

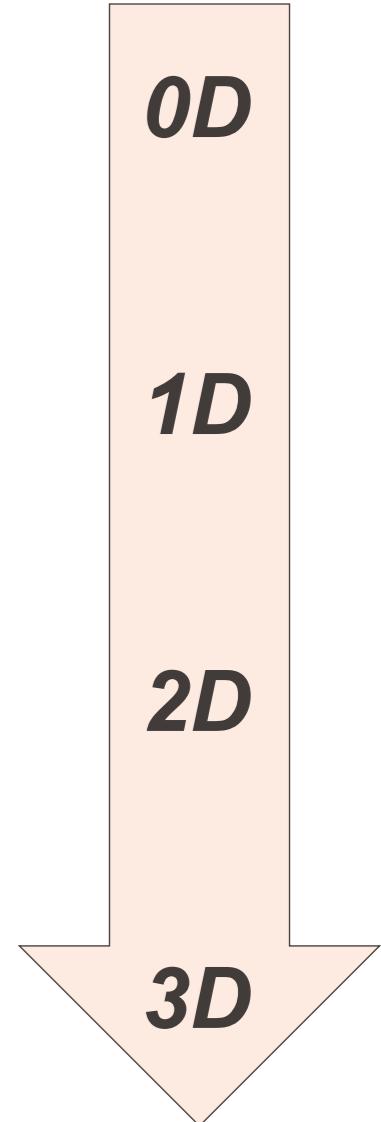
# Modèle CAD, partitionnement et maillage

Modélisation et simulation  
par éléments finis

# Structure d'un modèle CAD

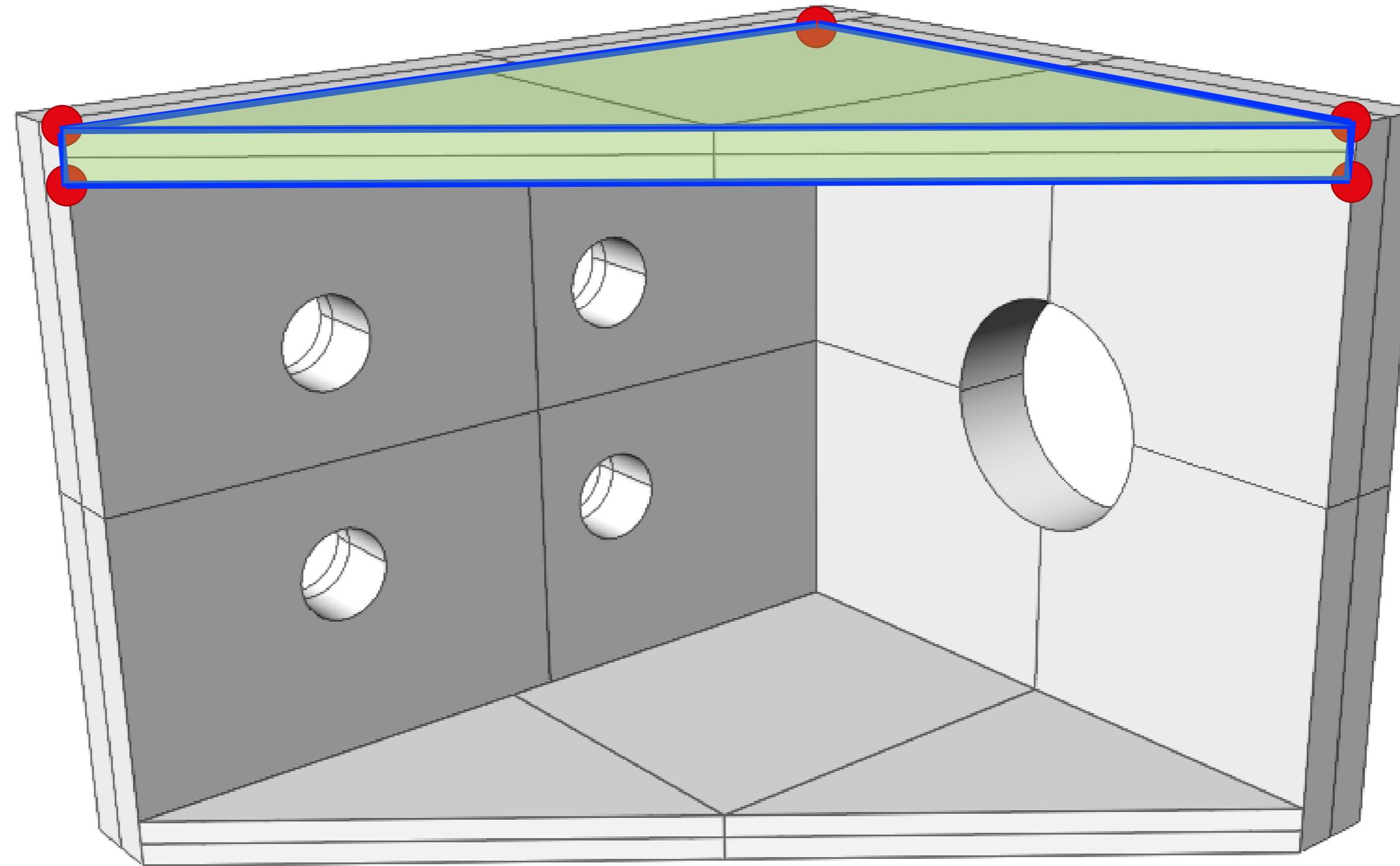
Modélisation et simulation par éléments finis

- Nœuds (0D) : coordonnées & système de coordonnées
- Arêtes (1D) : plusieurs sommets → ligne / courbe
- Surfaces (2D) : boucle fermée d'arêtes (sommets partagés), espace paramétrique 2D, vecteur normal = orientation
- Volumes (3D) : ensemble fermé de surfaces (arêtes partagées), orientation (normale) unifiée



# Exemple

Modélisation et simulation par éléments finis



Volume 3D CAD volume : toutes les arêtes sont partagées par les faces externes → pas d'arête “libre” → surface fermée → volume fermé.

# CAD import in ABAQUS / CAE (rappel semaine 3)

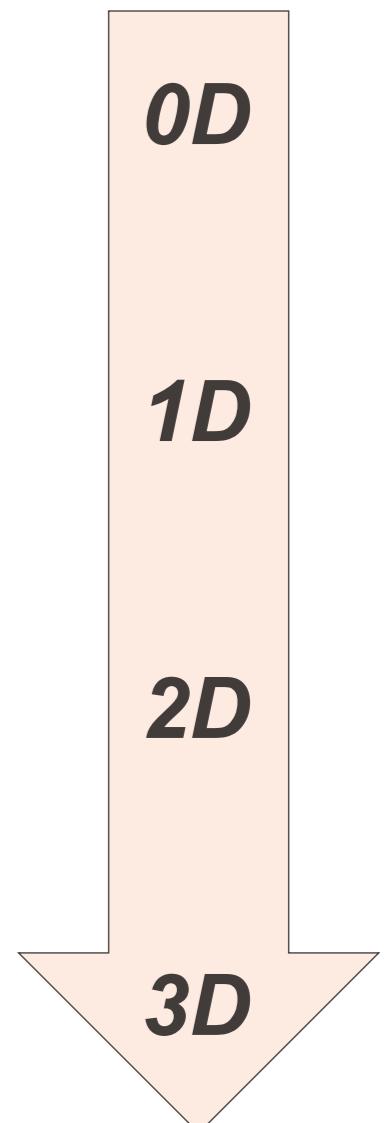
Modélisation et simulation par éléments finis

- Plusieurs formats supportés par Abaqus, par ex. :
  - STEP : universel, bon pour volumes & assemblages.
  - IGES : universel, bon pour surfaces, ok pour volumes.
  - SAT : moteur ACIS, format natif Abaqus, bon pour à peu près tout.
  - CATPart : format CATIA v5 format.
- Toujours vérifier la géométrie :
  - Vérifier dimensions / unités.
  - Déetecter free edges / invalid entities / petites arêtes et surfaces :
    - Réparer free edges : Tools → Query → **Geom. diagnostic**
    - Réparer petites faces : Tools → Geom. edit → Part → **Stitch**
    - Si problème avec maillage : Tools → Geom. edit → Face → **Repair small**
    - Si pb. avec opérations (par ex. partitionnement) : "**Convert to precise**" / "**Convert to analytical**" (nécessaire pour utiliser plans / cylindres pour esquisses).

# Maillage : principe de base

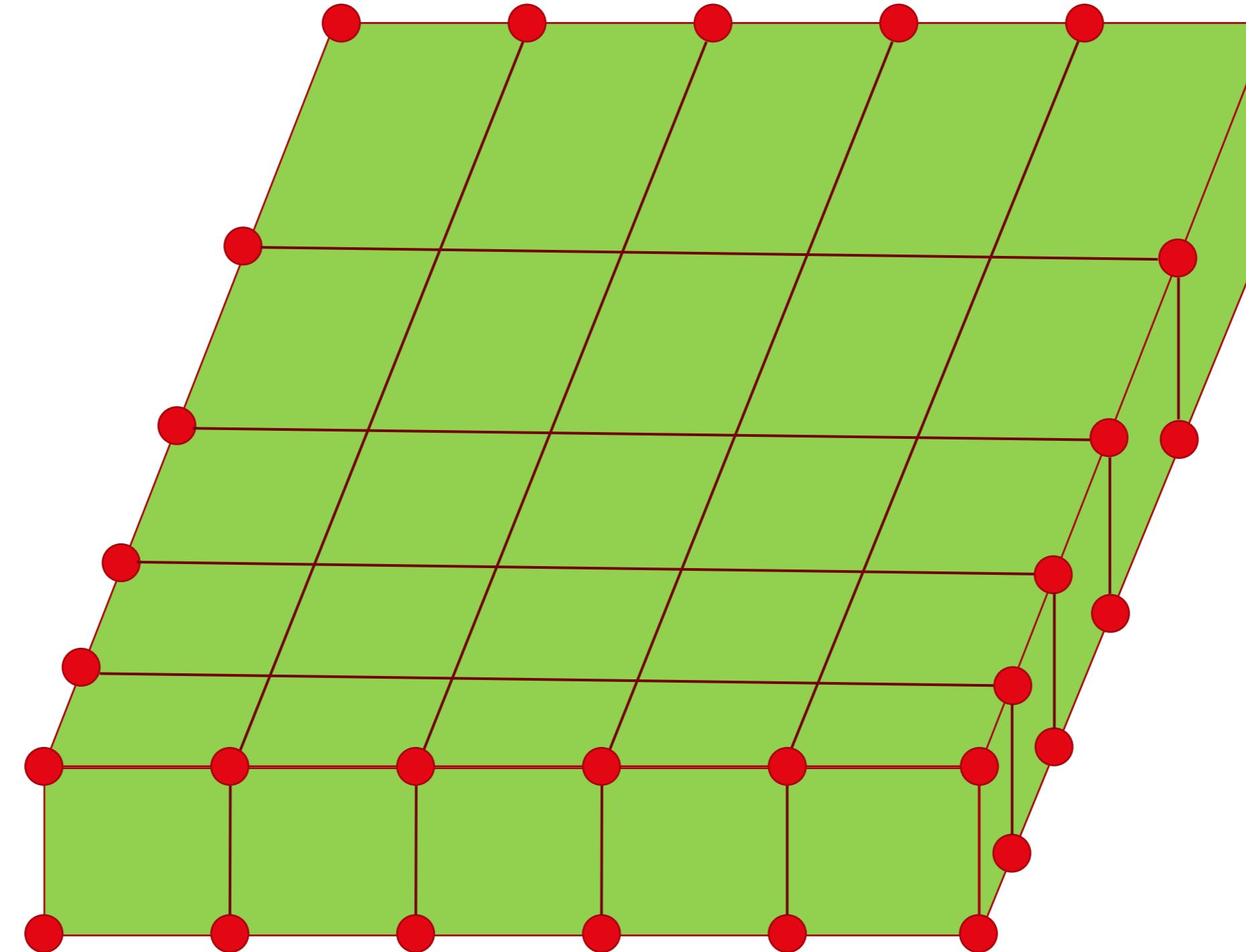
Génération du maillage basée sur la même hiérarchie que dans le modèle CAD :

- **0D** : les nœuds du modèle CAD deviennent des nœuds du maillage.
- **1D** : maillage des arêtes.
  - A partir d'une **taille / distribution** d'éléments définie par l'utilisateur.
- **2D** : propagation du maillage 1D vers surface 2D.
  - **Structuré**, ou **libre** ("advancing front" ou "medial axis").
- **3D** : propagation du maillage 2D vers volume 3D.
  - **Structuré, semi-structuré** (balayé), ou **libre**.



