

# Le modèle Ghyperbolic.mod 0.1

Pour TeXgraph 1.95

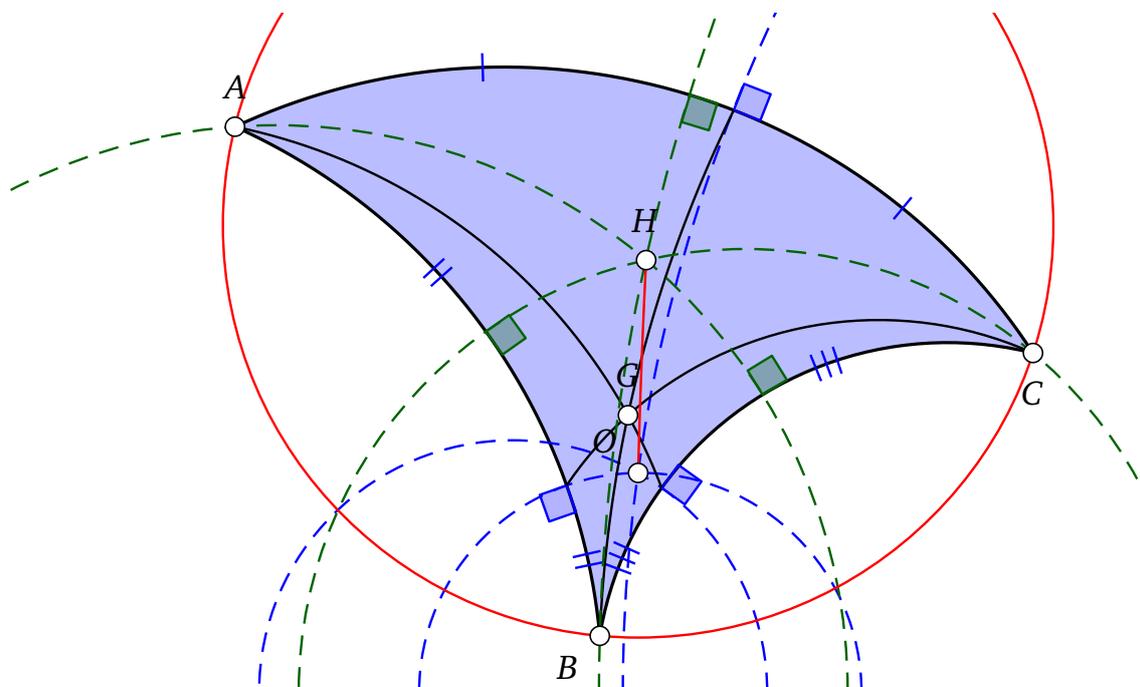
31 mars 2012

## Résumé

Ce modèle permet le dessin de figures élémentaires (droites, segments, lignes polygonales,...) en géométrie hyperbolique dans le demi-plan de Poincaré, ou dans le disque unité.

## Table des matières

	3.3	hSymC	3
<b>1 Introduction</b>	<b>2</b>	<b>4 Macros de calcul</b>	<b>3</b>
<b>2 Macros de dessin</b>	<b>2</b>	4.1 hMed	3
2.1 hDroite	2	4.2 hBissec	3
2.2 hLigne	2	4.3 hPerp	3
2.3 hCercle	2	4.4 hMil	3
2.4 hAngleD	2	4.5 hdist	3
2.5 hMarkseg	2	4.6 hInter	3
<b>3 Macros de transformation</b>	<b>3</b>	4.7 hInterL	3
3.1 hRot	3	4.8 hTangente	3
3.2 hSym	3	4.9 hPolyreg	4
		4.10 fdisc	4
		4.11 fplane	4



## 1 Introduction

Après le chargement du modèle, la variable globale *hmode* permet de sélectionner soit le mode *plane* (par défaut) pour le demi-plan de Poincaré ( $y > 0$ ), soit le mode *disc* (disque unité ( $|z| < 1$ ), en écrivant `hmode:=disc`).

Comme en géométrie euclidienne, les points sont gérés à l'aide de leur affixe, et l'utilisateur dispose de nouvelles macros de dessin pour représenter des droites, des lignes polygonales,... etc.

## 2 Macros de dessin

### 2.1 hDroite

- Syntaxe: `hDroite( < [A,B] > )` ou `hDroite( < A >, < B > )`
- Trace la droite définie par les deux points (affixes)  $\langle A \rangle$  et  $\langle B \rangle$ .

