

# PyRefra – traitement et inversion de donn es de sismique r fraction

Hermann Zeyen<sup>1</sup>, Emmanuel L ger<sup>1</sup>  
<sup>1</sup>: Laboratoire GEOPS, Universit  Paris Saclay

Nous pr sentons un nouveau logiciel open-source multi-plateformes en langage Python pour l'affichage, le traitement et le picking de donn es de sismique r fraction 2D ainsi que de donn es VSP. Diff rents types d'affichage des temps mesur s sont propos s permettant un contr le de la qualit  des picks. Le package pyGimli est int gr  pour permettre une inversion tomographique

des temps d'arriv e donnant des mod les de vitesse. L'utilisation extensive en enseignement a montr  que la prise en main du logiciel est intuitive et suffisamment simple pour s'adapter   des enseignements de tous niveaux.

## 1 - Installation du logiciel et pr paration des donn es

- La GUI est bas e sur PyQt5
- En plus de biblioth ques standard en science, le logiciel utilise pyGimli (Ruecker et al., 2017) et obspy (Beyreuther et al, 2010) (Fig. 1)
- Le logiciel lit des donn es en format SEG2 ou SEG1
- Les informations de g om trie (position des tirs et g ophones) sont donn es dans des fichiers ext rieurs
- D'autres fichiers permettent la correction d'erreurs d'acquisition communs (mauvais temps de d clenchement, erreurs dans la num rotation des tirs ou des g ophones,  limination de g ophones bruit s/hors fonctionnement...)