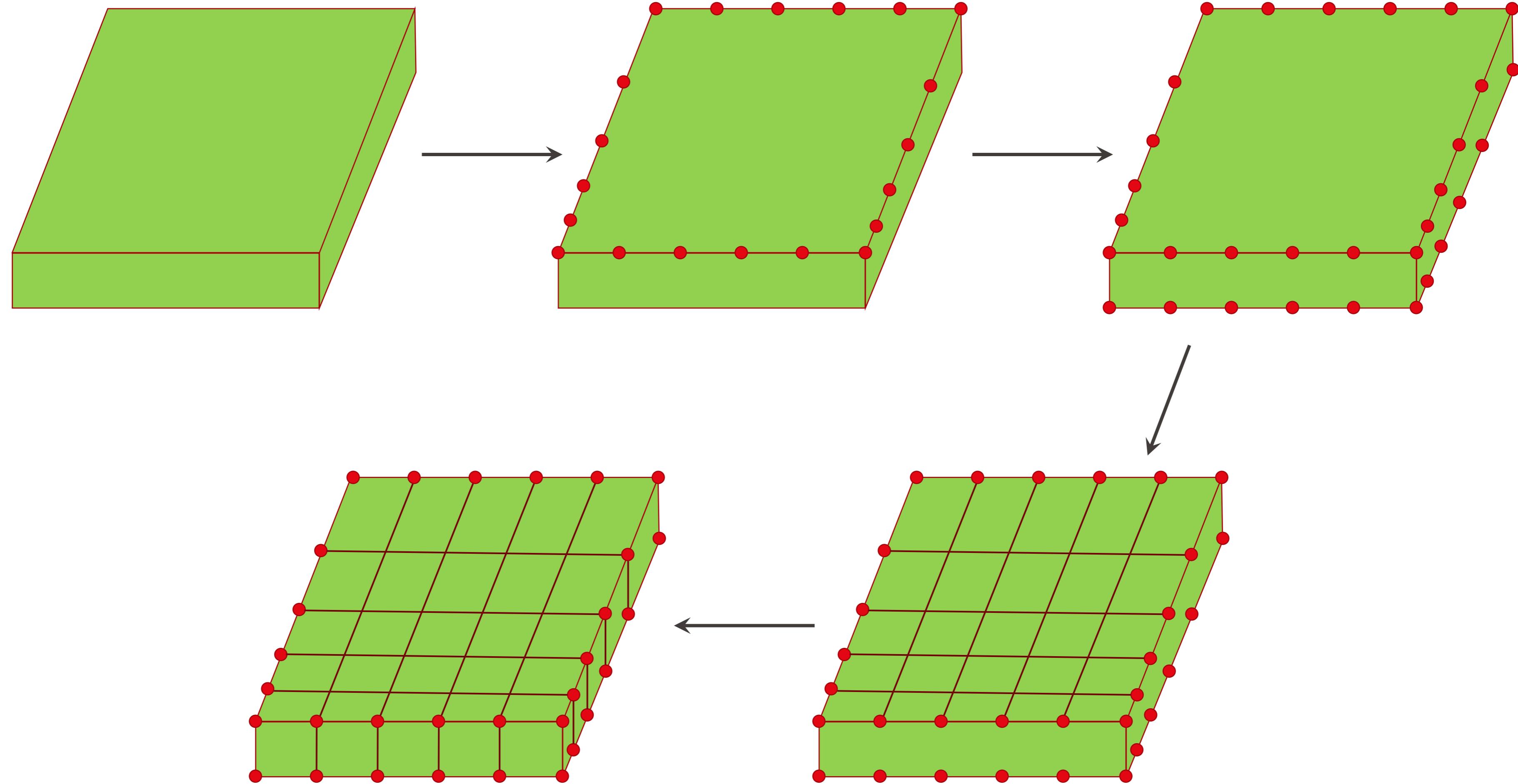
# Maillage : principe de base

Modélisation et simulation par éléments finis

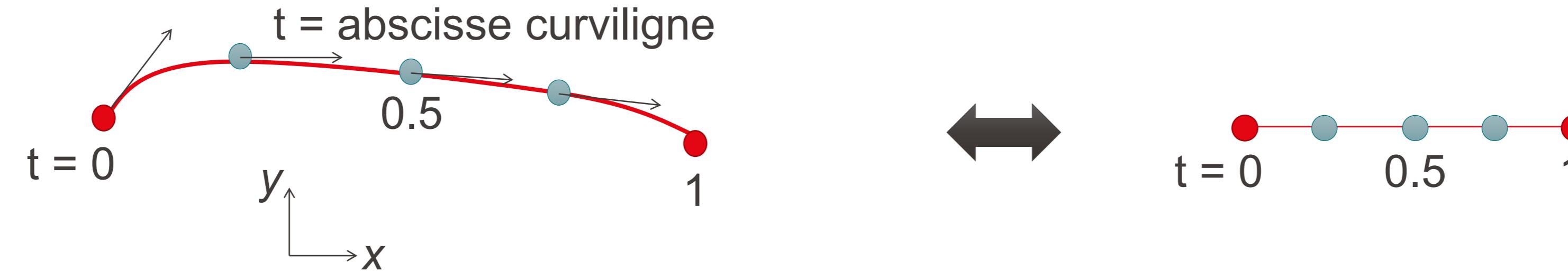


# Maillage : principe de base

Modélisation et simulation par éléments finis



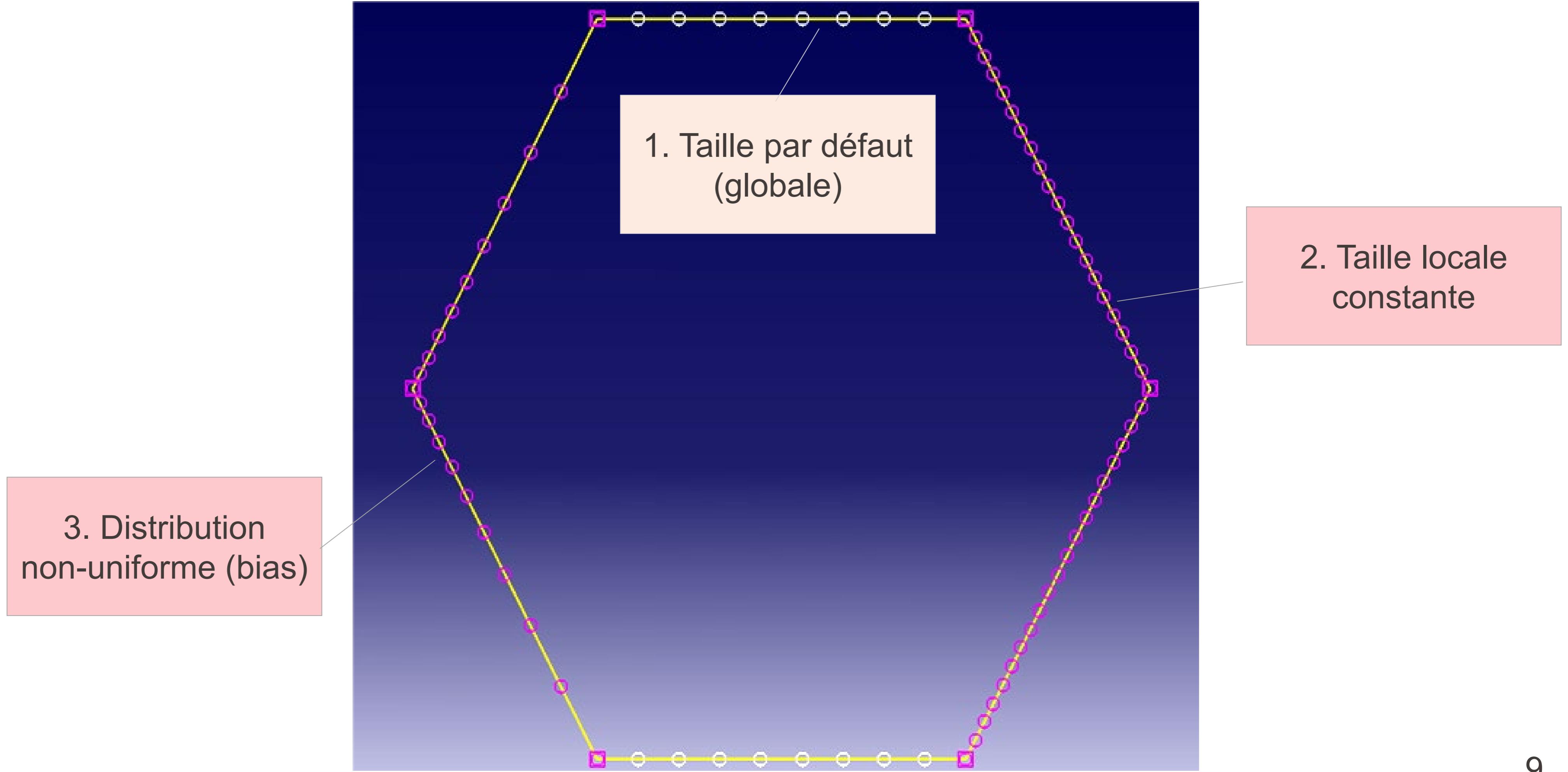
# Algorithmes de maillage 1D



- Méthode :
  - Utilise l'abscisse curviligne pour distribuer les nœuds le long des arêtes  
→ crée des éléments 1D.
- Définition :
  - Constant size : nb. d'éléments sur l'arête, ou taille d'élément
  - Variable size : nb. d'éléments avec distribution non-uniforme ("bias")
    - Bias = rapport des tailles du plus grand et plus petit éléments.
    - Sélectionner le sommet du côté du maillage fin.

# Algorithmes de maillage 1D

Modélisation et simulation par éléments finis



# Démonstration

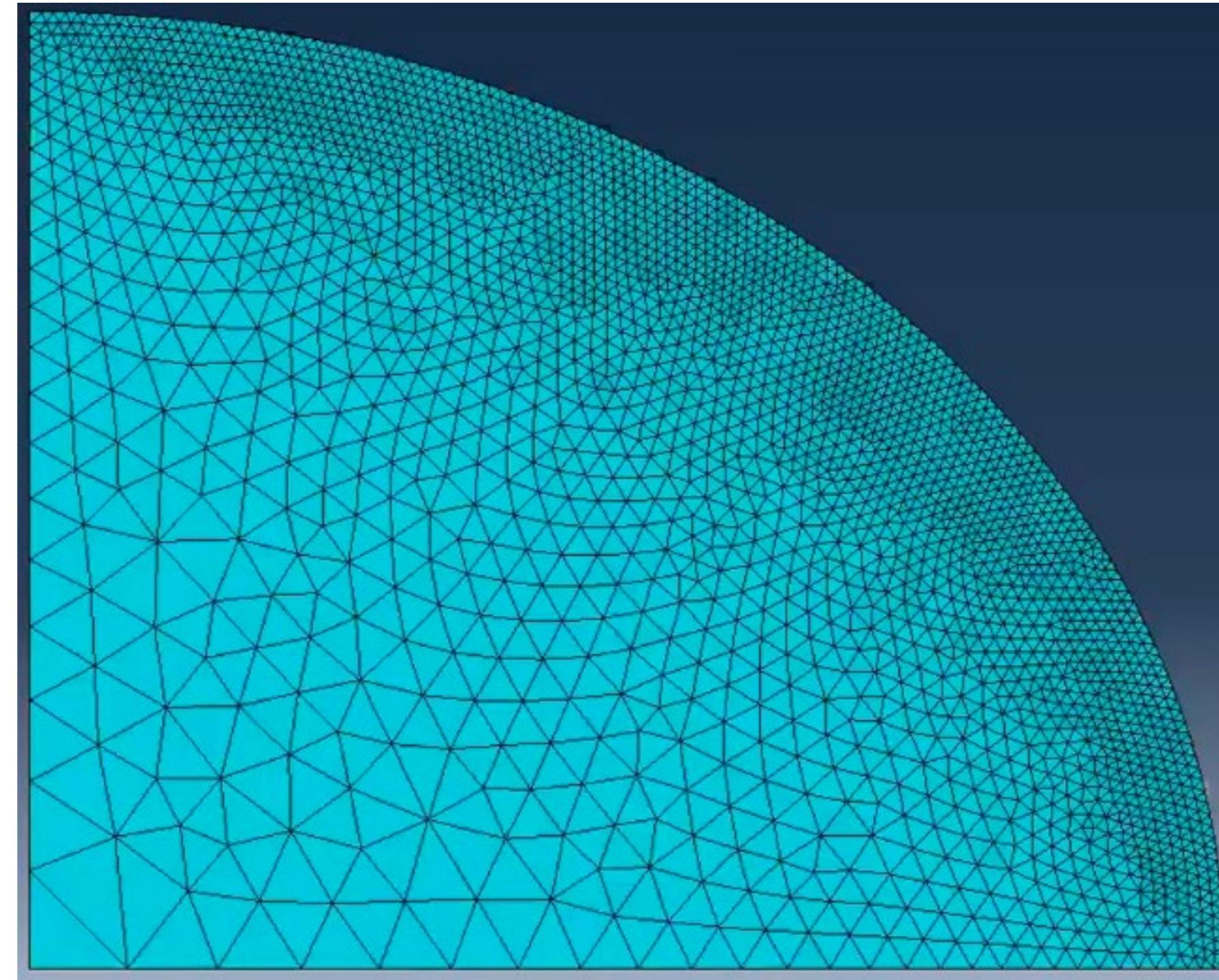
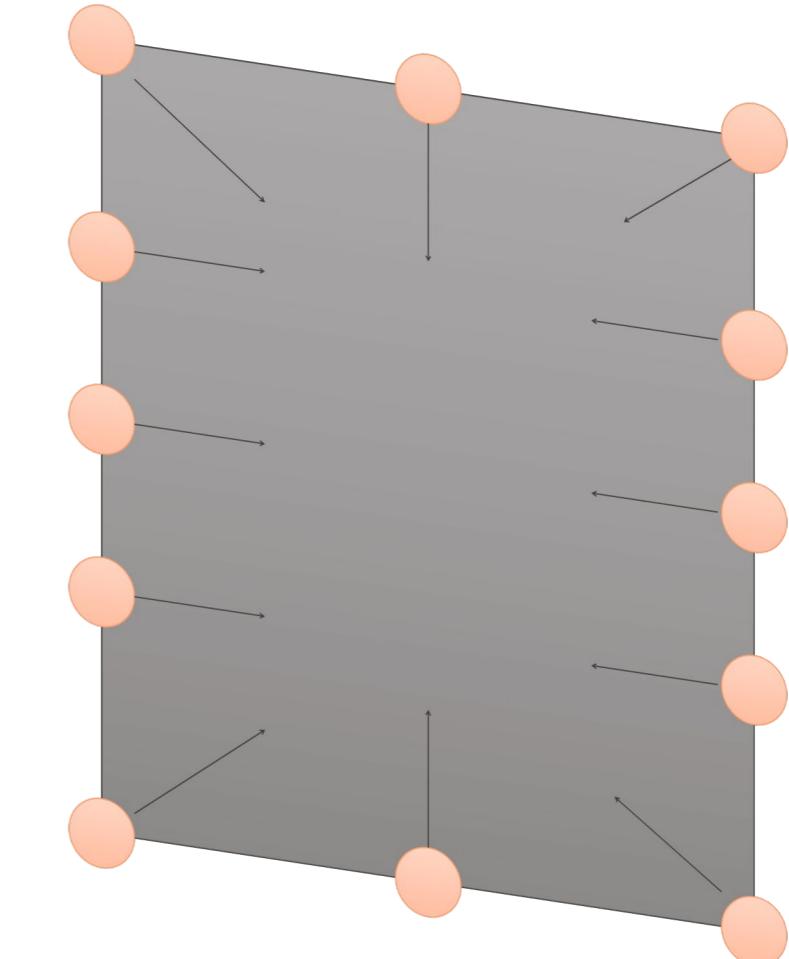


Illustration raffinement local et bias

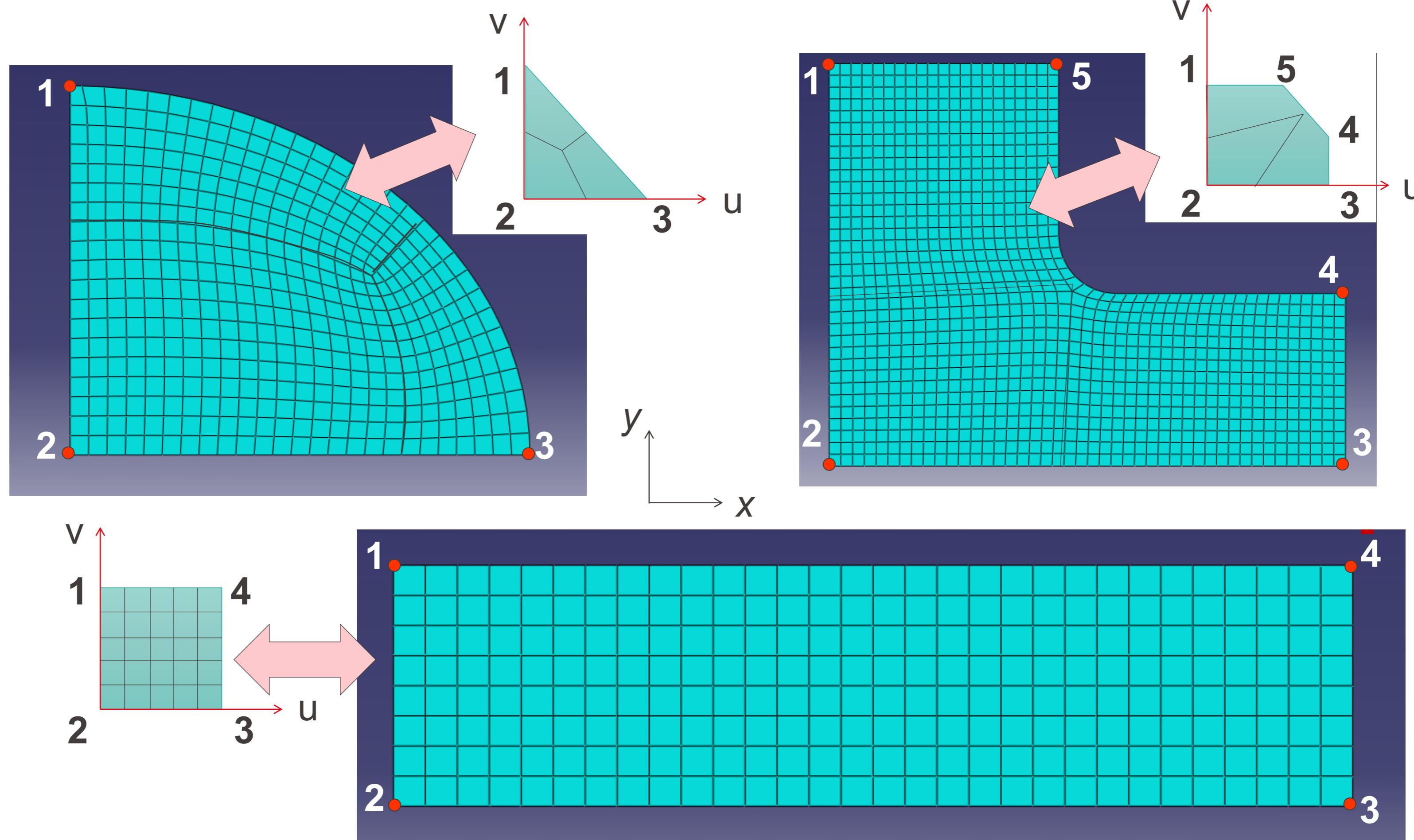
# Algorithmes de maillage 2D

- Méthodes :
  - Propager le maillage 1D sur la surface 2D
  - Surface gauche :
    - Si presque plane : projection sur meilleur plan.
    - Générale : maillage dans l'espace paramétrique.
  - Algorithmes :
    - **Structuré** (structured / mapped) : "projette" la surface sur une surface plus simple.
    - **Libre** : Delaunay triangulation / Advancing front / Medial axis.
- Définition:
  - Simplement sélectionner l'algorithme.
  - Taille de maillage héritée automatiquement des arêtes.



