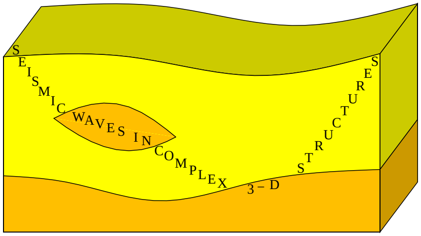
Interpolation within ray cells

Petr Bulant

Department of Geophysics, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic, http://sw3d.cz/staff/bulant.htm



http://sw3d.cz

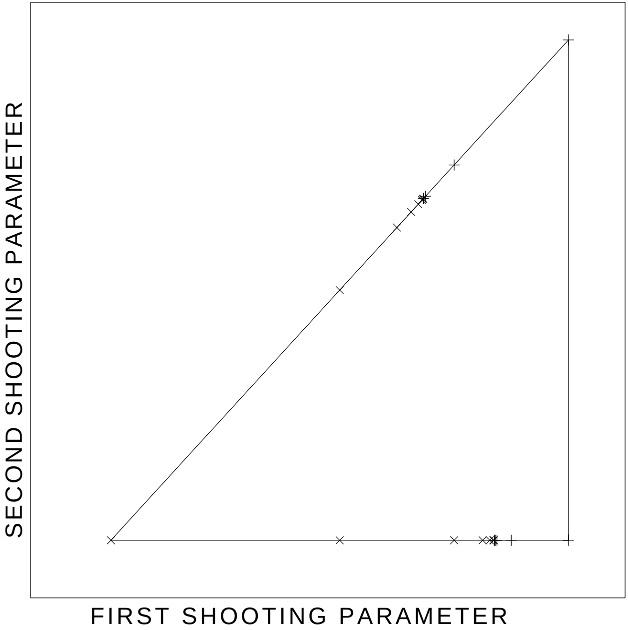
Controlled initial–value ray tracing

- *initial–value ray tracing* (Červený, Klimeš & Pšenčík, 1988) =>
ray history integer function (model blocks, interfaces, caustics, termination)

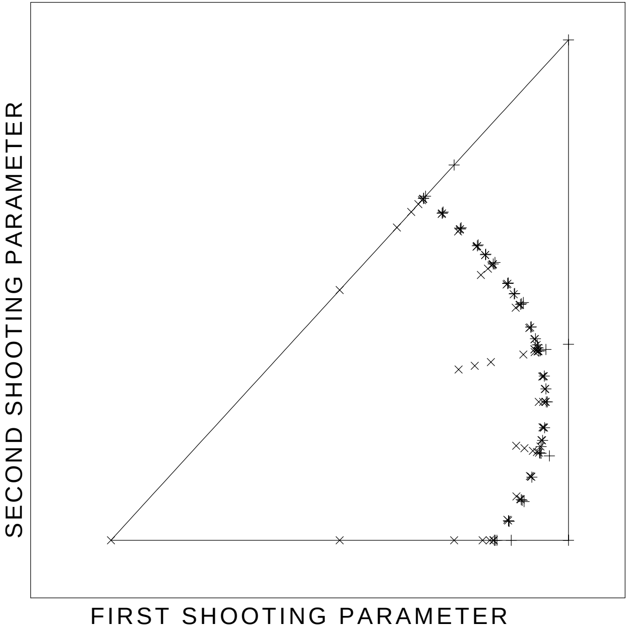
- continuity and smoothness of the travel time, its derivatives, spatial coordinates of rays, and many other quantities is guaranteed only between the rays of constant ray history

- *controlled initial–value ray tracing* (Bulant, 1999)
Domain of ray take–off parameters is covered by rays in order to identify regions of the same ray history. The emphasis is put on proper identification of the boundaries between the regions, and on keeping the demarcation belts as narrow as possible. The regions of the same ray history are then sampled by *homogeneous triangles*, which should not be too different from equilateral.

