

SW4 a VisIt - 3D seismické modelování a zobrazení

Seminář o seismologickém softwaru

Martin Labuta

MFF UK, GFÚ AVČR

04.12.2020

Obsah

- 1 Odkazy
- 2 Teoretický úvod k SW4
- 3 Instalace a spouštění SW4
- 4 SW4 funkcionalita
 - Input soubor
 - Hlavička zdrojového souboru
 - Seismické zdroje
 - Topografie
 - Materiálový model
 - Výstupní příkazy
 - Pluginy
- 5 VisIt
 - Instalace & Instalace SW4 pluginu
 - Visualizace

Obsah

- 1 Odkazy
- 2 Teoretický úvod k SW4
- 3 Instalace a spouštění SW4
- 4 SW4 funkcionalita
 - Input soubor
 - Hlavička zdrojového souboru
 - Seismické zdroje
 - Topografie
 - Materiálový model
 - Výstupní příkazy
 - Pluginy
- 5 VisIt
 - Instalace & Instalace SW4 pluginu
 - Visualizace

Užitečné odkazy

SW4 - geodynamcics

<https://geodynamics.org/cig/software/sw4/>

SW4 - github

<https://github.com/geodynamics/sw4/>

SW4 - instalace

<https://github.com/geodynamics/sw4/blob/master/doc/SW4-Installation.pdf>

SW4 - manuál

<https://github.com/geodynamics/sw4/blob/master/doc/SW4-UsersGuide.pdf>

O projektu SW4

<https://computing.llnl.gov/projects/serpentine-wave-propagation>

Články

- Sjögreen, B.; Petersson, N.A. (2012), A Fourth Order Accurate Finite Difference Scheme for the Elastic Wave Equation in Second Order Formulation, Journal of Scientific Computing, 52 (1) , 17-48, doi: 10.1007/s10915-011-9531-1, url: <http://link.springer.com/10.1007/s10915-011-9531-1>
- Petersson, N.A.; Sjögreen, B. (2012), Stable and efficient modeling of anelastic attenuation in seismic wave propagation, Communications in Computational Physics, 12 (01) , 193-225
- Petersson, N.A.; Sjögreen, B. (2015), Wave propagation in anisotropic elastic materials and curvilinear coordinates using a summation-by-parts finite difference method, Journal of Computational Physics, 299, 820-841, doi: 10.1016/j.jcp.2015.07.023, url: <http://linkinghub.elsevier.com/retrieve/pii/S0021999115004684>

Obsah

- 1 Odkazy
- 2 Teoretický úvod k SW4
- 3 Instalace a spouštění SW4
- 4 SW4 funkcionalita
 - Input soubor
 - Hlavička zdrojového souboru
 - Seismické zdroje
 - Topografie
 - Materiálový model
 - Výstupní příkazy
 - Pluginy
- 5 VisIt
 - Instalace & Instalace SW4 pluginu
 - Visualizace

Co je SW4

”SW4 je nástroj pro 3D seismické modelování vlnového pole s podmínkou volného povrchu na svrchním rozhraní, útlumovým supergridem ve vzdálené zóně a s možností bodových silových zdrojů a/nebo bodových zdrojů se zadaným momentovým tensorem. Časová funkce každého zdroje může být vybrána z předdefinovaných možností”

Využívá:

- výpočet pomocí metody konečných diferencí s polynomy 4. řádu
- explicitní časové schéma
- heterogenní 3D materiálový model
- kartézské souřadnice pro řešení vlnové rovnice
- křivočarou mřížku pro vystižení topografie

Vývoj SW4

- Vývoj v LLNL - Lawrence Livermore National Laboratory, Livermore, Kalifornie, USA
- Autory jsou Anders Petersson a Bjorn Sjogreen
- Předchůdcem SW4 (Seismic Waves, 4th order) je WPP (Wave Propagation Program) - vývoj na WPP začal před rokem 2006
- SW4 se stále vyvíjí - na githubu branch development
- SW4 je napsáno v C, výpočty běží ve Fortranu