### 2.2 hLigne

- Syntaxe: `hLigne( < liste de points >, < fermée (0/1) > )`
- Dessine la ligne polygonale passant la  $\langle$  liste de points  $\rangle$ . Le deuxième argument indique si la ligne doit être refermée.

### 2.3 hCercle

- Syntaxe: `hCercle( < A >, < r > )`
- Trace le cercle de centre  $\langle A \rangle$  et de rayon  $\langle r \rangle$  (réel strictement positif).

### 2.4 hAngleD

- Syntaxe: `hAngleD( < B >, < A >, < C >, < taille > )`
- Dessine un angle « droit » comme la macro `angleD`, en fait c'est un parallélogramme ( $BACD$ ) qui est dessiné,  $D$  étant calculé par la macro.

### 2.5 hMarkseg

- Syntaxe: `hMarkseg( < A >, < B >, < n >, < espacement >, < hauteur > )`
- Dessine  $n$  petits traits sur le segment défini par  $[A, B]$ , comme la macro `markseg`.



### Exemple

```
\begin{texgraph}[name=exemple1, file]
  \include "Ghyperbolic.mod";
  \graph image = [
    \hmode :=plane, \view(-5.1, 5, 0, 4.5), \marges(0, 0, 0, 0.5), \size(10), \labelSize :=scriptstyle,
    \arrows :=1, \axes(0, 1+i), \arrows :=0, A :=1+i, B :=1+4*i, C :=hRot(A, B, -pi/2), D :=3+2*i,
    \saveAttr(), \lineStyle :=userdash, \hDroite(A, B), \hDroite(B, C), \hDroite(C, A),
    \lineStyle :=solid, \fillStyle :=full, \fillColor :=seagreen, \fillOpacity :=0.5,
    \width :=8, \hLigne([A, B, C], 1), \fillOpacity :=0.9, \fillColor :=crimson, \hCercle(D, 0.75),
    \restoreAttr(), \dotStyle :=times, \dotScale :=1.5, \labelDot(A, "$A$", "E", 1), \labelDot(B, "$B$", "NE", 1), \labelDot(C, "$C$", "NO", 1),
    \labelDot(D, "$D$", "N", 1), \hMarkseg(A, B, 3, 0.1, 0.25), \hMarkseg(C, B, 3, 0.1, 0.25), \hAngleD(C, B, A, 0.25)
  ];
\end{texgraph}
```

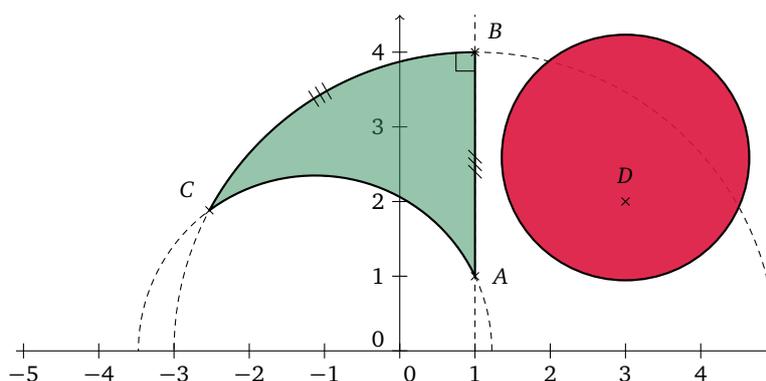


FIGURE 1: Macros de dessin

## 3 Macros de transformation

### 3.1 hRot

- Syntaxe: `hRot( < liste de points >, < centre >, < angle > )`
- Renvoie la liste des images de la < liste de points > par la rotation.

### 3.2 hSym

- Syntaxe: `hSym( < liste de points >, < axe > )`
- Renvoie la liste des images de la < liste de points > par la symétrie axiale, l'< axe > est une liste de deux points représentant une droite.

### 3.3 hSymC

- Syntaxe: `hSymC( < liste de points >, < centre > )`
- Renvoie la liste des images de la < liste de points > par la symétrie centrale.

## 4 Macros de calcul

### 4.1 hMed

- Syntaxe: `hMed( < A >, < B > )`
- Renvoie deux points de la médiatrice du segment défini par les affixes < A > et < B >.

### 4.2 hBissec

- Syntaxe: `hBissec( < B >, < A >, < C >, < extérieure (0/1) > )`
- Renvoie deux points de la bissectrice (intérieure ou extérieure) de l'angle  $\widehat{BAC}$ .

