



Titre: Étude de méthodes d'intégration temporelle basées sur des schémas de corrections différées d'ordres élevés et A-stables
Title:

Auteur: Simon Berthelin
Author:

Date: 2022

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Berthelin, S. (2022). Étude de méthodes d'intégration temporelle basées sur des schémas de corrections différées d'ordres élevés et A-stables [Master's thesis, Polytechnique Montréal]. PolyPublie. <https://publications.polymtl.ca/10489/>
Citation:

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/10489/>
PolyPublie URL:

Directeurs de recherche: André Garon, & Stéphane Étienne
Advisors:

Programme: Génie mécanique
Program:

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

**Étude de méthodes d'intégration temporelle basées sur des schémas de
corrections différées d'ordres élevés et A-stables**

SIMON BERTHELIN

Département de génie mécanique

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie mécanique

Août 2022

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Ce mémoire intitulé :

**Étude de méthodes d'intégration temporelle basées sur des schémas de
corrections différées d'ordres élevés et A-stables**

présenté par **Simon BERTHELIN**

en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

a été dûment accepté par le jury d'examen constitué de :

Sylvain TURENNE, président

André GARON, membre et directeur de recherche

Stephane ETIENNE, membre et codirecteur de recherche

André FORTIN, membre

REMERCIEMENTS

Je remercie tout d'abord André Garon et Stéphane Étienne, mes codirecteurs, pour l'encadrement de ma maîtrise. Merci plus particulièrement à André, pour sa pédagogie, les connaissances qu'il m'a apporté, mais aussi les croissants et les cafés au *Toi Moi et Café*. J'ai beaucoup appris tout en ayant vraiment passé deux années très agréables, l'ambiance de travail a toujours été au top.

Merci à Arthur, mon fidèle acolyte de bureau sans fenêtre. Il m'a énormément aidé dans la compréhension du code, du C++ ou même de Linux en général. Merci pour le temps qu'il m'a accordé sur ce mémoire.

Merci à Yves Bourgault, de l'Université d'Ottawa, pour les nombreuses discussions éclairantes que nous avons eu.

Merci également à Jérôme Vétel de m'avoir permis de faire l'expérience d'une charge de TD. Cela aura été extrêmement enrichissant.

Enfin je remercie ma famille et mes amis qui m'ont soutenu durant cette maîtrise. Même si une partie est loin, j'ai pu profiter de leur aide et je leur en suis reconnaissant.

RÉSUMÉ

Dans le domaine de la simulation numérique en mécanique des fluides, deux aspects sont à prendre en compte : la discrétisation spatiale et la discrétisation temporelle. Ces deux aspects vont de pair et influent tous deux sur la précision et le coût d'un calcul. Dans notre travail, nous nous intéressons à la discrétisation temporelle et plus particulièrement aux méthodes BDF – Backward Differentiations Formulae – car elles permettent d'effectuer des calculs stables dans des cas de problèmes raides dont font notamment partie les problèmes de Navier-Stokes. Mais ces méthodes sont stables pour des ordres de convergence de un et deux seulement. Pour augmenter l'ordre de convergence, nous construisons une méthode de correction différée, c'est-à-dire que nous modifions le schéma de discrétisation avec une correction. Cette correction est basée sur l'erreur de troncature propre à chaque BDF. Nous construisons ainsi une méthode que nous nommons DC5/BDF1 qui est un schéma BDF1 corrigé successivement quatre fois afin d'obtenir une méthode stable d'ordre cinq. De même, nous construisons une méthode DC5/BDF2, également à l'ordre cinq, mais basée cette fois sur le schéma BDF2. De plus nous construisons ces méthodes pour qu'elles soient fonctionnelles, c'est-à-dire stables et avec le bon taux de convergence, à pas de temps alterné. En effet le but est de faire un premier pas en direction d'un intégrateur permettant de fonctionner avec des pas de temps adaptatifs. Nous testons ensuite nos méthodes pour des EDO simples et des systèmes d'EDO. Nous vérifions que les taux de convergence sont bien atteints. Nous discutons de la raideur des équations ainsi que du pas de temps utilisé pour les simulations. Ensuite des tests sur des EDP en une dimension sont effectués pour comprendre l'influence des conditions limites sur nos intégrateurs. Nous remarquons alors que des pertes d'ordre peuvent se produire sur des corrections successives, et cela en fonction des conditions limites et en utilisant des pas de temps alternés. Pour remédier à ce problème, nous développons donc une méthode pour imposer différemment les conditions limites : la correction de Verwer. Précédemment nous utilisions des conditions de Dirichlet, le principe est maintenant d'imposer faiblement une équation différentielle du premier ordre aux bords de notre domaine. Pour le faire directement au sein des formes faibles, nous utilisons un multiplicateur de Lagrange. Cette correction permet de remédier au problème de perte d'ordre et fournit également des informations supplémentaires sur la simulation via le multiplicateur de Lagrange. Dans un problème de thermique, le multiplicateur correspond au flux de chaleur au bord. Dans un problème d'écoulement isotherme, le multiplicateur nous donne la résultante des forces aux frontières. Cette information est très utile dans le cadre d'une application aux Interactions Fluide-Structure, car cela évite le difficile calcul des forces en post-traitement. Nous fournis-

sons également des résultats en deux dimensions pour des problèmes de diffusion thermique et d'écoulements isothermes. Cela nous permet de discuter de l'effet de la correction de Verwer. Ces résultats nous permettent de valider nos méthodes. Enfin nous effectuons une simulation sur des tourbillons de Von Karman. Pour ce cas nous devons combiner nos intégrateurs DC/BDF1 et DC/BDF2 afin d'initialiser correctement la méthode DC/BDF2. Les résultats obtenus comparés à ceux de la littérature nous permettent d'apporter un élément de vérification à nos méthodes.

ABSTRACT

There are two main aspects of fluid numerical simulations : spatial discretization and temporal discretization. These two aspects go hand in hand and have an influence on the accuracy and the cost of a computation. We will focus our work on the temporal discretization and more specifically on the BDF – Backward Differentiation Formulae – methods because it allows stable computations for stiff problems. These problems are very sensitive to fluctuations of parameters and so can be unstable, Navier-Stokes problems correspond to this description. But the BDF methods are stable only for order one and two. So we build a new method that use a deferred correction to increase the order of convergence of the BDF. This correction is based on the truncation error of each BDF method. Thus we develop a method, named DC5/BDF1, where we successively correct a BDF1 method to get a stable method of order 5. By the same process we build a DC5/BDF2, also of order 5 but based on the BDF2. Besides we think these schemes in a way that it also works with alternate time-step. Indeed we open the possibility to improve our method with a adaptative time-step algorithm.

