cellular automaton over the nodes of the grid to calculate the deformed state. The parameters parsed from the input files by the preprocessor *fbfmesh* are (Fig. 2, Appendix A):

rows, cols: number of columns and rows of the mesh. The maximum size of the grids is 100 rows by 500 columns. However, these maximum values can be further increased (if enough memory is available) by changing the source code and recompiling it.

dx, dy: separation distance between neighboring nodes in the horizontal and vertical direction (Fig. 2).

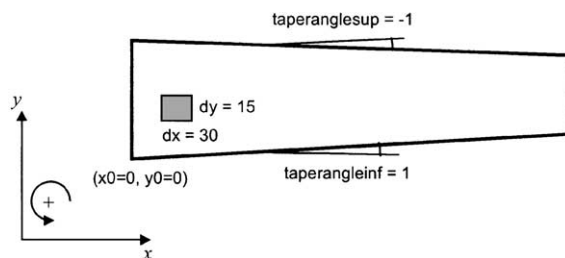


Fig. 2. Parameters used by preprocessor *fbfmesh* to generate grid of material points representing undeformed state. See text and Appendix A for further details.

x0, y0: coordinates of the node located at the lower left corner of the grid (Fig. 2).

taperangleinf, taperanglesup: optional angles (in degrees) that define a gradual thinning or thickening of the region in the horizontal direction (Fig. 2).

symbol: optional string of characters that defines the filling symbol of the mesh during plotting. Choices are “limestone”, “shale”, “beds”, “outline”, “hash”, and “lattice”.

color: optional integer that defines the color of the grid. The integer is used by *gnuplot* to plot the deformed state and must be specified according to its color scheme (see *fbfplot* below).

regionname: optional string of characters with the name of the model or region.

outpmesh: name of the output file storing the grid to be generated.

An example of an input file for *fbfmesh* is provided in Appendix A.

3.2. Program *fbfold*

The processor *fbfold* carries out the forward simulation of the deformation by iterating Eq. (1) over the nodes of the grids representing the stratigraphic units. It must be noted that *fbfold* handles one fault at a time and requires to define each fault in a separate file. However, this is a design shortcoming rather than an intrinsic limitation of the model. *Fbfold* can calculate the deformation of up to 50 faults. Again, this limit can be increased, if enough memory is available in the system, by changing the source code and recompiling it. As in the case of *fbfmesh*, the processor parses from an input file a series of parameters that define the geometry of the fault, its displacement, the meshes affected by the fault (output from *fbfmesh* or *fbfold*), and the name of the output files to store the deformed meshes and inactive faults. This is a list of the parameters parsed from the input file (Fig. 3):

nsegments: number of straight line segments constituting the fault trace.

x0, y0: starting point of the fault (Fig. 3). It should not be contained in the meshes affected by the fault; otherwise voids will result on the deformed grid.

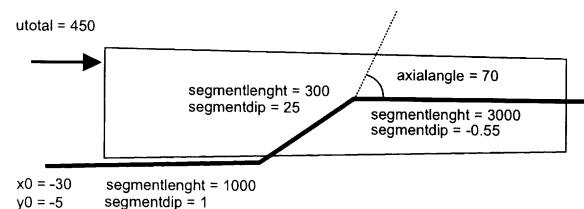


Fig. 3. Parameters used by processor to calculate the deformation induced by displacement along fault surface. See text and Appendix B for further details.

segmentlength, **segmentdip**: these parameters specify the length and dip (in degrees) of the straight segments of the fault (Fig. 3). As *fbfold* parses the information from each segment, it builds the geometry of the fault. Because of this, it is necessary to specify first the parameter **nsegments**, then **x0** and **y0**, followed by the various straight line segments of the fault.

axialangle: By default, *fbfold* uses as axial planes the lines bisecting the angles between adjacent straight fault segments. However, with this optional parameter it is possible to specify a different axial plane orientation. This parameter must be specified in conjunction with the parameters **segmentlength** and **segmentdip** (see Fig. 3 and example in Appendix B). Using **axialangle** it is possible to obtain fault-bend folds with constant bed length, in which the front of the propagating thrust sheet does not have a symmetrical axial plane (Suppe, 1983).

utotal: magnitude of the total displacement along the fault (Fig. 3).

inputmesh: string of characters with the name of the file containing the grid of material points in their initial configuration. It is possible to specify several meshes (stratigraphic units) by repeatedly using this parameter.

outputmesh: string of characters with the name of the file where the deformed mesh is going to be stored. If several input grids were specified (using **inputmesh**), then it becomes necessary to specify an equal number of output files. It is assumed that there is a direct correspondence in the naming scheme of input and output files, i.e., **inputmesh** file 1 is going to be stored in **outputmesh** file 1, **inputmesh** file 2 in **outputmesh** 2, and so on.

faultname: an optional string of characters with the name of the fault.

