

RAPPORT DE SIMULATION DE PROCESSUS

APPLICATION DE LA THEORIE DES ELEMENTS FINIS EN C++

THIBON Clément
Elèves-Ingénieurs à l'E.N.S.G.

Mai 2024

Notice analytique

Titre du document Document title	Rapport de simulation de processus Report of process simulation
Cadre et nature du travail Framework and nature of the work	Rapport de projet informatique Report of informatic project
Date de publication Issue date	16 mai 2024
Auteurs Authors	THIBON Clément
Encadrement Superviser	CUPILLARD Paul
Résumé Summary	Ce projet présente l'implémentation de la théorie des éléments finis en C++ sur des problèmes simples This project presents the implementation of the finite elements theory in C++ on simple problems
Mots-clés Keywords	Eléments finis ; C++ ; Dirichlet ; Neumann Finite éléments ; C++ ; Dirichlet ; Neumann
Nombres de pages Number of pages	11
Annexes Annexes	0

Table des matières

Introduction	5
Les tests	5
La quadrature	5
Test de la transformée.....	5
Test du jacobien.....	5
Test d'assemblages	6
Test de l'assemblage global.....	6
Simulations	6
Problème de Dirichlet pur.....	6
Problème de Dirichlet avec un terme source	7
Problème sinus bump.....	7
Problèmes de Dirichlet et Neumann	9
Problème du mug.....	9
Conclusion	11

Tables des figures

Figure 1 : Schéma du problème de Dirichlet pur.....	6
Figure 2 : Solution du problème Dirichlet pur à gauche sur le maillage square.mesh et à droite sur le maillage square_fine.mesh	7
Figure 3 : Solution du problème de Dirichlet avec un terme source avec à gauche le maillage square.mesh et à droite le maillage square_fine.mesh	7
Figure 4 : Solution du problème sinus bump avec à gauche le maillage square.mesh et à droite le maillage square_fine.mesh	8
Figure 5 : Différence entre la solution obtenue et la solution analytique avec (à gauche) le maillage square.mesh et (à droite) le maillage square_fine.mesh	8
Figure 6 : Schéma du problème de Dirichlet et Neumann	9
Figure 7 : Solution du problème de Dirichlet et Neumann avec à gauche le maillage square.mesh et à droite le maillage square_fine.mesh	9
Figure 8 : Schéma du problème du mug.....	10
Figure 9 : Solution du problème du mug avec en haut à gauche le maillage le plus grossier (mug_1), en haut à droite un maillage plus fin (mug_0_2) et en bas le maillage le plus fin (mug_0_5).....	10

Introduction

La théorie des éléments finis permet de simuler informatiquement des processus de diffusion, par exemple. Là où les solutions analytiques arrivent à leur limites, cette technique, en discrétisant le domaine étudié et à partir d'hypothèses, peut arriver à une approximation satisfaisante de la solution. Il s'agit dans ce rapport d'étudier l'application de cette technique en C++ à des processus simples de diffusion.

Les tests

L'implémentation des résolutions aux processus étudiés se fait par étapes. Chacune d'elle se doit d'être testable et testée. Ainsi, une erreur à l'une de ces étapes peut être facilement vérifiée.

La quadrature

Le premier de ces tests s'effectuera sur la quadrature, un point essentiel de la théorie des éléments finis. Celle-ci se définit par la position des points d'intégration ainsi que ses poids et son ordre. En calculant à l'ordre 2 sur le triangle de référence, on a

$$\int_{\Omega_e} f(x) dx = \int_{\Omega_e} x^2 dx = \frac{1}{2}$$

Et en sommant les poids de la quadrature d'ordre 2 par ce programme, nous obtenons bien la valeur 2.

