

1 Introduction

`mpgdd` est un ensemble d'outils pour la géométrie plane avec `METAPOST`. Cet ensemble se compose de plusieurs fichiers :

1. `gdd.mp` : c'est le fichier principal, il contient les structures et fonctions générales.
2. `gdd-arc.mp` : contient tout ce qui concerne les arcs de cercles.
3. `gdd-c2d.mp` : contient tout ce qui concerne les courbes du second degré.
4. `gdd-fct.mp` : contient quelques fonctions mathématiques usuelles.
5. `gdd-lbl.mp` : contient les fonctions relatives aux labels.
6. `gdd-plt.mp` : contient des fonctions facilitant la représentation de fonctions mathématiques.
7. `gdd-rep.mp` contient différents outils pour le tracé de figure dans un repère.
8. `gdd-tra.mp` contient les fonction permettant de gérer la transparence (code emprunté à Anthony PHAN).

Nous allons, dans la suite, décrire plus en détails chacune de ces fonctions. Il est à noter aussi que certaines fonctions s'appuient sur l'extension `graph.mp` présent dans toutes les bonnes distributions TEX .

2 Principe général de fonctionnement

`mpgdd` utilise des tables comme structure principale. Chaque objet est numéroté via le compteur `gdd0`, son type¹ est stocké dans la table `gddT[]` à la place `gddT[gdd0]`. Les propriétés des objets sont définies dans, là encore, des tables de type `numeric` qui sont `gddA[]`, `gddB[]`, ..., `gddF[]`.

Par exemple, pour un `Point` (type `mpgdd`), la première coordonnée se trouve dans `gddA[]` et la seconde dans `gddB[]` (les autres table ne sont pas utilisées pour un tel objet).

Il y a deux tables particulières `gddP[]` qui est du type `path` et `gddS[]` qui est du type `string`. Nous verrons plus tard quelle est leur utilité.

Bien entendu, lors d'une utilisation classique de `mpgdd`, l'appel à toutes ces tables n'est pas chose courante.

3 Les types

On peut, avec `mpgdd`, construire plusieurs types d'objets. Rappelons le, tout est *objet* dont le nombre est enregistré dans la variable `gdd0`. Le type d'objet, lui, est stocké dans la table `gddT[]`, et les tables `gddA[]` à `gddF[]` contiennent les propriétés des objets.

Nous allons ici décrire chaque type de l'extension `mpgdd` ainsi que leurs propriétés respectives.

Le type `point` Ce type correspond au point de l'espace euclidien. Pour être plus clair voici la fonction principale pour créer un tel objet :

```
vardef Point(expr a,b) =
  gddT[incr gdd0] = "point";
  gddA[gdd0] = a; gddB[gdd0] = b; gdd0
enddef;
```

1. Les types son propres à `mpgdd` et seront décrit plus tard.

```

input gdd;
input gdd-rep;
input gdd-lbl;
input gdd-tra;
input svgnames;

lblPreamble := "\usepackage[frenchb]{babel}
" &
" \usepackage[garamond]{
mathdesign}" &
"\usepackage{amsmath}";

labeloffset := 6;

gddTaillePoint := 3;
gddCouleurPoint := Yellow;

beginfig(1);

Repere(9,9,4,5,0.8,0.8);
Axes;
Debut;
Axes;
Graduations; Unites(1);

C1 = Cercle((1,1),3);
C2 = Cercle((-1,-2),1);

P1 = IntersectionCercles(C1,C2);
P2 = IntersectionCercles(C2,C1);

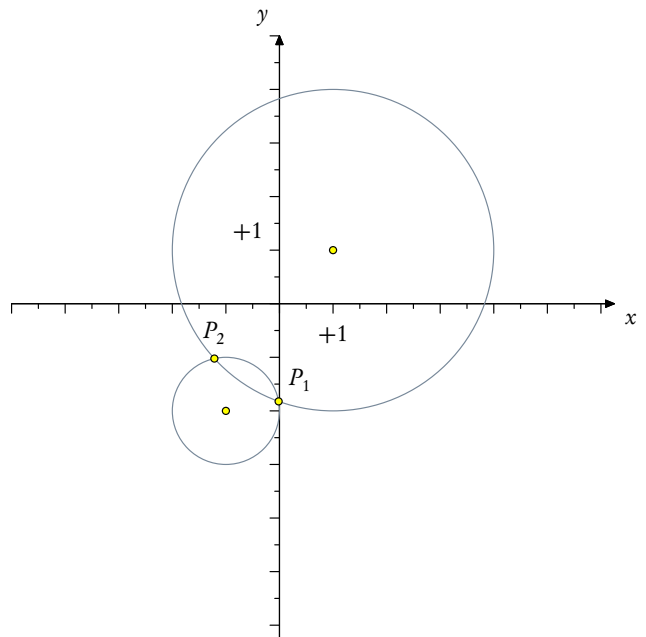
drawoptions(withcolor LightSlateGrey);
trace C1;
trace C2;

drawoptions();
pointe Point(1,1);
pointe Point(-1,-2);
pointe P1;
pointe P2;
label.urt(TEX("\(P_1\)"),PtR(P1));
label.top(TEX("\(P_2\)"),PtR(P2));

Fin;

endfig;
end

```



4 Fonctions générales

4.1 Relatives aux Points

4.2 Un peu de géométrie