

tkz-grapheur-altversion (v0.31f)

Version alternative de l'environnement graphique,
basée sur TikZ, xint et la normalisation interne.

Chargé automatiquement avec tkz-grapheur

Cédric Pierquet
c pierquet - at - outlook . fr
<https://github.com/cpierquet/latex-packages/tree/main/tkz-grapheur>

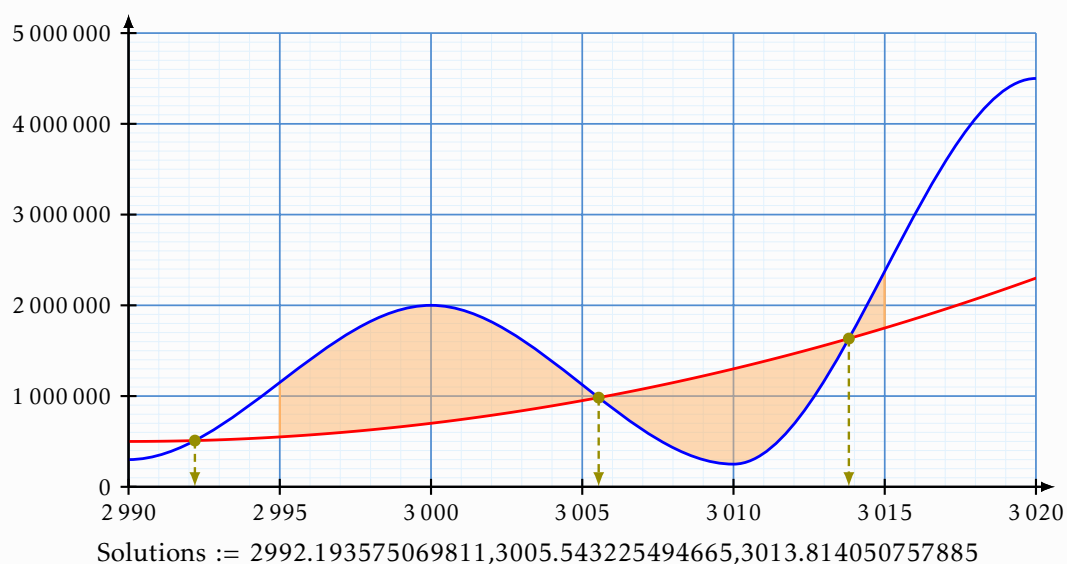
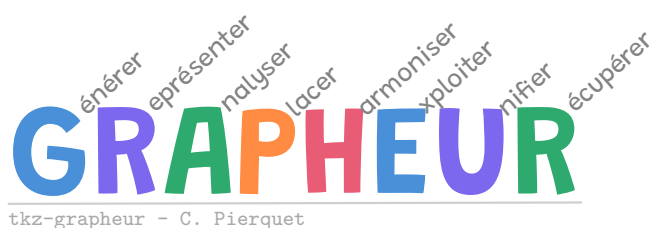


Table des matières

1	Introduction et philosophie	3
1.1	Problématique	3
1.2	Principe de la normalisation	3
1.3	Philosophie : tout est un path	3
1.4	Rétrocompatibilité	3
2	Environnement tkzGrph	4
2.1	Syntaxe	4
2.2	Clés disponibles	4
2.3	Nœuds créés	4
2.4	Axes et grilles	5
2.5	Styles des points	6
3	Définir et tracer des courbes	7
3.1	Courbe de fonction : <code>\tkzGrphDefineCurve</code>	7
3.2	Interpolation lisse : <code>\tkzGrphDefineInterpoCurve</code>	7
3.3	Spline de Hermite : <code>\tkzGrphDefineSpline</code>	7
3.4	Tracer des courbes existantes	7
4	Compléments	9
4.1	Récupération de coordonnées réelles	9
4.2	Création de nœuds : <code>\tkzGrphCreateNode(s)</code>	10
4.3	Marquage de points : <code>\tkzGrphMarkPts</code>	10
5	Analyse(s) et exploitation(s)	11
5.1	Intégrales : <code>\tkzGrphDrawIntg</code>	11
5.2	Intersections : <code>\tkzGrphFindItsc</code>	12
5.3	Images et antécédents	13
5.3.1	Image : <code>\tkzGrphFindY</code>	13
5.3.2	Images : <code>\tkzGrphFindListY</code>	13
5.3.3	Antécédents : <code>\tkzGrphFindX</code>	13
5.4	Extremums : <code>\tkzGrphFindMin/Max</code>	14
5.5	Points d'intérêt : <code>\tkzGrphInterestPoints</code>	15
6	Nuage et régression : statistiques	19
6.1	Nuage de points : <code>\tkzGrphScatterPlot</code>	19
6.2	Droite de régression : <code>\tkzGrphDrawRegLin</code>	19
6.3	Régressions : <code>\tkzGrphDrawFit</code>	20
7	Compléments sur les axes	21
7.1	Valeurs manuelles sur les axes	21
7.2	Placer un texte : <code>\tkzGrphPlaceText</code>	22
8	Annotations graphiques	23
8.1	Label sur un path : <code>\tkzGrphLabelPath</code>	23
8.2	Point sur un path : <code>\tkzGrphMarkOnPath</code>	23
8.3	Intervalle sur un axe : <code>\tkzGrphIntervalX/Y</code>	24

1 Introduction et philosophie

1.1 Problématique

L'environnement `GraphiqueTikz` de `tkz-grapheur` travaille directement en coordonnées `TikZ` — des valeurs extrêmes sur les axes (années, populations, distances...) génèrent des erreurs `dimension too large`.

L'environnement alternatif `tkzGrph` résout ce problème par une **normalisation interne** : `TikZ` travaille toujours dans $[0, L] \times [0, H]$ (en cm), quelle que soit l'amplitude réelle.

1.2 Principe de la normalisation

Pour une fenêtre $[X_{\min}, X_{\max}] \times [Y_{\min}, Y_{\max}]$ et des dimensions $L \times H$:

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \times L \quad y_{\text{norm}} = \frac{y - y_{\min}}{y_{\max} - y_{\min}} \times H$$

`TikZ` ne voit jamais que des coordonnées dans $[0, L] \times [0, H]$.

1.3 Philosophie : tout est un path

Toutes les courbes sont des **name path nommés** — fonction `xint`, interpolation ou spline. Cela permet :

- de **réutiliser** un tracé avec des styles différents via `\tkzGrphDrawExistingCurve(s)` ;
- de calculer des **intersections** entre n'importe quelles courbes ;
- de calculer des **intégrales** entre n'importe quelles courbes ;
- d'**unifier** la syntaxe indépendamment du type de courbe.

1.4 Rétrocompatibilité

L'environnement `tkzGrph` accepte les mêmes noms de clés que `GraphiqueTikz` (`Xmin`, `Xgrille`...) ainsi que leurs alias anglais (`width`, `xgrid`...).

2 Environnement tkzGrph

2.1 Syntaxe

```
\begin{tkzGrph}[options tikz]<clés>
...
\end{tkzGrph}
```

2.2 Clés disponibles

- `Xmin/Xmax` : bornes de l'axe Ox , $0/10$ par défaut;
- `Ymin/Ymax` : bornes de l'axe Oy , $0/10$ par défaut;
- `Largeur=...` / `width=...` : largeur (dimension pure), `8cm` par défaut;
- `Hauteur=...` / `height=...` : hauteur (dimension pure), `6cm` par défaut;
- `x=...` : unité x éventuelle (dimension pure), vide par défaut;
- `y=...` : unité y éventuelle (dimension pure), vide par défaut;
- `Origx=...` / `Origy=...` : position de l'origine, `{0}` par défaut;
- `Xgrille=...` / `xgrid=...` : pas de grille principale en x , `1` par défaut;
- `Xgrilles=...` / `xgrids=...` : pas de grille secondaire en x , `0.5` par défaut;
- `Xgrillei=...` / `xgridi=...` : pas de grille intermédiaire en x ;
- `Ygrille=...`, `Ygrilles=...`, `Ygrillei=...` : idem en y ;
- `Theme=...` / `theme=...` : thème de couleurs (`standard`, `gris`, `bleu`, `vert`, `chaud`, `contraste`);
- `NomFigure=...` / `figurename=...` : préfixe des nœuds pour plusieurs figures;
- `TailleGrad=...` / `gradsize=...` : taille des graduations (longueur ou dessus/dessous);
- `AffCadre` / `showframe` : affiche un cadre;
- `Milli` / `milli` : grille millimétrée.

☛ `Largeur` et `Hauteur` attendent des **dimensions pures** (`8cm`, `120mm`...).

☛ Les clés `x` et/ou `y` permettent de déterminer automatiquement la largeur et/ou la hauteur du graphique, et écrasent donc les clés `Largeur` et/ou `Hauteur`.

