

## Compléments de géométrie dans l'espace

Ici, tous les points mathématiques sont du type `color` de METAPOST. Pour chaque figure de géométrie dans l'espace

```
figureespace(x_1,y_1,x_2,y_2);
Initialisation(r,t,p,d);
finespace;
```

où  $r$ ,  $t$  et  $p$  sont les coordonnées sphériques de la position de l'observateur et  $d$  la distance à l'écran.

Il y a aussi le paramètre `typerepre` qui peut prendre les valeurs `proj` (défaut) et `persp`.

### Commandes propres à la 3D

`TraceAxes` Trace les axes du repère.

`TraceGrille(nb)` Trace une grille en 3D de l'origine sur un nombre `nb` d'unités.

`Graduations(nb)` Indique les graduations sur la grille.

`Projectionxy(A)`, `Projectionyz(A)`, `Projectionzx(A)`  
Trace la projection de  $A$  sur le plan  $Oxy$  (respectivement sur  $Oyz$  et  $Ozx$ ) en traits d'axes et en rouge.

`Projection(A)` Regroupe les trois projections précédentes.

### Points, droites, plans

`IntersectionDroite(A,B,C,D)` Détermine l'intersection des droites  $(AB)$  et  $(CD)$ .

`IntersectionPlanDroite(A,B,C,D,E)` Détermine l'intersection du plan  $(ABC)$  avec la droite  $(DE)$ .

`IntersectionPlanPlan(A,B,C,D,E,F)` Détermine l'intersection des plans  $(ABC)$  et  $(DEF)$ .

### Solides

`pointilles` Paramètre qui, par défaut, est à oui pour afficher les pointillés dans les solides ci-dessous.

`Cube(A,B,C,D,E,F,G,H)` Trace un cube  $ABCDEFGH$  et définit les points  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$ ,  $F$ ,  $G$ ,  $H$ .

`Pave(A,B,C,D,E,F,G,H)(6,4,3)` Trace un pavé droit  $ABCDEFGH$  de profondeur 6 unités, de largeur 4 unités et de hauteur 3 unités et définit les points  $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$ ,  $F$ ,  $G$ ,  $H$ .

## Notes

<sup>1</sup>uniquement valable en géométrie plane

<sup>2</sup>Merci Jean-Côme Charpentier

<sup>3</sup>uniquement en géométrie spatiale

<sup>4</sup>Ce peut être un autre objet (par exemple `!image!`) de METAPOST

<sup>5</sup>Crée par Anthony Phan

## Rappel des commandes de `geometriesyr16.mp` et `donymodule.mp`

Christophe Poulain

13 décembre 2006

### Commandes générales

`figure(x1,y1,x2,y2)` Commence une nouvelle figure et détermine le cadre qui va entourer la figure.

`figuremainlevee(x1,y1,x2,y2)` Commence une nouvelle figure et détermine le cadre qui va entourer la figure tracée à *main levée*.

`figureespace(x1,y1,x2,y2)` Commence une nouvelle figure et détermine le cadre qui va entourer la figure tracée *en 3D*.

`fin`, `finmainlevee`, `finespace` clôture la figure correspondante.

`feuille` C'est le cadre entourant la figure. On peut le tracer.

### Paramètres généraux

`marque_p` Indique le style du pointage des points : `non` (défaut), `plein`, `creux`, `croix`.

`marque_r` Rayon des cercles de pointage des points. (Défaut : 20).

`marque_a` Rayon des arcs de cercles de codage des angles. (Défaut : 20)

## Procédures d’affichage

`pointe(A)` Pointe le point  $A$  en accord avec `marque_p`.  
`nomme.pos(A)` Nomme le point  $A$  en accord avec `marque_p` et en suivant la position `pos`.  
`trace` Remplace `draw`  
`remplis` Remplace `fill`

## Procédures de codage

Elles doivent être précédées de la commande `trace`.

`codeperp(A,B,C,5)` Code l’angle droit  $\widehat{ABC}$ .

`Marqueangle(d1,d2,n)` Code<sup>1</sup> l’angle formé par les demi-droites  $d_1$  et  $d_2$  avec le codage `n` (1 ; 2 ; 3 : nombres traits ; 4 : croix).

`marqueangle(A,B,C,n)` Code<sup>1</sup> l’angle  $\widehat{ABC}$  de sens direct avec le codage `n` (1 ; 2 ; 3 : nombres traits ; 4 : croix).

`Codeangle(A,B,C,n,btex ... etex)` Code<sup>1</sup> l’angle  $\widehat{ABC}$  de sens direct avec le codage `n` et le texte `btex ... etex`.

`codesegments(A,B,C,D,n)` Code les segments  $[AB]$  et  $[CD]$  avec le codage `n` (1 ; 2 ; 3 : nombres de traits ; 4 : croix ; 5 : cercle creux).

`Codelongueur(A,B,C,D,E,F,n,...)` Code les segments  $[AB]$ ,  $[CD]$ ,  $[EF]$ , ... avec le même codage `n`.