Obsah

- 1 Odkazy
- 2 Teoretický úvod k SW4
- 3 Instalace a spouštění SW4**
- 4 SW4 funkcionalita
 - Input soubor
 - Hlavička zdrojového souboru
 - Seismické zdroje
 - Topografie
 - Materiálový model
 - Výstupní příkazy
 - Pluginy
- 5 VisIt
 - Instalace & Instalace SW4 pluginu
 - Visualizace

Instalace make

Prerekvizity Lapack, Blas, PROJ4 pro materiálové soubory rfile, MPI, exe soubor /sw4-v2.0/optimize/sw4

make.inc soubor

```
proj = yes
```

```
etree = no
```

```
SW4ROOT = /usr
```

```
FC = mpif90
```

```
CXX = mpicxx
```

```
EXTRA_LINK_FLAGS =-L/usr/lib -llapack -lblas -lgfortran
```

```
cd /my/installation/dir/sw4-v2.0
```

```
make clean
```

```
make
```

```
cd pytest
```

```
./test_sw4.py
```

Instalace Cmake

```
mkdir build
cd build
cmake -DPROJ4_HOME=/apps/all/PROJ/5.0.0-intel-2018a/ ..
make
ctest
```

executable in the build/bin directory

Spouštění

- Lokálně (příklad paralelizace na 8 vláknech)

```
cd do složky s in souborem
```

```
mpiexec -np 8 sw4 berkeley-r.in
```

- Na clusteru PBS (Portable Batch System)

- 1 Vytvoření run.sh souboru

- 2 Submit na cluster přes PBS

```
qsub -k oe run.sh
```

run.sh

```
#!/bin/sh

# embedded options to qsub - start with #PBS
# -- Name of the job --
#PBS -N mytina_test05
#PBS -r n
#PBS -q verylong
#PBS -l nodes=10:ppn=16
#PBS -l walltime=47:59:00

#ulimit -a
#ulimit -l unlimited

cd /home/labuta/projects/mytina/test05

# here follows the command you want to execute

# tyto 3 posledni radky jsou zalomene entrem, nutno odentrovat
mpiexec -f $PBS_NODEFILE -np 160
/home/labuta/Install/sw4-master/optimize/sw4
/home/labuta/projects/mytina/test05/test05.in
```

Obsah

- 1 Odkazy
- 2 Teoretický úvod k SW4
- 3 Instalace a spouštění SW4
- 4 SW4 funkcionalita**
 - Input soubor
 - Hlavička zdrojového souboru
 - Seismické zdroje
 - Topografie
 - Materiálový model
 - Výstupní příkazy
 - Pluginy
- 5 VisIt
 - Instalace & Instalace SW4 pluginu
 - Visualizace

Postup

- SW4 používá soubor vstupních příkazů (.in soubor)
- Některé příkazy jsou povinné ale mohou mít více volitelných parametrů, jiné příkazy je možno vynechat, jsou i příkazy které se navzájem vylučují
- Po skončení programu možná vizualizace výstupních dat - seismogramů a snapshotů vlnového pole buď ve Visitu nebo Matlabu/Octave

test.in

```
# test pro IT4I cluster
fileio pfs=1 path=out verbose=2 printcycle=10 nwriters=16
grid nx=401 ny=401 nz=201 h=10
supergrid width=600
refinement zmax=1000
refinement zmax=500
time t=4
block vp=3500 vs=2000 rho=2000
block vp=6000 vs=3500 rho=2700 z1=500
block vp=8000 vs=4500 rho=3300 z1=1000 z2=2000
source type=Ricker x=2000 y=2000 z=1000 fx=1. freq=2. t0=0.5
sac x=2000 y=2000 depth=0 file=1 writeEvery=100
image x=2000 mode=uz timeInterval=0.1
image y=2000 mode=uz timeInterval=0.1
image z=0 mode=uz timeInterval=0.1
```


Základní příkazy

```
fileio pfs=1 path=out verbose=2 printcycle=10 nwriters=16
grid nx=401 ny=401 nz=201 h=10
supergrid width=600
refinement zmax=1000
refinement zmax=500
time t=4
```

Základní příkazy

fileio path=... verbose=... printcycle=... pfs=... nwriters=...

