

tikz2d-fr

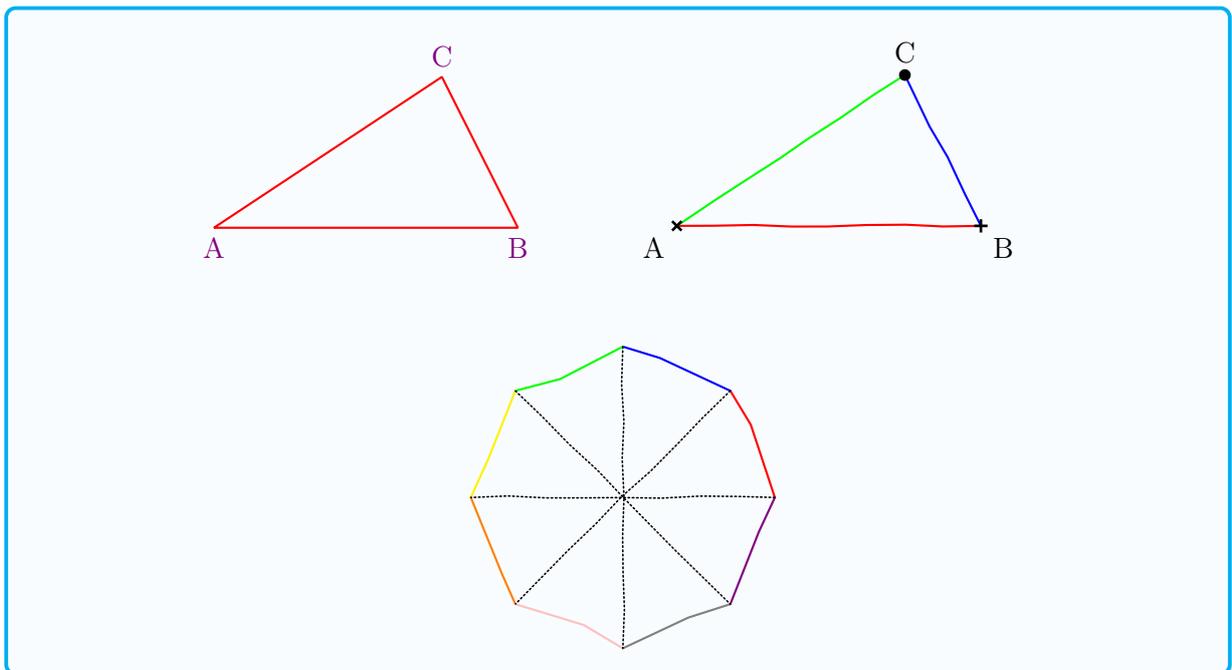
Quelques commandes simplifiées (fr)
pour utiliser avec TikZ.

Version 0.1.1 - 28 Mai 2023

Cédric Pierquet

c pierquet - at - outlook . fr
<https://github.com/cpierquet/tikz2dfr>

- ▶ Définir des points par coordonnées.
- ▶ Définir des points par transformations simples.
- ▶ Tracer des segments, avec option(s) de couleurs.
- ▶ Marquer des points.



L^AT_EX

pdfL^AT_EX

LuaL^AT_EX

TikZ

T_EXLive

MiK_TE_X

Table des matières

I	Historique	2
II	Introduction	3
1	Le package tikz2d-fr	3
1.1	Introduction	3
1.2	Autres solutions	3
1.3	Chargement du package, packages utilisés	3
III	Styles et commandes	4
2	Tracé à main levée	4
2.1	Idée	4
2.2	Utilisation basique	4
3	Définir des points par coordonnées	5
3.1	Commandes	5
3.2	Clés, options et arguments	5
4	Définir des points par transformation	7
4.1	Commandes	7
4.2	Clés, options et arguments	8
5	Tracer des segments	10
5.1	Commandes	10
5.2	Clés, options et arguments	10

Première partie

Historique

v0.1.1 : Ajout d'une clé `\[TaillePolice]` + Points définis par une transformation simple.

v0.1.0 : Version initiale.

