Algorithme de dichotomie

Dans le cas où les hypothèses nécessaire sont vérifiées, le théorème des valeurs intermédiaires nous assure l'existence d'au moins une solution à une équation du type $f(c) = k$. Cependant, il ne donne aucune information sur la valeur de cette solution. Il existe des algorithmes permettant néanmoins d'obtenir au moins une valeur approchée d'une solution de cette équation.

L'un d'eux est l'**algorithme de dichotomie** : considérons une fonction f continue sur un intervalle $[a, b]$ et supposons que $f(a) \leq k \leq f(b)$. On cherche à obtenir une valeur approchée de c. Le principe consiste, à chaque étape, à « couper »notre intervalle en deux, et à déterminer dans quelle partie se situe notre solution. À chaque étape, l'intervalle étudié est deux fois plus petit ; on arrête notre algorithme lorsque l'intervalle est assez petit.

Algorithme 1 Algorithme de dichotomie

fonction DICHOTOMIE(a, b, k, f, p) :
    Tant que $b - a \geq p$, **faire** :
        $m \leftarrow \dfrac{a+b}{2}$
        Si $f(m) \leq k$ **alors**
            $a \leftarrow m$
        Sinon
            $b \leftarrow m$
        Fin si
    Fin tant que
    renvoyer a
Fin fonction

Dans le cas où $f(a) \geqslant k \geqslant f(b)$, il suffit de changer le sens de l'inégalité $f(m) \leq k$.

Voici une implémentation de l'algorithme en Python :

```python
def dichotomie(a, b, k, f, p) :
    # f est une fonction continue sur [a;b]
    # k est un reel tel que f(a) <= k <= f(b)
    # p designe la precision voulue pour l'estimation de la solution
    while b - a >= p :
        m = (a + b) / 2
        if f(m) <= k :
                a = m
        else :
                b = m
    return a
```