Základní příkazy

grid nx=... ny=... nz=... x=... y=... z=... h=... lat=... lon=... az=...
mlat=... mlon=... proj=... ellps=... datum=... scale=... lat p=... lon p=...

Určení kroku mřížky

$$P = \frac{\min C_s}{hf_{max}} \quad (1)$$

kde chceme dostat P mezi 6 a 10

Základní příkazy

supergrid gp=... dc=...width=...

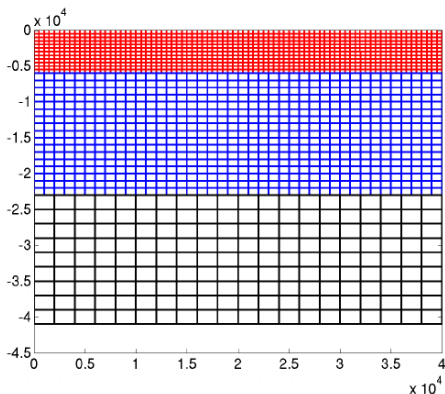
- Nejmenší doporučená šíře je gp=20 bodů
- Doporučeno je používat option width kvůli refinement



Základní příkazy

refinement zmax=..

- Je doporučeno zvolit hloubku rozhraní tak aby poměr C_s/h byl podobný přes celou výpočetní oblast
- Musí být nejméně 8 bodů na výšku pokud je pouze jedno rozhraní s další mřížkou
- Jinak uprostřed mřížky musí být nejméně 12 bodů



Základní příkazy

time t=... steps=... utcstart=...

Zdroje

```
source type=Ricker x=2000 y=2000 z=1000 fx=1. freq=2. t0=0.5
```

```
source x=... y=... z=... lat=... lon=... depth=... topodepth=... m0=...  
mxx=... mxy=... mxz=... myy=... myz=... mzz=... f0=... fx=... fy=...  
fz=... rake=... strike=... dip=... t0=... freq=... type=... ncy=... dfile=...
```

Pro určení t_0 autoři doporučují vztah

$$t_0 = 6\alpha \quad (2)$$

kde alfa je $1/\text{freq}$ pro Gausse a $2\pi/\text{freq}$ pro Rickera

Frekvenční obor časové funkce

- Fundamentální frekvence f_0 odpovídá
 - ▶ freq pro Rickerovu časovou funkci zdroje
 - ▶ $\text{freq}/2\pi$ pro Llu, Brune, Gauss
- Potom maximální frekvence f_{\max} přibližně odpovídá
 - ▶ $2.5 f_0$ pro Rickera a Gaussovské funkce
 - ▶ $4.0 f_0$ Brune

Maximální frekvence f_{\max} generuje nejkratší vlnové délky a tím ovlivňuje velikost mřížky

Určení kroku mřížky

$$P = \frac{\min C_s}{hf_{\max}} \quad (3)$$

Předdefinované časové funkce

- Rickerova funkce
- Integrál Rickerovy funkce

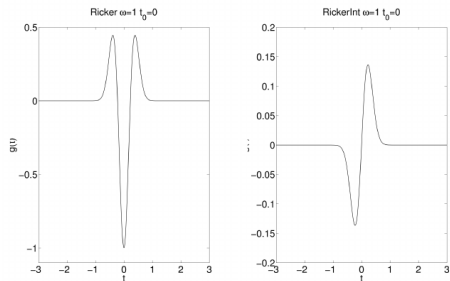


Figure 4.2: Ricker (left) and RickerInt (right) with $\omega = 1$ and $t_0 = 0$.