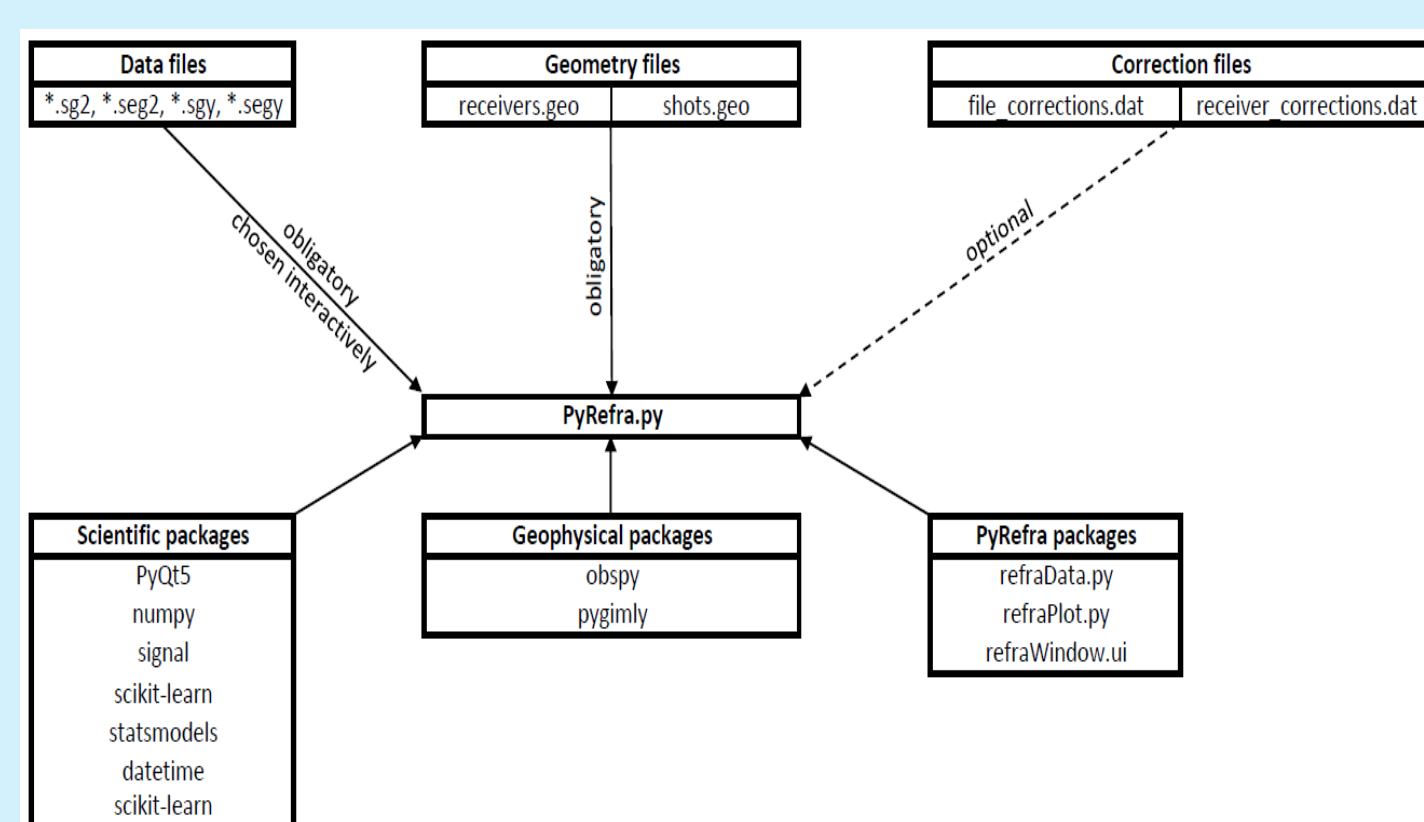


Figure 1: Biblioth ques et fichiers n cessaires pour PyRefra

File	Display	Utilities	Picking	Filter	Mute
Choose data files	Original data this screen	P-model	Manual picks	Frequency	Mute/recover trace
Save SEG1	Original data all shots	Tomography	Amplitude picks	Frequency filter for single trace	Mute air wave
save SU	Shot gather	Envelopes	STA-LTA	Air wave fk	Mute before line
save SEG2	Plot data of one file	False color	Correlation picks	Velocity filter	Mute after line
save Binary	Receiver gather	Change trace sign	Move picks		
save ASCII	Distance gather	Change sign	Uncertainty change		
save Headers	Choose component	Change colors tomo	Erase all picks		
save plot	Phase angles	Wave animation	Plot all picks		
Quit	Zoom	Attenuation	Plot calculated times		
	Zoom out	Pseudo-velocities	Store picks		
	Zoom in		Store Gimli format		
	Zoom initial				
	Trace normalize				
	Time gain				
	Distance gain				
	AGC				

Figure 2: Menus disponibles

## 2 – Fonctionnement g n ral

- L'utilisation de PyRefra est bas e sur des menus (Fig. 2)
- Tous les fichiers d'une ligne d'acquisition sont lus ensemble et il est possible de les acc der ind pendamment.
- L'affichage (Fig. 3) peut se faire par tir, par fichier, par g ophone ou par offset. En cas d'enregistrement avec des g ophones 3C, chaque composante peut s'afficher s par ment.