Deuxième partie

Introduction

1 Le package tikz2d-fr

1.1 Introduction



Le package propose des commandes basiques – francisées – pour travailler sur des figures simples en 2D, à l'aide de TikZ avec :

- de quoi simuler un tracé à *main levée* ;
- de quoi définir des points (et donc des nœuds) via `nom/coordonnées/positionlabel` ;
- `0.1.1` de quoi définir des points images par des transformations simples ;
- de marquer des points (croix, disques) ;
- de quoi tracer des segments (enchaînés ou non) en spécifiant les couleurs éventuelles.



On peut voir ce modeste package comme un *maigre* complément à TikZ et tkz-euclide¹ qui permet de faire énormément de choses en géométrie euclidienne.

Ainsi, les éventuel points (nœuds) créés par tikz2d-fr seront réutilisables dans toute commande classique de TikZ ou de tkz-euclide.

On peut donc *utiliser* le package tikz2d-fr comme *base* de travail avant des choses plus poussées !

1.2 Autres solutions



À noter qu'il existe d'autres solutions graphiques, comme :

- PSTricks avec pst-eucl² ;
- MetaPost avec ProfCollege et son environnement `geometrie`³.

1.3 Chargement du package, packages utilisés



Le package se charge, de manière classique, dans le préambule.

Il n'existe pas d'option pour le package, et `xcolor` n'est pas chargé.

```
\documentclass{article}
\usepackage{tikz2d-fr}
```



tikz2d-fr charge les packages suivantes :

- `tikz`, `xstring`, `simplekv`, `xintexpr` et `listofitems` ;
- les librairies `tikz.decorations` et `tikz.decorations.pathmorphing`.

Il est compatible avec les compilations usuelles en latex, pdflatex, lualatex ou xelatex.

1. <https://ctan.org/pkg/tkz-euclide>
2. <https://www.ctan.org/pkg/pst-eucl>
3. <https://www.ctan.org/pkg/profcollege>

Troisième partie

Styles et commandes

2 Tracé à main levée

2.1 Idée



L'idée est de *proposer* un style *tout prêt* pour simuler un tracé, en TikZ, à « main levée ». Il s'agit d'un style *basique* utilisant la librairie `decorations` avec `random steps`.

```
\tikzset{%
  mainlevee/.style args={#1et#2}{%
    decorate,decoration={random steps, segment length=#1,amplitude=#2}},
  mainlevee/.default={5mm et 0.6pt}
}
```

2.2 Utilisation basique



Il s'agit ni plus ni moins d'un style TikZ à intégrer dans les tracés et constructions TikZ!

Pour ce style, deux paramètres peuvent être précisés via `<mainlevee=#1 et #2>` :

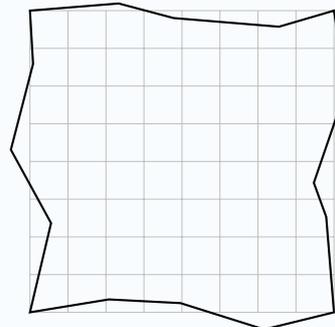
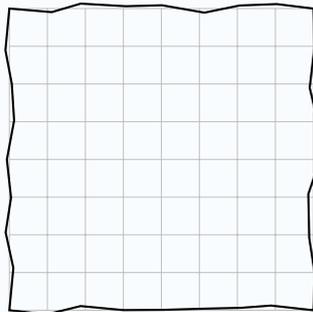
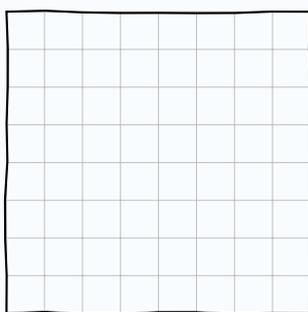
- `<#1>` correspond à l'option `segment length` (longueur des segments *types*) ;
défaut **<5mm>**
- `<#2>` correspond à l'option `amplitude` (amplitude maximale de la *déformation*).
défaut **<0.6pt>**