Předdefinované časové funkce

- Rickerova funkce
- Integrál Rickerovy funkce
- Gaussovská funkce
- Integrál Gaussovy funkce

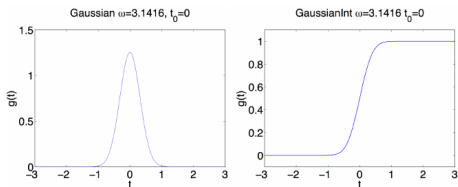


Figure 4.1: Gaussian (left) and GaussianInt (right) with $\omega = \pi$ and $t_0 = 0$.

Předdefinované časové funkce

- Rickerova funkce
- Integrál Rickerovy funkce
- Gaussovská funkce
- Integrál Gaussovy funkce
- Bruneho funkce
- Vyhlazená Bruneho funkce

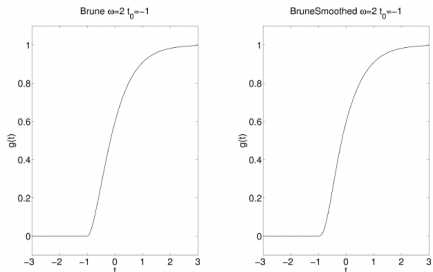


Figure 4.3: Brune (left) and BruneSmoothed (right) with $\omega = 2$ and $t_0 = -1$.

Předdefinované časové funkce

- Rickerova funkce
- Integrál Rickerovy funkce
- Gaussovská funkce
- Integrál Gaussovy funkce
- Bruneho funkce
- Vyhlazená Bruneho funkce
- Liuova funkce

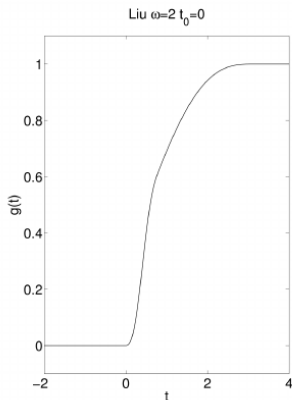


Figure 4.4: Liu time function with $\omega = 2$ and $t_0 = 0$.

Předdefinované časové funkce

- Rickerova funkce
- Integrál Rickerovy funkce
- Gaussovská funkce
- Integrál Gaussovy funkce
- Bruneho funkce
- Vyhlazená Bruneho funkce
- Liuova funkce
- Trojúhelníková funkce
- "Pila"

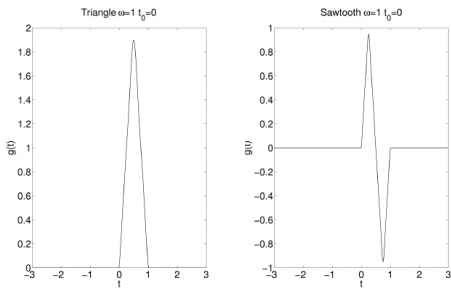


Figure 4.5: Triangle (left) and Sawtooth (right) with $\omega = 1$ and $t_0 = 0$.

Předdefinované časové funkce

- Rickerova funkce
- Integrál Rickerovy funkce
- Gaussovská funkce
- Integrál Gaussovy funkce
- Bruneho funkce
- Vyhlazená Bruneho funkce
- Liuova funkce
- Trojúhelníková funkce
- "Pila"
- "Rampa"
- Hladká křivka

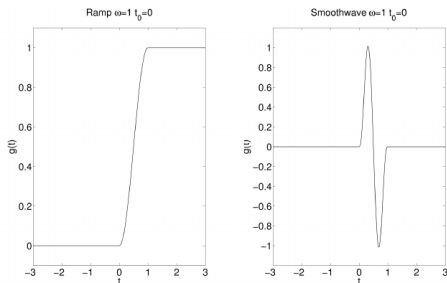


Figure 4.6: Ramp (left) and Smoothwave (right) with $\omega = 1$ and $t_0 = 0$.

