

# Manual do sistema GERAMODIN 1

**Luiz Alberto Santos<sup>\*†</sup>**

*\*Universidade Federal Fluminense (UFF)*

*Instituto de Geociências (EGG) - Sala 402*

*Departamento de Geologia e Geofísica (GGO)*

*Campus da Praia Vermelha*

*Av. Gen. Milton Tavares de Souza, s/n*

*Boa Viagem - Niterói - RJ - CEP: 24210-346*

*email: luizalb1@gmail.com*

*†Petróleo Brasileiro S.A. (Petrobras)*

*Av. República do Chile 330, Torre Leste, 12 andar*

*Centro - Rio de Janeiro, RJ - CEP: 20031-170*

(March 9, 2020)

Running head:

**ABSTRACT**

## INTRODUÇÃO

Este texto descreve o emprego do sistema *GERAMODIN1* criado para auxiliar as práticas de aula do curso Fundamentos de tomografia sísmica ministrado a partir do primeiro semestre de 2018 no Departamento de Geologia e Geofísica da Universidade Federal Fluminense. Este sistema *GERAMODIN1* tem finalidades didáticas e acadêmicas. Seu objetivo é permitir ao aluno em um ambiente de projeto: criar modelos de velocidade bidimensionais descritos por spline 1D variando lateralmente; modelar por traçado de raios os tempos de trânsito registrado na superfície a partir de difratores em profundidade; estimar por inversão tomográfica o campo de velocidade e; analisar os resultados.

## DESCRIÇÃO

O sistema *GERAMODIN1* é composto de quatro módulos e uma série de rotinas que rodam em sistema operacional Linux. Ele é composto de:

- interfaces em Python para o programa principal;
- interfaces em Python para as rotinas específicas de geração de modelo de velocidade, modelagem e inversão tomográfica e;
- rotinas desenvolvidas em C++ dos módulos específicos.

As interfaces em Python exibem o *frontend* contendo os campos a serem preenchidos além de botões de funções.

Além da interface gráfica e das rotinas citadas, o *GERAMODIN1* precisa que o pacote Seismic Unix (SU) (Stockwell e Cohen, 1998) esteja instalado para visualização de dados, modelos e gráficos.

O *geramodin1* demanda uma estrutura com subdiretórios dentro de um projeto:

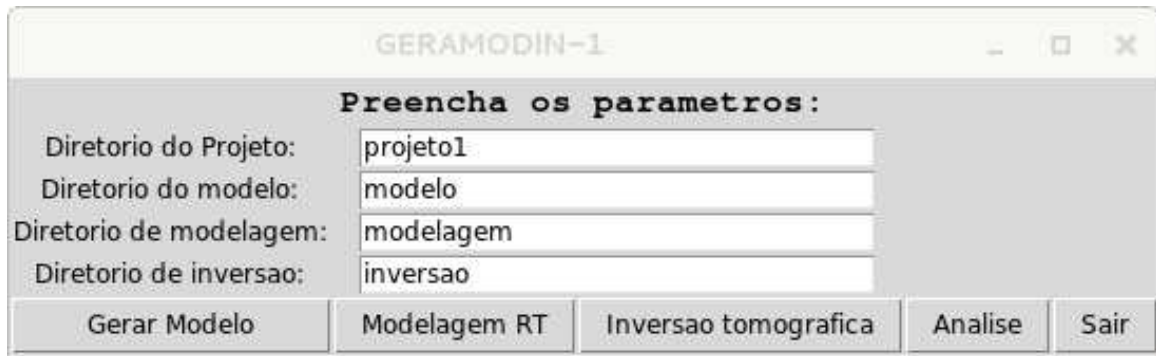


Figure 1: Interface do sistema geramodin 1.

- bin contendo os executáveis;
- modelo contendo o modelo de velocidade;
- modelagem contendo os resultados da modelagem inseridos no subdiretório DIFRAC;
- inversão com os resultados da inversão.

## APLICAÇÃO

Neste tópico apresenta-se uma aplicação de uso do *geramodin1* cujo demo está disponível no site *faladaterra.com*, abas *Home* e *Especialidades*. Na sessão Linha de Pesquisa Tomografia, clicando no formulário da Figura 1, será acionado o procedimento de *Download* do arquivo *demo\_geramodin1\_py.tar*. Depois de baixar este arquivo, copie-o para o diretório de trabalho que desejar. Em seguida você deverá executar o comando:

```
tar -xvf demo_geramodin1_py.tar
```

O comando acima irá descompactar os arquivos necessários para execução. O produto da descompactação estará contido no diretório *demo\_geramodin1\_py*. Para se assegurar que os programas rodarão, dentro do diretório *demo\_geramodin1\_py* execute por linha de comando:

```
chmod +x *
```

e em seguida:

```
chmod +x bin/*
```

Os comandos acima tornam os arquivos necessários ao uso do *geramodin1* em executáveis.

O diretório *demo\_geramodin1.py* possui um sub-diretório *bin* que contém os executáveis:

- *gerasplinegrid2d*: rotina para geração de modelos por splines 1D variando lateralmente.

- *rtrk4\_cfpv1*: rotina para traçado de raios cinemático e geração dos temps de trânsito registrados na superfície.

- *opasc*: rotina utilitária para operações com arquivos texto.

- *interpsplinetabrox*: rotina para interpolação de linhas (x,y) por spline. - *inversao\_green\_gcv0*:

rotina de inversão tomográfica não linear empregando gradiente conjugado voltada para CFP (difrações). Somente inverte o campo de velocidade que deve estar descrito por spline variando lateralmente.

Os requisitos para plena execução do sistema *geramodin1* são: sistema operacional Linux; ter o pacote livre Seismic Un\*x (SU) (Stockwell e Cohen, 1998) instalado e; as bibliotecas básicas do Python, sobretudo a Tkinter para uso da interface gráfica.

Para iniciar a aplicação, dentro do diretório *demo\_geramodin.py*, devemos digitar o comando:

```
geramodin1.sh
```

Trata-se de uma *shellscript* que chama a interface em Python.

Logo em seguida aparecerá a janela exibida na Figura 1. Trata-se do sistema principal que integra todas as funções envolvendo a criação de modelo, modelagem inversão e análise.

Vejamos cada uma das funções e rotinas correspondentes.

## Gerar modelo

Ao clicar no botão *GerarModelo* aciona-se a interface apresentada na Figura 2. Ela tem como objetivo criar um grid 2D representado por spline 1D variando lateralmente. A seguir descrevem-se os campos a serem preenchidos na interface.

Preencha os parametros:	
Diretorio do modelo:	modelo
Numero de nos de splines:	10
Prefixo do arquivo de modelagem:	vp
Numero de amostras na direcao x:	1000
Numero de amostras na direcao z:	500
Dimensao x em metros:	4.

0	1500
100	1500
200	1500
300	1600
400	1650
500	1700
600	1650
700	1600
800	1500
900	1500

Figure 2: Interface para geração de modelo de velocidade por spline 1D.

- Diretório do modelo: diretório onde será armazenado o modelo de velocidade criado.
- Numero de nós de splines: Número de nós da função spline 1D.
- Prefixo do arquivo de modelo: Digitar somente o nome do arquivo que antecede sem extensão.
- Numero de amostras na direção x: Número de amostras do grid de velocidade na direção x, horizontal.

- Numero de amostras na direção z: Número de amostras do grid de velocidade na direção z, vertical.
- Dimensao x em metros: Dimensão do grid (supostamente quadrado) em metros.