It is possible to use two additional parameters, **inputoldfaults** and **outputoldfaults**. These require a more detailed explanation. It is clear that the deformation induced by several faults can be modeled by using *fbfold* over its own output. As new faults are introduced the old ones become inactive and thus susceptible of being deformed by the currently active fault. The parameters **inputoldfaults** and **outputoldfaults** indicate the names of the files where to read and store these inactive faults.

Note that the faults should be defined in the direction of tectonic transport, e.g., if the displacement is from left to right, then **x0** and **y0** must occupy the leftmost position, and the parameters **segmentlength** and **segmentdip** must define successively segments located to the right of **x0** and **y0**. Appendix B illustrates, with an example, how to use these parameters.

3.3. Program *fbfplot*

Fbfplot takes as input the output files of the preprocessor and processor and generates as output a series of files with the structural model in a format

suitable for plotting. These files have a “tmp” extension and a bi-columnar structure. Thus, they can be easily imported into commercial spreadsheets or plotting programs. The use of *gnuplot*,⁶ however, is recommended to visualize the structural models obtained with FBF. *Gnuplot* is a free program to plot two- and three-dimensional data, which offers the advantage to have been ported to several platforms including Windows, UNIX, and Mac OS. It can also generate postscript files that can be imported into commercial vector-drawing programs. Moreover, *fbfplot* can generate a script file for *gnuplot*, which further simplifies the visualization of the cross-sections. The example in Appendix B illustrates the use of both, *fbfplot* and *gnuplot*.

4. Examples of cross-sections generated with FBF

The first example shows a possible faulting sequence for the Powell Mountain anticline of the Pine Mountain thrust sheet of the Appalachian thrust belt in Tennessee (Fig. 4). In this simulation, seven meshes with a rectangular geometry were employed to represent a series stratigraphic units with ages ranging from Cambrian to Pennsylvanian. The units are cut by three faults (Pine Mountain, Bales I, and Bales II thrust faults) of a flat-ramp-flat geometry that link three layer-parallel detachment surfaces (Woodward et al., 1985). The relative sequence of fault activation occurred in the direction of the tectonic transport.

Fig. 5 shows a comparison between a published section across part of the Sierra Madre Oriental fold-and-thrust belt in east-central Mexico (Suter, 1987) and a cross-section for the same area generated with FBF. This example shows the ability of FBF to deal with lithological units with complicated geometries. For example, three grids have been used to approximate the shape of the Valles-San Luis Potosi carbonate platform: a rectangular grid was used for the platform interior facies, and two laterally thinning grids were used for the platform margin facies. In a similar way, the grid employed for the basal unit (Las Trancas formation) tapers 0.5 along its base. In general, there is a good agreement between the structural model obtained with FBF and the section proposed by Suter (1987), specially in the upper part of the section in which there is a good structural control. On the other hand, the two sections differ at depth, like in the case of the inharmonic fold on the Lobo-Cinega thrust fault and the thrust fault in the core of the Pisaflores anticlinorium. These differences are due to the different assumptions made in the two techniques.

The cellular automaton used by FBF can be applied to model areas under extension as well. Fig. 6 shows an

⁶Gnuplot central. <http://www.gnuplot.info>.

