Learning Madagascar:

on a per need basis

Tariq Alkhalifah

Seismic Analysis Group (SWAG)
KAUST
swag.kaust.edu.sa

From the net

Q: What is the difference between a Ph.D. in mathematics and a large pizza? A: A large pizza can feed a family of four...

A mathematician, an engineer, and a computer scientist are vacationing together. They are riding in a car, enjoying the countryside, when suddenly the engine stops working. The mathematician: "We came past a gas station a few minutes ago. Someone should go back and ask for help." The engineer: "I should have a look at the engine. Perhaps, I can fix it." The computer scientist: "Why don't we just open the doors, slam them shut, and see if everything works again?"

The issue

The seemingly complex makeup:

- **LaTeX, python (scons????), and C.**
- **The file system (where is everything?).**
- **incomplete and not-up-to-date documentation.**
- **Any info on the libraries? (C and Python)?**

The objective

- **Getting from Madagascar what you need: even if we have to get dirty.**
- **Example paper: Scanning for the anisotropy parameter η, soon to be published**

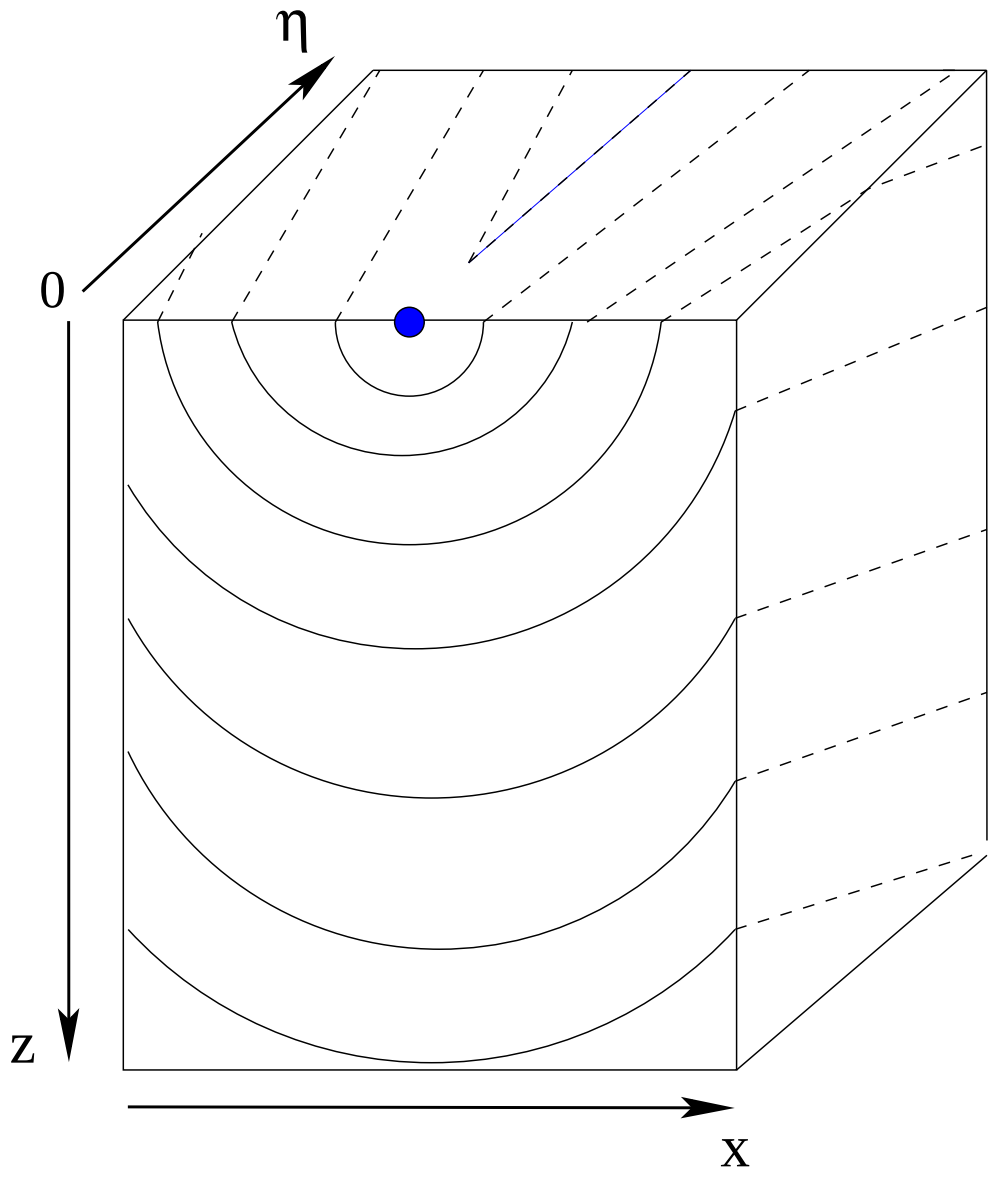
Paper: the issue

Uncertainty in anisotropic medium parameters in complex media: The trade-off between anisotropy and inhomogeneity