### 4.3 hPerp

- Syntaxe: `hPerp( < [A,B] >, < C > )`
- Renvoie deux points de la perpendiculaire à la droite définie par les affixes < A > et < B >, et passant par < C >.

### 4.4 hMil

- Syntaxe: `hMil( < A >, < B > )`
- Renvoie le milieu du segment défini par les deux points.

### 4.5 hdist

- Syntaxe: `hdist < A >, < B > )`
- Renvoie la distance hyperbolique de < A > à < B >.

### 4.6 hInter

- Syntaxe: `hInter( < droite 1 >, < droite 2 > )`
- Renvoie l'intersection de deux droites hyperboliques, chacune d'elles est une liste de deux points.

### 4.7 hInterL

- Syntaxe: `hInterL( < ligne 1 >, < ligne 2 > )`
- Renvoie la liste des points d'intersection des deux lignes polygonales, chacune d'elles est une liste de points.

### 4.8 hTangente

- Syntaxe: `hTangente( < [A,B] >, < C > )`
- Renvoie un vecteur unitaire tangent à la droite (AB) au point C, celui-ci est supposé appartenir à la droite, le vecteur est orienté le sens A vers B.

## 4.9 hPolyreg

- Syntaxe: `hpolyreg(⟨ n ⟩, ⟨ p ⟩, ⟨ milieux ⟩)`
- Renvoie les sommets d'un polygone régulier de type  $(n, p)$  ( $n$  sommets et  $p$  voisins par sommet) de centre 0 pour le disque unité, et de centre  $i$  pour le demi-plan de Poincaré. Le troisième argument  $\langle \text{milieux} \rangle$  est une variable qui contiendra en sortie la liste des milieux des côtés.

### Exemple

```

\begin{texgraph}[name=exemple2, file]
\include "Ghyperbolic.mod";
\graph image = [
  \hmode :=disc, \view(-1.1, 1.1, -1.1, 1.1), \Marges(0, 0, 0, 0), \size(8),
  \FillStyle :=full, \FillColor :=beige, \Cercle(0, 1),
  P :=hPolyreg(5, 6, m),
  \FillColor :=blue,
  \for z in Mix(P, m) By 2 do
    T :=[0, z[1], z[2]],
    \hLigne(T, 1), \hLigne(hSymC(T, z[2]), 1)
  \od,
  \FillStyle :=none, \Color :=red, \Width :=8, \hLigne(P, 1), \Color :=black, \Point(P)
];
\end{texgraph}

```

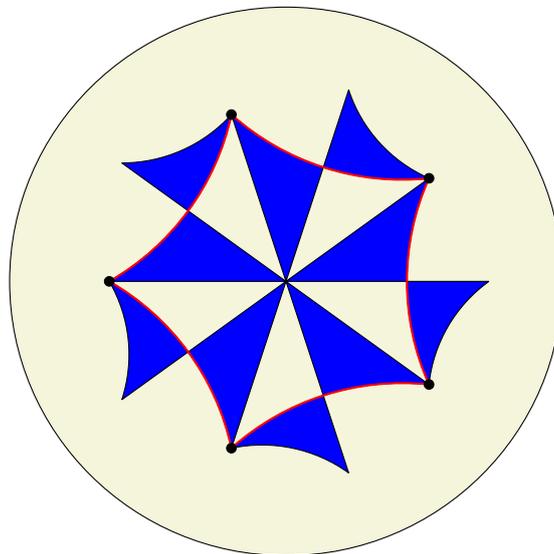


FIGURE 2: Exemple 2

## 4.10 fdisc

- Syntaxe: `fdisc(⟨ liste de complexes ⟩)`
- Renvoie la liste des images de la  $\langle \text{liste de complexes} \rangle$  par l'application  $z \mapsto \frac{z-i}{z+i}$ , celle-ci transforme le demi-plan de Poincaré en le disque unité, le complexe  $\langle z \rangle$  doit donc être dans le demi-plan de Poincaré.

## 4.11 fplane

- Syntaxe: `fplane(⟨ liste de complexes ⟩)`
- Renvoie la liste des images de la  $\langle \text{liste de complexes} \rangle$  par l'application  $z \mapsto i \frac{1+z}{1-z}$ , celle-ci est la réciproque de la précédente et transforme donc le disque unité en le demi-plan de Poincaré, le complexe  $\langle z \rangle$  doit donc être dans le disque unité :  $|z| < 1$ .

