

Plugin $\text{T}_{\text{E}}\text{Xgraph}$ pour $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$

TeXgraph par PATRICK FRADIN <http://texgraph.tuxfamily.org/index.html>

Plugin par EMMANUËL CORCELLE : <http://corcelle.wordpress.com/>

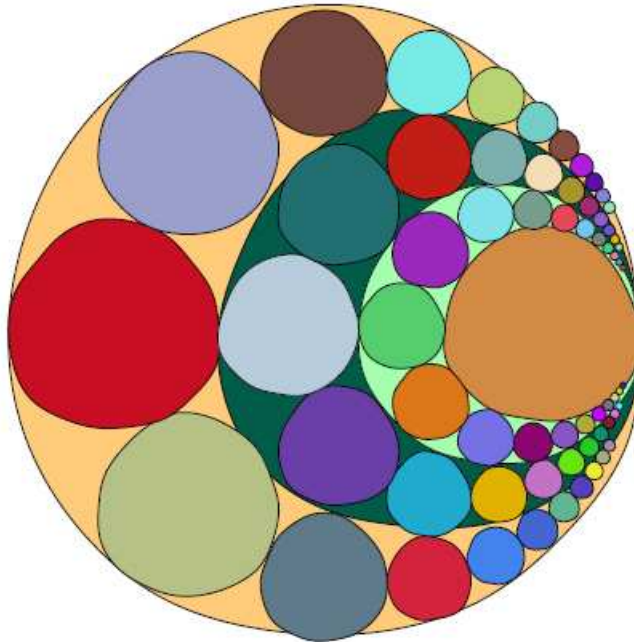


Figure 1. Exemple par P. FRADIN, tiré du forum

1 Présentation

$\text{T}_{\text{E}}\text{Xgraph}$ est un logiciel écrit par PATRICK FRADIN qui permet la création de graphiques mathématiques (comme les droites, les cercles, les courbes, les surfaces, les labels..) destinés à être utilisés dans des documents $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

- Site du projet : <http://texgraph.tuxfamily.org>
- Forum : <http://texgraph.forumpro.fr>

Le plugin permet d'insérer une session $\text{T}_{\text{E}}\text{Xgraph}$ en ligne de commande ou un « commutateur » commandes/figure à l'intérieur de $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$. Un traceur de courbes est aussi en construction.

2 Installation

1. Installer $\text{T}_{\text{E}}\text{Xgraph}$ dans sa dernière version (à partir du forum).
2. Décompresser l'archive du plugin `texgraph.zip` (disponible sur <http://corcelle.wordpress.com/>) dans le dossier `~/TeXmacs/plugins/` ou bien dans `/Dossier_TeXmacs/plugins/`.

Avertissement. Pour bénéficier du plugin sous Windows, mettre le chemin d'accès au dossier $\text{T}_{\text{E}}\text{Xgraph}$ dans votre variable d'environnement PATH. Pour cela, utiliser la modification du PATH dans l'onglet « avancé » des propriétés du poste de travail, voir par exemple l'installation de Xcas sur Les Maths Libres.

3 Le menu

Les entrées du menu TeXgraph retournent les codes des différents objets géométriques. Il ne reste plus qu'à remplacer si besoin est :

- le nom des objets : points, droites, segments, cercles, coniques...
- les variables : longueurs, angles...
- les étiquettes.
- les différents styles et codages.

Les informations sont notées **<en rouge>** et peuvent être laissées lors de la validation.

