

Extension `pst-eucl`
Géométrie euclidienne avec `PSTricks`

Version 1.3.2
Dominique RODRIGUEZ¹

28 mars 2005

¹domino.rodriquez@laposte.net

Résumé

L'extension `pst-eucl` permet de dessiner des figures géométriques en spécifiant des contraintes mathématiques. Il est ainsi possible de définir des points au moyen de transformations ou d'intersection. L'emploi des coordonnées est donc limité aux points de départ qui paramètrent en quelque sorte le dessin.

Remerciements

Je tiens à remercier tout particulièrement les personnes suivantes pour leur aide dans l'élaboration de ce paquetage :

- Denis GIROU pour ses critiques pertinentes et ses encouragements lors de la découverte de l'embryon initial et pour sa relecture du présent manuel ;
- Michael VULIS pour sa rapidité à tester la doc avec `VTeX` et il a ainsi permis de corriger un bug du code PostScript ;
- Manuel LUQUE et Olivier REBOUX pour leurs remarques et leurs exemples.
- Alain DELPLANQUE pour ses propositions de modifications sur le placement automatique des noms des points dans diverses fonctions et la possibilité de donner une suite de points dans `\pstGeonode`.

Avertissement

Ceci est la première version officiellement déposée sur le site d'archives CTAN.

LICENCE

Ce programme et sa documentation peuvent être distribués et/ou modifiés selon les termes de la « L^AT_EX Project Public License » disponible sur le site d'archives CTAN au répertoire `macros/latex/base/lpp1.txt`. Toutefois, vous pouvez m'envoyer un Email avec un petit commentaire. Ensuite vous devriez réfléchir à une donation¹ :

1. directement à l'équipe L^AT_EX3 ;
2. et/ou à moi pour le support de l'extension².

Une donation en temps selon compétence est possible : correction de la documentation, test des fonctionnalités, proposition d'améliorations...

¹surtout si vous utilisez un système d'exploitation payant (**des fenêtres** ou **MacOS**)! D'autre part, n'oubliez pas que L^AT_EX est utilisable gratuitement et que de nombreux utilisateurs achètent plusieurs centaines d'euros des logiciels de bien moins bonne qualité.

²je me satisfais amplement de 1 €, £1 ou \$1, mais j'accepte plus.

Table des matières

| | | | | | |
|----------|---|----------|----------|--|------------------------------|
| 1 | Manuel de l'utilisateur | 2 | | | |
| 1.1 | Spécifications particulières | 2 | | 1.5.5 | Fonction–droite 16 |
| 1.1.1 | Options de PSTricks | 2 | | 1.5.6 | Fonction–cercle 16 |
| 1.1.2 | Conventions | 2 | 2 | Galerie d'exemples | 18 |
| 1.2 | Objets de base | 2 | 2.1 | Géométrie élémentaire | 18 |
| 1.2.1 | Les points | 2 | 2.1.1 | Tracé de la bissectrice | 18 |
| 1.2.2 | Marquage des segments | 4 | 2.1.2 | Transformation de polygones et courbes | 18 |
| 1.2.3 | Triangles | 5 | 2.1.3 | Droites remarquables du tri- angle | 20 |
| 1.2.4 | Angles | 5 | 2.1.4 | Cercle d'EULER | 21 |
| 1.2.5 | Droites, demi-droites et seg- ments | 6 | 2.1.5 | Orthocentre et hyperbole | 21 |
| 1.2.6 | Cercles | 6 | 2.1.6 | Polygone régulier à 17 côtés | 23 |
| 1.2.7 | Arcs de cercles | 7 | 2.1.7 | Tangentes de cercles | 24 |
| 1.2.8 | Abscisse curviligne | 7 | 2.1.8 | Point de FERMAT d'un triangle | 24 |
| 1.2.9 | Courbe générique | 8 | 2.1.9 | Cercles ex-inscrits et inscrit d'un triangle | 25 |
| 1.3 | Transformations | 8 | 2.2 | Quelques lieux de points | 27 |
| 1.3.1 | Symétrie centrale | 9 | 2.2.1 | Parabole | 27 |
| 1.3.2 | Symétrie orthogonale | 9 | 2.2.2 | Hyperbole | 28 |
| 1.3.3 | Rotation | 9 | 2.2.3 | Cycloïde | 29 |
| 1.3.4 | Translation | 10 | 2.2.4 | Hypocycloïdes (astroïde et deltoïde) | 30 |
| 1.3.5 | Homothétie | 10 | 2.3 | Enveloppes de droites et de cercles | 31 |
| 1.3.6 | Projection orthogonale | 11 | 2.3.1 | Coniques | 31 |
| 1.4 | Objets particuliers | 11 | 2.3.2 | Cardioïde | 32 |
| 1.4.1 | Milieu | 11 | 2.4 | Homothéties et fractales | 33 |
| 1.4.2 | Centre de gravité d'un triangle | 11 | 2.5 | Géométrie hyperbolique : triangle et ses hauteurs | 33 |
| 1.4.3 | Cercle circonscrit d'un tri- angle et son centre | 12 | A | Glossaire des commandes | 34 |
| 1.4.4 | Médiatrice d'un segment | 12 | B | Les paramètres de pst-eucl | 38 |
| 1.4.5 | Bissectrices d'un angle | 13 | C | Compatibilité ascendantes de pst-eucl | 40 |
| 1.5 | Intersections | 13 | | | |
| 1.5.1 | Droite-droite | 13 | | | |
| 1.5.2 | Cercle–droite | 14 | | | |
| 1.5.3 | Cercle–cercle | 15 | | | |
| 1.5.4 | Fonction–fonction | 16 | | | |

Chapitre 1

Manuel de l'utilisateur

1.1 Spécifications particulières

1.1.1 Options de PSTricks

L'extension active le mode `\SpecialCoor`. Ce mode permet d'étendre la manière dont les coordonnées sont spécifiées en plus des classiques coordonnées cartésiennes. D'autre part le mode de tracé est mis à `dimen=middle`, c'est-à-dire que le positionnement des tracés est fait par rapport à leur milieu. Pour ces deux modes, je vous renvoie au manuel de l'utilisateur.

Enfin, il est supposé que le repère de travail est (ortho)normé.

1.1.2 Conventions

Pour la rédaction de ce manuel, j'ai utilisé les conventions classiques de dénomination des points en France :

- O est un centre (cercle, repère, symétrie, homothétie, rotation) ;
- I est l'unité de l'axe des abscisses, ou un milieu ;
- J est l'unité de l'axe des ordonnées ;
- A, B, C, D sont des points ;
- M est un point *paramètre* ;
- M' l'image de M par une transformation ;

Enfin, bien que s'agissant de nœuds au sens de PSTricks, je les ai résolument confondus avec des points dans le texte.

1.2 Objets de base

1.2.1 Les points

Repère par défaut

```
\pstGeonode[{par}]( $x_1, y_1$ ){ $\langle A_1 \rangle$ }( $x_2, y_2$ ){ $\langle A_2 \rangle$ }... ( $x_n, y_n$ ){ $\langle A_n \rangle$ }
```

Cette commande définit un ou plusieurs points géométriques associés avec un nœud. Chaque point possède un nom de nœud $\langle A_i \rangle$, qui définit par défaut l'étiquette placée sur le dessin. D'autre part, celle-ci est traitée par défaut en mode mathématique, le paramètre booléen `PtNameMath` (`true` par défaut) permet de modifier ce traitement et de traiter l'étiquette en mode normal. Elle est positionnée à une distance de `PointNameSep` (`1em` par défaut) du nœud, avec un angle de `PosAngle` (`0` par défaut). Il est également possible de proposer une étiquette différente du nom de nœud avec le paramètre `PointName` (`default` par

défaut), et une étiquette vide se spécifie en positionnant ce paramètre à `none` ; dans ce cas le point n'aura pas de nom sur le dessin.

Le symbole du point est donné par le paramètre `PointSymbol` (`*` par défaut). Le symbole est celui utilisé pour la commande `\psdot`. Il peut être également positionné à `none`, auquel cas le point n'est pas affichés sur la figure. Voici en rappel les valeurs utilisable pour le symbole :

- | | | |
|-----------------------------|------------------------------|------------------------------|
| - <code>*</code> : ● | - <code>otimes</code> : ⊗ | - <code>diamond*</code> : ◆ |
| - <code>o</code> : ○ | - <code>triangle</code> : △ | - <code>pentagon</code> : ⬠ |
| - <code>+</code> : + | - <code>triangle*</code> : ▲ | - <code>pentagon*</code> : ⬢ |
| - <code>x</code> : × | - <code>square</code> : □ | - <code> </code> : |
| - <code>asterisk</code> : * | - <code>square*</code> : ■ | |
| - <code>oplus</code> : ⊕ | - <code>diamond</code> : ◇ | |

D'autre part, ces symboles sont paramétrables avec des paramètres de `PSTricks` dont entre autre :

- son échelle avec le paramètre `dotscale`, la valeur de celui-ci est soit deux nombres différenciant ainsi le facteur d'échelle horizontale et verticale, ou soit un seul qui est la valeur identique pour les deux ;
- son angle de rotation avec le paramètre `dotangle`.

N'hésitez pas à consulter la documentation de `PSTricks` pour de plus amples détails.

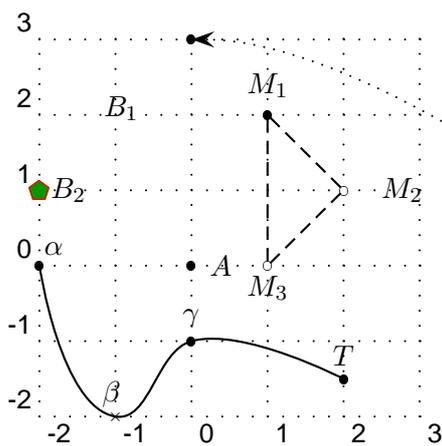
Les paramètres se spécifient explicitement dans la partie `<par>`. Les paramètres `PosAngle`, `PointSymbol`, `PointName` et `PointNameSep` peuvent recevoir :

- soit une valeur unique, commune à tous les points ;
- soit une liste de valeurs délimitée par des accolades `{ ... }` et séparées par des virgules *sans blancs*, permettant alors de différencier la valeur pour chaque point.

Dans le cas où une liste est spécifiée, elle peut comporter moins de valeurs que de points, dans ce cas la dernière valeur de la liste sera utilisée pour les points suivants.

Enfin, le paramètre `CurveType` (`none` par défaut) permet de relier les points au moyen d'une ligne :

- brisée ouverte `polyline` ;
- brisée fermée `polygon` ;
- courbe ouverte `curve`.



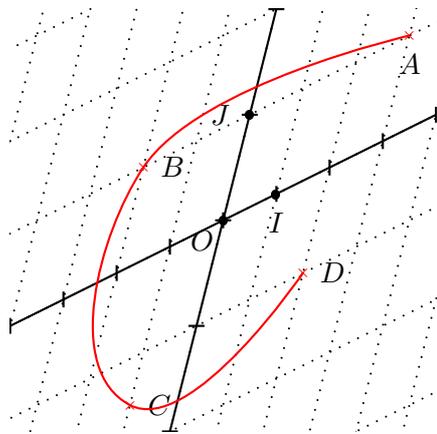
```
\pstGeonode{A}
\pstGeonode[PosAngle=-135, PointNameSep=1.3] (0,3){B_1}
\pstGeonode[PointSymbol=pentagon, dotscale=2, fillstyle=solid,
fillcolor=OliveGreen, PtNameMath=false,
PointName=$B_2$, linecolor=red] (-2,1){B2}
\pstGeonode[PosAngle={90,0,-90}, PointSymbol={*,o},
linestyle=dashed, CurveType=polygon,
PointNameSep={1em,2em,3mm}]
(1,2){M_1}(2,1){M_2}(1,0){M_3}
\pstGeonode[PosAngle={50,100,90}, PointSymbol={*,x,default},
PointNameSep=3mm, CurveType=curve,
PointName={\alpha,\beta,\gamma,default}]
(-2,0){alpha}(-1,-2){beta}(0,-1){gamma}(2,-1.5){T}
```

Il est évident que les nœuds ainsi définis sont utilisables par toutes les commandes utilisant les nœuds de `PSTricks`. Ainsi, il est possible de les référencer d'ici : `\nccurve{->}{ici}{B_1}`.

Repère personnalisé

```
\pstOIJGeonode[<par>](x1, y1){<A1>}{<O>}{<I>}{<J>}(x2, y2){<A2>}... (xn, yn){<An>}
```

Cette commande permet de placer des points dans un repère quelconque non forcément normé ni orthogonal, défini par le triplet $(O; I; J)$. On retrouve des traitements identiques à la commande précédente pour l'utilisation des paramètres.



```
\pstGeonode[PosAngle={-135,-90,180}]{O}(1,0.5){I}(0.5,2){J}
\pstLineAB[nodesep=10]{O}{I}
\pstLineAB[nodesep=10]{O}{J}
\multis(-5,-2.5)(1,0.5){11}{\psline(0,-.15)(0,.15)}
\multis(-2,-8)(0.5,2){9}{\psline(-.15,0)(.15,0)}
\psset{linestyle=dotted}%
\multis(-5,-2.5)(1,0.5){11}{\psline(-10,-40)(10,40)}
\multis(-2,-8)(0.5,2){9}{\psline(-10,-5)(10,5)}
\psset{PointSymbol=x,linestyle=solid}
\pstOIJGeonode[PosAngle={-90,0},CurveType=curve,
               linecolor=red]
(3,1){A}{O}{I}{J}(-2,1){B}(-1,-1.5){C}(2,-1){D}
```

1.2.2 Marquage des segments

Il est possible de tracer des segments en utilisant la commande `\ncline`. Pour en plus marquer les segments afin d'associer ceux ayant la même longueur, il existe la commande :

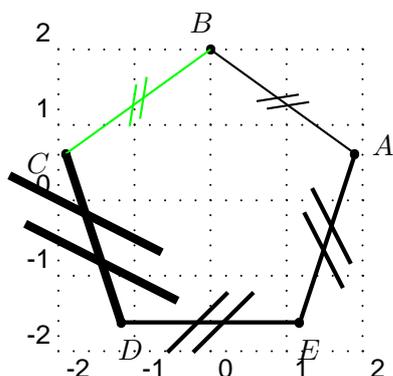
```
\pstSegmentMark[⟨par⟩]{⟨A⟩}{⟨B⟩}
```

Le symbole placé sur le segment est donné par le paramètre `SegmentSymbol`. Il prend comme valeur une commande utilisable en mode mathématique. Par défaut, sa valeur est `pstslashh`, et produit deux barres sur le segment. Le segment est également tracé.

Il existe plusieurs commandes prédéfinies dans ce paquetage pour marquer les segments :

- `\pstslash` : ;
- `\MarkHash` : ;
- `\MarkCros` : ;
- `\pstslashh` : ;
- `\MarkHashh` : ;
- `\MarkCross` : ;
- `\pstslashhh` : ;
- `\MarkHashhh` : ;

Les trois commandes de la famille `MarkHash` produisent une barre dont l'inclinaison est contrôlée par le paramètre `MarkAngle` (45 par défaut). D'autre part, leur taille et leur couleur dépendent de l'épaisseur et de la couleur du trait au moment du tracé, comme le montre l'exemple suivant.



```
\rput{18}{%
  \pstGeonode[PosAngle={0,90,180,-90}](2,0){A}(2;72){B}
  (2;144){C}(2;216){D}(2;288){E}
  \pstSegmentMark{A}{B}
  \pstSegmentMark[linecolor=green]{B}{C}
  \psset{linewidth=2\pslinewidth}
  \pstSegmentMark[linewidth=2\pslinewidth]{C}{D}
  \pstSegmentMark{D}{E}
  \pstSegmentMark{E}{A}}
```

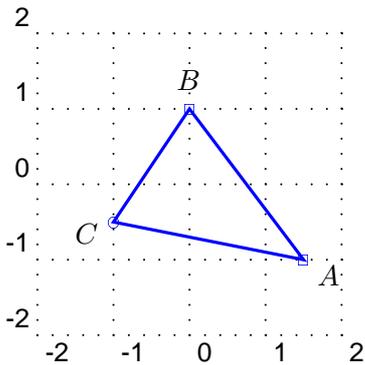
1.2.3 Triangles

Figures étudiées par excellence, elles possèdent leur macro pour les tracer rapidement :

```
\pstTriangle[⟨par⟩](xA,yA){⟨A⟩}(xB,yB){⟨B⟩}(xC,yC){⟨C⟩}
```

Afin de pouvoir placer avec précision les noms des points, il existe trois paramètres `PosAngleA`, `PosAngleB` et `PosAngleC`, qui s'associent respectivement aux nœuds `⟨A⟩`, `⟨B⟩` et `⟨C⟩`. Ils ont le même rôle que le paramètre `PosAngle` vu précédemment. Toutefois, si pour un point donné, aucun angle n'est spécifié, son nom va se positionner sur la bissectrice du sommet.

