Command-line, plotting, papers and beyond

Pengliang Yang
Xi'an Jiaotong University
&
TCCS, UT Austin

My journey to Madagascar

How do you know Madagascar? When I was a 1st phd student of gephysics, I have many goals:

- Want nice 2D seismic display?
- Want a 3D volume plotting?
- Want to write a paper for 'geophysics'?
- Want to include figures in a paper?
- Want to do further research based on other researchers' efforts?
- Want to share results in public?
- ...

What was I doing? I was learning Seismic Unix hard, but cannot use it neatly!

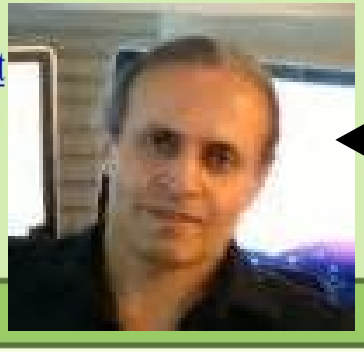
The Modeling and Imaging Lab

The WTOPI Research Consortium

International Symposium on

Geophysical Imaging with Localized Waves

Tianfuyuan Beach Spa Resort
Sanya, Hainan Island, China
July 24-28 2011



Congratulations to the success of Symposium!

Localized waves refer to waves localized in propagation direction-space location or/and in time-frequency. Localized waves exist both in real world as beam wave, wave packet (pulsed beam), soliton, coherent state, or in wavefield decomposition schemes as certain elementary waves having some preferred physic-mathematical properties. With phase-space localized waves, local AVA (amplitude variation with angle), local inversion for rock/reservoir parameters can be performed and the close link between imaging and inversion can be established. In fact, many different forms of localized waves,

Convened by:
Ru-Shan Wu, *University of California at Santa Cruz, USA*
Marteen de Hoop, *Purdue University, USA*
Bjorn Ursin, *Norwegian University of Science & Technology, Norway*
Jinghuai Gao, *Xi'an Jiaotong University, China*

International Organizing Committee:
Ru-Shan Wu, Maarten de Hoop, Bjorn Ursin, Jinghuai Gao, Chuck Mosher, Dave Yuen, Xiao-Bi Xie and Amy Kornberg

Sponsors:

IASPEI · SEG · EAGE European Association of Geoscientists & Engineers · National Science Foundation · CNPC BGP · NSFC National Natural Science Foundation of China · UC Santa Cruz · Xian Jiaotong University

Invited Speakers:

Tariq Alkhalifah, *King Abdullah University of Science and Technology, Saudi Arabia*
Ross Hill, *Chevron Corp, USA*
Stephane Jaffard, *Universite Paris, France*
Junru Jiao, *Petroleum Geo-Services, USA*


011年Mada ascar计算地球物理暑期学校
011 Madagascar Sc ol on Reproducible

Tariq bragged: With Madagascar, you can

- Do 2D seismic display in a nice way;
- Do a fantastic 3D volume plotting;
- Write a paper for 'geophysics‘ and many other prestigious journals;
- Include figures in a paper directly after running the codes;
- Obtain other researchers’ codes freely, conveniently
- Expose results more publically after publication
- …

Until I became a visiting phd student at UT Austin under the supervision of Prof. Sergey Fomel, I had to use Madagascar to finish my course work on 'Seismic Imaing'. I became a user, and then a developer!

I accepted all these features! That's not bragging!

Outline

- RSF data format
- **Play with command-line**
- **Plotting**
- **Create SConstruct for your computation**
- **Reproduce figures and papers**
- Public datasets and benchmark models
- Free open course
- Join the community: From user to developer

RSF data format

- RSF=regularly sampled data format :-) a story from Sergey

  Storing large-scale datasets in a text format may not be economical. RSF chooses the next best thing: it allows data values to be stored in a binary format but puts all data attributes in text files that can be read by humans and processed with universal text-processing utilities.

- *.rsf: header file contains information about origin, sampling interval, labels and legends of axes, data type (float, int), location of the actual data, size of data elements, etc.

- *.rsf@: binary data file

- Madagascar accesses the data according to header file and the path, using scons to build and run the codes.

  – SCons is a software construction tool (build tool, or make tool) implemented in Python, which uses Python scripts as “configuration files” for software builds. It is an easier, more reliable, and faster way to build software, solving a number of problems associated with other build tools, especially including the classic and ubiquitous make itself.

Play with command-line

- Similar to Seismic Unix (SU) using command-line,
  - compatible with shell scripts using pipes
  - Find a code you need (GUI):sfgui, sfbrower
- Almost all the commands have the prefix 'sf': sfprog
  - In Sconstruct, sf can be omitted.
  - Man page for command reference: just type the name
    - sfspike, sfbandpass
- The most frequently used commands (I believe):sfin, sfattr, sfpen

multiple programs, and do not want to save the intermediate files, then pipes will greatly reduce the number of files that you have to keep track of.

Command-line usage

Single program

```
[< in.rsf] sfprog [par1=] [par2=] [...]  [> out.rsf]
```

- Single input `< in.rsf`
- Single output `> out.rsf`
- Multiple parameters `par=val`

Multiple programs

```
[< in.rsf] sfprog1 [par=] | ...  | sfprogn [par=] [>
out.rsf]
```

- ONE task per program
- Data passed through pipes

Plotting

List of available plotting programs in Madagascar.	
sfbox	make box-line plots
sfcontour	make contour plots
sfcontour3	make contour plots of 3D surfaces
sfdots	plot signal with lollipops
sfgraph	create line plots, or scatter plots
sfgraph3	generate 3-D cube plots for surfaces.
sfgrey	create raster plots or 2D image plots
sfgrey3	create 3D image plots of panels (or slices) of a 3D cube
sfgrey4	generate movies of 3-D cube plots
sfplotrays	make plots of rays
sfthplot	make hidden-line surface plots
sfwiggle	plot data with wiggly traces

To actually create a plot, we can use the plotting programs on the com

```
sfspike n1=100 | sfnoise > file.rsf
sfgraph < file.rsf title="noise" > file.vpl
```

*.vpl: Figures are Vplot format

Shell scripting

Shell scripting is the first option for creating scripts. In shell scripti
scripting may be familiar to users of other packages such as SU,

```
#!/bin/bash
sfspike n1=100 > file.rsf
< file.rsf sfgraph > file.vpl
```

```
sfspike n1=1000 k1=300 > spike.rsf

sfin spike.rsf

sfattr < spike.rsf


sfbandpass fhi=2 phase=y < spike.rsf > filter.rsf


sfwiggle clip=0.02 title="Welcome to Madagascar" < filter.rsf > filter.vpl


sfpen < filter.vpl


sfspike n1=1000 k1=300 | sfbandpass fhi=2 phase=y | sfwiggle clip=0.02
title="Welcome to Madagascar" | sfpen
```

Interacting with files from the command line

Ultimately though, 95% of your time using Madagascar on the command line will be to inspect and view files tha

- **sfin** , used to get header information,
- **sfattr** , used to get file attributes,
- **sfwindow** , used to select portions of RSF files,
- and **sftransp** , used to reorder files.

Here are detailed usage examples and explanations of what the above programs do: **sfin** is one of the most use we have them in the correct order.

```
sfin file.rsf

file.rsf:
    in="/var/tmp/file.rsf@"
    esize=4 type=float form=native
    n1=100          d1=0.004        o1=0            label1="Time" unit1="s"
    n2=34           d2=0.1          o2=0            label2="Distance" unit2="km"
        3400 elements 13600 bytes
```

sfattr is also commonly used from the command line to check files for correct values. Most often, we use sfattr t reasonable. sfattr can be used to obtain basic statistics about the files as well.

```
sfattr < file.rsf

*****************************************
     rms =              1
    mean =              1
  2-norm =        58.3095
variance =              0
 std dev =              0
     max =              1 at 1 1
     min =              1 at 1 1
nonzero samples = 3400
  total samples = 3400
*****************************************
```

```
sfspike n1=100 | sfnoise > file.rsf
sfgraph < file.rsf title="noise" > file.vpl
```

Visualizing plots

In this example, we create a single trace full of noise and then send it to sfgr
visualize the data. To visualize the data we need to use a **pen** , which tells y
which will pick the best pen available for you. You can use **sfpen** to visualize

```
sfpen < file.vpl
```

This will pop up a screen on your window with the plot shown. Depending or
screen. Depending on the pen that you are using, there may be keyboard sh

