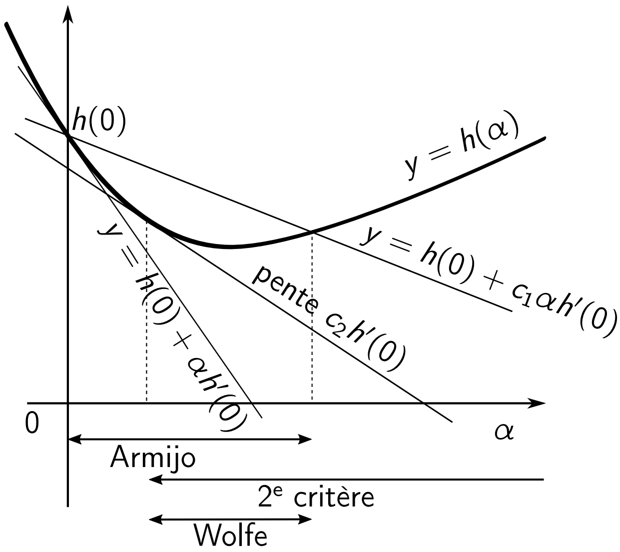
Méthodes numériques : optimisation.
Examen du 10 mai 2017 — Éléments de correction

1 Changement de variable affine pour Newton-Wolfe

1. La direction d est une direction de descente si et seulement si $\langle \nabla f(x), d\rangle < 0$ ce qui correspond à $h'(0) < 0$. Le pas α satisfait le critère de Wolfe si on a $h(\alpha) \leqslant h(0) + c_1 \alpha h'(0)$ (règle d'Armijo) et si $h'(\alpha) \geqslant c_2 h'(0)$ (deuxième condition).



2. On a alors, en posant $\widetilde{h}(t) = \widetilde{f}(\widetilde{x} + t\widetilde{d})$, que $\widetilde{h}(t) = \widetilde{f}(M(x+td)+b) = f(x+td) = h(t)$. On a donc $h = \widetilde{h}$, et tous les critères pouvant s'exprimer seulement à l'aide de la fonction h (ou $\widetilde{h}$), tels que le fait que d (ou $\widetilde{d}$) est une direction de descente, ou que α est un pas satisfaisant la règle de Wolfe.
3. On a
$$f(x+td) = f(x) + t\langle \nabla f(x), d\rangle + \frac{1}{2}t^2 \langle d, H_f(x)d\rangle + o(t^2).$$
Et aussi
$$\widetilde{f}(\widetilde{x}+t\widetilde{d}) = \widetilde{f}(\widetilde{x}) + t\langle \nabla \widetilde{f}(\widetilde{x}), \widetilde{d}\rangle + \frac{1}{2}t^2 \langle \widetilde{d}, H_{\widetilde{f}}(\widetilde{x})\widetilde{d}\rangle + o(t^2).$$
Si on a $\widetilde{d} = Md$ comme dans la question précédente, on obtient donc $f(x+td) = \widetilde{f}(\widetilde{x}+t\widetilde{d})$, ce qui donne donc
$$t\langle \nabla f(x), d\rangle + \frac{1}{2}t^2 \langle d, H_f(x)d\rangle - t\langle \nabla \widetilde{f}(\widetilde{x}), \widetilde{d}\rangle - \frac{1}{2}t^2 \langle \widetilde{d}, H_{\widetilde{f}}(\widetilde{x})\widetilde{d}\rangle = o(t^2).$$
En divisant par t et faisant tendre t vers 0, on obtient donc $\langle \nabla f(x), d\rangle - \langle \nabla \widetilde{f}(\widetilde{x}), \widetilde{d}\rangle = 0$ soit
$$\langle \nabla f(x), d\rangle = \langle \nabla \widetilde{f}(Mx+b), Md\rangle = \langle M^T \nabla \widetilde{f}(Mx+b), d\rangle.$$

Ceci étant valable pour d arbitraire, on a donc $\nabla f(x) = M^T \nabla \tilde{f}(Mx+b)$. En revenant à l'égalité précédente, on obtient donc

$$\frac{1}{2}t^2\langle d, H_f(x)d\rangle - \frac{1}{2}t^2\langle \tilde{d}, H_{\tilde{f}}(\tilde{x})\tilde{d}\rangle = o(t^2).$$