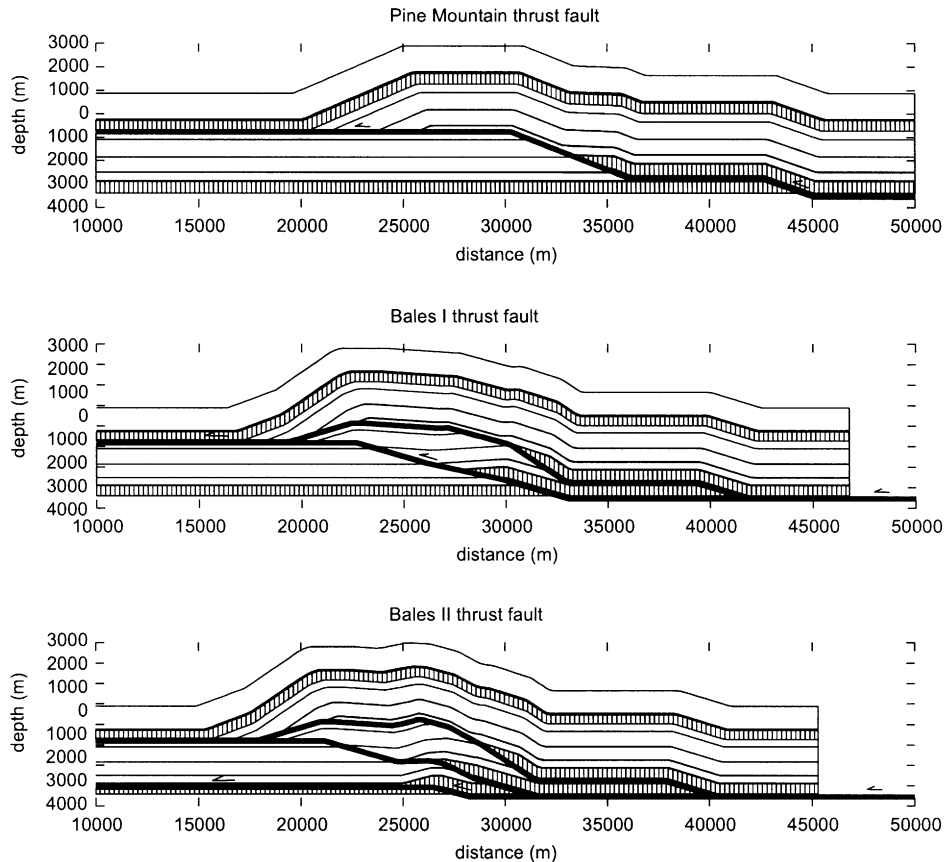


Fig. 4. Simulation of evolution of Powell Valley anticline (Appalachian thrust belt) carried out by FBF. Fault geometries and displacements are based on balanced cross-section by Woodward et al. (1985).

example with listric normal faults that merge in a detachment fault at depth; this faulting style is, for instance, observed along the margin of the Gulf of Mexico. Note that FBF cannot model the deformation of deep-seated normal faults because other processes such as isostasy and flexure of the crust are involved and cannot be reproduced by FBF (e.g., Contreras et al., 1997).

Finally, since FBF carries out a forward simulation, its applicability goes beyond the construction of balanced cross-sections. One of its potential uses is the simulation of more complicated geological systems and processes like the evolution of sedimentary basins affected by synsedimentary tectonism. Fig. 7 shows how the output of FBF was used in a numerical experiment involving sedimentation, erosion, and tectonism, based on the equation (Waltham, 1992)

$$\frac{\partial h}{\partial t} + v \cdot \nabla h = \kappa \frac{\partial^2 h}{\partial x^2} + s, \quad (2)$$

where h is the topographic elevation of the basin, v is the velocity field of the tectonic deformation produced by

faulting. The left-hand side of Eq. (2) represents the rate of change of the topographic elevation and includes an advective term ($v \cdot \nabla h$). The differential term on the right-hand side of Eq. (2) represents erosion or sedimentation (depending on the curvature of the topography), the constant κ controls the rate of erosion or sedimentation of the rocks, and s is a constant background sedimentation rate due to far sources. The values of the parameters employed in this simulation are $\kappa = 0.3 \text{ m}^2/\text{yr}$, $s = 0.02 \text{ mm}/\text{yr}$, and $|v| = 0.05 \text{ mm}/\text{yr}$.

The light gray layers in Fig. 7 precede thrusting and therefore have a uniform thickness, whereas the black layers were deposited during faulting and thin toward and on top of folds. This simple model is able to reproduce onlap patterns toward the core of the anticlines, which are often observed in active sedimentary basins (e.g., Trudgill et al., 1999; Hardy et al., 1996).