We test our methods with simple ODE and system of ODE. We check that we reach the expected convergence order. And we discuss about the impact of the stiffness and the variability of the time-step. We also test with PDE to understand the influence of the boundary conditions and of the time-step on the result. We note that alternate time step and specific boundary conditions can lead to an order reduction. So we also develop a different way to implement the boundary conditions inspired by the Verwer correction. Previously we used Dirichlet boundary condition, now the idea is to impose a differential equation of order one on the boundary. It is possible to do it directly into the weak form of the problem using a Lagrange multiplier. Using the Lagrange multiplier has the double benefit to solve the order reduction and to provide more information during the computation. The Lagrange multiplier correspond to the heat flux in a heat and mass transfer problem and it gives us the sum of forces on a bound in a Navier-Stokes problem. This information is very useful in Fluid Structure Interaction problem because it will avoid the difficult calculation of forces in post treatment. To have a full set test we simulate a heat and mass transfer problem and Navier Stokes problem in two dimensions. It is the occasion to talk about the effect of Verwer’s correction. Finally, to validate our scheme we simulate Von Karman vortex. For this particular situation, the DC5/BDF2 method will require the DC5/BDF1 method for the initialization. By comparing our result to others in the literature, we can validate our methods.

TABLE DES MATIÈRES

REMERCIEMENTS	iii
RÉSUMÉ	iv
ABSTRACT	vi
TABLE DES MATIÈRES	vii
LISTE DES TABLEAUX	x
LISTE DES FIGURES	xiv
LISTE DES SIGLES ET ABRÉVIATIONS	xvi
LISTE DES ANNEXES	xvii
CHAPITRE 1 INTRODUCTION	1
1.1 Contexte	1
1.2 Objectifs de recherche	2
1.3 Plan du mémoire	2
CHAPITRE 2 REVUE DE LITTÉRATURE	3
CHAPITRE 3 MÉTHODES BDF	5
3.1 Construction des schémas BDF à pas constants par utilisation de l'erreur de troncature	5
3.1.1 Erreur de troncature du BDF1 à pas constant	6
3.1.2 Augmentation de l'ordre de la méthode	7
3.2 Méthodes BDF d'ordre p à pas variables	8
3.2.1 Cas du BDF1	10
3.2.2 Cas du BDF2	10
3.3 Stabilité des méthodes BDF	10
CHAPITRE 4 MÉTHODES À CORRECTIONS DIFFERÉES	14
4.1 Construction des méthodes DC/BDF1	14
4.1.1 Méthode DC2/BDF1	14

4.1.2	Méthode DC3/BDF1	16
4.1.3	Méthodes DC/BDF1 d'ordres supérieurs	18
4.1.4	Démarrage de la méthode DC3/BDF1	21
4.2	Méthodes DC/BDF2	22
4.2.1	Erreur de troncature du BDF2	23
4.2.2	DC3/BDF2	25
4.2.3	DC4/BDF2 et DC5/BDF2	26
4.3	Utilisation de DC/BDF1 pour démarrer DC/BDF2	27
CHAPITRE 5 IMPOSITION DES CONDITIONS LIMITES ET PERTE D'ORDRE		29
5.1	Perte d'ordre	29
5.2	Imposition faible des conditions limites et correction de Verwer	30
5.3	Interprétation du multiplicateur	31
CHAPITRE 6 VERIFICATION POUR DES EDO		32
6.0.1	Solutions manufacturées	32
6.0.2	Norme de l'erreur	32
6.0.3	Initialisation des méthodes BDF	33
6.0.4	Choix du pas de temps	33
6.1	Pas de temps constant	33
6.1.1	Exemples à pas de temps alternés	37
6.1.2	Système de deux EDO	38
6.1.3	Système de deux EDO raides	39
CHAPITRE 7 VERIFICATION POUR DES EDP EN 1D		42
7.1	Erreurs de discrétisation	42
7.2	Tests à pas de temps constants	44
7.2.1	Conditions limites indépendantes du temps	45
7.2.2	Conditions limites dépendantes du temps	45
7.3	Tests à pas de temps alternés	46
7.3.1	Conditions limites indépendantes du temps	46
7.3.2	Conditions limites dépendantes du temps	46
7.4	Correction de Verwer	48
7.5	Initialisation du DC/BDF2 à l'aide du DC/BDF1	51
CHAPITRE 8 VERIFICATION ET VALIDATION DES EDP EN 2D		54
8.1	Équation de la chaleur en 2D	54

8.1.1	Exclusion des angles dans la géométrie	57
8.1.2	Interprétation du multiplicateur	58
8.2	Écoulements isothermes en 2D	58
8.2.1	Paramètres du solveur	59
8.2.2	Simulation à pas de temps constants	60
8.2.3	Simulation à pas de temps alternés	61
8.3	Allées de von Kármán	62
8.3.1	Paramètres des simulations	62
8.3.2	Étude de raffinement temporelle	68
8.3.3	Résultats	72
CHAPITRE 9 CONCLUSION		74
9.1	Synthèse	74
9.2	Limites de nos travaux	76
9.3	Améliorations futures et recommandations	77
RÉFÉRENCES		79
ANNEXES		82

LISTE DES TABLEAUX

Tableau 3.1	Table des différences divisées basée sur la solution u	8
Tableau 4.1	Table des différences divisées basée sur $F(t, u_1)$	15
Tableau 4.2	Table des différences divisées basée sur le résidu à l'ordre 3 F_3	18
Tableau 4.3	Table de différences divisées basée sur $F(t, u_2)$	25
Tableau 6.1	Solutions exactes et termes sources pour les tests DC/BDF appliqués à une EDO	33
Tableau 6.2	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF1 pour la solution $u_1(t) = t^2$ à pas de temps constants	34
Tableau 6.3	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF2 pour la solution $u_1(t) = t^2$ à pas de temps constants	34
Tableau 6.4	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF1 pour la solution $u_2(t) = t^4$ à pas de temps constants	34
Tableau 6.5	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF2 pour la solution $u_2(t) = t^4$ à pas de temps constants	35
Tableau 6.6	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF1 pour la solution $u_3(t) = e^t$ à pas de temps constants	35
Tableau 6.7	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF2 pour la solution $u_3(t) = e^t$ à pas de temps constants	35
Tableau 6.8	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF1 pour la solution $u_4(t) = -\frac{1}{t+1}$ à pas de temps constants	36
Tableau 6.9	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF2 pour la solution $u_4(t) = -\frac{1}{t+1}$ à pas de temps constants	36
Tableau 6.10	Ordre de convergence de la méthode DC/BDF1 à pas constants. "inté" signifie que la méthode intègre parfaitement l'équation.	36
Tableau 6.11	Ordre de convergence de la méthode DC/BDF2 à pas constants. "inté" signifie que la méthode intègre parfaitement l'équation.	37
Tableau 6.12	Erreur en norme $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF1 pour la solution $u_3(t) = e^t$ à pas de temps alternés	37
Tableau 6.13	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF2 pour la solution $u_3(t) = e^t$ à pas de temps alternés	37
Tableau 6.14	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du ressort à pas constants	39