Logo abaixo há um espaço no qual devem ser digitados pares de dados onde: a primeira coluna corresponde à posição x (horizontal) do nó da spline (em amostras) e; na segunda coluna deve ser informado o valor da velocidade em m/s. Note que o número de linhas deve ser o mesmo daquele informado no campo Numero de nos de splines.

Abaixo dos campos acima listados, há dois botões de ações:

Gerar Modelo;

Sair.

Com os parâmetros indicados, ao clicar no botão *GerarModelo* o arquivo de velocidade é gerado e, em seguida visualizado como na Figura 3. Esta é uma das funcionalidades que utiliza um recurso do SU. Finalmente, ao clicar em Sair, a interface de geração de modelo é encerrada e retornamos para a interface principal, Figura 1.

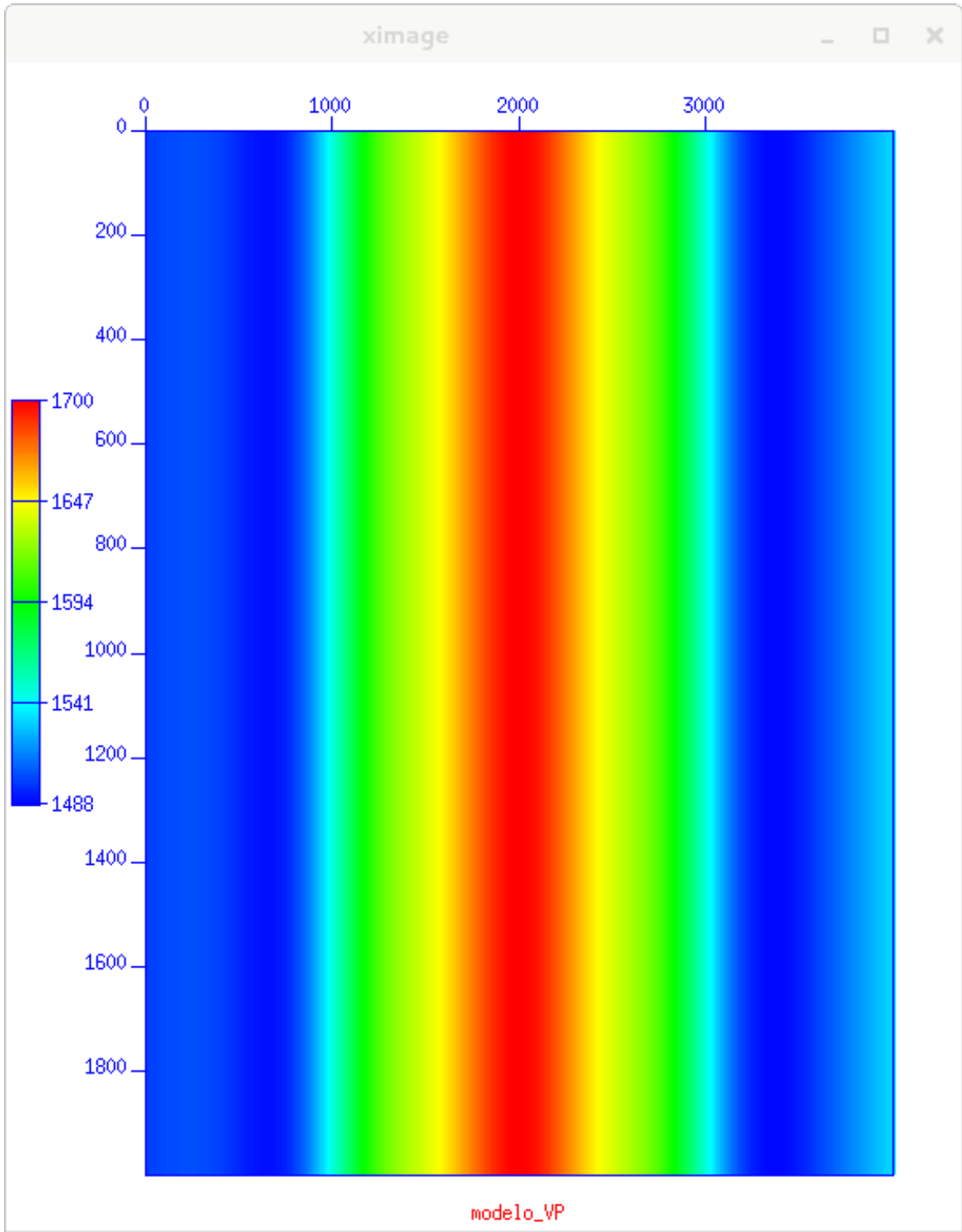
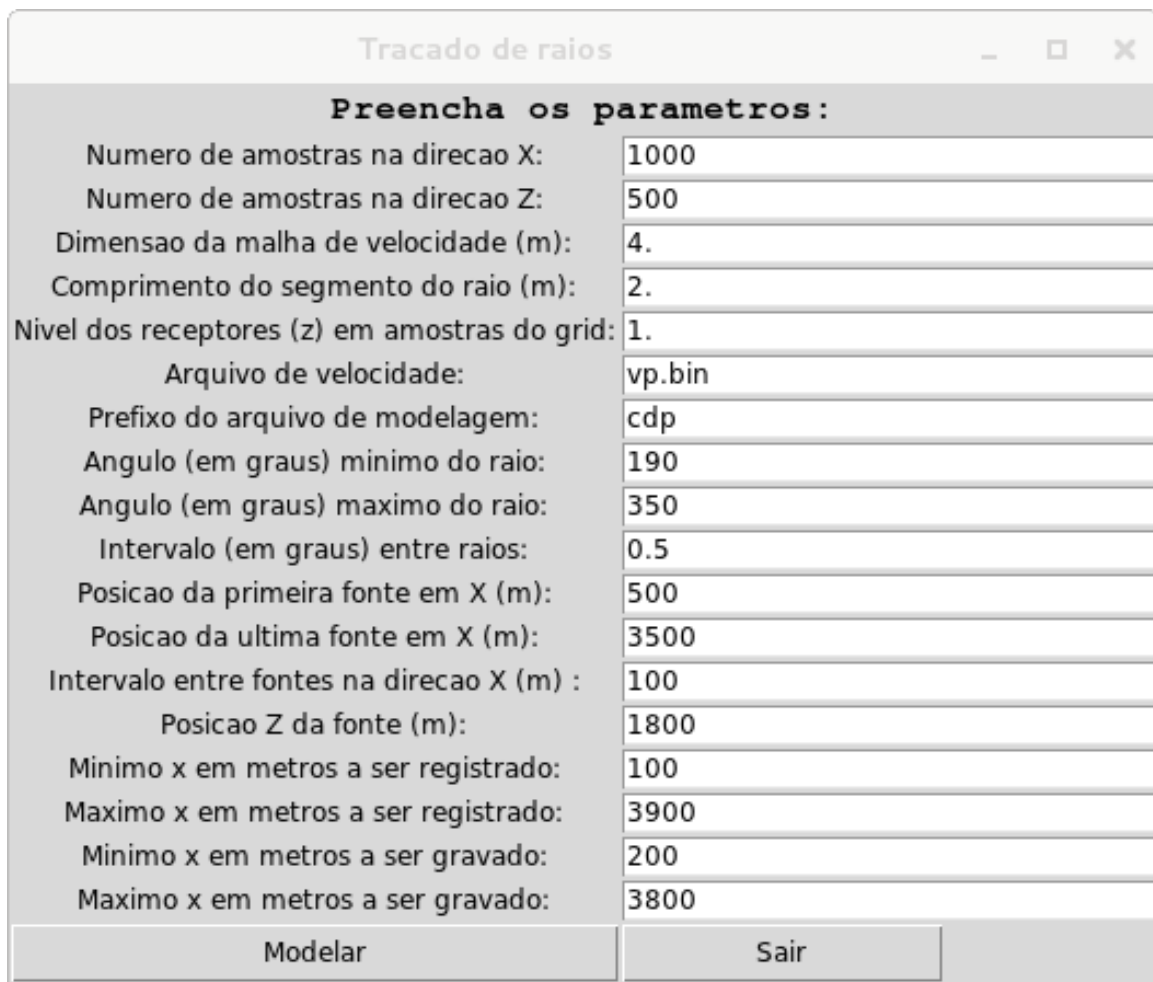


Figure 3: Modelo de velocidade gerado.