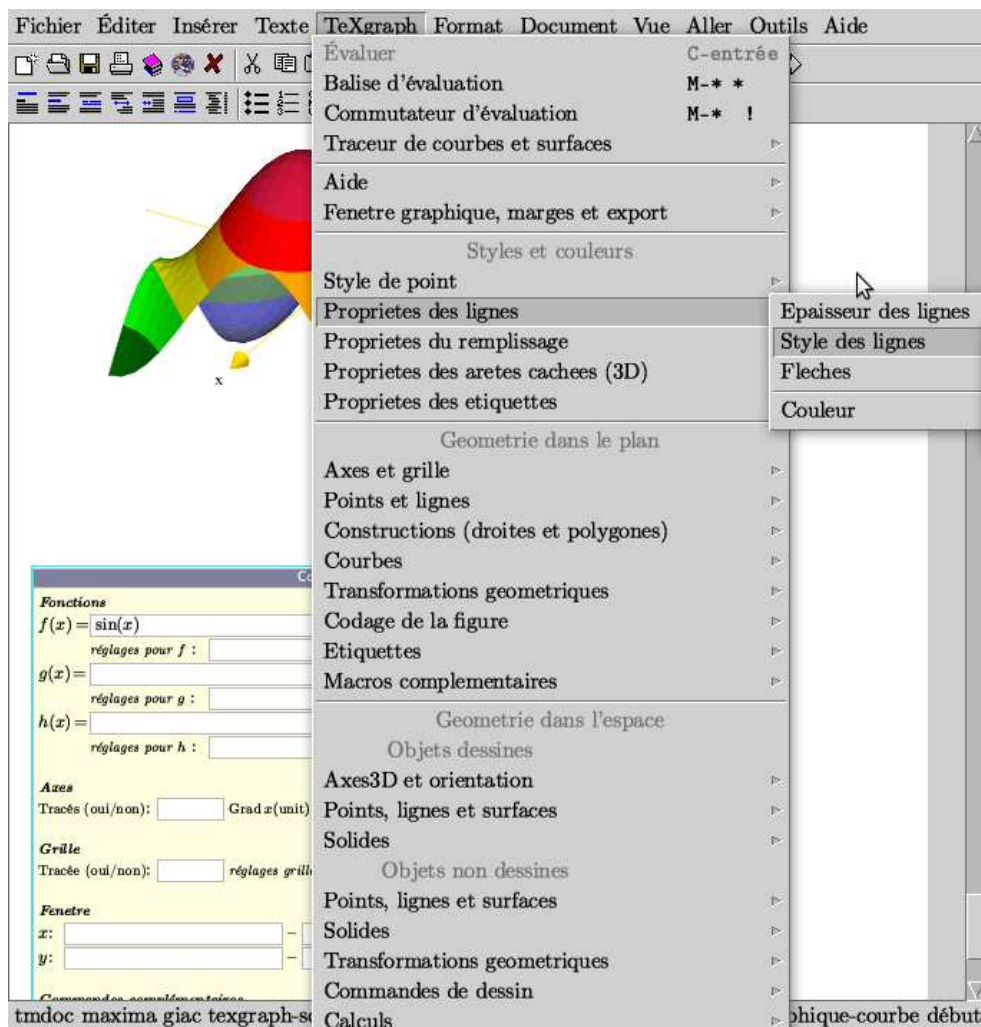


Figure 2. Menu spécifique à l'intérieur d'une session TeXgraph

4 Insertions de figures

4.1 Au sein d'une session TeXgraph

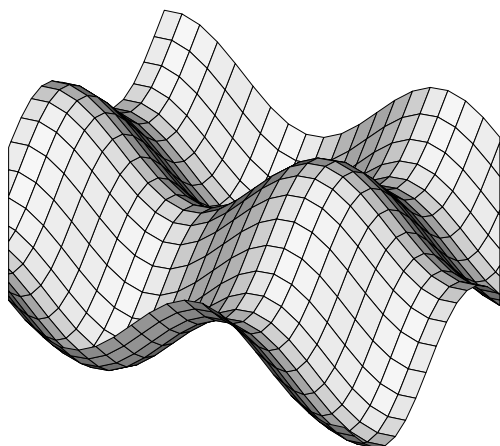
Ouvrir une session TeXgraph à partir de Insérer → Session ou bien de l'icône .

- Entrer les commandes au niveau de l'invite : `TeXgraph]`. Directement ou à partir du menu.
- Il est possible de passer à la ligne suivante avec `Shift-Entrée`.
- La validation se fait par `Entrée`, les commandes sont alors évaluées et la figure affichée en dessous.

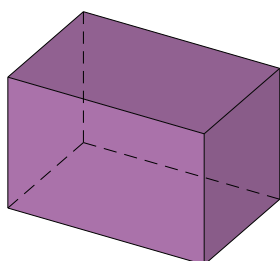
Exemple 1.

TeXmacs interface to TeXgraph.
To write LaTeX code inside labels, use `\\` instead of `\`.
For example, write `$$\pi` instead of `$$\pi`

```
TeXgraph] size(7), FillStyle:=full, FillColor:=white,
           Surface(u,v,cos(u)+sin(v),-5+5*i, -5+5*i,25+25*i,1)
```



```
TeXgraph] Fenetre(-0.5+3*i,4.5-2*i),
           FillStyle:=full, FillColor:=Light(purple,<facteur=>0.5), HideStyle:=dashed,
           P:= Parallelep(M(0,1.5,1), 2*vecI,3*vecJ,2*vecK), DrawPoly(P,4),
```



```
TeXgraph]
```

4.2 À tout endroit à l'aide du script TeXgraph

Choisir le script TeXgraph à partir de Document → Scripts.

Le menu TeXgraph donne alors accès à trois entrées supplémentaires :

- Commutateur d'évaluation qui ouvre une « boîte » `TeXgraph` dans laquelle on écrit le code.

Le passage du code à la figure (et réciproquement) se fait par la combinaison `Ctrl-*`.

Il faut pour cela être dans l'environnement de la « boîte » ou de la figure.

