Marmousi2 model

Trevor Irons

Data Type: *Synthetic 2D elastic model*
Source: *Allied Geophysical Laboratories (AGL)*
Location: *http://www.agl.uh.edu/*
Format: *SEGY*
Date of origin: *Development occured from 2002-2005*

INTRODUCTION

The Marmousi2 dataset is an extension and elastic upgrade of the classic Marmousi model. It was created by Allied Geophysical Laboratories (AGL). The Marmousi2 model has enjoyed widespread use and has been particularily insightful in amplitude versus offset (AVO) analysis, impedance inversion, multiple attenuation, and multicomponent imaging. AGL has publically released the data for reasearch use around the world.

Table 1 contains all the Marmousi2 files contained within the Madagascar repository.

```
-rw-r--r--    1 root    root    155653444  2005-05-05 05:30 density_marmousi-ii.segy
-rw-r--r--    1 root    root    155653444  2005-05-05 05:33 vp_marmousi-ii.segy
-rw-r--r--    1 root    root    155653444  2005-05-05 05:36 vs_marmousi-ii.segy
-rwxr-xr-x    1 root    root    4525264400 2005-05-05 07:06 obc_curl_v_1.segy
-rw-r--r--    1 root    root    2262634000 2005-05-05 07:54 obc_curl_v_2.segy
-rw-r--r--    1 root    root    4525264400 2005-05-05 09:30 obc_div_v_1.segy
-rw-r--r--    1 root    root    2262634000 2005-05-05 10:18 obc_div_v_2.segy
-rw-r--r--    1 root    root    4525264400 2005-05-05 11:50 obc_vx_1.segy
-rw-r--r--    1 root    root    2262634000 2005-05-05 12:38 obc_vx_2.segy
-rw-r--r--    1 root    root    3393949200 2005-05-05 13:45 obc_vz_1.segy
-rw-r--r--    1 root    root    3393949200 2005-05-05 14:55 obc_vz_2.segy
-rw-r--r--    1 root    root    4459728400 2005-05-05 16:27 surface_p1.segy
-rw-r--r--    1 root    root    2229866000 2005-05-05 17:14 surface_p2.segy
```

Table 1: A list of all files contained in the Marmousi2 repository

MODEL

The Marmousi2 model completely encapsulates the origional Marmousi model which was based on the Northern Quenguela Trough in the Quanza Basin of Angola. Lithologies include sandstones,shales, limestones and marls.

In total the Marmousi2 model is 3.5 km in depth and 17 km across. The model contains 199 horizons which make the model stratigraphicaly more complex than its predecessor. Additionally the water layer was extended to 450 meters.

As Marmousi2 is an elastic model both shear and primary velocities must be defined across the entire model. Additionally a density model is included. The files *vp_marmousi-ii.segy*, *vs_marmousi-ii.segy*, and *density_marmousi-ii.segy* contain the velocity and density

n1=2801	o1=0	d1=0.001249	label1=Depth	unit1=km
n2=13601	o2=0	d2=0.001249	label2=Position	unit2=km

Table 2: Header information for Marmousi2 velocity and density models

models for Marmousi2. These three files all share the same data spacing and their headers should be formatted similarily. This header format is shown in table 2

The file *marmousi2/model/SConstruct* is a SCons script that fetches the three model files (VP, VS, and Density), appends the header information as necessary and produces plots of the models. This file is reproduced in table 3 and the models themselves are shown in figures 1, 2, and 3.

```
from rsf.proj import *
# Fetch Files from repository
modelFiles=['vp_marmousi-ii.segy','vs_marmousi-ii.segy','density_marmousi-ii.segy']
outputFiles=['vp','vs','density']
for file in modelFiles:
    Fetch(file,"marm2")
# Convert Files to RSF
counter=0
for file in modelFiles:
    if file is 'vp_marmousi-ii.segy' or file is 'vs_marmousi-ii.segy':
        Flow(outputFiles[counter],file, ''' segyread tape=$SOURCE | put
            d1=.001249 d2=.001249 o1=0 o2=0 label1=Depth label2=Distance
            unit1=km unit2=km''')
    if file is 'density_marmousi-ii.segy':
        Flow(outputFiles[counter],file, ''' segyread tape=$SOURCE | put
            d1=.001249 d2=.001249 o1=0 o2=0 label1=Depth label2=Distance
            unit1=km unit2=km''')
    counter = counter+1
# Plotting Section
title=['Compressional\ Velocity\ Model','Shear\ Velocity\ Model','Density\ Model']
counter=0
for file in outputFiles:
    Flow(file+'small',file,'prep4plot inp=$SOURCE out=$TARGET',stdin=0,stdout=-1)
    if file is 'vp' or file is 'vs':
        Result(file,file+'small',
               '''
               grey  color=I gainpanel=a allpos=y title=%s
               scalebar=y screenratio=.205 screenht=2
               scalebar=y barlabel=Velocity barunit=km/s
               labelsz=4 wheretitle=t titlesz=6 barreverse=y
               ''' % title[counter])
    if file is 'density':
        Result(file,file+'small',
               '''
               grey  color=I gainpanel=a allpos=y title=%s
               scalebar=y screenratio=.205 screenht=2
               scalebar=y barlabel=Density barunit="g/cm\^3\_"
               labelsz=4 wheretitle=t titlesz=6 barreverse=y
               ''' % title[counter])
    counter = counter+1
End()
```