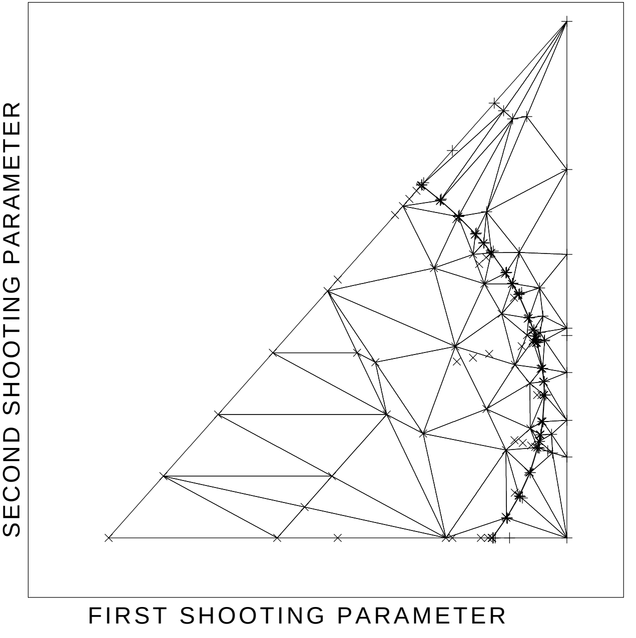
Coverage of the ray domain by homogeneous triangles



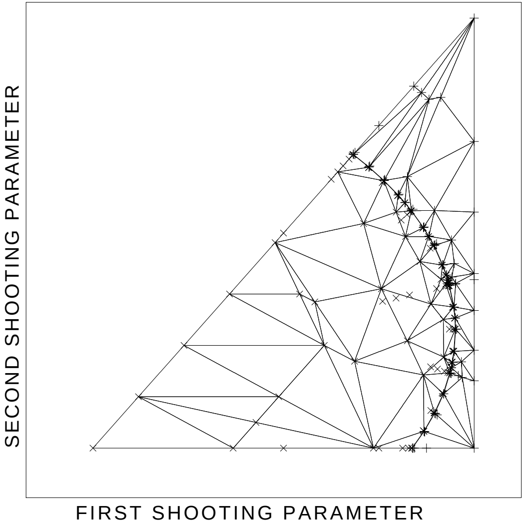
Coverage of the ray domain by homogeneous triangles



Coverage of the ray domain by homogeneous triangles

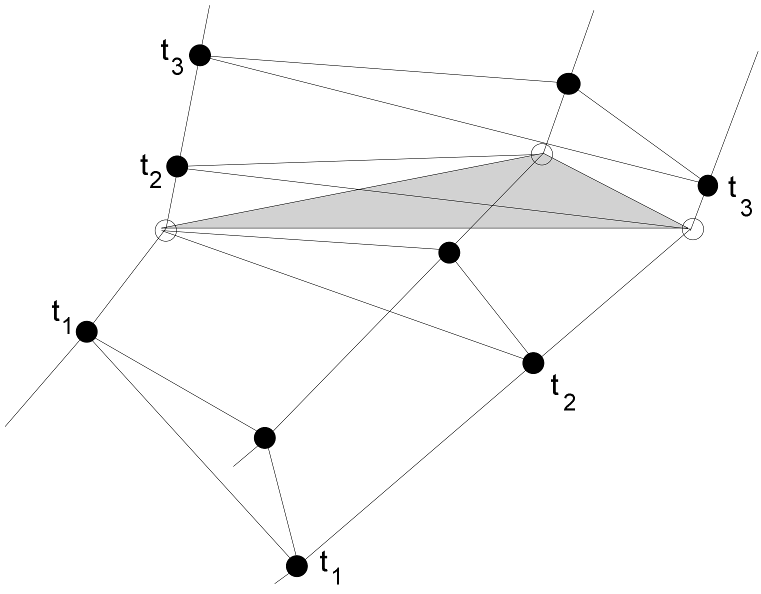


Applications of the controlled initial–value ray tracing



- boundary–value (two–point) ray tracing
- interpolation within ray cells

Interpolation within ray cells - decomposition of the ray tube into ray cells



Interpolation of the ray-theory Green tensor within ray cells

Interpolation of the ray-theory Green tensor within ray cells using the algorithm designed by Bulant & Klimeš (1999) has been proved especially efficient for calculating the ray-theory Green tensor at the nodes of dense 3-D grids.