### Exemple

```

\begin{texgraph}[name=preface, file]
\include "Ghyperbolic.mod";
\graph image = [
  \hmode :=plane, \view(-5, 5, 0, 6), \Marges(0, 0, 0, 0.5), \size(10), a :=-3+5*i, b :=0.2+0.5*i, c :=4+3*i,
  \Width :=8, \FillStyle :=full, \FillColor :=Rgb(0.73, 0.74, 1),
];

```

```

hLigne([a, b, c], 1), {le triangle}
FillStyle :=none, Color :=blue, Width :=6, LineStyle :=dashed,
{tracé des médianes et calcul de l'intersection}
Med_ab :=hMed(a, b), Med_bc :=hMed(b, c), Med_ca :=hMed(c, a),
hDroite(Med_ab), hDroite(Med_bc), hDroite(Med_ca), O :=hInter(Med_ab, Med_ca),
LineStyle :=solid, FillStyle :=full, FillColor :=blue, FillOpacity :=0.3,
{Calculs des milieux, marquages des angles droits et des segments de meme longueur}
c' :=hMil(a, b), hAngleD(b, c', hSymC(Med_ab[2], c'), 0.25), hMarkseg(a, c', 2, 0.1, 0.25), hMarkseg(b, c', 2, 0.1, 0.25),
a' :=hMil(c, b), hAngleD(c, a', hSymC(Med_bc[2], a'), 0.25), hMarkseg(c, a', 3, 0.1, 0.25), hMarkseg(b, a', 3, 0.1, 0.25),
b' :=hMil(c, a), hAngleD(c, b', Med_ca[2], 0.25), hMarkseg(a, b', 1, 0.1, 0.25), hMarkseg(c, b', 1, 0.1, 0.25),
FillStyle :=none,
Color :=black,
{tracé des médianes et calcul de G}
hLigne([a, a'], 0), hLigne([b, b'], 0), hLigne([c, c'], 0), G :=hInter([a, a'], [b, b']),
Color :=darkgreen, LineStyle :=dashed,
{tracé des hauteurs et calcul de H}
Hc :=hPerp([a, b], c), Hb :=hPerp([a, c], b), Ha :=hPerp([c, b], a), hDroite(Hc), hDroite(Hb), hDroite(Ha), H :=hInter(Hc, Hb),
LineStyle :=solid, FillStyle :=full, FillColor :=darkgreen,
{marquage des angles droits pour les hauteurs}
c' :=hInter(Hc, [a, b]), hAngleD(c, c', b, 0.25),
b' :=hInter(Hb, [a, c]), hAngleD(b, b', c, 0.25),
a' :=hInter(Ha, [b, c]), hAngleD(a, a', c, 0.25), FillStyle :=none,
Color :=red,
{cercle circonscrit et segment [OH]}
hCercle(O, hDist(a, O)), hLigne([O, H], 0),
Color :=black, DotStyle :=dotcircle, DotScale :=1.5, FillColor :=white,
{placement des labels}
LabelDot(O, "$O$", "NO", 1), LabelDot(G, "$G$", "N", 1), LabelDot(H, "$H$", "N", 1), LabelDot(a, "$A$", "N", 1),
LabelDot(b, "$B$", "SO", 1), LabelDot(c, "$C$", "S", 1),
];
\end{texgraph}

```

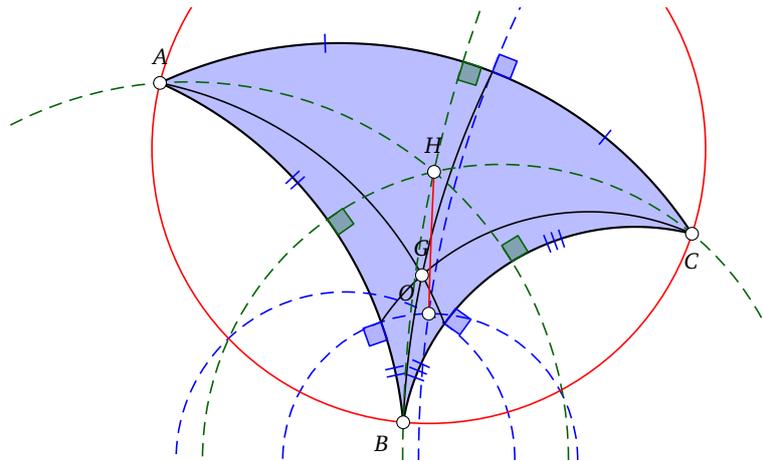


FIGURE 3: Exemple de la première page