

TP analyse d'images de cryo-EM

Relion 2.1

Pre-processing

Support: Cryo-EM Images de beta-galactosidase, acquises sur un Titan Krios équipé d'une caméra à détecteur direct d'électrons Falcon II. Images extraites du tutorial de Relion 2.1. Pour le tutorial complet, voir ici : https://www2.mrc-lmb.cam.ac.uk/relion/index.php?title=Main_Page

Le terme de pré-processing désigne toutes les étapes initiales du traitement d'images : alignement des frames, estimation de la ctf des images, picking, extraction et tri des particules. Ici, pour des questions de temps, nous n'aborderons que brièvement les deux premiers items, et nous concentrerons sur la prise en main de relion et le picking des particules.

Relion 2.1 permet d'inclure dans son pipeline l'utilisation de quelques programmes externes permettant l'alignement des frames (MotionCor2 ou Unblur), l'estimation de la ctf (Gctf ou Ctffind4), ou encore l'estimation de la résolution locale des cartes obtenues (ResMap) Il faut installer ces programmes (ie. les télécharger pour la plupart) en plus de l'installation de Relion.

A. Prise en main de Relion2.1

Si besoin, dézipper le dossier de travail :

```
tar zxvf Oleron_practical_2018_EM1.tar.gz
```

Aller dans le répertoire Oleron_Practical_2018

```
cd Oleron_practical_2018_EM1/
```

lister le contenu du répertoire :

```
ls
```

Frames_raw : sous-répertoire contenant des frames brutes utilisées dans ce TP (visualisation & alignement avec unblur si le temps le permet)

My_workdir : sous-répertoire où de travail

Oleron_EM_practicals_preprocessing : sous-répertoire où des résultats pré-calculés pour le preprocessing sont stockés, en cas de manque de temps pour faire soi-même le travail.

Aller dans le sous-répertoire de travail

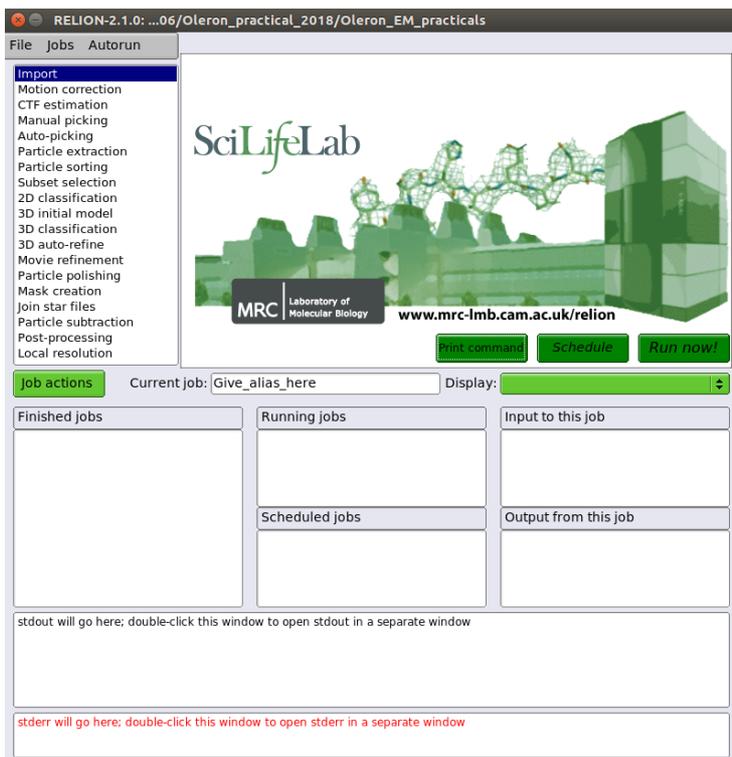
```
cd My_workdir
```

(NB : commande pour vérifier où on est dans l'arborescence : pwd)

