

Université de Bordeaux

Master 2 : Modélisation Numérique et Calcul Haute Performance

RAPPORT TP FREEFEM ++

Nastorg Matthieu & Philibert Thomas

Question 1 :

On s'intéresse à l'étude par éléments finis, via le langage de programmation FREEFEM++, du problème suivant :

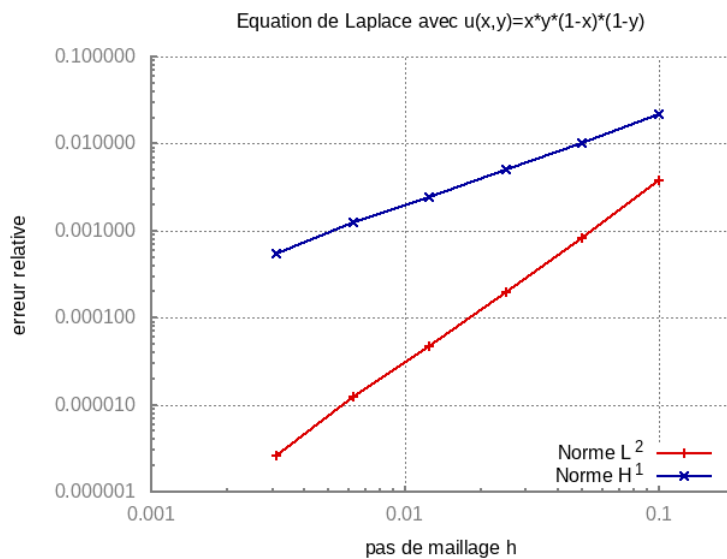
$$\begin{cases} -\Delta u = f & \text{si } u \in \Omega \\ u = 0 & \text{si } u \in \partial\Omega \end{cases}$$

Ici, nous avons choisi la fonction : $x(1-x)y(1-y)$. La résolution de ce problème nous est donné dans les fichiers fournis. On précise que Ω est un domaine carré $(0,1) \times (0,1)$. De plus nous utilisons pour notre résolution les éléments finis $P1$.

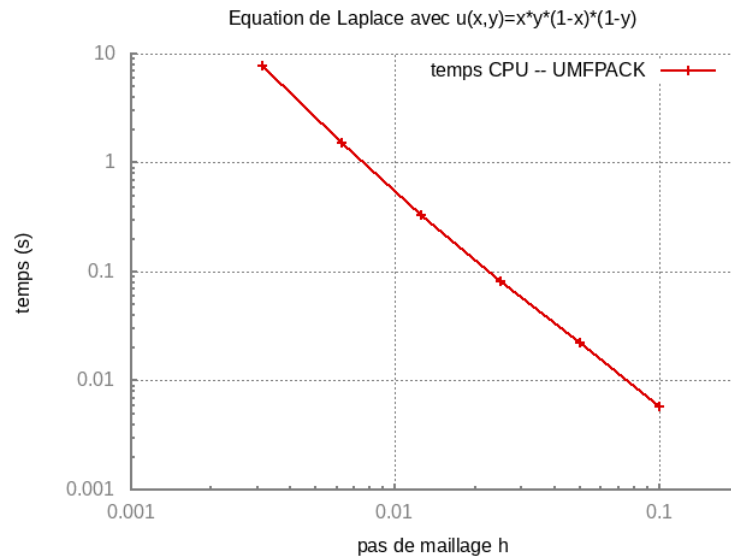
On nous demande d'évaluer l'erreur sur la semi-norme H^1 définie par :

$$E_{H^1} = \left(\int_{\Omega} |\nabla u - \nabla u_h|^2 dx \right)^{\frac{1}{2}}$$

On affiche, ci-dessous, le graphique d'erreur L^2 et H^1 relatif à la résolution approchée du problème. L'erreur est affichée en fonction d'une boucle de maillage de plus en plus fin.



De ces graphes, on peut dire que plus le maillage s'affine, plus les erreurs diminuent. En effet, en rajoutant des points dans le maillage, l'interpolation via les éléments finis $P1$ est plus précise et garantit donc une approximation de plus en plus juste (en atteste le tableau min-max décrit plus loin). Il est cependant important de trouver un juste milieu entre précision et temps de calcul... C'est pour cela que l'on affiche également ci-dessous le temps de calcul CPU en fonction de la finesse du maillage. On s'attend à ce que la courbe augmente plus le maillage s'affine. Résultat confirmé par le graphique ci-dessous.