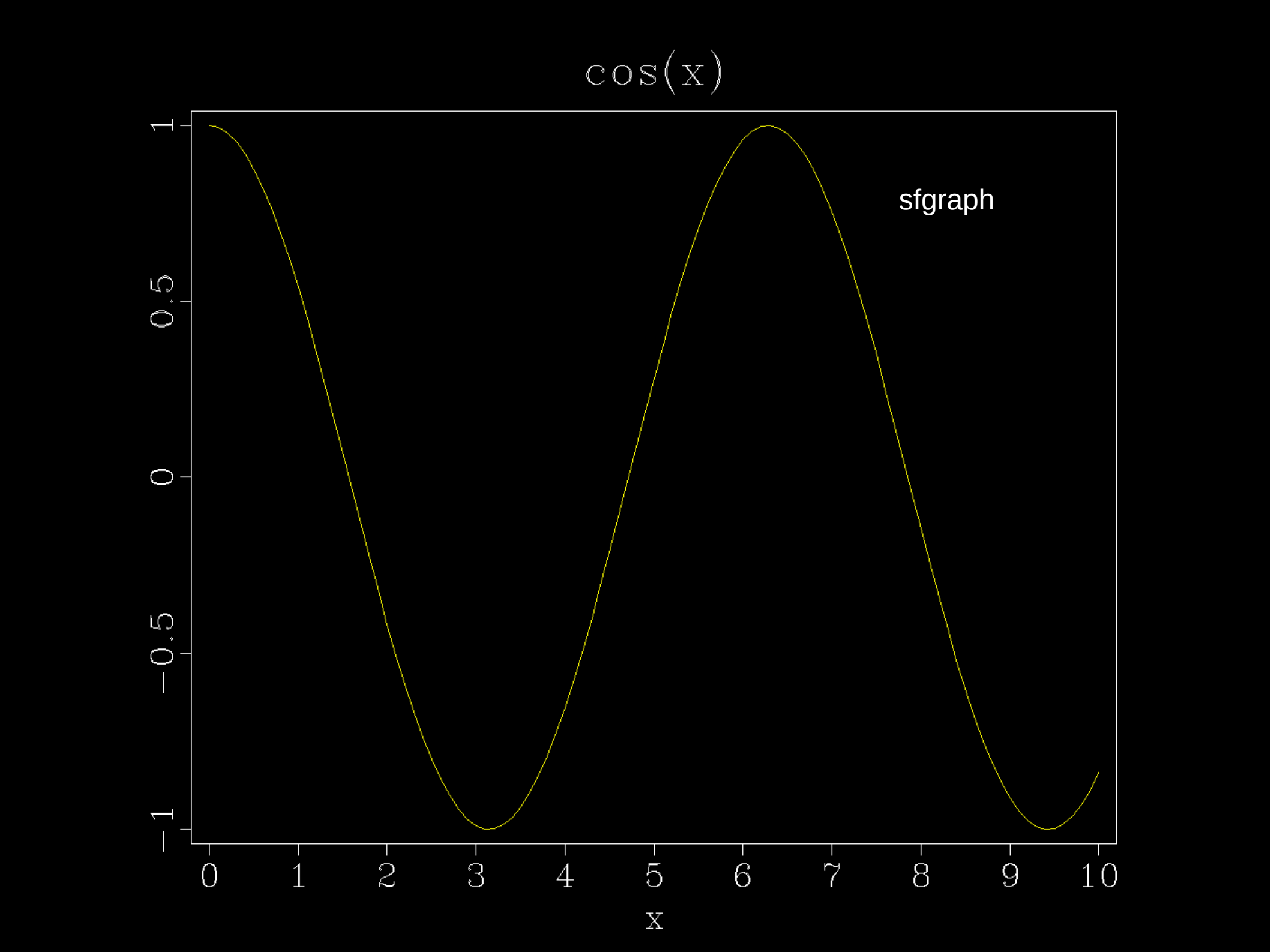
Converting VPLOT to other formats

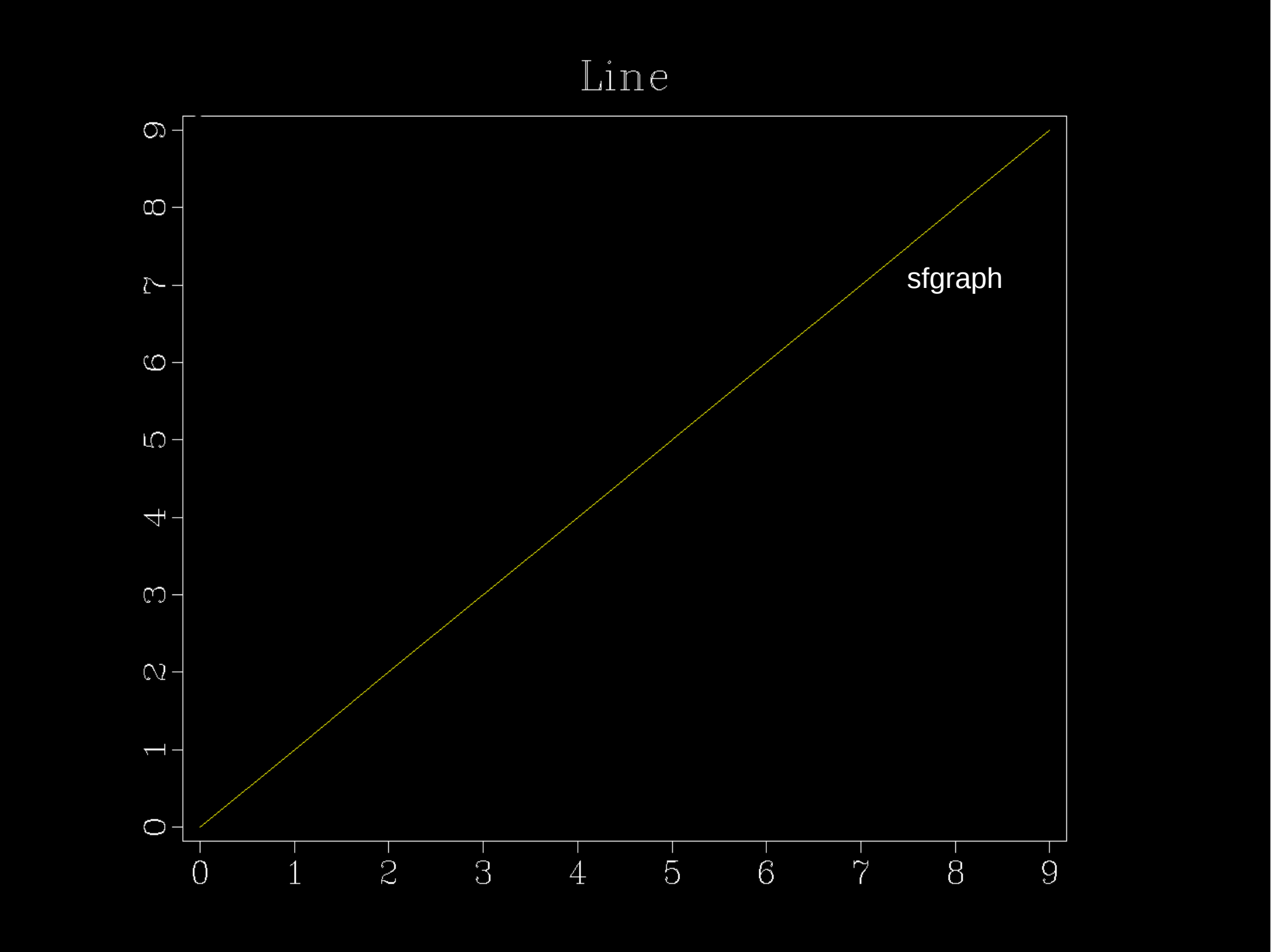
If you want to build reports or documents using other programs, or want to s
plot to another format use the tool **vpconvert** . **vpconvert** allows you to con

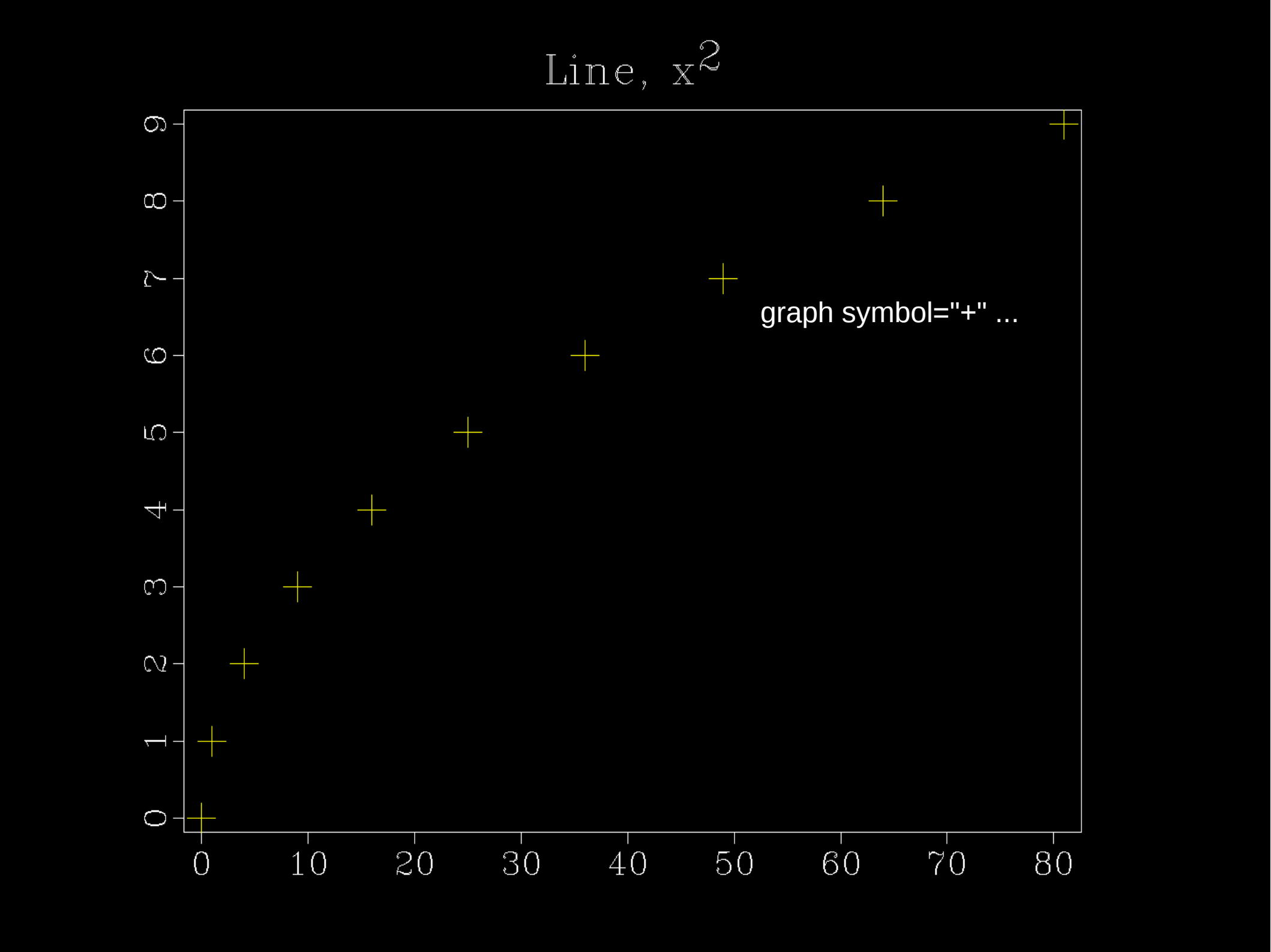
- avi,
- eps,
- gif,
- jpeg/jpg,
- mpeg/mpg (movie format),
- pdf,
- png,
- ppm,
- ps,
- svg,
- and tif.