Předdefinované časové funkce

- Rickerova funkce
- Integrál Rickerovy funkce
- Gaussovská funkce
- Integrál Gaussovy funkce
- Bruneho funkce
- Vyhlazená Bruneho funkce
- Liuova funkce
- Trojúhelníková funkce
- "Pila"
- "Rampa"
- Hladká křivka
- Dirac

Zdroje

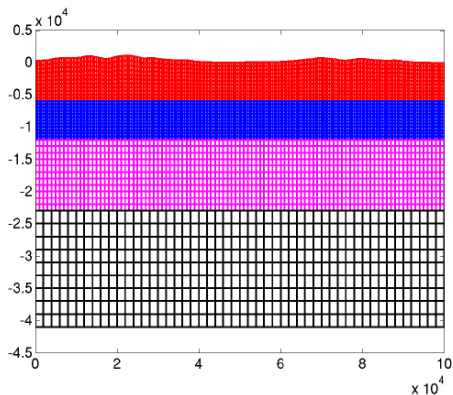
rupture file=... Vyžaduje soubor SRF (Standard Rupture Format)

Info o SRF

http://equake-rc.info/static/paper/SRF-Description-Graves_2.0.pdf

Topografie

topography input=... file=...
resolution=... zmax=... order=...
smooth=... gaussianAmp=...
gaussianXc=... gaussianYc=...
gaussianLx=... gaussianLy=...



Příkazy pro materiálový model

attenuation phasefreq=... nmech=... maxfreq=... minppw=...
qmultiplier=...

anisotropy

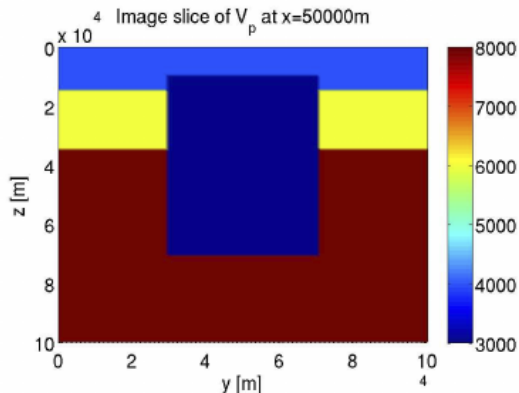
randomize amplitude=... lengthscale=... lengthscalez=... gradient=...
sdthreshold=... seed1=... seed2=... seed3=...

block vp=... vs=... rho=... qp=... qs=... vpgrad=... vsgrad=...
rhograd=... absdepth=... x1=... x2=... y1=... y2=... z1=... z2=...

efile; ifile; pfile; rfile; sfile (pouze development branch na Githubu)

Příkaz block

```
block vp=4000 vs=2500 rho=2000
block vp=6000 vs=3500 rho=2700 z1=15000
block vp=8000 vs=4500 rho=3300 z1=35000 z2=100000
block vp=3000 vs=2000 rho=1000 x1=4000 x2=8000 y1=3000 y2=7000 z1=10000 z2=70000
```

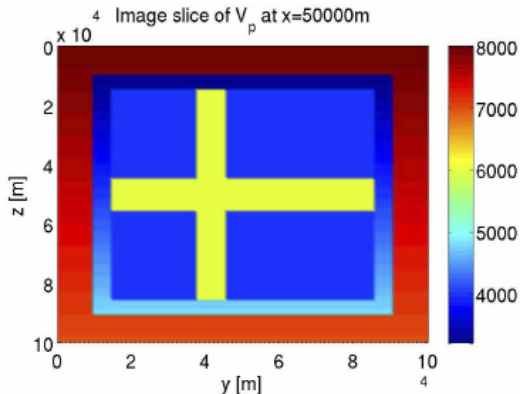


Příkaz block

```

block vp=8000 vs=4500 rho=3300 vpgrad=-0.01
block vp=3000 vs=2000 rho=1000 x1=1e4 x2=9e4 y1=1e4 y2=9e4 z1=1e4 z2=9e4 vpgrad=0.2
block vp=4000 vs=2500 rho=2000 x1=15e3 x2=85e3 y1=15e3 y2=85e3 z1=15e3 z2=85e3
block vp=6000 vs=3500 rho=2700 x1=15e3 x2=85e3 y1=15e3 y2=85e3 z1=45e3 z2=55e3
block vp=6000 vs=3500 rho=2700 x1=15e3 x2=85e3 z1=15e3 z2=85e3 y1=38e3 y2=45e3