Par suite, nous donnons le tableau des minimums et maximums des solutions approchées et exactes en fonction du pas du maillage. Il est fortement utile de ressortir ces valeurs pour analyser la justesse de notre solution. En effet, on s'attend à ce que plus le maillage s'affine, plus les maximums et minimums approchés se rapprochent de leurs valeurs exactes respectives. En plus du graphique de diminution de l'erreur, ces dernières nous donnent un second indice quant à la justesse de notre approximation, d'où l'intérêt de savoir les extraire et les analyser. Nous voyons que, dans notre cas, les valeurs approchées tendent bien vers celles exactes.

pas	minimum approché	minimum exact	maximum approché	maximum exact
0.1	$8.6 * 10^{-63}$	0	0.0622894	0.0625
0.05	$2.4 * 10^{-63}$	0	0.0624948	0.0625
0.025	$6.4 * 10^{-64}$	0	0.0624861	0.0625
0.0125	$1.6 * 10^{-64}$	0	0.0624961	0.0625
0.00625	$4.3 * 10^{-65}$	0	0.0624987	0.0625
0.003125	$1.05 * 10^{-65}$	0	0.0624996	0.0625

Question 2

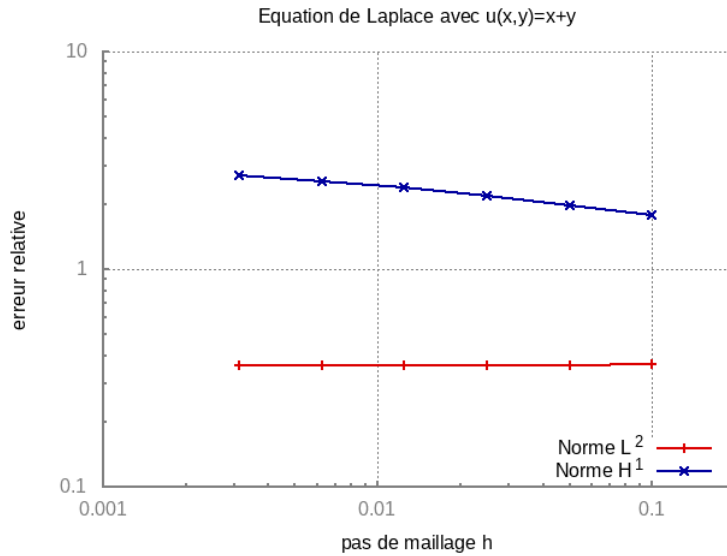
Nous allons rééditer l'étude faite à la question 1 pour la fonction exacte suivante : $u(x, y) = x + y$. Pour cela, il nous faut calculer le second membre f .

$$-\Delta u = f \Rightarrow f = 0$$

De plus, il va falloir changer les conditions de bord, que l'on implémentera de la façon suivante :

- Sur $\Gamma 1 : y = 0 \Rightarrow u(x, y) = x$
- Sur $\Gamma 2 : x = 1 \Rightarrow u(x, y) = 1 + y$
- Sur $\Gamma 3 : y = 1 \Rightarrow u(x, y) = 1 - x$
- Sur $\Gamma 4 : x = 0 \Rightarrow u(x, y) = y$

On résoud et on affiche le graphique d'erreur en fonction de la finesse du maillage.



On remarque que, quelque soit le pas de maillage, l'erreur reste constante. Cela provient du fait que les éléments P1 utilisés pour l'étude sont exacts. En effet, la fonction $u(x, y)$ approximée est linéaire et, comme les éléments P1 font des interpolations linéaires entre points du maillage, il est logique qu'ils soient, dans ce cas, exacts.