Another possible use of FBF is direct modeling of potential fields, where the goal is to reproduce an anomaly function in experimentally measured data originated by the subsurface structure. The calculation

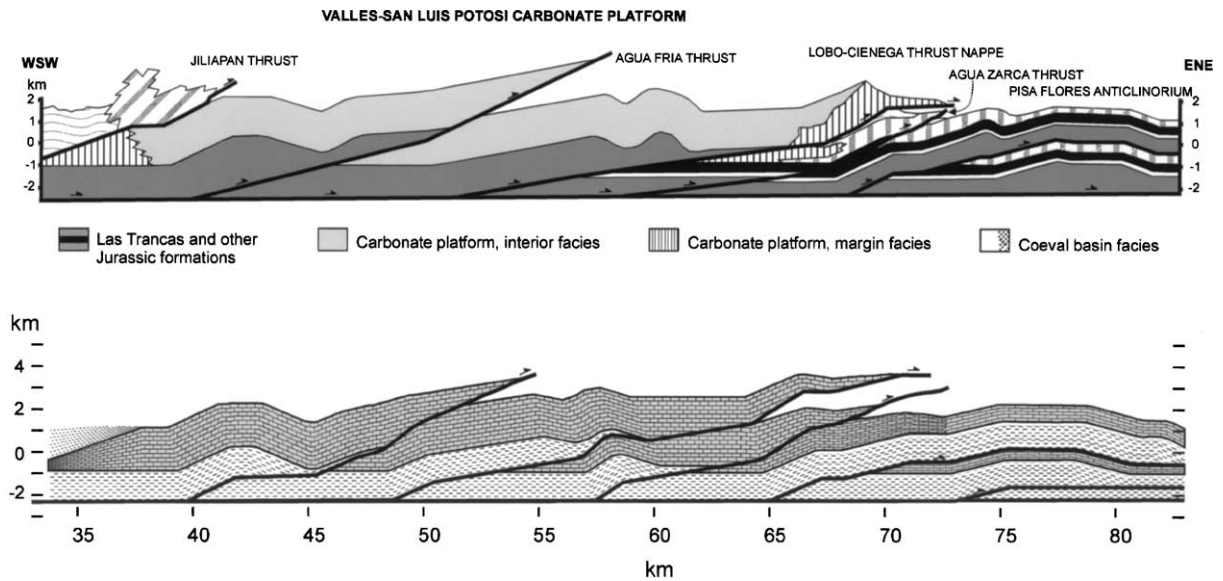


Fig. 5. Comparison between area-balanced section across Sierra Madre Oriental fold-and-thrust belt in east-central Mexico (Suter, 1987) and synthetic cross-section generated for same area by FBF. Note good agreement between two sections at upper structural levels. Discrepancies at depth result from different assumptions made by methodologies. Limestone and shale filling patterns of synthetic cross-section were generated by FBF package.

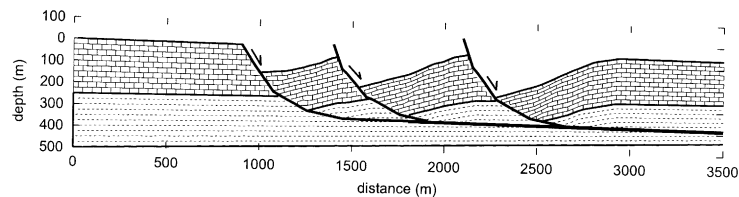


Fig. 6. Numerical simulation involving listric normal faults. Fault activation was from right to left. Two meshes of material points were employed in this simulation.

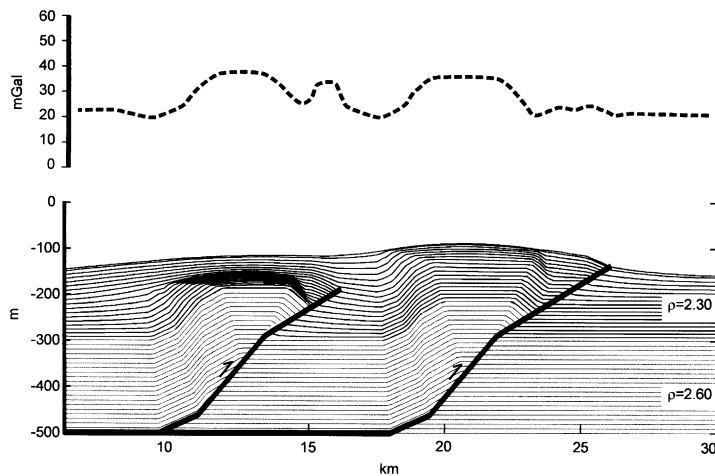


Fig. 7. Computer simulation incorporating deformation by thrust faulting and syntectonic sedimentation. Figure also shows gravity anomaly generated by synthetic structures. Density values are in g/cm³.

of such potential fields is an important constrain in the construction of balanced cross-sections (Kulander and Dean, 1978). Fig. 7 exemplifies this by showing the gravimetric response of the structure generated by the numerical experiment described above. In this model the basal layers have a density of 2.6 g/cm^3 (compacted sediments) and the syntectonic layers a density of 2.3 g/cm^3 (uncompacted sediments). As expected, the computed gravimetric anomaly curve presents maxima at the crest of anticlines and minima where the uncompacted sediments are thickest.