## Modelagem RT

Clicando no botão *Modelagem\_RT* abre-se uma nova interface, Figura ???. Esta interface destina-se a calcular os tempos de trânsito registrados na superfície para uma ou mais fontes, difratores, em profundidade. Vejamos os campos a serem preenchidos na interface Modelagem RT.



Preencha os parametros:	
Numero de amostras na direcao X:	1000
Numero de amostras na direcao Z:	500
Dimensao da malha de velocidade (m):	4.
Comprimento do segmento do raio (m):	2.
Nivel dos receptores (z) em amostras do grid:	1.
Arquivo de velocidade:	vp.bin
Prefixo do arquivo de modelagem:	cdp
Angulo (em graus) minimo do raio:	190
Angulo (em graus) maximo do raio:	350
Intervalo (em graus) entre raios:	0.5
Posicao da primeira fonte em X (m):	500
Posicao da ultima fonte em X (m):	3500
Intervalo entre fontes na direcao X (m) :	100
Posicao Z da fonte (m):	1800
Minimo x em metros a ser registrado:	100
Maximo x em metros a ser registrado:	3900
Minimo x em metros a ser gravado:	200
Maximo x em metros a ser gravado:	3800

Modelar      Sair

Figure 4: Interface para modelagem por traçado de raios.

-Numero de amostras na direcao X: Número de amostras do modelo de velocidade na direção horizontal.

-Numero de amostras na direcao Z: Número de amostras do modelo de velocidade na direção



vertical.

-Dimensao da malha de velocidade (m): Dimensão da malha ou grid em metros.

-Comprimento do segmento do raio (m): Comprimento do segmento de raio em metros.

-Nível dos receptores (z) em amostras do grid: Nível dos receptores na vertical em amostras de grid.

-Arquivo de velocidade: Nome completo do arquivo de velocidade. O arquivo já deve estar no diretório modelo.

-Prefixo do arquivo de modelagem: Nome do arquivo sem a extensão. A extensão binária, texto, segy, SU, etc.. será fornecida pelo próprio sistema. Por ora, somente há as extensões binária (bin) e texto (txt).

-Angulo minimo do raio (em graus): Para a modelagem serão modelados vários raios compondo um leque que parte de um ponto difrator em uma coordenada  $(x_i, z_i)$ . Neste campo informa-se o ângulo mínimo de acordo com a convenção a seguir:

.....270

...180.....|.....0

.....090

-Angulo (em graus) maximo do raio: Neste campo informa-se o ângulo máximo de acordo com a convenção adotada acima.

-Intervalo (em graus) entre raios: Informa-se aqui o ângulo entre raios consecutivos para o leque entre os ângulos mínimo e máximo.

-Posicao da primeira fonte em X (m): Informa-se neste campo a posição em x, horizontal, da primeira fonte difratora.

-Posicao da ultima fonte em X (m): Informa-se neste campo a posição em x, horizontal, da

última fonte difratora.

-Intervalo entre fontes na direção X (m) : E aqui informa-se o intervalo regular entre cada um dos difratores.

-Posição Z da fonte (m): Neste campo preenchemos a profundidade dos difratores.

-Mínimo x em metros a ser registrado: Preenche-se aqui a posição mínima onde se deseja registrar as chegadas de raios na superfície. Na prática, todos os raios a esquerda desta posição não serão registrados. Recomenda-se que esta posição/valor em X seja inferior ao mínimo X a ser gravado.

-Máximo x em metros a ser registrado: Todos os raios a direita desta posição também não serão registrados. Assim, recomenda-se que esta posição/valor em X seja superior ao máximo X a ser gravado.

-Mínimo x em metros a ser gravado: A curva de tempo vs posição de cada difrator será interpolada regularmente a cada 10m entre o valor informado neste campo e o campo seguinte (Máximo x em metros a ser gravado). Assim recomenda-se que este campo tenha valor superior ao valor informado no campo Mínimo x em metros a ser registrado.

-Máximo x em metros a ser gravado: Posição máxima para o receptor. Recomenda-se que este campo tenha valor inferior ao valor informado no campo Máximo x em metros a ser registrado.

Abaixo dos campos acima listados, há dois botões de ações:

- Modelar: Ao clicar neste botão inicia-se a modelagem por traçado de raios. Com os valores informados na Figura 4 serão geradas curvas de tempos de trânsito (tempo vs posição x) registradas na superfície para 31 difratores  $((3500-500)/100 + 1)$ . Cada uma das curvas geradas é exibida para controle de qualidade da modelagem. Neste caso serão exibidas 31

curvas com o fundo do modelo de velocidade empregado. Não há relação entre a curva e o fundo apresentado. Se o resultado da modelagem estiver satisfatório para o usuário, sugere-se fechar todas as janelas empregando-se o comando *killall ximage* por linha de comando. Assim todas as janelas com a função SU *ximage* serão encerradas. As Figuras 5, 6 e 7 são exemplos de curvas de tempo de trânsito dos difrtores nas posições (500,1800), (2000,1800) e (3500,1800) respectivamente. Os arquivos gerados são adequadamente armazenados no diretório modelagem/DIFRAC.

- Sair: Ao clicar neste botão a janela de modelagem é encerrada.

