

Tutoriel : Maillage

Création de différents éléments sous Gmsh pour
l'utilisation du maillage sous Herezh

Gaëtan ROMAN

13/06/2017

Sommaire

I)	Introduction :	3
II)	Principe d'Herezh :	3
III)	Les éléments d'un maillage :	4
IV)	Création d'éléments sous Gmsh :	5
1)	Elément Bielles 1D	5
2)	Elément triangle 2D	6
3)	Elément quadrangle 2D	9
4)	Elément Tétraèdre 3D	12
5)	Elément Hexaèdre 3D	15
V)	Créer un maillage au format her :	17
VI)	Conclusion :	18

I) Introduction :

L'objectif de ce document est d'expliquer comment obtenir les différents éléments selon la dimension de l'espace sous Gmsh. Stamm peut être aussi utilisé mais pour des structures simples comme une plaque, une poutre, une barre, un tube etc...

Nous rappellerons aussi le principe d'Herezh et la méthode des éléments finis brièvement.

II) Principe d'Herezh :

La méthode des éléments finis est une méthode de résolution approchée d'équations aux dérivées partielles. Elle permet par calcul d'éléments finis d'obtenir une solution approchée de la solution. Cette méthode est mise en œuvre via des codes de calcul informatique (Herezh, Abaqus etc), cela est très utilisé industriellement.

Ce type de résolution permet de résoudre de nombreux problèmes physiques soumis à des conditions limites. Comme de la conduction thermique, de l'électromagnétisme, de la mécanique des solides etc.

On se restreint sur de la mécanique de solide. A l'aide de ces logiciels il est possible de résoudre divers problèmes tel que :

- Dimensionner des structures importantes
- Dimensionner des pièces de sécurité
- Simulation numérique d'un essai de traction par exemple
- Amélioration d'un composant
- etc

Herezh est un outil permettant de faire des simulations numériques, du dimensionnement. Ce logiciel de calcul par éléments finis est un solveur. Il utilise différents fichiers comme :

- Un fichier « nom.info » contenant la mise en donnée pour le type du calcul (loi de comportement, le maillage, les conditions limites, le chargement etc)
- Un fichier « nom.her » contenant le maillage de la structure.

Pour créer ce maillage il nous faut un mailleur comme gmsh (par exemple) nous fournissant un fichier « nom.msh ». Ce fichier créé est ensuite converti en un fichier « nom.her » par l'intermédiaire de msh2her.pl utilisable par Herezh.

Le logiciel gmsh peut nous créer des pièces volumiques, surfaciques, linéaires via le fichier « nom.geo » puis le mailler. Cependant, gmsh n'est pas un modéleur, il ne permet pas de créer des géométries complexes à partir de rien. Le logiciel gmsh est plus spécifiquement dédié à la création de maillage à partir d'une géométrie donnée. Il possède néanmoins quelques outils permettant de créer des géométries assez simples, et surtout de modifier des géométries déjà existantes pour les adapter de manière à obtenir au final un maillage correct. Pour créer une géométrie complexe il faut utiliser un modéleur 3D comme SolidWorks, FreeCAD... puis exporter la géométrie au format step par exemple, et enfin importer cette géométrie dans gmsh pour pouvoir la mailler. En effet, gmsh est capable de lire le format step.

A la fin d'un calcul par éléments finis il est possible de visualiser nos résultats via gmsh. En effet, Herezh nous fourni un fichier « nom.Cvisu » qui nous permet de sauvegarder les grandeurs à visualiser comme :

- des tableaux donnant différentes informations sur des grandeurs aux nœuds, aux points d'intégrations via gnuplot (tracé de courbe)
- des isovaleurs aux nœuds pour gmsh

III) Les éléments d'un maillage :

Un maillage est la discrétisation spatiale d'un milieu continu voire une modélisation géométrique d'un domaine par des éléments proportionnés finis et bien définis. L'objectif d'un maillage est de procéder également à une simplification d'un système par un modèle fini dans l'optique d'une simulation numérique.

Un maillage est défini par son repère, par ses points possédant des coordonnées et par des cellules. De plus, un maillage peut être caractérisé par sa dimension (1D, 2D et 3D), sa finesse, la géométrie des cellules, le degré de l'élément (degré des polynômes d'interpolation) qui est donc le degré du polynôme servant à décrire les côtes ou arrêtes des éléments.

Herezh peut utiliser différents éléments selon la dimension du maillage. Ce maillage est défini dans le fichier.her pour cela :

- Utilisation du logiciel stamm pour des structures simples
- Utilisation de gmsh puis de msh2her.pl pour obtenir les différents éléments ci-dessous :

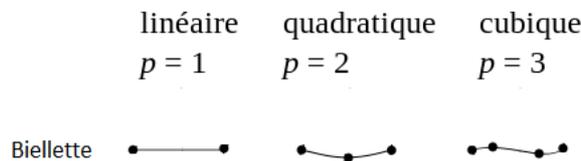


Figure 1: Élément 1D

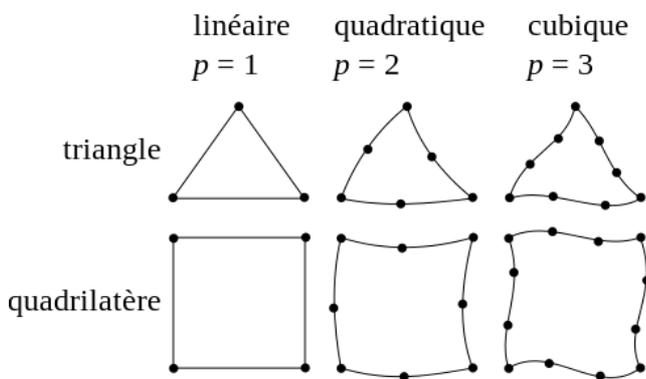


Figure 2: Éléments 2D

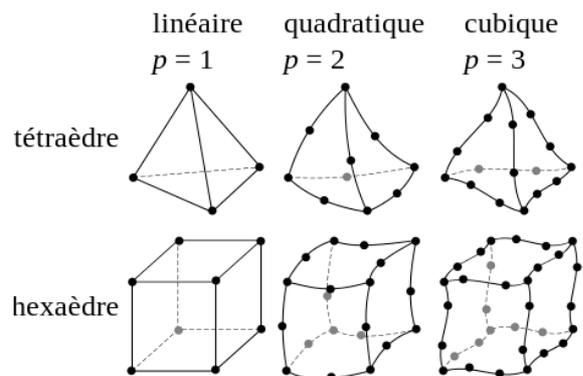


Figure 3: Éléments 3D

IV) Création d'éléments sous Gmsh :

Nous allons créer les différents éléments observés sur la figure 1, 2 ou 3 sous Gmsh.

1) Élément Biellettes 1D

Ci-dessous le fichier.geo permettant de créer une chaîne constituée de 4 éléments biellettes.

```
//+ Définitions des paramètres
ma = DefineNumber[ 10, Name "Parameters/maillage" ]; // taille du maillage

//+ Définition des points
Point(1) = {0, 0, 0, ma};
Point(2) = {10, 0, 0, ma};
Point(3) = {20, 0, 0, ma};
Point(4) = {30, 0, 0, ma};
Point(5) = {40, 0, 0, ma};

//+ Définitions des lignes
Line(1) = {1, 2};
Line(2) = {2, 3};
Line(3) = {3, 4};
Line(4) = {4, 5};

//+ Définition des références
Physical Line("chaîne") = {1, 2, 3, 4};
```

Paramètre permettant de régler la taille du maillage. Ici 1 nœud tous les 10 mm

On définit nos points

On relie nos points par des lignes

On crée une référence englobant toutes les lignes pour que le logiciel msh2her.pl puisse récupérer nos 4 éléments biellettes.