Tableau 6.15	Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du ressort à pas constants	39
Tableau 6.16	Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système ressort pour une simulation à pas de temps constants avec un coefficient de raideur de 1000	40
Tableau 6.17	Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système ressort pour une simulation à pas de temps constants avec un coefficient de raideur de 1000	40
Tableau 7.1	Solutions manufacturées dont les conditions limites dépendent ou non du temps	44
Tableau 7.2	Erreur et ordre de convergence de la méthode DC/BDF1 à pas de temps constants pour la solution \mathbf{u}_1 donc à condition limites indépendantes du temps.	45
Tableau 7.3	Erreur et ordre de convergence de la méthode DC/BDF2 à pas de temps constants pour la solution \mathbf{u}_1 donc à condition limites indépendantes du temps.	45
Tableau 7.4	Erreur et ordre de convergence de la méthode DC/BDF1 à pas de temps constants pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps.	45
Tableau 7.5	Erreur et ordre de convergence de la méthode DC/BDF2 a pas de temps constant pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps.	45
Tableau 7.6	Erreur et ordre de convergence de la méthode DC/BDF1 à pas de temps alternés pour la solution \mathbf{u}_1 donc à condition limites indépendantes du temps.	46
Tableau 7.7	Erreur et ordre de convergence de la méthode DC/BDF2 à pas de temps alternés pour la solution \mathbf{u}_1 donc à condition limites indépendantes du temps.	46
Tableau 7.8	Erreur et ordre de convergence de la méthode DC/BDF1 à pas de temps alternés pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps.	47
Tableau 7.9	Erreur et ordre de convergence de la méthode DC/BDF2 à pas de temps alternés pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps.	47

Tableau 7.10	Erreur et ordre de convergence de la méthode DC/BDF2 à pas de temps constants pour la solution manufacturée \mathbf{u}_2 donc à condition limites dépendantes du temps avec la méthode de Verwer.	49
Tableau 7.11	Erreur et ordre de convergence de la méthode DC/BDF2 à pas de temps alternés pour la solution manufacturée \mathbf{u}_2 donc à condition limites dépendantes du temps avec la méthode de Verwer.	49
Tableau 7.12	Erreur et ordre de convergence de la méthode DC/BDF1 à pas de temps alternés pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps avec la méthode de Verwer.	50
Tableau 7.13	Taux de convergence des méthodes DC/BDF1 et DC/BDF2 pour l'équation de la chaleur en une dimension spatiale.	51
Tableau 7.14	Erreur et ordre de convergence de la méthode DC/BDF2 à pas de temps constants pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps avec méthode de Verwer et avec initialisation DC/BDF1	52
Tableau 7.15	Erreur et ordre de convergence de la méthode DC/BDF2 à pas de temps alternés pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps avec méthode de Verwer et avec initialisation DC/BDF1	52
Tableau 8.1	Erreurs et ordres de convergence de la méthode DC/BDF2 à pas de temps constant par rapport à la solution (8.2) sur un carré. Mesure de l'erreur sur λ en norme L^2	56
Tableau 8.2	Erreurs et ordres de convergence de la méthode DC/BDF2 à pas de temps constants par rapport à la solution (8.2) sur un carré. Mesure de l'erreur sur λ en norme L^2 en excluant les coins	57
Tableau 8.3	Erreurs et ordres de convergence de la méthode DC/BDF2 à pas de temps constants par rapport à la solution 8.2 sur un carré. Mesure de l'erreur de l'intégrale de λ	58
Tableau 8.4	Erreurs et ordres de convergence de la méthode DC/BDF2 à pas de temps constants pour le problème (8.5) sur un carré. Mesure de l'erreur de l'intégrale de λ_u	61
Tableau 8.5	Erreurs et ordres de convergence de la méthode DC/BDF2 à pas de temps alternés pour le problème 8.5 sur un carré. Mesure de l'erreur de l'intégrale de λ_u	61

Tableau 8.6	Données comparatives pour les allées de von Kármán	73
Tableau A.1	Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système ressort pour une simulation à pas de temps alternés	82
Tableau A.2	Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système ressort pour une simulation à pas de temps alternés	82
Tableau B.1	Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système matriciel à pas constants	83
Tableau B.2	Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système matriciel à pas constants	84
Tableau B.3	Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système matriciel pour une simulation à pas de temps alternés	84
Tableau B.4	Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système matriciel pour une simulation à pas de temps alternés	84
Tableau B.5	Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système de 3 EDO pour une simulation à pas de temps constants avec un coefficient de raideur de 100	85
Tableau B.6	Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système de 3 EDO pour une simulation à pas de temps constants avec un coefficient de raideur de 100	86

LISTE DES FIGURES

Figure 3.1	Régions de stabilité des méthodes BDF1, BDF2 et BDF3 (en vert)	12
Figure 4.1	Principe de l'initialisation de la première valeur pour la méthode DC3/BDF1	22
Figure 4.2	Utilisation de la méthode DC/BDF1 pour initialiser DC/BDF2	28
Figure 7.1	Répartition de l'erreur sur le domaine pour plusieurs de pas de temps constants	47
Figure 7.2	Répartition de l'erreur sur le domaine pour plusieurs de pas de temps alternés	48
Figure 8.1	Maillages de la géométrie carré composés de 40, 160, 640 et 2560 éléments	55
Figure 8.2	Champ de température pour la solution (8.2) dans le cas d'un maillage de 40 et de 2560 éléments	56
Figure 8.3	Flux numériques et théorique sur le coin d'un carré	57
Figure 8.4	Champ de vitesses dans le cas du problème (8.5)	60
Figure 8.5	Géométrie utilisée pour générer des allées de von Kármán . . .	62
Figure 8.6	Champ de vitesse horizontale imposé initialement afin de provoquer des instabilités	63
Figure 8.7	Maillage de 93338 éléments utilisé pour capturer les allées de von Kármán	64
Figure 8.8	Vitesses horizontale u , verticale v et norme du vecteur vitesse $ \vec{u} $ en aval du cercle pour un lâché de tourbillons de von Kármán. . .	66
Figure 8.9	Évolution du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Simulation avec un pas de temps de 0.03125.	67
Figure 8.10	Évolution du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Simulation avec un pas de temps de 0.03125.	68
Figure 8.11	Évolution du coefficient de traînée en fonction du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Méthode BDF2 pour plusieurs pas de temps.	69