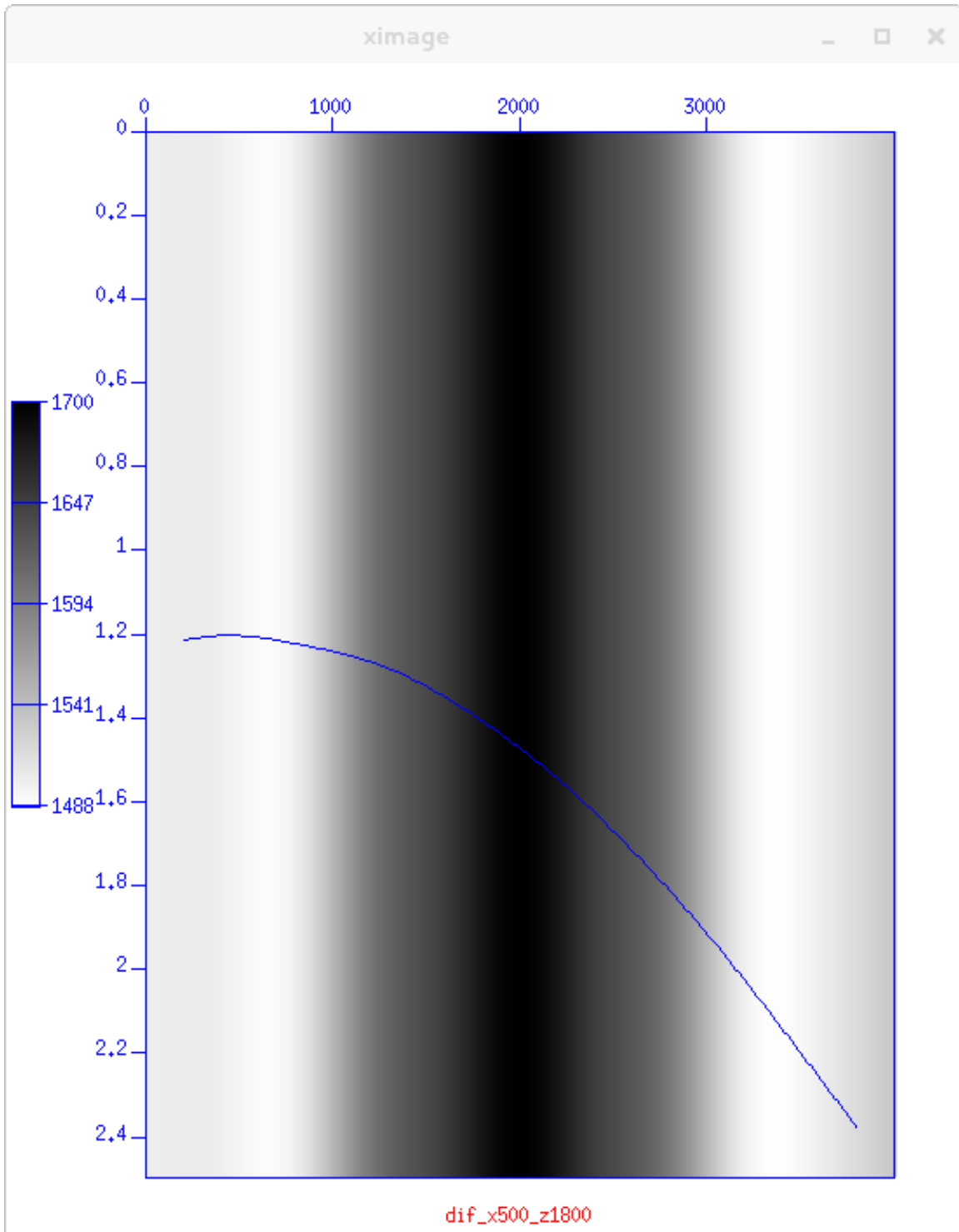


Figure 5: Curva de tempo de trânsito do difrator na posição (500,1800). Eixo horizontal em metros e o vertical em segundos.

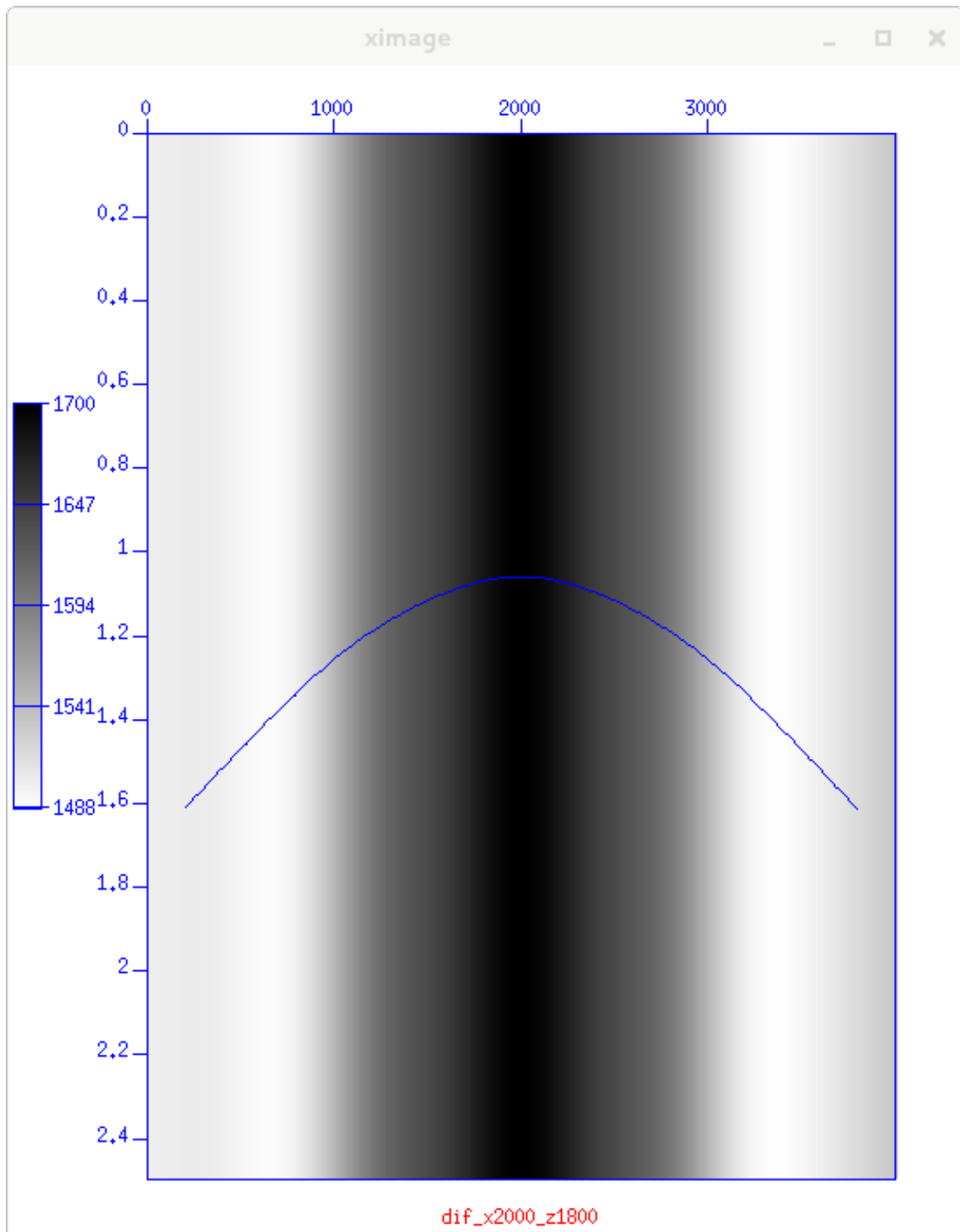


Figure 6: Curva de tempo de trânsito do difrator na posição (2000,1800). Eixo horizontal em metros e o vertical em segundos.