Table 3: SCons script generating images of the Marmousi2 model

SHOTS

Three sets of data were collected over this model. A near surface streamer survey, a vertical souding profile (VSP), and an ocean bottom cable (OBC) survey. Several sets of shot records are included in the Marmousi2 repository; multicomponent OBC data found in *obc_vx_#.segy* and*obc_vz_#.segy*, reduced data from the OBC cable found in *obc_div_v_#.segy* and *obc_curl_v_#.segy*, and streamer cable data found in *surface_p#.segy*. Each of these files was split into components to make them more managable. The # symbol

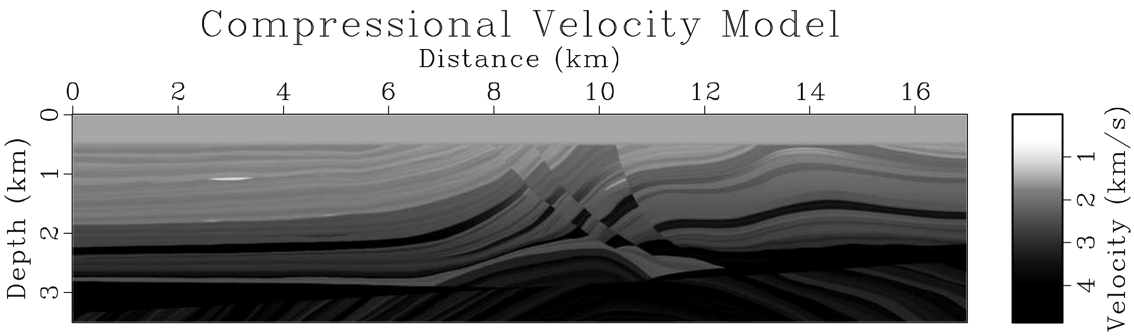

Figure 1: Marmousi2 P-wave velocity model

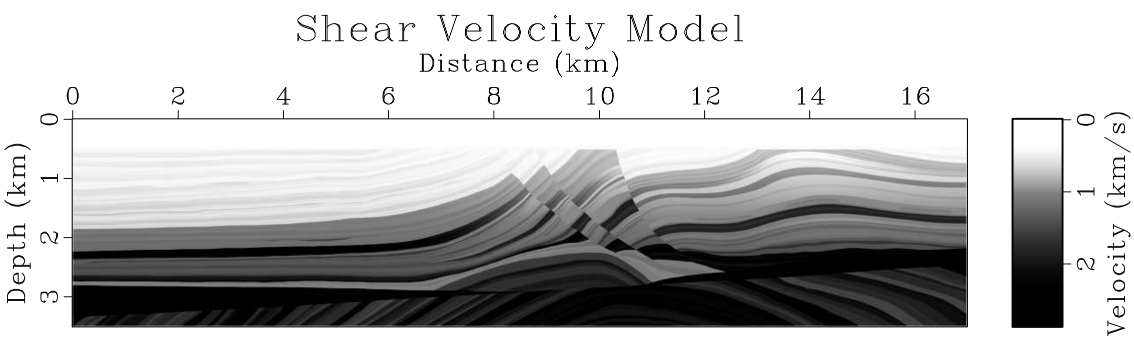

Figure 2: Marmousi2 S-wave velocity model

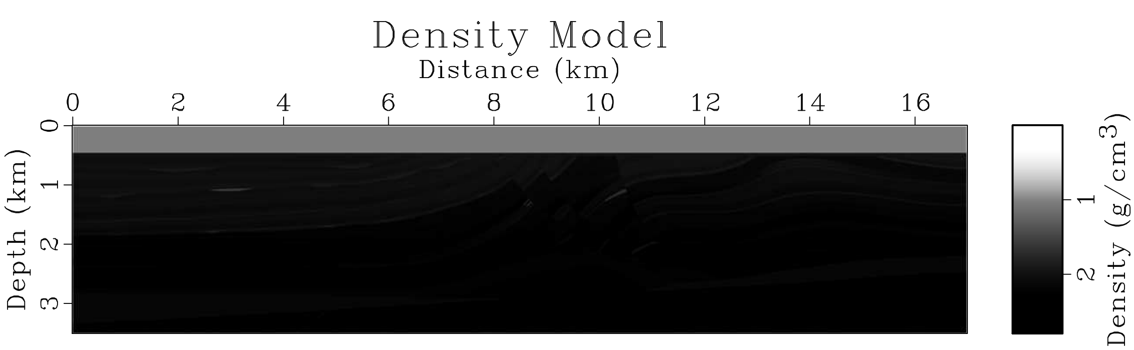

Figure 3: Marmousi2 Density Model

above correspont to either part number 1 or 2.

In all cases the source was an airgun located on a ship at depth of 10 m. The source began firing at 3 000 m along the horizontal x coordinate and continued firing every 25 m until 14 975 m.

OBC Surveys

The OBC cable was placed on the ocean floor at a depth of roughly 450 m. Multicomponent phones were spaced every 12.32 m along the entire length of the model. As the model is 2D only the x and z components of the wavefield were measured. Marmousi2 OBC survey data should have header information configured as shown in table 4.

n1=2500	o1=0	d1=0.002	label1=Depth Z	unit1=s
n2=1381	o2=0	d2=12.32	label2=Position X	unit2=m
n3=480	o3=3000	d3=25	label3=Shot-Coord	unit2=m

Table 4: Header information for Marmousi2 ocean bottom cable surveys

OBC Vz data

The file *marmousi2/vz /SConstruct* contains a SCons script that fetches the Vz component data files from the OBC survey, concatinates the segments, appends the header making a three axis file, (time, offset, and shot) and produces several plots of the data. This file is reproduced in table 5.

```
from rsf.proj import *
# Fetch Files from repository
Fetch("obc_vz_1.segy","marm2")
Fetch("obc_vz_2.segy","marm2")

# Convert Files to RSF and update header
Flow('obc_vz_1','obc_vz_1.segy',