On retrouve les mêmes résultats que la question 1 concernant l'augmentation du temps de calcul CPU. Finalement, on affiche le tableau des minimums et maximums. On remarque que les résultats sont en adéquation avec nos attentes, ce qui permet de justifier la qualité de notre approximation pour cette étude.

pas	minimum approché	minimum exact	maximum approché	maximum exact
0.1	$1 * 10^{-31}$	0	1.9	2
0.05	$5 * 10^{-32}$	0	1.95	2
0.025	$2.5 * 10^{-32}$	0	1.975	2
0.0125	$1.25 * 10^{-32}$	0	1.9875	2
0.00625	$6.25 * 10^{-33}$	0	1.99375	2
0.003125	$3.125 * 10^{-33}$	0	1.99687	2

Question 3

Nous allons reproduire l'analyse pour la fonction $u(x, y) = \cos(2\pi x)y(1 - y)$ avec les conditions de bord mixtes suivantes :

- $u(x, y) = 0$ pour $\{y = 0\}$ et $\{y = 1\}$ (Condition de DIRICHLET)
- $\frac{\partial u}{\partial n} = 0$ pour $\{x = 0\}$ et $\{x = 1\}$ (Condition de NEUMANN)

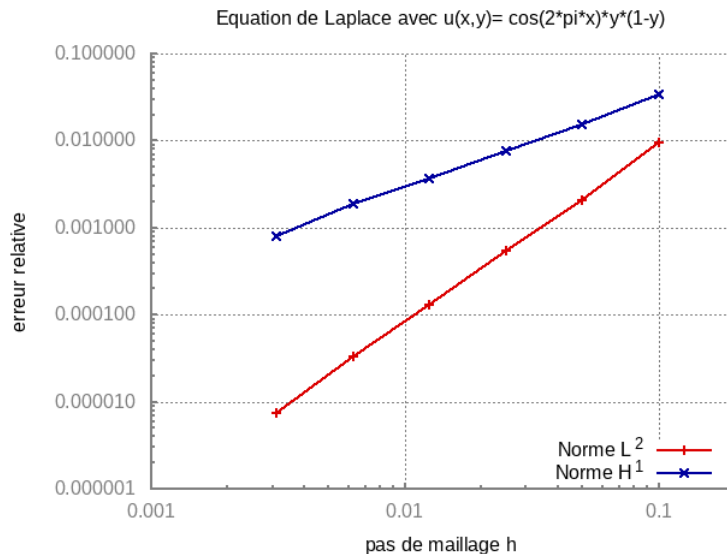
Pour commencer, calculons le second membre f :

$$-\Delta u = f \implies f = \cos(2\pi x)(2 - 4\pi^2 y(y - 1))$$

Concernant les conditions de bord :

- DIRICHLET : On implémente comme depuis le début sur Γ_1 et Γ_3
- NEUMANN : On ne fait rien car, dans la formulation variationnelle implémentée, nous avons déjà retiré le terme $\int_{\partial\Omega} \frac{\partial u}{\partial n} \cdot \nu ds$. Sans précisions sur les termes de bord (comme pour DIRICHLET par exemple), FREEFEM++ interprétera cela comme $\frac{\partial u}{\partial n} = 0$.

Comme pour les questions 1 et 2, nous affichons le graphique d'erreur. Comme on peut s'y attendre, l'erreur converge bien vers 0 quand le maillage s'affine.



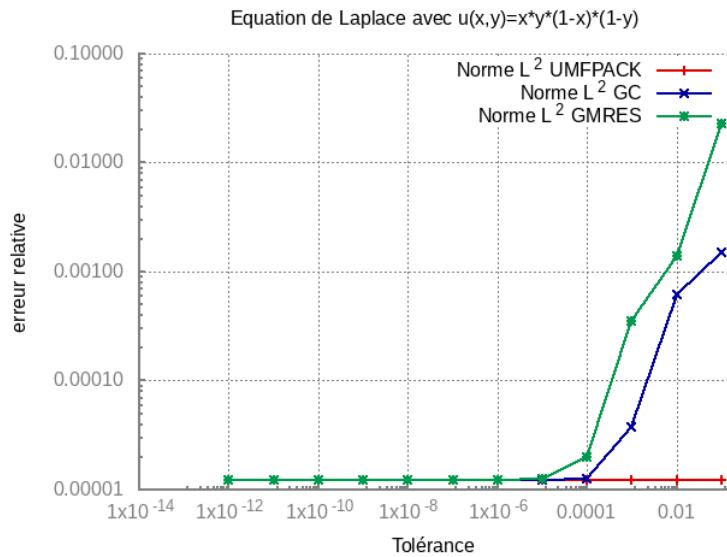
Finalement le calcul des maximums et minimums nous permettent de bien justifier la pertinence de notre approximation.