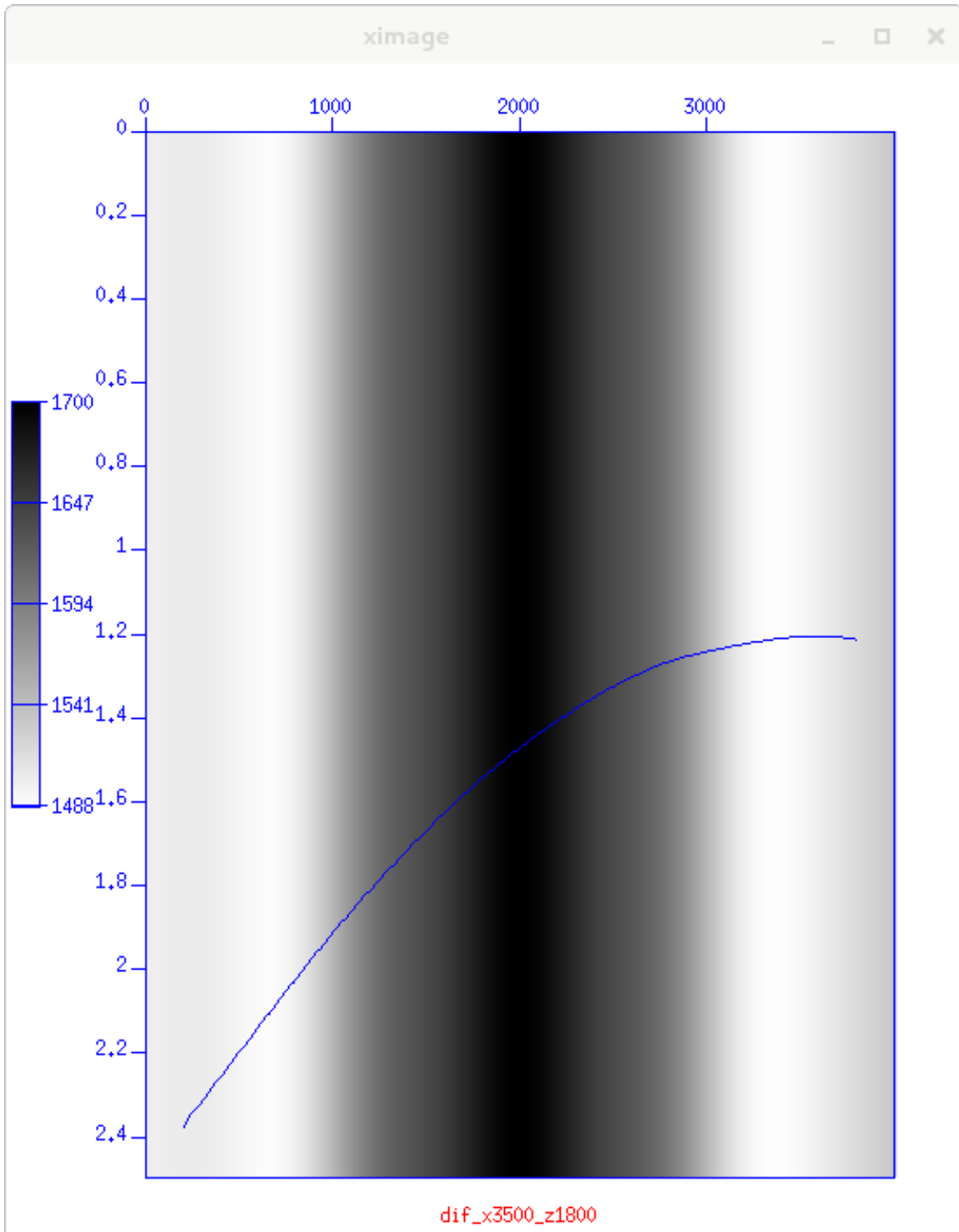


Figure 7: Curva de tempo de trânsito do difrator na posição (3500,1800). Eixo horizontal em metros e o vertical em segundos.

## Inversão tomográfica de difrações

Passamos agora ao processo de inversão tomográfica no qual deseja-se estimar o campo de velocidades a partir das curvas de tempo de trânsito calculadas com a modelagem do subtópico anterior. Clicando no botão Inversão tomográfica (Figura 1) abre-se a interface exibida na Figura 8.

Preencha os parametros:	
Numero de amostras na direcao X:	1000
Numero de amostras na direcao Z:	500
Dimensao da malha de velocidade (m):	4.
Comprimento do segmento do raio (m):	2.
Arquivo de parametros de inversao:	file_dados.txt
Intervalo (em graus) entre raios:	1.0
Arquivo de residuos:	residuo.txt
Abertura angular (em graus):	40.0
Tolerancia do residuo:	0.001
Numero de NOS do modelo (spline hz):	5
Minimo X para aquisicao:	100.
Maximo X para aquisicao:	3900.
Escalar para regular o tamanho do passo:	0.04
Inverter	
Sair	

Figure 8: Interface para inversão tomográfica.

- Numero de amostras na direcao X: Número de amostras do modelo de velocidade na direção horizontal.
- Numero de amostras na direcao Z: Número de amostras do modelo de velocidade na direção vertical.
- Dimensao da malha de velocidade (m): Dimensão da malha ou grid em metros.
- Comprimento do segmento do raio (m): Comprimento do segmento de raio em metros.

- Arquivo de parametros de inversao: Arquivo contendo informações do modelo inicial discretizado por splines, as coordenadas dos nós, as velocidades nos nós, o desvio máximo, uma lista dos arquivos com os tempos de trânsito dos difratores e as coordenadas dos difratores.
- Intervalo (em graus) entre raios: Informa-se aqui o ângulo entre raios consecutivos para o leque de raios.
- Arquivo de residuos: Arquivo contendo os resíduos em segundos para cada iteração.
- Abertura angular (em graus): Abertura angular em relação a normal. O leque de raios contemplará uma abertura correspondente ao dobro do valor informado.
- Tolerancia do residuo: Valor mínimo de resíduo em segundos para limitar as iterações.
- Numero de NOS do modelo (spline hz): Número de nós de splines do modelo.
- Minimo X para aquisicao: Valor mínimo em metros para registro de tempos no eixo X.
- Maximo X para aquisicao: Valor máximo em metros para registro de tempos no eixo X.
- Arquivo com horizonte mais profundo conhecido: Não empregado no momento (deixar como está).
- Modelo de velocidade de referencia: Não empregado no momento (deixar como está).
- Janela para suavizacao: Janela de suavização em amostras. Deixar em 1, por ora.
- Escalar para regular o tamanho do passo: Escalar para calibrar os passos da inversão.

Com os parâmetros indicados na Figura 8, ao clicar no botão *Inverter* o processo de inversão é iniciado. Para esta configuração, depois de 9 iterações o resíduo de tolerância é alcançado (0.001s). Assim, a partir de um modelo homogêneo, o processo de inversão tomográfica alcançou o modelo apresentado na Figura 9.

Ao clicar no botão *Sair* a interface se encerra.



