

# OPTIMISATION DE LA RECONSTRUCTION DE TRACES DANS DUNE EN PYTHON

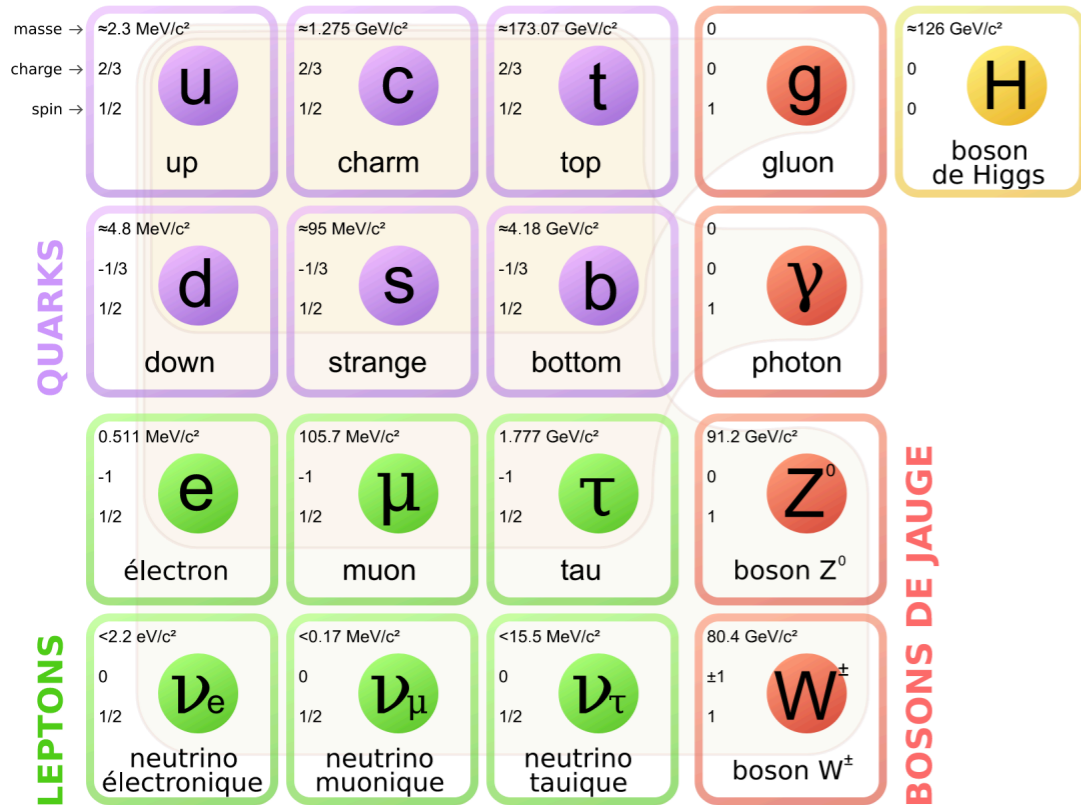


LARDON

*Liquid Argon Reconstruction Done in  
pythON*

*Laura Zambelli (LAPP)  
Gray Scott Reloaded School  
July 11<sup>th</sup> 2024 - Annecy*

# Les oscillations de neutrinos



- La matière est constituée d'un assemblage de constituants élémentaires.
- Les interactions entre ces constituants sont décrites par le Modèle Standard.
- Parmi les 17 particules élémentaires, il existe trois types ('saveurs') de neutrinos :  $\nu_e$ ,  $\nu_\mu$  et  $\nu_\tau$ .

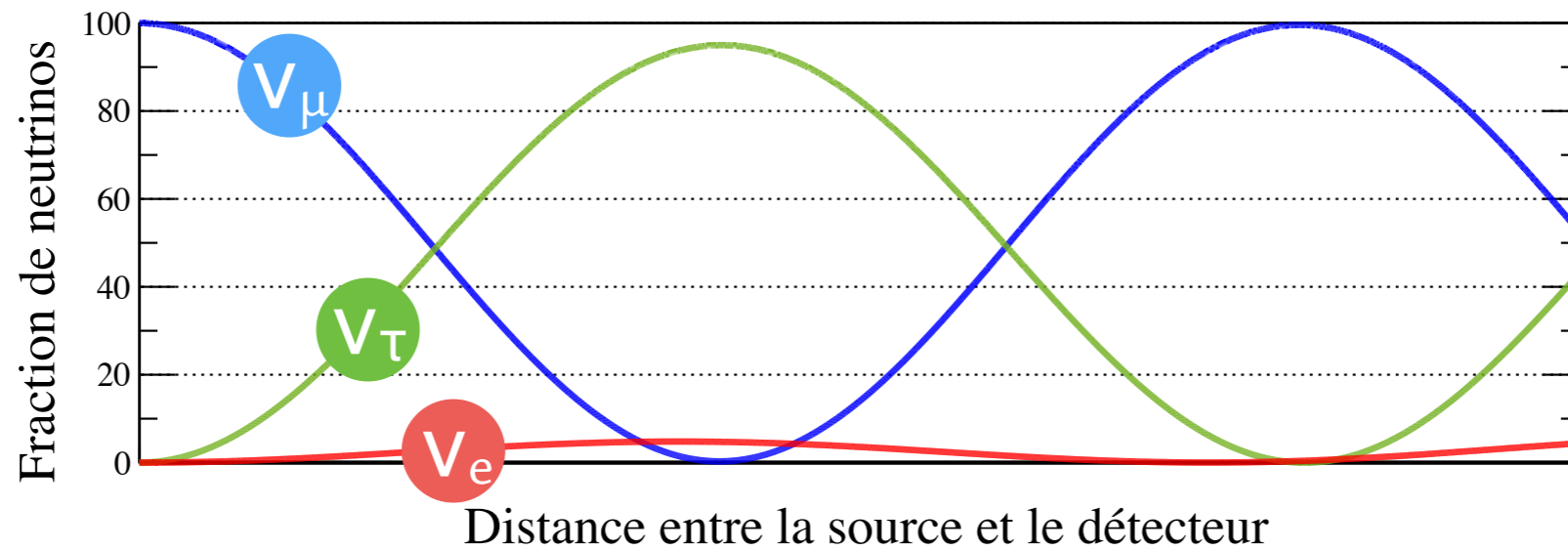
Les neutrinos :

- N'ont pas de charge électrique
- Ont une masse très faible (encore inconnue)
- Interagissent très peu avec la matière

# Les oscillations de neutrinos

Dans les années 2000, en étudiant les neutrinos émis par diverses sources (réacteur nucléaire, soleil, rayons cosmiques) il a été découvert que les neutrinos **oscillent**:

-> Un neutrino créé selon une certaine saveur peut être détecté en étant d'une saveur différente



La probabilité qu'un neutrino oscille ('change de saveur') dépend:

- De l'énergie du neutrino
- De la distance parcourue entre la source et le détecteur

Théoriquement, les oscillations de neutrinos sont décrites par un modèle possédant 6 paramètres, dont 5 sont aujourd'hui connus avec une plus ou moins grande précision.

Comprendre les mécanismes régissant les oscillations de neutrinos aidera à comprendre l'Univers, et en particulier l'origine de l'asymétrie matière - antimatière.

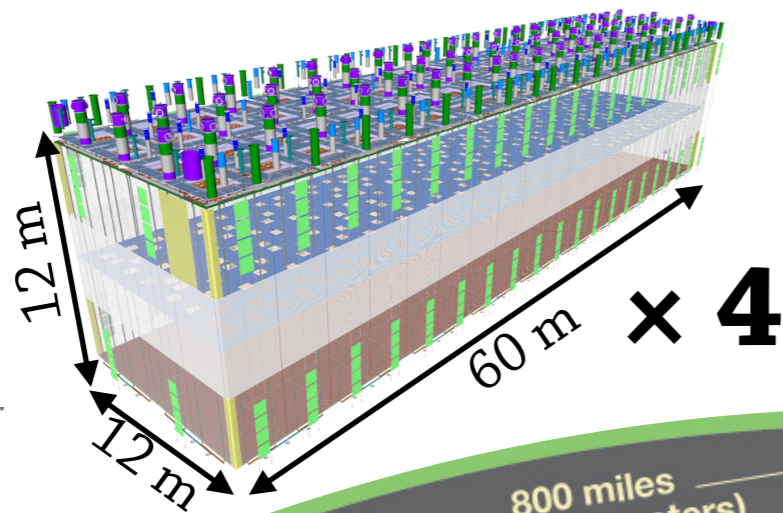
# L'expérience DUNE

DUNE ('Deep Underground Neutrino Experiment') est une expérience en cours de construction aux États-Unis qui vise à déterminer avec une grande précision les paramètres régissant les oscillations de neutrinos.

## Principe :

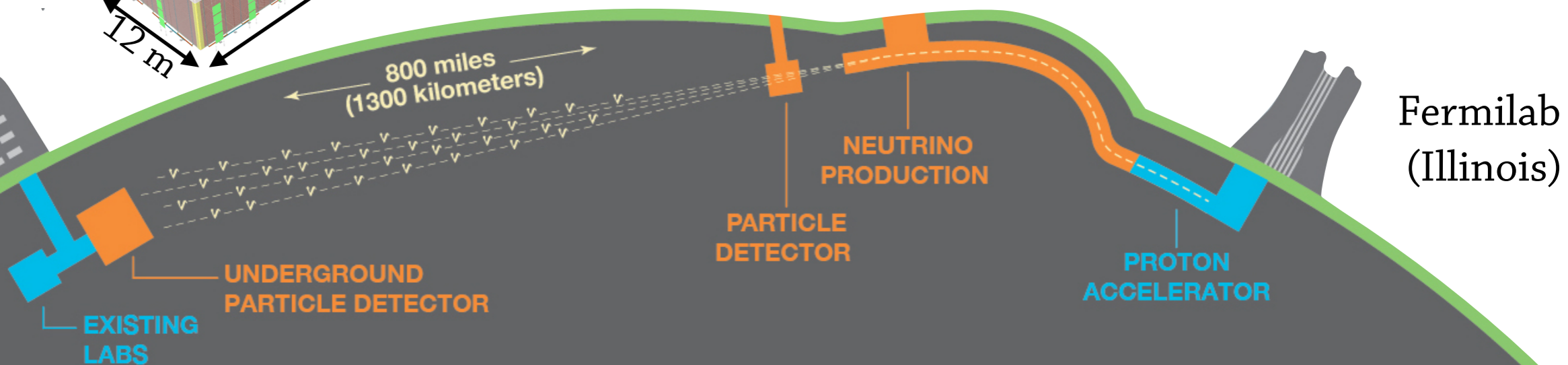
Au Fermilab, près de Chicago, un faisceau de neutrino de type  $\nu_\mu$  est créé à une énergie connue grâce à un accélérateur de proton.

Dans une mine du Dakota du Sud, 1300 km du Fermilab, un détecteur à 1.5km de profondeur observe le faisceau de neutrino et mesure la fraction de neutrino de chaque saveur.