The discretized ray-theory Green tensor can be used for various applications including the ray-based Born approximation, non-linear determination of seismic hypocentres, studies of seismic sources, or Kirchhoff prestack depth migration.

Interpolation within ray cells - numerical realization
Models for ray tracing:

- package MODEL

- general 3-D layered and block isotropic or anisotropic structures, containing isolated bodies, pinchouts, etc.

- structural interfaces composed of one or several smooth surfaces defined implicitly as the zero isosurfaces of given functions

- elastic parameters discontinuous only along structural interfaces, inside the layers and blocks elastic parameters vary smoothly in all three dimensions

Interpolation within ray cells - numerical realization
Step 1 – controlled initial–value ray tracing:

- package CRT, program `crt.for`

- P and S waves in general 3-D layered and block isotropic structures

- P-wave ray tracing in anisotropic structures without interfaces

- common S-wave rays in anisotropic structures without interfaces

- SH and SV reference rays in anisotropic structures with considerable transversely isotropic component (without interfaces)

The quantities calculated along rays are stored in the output files with given time step, and the quantities at structural interfaces and at the end surface are stored as well. Also the file with the indices of the rays corresponding to the vertices of homogeneous triangles (i.e. ray tubes) is stored. The quantities stored in the points along rays include coordinates, ray history, travel time and its derivatives, ray propagator matrix, and amplitudes of the Green function.

Interpolation within ray cells - numerical realization
Step 2 – interpolation within ray cells:

- package CRT, program `mtt.for`

- quantities may be interpolated to the gridpoints of arbitrary regular rectangular 3-D grid of points, or to the individual points specified in an input file

- interpolated quantities written in the form of multivalued grids

Interpolation within ray cells - interpolated quantities

- number of arrivals at each point
- real and imaginary travel times of all determined arrivals
- ray history
- slowness vector
- second derivatives of travel time with respect to the coordinates of the point
- components of the 4×4 ray propagator matrix in ray–centred coordinates
- sum of squares of Gaussian beam widths (for Gaussian beam migrations)
- real or imaginary part of the vectorial amplitude of the Green function, norm of the vectorial amplitude of the Green function, and amplitude modified for use in the Kirchoff integral
- coordinates of the initial point of the ray corresponding to the arrival (suitable mostly for linear or planar sources)
- derivatives of travel time with respect to the coordinates of the source (i.e. the vector opposite to the slowness vector in the source)
- user–defined quantities

Further processing of the interpolated quantities

- program `grdborn.for` to calculate the Born approximation of the wavefield at specified receivers
- program `mgrd.for` to convert a multivalued grid into several singlevalued grids
- program `mttgrd.for` to convert a multivalued grid into several singlevalued grids sorted according to the ray history
- program `grdcal.for` to perform vectorial calculations with singlevalued gridded data enables to perform all the basic operations like addition, subtraction, multiplication, division, exponentiation, logarithms, goniometric functions, and many other operations
- program `grdnorm.for` to calculate the spatial density of the Lebesgue norm of gridded data, mostly used for calculation of maximum or average value of the gridded data
- program `grdps.for` to display gridded data in PostScript
- program `grdmigr.for` to perform common-source Kirchhoff migration using gridded travel times and amplitudes

Numerical examples - package DATA (1/4)

`len/leni-mtt.h`
3-D model, 2-D target grid,
calculation of travel times,
pictures of travel times sorted according to their value
`hes/hes-mtt.h`
2-D model, 2-D target grid,
calculation of travel times,
pictures of rays together with the numbers of arrivals, pictures of travel times sorted according to their value or according to the ray history
`mar/mar-crt.h`
2-D model, 2-D target grid,
calculation of numbers of arrivals and the widths of Gaussian beams

