Exercice 4 ♣ Simulation du mouvement d'une prothèse de main (25 min.)

La figure 3.25 montre une prothèse de main motorisée. La mise au point de ce type de prothèse nécessite de longues phases d'optimisation pour s'adapter au mieux aux mouvements naturels et aux besoins des personnes. L'exercice vise à simuler le mouvement d'ouverture et de fermeture de la pince.

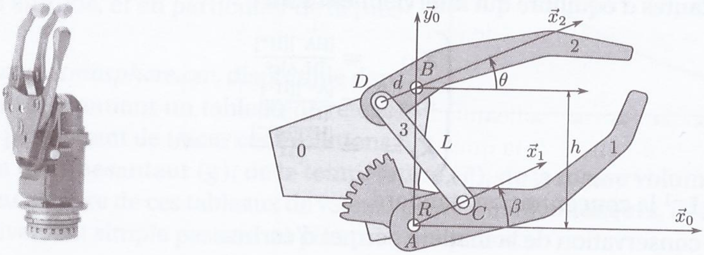

Figure 3.25. Prothèse de main et modélisation du mouvement des doigts.

Un moteur commande l'angle β. La loi de fermeture de la pince dépend des angles θ et β des doigts, dont les mouvements sont liés par la barre CD. En exprimant la contrainte de distance imposée par la barre, il est assez simple d'obtenir l'équation liant θ et β :

$$\|\vec{DC}\| = L \Longleftrightarrow (R\cos\beta + d\,\cos\theta)^2 + (R\sin\beta - h + d\,\sin\theta)^2 - L^2 = 0.$$

En résolvant cette équation non linéaire, il est possible de déduire pour tout angle β imposé par le moteur à la pince 1, l'angle θ de la pince 2.

En appliquant des opérations de rotation et de translation des polylignes représentant les pinces, il est alors possible de simuler le mouvement.

Données : $h = 40\,\text{mm}$, $L = 44\,\text{mm}$, $R = 15\,\text{mm}$ et $d = 10\,\text{mm}$.

1) Proposer une fonction `resolution()` basée sur la méthode de Newton, permettant de résoudre l'équation et renvoyant θ pour toute valeur de β donnée comme argument.
2) Proposer un algorithme permettant de définir un tableau d'environ 10 valeurs pour β entre $-0{,}5$ et $0{,}3$ rad, et de calculer le tableau des angles θ correspondants.

Une rotation d'un angle θ dans le plan peut être représentée par une matrice 2×2 et tout vecteur V peut subir cette rotation en le multipliant par la matrice sous la forme :

$$\vec{V}' = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} V_x \\ V_y \end{bmatrix}.$$

Les lignes ci-dessous permettent de donner une représentation simplifiée des pinces et de la bielle 3. En multipliant la matrice de rotation par une des polylignes `pince1` ou `pince2`, celle-ci subit la rotation voulue (autour de son axe de liaison au bâti).

Python

```
pince1=array([[0,10,20,40,50],
       [0,0, -5,0,10]])
pince2=array([[-10,0,10,40,50],
       [0, 0, 0,-5,-15]])
```

Scilab

```
// Polylignes
pince1=[0,10,20,40,50;
        0,0, -5,0,10]
pince2=[-10,0,10,40,50;
         0, 0, 0,-5,-15]
```

3) Proposer un programme permettant de visualiser le mouvement des pinces pour les 10 positions calculées à la question précédente.