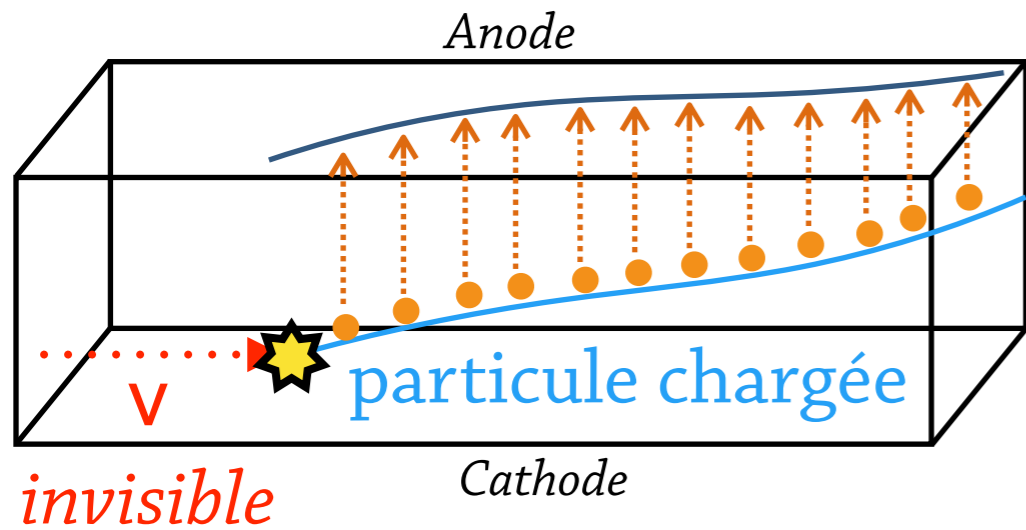
Ce détecteur est constitué de 4 modules géants utilisant la technologie de chambre à projection temporelle avec de l'argon liquide.

SURF  
(Dakota  
du Sud)



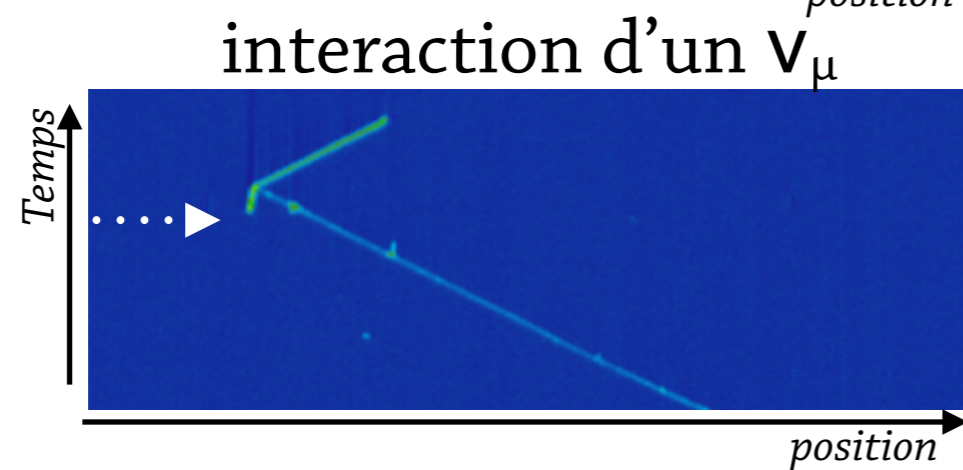
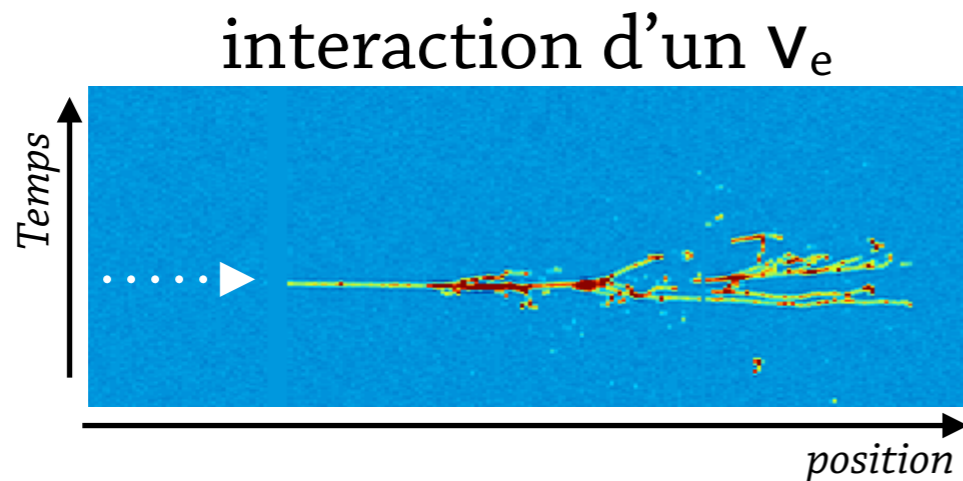
Fermilab  
(Illinois)

# Les Chambres à Projection Temporelles (TPC)



1. Les neutrinos interagissent avec les atomes d'argon ce qui génère des particules chargées
2. Lorsqu'une particule chargée traverse de la matière, elle arrache des électrons au milieu
3. Grâce à un champ électrique, ces électrons sont dirigés vers le haut du détecteur (anode)
4. Les électrons sont collectés sur l'anode, et forment des images 2D. Le temps d'arrivée des électrons donne la 3<sup>ème</sup> dimension.

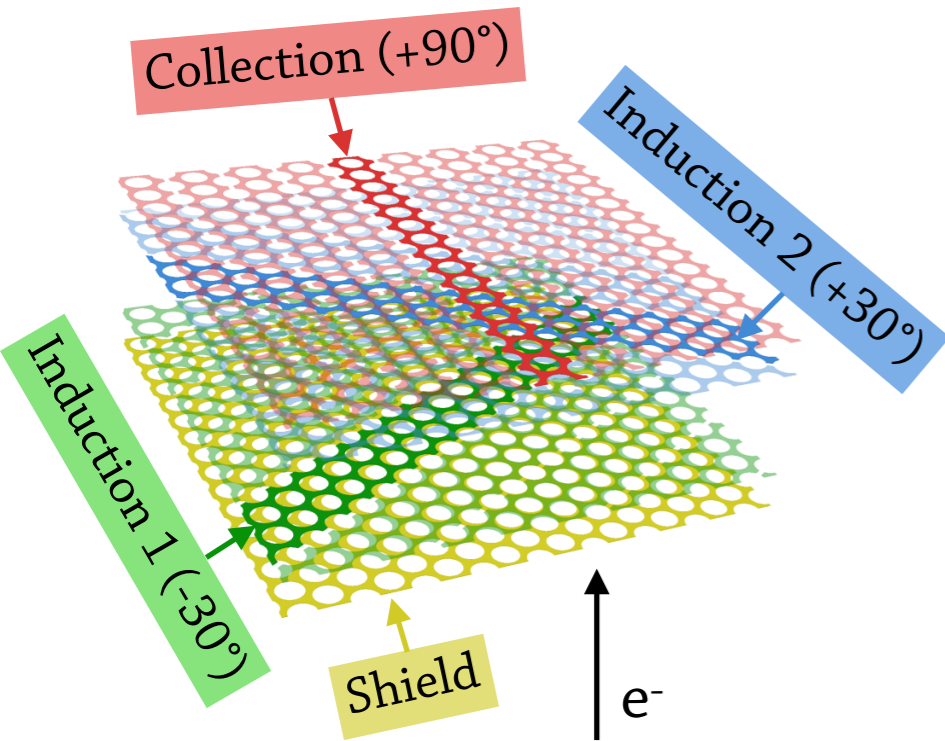
Images 2D obtenues dans le détecteur



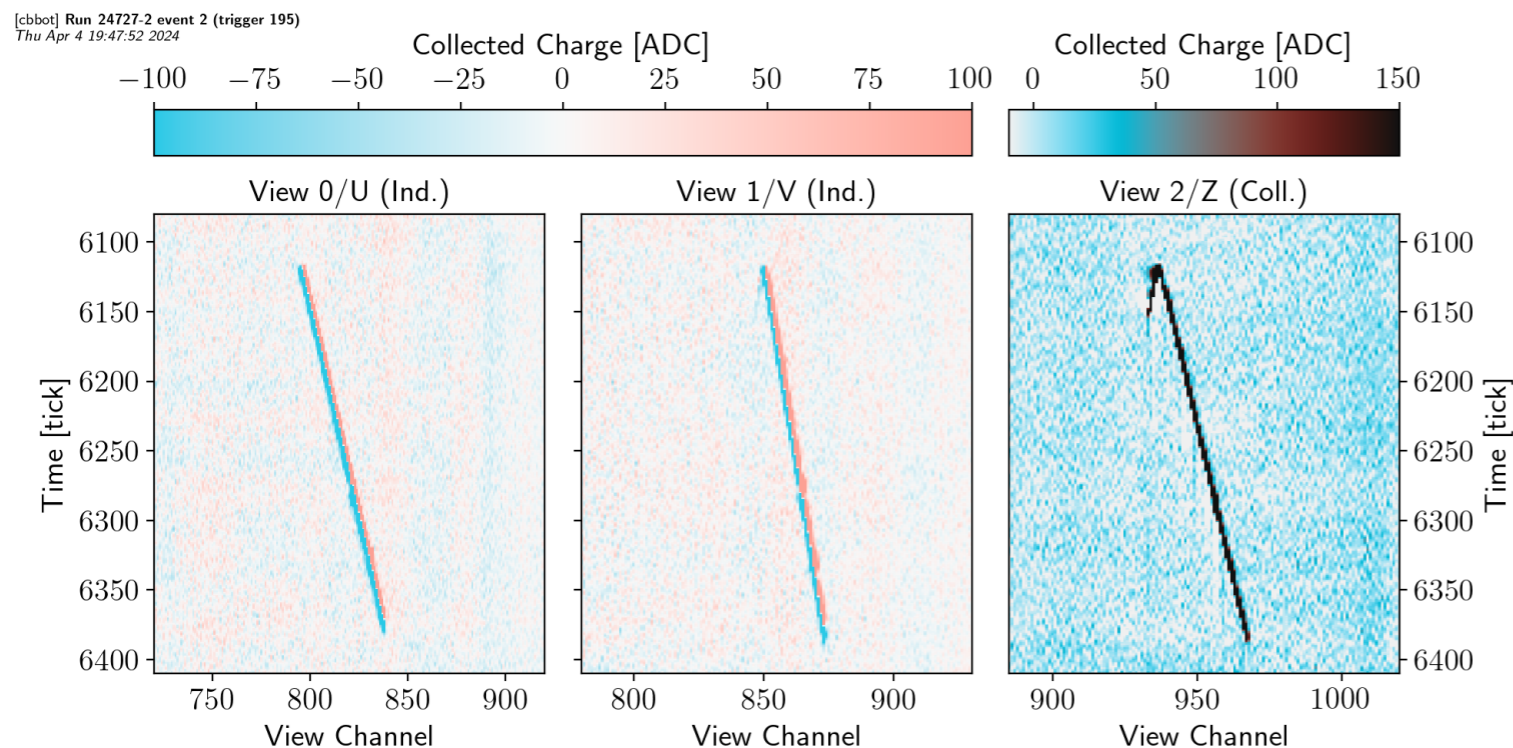
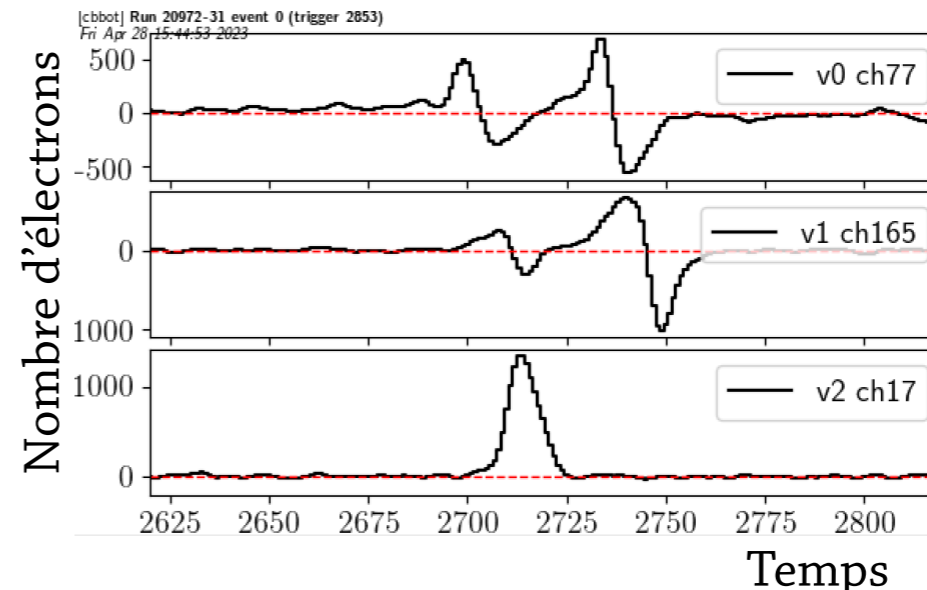
Cette technologie permet de distinguer les interactions dues à un  $\nu_e$  et à un  $\nu_\mu$