- L'affichage de traces peut se faire en normalisant chaque trace, avec un gain de distance, de temps ou AGC (automatic gain control)
- Diff rentes options de filtrage (fr quence, vitesse) ainsi que de muting sont pr vues

## 3 - Picking

- Le picking peut se faire manuellement en cliquant la position de la premi re arriv e et son incertitude (Fig. 4).
- Diff rentes options de picking (semi)automatique sont   disposition.
- La m thode semi-automatique pr f r e se base sur la corr lation crois e entre les traces et une trace ma tre. Elle utilise en plus les informations de mesures faites sur des tirs   proximit    offset  gal. La comparaison avec des temps mesur s manuellement est encourageante (Fig. 5).
- Suite au picking automatique, il est possible de bouger les picks, de changer leur incertitude ou de les effacer.
- Diff rentes possibilit s de contr le de qualit  des picks sont disponibles: L'affichage par offset permet la d tection de mesures aberrantes, surtout pour des pairs de picks o  les positions de tir et capteur sont invers s. L'affichage de pseudo-sections pour les vitesses moyenn es pour les diff rentes combinaisons tir-capteur et la repr sentation des lenteurs locales (inverse de la vitesse entre deux capteurs voisins) permettent la d tection de probl mes avec des tirs ou capteurs (Fig. 6). Finalement, l'affichage de tous les temps mesur s sur un graphique aussi permet la d tection d'erreurs (Fig. 7).