Numerical examples - package DATA (2/4)

`mar/mar-mcrt.h`

2-D model, 2-D target grid,

calculation of numbers of arrivals, travel times, and amplitudes for Kirchhoff migration (sorting according to amplitudes)

optional pictures of rays together with the numbers of arrivals, and of travel times sorted according to their value

`mar/mgb-opt1.h`

2-D model, 2-D target grid,

calculation of travel times as an input for calculation of optimum initial parameters of Gaussian beams

`mar/mgp-mig1.h`

2-D model, 2-D target grid,

calculation of travel times and amplitudes for Gaussian packet migration

Numerical examples - package DATA (3/4)

`p1i/p1-mtt.h`
2-D model, 2-D target grid,
calculation of slowness vectors, travel times and ray amplitudes,
pictures of rays together with velocity, model block indices, and numbers of arrivals

`born/m2d-mtt.h`
2-D model, 2-D target grid,
calculation of slowness vectors, travel times and amplitudes for Born approximation

`vgr/vgr-q2.h`
1-D constant velocity gradient model, 2-D target grid,
calculation of geometrical spreading from gridded slowness vectors

`wb/wb2-loc.h`
3-D model, 3-D target grid,
calculation of travel times for kinematic hypocentre determination

Numerical examples - package DATA (4/4)

`98/98-mtt.h`

3-D model, 2-D target grid,
calculation of travel times,
pictures of travel times sorted according to their value or according to the ray history

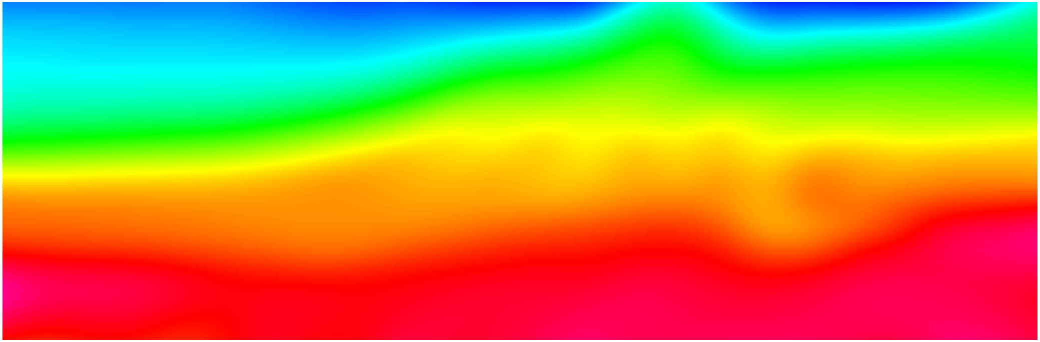
`qi/*-pfa.h`

six 2-D anisotropic models, 2-D target grid,
calculation of travel times using prevailing-frequency approximation of the coupling-ray theory,
pictures of travel times and their relative differences

Numerical examples - history file `mar-mcrt.h`

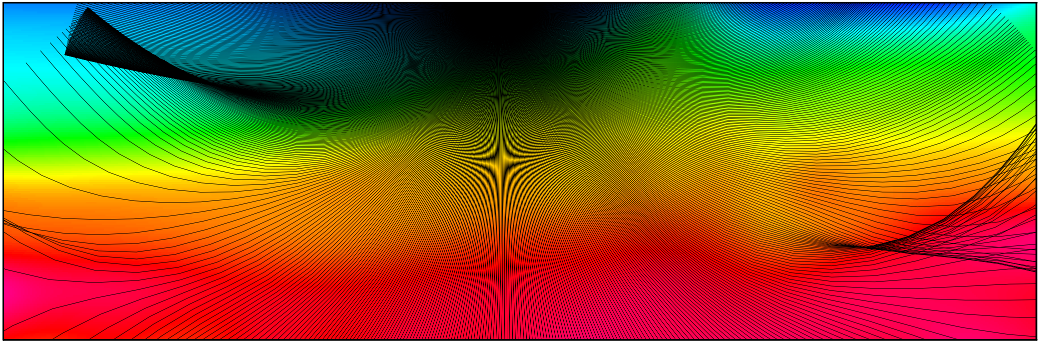
- smooth 2-D isotropic model Marmousi
- calculation of P-wave travel time