# Les signaux recueillis

Les anodes sont faites de trois plans de lectures successifs (ou 'vues') équipés de canaux de lecture avec des orientations différentes  $\leftrightarrow$  3 images 2D vues avec trois angles différents



Le nombre d'électrons digitisés (en 'ADC') vus par chaque canal est enregistré toutes les 512 ns pendant 4 ms.



**La reconstruction des événements de DUNE cherche à retrouver la position en 3D de chaque particule chargée qui a traversé le détecteur**

# La reconstruction des événements

---

La collaboration DUNE utilise le software LArSoft pour reconstruire ses données:

- Software commun à toutes les expériences utilisant des TPCs à Argon Liquide
- Écrit en C++ avec une interface en ROOT
- LArSoft a des interfaces avec d'autres software, par exemple utilisant du machine learning pour la reconnaissance de traces
- LArSoft est très difficile à comprendre, et encore plus à maîtriser

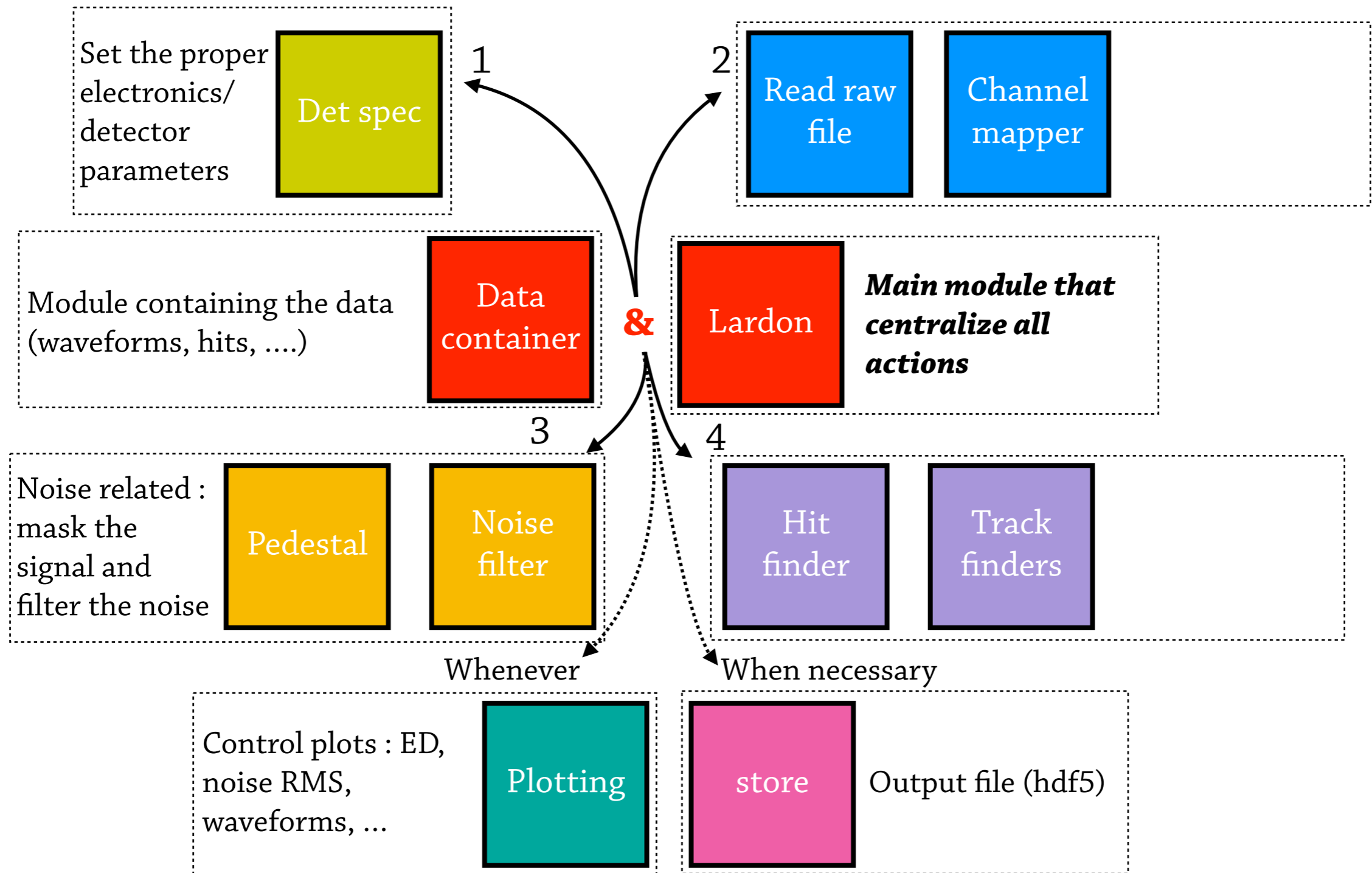
Le software LARDON (Liquid Argon Reconstruction Done in pythON):

- Est né d'un défi personnel d'apprendre le python
- L'idée d'origine était de seulement arriver à décoder et les visualiser les données
- Grâce au confinement, LARDON peut désormais faire la reconstruire en 3D des traces en un temps largement inférieur à LArSoft avec des résultats similaires

Le code est public (mais pas encore les données) : <https://github.com/dune-lardon/lardon>

# Principe de la reconstruction LARDON

## Principe de LARDON





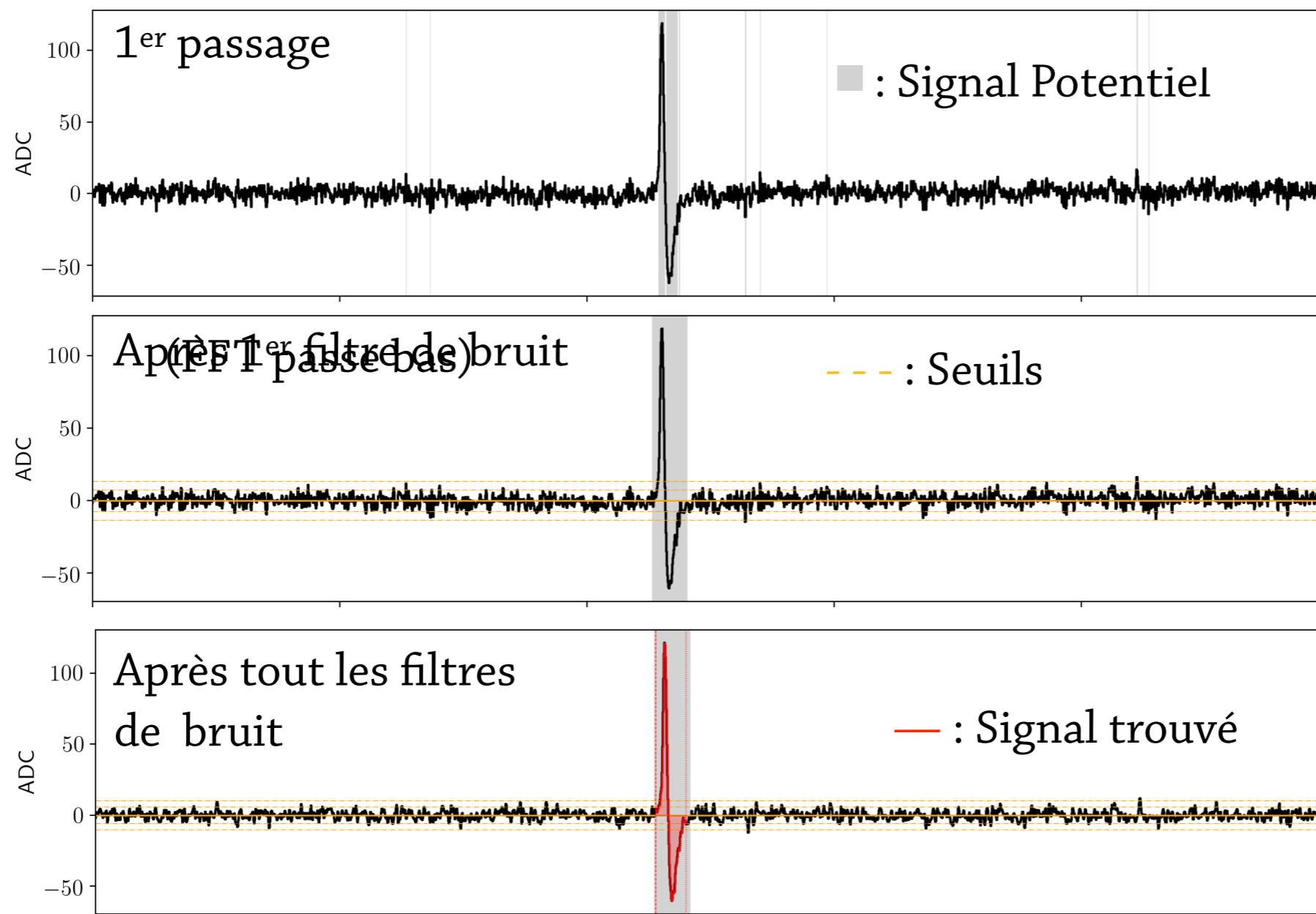
# Filter le bruit

Les données sont d'abord décodées depuis un fichier binaire

Pour chaque canal, une distinction grossière bruit/signal est faite pour chaque point de donnée:

- d'abord basée sur un seuil fixe (ex:  $|ADC| > 10$ )
- puis par rapport à un seuil basé sur le RMS du bruit