Il existe également des paramètres pour contrôler le symbole utilisé pour représenter les points : `PointSymbolA`, `PointSymbolB` et `PointSymbolC`, et des paramètres pour le nom de chacun des points : `PointNameA`, `PointNameB` et `PointNameC`. Ces deux ensembles de paramètres sont liés à `PointSymbol` et `PointName`. La gestion de la valeur par défaut suit les mêmes règles que celles vues précédemment.



```
\pstTriangle[PointSymbol=square, PointSymbolC=o,
linecolor=blue, linewidth=1.5\pslinewidth]
(1.5,-1){A}(0,1){B}(-1,-.5){C}
```

1.2.4 Angles

Chaque angle est déterminé par trois points. Le sommet de l'angle est le second. L'ordre a de l'importance en sachant que le sens direct ou trigonométrique est le sens inverse des aiguilles d'une montre. Tout d'abord, il est possible de marquer un angle droit :

```
\pstRightAngle[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨C⟩}
```

Le symbole utilisé est contrôlé par le paramètre `RightAngleType` (default par défaut). Ses valeurs possibles sont :

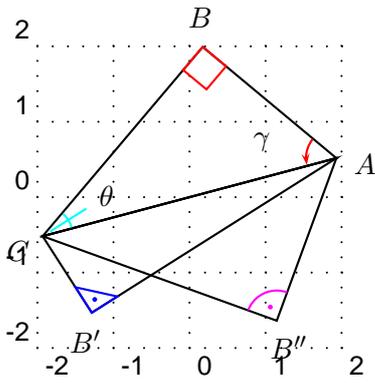
- `default` : symbole standard ;
- `german` : symbole allemand (proposé par U. DIRR) ;
- `suissieromand` : symbole suisse romand (proposé par P. SCHNEWLIN).

L'autre paramètre auquel est sensible cette commande, hormis bien entendu ceux de contrôle du trait, est `RightAngleSize` qui permet de contrôler la longueur du carré utilisé (0.28 unit par défaut). Le symbole est placé sur l'angle de sommet `⟨B⟩`.

Pour les autres angles, il y a la commande :

```
\pstMarkAngle[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨C⟩}{⟨label⟩}
```

L'étiquette définie par le paramètre `label` peut être tout texte T_EX. Elle est placée à `LabelSep` (1 unit par défaut) du point dans la direction angulaire de la bissectrice intérieure modifiée par `LabelAngleOffset` (0 par défaut) et positionnée par rapport à `LabelRefPt` (c par défaut). D'autre part l'arc de cercle utilisé pour le marquage a pour rayon `MarkAngleRadius` (.4 unit par défaut) et il est possible d'y adjoindre une flèche en début ou fin d'arc en utilisant le paramètre `arrows`. Enfin, on peut marquer l'angle avec un marqueur produit par une commande T_EX donnée en argument du paramètre `Mark`.



```

\psset{PointSymbol=none}
\pstTriangle(2;15){A}(2;85){B}(2;195){C}
\psset{PointName=none}
\pstTriangle[PointNameA=default](2;-130){B'}(2;15){A'}(2;195){C'}
\pstTriangle[PointNameA=default](2;-55){B''}(2;15){A''}(2;195){C''}
\pstRightAngle[linecolor=red]{C}{B}{A}
\pstRightAngle[linecolor=blue, RightAngleType=suisseromand]{A}{B'}{C}
\pstRightAngle[linecolor=magenta, RightAngleType=german]{A}{B''}{C}
\psset{arcsep=\pslinewidth}
\pstMarkAngle[linecolor=cyan, Mark=MarkHash]{A}{C}{B}{\theta}
\pstMarkAngle[linecolor=red, arrows=->]{B}{A}{C}{\gamma}

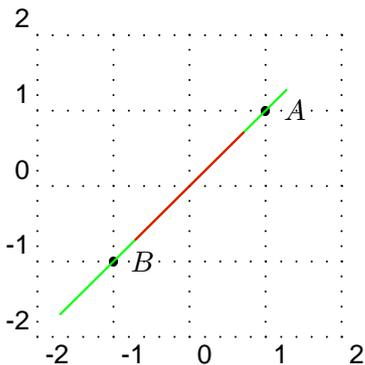
```

1.2.5 Droites, demi-droites et segments

Rien de plus utile qu'une droite!

```
\pstLineAB[par]{A}{B}
```

Afin de contrôler sa longueur¹, les deux paramètres `nodesepA` et `nodesepB` indiquent l'abscisse des extrémités de la droite. Une abscisse négative spécifie un dépassement au-delà des points, tandis qu'une abscisse positive indique le contraire. Si les deux valeurs sont identiques, on peut utiliser alors le paramètre `nodesep`. La valeur par défaut de ces paramètres est nulle.



```

\pstGeonode(1,1){A}(-1,-1){B}
\pstLineAB[nodesepA=-.4, nodesepB=-1, linecolor=green]{A}{B}
\pstLineAB[nodesep=.4, linecolor=red]{A}{B}

```

1.2.6 Cercles

Un cercle peut être défini soit par son centre et un point de sa circonférence, soit par deux points diamétralement opposés. Il existe donc deux fonctions :

```
\pstCircleOA[par]{O}{A}
```

```
\pstCircleAB[par]{A}{B}
```

Pour la première fonction, il est également possible d'omettre le second point, et de spécifier un rayon ou un diamètre avec les paramètres `Radius` et `Diameter` au moyen des deux fonctions suivantes :

```
\pstDistAB{A}{B}
```

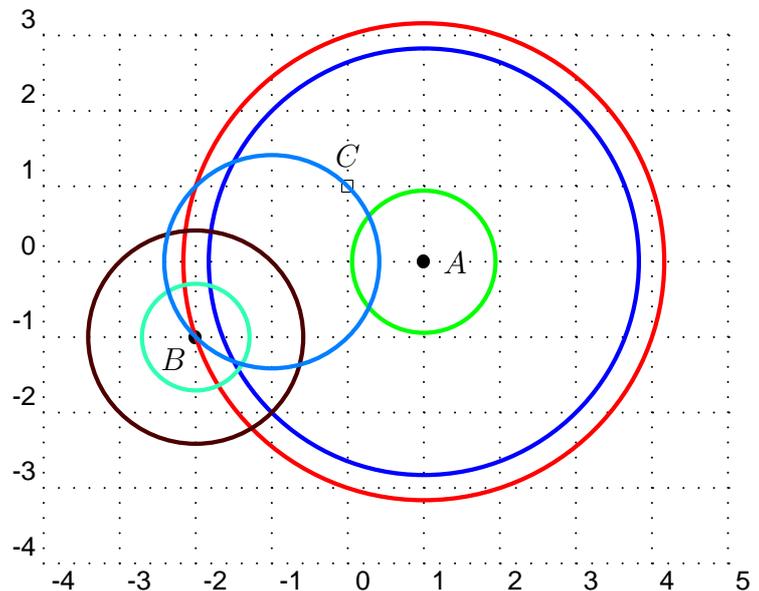
```
\pstDistVal{x}
```

¹Ce qui est le comble pour une droite!

La première fonction permet de spécifier une distance entre deux points et la seconde directement une valeur numérique qui peut être un calcul en code PostScript. Cette dernière distance est spécifiée en multiple de l'unité `\unit` et elle n'est pas une distance. Ces distances peuvent être modifiées par un coefficient multiplicatif spécifié préalablement par le paramètre `DistCoef`. Ce paramètre peut être un calcul en code PostScript.

Avec cette extension, il devient possible de tracer :

- le cercle de centre A passant par B ;
- le cercle de centre A de rayon les deux tiers de AC ;
- le cercle de centre A de rayon BC ;
- le cercle de centre B de rayon AC ;
- le cercle de centre B de diamètre AC ;
- le cercle de diamètre BC ;



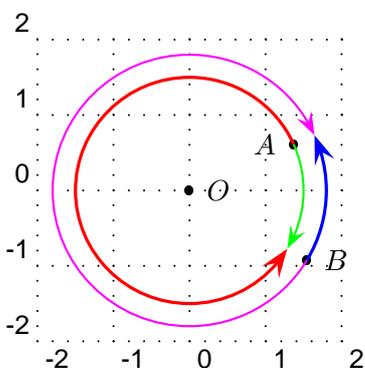
```
\psset{linewidth=2\pslinewidth}
\pstGeonode[PosAngle={0,-135,90},PointSymbol={*,*,square}](1,0){A}(-2,-1){B}(0,1){C}
\pstCircleOA[linecolor=red]{A}{B}
\pstCircleOA[linecolor=green,DistCoef=2/3,Radius=\pstDistAB{A}{C}]{A}{C}
\pstCircleOA[linecolor=blue,Radius=\pstDistAB{B}{C}]{A}{C}
\pstCircleOA[linecolor=Sepia,Radius=\pstDistAB{A}{C}]{B}{C}
\pstCircleOA[linecolor=Aquamarine,Diameter=\pstDistAB{A}{C}]{B}{C}
\pstCircleAB[linecolor=RoyalBlue]{B}{C}
```

1.2.7 Arcs de cercles

```
\pstArcOAB[<par>]{<O>}{<A>}{<B>}
```

```
\pstArcnOAB[<par>]{<O>}{<A>}{<B>}
```

Ces deux fonctions tracent des arcs de cercle, le centre étant donné par O , le rayon par OA , l'angle de départ par A et celui d'arrivée par B . Enfin la première trace l'arc dans le sens direct et la seconde dans le sens indirect. Il n'est pas nécessaire que les deux points soient à égales distances de O .



```
\pstGeonode[PosAngle={180,0}](1.5;24){A}(1.8;-31){B}
\pstGeonode{O}
\psset{arrows=->,arrowscale=2}
\pstArcOAB[linecolor=red,linewidth=1.5\pslinewidth]{O}{A}{B}
\pstArcOAB[linecolor=blue,linewidth=1.5\pslinewidth]{O}{B}{A}
\pstArcnOAB[linecolor=green]{O}{A}{B}
\pstArcnOAB[linecolor=magenta]{O}{B}{A}
```

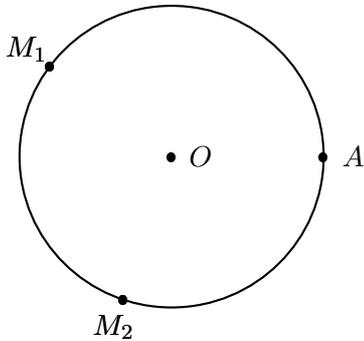
1.2.8 Abscisse curviligne

Un point peut être positionné sur un cercle au moyen d'une abscisse curviligne.

```
\pstCurvAbsNode[⟨par⟩]{⟨O⟩}{⟨A⟩}{⟨B⟩}{⟨Abs⟩}
```

Le point $\langle B \rangle$ est placé sur le cercle de centre $\langle O \rangle$ passant par $\langle A \rangle$, avec l'abscisse curviligne $\langle Abs \rangle$. L'origine est en $\langle A \rangle$, et le sens est par défaut le sens trigonométrique (inverse des aiguilles d'une montre). Le paramètre `CurvAbsNeg` (`false` par défaut) permet de modifier le sens de parcours du cercle.

Si le paramètre `PosAngle` n'est pas spécifié, l'étiquette associée au point va se placer automatiquement de manière à être aligné avec le centre du cercle et le point.



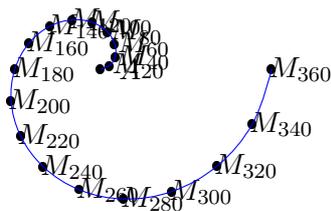
```
\pstGeonode{0}(2,0){A}
\pstCircleOA{0}{A}
\pstCurvAbsNode{0}{A}{M_1}{\pstDistVal{5}}
\pstCurvAbsNode[CurvAbsNeg=true]{0}{A}{M_2}{\pstDistAB{A}{M_1}}
```

1.2.9 Courbe générique

Il est tout à fait possible de générer des points au moyen d'une boucle et de leur donner à chacun un nom générique composé d'un radical et d'un nombre. La commande suivante permet de relier ces points au moyen d'une courbe interpolée.

```
\pstGenericCurve[⟨par⟩]{⟨Radical⟩}{⟨n1⟩}{⟨n2⟩}
```

La courbe est tracée sur les points dont le nom comporte le radical $\langle Radical \rangle$, et comme suffixe un nombre allant de $\langle n_1 \rangle$ à $\langle n_2 \rangle$. Pour gérer les effets de bords, les paramètres `GenCurvFirst` et `GenCurvLast` permettent de désigner explicitement le premier et le dernier point. Le paramètre `GenCurvInc` permet de spécifier l'incrément de boucle pour passer d'un point à un autre (1 par défaut).



```
\psset{unit=.00625}
\pstGeonode{A}
\multido{\n=20+20}{18}{\pstGeonode[PointName=M_{\n}]{\n;\n}{M_{\n}}}
\pstGenericCurve[GenCurvFirst=A, GenCurvInc=20,
linecolor=blue, linewidth=.5\pslinewidth]{M_{20}}{M_{360}}
```

1.3 Transformations

Les transformations du plan permettent de coder facilement des propriétés géométriques. Il est donc possible de construire les images de points par toutes les transformations classiques du plan. Toutes ces commandes partagent le même schéma syntaxique et deux paramètres.

La syntaxe commune place en fin de description deux listes de points, dont la seconde est optionnelle et de cardinal plus faible ou égal. Ces deux listes contiennent les points antécédents et les points images. Dans le cas où les images sont absentes dans leur totalité ou pour une partie (liste moins longue), le nom de l'image sera automatiquement calqué sur l'antécédent auquel sera adjoint un '.

Le premier paramètre commun est `CodeFig`. Il permet de faire apparaître les propriétés caractéristiques liées à la transformation et à la construction de l'image. Par défaut, ce paramètre a pour valeur `false`, il doit être mis

à `true` pour activer ce tracé optionnel. Ce tracé se fait en utilisant le style `CodeFigStyle` (dashed par défaut) et suivant la couleur `CodeFigColor` (cyan par défaut).

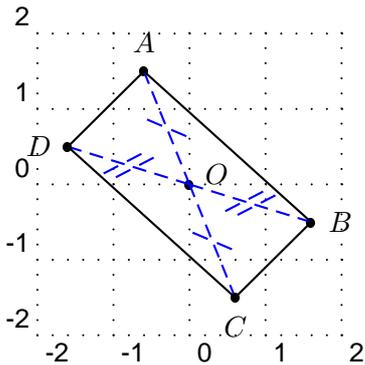
Le second paramètre est `CurveType` qui permet de faire apparaître une courbe passant par les images, et donc de pouvoir rapidement tracer le transformé d'une figure.

1.3.1 Symétrie centrale

```
\pstSym0[⟨par⟩]{⟨O⟩}{⟨M1, M2, ⋯, Mn⟩}{⟨M'1, M'2, ⋯, M'p⟩}
```

Construit le point symétrique par rapport au point O . Les paramètres classiques de la création de point sont utilisables comme dans toutes les fonctions suivantes.

Elle peut servir à la construction d'un parallélogramme de centre connu.

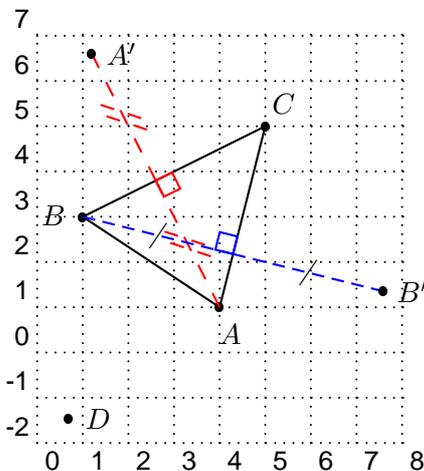


```
\psset{CodeFig=true}
\pstGeonode[PosAngle={20,90,0}]{O}{(-.6,1.5){A}(1.6,-.5){B}
\pstSym0[CodeFigColor=blue, PosAngle={-90,180}]{O}{A, B}[C, D]
%\pstSym0[SegmentSymbol=pstslash, PosAngle=180]
% {O}{B}{D}
\pstLineAB{A}{B}\pstLineAB{C}{D}
\pstLineAB{A}{D}\pstLineAB{C}{B}
```

1.3.2 Symétrie orthogonale

```
\pstOrtSym[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨M1, M2, ⋯, Mn⟩}{⟨M'1, M'2, ⋯, M'p⟩}
```

Construit le point symétrique M'_i de chacun des M_i par rapport à la droite (AB) .



```
\pstTriangle(1,3){B}(5,5){C}(4,1){A}
\pstOrtSym{A}{B}{C}[D]
\psset{CodeFig=true}
\pstOrtSym[dash=2mm 2mm, CodeFigColor=red]
{C}{B}{A}
\pstOrtSym[SegmentSymbol=pstslash, linestyle=dotted,
dotsep=3mm, CodeFigColor=blue]
{C}{A}{B}
```

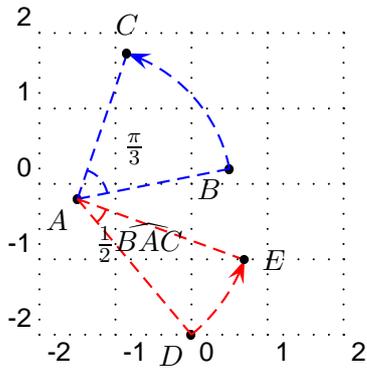
1.3.3 Rotation

```
\pstRotation[⟨par⟩]{⟨O⟩}{⟨M1, M2, ⋯, Mn⟩}{⟨M'1, M'2, ⋯, M'p⟩}
```

Construit l'image d'un point par rotation. L'angle de rotation est donné par le paramètre `RotAngle` (60 par défaut). Celui-ci peut-être un angle orienté défini par trois points. Il faut alors utiliser la fonction :

```
\pstAngleAOB{⟨A⟩}{⟨O⟩}{⟨B⟩}
```

Pensez à l'utiliser pour construire un carré ou un triangle équilatéral. Le paramètre booléen `CodeFig` permet de placer un arc avec une flèche (en bout) entre le point et son image, et si le paramètre `TransformLabel` (`none` par défaut) contient un texte il est placé sur l'angle correspondant en mode mathématique.