Numerical examples - history file `mar-mcrt.h`



P-wave slowness in the smooth 2-D model Marmousi

Numerical examples - history file `mar-mcrt.h`

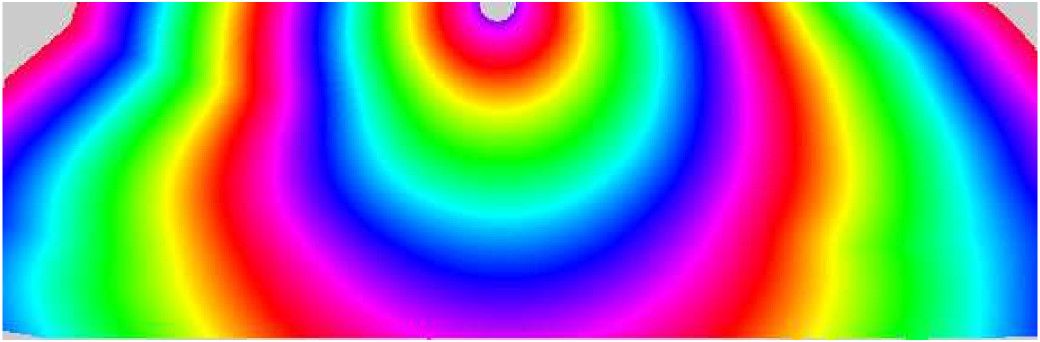


P-wave slowness and calculated rays

Numerical examples - history file `mar-mcrt.h`

Numbers of arrivals

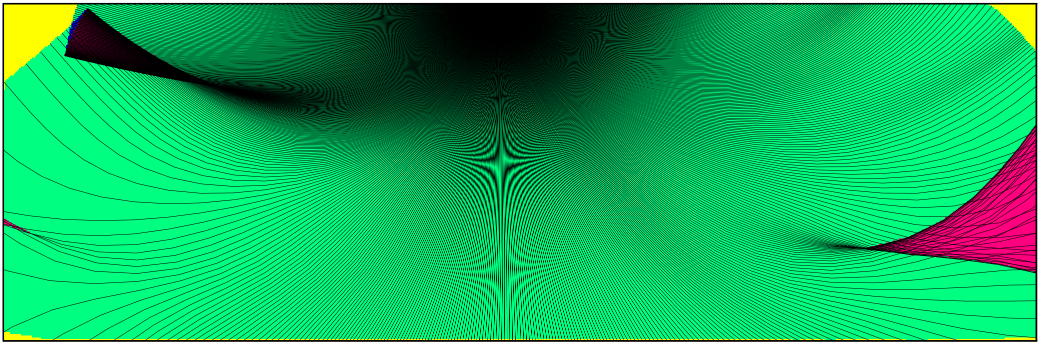
Numerical examples - history file `mar-mcrt.h`