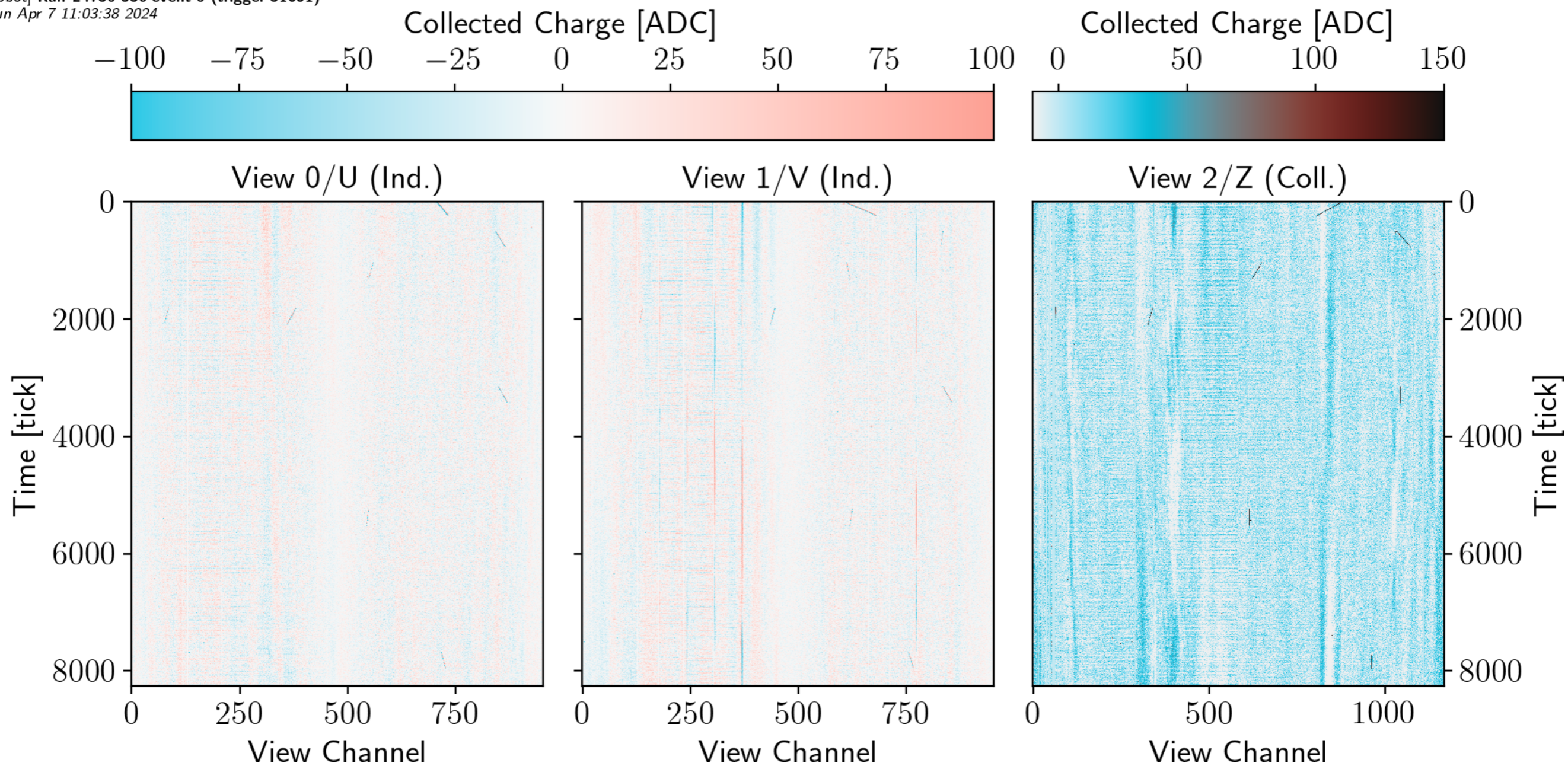
Plusieurs filtres sont appliqués aux données pour réduire le bruit et mieux distinguer le signal