5. Conclusions

FBF is a series of computer programs to carry out a two-dimensional forward simulation of fault-related folding. The package consists of a preprocessor to define the undeformed state of geological formations; a processor, which uses a cellular automaton to displace material points parallel to the surface of a fault of known shape and displacement; and a postprocessor to convert the output of the processor to a format appropriate for plotting. The cross-sections generated with these programs are area-balanced and develop fault-related folds similar to the observed in natural examples. The cellular automaton can be used to model the deformation caused

by shortening as well as extension. Other possible uses of FBF include the computation of potential anomaly fields of the structural models generated with the package (e.g., gravimetric anomaly curve). Therefore, it is possible to test the consistency of the model by comparing the computed anomaly curve with experimentally measured data. The displacement field generated by FBF can also be used in the simulation of more complex geological systems such as the evolution of sedimentary basins affected by synsedimentary tectonism.

Acknowledgements

The author is grateful to Max Suter for a thorough review of this article and to reviewer Richard Thisli for his helpful comments. Support from CICESE Grant No. 644107 is acknowledged.

Appendix A

This is an example of an input file to `fbfmesh`, which illustrates the use of the parameters of this program. See text and Fig. 2 for details.

```
#=====
# Sample input file to fbfmesh. This is going to generate the
# initial undeformed state.
# Length units in meters, angles in degrees.
#=====
# Name of the region
regionname = limestone_unit
# Filling symbol to be used during plotting
symbol = limestone
# The following parameters define the size of the grid
cols = 200
rows = 15
# The following parameters define the coordinates of the node
# located on the lower left of the grid
x0 = 0.0
y0 = 0.0
# These parameters define the horizontal (x direction) and vertical
# separations (y direction) between the grid nodes
dx = 50.0
dy = 30.0
# The grid tapers along the x direction.
taperanglesup = -1
taperangleinf = 1
# Output file storing the resulting grid mesh
outputmesh = limestone.mesh
```

The limestone.mesh file containing the undeformed mesh is obtained by the following command
fbfmesh *<file with the mesh parameters>*

Appendix B

This is an example of an input file to fbplot, which illustrates the use of the parameters of this program. See text and Fig. 3 for details.

In the previous example fbplot is going to store the results of the simulation in the files deformed-limestone.mesh and old.fault. To create a graph with the deformed state it is necessary, first, to use postprocessor fbplot to convert the output of fbplot to a format suitable for plotting. Fbplot is used in the following way:

fbplot -m *<files with deformed meshes>* -f *<files with inactive faults>* -o *<output file with gnuplot scrip>*

For our example fbplot, must be used as follows:

fbplot -m deformed-limestone.mesh -f old.fault -o faultbendfold.plt

```
#=====
# Sample input file to fbplot. This file specifies a fault
# with a flat-ramp-flat geometry
# Length units in meters, angles in degrees.
#=====
# Name of the fault
faultname = thrust_ramp
# Number of straight line segments defining the geometry of the
# fault
nsegments = 3
# starting point of the fault
x0 = -30.0
y0 = -5.0
# This is the first segment, a flat
segmentlength = 1000.0
segmentdip = 1.0
# #
# 2nd segment, a ramp. Note the use of the parameter
# axialangle to define an axial plane on the upper part of
# the ramp with an orientation different from that of a
# bisectrix
segmentlength = 300.0
segmentdip = 25.0
axialangle = 70
# #
# 3rd segment, a flat
segmentlength = 30000
segmentdip = -0.55
# #
# Displacement along the fault surface
utotal = 450.0
# Input file storing the mesh in the initial state
inputmesh = limestone.mesh
# Output file storing the mesh in the deformed state
outputmesh = deformed-limestone.mesh
# Output file to store the inactive faults
outputoldfaults = old.fault
```

The program fbplot (processor module) is used in the following way:
fbplot *<file with fault parameters>*

Finally, to display the structural model (Fig. 8) with gnuplot, one must type
gnuplot faultbendfold.plt

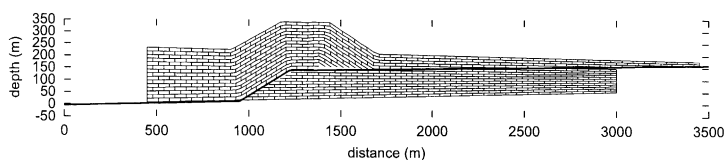


Fig. 8. Fault-bend fold resulting from sample input files listed in Appendixes A and B.