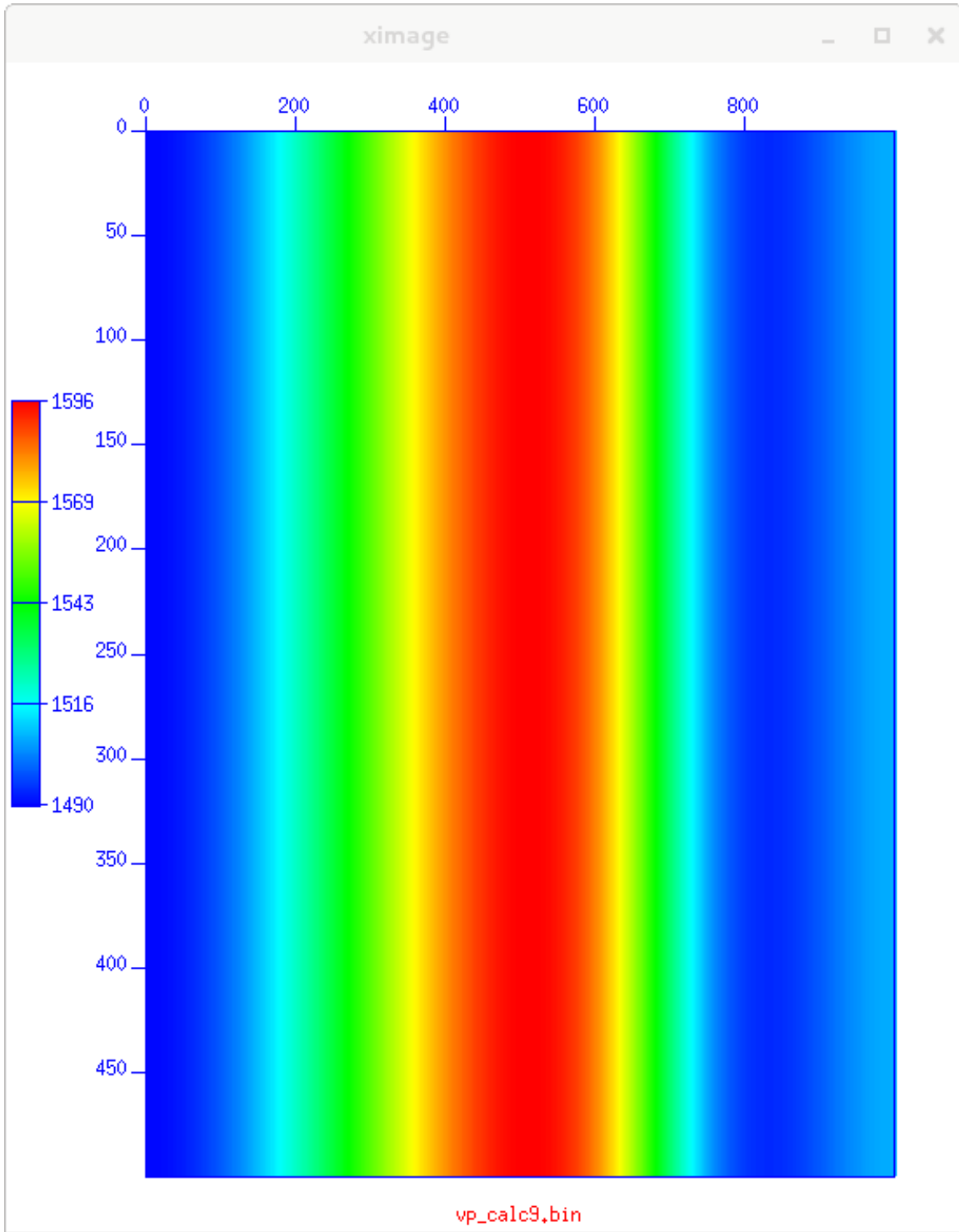
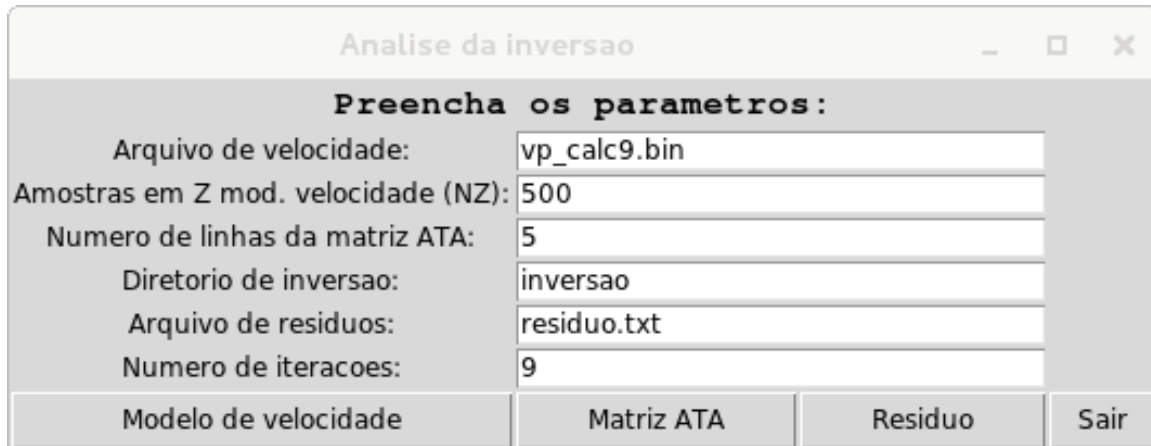


Figure 9: Modelo de velocidade calculado com inversão tomográfica não linear.

## Análise

Ao clicar no botão análise a interface observada na Figura 10 é exibida.



Preencha os parametros:			
Arquivo de velocidade:	vp_calc9.bin		
Amostras em Z mod. velocidade (NZ):	500		
Numero de linhas da matriz ATA:	5		
Diretorio de inversao:	inversao		
Arquivo de residuos:	residuo.txt		
Numero de iteracoes:	9		
Modelo de velocidade	Matriz ATA	Residuo	Sair

Figure 10: Interface de análise.

- Arquivo de velocidade: Arquivo de velocidade para observação. Qualquer um dos modelos de velocidade gerados nas iterações pode ser selecionado.
- Amostras em Z mod. velocidade (NZ): Número de amostras na vertical do modelo de velocidade.
- Numero de linhas da matriz ATA: Número de linhas da matriz ATA corresponde ao número de incógnitas do problema inverso. Neste exemplo ele corresponde ao número de nós de splines para caracterização do modelo de velocidade.
- Diretorio de inversao: Diretório onde se gravam os arquivos de inversão.
- Arquivo de residuos: Nome do arquivo onde se gravou os resíduos por iterações.
- Numero de iteracoes: Número de iterações total.

Com os parâmetros preenchidos na na Figura 10 e clicando-se no botão *Modelo de Velocidade*, aparecerá o modelo de velocidade da iteração 9, Figura 9. O modelo invertido

na nona iteração se assemelha ao modelo original, Figura 3.

Ao clicar no botão *Matriz ATA* aparecerá a matriz quadrada ATA exibida na Figura 11. Trata-se de uma matriz 5x5 pois temos 5 graus de liberdade correspondentes ao número de nós de splines que empregamos para descrever o modelo de velocidade.

Clicando no botão *Residuo* aparecerá a curva de resíduos a cada iteração, Figura 12. Note que havia espaço para mais iterações pois na nona iteração a curva não "pousou" ou atingiu um ponto de estabilidade.

Notando espaço para obtenção de melhor resultado na inversão tomográfica reduzimos a tolerância e rodamos nova inversão. Depois de 11 iterações observa-se que a convergência foi alcançada na curva resíduo por iteração - Figura 13. E por conta deste número adicional de passos, obtemos um modelo de velocidade mais próximo do original - Figura 14.

Finalmente, ao clicar no botão *Sair* a interface se encerra.