Figure 8.12	Évolution du coefficient de traînée en fonction du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Méthode DC5/BDF2 pour plusieurs pas de temps.	70
Figure 8.13	Évolution du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Simulation pour un pas de temps de 0.03125 pour les différentes méthodes DC/BDF2.	71
Figure 8.14	Erreurs d'approximation sur le multiplicateur en fonction du nombre de pas de temps pour les méthodes DC/BDF2.	72

LISTE DES SIGLES ET ABRÉVIATIONS

IFS	Interaction Fluide-Structure
BDF	Backward differentiations formulae
DC	Deferred correction
EDO	Équations aux dérivées ordinaires
EDP	Équations aux dérivées partielles
ODE	Ordinary differential equation
PDE	Partial differential equation
IVP	Initial value problem
BVP	Boundary value problem

LISTE DES ANNEXES

Annexe A	Résultats complémentaires pour des systèmes matriciels en 0D à pas alternés	82
Annexe B	Résultats complémentaires sur un système de trois équations	83

CHAPITRE 1 INTRODUCTION

En ingénierie, l'étude de la mécanique des fluides soulève de nombreux problèmes. Une majeure partie d'entre eux sont maintenant résolus numériquement, mais il existe autant de manières de résoudre numériquement un problème qu'il existe de situations différentes. Dans notre cas, nous allons nous intéresser à des problèmes d'interactions fluide-structure (IFS). Dans ce type de situation, le fluide a un impact sur le mouvement de la structure et, inversement, le mouvement de la structure modifie l'écoulement du fluide. Les structures de plateforme pétrolière en mer, les cheminées d'usine ou les systèmes vasculaires constituent de très bons exemples d'interactions fluide-structure. Ces problèmes sont souvent considérés comme raides, c'est-à-dire qu'une faible modification d'un paramètre influence fortement le résultat. En particulier, les problèmes d'IFS posent problèmes, car de nombreuses méthodes numériques sont instables lors des simulations.

1.1 Contexte

Lors d'une simulation numérique, plusieurs aspects sont à prendre en compte et influent sur la précision et le coût du calcul. La discrétisation spatiale et le choix du maillage sont primordiaux. De notre côté, nous nous intéresserons à l'aspect temporel, avec la discrétisation temporelle et le choix d'intégrateur. Pour répondre aux enjeux soulevés par les problèmes raides, il existe de nombreuses approches regroupées en deux familles de méthodes distinctes. Les méthodes explicites, parfaitement illustrées par la méthode d'Euler explicite, sont simples à mettre en place, mais présentent des conditions à respecter afin d'en assurer la stabilité. Elles ne seront donc pas considérées dans notre étude. À l'inverse, les méthodes implicites sont généralement plus stables, mais la résolution du système peut être non linéaire.

Nous nous concentrerons par la suite sur les méthodes implicites, car nous cherchons à produire des méthodes inconditionnellement stables. Parmi ces méthodes, nous pouvons citer les classiques Runge-Kutta implicites qui peuvent être d'ordres élevés, mais qui sont lourdes à mettre en œuvre. Nous nous intéressons également aux méthodes BDF, pour *Backward Differentiation Formulae*, dont BDF1 qui correspond en fait à un schéma d'Euler implicite ou BDF2 aussi connu sous le nom de la méthode de Gear.

1.2 Objectifs de recherche

Les méthodes BDF sont réputées pour être adaptées aux problèmes raides. Nous allons donc nous concentrer sur ces schémas. Cependant, nous verrons qu'elles sont inconditionnellement stables pour les méthodes d'ordre 1 et 2 et qu'elles deviennent conditionnellement stables à partir de l'ordre 3.

Notre but sera donc de **construire des méthodes de corrections différées issues des schémas BDF1 et BDF2 pour en augmenter la précision (l'ordre) tout en préservant la stabilité** dans le cas de problèmes raides. En outre, en vue d'adapter le pas de temps, nous développerons également des méthodes fonctionnelles et stables avec des pas de temps alternés. Cela permettra d'ouvrir la voie pour des schémas à pas de temps adaptatifs.

Nous mettrons en œuvre ces schémas dans un code d'éléments finis développé au sein du laboratoire. L'un des objectifs sera donc également d'adapter la théorie de nos méthodes à la pratique des éléments finis.

1.3 Plan du mémoire

Nous étudierons tout d'abord les propriétés des schémas BDF pour en extraire les informations nécessaires menant à la construction de corrections différées stables et d'ordres élevés. Nous aborderons également des éléments théoriques concernant la manière d'imposer les conditions limites d'un problème dans le cadre des éléments finis. Par la suite, nous verrons le comportement de nos schémas pour des EDO. Puis nous présenterons les résultats de simulations dans une situation unidimensionnelle, ce qui constituera des situations d'EDP. Enfin, nous étudierons plusieurs équations de Navier-Stokes et de thermique en deux dimensions, pour vérifier et valider nos méthodes. Nous terminerons par la présentation de nos résultats, de nos observations et par les questions à approfondir concernant notre travail.

CHAPITRE 2 REVUE DE LITTÉRATURE

Les méthodes d'intégration temporelle sont une composante essentielle de la simulation en mécanique des fluides, c'est donc naturellement que de très nombreux auteurs se sont penchés sur le sujet. Elles impactent directement la précision et donc le coût du calcul. Plusieurs stratégies existent afin d'assurer la précision, tout en conservant la stabilité, lors d'une simulation. Une des approches couramment utilisées consiste à faire varier le pas de temps pour s'adapter à une erreur cible. Birken et al. [1] proposent ainsi une méthode à pas de temps adaptatif permettant de contrôler l'erreur d'intégration temporelle. C'est également le cas de Mayr et al. qui se concentrent sur une erreur a posteriori [2]. De la même manière, John et Rang [3] présentent plusieurs algorithmes d'adaptation du pas de temps. L'ensemble de ces auteurs insistent sur la nécessité d'utiliser ces méthodes pour simuler les équations de Navier-Stokes ou dans le cas de problèmes d'interactions fluide-structure. Parmi les schémas numériques classiques, nous pouvons mettre l'accent sur les méthodes BDF (Backward Differentiations Formulae) très fréquemment utilisées en raison de leur simplicité de mise en place. Ces méthodes sont la plupart du temps utilisées à l'ordre 2, car elles sont, au-delà, sujettes aux instabilités. Autour de cette problématique, des auteurs ont donc développé au cours des années 1980, des méthodes à pas de temps adaptatifs spécifiques aux BDF [4, 5]. Skelboe et al. [6] ont développé un test d'instabilité des BDF d'ordre élevé afin de pouvoir appliquer ces schémas à des équations raides.