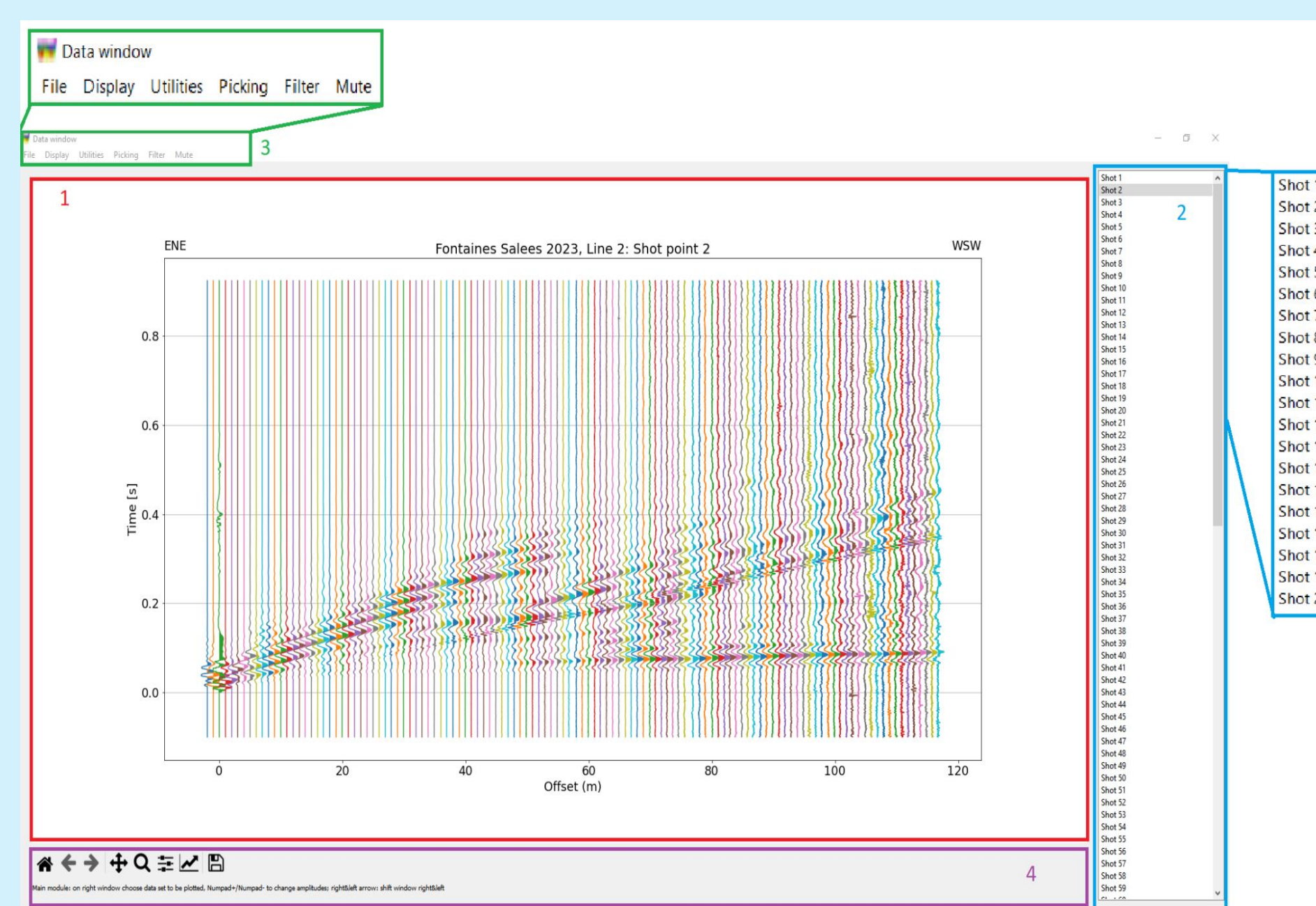


Figure 3: Aspect g n ral de la fen tre principale Cadre rouge: Affichage des donn es; cadre bleu: liste des fichiers/tirs/capteurs/distances disponibles; cadre vert: menus; cadre violet: Aide et outils PyQt.