```



Materiálové soubory

- pfile
- rfile

Ukázat Matlab scripty

Výstupní příkazy

rec x=... y=... z=... lat=... lon=... depth=... topodepth=... sta=...
file=... writeEvery=... nsew=... usgsformat=... sacformat=... variables=...

image x=... y=... z=... time=... timeInterval=... cycle=...
cycleInterval=... file=... mode=... precision=...

volimage file=... mode=... precision=... time=... timeInterval=... cycle=...
cycleInterval=... startTime=...

Pluginy a funkce o kterých se nezmiňuje žádný manuál ale přesto jsou přiloženy a jediná dokumentace je krátké README o funkcích ale nic o pluginech

- Pluginy pro VisIt
 - ▶ SW4image
 - ▶ Volimage
- Funkce pro Matlab
 - ▶ readimage.m
 - ▶ plotimage.m
 - ▶ readimage3d.m
 - ▶ readsac.m
 - ▶ writesac.m
 - ▶ atd.

Obsah

- 1 Odkazy
- 2 Teoretický úvod k SW4
- 3 Instalace a spouštění SW4
- 4 SW4 funkcionalita
 - Input soubor
 - Hlavička zdrojového souboru
 - Seismické zdroje
 - Topografie
 - Materiálový model
 - Výstupní příkazy
 - Pluginy
- 5 VisIt
 - Instalace & Instalace SW4 pluginu
 - Visualizace

Visit - užitečné odkazy

Visit LLNL

<https://wci.llnl.gov/simulation/computer-codes/visit/>

Visit repozitář

<https://portal.nersc.gov/project/visit/releases/>

Visit github

<https://github.com/visit-dav/visit>

Visit wiki kompilace

https://www.visitusers.org/index.php?title=Build_visit_overview

Visit wiki kompilace pluginů

http://www.visitusers.org/index.php?title=Building_plugins_using_CMake

O VisItu

- Je to open source software určené k vizualizaci, animacím a analýze dat
- Předností je možnost paralelizace a komunikace klient-server
- Vizualizace velkého množství typů souborů a možnost instalace pluginů pro další typy
- Využívá různé knihovny, např. QT pro uživatelské rozhraní, Python, VTK (Visualization ToolKit) pro vizualizační algoritmy
- Windows, Mac, UNIX (Linux atd,)

Kompilace VisItu ze zdrojových kódů

```
wget http://portal.nersc.gov/svn/visit/trunk/releases/3.0.0/build_visit3_0_0
chmod 700 build_visit3_0_0
env ZLIB_FILE="zlib-1.2.11.tar.gz" ZLIB_VERSION="1.2.11" ./build_visit3_0_0 --
makeflags -j4 --mpich --openssl --zlib --gdal
chmod 755 visit-install3_0_0
./visit-install3_0_0 3.0.0 linux-x86_64 /usr/local/visit
```

Kompilace SW4 pluginu ze zdrojových kódů

```
cd /home/martin/Downloads/sw4-master/tools/sw4img  
xml2cmake -clobber -public sw4.xml  
cmake .
```

Pak je nutné projít adresáře

```
/sw4-master/tools/sw4img/CMakeFiles/Esw4imgDatabase_par.dir  
/sw4-master/tools/sw4img/CMakeFiles/Esw4imgDatabase_ser.dir  
/sw4-master/tools/sw4img/CMakeFiles/Isw4imgDatabase.dir  
/sw4-master/tools/sw4img/CMakeFiles/Msw4imgDatabase.dir
```

a zkontrolovat cesty v souboru build.make v každém adresáři a popř. ručně opravit.

Nakonec

```
sudo make
```

Galerie