Une approche différente concerne la correction différée. Le principe est d'augmenter l'ordre de convergence d'une méthode en la modifiant par une correction bien choisie. Cette idée remonte à la fin des années cinquante avec Fox qui s'intéresse à la correction de l'intégrateur numérique dans le cas d'une méthode de relaxation [7] ou linéaire [8, 9]. Par la suite, Peyrera [10] et Setter [11] ont poursuivi selon cette démarche et montrent de nouvelles façons de corriger les problèmes aux conditions initiales (Initial Value Problem) ou les problèmes aux conditions limites (Boundary Value Problem). Plus récemment encore, les auteurs cherchent à améliorer la précision et la rapidité du calcul. Alors Dutt et al. [12] explorent la possibilité d'une correction basée sur l'intégrale de Picard. De même Minion et al. [13, 14] propose une approche différente de la correction basée sur des éléments spectraux. Citons également le travail de Kress [15] et Gustafsson [16] qui ont développé des corrections différées pour des schémas de Crank-Nicolson. Dans leur travail, cette méthode d'intégration temporelle est utilisée conjointement avec une discrétisation spatiale par différences finies. Dans notre travail, nous adapterons ces travaux pour une discrétisation spatiale par la méthode des éléments finis.

De notre côté, nous baserons notre travail sur des méthodes BDF. Dans ce domaine, Geng [17] est le premier à mettre en œuvre en 1985 une correction pour des méthodes BDF multi-pas. Cependant, ces méthodes sont encore à pas constants. Au vu de notre objectif, notons également les travaux de Guermond et Mineev [18] qui utilisent des méthodes BDF pour des écoulements modélisés par les équations de Navier-Stokes.

CHAPITRE 3 MÉTHODES BDF

Lors d'une simulation numérique, les enjeux portent sur le contrôle simultané de la stabilité et la précision du calcul. En effet, en considérant un coût de calcul constant, la maximisation de la précision se fait généralement au détriment de la stabilité. Pour les méthodes d'intégration temporelle usuelles [19, 20], il est souvent impossible de satisfaire simultanément ces deux enjeux. Par exemple, il est démontré que les méthodes d'intégration aux différences divisées sont conditionnellement stables au-delà de l'ordre 2¹. Dans ce contexte, des techniques de mitigation sont utilisées pour minimiser le coût du calcul en modifiant simultanément l'ordre de la méthode d'intégration ainsi que son pas de temps à l'aide d'indicateurs de stabilité [21]. Malheureusement, ces méthodes de contrôle se basent sur des heuristiques, dont l'universalité est discutable. Notre but est donc d'explorer la construction de méthodes d'intégration inconditionnellement stables, et cela sans limite sur la précision du schéma. Nous allons maintenant détailler la construction des schémas BDF.

Notations Dans les développements qui vont suivre, nous utilisons les notations suivantes :

- $u(t)$ est la solution exacte,
- t^n est le temps discrétisé au n -ième pas de temps,
- u^n est la solution numérique obtenue au temps t^n ,
- u_p^n est la solution numérique d'ordre p au temps t^n ,
- F_p^n correspond au résidu calculé depuis la solution u_p^n ,
- $k_1 = t^n - t^{n-1}$ et $k_2 = t^{n-1} - t^{n-2}$,
- $\omega_n = \frac{t^n - t^{n-1}}{t^{n-1} - t^{n-2}}$ est le rapport entre deux pas de temps consécutifs,
- $\mathcal{E}_p(t^n)$ est l'erreur de troncature d'ordre p d'un schéma numérique.

3.1 Construction des schémas BDF à pas constants par utilisation de l'erreur de troncature

Soit l'équation différentielle ordinaire en la variable $u(t)$:

$$\frac{du}{dt}(t) = f(t, u(t)) . \tag{3.1}$$

Pour construire la méthode BDF, nous discrétisons la dérivée $\frac{du}{dt}$. Dans un premier temps, nous pouvons approximer cette dérivée par une formule aux différences divisées [19]. C'est la

1. Nous rappelons que plus l'ordre de discrétisation d'une méthode est élevé, plus celle-ci est précise.

méthode d'Euler implicite :

$$\frac{u^n - u^{n-1}}{k_1} = f(t^n, u^n) . \quad (3.2)$$

3.1.1 Erreur de troncature du BDF1 à pas constant

Pour obtenir ce terme d'erreur, nous remplaçons les solutions numériques u^n et u^{n-1} par la solution exacte évaluée aux temps t^n et t^{n-1} :

$$\frac{u(t^n) - u(t^{n-1})}{k_1} - f(t^n, u(t^n)) = \mathcal{E}(t^n) . \quad (3.3)$$

Pour exprimer l'erreur, nous cherchons à faire apparaître l'EDO (3.1) évaluée en t^n . Pour cela, nous développons le terme $u(t^{n-1})$ au voisinage de t^n à l'aide d'un développement en série de Taylor :

$$u(t^{n-1}) = u(t^n) - \frac{du}{dt}(t^n)k_1 + u''(t^n)\frac{k_1^2}{2!} - u'''(t^n)\frac{k_1^3}{3!} + \dots .$$

En divisant par le pas de temps, nous retrouvons bien l'EDO dont les termes s'annulent par définition. Ainsi, nous voyons apparaître le terme d'erreur \mathcal{E} .

$$\underbrace{\frac{du}{dt}(t^n) - f(t^n, u(t^n))}_0 - \underbrace{u''(t^n)\frac{k_1}{2!} + u'''(t^n)\frac{k_1^2}{3!} + \mathcal{O}(k_1^3)}_{\mathcal{E}} = \mathcal{E}(t^n) \quad (3.4)$$

L'erreur de troncature d'une méthode BDF1 est donc :

$$\boxed{\mathcal{E}(t^n) = -u''(t^n)\frac{k_1}{2!} + u'''(t^n)\frac{k_1^2}{3!} + \dots} . \quad (3.5)$$

Définition 1. Ordre de convergence

De manière générale, l'erreur de troncature d'un schéma d'intégration s'écrit sous la forme $\mathcal{E} = c_p k^p + c_{p+1} k^{p+1} + \dots$ avec les coefficients c_p différents de 0 et k le pas de temps. Lorsque le pas de temps k est suffisamment petit, l'erreur est dominée par le premier terme de la série, c'est-à-dire $c_p k^p$. On dit alors que la méthode est d'ordre p ou $\mathcal{O}(k^p)$.