     '''segyread read=d |
     put n2=1381 n3=240 o1=0 o2=0 o3=3000
     d2=12.32 d3=25''')
Flow('obc_vz_2','obc_vz_2.segy',
     '''segyread read=d |
     put n2=1381 n3=240 o1=0 o2=0 o3=11000
     d2=12.32 d3=25
     ''')
# Concatinate Datasets
Flow('vz',['obc_vz_1','obc_vz_2'],'cat ${SOURCES[0:2]} axis=3',stdin=0)

# Plot Data
Result('zero','vz',
       '''window $SOURCE  min2=0 max2=0 size2=1 |
       grey  color=I gainpanel=a label2=Position\ X unit2=m
       title=Zero\ Offset\ Data''')

Result('zero2','vz',
       '''window $SOURCE  min2=0 max2=0 size2=1 |
       grey  color=I gainpanel=a label2=Position\ X unit2=m
       title=Zero\ Offset\ Data''')

End()
```

Table 5: SCons script generating images of the Marmousi2 Vz data

OBC Vx data

Similar to the Vz data the file *marmousi2 /vx /SConstruct* contains a list of rules that tell Madagascar to gather the Vx data files, append the header and produce plots of the data. This script is reproduced in table 6

```
from rsf.proj import *
# Fetch Files from repository
Fetch("obc_vz_1.segy","marm2")
Fetch("obc_vz_2.segy","marm2")

# Convert Files to RSF and update header
Flow('obc_vz_1','obc_vz_1.segy',
     '''segyread read=d |
     put n2=1381 n3=240 o1=0 o2=0 o3=3000
     d2=12.32 d3=25''')
Flow('obc_vz_2','obc_vz_2.segy',
     '''segyread read=d |
     put n2=1381 n3=240 o1=0 o2=0 o3=11000
     d2=12.32 d3=25
     ''')
# Concatinate Datasets
Flow('vz',['obc_vz_1','obc_vz_2'],'cat ${SOURCES[0:2]} axis=3',stdin=0)

# Plot Data
Result('zero','vz',
       '''window $SOURCE  min2=0 max2=0 size2=1 |
       grey  color=I gainpanel=a label2=Position\ X unit2=m
       title=Zero\ Offset\ Data''')

Result('zero2','vz',
       '''window $SOURCE  min2=0 max2=0 size2=1 |
       grey  color=I gainpanel=a label2=Position\ X unit2=m
       title=Zero\ Offset\ Data''')

End()
```

Table 6: SCons script generating images of the Marmousi2 Vx data

OBC div data

The divergence operator was applied to the multicomponent OBC Pluto dataset. These files are *obc_div_v_1.segy* and *obc_div_v_2.segy*. Taking the divergence separates out the acoustic component of the data.