# Algorithmes de maillage 2D structuré

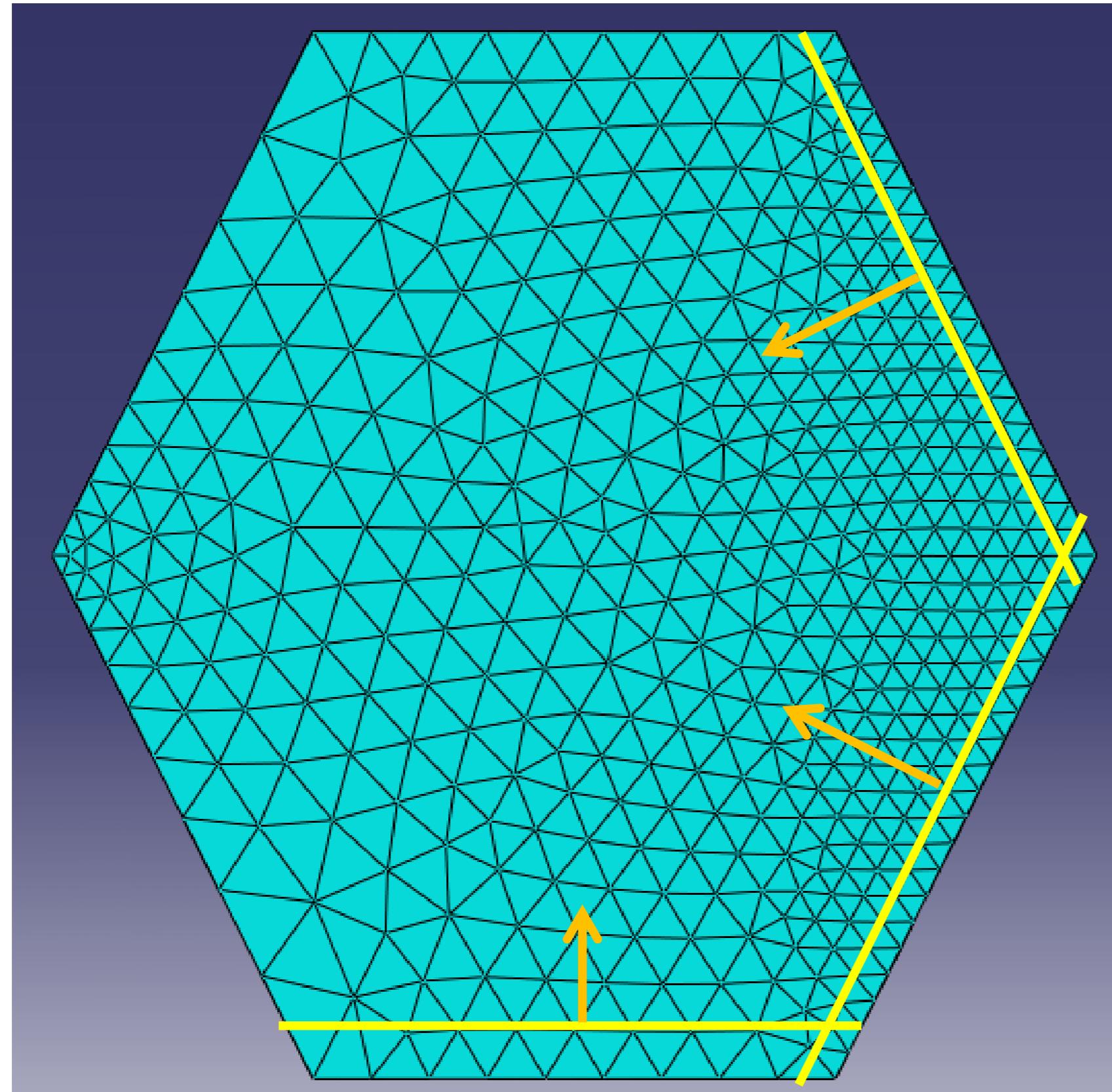
Modélisation et simulation par éléments finis



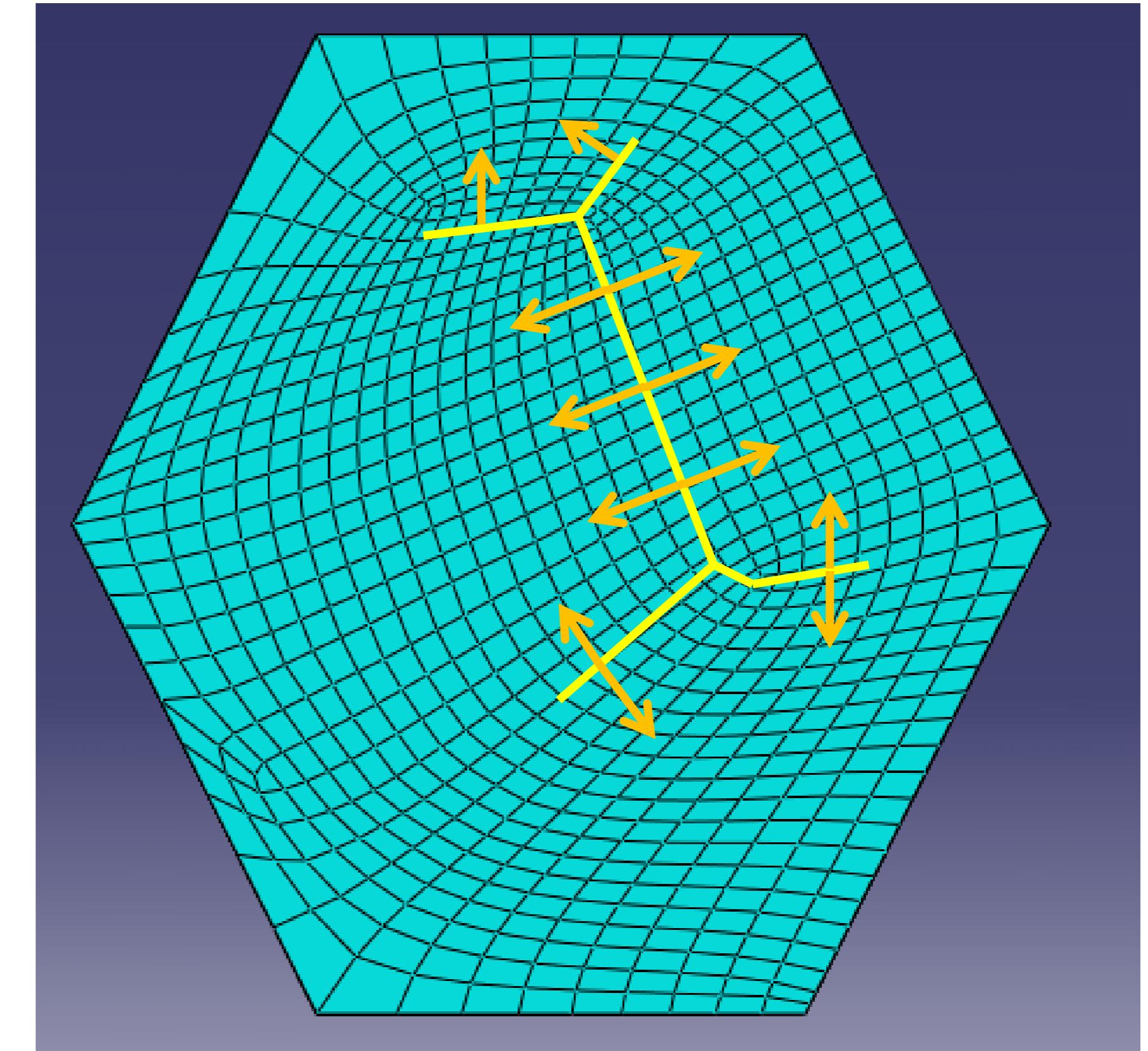
Maillage mapped / structured : fonctionne pour surfaces avec 3, 4 ou 5 coins et arêtes.

# Algorithmes de maillage 2D libre

Modélisation et simulation par éléments finis



Advancing front



Medial axis

# Algorithmes de maillage 3D

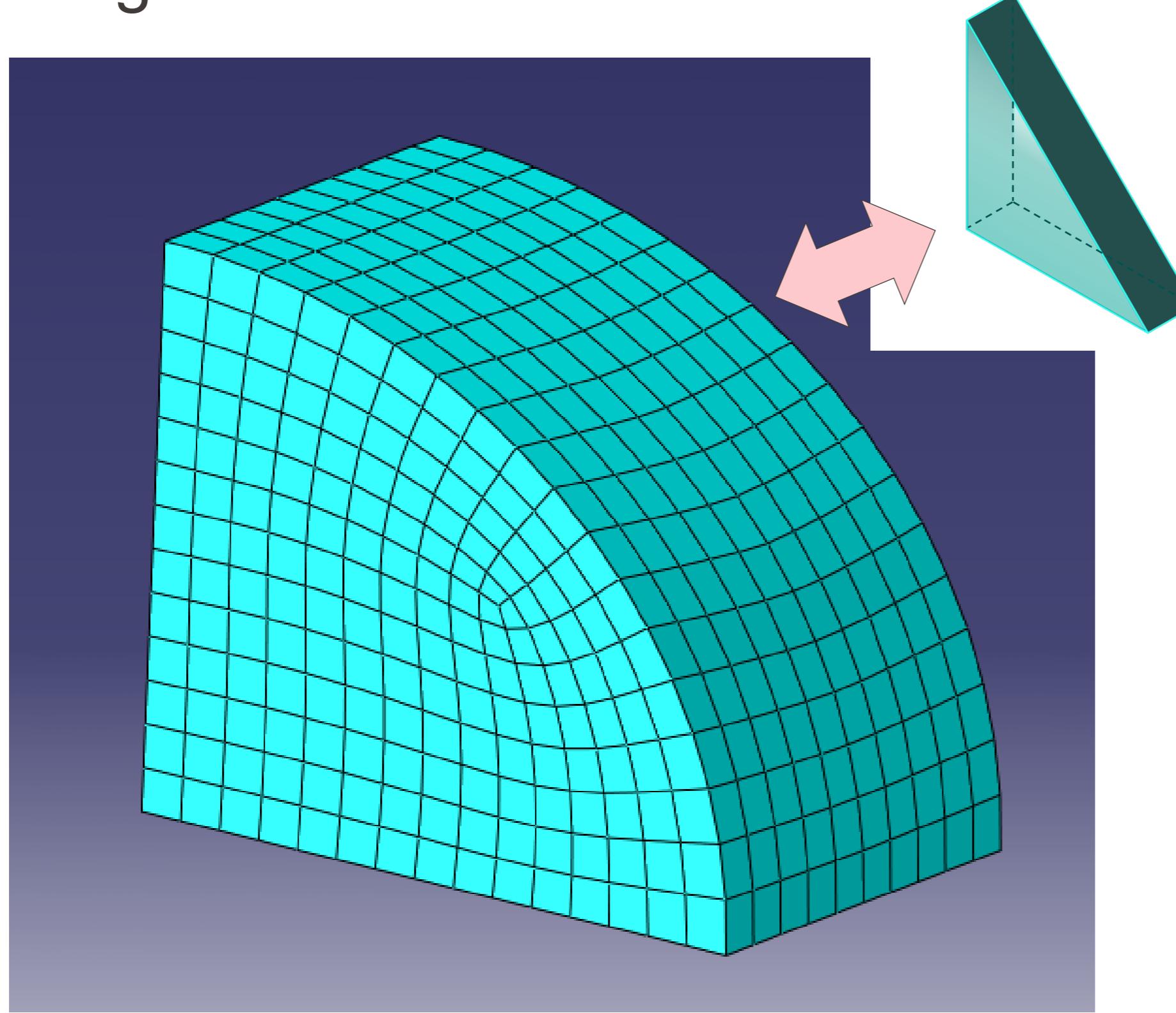
- Méthodes :
  - Propager le maillage 2D dans le volume 3D.
  - Algorithmes :
    - **Structuré** (structured / mapped) : "projette" le volume sur un volume plus simple ( $\rightarrow$  hexaèdres).
    - **Semi-structuré** (balayé) : extrusion / balayage d'un maillage 2D libre ( $\rightarrow$  hexaèdres ou prismes).
    - **Libre** : Delaunay / Advancing Front ( $\rightarrow$  tétraèdres).
- Définition :
  - Simplement sélectionner l'algorithme.
  - Taille de maillage héritée automatiquement des surfaces et arêtes.