De même, ceci nous donne $\langle d, H_f(x)d\rangle = \langle d, M^T H_{\tilde{f}}(\tilde{x})M\, d\rangle$.
En notant A la matrice symétrique $H_f(x) - M^T H_{\tilde{f}}(\tilde{x})M$, on a donc que $\langle d, Ad\rangle = 0$ pour tout $d \in \mathbb{R}^n$.
Par polarisation, on obtient donc que $\langle e_i, Ae_j\rangle = \frac{1}{2}[\langle (e_i+e_j), A(e_i+e_j)\rangle - \langle e_i, Ae_i\rangle - \langle e_j, Ae_j\rangle] = 0$.
Autrement dit que $a_{ij} = 0$, donc $A = 0$. C'est à dire $H_f(x) = M^T H_{\tilde{f}}(\tilde{x})M$.

Si $H_f(x)$ est symétrique définie positive, pour tout $\tilde{d} \neq 0$ fixé, en posant $d = M^{-1}\tilde{d}$, on obtient que $\langle \tilde{d}, H_{\tilde{f}}(\tilde{x})\tilde{d}\rangle = \langle d, H_f(x)d\rangle > 0$ puisque $d \neq 0$.

Réciproquement si $H_{\tilde{f}}(\tilde{x})$ est symétrique définie positive, alors pour $d \neq 0$, en posant $\tilde{d} = Md \neq 0$, on obtient également $\langle d, H_f(x)d\rangle = \langle \tilde{d}, H_{\tilde{f}}(\tilde{x})\tilde{d}\rangle > 0$.

4. On a $x_{k+1} = x_k + \alpha_k d_k$, avec $d_k = -H_f(x_k)^{-1}\nabla f(xk)$, et α_k qui satisfait la règle de Wolfe au point x_k. Montrons que $\tilde{x}_{k+1} = \tilde{x}_k + \alpha_k \tilde{d}_k$ où $\tilde{d}_k = -H_{\tilde{f}}(\tilde{x}_k)^{-1}\nabla f(\tilde{x}_k)$ (ce qui correspond à la méthode de Newton pour $\tilde{f}$). On a donc

$$d_k = -(M^T H_{\tilde{f}}(\tilde{x})M)^{-1}(M^T\nabla\tilde{f}(\tilde{x})) = -M^{-1}H_{\tilde{f}}(\tilde{x}_k)^{-1}\nabla f(\tilde{x}k) = M^{-1}\tilde{d}_k.$$

On a donc $\tilde{d}_k = Md_k$, et donc $\tilde{x}_{k+1} - \tilde{x}_k = M(x_{k+1} - x_k) = \alpha_k M d_k = \alpha_k \widetilde{d_k}$.
Comme on a $\widetilde{d_k} = Md_k$, d'après la question 2, on obtient bien que si d_k est une direction de descente au point x_k, alors $\tilde{d}_k$ est une direction de descente au point $\tilde{x}_k$, et que si α_k satisfait la règle de Wolfe pour f pour la direction d_k au point x_k, alors ce même α_k satisfait la règle de Wolfe pour $\tilde{f}$ pour la direction $\tilde{d}_k$ au point $\tilde{x}_k$.

2 Descente de gradient et règle de Wolfe

1. (a) Le pas $\alpha_{\min}$ vérifie la règle d'Armijo (on a $h(\alpha_{\min}) \leqslant h(0) + c_1\alpha_{\min}h'(0)$). Mais il ne vérifie pas la deuxième condition (on a $h'(\alpha_{\min}) = \langle \nabla f(x + \alpha_{\min}d), d\rangle < c_2h'(0)$, sinon la ligne 34 serait effectuée, et on n'entrerait pas dans la troisième boucle). Le pas $\alpha_{\max}$, lui, ne vérifie pas la règle d'Armijo $h(\alpha_{\max}) > h(0) + c_1h'(0)$.

