Journal latticeMech 3d

# 4/01/2020

* J’ai fait des essais avec qt. Mais à bien y réfléchir pour ne pas recommencer tout le travail réalisé jusqu’à présent je crois qu’il faudrait que j’utilise wxpython avec une fenêtre openGL
* J’ai installé wxformbuilder avec lequel je vais essayer de recréer l’interface utilisé auparavant.
* Je pense qu’il faut que je fasse des essais pour incorporer GLUMPY dans le code avec une interface openGL

5/01/2020

* J’ai réussi à incorporer dans un code wx une fenêtre graphique opengl

10.02.2020

\* je reprend le fichier gui main.py :

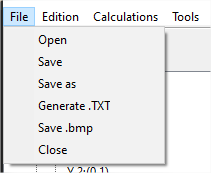
TODO :

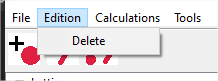
1. il faut permettre l’appel de la partie OPENGL en switch d’une fenêtre « normale »
2. ajouter l’affichage des images de limites élastiques
3. ~~décomposer le code latticeMech en modules~~
4. ~~ajouter tout l’ancien code dans le nouveau~~
5. faire des tests avec des indications des éléments manquants
6. créer un site web contenant cv et code

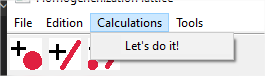
11.02.2020

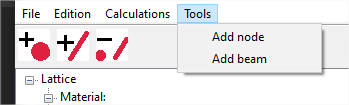
- décomposition du fichier **gui main.py** en son appel de l’interface graphique et incorporation de l’opengl

- ajout du menu









18.02.2020

* Je continue la décomposition du code en modules TODO4
* J’ai décomposer en module..
* Je fait une explication sur le nom des différents éléments désignés dans la version précédente du code située dans *H:\OneDrive - Académie de Strasbourg\travaux de recherche\en cours\latticeMech2 3D 2020\DiscreteLatticeMech-LatticeMech*
* Je met le résultat dans .\*documentation\Désignations éléments.docx*

25.02.2020

Il faudrait :

* Abandonner provisoirement la 3D
* S’intéresser au contrôle de cohérence de la topologie
* Faire l’affichage des limites élastiques
* Utiliser une modélisation de sous-structure
* Obtenir un code python portable