Paper: scope

- **Perturbation of the eikonal equation for small η**
- **Fast marching approach to solve the equations**
- **A homogeneous approximation**
- **A Marmousi example**

Scanning for η



First step: the paper template

- From the book directory, or one of your own papers
- Read the paper and see if you could form a good introduction from that paper.
- Copy it to your directory compile it (the whole directory)
- Remove unwanted examples and corresponding directories (compile again).
- Write your abstract (a road map).

For the η scan paper

"A variational formulation of the fast marching eikonal solver"
Sergey Fomel

book/sep/fmeiko

The coefficients of powers of η

$$v_v^2 \frac{\partial \tau_0}{\partial z}\frac{\partial \tau_i}{\partial z} + v^2 \frac{\partial \tau_0}{\partial y}\frac{\partial \tau_i}{\partial y} +$$

$$v^2 \frac{\partial \tau_0}{\partial x}\frac{\partial \tau_i}{\partial x} = f_i(x, y, z)$$

with $i = 1, 2, 3.$

$$\tau(x, y, z) \approx \tau_0(x, y, z) + \tau_1(x, y, z)\eta + \tau_2(x, y, z)\eta^2 + \tau_3(x, y, z)\eta^3$$

Second step: An equivalent code

- Covers the dimensions of the input and output fields (also velocity if needed)
- Locate whether their is an example that runs it (search SConstruct)
- Copy the code (and all related files) to your directly and compile it
- fetch the example directory that runs it to your papers directory and run the example
- Delete unwanted parts and then compile it again (The art of deletion).

Thus I used "Meikonal.c"

Modifications 1

Meikonal.c

```
/* Fast marching eikonal solver (3-D). */
/*
  Copyright (C) 2004 University of Texas at Austin

  This program is free software; you can redistribu
  it under the terms of the GNU General Public Lice
```

MeikEta.c

```
/* Eta differntial eikonal solver (3-D). */
/*
  Copyright (C) 2009 KAUST

  This program is free software; you can redistribu
  it under the terms of the GNU General Public Lice
```

Modifications 2

Meikonal.c

```
#include <math.h>

#include<rsf.h>

#include "fastmarch.h"
```

MeikEta.c

```
#include <math.h>

#include<rsf.h>

#include "fastEta.h"
```

Modifications 3

Meikonal.c

```
int b1, b2, b3, n1, n2, n3, i, nshot, ndim, is,
float br1, br2, br3, o1, o2, o3, d1, d2, d3, slo
float **s, *t, *v;
char *sfile;
bool isvel, sweep, plane[3];
sf_file vel, time, shots;
```

MeikEta.c

```
int b1, b2, b3, n1, n2, n3, i, nshot, ndim, is,or
float br1, br2, br3, o1, o2, o3, d1, d2, d3, slo
float **s, *t, *bt;
char *sfile;
bool isbtime, plane[3];
sf_file btime, time, shots;
```

Modifications 4

Meikonal.c

```
if(!sf_getint("order",&order)) order=2;
/* [1,2] Accuracy order */
```

MeikEta.c

```
if(!sf_getint("order",&order)) order=2;
/* [1,2] Accuracy order */

if(!sf_getint("sorder",&sorder)) sorder=2;
/* [1,2,3] Accuracy order of the source perturb

if(!sf_getfloat("eta",&eta)) eta=0.0;
/* The value of the constant eta */
```

The heart

MeikEta.c

```
for( is = 0; is < nshot; is++) {
     fastEta(t,bt,p, plane,
                  n3,n2,n1,
                  o3,o2,o1,
                  d3,d2,d1,
                  s[is][2],s[is][1],s[is][0],
                  b3,b2,b1,
                  order,sorder,eta);

     sf_floatwrite (t,n123,time);
}
```

Modifications subroutine

fastmarch.c

```
#include <rsf.h>
/*^*/

#include "fastmarch.h"
```

fastEta.c

```
#include <rsf.h>
/*^*/

#include "fastEta.h"
#include "neighbors.h"
#include "pqueue.h"
```

Modifications subroutine

fastmarch.c

```
#include <rsf.h>
/*^*/

#include "fastmarch.h"
```

fastEta.c

```
#include <rsf.h>
/*^*/

#include "fastEta.h"
#include "neighbors.h"
#include "pqueue.h"
```