   (b) — Pour la première boucle, on divise $\alpha_{\min}$ par deux jusqu'à ce qu'il satisfasse la règle d'Armijo. Ceci est possible parce que pour $\alpha > 0$ suffisamment petit, α satisfait toujours la règle d'Armijo, puisque d est une direction de descente, et que donc $h(\alpha) = h(0) + \alpha h'(0) + o(\alpha)$ ce qui donne $h(\alpha) < h(0) + \alpha c_1 h'(0)$ lorsque α est suffisamment petit (car $0 < c_1 < 1$ et $h'(0) > 0$).
   — Pour la deuxième boucle, si tous les pas satisfont la règle d'Armijo, alors la fonction f ne serait pas bornée inférieurement, puisqu'on aurait $h(\alpha) \leqslant h(0) + \alpha c_1 h'(0) \to -\infty$ (on a aussi ici utilisé le fait que $h'(0) < 0$, i.e. d est une direction de descente).
   — Enfin pour la troisième boucle, si elle ne s'arrêtait pas, on aurait des $\alpha_{\min}$ et $\alpha_{\max}$ aussi proches que l'on veut d'une limite α_* (avec $\alpha_{\min} < \alpha_* < \alpha_{\max}$) vérifiant $\frac{h(\alpha_{\max})-h(\alpha_{\min})}{\alpha_{\max}-\alpha_{\min}} > c_1h'(0)$ et $h'(\alpha_{\min}) < c_2h'(0)$. Avec le théorème des accroissement finis, ceci donne l'existence de α_c aussi proche que l'on veut de α_* avec $h'(\alpha_c) > c_1h'(0)$. On utilise alors le fait que la fonction est C^1 pour passer à la limite dans ces deux inégalités, et obtenir $c_2h'(0) \geqslant h'(\alpha_*) \geqslant c_1h'(0)$. La contradiction utilise alors aussi le fait que $h'(0) < 0$.

   En résumé, l'hypothèse que d est une direction de descente est utilisée pour les trois boucles, l'hypothèse que f est bornée inférieurement est utilisée pour la deuxième boucle, et l'hypothèse que f est C^1 est utilisée pour la dernière boucle.

(c) Si `Nmax` est suffisamment grand, on ne sort donc des boucles que par les trois conditions `break` des lignes 29, 37 et 48, ou par la condition `return` de la ligne 35. Dans ce dernier cas, l'algorithme renvoie donc bien un pas satisfaisant la règle de Wolfe. Et si on est sorti de chacune des boucles par les instructions `break`, on a donc que le pas α correspondant à la variable `a` vérifie bien les deux conditions de la règle de Wolfe. On a donc bien existence d'un pas satisfaisant la règle de Wolfe, et l'algorithme renvoie bien pour `a` un tel pas si `Nmax` est suffisamment grand.

2. On a mis les variables d'entrées `h0` et `gfx` dans la fonction pour éviter de calculer plusieurs fois les mêmes choses. En effet si la méthode de descente nous demande de calculer la valeur de la fonction et du gradient avant de chercher un pas satisfaisant la méthode de Wolfe, ce peut être économique d'utiliser ces valeurs déjà calculées.

   Si on suppose que la valeur de la variable `ainit` est un pas qui satisfait la règle de Wolfe au point x pour la direction d, la fonction `f` est évaluée une seule fois à la ligne 24, puis la condition `if` de la ligne 28 est satisfaite dès le premier passage dans la première boucle. Ensuite au premier passage dans la deuxième boucle, la fonction `gradf` est évaluée une fois à la ligne 33, et dès la ligne 34, la condition `if` est satisfaite et l'algorithme renvoie la valeur de `amin`, qui correspond à la valeur de `ainit` (d'après la ligne 23, puisqu'ensuite la variable `amin` n'est jamais modifiée).

   Il a donc une seule évaluation de chaque fonction `f` et `gradf` dans ces conditions.

3. Voilà un exemple de code possible (avec une boucle `for` ou `while`)

```
def descenteGradWolfe(x0,tol,ainit=1,Nmax=1000):
    global compteurappels
    compteurappels=0
    listecompteur=[]
    listenormes=[]
    listef=[]
    x=x0
    gfx=gradf(x0)
    ngfx=norm(gfx)
    fx=f(x)
    listenormes.append(ngfx)
    listef.append(fx)
    listecompteur.append(compteurappels)
    a=ainit
    for i in range(Nmax):
        if ngfx<=tol:
            break
        d=-gfx
        a,fx,gfx=pasWolfe(x,d,a,fx,gfx)
        x=x+a*d
        ngfx=norm(gfx)
        listenormes.append(ngfx)
        listef.append(fx)
        listecompteur.append(compteurappels)
    return x,listecompteur,listenormes,listef
```

```
def descenteGradPasFixe(x0,tol,a=1,Nmax=1000):
    global compteurappels
    compteurappels=0
    listecompteur=[]
    listenormes=[]
    listef=[]
    x=x0
    gfx=gradf(x0)
    ngfx=norm(gfx)
    fx=f(x)
    listenormes.append(ngfx)
    listef.append(fx)
    listecompteur.append(compteurappels)
    i=0
    while ngfx>tol and i<Nmax:
        i+=1
        x=x-a*gfx
        fx=f(x)
        gfx=gradf(x)
        ngfx=norm(gfx)
        listenormes.append(ngfx)
        listef.append(fx)
        listecompteur.append(compteurappels)
    return x,listecompteur,listenormes,listef
```

