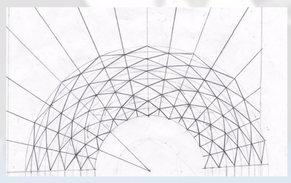
SUBDIVISIONS, SUBDIVISONS !

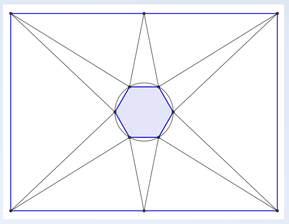
NOTRE OBJECTIF :
COMPRENDRE LA CONSTRUCTION DU DÔME DU BRITISH MUSEUM À LONDRES

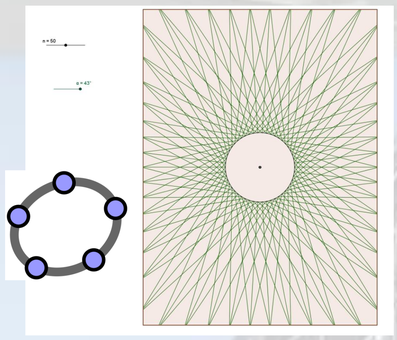
GAP

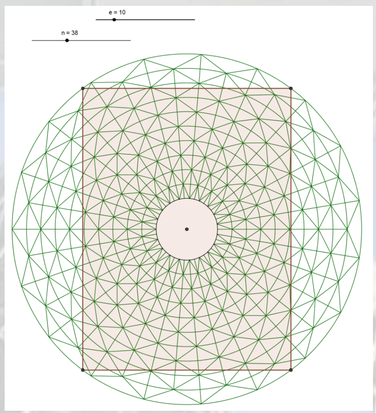
Notre chercheur, M. Raffin, nous présente son domaine de recherche.

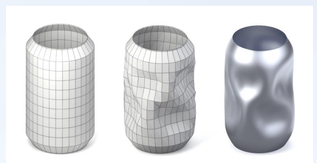
PREMIÈRES RÉFLEXIONS ET ESSAIS:





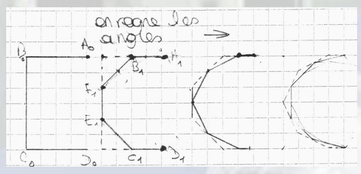


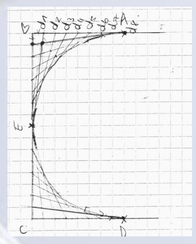




$$A_{n+1} = 3T_0 . 4^n + 2 . A_n$$

Au début, nous avons réfléchi à des façons d'adoucir des objets anguleux, comme par exemple un [ ] **.**





La subdivision consiste à ajouter ou transformer des points et des arêtes pour rendre la courbe ou la surface plus lisse.
Certaines techniques n'étaient pas possibles pour nous car elles changent la taille de l'objet.
Puis nous avons essayé de relier les sommets d'un hexagone central à un rectangle extérieur.
Nous avons utilisé Geogebra pour tester nos idées et augmenter le nombre de côtés du polygone central.

LONDON

PROBLÉMATIQUE :

Quand doit-on arrêter de subdiviser ?

IL VA FALLOIR UN ALGORITHME !

On s'est vite rendu compte que pour voir ce que donneraient nos idées après plusieurs opérations, il allait falloir automatiser les calculs avec un algorithme.
On part d'un pavage avec des triangles et on casse chaque triangle en quatre en utilisant les médianes.
Nous avons utilisé Scilab pour coder cet algorithme. Au début on a eu beaucoup de mal, même pour faire des choses qui nous paraissaient simples. On a 280 lignes de code en tout !