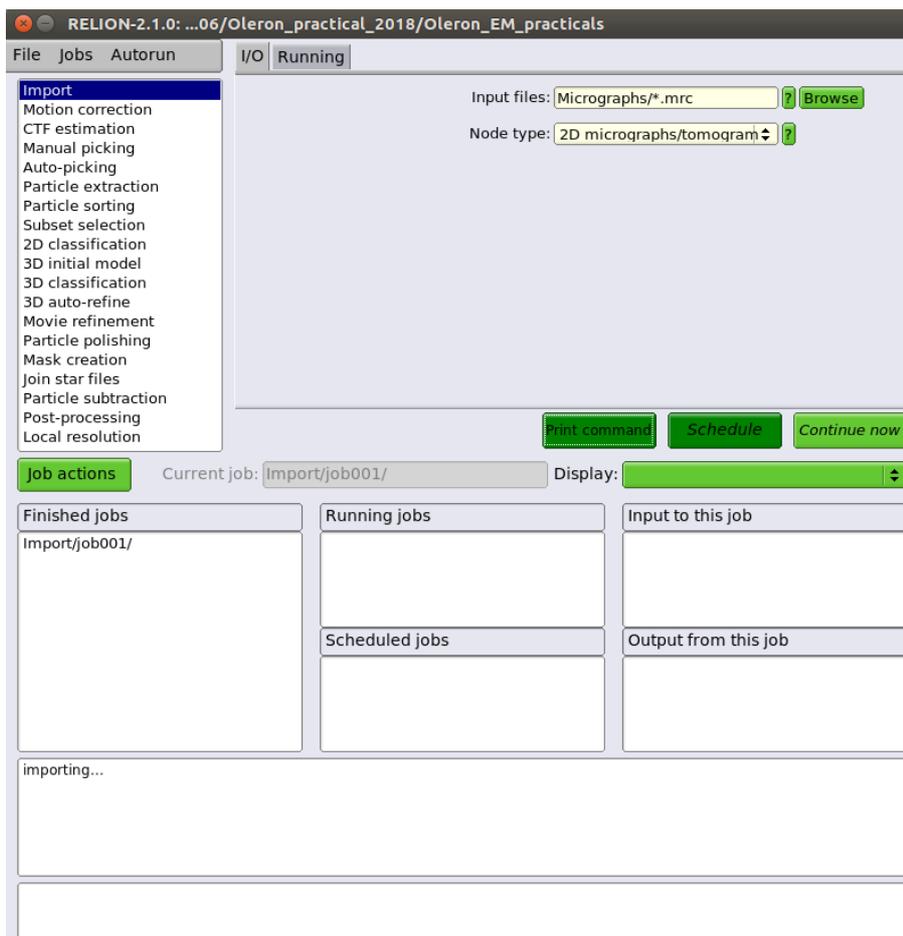
lancer relion

```
relion
```

l'interface graphique de Relion 2.1 apparaît. Le panel de gauche montre toutes les opérations (tâches / jobs) possibles de Relion ; le panel de droite contiendra les différents paramètres à entrer pour effectuer ces tâches. Les fenêtres en-dessous permettent de voir ce qui a été fait dans ce répertoire / avec ce jeu de données (pour l'instant tout est vide car on n'a pas encore commencé à travailler).



B. Import des images brutes (Micrographs)



Sélectionner la tâche **Import** en cliquant dessus. Dans le panel de droite, les différents inputs/options de cette tâche devront être remplis.

Input files (images/fichiers à importer) : **Micrographs/*.mrc** (toutes les images du sous-répertoire Micrographs terminant par « .mrc »)

Node type : type d'images à importer (mrc movies, mrc realigned images, tomogrammes...)