4. Voilà un exemple de code possible

```
x0=array([2,2])
x,lc,ln,lf=descenteGradWolfe(x0,1e-5)
semilogy(lc,array(lf)+1)
semilogy(lc,ln)
```

```
x0=array([2,2])
x,lc,ln,lf=descenteGradWolfe(x0,1e-5,0.2)
semilogy(lc,array(lf)+1)
semilogy(lc,ln)
```

3 Taux de convergence de la méthode du gradient conjugué

1. On a $Ax_* + b = 0$. Donc on peut remplacer b par $-Ax_*$ dans les expressions de $f(x)$ et $f(x_*)$. On obtient

$$f(x) - f(x_*) = \tfrac{1}{2}\langle x, Ax\rangle + \langle b, x\rangle - \tfrac{1}{2}\langle x_*, Ax_*\rangle - \langle b, x_*\rangle = \tfrac{1}{2}\langle x, Ax\rangle - \langle Ax_*, x\rangle + \tfrac{1}{2}\langle x_*, Ax_*\rangle$$

Et d'autre part

$$\tfrac{1}{2}\|x - x_*\|_A = \tfrac{1}{2}\langle x - x_*, A(x - x_*)\rangle = \tfrac{1}{2}\langle x, Ax\rangle - \tfrac{1}{2}\langle x, Ax_*\rangle - \tfrac{1}{2}\langle x_*, Ax\rangle + \tfrac{1}{2}\langle x_*, Ax_*\rangle,$$

qui est bien égal au terme de droite de l'équation précédente puisque A est symétrique et qu'on a donc $\langle x, Ax_*\rangle = \langle x_*, Ax\rangle$.

2. On a (dans le cas où $p_k \neq 0$) que p_k est une direction de descente au point x_k et $\alpha_k > 0$ est le pas optimal dans la direction p_k. Autrement dit on a $f(x_k + tp_k) \geqslant f(x_k + \alpha_k p_k) = f(x_{k+1})$ avec égalité si et seulement si $t = \alpha_k$. En soustrayant $f(x_*)$ des deux côtés, on a donc $f(x_k + tp_k) - f(x_*) \geqslant f(x_{k+1}) - f(x_*)$ avec égalité si et seulement si $t = \alpha_k$.
Ou encore d'après la première question $\|x_k + tp_k - x_*\|_A^2 = \|e_k + tp_k\|_A^2 \geqslant \|x_{k+1} - x_*\|_A^2 = \|e_{k+1}\|_A^2$ avec égalité si et seulement si $t = \alpha_k$.
Pour $0 \leqslant i < k$, on a $\langle p_i, Ap_k\rangle = 0$. Donc si $v \in \mathrm{Vect}(p_0, p_1, \ldots, p_{k-1})$, on a $\langle v, Ap_k\rangle = 0$.
Donc, en développant

$$\begin{aligned}\|e_k + v + tp_k\|_A^2 &= \|e_k + v\|_A^2 + t^2\|p_k\|_A^2 + 2t\langle e_k + v, p_k\rangle \\ &= \|e_k + v\|_A^2 - \|e_k\|_A^2 + \|e_k\|_A^2 + t^2\|p_k\|_A^2 + 2t\langle e_k, p_k\rangle \\ &= \|e_k + v\|_A^2 - \|e_k\|_A^2 + \|e_k + tp_k\|_A^2.\end{aligned}$$

