ECOLE NATIONALE SUPERIEURE D'ARTS ET METIERS

Mixed finite elements for the thin plate bending problem

TIBA Azzeddine
February 1, 2020

Abstract The project consists of the numerical solution of the biharmonic equation describing the trasnversal displacement of a thin plate under the effect of external and internal loads. The solution is based on the P2 finite element method, a mixed formulation was used in order to solve the problem. As for the time discretization, the method of finite differences was used, where two implicit time integration schemes were compared; The backward Euler scheme, and the HHT method. The HHT method is of value due to the improved numerical damping introduced while maintaining a second order accuracy. Error estimates are provided for the corresponding discrete problem. We then implemented a code for the problem on *Matlab* and showed the numerical and graphical results.
In order to show the benefits of using the conjugate gradients method on the problem, preconditioners were used on the block matrices of the problem enabled using the *C++* open source software *deal.ii*.

Résumé Le projet consiste à la résolution numérique de l'équation biharmonique qui décrit le déplacement transversal d'une plaque soumise aux effort extérieurs et intérieurs. La résolution s'effectue par la méthode des éléments finis de type P2, des éléments finis mixtes ont été utilisés après la formulation faible du problème. Pour la discrétisation temporelle, nous avons utilisé la méthode de différences finies, deux schémas implicites ont été appliqués afin d'approcher la dérivée temporelle; le schéma décentré arrière, et la méthode HHT. La méthode HHT, contrairement au schéma arrière, permet de contrôler la quantité d'amortissement numérique, et permet une approximation de second ordre. Des estimations d'erreurs sont fournies pour le problème discret. Un programme de ce calcul a été implémenté sur *Matlab* pour ensuite afficher les résultats numériques et graphiques.
Le problème statique a aussi été résolu en utilisant le code Open Source en *C++* *deal.ii*, des préconditionneurs ont été utilisé en manipulant les matrices blocs du problème afin de pouvoir utiliser la méthode du gradient conjugué pour le résoudre.

Contents

1 Introduction

1.1 The biharmonic equation

The problem describes the bending of thin plates, where the thickness h is very small comparing to the other two dimensions. (usually $\frac{h}{L}10$). Using the Kirchoff-Love theory where transverse shear is negligible, and the plane section stays perpendicular to the mid surface.
Solving the bending problem of this plate, i.e finding the transverse displacement leads to a dynamic (or static) bidimensional partial differential equation1. The boundary conditions considered in this parts describe the fixed plates borders 21, and the initial conditions of a plate in rest at $t = 0$. We note that $u(\mathbf{x}, t) = u(x, y, t)$

$$\boxed{\rho h \frac{\partial^2 u}{\partial^2 t}(\mathbf{x}, t) + D\Delta(\Delta u(\mathbf{x}, t)) = g(\mathbf{x}, t) \quad , \forall(\mathbf{x}, t) \in \Omega[0, \infty[} \tag{1}$$

- Initial conditions :

$$u(\mathbf{x}, 0) = 0 \qquad , \qquad \forall \mathbf{x} \in \Omega \tag{2}$$

- Boundary conditions : Fixed

$$u(\mathbf{x}, t) = \frac{\partial u}{\partial n}(\mathbf{x}, t) = 0 \qquad , \qquad \forall(\mathbf{x}, t) \in \partial\Omega[0, \infty[\tag{3}$$

The finite element used in this part are P_2 quadratic Lagrange elements. We note these elements here as T_k elements.

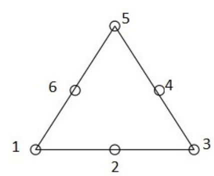


Figure 1: 6 nodes quadratic triangular element.