Seules les éléments appartenant à un groupe Physical, seront disponible dans le fichier de maillage crée par gmsh

Ensuite sous gmsh on peut mailler en 1D notre chaîne selon une interpolation linéaire (ordre 1), une interpolation quadratique (ordre 2) ou une interpolation cubique (ordre 3) comme suit :

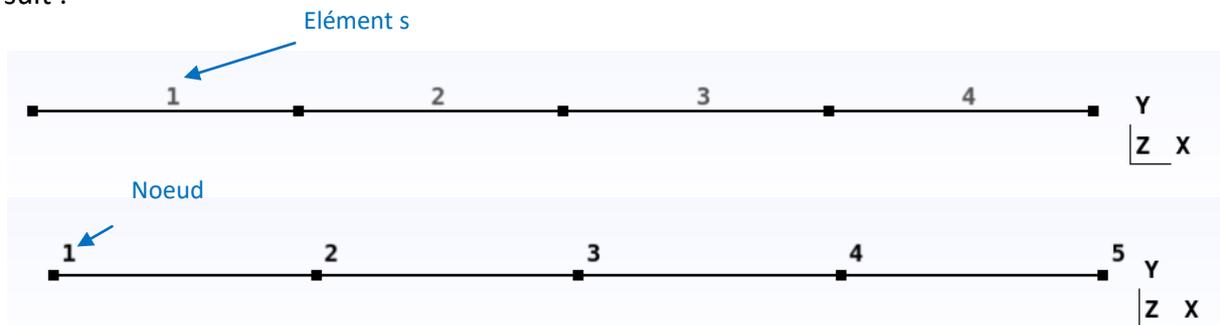


Figure 4: Eléments Biellettes d'ordre 1



Figure 5: Eléments Biellettes d'ordre 2

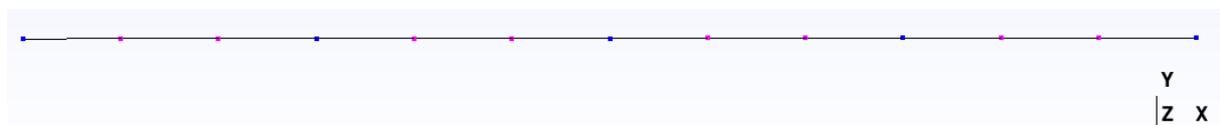


Figure 6: Eléments Biellettes d'ordre 3

2) Élément triangle 2D

Ci-dessous le fichier.geo permettant de créer une plaque en 2D composée d'éléments triangles non structurée :

```
//+ Paramètre
a = DefineNumber[ 30, Name "Parameters/longueur" ];
b = DefineNumber[ 20, Name "Parameters/largeur" ];
ma = DefineNumber[ 4, Name "Parameters/maillage" ];

//+ Définitions des points
Point(1) = {0, 0, 0, ma};
Point(2) = {a, 0, 0, ma};
Point(3) = {a, b, 0, ma};
Point(4) = {0, b, 0, ma};

//+ Définitions des lignes
Line(1) = {1, 2};
Line(2) = {2, 3};
Line(3) = {3, 4};
Line(4) = {4, 1};

//+ Définition de la surface
Line Loop(5) = {3, 4, 1, 2};
Plane Surface(6) = {5};

//Définitions des références
Physical Surface("Surface") = {6};
```

Paramètres de dimension pour nos points

On définit nos points

Paramètre permettant de régler la taille du maillage. Ici, 30/4 éléments selon la longueur et 20/4 éléments selon la largeur

On relie nos points par des lignes

On crée notre surface

On crée une référence englobant la surface pour que le logiciel msh2her.pl puisse récupérer nos éléments triangles

Ensuite sous gmsh on peut mailler en 2D notre plaque cependant comme nous le remarquerons il s'agit d'un maillage non structuré :

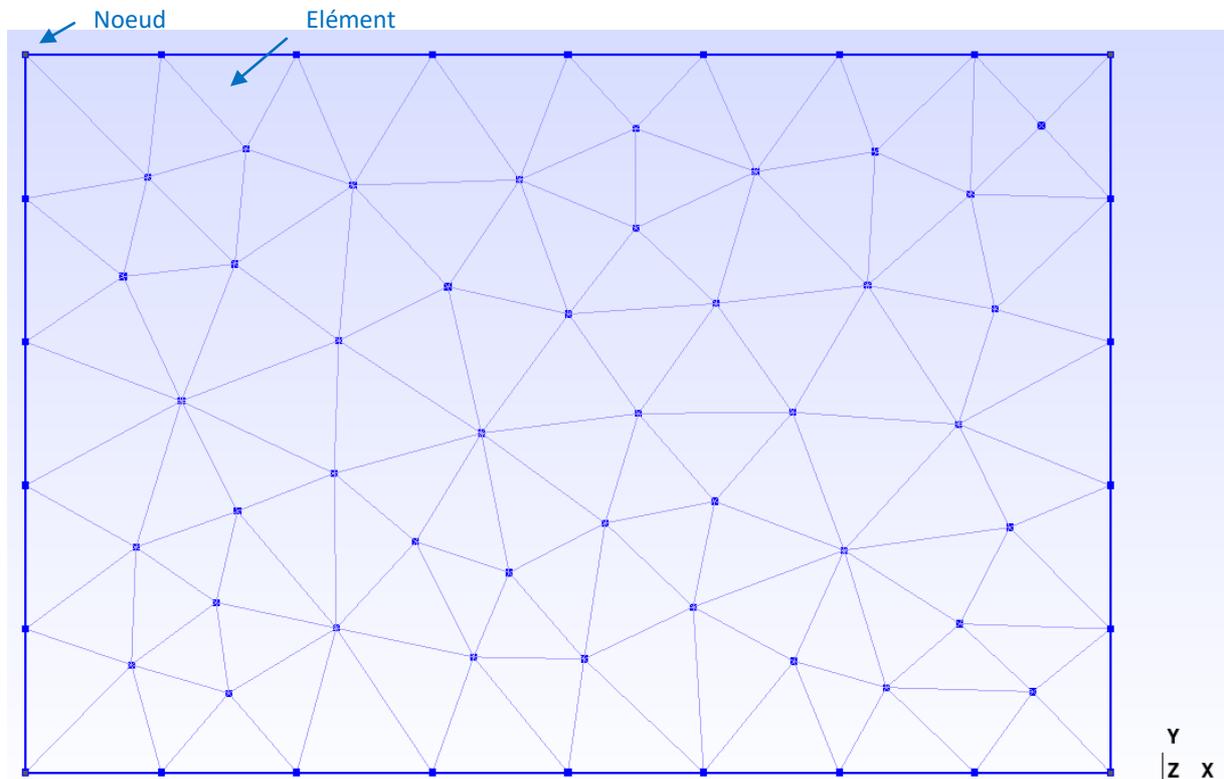


Figure 7: Éléments triangle non structuré d'ordre 1

En général, un maillage structuré permet d'obtenir des éléments avec un contour rectangulaire (quadrangle, hexaèdre) qui produisent de meilleurs résultats. Cependant ceci n'est vrai que dans le cas où il n'y a pas une trop grande distorsion des mailles. Un maillage structuré est également régulier ce qui apporte en général de meilleurs résultats (mais ce n'est pas systématique !)