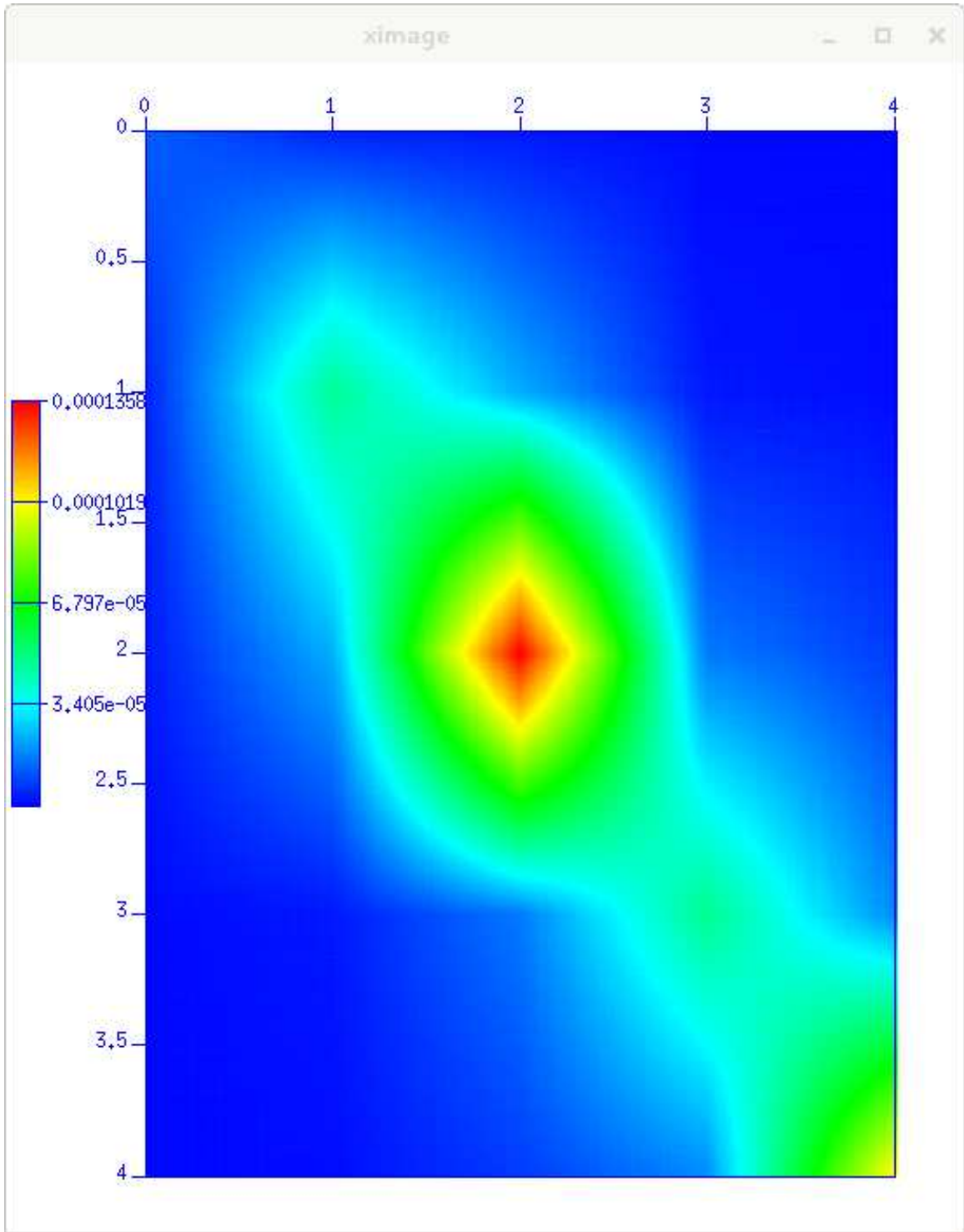


Figure 11: Matriz  $ATA$ .

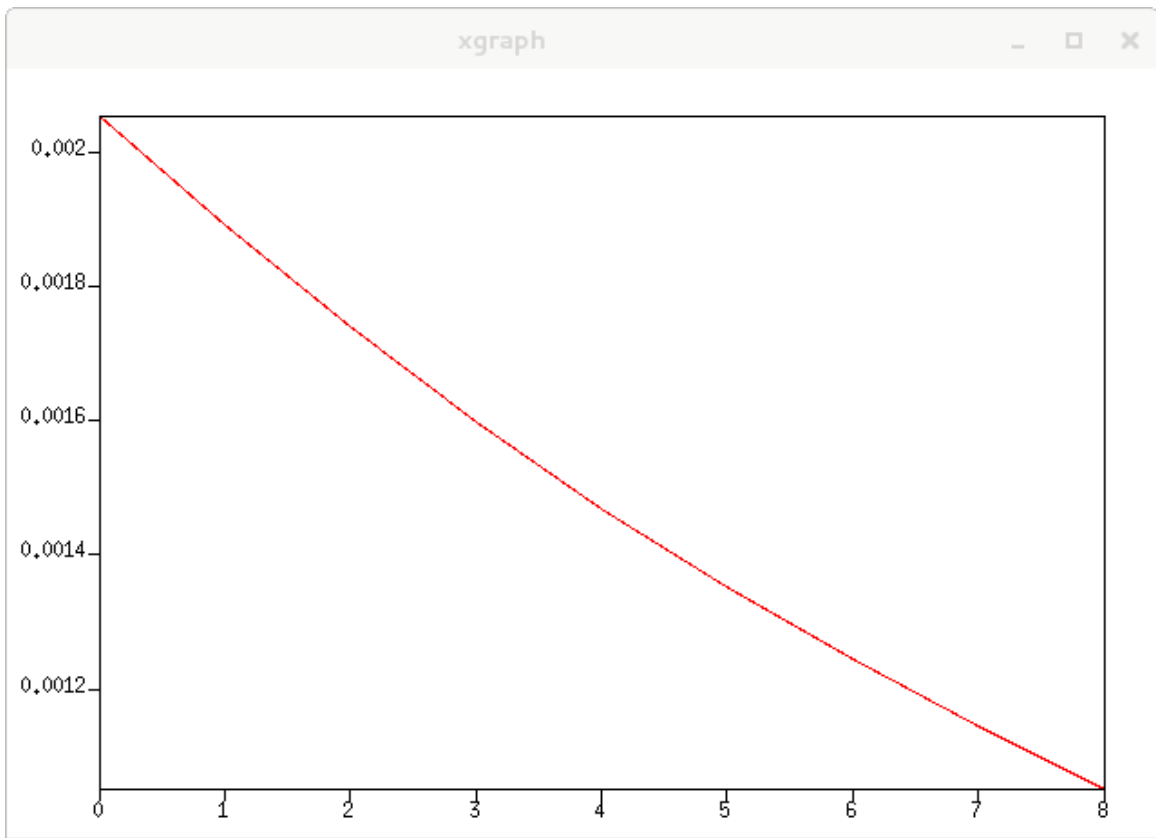


Figure 12: Curva de resíduo até a iteração 9.