On veut montrer que si $e \in e_0 + \mathrm{Vect}(p_0, p_1, \ldots, p_{k-1})$, alors $\|e\|_A^2 \geqslant \|e_k\|_A^2$, avec égalité si et seulement si $e = e_k$. On a déjà par récurrence immédiate par la formule $e_{k+1} = e_k + \alpha_k p_k$ que pour $k \geqslant 1$, $e_k \in e_0 + \mathrm{Vect}(p_0, p_1, \ldots, p_{k-1})$.
Le cas $k = 1$ correspond au début de la question : si $e = e_0 + tp_0$ alors $\|e_0 + tp_0\|_A^2 \geqslant \|e_1\|_A^2$, avec égalité si et seulement si $t = \alpha_1$, autrement dit $e = e_1$. Si on suppose que le résultat est vrai au rang k, montrons le au rang $k + 1$.
Si $e \in e_0 + \mathrm{Vect}(p_0, p_1, \ldots, p_k)$, on a que $e - e_k$ appartient à $\mathrm{Vect}(p_0, p_1, \ldots, p_k)$, et s'écrit donc de la forme $e - e_k = v + tp_k$, où $v \in \mathrm{Vect}(p_0, p_1, \ldots, p_{k-1})$ et $t \in \mathbb{R}$. Donc

$$\|e\|_A^2 = \|e_k + v + tp_k\|_A^2 = \|e_k + v\|_A^2 - \|e_k\|_A^2 + \|e_k + tp_k\|_A^2 \geqslant \|e_k + tp_k\|_A^2 \geqslant \|e_{k+1}\|.$$

La première inégalité vient de l'hypothèse de récurrence et du fait que $e_k + v \in e_0 + \mathrm{Vect}(p_0, \ldots, p_{k-1})$. C'est une égalité si et seulement si $e_k + v = e_k$, autrement dit si $v = 0$. La deuxième inégalité est une égalité si et seulement si $t = \alpha_k$. Autrement dit on a bien l'inégalité demandée, qui est une égalité si et seulement si $e = e_k + \alpha_k p_k = e_{k+1}$.

3. On sait que $r_k = \nabla f(x_k)$ dans la méthode du gradient conjugué (avec les notations du cours). On a donc $r_k = Ax_k + b = Ax_k - Ax_* = Ae_k$.
On a vu à la question précédente que $e_k - e_0 \in \mathrm{Vect}(p_0, p_1, \ldots, p_{k-1})$. Mais cet espace est le même que $\mathrm{Vect}(r_0, Ar_0, \ldots, A^{k-1}r_0)$ d'après le cours. Et donc on obtient que $e_k - e_0 \in \mathrm{Vect}(r_0, Ar_0, \ldots, A^{k-1}r_0)$.
En remplaçant r_0 par Ae_0, on obtient le résultat voulu.
On a donc des réels $a_1, a_2, \ldots, a_k$ tels que $e_k - e_0 = \sum_{i=1}^{k} a_i A^i e_0$, donc en notant $a_0 = 1$ et en posant $P_k = \sum_{i=0}^{k} a_i X^i$, on obtient que $P(0) = a_0 = 1$ et que $e_k = P_k(A)e_0$.
Enfin si on a un polynôme $P \in \mathbb{R}_k[X]$ tel que $P(0) = 1$, alors en posant $e = P(A)e_0$, on a bien $e \in e_0 + \mathrm{Vect}(p_0, p_1, \ldots, p_{k-1})$. D'après la question précédente, on a donc que $\|e\|_A^2 \geqslant \|e_k\|_A^2$, autrement dit que $\|P(A)e_0\|_A^2 \geqslant \|P_k(A)e_0\|$, ce qui donne que P_k est bien un minimiseur du problème donné.

4. On écrit $e_0 = \sum_{i=1}^n t_i v_i$, et on a donc $Ae_0 = \sum_{i=1}^n \lambda_i t_i v_i$, et plus généralement $A^j e_0 = \sum_{i=1}^n \lambda_i^j t_i v_i$, et donc $P(A)e_0 = \sum_{i=1}^n P(\lambda_i) t_i v_i$.
La base v_i étant orthogonale, on a

$$\|P(A)e_0\|_A^2 = \langle P(A)e_0, AP(A)e_0\rangle = \sum_{i=1}^n \lambda_i P(\lambda_i)^2 t_i^2 \|v_i\|^2.$$

On a donc aussi $\|e_0\|_A^2 = \sum_{i=1}^n \lambda_i t_i^2 \|v_i\|^2$. Et donc

$$\|P(A)e_0\|_A^2 \leqslant \sum_{i=1}^n \lambda_i \Big(\max_{\lambda\in\Lambda} |P(\lambda)|^2\Big) t_i^2 \|v_i\|^2 = \max_{\lambda\in\Lambda} |P(\lambda)|^2 \|e_0\|_A^2.$$

D'après la question 3, on sait que pour tout polynôme P de $\mathbb{R}_k[X]$ tel que $P(0) = 1$, on a

$$\|e_k\|_A^2 = \|P_k(A)e_0\|_A^2 \leqslant \|P(A)e_0\|_A^2.$$

On obtient donc bien le résultat voulu.