`marque_para(d1,d2,t)` Code l’information que les droites ( $d_1$ ) et ( $d_2$ ) sont parallèles. Le paramètre  $t$  indique la position du codage.

`marquesegment(A,B)` Indique le codage des extrémités du segment  $[AB]$ .

`marquedemidroite(A,B)` Indique le codage de l’extrémité de la demi-droite  $[AB]$ .

`coupdecompas(A,B,n)` Permet de tracer<sup>1</sup> *un coup de compas* de centre  $A$  passant par  $B$  avec « un écart à  $B$  égal à  $\pm n^\circ$  ».

## Objets mathématiques

### Points

`iso(A,B,C,...)` Détermine l’isobarycentre du système  $(A, B, C, \dots)$ .

`milieu(A,B)` Détermine le milieu du segment  $[AB]$ . (Différence visible avec `iso` lors d’une figure à main levée.)

`projection(A,B,C)` Détermine<sup>1</sup> la projection de  $A$  sur la droite  $(BC)$ .

`CentreCercleC(A,B,C)` Détermine<sup>1</sup> le centre du cercle circonscrit à  $ABC$ .

`CentreCercleI(A,B,C)` Détermine<sup>1</sup> le centre du cercle inscrit à  $ABC$ .

`Orthocentre(A,B,C)` Détermine<sup>1</sup> l’orthocentre de  $ABC$ .

### Autres objets : droites

Les commandes doivent être précédées de `trace`.

`segment(A,B)` Trace le segment  $[AB]$  (*sans le codage des extrémités*).

`droite(A,B)` Trace la droite  $(AB)$ .

`demidroite(A,B)` Trace la demi-droite  $[AB]$ .

`bissectrice(A,B,C)` Trace<sup>1</sup> la bissectrice de  $\widehat{CBA}$ .

`parallele(A,B,C)` Trace<sup>1</sup> la parallèle à la droite  $(AB)$  passant par  $C$ .

`perpendiculaire(A,B,C)` Trace<sup>1</sup> la perpendiculaire à la droite  $(AB)$  passant par  $C$ .

`mediatrice(A,B)` Trace<sup>1</sup> la médiatrice du segment  $[AB]$ .

### Autres objets : polygones

Les commandes doivent être précédées de `trace`.

`triangleqcq(A,B,C)` Trace<sup>1</sup> un triangle *vraiment quelconque*<sup>2</sup> et définit les points  $A$ ,  $B$  et  $C$ .

`polygone(A,B,C,...)` Trace le polygone  $ABC \dots$

`chemin(A,B,C,...)` Trace le chemin polygonal *non fermé*  $ABC \dots$

### Autres objets : cercles

`cercles(A,3cm)` Trace<sup>1</sup> le cercle de centre  $A$ , de rayon 3 cm.

`cercles(A,B)` Trace<sup>1</sup> le cercle de centre  $A$  et qui passe par  $B$ .

`cercles(A,B,A,B,C)` Trace<sup>3</sup> le cercle de centre  $A$  passant par  $B$  dans le plan  $(ABC)$ .

`arcercle(A,B,O)` Trace<sup>1</sup> l’arc de cercle  $AB$  de sens direct et de centre  $O$ .

`pointarc(cc,ang)` Repère le point du cercle `cc` et d’angle `ang` (au sens METAPOST).

### Autres objets : les transformations

`rotation(A,B,50)` Détermine<sup>1</sup> l’image de  $A^4$  par la rotation de centre  $B$  et d’angle  $50^\circ$ .

`symetrie(A,B)` Détermine<sup>1</sup> l’image de  $A^4$  par la symétrie centrale de centre  $B$ .

`symetrie(A,B,C)` Détermine<sup>1</sup> l’image de  $A^4$  par la symétrie axiale d’axe  $(BC)$ .

## Divers

Les commandes doivent être précédées de `trace`.

`hachurage(cc,ang,ec,tr)` Hachure complètement le chemin *fermé* `cc`, avec des hachures inclinées d’`ang` espacées de `ec` et de motifs `tr`. (0 : traits pleins ; 1 : pointillés ; 2 : traits d’axes)

`cotation(A,B,ec,dec,btex ... etex)` Trace<sup>1</sup> une flèche de cotation du segment  $[AB]$  décalée de `ec` ; nommée, sur le dessus, par le texte décalé de `dec`.

`cotationmil(A,B,ec,dec,btex ... etex)` Trace une flèche de cotation du segment  $[AB]$  décalée de `ec` et dans laquelle une ouverture par rapport au milieu de `dec` a été faite pour nommé par le texte.

`appellation(A,B,dec,btex ... etex)` Affiche le long du segment  $[AB]$  avec un décalage `dec` le texte.

`transparence cc` <sup>5</sup> Remplit le chemin fermé `cc` en créant une transparence avec la couleur définie par `fillcolor` (défaut :gris).