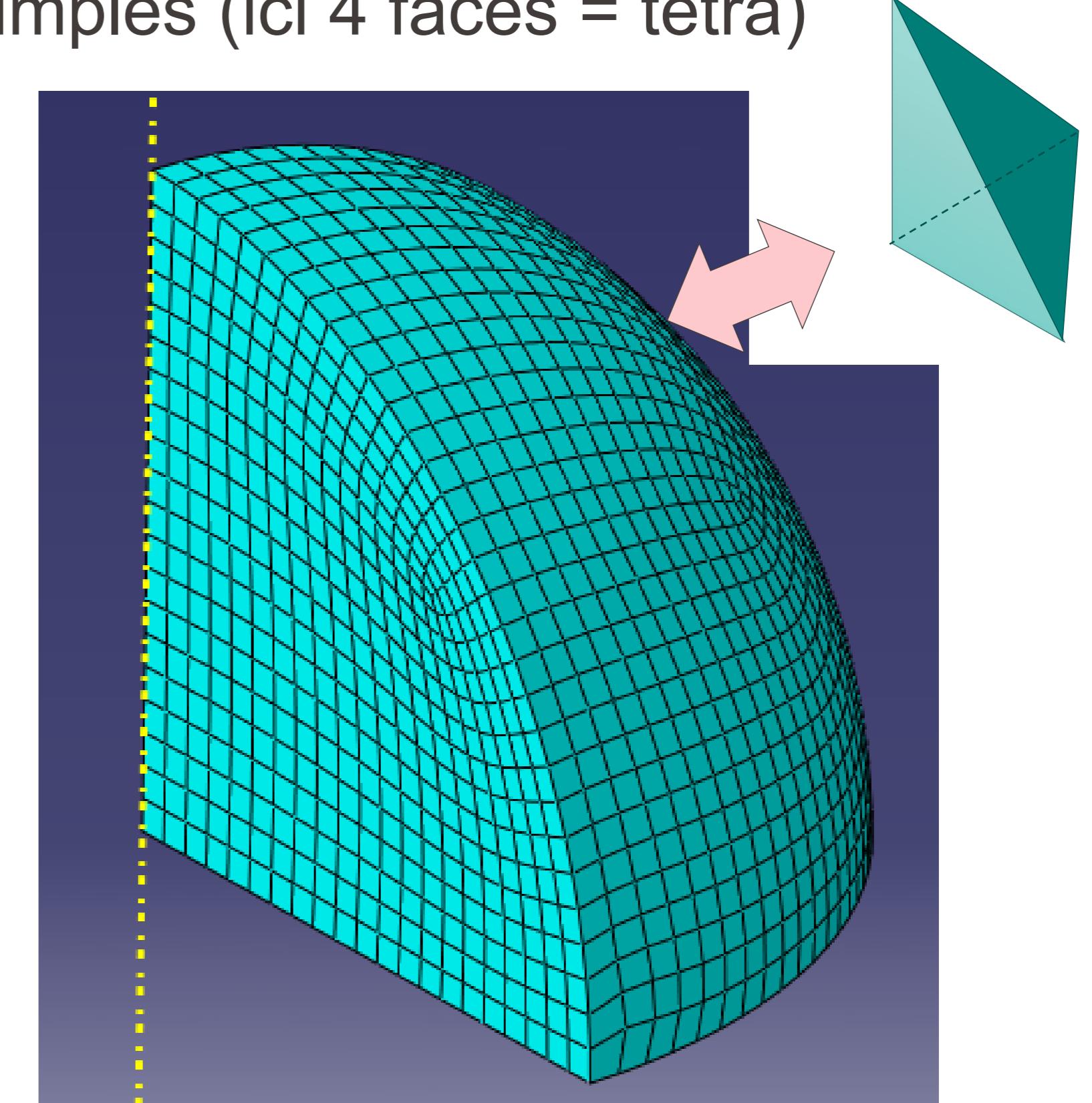
# Algorithmes de maillage 3D structuré

Modélisation et simulation par éléments finis

Hexaèdres : extrusion d'un maillage 2D structuré



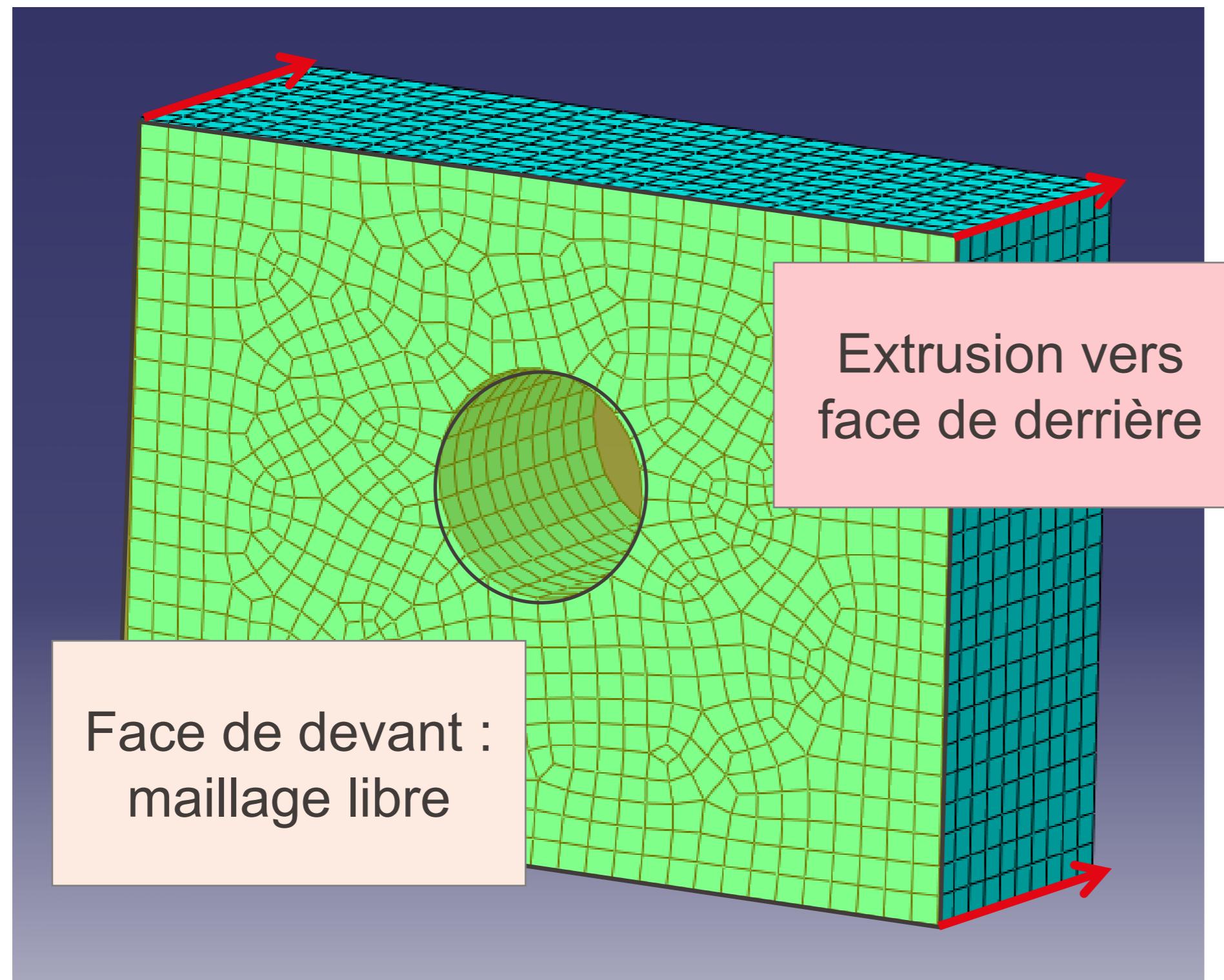
Hexaèdres : pour primitives 3D simples (ici 4 faces = tétra)



Ne fonctionne que pour volumes simples : volumes avec 4 faces, ou volumes d'extrusion avec 3, 4 ou 5 faces latérales.

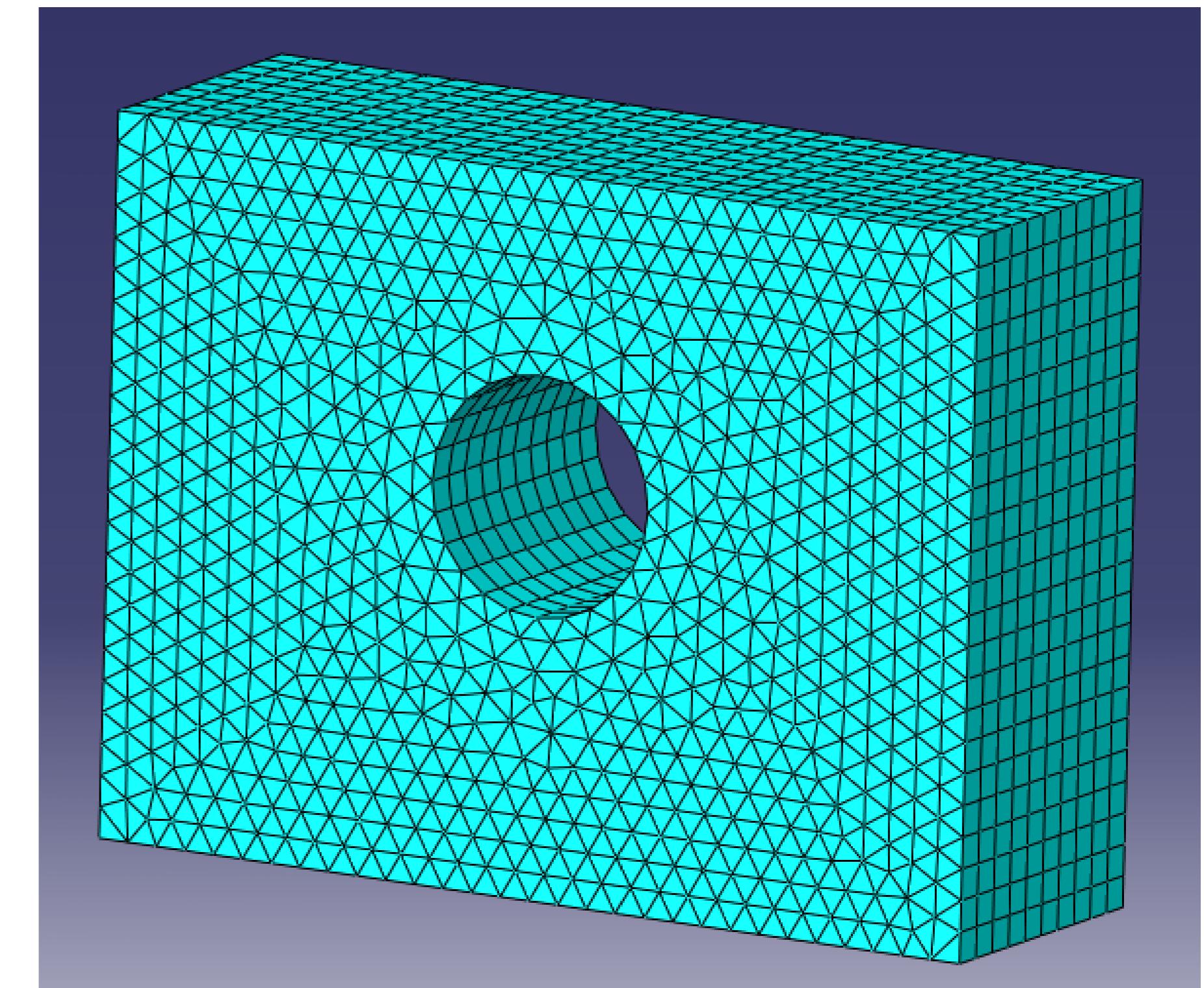
# Algorithmes de maillage 3D balayé (sweep)

Hexaèdres : maillage 2D libre  
quadrangles + extrusion



Modélisation et simulation par éléments finis

Prismes : maillage 2D libre  
triangles + extrusion

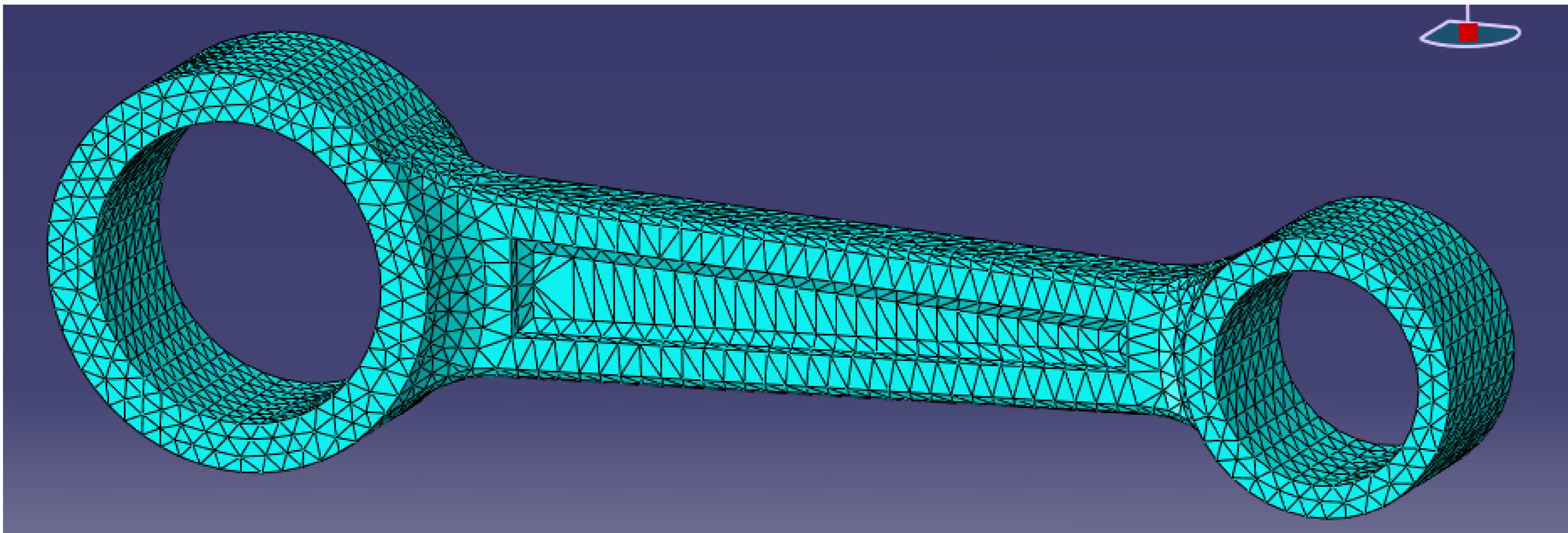


Seulement pour volumes d'extrusion / révolution : 2 faces similaires  
(devant et derrière) + faces latérales quadrangulaires.

# Algorithmes de maillage 3D libre

Maillage libre avec tétraèdres : maillage 2D libre triangles (2D advancing front) + génération libre de tétraèdres (3D advancing front).

Modélisation et simulation par éléments finis

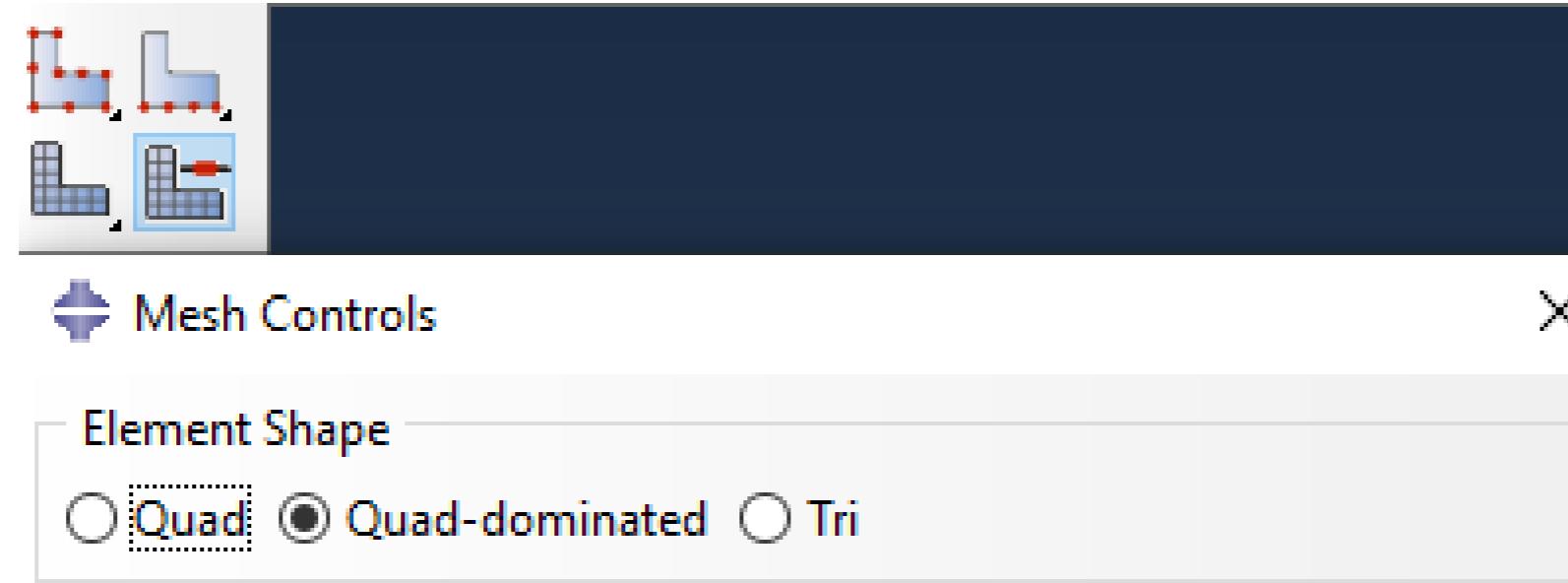


Algorithme **le plus général** dans Abaqus. Permet de mailler n'importe quelle géométrie. Très flexible pour raffiner localement.

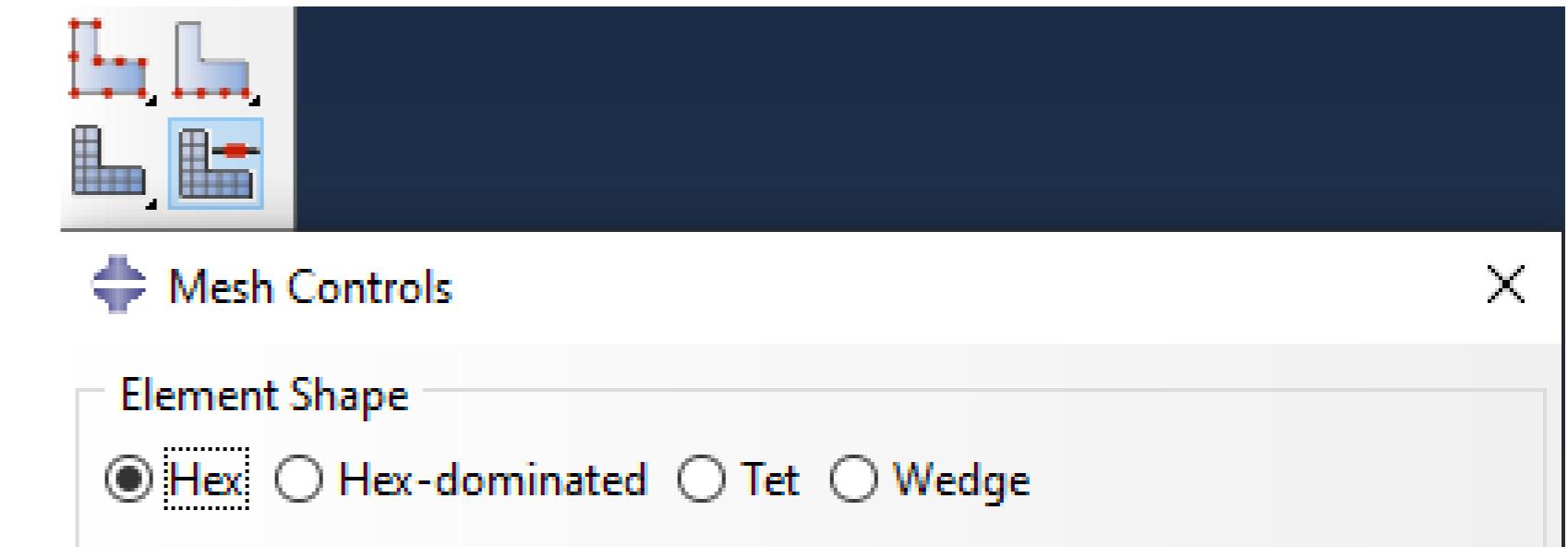
# Démonstration

Choix d'éléments :