- Dans le cas d'un maillage quelconque il faut définir la position de chaque nœud et la composition de chaque polygone ou polyèdre (la définition est dite explicite).
- Dans le cas d'un maillage régulier, la composition des polygones ou polyèdres voire la position des points peut être déduites d'une règle de construction (la définition est dite implicite car le maillage est régulier, il découle forcément d'une logique sans être énoncé formellement).

Ci-dessous le fichier.geo permettant de créer une plaque en 2D composée d'éléments triangles structurés :

```
//+ Paramètre
a = DefineNumber[ 30, Name "Parameters/longueur" ];
b = DefineNumber[ 20, Name "Parameters/largeur" ];
ma = DefineNumber[ 4, Name "Parameters/maillage" ];

//+ Définitions des points
Point(1) = {0, 0, 0, ma};
Point(2) = {a, 0, 0, ma};
Point(3) = {a, b, 0, ma};
Point(4) = {0, b, 0, ma};

//+ Définitions des lignes
Line(1) = {1, 2};
Line(2) = {2, 3};
Line(3) = {3, 4};
Line(4) = {4, 1};

//+ Définition de la surface
Line Loop(5) = {3, 4, 1, 2};
Plane Surface(6) = {5};

//+ Définition du maillage ajusté
Transfinite Line {2, 4} = 6 Using Progression 1;
Transfinite Line {1, 3} = 11 Using Progression 1;
Transfinite Surface {6};

//Définitions des références
Physical Surface("Surface") = {6};
```

Paramètres de dimension pour nos points

On définit les coordonnées des points

On relie nos points par des lignes

On crée notre surface

Définit le nombre d'élément +1 sur sa largeur (5 éléments)

Définit le nombre d'élément +1 sur sa longueur (10 éléments)

Permet de structurer le maillage sur toute sa surface

On crée une référence englobant la surface pour que le logiciel msh2her.pl puisse récupérer nos éléments triangles

Ensuite sous gmsh on peut mailler en 2D notre plaque composée d'éléments triangles structurés selon une interpolation linéaire (ordre 1), une interpolation quadratique (ordre 2) ou une interpolation cubique (ordre 3) comme suit :

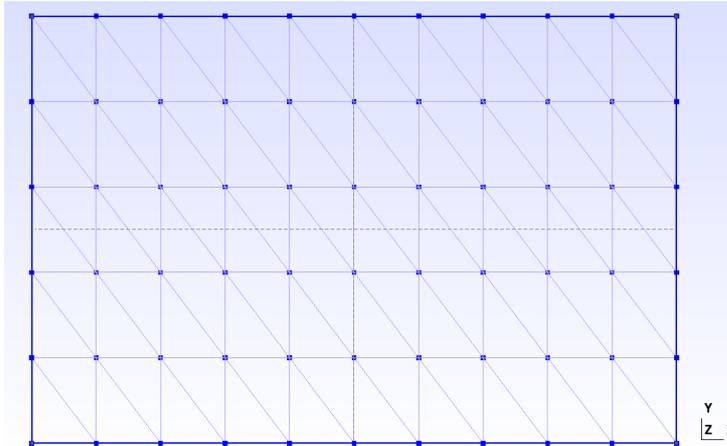


Figure 8: Eléments triangles structuré d'ordre 1

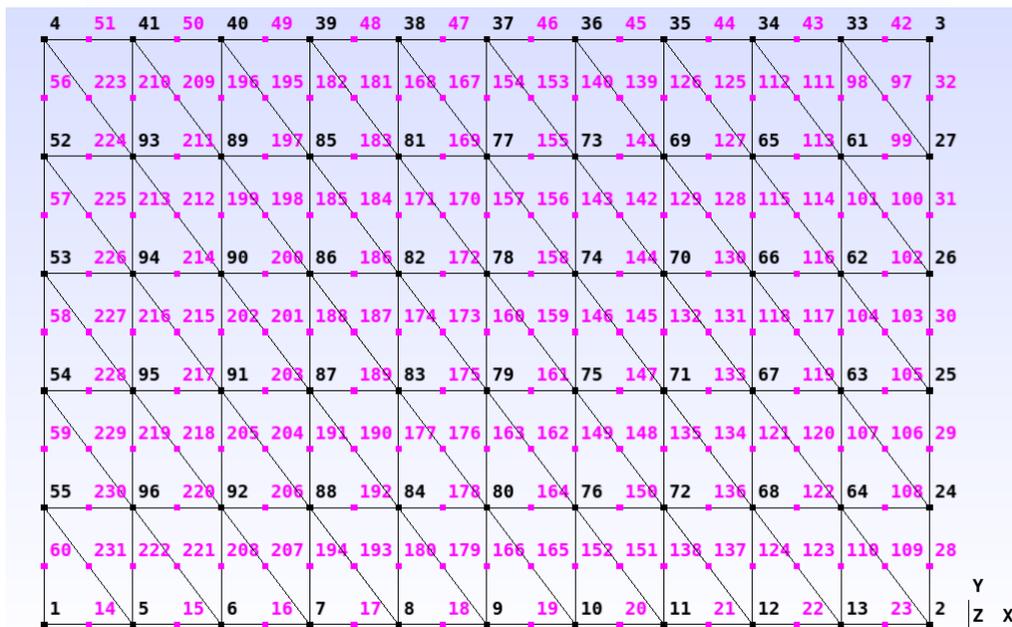
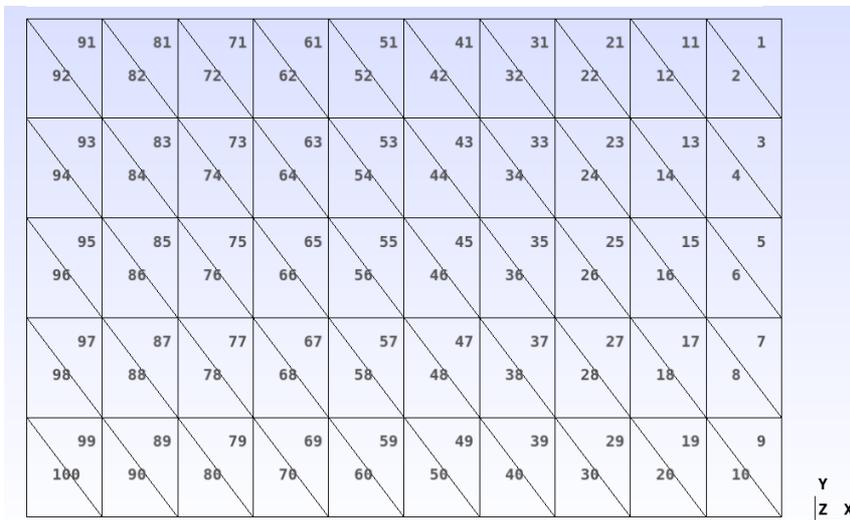