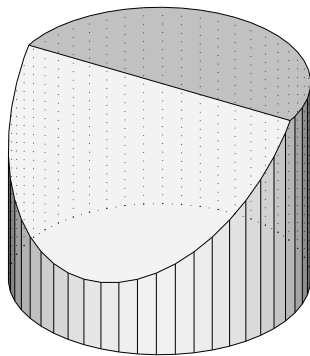
Exemple 2.

```

TeXgraph
Fenetre(-3+3*i,3-2*i),
FillStyle:=full, FillColor:=white,
c:=Cylindre([1+i,0],M(0,0,3),2,50),
plan:= [M(4,0,0), -vecI-vecK],
S:=Section(plan, c), DrawPoly(S,4),

```

On valide les commandes avec `Ctrl-*`, la « boîte » est alors remplacée par la figure :



Il est possible de revenir à la « boîte » de commandes avec `Ctrl-*` (si le curseur est à côté de la figure).

- Balise d'évaluation qui permet de taper le code puis valider pour obtenir la figure mais sans espoir de retour en arrière.
- Il est aussi possible de taper les commandes directement à l'intérieur de tout environnement mathématique « `$$` » puis de le valider par un `Ctrl-Entrée` pour obtenir la figure.

4.3 Traceur de courbes

Pour avoir accès au traceur de courbes :

1. Choisir le script TeXgraph à partir de Document → Scripts.
2. Appeler le traceur de courbe à partir du menu TeXgraph → Traceur de courbes (le package `texgraph-scripts.tm` se charge automatiquement).

Exemple 3.

Courbes TeXgraph

Fonctions

$f(x) =$

réglages pour f

$g(x) =$

réglages pour g

$h(x) =$

réglages pour h

Axes

Tracés (oui/non): Grad x (unit): Grad y (unit):

Origine du repère: réglages axes:

Grille

Tracée (oui/non): réglages grille:

Fenêtre

Largeur (+i*haut: ou échelle / Ox et échelle / Oy

x : -

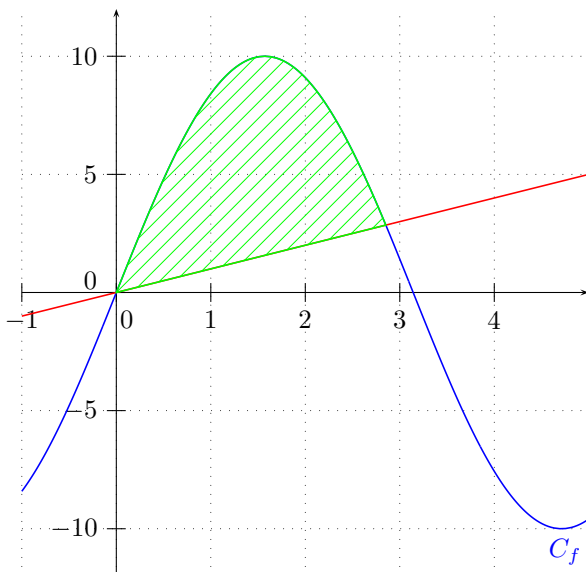
y : -

Commandes complémentaires

Pour les réglages, menu : TeXgraph → Propriétés des lignes

par exemple : `Color:=blue, LineStyle:=dashed,`

donnera après un **Ctrl-*** :



Par défaut, les courbes sont en traits pleins bleu (pour f), rouge (pour g) et vert (pour h). La fenêtre est définie pour $x \in [-5; 5]$ et $y \in [-5; 5]$, les axes sont tracés en noir et la grille en gris.

Il est possible de laisser des champs vides. Des réglages complémentaires sont possibles par exemple à partir du menu TeXgraph :

Couleur — `Color:=gray`, (ou `black`, `white`, `red`, `blue`, `yellow`, `green`, `orange...`)

Style de trait — `LineStyle:=solid`, (—) ou `LineStyle:=dashed`, (--) ou `LineStyle:=dotted`, (...)

Étiquette — `Label(2+4*i, "C_f")`, (C_f au point de coordonnées (2;4))

Insérer une étiquette à partir du menu et effectuer les modifications.

Point et étiquette — `LabelDot(2+i, "A", "NO", <mark=>1, <dist=>0.25)`,

Domaine entre f et g — `FillStyle:=bdiag`, `FillColor:=blue`, `domaine3(f(t),g(t))`,

Toutes les commandes complémentaires et réglages doivent être suivis d'une virgule.

5 Exemples

De nombreux exemples sont tirés des sites *TeXgraph* et de la documentation du logiciel *TeXgraph.pdf*.

TeXmacs interface to TeXgraph.