Test de la transformée

Les éléments constitutifs des maillages auxquels nous nous intéressons sont le bord (*edge*) et le triangle. Il convient de tester s'ils sont correctement rendus par notre programme. On test donc ici la fonction *transform* d'une instance de la classe *ElementMapping* sur un point de l'élément de référence dont on connaît la transformation dans l'espace réel. Le test a été effectué sur les quatrièmes triangles et bords du maillage *square.mesh* avec pour point (0.2, 0.4). La transformée du point pour le bord 4 a donné (0.671731, 0.929517) et pour le triangle 4 donne en effet ce que l'on cherchait.

Test du jacobien

Le jacobien est un outil mathématique au cœur de la théorie des éléments finis. Il nous faut donc tester sa réalisation dans le programme. La fonction *jacobian_matrix* de la classe *ElementMapping* permet de calculer le jacobien pour n'importe quel point dans un élément donné. Nous le testons ici pour le point (0.2, 0.4) du quatrième bord et du quatrième triangle. Les résultats sont cohérents avec ce que l'on aurait pu attendre. Il va de même pour la fonction *jacobian* de la même classe, qui calcule le déterminant de cette matrice.

Test d'assemblages

Le calcul des matrices et vecteurs élémentaires sont aussi nécessaires. Toujours pour les quatrièmes triangles et bords, et pour une quadrature d'ordre 2, nous calculons F_e et K_e , grâce respectivement aux fonctions *assemble_elementary_vector* et *assemble_elementary_matrix*, et obtenons des résultats cohérents.

Test de l'assemblage global

Après avoir créé les matrices élémentaires, il convient de les rassembler dans une matrice global, et de même pour les vecteurs. Ce sont ici les fonctions *local_to_global_matrix* et *local_to_global_vector* qui sont testées. Le résultat obtenu est satisfaisant et nous pouvons ainsi passer aux simulations

Simulations

Toutes les fonctions essentielles ayant été testées, il est temps de les utiliser pour résoudre des problèmes de diffusion. Tous les maillages suivants sont triangulaires. En ce qui concerne les problèmes quatre premiers problèmes, il s'effectueront sur deux maillages, *square.mesh* et *square_fine.mesh*, le premier étant un maillage moins fin que le second.

Problème de Dirichlet pur

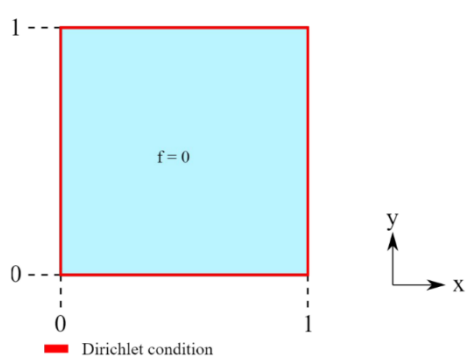


Figure 1 : Schéma du problème de Dirichlet pur

Le premier problème à résoudre (Figure 1) est une diffusion, avec pour coefficient de diffusion 1, sur un carré de longueur 1, avec pour toute condition aux bords, une condition de Dirichlet $u = x+y$. La solution donnée par notre programme pour les deux maillages et présentée Figure 2. Au vu de la forme de la condition de Dirichlet, on s'attend à ce que les valeurs augmentent de façon continue du coin bas gauche vers le coin haut droit. C'est bien ce que l'on observe. De plus les valeurs semblent cohérentes : 2 pour le point (1, 1) et des valeurs très faibles, proches de 0, en (0, 0). La résolution de ce premier problème est donc concluante. Le maillage plus fin offre une

meilleure résolution. Il est intéressant de remarquer que les valeurs les plus faibles des deux maillages ne sont pas les mêmes, en effet le maillage le plus fin offrira, pour des valeurs théoriquement nulles, des valeurs calculées plus faibles qu'un maillage plus grossier.