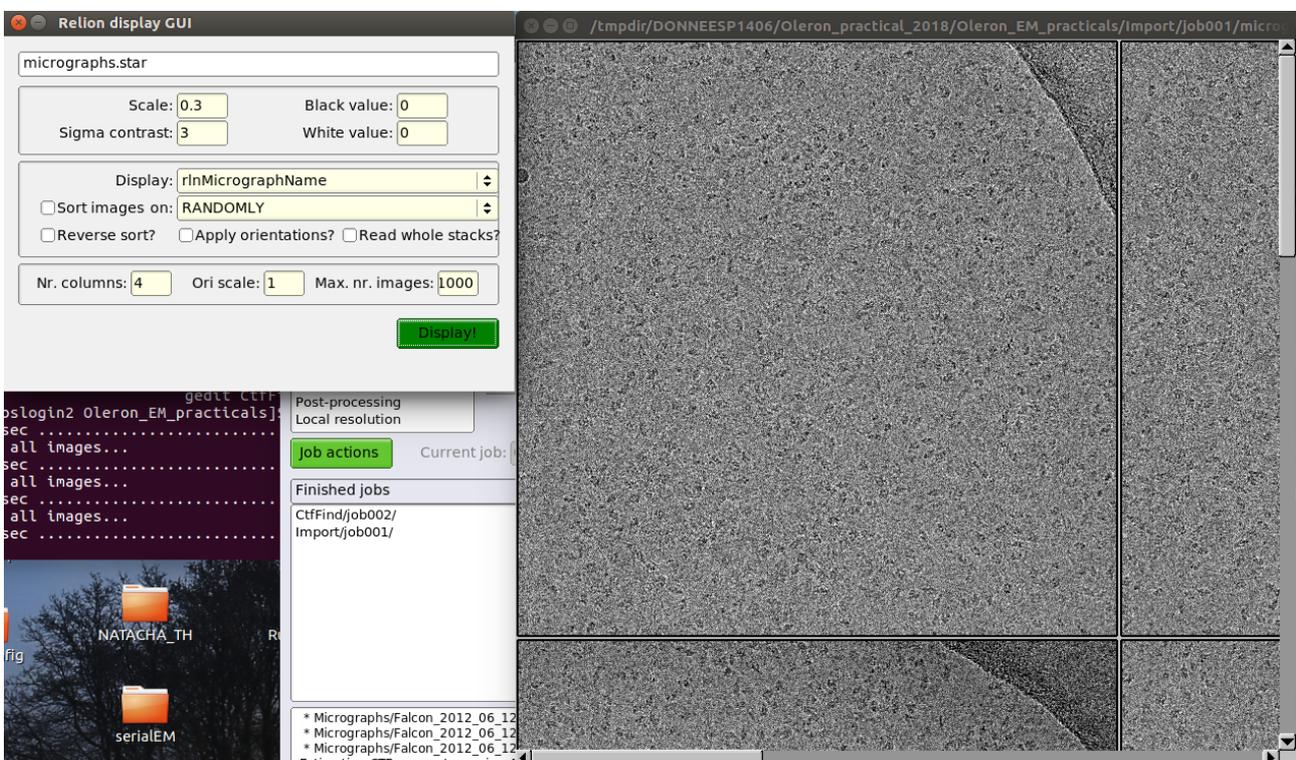
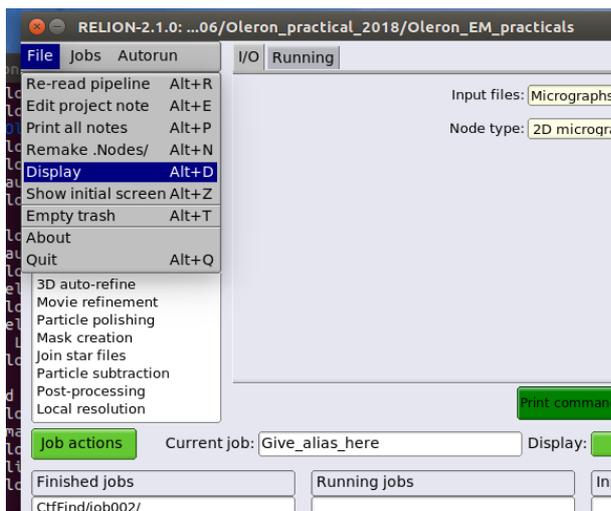
ici le réalignement des movies a déjà été fait, on importe donc des images mrc réalignées : choisir l'option **2D micrographs/tomograms**.

Appuyer sur **Run now !** (milieu – droite du GUI). Lorsque l'import est terminé, la tâche Import/job001 apparaît dans la fenêtre « finished jobs »

Dans le sous-répertoire Oleron_EM_practicals, un sous-répertoire Import contenant le sous-répertoire job001 est

apparu. Import/job001/ contient un fichier texte « micrographs.star » qui liste tous les fichiers à utiliser pour la suite du traitement d'images.

On peut afficher les images brutes en allant sur File/Display (haut-gauche du GUI) et choisir : Import/job001 : micrographs.star.



NB : La visualisation des micrographs sera meilleure (car plus d'options de filtres passe-bande) lors de l'étape de sélection des particules (« picking »).

C. Estimation de la ctf des images brutes

ce travail a déjà été fait en amont du TP pour gagner du temps, avec le programme CtfFind4. Voici les paramètres utilisés pour ce faire :

cliquer sur la tâche **CTF estimation**

Onglet I/O

input micrographs star file : Import/job001/micrographs.star

use micrographs without dose-weighting : no

Spherical aberration (mm): 2

voltage (kV) : 300

Amplitude contrast : 0.1

Magnified pixel size (Angstroms) : 3.54 (images binnées 2x, taille originale de pixel : 1.77Å)

amount of astigmatism (A): 100

Onglet **searches** (paramètres généraux pour Ctffind4, peuvent s'adapter à Gctf):

FFT box size (pix):512

Minimum resolution (A): 30

Maximum resolution (A): 7.1

Minimum defocus cvalue (A): 5000

Maximum defocus cvalue (A): 50000

Defocus step size (A): 500

Amount of astigmatism (A): 100

Estimate phase shifts : No (utilisé que dans le cas d'images acquises avec des phase plates)

Onglet **Ctffind4**

Use CTFFIND-4.1? : Yes

CTFFIND-4.1 executable: /adresse/de/l'exécutable/ctffind4

Estimate Thon rings from movies : No

Estimate CTF on window size (pix): -1

Ctffind4 génère un dossier CtfFind/job002/Micrographs avec des fichiers supplémentaires pour chaque micrograph :

le fichier .ctf est une image (au format mrc) contenant le spectre de puissance de l'image, et le modèle de ctf théorique pour cette image dans le quart supérieur gauche. Le fichier .log contient les résultats de l'estimation de la ctf pour chaque image, et le fichier .com correspond au fichier de commande utilisé pour lancer ctffind4.