The file *marmousi2 /div /SConstruct* contains a list of rules that tell Madagascar to gather the div data files, append the header and produce plots of the data. This script is reproduced in table 7 and a plot of shot 50 is shown in figure **??**

OBC curl data

Similarily the curl operator was applied to the Pluto OBC data. These files are *obc_curl_v_1.segy* and *obc_curl_v_2.segy* These curl data contain only data generated by the elastic component of the field.

The file *marmousi2 /curl /SConstruct* contains a list of rules that tell Madagascar to gather the curl data files, append the header and produce plots of the data. This script is reproduced in table 8

```
from rsf.proj import *
# Fetch Files from repository
Fetch("obc_div_v_1.segy","marm2")
Fetch("obc_div_v_2.segy","marm2")

# Convert Files to RSF and update header
Flow('obc_div_v_1','obc_div_v_1.segy', '''segyread tape=$SOURCE |
        put n2=1381 n3=320 o1=0 o2=0 o3=3000
        d2=12.32 d3=25 label1=Depth\ Z label2=Distance\ X label3=Shot-Cord
        unit1=s unit2=m unit2=m''')

Flow('obc_div_v_2','obc_div_v_2.segy', '''segyread tape=$SOURCE |
        put n2=1381 n3=160 o1=0 o2=0 o3=11025
        d2=12.32 d3=25 label1=Depth\ Z label2=Distance\ X label2=Shot-Cord
        unit1=s unit2=m unit3=m''')
# Concatinate Datasets
Flow('div',['obc_div_v_1','obc_div_v_2'],'cat ${SOURCES[0:2]} axis=3',stdin=0)

# Plot Data
Result('movie','div','''window $SOURCE
               min3=4250 max3=4250 n3=1 |
               grey color=I gainpanel=a
               title=OBC\ Div\ Shot\ 50''')

End()
```