pas	minimum approché	minimum exact	maximum approché	maximum exact
0.1	-0.248208	-0.25	0.250697	0.25
0.05	-0.250084	-0.25	0.250316	0.25
0.025	-0.249807	-0.25	0.250188	0.25
0.0125	-0.249946	-0.25	0.250057	0.25
0.00625	-0.249983	-0.25	0.250023	0.25
0.003125	-0.249995	-0.25	0.250006	0.25

Question 4

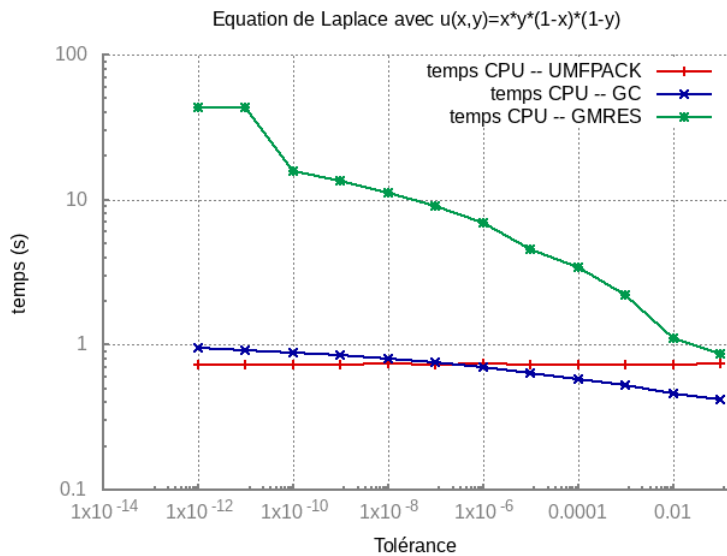
Pour les questions 4 et 5, nous allons d'avantage nous focaliser sur les méthodes de résolution de systèmes linéaires dont une directe (la méthode LU de UMFPACK) et deux itératives (la méthode du gradient conjugué CG ainsi que la généralisation de la méthode de minimisation du résidu GMRES). Pour cela, nous allons utiliser le problème initial de ce TP ainsi que la norme L2. Nous allons donc fixer notre maillage le plus fin possible pour résoudre le système linéaire dans un temps raisonnable avec UMFPACK soit 160 points par cotés du domaine considéré ($k = 4$).

Voici le graphe d'erreur en fonction de la tolérance donnée aux différentes méthodes :



On remarque que l'erreur engendrée par les méthodes itératives décroît rapidement avec la tolérance. De plus, à partir d'une tolérance de l'ordre de 10^{-5} , ces méthodes sont aussi précises que la méthode LU de UMPFPAK. Ce qui n'est pas étonnant puisque la méthode directe a une erreur de l'ordre de 10^{-5} aussi.

On s'intéresse maintenant au temps de calcul de ces différentes méthodes en fonction de la tolérance :



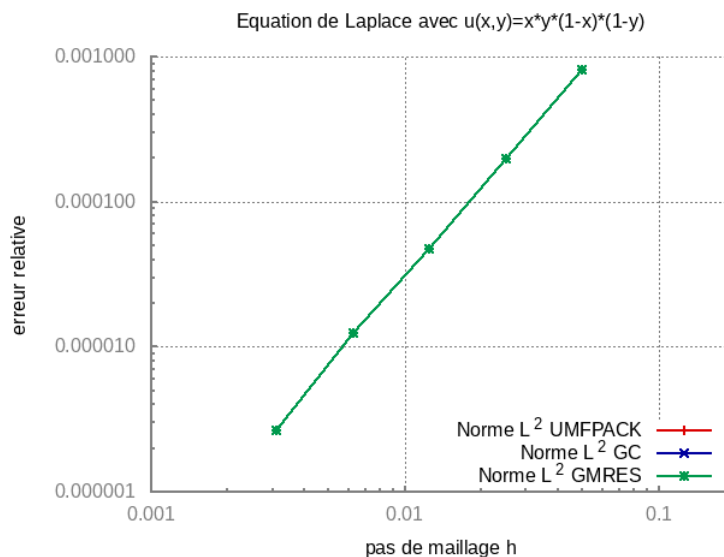
Ce graphe nous montre que la méthode GMRES met plus de temps que la méthode directe et que cette différence croît grandement avec la diminution de la tolérance. Il serait donc déconseillé d'utiliser cette méthode dans ce cas.