On visualise les anneaux de Thon de chacune des images en utilisant display (menu en haut à gauche du GUI de Relion): CtfFind/job002/micrographs_ctf.star

NB : pour ce jeu de données « idéal » on n'a gardé que de très bonnes images, avec de beaux spectres de puissance, donc cette étape de visualisation n'est pas très informative. Mais pour un jeu de données plus conséquent, l'inspection systématique des anneaux de Thon permettra d'éliminer rapidement les images contenant des défauts (astigmatisme, drift, contamination par de la glace cristalline)... Ceci peut être effectué avec la tâche « Subset selection».

D. Sélection (« picking ») et extraction des particules

Dans Relion, la sélection des particules est semi-automatique, et se déroule en 3 étapes consécutives : **(1)**, picking manuel sur quelques micrographs ; extraction des particules. **(2)**, classification 2D des particules extraites, choix des classes 2D modèles pour **(3)**, auto-picking sur tous les micrographs par recherche d'homologie entre les classes 2D (modèles) et les particules potentielles dans les micrographs.

D1. picking manuel des particules

Cette première étape permet de se familiariser avec le jeu de données, et l'aspect/les différentes vues des particules dans la glace.

cliquer sur la tâche **Manual picking**.

Onglet **I/O**

Choisir comme fichier d'entrée le fichier CtfFind/job002/micrographs_ctf.star.

Onglet **Display**, remplir :

Particle diameter (A): 200 (diamètre du cercle affiché autour de la particule)

Scale for micrograph : 0.3 (dépend de la taille de l'écran!)

Sigma contrast : 3 (noir : 3x deviation standard en dessous de la moyenne, blanc : 3x deviation standard au dessus.)

White value : 0

Black value : 0

low-pass filter (A) : 20 (filtre passe-bas permettant de mieux voir les particules si images très bruitées)

High-pass filter (A) : 0

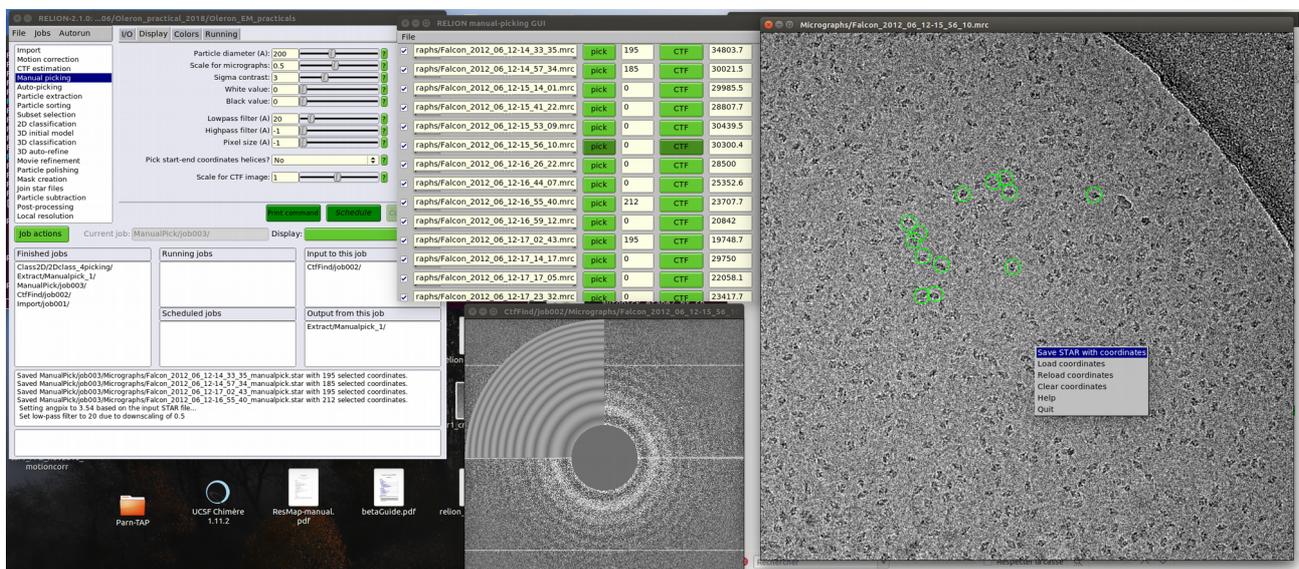
Pixel Size : 3.54

Scale for CTF image : 1 (échelle de l'affichage du spectre de puissance)

Lancer la tâche en cliquant sur **Run now !**

Une fenêtre s'ouvre qui permet de cliquer et visualiser l'image brute et/ou son spectre de puissance et la ctf modélisée.