First arrival travel times

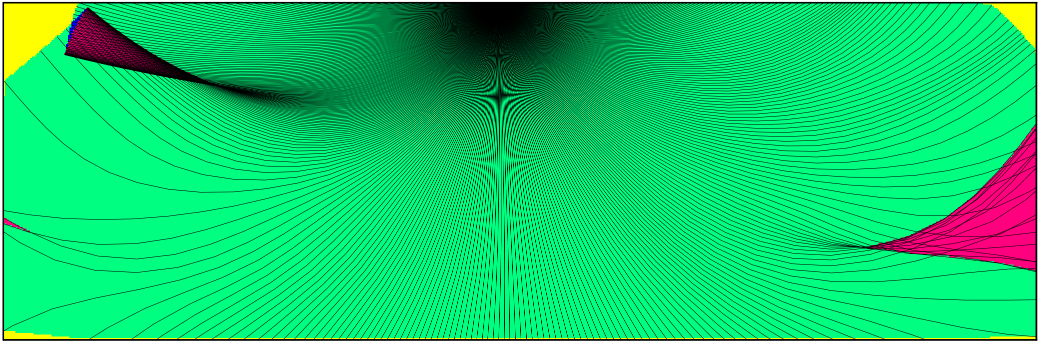
Second and third arrival travel times

Numerical examples



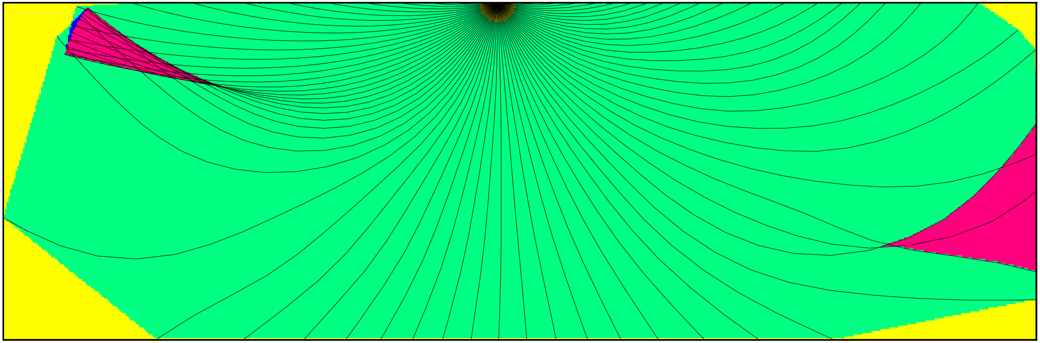
720 rays

Numerical examples



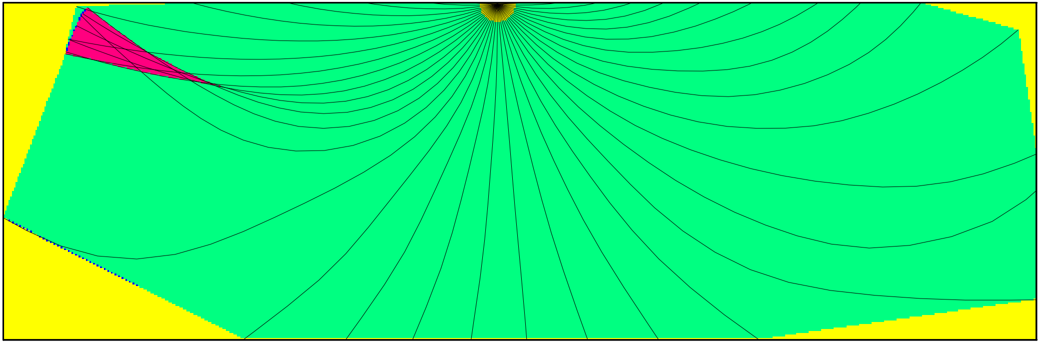
360 rays

Numerical examples



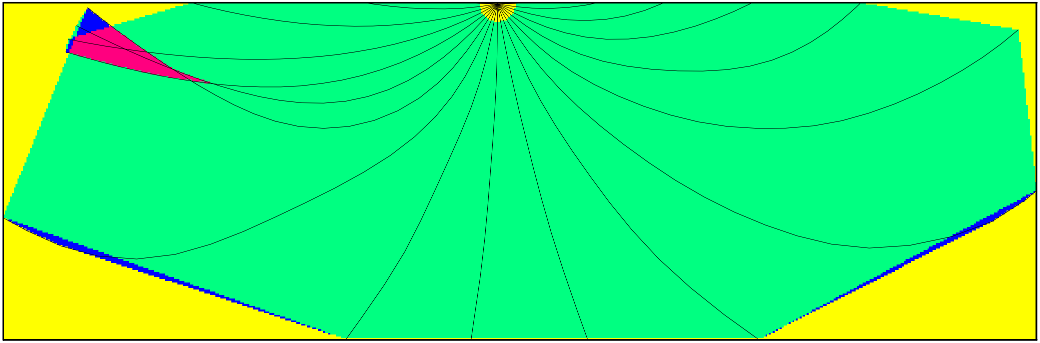
72 rays

Numerical examples



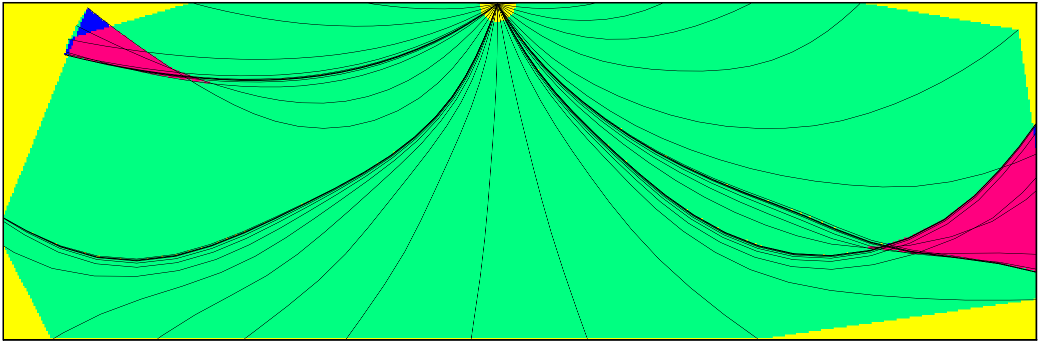
36 rays

Numerical examples