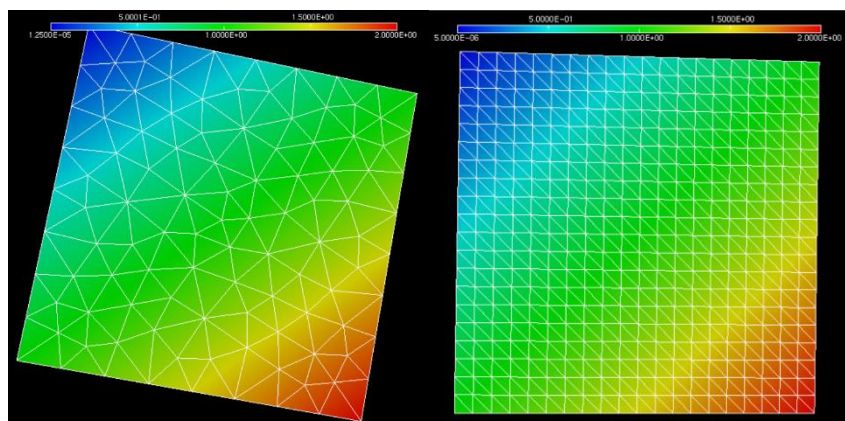


Figure 2 : Solution du problème Dirichlet pur à gauche sur le maillage square.mesh et à droite sur le maillage square_fine.mesh

Problème de Dirichlet avec un terme source

Ce problème ci est similaire au précédent, mais les conditions de Dirichlet sont nulles et l'on introduit un terme source égal à 1. La solution se situe Figure 3. Comme espéré, on trouve un gradient de valeurs entre le centre et le bord. Au bord, des valeurs très faibles proches de 0 trahissent les conditions de Dirichlet. Le centre et ses valeurs plus élevées se doit au terme source. Les conclusions précédentes sur les différences entre les deux maillages sont conservées.

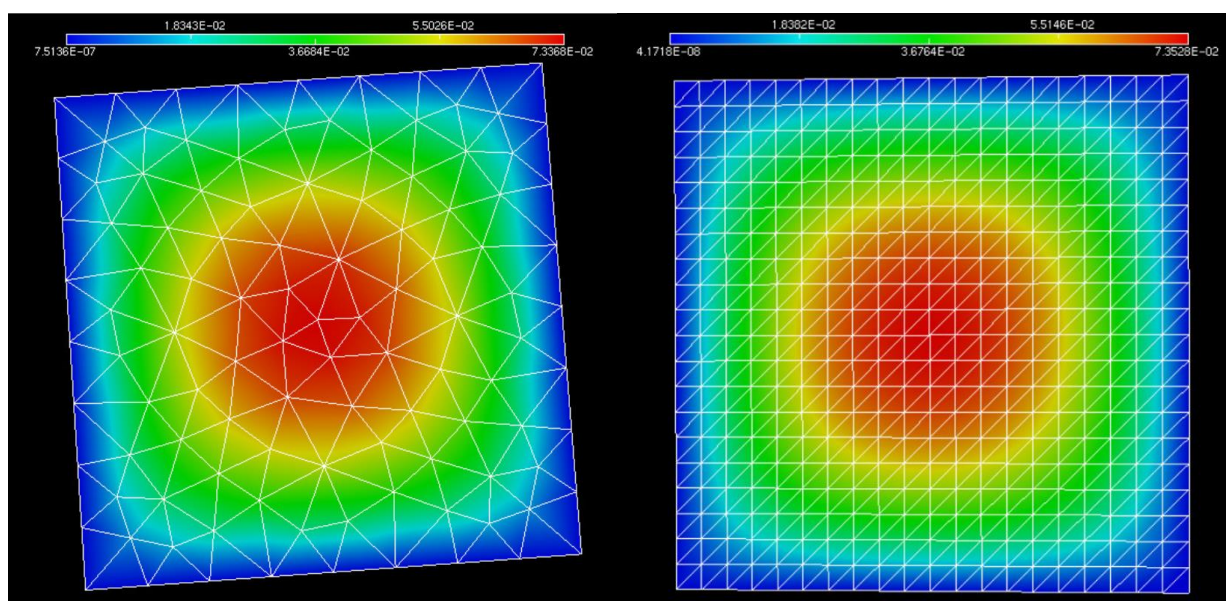


Figure 3 : Solution du problème de Dirichlet avec un terme source avec à gauche le maillage square.mesh et à droite le maillage square_fine.mesh