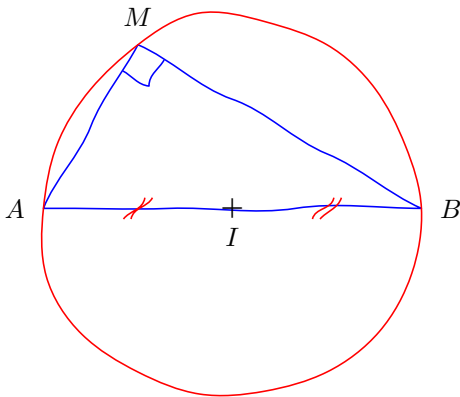
For LaTeX code, use \backslash instead of \backslash for example, $\backslash\pi$ instead of π

Rappels : Fenetre(-4+3*i,4-3*i,1+i), Marges(0.25,0.25,0.25,0.25), size(5)

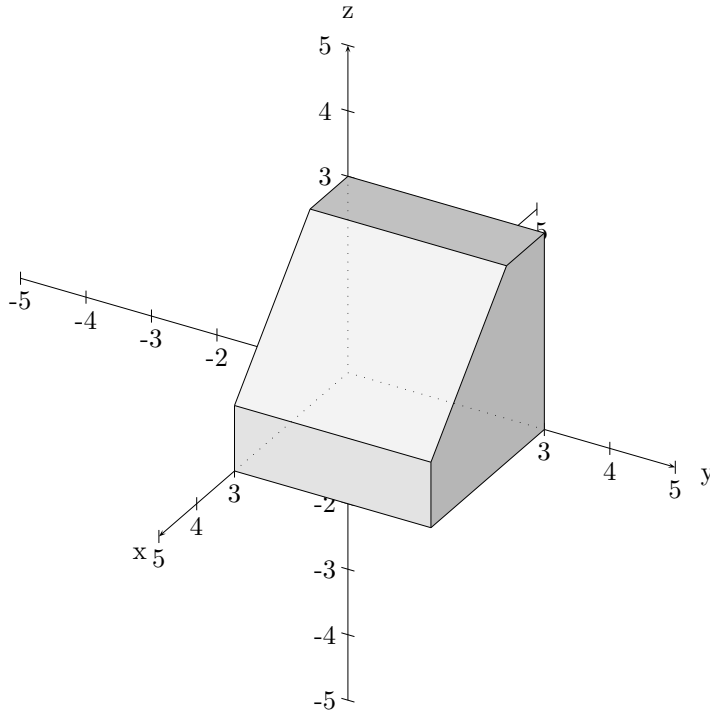
```
TeXgraph] Load("trembling.mac"),
Fenetre(-3+2.5*i,3-2*i,1+i), size(8), a:=0, b:=120, R:=2,
I:=0, A:=R*exp(i*(a+pi)), B:=R*exp(i*a*deg), C:=R*exp(i*b*deg),
Width:=6, Color:=blue,
tLigne([A,B,C],1), tLigne(angleD(A,C,B,<rayon=>0.4),0),

Color:=red,
tCercle([I,R],0.05),
tLigne(markseg(A,I,<nb=>2,<esp=>0.1,<long=>0.4), 0),
tLigne(markseg(B,I,<nb=>2,<esp=>0.1,<long=>0.4), 0),

Color:=black, DotStyle:=cross,
LabelDot(A,"A","O",<mark=>0,<dist=>0.25),
LabelDot(B,"B","E",<mark=>0,<dist=>0.25),
LabelDot(C,"M","N",<mark=>0,<dist=>0.25),
LabelDot(I,"I","S",<mark=>1,<dist=>0.25),
```



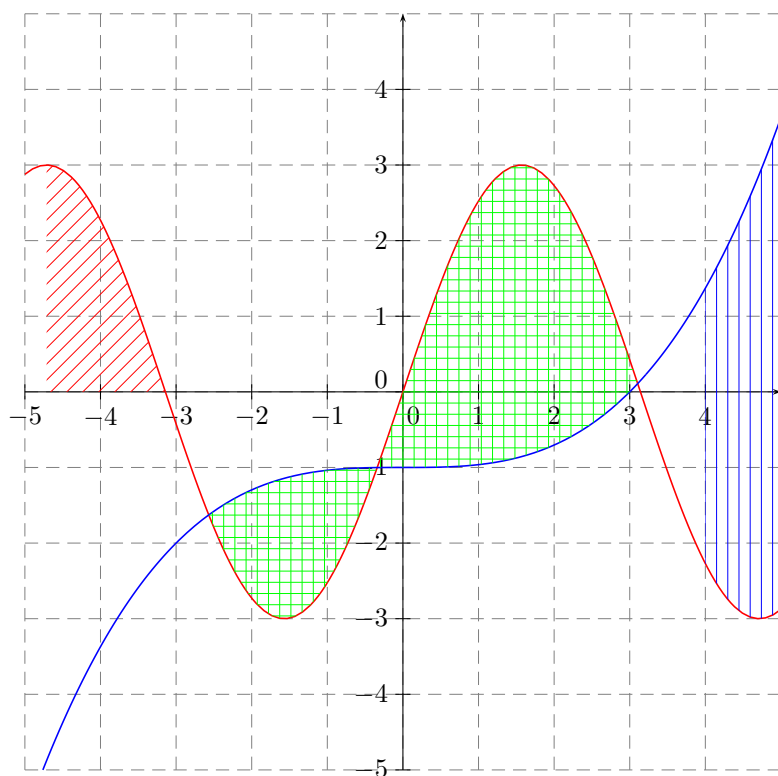
```
TeXgraph] Arrows:=1, Axes3D(0,0,0,1,1,1), Arrows:=0,
FillStyle:=full, FillColor:=white,
cube:= Parallelep(Origin, 3*vecI, 3*vecJ, 3*vecK),
plan:= [dot(4,0,0), -vecI-vecK],
S:=Section(plan, cube), DrawPoly(S,4)
```