1.2 Problem formulation

The problem 1-->21 can thus be simplified by writing : $u_0(\mathbf{x}) = 0$ and $\dfrac{D}{\rho h} = 1$

$$\begin{cases} \dfrac{\partial^2 u}{\partial^2 t}(\mathbf{x}, t) + \Delta(\Delta u(\mathbf{x}, t)) = g(\mathbf{x}, t) & \forall(\mathbf{x}, t) \in \Omega[0, \infty[\\ u(\mathbf{x}, 0) = \dot{u}(\mathbf{x}, 0) = 0 & \forall \mathbf{x} \in \Omega \\ u(\mathbf{x}, t) = \dfrac{\partial u}{\partial n}(\mathbf{x}, t) = 0 & \forall(\mathbf{x}, t) \in \partial\Omega[0, \infty[\end{cases} \tag{4}$$

2 Time discretization

The equations 4 could be solved using implicit or explicit time integration schemes, explicit schemes require less computation time, they have however conditional stability; very small time steps can significantly increase the CPU time needed. For the implicit time integration schemes, the computing time is higher because of the need to solve the linear system of equations and storing the system's matrix at each time step. Nonetheless, these schemes are unconditionally stable, making the time step

dependent only on the nature of the problem, reducing thus the computational time.
When investigating these characteristics, implicit time integration schemes are adopted for the time discretization of the plate's bending problem, since only linear small displacements are considered here and since the time step chosen is significant.
The implicit time integration schemes have a numerical damping to the dynamic response, which reduces the magnitude of the higher modes' response, those modes being the ones that don't imply physical interpretations. Numerous existent implicit schemes introduce (and sometimes control) a numerical damping while minimizing the dissipation of the lower modes' responses: Wilson's method [1] that is less and less used due to its significant low frequencies' damping properties[2], Newmark method [3] and the generalized α method[4].
We choose here the implicit **backward Euler scheme**, and **Hilber-Hughes-Taylor (HHT)** method [5]. These two are usually used in the commercial finite element software.[1]

2.1 Backward Euler scheme :

It's an first order scheme, unconditionally stable, where:

$$\ddot{U}= \frac{U^{n-2}-2U^{n-1}+U^{n}}{h^2} \tag{5}$$

where U^n is the solution at the moment $t_n=(n-1)h$ and $h=t_{n+1}-t_n$ subsequently 5 becomes :

$$\boxed{\frac{U^{n-2}-2U^{n-1}+U^{n}}{h^2}+\Delta(\Delta(U^n))=g^n} \tag{6}$$

The initial condition gives $\dfrac{\partial U}{\partial t}(x,y,0)=0$ and using the centered integration scheme, we introduce a fictitious point on the time grid U^{-1} that verifies $U^{-1}=U^1$ and replacing it in 6 gives $\dfrac{2U^1}{h^2}+\Delta(\Delta(U^1))=f^1$ it is thus the same as solving 6 with $U^{n-1}=U^{n-2}=0$ and a time step h'$=\dfrac{h}{\sqrt{2}}$.

2.2 HHT method:

$$u_{i+1}\approx x_i+h^2[(1/2-\beta)\ddot{u}_i+\beta\ddot{u}_{i+1}] \tag{7}$$

and

$$\dot{u}_{i+1}\approx \dot{u}_i+h[(1-\gamma)\ddot{u}_i+\gamma\ddot{u}_{i+1}] \tag{8}$$

give

$$[M+h^2(1-\alpha)\beta K]\ddot{u}_{i+1}+h^2(1-\alpha)(1/2-\beta)K\ddot{u}_i+h(1-\alpha)K\dot{u}_i+Kx_i=(1-\alpha)f_{i+1}+\alpha f_i \tag{9}$$

It is an implicit time integration scheme that, with the α parameter, introduces numerical damping while keeping a **second order approximation** (which is not possible with the Newmark method).

By choosing :

$$0\leq\alpha\leq 1/3 \quad , \quad \beta=\frac{(1+\alpha)^2}{4} \quad , \quad \gamma=1/2+\alpha$$

It is unconditionally stable. Matrices **K**, **M** and **f** verify:
$M\ddot{\mathbf{u}}+K\mathbf{u}=f$ (See Section 4).

[1] Le code *Comsol* for example uses the explicit centered Euler scheme, the backward Euler scheme and the generalized α scheme. We note that the generalized α scheme is a generalization of Newmark and HHT methods.