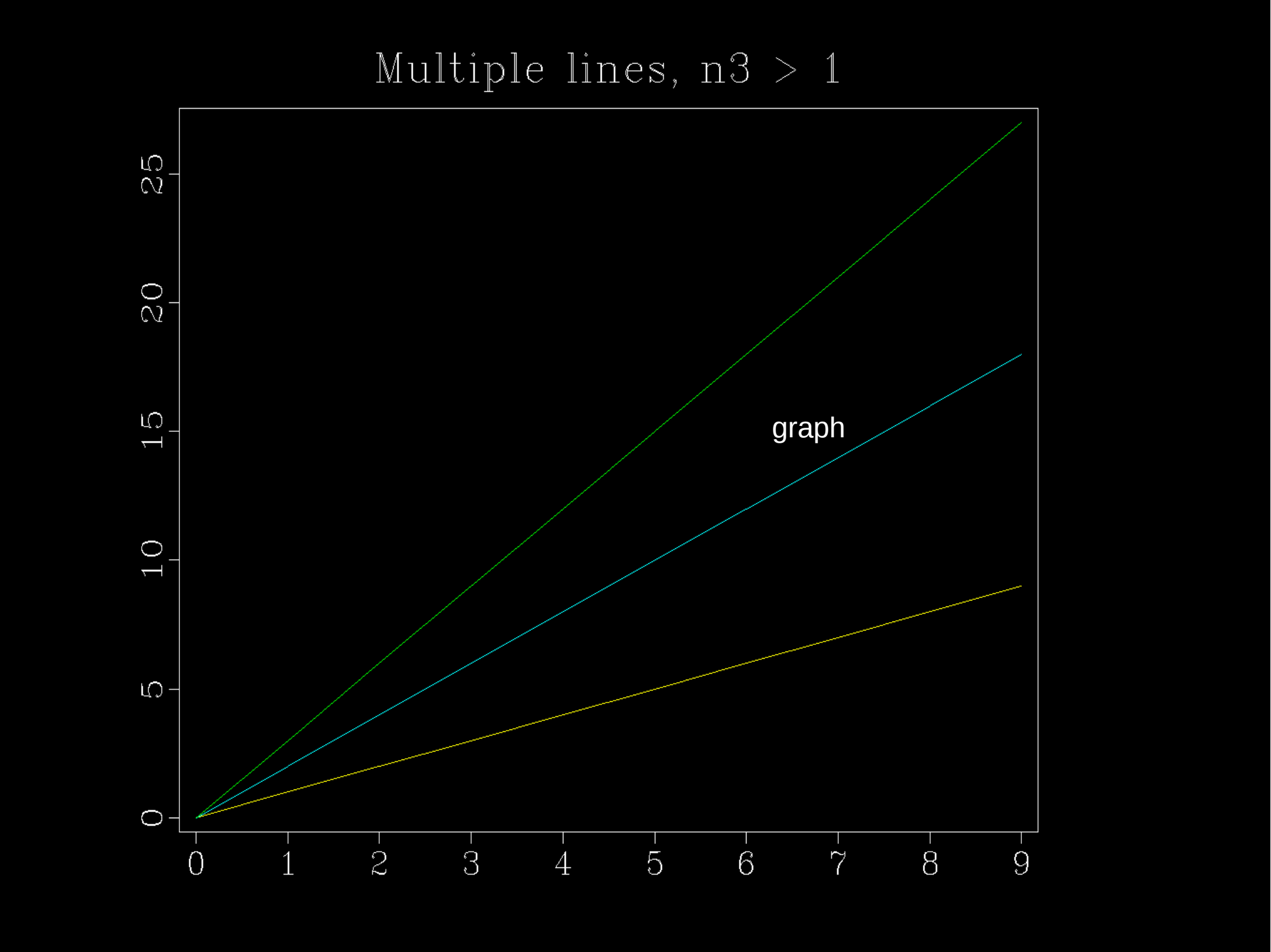
Here's an example of how to use **vpconvert** :