2.3 Nœuds créés

- `grph-ne`, `grph-nw`, `grph-se`, `grph-sw` : coins de la fenêtre;
- `grph-n`, `grph-s`, `grph-e`, `grph-w`, `grph-c` : milieux des bords et centre;
- `grph-axox-w`, `grph-axox-e`, `grph-axoy-s`, `grph-axoy-n` : bords des axes;
- `grph-orig` : origine du repère.

2.4 Axes et grilles

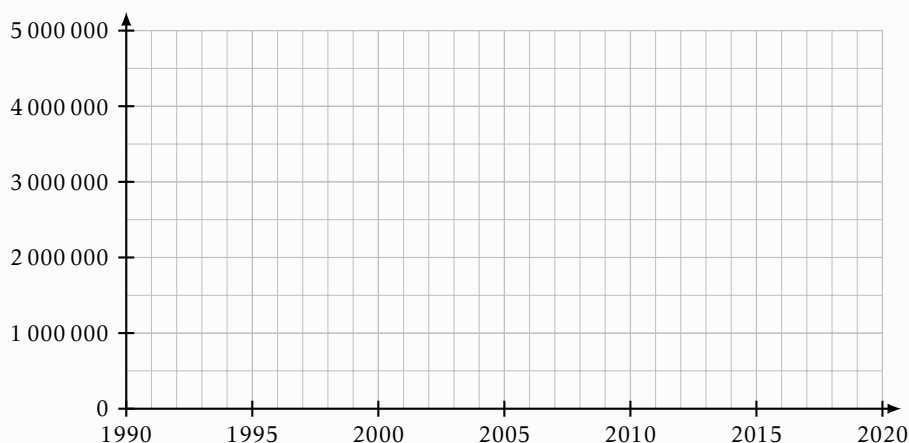
```
\tkzGrphDrawGridAxis[clés]{liste valeurs X}{liste valeurs Y}
% alias FR :
\tkzGrphTracerAxesGrille[clés]{liste valeurs X}{liste valeurs Y}
```

Les listes sont des **valeurs réelles** — la conversion est automatique.

Les [clés] disponibles sont :

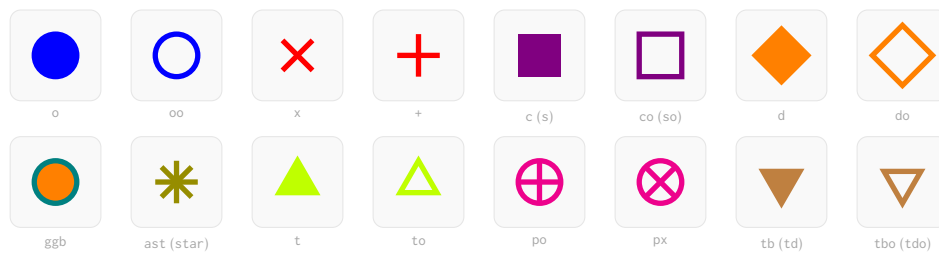
- Grille / grid : booléen, true par défaut;
- GrilleIntermediaire / intermediategrid : booléen;
- Grads / labels : booléen, true par défaut;
- AxeOy / yaxis : booléen, true par défaut;
- Fleches / arrows : booléen, true par défaut;
- Elargir / enlarge : élargissement des axes;
- Police / font : police des labels;
- Format / format : format des labels (num, annee, frac...);
- Derriere / behind : grille en arrière-plan;
- Devant / front : axes en avant-plan (sans grille).

```
\begin{tkzGrph}%
  <Xmin=1990,Xmax=2020,Ymin=0,Ymax=5000000,
  Largeur=10cm,Hauteur=5cm,Xgrille=5,Xgrilles=1,
  Ygrille=1000000,Ygrilles=500000,Origx=1990>
  \tkzGrphTracerAxesGrille[Elargir=2.5mm,Police=\small,Format=annee/num]%
    {1990,1995,2000,2005,2010,2015,2020}%
    {0,1000000,2000000,3000000,4000000,5000000}
\end{tkzGrph}
```



2.5 Styles des points

Les styles disponibles pour les points sont : `o` (disque plein), `oo` (cercle vide), `x` (croix diagonale), `+` (croix droite), `c/s` (carré plein), `co/so` (carré vide), `d` (diamant plein), `do` (diamant vide), `ggb` (style GeoGebra). La clé `Couleur=A/B` permet de différencier la couleur du contour et du fond.



Ces styles sont accessibles dans toute commande permettant de marquer des points, via les clés partagées :

- `Style / style`;
- `Taillec / sizec`;
- `Taillex / sizex`;
- `Tailleo / sizeo`;
- `Epaisseur / thick`.

3 Définir et tracer des courbes

3.1 Courbe de fonction : `\tkzGrphDefineCurve`

```
\tkzGrphDefineCurve[clés]<nom fct>{expression xint}  
% alias FR :  
\tkzGrphDefinirCourbe[clés]<nom fct>{expression xint}
```

Les [clés] disponibles sont :

- Couleur / color : couleur du tracé, `black` par défaut;
- Nom / name : nom du path créé;
- Debut / start : abscisse de départ (réelle), `Xmin` par défaut;
- Fin / end : abscisse de fin (réelle), `Xmax` par défaut;
- Pas / step : pas du tracé, calculé automatiquement par défaut;
- Trace / draw : booléen, `true` par défaut.

3.2 Interpolation lisse : `\tkzGrphDefineInterpoCurve`

```
\tkzGrphDefineInterpoCurve[clés]{x/y $ x/y $ ...}  
% alias FR :  
\tkzGrphDefinirCourbeInterpo[clés]{x/y $ x/y $ ...}
```

Séparateur `$` entre les points, `/` entre les coordonnées. Clé spécifique : `tension` (défaut `0.5`).

3.3 Spline de Hermite : `\tkzGrphDefineSpline`

```
\tkzGrphDefineSpline[clés]{x/y/f'(x) $ x/y/f'(x) $ ...}  
% alias FR :  
\tkzGrphDefinirCourbeSpline[clés]{x/y/f'(x) $ x/y/f'(x) $ ...}
```

Remarques :

- Pour chaque point : abscisse / ordonnée / dérivée (toutes en valeurs réelles).
- La dérivée est normalisée automatiquement.
- Clé spécifique (Hermite) : `Alt` (mode alternatif Bézier, conseillé pour les courbes oscillantes).

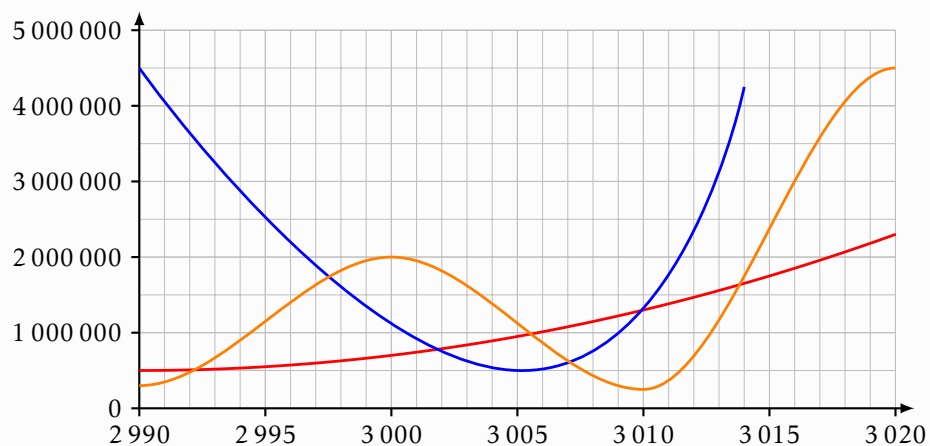
3.4 Tracer des courbes existantes

```
% tracer UN path existant avec un style  
\tkzGrphDrawExistingCurve[options draw]{nom path}  
  
% tracer PLUSIEURS paths en une fois : nom/couleur,...  
\tkzGrphDrawExistingCurves[options draw]{nom/couleur, nom/couleur, ...}  
% alias FR :  
\tkzGrphTracerCourbe / \tkzGrphTracerCourbes
```

```

\begin{tkzGrph}%
  <Xmin=2990,Xmax=3020,Ymin=0,Ymax=5000000,
  Largeur=10cm,Hauteur=5cm,Xgrille=5,Xgrilles=1,
  Ygrille=1000000,Ygrilles=500000,Origx=2990,%
  TailleGrad=0pt/3pt>
  \tkzGrphTracerAxesGrille[Elargir=2.5mm]%
    {2990,2995,3000,3005,3010,3015,3020}%
    {0,1000000,2000000,3000000,4000000,5000000}
  \tkzGrphDefinirCourbe[Nom=cf]<f>{2000*(x-2990)^2+500000}
  \tkzGrphDefinirCourbeInterpo[Tension=1,Nom=interpo]%
    {2990/4500000 $ 3005/500000 $ 3014/4250000}
  \tkzGrphDefinirCourbeSpline[Alt,Nom=sp]%
    {2990/300000/0.5 $ 3000/2000000/0 $ 3010/250000/0 $ 3020/4500000/0.8}
  % tracer avec des couleurs différentes
  \tkzGrphTracerCourbes{cf/red,interpo/blue,sp/orange}
\end{tkzGrph}