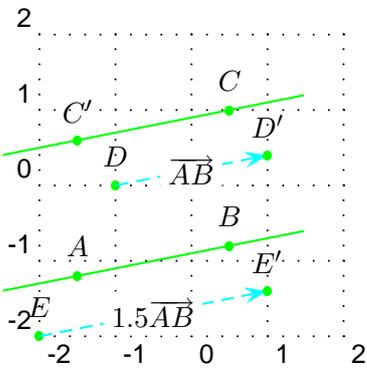


```
\psset{arrowscale=2}
\pstGeonode[PosAngle=-135](-1.5,-.2){A}(.5,.2){B}(0,-2){D}
\pstRotation[PosAngle=90, RotAngle=60, CodeFig=true,
CodeFigColor=blue,
TransformLabel=\frac{\pi}{3}] {A}{B}{C}
\pstRotation[AngleCoef=.5, RotAngle=\pstAngleAOB]{B}{A}{C},
CodeFigColor=red, CodeFig=true,
TransformLabel=\frac{1}{2}\widehat{BAC}] {A}{D}{E}
```

1.3.4 Translation

```
\pstTranslation[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨M1, M2, ..., Mn⟩}{⟨M'1, M'2, ..., M'p⟩}
```

Construit le translaté de vecteur \overrightarrow{AB} du point $\langle M \rangle$. À utiliser pour construire la parallèle passant par un point.



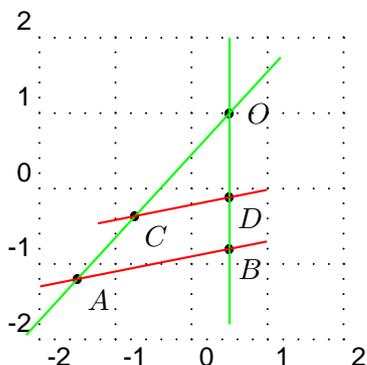
```
\psset{linecolor=green, nodesep=-1, PosAngle=90,arrowscale=2}
\pstGeonode(-1.5,-1.2){A}(.5,-.8){B}(.5,1){C}(-1,0){D}(-2,-2){E}
\pstTranslation{B}{A}{C}
\psset{CodeFig=true, TransformLabel=default}
\pstTranslation{A}{B}{D}
\pstTranslation[DistCoef=1.5]{A}{B}{E}
\pstLineAB{A}{B}\pstLineAB{C}{C'}
```

Il est également possible d'utiliser le paramètre `DistCoef` pour modifier le vecteur de translation avec un coefficient multiplicatif. Le paramètre booléen `CodeFig` permet de placer le vecteur de translation entre le point et son image avec au milieu un texte qui par défaut est les deux points ou le contenu du paramètre `TransformLabel` (`none` par défaut).

1.3.5 Homothétie

```
\pstHomO[⟨par⟩]{⟨O⟩}{⟨M1, M2, ..., Mn⟩}{⟨M'1, M'2, ..., M'p⟩}
```

Construit l'image d'un point par homothétie. Le coefficient se spécifie avec le paramètre `HomCoef`. Incontournable pour les situations de THALÈS.

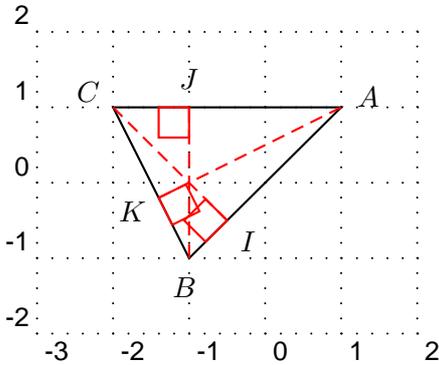


```
\pstGeonode[PosAngle={0,-45}](.5,1){O}(-1.5,-1.2){A}(.5,-.8){B}
\pstHomO[HomCoef=.62, PosAngle=-45]{O}{A,B}{C,D}
\psset{linecolor=green, nodesep=-1}
\pstLineAB{A}{O}\pstLineAB{B}{O}
\psset{linecolor=red, nodesep=-.5}
\pstLineAB{A}{B}\pstLineAB{C}{D}
```

1.3.6 Projection orthogonale

```
\pstProjection[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨M1, M2, ⋯, Mn⟩}{⟨M'1, M'2, ⋯, M'p⟩}
```

Projette orthogonalement les point M_i sur la droite (AB) . Très important pour tracer les hauteurs d'un triangle. Le nom est aligné avec le point et son projeté comme le montre l'exemple.



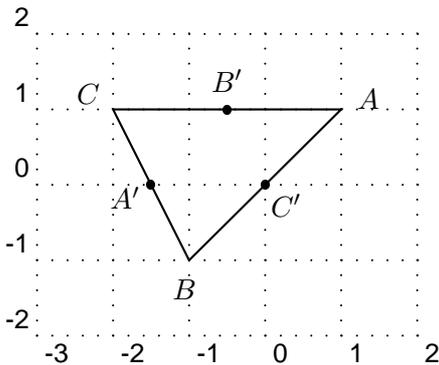
```
\psset{PointSymbol=none,CodeFig=true,CodeFigColor=red}
\pstTriangle(1,1){A}(-2,1){C}(-1,-1){B}
\pstProjection{A}{B}{C}[I]
\pstProjection{A}{C}{B}[J]
\pstProjection{C}{B}{A}[K]
```

1.4 Objets particuliers

1.4.1 Milieu

```
\pstMiddleAB[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨I⟩}
```

Construit le milieu I du segment $[AB]$. Par défaut, le nom du point est positionné automatiquement sous (ou sur) le segment.

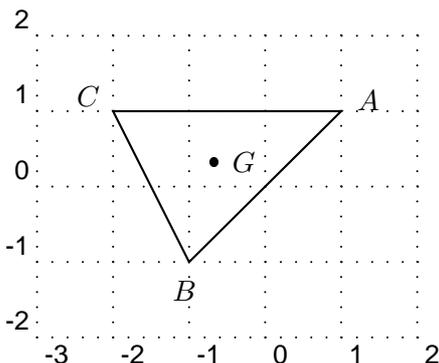


```
\pstTriangle[PointSymbol=none](1,1){A}(-1,-1){B}(-2,1){C}
\pstMiddleAB{A}{B}{C'}
\pstMiddleAB{C}{A}{B'}
\pstMiddleAB{B}{C}{A'}
```

1.4.2 Centre de gravité d'un triangle

```
\pstCGravABC[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨C⟩}{⟨G⟩}
```

Construit le centre de gravité d'un triangle.

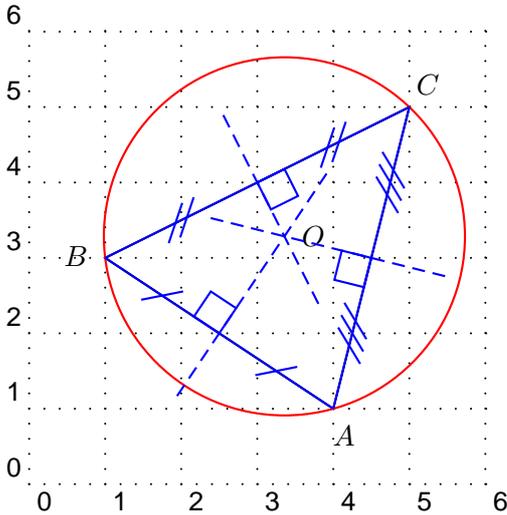


```
\pstTriangle[PointSymbol=none](1,1){A}(-1,-1){B}(-2,1){C}
\pstCGravABC{A}{B}{C}{G}
```

1.4.3 Cercle circonscrit d'un triangle et son centre

```
\pstCircleABC[par]{A}{B}{C}{O}
```

Cette commande permet de tracer le cercle passant par trois points et de positionner le centre. Le tracé du cercle est contrôlé par le paramètre booléen `DrawCirABC` (`true` par défaut). D'autre part, les traits de constructions peuvent être apparent (médiatrices) avec le paramètre `CodeFig`. Les milieux sont aussi marqué sur chacun des segments avec trois marques différentes donnés par les paramètres `SegmentSymbolA`, `SegmentSymbolB` et `SegmentSymbolC`. La création du point est sensible à tous les paramètres classiques déjà vus.



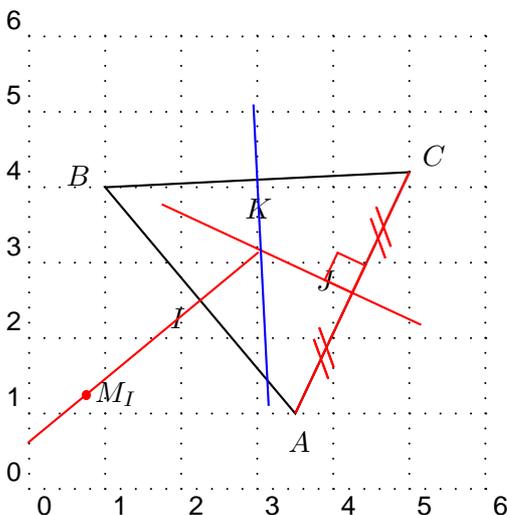
```
\pstTriangle[PointSymbol=none](4,1){A}(1,3){B}(5,5){C}
\pstCircleABC[CodeFig=true, CodeFigColor=blue,
linecolor=red, PointSymbol=none]{A}{B}{C}{O}
```

1.4.4 Médiatrice d'un segment

```
\pstMediatorAB[par]{A}{B}{I}{M}
```

La médiatrice d'un segment est la droite perpendiculaire au milieu de ce segment. Le segment est $[AB]$, le milieu est I , et M est un point de la médiatrice, construit par une rotation de 90° du point B par rapport à I . L'ordre des deux points est important, il permet de contrôler la position de la médiatrice. La commande crée les points M et I . La construction est sensible aux paramètres suivant :

- `CodeFig`, `CodeFigColor` et `SegmentSymbol` pour marquer l'angle droit et le milieu ;
- `PointSymbol` et `PointName` pour contrôler l'apparence des deux nouveaux points, chacun pouvant être spécifiés séparément au moyen des paramètres `...A` et `...B` ;
- les paramètres de tracés des droites.



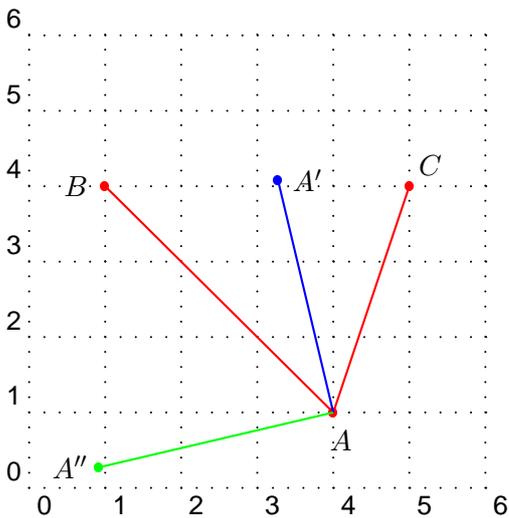
```
\pstTriangle[PointSymbol=none](3.5,1){A}(1,4){B}(5,4.2){C}
\psset{linecolor=red, CodeFigColor=red, nodesep=-1}
\pstMediatorAB[PointSymbolA=none]{A}{B}{I}{M_I}
\psset{PointSymbol=none, PointNameB=none}
\pstMediatorAB[CodeFig=true]
{A}{C}{J}{M_J}
\pstMediatorAB[PosAngleA=45, linecolor=blue]
{C}{B}{K}{M_K}
```

1.4.5 Bissectrices d'un angle

```
\pstBissectBAC[⟨par⟩]{⟨B⟩}{⟨A⟩}{⟨C⟩}{⟨M⟩}
```

```
\pstOutBissectBAC[⟨par⟩]{⟨B⟩}{⟨A⟩}{⟨C⟩}{⟨M⟩}
```

Il existe deux bissectrices pour un angle géométrique donné : l'intérieure et l'extérieure ; donc il y a deux commandes. L'angle est spécifié dans le sens trigonométrique. Les résultats de ces deux commandes sont la bissectrice et un point de celle-ci qui est positionné par une rotation du point B .



```
\psset{CurveType=polyline,linecolor=red}
\pstGeonode[PosAngle={180,-75,45}](1,4){B}(4,1){A}(5,4){C}
\pstBissectBAC[linecolor=blue]{C}{A}{B}{A'}
\pstOutBissectBAC[linecolor=green,PosAngle=180]
{C}{A}{B}{A''}
```

1.5 Intersections

Un autre moyen de définir un point est de spécifier une intersection. Six types d'intersections sont gérées :

- droite-droite ;
- droite-cercle ;
- cercle-cercle ;
- fonction-fonction ;
- fonction-droite ;
- fonction-cercle.

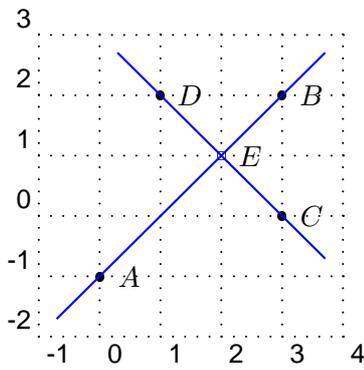
Une intersection peut ne pas exister : cas des droites parallèles par exemple. Dans ce cas, le ou les points concernés se retrouvent placés à l'origine. En fait, il est supposé que l'utilisation de ces commandes implique l'existence de l'intersection.

1.5.1 Droite-droite

```
\pstInterLL[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨C⟩}{⟨D⟩}{⟨M⟩}
```

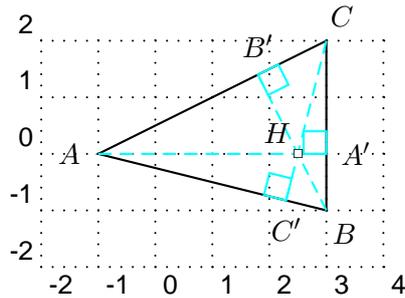
Construit le point d'intersection de la droite (AB) avec la droite (CD) . Option essentielle de la construction de figures du triangle, elle permet de positionner les points remarquables. Les paragraphes suivants proposent différents exemples :

basique



```
\pstGeonode(0,-1){A}(3,2){B}(3,0){C}(1,2){D}
\pstInterLL[PointSymbol=square]{A}{B}{C}{D}{E}
\psset{linecolor=blue, nodesep=-1}
\pstLineAB{A}{B}\pstLineAB{C}{D}
```

orthocentre



```
\psset{CodeFig=true, PointSymbol=none}
\pstTriangle[PosAngleA=180](-1,0){A}(3,-1){B}(3,2){C}
\pstProjection[PosAngle=-90]{B}{A}{C}
\pstProjection{B}{C}{A}
\pstProjection[PosAngle=90]{A}{C}{B}
\pstInterLL[PosAngle=135, PointSymbol=square]
{A}{A'}{B}{B'}{H}
```

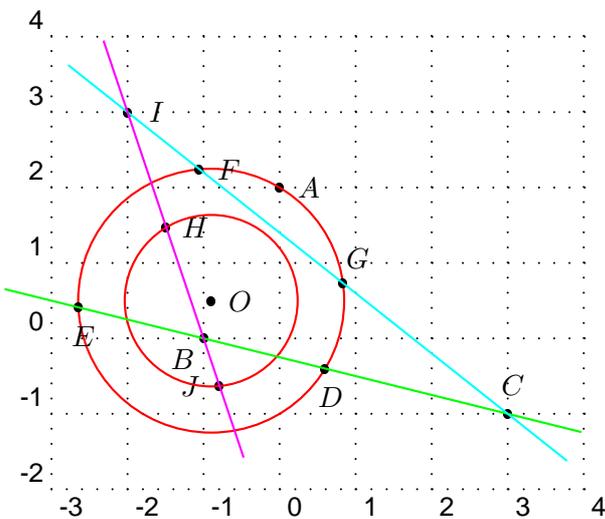
1.5.2 Cercle–droite

```
\pstInterLC[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨O⟩}{⟨C⟩}{⟨M₁⟩}{⟨M₂⟩}
```

Construit le ou les points d'intersection de la droite (AB) avec le cercle de centre O passant par OC . L'intersection entre un cercle et une droite donne en cas d'existence un ou deux points. En fait le cas tangent représente un point double au sens des racines d'un polynôme.

Le cercle est spécifié avec son centre et soit un point de sa circonférence, soit son rayon ou diamètre spécifié au moyen des paramètres `Radius` et `Diameter` et modulé par le paramètre coefficient multiplicatif `DistCoef`.

La place des deux points est telle que les vecteurs \overrightarrow{AB} et $\overrightarrow{M_1M_2}$ soient colinéaires. Ainsi si on échange les points qui définissent la droite, les deux points d'intersection seront aussi échangés. Si l'intersection n'existe pas, les deux points sont placés au centre du cercle.