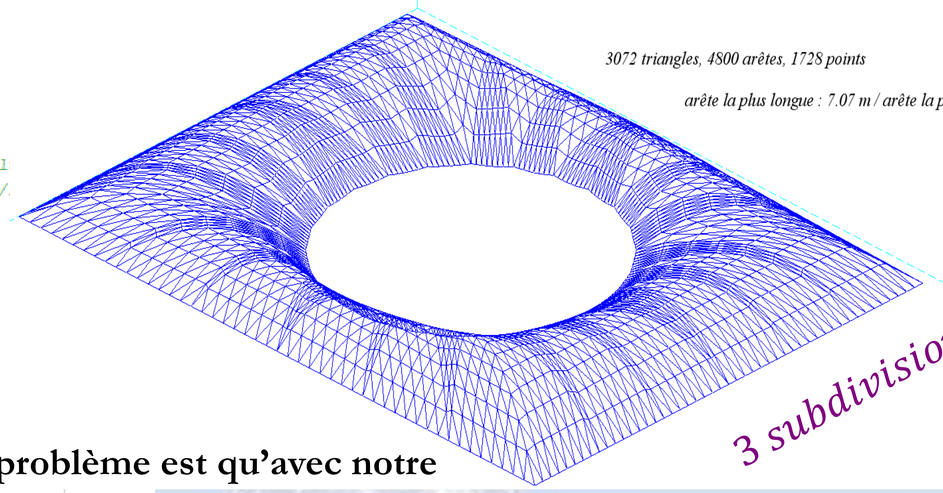
```
//
//·MATh.en.JEANs·Aristide·Briand·GAP·2013·:·19/03/2014


R=22;L=97/2;l=73/2;hauteurtoit=10//·Dimensions·réelles··en·m,·de·la·cour
Dimaretefinale=5;//·seuil·pour·arêter·de·subdiviser

function·l=longueur3D(p1,p2)
//calcule·la·longueur·entre·deux·points
····l=sqrt((p1(1)-p2(1))^2+(p1(2)-p2(2))^2+(p1(3)-p2(3))^2)
endfunction

function·[ct,cv,lum,p,aire]=performance(pts3D,arts,trgls)
//·donne·le·coût·de·la·tubulure·(ct),·du·verre·(cv),·la·luminosité·(lum),·le·poids·(p)
//·et·l'aire·(aire)·de·la·structure
····coutunitaireverre=350;coutunitairetube=40;diametretube=0.15;poidsunitairetube=8;poids
····//·caractéristiques·du·verre·et·du·tube·utilisés
····longtotaletube=0;surftotaleverre=0;surftotaletube=0;
····for·i=1:length(trgls)//calcul·de·l'aire·des·triangles·avec·Héron
········P1=pts3D(trgls(i)(1));P2=pts3D(trgls(i)(2));P3=pts3D(trgls(i)(3))
········//identification·des·points·du·triangle·n°i
········a=longueur3D(P1,P2);b=longueur3D(P2,P3);c=longueur3D(P1,P3)
········p=(1/2)*(a+b+c)//demi-périmètre
········Airetriangle=sqrt(p*(p-a)*(p-b)*(p-c))//formule·de·Héron
········surftotaleverre=surftotaleverre+Airetriangle
····end
····for·i=1:length(arts)//calcul·des·longueurs·des·arêtes
········P1=pts3D(arts(i)(1));P2=pts3D(arts(i)(2));
········//identification·des·points·de·l'arête·n°i
········longtotaletube=longtotaletube+longueur3D(P1,P2)
····end
····surftotaletube=diametretube*longtotaletube
····//calcul·des·indicateurs
····ct=longtotaletube*coutunitairetube;//·coût·du·tube
····cv=surftotaleverre*coutunitaireverre;//·coût·du·verre
····lum=(surftotaleverre-surftotaletube)/surftotaleverre;//·calcul·de·la·l
····p=longtotaletube*poidsunitairetube+surftotaleverre*poidsunitaireverre/
····aire=surftotaleverre-surftotaletube//·aire·totale
endfunction

function·z=aretelapluslongue(arts,pts3D)
//calcul·de·l'arête·la·plus·longue
····longmax=0;
····for·i=1:length(arts)
········long=longueur3D(pts3D(arts(i)(1)),pts3D(arts(i)(2)));
········if·(long>longmax)·then·longmax=long
····end
····format(5);z=longmax;format()
endfunction
```

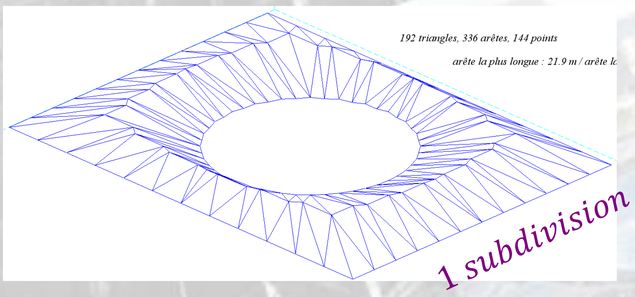
$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

NOS RÉSULTATS :

Au début, le toit est très grossier mais rapidement son aspect devient très joli.
Merci l'affichage en 3D de Scilab ! *quand on a compris comment ça marche !*

MeJ GAP :





Le problème est qu'avec notre technique, on atteint (très !) vite des longueurs d'arêtes très petites et donc certains triangles contiennent peu de verre et ça fait baisser très vite la luminosité du toit.