3 Variational form

A weak form of the problem 4 gives :

$$\int_\Omega (\Delta^2 u).v.d\mathbf{x} = \int_\Omega \Delta(\Delta u).v.d\mathbf{x} \qquad \forall\ v \in H_0^2(\Omega) \tag{10}$$

$$\iff \int_\Omega \Delta u.\Delta v.d\mathbf{x} + \int_{\partial\Omega} [v.(\nabla(\Delta u).n) - (\nabla v.n).\Delta u].d\Gamma = \int_\Omega \Delta u.\Delta v.d\mathbf{x} \tag{11}$$

$$\iff \int_\Omega \Delta u.\Delta v.d\mathbf{x} = \int_\Omega g.v.d\mathbf{x} \qquad \forall\ v \in H_0^2(\Omega) \tag{12}$$

Green's formula and boundary conditions on v ($v = 0,\ (\nabla v.n) = 0\ on\ d\Omega$), lead to 12
Lagrange finite elements require conform elements with C^1 shape functions, these functions can sometimes be hard to construct.
Another approach to solve this problem is the use of nonconforming finite elements, Hermite (Argyris) finite elements or **mixed finite elements**. This form is used usually in fluid mechanics PDE.[6, 7, 8]
This leads to the next problem:

$$\begin{cases} D.\Delta U^n = W & On\ \Omega \\ \Delta W = g^n - \rho H \dfrac{U^{n-2} - 2U^{n-1} + U^n}{h^2} & On\ \Omega \\ U = \dfrac{\partial U}{\partial n} = 0 & On\ \partial\Omega \end{cases} \tag{13}$$

Where U is **the displacement** and W is **the bending moment.** The discretization gives :

$$\begin{bmatrix} D.\mathbf{K} & \mathbf{M} \\ \dfrac{\rho.H}{h}\mathbf{M} & -\mathbf{K} \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ \vdots \\ u_N \\ w_1 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \int_\Omega (g^n - \rho H \dfrac{U^{n-2} - 2U^{n-1}}{h^2})\varphi_1 \\ \vdots \\ \int_\Omega (g^n - \rho H \dfrac{U^{n-2} - 2U^{n-1}}{h^2})\varphi_N \end{bmatrix} \tag{14}$$

Where $\quad K_{ij} = \int_\Omega \nabla\varphi_i.\nabla\varphi_j \quad , \quad M_{ij} = \int_\Omega \varphi_i.\varphi_j$
so that $\varphi_i \in P_2$ represent **shape functions** .
The mixed formulation of unknowns of different spaces require the use of finite elements that satisfy the Ladyzhenskaya-Babuska-Brezzi (LBB) condition [9, 10] in order to insure that the discrete solution is stable and will converge to the exact solution.

4 Implementing the method(*Matlab*)

The computation of the matrices **K** and **M** coefficients is done by assembling elementary matrices Ke and Me for each element. In order to solve the integrals of the matrices' coefficients, using Gauss quadrature and reference elements transformations, isoparametric elements were used[2]. Reference

[2]An element is isoparametric if the same shape functions are used for geometry and for displacement interpolation. We should mention that in this case, $P1$ Lagrange geometry description in this case are sufficient since the mesh used is regular.

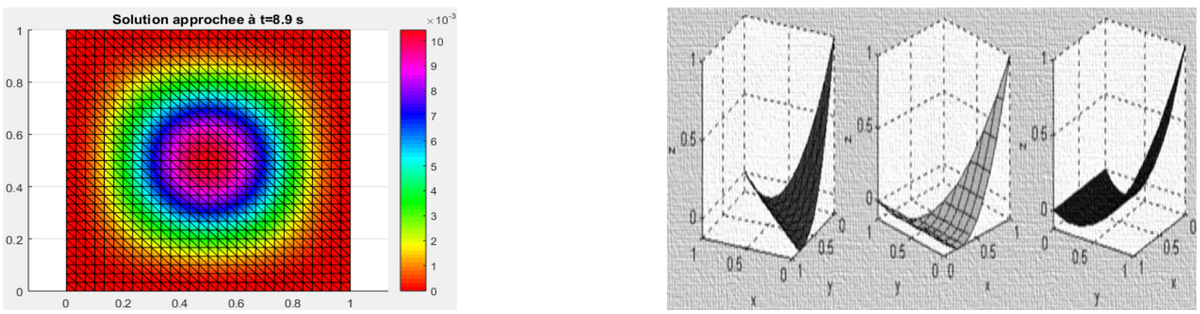
shape functions should thus be computed (**Fig.** 2b).