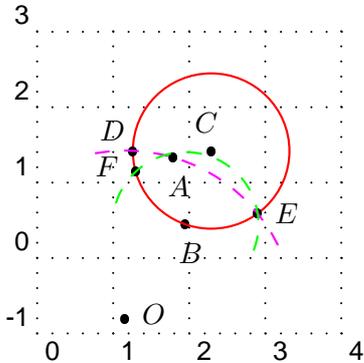


```
\pstGeonode[PosAngle={-135,80,0}](-1,0){B}(3,-1){C}(-.9,.5){O}
\pstGeonode(-2,3){I}
\pstCircleOA[linecolor=red]{O}{A}
\pstInterLC[PosAngle=-80]{C}{B}{O}{A}{D}{E}
\pstInterLC[PosAngleB=60, Radius=\pstDistAB{O}{D}]
{I}{C}{O}{F}{G}
\pstInterLC[PosAngleB=180, DistCoef=1.3,
Diameter=\pstDistAB{O}{D}]
{I}{B}{O}{H}{J}
\pstCircleOA[linecolor=red, DistCoef=1.3,
Diameter=\pstDistAB{O}{D}]{O}{A}
\psset{nodesep=-1}
\pstLineAB[linecolor=green]{E}{C}
\pstLineAB[linecolor=cyan]{I}{C}
\pstLineAB[linecolor=magenta]{J}{I}
```

1.5.3 Cercle–cercle

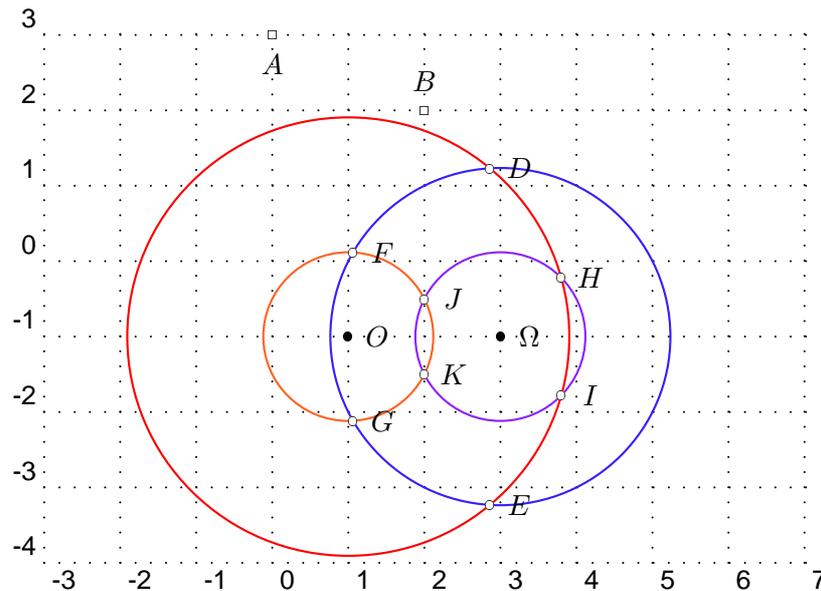
```
\pstInterCC[⟨par⟩]{⟨O1⟩}{⟨B⟩}{⟨O2⟩}{⟨C⟩}{⟨M1⟩}{⟨M2⟩}
```

Fonction au comportement identique à la précédente. Les paramètres booléens `CodeFigA` et `CodeFigB` permettent de faire apparaître les arcs au niveau des intersections. Par cohérence `CodeFig` permet de faire apparaître les deux arcs. Afin de contrôler dans quel sens vont-êre ces arcs, les paramètres booléens `CodeFigArc` et `CodeFigBarc` permettent de choisir dans le sens direct ou indirect.



```
\rput{10}{%
  \pstGeonode[PosAngle={0,-90,-90,90}]
    (1,-1){O}(2,1){A}(2,0.1){B}(2.5,1){C}
  \pstCircleOA[linecolor=red]{C}{B}
  \pstInterCC[PosAngleA=135, CodeFigA=true, CodeFigArc=false,
    CodeFigColor=magenta]{O}{A}{C}{B}{D}{E}
  \pstInterCC[PointSymbolB=none, PointNameB=none,
    PosAngleA=170, CodeFigA=true, CodeFigArc=false,
    CodeFigColor=green]{B}{E}{C}{B}{F}{G}
```

Voici un deuxième exemple plus complet, intégrant les spécifications de cercle avec des rayons et des diamètres. Il existe dans ce cas des paramètres `RadiusA`, `RadiusB`, `DiameterA` et `DiameterB`. Toutefois, même dans le cas où le diamètre est spécifié, il faut donner un centre. On ne peut pas spécifier un cercle avec deux points diamétralement opposés.



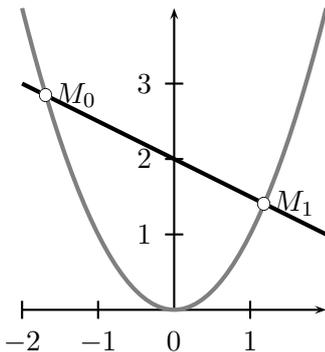
```
\pstGeonode[PointName={\Omega,0}] (3,-1){Omega}(1,-1){O}
\pstGeonode[PointSymbol=square, PosAngle={-90,90}] (0,3){A}(2,2){B}
\psset{PointSymbol=o}
\pstCircleOA[linecolor=red, DistCoef=1 3 10 div add, Radius=\pstDistAB{A}{B}]{O}{}
\pstCircleOA[linecolor=Orange, Diameter=\pstDistAB{A}{B}]{O}{}
\pstCircleOA[linecolor=Violet, Radius=\pstDistAB{A}{B}]{Omega}{}
\pstCircleOA[linecolor=Purple, Diameter=\pstDistAB{A}{B}]{Omega}{}
\pstInterCC[DistCoef=1 3 10 div add, RadiusA=\pstDistAB{A}{B},
  DistCoef=none, RadiusB=\pstDistAB{A}{B}]{O}{}{Omega}{}{D}{E}
\pstInterCC[DiameterA=\pstDistAB{A}{B}, RadiusB=\pstDistAB{A}{B}]{O}{}{Omega}{}{F}{G}
\pstInterCC[DistCoef=1 3 10 div add, RadiusA=\pstDistAB{A}{B},
  DistCoef=none, DiameterB=\pstDistAB{A}{B}]{O}{}{Omega}{}{H}{I}
\pstInterCC[DiameterA=\pstDistAB{A}{B}, DiameterB=\pstDistAB{A}{B}]{O}{}{Omega}{}{J}{K}
```

1.5.4 Fonction–fonction

```
\pstInterFF[⟨par⟩]{⟨f⟩}{⟨g⟩}{⟨x0⟩}{⟨M⟩}
```

Cette fonction permet de positionner un point à une intersection entre les deux fonctions f et g . La valeur x_0 est une valeur approchée de l'abscisse de cette intersection. Il est donc tout à fait possible de l'utiliser plusieurs fois si plusieurs intersections existent. Chacune des fonctions est décrite en PostScript de manière identique à la fonction `\psplot` de `PSTricks`. Il est évidemment possible de spécifier une fonction constante, et par exemple de chercher des racines.

La recherche de l'intersection se fait en utilisant l'algorithme de NEWTON, et évidemment il peut arriver que cela n'aboutisse pas. Dans ce cas, le point est placé à l'origine. D'autre part, la recherche peut tomber dans un piège (minimum local proche de zéro), ou encore se mettre à osciller et donc ne jamais finir, il faudra alors modifier x_0 .

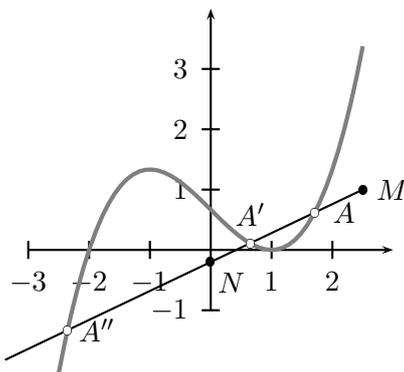


```
\psaxes{->}(0,0)(-2,0)(2,4)
\psset{linewidth=2\pslinewidth}
\psplot[linecolor=gray]{-2}{2}{x 2 exp}
\psplot{-2}{2}{2 x 2 div sub}
\psset{PointSymbol=o}
\pstInterFF{2 x 2 div sub}{x 2 exp}{1}{M_1}
\pstInterFF{2 x 2 div sub}{x 2 exp}{-2}{M_0}
```

1.5.5 Fonction–droite

```
\pstInterFL[⟨par⟩]{⟨f⟩}{⟨A⟩}{⟨B⟩}{⟨x0⟩}{⟨M⟩}
```

Positionne un point à une intersection de la fonction f et de la droite (AB) .

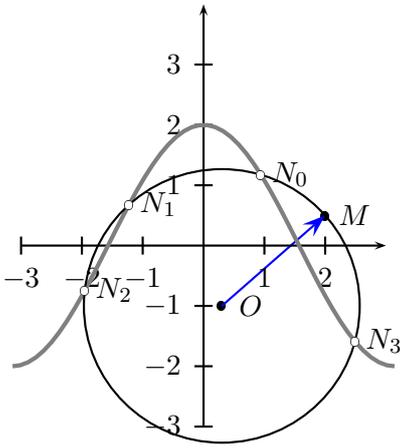


```
\def\F{x 3 exp 3 div x sub 2 3 div add .0001 add}
\psaxes{->}(0,0)(-3,-1)(3,4)
\psplot[linewidth=2\pslinewidth, linecolor=gray]{-2.5}{2.5}{\F}
\psset{PointSymbol=*}
\pstGeonode[PosAngle={-45,0}](0,-.2){N}(2.5,1){M}
\pstLineAB[nodesepA=-3cm]{N}{M}
\psset{PointSymbol=o}
\pstInterFL{\F}{N}{M}{2}{A}
\pstInterFL[PosAngle=90]{\F}{N}{M}{0}{A'}
\pstInterFL{\F}{N}{M}{-2}{A''}
```

1.5.6 Fonction–cercle

```
\pstInterFC[⟨par⟩]{⟨f⟩}{⟨O⟩}{⟨A⟩}{⟨x0⟩}{⟨M⟩}
```

Positionne un point à une intersection de la fonction f et du cercle de centre O passant par A . L'intersection est trouvée en utilisant l'algorithme itératif de NEWTON avec comme valeur initiale x_0 . Le point aura donc une abscisse *proche* de x_0 .



```

\def\F{x 180 mul 3.1415926 div cos 2 mul}
\pstGeonode(0.3,-1){O}(2,.5){M}
\ncline[linecolor=blue, arrowscale=2]{->}{O}{M}
\psaxes{->}(0,0)(-3,-3)(3,4)
\psplot[linewidth=2\pslinewidth, linecolor=gray]{-3.14}{3.14}{\F}
\psset{PointSymbol=*}
\pstCircleOA{O}{M}
\psset{PointSymbol=o}
\pstInterFC{\F}{O}{M}{1}{N_0}
\pstInterFC{\F}{O}{M}{-1}{N_1}
\pstInterFC{\F}{O}{M}{-2}{N_2}
\pstInterFC{\F}{O}{M}{2}{N_3}

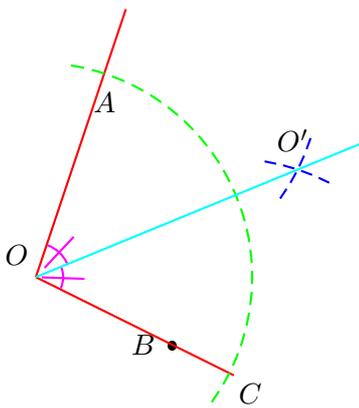
```

Chapitre 2

Galerie d'exemples

2.1 Géométrie élémentaire

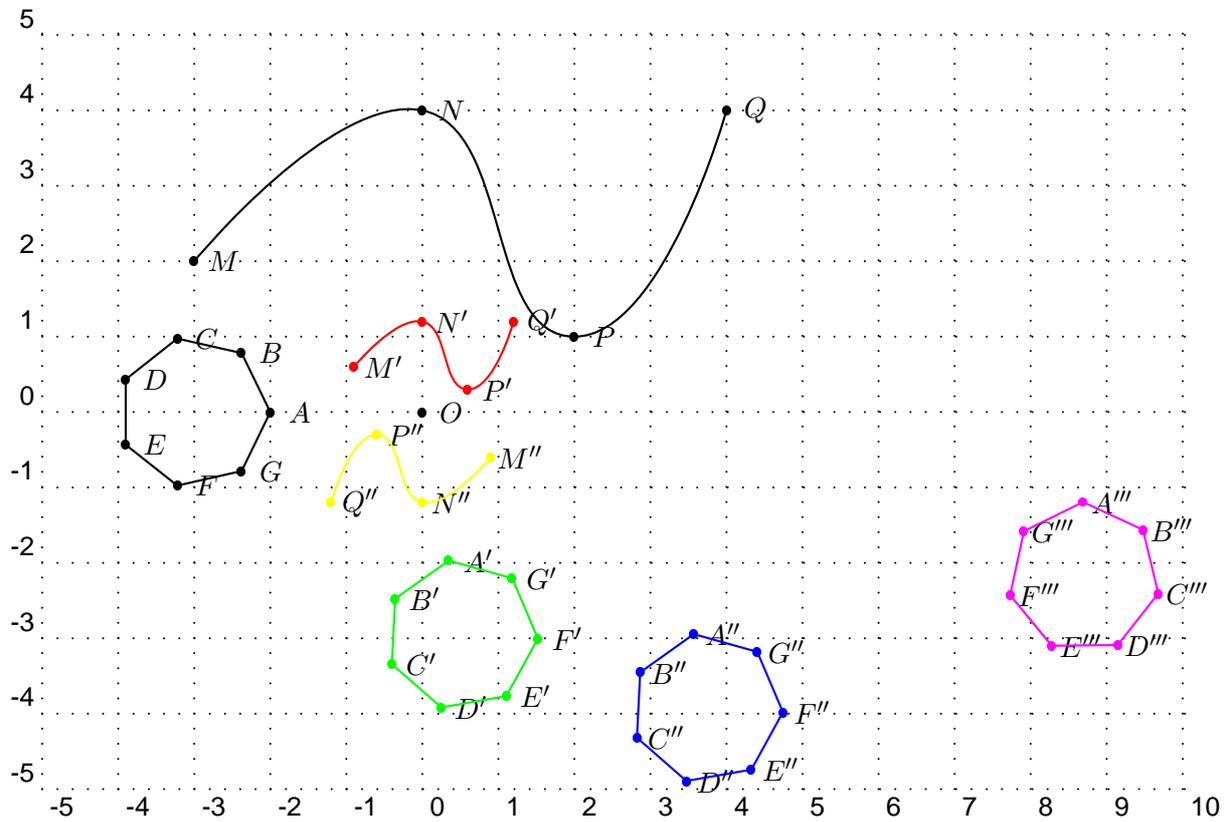
2.1.1 Tracé de la bissectrice



```
\psset{PointSymbol=none}
\pstGeonode[PosAngle={180,130,-90}, PointSymbol={default,none}]
(2,0){B}(0,1){O}(1,4){A}
\pstLineAB[nodesepB=-1, linecolor=red]{O}{A}
\pstLineAB[nodesepB=-1, linecolor=red]{O}{B}
\pstInterLC[PointSymbolA=none, PosAngleB=-45]{O}{B}{O}{A}{G}{C}
\psset{arcsepA=-1, arcsepB=-1}
\pstArcOAB[linecolor=green, linestyle=dashed]{O}{C}{A}
\pstInterCC[PointSymbolB=none, PointNameB=none, PosAngleA=100]{A}{O}{O'}
\pstArcOAB[linecolor=blue, linestyle=dashed]{A}{O'}{O'}
\pstArcOAB[linecolor=blue, linestyle=dashed]{C}{O'}{O'}
\pstLineAB[nodesepB=-1, linecolor=cyan]{O}{O'}
\psset{arcsep=\pslinewidth, linecolor=magenta, Mark=MarkHash}
\pstMarkAngle{C}{O}{O'}{}
\pstMarkAngle[MarkAngleRadius=.5]{O'}{O}{A}{}
```

2.1.2 Transformation de polygones et courbes

Voici comment utiliser le paramètre `CurveType` avec des transformations.