Sélectionner environ 800-1000 particules sur ~4-5 micrographs. On peut choisir des micrographs avec des valeurs de defocus assez variés. Clic gauche de la souris pour la sélection, intermédiaire pour supprimer une particule sélectionnée, clic droit pour afficher un menu contextuel avec lequel enregistrer les coordonnées (option **save coordinates**)



ceci va permettre d'enregistrer, pour chaque image traitée, un fichier contenant les coordonnées du centre des particules sélectionnées (voir ManualPick / job003 / Micrographs / nomdumicrograph_manualpick.star). Il faut maintenant extraire ces particules dans des boites individuelles carrées (**dimensions : toujours un nombre pair de pixels !!!**). Pour cela, double-cliquer sur la tâche **particle extraction**

comme plusieurs extractions seront faites, on peut donner un alias à cette tâche : Manualpick_1

onglet I/O

micrograph STAR file: CtfFind/job002/micrographs_ctf.star

Coordinate-file suffix: ManualPick/job003/coords_suffix_manualpick.star

OR re-extract refined particles? **No**

Manually set pixel size? **No** (nécessaire que si le fichier d'entrée des micrographs ne contient pas d'information de ctf)

Onglet extract (paramètres d'extraction)

Particle box size (pix): **100** (toujours un nombre pair!)

Invert contrast? **Yes** (pour avoir des particules blanches sur fond noir)

Normalize particles? **Yes**

Diameter background circle (pix): 60 (Les particules seront normalisées à une valeur moyenne de 0 et une déviation standard de 1 pour tous les pixels du « background ». Cette aire correspond à l'extérieur du cercle dont le diamètre est précisé ici.

Stddev for white dust removal : -1

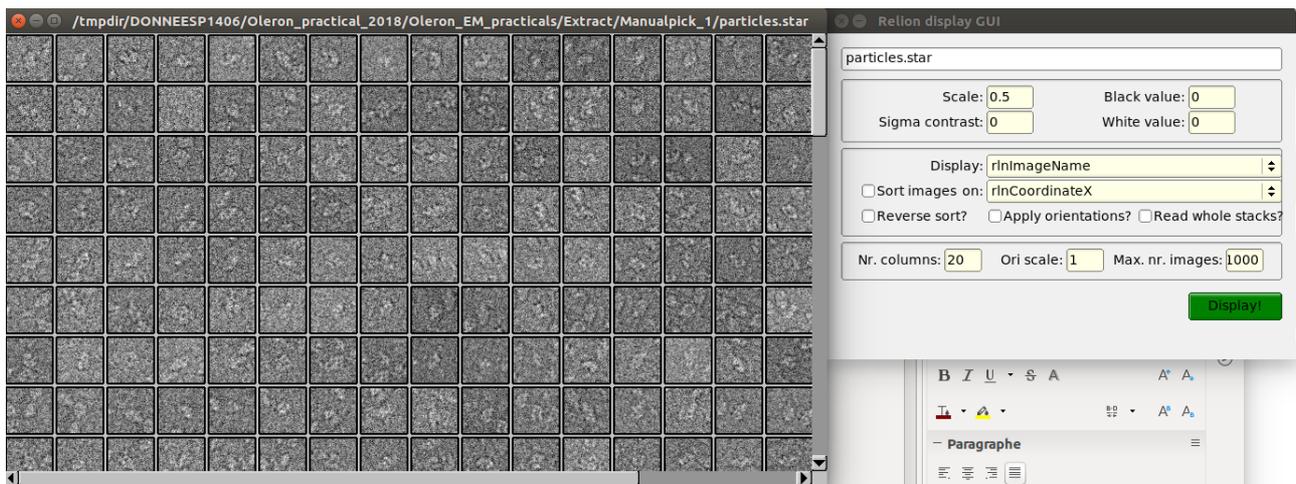
Stddev for black dust removal : -1

Rescale particles : No

Onglet Helix : Ignorer – les particules ne sont pas hélicoidales.

Cliquer sur **Run now !**

Lorsque la tâche est terminée, on peut afficher les particules extraites en allant dans Display : Extract/manualpick_1/particles.star.



D2. Classification 2D

Processus itératif dans lequel les images de particules ayant des orientations similaires dans l'espace vont être groupées, alignées dans le plan puis sommées, ce qui permettra d'augmenter le rapport signal/bruit de chacune des classes. Les meilleures classes 2D seront utilisées pour servir de modèle à l'autopicking (processus de reconnaissance de formes dans les images brutes).