Figure 9: Eléments triangles structuré d'ordre 2

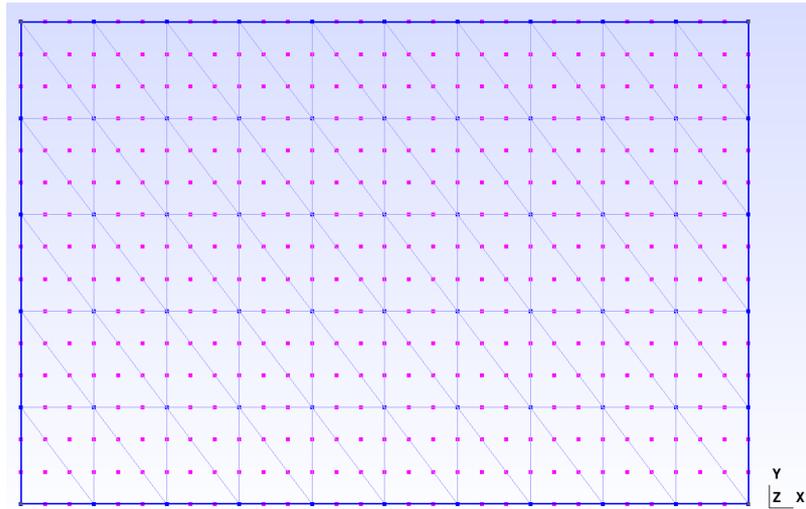


Figure 10: Eléments triangles structuré d'ordre 3

3) Elément quadrangle 2D

Ci-dessous le fichier.geo permettant de créer une plaque en 2D composée d'éléments quadrangles non structurée :

```
//+ Paramètre
a = DefineNumber[ 30, Name "Parameters/longueur" ];
b = DefineNumber[ 20, Name "Parameters/largeur" ];
ma = DefineNumber[ 4, Name "Parameters/maillage" ];

//+ Création de la plaque
Point(1) = {0, 0, 0, ma};

Extrude {a, 0, 0} {
  Point{1};
}

Extrude {0, b, 0} {
  Line{1};
}

//+ Définition du maillage
Recombine Surface {5};

//+ Définitions des références
Physical Surface("Surface") = {5};
```

Paramètres de dimension pour la création de la plaque

On définit la coordonné du point

On extrude notre point. Cela nous crée une ligne

On extrude notre ligne. Cela nous crée une surface

Permet d'obtenir des éléments quadrangles sur toute notre surface

Paramètre permettant de régler la taille du maillage. Ici, 30/4 éléments selon la longueur et 20/4 éléments selon la largeur

On crée une référence englobant la surface pour que le logiciel msh2her.pl puisse récupérer nos éléments quadrangles

Ensuite sous gmsh on peut mailler en 2D notre plaque cependant comme nous le remarquerons ce maillage ne sera pas structuré :

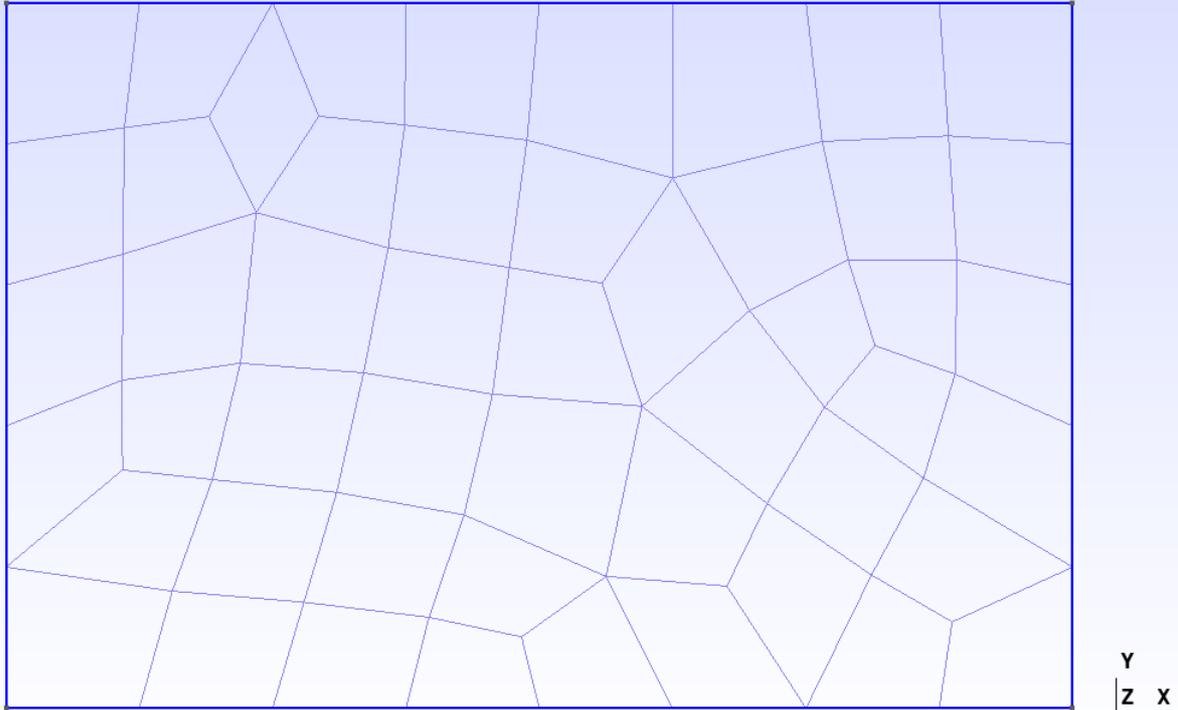


Figure 11: Eléments quadrangles non structurés

Ci-dessous le fichier.geo permettant de créer une plaque en 2D composée d'éléments quadrangles structurés :

```
//Paramètre
a = DefineNumber[ 30, Name "Parameters/longueur" ];
b = DefineNumber[ 20, Name "Parameters/largeur" ];
ma = DefineNumber[ 4, Name "Parameters/maillage" ];
ta = DefineNumber[ 8, Name "Parameters/tranche_a" ];
tb = DefineNumber[ 4, Name "Parameters/tranche_b" ];

//+ Création de la plaque
Point(1) = {0, 0, 0, ma};

Extrude {a, 0, 0} {
  Point{1}; Layers{ta};Recombine;
}
Extrude {0, b, 0} {
  Line{1}; Layers{tb};Recombine;
}

//+ Définitions des références
Physical Surface("Surface") = {5};
```

Paramètres de dimension pour la création de la plaque

Nombre d'élément selon la longueur

Nombre d'élément selon la largeur

On définit la coordonné du point

On extrude notre point. Cela nous crée une ligne. En ajoutant la dépendance du nombre d'élément sur sa longueur.