2D

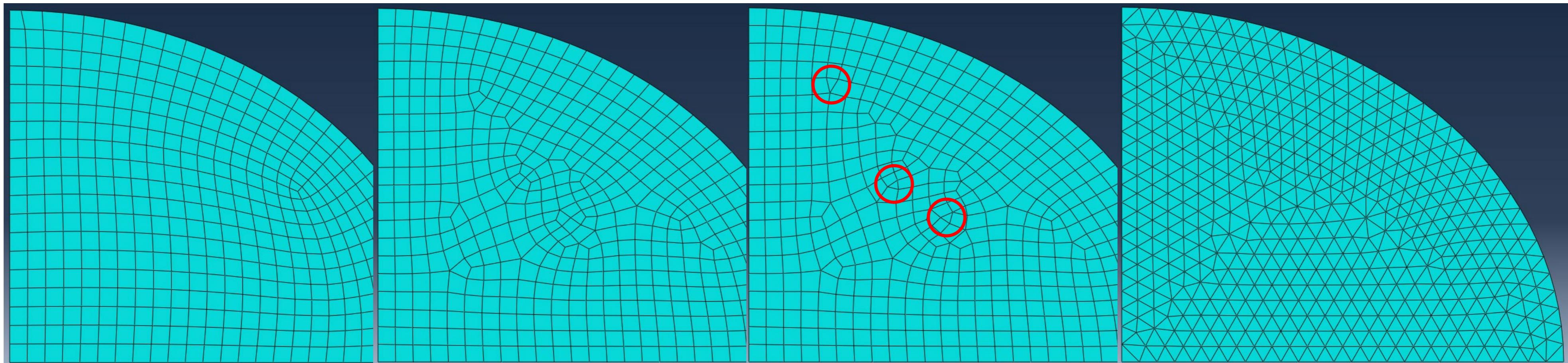


3D



Modélisation et simulation par éléments finis

Exemples 2D :



Quad structuré

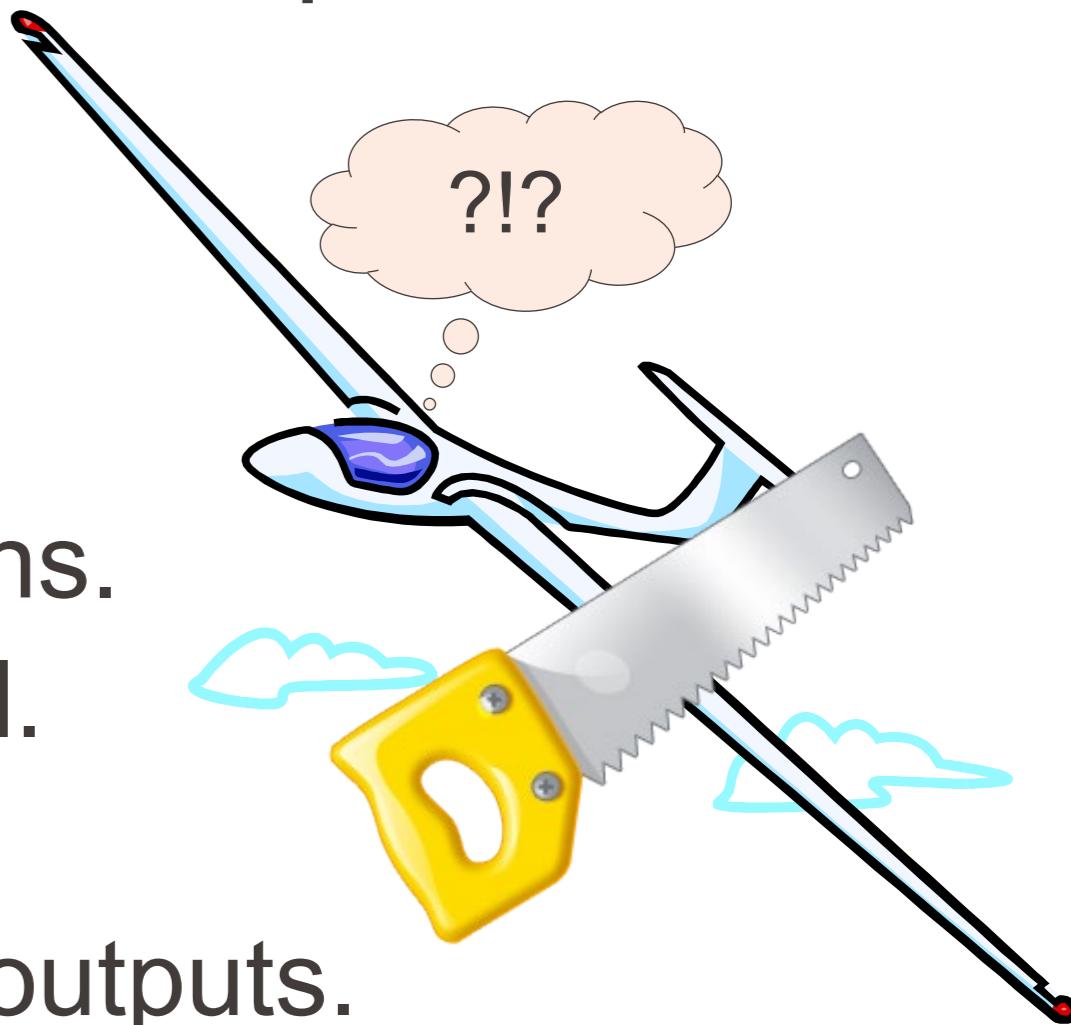
Quad libre

Quad-dominant

Tri libre

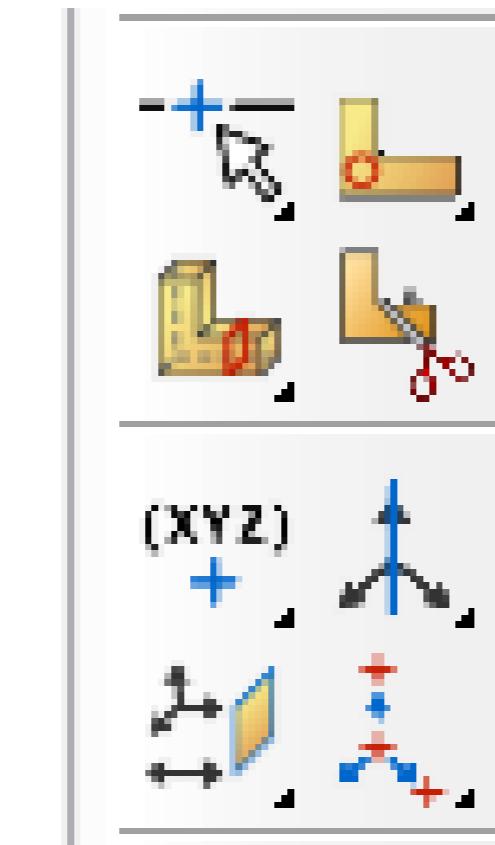
# Partitionnement

- But : **décomposer la géométrie** en volumes / faces plus simples.
- Méthode : couper les arêtes, faces et volumes avec des plans, extrusions, esquisses etc...
- Utile pour :
  - Utiliser **maillage structuré / balayé** dans certaines régions.
  - Améliorer la qualité du maillage, définir **raffinement local**.
  - Définir **matériaux différents** dans différentes régions.
  - Créer nouvelles faces / arêtes pour **conditions limites / outputs**.
  - Profiter des **symétries**.
  - Maintenir la **continuité** d'une pièce !
- Inconvénient : si mal utilisé, peut générer un grand nombre de petites faces / arêtes → génère petits éléments de mauvaise qualité.



# Partitionnement

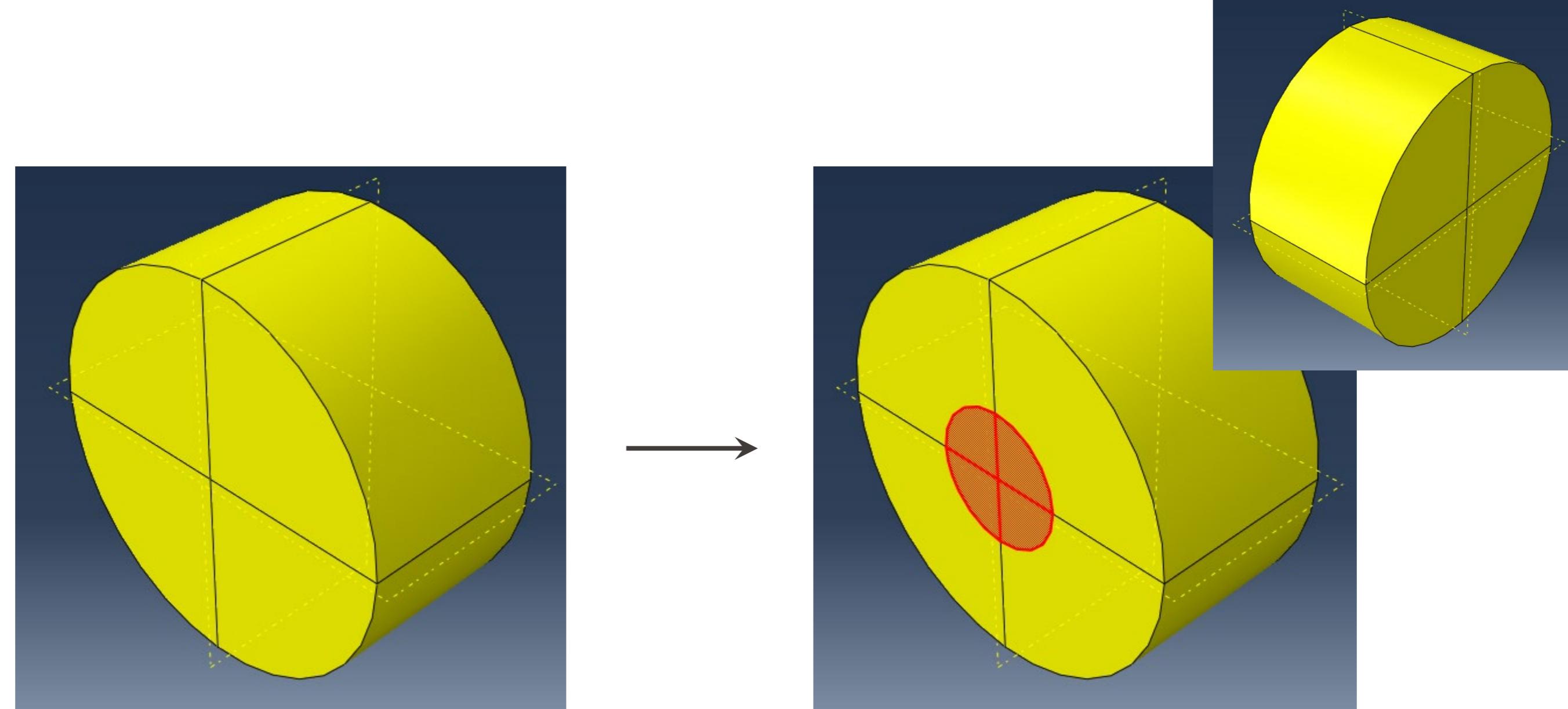
Modélisation et simulation par éléments finis



Outils de **partitionnement** (arêtes / faces / volumes)

Outils de **construction** (datum point / axis / plane / coordinate system)

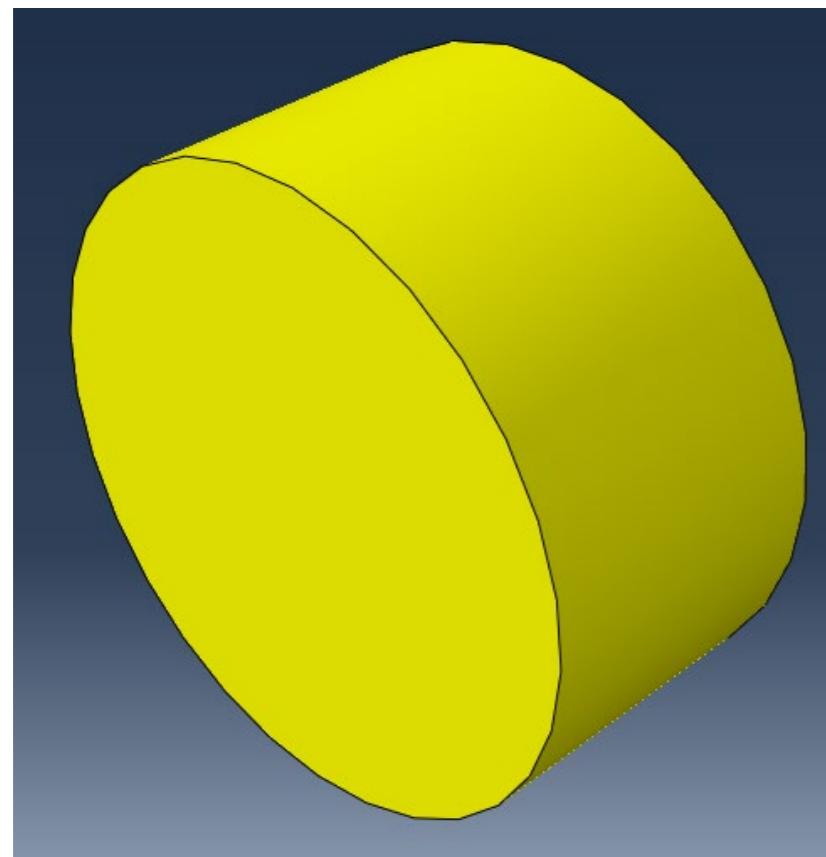
# Démonstration



Partitionnement de certaines faces (avec une esquisse)  
→ possibilité d'appliquer un chargement dans une sous-région.

# Démonstration

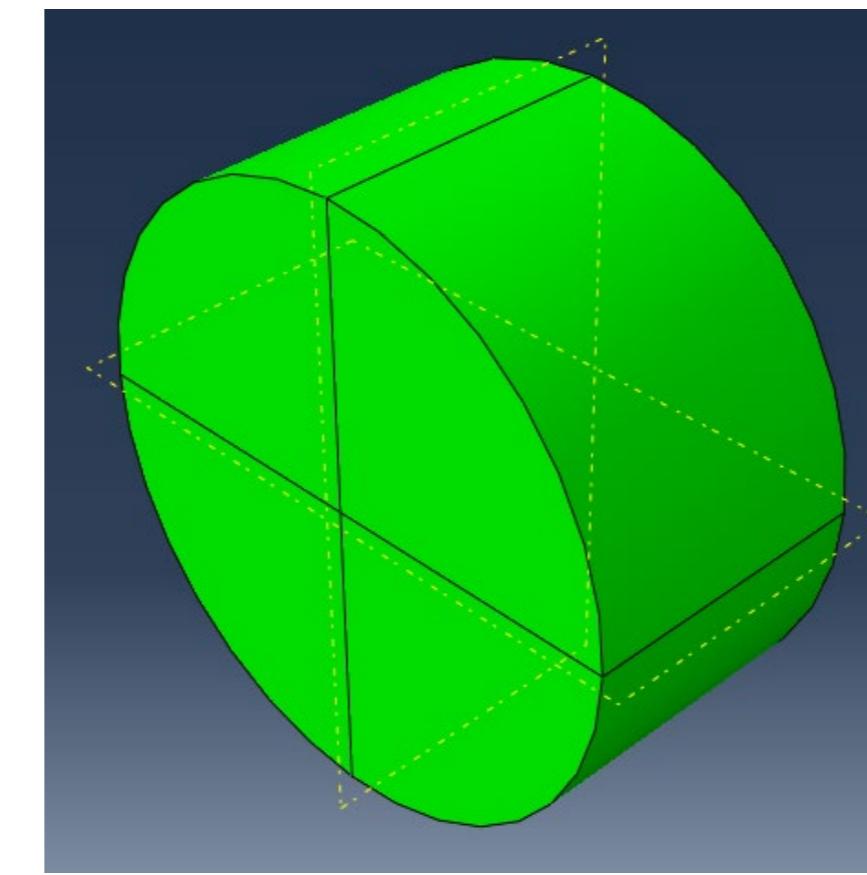
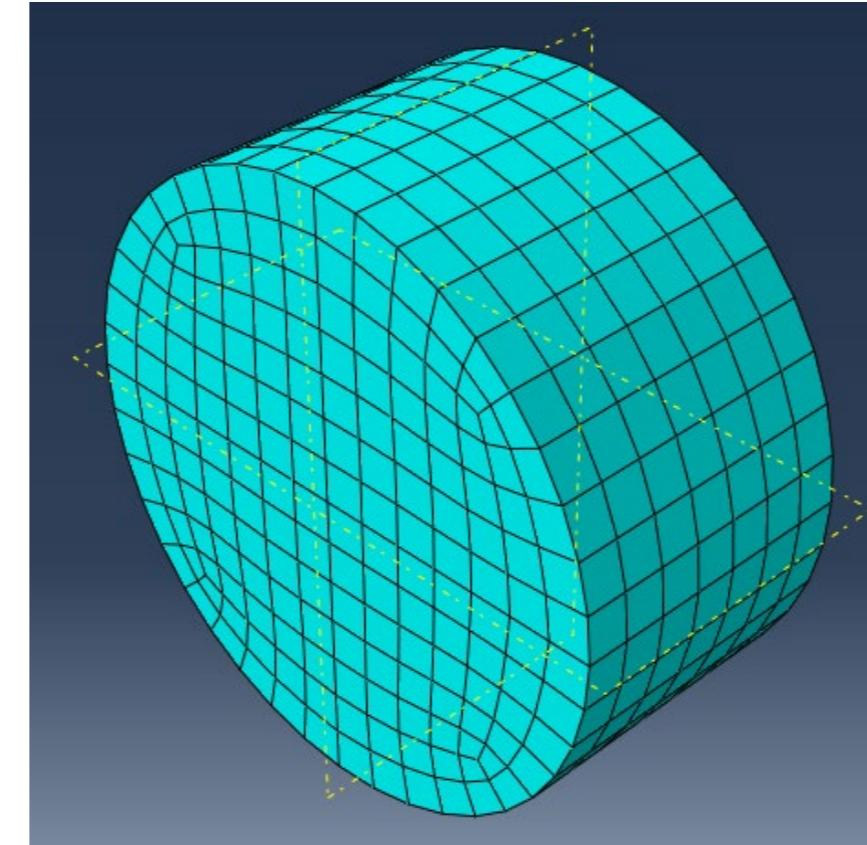
Modélisation et simulation par éléments finis



