1 Objectif du TP

Le but de ce TP est de tester différentes méthodes d'optimisation sans contraintes. Pour faire ces tests, vous pourrez soit utiliser une fonction $f : \mathbb{R}^n \to \mathbb{R}$ ($n \geq 2$) de votre choix, soit utiliser la fonction de Rosenbrock :

$$f(x) = 50(x_1^2 - x_2)^2 + (x_1 - 1)^2,\ x = (x_1, x_2)^T$$

La programmation se fera dans MATLAB. Pour vous faire une idée de la fonction de Rosenbrock, utilisez la fonction `contour` de MATLAB sur le rectangle $[-2, 2] \times [-1, 3]$.

Le travail qui vous est demandé consiste à tester les algorithmes suivants :

1. méthode de la plus forte pente,
2. algorithme de Fletcher-Reeves,
3. algorithme de Polak-Ribière,
4. algorithme de Newton, avec ou sans recherche linéaire,
5. algorithme DFP,
6. algorithme BFGS.

Vous résumerez ensuite dans un tableau les performances de ces algorithmes.

2 Recherche linéaire

Dans chacun des algorithmes utilisés, vous aurez besoin de faire une recherche linéaire consistant à minimiser la fonction :

$$q(t) = f(x + td),$$

où d est une direction de recherche qui dépend de l'algorithme utilisé. Nous utiliserons la méthode suivante, qui consiste à décider si t est accepté, ou bien considéré comme trop petit ou trop grand.

ALGORITHME TYPE DE RECHERCHE LINÉAIRE :

1. choix de $t > 0$, $t_g = 0$, $t_d = 0$.
2. test :
   - a) la valeur de t convient, fin de la recherche linéaire,
   - b) on considère que t est trop grand, on pose $t_d = t$, (t_d pour borne droite),
   - c) on considère que t est trop petit, on pose $t_g = t$, (t_g pour borne gauche),
3. si $t_d = 0$, on choisit $t > t_g$ (par exemple $t \leftarrow 10\,t$)
   sinon on choisit $t \in]t_g, t_d[$ (par exemple $t \leftarrow (t_g + t_d)/2$), puis retour en 2).

Pour décider du test ci-dessus, nous utiliserons la règle de Wolfe suivante.

Règle de Wolfe :

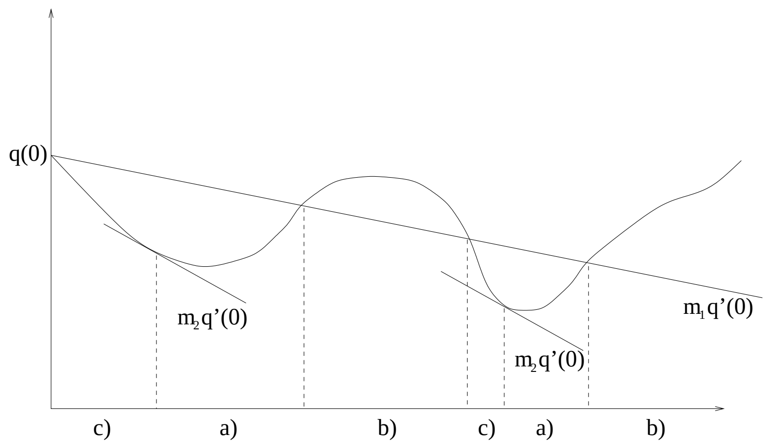

Fig. 1 – Règle de Wolfe

- a) $q(t) \leq q(0) + m_1 t q'(0)$ et $q'(t) \geq m_2 q'(0)$,
- b) $q(t) > q(0) + m_1 t q'(0)$,
- c) $q(t) \leq q(0) + m_1 t q'(0)$ et $q'(t) < m_2 q'(0)$.

Les paramètres m_1 et m_2 sont choisis tels que $0 < m_1 < 0,5 < m_2 < 1$.

La fonction à optimiser et sont gradient étant calculés par les fonctions MATLAB `fonct` et `gradf`, écrites respectivement dans les fichiers (m-files) `fonct.m` et `gradf.m`, vous programmerez cette recherche linéaire en suivant le format :

```
function t = rlin(x,t,d,fonct,gradf)
...
y = feval(fonct,x) ;
...
grady = feval(gradf,x) ;
...
```

L'appel à `rlin` se fait alors par la commande :

```
t = rlin(x,t,d,'fonct','gradf') ;
```

3 Travail à faire

Vous rédigerez un court rapport de 2 ou 3 pages contenant le tableau mentionné au §1 accompagné de vos commentaires.