Recombine permet d'avoir des éléments quadrangles

On extrude notre ligne. Cela nous crée une surface. En ajoutant la dépendance du nombre d'élément sur sa largeur

On crée une référence englobant la surface pour que le logiciel msh2her.pl puisse récupérer nos éléments quadrangles

Ensuite sous gmsh on peut mailler en 2D notre plaque composée d'éléments quadrangles structurés selon une interpolation linéaire (ordre 1), une interpolation quadratique (ordre 2) ou une interpolation cubique (ordre 3) comme suit :

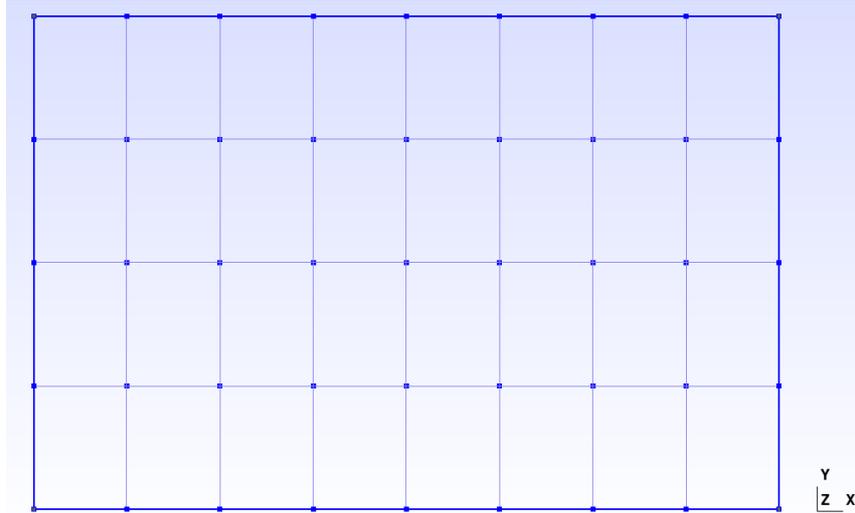


Figure 12: Eléments quadrangles structurés d'ordre 1

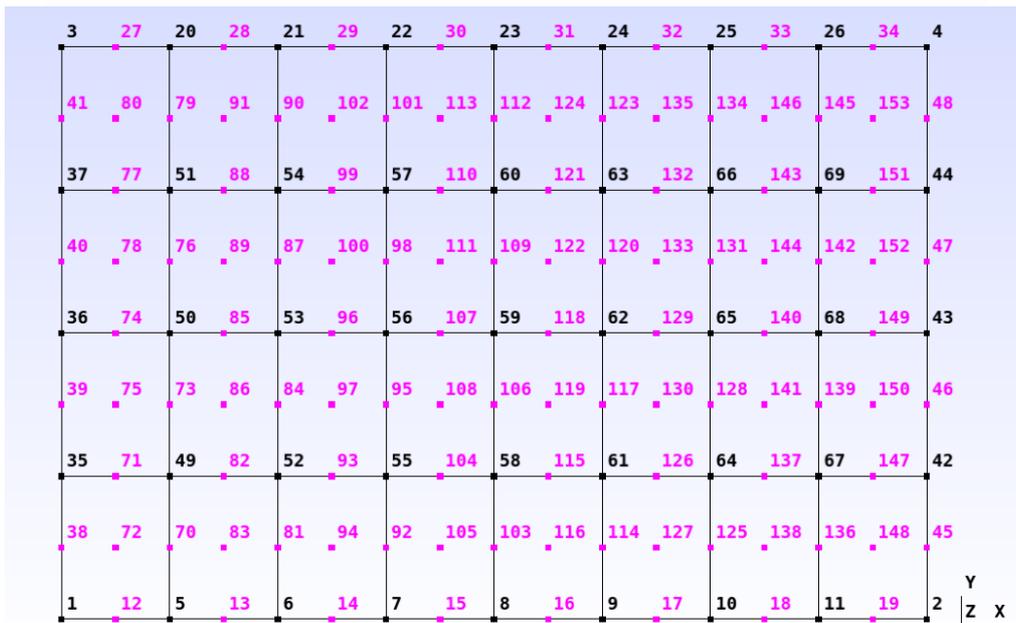
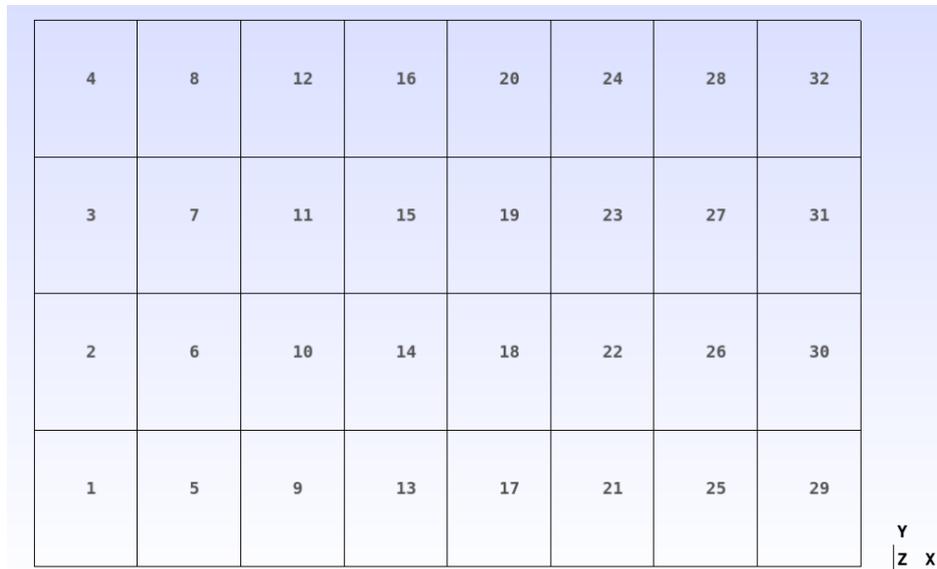


Figure 13: Eléments quadrangles structurés d'ordre 2

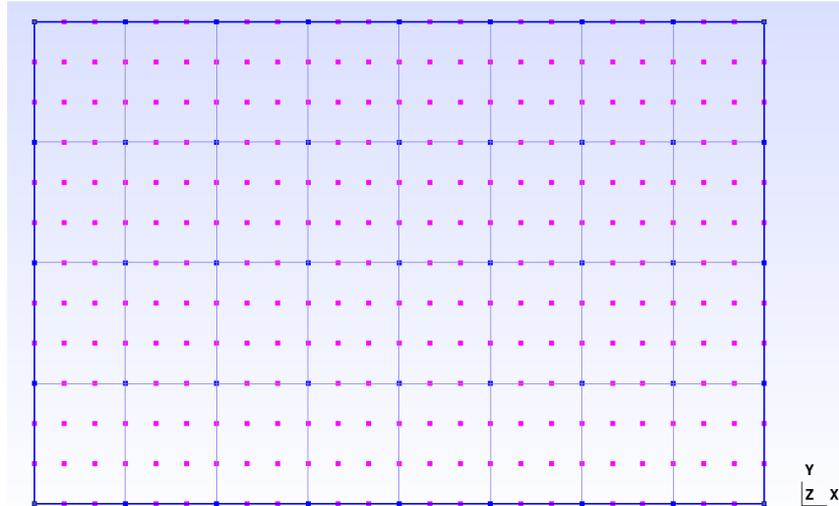


Figure 14: Eléments quadrangles structurés d'ordre 3

4) Elément Tétraèdre 3D

Ci-dessous le fichier.geo permettant de créer une plaque en 3D composée d'éléments tétraèdre non structurée:

```
//+ Définitions des Paramètres
a = DefineNumber[ 30, Name "Parameters/longueur" ];
b = DefineNumber[ 20, Name "Parameters/largeur" ];
h = DefineNumber[ 3, Name "Parameters/hauteur" ];
ma = DefineNumber[ 4, Name "Parameters/maillage" ];

//+ Création de la plaque
Point(1) = {0, 0, 0, ma};

Extrude {a, 0, 0} {
  Point{1};
}

Extrude {0, b, 0} {
  Line{1};
}

Extrude {0, 0, h} {
  Surface{5};
}

//+ Définition du maillage
Transfinite Line {10, 8} = 4 Using Progression 1;
Transfinite Line {9, 7} = 6 Using Progression 1;
Transfinite Line {17, 21, 12, 13} = 2 Using Progression 1;

//+ Définition des références
Physical Volume("Volume") = {1};
```

Paramètres de dimension pour nos points

On crée la coordonnée du point

On extrude notre point. Cela nous crée une ligne.