Les valeurs `<mainlevee=5mm et 0.6pt>` donnent des résultats – à mon sens – satisfaisants, mais l'utilisateur pourra modifier à loisir ces paramètres !

```
%la grille a été rajoutée pour la sortie
\begin{tikzpicture}
  \draw[thick,mainlevee] (0,0) rectangle (4,4) ;
\end{tikzpicture}

\begin{tikzpicture}
  \draw[thick,mainlevee=5mm et 2pt] (0,0) rectangle (4,4) ;
\end{tikzpicture}

\begin{tikzpicture}
  \draw[thick,mainlevee=10mm et 3mm] (0,0) rectangle (4,4) ;
\end{tikzpicture}
```



3 Définir des points par coordonnées

3.1 Commandes



Les commandes *simplifiées* et *francisées* disponibles sont :

- `\DefinirPoint` pour définir un point ;
- `\DefinirPoints` pour définir des points ;
- `\MarquerPoint` pour marquer (matérialiser) un point ;
- `\MarquerPoints` pour marquer (matérialiser) des points.

```
\begin{tikzpicture}[options tikz]
  %créer/placer/nommer un point
  \DefinirPoint(*)[clés]{noeud}{coordonnées}<label>
  %créer/placer/nommer plusieurs points, sans modification des labels !
  \DefinirPoints(*)[clés]{liste}
  %marquer un point
  \MarquerPoint[clés]{point}
  %marquer plusieurs points
  \MarquerPoints[clés]{liste}
\end{tikzpicture}
```

3.2 Clés, options et arguments



Les versions étoilées désactivent l’affichage des labels des points.

L’argument optionnel et entre [...] propose les **<clés>** suivantes (communes ou spécifiques) :

- **<PosLabel>** pour préciser la position (francisée) du label pour les points ; défaut : ****
-  **<TaillePolice>** pour spécifier la taille de la police du label ;
défaut : **<\normalsize>**
- **<StyleMarque>** parmi **<x/o/+>** pour spécifier le style de la marque des points ;
défaut : **<o>**
- **<Math>** : booléen pour afficher le label en mode math ; défaut : **<false>**
- **<TailleMarque>** pour spécifier la taille de la marque des points (disque/ croix) ;
défaut : **<2pt>**
- **<Couleur>** pour paramétrer la couleur. défaut : **<black>**



Les positions pour les labels des points sont *francisées* :

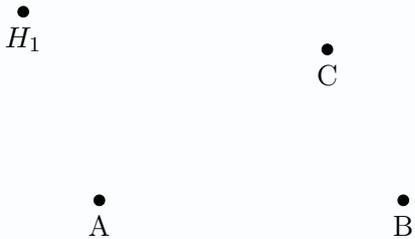
- **** : bas
- **<h>** : haut
- **<g>** : gauche
- **<d>** : droite
- **<hg>** : haut gauche
- ...



De manière un peu plus spécifique :

- le `<label>` pour la commande `\DefinirPoint` est optionnel, et est identique à `{nœud}` ;
- la `{liste}` pour la commande `\DefinirPoints` est à donner – par exemple – sous la forme `A/xa,ya/poslabel B/alpha:r/poslabel ...`

```
\begin{tikzpicture}
  \DefinirPoint{A}{0,0}
  \DefinirPoints{B/4,0/ C/3,2/}
  \DefinirPoints[Math]{H_1/-1,2.5/}
  \MarquerPoints{A,B,C,H_1}
\end{tikzpicture}
```



```
\begin{tikzpicture}
  \DefinirPoints[DecalLabel=1pt,Couleur=blue,Math]{A/0,0/bg B/4,0/bd}
  \DefinirPoints[DecalLabel=1pt,Couleur=red,TaillePolice=\LARGE]{C/3,2/h}
  \MarquerPoints[StyleMarque=x,Couleur=violet]{A}
  \MarquerPoints[Couleur=orange]{B}
  \MarquerPoint[StyleMarque=+,Couleur=red]{C}
\end{tikzpicture}
```



4 Définir des points par transformation

4.1 Commandes



0.1.1 Les commandes *simplifiées* et *francisées* disponibles sont :

- `\ImagePoint` pour définir un point par transformation ;
- `\ImagesPoints` pour définir des points par transformation.