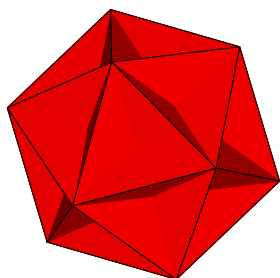
```
TeXgraph] Color:=gray, LineStyle:=dashed, Grille(0,1+i),
Color:=black, LineStyle:=solid,
Arrows:=1, Width:=4, Axes(0,1+i), Arrows:=0,

Color:=red, Width:=6, Courbe(t+i*3*sin(t)),
Color:=blue, Width:=6, Courbe(t+i*(t^3/27-1)),

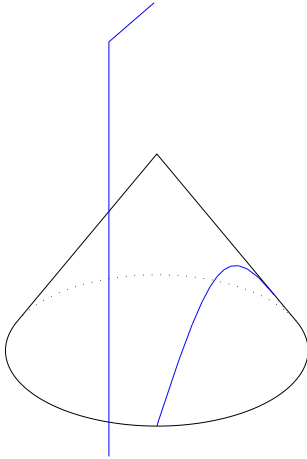
FillColor:=red, FillStyle:=bdiag ,LineStyle:=noline,
domaine1(3*sin(t),-3*pi/2,-pi),
FillColor:=blue, FillStyle:=vertical, LineStyle:=noline,
domaine2(3*sin(t),(t^3/27-1),4,5),
FillColor:=green, FillStyle:=hvcross, LineStyle:=noline,
tMin:=Xmin, tMax:=Xmax, domaine3(3*sin(t),(t^3/27-1)),
```



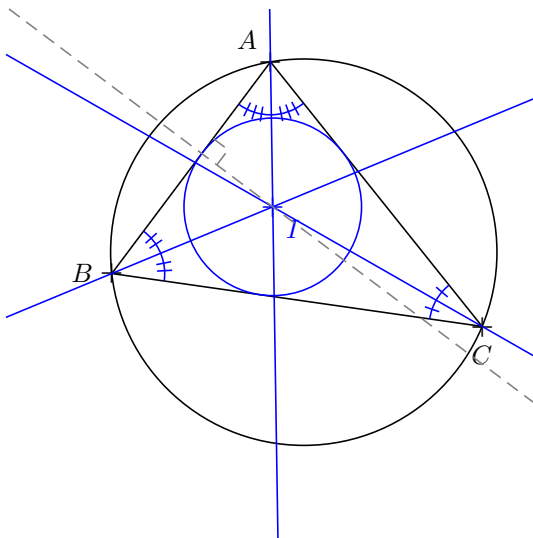
```
TeXgraph] Load("PolyedresII.mac"),
Fenetre(-2+2*i,2-2*i,1+i), size(8), Width:=8,
GdDodecaedre(Origin, [1, 0], P, A),
Build3D(P, A), Display3D()
```



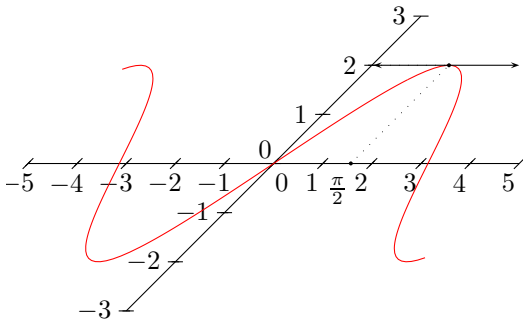
```
TeXgraph] Fenetre(-3+2*i,3-4*i,1+i),
Dcone( Origin, -3*vecK, 2, 2),
$C:=Cone(Origin, -3*vecK, 2, 35), FillStyle:=none,
$L:=Intersection( [[i,0], -vecJ], C,$S),
Del(L,1,5), Color:=blue, DrawAretes(L,0),
DrawPlan( [ [i,0], -vecJ], vecK, 7, 6,2)
```