$$K_{ij} = \int_0^1\int_0^1 (J^*)^t \nabla\hat{\phi}_i.(J^*)^t \nabla\hat{\phi}_j det(J) dsdt \tag{15}$$

$$M_{ij} = \int_0^1\int_0^1 \hat{\phi}_i\hat{\phi}_j det(J) dsdt \tag{16}$$

Where J is the jacobian matrix of the transformation from (x, y) to (s, t), J^* from (s,t) to (x,y)[3] and $\hat{\phi}$ is the shape function for the reference element. (**Fig** 2b).

For the HHT method, we can rewrite 14 as $M^*\ddot{\mathbf{u}} + K^*\mathbf{u} = f$ où $M^* = \rho HM$ and $K^* = D((K - M)M^{-1} + I)K$. After solving 14, we get the solution vector and we can visualize the displacement and the bending moment of the studied plate.



(a) affichage de la solution approchée de ue h=0.08s avec 61x61 noeuds

(b) Les fonctions de forme P2

Figure 2: Shape functions and solution visualization

5 Method validation

In order to highlight convergence, we choose an exact solution $u_e(x, y, t) = t^2 e^{-t} cos(\pi x)^2 sin(\pi x)^2$ and a function g that verifies 1. The chosen problem is around a $[0, 1]x[0, 1]$ plate and $\frac{D}{\rho h} = 1$. The approximate solution by the numerical *Matlab* program can be validated by verifying the convergence of this solution to the exact solution, this convergence is evaluated by computing the error. This error is computed as the norms 17, 18 in their respective spaces L^2 and H^1; $||u - u_h||_{L^2}$ and $||u - u_h||_{H^1}$, and show the convergence of these norms towards zero when the mesh is smaller.

$$||f||_{L^2} = (\int_\Omega |f(x)|^2 dx)^{1/2} \tag{17}$$

$$||f||_{H^1} = (\int_\Omega \Delta |f(x)|^2 dx + \int_\Omega |f(x)|^2 dx)^{1/2}. \tag{18}$$

The computation of these errors is implemented in *Matlab* in the programs *H1_Error.m* and *L2_Error.m* .

The table 1 shows the numerical convergence of the approximate solution when the mesh is finer while keeping the time step fixed.

[3]Generally, $J^* \neq J^{-1}$ for *P2elements*.

The table 2 shows the numerical convergence when decreasing the time step while keeping the mesh size fixed.
The error computation was done for the backward Euler scheme.

Table 1: Convergence -Space-

Number of nodes	$\|\|u-u_h\|\|_{L^2}$			$\|\|u-u_h\|\|_{H^1}$			Computation time
	t=0.3 s	t= 2 s	t= 5.1 s	t=0.3 s	t= 2 s	t= 5.1 s	
23x23	0.2265	3.4502	0.9976	0.281	3.6399	1.0547	164.87 s
33x33	0.1903	2.4207	0.7046	0.1953	2.484	0.7231	240.34 s
51x51	0.06015	0.6449	0.1958	0.06076	0.6516	0.1979	1300.30 s
61x61	0.0113	0.3316	0.0838	0.01146	0.3345	0.0864	2499.01 s

The time step is $h = 0.1s$ and the final instant is $9s$ with 16 quadrature points (1 CPU 2.30 GHz)