5. Le pire des cas est quand la plus grande valeur propre vaut 1, alors le nombre de conditionnement vaut κ puisque la plus petite valeur propre est $\frac{1}{\kappa}$. D'après le cours, on sait qu'on a cette estimation :

$$\|e_k\|_A \leqslant 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|e_0\|_A.$$

Si on prend le polynôme $P(X) = (1-\kappa X)(1-X)^{k-1}$, on a bien $P(0) = 1$. Pour $\lambda \in \Lambda$, on a donc soit $\lambda = \frac{1}{\kappa}$ et alors $P(\lambda) = 0$, soit $\lambda \in [1-\rho, 1]$ et alors $P(\lambda) \in [(1-\kappa(1-\rho))\rho^{k-1}, 0]$, donc $|P(\lambda)| \leqslant |\kappa - \kappa\rho - 1|\rho^{k-1} \leqslant \kappa\rho^{k-1}$. Et donc on a $\max_{\lambda\in\Lambda} |P(\lambda)| \leqslant \kappa\rho^{k-1}$, ce qui nous donne bien

$$\|e_k\|_A \leqslant \kappa\rho^{k-1}\|e_0\|_A.$$

La convergence est donc plus rapide que ce qui est donné par le théorème si ρ est suffisamment petit (le taux de convergence est ρ, alors qu'il serait proche de 1 si κ est grand). Par exemple, si $\rho = \frac{1}{10}$ et $\kappa = 100$, le théorème donnerait $\|e_k\|_A \leqslant 2(\frac{9}{11})^{k-1}\|e_0\|_A$ alors qu'on a en fait au moins $\|e_k\|_A \leqslant \frac{1}{10^{k-3}}\|e_0\|_A$, qui est une meilleure estimation dès que $k \geqslant 3$ et qui converge bien plus rapidement vers 0.

6. * On peut en fait montrer que les polynôme de Tchebychev $T_k \in \mathbb{R}_k[X]$ sont donnés par les relations suivantes

$$T_0 = 1, \quad T_1 = X, \quad T_{k+1} = 2XT_k - T_{k-1} \text{ pour } k \geqslant 1.$$

On obtient immédiatement que $P_k(0) = 1$ (attention ici, petit conflit de notation avec la question 3, ce n'est pas le P_k qui minimise le problème). D'après l'expression pour $x \in [-1, 1]$, on a immédiatement que $|T_k(x)| \leqslant 1$ pour $x \in [-1, 1]$. Et si $x \in [\ell, L]$, alors $\frac{L+\ell-2x}{L-\ell} \in [-1, 1]$, et donc $|P_k(x)| \leqslant \frac{1}{T_k(\frac{L+\ell}{L-\ell})}$.
Si toutes les valeurs propres sont dans $[\ell, L]$, on a donc que $|P_k(\lambda)| \leqslant |T_k(\frac{L+\ell}{L-\ell})|^{-1}$ dès que $\lambda \in \Lambda$.
On a donc $\|e_k\|_A \leqslant |T_k(\frac{L+\ell}{L-\ell})|^{-1}\|e_0\|_A$.
On prend $x = \frac{L+\ell}{L-\ell}$. On a $x \pm \sqrt{x^2-1} = \frac{(L+\ell)\pm 2\sqrt{L\ell}}{L-\ell} = \frac{(\sqrt{L}\pm\sqrt{\ell})^2}{(\sqrt{L}+\sqrt{\ell})(\sqrt{L}-\sqrt{\ell})}$, et donc

$$T_k(\frac{L+\ell}{L-\ell}) = \frac{1}{2}\left[\left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k + \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k\right],$$

ce qui donne la première partie du résultat.
La deuxième inégalité provient simplement du fait que $\left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k + \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k \geqslant \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k$, et donc

$$\left[\left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k + \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^k\right]^{-1} \leqslant \left(\frac{\sqrt{L}+\sqrt{\ell}}{\sqrt{L}-\sqrt{\ell}}\right)^{-k} = \left(\frac{\sqrt{L}-\sqrt{\ell}}{\sqrt{L}+\sqrt{\ell}}\right)^k.$$