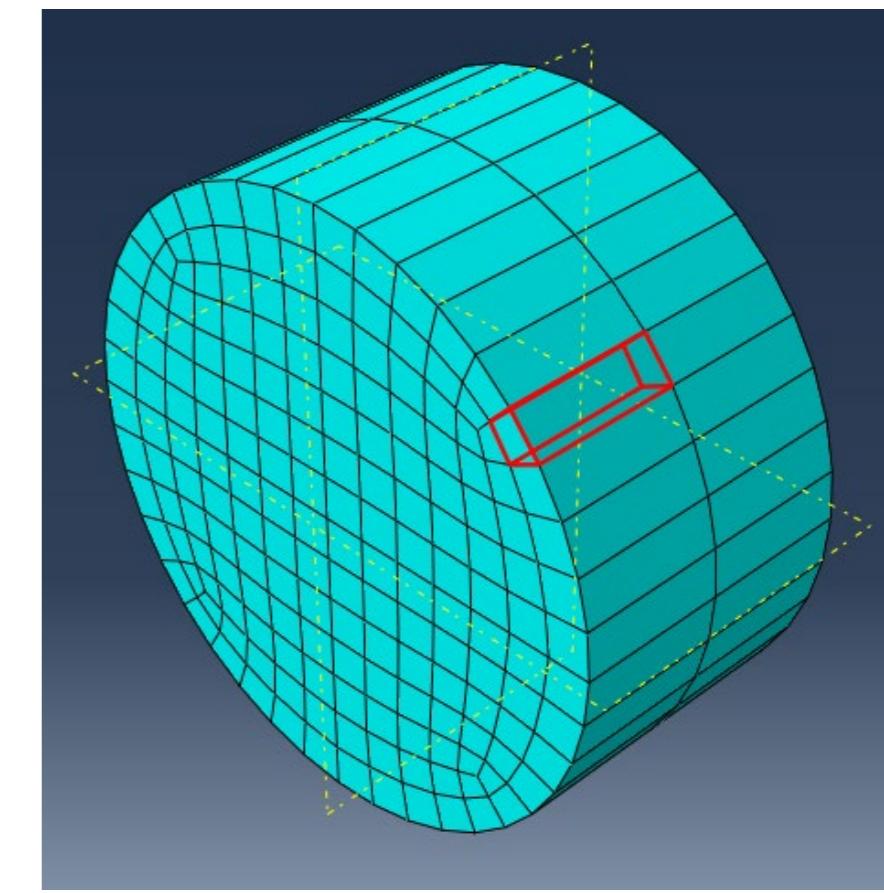
Pièce avec maillage hexa balayé



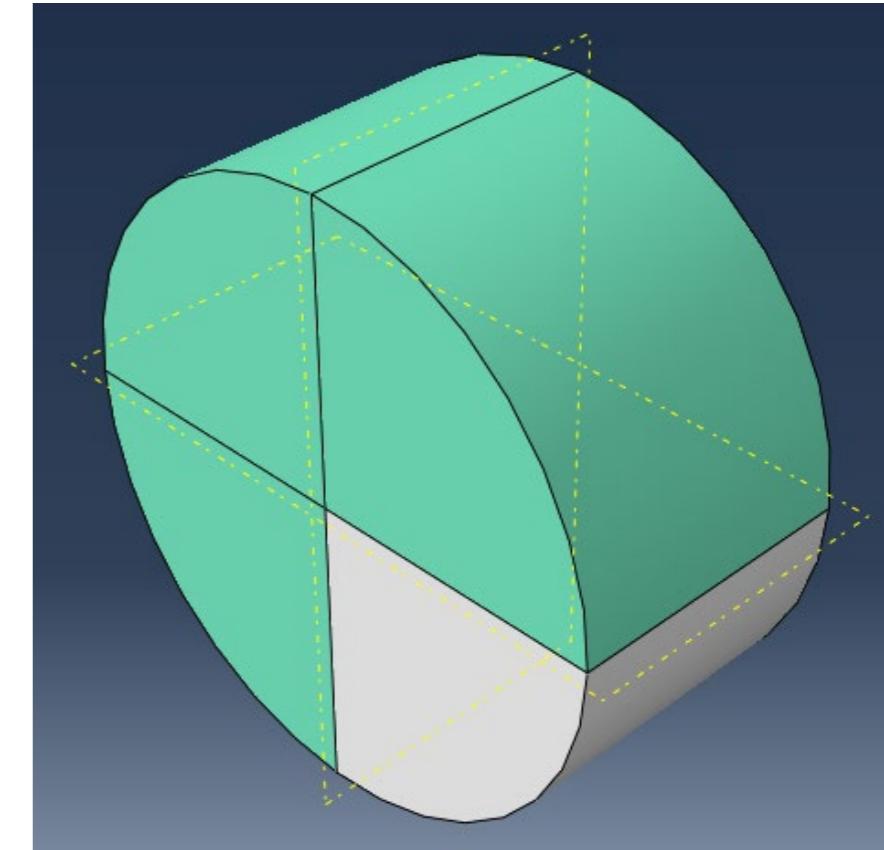
Partitionnement du volume en 4  
(avec 2 datum planes) → maillage hexa structuré possible