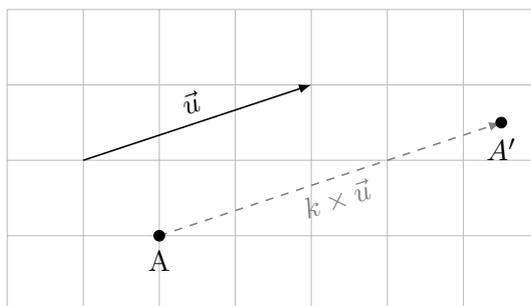


Les *transformations* disponibles sont (l'écriture *mathématique* sert à illustrer la syntaxe \LaTeX utilisée) :

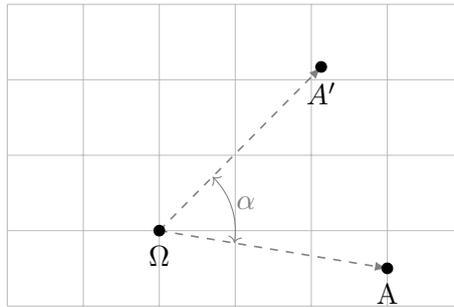
- les translations glissées $t(A) = A + k \times \vec{u}$;
- les rotations $r(A) = A + \text{rot}(\Omega, \theta)$;
- les symétries centrales $s(A) = A + \text{sym}(I)$.

```
\begin{tikzpicture}[options tikz]
  %création des points utiles
  \DefinirPoint... \DefinirPoints
  %point défini par une transformation
  \ImagePoint(*)[clés]{formule}<label point>
  %points définis par des transformations, sans modification des labels !
  \ImagesPoints(*)[clés]{formule § formule § ...}
\end{tikzpicture}
```

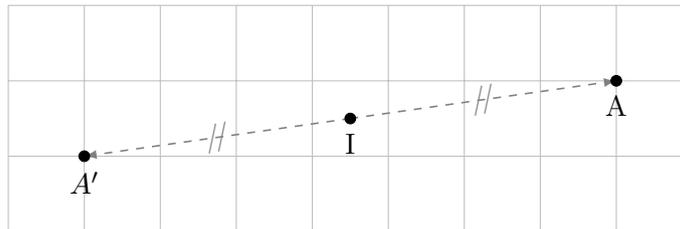
```
\begin{tikzpicture}[options tikz]
  %création des points utiles
  \DefinirPoint...
  %point défini par une translation glissée
  \ImagePoint(*)[clés]{PtArr=PTDep+t(k,DebVect>FinVect)}
\end{tikzpicture}
```



```
\begin{tikzpicture}[options tikz]
  %création des points utiles
  \DefinirPoint...
  %point défini par une rotation
  \ImagePoint{PtArr=PTDep+r(centre,angle)}
  %point défini par une symétrie centrale
\end{tikzpicture}
```



```
\begin{tikzpicture}[options tikz]
  %création des points utiles
  \DefinirPoint...
  \ImagePoint{PtArr=PTDep+s(centre)}
\end{tikzpicture}
```



4.2 Clés, options et arguments



Les versions étoilées désactivent l'affichage des labels des points.

L'argument optionnel et entre [...] propose les **clés** suivantes (communes ou spécifiques) :