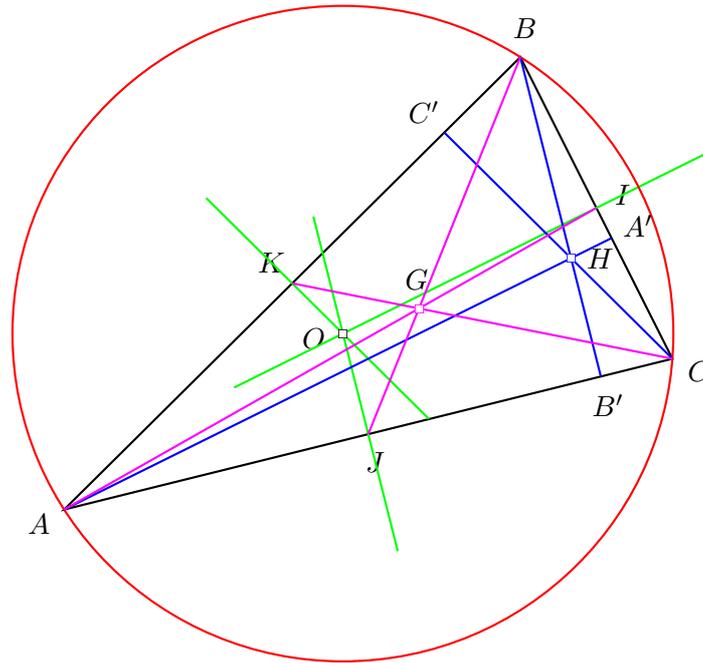


```

\pstGeonode{0}
\rput(-3,0){\pstGeonode[CurveType=polygon](1,0){A}(1;51.43){B}(1;102.86){C}
(1;154.29){D}(1;205.71){E}(1;257.14){F}(1;308.57){G}}
\rput(-4,-1){\pstGeonode[CurveType=curve](1,3){M}(4,5){N}(6,2){P}(8,5){Q}}
\pstRotation[linecolor=green, RotAngle=100, CurveType=polygon]{0}{A, B, C, D, E, F, G}
\pstHomO[linecolor=red, HomCoef=.3, CurveType=curve]{0}{M,N,P,Q}
\pstTranslation[linecolor=blue, CurveType=polygon]{C}{0}{A', B', C', D', E', F', G'}
\pstSymO[linecolor=yellow, CurveType=curve]{0}{M',N',P',Q'}
\pstOrtSym[linecolor=magenta, CurveType=polygon]{Q}{F''}
{A', B', C', D', E', F', G'}{A''', B''', C''', D''', E''', F''', G'''}

```

2.1.3 Droites remarquables du triangle

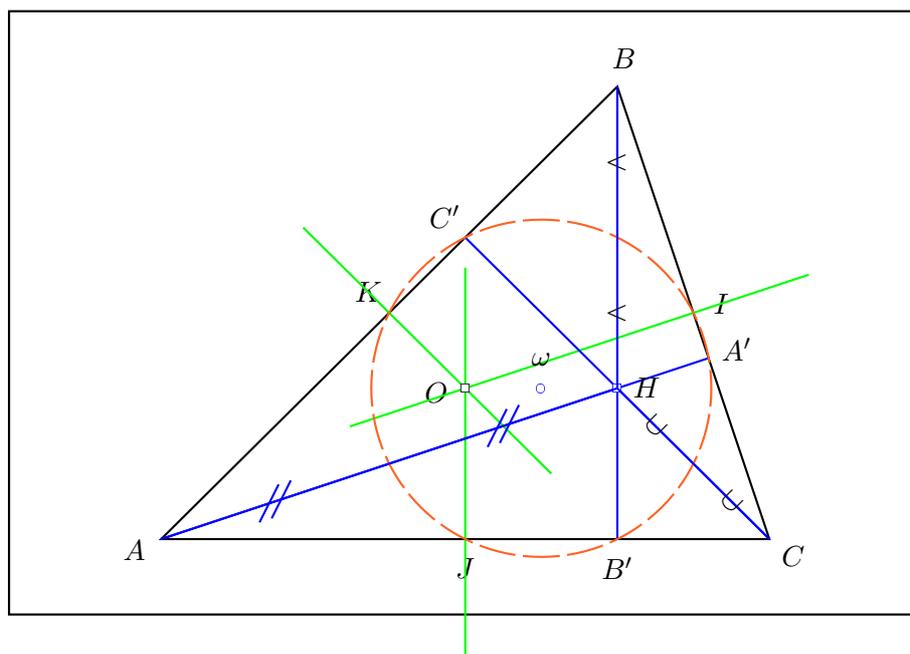


```

\psset{PointSymbol=none}
\pstTriangle[PointSymbol=none](-2,-1){A}(1,2){B}(2,0){C}
% les médiatrices
{% encapsulation de modif paramètres
  \psset{linestyle=none, PointNameB=none}
  \pstMediatorAB{A}{B}{K}{KP}
  \pstMediatorAB[PosAngleA=-40]{C}{A}{J}{JP}
  \pstMediatorAB[PosAngleA=75]{B}{C}{I}{IP}
}% fin
\pstInterLL[PointSymbol=square, PosAngle=-170]{I}{IP}{J}{JP}{O}
{% encapsulation de modif paramètres
  \psset{nodesep=-.8, linecolor=green}
  \pstLineAB{O}{I}\pstLineAB{O}{J}\pstLineAB{O}{K}
}% fin
\pstCircleOA[linecolor=red]{O}{A}
% pour que le symbol de O soit sur et non sous les droites
\psdot[dotstyle=square](O)
% les hauteurs et l'orthocentre
\pstProjection{B}{A}{C}
\pstProjection{B}{C}{A}
\pstProjection{A}{C}{B}
\psset{linecolor=blue}\ncline{A}{A'}\ncline{C}{C'}\ncline{B}{B'}
\pstInterLL[PointSymbol=square]{A}{A'}{B}{B'}{H}
% les médianes et le centre de gravité
\psset{linecolor=magenta}\ncline{A}{I}\ncline{C}{K}\ncline{B}{J}
\pstCGravABC[PointSymbol=square, PosAngle=95]{A}{B}{C}{G}

```

2.1.4 Cercle d'Euler



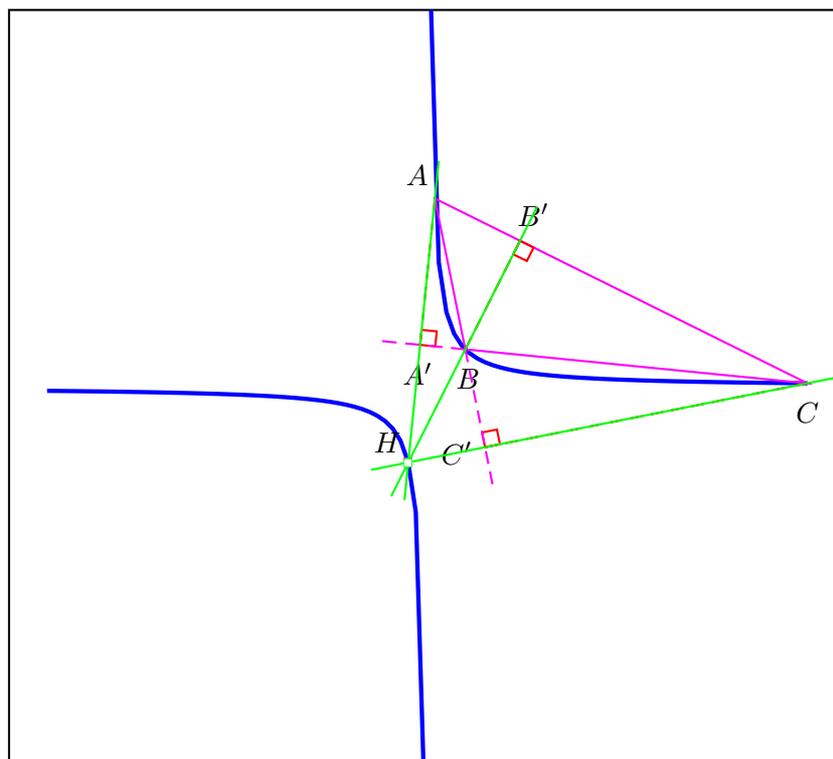
```

\psset{PointSymbol=none}
\pstTriangle(-2,-1){A}(1,2){B}(2,-1){C}
{% encapsulation de modif paramètres
  \psset{linestyle=none, PointSymbolB=none, PointNameB=none}
  \pstMediatorAB{A}{B}{K}{KP}
  \pstMediatorAB{C}{A}{J}{JP}
  \pstMediatorAB{B}{C}{I}{IP}
}% fin
\pstInterLL[PointSymbol=square, PosAngle=-170]{I}{IP}{J}{JP}{O}
{% encapsulation de modif paramètres
  \psset{nodesep=-.8, linecolor=green}
  \pstLineAB{O}{I}\pstLineAB{O}{J}\pstLineAB{O}{K}
}% fin
\psdot[dotstyle=square](O)
\pstProjection{B}{A}{C}
\pstProjection{B}{C}{A}
\pstProjection{A}{C}{B}
\psset{linecolor=blue}\ncline{A}{A'}\ncline{C}{C'}\ncline{B}{B'}
\pstInterLL[PointSymbol=square]{A}{A'}{B}{B'}{H}
% le cercle d'Euler (centre au milieu de [OH])
\pstMiddleAB[PointSymbol=o, PointName=\omega]{O}{H}{omega}
\pstCircleOA[linecolor=Orange, linestyle=dashed, dash=5mm 1mm]{omega}{B'}
\psset{PointName=none}
% il passe par le milieu des segments joignant l'orthocentre et les sommets
\pstMiddleAB{H}{A}{AH}\pstMiddleAB{H}{B}{BH}\pstMiddleAB{H}{C}{CH}
\pstSegmentMark{H}{AH}\pstSegmentMark{AH}{A}
\psset{SegmentSymbol=wedge}\pstSegmentMark{H}{BH}\pstSegmentMark{BH}{B}
\psset{SegmentSymbol=cup}\pstSegmentMark{H}{CH}\pstSegmentMark{CH}{C}

```

2.1.5 Orthocentre et hyperbole

L'orthocentre d'un triangle dont les points sont situés sur les branches de l'hyperbole $\mathcal{H} : y = a/x$ appartient à cette hyperbole.



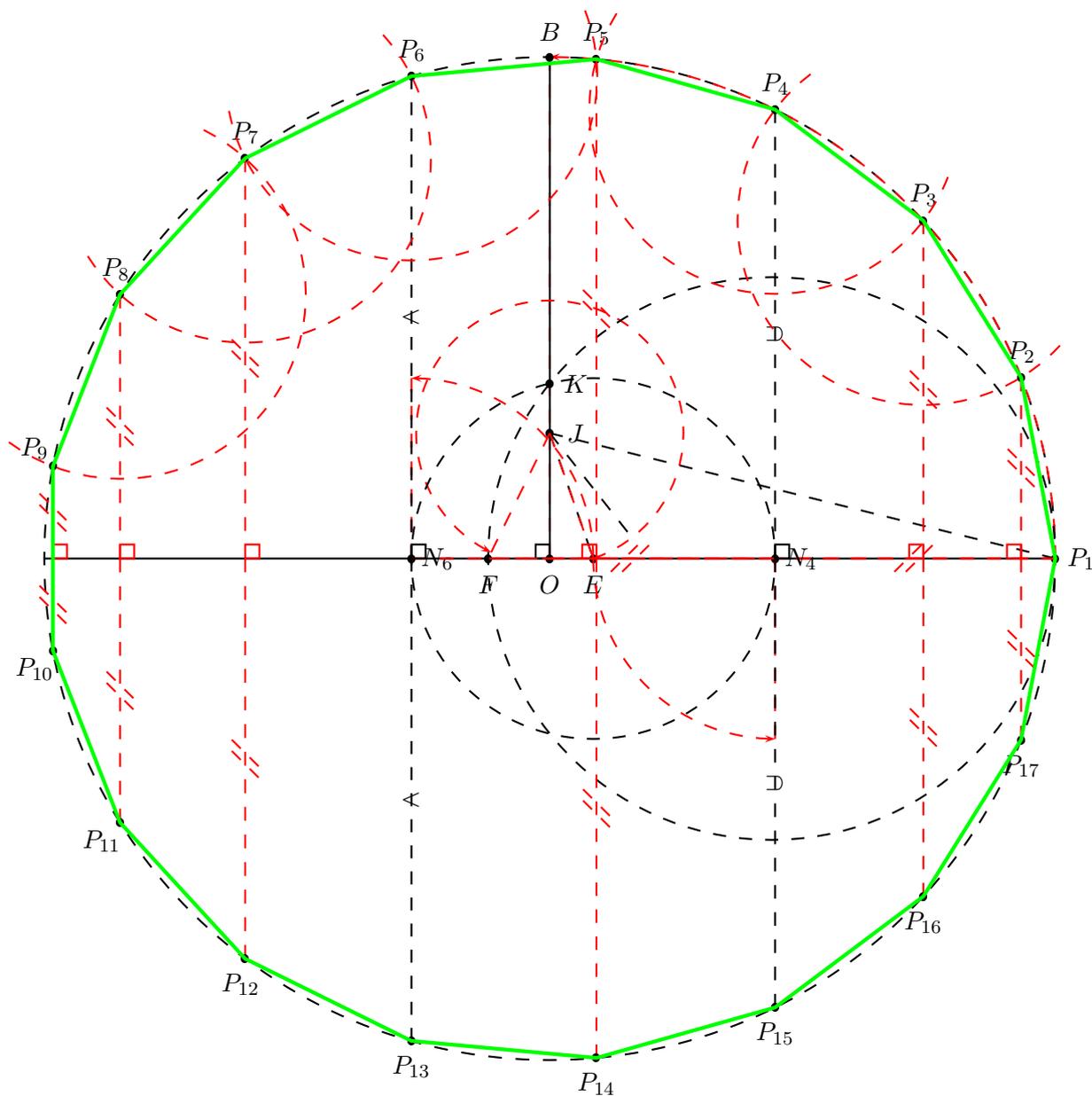
```

\psset{linecolor=blue, linewidth=2\pslinewidth}
\psplot{-10}{-.1}{1 x div}
\psplot{.1}{10}{1 x div}
\psset{PointSymbol=none, linewidth=.5\pslinewidth}
\pstTriangle[linecolor=magenta, PosAngleB=-85, PosAngleC=-90](.2,5){A}(1,1){B}(10,.1){C}
%\pstTriangle[linecolor=magenta, PosAngleB=-135](.2, 5){A}(-1,-1){B}(10,.1){C}
\psset{linecolor=magenta, CodeFig=true, CodeFigColor=red}
\pstProjection{B}{A}{C}
\ncline[nodesepA=-1, linestyle=dashed, linecolor=magenta]{C'}{B}
\pstProjection{B}{C}{A}
\ncline[nodesepA=-1, linestyle=dashed, linecolor=magenta]{A'}{B}
\pstProjection{A}{C}{B}
\pstInterLL[PosAngle=135, PointSymbol=square]{A}{A'}{B}{B'}{H}
\psset{linecolor=green, nodesep=-1}
\pstLineAB{A}{H}\pstLineAB{B'}{H}\pstLineAB{C}{H}
\psdot[dotstyle=square](H)

```

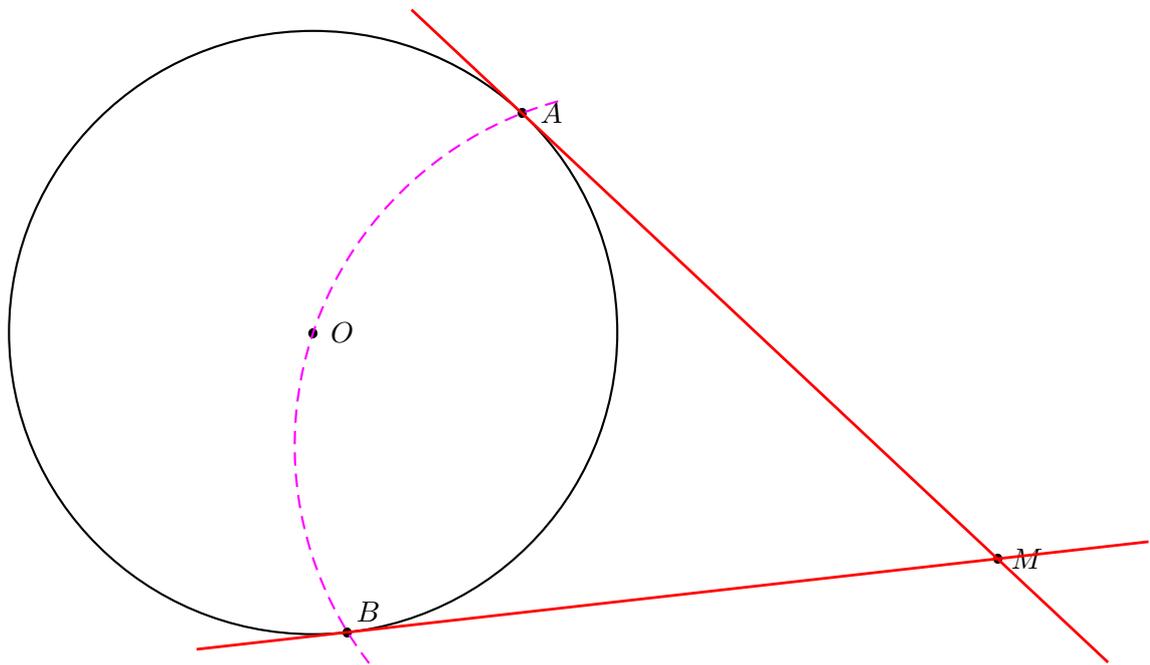
2.1.6 Polygone régulier à 17 côtés

Remarquable construction due à K. F. GAUSS qui a démontré par ailleurs qu'il était possible de construire les polygones à $2^{2^p} + 1$ côtés avec en prime la méthode général de construction. Le polygone régulier suivant de cette famille possède donc 257 côtés !

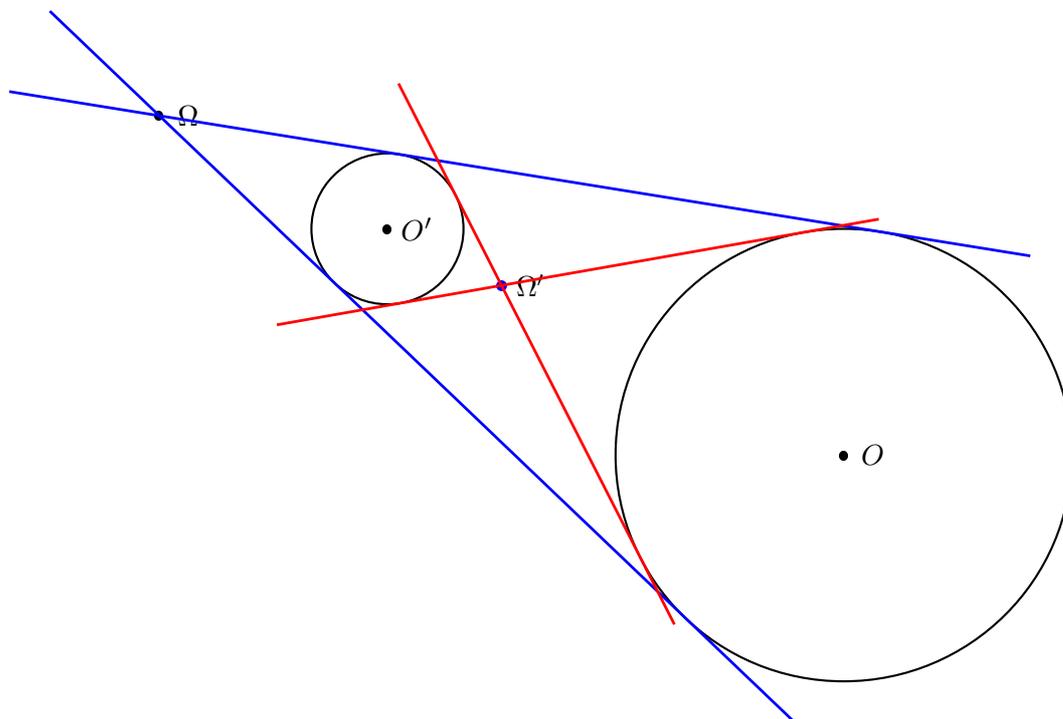


2.1.7 Tangentes de cercles

Construction des tangentes à un cercle passant par un point donné.