La méthode CG, quant à elle, met moins de temps que la méthode LU pour une tolérance supérieur à 10^{-5} et plus de temps lorsque cette tolérance passe sous cette limite. Il peut donc être avantageux d'utiliser cette méthode si l'on estime que l'erreur commise sur la résolution du système linéaire a moins d'importance que le gain de temps généré.

Question 5

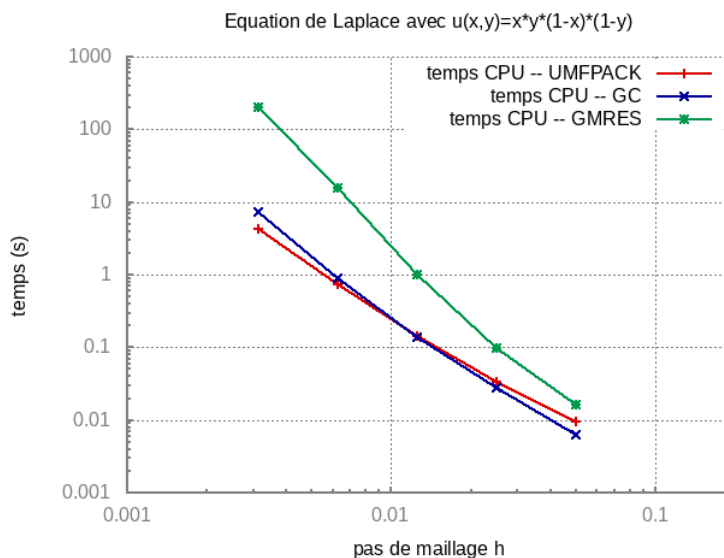
Dans cette section, on s'intéresse à l'impact de la finesse du maillage sur le temps de calcul ainsi qu'à l'erreur commise par les deux méthodes itératives utilisées précédemment. Pour cela, nous fixons la tolérance des ces deux méthodes à 10^{-10} et nous utilisons la suite de maillage utilisée dans les premières questions.

Nous obtenons le graphe d'erreur ci-dessous :



Les trois courbes sont confondues, ce qui indique qu'avec une tolérance de 10^{-10} , l'erreur commise sur la résolution du système linéaire est négligeable par rapport à l'erreur de la méthode employée.

Comparons maintenant le temps de calcul de ces différentes méthodes :



On s'aperçoit dans un premier temps que la méthode GMRES met toujours plus de temps que la méthode directe quelque soit la taille de maillage et que cette différence s'accroît avec la diminution de la taille de

ce maillage.

Dans un deuxième temps, on s'aperçoit que la méthode CG met moins de temps que la méthode directe pour un maillage grossier et qu'à partir d'une certaine limite, cette méthode devient moins efficace que la méthode directe.

Question 6

On considère dans cette question un cas fortement anisotrope avec des conditions de DIRICHLET homogènes. Plus précisément on s'intéresse à la résolution du problème suivant :

$$\begin{cases} -\operatorname{div}(a(x)\nabla u) = f & \text{si } u \in \Omega \\ u = 0 & \text{si } u \in \partial\Omega \end{cases}$$

où $a(x) = \operatorname{diag}(1, 10^r)$. De plus, nous utiliserons pour second membre f la fonction suivante : $f = 4\pi^2(1 + 10^r)\sin(2\pi x)\sin(2\pi y)$ dont la solution exacte, dans ce cas, est $u(x, y) = \sin(2\pi x)\sin(2\pi y)$.

Pour commencer, nous allons réécrire la formulation variationnelle correspondant au problème étudié. Soit v une fonction suffisamment régulière, en intégrant on a :

$$\begin{aligned} &\Rightarrow -\int_{\Omega} \operatorname{div}(a(x)\nabla u) \cdot v \, dx - \int_{\Omega} f \cdot v \, dx = 0 \\ &\Rightarrow -\int_{\Omega} \nabla \cdot (a(x)\nabla u) \cdot v \, dx - \int_{\Omega} f \cdot v \, dx = 0 \end{aligned}$$