Double cliquer sur la tâche **2D classification**.

Onglet I/O

Input : Extract/job004/particles.star (utiliser le bouton Browse)

Onglet CTF

Do CTF-correction?: Yes (correction de phase+amplitude sur chacune des particules extraites)

Have data been phase-flipped? : No

Ignore CTFs until first peak?: No

Onglet Optimisation

Number of classes: 10 (~100 particules par classes en cryo, 20-50 pour des images de coloration négative)

Regularisation parameter T: 2

Number of iterations: 25

Use subsets for initial updates? No (utilisé pour accélérer les calculs pour de très grands datasets)

Mask diameter (A): 200 (appliqué pour masquer le bruit autour de la particule et d'éventuelles particules trop proches de celle d'intérêt. Attention au diamètre du masque : il ne faut pas qu'il empiète sur la particule d'intérêt!)

Mask individual particles with zeros? Yes

Limit resolution E-step to (A): -1

Onglet Compute

Use parallel disc I/O? Yes

Number of pooled particles: 3

Pre-read all particles into RAM? Yes

Combine iterations through disc? No

Use GPU acceleration? No

Which GPUs to use: 0

Sélection des meilleures classes pour auto-picking

Double-cliquer sur l'onglet Subset selection

Onglet I/O :

Select classes from model .star : Class2D/job005

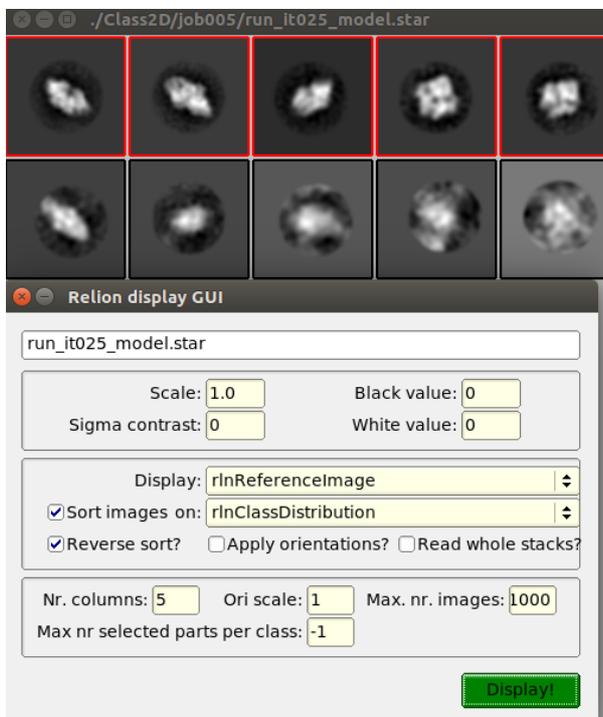
Onglet Class Options

Re-center the class averages : Yes

Regroup the particles : No

Pour plus de clarté, on peut donner un alias à ce job (fenêtre currentjob : give alias here au centre du GUI), comme par ex. select_4_autopick

Run now!



Une fenêtre d'affichage apparait: cocher Sort Images on rlnClassDistribution, et éventuellement reverse sort pour avoir les classes les plus peuplées en haut (autres paramètre : scale 1.0 ; Sigma contrast : 0 ; Black value : 0 ; White value:0). Display !

Une fenêtre s'affiche avec les classes 2D (leur moyenne : class averages) triées en fonction de leur peuplement. Sélectionner 4-6 class averages représentant des vues différentes de la particule. Pour cela, effectuer un **click gauche** sur les class averages d'intérêt, celles qui seront sélectionnées sont encadrées en rouge. Si plusieurs classes similaires apparaissent, n'en choisir qu'une seule. Ne pas prendre de mauvais class average (peu de détail dans la particule, contraste entre la particule : blanche et le bruit : noir peu évident).

Avant de fermer cette fenêtre, ne pas oublier d'enregistrer cette sélection par un **click droit : save selected classes.**

D3. Autopicking

L'autopicking un processus de reconnaissance de forme, où toutes les images brutes (champs de particules / micrographies) vont être « scannées » par chacun des class averages choisis comme références, et chaque zone de

l'image brute ressemblant à l'une des références sera enregistré comme particule. Avant de lancer l'autopicking sur l'ensemble des images, trois paramètres doivent être optimisés sur 2-3 images brutes, afin de définir les critères de ressemblance : le seuil de picking (picking threshold), la distance minimale inter-particule (minimum inter particle distance), et le maximum de déviation standard du bruit de fond (maximum stdev noise). Pour cela, on va choisir 2 ou 3 images type grâce au **Subset selection**. Dans l'onglet **I/O**, choisir la ligne « select from picked particles » : `manualPick/job003/coords_suffix_manualpick.star`

Pour faciliter la compréhension du processus, on peut donner un alias à ce job, comme par exemple `2mics4autopick`.