Possibilité d'utiliser éléments allongés

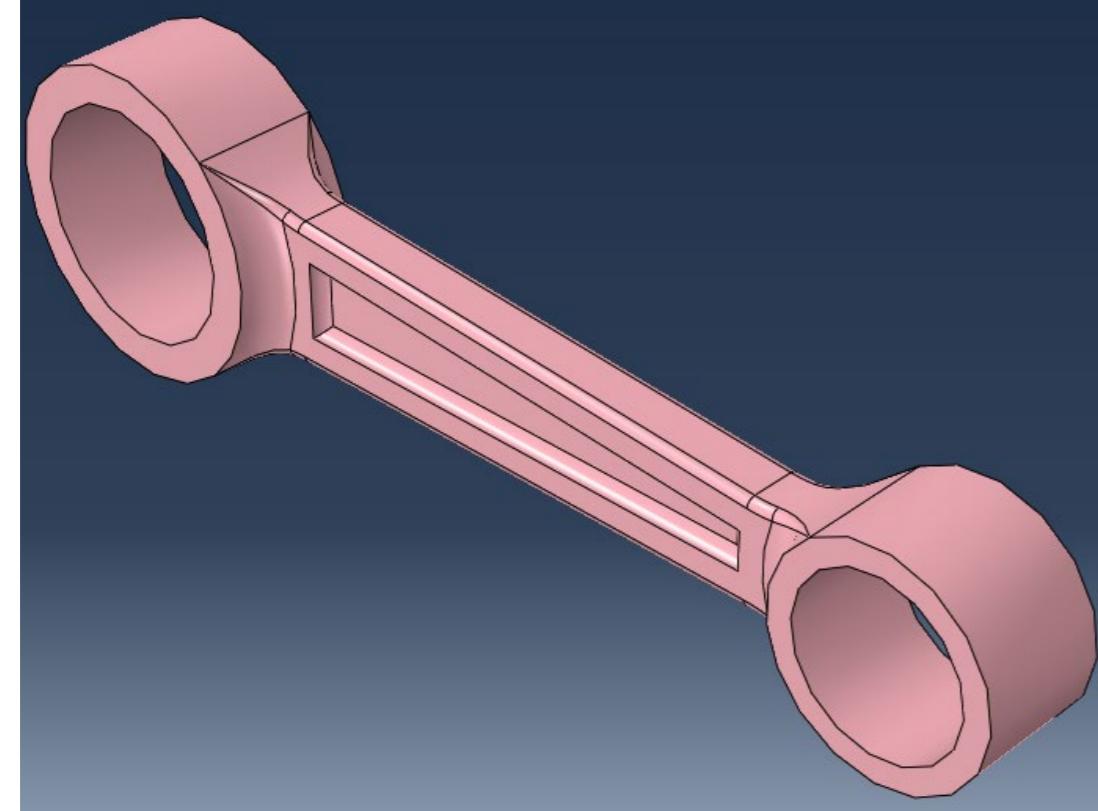


Possibilité d'appliquer des matériaux différents dans différents volumes

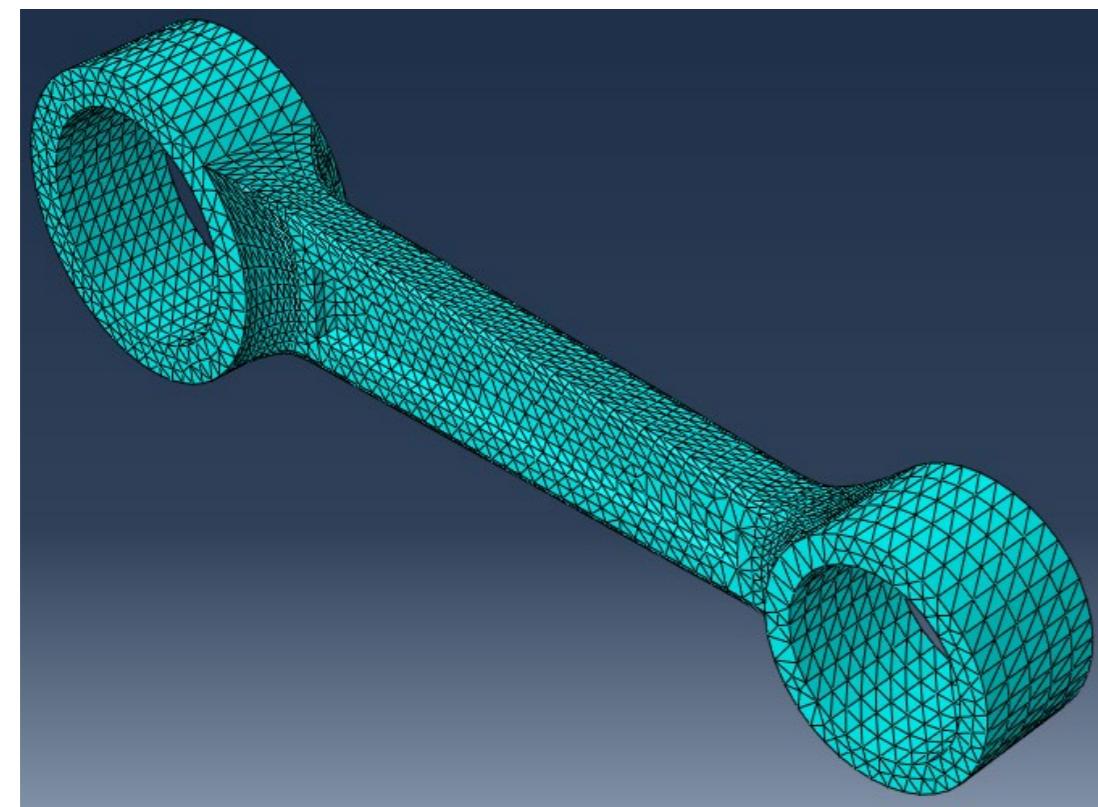


# Démonstration

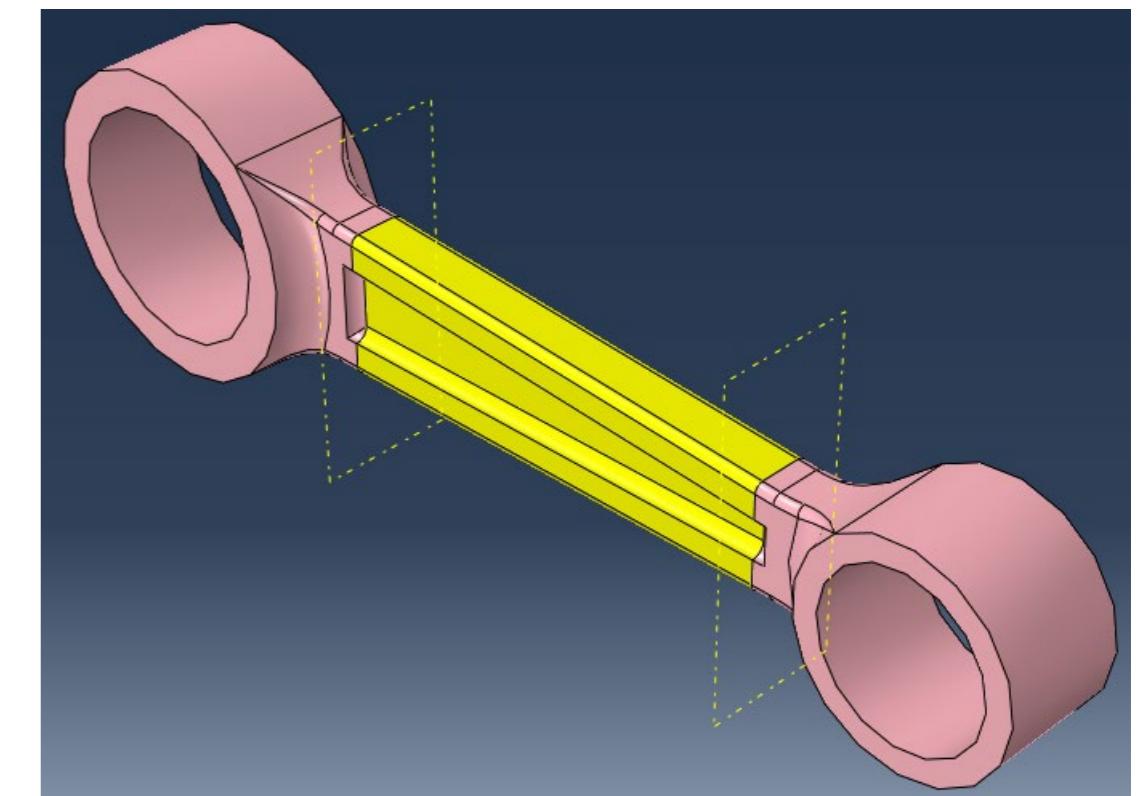
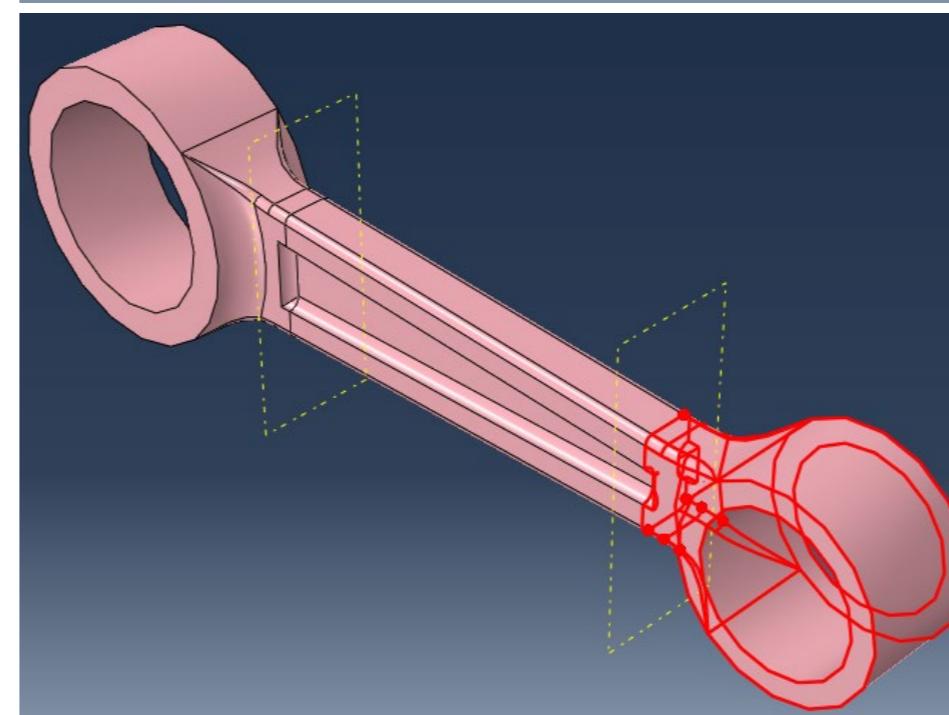
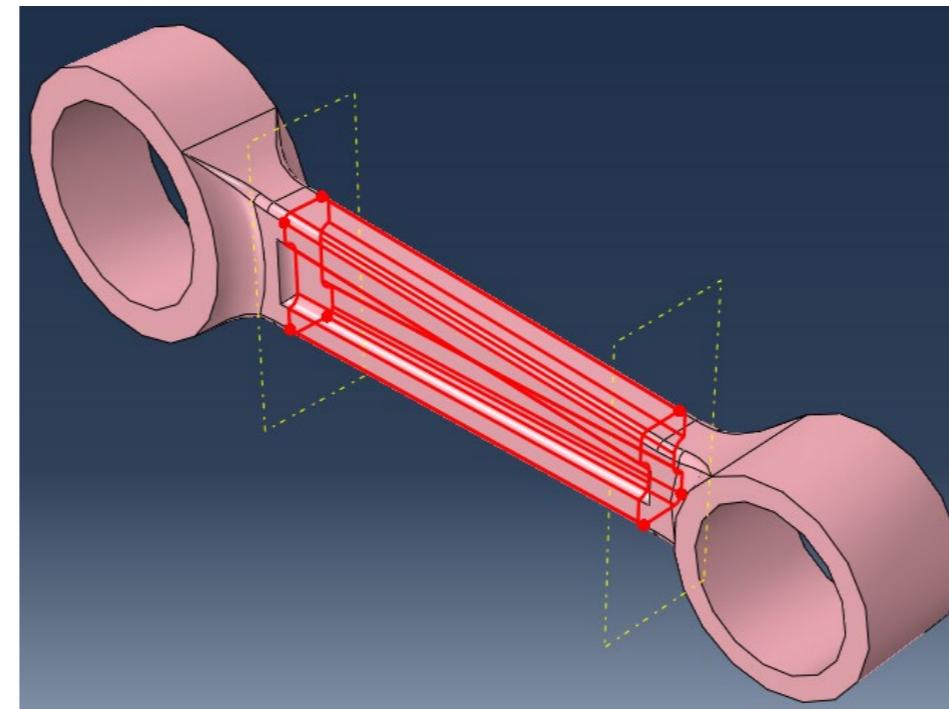
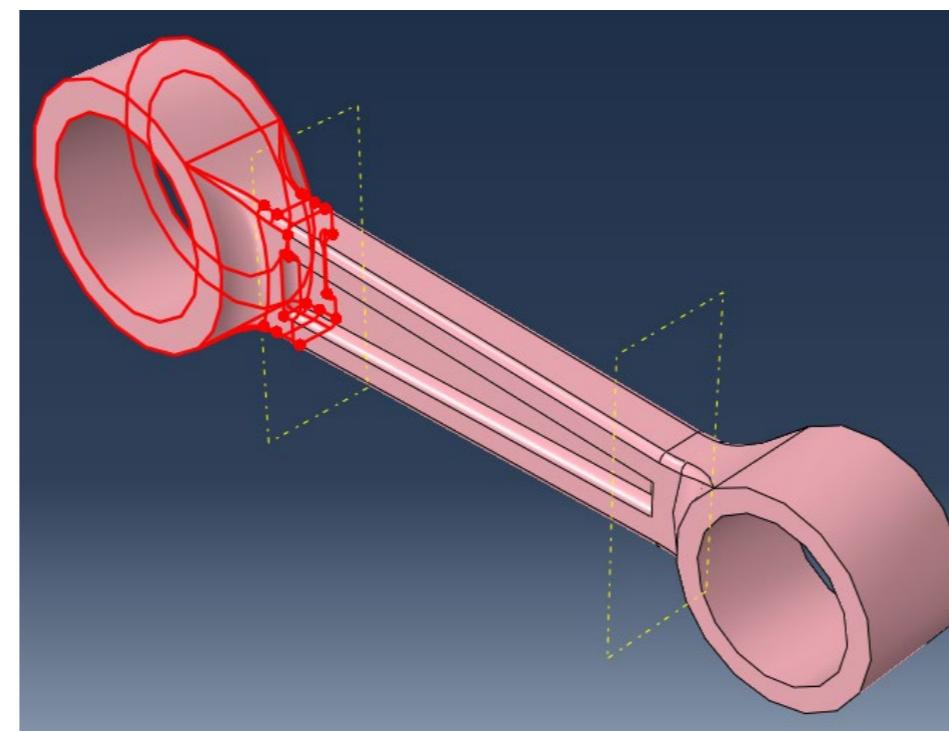
Modélisation et simulation par éléments finis



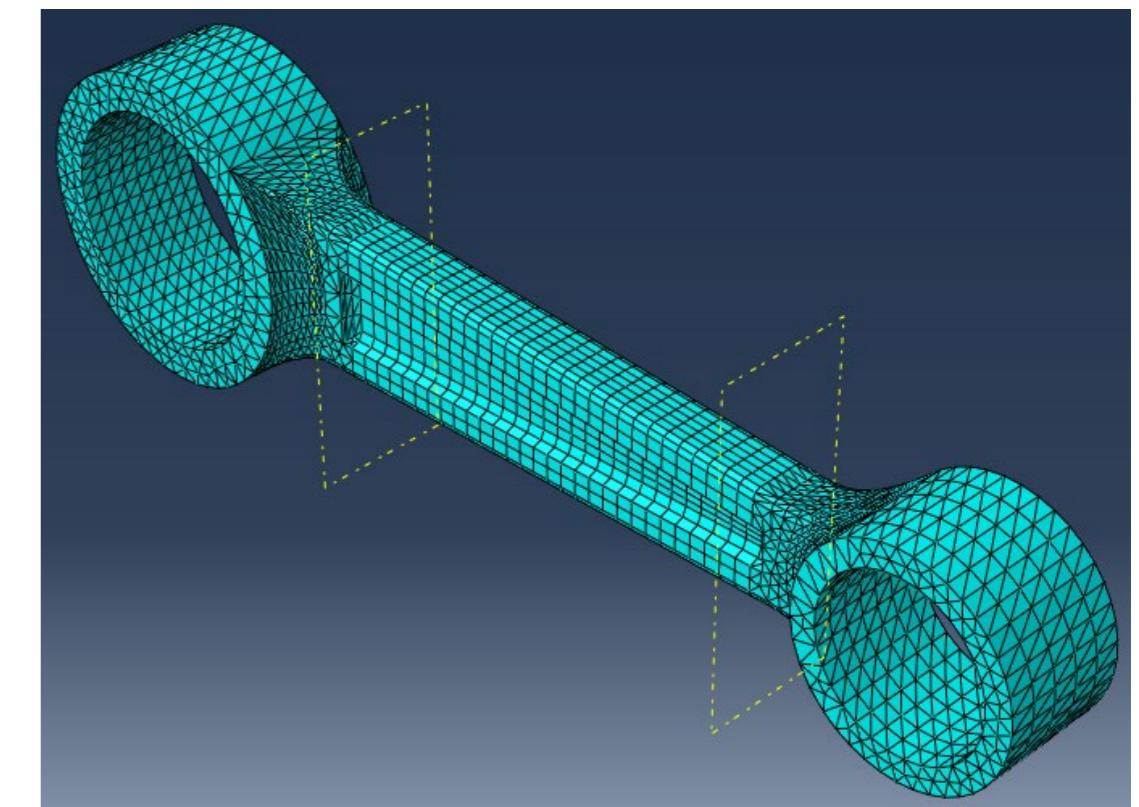
Pièce maillable avec tétra



Partitionnement du volume en 3  
(avec 2 datum planes)



Partie centrale maillable  
avec prismes



# CAD & maillage : problème de continuité

- La continuité du champ de déplacements est assurée par un maillage congruent : à une interface, tous les nœuds sont partagés.
- Un maillage congruent est généré si et seulement si les faces et arêtes sont partagées dans le modèle CAD. Quand on importe une géométrie avec plusieurs pièces, besoin de fusionner les faces & arêtes. Toujours vérifier les “free edges” !
- Des méthodes de maillage incompatibles peuvent générer des nœuds “suspendus” (hanging nodes) et/ou des discontinuités de déplacements. Exemple : transitions entre éléments linéaires et quadratiques ; transitions entre tétra et hexa.
- Quand il n'est pas possible d'avoir une interface partagée, il faut imposer la continuité des déplacements via des contraintes cinématiques → équations supplémentaires (à éviter autant que possible).

# CAD & maillage : problème de continuité

## Deux régions CAD déconnectées

Deux faces (superposées)

→ nœuds dupliqués

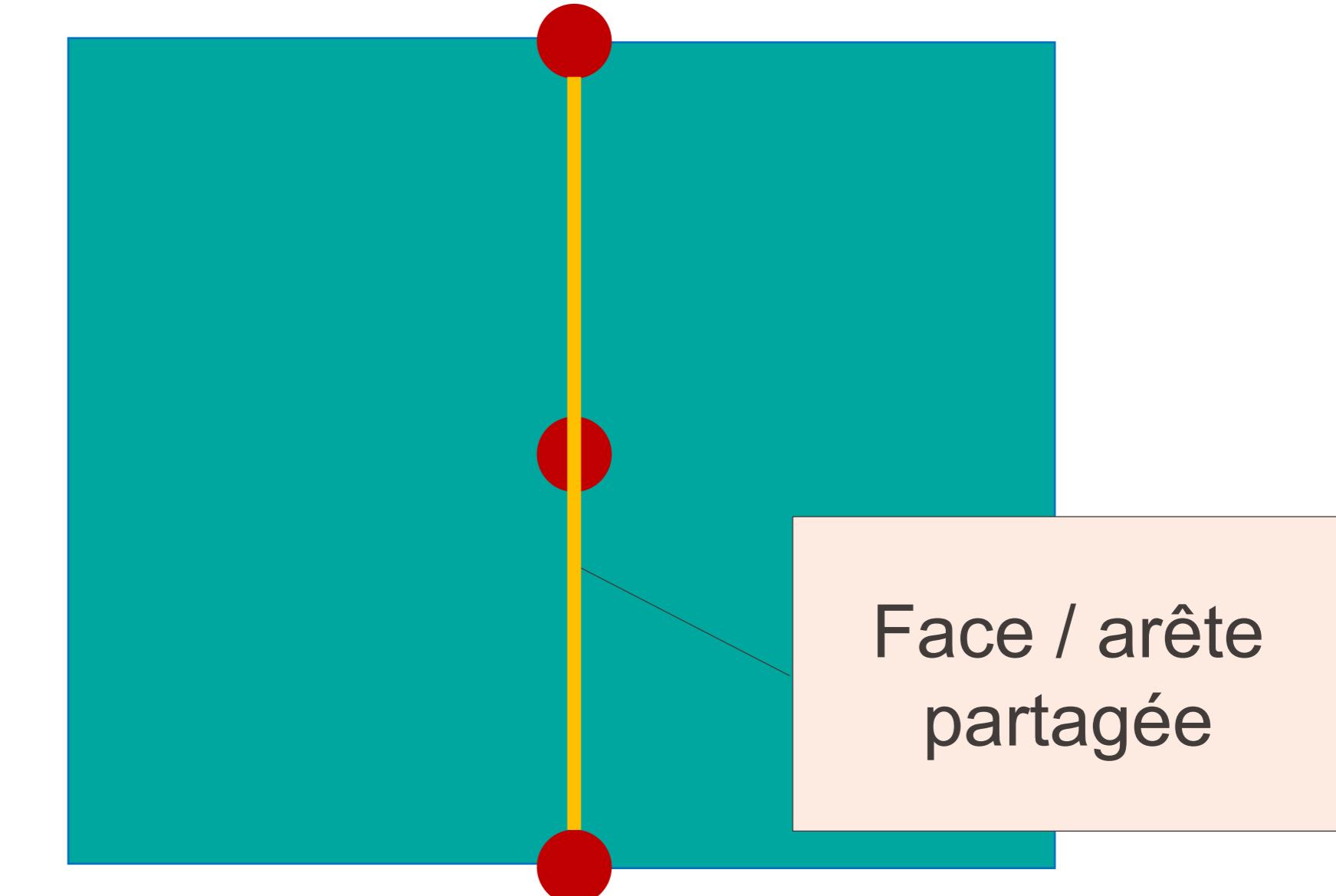
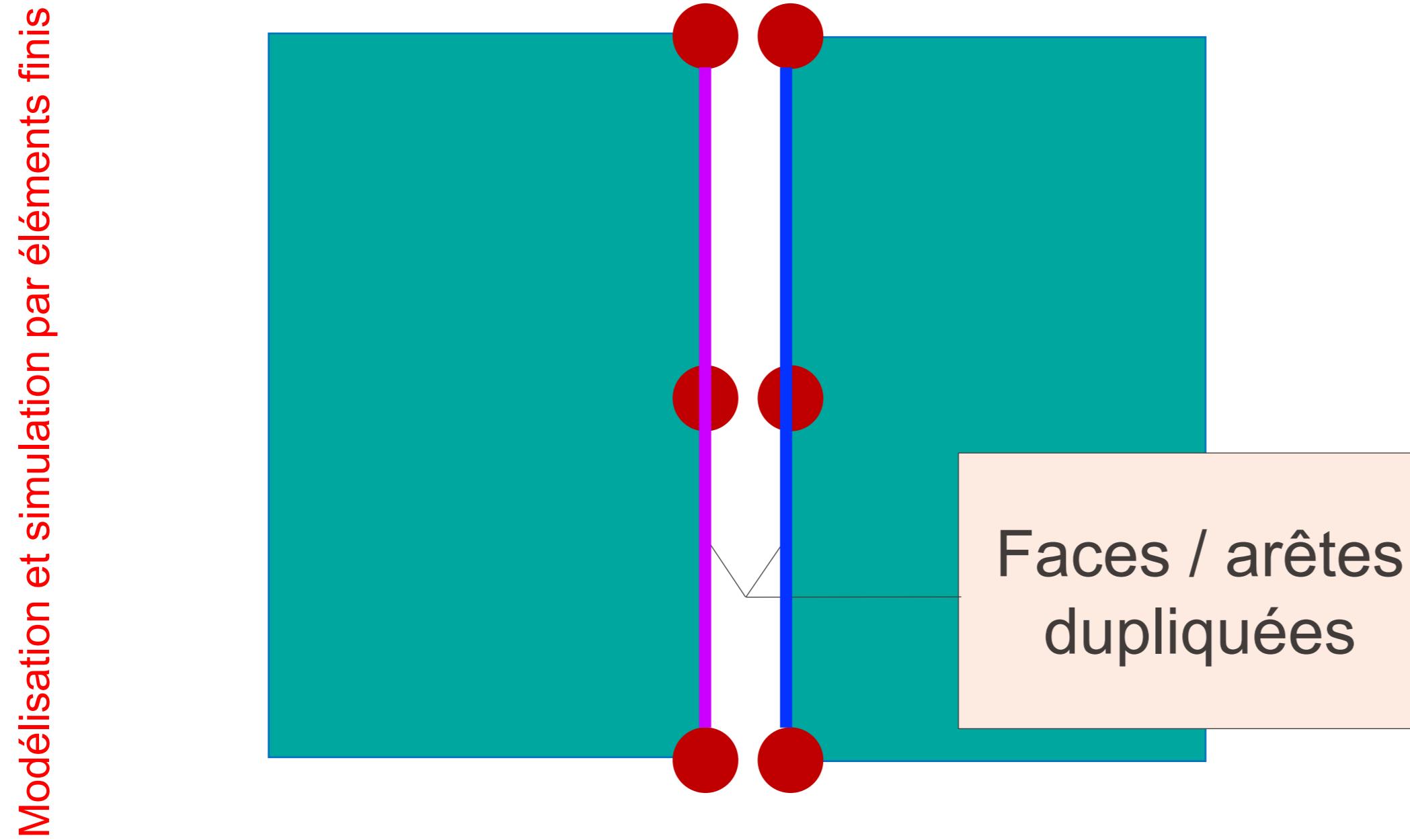
→ discontinuité des déplacements