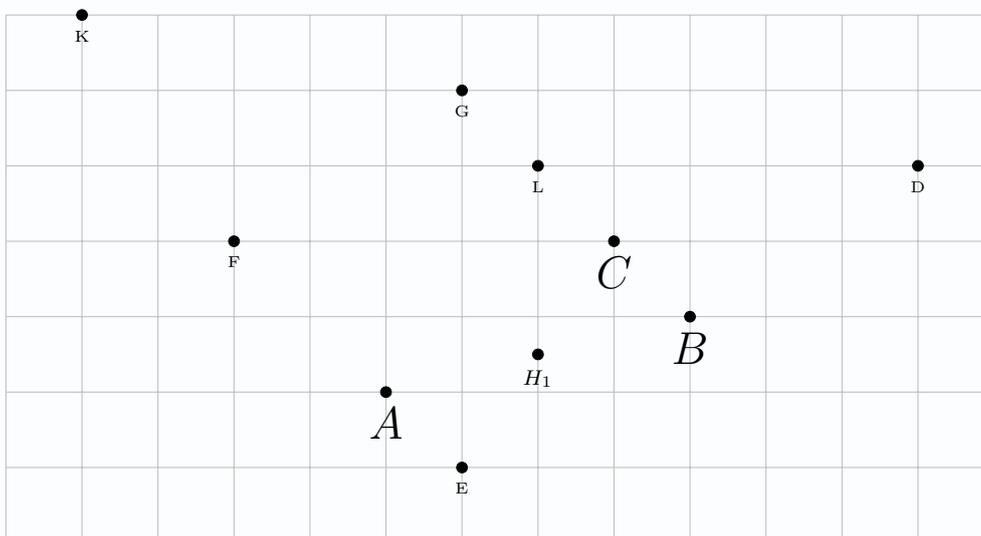
- **PosLabel** pour préciser la position (francisée) du label pour les points; défaut : ****
- **TaillePolice** pour spécifier la taille de la police du label; défaut : **<\normalsize>**
- **StyleMarque** parmi **<x/o/+>** pour spécifier le style de la marque des points; défaut : **<o>**
- **Math** : booléen pour afficher le label en mode math; défaut : **<false>**
- **TailleMarque** pour spécifier la taille de la marque des points (disque/ croix); défaut : **<2pt>**
- **Couleur** pour paramétrer la couleur. défaut : **<black>**



On donne par exemple $A(0;0)$, $B(4;1)$ et $C(3;2)$ et on définit les points :

- D image de A par la translation de vecteur \overrightarrow{AB} ;
- E image de A par la translation de vecteur \overrightarrow{CB} ;
- F image de A par la translation-glissée de vecteur $2\overrightarrow{DB}$;
- G image de F par la translation de vecteur \overrightarrow{BD} ;
- K image de G par la rotation de centre F et d'angle 90° ;
- L image de B par la symétrie centrale de centre C ;
- H_1 image de A par la translation-glissée de vecteur $-0,5\overrightarrow{BA}$.

```
\draw[thin,lightgray] (-5,-2) grid (8,5) ;
\DefinirPoints[Math,TaillePolice=\LARGE]{A/0,0/ B/4,1/ C/3,2/}
\ImagesPoints[TaillePolice=\tiny]{D=C+t(A>B) § E=A+t(C>B) § F=A+t(2,B>C) §
  G=F+t(B>D) § K=G+r(F,90) § L=B+s(C)}
\ImagePoint[Math,TaillePolice=\scriptsize]{H_1=A+t(-0.5,B>A)}
\MarquerPoints{A,B,C,D,E,F,G,K,L}
\MarquerPoint{H_1}
```



5 Tracer des segments

5.1 Commandes



La commande *simplifiée* et *francisée* pour tracer des segments est `\TracerSegments`.

Le fonctionnement de cette commande permet de définir les segments :

- sous la forme `A/B C/D ...` dans le cas de segments *non forcément contigus* ;
- sous la forme `A B C ...` dans le cas de segments *contigus* (avec possibilité de *fermer* le chemin).

```
%version liste de segments (AB) et (CD) et ...
\begin{tikzpicture}
  \TracerSegments[clés]<options tikz>{ptA/ptB ptC/ptD ...}
\end{tikzpicture}

%version suite de points (AB) et (BC) et ...
\begin{tikzpicture}
  \TracerSegments[Suite,clés]<options tikz>{ptA ptB ptC ...}
\end{tikzpicture}
```

5.2 Clés, options et arguments



Concernant les clés, qui correspondent au premier argument, optionnel et entre [...] (certaines clés sont inactives dans certaines situations...) :

- `<Cap>` : type de *jointure* pour les segments individuels ; défaut `<round>`
- `<Join>` : type de *jointure* pour les segments enchaînés ; défaut `<miter>`
- `<Couleurs>` : sous la forme `<couleur unique>` ou `<liste,de,couleurs>` (cohérente avec le nombre de segments !); défaut `<black>`
- `<Cycle>` : booléen pour préciser qu'on veut *fermer* le chemin ; défaut `<false>`
- `<Suite>` : booléen pour préciser qu'on veut travailler avec des points enchaînés. défaut `<false>`



Le deuxième argument, optionnel et entre `<...>` correspond aux options, en langage TikZ à passer à l'ensemble des segments.