```



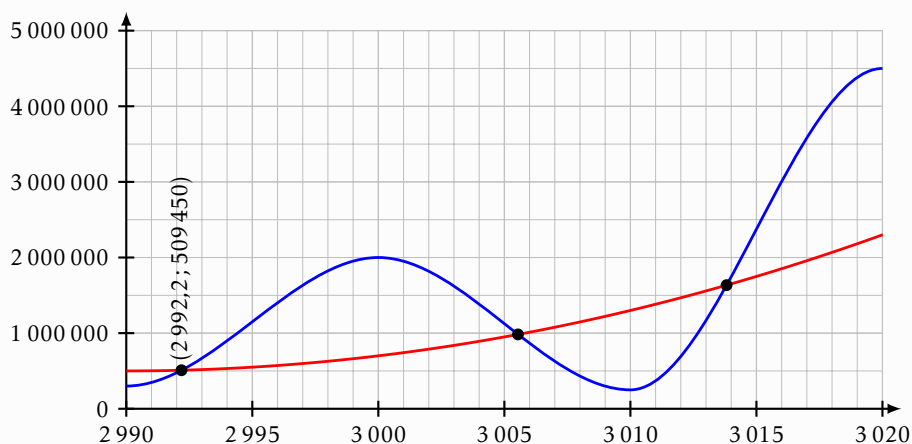
4 Compléments

4.1 Récupération de coordonnées réelles

```
\tkzGrphGetX{(nœud)}[\mavaleur] % abscisse réelle
\tkzGrphGetY{(nœud)}[\mavaleur] % ordonnée réelle
\tkzGrphGetXY{(nœud)}[\monx][\mony]
% alias FR :
\tkzGrphRecupX / \tkzGrphRecupY / \tkzGrphRecupXY
```

Dénormalise les coordonnées d'un nœud TikZ existant vers les vraies valeurs.

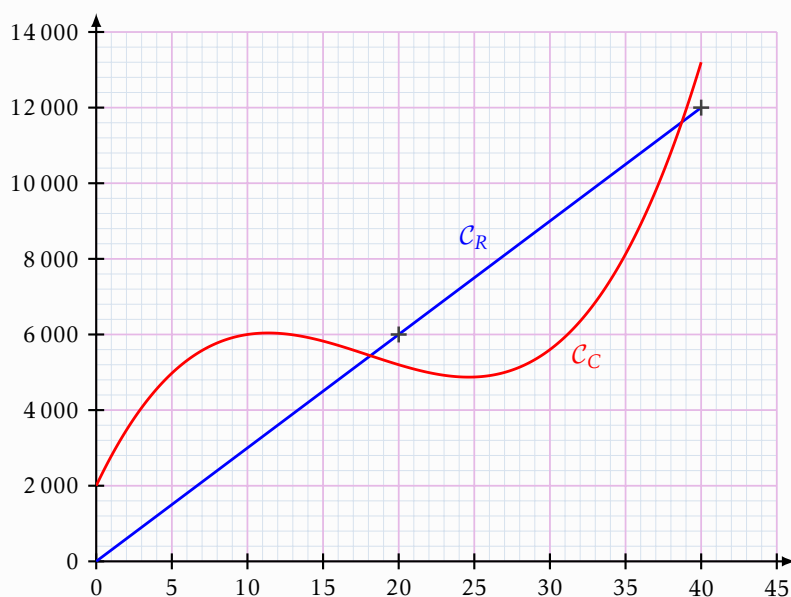
```
\begin{tkzGrph}%
  <Xmin=2990,Xmax=3020,Xgrille=5,Xgrilles=1,
  Ymin=0,Ymax=5000000,Ygrille=1000000,Ygrilles=500000,
  Largeur=10cm,Hauteur=5cm,Origx=2990>
  \tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
    {2990,2995,3000,3005,3010,3015,3020}%
    {0,1000000,2000000,3000000,4000000,5000000}
  \tkzGrphDefineCurve[name=cf]<f>{2000*(x-2990)^2+5000000}
  \tkzGrphDefineSpline[alt,name=sp]%
    {2990/3000000/0.5 $ 3000/2000000/0 $ 3010/2500000/0 $ 3020/4500000/0.8}
  \tkzGrphDrawExistingCurves{cf/red,sp/blue}
  \tkzGrphFindItsc[show,name=K]{cf}{sp}
  % récupérer et afficher les coordonnées réelles
  \tkzGrphGetXY{(K-1)}[\xun][\yun]
  \node[right,font=\small,rotate=90] at (K-1)
    {$(\ArrondirNum[1]{\xun}\,,\,\ArrondirNum[0]{\yun})$} ;
\end{tkzGrph}
```



4.2 Création de nœuds : \tkzGrphCreateNode(s)

```
% version un seul nœud
\tkzGrphCreateNode[nom]{(x,y)}
% version plusieurs nœuds
\tkzGrphCreateNode{(xA,yA) $ nomA , (xB,yB) $ nomB , ... }
% aliases FR :
\tkzGrphCreerNoeud et \tkzGrphCreerNoeuds
```

```
\begin{tkzGrph}%
<Xmin=0,Xmax=45,Xgrille=5,Xgrilles=1,%
Ymin=0,Ymax=14000,Ygrille=2000,Ygrilles=400,%
Largeur=9cm,Hauteur=7cm,Theme=seyes>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]{0,5,...,45}{0,2000,...,14000}
\tkzGrphDefineCurve[name=CR,end=40]<R>{300*x}
\tkzGrphDefineCurve[name=CC,end=40]<C>{x^3-54*x^2+840*x+2000}
\tkzGrphDrawExistingCurves{CR/blue,CC/red}
\tkzGrphCreateNodes{{(20,6000)} $ ptA , {(40,12000)} $ ptB}
\tikzset{pflpoint/.style={line width=1.25pt}}
\tkzGrphMarkPts*[style=+,size=3pt,color=darkgray]{(ptA),(ptB)}
\tkzGrphPlaceText[pos=below,color=red]{(32.5,6000)}{ $\mathcal{C}_C$ }
\tkzGrphPlaceText[pos=above,color=blue]{(25,8000)}{ $\mathcal{C}_R$ }
\end{tkzGrph}
```



4.3 Marquage de points : \tkzGrphMarkPts

```
% version normale : avec label
\tkzGrphMarkPts[clés]<police>{liste de nœuds TikZ}
% version étoilée : sans label
\tkzGrphMarkPts*[clés]{liste de nœuds TikZ}
% alias FR :
\tkzGrphMarquerPts
```

Les nœuds sont donnés en coordonnées normalisées (à faire en amont) ou via des nœuds TikZ existants. Clés : Couleur/color, Style/style (o, x, +, ggb...).

5 Analyse(s) et exploitation(s)

5.1 Intégrales : \tkzGrphDrawIntg

```
% sous une courbe (axe Ox)
\tkzGrphDrawIntg[clés]{nom path}{borne A}{borne B}
% entre deux courbes
\tkzGrphDrawIntg[clés]{nom path 1}[nom path 2]{borne A}{borne B}
% alias FR :
\tkzGrphIntegrale[clés]{...}
```

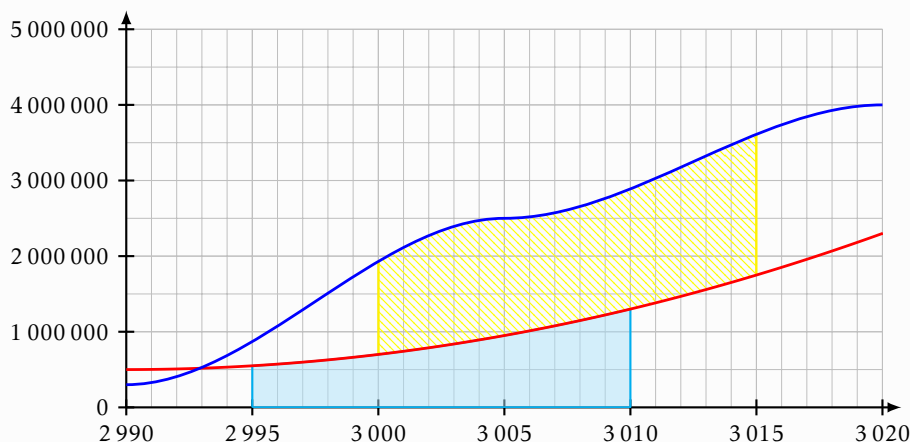
Les bornes sont des **valeurs réelles** ou des **nœuds TikZ** ((ITSC-1)). Le paramètre optionnel [nom path 2] active le mode *entre deux courbes*.