Dans notre cas lorsque le pas de temps k_1 tend vers 0, le terme dominant est $u''(t^n)\frac{k_1}{2!}$. Pour un pas de temps assez petit, l'erreur est donc proportionnelle à k_1 . La méthode BDF1 est donc d'ordre 1.

3.1.2 Augmentation de l'ordre de la méthode

Pour augmenter l'ordre de la méthode d'Euler (3.2), une stratégie consiste donc à ajouter dans le schéma une expression de manière à annuler le terme d'ordre 1 dans l'erreur. Nous choisissons donc de soustraire au membre de gauche de l'équation (3.2) le premier terme de l'erreur de la relation (3.5). Dans notre situation cela donne :

$$\frac{u^n - u^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2!} = f(t^n, u^n) . \quad (3.6)$$

Pour vérifier nous pouvons recalculer l'erreur en reprenant le travail effectué à l'équation (3.4) :

$$\underbrace{\frac{du}{dt}(t^n) - f(t^n, u(t^n))}_{\emptyset} + \cancel{u''(t^n) \frac{k_1}{2!}} - \cancel{u''(t^n) \frac{k_1}{2!}} + \underbrace{u'''(t^n) \frac{k_1^2}{3!} + O(k_1^3)}_{\varepsilon} = \mathcal{E}(t^n). \quad (3.7)$$

Nous avons bien maintenant une erreur dominée par un terme d'ordre 2 : $\mathcal{E}_2(t^n) = u'''(t^n) \frac{k_1^2}{3!}$. Nous avons construit une méthode d'ordre 2. Cette correction est basée sur la solution exacte, nous devons trouver un succédané à cela. Nous cherchons une approximation permettant de modéliser la dérivée seconde exacte. Pour cela nous utilisons les résultats numériques aux pas de temps précédents u^n , u^{n-1} et u^{n-2} pour approcher la dérivée au temps t^n . Notons que cette approche est un choix ; nous verrons par la suite qu'il est possible d'approximer la dérivée de manière différente. Nous utilisons une table de différences divisées à pas constants (nous développerons la méthode générale dans la section 4.1.1) pour écrire l'approximation suivante :

$$\boxed{u''(t^n) = \frac{u^n - 2u^{n-1} + u^{n-2}}{2k_1} + \mathcal{O}(k_1)} . \quad (3.8)$$

En substituant cette approximation dans l'équation (3.6), nous obtenons :

$$\frac{3u^n - 4u^{n-1} + u^{n-2}}{2k_1} = f(t^n, u^n) . \quad (3.9)$$

De cette manière, nous venons de construire la méthode de Gear d'ordre 2 [22] à pas de temps constants. En effet, l'erreur est maintenant dominée par le terme $u'''(t^n) \frac{k_1^2}{3!}$ et l'ordre de convergence de l'erreur est de 2. De la même façon, nous pourrions construire un schéma à l'ordre 3 en intégrant le terme $u'''(t^n) \frac{k_1^2}{3!}$ à notre discrétisation. Nous aurions ainsi un BDF3.

Remarque 3.1.1. *Nous pouvons formaliser les écritures des schémas BDF1 et BDF2. En*

introduisant les coefficients c_i , nous pouvons écrire la discrétisation comme une somme des $p + 1$ solutions pondérées par des coefficients :

$$\frac{\partial}{\partial t}u(t^n) \approx \boxed{\sum_{i=0}^p c_i^n u^{n-i} = f(t^n, u^n)}. \quad (3.10)$$

Et nous remarquons alors que le nombre de solutions antérieures utilisées, p , correspond à l'ordre de convergence du schéma.

Nous avons ici décrit une manière d'utiliser l'erreur de troncature pour augmenter l'ordre des méthodes BDF à pas constant. Cette philosophie sera à la base de nos méthodes à corrections différées, il est donc important de comprendre le lien entre l'ordre de la méthode et l'erreur de troncature. Nous allons maintenant formaliser la construction des méthodes BDF dans les cas où les pas de temps sont variables.

3.2 Méthodes BDF d'ordre p à pas variables

Considérons de nouveau l'équation type (3.1). Un schéma BDF d'ordre p prend en compte les p solutions précédentes pour construire un polynôme d'interpolation adéquatement choisi. Nous pouvons l'écrire sous forme de somme :

$$\frac{\partial}{\partial t}u(t^n) \approx \boxed{\sum_{i=0}^p c_i^n u^{n-i} = f(t^n, u^n)}.$$

Nous cherchons à déterminer les coefficients c_i^n dans le cas d'un pas de temps non constant. Pour cela nous cherchons une manière alternative d'obtenir une approximation de la dérivée première $\frac{d}{dt}u(t^n)$ à l'aide des solutions antérieures. Tout d'abord, nous cherchons le polynôme d'interpolation des solutions numériques qui permet d'approximer la solution $u(t)$. Pour cela, nous exprimons la solution $u(t)$ à l'aide de la table 3.1 des différences divisées.

Tableau 3.1 Table des différences divisées basée sur la solution u

	t^n	u^n			
$k_1 = t^n - t^{n-1}$			δ_{11}		
	t^{n-1}	u^{n-1}		δ_{12}	
$k_2 = t^{n-1} - t^{n-2}$			δ_{21}		δ_{13}
	t^{n-2}	u^{n-2}		δ_{22}	
$k_3 = t^{n-2} - t^{n-3}$			δ_{31}		
	t^{n-3}	u^{n-3}			

Dans laquelle les coefficients sont les suivants :

$$\begin{aligned}
\delta_{11} &= \frac{u^{n-1} - u^n}{t^{n-1} - t^n}, \\
\delta_{12} &= \frac{\delta_{21} - \delta_{11}}{t^{n-2} - t^n} = \frac{\frac{u^{n-2} - u^{n-1}}{t^{n-2} - t^{n-1}} - \frac{u^{n-1} - u^n}{t^{n-1} - t^n}}{t^{n-2} - t^n}, \\
\delta_{13} &= \frac{\delta_{12} - \delta_{22}}{t^{n-3} - t^n}.
\end{aligned} \tag{3.11}$$

La solution $u(t)$ est approximée à l'aide du polynôme d'interpolation de Newton $P(t)$. Ce polynôme est de degré p et passe par $p + 1$ solutions.

$$\begin{aligned}
u(t) \approx P(t) &= u^n + \delta_{11}(t - t^n) + \delta_{12}(t - t^n)(t - t^{n-1}) \\
&+ \delta_{13}(t - t^n)(t - t^{n-1})(t - t^{n-2}) + \delta_{14}(t - t^n)(t - t^{n-1})(t - t^{n-2})(t - t^{n-3}) \\
&+ \dots \\
&= u^n + \sum_{i=1}^p \delta_{1i} \prod_{j=1}^i (t - t^{n-j+1})
\end{aligned} \tag{3.12}$$