Une fenêtre contenant la liste de toutes les images brutes s'ouvre : choisir 2-3 micrographs parmi elles, avec si possible des valeurs de defocus et des épaisseurs de glace différents (on peut pour cela utiliser l'option invert selection. Lorsque 2 ou 3 micrographs ont été choisis, **ne pas oublier de sauver la sélection** (en haut à gauche de la fenêtre : **File/Save selection**).

Ensuite, sélectionner la tâche **Autopicking**

onglet **I/O** :

Input micrographs for autopick : aller chercher avec browse dans le répertoire `Select/2mics4autopick/micrographs_select.star`

References : aller chercher avec le bouton browse le répertoire `Select/select_4_autopick/class_averages.star`

Or use Gaussian blob ? **No**

Pixel size in micrographs : **-1** (ce paramètre sera ignoré, car la taille de pixel est contenue dans les fichiers .star depuis l'estimation de la ctf)

Mask diameter (A) : **200** (comme pour la classification 2D)

Low-pass filter frequencies (A) : **20** (filtre passe-bas appliqué sur les références, pour réduire les effets de biais dus au modèle, aka « Eistein from noise »)

Highpass filter (A) : **-1** (si on entre une valeur positive, un filtre passe-haut sera appliqué aux images brutes ; peut être utile en cas de gradient d'épaisseur de glace sur les images brutes)

Pixel size in references : **-1** (ignoré ici car les références ont la même taille de pixel que les images brutes)

Angular sampling : **5** (convient dans une grande majorité de cas)

References have inverted contrast : **Yes** (dans les images brutes, les particules sont noires sur fond blanc, alors que c'est l'inverse pour les class averages servant de référence)

Are References CTF corrected ? **Yes** (la classification 2D a été faite avec correction de CTF)

Ignore CTFs until first peak ? **No**

onglet **Autopicking** :

Picking threshold : **0.8** (valeur seuil de pic de corrélation dans les FOM maps en deça duquel les particules ne sont pas sélectionnées)

Minimum inter-particle distance (A) : **200** (distance minimum entre 2 particules ; souvent 50-60 % du diamètre de la particule)

Maximum stdev noise : **1.5** (Les pics de corrélation dans les zones où la déviation standard du bruit de fond des images brutes normalisées dépasse cette valeur ne seront pas pris en compte. Ce paramètre permet de réduire les artefacts de picking (glace contaminée, bords de carbone...))

Write FOM maps : **Yes**

Read FOM maps : **No**

Shrink factor : 0 (filtre les images brutes à la résolution du filtre passe bas des références ; 1 : pas de filtre. Ce facteur permet d'accélérer les calculs, mais réduit la précision de picking)

Use GPU acceleration : **No**

Which GPU to use : si vide, le programme essaiera de trouver tout seul quels GPUs il peut utiliser.

onglet **Running** :

Number of MPI procs : **1**

Submit to queue : **no**

On peut utiliser un alias comme `optim_param_autopick`

La partie la plus coûteuse en calcul est de calculer la figure de mérite (figure de mérite, FOM) de chacune des positions détectées. Il s'agit d'un coefficient de corrélation entre chacune des références et chacune des positions sur les images brutes. Ce calcul est suivi d'un algorithme (beaucoup plus rapide) de détection de pics, prenant en compte le seuil minimum, la distance interparticule et le niveau de bruit mentionnés ci-dessus. Ces paramètres doivent être optimisés, le programme va générer des « FOM maps », deux fichiers de la taille des images brutes par image brute et par référence. Ces FOM maps vont être utilisés pour optimiser les 3 paramètres rapidement

Afficher les résultats du job `optim_param_autopick` : fenêtre Display au centre-gauche du GUI : choisir out : `coords_suffix_autopick.star`

Vous pouvez choisir de picker plus ou moins de particules. Pour cela, dans la fenêtre finished jobs, double-cliquer sur le job `Autopick/optim_param_autopick` qui vient de se terminer, et aller à l'onglet autopicking

Picking threshold : `0.6` (ou ce que vous voulez essayer)

Minimum inter-particle distance (A) : `200`

Maximum stddev noise : `1.6` (ou ce que vous voulez essayer)

Write FOM maps : `No`

Read FOM maps : `Yes`

cliquer sur `continue now` puis aller dans le job actuel : Display : `Out:coords_suffix_autopick.star`. Le picking vous paraît-il optimal ? Si non, vous pouvez essayer de recommencer comme ci-dessus pour que cela vous paraisse mieux. Pour ce TP, les paramètres utilisés sont : picking threshold 0.4 ; minimum inter-particle distance : 170, maximum stddev noise : 1.0.