Table 7: SCons script generating images of the Marmousi2 Vx data

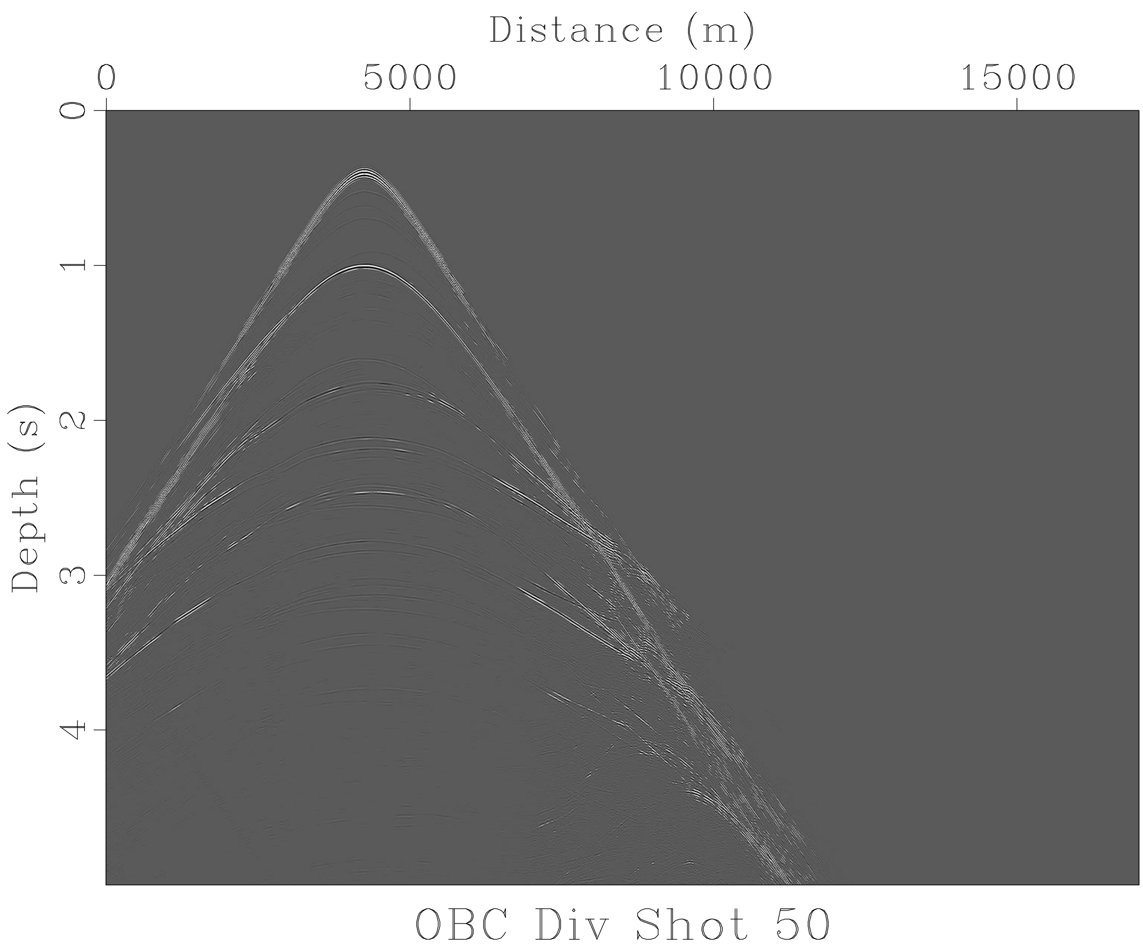

Figure 4: Marmousi2 shot 50 of div data.

```
from rsf.proj import *
import fdmod
# Fetch Files from repository
Fetch("obc_curl_v_1.segy","marm2")
Fetch("obc_curl_v_2.segy","marm2")

# Convert Files to RSF and update header
Flow('obc_curl_v_1','obc_curl_v_1.segy','''segyread tape=$SOURCE
        | put n2=1381 n3=320 o1=0 o2=0 o3=3000
        d2=12.32 d3=25 label1=Z label2=X label3=Shot
        unit1=s unit2=m unit2=m''',stdin=0)
Flow('obc_curl_v_2','obc_curl_v_2.segy','''segyread tape=$SOURCE
        | put n2=1381 n3=160 o1=0 o2=0 o3=11025
        d2=12.32 d3=25 label1=Z label2=X label2=Shot
        unit1=s unit2=m unit3=m''',stdin=0)

# Use fdmod for c graphing function
par = {
    'nt':2500, 'dt':0.002,'ot':0,  'lt':'t','ut':'s',
    'nx':1381, 'ox':0, 'dx':12.32 ,'lx':'x','ux':'km',
    'nz':480, 'oz':0,  'dz':12.32 ,'lz':'z','uz':'km',
    }
# add F-D modeling parameters
fdmod.param(par)



# Concatinate Datasets
Flow('curl',['obc_curl_v_1','obc_curl_v_2'],'cat ${SOURCES[0:2]} axis=3',stdin=0)

# Plot Data
Result('curlShot50','curl','''window $SOURCE
               min3=4250 max3=4250 size3=1 |
               grey color=I gainpanel=a
               title=OBC\ Curl\ Shot\ 50''')


Result('movie','curl','''window $SOURCE
               j3=20 |
               grey color=I gainpanel=a
               title=OBC\ Curl\ Shot\ 50''')

Result('curlFD','curl',fdmod.cgrey('j3=20 bias=1.5 pclip=98',par))


End()
```

Table 8: SCons script generating images of the Marmousi2 curl data

Streamer Surveys

The streamer survey was not traditional in the sense that it employed a 17 km long static streamer which spanned the entire model. In total there were 1 361 single component hydrophones spaced every 12.5 m at a depth of 5 m. This unrealistic geometry was chosen both for simplicity and to allow maximum utility of the data. The table 9 outlines the values that streamer data files headers should have.

n1=2500	o1=0	d1=0.002	label1=Depth Z	unit1=s
n2=1361	o2=0	d2=12.5	label2=Position X	unit2=m
n3=480	o3=3000	d3=25	label3=Shot-Coord	unit2=m

Table 9: Header information for Marmousi2 streamer surveys

FINITE DIFFERENCE MODELING

Madagascar may be used to perform finite difference modeling of the wavefield and receiver data. The tools to perform these tasks are found in the fdmod package.

Note, these processes are somewhat computationally intensive. I performed the majority of these models on a machine with a 3 GHz processor and 1.5 MB of RAM and most of the models took on the order of 3 hours to perform.