☛ Nécessite la librairie TikZ `spath3` (\usetikzlibrary{spath3}).

Les [clés] disponibles sont :

- Couleurs / colors : couleur(s) (bord/fond ou unique), gray par défaut;
- Style / style : fill (défaut) ou hachures;
- Hachures / hatch : pattern TikZ, north west lines par défaut;
- Opacite / opacity : opacité du fond, 0.5 par défaut;
- Bord / border : booléen, true par défaut, traits verticaux aux bornes;
- Bornes / bound : abs (défaut) ou noeuds.

```
\begin{tkzGrph}%
<Xmin=2990,Xmax=3020,Xgrille=5,Xgrilles=1,
Ymin=0,Ymax=5000000,Ygrille=1000000,Ygrilles=500000,
Largeur=10cm,Hauteur=5cm,Origx=2990>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
{2990,2995,3000,3005,3010,3015,3020}%
{0,1000000,2000000,3000000,4000000,5000000}
\tkzGrphDefineCurve[draw=false,name=cf]<f>{2000*(x-2990)^2+5000000}
\tkzGrphDefineSpline[alt,draw=false,name=sp]%
{2990/3000000/0.5 $ 3005/2500000/0 $ 3020/4000000/0.8}
% sous une courbe
\tkzGrphDrawIntg[colors=cyan/cyan!30]{cf}{2995}{3010}
% entre deux courbes
\tkzGrphDrawIntg[colors=yellow,style=hachures]{sp}{cf}{3000}{3015}
\tkzGrphDrawExistingCurves{cf/red,sp/blue}
\end{tkzGrph}
```



5.2 Intersections : \tkzGrphFindItsc

```
\tkzGrphFindItsc[clés]{nom path 1}{nom path 2}<nb>
% alias FR :
\tkzGrphIntersections[clés]{nom path 1}{nom path 2}<nb>
```

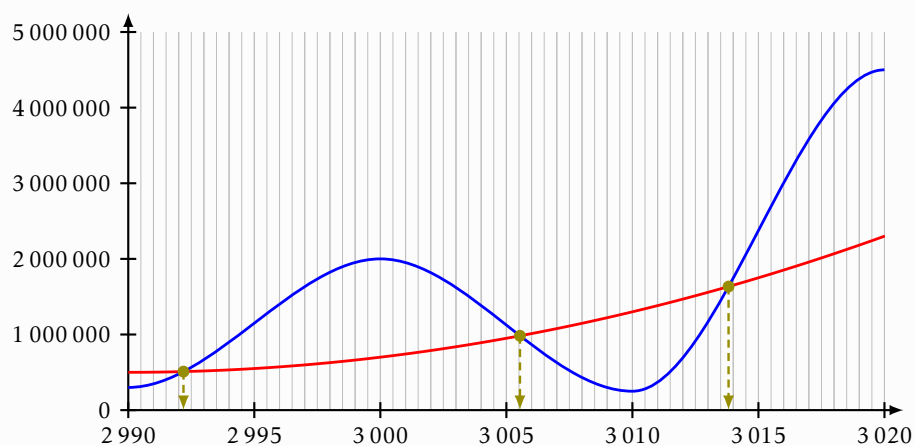
Les nœuds sont nommés S-1, S-2...(ou le préfixe choisi via `Nom/name`).

Les [clés] disponibles sont :

- `Nom / name` : préfixe des nœuds, S par défaut;
- `Couleur / color` : couleur des points et traits;
- `Aff / show` : booléen, true par défaut;
- `Traits / lines` : booléen, traits vers les axes;
- `ListeRes / reslist` : nom d'une macro pour la liste des abscisses réelles.

```
\begin{tkzGrph}<Xmin=2990,Xmax=3020,Ymin=0,Ymax=5000000,
  Largeur=10cm,Hauteur=5cm,Origx=2990>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
  {2990,2995,3000,3005,3010,3015,3020}%
  {0,1000000,2000000,3000000,4000000,5000000}
\tkzGrphDefineCurve[name=cf]<f>{2000*(x-2990)^2+5000000}
\tkzGrphDefineSpline[alt,name=sp]%
  {2990/300000/0.5 $ 3000/2000000/0 $ 3010/250000/0 $ 3020/4500000/0.8}
\tkzGrphDrawExistingCurves{cf/red,sp/blue}
\tkzGrphFindItsc[show,lines,reslist=LISTRES,color=olive]{cf}{sp}
\end{tkzGrph}
```

Solutions : `\LISTRES`



Solutions : 2992.189939992727,3005.539579948395,3013.814659920914

5.3 Images et antécédents

5.3.1 Image : \tkzGrphFindY

```
\tkzGrphFindY[clés]{nom path}{x réel}<nom nœud>  
% alias FR :  
\tkzGrphTrouverY[clés]{nom path}{x réel}<nom nœud>
```

Intersection avec la droite verticale $x = x_{\text{réel}}$. Le nœud <nom nœud> est réutilisable ensuite.

5.3.2 Images : \tkzGrphFindListY

```
\tkzGrphFindListY[clés]{nom path}{x1,x2,...}  
% alias FR :  
\tkzGrphTrouverListeY[clés]{nom path}{x1,x2,...}
```

5.3.3 Antécédents : \tkzGrphFindX

```
\tkzGrphFindX[clés]{nom path}{y réel}<nb>  
% alias FR :  
\tkzGrphTrouverX[clés]{nom path}{y réel}<nb>
```

<nb> reçoit le nombre de solutions. Clé `reslist/ListeRes` pour stocker la liste des abscisses.

Les [clés] communes à \tkzGrphFindY et \tkzGrphFindX :

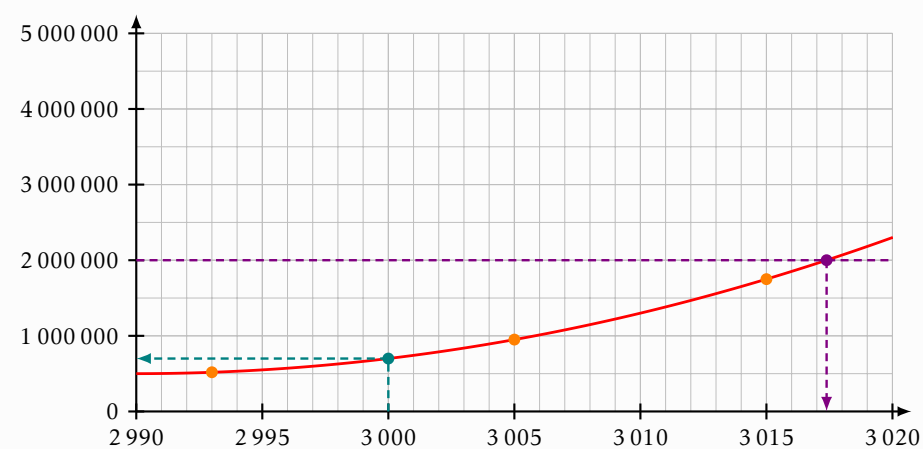
- `Couleur / color` : couleur du point et des traits;
- `Style / style` : style du point (o, x, ggb...);
- `Aff / show` : booléen, true par défaut;
- `Traits / lines` : booléen, traits vers les axes;
- `AffDroite / showline` : booléen, affiche la droite horizontale/verticale.

```

\begin{tkzGrph}%
  <Xmin=2990,Xmax=3020,Xgrille=5,Xgrilles=1,
  Ymin=0,Ymax=5000000,Ygrille=1000000,Ygrilles=500000,
  Largeur=10cm,Hauteur=5cm,Origx=2990>
  \tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
    {2990,2995,3000,3005,3010,3015,3020}%
    {0,1000000,2000000,3000000,4000000,5000000}
  \tkzGrphDefineCurve[trace,name=cf,color=red]<f>{2000*(x-2990)^2+500000}
  % image en x=3000
  \tkzGrphFindY[show,lines,color=teal]{cf}{3000}
  % images multiples
  \tkzGrphFindListY[show,color=orange]{cf}{2993,3005,3015}
  % antécédents de y=2000000
  \tkzGrphFindX%
    [show,lines,showline,color=violet,reslist=ANTS]{cf}{2000000}
\end{tkzGrph}