On extrude notre ligne. Cela nous crée une surface.

On extrude notre surface. Cela nous crée un volume.

Nombre d'élément +1 sur sa largeur (3 éléments)

Nombre d'élément +1 sur sa longueur (5 éléments)

Nombre d'élément +1 sur sa hauteur (1 élément)

Permet de récupérer nos éléments tétraédriques sous msh2her.pl

Ensuite sous gmsh on peut mailler en 3D notre plaque composée d'éléments tétraèdres non structurée selon une interpolation linéaire (ordre 1), une interpolation quadratique (ordre 2) ou une interpolation cubique (ordre 3) comme suit :

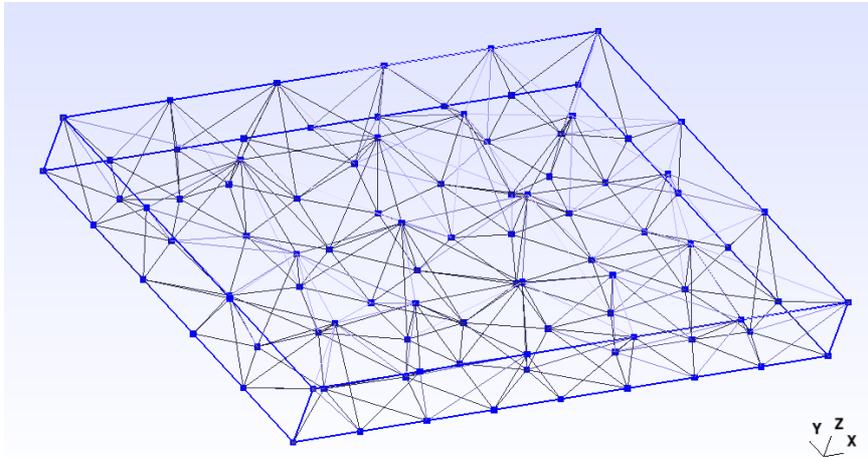


Figure 15: Eléments tétraèdre non structurée d'ordre 1

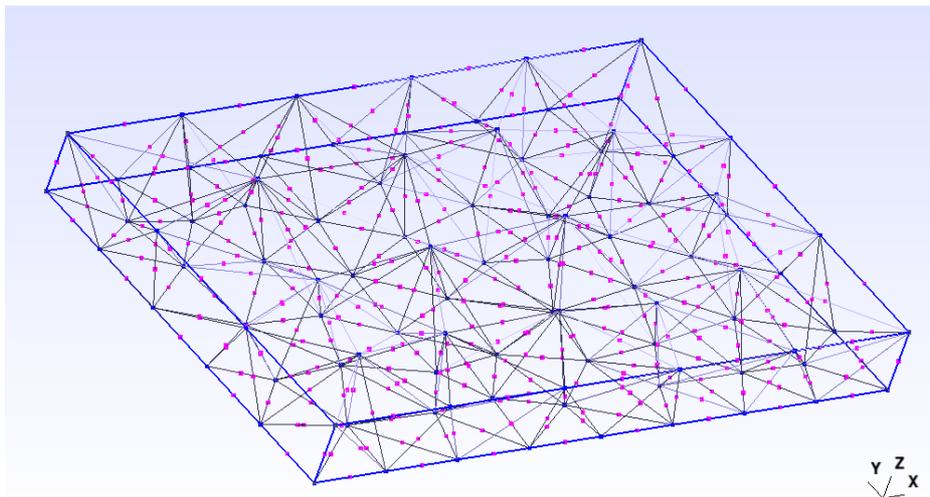


Figure 16: Eléments tétraèdre non structurée d'ordre 2

Ci-dessous le fichier.geo permettant de créer une plaque en 3D composée d'éléments tétraèdres structurés :

```
//+ Définitions des Paramètres
a = DefineNumber[ 30, Name "Parameters/longueur" ];
b = DefineNumber[ 20, Name "Parameters/largeur" ];
h = DefineNumber[ 3, Name "Parameters/hauteur" ];
ma = DefineNumber[ 4, Name "Parameters/maillage" ];
ta = DefineNumber[ 5, Name "Parameters/tranche_a" ];
tb = DefineNumber[ 3, Name "Parameters/tranche_b" ];
th = DefineNumber[ 1, Name "Parameters/tranche_h" ];

//+ Création de la plaque
Point(1) = {0, 0, 0, ma};
Extrude {a, 0, 0} {
```

Paramètres de dimension pour nos points

Nombre d'élément sur sa longueur

Nombre d'élément sur sa largeur

Nombre d'élément sur sa hauteur

On crée la coordonnée de notre point

On extrude notre point. Cela nous crée une ligne dépendant du nombre d'élément sur sa longueur

```

Point{1}; Layers{ta}; ←
}

Extrude {0, b, 0} {
  Line{1}; Layers{tb}; ←
}

Extrude {0, 0, h} {
  Surface{5}; Layers{th}; ←
}

//+ Définition des références
Physical Volume("Volume") = {1}; ←

```

On extrude notre ligne. Cela nous crée une surface dépendant du nombre d'élément sur sa largeur

On extrude notre surface. Cela nous crée un volume dépendant du nombre d'élément sur sa hauteur

Permet de récupérer nos éléments tétraédriques sous msh2her.pl

Ensuite sous gmesh on peut mailler en 3D notre plaque composée d'éléments tétraèdres structurée selon une interpolation linéaire (ordre 1), une interpolation quadratique (ordre 2) ou une interpolation cubique (ordre 3) comme suit :