Le troisième argument, obligatoire et entre `{...}` correspond à la suite des segments à tracer :

- sous la forme `ptA/ptB ptC/ptD ...` pour des segments individuels ;
- sous la forme `ptA ptB ptC ...` (avec la clé `<Suite>`) pour des segments enchaînés.

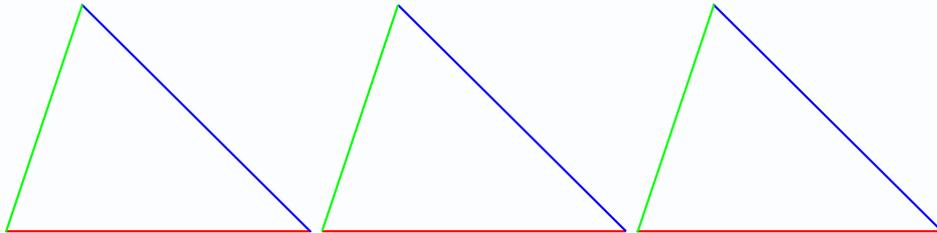


À noter que dans le cas d'une succession de segments (`<Suite>`) monochromes, cela revient à un chemin et donc ce sera la clé `<Join=...>` qui sera utile.

```

\begin{tikzpicture}
  \TracerSegments[Suite,Couleurs={red,blue,green}]{0,0 4,0 1,3 0,0}
\end{tikzpicture}
\begin{tikzpicture}
  \TracerSegments[Suite,Cap=butt,Couleurs={red,blue,green}]{0,0 4,0 1,3 0,0}
\end{tikzpicture}
\begin{tikzpicture}
  \TracerSegments[Suite,Cap=rect,Couleurs={red,blue,green}]{0,0 4,0 1,3 0,0}
\end{tikzpicture}

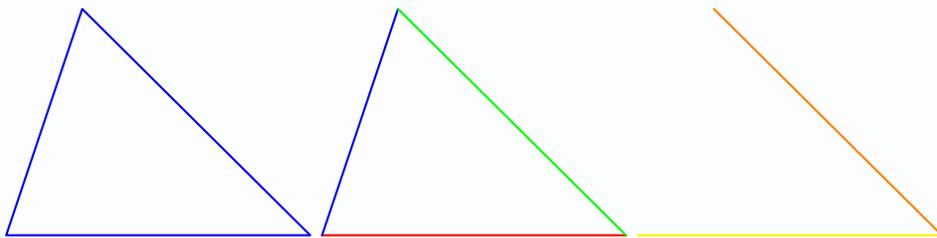
```



```

\begin{tikzpicture}
  \TracerSegments[Cycle,Suite,Couleurs=blue,Join=round]{0,0 4,0 1,3}
\end{tikzpicture}
\begin{tikzpicture}
  \TracerSegments[Couleurs={red,blue,green}]{0,0/4,0 1,3/0,0 4,0/1,3}
\end{tikzpicture}
\begin{tikzpicture}
  \TracerSegments[Suite,Couleurs={yellow,orange}]{0,0 4,0 1,3}
\end{tikzpicture}

```



```

\begin{tikzpicture}
  \DefinirPoints[DecalLabel=1pt]{A/0,0/bg B/4,0/bd C/3,2/h}
  \TracerSegments[Suite,Cycle,Couleurs={red,blue,green}]{A B C}
  \MarquerPoints[StyleMarque=x]{A,B,C}
\end{tikzpicture}
\begin{tikzpicture}
  \DefinirPoints[DecalLabel=1pt,Math]{A/0,0/bg B/4,0/bd C/3,2/h}
  \TracerSegments[Couleurs={yellow,red}]{A/B B/C}
  \TracerSegments[Couleurs=blue]{C/A}
  \MarquerPoints{A,B,C}
\end{tikzpicture}

```