Choices

fastmarch.c

```
sf_pqueue_start();
sf_neighbors_init (in, d, n, order, time);
```

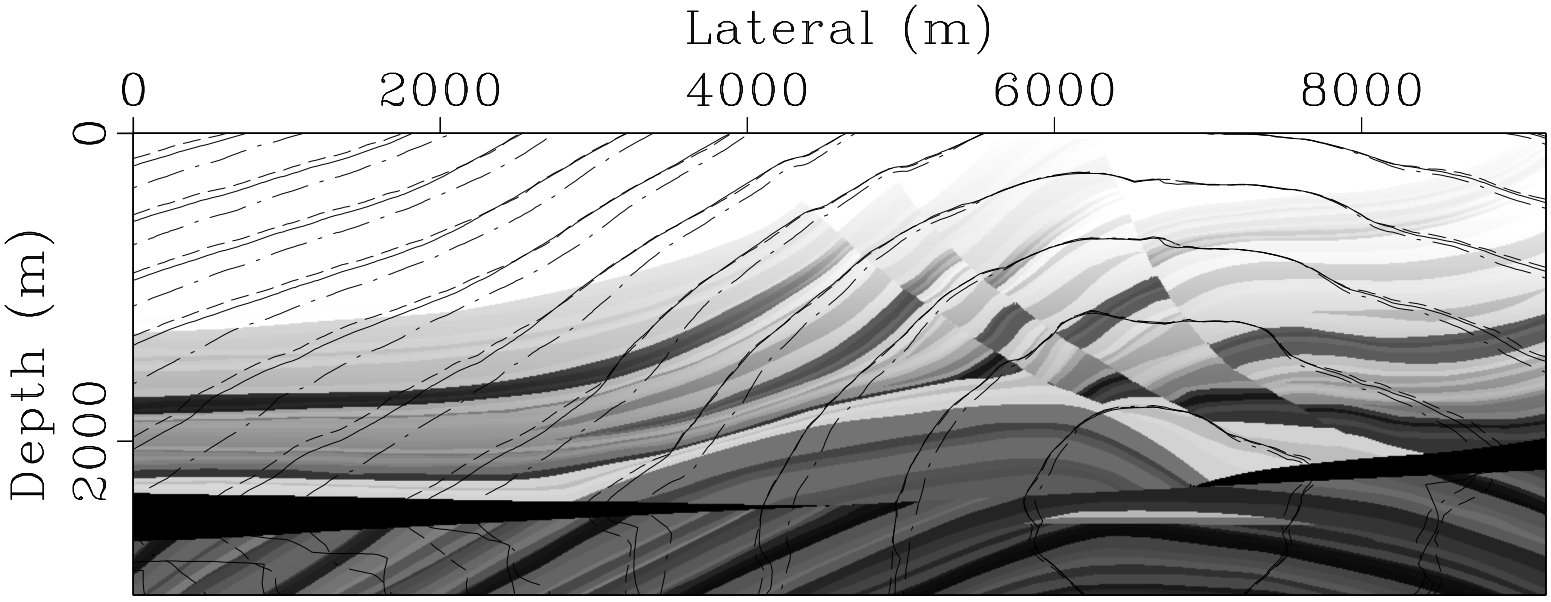
fastEta.c

```
void updateds (int p1, int p2, int p3, float* tj, f
               float s, float dy)
```

