1 Barycentre d'un système de points matériels

1.1 Barycentre de deux points

On définit le barycentre de deux points A et B du plan affectés des coefficients de pondération a et b (avec la somme $a+b$ non nulle) comme l'unique point G vérifiant la relation vectorielle

$$\boxed{a\,\vec{GA} + b\,\vec{GB} = \vec{0}}$$

Relation de Chasles

En utilisant la relation de Chasles, on peut écrire:

$$a\,\vec{GA} + b\,(\vec{GA} + \vec{AB}) = \vec{0}$$

soit

$$(a+b)\,\vec{GA} + b\,\vec{AB} = \vec{0}$$

ou encore

$$(a+b)\,\vec{AG} = b\,\vec{AB}$$

Comme $a+b$ est non nul

$$\boxed{\vec{AG} = \frac{b}{a+b}\vec{AB}}$$

On note $G = \text{bar}\{(A,a),(B,b)\}$

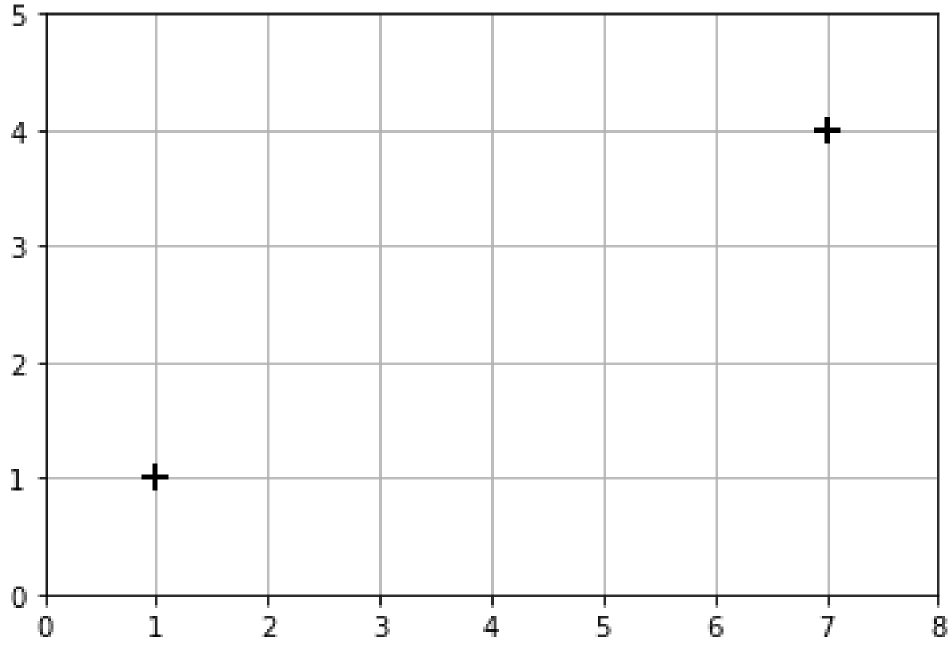
Remarque

Le point G appartient à la droite (AB).

Exemple

Situer sur la figure ci-dessous le barycentre G des points $A(1,1)$ et $B(7,4)$ affectés des coefficients $a=5$ et $b=10$.

[4]:
```
import matplotlib.pyplot as plt
plt.plot([1,7],[1,4],'+k',ms=10,mew=2)
plt.xlim([0,8])
plt.ylim([0,5])
plt.grid()
```



1.2 Barycentre d'un ensemble de points matériels

Soit $\{M_i\}$ un ensemble de N points du plan de coordonnées x_i, y_i.

Soit $\{\alpha_i\}$ un ensemble de N scalaires (de somme non nulle).

Le barycentre des points $\{M_i\}$ affectés des coefficients $\{\alpha_i\}$ est l'unique point G vérifiant la relation vetorielle:

$$\boxed{\sum_{i=1...N} \alpha_i G\vec{M}_i = \vec{0}}$$

1.3 Coordonnées du barycentre

Principe:

Il suffit d'introduit l'origine O du repère dans la relation de Chasles:

Cas du barycentre de deux points

$$a\,\vec{GA} + b\,\vec{GB} = \vec{0}$$

En introduisant l'origine O du repère:

$$a\,(\vec{GO} + \vec{OA}) + b\,(\vec{GO} + \vec{OB}) = \vec{0}$$

soit

$$(a+b)\,\vec{GO} + a\,\vec{OA} + b\,\vec{OB} = \vec{0}$$

soit

$$(a+b)\,\vec{OG} = a\,\vec{OA} + b\,\vec{OB}$$

soit

$$\boxed{\vec{OG} = \frac{a\,\vec{OA} + b\,\vec{OB}}{a+b}}$$

Les coordonnées x_G et y_G sont donc

$$x_G = \frac{ax_A + bx_B}{a+b}$$

et

$$y_G = \frac{ay_A + by_B}{a+b}$$

Cas du barycentre de N points

$$\sum_{i=1...N} \alpha_i \vec{GM_i} = \vec{0}$$

Puis

$$\sum_{i=1...N} \alpha_i (\vec{GO} + \vec{OM_i}) = \vec{0}$$

soit

$$\sum_{i=1...N} \alpha_i \vec{GO} + \sum_{i=1...N} \alpha_i \vec{OM_i} = \vec{0}$$