Ces méthodes  
sont accélérées  
par numba

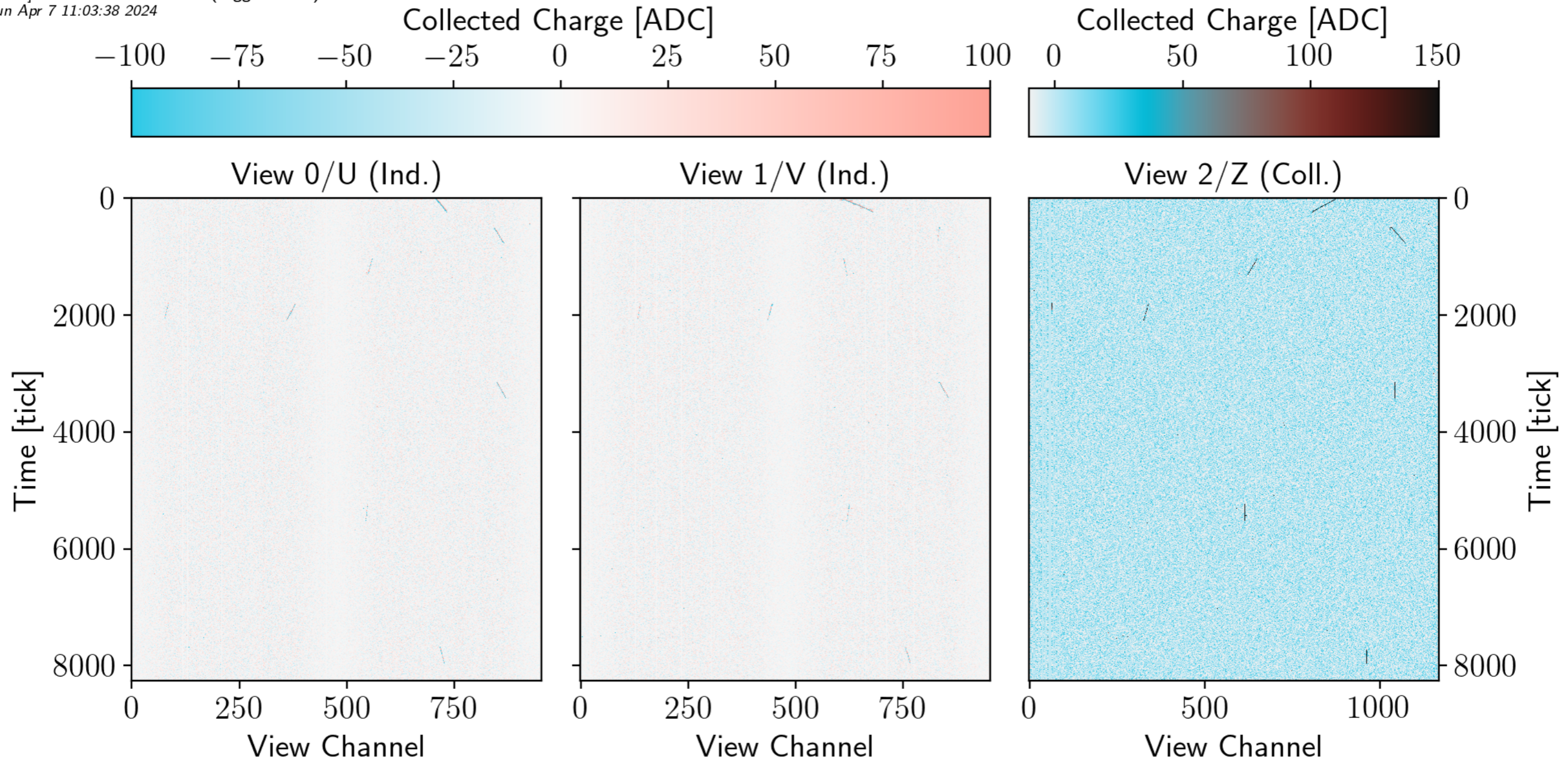
# Example - événement brut

[cbbot] Run 24730-330 event 0 (trigger 31681)  
Sun Apr 7 11:03:38 2024



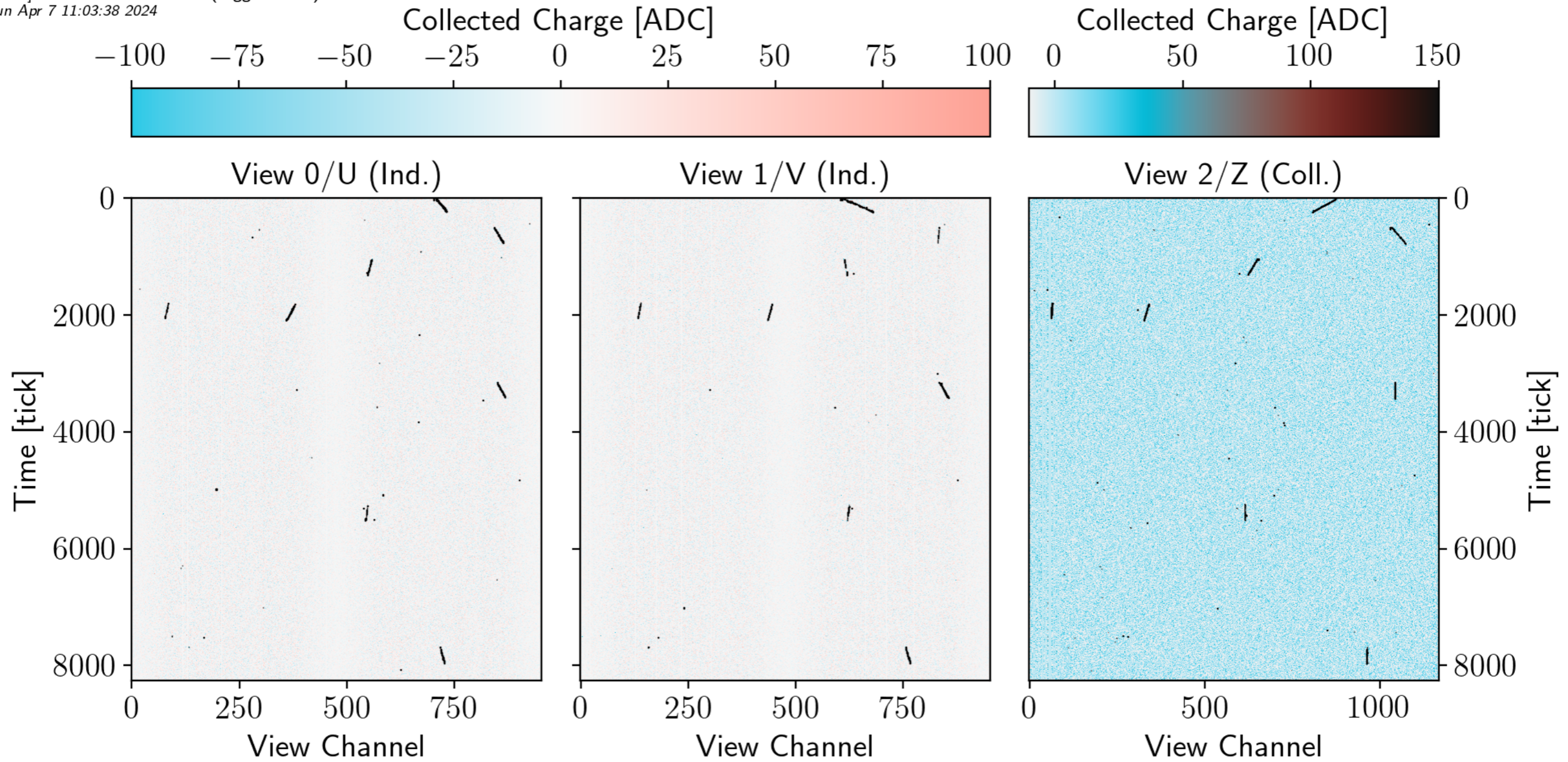
# Example - après le filtrage du bruit

[cbbot] Run 24730-330 event 0 (trigger 31681)  
Sun Apr 7 11:03:38 2024



# Example - Signal trouvé

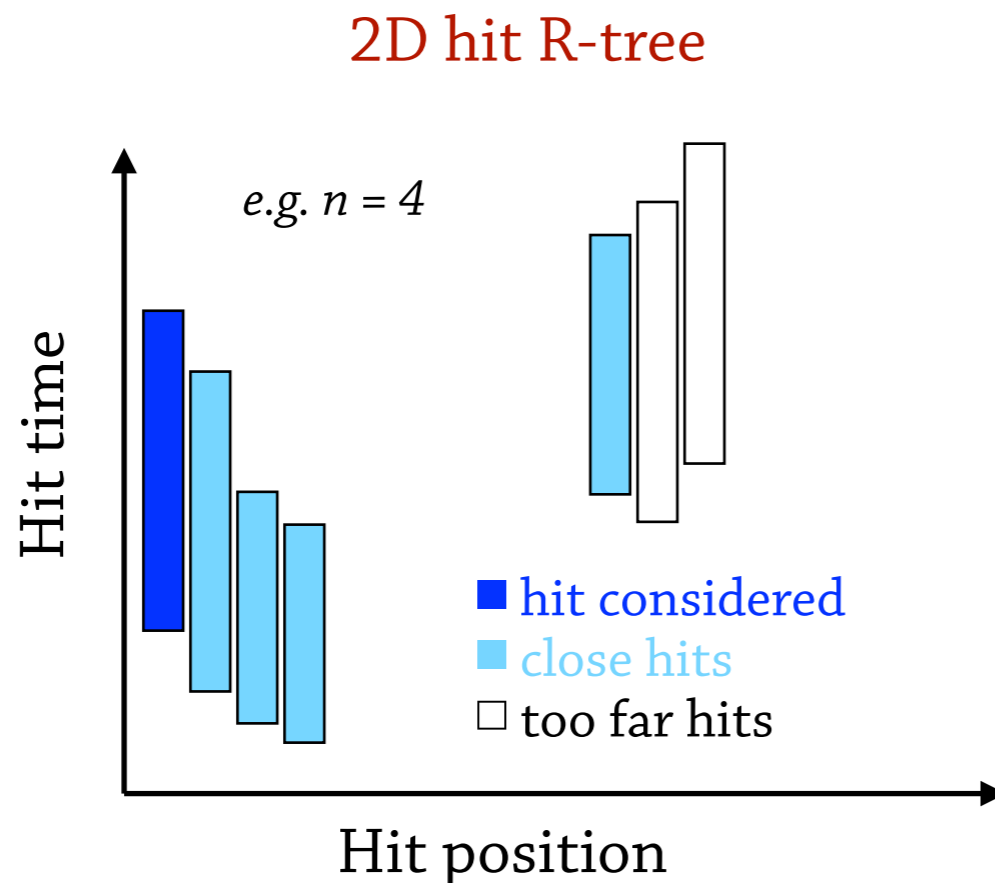
[cbbot] Run 24730-330 event 0 (trigger 31681)  
Sun Apr 7 11:03:38 2024



# Reconstruction en 2D

Les signaux générés par une trace seront physiquement proches en temps et en espace dans chaque vues. La première étape de la reconstruction est de regrouper ces signaux ensemble.

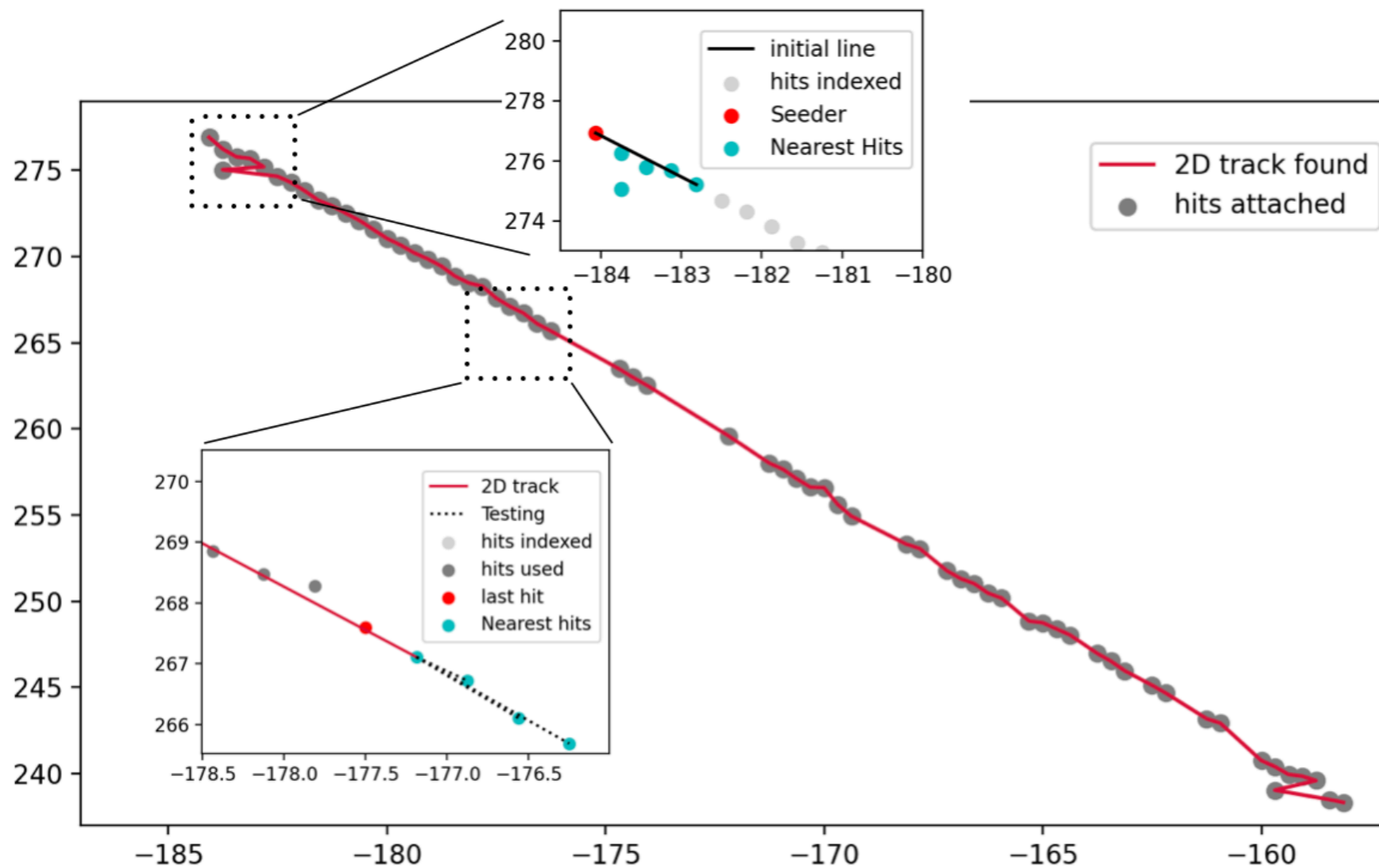
Les signaux (ou hits) trouvés sont indexés dans un Rectangular-tree (R-tree) en 2 dimensions: leurs positions (via le canal où ils ont été trouvés) et leur temps d'arrivée. Un R-tree est généré pour chaque vue.



Pour un hit donné, le R-tree retourne les  $n$  autres hits les plus proche.

# Reconstruction en 2D

Avec la collection de hits proche, les traces en 2D sont reconstruites grâce à un algorithme de semblable à un filtre de Kalman:



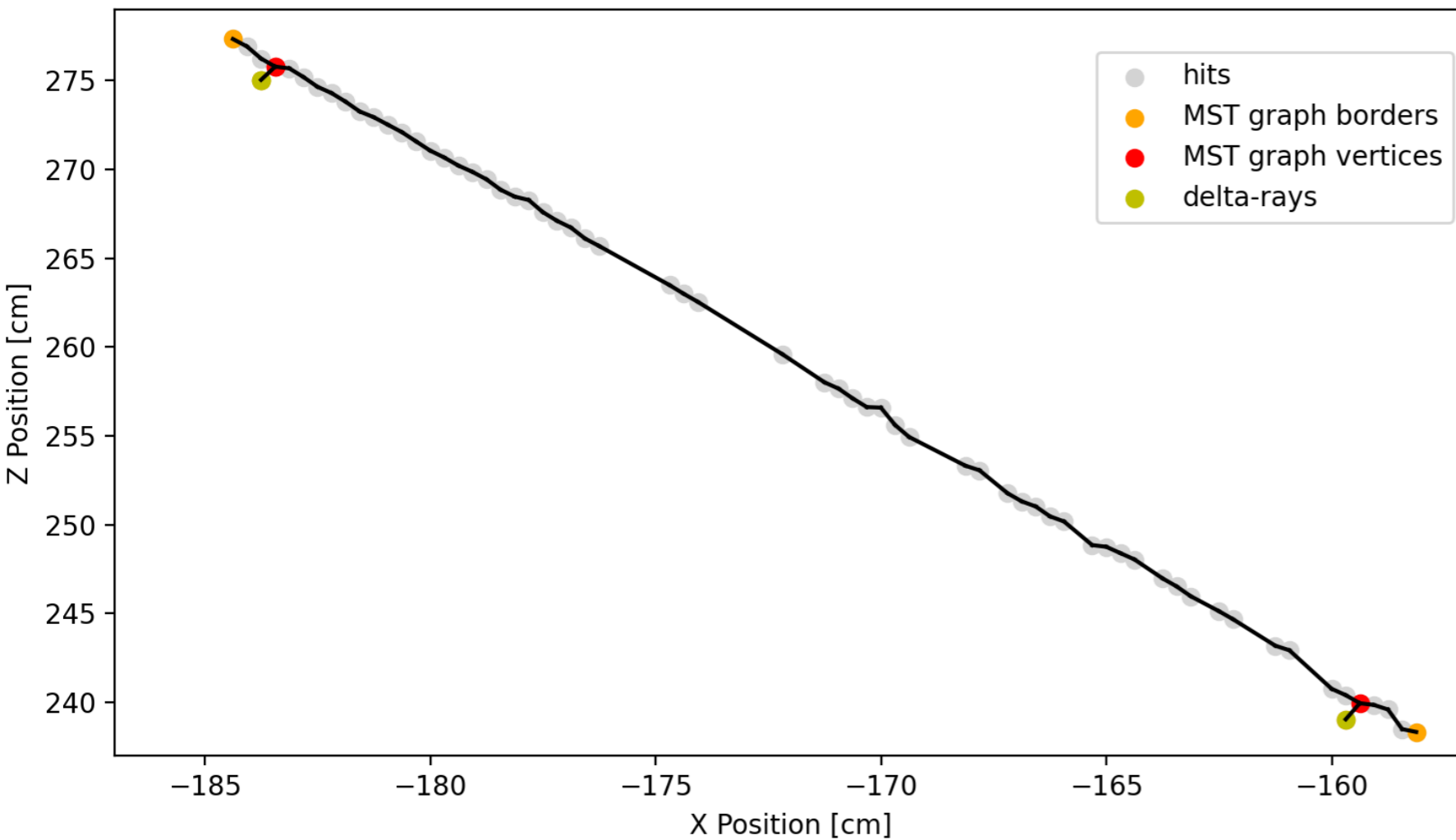
La trace est construite de par itération où chaque hit est testé puis ajouté à la trace, ce qui modifie les paramètres de la trace.