Table 2: Convergence -Time-

Time step	$\|\|u-u_h\|\|_{L^2}$			$\|\|u-u_h\|\|_{H^1}$			Computation time
	t=0.3 s	t= 2 s	t= 5.1 s	t=0.3 s	t= 2 s	t= 5.1 s	
0.2 s	1.664	21.277	6.089	1.756 s	22.455	6.426	52.82 s
0.15 s	0.5336	10.41	2.9918	0.56299	10.996	3.157	79.81 s
0.1 s	0.2265	3.4502	0.9976	0.281	3.6399	1.0547	164.87 s
0.06 s	0.00197	0.1154	0.03094	0.003397	0.12164	0.03834	166.88 s

The mesh used is of $23x23$ nodes and the final instant of $9s$ with 16 quadrature points (1 CPU 2.30 GHz)

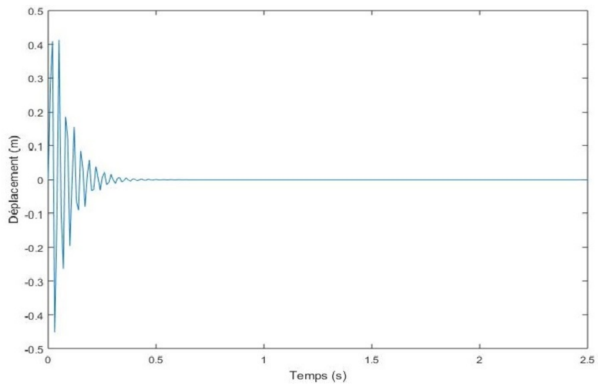
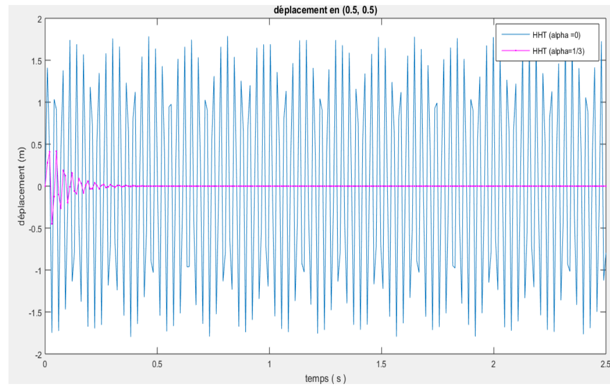
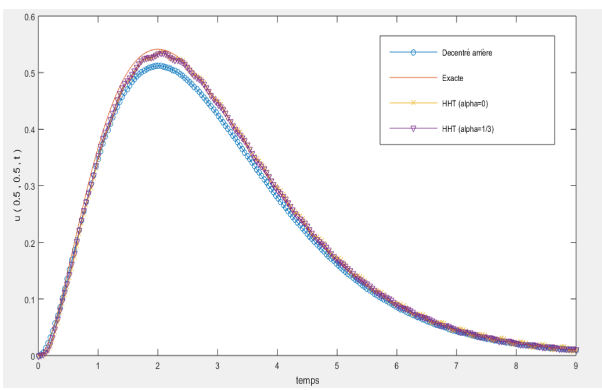
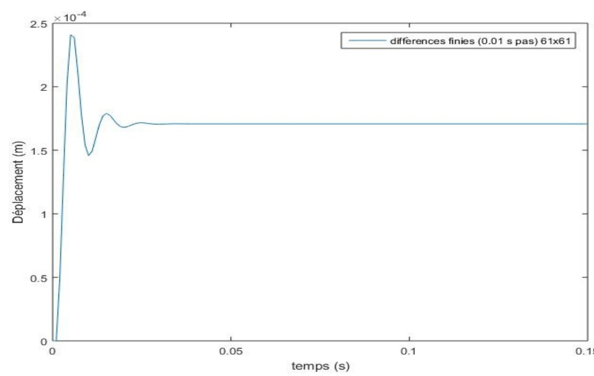
6 Numerical example

A numerical example was done on the problem 4, with the function ue. **Fig** 6 (c) shows the approximate solution with the two methods : HHT with two different values for the parameter α, 0 and 1/3, and the backward Euler scheme, the three results are compared with the exact solution in $(0.5, 0.5)$.