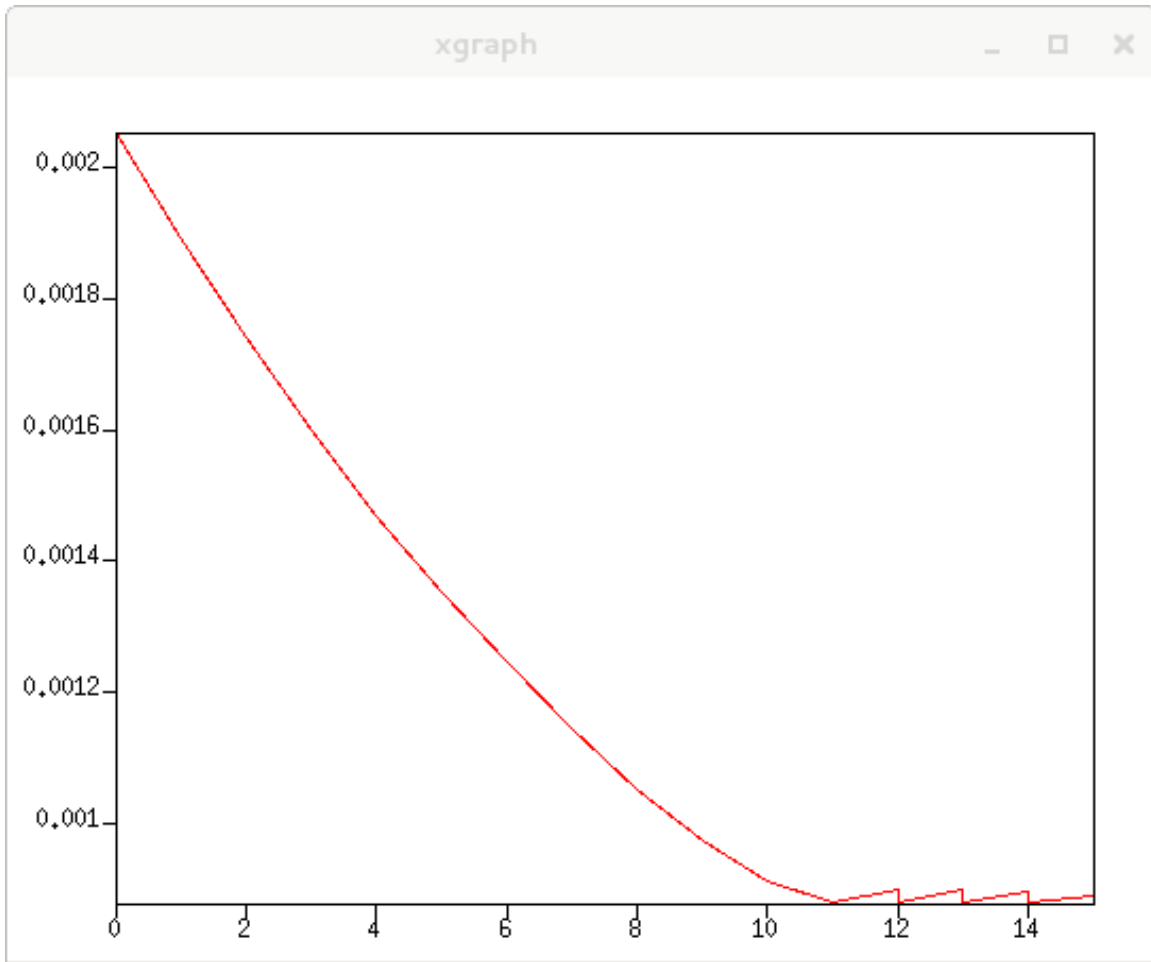


Figure 13: Curva de resíduo até a iteração 15.

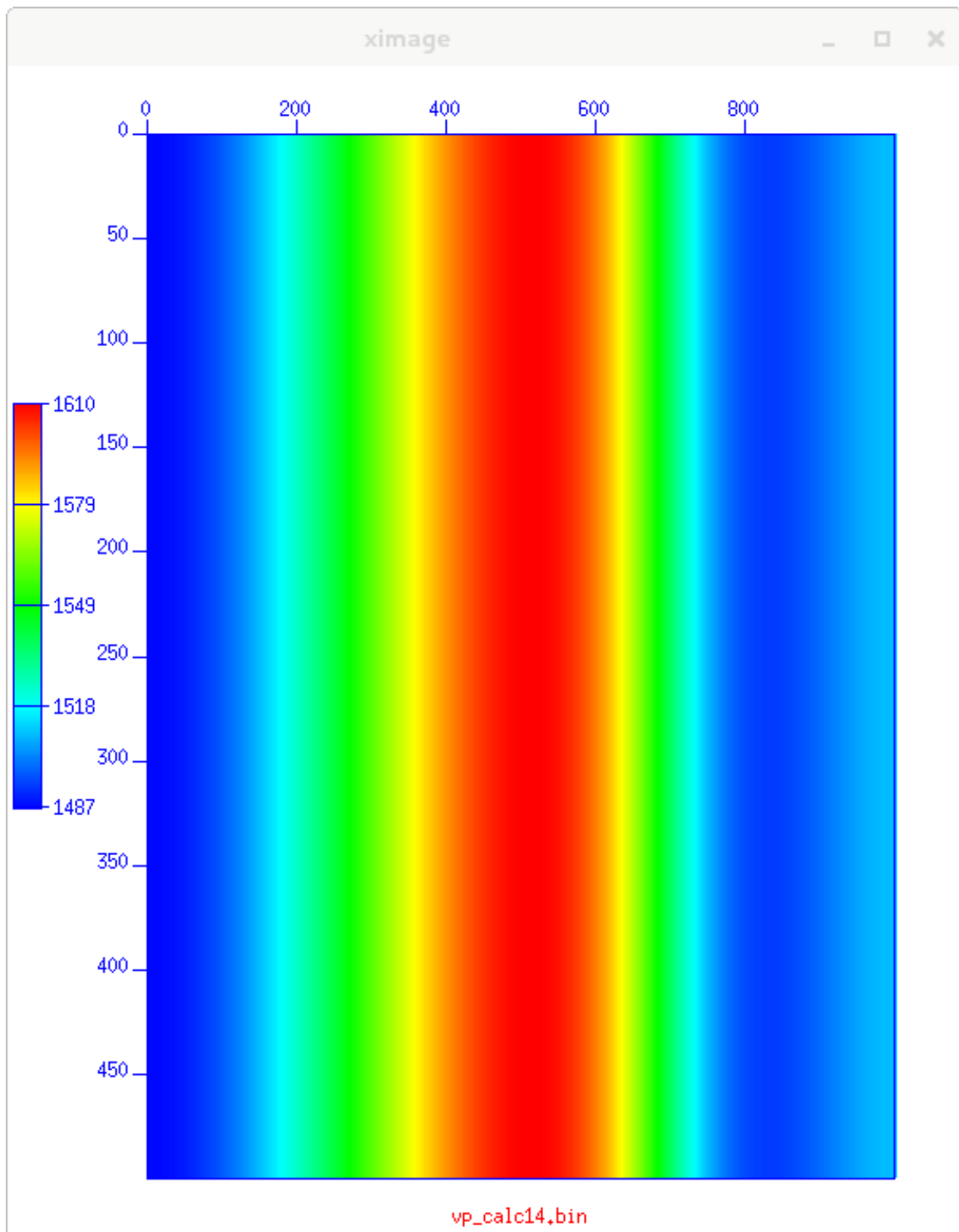


Figure 14: Modelo de velocidade calculado com inversão tomográfica não linear no passo

14.

## DISCUSSÃO E RECOMENDAÇÕES

O programa *geramodin1* tem sua aplicação voltada para finalidades acadêmicas e didáticas. O pacote pode ser baixado pelo site [faladaterra.com](http://faladaterra.com), aba Especialidades e o passo a passo está descrito no tópico Aplicações. Periodicamente estarão sendo carregadas novas versões onde serão incorporadas melhorias e sugestões dos usuários.

O sistema ora apresentado está sujeito às variações de distribuição do Linux, às versões do Python e dependente da biblioteca Tkinter. Temos rodado o *geramodin1* em distribuições Linux CENTOS, Fedora e Ubuntu e com versão Python 3.0. Eventualmente temos que adicionar manualmente a citada biblioteca Tkinter.

## CONCLUSÕES

O uso do *geramodin1* na disciplina Fundamentos de tomografia sísmica e em cursos da Pós Graduação no Departamento de Geologia e Geofísica da Universidade Federal Fluminense têm se mostrado como importante ferramenta didática e de desenvolvimento do corpo docente. Ferramentas práticas como esta coloca os alunos mais próximos do dia a dia de suas vidas profissionais acelera a absorção do conhecimento e promove o pensamento analítico e crítico necessário na resolução de problemas inversos nas geociências.

## AGRADECIMENTOS

Gostaríamos de agradecer aos colegas Eduardo Filpo pelas discussões e sugestões na seara tomográfica. E também, aos alunos da Universidade Federal Fluminense, motivadores deste projeto.



## REFERÊNCIAS

Stockwell, J.W., Cohen, J.K., 1998, The New SU Users Manual, Version 2.2, Seismic Unix Project, Center of Wave Phenomena Colorado School of Mines.