

SansContact/cube_dyna_sd

- cube hpp en traction simple:

> génération: `python gen_cube_tet.py`

> faire le calcul: `python command.py`

> post-traitement: `./post_FvsU.py`

> tracer fz_uz.txt `xmgrace fz_uz.txt`

> refaire le calcul après avoir activé l'amortissement de Rayleigh

> tracer fz_uz.txt

SansContact/cube_quasi_sd

- cube hpp quasi statique en traction simple:

- > génération

- > faire le calcul

hummm, pourquoi ça ne marche pas ?

- > bloquer les mouvements de corps rigide

- > refaire les calculs en prenant les maillages «fin» et «tres_fin»

- > comparer les temps de calculs obtenus en remplaçant le stockage par défaut avec le stockage «exploded»

SansContact/beam_Id

- balancier:

- > génération

- > faire le calcul

- > visualiser le mouvement avec paraview

- > tracer le déplacement vertical d'un point à l'extrémité du balancier

- > refaire le calcul avec un peu d'amortissement

AvecContact/Bascule/Id|coro

- basculement d'un bloc posé sur une fondation
 - > génération
 - > faire le calcul en mesurant le temps de calcul
 - > visualiser le mouvement avec paraview
 - > tracer le déplacement moyen suivant z de la face supérieure du bloc
 - > mettre un peu d'amortissement

AvecContact/Comprime_2D

- écrasement d un bloc
 - > génération
 - > faire le calcul en mesurant le temps de calcul
 - > générer les fichiers vtk avec `command_post.py`
 - > visualiser le mouvement avec paraview
 - > changer la position des points de contact

AvecContact/Comprime_3D

- écrasement d un bloc
 - > génération
 - > faire le calcul en mesurant le temps de calcul
 - > visualiser le mouvement avec paraview
 - > changer la position des points de contact