Problème sinus bump

Ce problème est le même que le précédent à ceci près que le terme source est ici égal à $2\pi^2 * \sin(\pi x) * \sin(\pi y)$. Le résultat présenté en Figure 4 ressemble sensiblement à la Figure 3, en raison d'une formulation très proche. Les mêmes remarques sur les deux maillages restent applicables. Nous sommes en

mesure de réaliser la solution analytique de formule $u = \sin(\pi * x) * \sin(\pi * y)$, cohérente avec l'allure de la figure 4, et comparer celle-ci avec ce que l'on obtient. Cette analyse se situe en figure 5.

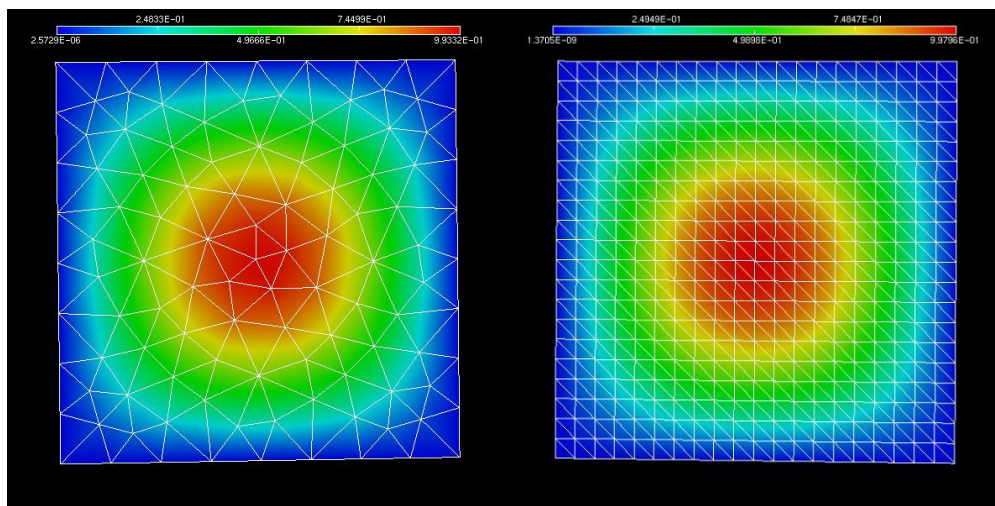


Figure 4 : Solution du problème sinus bump avec à gauche le maillage square.mesh et à droite le maillage square_fine.mesh