-> autorise des traces légèrement courbées, contrairement à approche plus stricte comme un ajustement linéaire

# Reconstruction en 2D

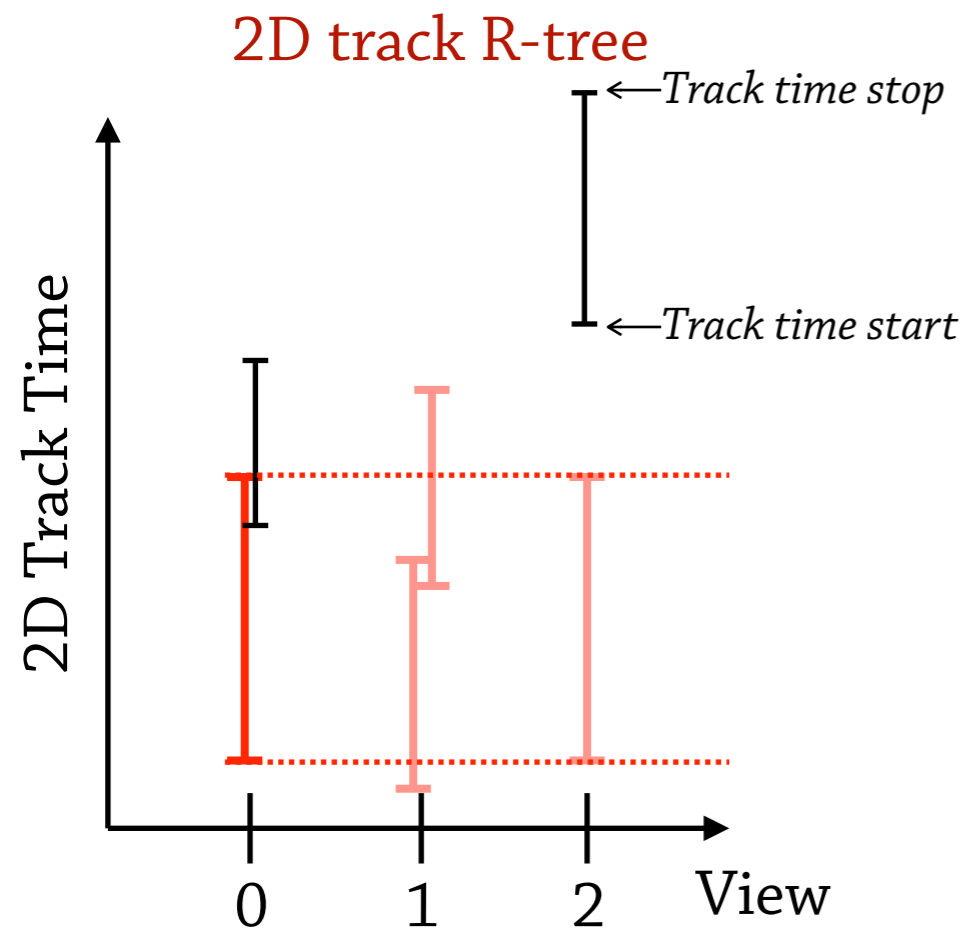
Une fois la trace reconstruite, les hits la constituant sont ré-ordonnés par un graphe MST (Minimum Spanning Tree) d'où on peut en extraire:

- les points extrêmes de la trace
- les hits n'appartenant pas complètement à la trace (les 'delta-rays')



# Reconstruction en 3D

Une fois les traces trouvées dans chaque vue, il faut les assembler afin de déterminer les coordonnées 3D de chaque hit. Une même trace sera vue au même moment dans chaque des vues.



Toutes les traces trouvées sont indexées dans un nouvel R-tree à deux dimensions: la vue dans laquelle elles ont été reconstruites, et leurs étalage dans le temps

Pour **une trace dans une vue**, le R-tree donne toutes **les intersections de traces dans les autres vues**.

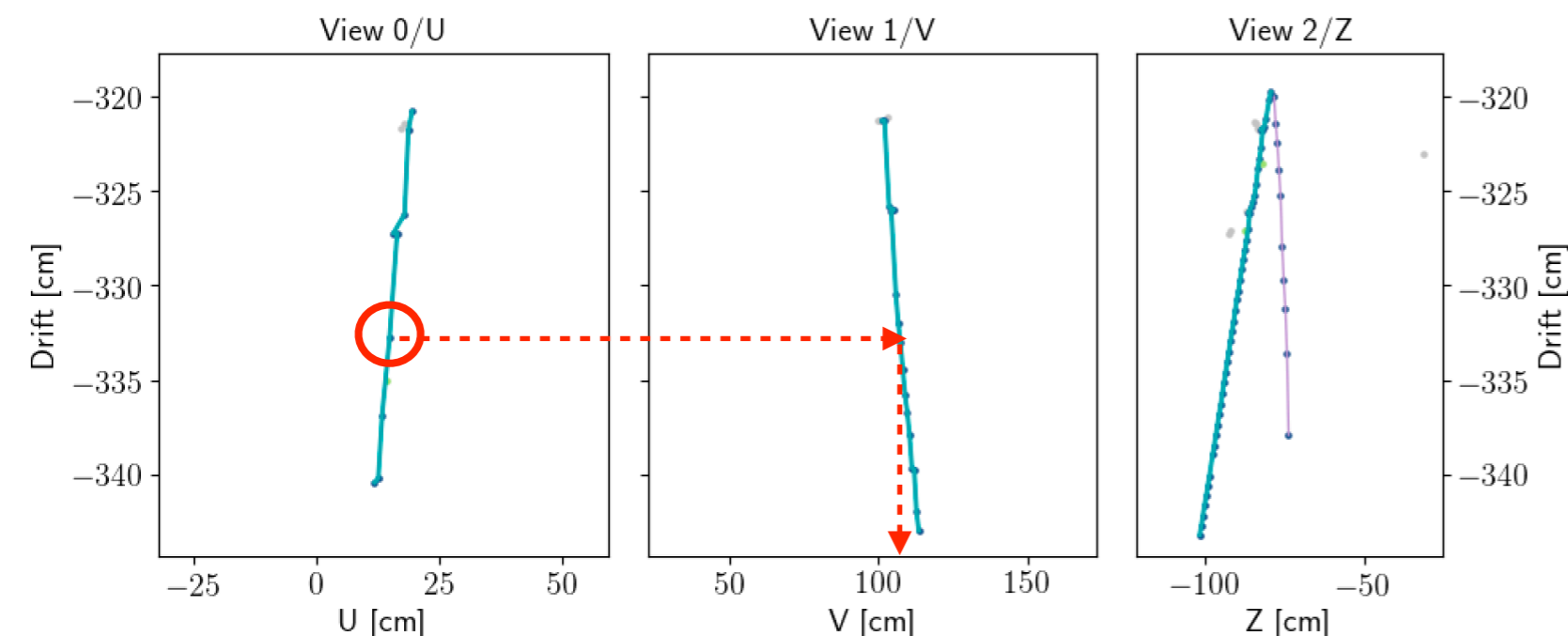
Dans le cas d'ambiguïtés, d'autres critères comme la somme du nombre d'électrons entre en compte pour déterminer qui est le meilleur match



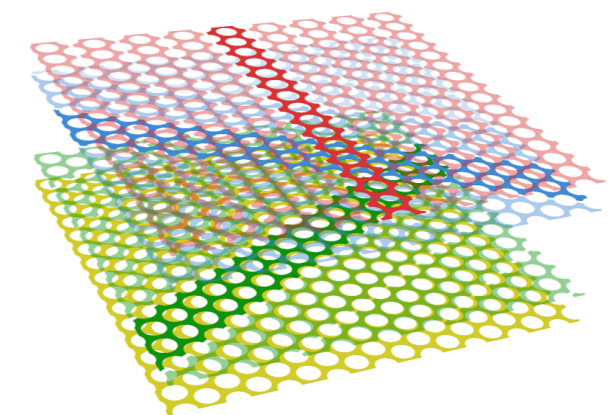
# Reconstruction en 3D

Avec l'assemblage des traces dans chacune des vues, la coordonnées en 3D de chaque hit est déterminée.

La coordonnées manquante de chaque hit de la vue  $i$  est calculée par la position interpolée de la trace sur la vue  $j$  au même temps.

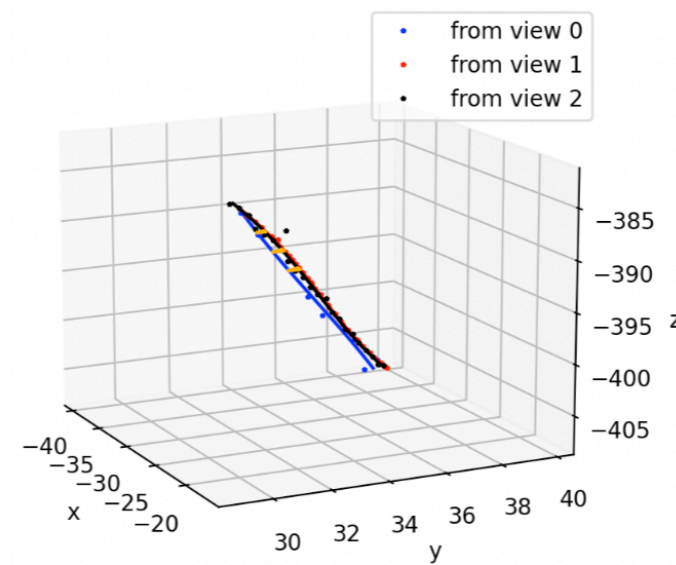


<-> le signal vue par sur ces trois canaux correspond au même dépôt d'énergie

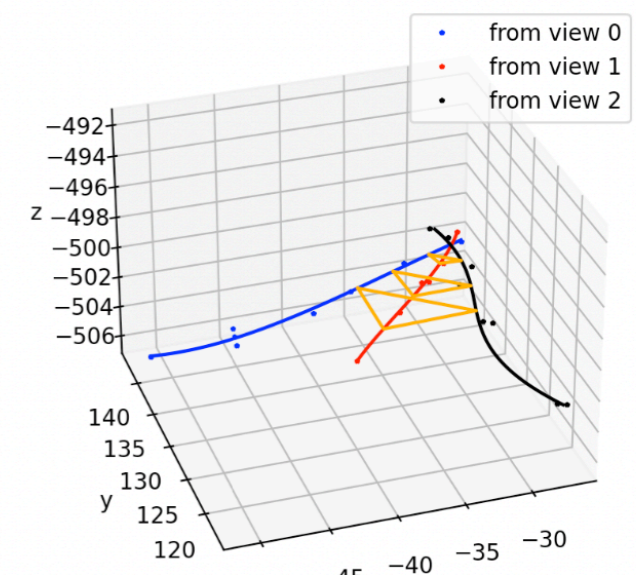


Afin de s'assurer que l'association des traces 2D est correcte, la distance entre les trois traces 3D reconstruites est calculée