## Deux régions CAD connectées

Une seule face

→ nœuds partagés

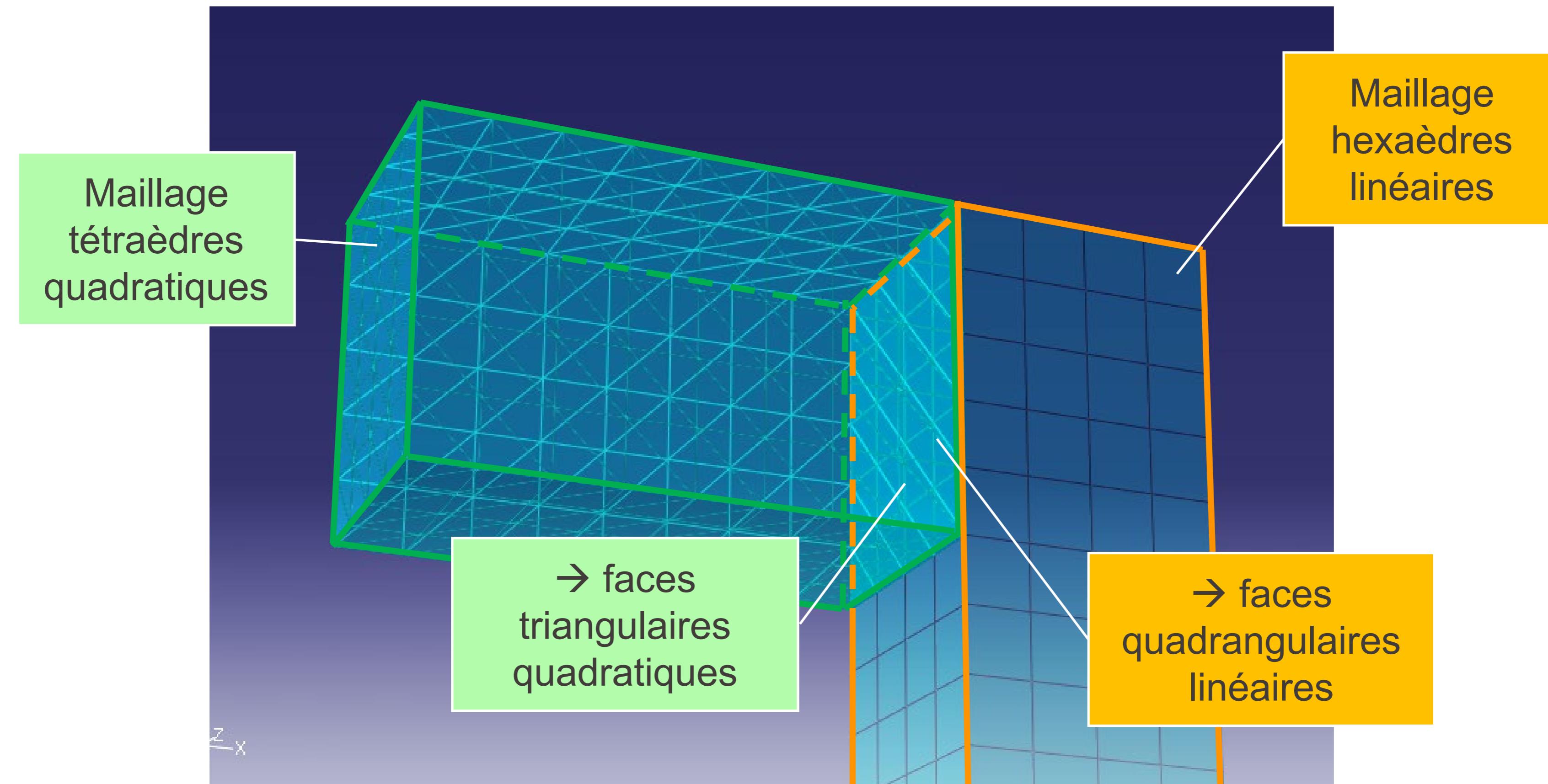
→ déplacements continus



# Maillages incompatibles

- Les faces du maillage doivent être similaires.
- Les fonctions de base doivent être compatibles.

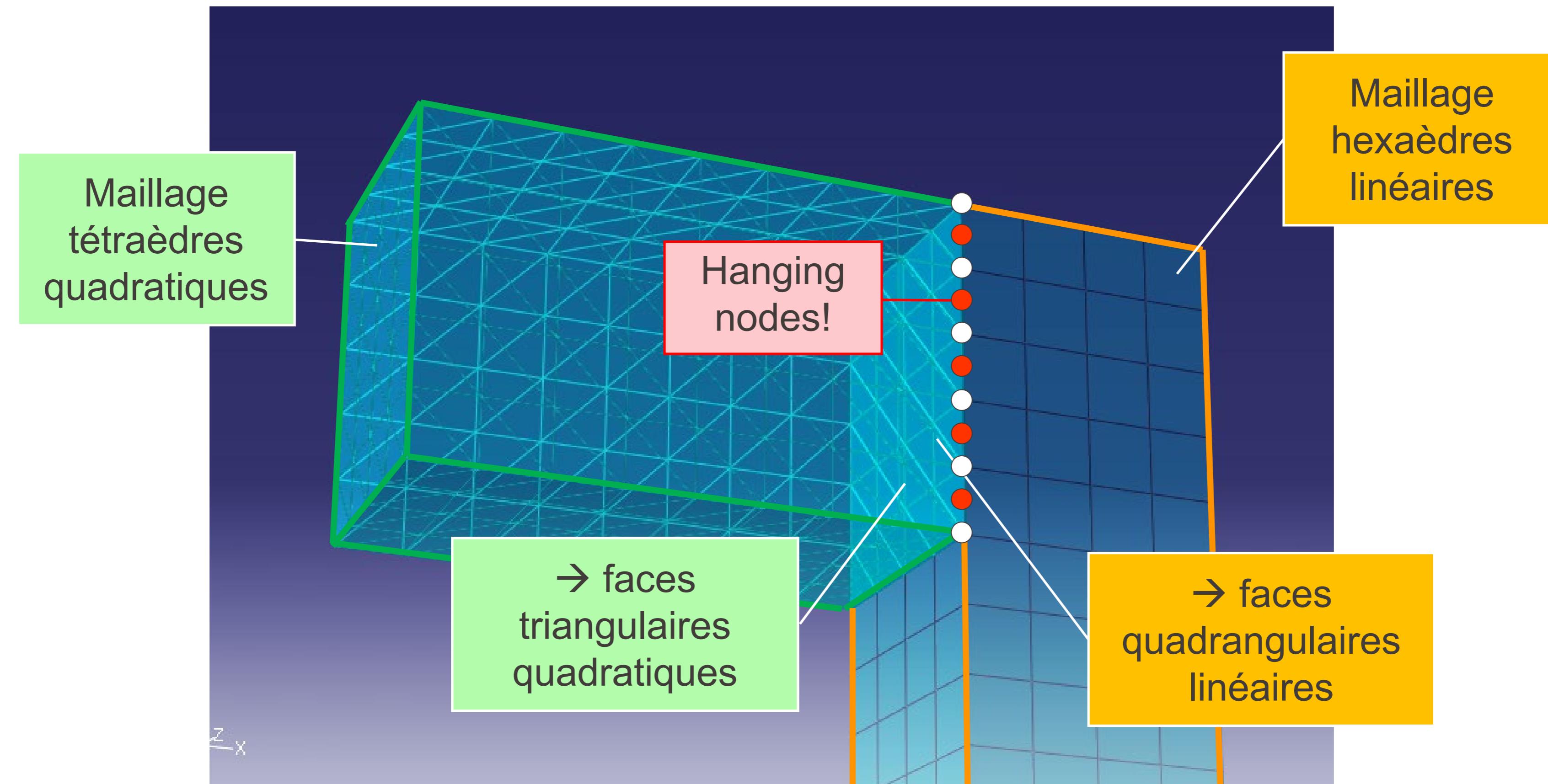
Modélisation et simulation par éléments finis



# Maillages incompatibles

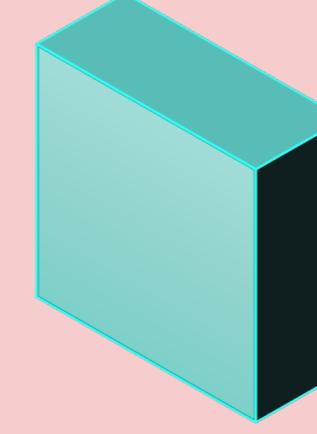
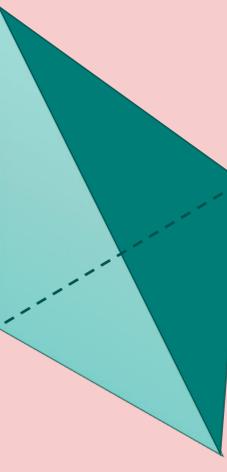
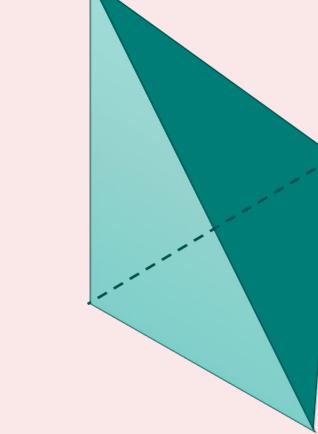
- Les faces du maillage doivent être similaires.
- Les fonctions de base doivent être compatibles.

Modélisation et simulation par éléments finis



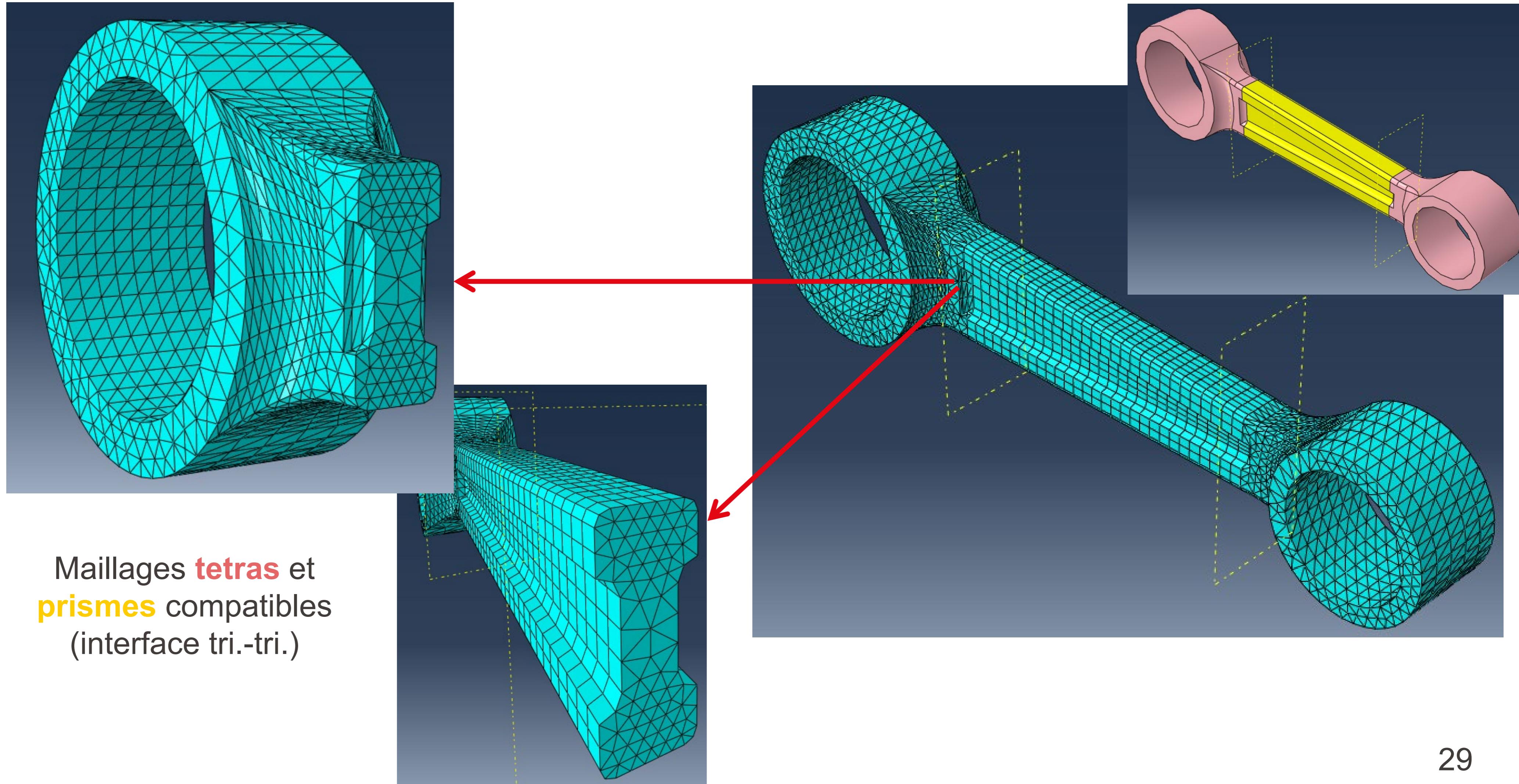
# Eléments compatibles

Modélisation et simulation par éléments finis

<p>... compatible avec...</p> 			
	<b>OUI</b>	<b>PARTIELLEMENT :</b> si interface = quad.	<b>NON</b>
	-	<b>OUI</b> (quad.-quad. ou tri.-tri.)	<b>PARTIELLEMENT :</b> si interface = tri.
	-	-	<b>OUI</b>

# Maillages et éléments compatibles : illustration

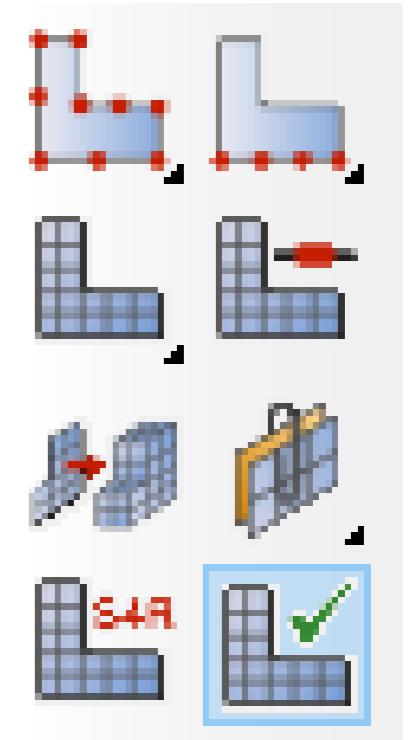
Modélisation et simulation par éléments finis



# Qualité du maillage

- **Critères :**

- Géométrie : distorsion, rapport d'aspect, angles min / max, etc.
- Eléments finis : le **jacobien doit être positif !**



- **Influence :**

- Mauvaise qualité = **mauvaise convergence** quand on raffine.
- Grandes **discontinuités** du champ de **contraintes**.
- Certains éléments peuvent se "bloquer" quand le rapport d'aspect est grand.
- Erreurs numériques d'arrondi et **singularités**.
- Résolution impossible **si le jacobien est négatif !**

- **Recommandations :**

- Souvent préférable d'avoir un maillage de **tétraèdres** quadratiques **de bonne qualité** plutôt qu'un maillage d'**hexaèdres très déformés** !
- Problèmes avec petites arêtes et jonctions quasi-tangentes (éléments très petits / "pointus" ) → "virtual topology" (ignore certaines arêtes pour le maillage).

# Recommandations CAD (logiciel CAD)

- Créer des pièces "propres" pour l'analyse EF :
  - Eviter **petites surfaces / arêtes**.
  - Eviter **jonctions quasi-tangentes** (angles très petits).
  - Essayer de minimiser le nombre de faces dans le modèle.
  - Préférer un "sweep" / "loft" simple aux opérations complexes de coupe / extrusion (→ permet d'utiliser un maillage structuré).
- Supprimer les détails géométriques qui ne sont pas importants :
  - **Evaluer quels détails sont importants pour le but de la simulation !**
  - Détails typiques : congés / chanfreins, petits trous / perçages, certains composants (vis, écrous, rivets etc.).
- **Pièces complexes / assemblages** : souvent très long de "réparer" la géométrie et de résoudre les problèmes de maillage → plus facile de reconstruire un modèle CAD propre pour l'analyse EF.

# Recommandations CAD et maillage (Abaqus)

- Toujours **vérifier** la géométrie (free edges, etc.).
- Si nécessaire : **réparer** la géométrie ou essayer un format différent.
- **Partitionner** pour créer volumes plus simples et utiliser symétries.
- Choix de la méthode de maillage (si possible) :  
Hexa structuré > Hexa balayé > Prisme balayé > Tetra libre
- Utiliser maillages **compatibles** aux interfaces !
- Vérifier la **qualité** du maillage (au minimum pas de "Analysis Error").
- Définir **raffinements locaux** où nécessaire.
- Utiliser "**virtual topology**" si nécessaire.
- Ne pas essayer à tout prix de créer un maillage hexa structuré. Temps mieux investi pour étude de convergence et raffinements locaux pertinents.