```

Antécédents : \ANTS



Antécédents : 3017.387449909050

5.4 Extremums : \tkzGrphFindMin/Max

```

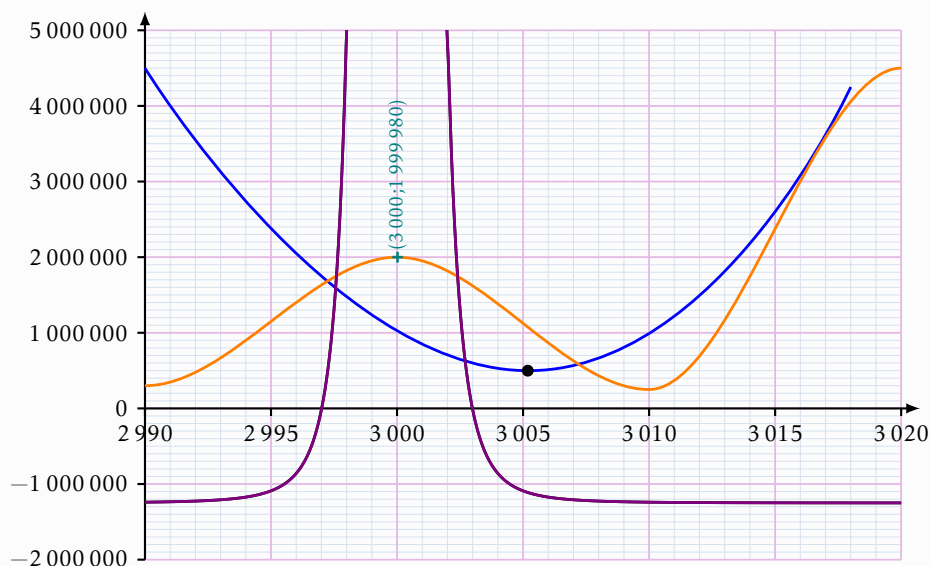
\tkzGrphFindMin[clés]{nom path}[nœud]
\tkzGrphFindMax[clés]{nom path}[nœud]

```

```

\begin{tkzGrph}%
  <xmin=2990,Xmax=3020,Xgrille=5,Xgrilles=1,%
  Ymin=-2000000,Ymax=5000000,Ygrille=1000000,Ygrilles=100000,Origx=2990,%
  Largeur=10cm,Hauteur=70mm,Theme=seyes,TailleGrad=0pt/1mm>
  \tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
  {2990,2995,3000,3005,3010,3015,3020}%
  {-2000000,-1000000,0,1000000,2000000,3000000,4000000,5000000}
  \tkzGrphDefineCurve[name=cf,trace,forbiddenvalues={3000}]%
  <f>{-1250000 + 100000000/(x-3000)^4}
  \tkzGrphDefineInterpoCurve[tension=1,name=interpo]%
  {2990/4500000 $ 3005/500000 $ 3018/4250000}
  \tkzGrphDefineSpline[alt,name=spline]%
  { 2990/300000/0.5 $ 3000/2000000/0 $ 3010/250000/0 $ 3020/4500000/0.8 }
  \tkzGrphDrawExistingCurves{interpo/blue,spline/orange,cf/violet}
  % \tkzGrphFindItsc[show,lines,reslist=LISTRES,Couleur=olive]{cf}{spline}
  %max/min
  \tkzGrphFindMin{interpo}
  \tkzGrphFindMax[start=2990,end=3010,color=teal,style=+]{spline}
  \tkzGrphGetXY{(grph-max)}[\mylocalxmax][\mylocalymax]
  \draw[teal] (grph-max) node[right,rotate=90,scale=0.75]
  ↳ {(\\ArrondirNum[0]{\mylocalxmax};\\ArrondirNum[0]<exponent-mode=fixed>{\mylocalymax})} ;
\end{tkzGrph}

```



5.5 Points d'intérêt : \tkzGrphInterestPoints

La commande `\tkzGrphInterestPoints` est un *wrapper* qui combine en une seule commande la recherche des **racines**, du **maximum global** et du **minimum global** d'une courbe définie via un name path.

Elle est compatible avec les trois types de courbes (`\tkzGrphDefineCurve`, `\tkzGrphDefineSpline`, `\tkzGrphDefineInterpoCurve`).

🔴 Les *limitations* sont les mêmes que pour `\tkzGrphFindMin/Max` :

- pas de gestion d'extremums multiples (seul le global sera traité);
- pour des courbes sans minimum visible, utiliser `Minimum=false`;
- le temps de compilation peut être plus long.

```

%dans l'environnement tkzGrph
\tkzGrphInterestPoints[clés]{nom path}
% alias FR :
\tkzGrphPointsInteret[clés]{nom path}

```

Les [clés], optionnelles, disponibles sont :

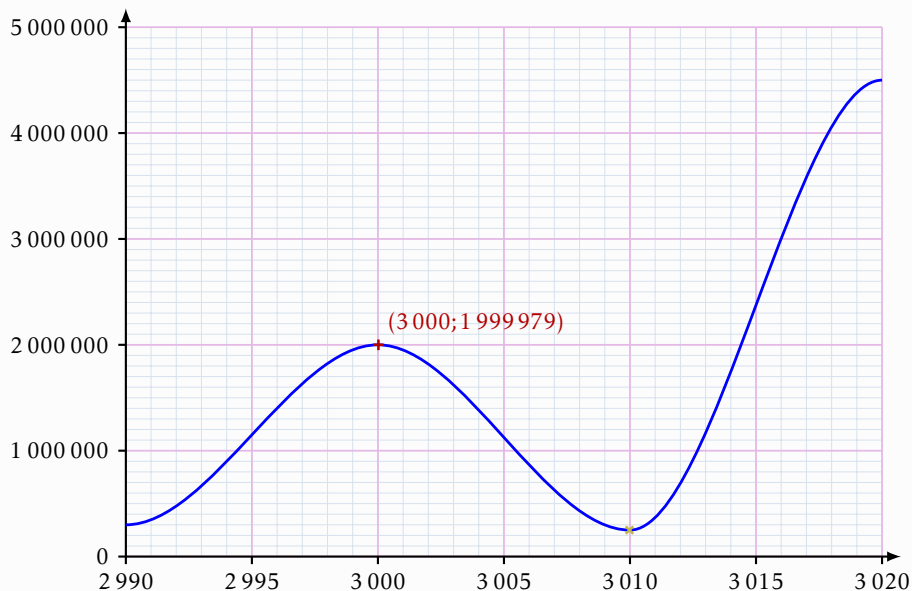
- `Couleur` / `color` : couleur globale (black par défaut);