Bonne association

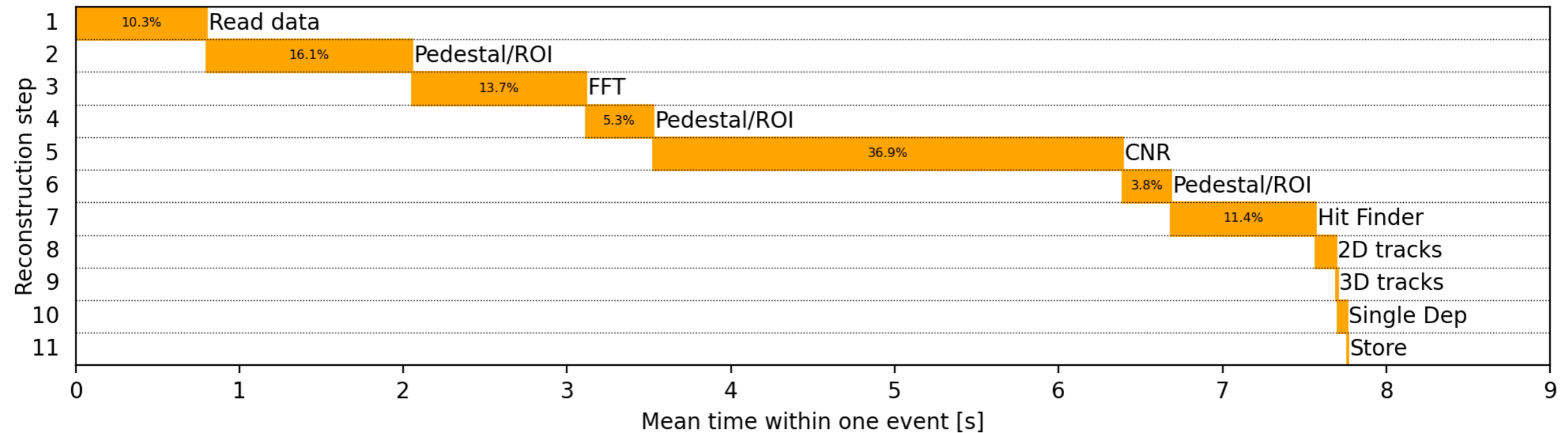


Mauvaise association



# Performances

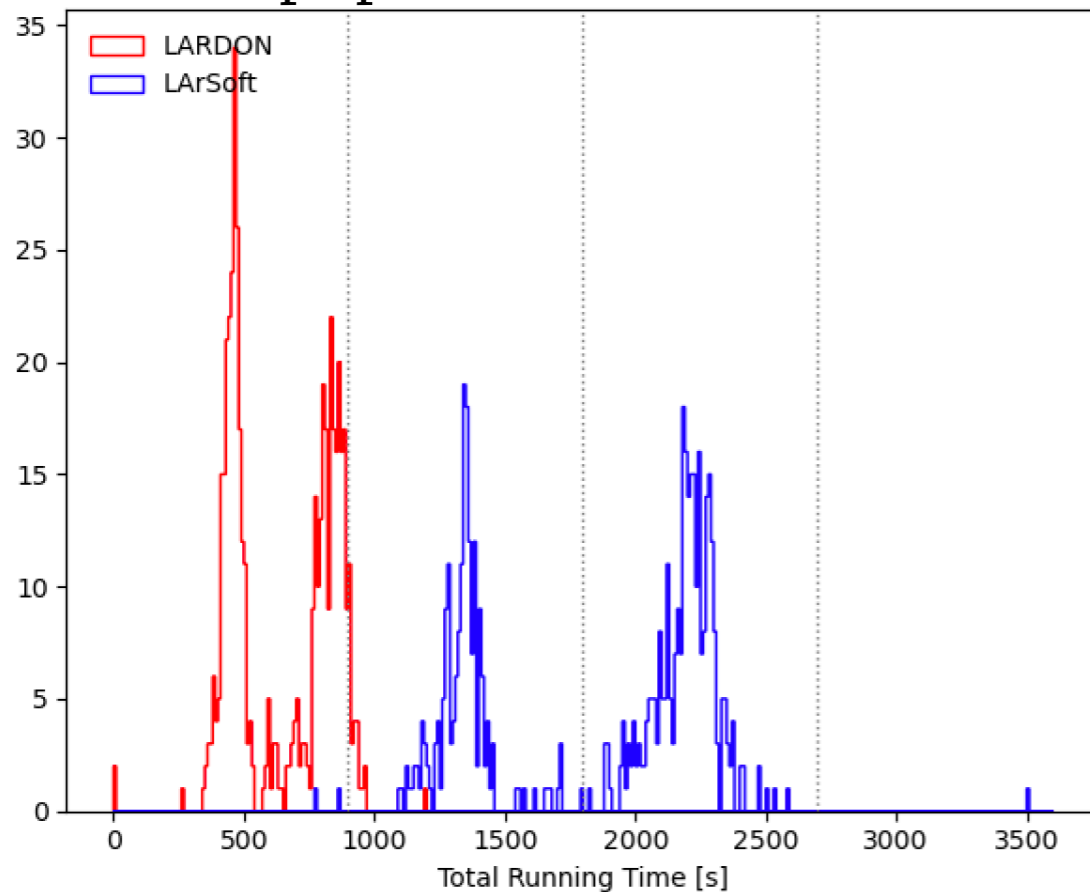
Pour reconstruire un événement d'une durée de 4ms, il faut ~7.7s en moyenne. Le traitement du signal (décodage des données, filtrage du bruit, séparation bruit/signal) représente ~85% du temps



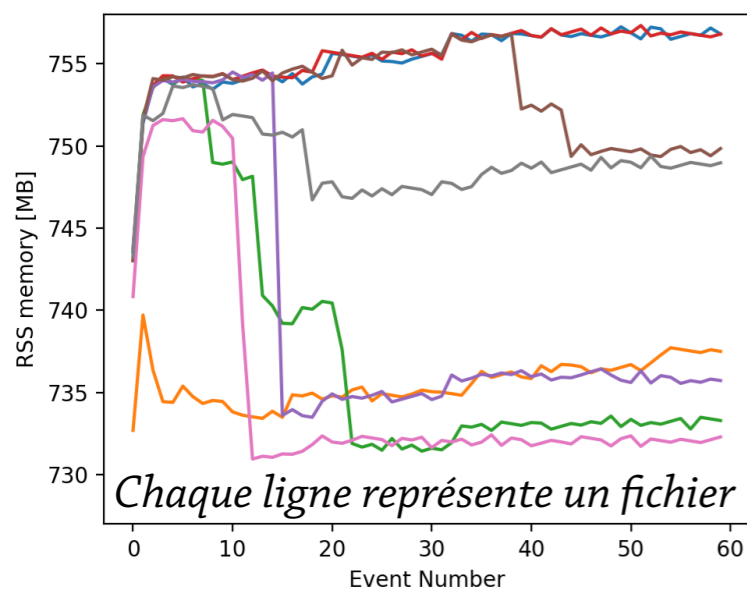
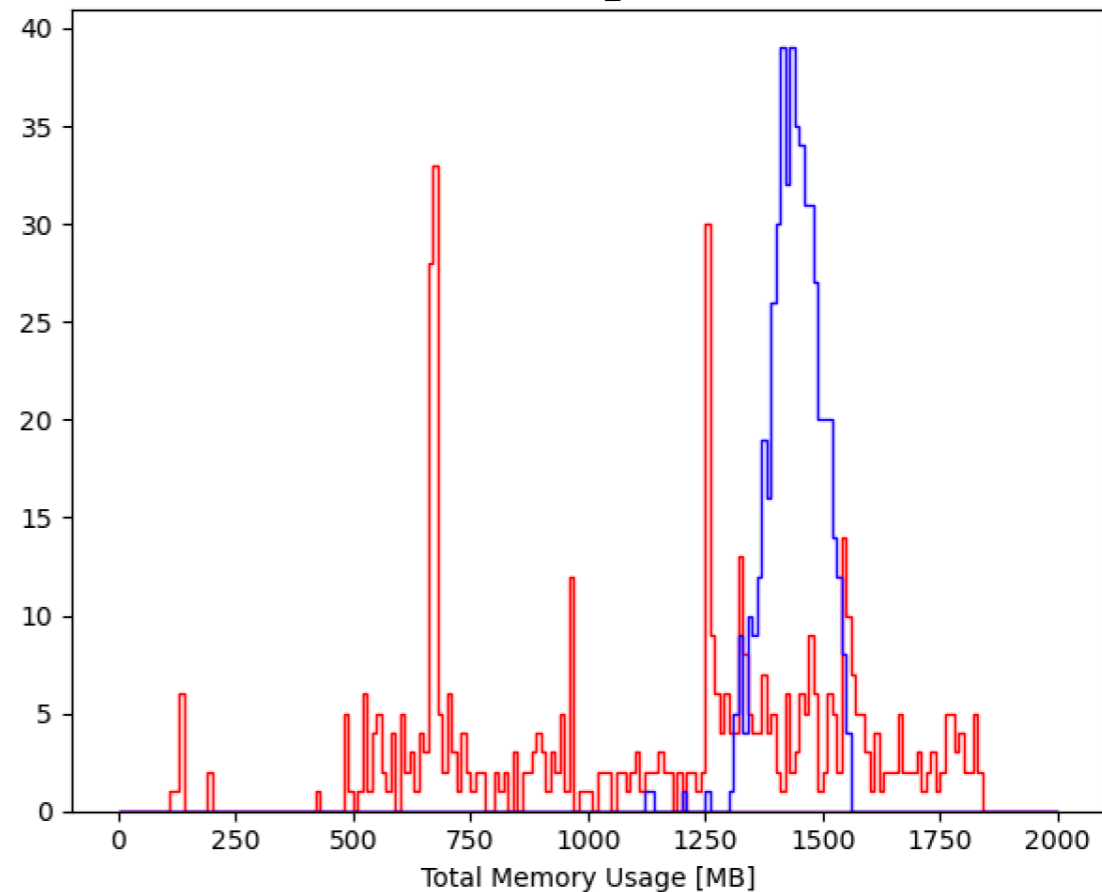
# Performances

Reconstruction de 512 fichiers contenant chacun 60 événements par LARDON et LArSoft sur le système de batch du CERN