The approximations used a time step of $0.15s$ and a mesh of $11x11$ nodes. There is very insignificant difference between the 3 solutions, and the backward Euler scheme shows a slightly bigger error that decreases by further refining the mesh.

In order to highlight the efficacy of the HHT method, which lies in its ability to control the numerical damping with only the α parameter, while keeping a second order accuracy, a thin steel $[0, 1]x[0, 1]$ plate is considered, it has a $1cm$ thickness, and loaded by a **3 Kpa** pressure P. This load generates plate vibrations that represent the superposition of several vibration modes, the numerical damping enables the dissipation of the response's magnitude according to the higher frequencies that don't have physical interpretations. **Fig** 6 (d) shows the effect of this damping introduced naturally by the backward Euler scheme before stabilizing at $1.7 * 10^{-4}$m.[4] **Fig** 6 (b) shows the impact due to changing the parameter α from 0 to $\frac{1}{3}$ on the numerical damping of the response. This solution was done on a $11x11$ mesh and a time step of $0.01s$, refining the mesh would reduce the approximate displacement's value towards the exact value. 6 (a) shows that an α value of 0.005 can reduce the responses' amplitude in the higher frequencies but not as strongly as for an α value of $\frac{1}{3}$. Increasing α enables more numerical damping.

[4]The static response given by *Ansys* is $2 * 10^{-4}$m.

Displacement of the point (0.5,0.5)	
FIG 3 (a)	FIG 3 (b)
With a 31x31 mesh (α=1/3) 	Using two values 
FIG 3 (c)	FIG 3 (d)
Using different schemes 	using the backward Euler scheme 

7 Preconditioners for the resolution of the static problem

In this section, we treat the static version of 1,:

$$\boxed{D\Delta(\Delta u(\mathbf{x},t)) = g(\mathbf{x},t) \quad , \ \forall(\mathbf{x},t) \in \Omega[0,\infty[} \tag{19}$$

- Initial conditions :

$$u(\mathbf{x},0) = 0 \quad , \quad \forall \mathbf{x} \in \Omega \tag{20}$$

- Boundary conditions : Pinned

$$u(\mathbf{x},t) = M\,(\mathbf{x},t) = 0 \qquad , \qquad \forall(\mathbf{x},t) \in \partial\Omega[0,\infty[\tag{21}$$

where $M(\mathbf{x},t)$ is the bending moment.

$$M(\mathbf{x}) = \frac{\partial^2 u}{\partial x^2} + \nu\frac{\partial^2 u}{\partial y^2} \tag{22}$$

with ν the Poisson coefficient.
The mixed form for this static problem leads to the static form of 13, into a linear system of the form:

$$\begin{bmatrix} \mathbf{M} & D.\mathbf{K} \\ \mathbf{K^T} & \mathbf{0} \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ \vdots \\ w_N \\ u_1 \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \int_\Omega (g^n \varphi_1) \\ \vdots \\ \int_\Omega (g^n \varphi_N) \end{bmatrix} \tag{23}$$

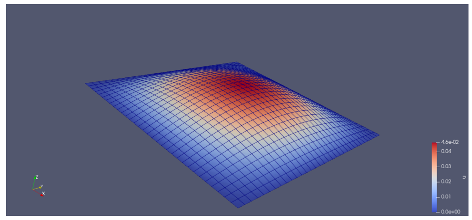
where

$$w(\mathbf{x}) = \Delta v(\mathbf{x}) \tag{24}$$

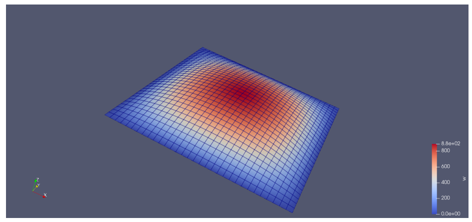
We should mention that, we are studying here the pinned problem (where the bending moment $M = 0$ at the boundaries) instead of the fixed boundaries problem, as the former enables us to have a simple Dirichlet problem ($u = 0$ and $w = 0$ at the boundaries), whether the fixed borders would require more complicated methods, since there is no Dirichlet condition for the w variable, it would need special elements like the discontinuous Galerkin and some other mathematical tricks[11].
We use the quadratic Lagrange finite elements in order to be able to retrieve the second derivative and thus the bending moment field in the plate.
We implemented the problem in the $C++$ Open Source software *deal.ii*[12, 13] At first we solve 23 by the UMFPACK direct solver, obtaining the results showed in **Fig.** 3