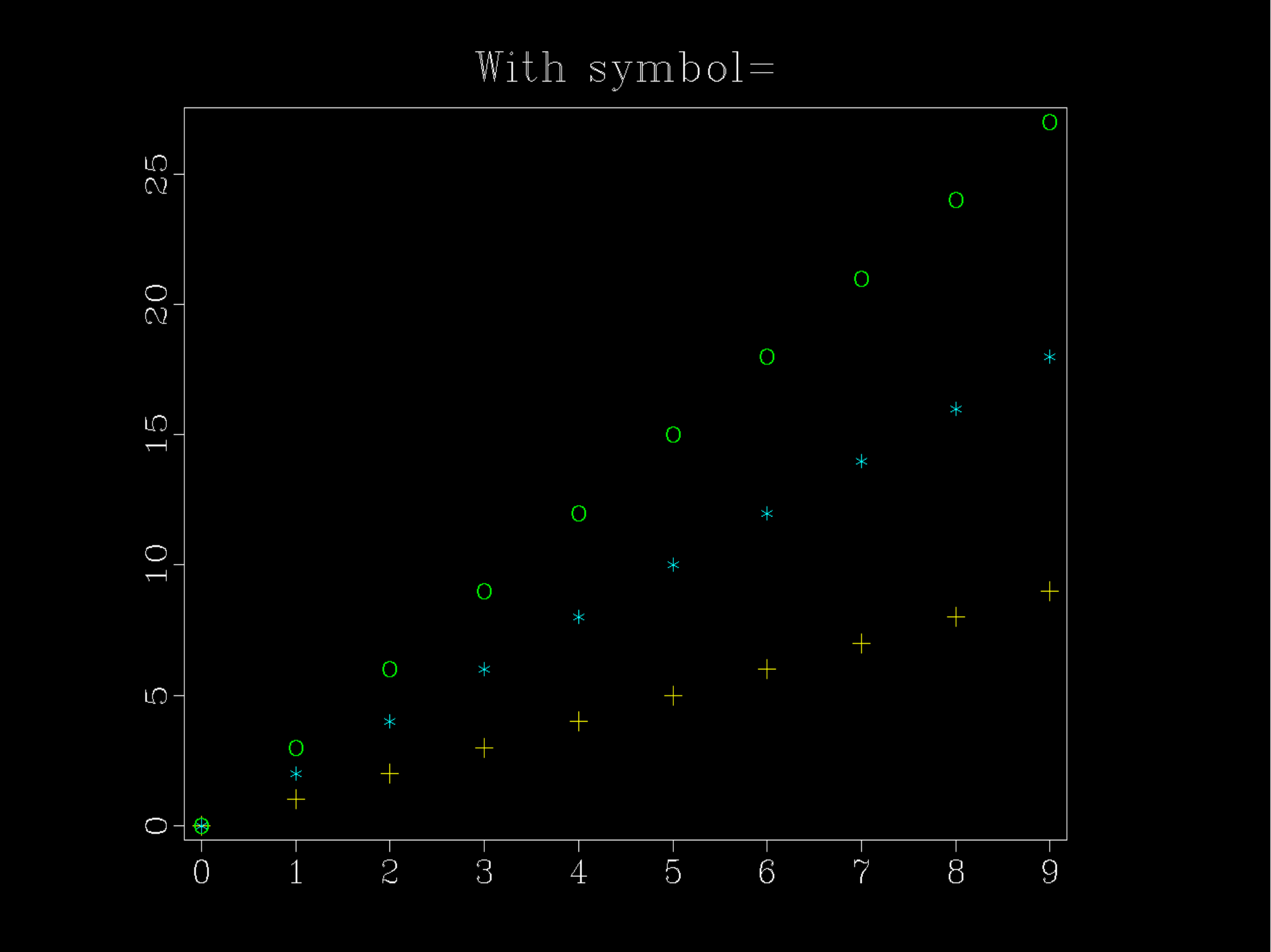
```
vpconvert file.vpl format=jpeg
```


cos(x)
sfgraph
1
0.5
0
−0.5
−1
0
1
2
3
4
5
6
7
8
9
10
x

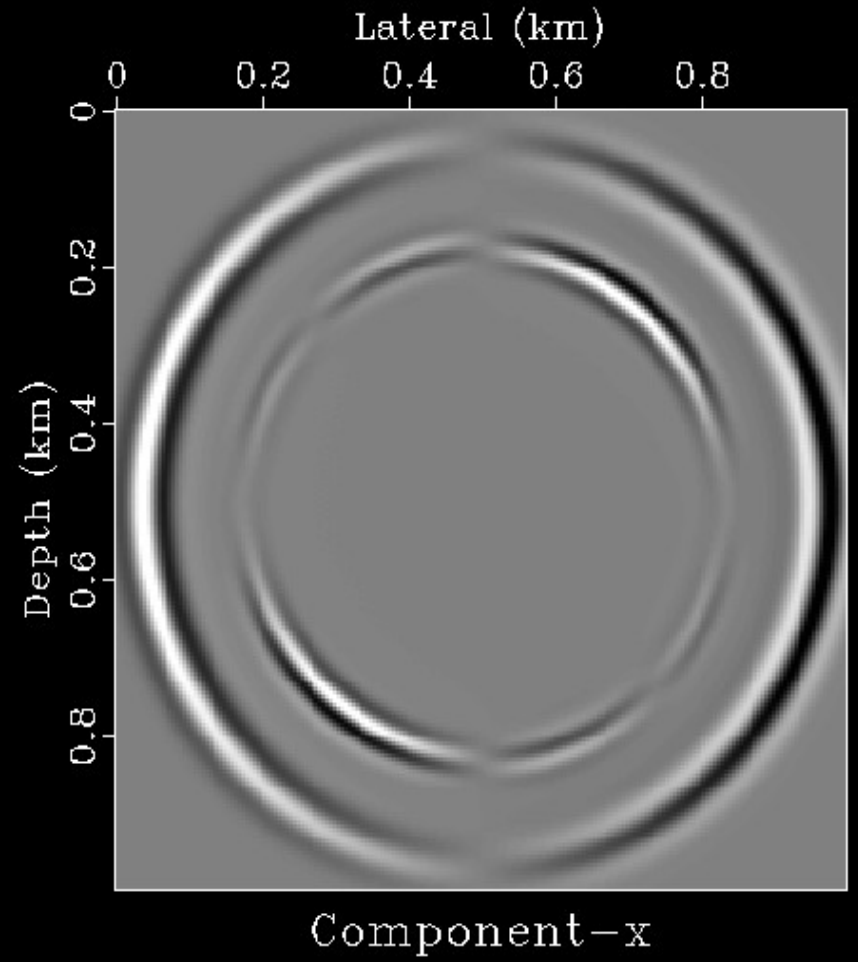

Line
sfgraph
0
1
2
3
4
5
6
7
8
9

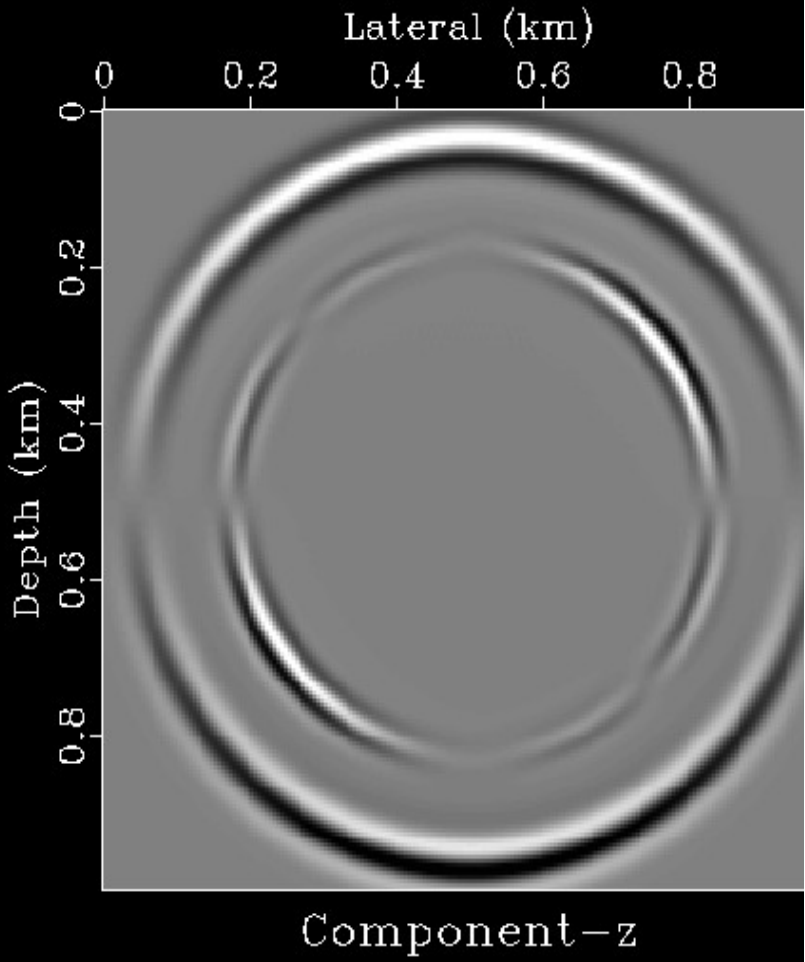

Line, x2
graph symbol="+" ...
0
1
2
3
4
5
6
7
8
9
0
10
20
30
40
50
60
70
80

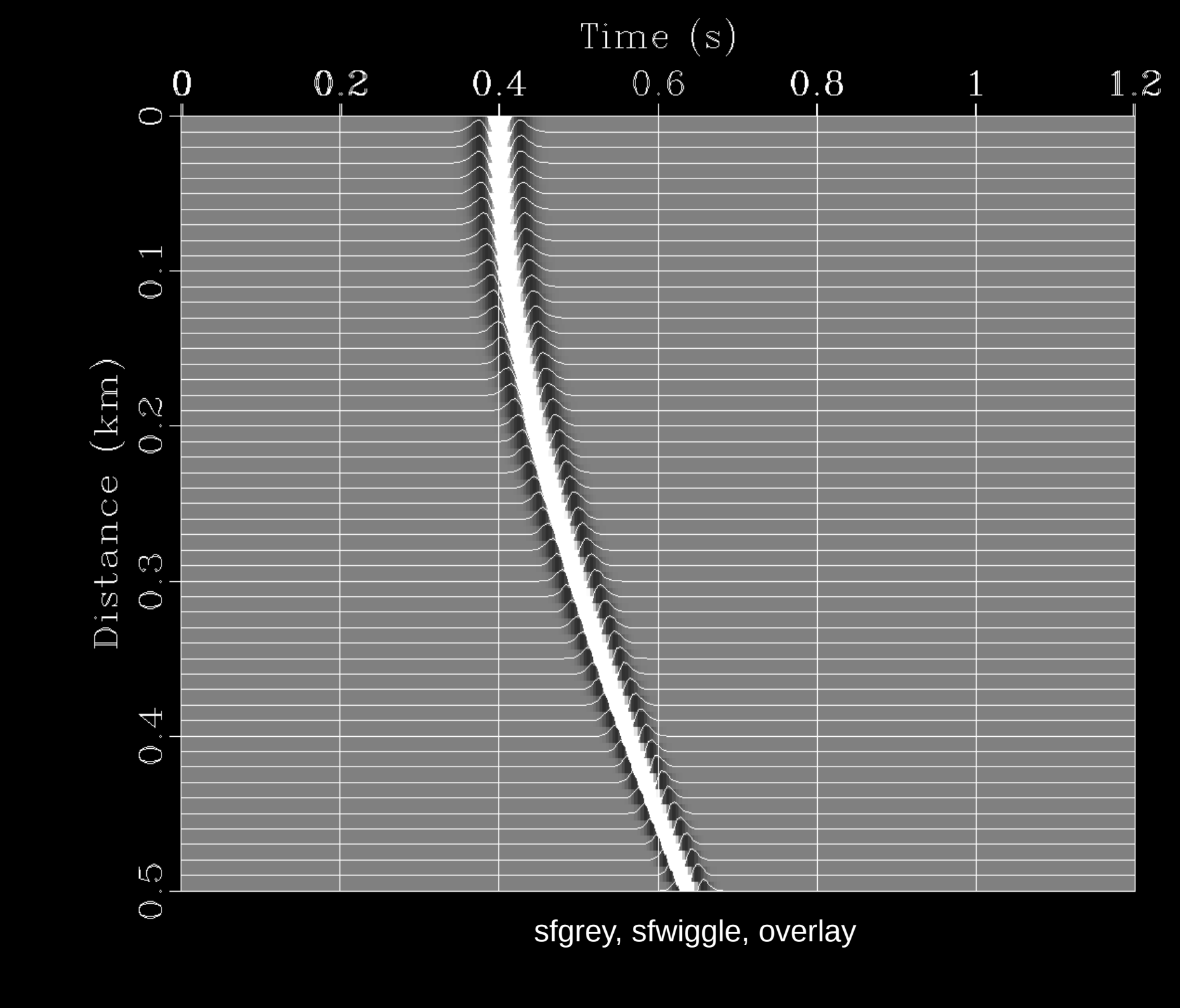

Multiple lines, n3 > 1
0
5
10
15
20
25
0
1
2
3
4
5
6
7
8
9
graph


With symbol=
0
5
10
15
20
25
0
1
2
3
4
5
6
7
8
9

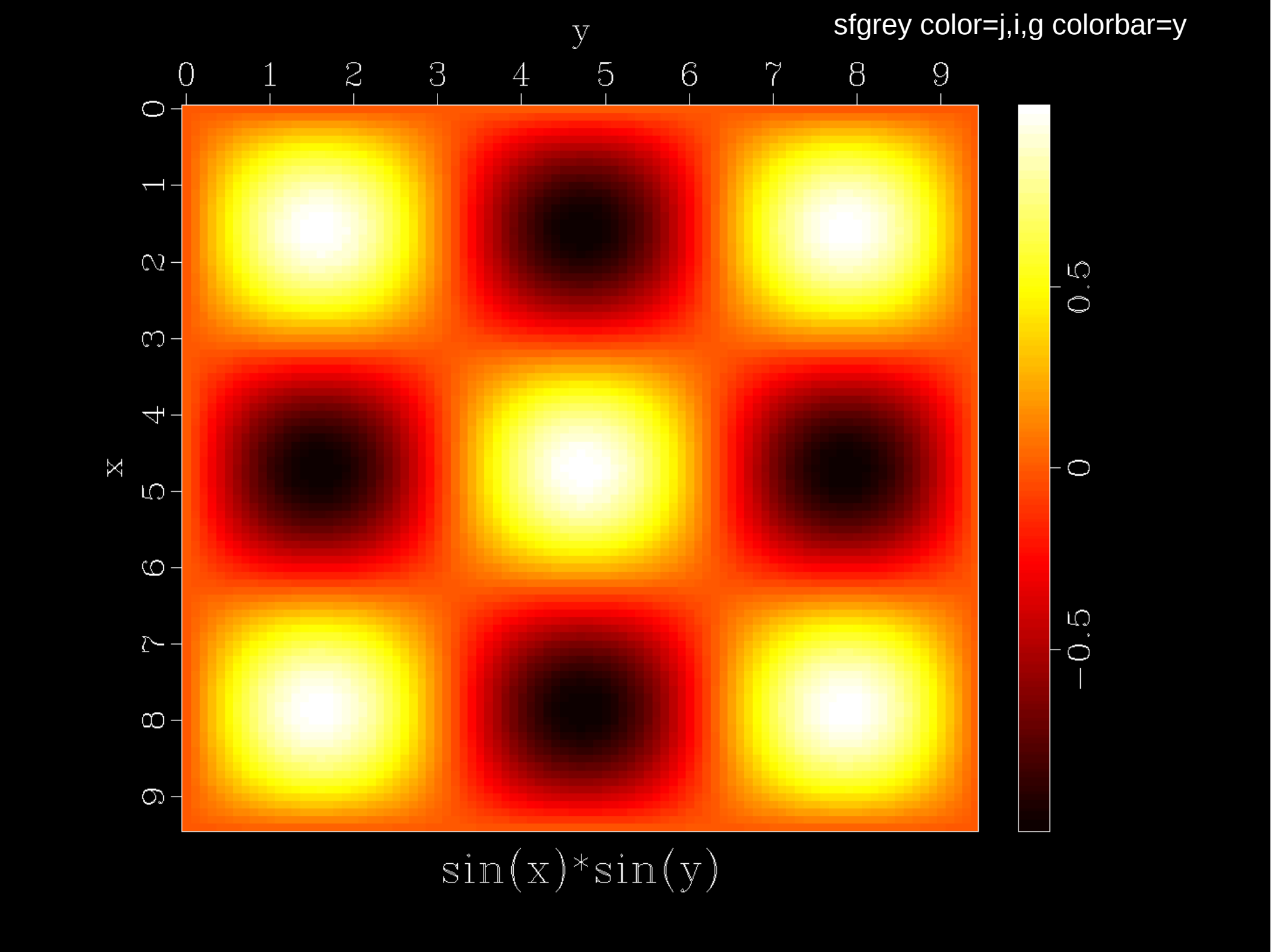
Plot(...,'sfgrey color=i ...');
Result(...,'SideBySideIso')