Temps pour reconstruire 1 fichier



Mémoire utilisée pour 1 fichier

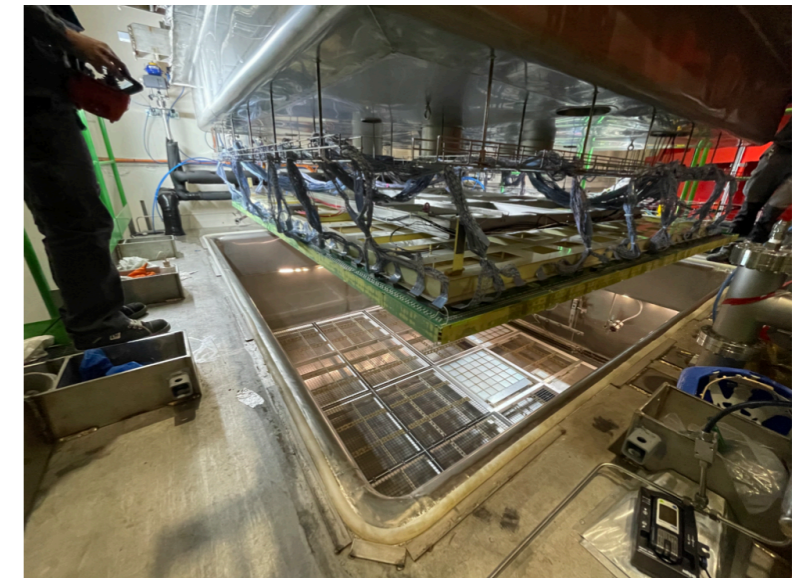
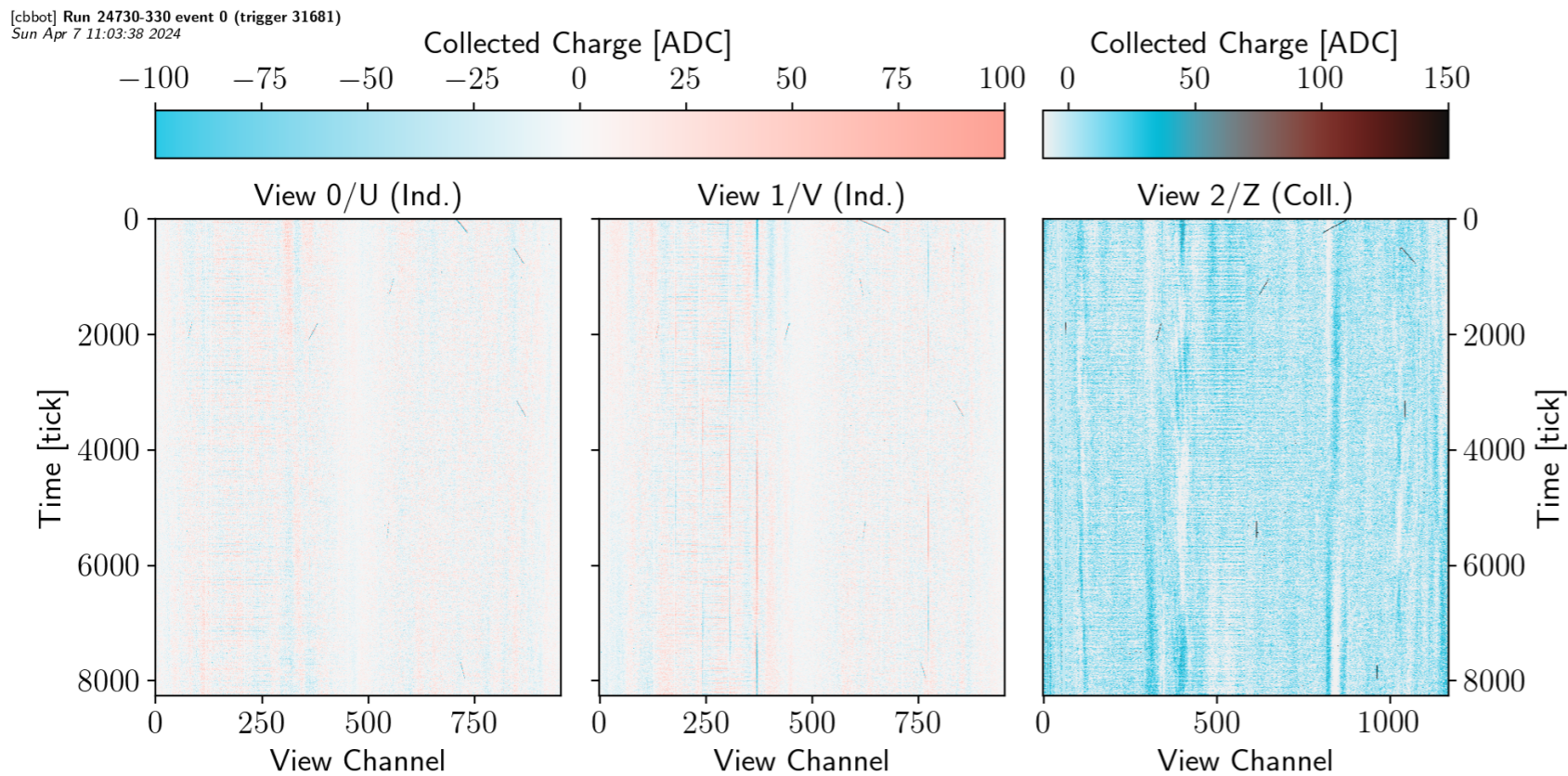


En termes de mémoire pour LARDON:

- Augmentation au début du processus
- Mémoire allouée par numba ?
- Légère augmentation au cours de la reconstruction
- Diminution soudaine de temps en temps - mystère

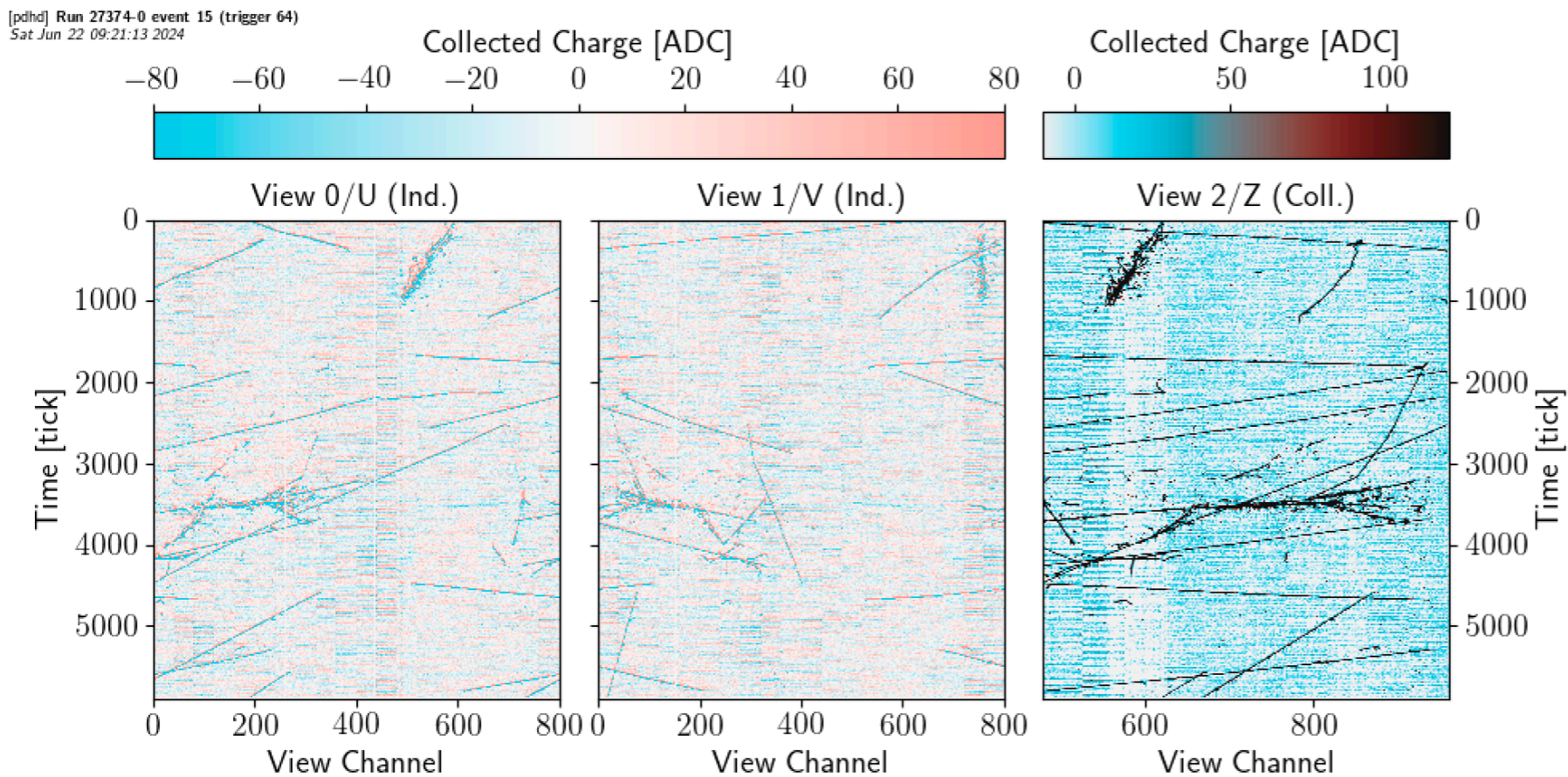
# Vers le futur

Les données jusque là reconstruite avec LARDON sont issus d'un démonstrateur avec une anode de  $3 \times 3.4 \text{ m}^2$  et de 20 cm de distance entre l'anode et la cathode  
-> 1 module d'anode, avec 3,072 canaux

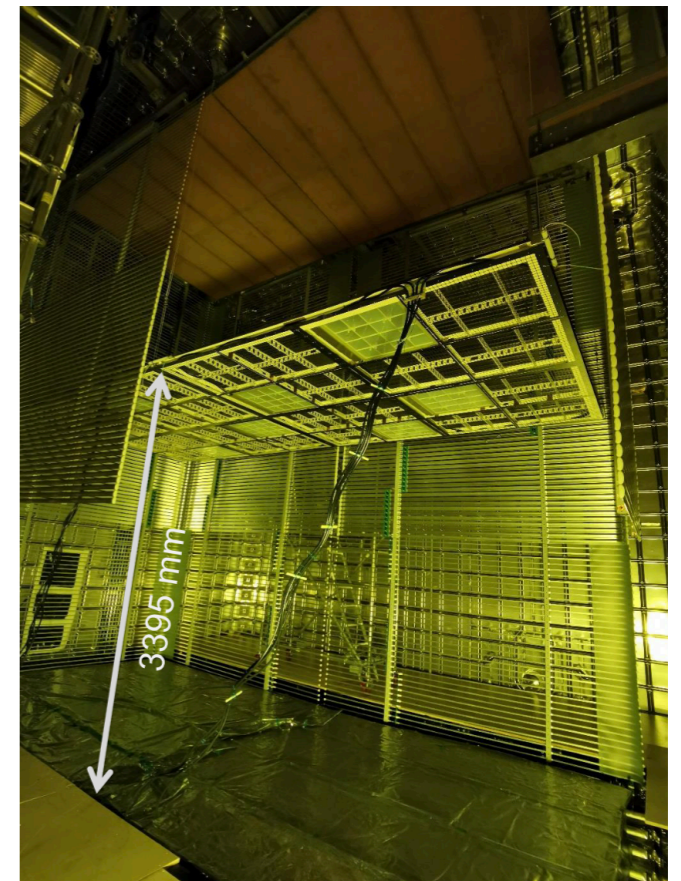


# Vers le futur

A l'automne 2024, un prototype de DUNE prendra des données au CERN. Il sera constitué de 4 modules d'anode avec 3.4 m de distance entre l'anode et la cathode  
-> 12,228 canaux à traiter



*(Données d'un détecteur similaire qui prend des données en ce moment)*



Le détecteur de DUNE aura 160 modules d'anode, avec 6 m de distance entre l'anode et la cathode  
-> 491,520 canaux à traiter !

**-> De nouvelles méthodes d'optimisation de temps de calcul et de mémoire seront nécessaires pour reconstruire ces futures données!**