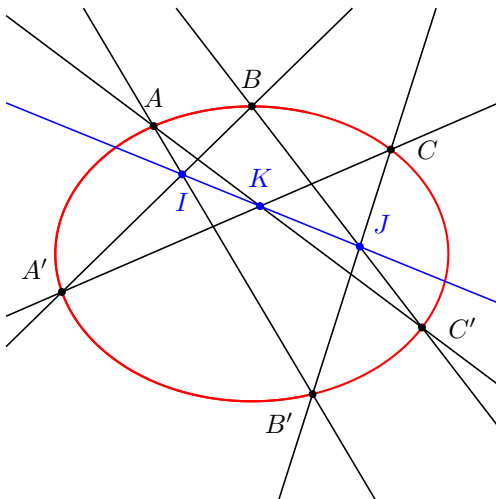
```
TeXgraph] size(7.5), A:=4*i, B:=-3, C:=4-i, Width:=6,
Ligne([A,B,C],1), Cercle(A,B,C),
DotStyle:=cross, LabelDot(A,"$A$","NO",1), LabelDot(B,"$B$","O",1),
LabelDot(C,"$C$","S",1),
Color:=blue,
Dbissec(B,A,C,1),Dbissec(C,B,A,1), Dbissec(A,C,B,1),
I:=bissec(B,A,C,1) Inter bissec(C,B,A,1), LabelDot(I,"$I$","SE",1), Cercle(I,abs(I-
proj(I,[A,B]))), L:=[A,C,B],
for k from 1 to 3 do
Arc(L[1],L[2],L[3],1), markangle(L[1],L[2],I,1,k,0.1,0.2),
markangle(L[3],L[2],I,1,k,0.1,0.2), permute(L),
od,
Color:=gray,LineStyle:=dashed,
Dperp([A,B],I,1)
```



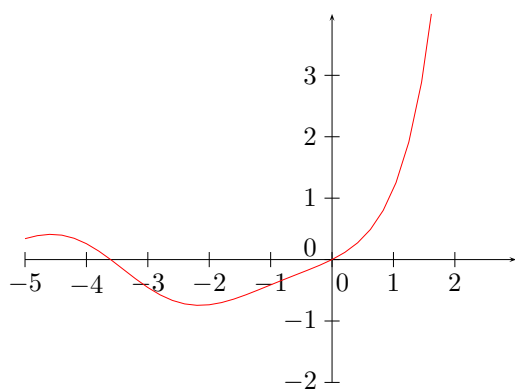
```
TeXgraph] Fenetre(-5+3*i,5-3*i,1+i), size(7), SetMatrix([0,1,1+i]),
Width:=4, axes(0,1+i), Color:=red, Courbe(t+i*2*sin(t)),
Color:=black, Arrows:=2, tangente(2*sin(x), pi/2, 2),
Arrows:=0, LineStyle:=dotted, Ligne( [2*i,pi/2+2*i, pi/2], 0), Point(pi/2+2*i),
LabelDot( pi/2, "$\\frac{\\pi}{2}$","S",1), IdMatrix()
```



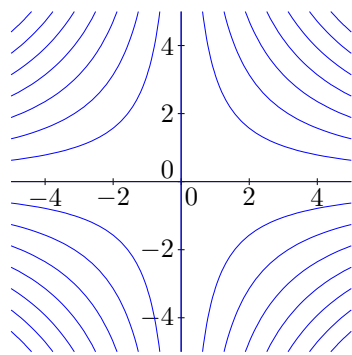
```
TeXgraph] size(7),
NewMac("f", "4*cos(t)+3*i*sin(t)",t),
Marges(0,0,0,0), Width:=8, Color:=red, Courbe(f(t)),
Width:=6,Color:=black, A:=f(2*pi/3),B:=f(pi/2),C:=f(pi/4),
C':=f(-pi/6),B':=f(-2*pi/5),A':=f(-11*pi/12),
I:=[A,B'] Inter [B,A'],J:=[B,C'] Inter [C,B'], K:=[C,A'] Inter [A,C'],
Droite(A,B'),Droite(A,C'), Droite(B,A'),Droite(B,C'),
Droite(C,A'),Droite(C,B'), DotStyle:=bigdot,
LabelDot(A,"$A$","N",1), LabelDot(A',"$A'$","NO",1),
LabelDot(B,"$B$","N",1), LabelDot(B',"$B'$","SO",1,0.35),
LabelDot(C,"$C$","E",1,0.35), LabelDot(C',"$C'$","E",1,0.35),
Color:=blue,Droite(I,K), LabelDot(I,"$I$","S",1),
LabelDot(J,"$J$","NE",1), LabelDot(K,"$K$","N",1)
```



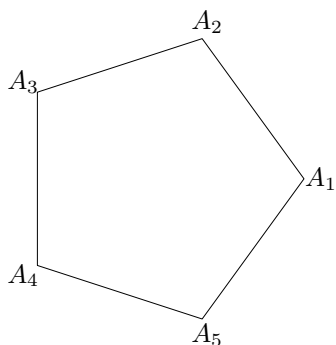
```
TeXgraph] Fenetre(-5+4*i,3-2*i,1+i),size(7),Arrows:=1,Axes(0,1+i),
Arrows:=0,Color:=red,
EquaDif(y+i*(t*x+y+sin(t)),0,i/2,1)
```