Complex examples

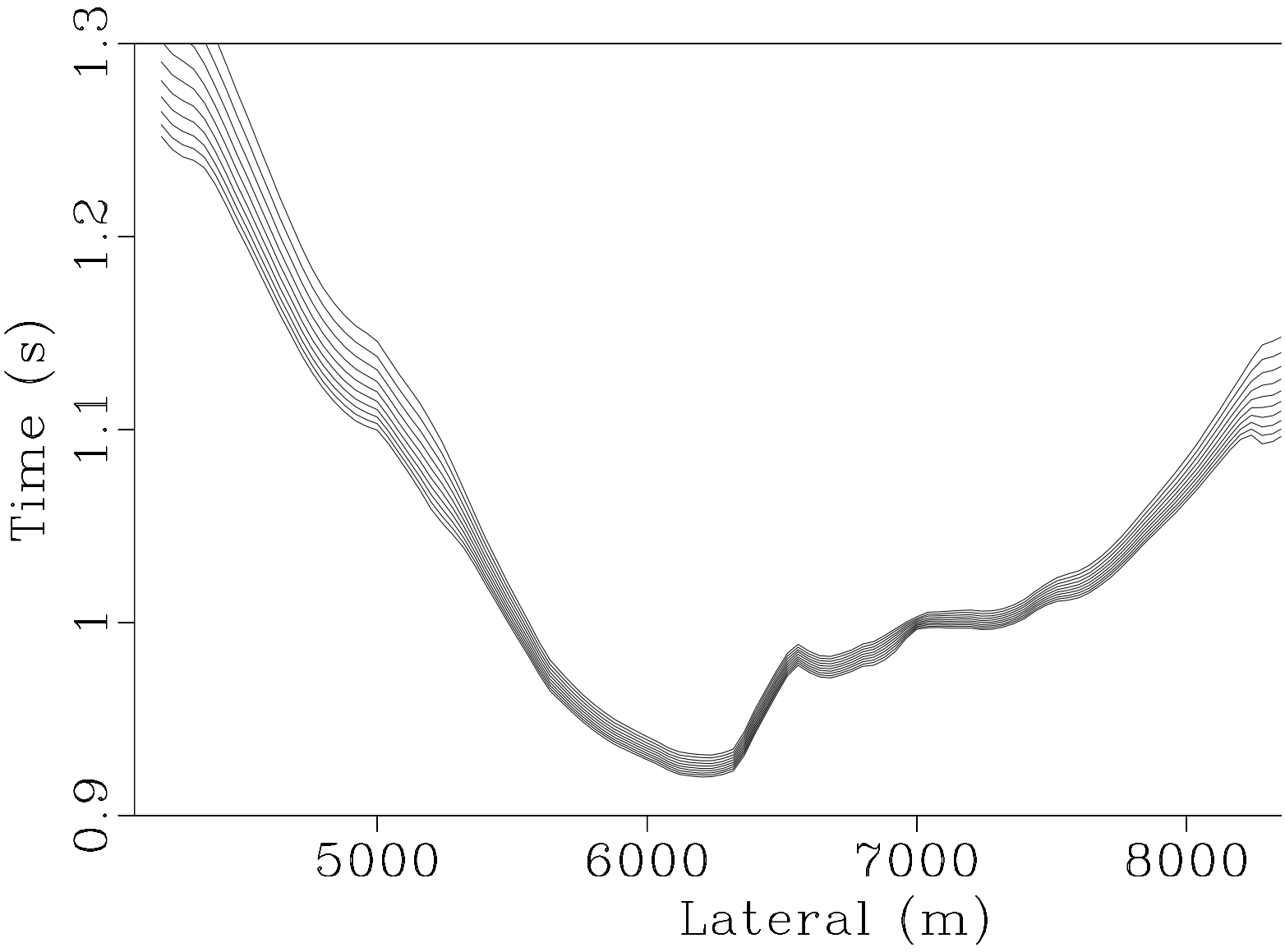
- Go to the Data directory under book
- Pick the example you need 2-D versus 3-D and the message you want to deliver
- copy that directory to your papers directoy
- In my case, the Marmousi example
- Alternatively, you could find the examples in some of the reproducible papers

Marmousi Result



$\eta = 0.1$

A scan for η



$\eta = 0.0 - 0.5$

Documentation

Madagascar Programming Reference Manual

courtesy of SWAG

RSFSRC/book/rsf/manual

Content

- Covers all available data types and subroutines under rsf.h
- Includes a description (beyond what was given by the developer)
- It organizes them in terms of usage and purpose
- we plan to have the manual get built automatically to include updates in the subroutine

Content

- *An example: Finite-Difference modeling*
- *Data types*
- *Preparing for input*
- *Operations with RSF files*
- *Error handling*
- *Linear operators*
- *Data analysis*
- *Filtering*
- *Solvers*
- *Interpolation*
- *Smoothing*
- *Ray tracing*
- *General tools*
- *Geometry*
- *Miscellaneous*
- *System*

Pointers

- Be patient, do not give up
- Use rsf-user@lists.sourceforge.net
- SConstruct controls everything, try, at least, to understand it
- You could always run RSF like SU

Final remarks

- Make things simple
- Get to know Madagascar
- Always compile (makes error search easier)
- Reproducible is good (very good)
- Love thy neighbor $\longrightarrow$ contribute

I thank the supporters of SWAG

http://swag.kaust.edu.sa

Madagascar Challenges

- A document for developers to encourage best practices
- This includes min. documentation and examples for system codes
- A steering committee???
- Accept sponsorship? to cover some improvement costs?

The Madagascar philosophy

- Democracy? Do we have really have that?
- Capitalist? It is free!!
- Socialist? Equal distribution regardless of work
- Helpful?