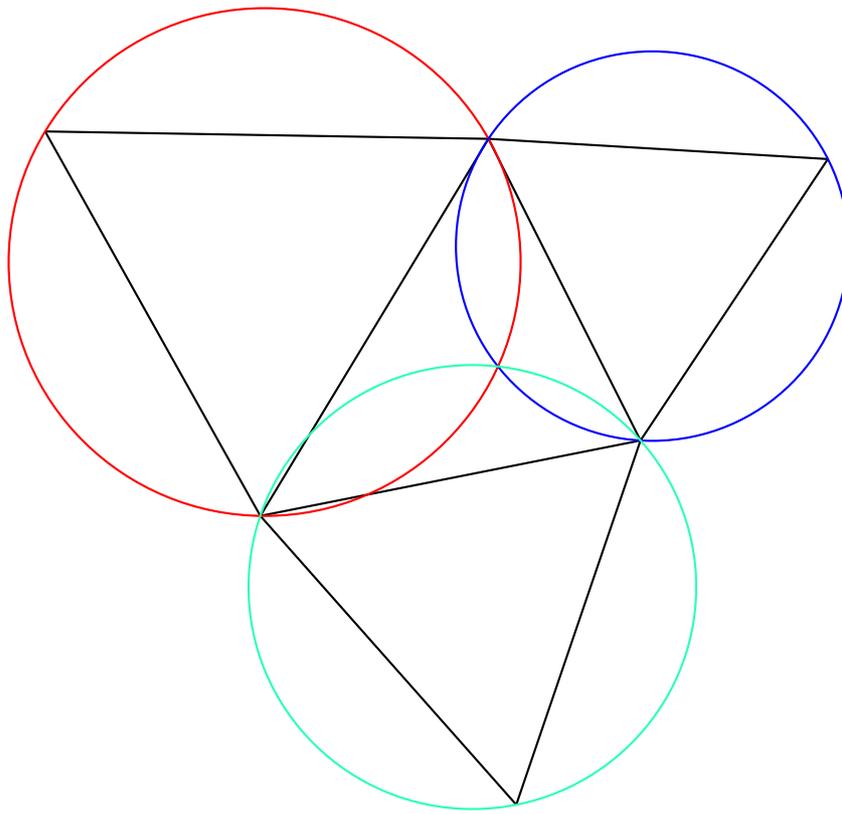


Construction des tangentes à deux cercles.



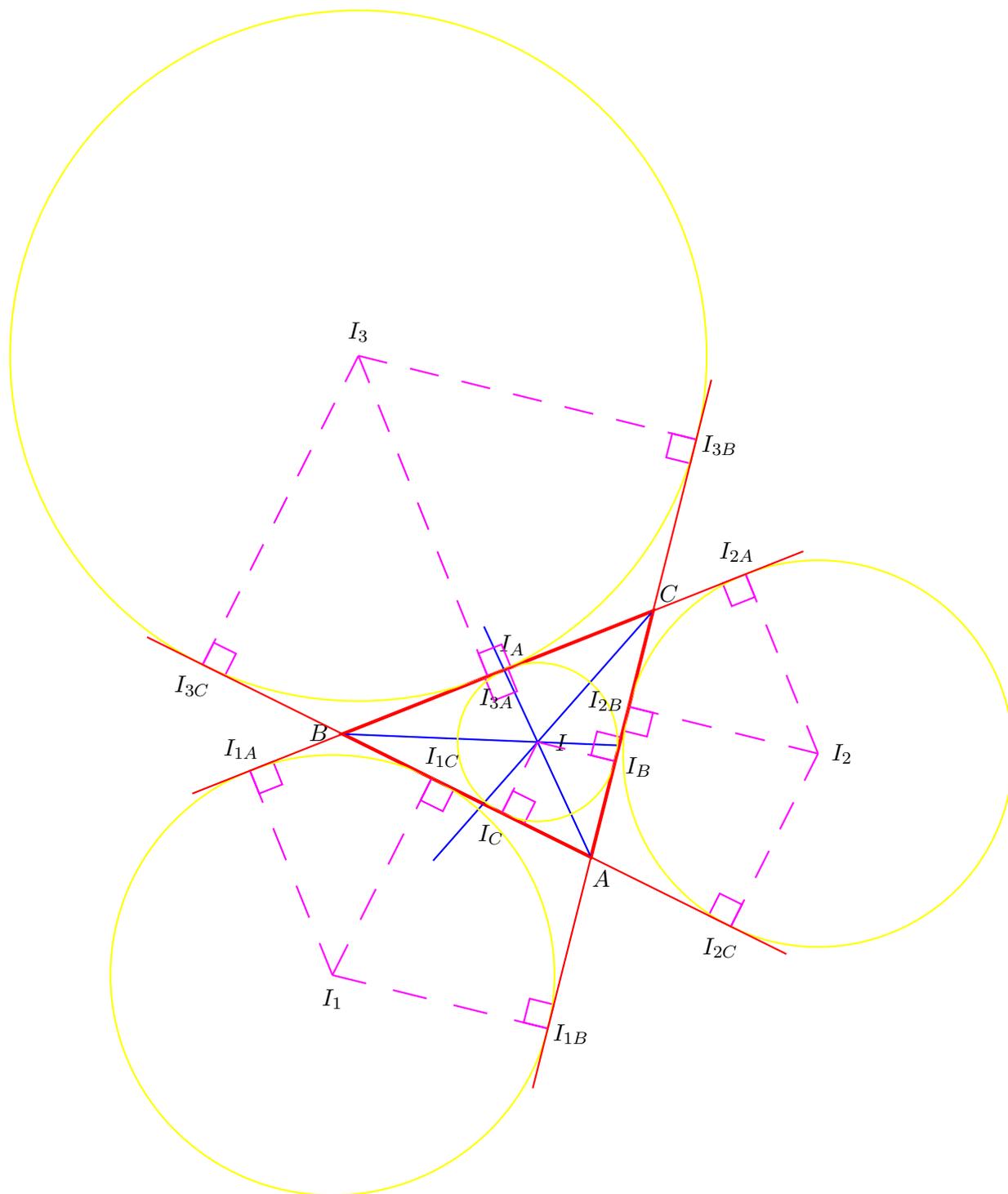
2.1.8 Point de Fermat d'un triangle

Figure de Manuel LUQUE.



2.1.9 Cercles ex-insrits et inscrit d'un triangle

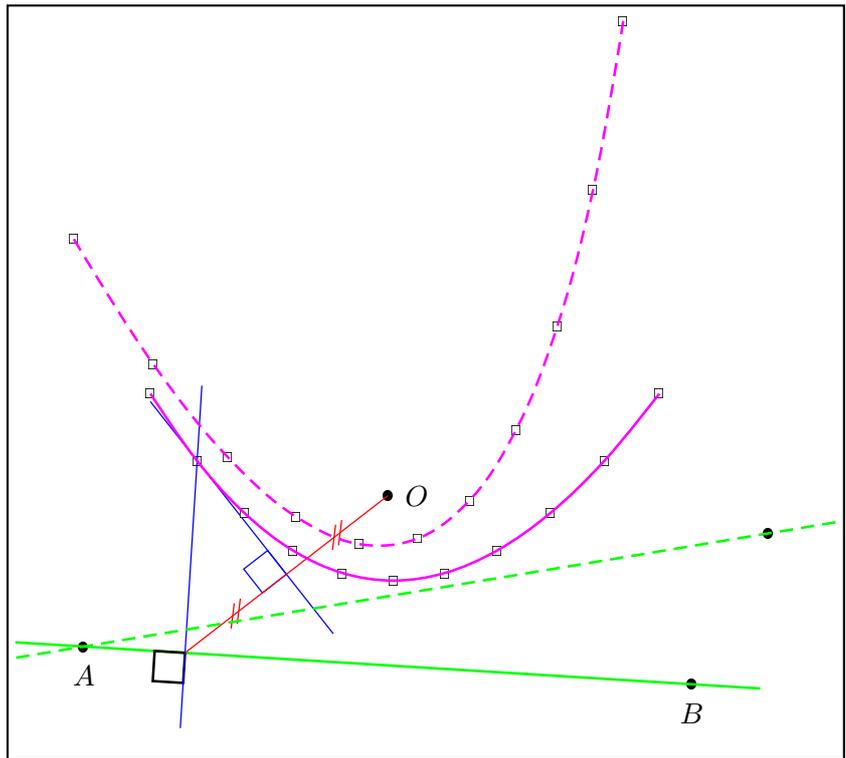
Les centres de ces cercles sont les intersections des bissectrices intérieures et extérieures.



2.2 Quelques lieux de points

2.2.1 Parabole

La parabole est l'ensemble des points situés à égale distance d'un point : le foyer, et d'une droite : la directrice.



```

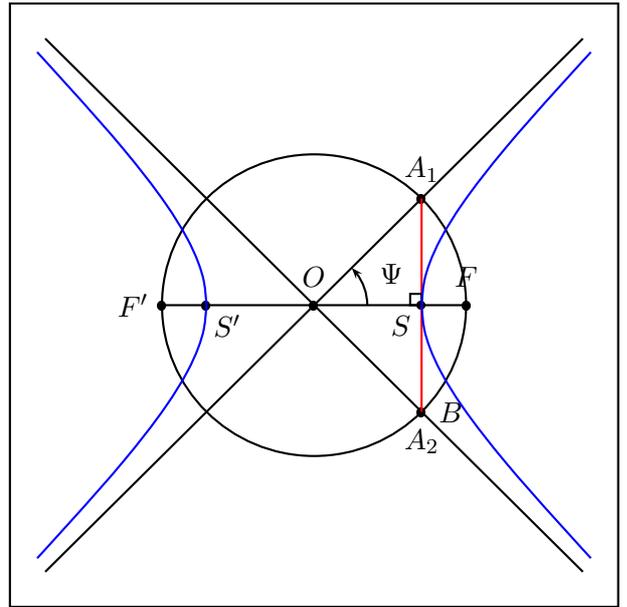
\psset{linewidth=1.2\pslinewidth}\renewcommand{\NbPt}{11}
\pstGeonode[PosAngle={0,-90}](5,4){O}(1,2){A}(9,1.5){B}
\newcommand{\Parabole}[1][100]{%
  \pstLineAB[nodesep=-.9, linecolor=green]{A}{B}
  \psset{RotAngle=90, PointSymbol=none, PointName=none}
  \multido{\n=1+1}{\NbPt}{%
    \pstHomO[HomCoef=\n\space \NbPt\space 1 add div]{A}{B}[M\n]
    \pstMediatorAB[linestyle=none]{M\n}{O}{M\n_I}{M\n_IP}
    \pstRotation{M\n}{A}[M\n_P]
    \pstInterLL[PointSymbol=square, PointName=none]{M\n_I}{M\n_IP}{M\n}{M\n_P}{P_\n}
    \ifnum\n=#1
      \bgroup
      \pstRightAngle{A}{M\n}{M\n_P}
      \psset{linewidth=.5\pslinewidth, nodesep=-1, linecolor=blue}
      \pstLineAB{M\n_I}{P_\n}\pstLineAB{M\n}{P_\n}
      \pstRightAngle{P_\n}{M\n_I}{M\n}
      \psset{linecolor=red}\pstSegmentMark{M\n}{M\n_I}\pstSegmentMark{M\n_I}{O}
      \egroup
    \fi}}%fin multido-newcommand
\Parabole[2]\pstGenericCurve[linecolor=magenta]{P_}{1}{\NbPt}
%% Nouvelle parabole avec un nouveau point B
\pstGeonode[PointSymbol=*, PosAngle=-90](10,3.5){B}
\psset{linestyle=dashed}
\Parabole\pstGenericCurve[linecolor=magenta]{P_}{1}{\NbPt}

```

2.2.2 Hyperbole

L'hyperbole est composée des points dont la différence des distances avec les foyers est constant. Pour se rafraîchir la mémoire, on peut aller voir le « Cours de mathématiques spéciales », vol. 2, RAMIS, DESCHAMPS, ODOUX, pp. 226–227.

```
%% QQ RAPPELS : a=\Sommet, c=\PosFoyer,
%% b^2=c^2-a^2, e=c/a
%% pour une hyperbole -> e>1, donc c>a,
%% ici on choisi a=\sqrt{2}, c=2, e=\sqrt{2}
%% M est sur H <=> |MF-MF'|=2a
```



```
\newcommand{\Sommet}{1.4142135623730951}
\newcommand{\PosFoyer}{2}
\newcommand{\HypAngle}{0}
\setcounter{i}{1}
\newcounter{CoefDiv}\setcounter{CoefDiv}{20}
\newcounter{Inc}\setcounter{Inc}{2}
\newcounter{n}\setcounter{n}{2}
%% rayon des cercles successifs utilisés pour trouver les points de H
%% on choisit \Rii=\Ri+2\Sommet (définition de l'hyperbole)
\newcommand{\Ri}{% c'est du postscript
  \PosFoyer\space\Sommet\space sub \arabic{i}\space\arabic{CoefDiv}\space div add}
\newcommand{\Rii}{\Ri\space\Sommet\space 2 mul add}
\pstGeonode[PosAngle=90]{0}{\PosFoyer;\HypAngle}{F}
\pstSymO[PosAngle=180]{0}{F}\pstLineAB{F}{F'}
%% TRACÉ DES ASYMPTOTES
\pstCircleOA{0}{F}
%% positionnement des deux sommets de H
\pstGeonode[PosAngle=-135]{\Sommet;\HypAngle}{S}
\pstGeonode[PosAngle=-45]{-\Sommet;\HypAngle}{S'}
%% l'intersection de la droite perpendiculaire à (FF') passant par S,
%% coupe les asymptotes sur le cercle de diamètre [FF'] (cette droite est une tangente)
\pstRotation[RotAngle=90, PointSymbol=none]{S}{0}{B}
\pstInterLC[PosAngleA=90, PosAngleB=-90]{S}{B}{0}{F}{A_1}{A_2}
\pstLineAB[nodesepA=-3,nodesepB=-5]{A_1}{0}
\pstLineAB[nodesepA=-3,nodesepB=-5]{A_2}{0}
%% cos(\Psi)=OS/OF (c-a-d \Sommet/\PosFoyer)
%% ici \sqrt{2}/2, donc \Psi=45 => hyperbole équilatère
\pstMarkAngle[LabelSep=.8, MarkAngleRadius=.7, arrows=->,
  LabelSep=1.1]{F}{0}{A_1}{\Psi}
\ncline[linecolor=red]{A_1}{A_2}
\pstRightAngle[RightAngleSize=.15]{A_1}{S}{0}
\psset{PointName=none}
\whiledo{\value{n}<8}{%
  \psset{RadiusA=\pstDistVal{\Ri},RadiusB=\pstDistVal{\Rii},PointSymbol=none}
```

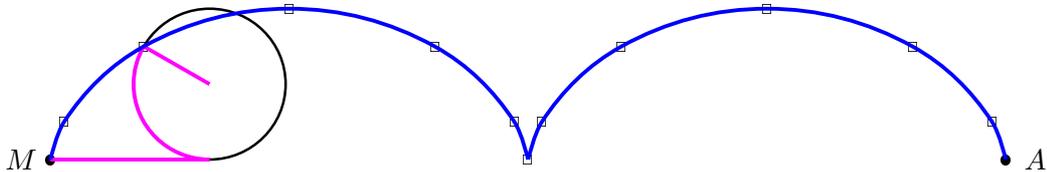
```

\pstInterCC{F}{F'}{M\arabic{n}}{P\arabic{n}}
\pstInterCC{F'}{F}{M'\arabic{n}}{P'\arabic{n}}
%% bcp de points au début, moins ensuite
%% n -> numéro du point, i -> taille des cercles
%% Inc -> incrément variable de i (2^n)
\stepcounter{n}\addtocounter{i}{\value{Inc}}
\addtocounter{Inc}{\value{Inc}}%% fin de whiledo
\psset{linecolor=blue}
%% tracé des quatres 1/2 branches de l'hyperbole
\pstGenericCurve[GenCurvFirst=S]{M}{2}{7}
\pstGenericCurve[GenCurvFirst=S]{P}{2}{7}
\pstGenericCurve[GenCurvFirst=S']{M'}{2}{7}
\pstGenericCurve[GenCurvFirst=S']{P'}{2}{7}
%% pour vérif le trace paramétrique
%\parametricplot[linecolor=black, linewidth=.25\pslinewidth]{-1}{1}
% {t dup tx@EclDDict begin sh exch ch end \Sommet\space mul exch
%   \PosFoyer\space dup mul \Sommet\space dup mul sub sqrt mul}
%\parametricplot[linecolor=black, linewidth=.25\pslinewidth]{-1}{1}
% {t dup tx@EclDDict begin sh exch ch end neg \Sommet\space mul exch
%   \PosFoyer\space dup mul \Sommet\space dup mul sub sqrt mul}