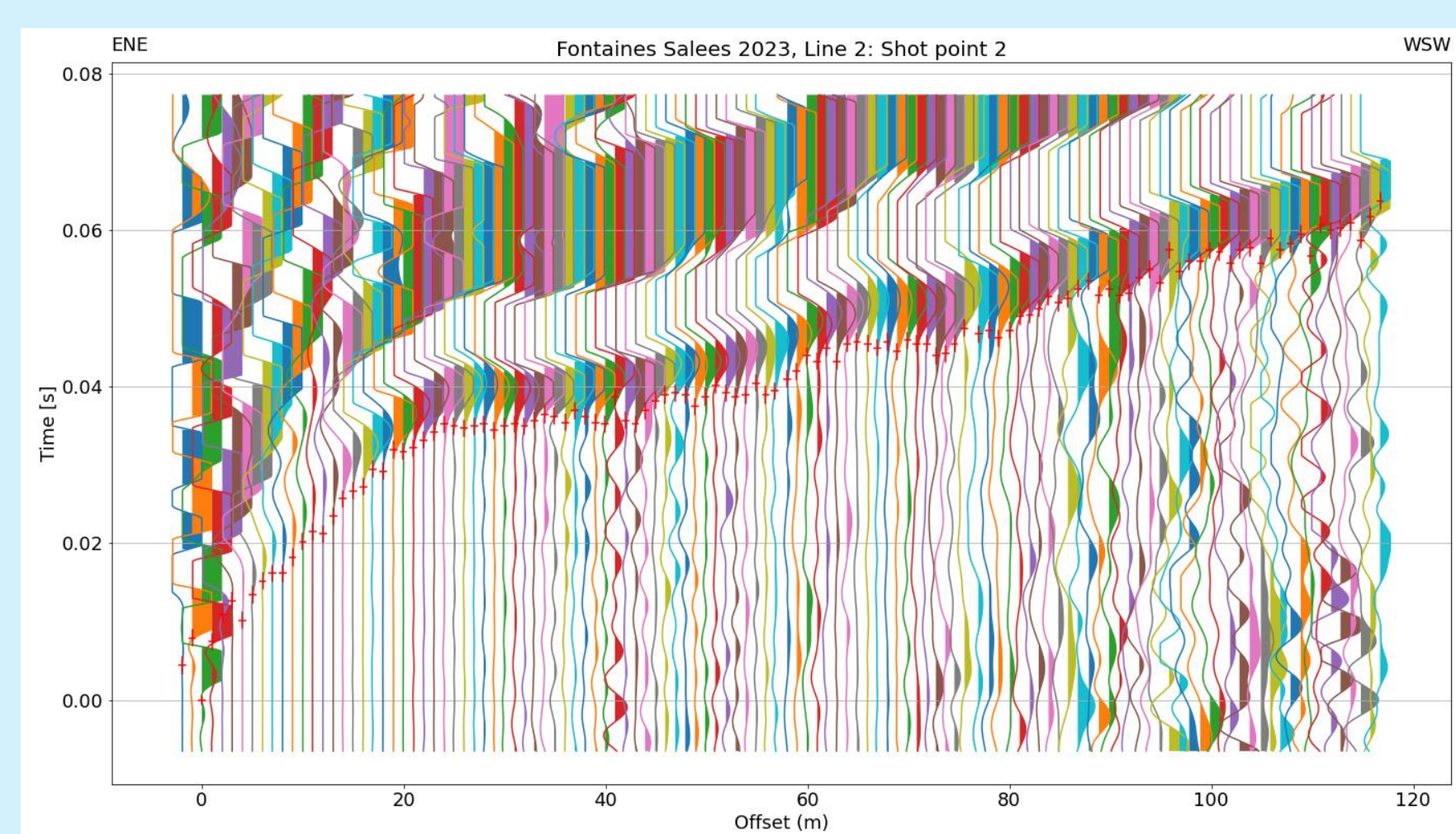


Figure 4: Exemple de picks avec leurs marges d'incertitude

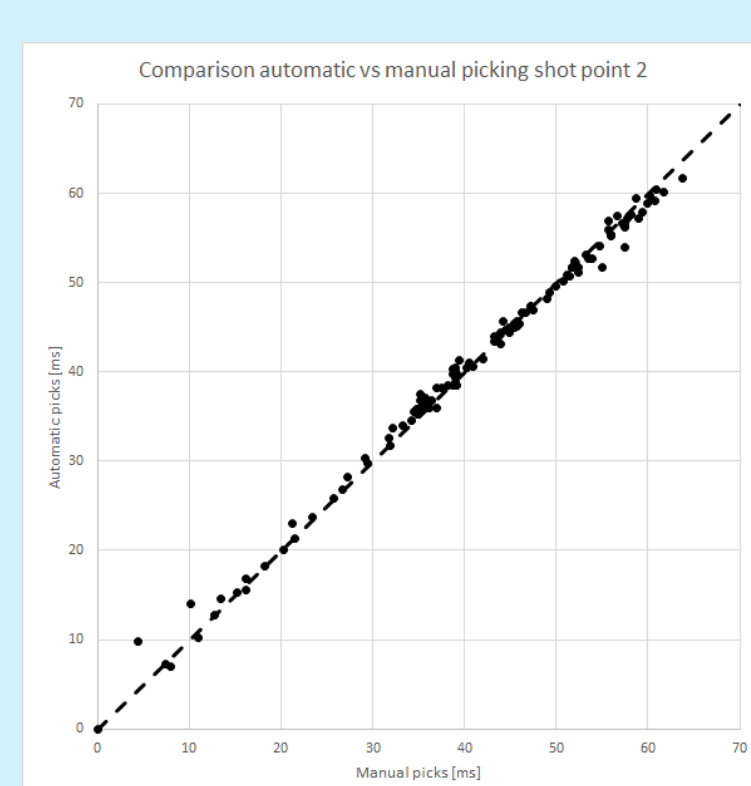


Figure 5: Comparaison de picks automatiques et manuels faits sur le tir montr  dans la figure 4 (offsets de 0   70m).