We can alter the system 23 to obtain :

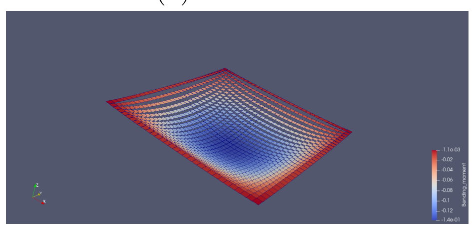
$$\begin{bmatrix} \frac{1}{D}\mathbf{M} & \mathbf{K} \\ \mathbf{K^T} & \mathbf{0} \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ \vdots \\ w_N \\ u_1 \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \int_\Omega (g^n \varphi_1) \\ \vdots \\ \int_\Omega (g^n \varphi_N) \end{bmatrix} \tag{25}$$



(a) Displacement solution



(b) w solution



(c) Bending moment solution

Figure 3: Solution obtained by deal.ii

The benefit of using this system is showing some preconditioners to solve this system, as the iterative solvers fail for this case because the left hand matrix has zeros in the diagonal. We should mention that the lower right block is not entirely zero as it contains coefficients due to the implementation of Dirichlet conditions globally (elements on the diagonal), we note this block **BC**.
From 25, if we multiply the first row block by $K^T M$ and substract the second row we have:

$$(K^T M^{-1} K - BC)U = -G \tag{26}$$

$$MW = -KU \tag{27}$$

with $W_i = w_i$, $U_i = u_i$ and $G_i = \int_\Omega (g^n \varphi_1)$.
The matrix $S = (K^T M^{-1} K - BC)$ is called the *Schur complement of the left hand matrix.* $K^T M^{-1} K$ is a positive definite matrix but S is not necessarily.
Here we consider S to be a positive-definite matrix (which we verify empirically). The simplest way to ensure that is to implement the Dirichlet conditions naturally in the weak form in order to guarantee that the lower right block is entirely zero.
To solve the system using the conjugate gradient method, we use

$$\tilde{S}^{-1} = [K^T (diag M^{-1}) K]^{-1} \tag{28}$$

as a preconditioner with only inversing the diagonal of M as a Jacobi preconditioner, we show in the **Figure** 4 the L_2 error between the solution obtained using the UMFPACK solver and the one using the conjugate gradient with the block matrices.