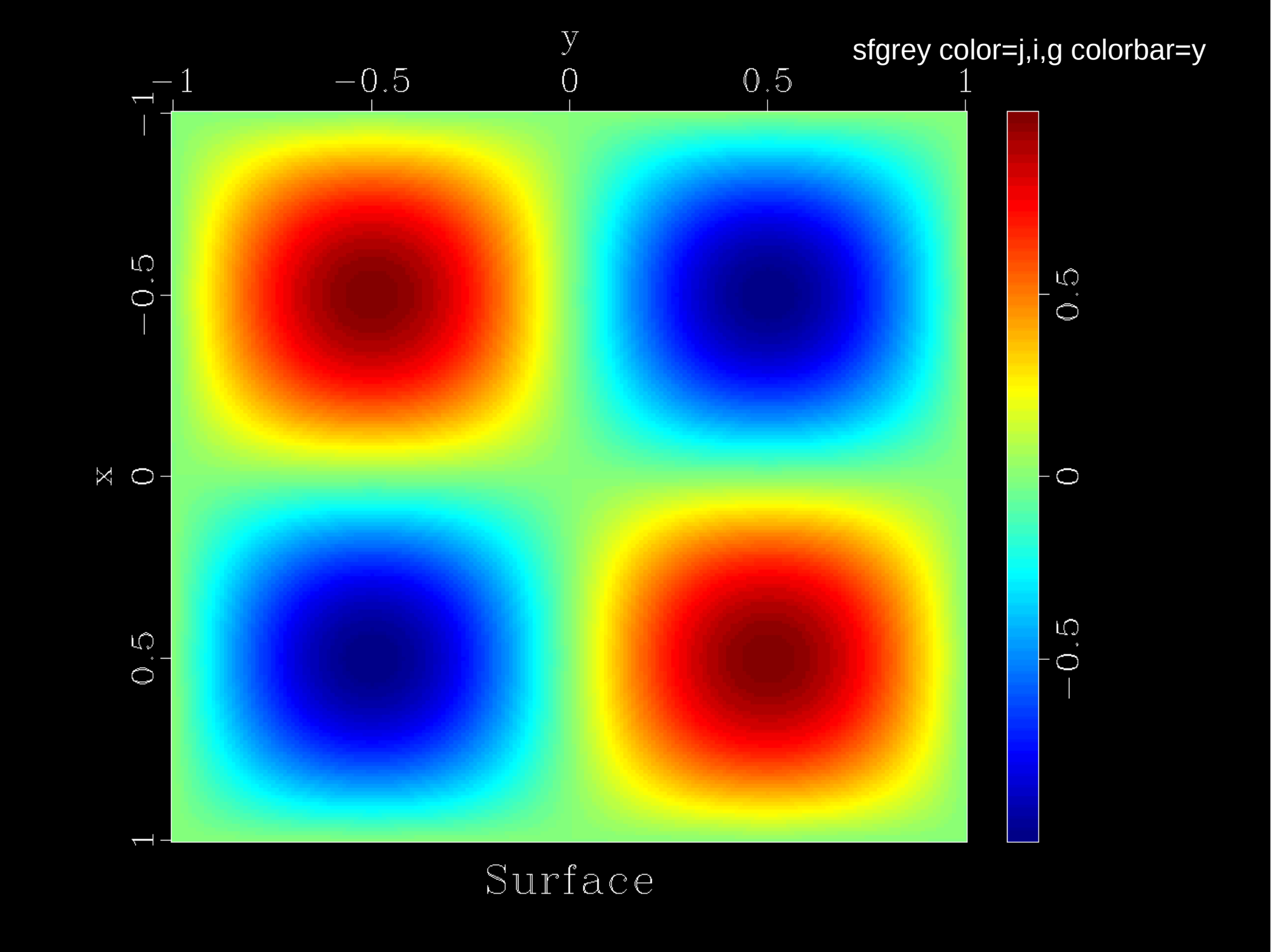


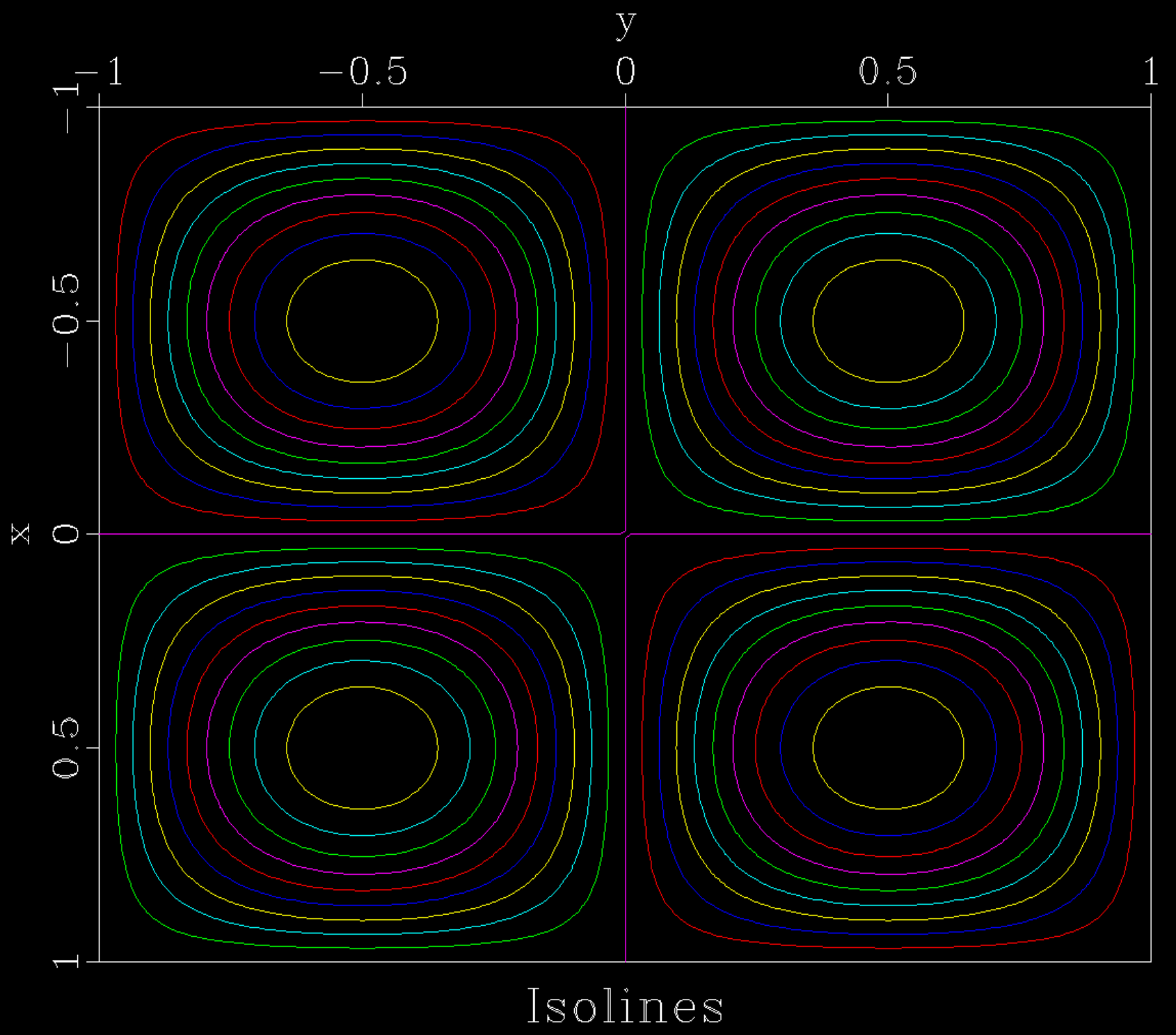




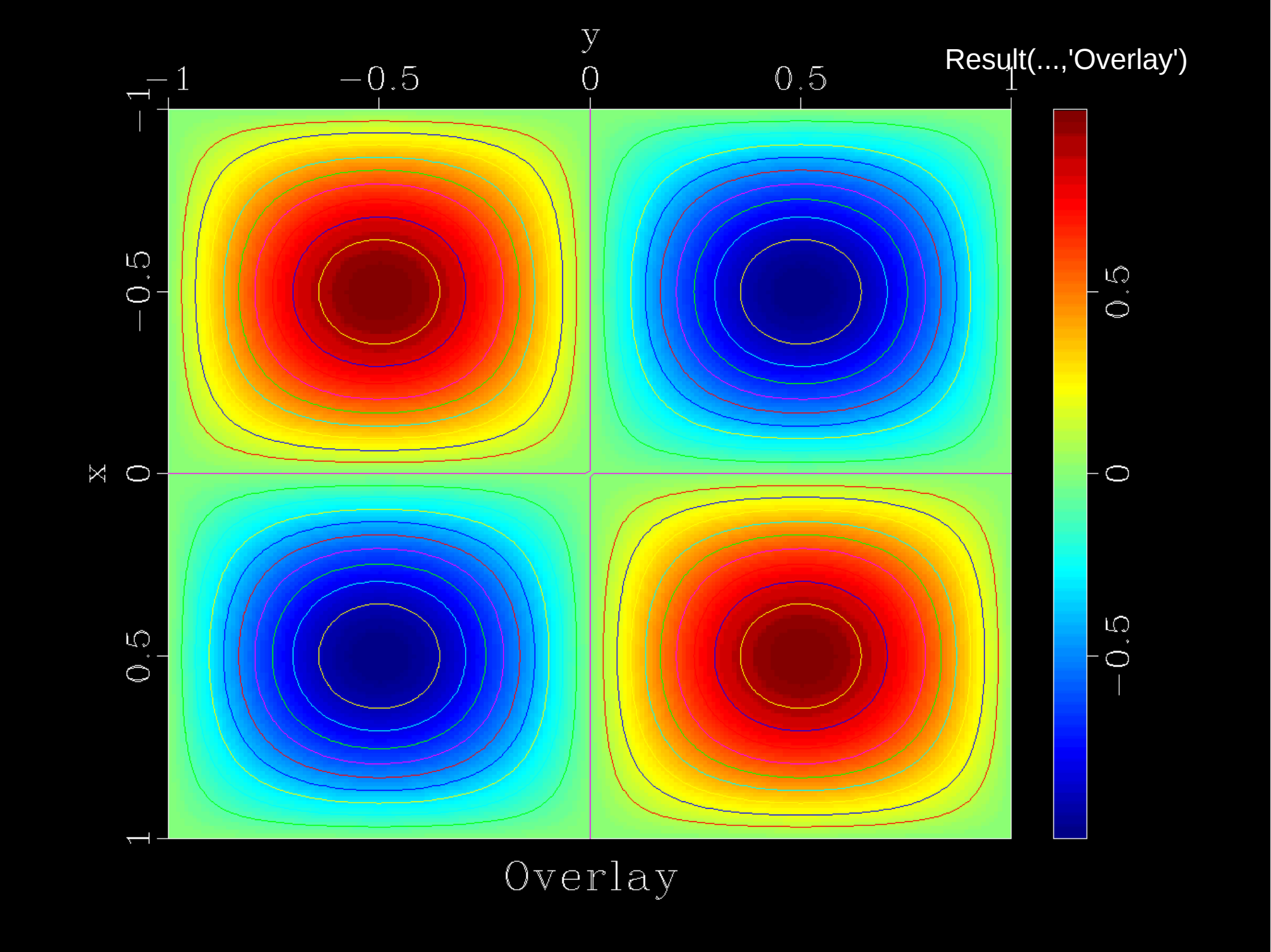
sfgrey, sfwiggle, overlay

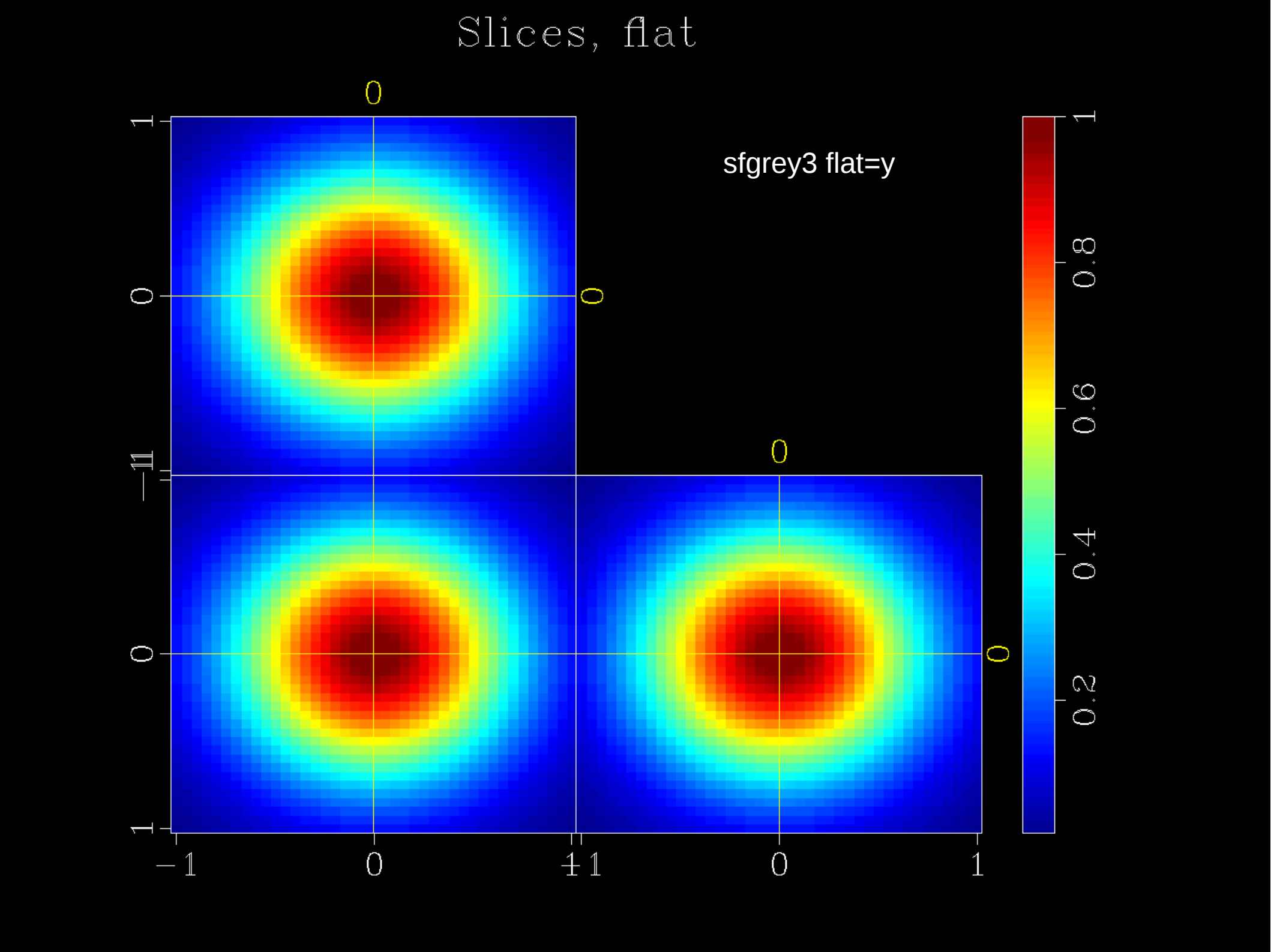

sfgrey color=j,i,g colorbar=y
y
0 1 2 3 4 5 6 7 8 9
x
0 1 2 3 4 5 6 7 8 9
0.5
0
−0.5
sin(x)*sin(y)

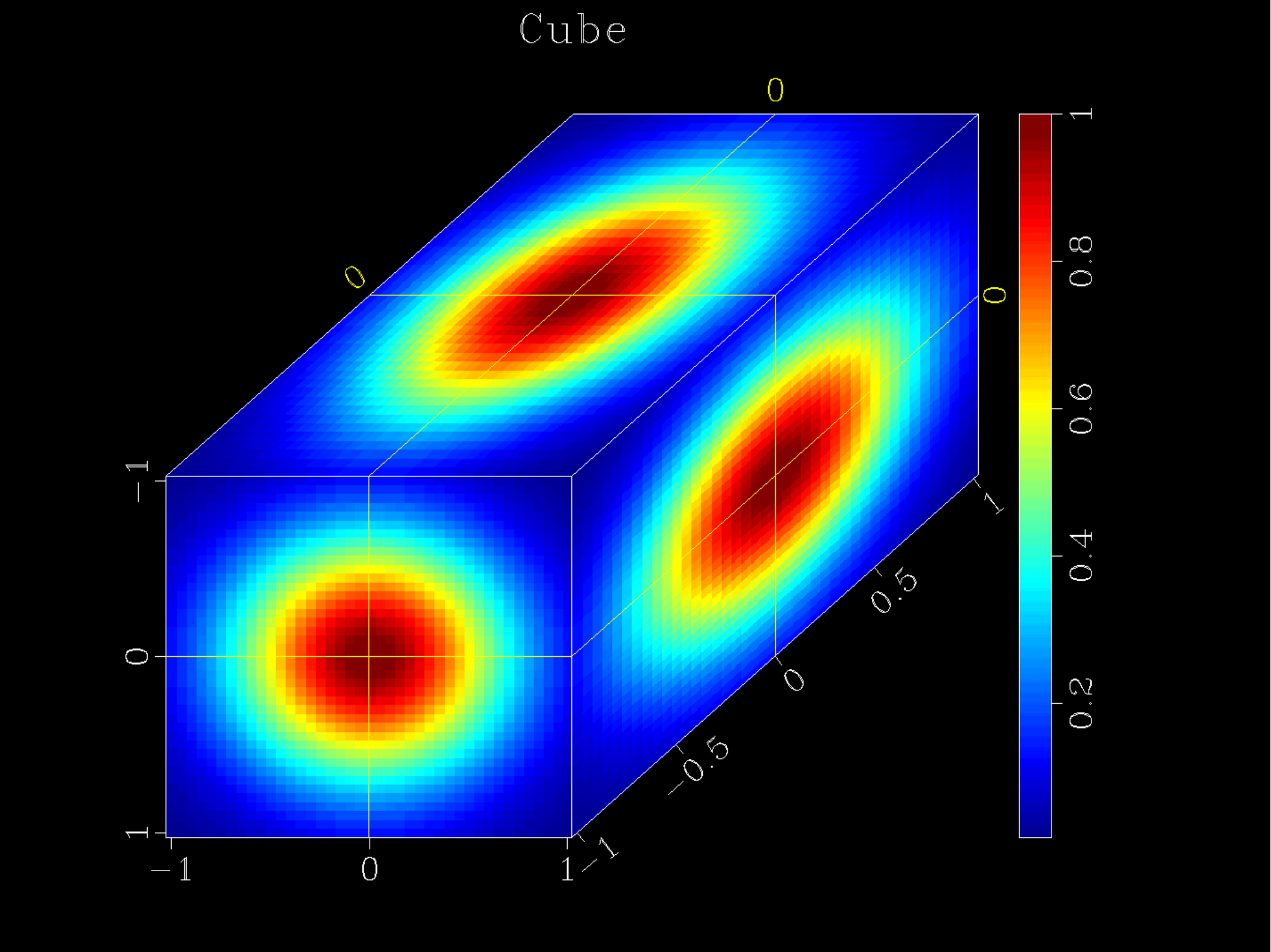

sfgrey color=j,i,g colorbar=y
y
−1
−0.5
0
0.5
1
x
−1
−0.5
0
0.5
1
0.5
0
−0.5
Surface



sfcontour


Result(...,'Overlay')
y
−1
−0.5
0
0.5
1
x
−1
−0.5
0
0.5
1
0.5
0
−0.5
Overlay


Slices, flat
sfgrey3 flat=y
0
0
0
0
1
0
−1
1
0
−1
−1
0
1
0.2
0.4
0.6
0.8
1


Cube
0
0
0
−1
0
1
−1
0
1
−1
−0.5
0
0.5
1
0.2
0.4
0.6
0.8
1

Create SConstruct for your computation

- Different programming tools
  - C/C++
  - Fortran (f77, f90)
  - Matlab
  - CUDA C/C++
  - Java
  - Python
  - ...

RSFSRC/api/