```

2.2.3 Cycloïde

La roue roule de M à A . Les points du cercle parcourent des cycloïdes. L'intérêt de cet exemple est de faire coïncider les points de rebroussement avec des points calculés.



```

\renewcommand{\NbPt}{11}\psset{linewidth=1.2\pslinewidth}
\pstGeonode[PointSymbol={*,none}, PointName={default,none}, PosAngle=180]{M}{0,1}{0}
%% 4*pi=12.5663706144
\pstGeonode(12.5663706144,0){A}
\pstTranslation[PointSymbol=none, PointName=none]{M}{A}{0}[B]
\multido{\n=1+1}{\NbPt}{%
  \pstHomO[HomCoef=\n\space \NbPt\space 1 add div,
    PointSymbol=none, PointName=none]{0}{B}[0\n]
  \pstProjection[PointSymbol=none, PointName=none]{M}{A}{0\n}[P\n]
  %\pstCircleOA[linestyle=dashed, linecolor=red]{0\n}{P\n}
  \pstCurvAbsNode[PointSymbol=square, PointName=none, CurvAbsNeg=true]
    {0\n}{P\n}{M\n}{\pstDistAB{0}{0\n}}
  \ifnum\n=2%affichage du second cercle
    \bgroup
    \pstCircleOA{0\n}{M\n}
    \psset{linecolor=magenta, linewidth=1.5\pslinewidth}
    \pstArcnOAB{0\n}{P\n}{M\n}
    \ncline{0\n}{M\n}\ncline{P\n}{M}
    \egroup

```

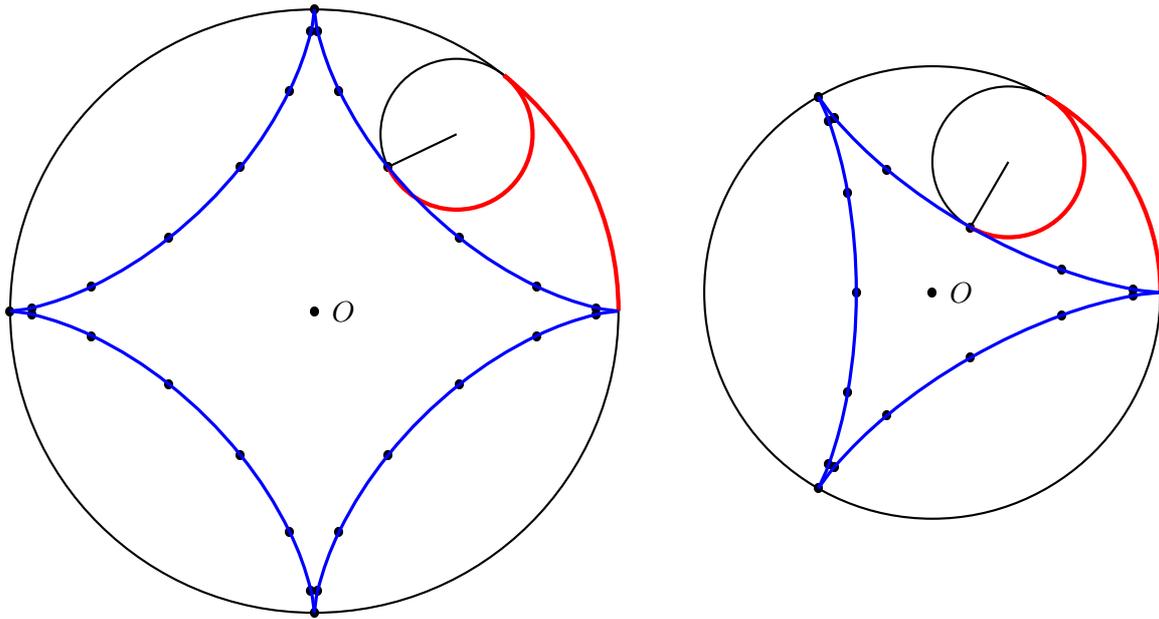
```

\fi}% fin du multido
\psset{linecolor=blue, linewidth=1.5\pslinewidth}
\pstGenericCurve[GenCurvFirst=M]{M}{1}{6}
\pstGenericCurve[GenCurvLast=A]{M}{6}{\NbPt}
%% juste pour la vérification
%\parametricplot[linecolor=green, linewidth=.5\pslinewidth]{0}{12.5663706144}
% {t t 3.1415926 div 180 mul sin sub 1 t 3.1415926 div 180 mul cos sub}

```

2.2.4 Hypocycloïdes (astroïde et deltoïde)

Une roue roule à l'intérieure d'un cercle et selon le rapport des rayons, on obtient une astroïde, une deltoïde et dans le cas générale des hypocycloïdes.



```

\newcommand{\HypoCyclo}[4][100]{%
  \def\R{#2}\def\petitR{#3}\def\NbPt{#4}% définitions locales pour lisibilité
  %% Selon la valeur de R sur petitR :
  %% 4 une astroïde -- 3 une deltoïde -- 2 un diamètre (!)
  \def\Anglen{\n\space 360 \NbPt\space 1 add div mul}
  \psset{PointSymbol=none,PointName=none}
  \pstGeonode[PointSymbol=*,none],PointName={default,none}, PosAngle=0]{O}{(\R;0){P}
  \pstCircleOA{O}{P}
  \pstHomO[HomCoef=\petitR\space\R\space div]{P}{O}{M}
  \multido{\n=1+1}{\NbPt}{%
    \pstRotation[RotAngle=\Anglen]{O}{M}{M\n}
    \rput(M\n){\pstGeonode(\petitR;0){Q}}
    \pstRotation[RotAngle=\Anglen]{M\n}{Q}{N}
    \pstRotation[RotAngle=\n\space -360 \NbPt\space 1 add div
    mul \R\space\petitR\space div mul, PointSymbol=*, PointName=none]%
    {M\n}{N}{N\n}
  \ifnum\n=#1
    \pstCircleOA{M\n}{N\n}\ncline{M\n}{N\n}%
    {\psset{linecolor=red, linewidth=2\pslinewidth}
    \pstArcOAB{M\n}{N\n}{N}\pstArcOAB{O}{P}{N}}
  \fi}}%fin multido-newcommand

```

```

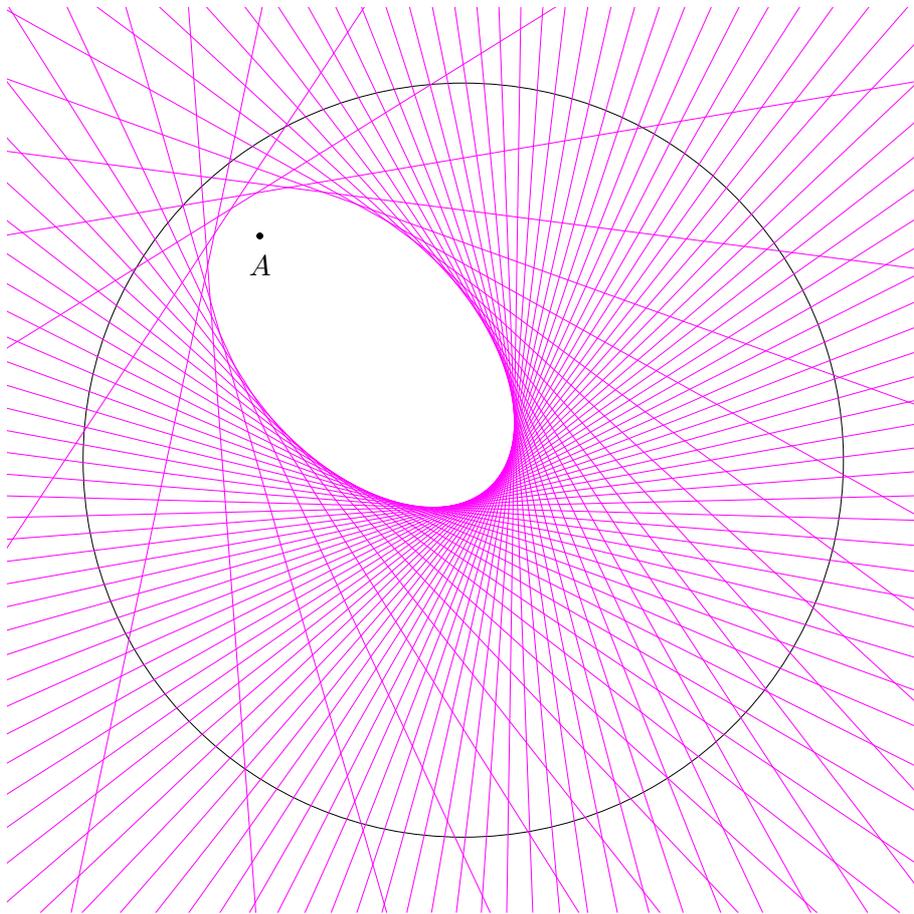
\HypoCyclo[4]{4}{1}{27}
\psset{linecolor=blue,linewidth=1.5\pslinewidth}
%% il est préférable de mettre les quatre branches séparément à cause
%% des points de rebroussement
\pstGenericCurve[GenCurvFirst=P]{N}{1}{7}
\pstGenericCurve{N}{7}{14}\pstGenericCurve{N}{14}{21}
\pstGenericCurve[GenCurvLast=P]{N}{21}{27}

```

2.3 Enveloppes de droites et de cercles

2.3.1 Coniques

On considère un cercle et un point A extérieur à ce cercle. L'ensemble des médiatrices des segments formés par A et les points du cercle forme les deux coniques selon la place de A : s'il est à l'intérieur du cercle l'ellipse, à l'extérieur l'hyperbole. A et le centre du cercle en sont les foyers (figure d'O. REBOUX).



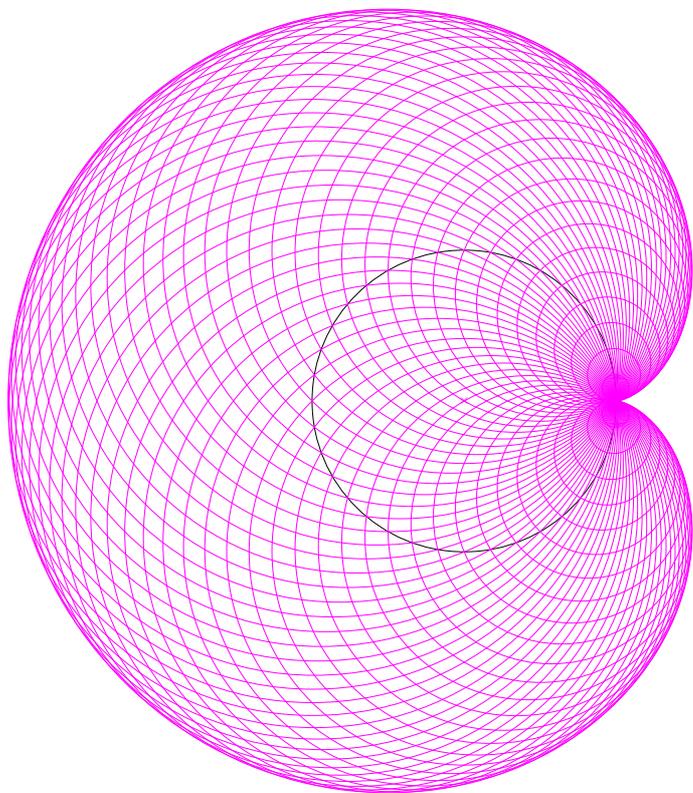
```

\psset{linewidth=0.4\pslinewidth,PointSymbol=none, PointName=none}
\pstGeonode[PosAngle=-90, PointSymbol={none,*,none}, PointName={none,default,none}]
  {0}(4;132){A}(5,0){O'}
\pstCircleOA{0}{O'}
\multido{\n=5+5}{72}{%
  \pstGeonode(5;\n){M_\n}
  \pstMediatorAB[nodesep=-15,linecolor=magenta]
    {A}{M_\n}{I}{J}}% fin multido

```

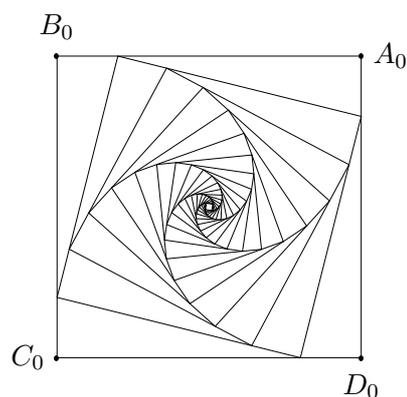
2.3.2 Cardioïde

La cardioïde est l'enveloppe des cercles centrés sur un cercle et passant par un point fixé de ce cercle.



```
\psset{linewidth=0.4\pslinewidth,PointSymbol=x,
      nodesep=0,linecolor=magenta}
\pstGeonode[PointName=none]{0}(2,0){0'}
\pstCircleOA[linecolor=black]{0}{0'}
\multido{\n=5+5}{72}{%
  \pstGeonode[PointSymbol=none, PointName=none]
    (2;\n){M_\n}
  \pstCircleOA{M_\n}{0'}}
```

2.4 Homothéties et fractales



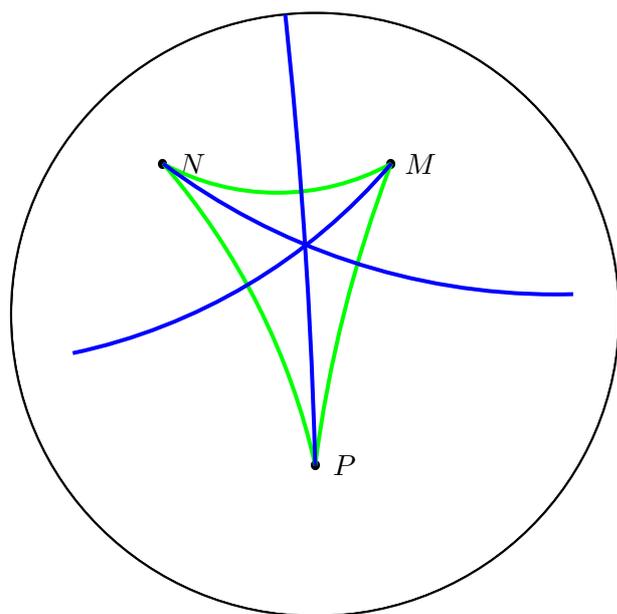
```

\pstGeonode[PosAngle={0,90}](2,2){A_0}(-2,2){B_0}%
\psset{RotAngle=90}
\pstRotation[PosAngle=270]{A_0}{B_0}[D_0]
\pstRotation[PosAngle=180]{D_0}{A_0}[C_0]
\pspolygon(A_0)(B_0)(C_0)(D_0)%
\psset{PointSymbol=none, PointName=none, HomCoef=.2}
\multido{\n=1+1,\i=0+1}{20}{%
  \pstHom0[PosAngle=0]{B_\i}{A_\i}[A_\n]
  \pstHom0[PosAngle=90]{C_\i}{B_\i}[B_\n]
  \pstHom0[PosAngle=180]{D_\i}{C_\i}[C_\n]
  \pstHom0[PosAngle=270]{A_\i}{D_\i}[D_\n]
  \pspolygon(A_\n)(B_\n)(C_\n)(D_\n)}% fin multido

```

2.5 Géométrie hyperbolique : triangle et ses hauteurs

Le comble pour de la géométrie euclidienne !