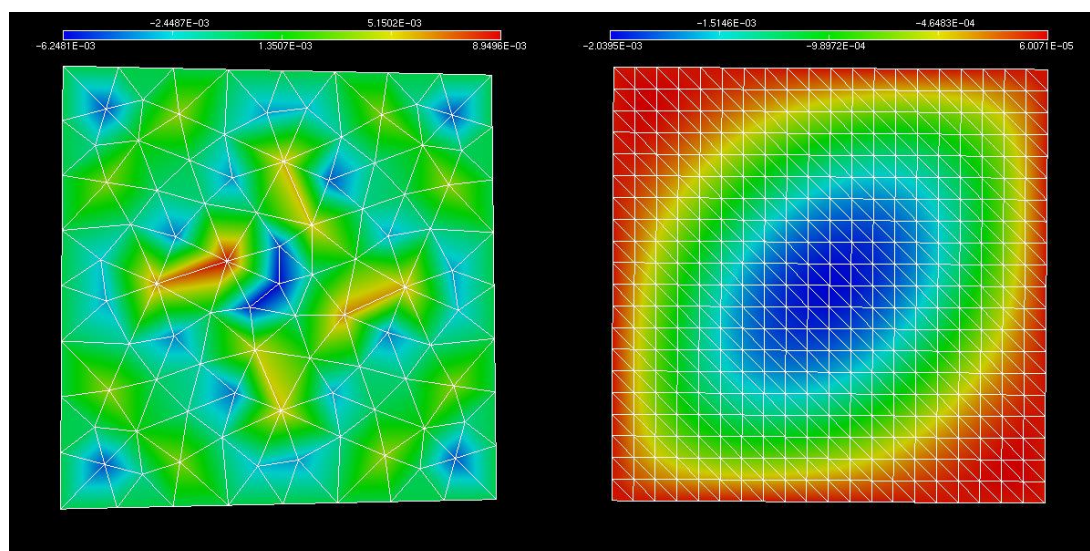


Figure 5 : Différence entre la solution obtenue et la solution analytique avec (à gauche) le maillage square.mesh et (à droite) le maillage square_fine.mesh

On observe sur la figure 5 dans le maillage le plus grossier que l'erreur est plutôt homogène et faible à part sur les côtés et sur les nœuds où l'erreur peut être élevée (maximum -0,1). Pour ce qui est du maillage fin, on remarque des formes ellipsoïdes concentriques avec une erreur minimale en vert et de part et d'autre une erreur augmentant dans un sens inverse (maximum au centre -0,002). Le maillage fin présente donc des erreurs moins localisée et suivant une tendance très lisible.

Problèmes de Dirichlet et Neumann

Le problème de Dirichlet et Neumann (Figure 6) amène une nouvelle condition : celle de Neumann, valant $\sin(\pi y)$. Elle vaut pour le bord gauche, et les bords hauts et bas (pour lesquels elle vaut 0). Sur le bord droit, on applique une condition de Dirichlet nulle. Le terme source vaut 1 et le coefficient de diffusion vaut 1.

La solution doit être symétrique par rapport à l'axe horizontal en $y = 0.5$. C'est bien ce qu'on observe sur la solution donnée par notre programme en Figure 7. Des valeurs faibles sur le bord soumis à des conditions de Dirichlet nulle et des valeurs fortes sur le bord gauche, et d'autant plus fortes qu'elle sont proches de $y=0.5$ (surtout visible sur le maillage le plus fin) sont cohérentes avec le problème posé.

La condition de Neumann semble donc avoir été implémentée avec succès. Cependant, il est étonnant de trouver un tel écart de valeurs entre les valeurs les plus fortes de square.bb (2) d'une part et de square_fine.bb (4) d'autre part, tandis que les écarts étaient jusqu'ici inférieurs à 0,05.