- `CouleurRacines` / `colorroots` : couleur des racines (défaut → `Couleur`);
- `CouleurMax` / `colormax` : couleur du maximum (défaut → `Couleur`);
- `CouleurMin` / `colormin` : couleur du minimum (défaut → `Couleur`);
- `Style` / `style` : style global des points (0 par défaut);
- `StyleRacines` / `styleroots` : style des racines (défaut → `Style`);
- `StyleMax` / `stylemax` : style du maximum (défaut → `Style`);
- `StyleMin` / `stylemin` : style du minimum (défaut → `Style`);
- `Local` / `local` : booléen pour les extrema locaux, en *test* (`false` par défaut);
- `Racines` / `roots` : booléen (`true` par défaut);
- `Maximum` / `maximum` : booléen (`true` par défaut);
- `Minimum` / `minimum` : booléen (`true` par défaut);
- `NomMax` / `namemax` : nom du nœud du maximum (`grph-max` par défaut);
- `NomMin` / `namemin` : nom du nœud du minimum (`grph-min` par défaut);
- `NomRacines` / `nameroots` : préfixe des nœuds des racines (`grph-rac` par défaut);
- `Debut` / `start` : début de la recherche (fenêtre complète par défaut);
- `Fin` / `end` : fin de la recherche (fenêtre complète par défaut).

```

\begin{tkzGrph}%
<Xmin=2990,Xmax=3020,Xgrille=5,Xgrilles=1,%
Ymin=0,Ymax=5000000,Ygrille=1000000,Ygrilles=100000,Origx=2990,%
Largeur=10cm,Hauteur=70mm,Theme=seyes,TailleGrad=0pt/1mm>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
{2990,2995,3000,3005,3010,3015,3020}%
{0,1000000,2000000,3000000,4000000,5000000}
\tkzGrphDefineSpline[alt,name=spline]%
{2990/300000/0.5$3000/2000000/0$3010/250000/0$3020/4500000/0.8}
\tkzGrphDrawExistingCurves{spline/blue}
% points d'intérêt de la spline
\tkzGrphInterestPoints[%
Debut=2995,Fin=3012,%
CouleurMax=red!70!black,StyleMax=+,
CouleurMin=yellow!70!black,StyleMin=x,
NomMax=monmax,NomMin=monmin%
]{spline}
% récupération des coordonnées du maximum
\tkzGrphGetXY{(monmax)}[\xmax][\ymax]
\node[above right,font=\small,red!70!black] at (monmax)
{$(\ArrondirNum[0]{\xmax});%
\ArrondirNum[0]<exponent-mode=fixed>{\ymax})$} ;
\end{tkzGrph}

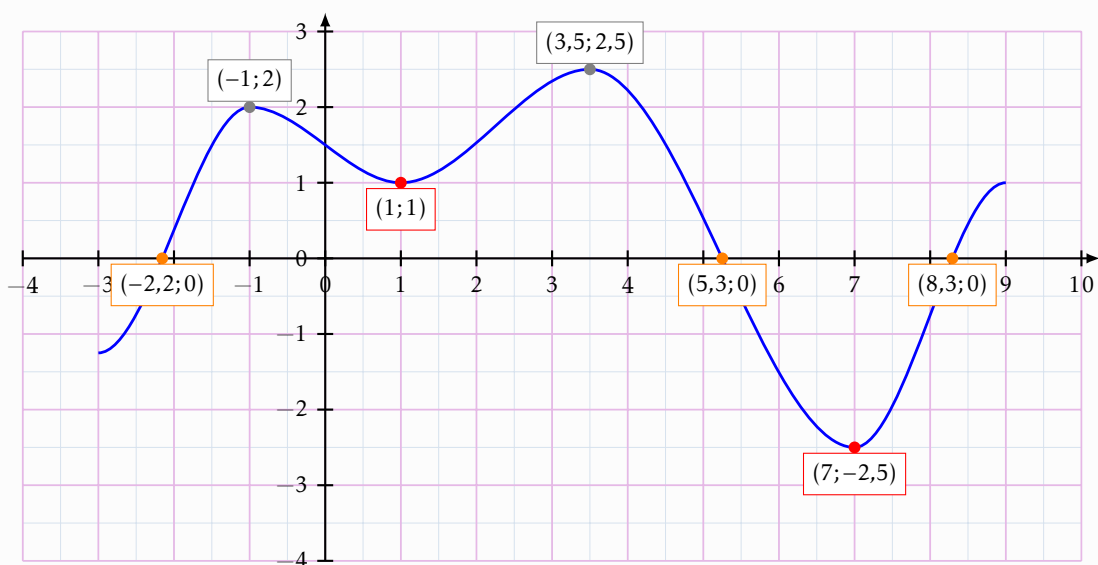
```



```

\begin{tkzGrph}%
<Xmin=-4,Xmax=10,Xgrille=1,Xgrilles=0.5,%
Ymin=-4,Ymax=3,Ygrille=1,Ygrilles=0.5,%
Largeur=14cm,Hauteur=7cm,Theme=seyes>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]{-4,-3,...,10}{-4,-3,...,3}
\tkzGrphDefineSpline[alt,name=spline,Couleur=red]%
{ -3/-1.25/0 $ -1/2/0 $ 1/1/0 $ 3.5/2.5/0 $ 7/-2.5/0 $ 9/1/0}
\tkzGrphDrawExistingCurves{spline/blue} % points d'intérêt de la spline
%\tkzGrphFindMax[Aff=true]{spline}[\pflpinommax]%
\tkzGrphPointsInteret[Local,CouleurMax=gray,CouleurMin=red,CouleurRacines=orange]{spline}
%tests
\foreach \i in {1,...,\pflstlocalmaxn}{%
\tkzGrphGetXY{(grph-max-\i)}[\tmpx][\tmpy]%
\node[draw=gray,fill=white,above=2pt,font=\small] at (grph-max-\i)
{${\ArroundirNum[1]{\tmpx};\ArroundirNum[1]{\tmpy}}$} ;%
}%
\foreach \i in {1,...,\pflstlocalminn}{%
\tkzGrphGetXY{(grph-min-\i)}[\tmpx][\tmpy]%
\node[draw=red,fill=white,below=2pt,font=\small] at (grph-min-\i)
{${\ArroundirNum[1]{\tmpx};\ArroundirNum[1]{\tmpy}}$} ;%
}%
\foreach \i in {1,...,\myantec}{%
\tkzGrphGetXY{(grph-rac-\i)}[\tmpx][\tmpy]%
\node[draw=orange,fill=white,below=2pt,font=\small] at (grph-rac-\i)
{${\ArroundirNum[1]{\tmpx};\ArroundirNum[1]{\tmpy}}$} ;%
}
\end{tkzGrph}

```



6 Nuage et régression : statistiques

6.1 Nuage de points : \tkzGrphScatterPlot

```
\tkzGrphScatterPlot[clés]{liste X}{liste Y}
% alias FR :
\tkzGrphNuage[clés]{liste X}{liste Y}
```

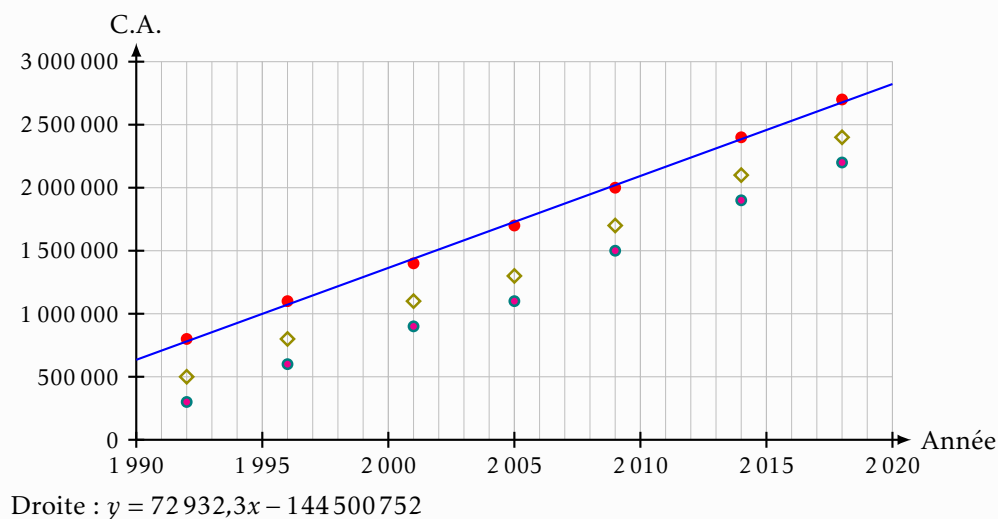
6.2 Droite de régression : \tkzGrphDrawRegLin

```
\tkzGrphDrawRegLin[clés tikz]{liste X}{liste Y}
% alias FR :
\tkzGrphRegLin[clés tikz]{liste X}{liste Y}
```

Calcule et trace la droite de régression par la méthode des moindres carrés. Les coefficients sont stockés dans $\backslash\text{pflstrega}$ (a) et $\backslash\text{pflstregb}$ (b).

```
\begin{tkzGrph}
  <Xmin=1990,Xmax=2020,Ymin=0,Ymax=3000000,Largeur=10cm,Hauteur=5cm,%
  Xgrille=5,Xgrilles=1,Ygrille=500000,Origx=1990>
  \tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]{1990,1995,2000,2005,2010,2015,2020}%
  {0,500000,1000000,1500000,2000000,2500000,3000000}
  \tkzGrphScatterPlot[color=red]{1992,1996,2001,2005,2009,2014,2018}%
  {800000,1100000,1400000,1700000,2000000,2400000,2700000}
  \tkzGrphScatterPlot[color=olive,style=do]{1992,1996,2001,2005,2009,2014,2018}%
  {500000,800000,1100000,1300000,1700000,2100000,2400000}
  \tkzGrphScatterPlot[color=teal/magenta,style=ggb]{1992,1996,2001,2005,2009,2014,2018}%
  {300000,600000,900000,1100000,1500000,1900000,2200000}
  \tkzGrphDrawRegLin[blue,thick]{1992,1996,2001,2005,2009,2014,2018}%
  {800000,1100000,1400000,1700000,2000000,2400000,2700000}
  %labels
  \tkzGrphPlaceTexte[Position=right]{(grph-axox-ee)}{Année}
  \tkzGrphPlaceTexte[Position=above]{(grph-axoy-nn)}{C.A.}
\end{tkzGrph}

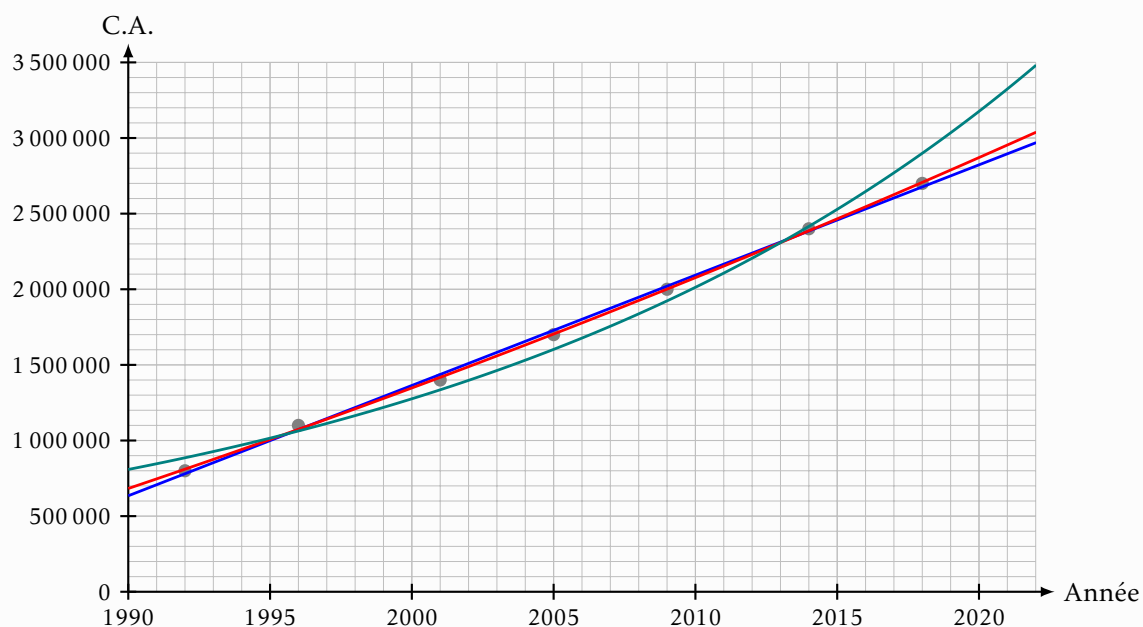
Droite :  $y = \text{ArrondirNum}[1]{\backslash\text{pflstrega}} x + \text{ArrondirNum}[0]{\text{exponent-mode=fixed}}{\backslash\text{pflstregb}}$ 
```