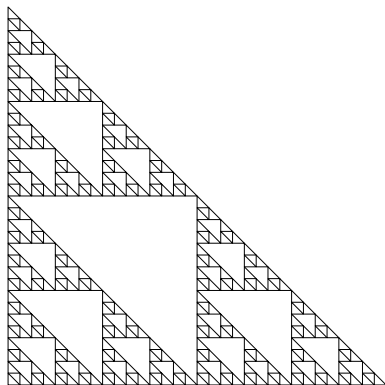
```
TeXgraph] Axes(0,2+2*i), size(5), Color:=blue,  
Implicit(sin(x*y))
```



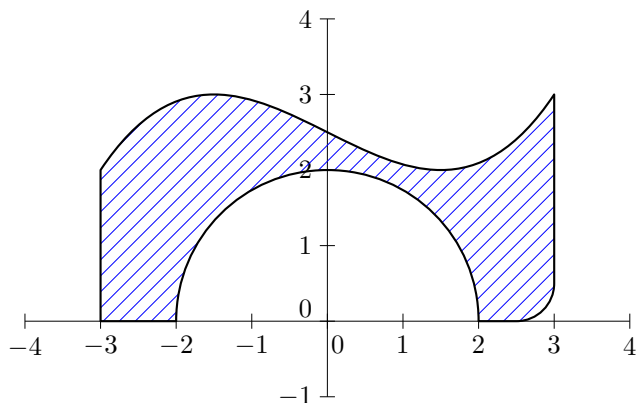
```
TeXgraph] size(7),  
Ligne( for k from 0 to 4 do 3*exp(2*i*k*pi/5) od,1),  
for k from 0 to 4 do  
Label(3.35*exp(2*i*k*pi/5), ["$A_{",k+1,"}$"]),  
od
```



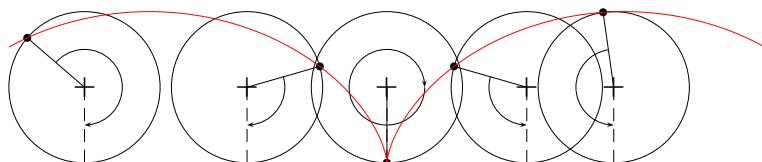
```
TeXgraph] Fenetre(-0.5+1*i,1.5-0.1*i,5+5*i),
L:= [0,1,i],
for k from 1 to 5 do
L:= [Map(z/2,z,L), Map((1+z)/2,z,L), Map((i+z)/2,z,L)]
od,
MapBy([FillColor:=Rgb(Rand(),Rand(),Rand()), Ligne(z,1)],z,L,3)
```



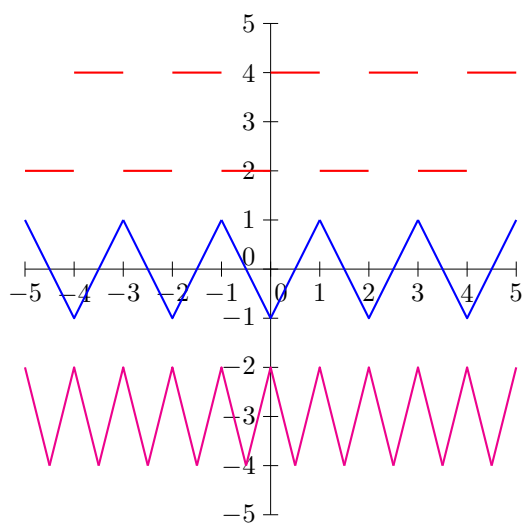
```
TeXgraph] Fenetre(-4+4*i,4-i,1+i), Axes(0,1+i),
FillStyle:=bdiag, FillColor:= blue, Width:=8,
Path([-3+2*i,-3,-2,line, 0,2,2,-1,arc, 3,3+3*i,0.5,linearc, 1,-1+5*i,-3+2*i,bezier], 1)
```



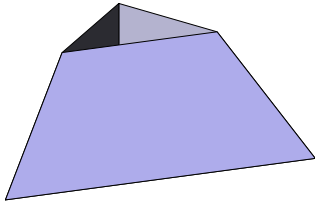
```
TeXgraph] Seg(-5,5),
for t in [-4,-1.85,0,1.85,3] do
M:=t-sin(t)+i*(1-cos(t)), I:=t+i,
DotStyle:=cross, Point(I), DotStyle:=bigdot,
Point(M), Cercle(I,1), Seg(M,I),
Arrows:=1, Arc(M,I,t,0.5,t), Arrows:=0,
LineStyle:=dashed, Seg(I,t), LineStyle:=solid
od,
Color:=red,Courbe(t-sin(t)+i*(1-cos(t)))
```