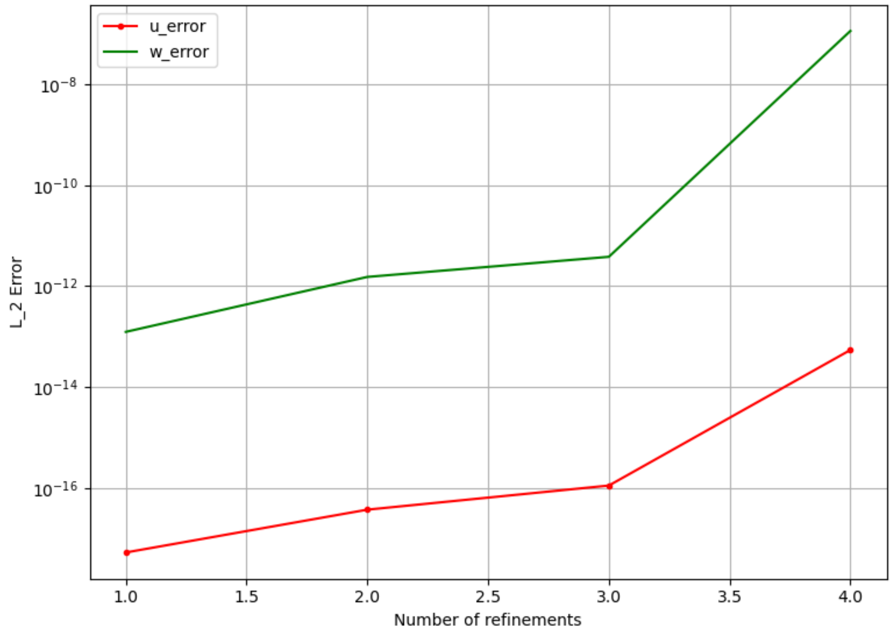


Figure 4: L_2 error between the UMFPACK solution and the conjugate gradient solution

8 Conclusion

The solution of the plate bending problem was solved using the finite element method. A mixed form was used in order to avoid using complicated finite elements, and only use the quadratic Lagrange finite elements. Implicit time integration schemes were used to benefit from their stability properties. The HHT method allows enhanced properties regarding the numerical damping and the accuracy of the discrete solution in comparison with the backward Euler scheme. However, the algorithm is slow and computationally expensive as it needs a fine mesh and a small time step. A study should be done to optimise the computation cost in regards to using nonconforming finite elements.
The static solution presented shows the low accuracy of the block matrices solution and algorithms with Dirichlet conditions implemented in the weak form should be implemented to insure having a positive definite Schur complement.

Bibliography

[1] E. Wilson. A computer program for the dynamic stress analysis of underground structures. 1968.

[2] Klaus-Jürgen Bathe and Gunwoo Noh. Insight into an implicit time integration scheme for structural dynamics. *Computers Structures*, 98-99:1–6, 2012.

[3] Nathan M. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division, ASCE*, 85(EM 3), 1959.

[4] Gregory M. Hulbert and Jintai Chung. Explicit time integration algorithms for structural dynamics with optimal numerical dissipation. *Computer Methods in Applied Mechanics and Engineering*, 137(2):175–188, 1996.

[5] Hans M. Hilber, Thomas J. R. Hughes, and Robert L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 5(3):283–292, 1977.

[6] Chapter 6 - finite element methods for the plate problem. In Philippe G. Ciarlet, editor, *The Finite Element Method for Elliptic Problems*, volume 4 of *Studies in Mathematics and Its Applications*, pages 333–380. Elsevier, 1978.

[7] P. G. Ciarlet and P. A. Raviart. A mixed finite element method for the biharmonic equation. Math. Aspects finite Elem. partial Differ. Equat., Proc. Symp. Madison 1974, 125-145 (1974)., 1974.

[8] Xiao liang Cheng, Weimin Han, and Hong ci Huang. Some mixed finite element methods for biharmonic equation. *Journal of Computational and Applied Mathematics*, 126(1):91–109, 2000.

[9] D. N. Arnold, F. Brezzi, and M. Fortin. A stable finite element for the stokes equations. 21:337 – 344, 1984.

[10] Ivo Babuška. The finite element method with lagrangian multipliers. *Numer. Math.*, 20(3):179–192, June 1973.

[11] Susanne Brenner and Li-yeng Sung. C0 interior penalty methods for fourth order elliptic boundary value problems on polygonal domains. *Journal of Scientific Computing*, 22-23:83–118, 06 2005.

[12] Daniel Arndt, Wolfgang Bangerth, Bruno Blais, Thomas C. Clevenger, Marc Fehling, Alexander V. Grayver, Timo Heister, Luca Heltai, Martin Kronbichler, Matthias Maier, Peter Munch, Jean-Paul Pelteret, Reza Rastak, Ignacio Thomas, Bruno Turcksin, Zhuoran Wang, and David Wells. The `deal.II` library, version 9.2. *Journal of Numerical Mathematics*, 28(3):131–146, 2020.

[13] Daniel Arndt, Wolfgang Bangerth, Denis Davydov, Timo Heister, Luca Heltai, Martin Kronbichler, Matthias Maier, Jean-Paul Pelteret, Bruno Turcksin, and David Wells. The deal.II finite element library: Design, features, and insights. *Computers & Mathematics with Applications*, 2021.