This directory contains development kits implementing the Madagascar API for several programming languages.

- High performance computing (do things in parallel)
  - Pscons
  - OpenMP (sfomp)
  - Mpi (sfmpi, sfbatch):
    - submit a job script on cluster (Stampede)
  - cuda

- RSFSRC/book/rsf/bash/omphello
- RSFSRC/book/rsf/bash/sample1
- RSFSRC/book/rsf/bash/sample2

Create SConstruct for your computation

- Simple python syntax:
  - from rsf.proj import *
  - Fetch() ← Defined in Madagascar's rsf.proj package (Sergey)
  - Flow()
  - Plot()
  - Result()
  - End()

Result(): creates Vplot figures in a subdirectory 'Fig'. These figures will be used to generate papers with latex.

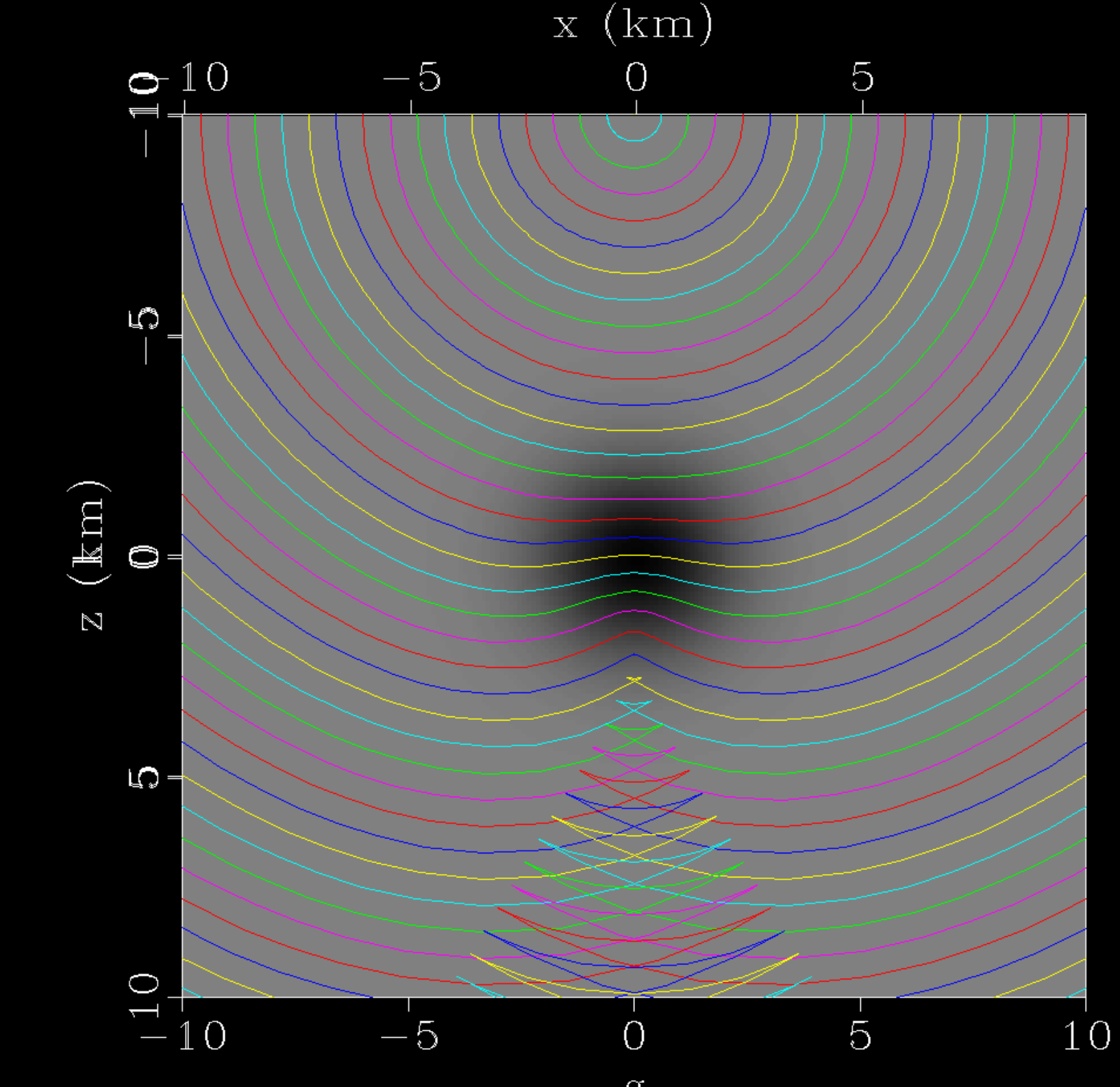
Plot(): similar to Result(), but the resulting figures may not be used in the paper.