NB : si la tâche Auto-Pick/Optim_param_autopick reste bloquée dans la fenêtre Running jobs, faire un click droit dans celle-ci. Si elle ne bascule toujours pas dans les finished jobs, on peut la sélectionner et cliquer dans job actions (centre droit du GUI) : marked as finished.

Une fois que vous avez trouvé des paramètres d'autopicking optimaux, vous pouvez les appliquer à l'ensemble de votre dataset. Pour cela, double-cliquez sur la tâche `Auto-picking` dans l'onglet I/O, remplacez le fichier star d'entrée contenant les 2 micrographs par le fichier star contenant tous les fichiers dont la ctf a été déterminée : `CtfFind/job002/micrographs_ctf.star` ; ne modifiez pas les autres paramètres, sauf, dans l'onglet autopicking

Picking threshold : `0.4`

Minimum inter-particle distance (A) : `110`

Maximum stddev noise : `1.1` (ou ce que vous voulez essayer)

Write FOM maps : `No`

Read FOM maps : `No`

Vous pouvez donner un alias à cette tâche, par exemple `pick_all_micrographs`. Une fois la tâche terminée, vous pouvez vérifier le picking de quelques-uns des micrographs, en cliquant sur la tâche `pick_all_micrographs` et en allant afficher le `out:coords_suffix_autopick.star` (dans la fenêtre Display au centre gauche du GUI).

Si le picking vous convient, vous devez maintenant **extraire les particules auto-pickées**.

Pour cela, double-cliquer sur la tâche `Particle extraction`, et remplacer les fichiers d'entrée (onglet I/O) ainsi :
micrographs star file : `CtfFind/job002/micrographs_ctf.star`

input coordinates : utiliser le bouton Browse pour aller choisir le fichier `coords_suffix_autopick.star` du job `pick_all_micrographs`.

re-extract refined particles : No

manually set pixel size : No

Onglet Extract :

Particle box size (pix): 100 (comme lors du manual picking Attention, si on change ce paramètre on ne pourra pas faire le sorting des particules ensuite car celles-ci n'auront plus la même taille de boîte d'extraction que leurs références)

Invert contrast? Yes (pour avoir des particules blanches sur fond noir)

Normalize particles? Yes

Diameter background circle (pix): 60

Stddev for white dust removal : -1

Stddev for black dust removal : -1

Rescale particles : No

Vous pouvez attribuer un alias à cette tâche, comme par exemple `all_mics_autopicked`. Pour lancer la tâche, n'oubliez pas d'appuyer sur **Run now !**

A la fin de l'extraction, la fenêtre de résultats de la tâche actuelle (en bas du GUI) vous donne le nombre total de particules : combien en avez-vous extrait ? (si vous avez utilisé les paramètres ci-dessus, vous devriez en avoir 8835)

E. Tri des particules

Il est possible de trier les particules en fonction de leur différence par rapport à leur référence (CTF-correctée). Pour cela, double-cliquer sur **Particle Sorting**

dans l'onglet **I/O**

Input particles to be sorted : avec le bouton **Browse**, choisir : `Extract/all_mics_autopicked/particles.star`

are these from an extract job : Yes

Autopicking references : `Select/select_4_autopicking/class_averages.star`

dans l'onglet **References**

pixel size in references (A) : -1

are references CTF corrected : Yes

ignore CTFs until first peak : No

on peut donner l'alias `sort_all_extracted_particles` à cette tâche

Run now !

On va maintenant pouvoir afficher et trier les particules en fonction de leur « Z-score » (score Z établi par la ressemblance à du bruit blanc de la différence entre une particule et sa référence ; plus le Z-score est élevé et plus la particule a des chances d'être un faux positif très contrasté : carbone, agrégats, saletés...) [Scheres, 2015 ; [10.1016/j.jsb.2014.11.010](https://doi.org/10.1016/j.jsb.2014.11.010)]

pour cela il faut utiliser la tâche **Subset Selection**

onglet **I/O**

ligne Or select from particles.star : utiliser le bouton **Browse** pour sélectionner `Sort/sort_all_extracted_particles/particles_sort.star`

ensuite cliquer sur **Run now !**

Les particules sont affichées par Z-score croissant. On peut descendre dans la fenêtre d'affichage, et décider d'un seuil en dessous duquel on ne gardera pas les particules. Pour cela, cliquer (gauche) sur la dernière particule voulue, puis click droit : select all above , et re-click droit : save selected particles

Combien de particules reste-t-il après sélection ? (regarder dans la fenêtre du bas du GUI).