Nous pouvons alors exprimer la dérivée de $P(t)$ et l'évaluer au temps t^n pour obtenir une expression approchée de $\frac{d}{dt}u(t^n)$.

$$\begin{aligned}
\frac{d}{dt}u(t^n) \approx \frac{d}{dt}P(t^n) &= \delta_{11} + \delta_{12}(t^n - t^{n-1}) \\
&+ \delta_{13}(t^n - t^{n-1})(t^n - t^{n-2}) + \delta_{14}(t^n - t^{n-1})(t^n - t^{n-2})(t^n - t^{n-3}) \\
&+ \dots \\
&= \delta_{11} + \sum_{i=2}^p \delta_{1i} \prod_{j=1}^{i-1} (t^n - t^{n-j})
\end{aligned} \tag{3.13}$$

Les expressions de la dérivée obtenues par la méthode BDF et par le polynôme de Newton sont équivalentes, on peut donc les comparer pour identifier les coefficients c_i^n :

$$\frac{d}{dt}u(t^n) \approx \delta_{11} + \sum_{i=2}^p \delta_{1i} \prod_{j=1}^{i-1} (t^n - t^{n-j}) \approx \sum_{i=0}^p c_i^n u^{n-i} \tag{3.14}$$

3.2.1 Cas du BDF1

Un schéma BDF à l'ordre 1 correspond à $p = 1$. Donc, en écrivant (3.14) avec les coefficients issus de la table de différence divisée 3.1, nous obtenons le résultat suivant :

$$\begin{aligned}\frac{d}{dt}u(t^n) &\approx \delta_{11} = c_0^n u^n + c_1^n u^{n-1}, \\ \frac{u^{n-1} - u^n}{t^{n-1} - t^n} &= c_0^n u^n + c_1^n u^{n-1}.\end{aligned}\tag{3.15}$$

Nous retrouvons bien la formule d'Euler avec les coefficients $c_0^n = \frac{1}{k_1}$ et $c_1^n = -\frac{1}{k_1}$.

3.2.2 Cas du BDF2

De la même façon, nous pouvons exprimer le BDF2 pour $p = 2$.

$$\begin{aligned}\frac{d}{dt}u(t^n) &\approx \delta_{11} + \delta_{12}(t^n - t^{n-1}) = c_0^n u^n + c_1^n u^{n-1} + c_2^n u^{n-2} \\ \frac{u^{n-1} - u^n}{t^{n-1} - t^n} + \frac{\frac{u^{n-2} - u^{n-1}}{t^{n-2} - t^{n-1}} - \frac{u^{n-1} - u^n}{t^{n-1} - t^n}}{t^{n-2} - t^n} (t^n - t^{n-1}) &= c_0^n u^n + c_1^n u^{n-1}\end{aligned}\tag{3.16}$$

Nous faisons apparaître les coefficients de la formule de Gear.

$$\begin{aligned}c_0^n &= \frac{1}{t^n - t^{n-1}} + \frac{1}{t^n - t^{n-2}} = \frac{1}{k_1} + \frac{1}{k_1 + k_2} \\ c_1^n &= \frac{1}{t^{n-1} - t^n} + \frac{1}{t^{n-2} - t^{n-1}} = -\frac{1}{k_1} - \frac{1}{k_2} \\ c_2^n &= \frac{t^n - t^{n-1}}{(t^{n-1} - t^{n-2})(t^n - t^{n-1})} = \frac{k_1}{k_2} \frac{1}{k_1 + k_2}\end{aligned}\tag{3.17}$$

Nous avons vu comment construire de manière formelle les méthodes BDF. Nous pourrions pousser cette démarche pour des méthodes d'ordres supérieurs, BDF3 ou BDF4, mais nous allons voir que ces méthodes ne remplissent pas les critères de stabilité qui nous intéressent. Nous ne travaillerons donc qu'à partir des schémas BDF1 et BDF2.

3.3 Stabilité des méthodes BDF

Les méthodes BDF1 et BDF2 sont inconditionnellement stables à pas constants [23]. Nous allons voir ce que cela signifie exactement. La stabilité est définie en étudiant le problème

suivant, aussi appelé équation test de Dahlquist² [20]. Dans ce problème, u représente une perturbation à la solution. Une méthode est stable si la perturbation décroît vers zéro :

$$\begin{aligned}\frac{du}{dt} &= \lambda u, \\ u(0) &= u_0.\end{aligned}\tag{3.18}$$

Avec $\lambda = Re(\lambda) + iIm(\lambda)$ un nombre complexe. Il nous faut déterminer les conditions sur λ pour que la solution converge. La convergence se traduit par :

$$\left| \frac{u^{n+1}}{u^n} \right| < 1.\tag{3.19}$$

Or, dans le cas d'un schéma BDF1, nous avons que :

$$u^n = u^{n+1} (1 - k\lambda).\tag{3.20}$$

Donc la condition de stabilité devient :

$$\left| \frac{u^{n+1}}{u^n} \right| = \left| \frac{1}{1 - k\lambda} \right| < 1.\tag{3.21}$$

En explicitant l'écriture complexe de λ nous obtenons alors la condition suivante :

$$(1 - kRe(\lambda))^2 + (kIm(\lambda))^2 > 1.\tag{3.22}$$

La zone d'instabilité sera donc comprise dans un cercle de rayon 1 centré en $(1, 0)$. Nous représentons ce résultat dans le plan $(kRe(\lambda), kIm(\lambda))$ ainsi, que les zones de stabilité et d'instabilité des méthodes BDF sur les figures 3.1.

2. Pour des problèmes non linéaires il est toujours possible de linéariser autour d'un point d'intérêt et donc de se rapporter à l'équation de Dahlquist linéaire

