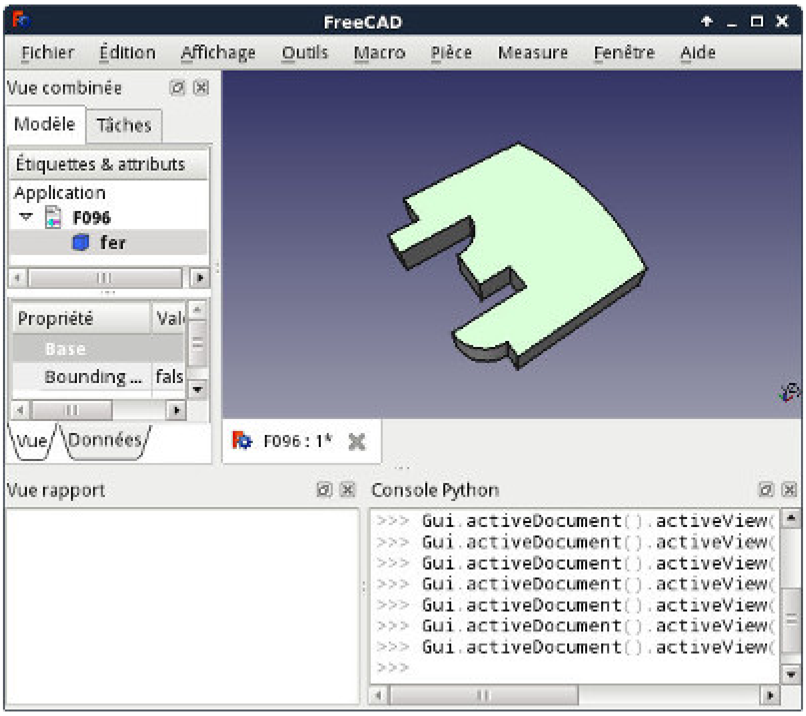
Arc de cercle pour FreeCAD

Mourad Arnout

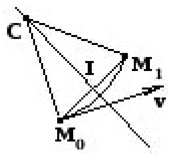
17 mai 2016

Résumé

Cet article fait partie d'un projet d'atelier menuiserie pour FreeCAD. La syntaxe de ce logiciel pour concevoir une section comprenant des rainures, des congés et des languettes est effroyablement compliquée. Les concepteurs de FreeCAD ont dû avoir du Java au biberon. Dans cet article on va définir un arc par un point de départ, un point d'arrivée et le vecteur tangent au premier point.



1 Coordonnées du centre de l'arc de cercle



Il s'agit de tracer un arc de cercle partant de M_0, arrivant en M_1 et tangent à $\vec{v}$ en M_0. Dans FreeCAD le vecteur $\vec{v}$ sera défini par son angle a avec l'axe des abscisses.

Soit C le centre de ce cercle et I le milieu du segment $[M_0M_1]$.

Un point $M(x, y)$ appartient à la droite (M_0C) si et seulement si les vecteurs $\overrightarrow{v}(\cos a, \sin a)$ et $\overrightarrow{M_0C}(x - x_0, y - y_0)$ sont orthogonaux :

$$(x - x_0)\cos a + (y - y_0)\sin a = 0 \tag{1}$$

Le centre C appartient aussi à la médiatrice de $[M_0, M_1]$. On en fait autant avec $\overrightarrow{M_0M_1}$ et $\overrightarrow{IM}$ qui sont orthogonaux :

$$(x_1 - x_0)(x - \frac{x_0 + x_1}{2}) + (y_1 - y_0)(y - \frac{y_0 + y_1}{2}) \tag{2}$$

On aurait pu aussi écrire que $MM_0^2 = MM_1^2$.

On va maintenant arranger un peu nos affaires.

$$\cos a.x + \sin a.y = x_0 \cos a + y_0 \sin a \tag{3}$$

$$(x_1 - x_0)x + (y_1 - y_0)y = \frac{x_1^2 - x_0^2 + y_1^2 - y_0^2}{2} \tag{4}$$

Il ne reste plus qu'à résoudre le système de deux équations à deux inconnues x et y. Un de mes anciens profs de physique prétendait "il n'y a qu'une solution, c'est la bonne". Ici la bonne solution c'est celle des déterminants que je rappellerai à la fin de cet article. On obtient aussitôt : u

$$x = \frac{(y_1 - y_0)(x_0 \cos a + y_0 \sin a) - \frac{1}{2}(x_1^2 - x_0^2 + y_1^2 - y_0^2)\sin a}{(y_1 - y_0)\cos a - (x_1 - x_0)\sin a} \tag{5}$$