References

- Chamberlain, R.T., 1910. The Appalachian folds of Central Pennsylvania. *Journal of Geology* 18, 21–30.
- Contreras, J., 1991. Kinematic modeling of cross-sectional sequences by computer simulation: coding and implementation of the model. *Computers & Geosciences* 17, 1197–1217.
- Contreras, J., Scholz, C.H., King, G.C.P., 1997. A model of rift basin evolution constrained by first-order stratigraphic observations. *Journal of Geophysical Research* 102, 7673–7690.
- Contreras, J., Suter, M., 1990. Kinematic modeling of cross-sectional deformation sequences by computer simulation. *Journal of Geophysical Research* 95, 21913–21929.
- Contreras, J., Suter, M., 1997. A kinematic model for the formation of duplex systems with a perfectly planar roof thrust. *Journal of Structural Geology* 19, 269–278.
- Cruickshank, K.M., Neavel, K.E., Zhao, G., 1989. Computer simulation of growth of duplex structures. *Tectonophysics* 164, 1–12.
- Hardy, S., Poblet, J., McClay, K., Waltham, D., 1996. Mathematical modeling of growth strata associated with fault-related fold structures. In: Buchanan, P.G., Nieuwland, D.A. (Eds.), *Modern Development in Structural Interpretation, Validation and Modelling*. Geological Society Special Publication No 99, London, pp. 265–282.
- Kligfield, R., Geiser, P., Geiser, J., 1986. Construction of geologic cross sections using microcomputer systems. *Geobyte* 1, 60–66.
- Kulander, W.B., Dean, S., 1978. Gravity, magnetics, and structure, Allegheny Plateau/Western Valley and Ridge in West Virginia and adjacent states. *Western Virginia Geological and Economic Survey, Report of Investigation RI-27*, 97pp.
- Mitra, S., 1986. Duplex structures and imbricate thrust systems: geometry, structural position, and hydrocarbon potential. *American Association of Petroleum Geologists Bulletin* 70, 1087–1112.
- Moretti, I., Larrère, M., 1989. Computer-aided construction of balanced geological cross sections. *Geobyte* 4, 16–24.
- Suppe, J., 1983. Geometry and kinematics of fault-bend folding. *American Journal of Science* 283, 684–721.
- Suter, M., 1981. Strukturelles Querprofil durch den nordwestlichen Faltenjura, Mt-Terri-Randüberschiebung-Freiburg. *Eclogae Geologicae Helveticae* 74, 261–278.
- Suter, M., 1987. Structural traverse across the Sierra Madre Oriental fold-thrust belt in east-central Mexico. *Geological Society of America Bulletin* 98, 249–264.
- Toffoli, T., 1984. Cellular automata as an alternative to differential equations in modeling physics. *Physica* 10D (1–2), 117–127.
- Trudgill, B.C., Rowan, M.G., Fiduk, J.C., Weimer, P., Gale, P.E., Korn, B.E., Phair, R.L., Gafford, W.T., Roberts, G., Dobbs, S.W., 1999. The Perdido Fold Belt, Northwestern Deep Gulf of Mexico, Part 1: structural geometry, evolution, and regional implications. *American Association of Petroleum Geologists Bulletin* 83, 88–96.
- Waltham, D., 1992. Mathematical modeling of sedimentary basin processes. *Marine and Petroleum Geology* 12, 153–163.
- Wessel, P., Smith, W.H.F., 1991. Free software helps map and display data. *EOS Transactions, American Geophysical Union* 72 (441), 445–446.
- Woodward, N.B., Boyer, S.E., Suppe, J., 1985. *An Outline of Balanced Cross-sections*, 2nd Edn. Department of Geological Sciences, University of Tennessee, Studies in Geology, Vol. 11, 170pp.