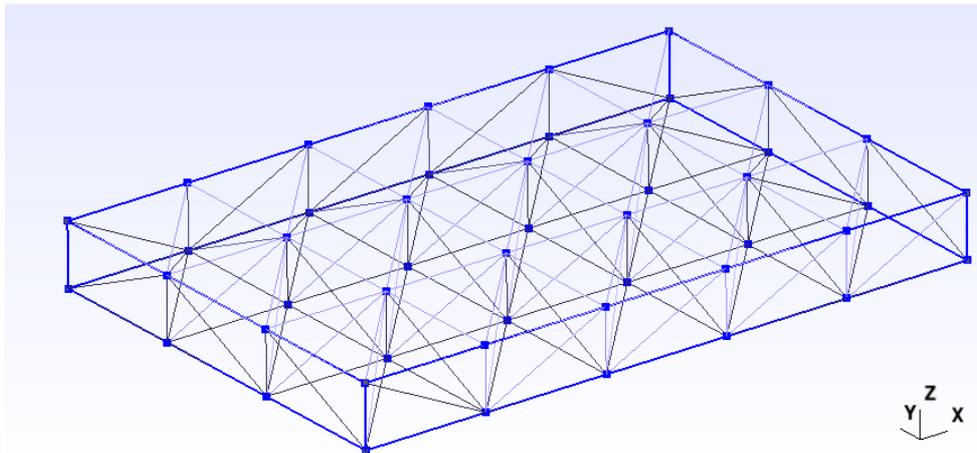


Figure 17: Éléments tétraèdres structurée d'ordre 1

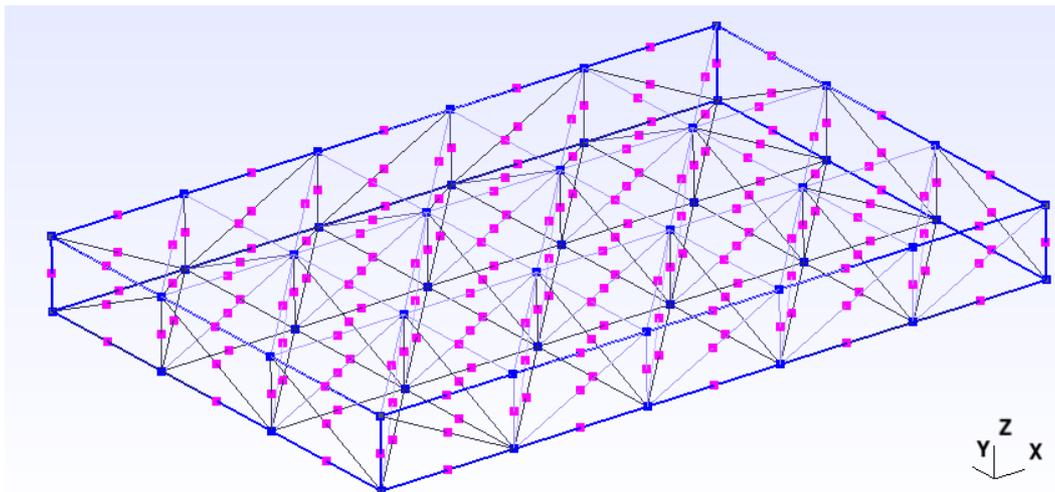


Figure 18: Éléments tétraèdres structurée d'ordre 2

5) Élément Hexaèdre 3D

Ci-dessous le fichier.geo permettant de créer une plaque en 3D composée d'éléments hexaèdres structurés:

```
//+ Définitions des Paramètres
a = DefineNumber[ 30, Name "Parameters/longueur" ];
b = DefineNumber[ 20, Name "Parameters/largeur" ];
h = DefineNumber[ 3, Name "Parameters/hauteur" ];
ma = DefineNumber[ 4, Name "Parameters/maillage" ];
ta = DefineNumber[ 8, Name "Parameters/tranche_a" ];
tb = DefineNumber[ 4, Name "Parameters/tranche_b" ];
th = DefineNumber[ 1, Name "Parameters/tranche_h" ];

//+ Création de la plaque
Point(1) = {0, 0, 0, ma};

Extrude {a, 0, 0} {
  Point{1}; Layers{ta};Recombine;
}

Extrude {0, b, 0} {
  Line{1}; Layers{tb};Recombine;
}

Extrude {0, 0, h} {
  Surface{5}; Layers{th};Recombine;
}

//+ Définition des références
Physical Volume("Volume") = {1};
```

Paramètres de dimension de la plaque

Nombre d'élément sur sa longueur

Nombre d'élément sur sa largeur

Nombre d'élément sur sa hauteur

On crée la coordonnée de notre point

On extrude notre point. Cela nous crée une ligne dépendant du nombre d'élément sur sa longueur

Recombine permet d'avoir des éléments quadrangles

On extrude notre ligne. Cela nous crée une surface dépendant du nombre d'élément sur sa largeur

On extrude notre surface. Cela nous crée un volume dépendant du nombre d'élément sur sa hauteur

Permet de récupérer nos éléments hexaédriques sous msh2her.pl

Ensuite sous gmsh on peut mailler en 3D notre plaque composée d'éléments hexaèdres selon une interpolation linéaire (ordre 1), une interpolation quadratique (ordre 2) ou une interpolation cubique (ordre 3) comme suit :

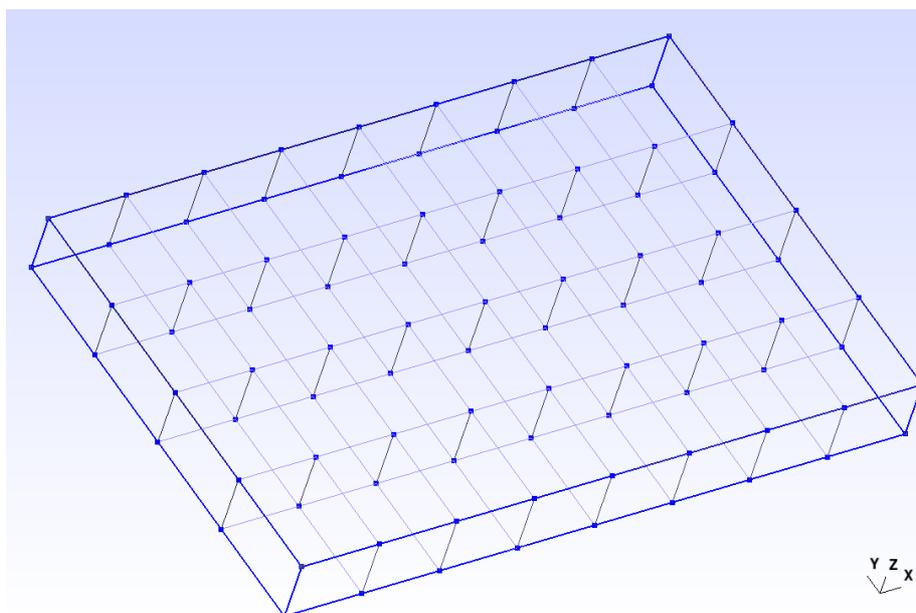


Figure 19: Éléments hexaèdres structurés d'ordre 1

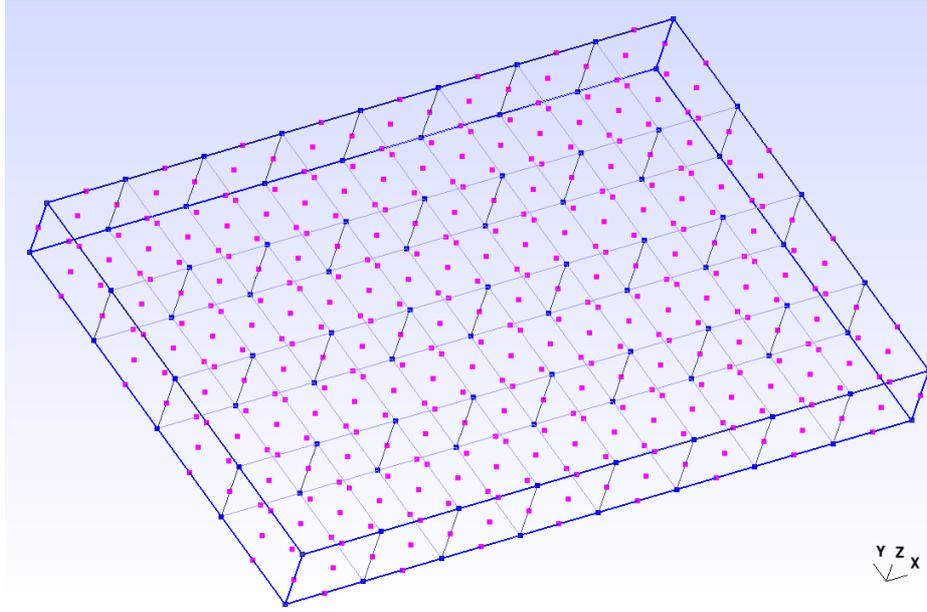


Figure 20: Eléments hexaèdres structurés d'ordre 2

V) Créer un maillage au format her :