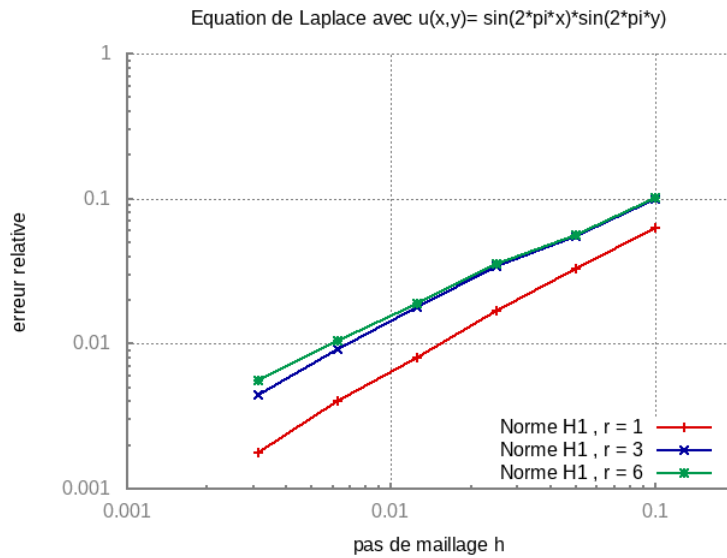
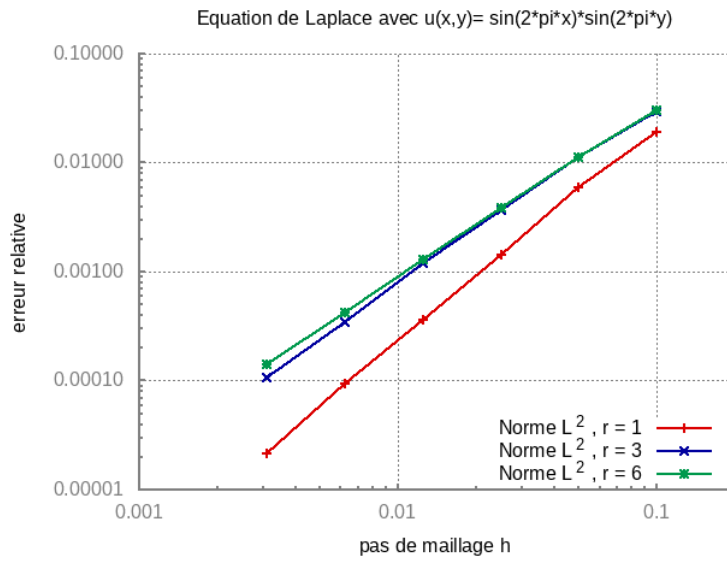
Formule de Green :

$$\Rightarrow \int_{\Omega} a(x)\nabla u \cdot \nabla v \, dx - \int_{\partial\Omega} \frac{a(x)\nabla u}{\partial n} v \, ds - \int_{\Omega} f \cdot v \, dx = 0$$

Comme $u = 0$ sur $\partial\Omega$:

$$\begin{aligned} &\Rightarrow \int_{\Omega} a(x)\nabla u \cdot \nabla v \, dx - \int_{\Omega} f \cdot v \, dx = 0 \\ &\Rightarrow \int_{\Omega} \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + 10^r \frac{\partial u}{\partial y} \frac{\partial v}{\partial y} \, dx - \int_{\Omega} f \cdot v \, dx = 0 \end{aligned}$$

On implémente les conditions de DIRICHLET comme depuis le début. On effectue une boucle pour $r = 1, 3, 6$. Finalement, on affiche le graphique d'erreur L2 puis H1 en fonction de la finesse du maillage.



Pour finir, nous affichons les valeurs de maximums et minimums approchées et exactes afin de justifier la pertinence de notre approximation.

pas	minimum exact	minimum approché $r = 1$	minimum approché $r = 3$	minimum approché $r = 6$
0.1	-0.951057	-0.957207	-0.957007	-0.957024
0.05	-1	-0.998645	-1.00135	-1.00146
0.025	-0.998981	-0.999503	-0.999291	-0.999252
0.0125	-0.999241	-0.999309	-0.999391	-0.999398
0.00625	-0.999807	-0.999836	-0.999764	-0.99977
0.003125	-1	-1.00004	-1.00008	-1.00008

pas	maximum exact	maximum approché r = 1	maximum approché r = 3	maximum approché r = 6
0.1	0.951057	0.961589	0.97398	0.974181
0.05	1	0.989663	0.986455	0.986449
0.025	1	1.0004	1.00084	1.00085
0.0125	0.999241	0.999171	0.999194	0.999196
0.00625	0.999868	0.999834	1.00011	1.00011
0.003125	1	0.99997	0.999953	0.99995