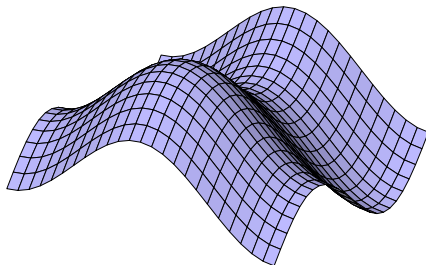
```
TeXgraph] size(7),Axes(0,1+i), Width:=8,
Color:=red, periodic(if t>0 then 4 else 2 fi,-1,1,5,1),
Color:=blue, periodic(2*abs(t)-1, -1,1),
Color:=magenta, periodic(-2-4*abs(t), -0.5, 0.5,5)
```



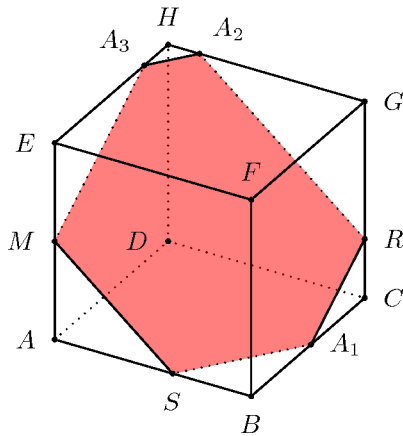
```
TeXgraph] Fenetre(-1+1*i,1-1*i,3+3*i),P:=Tetra(Origin, vecI,vecJ,vecK),  
S:=ClipFacet(P,[M(0,0,0.5),-vecK]), FillStyle:=full, FillColor:=Rgb(0.74,0.73,1),  
DrawPolyNC(S,1)
```



```
TeXgraph] size(7),FillColor:=Rgb(0.74,0.73,1), FillStyle:=full,  
S:=GetSurface( [u+i*v,sin(u)+cos(v)], pi*(-1+i),pi*(-1+i)),  
Ligne( Proj3D(SortFacet(S,i)), 1)
```



```
TeXgraph] view(-1.5,5.5,-1.6,5),size(7.5),
$A:=Origin, $B:=[3*i,0], $C:=[-3+3*i,0], $D:=[-3,0], $E:=[0,3], $F:=[3*i,3], $G:=[-3+3*i,3],
$H:=[-3,3], $M:=A+0.5*(E-A), $R:=C+0.3*(G-C), $S:=A+0.6*[B-A],
$plan:=[M, prodvec(M-S,R-S)], $A1:=interDP([B,C-B], plan), $A2:=interDP([H,G-H],
plan), $A3:=interDP([H,E-H], plan), Width:=6, Cube:=Parallelep(A, B-A, D-A, E-A),
$L:=Intersection(plan, Cube, face), DrawPoly(Cube, 0), SaveAttr(),FillOpacity:=0.5,
LineStyle:=noline, FillStyle:=full, FillColor:=red, DrawPoly(face,1),
RestoreAttr(),DrawPoly(Cube, 1),DrawAretes(L, 0), LabelDot(Proj3D(A), "$A$", "O", 1),
LabelDot(Proj3D(B), "$B$", "S", 1), LabelDot(Proj3D(C), "$C$", "E", 1),
LabelDot(Proj3D(D), "$D$", "O", 1), LabelDot(Proj3D(E), "$E$", "O", 1),
LabelDot(Proj3D(F), "$F$", "N", 1), LabelDot(Proj3D(G), "$G$", "E", 1),
LabelDot(Proj3D(H), "$H$", "N", 1), LabelDot(Proj3D(M), "$M$", "O", 1),
LabelDot(Proj3D(R), "$R$", "E", 1), LabelDot(Proj3D(S), "$S$", "S", 1),
LabelDot(Proj3D(A1), "$A_1$", "E", 1), LabelDot(Proj3D(A2), "$A_2$", "NE", 1),
LabelDot(Proj3D(A3), "$A_3$", "NO", 1)
{export=pdfc}
```



```
TeXgraph] {export=eps}
Marges(0,0,0,0), view(-5,5,-5,5), size(12,1),
S:=GetSurface([u+i*v,3*(sin(u)+cos(v))], pi*(-1+i), pi*(-1+i)),
R:=rectangle3d(S), z1:=Zde(R[1,2]), z2:=Zde(R[3,2]), nb:=8,
pas:=(z2-z1)/nb, transformbox3d(R), Mat:=GetMatrix3D(), IdMatrix3D(),
view3D(-4,4,-4,4,-4,4),
palette:=[purple, darkblue, blue, darkgreen, green, yellow, orange, red],
backcolor:=gray, z:=z1,

Build3D(
for k from 1 to nb-1 do
  Inc(z,pas),
  S:=ClipFacet(S,[M(0,0,z),-vecK], S'), Inc(k,1),
  bdFacet(S,[color:=palette[k], smooth:=1, matrix:=Mat]), Re(jump)-i,
  Echange(S,S')
od,
  bdFacet(S,[color:=crimson, smooth:=1, matrix:=Mat]),
  bdAxes([0,0], [color:=gold, arrows:=1, arrowsscale:=1.5])
),
Display3D()
```