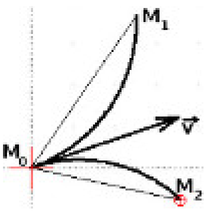
$$y = \frac{\frac{1}{2}(x_1^2 - x_0^2 + y_1^2 - y_0^2)\cos a - (x_1 - x_0)(x_0 \cos a + y_0 \sin a)}{(y_1 - y_0)\cos a - (x_1 - x_0)\sin a} \tag{6}$$

D'où le code python ci-dessous avec les notations $M_0(x0, y0)$ $M_1(x, y)$ et $C(X, Y)$:

```
p = x*x + y*y - x0*x0 - y0*y0
q = x0*cos(a) + y0*sin(a)
d = (x - x0)*sin(a) - (y - y0)*cos(a)
X = (.5*sin(a)*p - (y-y0)*q) / d # center of arc
Y = ((x-x0)*q -.5*cos(a)*p) / d
r = hypot(X - x, Y - y)
a = degrees(copysign(acos((x0 - X)/r), y0 - Y))
b = degrees(copysign(acos((x - X)/r), y - Y))
e = Part.makeCircle(r, fv(X, Y, 0), fv(0, 0, 1), a, b)
```

2 Orientation du plan

Le diable se cache dans les détails, et en math le diable c'est souvent un cauchemar d'orientation. FreeCAD trace ces arcs toujours dans le sens trigonométrique. Entre 30 ° et 45 ° il fait un petit arc, mais entre 45 ° et 30 ° il fait le grand tour.



Il faut donc bien lui indiquer l'angle de départ et l'angle d'arrivée. Voyez le petit dessin. Pour l'arc M_0M_1 il faut lui donner en premier l'angle en M_0 alors que pour l'arc M_0M_2 il faut donner l'angle à l'arrivée en premier. La solution miracle est le produit vectoriel $\vec{v} \wedge \overrightarrow{M_0M_1}$. S'il est positif le sens est trigonométrique, sinon il est rétrograde.

Dans notre cas simple :

$$\vec{v} \wedge \overrightarrow{M_0M_1} = (0, 0, (y_1 - y_0)\cos a - (x_1 - x_0)\sin a) \tag{7}$$

Finalement le code python de la fonction arcTo est :

```
def arcTo(self, alpha, x, y):
    """ a is the start angle of the arc from the current point
    (x, y) is the end point
    draw an arc from the current point to (x, y)
    update the current point
    """
    a = radians(alpha)
    x0, y0 = self.x, self.y # start point
    self.x, self.y = x, y # update current start point
    p = x*x + y*y - x0*x0 - y0*y0
    q = x0*cos(a) + y0*sin(a)
    d = (x - x0)*sin(a) - (y - y0)*cos(a)
    X = (.5*sin(a)*p - (y-y0)*q) / d # center of arc
    Y = ((x-x0)*q -.5*cos(a)*p) / d
    r = hypot(X - x, Y - y)
    g = (y - y0)*cos(a) - (x - x0)*sin(a)
    a = degrees(copysign(acos((x0 - X)/r), y0 - Y))
    b = degrees(copysign(acos((x - X)/r), y - Y))
    if g > 0:
        e = Part.makeCircle(r, fv(X, Y, 0), fv(0, 0, 1), a, b)
    else:
        e = Part.makeCircle(r, fv(X, Y, 0), fv(0, 0, 1), b, a)
    self.addEdge(e)
```

3 Rappels

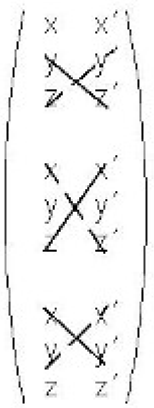
3.1 Système linéaire de deux équations à deux inconnues

Si $ab' - a' \neq 0$: $\left\{\begin{array}{ll} ax + by & = c \\ a'x + b'y & = c' \end{array}\right. \iff \left\{\begin{array}{ll} x & = \dfrac{b'c - bc'}{ab' - a'b} \\ y & = \dfrac{ac' - a'c}{ab' - a'b} \end{array}\right.$

On se souviendra du schema opératoire ci-dessous :

3.2 Produit vectoriel

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \wedge \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} yz' - zy' \\ zx' - xz' \\ xy' - yx' \end{pmatrix}$$



Le produit vectoriel $\vec{u} \wedge \vec{v}$ est un vecteur $\vec{w}$:

1. Orthogonal à $\vec{u}$ et à $\vec{v}$.
2. Le trièdre $(\vec{u}, \vec{v}, \vec{w})$ est direct si et seulement si l'angle $(\vec{u}, \vec{v})$ est positif. *C'est cette propriété qu'on utilise au paragraphe 2*
3. Sa norme $||\vec{w}||$ est l'aire du parallélogramme de côtés $\vec{u}$ et $\vec{v}$