Le logiciel msh2her.pl permet d'un format msh (obtenu sous gmesh lors de la sauvegarde du maillage) de créer la mise en donnée du maillage sous un format lisible par Herezh (fichier.her). Pour cela, il faut dans un premier temps à la racine de votre répertoire de travail exécuter sur un terminal la commande suivante « msh2her.pl nom_fichier.msh » vous devez obtenir cela :

```
roman@roman-X751LJ:~/Documents/Stage_annee2/tuto_1ere_annee/Elements_dispo/eleme
nt2D/quadrangle$ msh2her.pl quadrangle_strustructuree_m1.msh ← Commande pour ouvrir
stamm

#####
#
#                               msh2her.pl
#   transformation d'un maillage genere par le programme gmesh
#   en un maillage utilisable par Herezh++
#####
# licence GPL,                               gerard.rio@univ-ubs.fr
#                               http://www-lg2m.univ-ubs.fr/~rio
# version 2.2
#####

lecture des tags (preparation a la creation des references)
  format de sauvegarde du maillage : msh 2.2
  nombre total de references lus : 1
les references                               Références lisibles si elles appartiennent à un Physical Group dans
1 => Surface                               le fichier.geo
lecture des coordonnees des noeuds           nombre total de noeuds lus : 153
lecture des elements                         nombre total d'elements lus : 32
nombre d'elements du type quadrangle        nombre d'elements du type quadrangle quadratique complet a 9 noeuds = 32 ← Lecture des différents
nombre initial d'elements differents 1      noeuds, éléments etc

=== choix d'elements: voulez-vous ?
conserver les elements du type quadrangle   quadratique complet a 9 noeuds (rep o ← Choix des éléments
ou n) o                                     qu'on veut récupérer
====> ok, on conserve les elements         quadrangle quadratique complet a 9 noeuds
quadrangle quadratique complet a 9 noeuds:  nombre de points d'integration par default 0 ? (rep o/n) o
                                             ← Possibilité de changer
                                             le nombre de points
                                             d'intégrations si besoin

#####
#                               msh2her.pl
#   fin creation du fichier de maillage .her
#####
```

Figure 21: Terminal msh2her.pl

Récapitulons les différents éléments possibles avec un Physical Group :

- Physical Point => permet d'avoir un élément point
- Physical Line => permet d'avoir des références d'élément biellettes
- Physical Surface => permet d'avoir des éléments triangles ou quadrangles
- Physical Volume => permet d'avoir des éléments tétraèdres ou hexaèdres

Remarque : On crée des références de lignes pour un blocage et une force puis une référence de surface pour récupérer nos éléments quadrangles par exemple. Sous msh2her.pl il va lire nos 3 références donc nous avons la possibilité d'avoir à la fois des éléments quadrangles mais aussi des éléments biellettes. Si nous voulons seulement nos éléments quadrangles il suffit d'accepter seulement ce choix là et refuser nos segments linéaires. Par contre, lors de la création du fichier.her il y aura une référence de nœud lié au blocage et à la force ainsi qu'une référence d'élément associé à ces 2 références sans pour autant avoir des éléments biellettes.

Ensuite pour visualiser ce maillage, il vous est possible grâce à `hz_visuMail.pl` de voir le maillage créé. Il vous suffit dans un terminal à la racine de vos fichiers d'utiliser la commande suivante « `hz_visuMail.pl nom_fichier.her` » vous devez obtenir cela :

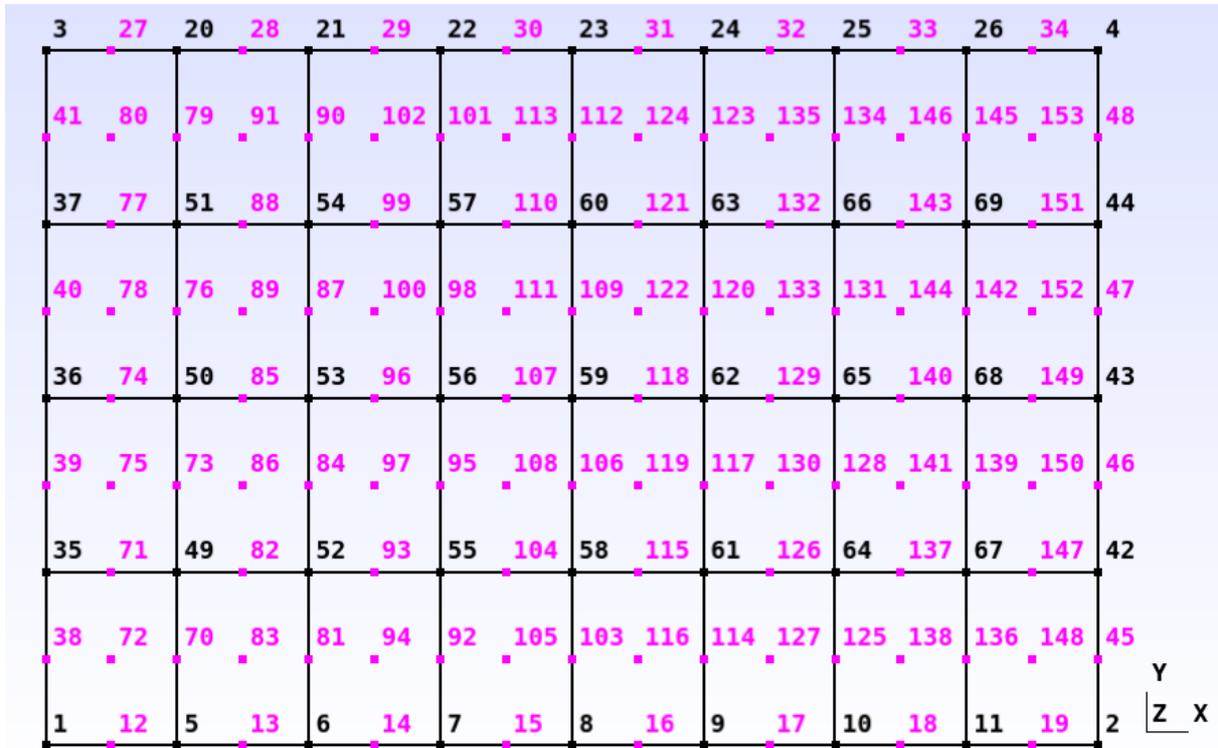


Figure 22: Quadrangles structurés d'ordre 2

VI) Conclusion :

Il vous est dorénavant possible de créer n'importe quelle type d'élément sous `gmsh` peu importe la dimension de travail. Nous pouvons cumuler sur un même maillage différent élément. Mais vous êtes aussi capables d'optimiser le temps de calcul via le maillage utilisé. A noter que pour créer le fichier `.her` il vous est nécessaire de sauvegarder le maillage créé sous `gmsh` vous générant ainsi un fichier `.msh`. Ensuite, grâce au logiciel `msh2her.pl` et le fichier `.msh` cela vous génère selon vos choix le fichier `.her` utile pour `Herezh`. D'ailleurs, votre maillage créé peut être visualisé via `hz_visuMail.pl`