Annexe A

Glossaire des commandes

Voici la liste des commandes définies par `pst-eucl`. Chacune est proposée avec une courte description ainsi que la liste des paramètres `pst-eucl` qui permettent de la contrôler. Il est tout à fait possible que d'autres paramètres `PSTricks` puissent être utilisés, en particulier ceux contrôlant le trait (épaisseur, couleur, style).

| | |
|--|----|
| <code>\pstAngleAOB</code> { $\langle A \rangle$ }{ $\langle O \rangle$ }{ $\langle B \rangle$ }..... | 9 |
| Permet d'utiliser la mesure de \widehat{AOB} (orienté) pour le paramètre <code>RotAngle</code> . Paramètres : <code>AngleCoef</code> | |
| <code>\pstArcnOAB</code> [$\langle par \rangle$]{ $\langle O \rangle$ }{ $\langle A \rangle$ }{ $\langle B \rangle$ }..... | 7 |
| Trace l'arc de cercle de centre O , de rayon OA , délimité par l'angle \widehat{AOB} dans le sens trigonométrique inverse. | |
| <code>\pstArcOAB</code> [$\langle par \rangle$]{ $\langle O \rangle$ }{ $\langle A \rangle$ }{ $\langle B \rangle$ }..... | 7 |
| Trace l'arc de cercle de centre O , de rayon OA , délimité par l'angle \widehat{AOB} dans le sens trigonométrique. | |
| <code>\pstBissectBAC</code> [$\langle par \rangle$]{ $\langle B \rangle$ }{ $\langle A \rangle$ }{ $\langle C \rangle$ }{ $\langle M \rangle$ }..... | 13 |
| Construit la bissectrice intérieure de l'angle \widehat{BAC} ainsi qu'un point M de la bissectrice image de B par rotation autour de A . Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> | |
| <code>\pstCGravABC</code> [$\langle par \rangle$]{ $\langle A \rangle$ }{ $\langle B \rangle$ }{ $\langle C \rangle$ }{ $\langle G \rangle$ }..... | 11 |
| Construit le centre de gravité G du triangle ABC . Paramètres : <code>PointName</code> , <code>PointNameSep</code> , <code>PosAngle</code> , <code>PointSymbol</code> , <code>PtNameMath</code> | |
| <code>\pstCircleAB</code> [$\langle par \rangle$]{ $\langle A \rangle$ }{ $\langle B \rangle$ } | 6 |
| Construit le cercle de diamètre AB . Paramètres : <code>Radius</code> , <code>Diameter</code> . | |
| <code>\pstCircleABC</code> [$\langle par \rangle$]{ $\langle A \rangle$ }{ $\langle B \rangle$ }{ $\langle C \rangle$ }{ $\langle O \rangle$ }..... | 12 |
| Construit le centre O du cercle circonscrit du triangle ABC . Paramètres : <code>PointName</code> , <code>PointNameSep</code> , <code>PosAngle</code> , <code>PointSymbol</code> , <code>PtNameMath</code> , <code>DrawCirABC</code> , <code>CodeFig</code> , <code>CodeFigColor</code> , <code>CodeFigStyle</code> , <code>SegmentSymbolA</code> , <code>SegmentSymbolB</code> , <code>SegmentSymbolC</code> | |

| | |
|--|----|
| <code>\pstCircleOA[⟨par⟩]{⟨O⟩}{⟨A⟩}</code> | 12 |
| Construit le cercle de centre O passant par A . | |
| Paramètres : <code>Radius</code> , <code>Diameter</code> . | |
| <code>\pstCurvAbsNode[⟨par⟩]{⟨O⟩}{⟨A⟩}{⟨B⟩}{⟨Abs⟩}</code> | 8 |
| Place un point sur un cercle au moyen de son abscisse curviligne. | |
| Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> , <code>CurvAbsNeg</code> | |
| <code>\pstDistAB{⟨A⟩}{⟨B⟩}</code> | 8 |
| Permet de spécifier la distance AB pour les paramètres <code>Radius</code> et <code>Diameter</code> . | |
| Paramètres : <code>DistCoef</code> . | |
| <code>\pstDistVal{⟨x⟩}</code> | 6 |
| Permet d'utiliser une valeur numérique pour les paramètres <code>Radius</code> et <code>Diameter</code> . | |
| Paramètres : <code>DistCoef</code> . | |
| <code>\pstGenericCurve[⟨par⟩]{⟨Radical⟩}{⟨n₁⟩}{⟨n₂⟩}</code> | 8 |
| Trace une courbe interpolée au moyen d'une famille de points ayant une convention de dénomination utilisant un radical et un numéro. | |
| Paramètres : <code>GenCurvFirst</code> , <code>GenCurvInc</code> , <code>GenCurvLast</code> | |
| <code>\pstGeonode[⟨par⟩](x₁, y₁){⟨A₁⟩}(x₂, y₂){⟨A₂⟩}... (x_n, y_n){⟨A_n⟩}</code> | 2 |
| Création d'une liste de points dans le repère d'origine. | |
| Paramètres : <code>PointName</code> , <code>PointNameSep</code> , <code>PosAngle</code> , <code>PointSymbol</code> , <code>PtNameMath</code> | |
| <code>\pstHomO[⟨par⟩]{⟨O⟩}{⟨M₁, M₂, ..., M_n⟩}{⟨M'₁, M'₂, ..., M'_p⟩}</code> | 10 |
| Construit l'image M'_i de chacun des M_i par l'homothétie de centre O et de rapport <code>HomCoef</code> . | |
| Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> , <code>HomCoef</code> | |
| <code>\pstInterCC[⟨par⟩]{⟨O₁⟩}{⟨B⟩}{⟨O₂⟩}{⟨C⟩}{⟨M₁⟩}{⟨M₂⟩}</code> | 15 |
| Place le(s) point(s) d'intersection entre le cercle de centre O_1 passant par B et le cercle de centre O_2 passant par C . | |
| <code>\pstInterFC[⟨par⟩]{⟨f⟩}{⟨O⟩}{⟨A⟩}{⟨x₀⟩}{⟨M⟩}</code> | 16 |
| Place <i>un</i> point d'intersection entre le cercle de centre O passant par A et la courbe représentative de la fonction f . À priori l'abscisse du point sera proche de x_0 (valeur initiale de l'algorithme de NEWTON). | |
| <code>\pstInterFF[⟨par⟩]{⟨f⟩}{⟨g⟩}{⟨x₀⟩}{⟨M⟩}</code> | 16 |
| Place <i>un</i> point d'intersection entre les deux courbes représentatives des fonctions f et g . À priori l'abscisse du point sera proche de x_0 (valeur initiale de l'algorithme de NEWTON). | |
| <code>\pstInterFL[⟨par⟩]{⟨f⟩}{⟨A⟩}{⟨B⟩}{⟨x₀⟩}{⟨M⟩}</code> | 16 |
| Place <i>un</i> point d'intersection entre la droite (AB) et la courbe représentative de la fonction f . À priori l'abscisse du point sera proche de x_0 (valeur initiale de l'algorithme de NEWTON). | |

| | | |
|--------------------------------|--|----|
| <code>\pstInterLC</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \} \{ \langle O \rangle \} \{ \langle C \rangle \} \{ \langle M_1 \rangle \} \{ \langle M_2 \rangle \}$ | 14 |
| | Place le(s) point(s) d'intersection entre la droite (AB) et le cercle de centre O passant par C . | |
| | Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> , <code>PointSymbolA</code> , <code>PosAngleA</code> , <code>PointNameA</code> , <code>PointSymbolB</code> , <code>PosAngleB</code> , <code>PointNameB</code> , <code>Radius</code> , <code>Diameter</code> | |
| <code>\pstInterLL</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \} \{ \langle C \rangle \} \{ \langle D \rangle \} \{ \langle M \rangle \}$ | 13 |
| | Place le point d'intersection entre les droites (AB) et (CD) . | |
| | Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> | |
| <code>\pstLineAB</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \}$ | 6 |
| | Trace la droite (AB) . | |
| <code>\pstMarkAngle</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \} \{ \langle C \rangle \} \{ \langle label \rangle \}$ | 5 |
| | Marque l'angle \widehat{ABC} spécifié dans le sens trigonométrique | |
| | Paramètres : <code>MarkAngleRadius</code> , <code>LabelAngleOffset</code> , <code>Mark</code> | |
| <code>\pstMediatorAB</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \} \{ \langle I \rangle \} \{ \langle M \rangle \}$ | 12 |
| | Construit la médiatrice du segment $[AB]$, son milieu I et un point M de la médiatrice qui est l'image par quart de tour de B . | |
| | Paramètres : <code>PointName</code> , <code>PointNameSep</code> , <code>PosAngle</code> , <code>PointSymbol</code> , <code>PtNameMath</code> , <code>CodeFig</code> , <code>CodeFigColor</code> , <code>CodeFigStyle</code> , <code>SegmentSymbol</code> | |
| <code>\pstMiddleAB</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \} \{ \langle I \rangle \}$ | 11 |
| | Construit le milieu I du segment $[AB]$. | |
| | Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> , <code>SegmentSymbol</code> , <code>CodeFig</code> , <code>CodeFigColor</code> , <code>CodeFigStyle</code> | |
| <code>\pstOIJGeonode</code> | $\langle par \rangle (x_1, y_1) \{ \langle A_1 \rangle \} \{ \langle O \rangle \} \{ \langle I \rangle \} \{ \langle J \rangle \} (x_2, y_2) \{ \langle A_2 \rangle \} \dots (x_n, y_n) \{ \langle A_n \rangle \}$ | 4 |
| | Création d'une liste de points dans le repère $(O; I; J)$. | |
| | Paramètres : <code>PointName</code> , <code>PointNameSep</code> , <code>PosAngle</code> , <code>PointSymbol</code> , <code>PtNameMath</code> | |
| <code>\pstOrtSym</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \} \{ \langle M_1, M_2, \dots, M_n \rangle \} \{ \langle M'_1, M'_2, \dots, M'_p \rangle \}$ | 9 |
| | Construit le symétrique M'_i de chacun des M_i par rapport à la droite (AB) . | |
| | Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> , <code>CodeFig</code> , <code>CodeFigColor</code> , <code>CodeFigStyle</code> | |
| <code>\pstOutBissectBAC</code> | $\langle par \rangle \{ \langle B \rangle \} \{ \langle A \rangle \} \{ \langle C \rangle \} \{ \langle M \rangle \}$ | 13 |
| | Construit la bissectrice extérieure d'un angle ainsi qu'un point M de la bissectrice image de B par rotation autour de point A . | |
| | Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> | |
| <code>\pstProjection</code> | $\langle par \rangle \{ \langle A \rangle \} \{ \langle B \rangle \} \{ \langle M_1, M_2, \dots, M_n \rangle \} \{ \langle M'_1, M'_2, \dots, M'_p \rangle \}$ | 11 |
| | Construit l'image M'_i de chacun des M_i par projection orthogonale sur (AB) . | |
| | Paramètres : <code>PointSymbol</code> , <code>PosAngle</code> , <code>PointName</code> , <code>PointNameSep</code> , <code>PtNameMath</code> , <code>CodeFig</code> , <code>CodeFigColor</code> , <code>CodeFigStyle</code> | |

| | |
|--|----|
| <code>\pstRightAngle[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨C⟩}</code> | 5 |
| <p>Marque l'angle droit \widehat{ABC} spécifié dans le sens trigonométrique. Paramètres : <code>RightAngleType</code>, <code>RightAngleSize</code>, <code>RightAngleSize</code></p> | |
| <code>\pstRotation[⟨par⟩]{⟨O⟩}{⟨M₁, M₂, ..., M_n⟩}{⟨M'₁, M'₂, ..., M'_p⟩}</code> | 9 |
| <p>Construit l'image M'_i de chacun des M_i par rotation d'angle <code>RotAngle</code> autour de O dans le sens trigonométrique. Paramètres : <code>PointSymbol</code>, <code>PosAngle</code>, <code>PointName</code>, <code>PointNameSep</code>, <code>PtNameMath</code>, <code>RotAngle</code></p> | |
| <code>\pstSegmentMark[⟨par⟩]{⟨A⟩}{⟨B⟩}</code> | 4 |
| <p>Permet de marquer le segment $[AB]$ en son milieu avec la marque définie par <code>SegmentSymbol</code>. Paramètres : <code>SegmentSymbol</code></p> | |
| <code>\pstSym0[⟨par⟩]{⟨O⟩}{⟨M₁, M₂, ..., M_n⟩}{⟨M'₁, M'₂, ..., M'_p⟩}</code> | 9 |
| <p>Construit le symétrique M'_i de chacun des M_i par rapport à O. Paramètres : <code>PointSymbol</code>, <code>PosAngle</code>, <code>PointName</code>, <code>PointNameSep</code>, <code>PtNameMath</code>, <code>CodeFig</code>, <code>CodeFigColor</code>, <code>CodeFigStyle</code></p> | |
| <code>\pstTranslation[⟨par⟩]{⟨A⟩}{⟨B⟩}{⟨M₁, M₂, ..., M_n⟩}{⟨M'₁, M'₂, ..., M'_p⟩}</code> | 10 |
| <p>Construit l'image M'_i de chacun des M_i par la translation de vecteur \overrightarrow{AB}. Paramètres : <code>PointSymbol</code>, <code>PosAngle</code>, <code>PointName</code>, <code>PointNameSep</code>, <code>PtNameMath</code>, <code>DistCoef</code></p> | |
| <code>\pstTriangle[⟨par⟩](x_A, y_A){⟨A⟩}(x_B, y_B){⟨B⟩}(x_C, y_C){⟨C⟩}</code> | 5 |
| <p>Définit et trace un triangle. Paramètres : <code>PointName</code>, <code>PointNameSep</code>, <code>PointSymbol</code>, <code>PointNameA</code>, <code>PosAngleA</code>, <code>PointSymbolA</code>, <code>PointNameB</code>, <code>PosAngleB</code>, <code>PointSymbolB</code>, <code>PointNameC</code>, <code>PosAngleC</code>, <code>PointSymbolC</code></p> | |

Annexe B

Les paramètres de `pst-eucl`

| Paramètre | Défaut | Signification |
|-------------------------------|----------------------|--|
| <code>PointSymbol</code> | <code>default</code> | Symbole utilisé pour marquer un point. |
| <code>PointSymbolA</code> | <code>default</code> | idem pour le premier point d'une commande en créant plusieurs. |
| <code>PointSymbolB</code> | <code>default</code> | pour le second... |
| <code>PointSymbolC</code> | <code>default</code> | pour le troisième... |
| <code>PointName</code> | <code>default</code> | Étiquette placé auprès du point. |
| <code>PointNameA</code> | <code>default</code> | idem pour le premier point d'une commande en créant plusieurs. |
| <code>PointNameB</code> | <code>default</code> | pour le second... |
| <code>PointNameC</code> | <code>default</code> | pour le troisième... |
| <code>PtNameMath</code> | <code>true</code> | paramètre booléen spécifiant si le style mathématique doit être utilisé pour l'étiquette d'un point. |
| <code>SegmentSymbol</code> | <code>default</code> | Symbole utilisé pour marquer un segment |
| <code>SegmentSymbolA</code> | <code>default</code> | idem pour le premier segment d'une commande en marquant plusieurs. |
| <code>SegmentSymbolB</code> | <code>default</code> | pour le second... |
| <code>SegmentSymbolC</code> | <code>default</code> | pour le troisième... |
| <code>Mark</code> | <code>default</code> | Symbole utilisé pour marquer un angle... |
| <code>MarkAngle</code> | <code>default</code> | Angle à utiliser pour le symbole précédent... |
| <code>PointNameSep</code> | <code>1em</code> | Distance séparant le symbole du point de son étiquette. |
| <code>PosAngle</code> | <code>undef</code> | Position de l'étiquette autour du point. |
| <code>PosAngleA</code> | <code>undef</code> | idem pour le premier point d'une commande en créant plusieurs. |
| <code>PosAngleB</code> | <code>undef</code> | pour le second... |
| <code>PosAngleC</code> | <code>undef</code> | pour le troisième... |
| <code>RightAngleSize</code> | <code>.4</code> | Taille du symbole de l'angle droit |
| <code>RightAngleType</code> | <code>default</code> | Type de la marque de l'angle droit, valeur possible : <code>german</code> et <code>suisseromand</code> |
| <code>MarkAngleRadius</code> | <code>.4</code> | Rayon de l'arc de cercle marquant un angle. |
| <code>LabelAngleOffset</code> | <code>0</code> | Offset angulaire permettant de décaler l'étiquette d'un angle. |
| <code>LabelSep</code> | <code>1</code> | Distance entre le sommet de l'angle et l'étiquette. |
| <code>LabelRefPt</code> | <code>c</code> | Point de référence <code>T_EX</code> utilisé pour placer l'étiquette d'un angle. |
| <code>HomCoef</code> | <code>.5</code> | Coefficient de l'homothétie. |
| <code>RotAngle</code> | <code>60</code> | Angle utilisé pour la rotation. |
| ... à suivre ... | | |

| Paramètre | Défaut | Signification |
|----------------|--------|---|
| DrawCirABC | true | Paramètre booléen indiquant si le cercle circonscrit doit être tracé. |
| CodeFig | false | Paramètre booléen indiquant si la construction doit être codée. |
| CodeFigA | false | idem pour la première construction d'une commande en générant plusieurs. |
| CodeFigB | false | La seconde. |
| CodeFigColor | cyan | Couleur du trait pour coder la construction. |
| CodeFigStyle | dashed | Style du trait pour coder la construction. |
| CodeFigArc | true | paramètre booléen spécifiant le sens de l'arc au niveau de la première intersection entre deux cercles. |
| CodeFigBarc | true | idem pour la seconde. |
| Radius | none | Spécification du rayon du cercle. |
| RadiusA | undef | Spécification du rayon du premier cercle. |
| RadiusB | undef | Spécification du rayon du second cercle. |
| Diameter | none | Spécification du diamètre du cercle. |
| DiameterA | undef | Spécification du rayon du premier cercle. |
| DiameterB | undef | Spécification du rayon du second cercle. |
| DistCoef | none | Coefficient multiplicatif permettant de modifier une distance/vecteur. |
| AngleCoef | none | Coefficient multiplicatif permettant de modifier la mesure d'un angle. |
| CurvAbsNeg | false | Paramètre booléen indiquant que les abscisses curvilignes sont dans le sens anti-trigonométrique |
| GenCurvFirst | none | Nom du premier point d'une courbe générique (effet de bord). |
| GenCurvLast | none | Nom du dernier point d'une courbe générique (effet de bord). |
| GenCurvInc | none | valeur à utiliser comme incrément dans une courbe générique. |
| CurveType | none | Mode de tracé de courbe pour relier une liste de points. |
| TransformLabel | none | étiquette utilisée pour marquer la rotation ou la translation. |

Annexe C

Compatibilité ascendantes de `pst-eucl`

Nouveauté pour cette version, des commandes ont changé de syntaxe sans changer de nom, la compatibilité ascendante n'est donc pas maintenue. Toutefois, pour éviter à certains utilisateurs pressés de mettre à jour leur fichier dans l'urgence, il est possible de retrouver l'ancienne syntaxe en activant l'option `old` à l'inclusion de l'extension `\usepackage[old]{pst-eucl}`. Pour cette version cela concerne toutes les commandes des transformations. Il faut donc se reporter à l'ancien manuel pour avoir la syntaxe.