6.3 Régressions : \tkzGrphDrawFit

```
\tkzGrphDrawFit%
  [clés courbe]<nom courbe>{type}<arrondis>{listeX}{listeY}
% alias FR :
\tkzGrphTracerAjustement
```

```
\begin{tkzGrph}%
  <Xmin=1990,Xmax=2022,Ymin=0,Ymax=3500000,
  Largeur=12cm,Hauteur=7cm,
  Xgrille=5,Xgrilles=1,
  Ygrille=500000,Ygrilles=100000,
  Origx=1990>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small,format=year/num]%
  {1990,1995,2000,2005,2010,2015,2020}%
  {0,500000,1000000,1500000,2000000,2500000,3000000,3500000}
%data
\def\LX{1992,1996,2001,2005,2009,2014,2018}
\def\LY{800000,1100000,1400000,1700000,2000000,2400000,2700000}
% nuage de points
\tkzGrphScatterPlot[color=gray,style=o,size=2pt]{\LX}{\LY}
% ajustement linéaire
\tkzGrphDrawFit[color=blue,trace]<ajlin>{lin}{\LX}{\LY}
% ajustement quadratique
\tkzGrphDrawFit[color=red,trace]<ajquad>{quad}{\LX}{\LY}
% ajustement exponentiel alt
\tkzGrphDrawFit[color=teal,trace]<ajexp>{expalt}{\LX}{\LY}
% label axe
\tkzGrphPlaceTexte[Position=right]{(grph-axox-ee)}{Année}
\tkzGrphPlaceTexte[Position=above]{(grph-axoy-nn)}{C.A.}
\end{tkzGrph}
```



7 Compléments sur les axes

7.1 Valeurs manuelles sur les axes

Ces commandes permettent d'ajouter manuellement des valeurs ou des labels qualitatifs sur les axes, en complément de `\tkzGrphDrawGridAxis`.

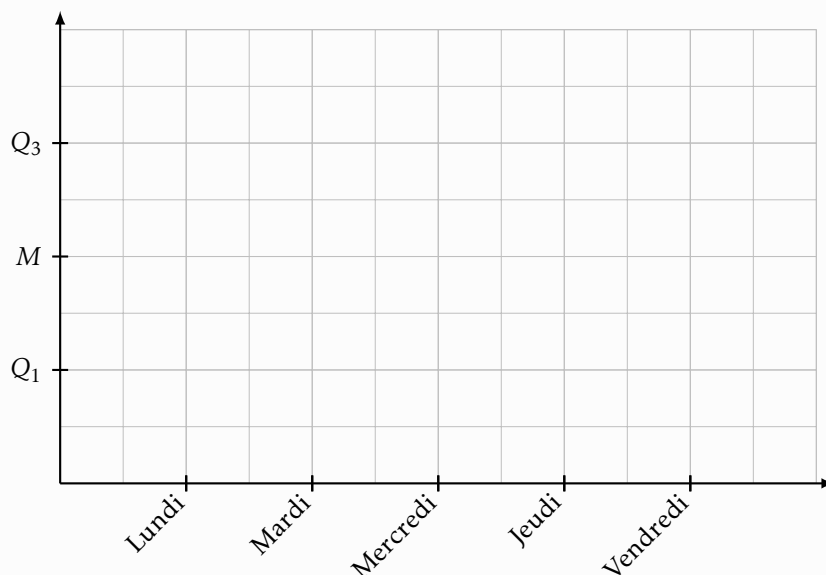
```
% valeurs numériques formatées
\tkzGrphAddValX[clés]{liste valeurs X réelles}{liste labels}
\tkzGrphAddValY[clés]{liste valeurs Y réelles}{liste labels}
% alias FR :
\tkzGrphAjouterValeursX / \tkzGrphAjouterValeursY

% labels qualitatifs (avec rotation possible)
\tkzGrphAddLabelsX[clés]{liste valeurs X réelles}{liste labels}
\tkzGrphAddLabelsY[clés]{liste valeurs Y réelles}{liste labels}
% alias FR :
\tkzGrphAjouterLabelsX / \tkzGrphAjouterLabelsY
```

Les [clés] disponibles sont :

- Traits / `thicks` : booléen, affiche les tirets de graduation ;
- Rotation / `angle` : angle de rotation des labels (0 par défaut) ;
- Police / `font` : police des labels.

```
\begin{tkzGrph}<Xmin=0,Xmax=6,Ymin=0,Ymax=4,
  Largeur=10cm,Hauteur=6cm>
\tkzGrphDrawGridAxis[Grads=false,enlarge=2.5mm]{}{}
% valeurs numériques personnalisées
\tkzGrphAddValX[Traits]{1,2,3,4,5}{,,,}
\tkzGrphAddValY[Traits]{1,2,3}{$Q_1$, $M$, $Q_3$}
% labels qualitatifs avec rotation
\tkzGrphAddLabelsX[Rotation=45]{1,2,3,4,5}%
  {Lundi,Mardi,Mercredi,Jeudi,Vendredi}
\end{tkzGrph}
```



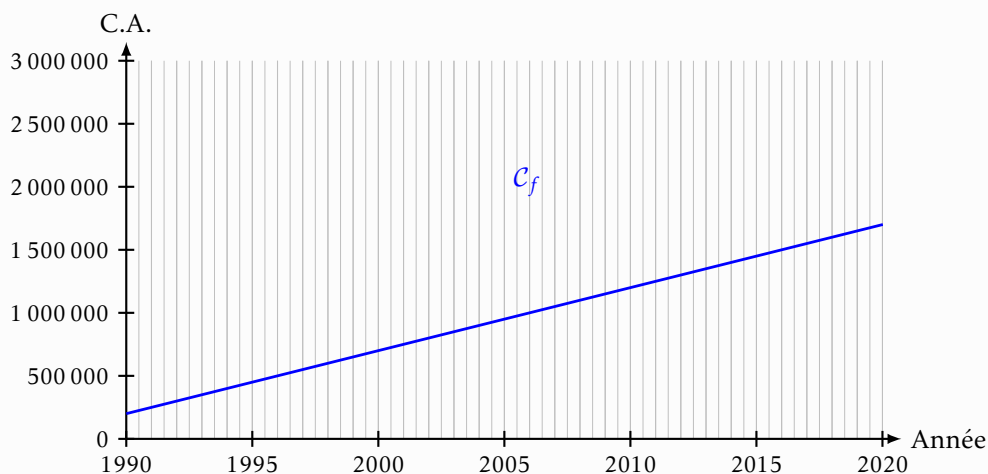
7.2 Placer un texte : \tkzGrphPlaceText

```
% coordonnées réelles
\tkzGrphPlaceText[clés]({{x},{y}}){texte}
% nœud TikZ existant
\tkzGrphPlaceText[clés]((nom-noeud)){texte}
% alias FR :
\tkzGrphPlaceTexte[clés]{...}{texte}
```

Les coordonnées réelles doivent être protégées avec $\{x\}, \{y\}$ pour éviter les conflits de parsing. Les nœuds existants (grph-axox-ee, grph-axoy-nn...) sont utilisables directement.