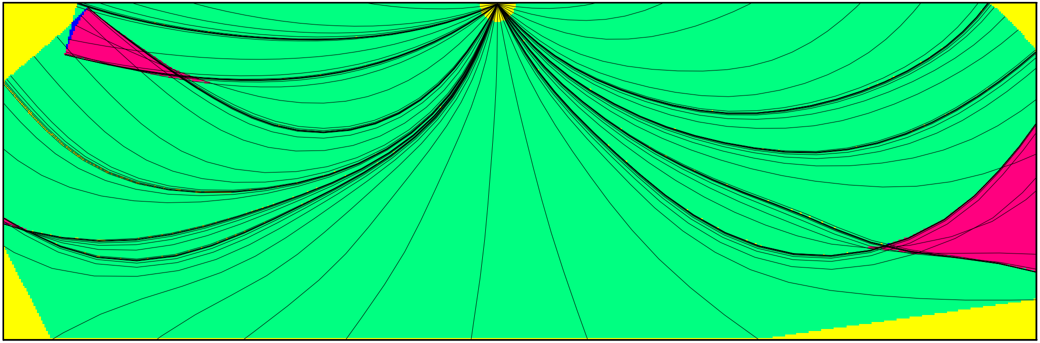
18 rays

Numerical examples



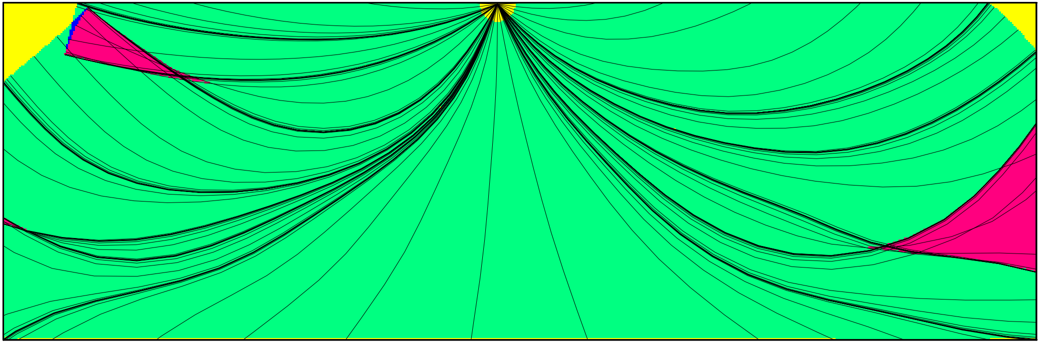
Ray history: caustics

Numerical examples



Ray history: caustics + reason of termination

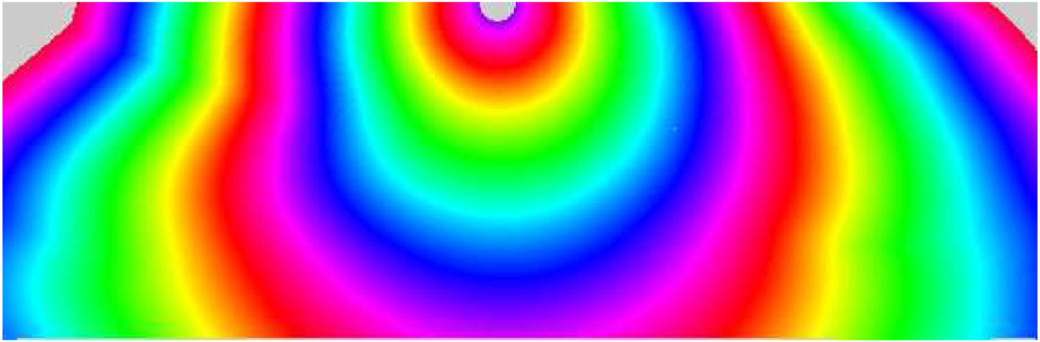
Numerical examples



Ray history: caustics + reason of termination + model boundaries

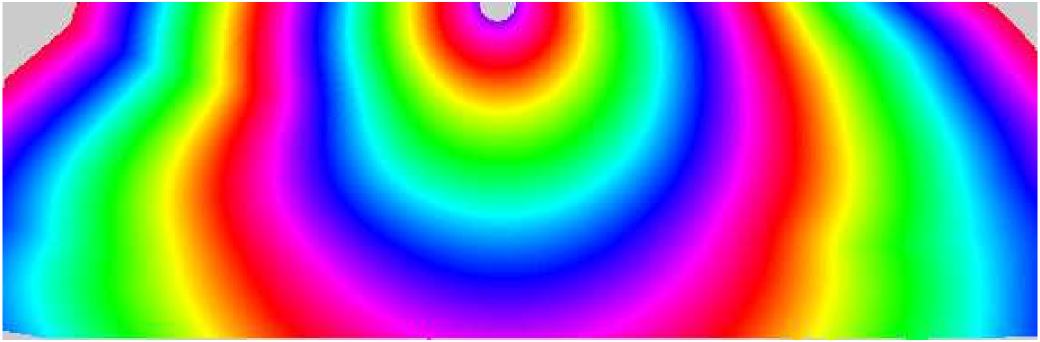
213 rays

Numerical examples



First arrivals - 213 rays

Numerical examples



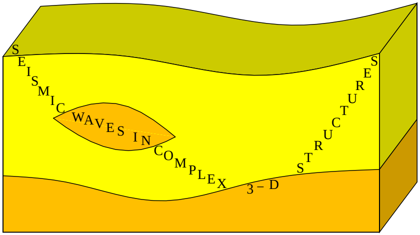
First arrivals - 720 rays

Second and third arrivals - 213 rays

Second and third arrivals - 720 rays

Acknowledgements

The research has been supported by the Grant Agency of the Czech Republic within project "Asymptotic full-waveform inversion of seismic data in complex media" and by the consortium "Seismic Waves in Complex 3-D Structures"



http://sw3d.cz

Thank you for your attention!