Always start with **from rsf.prog import ***, end with **End()**

  - scons
  - scons view
  - scons -c
  - sfpen fig.vpl
  - vpconvert format=pdf fig.vpl

Create SConstruct for your computation

```
from rsf.proj import *

# create a model
amp=1
rad=1
Flow('model',None,
     '''
     math n1=301 d1=0.01 o1=0 n2=1001 d2=0.01 o2=0 output="1+2*x1+0.5*x2" |
     math output="input+%g*exp(-((x1-1.5)*(x1-1.5)+(x2-5)*(x2-5))/(%g*%g))" |
     put label1=Depth unit1=km label2=Position unit2=km
     ''' % (amp,rad,rad))

# plot the model
Plot('model','model','grey color=j scalebar=y label1=Depth unit1=km label2=Position unit2=km barlabel=Velocity
barunit=km/s barreverse=y title=Model allpos=y')

# do a ray-tracing
Flow('ray','model','rays2 yshot=5 nt=500 dt=0.001 a0=180 nr=1')

# plot the ray
Plot('ray','graph transp=y yreverse=y min1=0 max1=3 min2=0 max2=10 wantaxis=n wanttitle=n scalebar=y plotcol=7
plotfat=3')

# overlay model and ray
Result('overlay','model ray','Overlay')

End()
```


x (km)
−10
−5
0
5
z (km)
−10
−5
0
5
10

Reproduce figures and papers

- To run the examples:
  - **scons**
  - **scons view** (sfpen fig.vpl)
- To reproduce a paper:
  - Do the computation and generate the figures:
    - **sftour scons lock**
  - Generate the pdf file: **scons pdf**
    - **scons read**
  - Remove the unwanted things:
    - **scons -c**

Reproduce figures and papers

- Paper Sconstruct
  - from rsf.tex import * tells Madagascar to import Python packages for processing TeX files
  - End(name,lclass,options,use)
    - name - name of the root tex file to build, paper.tex.
    - lclass - name of the LaTeX class file to use.
    - options - document options for LaTeX class file.
    - use - names of LaTeX packages to import during compilation.

```
from rsf.tex import *

End('paper',lclass='eageabs',options='11pt',

use='times,natbib,color,amssymb,amsmath,amsbsy,graphicx,fancyhdr')
```