Les [clés] disponibles sont : Couleur/color, Police/font, Position/anchor, DecalX/shiftx, DecalY/shifty.

```
\begin{tkzGrph}<Xmin=1990,Xmax=2020,Ymin=0,Ymax=3000000,
  Largeur=10cm,Hauteur=5cm,Origx=1990>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small,Format=annee/num]%
  {1990,1995,2000,2005,2010,2015,2020}%
  {0,500000,1000000,1500000,2000000,2500000,3000000}
\tkzGrphDefineCurve[trace,name=cf,color=blue]<f>{50000*(x-1990)+200000}
% label sur un nœud existant
\tkzGrphPlaceTexte[Position=right]((grph-axox-ee)){Année}
\tkzGrphPlaceTexte[Position=above]((grph-axoy-nn)){C.A.}
% label en coordonnées réelles
\tkzGrphPlaceTexte[Position=above right,color=blue]%
  ({2005},{1850000}){ $C_f$ }
\end{tkzGrph}
```



8 Annotations graphiques

8.1 Label sur un path : `\tkzGrphLabelPath`

Permet de placer un label directement sur un `name path` à une position donnée en **longueur d'arc** ($t \in [0, 1]$).

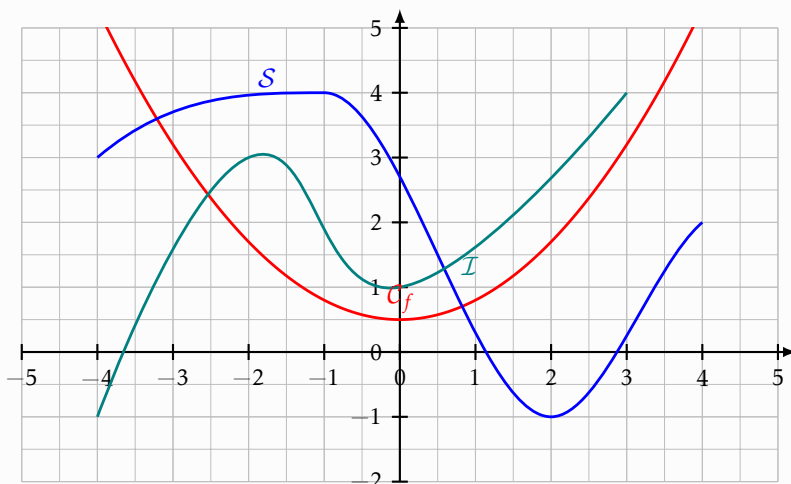
● La position t est mesurée en longueur d'arc le long du path — $t=0.5$ correspond au milieu *géométrique* du tracé, pas au milieu des abscisses.

```
\tkzGrphLabelPath[clés]<options tikz>\{nom path\}{label}
% alias FR :
\tkzGrphEtiquettePath[clés]<options tikz>\{nom path\}{label}
```

Les [clés] disponibles sont :

- t : position en longueur d'arc (0.5 par défaut);
- Couleur / color : couleur du label (black par défaut);
- Police / font : police du label;
- Position / pos : positionnement du label (above par défaut);
- DecalX / shiftx et DecalY / shifty : décalages en pt.

```
\begin{tkzGrph}<Xmin=-5,Xmax=5,Ymin=-2,Ymax=5,
  Largeur=10cm,Hauteur=6cm>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
  {-5,-4,...,5}{-2,-1,...,5}
\tkzGrphDefineCurve[trace,name=cf,color=red]<f>{0.3*x^2+0.5}
\tkzGrphDefineSpline[trace,alt,name=sp,color=blue]%
  {-4/3/1 $ -1/4/0 $ 2/-1/0 $ 4/2/1}
\tkzGrphDefineInterpoCurve[trace,name=interpo,color=teal,tension=0.8]%
  {-4/-1 $ -2/3 $ 0/1 $ 3/4}
% labels au milieu (t=0.5 par défaut)
\tkzGrphLabelPath[color=red,pos=above]{cf}{\mathcal{C}_f}
% à t=0.25
\tkzGrphLabelPath[color=blue,pos=above,t=0.25]{sp}{\mathcal{S}}
% à t=0.75
\tkzGrphLabelPath[color=teal,pos=below,t=0.75]{interpo}{\mathcal{I}}
\end{tkzGrph}
```



8.2 Point sur un path : `\tkzGrphMarkOnPath`

Permet de placer et marquer un point sur un `name path`, à une position en longueur d'arc ou à une abscisse réelle donnée.

```

\tkzGrphMarkOnPath[clés]{nom path}
% alias FR :
\tkzGrphMarquerSurPath[clés]{nom path}

```

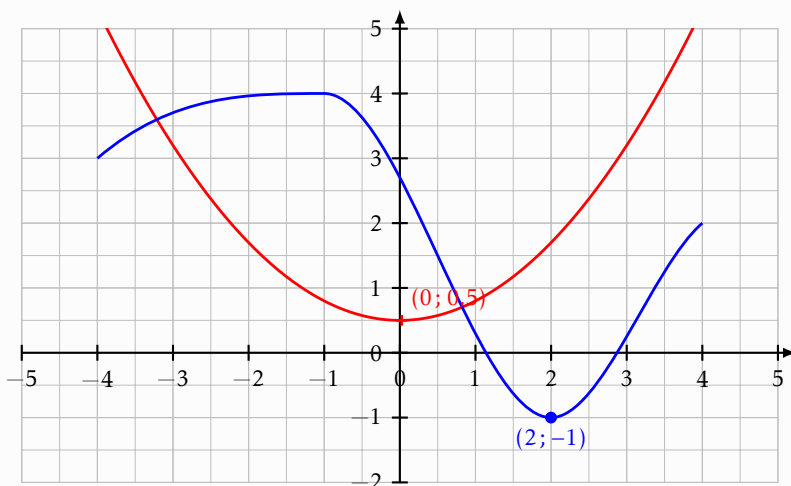
Les [clés] disponibles sont :

- `t` : position en longueur d'arc (0.5 par défaut) — si `x` est renseigné, `t` est ignoré;
- `x` : abscisse réelle — le point est l'intersection avec la droite verticale $x = x_{\text{réel}}$;
- `Couleur` / `color` : couleur du point;
- `Style` / `style` : style du point (o, x, +, ggb...);
- `Nom` / `name` : nom du nœud créé (grph-mop par défaut);
- `Aff` / `show` : booléen, true par défaut.

```

\begin{tkzGrph}<Xmin=-5,Xmax=5,Ymin=-2,Ymax=5,
  Largeur=10cm,Hauteur=6cm>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
  {-5,-4,...,5}{-2,-1,...,5}
\tkzGrphDefineCurve[trace,name=cf,color=red]<f>{0.3*x^2+0.5}
\tkzGrphDefineSpline[trace,alt,name=sp,color=blue]%
  {-4/3/1 $ -1/4/0 $ 2/-1/0 $ 4/2/1}
% mode t : milieu du path
\tkzGrphMarkOnPath[t=0.5,color=red,style=+,name=ptA]{cf}
% mode x : abscisse réelle
\tkzGrphMarkOnPath[x=2,color=blue,style=ggb,name=ptB]{sp}
% récupérer les coordonnées réelles
\tkzGrphGetXY{(ptA)}{\xa}{\ya}
\tkzGrphGetXY{(ptB)}{\xb}{\yb}
\node[above right,font=\small,red] at (ptA)
  {$(\ArroundNum[1]{\xa}\,,\,\ArroundNum[1]{\ya})$} ;
\node[below,font=\small,blue] at (ptB)
  {$(\ArroundNum[1]{\xb}\,,\,\ArroundNum[1]{\yb})$} ;
\end{tkzGrph}

```



8.3 Intervalle sur un axe : \tkzGrphIntervalX/Y

Permet de mettre en évidence un intervalle sur l'axe Ox ou Oy .

```

\tkzGrphIntervalX[clés]{borne a}{borne b}
\tkzGrphIntervalY[clés]{borne a}{borne b}
% alias FR :
\tkzGrphIntervalleSurOx / \tkzGrphIntervalleSurOy

```

Les bornes sont des **valeurs réelles**. Les [clés] disponibles sont :

- Couleur / color : couleur (black par défaut);
- Style / style : segment (défaut) ou arrow/fleche;
- Epaisseur / thick : fin/thin, standard/normal (défaut), epais/thick.

```
\begin{tkzGrph}<Xmin=-5,Xmax=5,Ymin=-2,Ymax=5,
  Largeur=10cm,Hauteur=6cm>
\tkzGrphDrawGridAxis[enlarge=2.5mm,font=\small]%
  {-5,-4,...,5}{-2,-1,...,5}
\tkzGrphDefineCurve[trace,name=cf,color=red]<f><0.3*x^2+0.5>
% intervalle sur Ox - style segment épais
\tkzGrphIntervalX[color=blue,thick=epais]{-2}{3}
% intervalle sur Ox - style flèche
\tkzGrphIntervalX[color=orange,style=arrow]{1}{4}
% intervalle sur Oy - fin
\tkzGrphIntervalY[color=teal,thick=fin]{1}{4}
\end{tkzGrph}
```