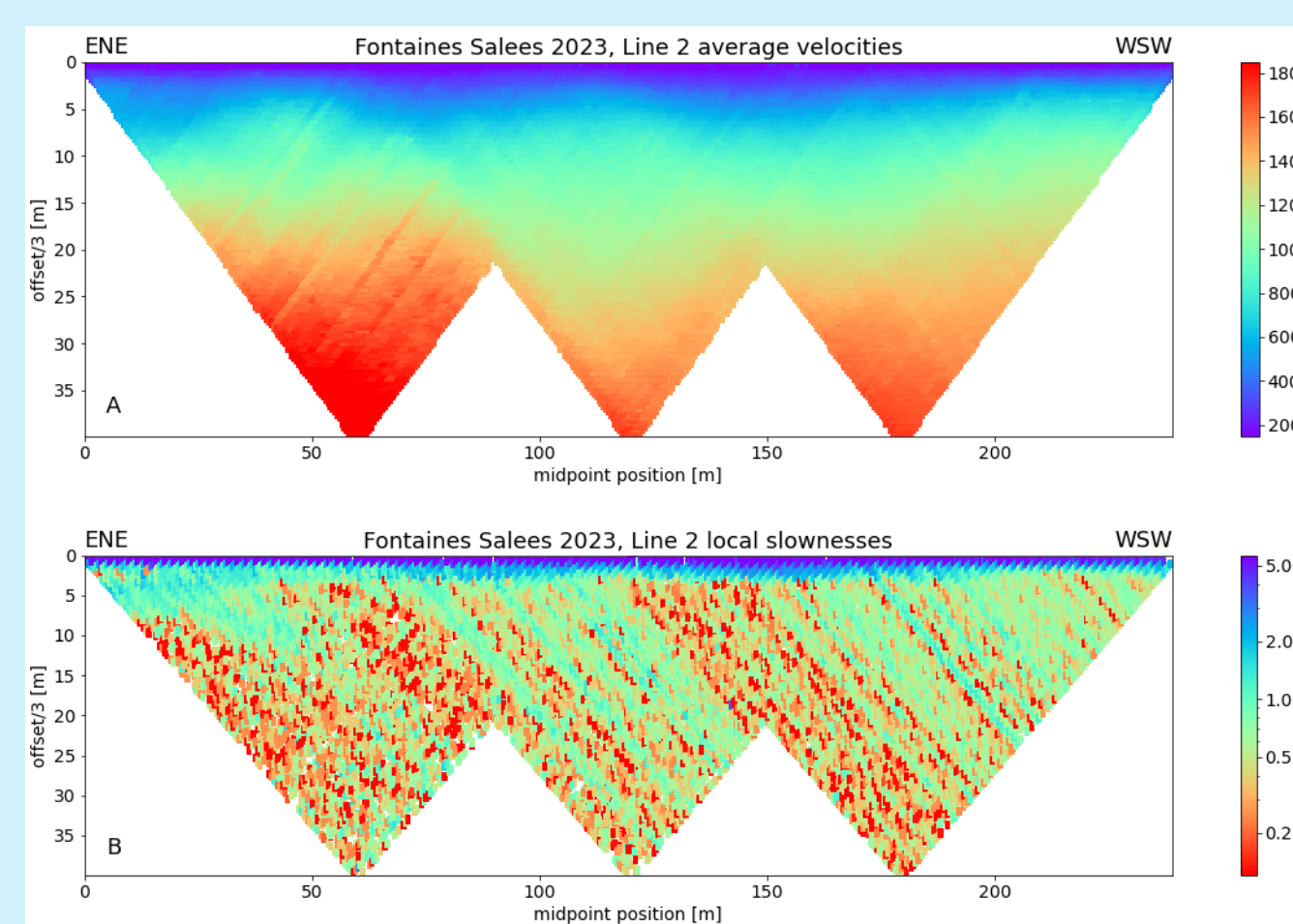


Figure 6: Exemple de pseudo-sections de vitesse moyenne et lenteur locale

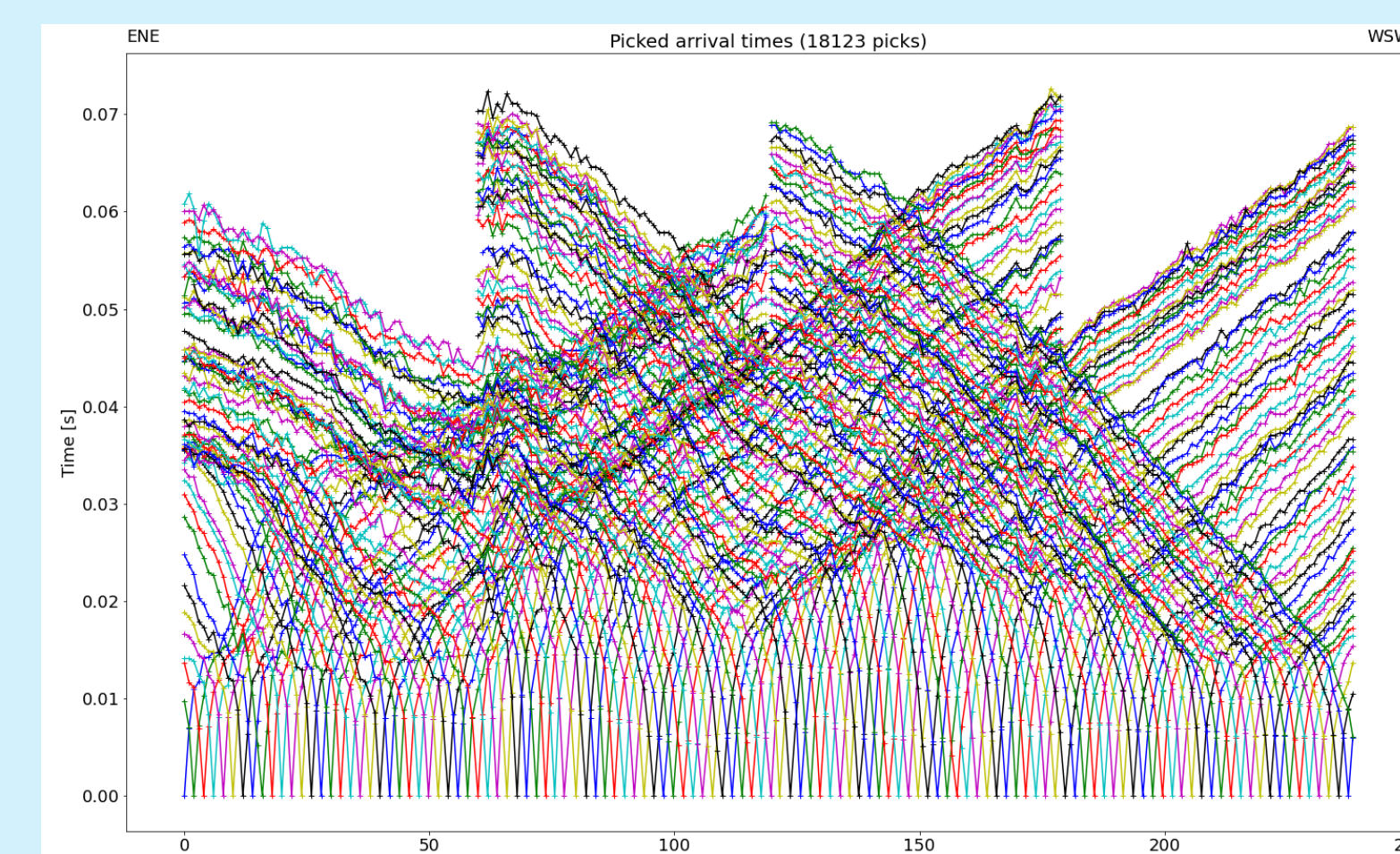


Figure 7: Affichage des temps d'arriv e mesur s   l'origine des pseudo-sections de la figure 6.

## 4 – Inversion tomographique des temps d'arriv e

Une fois tous les temps d'arriv e mesur s, il est possible de lancer une inversion tomographique. Pour cela, le logiciel utilise les fonctions de pyGimli. L'utilisateur a la possibilit  de d finir un mod le 1D initial simple (vitesses en haut et en bas du mod le, gradient constant) ainsi que diff rents param tres de contr le de l'inversion (param tres de lissage, nombre d'it rations). Les it rations se terminent si le  $\chi^2$  du misfit passe en-dessous de 1, si le nombre limite d'it rations est d pass  ou si entre deux it rations la r duction du misfit devient trop petite. Le r sultat est affich  comme section des vitesses o  les vitesses ne sont marqu es qu'o  des rais sont pass s lors des diff rentes it rations. En plus diff rents graphiques donnant des indications de la qualit  de l'ajustement sont pr sent s (Fig. 8).

## Conclusion - Perspectives

- Logiciel facile   prendre en main, apte pour l'enseignement
- Affichages vari s offerts
- Diff rentes m thodes de picking permettant des mesures rapides (les >18000 picks de la figure 7 se sont faits en 3 jours)
- Importance donn e aux possibilit s de contr le de qualit  des temps mesur s
- Inversion tomographique int gr e
- Perspectives: Int gration du traitement et de l'inversion 3D

## R f rences

Beyreuther, M., R. Barsch, L. Krischer, T. Megies, Y. Behr, and J. Wassermann, 2010 – ObsPy: A Python Toolbox for Seismology. Seismological Research Letters 81(3), 530-533.  
Ruecker, C., T. G nther and F.M. Wagner, 2017 – pyGIMLI: An open-source library for modelling and inversion in geophysics. Computers & Geosciences 109, 106-123. doi: 10.1016/j.cageo.2017.07.011  
Zeyen, H. and L ger, E., 2023 – PyRefra – Refraction seismic data treatment and inversion. Comp.&Geosci. submitted