Exercises

- Example for visualization

  cd $RSFSRC/book/rsf/bash/plot

  scons

  scons view

  sfpen fig.vpl

- Exercise 1: adapt command-line to SConstruct

  – examine the script: example/SConstruct

Exercises

- Exercise 2: customize your paper

  cd $RSFSRC/book/bei/dwnc

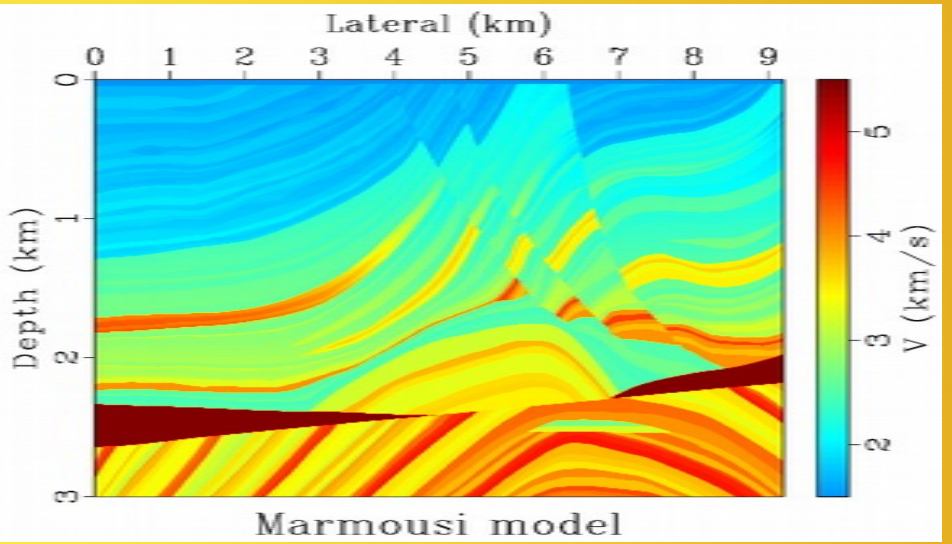
  sftour scons lock

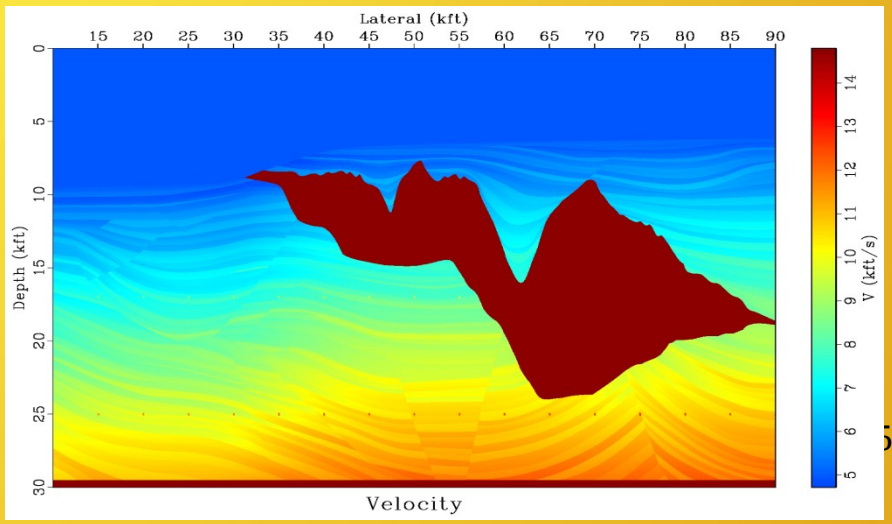
  scons read

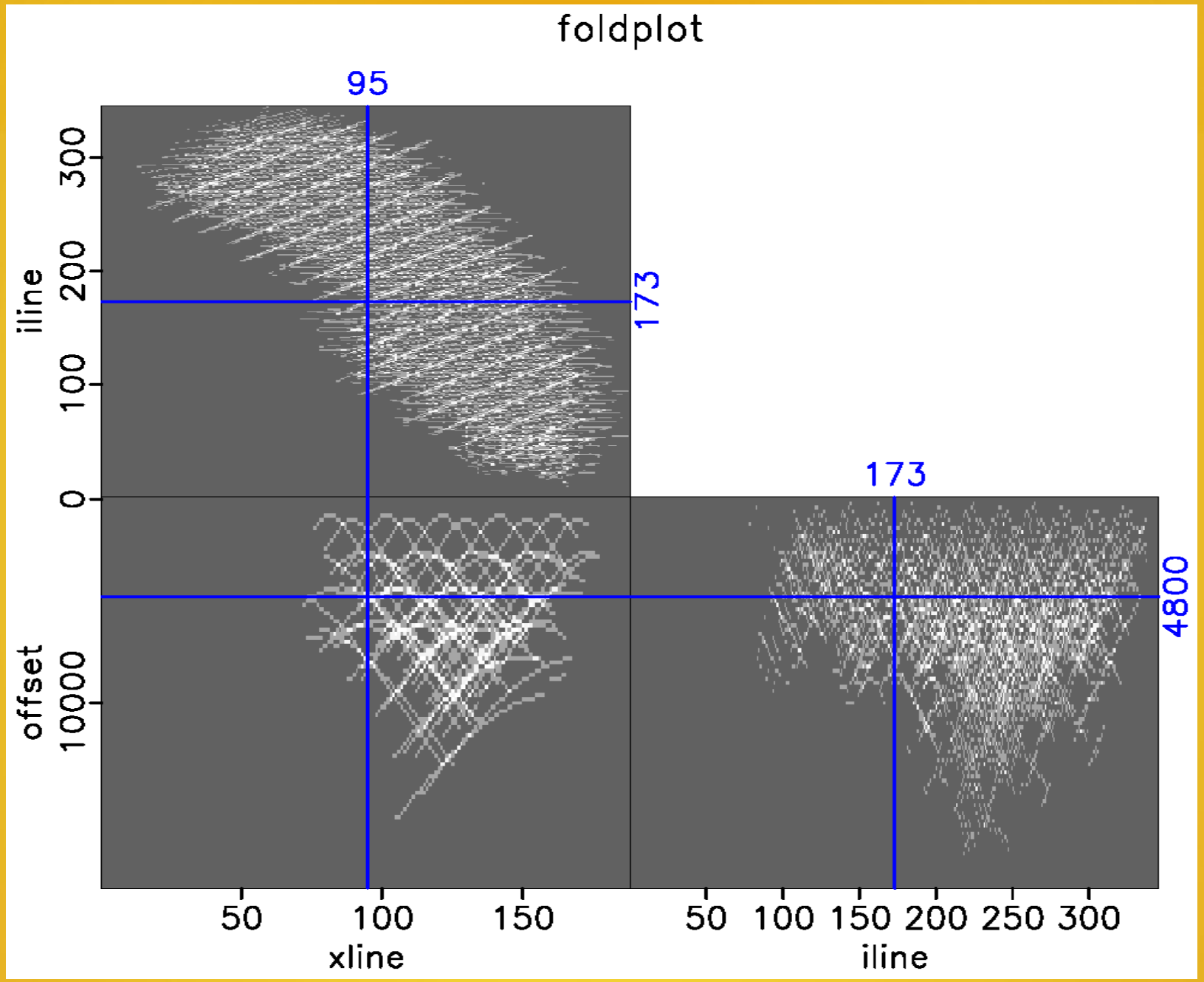
  sftour scons -c

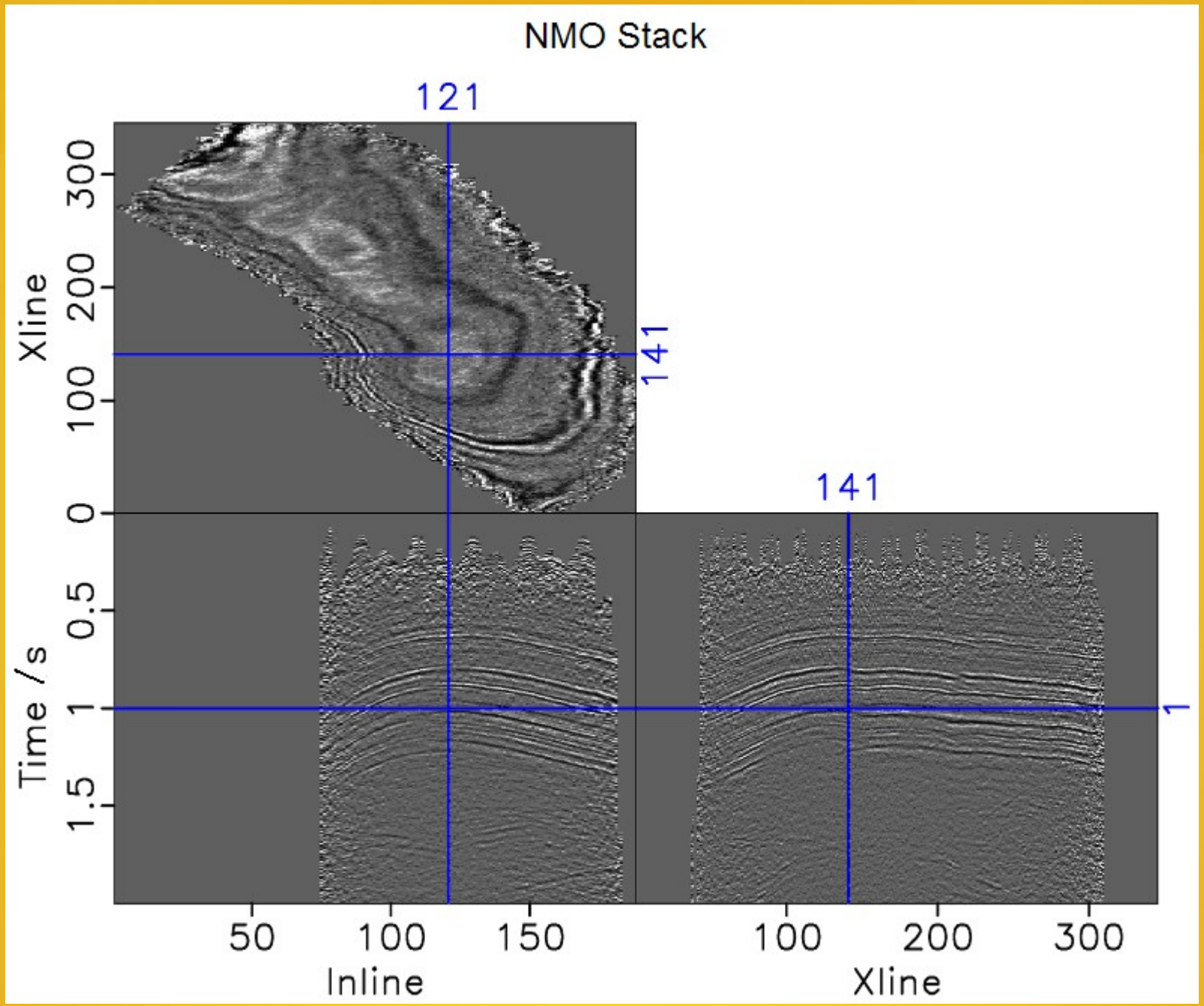
Public datasets and benchmark models

- RSFSRC/book/data (download from the server)
  - Marmousi
  - Sigsbee
  - BP
  - Hess VTI
  - Teapotdome
  - Pluto
  - Overthrust
  - Seam (new!!)
- Acknowledgement:
  - Karl Scheicher, Sergey Fomel






foldplot
95
173
173
4800
iline
0
100
200
300
offset
10000
50
100
150
xline
50
100
150
200
250
300
iline


NMO Stack
121
141
141
1
Xline
0
100
200
300
Time /s
0.5
1
1.5
50
100
150
Inline
100
200
300
Xline

Reproducible Documents

getting madagascar
- download
- Installation
- SVN repository
- SEGTeX

introduction
- Package overview
- Tutorial
- Hands-on tour
- Reproducible documents

user documentation
- List of programs
- Common programs
- The RSF file format
- Reproducibility with SCons

developer documentatio
- Adding programs
- Contributing programs
- API demo: clipping data
- API demo: explicit finite differences

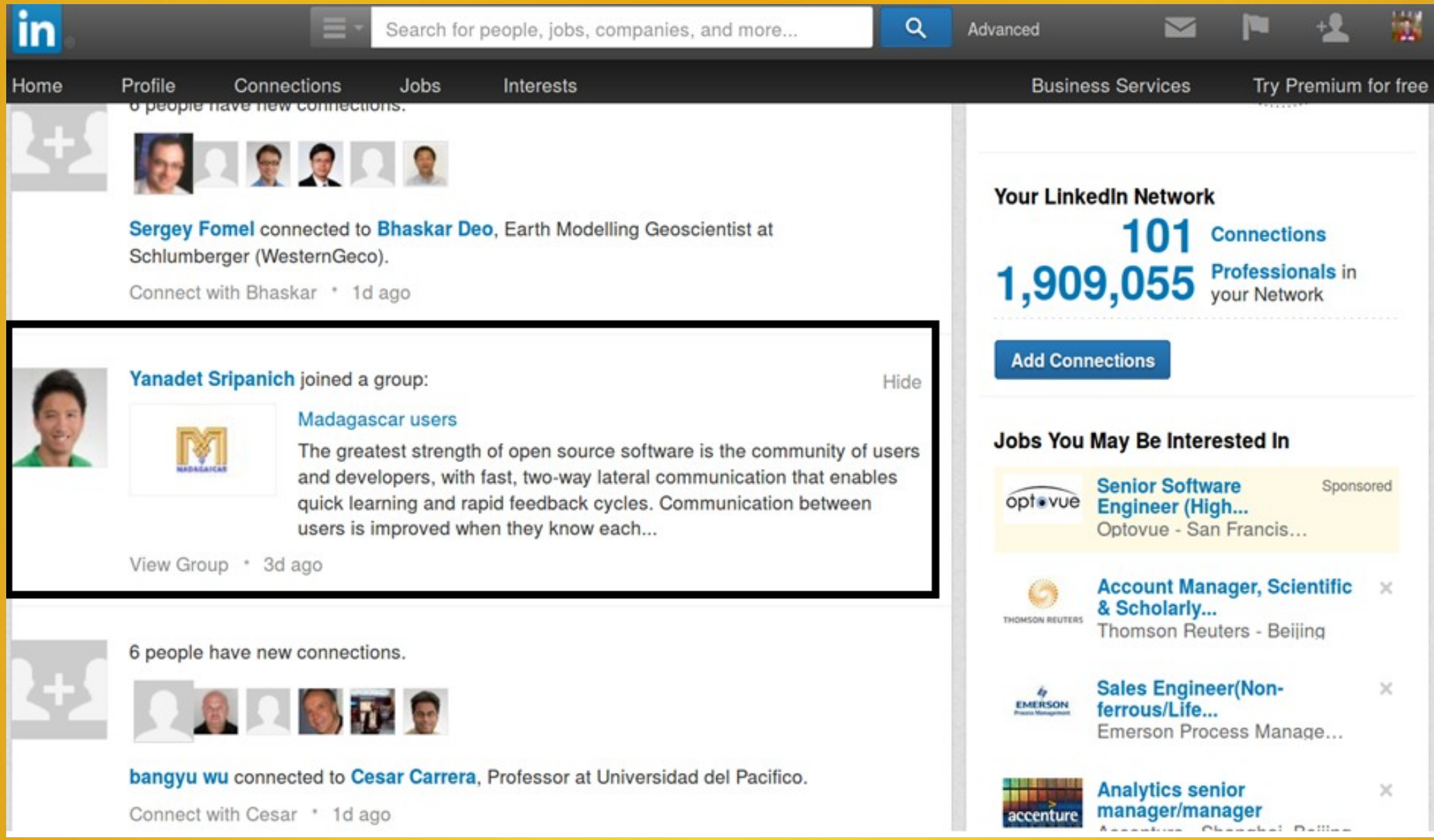
community
- User mailing list

Basic Earth Imaging

- Imaging in shot-geophone space by *Jon F. Claerbout*
- Downward continuation by *Jon F. Claerbout*
- Waves and Fourier sums by *Jon F. Claerbout*
- Zero-offset migration by *Jon F. Claerbout*
- Moveout, velocity, and stacking by *Jon F. Claerbout*
- Waves in strata by *Jon F. Claerbout*
- Adjoint operators by *Jon F. Claerbout*
- Field recording geometry by *Jon F. Claerbout*

Join the community: From user to developer

- From user to developer
  - Add and update the codes using 'svn', a revision control system:
    - svn add foo.c
    - svn commit -m “message about foo.c”
  - Attending the working workshop to add new features;
- Keep eyes close on the updates of Sergey's release;
  - Madagascar development blog
- Trouble shooting:
  - Join the ’Madagascar’ group on LinkedIn;
  - Ask questions;
- Madagascar is a community effort
  - Stand on the shoulder of giants;
  - Develop and share your research with others;
  - Ready to be exceeded and challenged!


Search for people, jobs, companies, and more...
Advanced
Home
Profile
Connections
Jobs
Interests
Business Services
Try Premium for free
Sergey Fomel connected to Bhaskar Deo, Earth Modelling Geoscientist at Schlumberger (WesternGeco).
Connect with Bhaskar • 1d ago
Yanadet Sripanich joined a group:
Hide
Madagascar users
The greatest strength of open source software is the community of users and developers, with fast, two-way lateral communication that enables quick learning and rapid feedback cycles. Communication between users is improved when they know each...
View Group • 3d ago
6 people have new connections.
bangyu wu connected to Cesar Carrera, Professor at Universidad del Pacifico.
Connect with Cesar • 1d ago
Your LinkedIn Network
101 Connections
1,909,055 Professionals in your Network
Add Connections
Jobs You May Be Interested In
Senior Software Engineer (High...
Sponsored
Optovue - San Francis...
Account Manager, Scientific & Scholarly...
Thomson Reuters - Beijing
Sales Engineer(Non-ferrous/Life...
Emerson Process Manage...
Analytics senior manager/manager

Sergey Fomel

Running Madagascar in the cloud

Sergey Fomel
Professor at The University of Texas at Austin

Madagascar in the cloud ahay.org
SageMathCloud is a free cloud computing platform for computational mathematics created by William Stein, the leader of the Sage project. SageMathCloud provides a rich environment, which allows one, for example, to easily install Madagascar and to...

Like (5) • Comment • Follow • 5 months ago

Mehdi E. Far, Julien Moreau and 3 others like this

Add a Comment...

Sergey Fomel

New paper: deblending using NMO

Sergey Fomel
Professor at The University of Texas at Austin

Deblending using NMO median filtering reproducibility.org
A new paper is added to the collection of reproducible documents: Deblending using normal moveout and median filtering in common-midpoint gathers The benefits of simultaneous source acquisition are compromised by the challenges of dealing with...

Like (2) • Comment • Follow • 1 month ago

Julien Moreau, Tobias Stål like this

Add a Comment...

Sergey Fomel

New paper: robust time-to-depth conversion


2014/ 8/ 2

Thank you!

Welcome to join us!