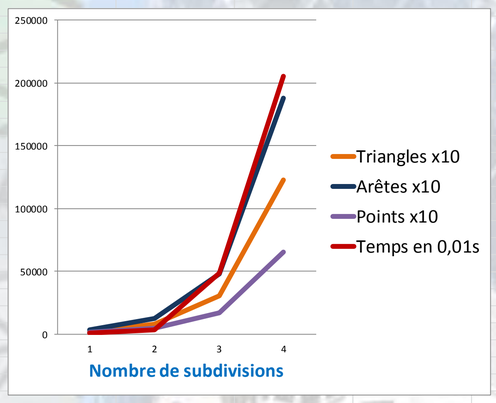
COMMENT EST FAIT NOTRE PROGRAMME ?

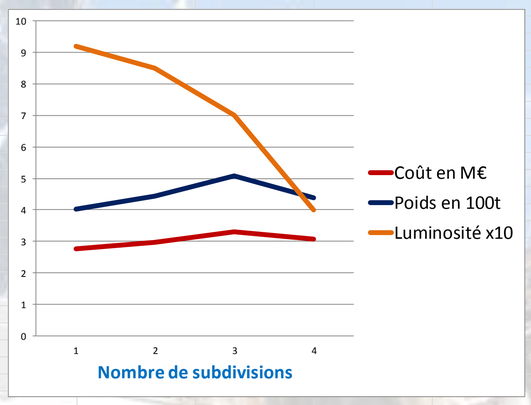
- ✓ Générer la liste de points 2D (sur le cercle central et le rectangle extérieur) et d'arêtes de départ
- ✓ Tant que toutes les arêtes ne sont pas plus petites qu'un certain seuil :
  - ❑ Chercher les triangles
  - ❑ Dans chaque triangle rajouter les médianes
  - ❑ Générer la liste de points en 3D (calcul, pour chaque point, de sa hauteur)
  - ❑ Afficher le résultats en 3D
  - ❑ Calculer puis afficher les indicateurs de performance du toit (coût, poids, luminosité …)
  - ❑ Calculer la taille de la plus grande arête

Pour que le programme soit plus clair et facile à améliorer, nous avons utilisé des fonctions qui effectuent les différentes tâches :

```
function [y]=trouvertriangles(arts, pts)
function [y, x]=rajouteraretesmedianes(y, x, trgls)
function y=genererpoints3D(pts)
function afficherresultats(pts3D, arts, trgls, n)
function z=aretelapluslongue(arts, pts3D)
```

Subdivisions	Triangles x10	Arêtes x10	Points x10	Temps en 0,01s	Coût en M€	Poids en 100t	Luminosité x10	Aire totale en m²
1	1920	3360	1440	1119	2,755	4,026	920%	6829
2	7680	12480	4800	3416	2,978	4,434	850%	6438
3	30720	48000	17280	48690	3,306	5,077	700%	5337
4	122880	188160	65280	205220	3,08	4,393	400%	3051
réel	33120					793		6000





ON VOIT APPARAÎTRE UN OPTIMUM POUR 3 SUBDIVISIONS ET ON A EU LA SURPRISE QU'ALORS, « NOTRE » TOIT ÉTAIT ASSEZ PROCHE DU VRAI SI ON REGARDE LE NOMBRE DE TRIANGLES, LE POIDS OU LA SURFACE !

Jade-Laura-Solène-Alexandra-Elsa-Alexia-Sarah-Pauline-Lise-Vladimir-Johan-Nicolas-Zélia-Matthieu-Killian-Brandon-Ludovic

Romain Raffin (maître de conférences)
Emmanuel Berton (professeur)
Françoise Hirtz-Villard (professeur)
Christian Marchal (professeur)

Lycée Aristide Briand Polyvalent Régional

académie d'aix-marseille

Région PACA