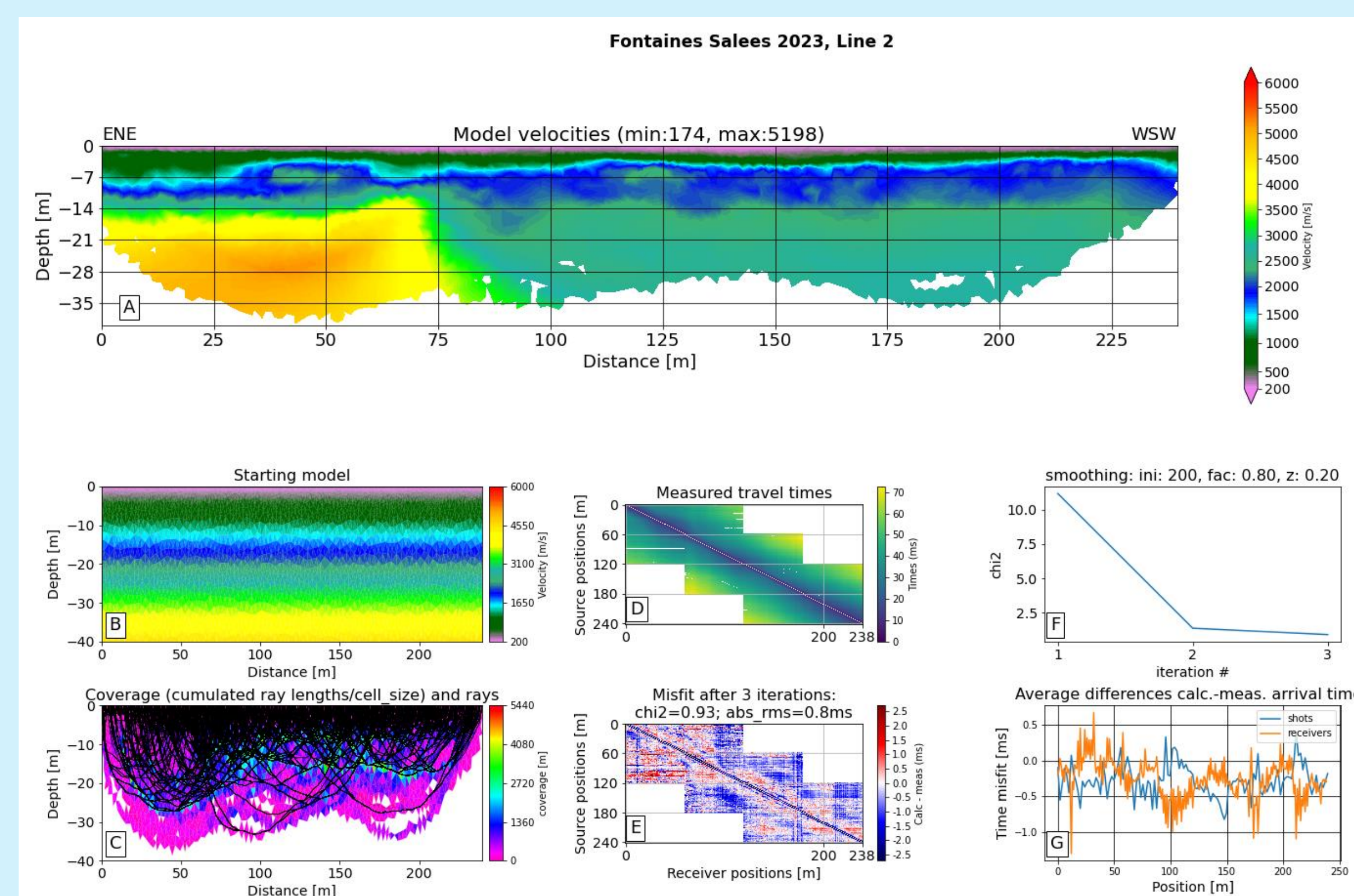


Figure 8: Mod le tomographique r sultant de l'inversion des donn es montr es dans la Fig. 7. A: mod le final des vitesses; B: Mod le initial; C : Mesure de la r solution (longueur somm e des rais traversant chaque cellule) et distribution des rais du mod le final (lignes noires). D : Temps d'arriv e mesur s comme fonction des positions des r cepteurs (axe X) et des sources (axe Y). E : Erreur d'ajustement entre les temps calcul s et mesur s. F : Evolution de la mesure  $\chi^2$  des erreurs d'ajustement en fonction du num ro de pas d'it ration. G : Erreur moyen pour tous les tirs et r cepteurs en fonction de leur position, permettant de localiser d'eventuels probl mes avec l'un de ces points.

## Remerciements

Nous remercions les  tudiants de plusieurs promotions de Licence et Master ST d'avoir test  le logiciel et d couvert des bugs et instabilit s.