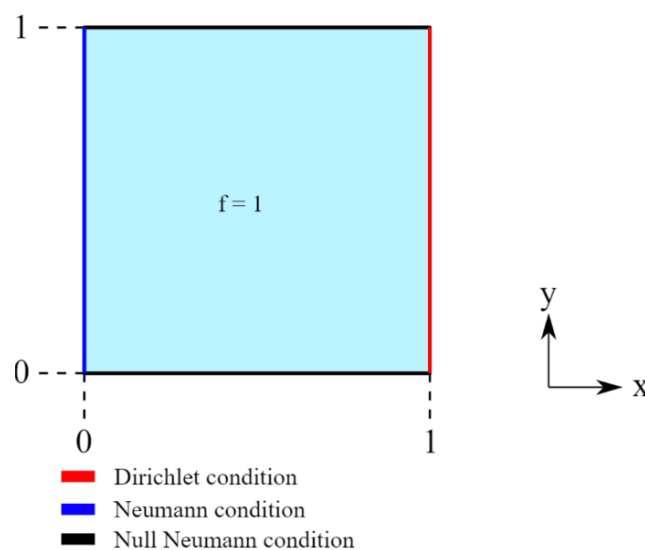


Figure 6 : Schéma du problème de Dirichlet et Neumann

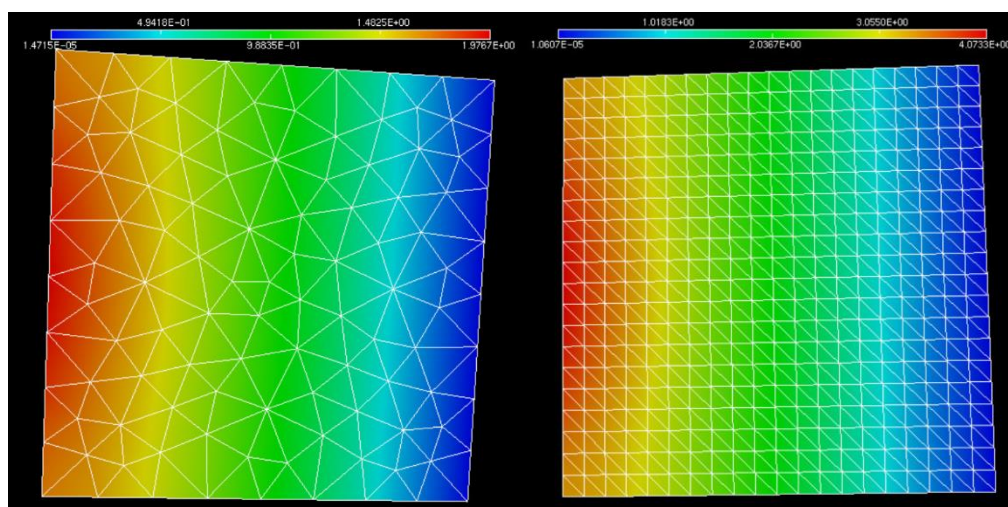


Figure 7 : Solution du problème de Dirichlet et Neumann avec à gauche le maillage square.mesh et à droite le maillage square_fine.mesh

Problème du mug

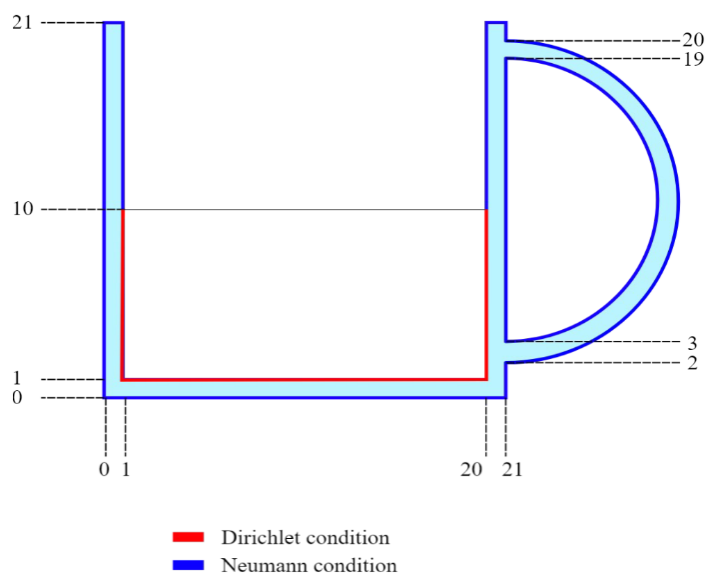


Figure 8 : Schéma du problème du mug

Nous nous intéressons ici à la diffusion de la chaleur dans un mug à moitié rempli d'eau bouillante (Figure 8). La condition de Dirichlet imposée par l'eau bouillante vaut 100 et la condition de Neumann ailleurs est considérée valoir -0,1. Le coefficient de diffusion vaut 1. La solution présentée en Figure 9 est cohérente avec ce problème, avec les parties rouges les parties les plus chaudes et les parties bleus les plus froides. Les trois maillages montre la même tendance.