En remarquant que le terme $\vec{GO}$ peut être sortie de la somme:

$$\left(\sum_{i=1...N} \alpha_i\right) \vec{GO} + \sum_{i=1...N} \alpha_i \vec{OM_i} = \vec{0}$$

en changeant le signe:

$$\left(\sum_{i=1...N} \alpha_i\right) \vec{OG} = \sum_{i=1...N} \alpha_i \vec{OM_i}$$

soit

$$\boxed{\vec{OG} = \frac{\sum_{i=1...N} \alpha_i \vec{OM_i}}{\sum_{i=1...N} \alpha_i}}$$

Les coordonnées x_G, y_G du barycentre sont donc

$$x_G = \frac{\alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_N x_N}{\sum_{i=1...N} \alpha_i} = \frac{\sum_{i=1...N} \alpha_i x_i}{\sum_{i=1...N} \alpha_i}$$

$$y_G = \frac{\sum_{i=1...N} \alpha_i y_i}{\sum_{i=1...N} \alpha_i}$$

1.4 Application calcul des coordonnées du barycentre en Python

Ecrire une fonction `barycentre(xi,yi,ai)` qui, à partir des deux listes Python de coordonnées *xi*, *yi* et de la liste Python des coefficients *ai* renvoie le *tuple* (xg,yg) des coordonnées du barycentre des points $M_i(x_i, y_i)$ affectés des coefficients a_i.

On s'aidera de la trame ci-dessous en remarquant que l'algorithme nécessite de calculer la somme de trois listes de même nombre d'éléments.

Remarque: les listes *xi*, *yi* et *ai* sont supposées contenir un même nombre de flottants et la somme des valeurs numériques de la liste *ai* est supposée être non nulle.

[16]:
```
def barycentre(xi, yi, ai): # à compléter
    ''' renvoie les coordoonées (xg,yg) du barycentre
    des points de coordonnées (xi,yi)
    affectés des coefficients  ai '''
    N = ... # nombre de points
    xg, yg, s = 0, 0, 0 # mise à zéro
    for k in range(N):
        xg = xg + ...
        yg = yg + ...
        s  = s  + ...
    return xg/s, yg/s
```

Application

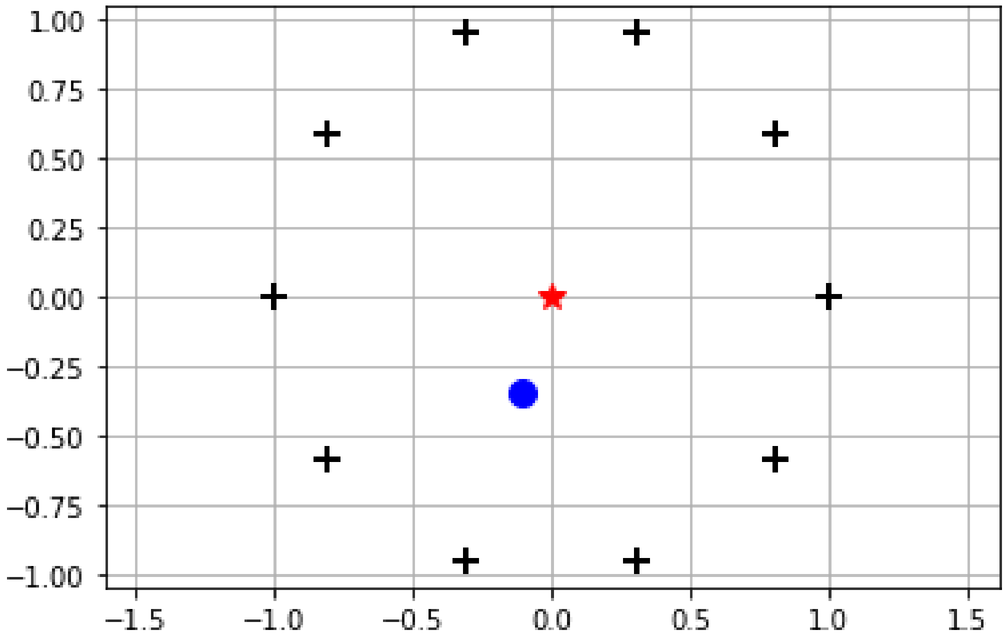
Déterminer les coordonnées des barycentres des points de coordonnées xj et yj: - d'une part lorsqu'ils sont affectés tous du même coefficient 1 (on parle d'isobarycentre) - d'autre part lorsqu'ils sont affectés des coefficients *ai=alphaj*

[22]:
```
## Validation (à compléter)
import numpy as np
Nb = 10
thetaj = np.array([k/Nb*360 for k in range(Nb)]) # angles en degrés
xj =  np.cos(thetaj*np.pi/180) # conversion en radians
yj =  np.sin(thetaj*np.pi/180) # conversion en radians
aj1 = [1 for k in range(Nb)] # coefficients tous égaux

xg1,yg1 = ...# premier barycentre
xg2,yg2 = ...# second barycentre

plt.plot(xj,yj,'+k',ms=10,mew=2) # affichage des points
plt.plot(xg1,yg1,'*r',ms=10)      # affichage du 1er barycentre
plt.plot(xg2,yg2,'ob',ms=10)      # affichage du second barycentre
plt.grid()
plt.axis('equal')
```

[22]:
```
(-1.1, 1.1, -1.046162167924669, 1.046162167924669)
```



[23]:
```
## Remarque : méthode alternative utilisant la fonction sum()
##            et la syntaxe de Numpy
def bar(xi,yi,ai):
    s = np.sum(ai)
    xg = np.sum(ai*xi)/s
    yg = np.sum(ai*yi)/s
    return xg,yg
```