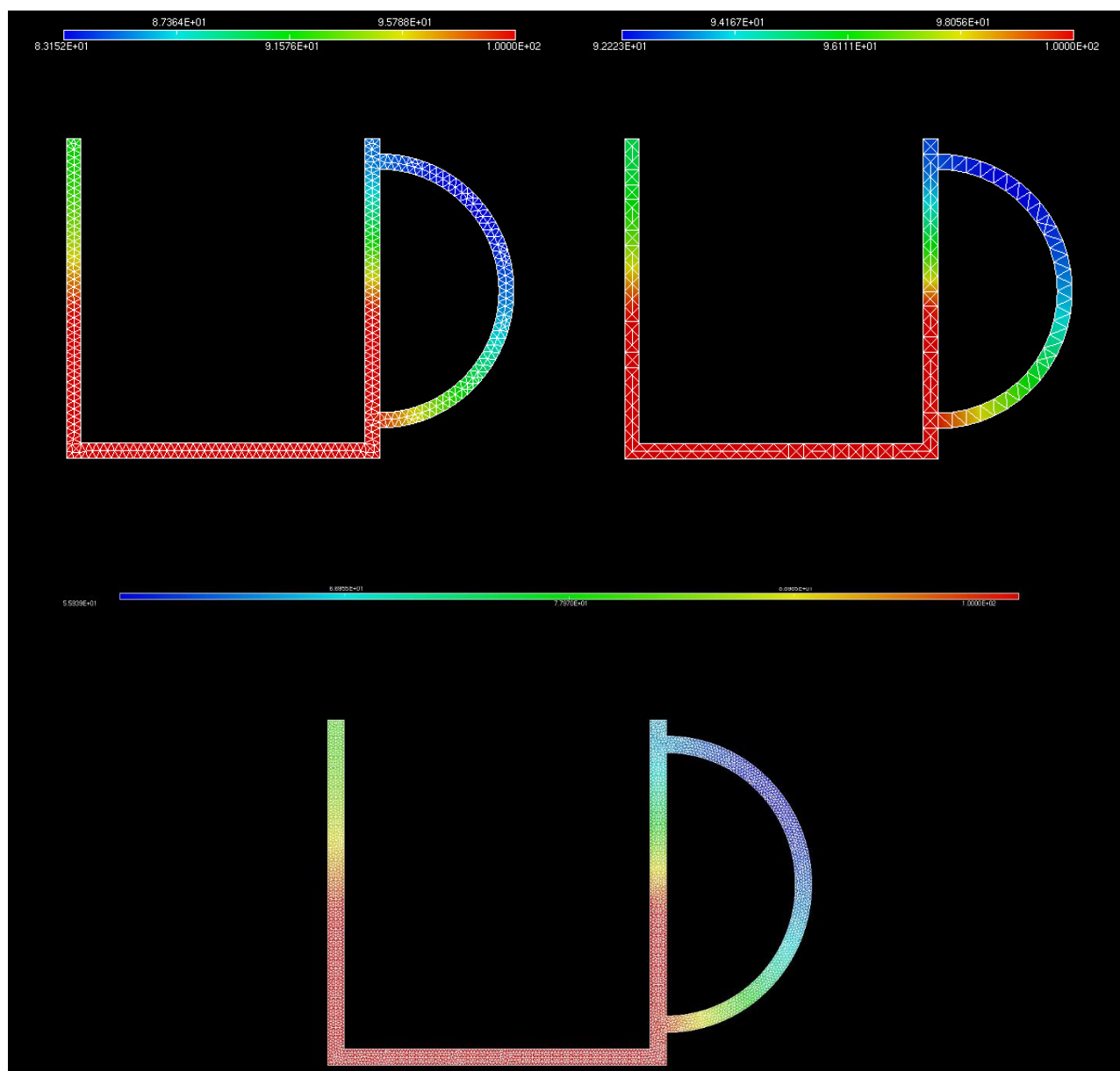


Figure 9 : Solution du problème du mug avec en haut à gauche le maillage le plus grossier (mug_1), en haut à droite un maillage plus fin (mug_0_2) et en bas le maillage le plus fin (mug_0_5)

Conclusion

L'implémentation de la théorie des éléments finis en C++ prouve l'utilité de cette méthode dans la modélisation de processus physiques de diffusion sur ces problèmes simples et suggère son applicabilité sur des problèmes plus complexes.