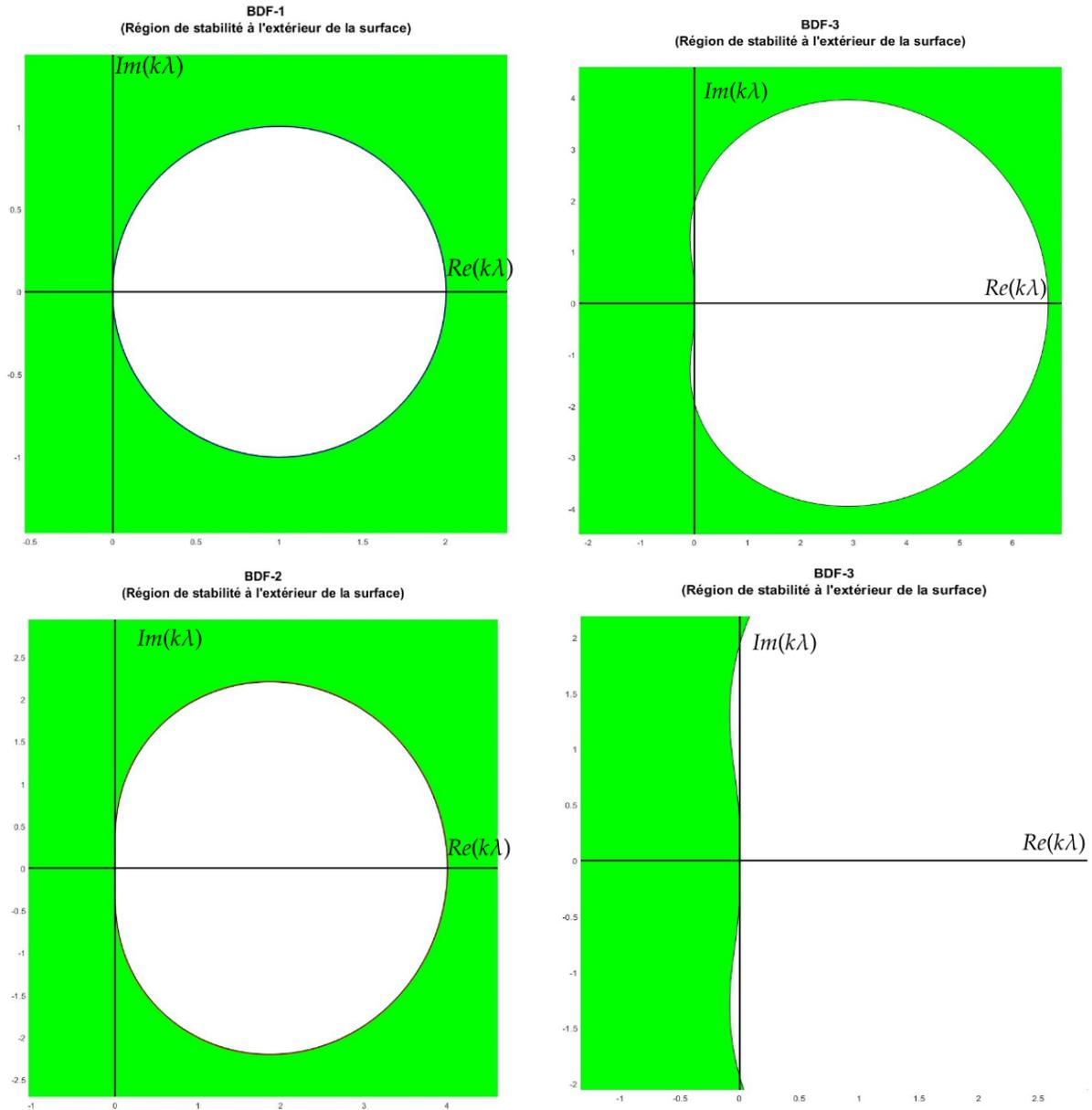


Figure 3.1 Régions de stabilité des méthodes BDF1, BDF2 et BDF3 (en vert).

Dans les cas BDF1 et BDF2, la région d'instabilité se situe strictement à droite de l'axe imaginaire. Cela signifie que la méthode est inconditionnellement stable ; pour tout problème physiquement stable ($Re(\lambda < 0)$), le schéma numérique est stable indépendamment du choix du pas de temps.

Définition 2. *A-stabilité* [19]

Une méthode est dite *A-stable*, si sa zone de stabilité absolue comprend tout le demi plan à gauche de l'axe imaginaire, c'est-à-dire lorsque $k\lambda_R < 0$.

On peut donc qualifier les méthodes BDF1 et BDF2 de *A-stables*. Pour la méthode BDF3, il existe une petite zone d'instabilité à gauche de l'axe imaginaire. Ce phénomène sera de plus en plus important pour des méthodes BDF d'ordres plus élevés. Ces méthodes ne sont pas *A-stables*. Par la suite, nous cherchons à développer des méthodes *A-stables*, nous allons donc faire usage des méthodes BDF1, BDF2.

Remarque 3.3.1. *Stabilité à pas variables*

Nous avons évoqué la stabilité des méthodes BDF à pas constant. Notons qu'à pas variable cette stabilité est plus restrictive. Le schéma BDF1 est stable quel que soit le pas de temps. En revanche dans le cas du BDF2, le taux d'accroissement du pas de temps ne peut pas dépasser une certaine valeur [24]. Une démonstration inspirée de Crouzeix et Mignot [25] est fournie par Bourgault et al. [23] pour aboutir au résultat suivant : si $\omega_n < 1 + \sqrt{2}$, la méthode BDF2 est stable. Cependant dans notre cas nous effectuerons seulement des études à pas de temps alternés deux à deux. Dans ce cas particulier les auteurs ont également montré que le taux d'accroissement pouvait être supérieur à $1 + \sqrt{2}$ tout en conservant la stabilité.

Conclusion

Dans cette section nous avons développé une approche intuitive pour construire les méthodes BDF à pas constants. Puis nous avons formalisé les méthodes BDF1 et BDF2 à pas variables. Nous avons vu que seules ces méthodes sont *A-stables* parmi les méthodes BDF. Nous continuerons donc de travailler avec ces deux méthodes seulement, car nous cherchons à produire des méthodes *A-stables* d'ordre supérieurs.

CHAPITRE 4 MÉTHODES À CORRECTIONS DIFFÉRÉES

Nous allons maintenant développer des méthodes à corrections différées basées sur les méthodes BDF1 et BDF2. Nous allons construire des méthodes corrigées *A-stables* allant jusqu'à l'ordre 5.

4.1 Construction des méthodes DC/BDF1

Nous allons développer une méthode basé sur le schéma BDF1 afin d'augmenter l'ordre de convergence de l'erreur tout en conservant la propriété de *A-stabilité* de la méthode BDF1. Pour cela nous ajoutons une correction \mathcal{D}_p au schéma d'Euler $\frac{u^n - u^{n-1}}{k_1}$. Pour augmenter l'ordre du schéma, nous utilisons l'erreur de troncature de ce dernier pour le corriger. Nous avons déjà fait ce travail dans la section 3.1. Cependant nous avons vu qu'augmenter l'ordre pouvait amener à des instabilités. Nous allons donc présenter une nouvelle manière de construire la correction afin de préserver la *A-stabilité* propre à la méthode BDF1.

De manière générale ce nouveau schéma à corrections différées peut s'écrire de la façon suivante :

$$\boxed{\frac{u^n - u^{n-1}}{k_1} + \mathcal{D}_p = F(t^n, u^n)} . \quad (4.1)$$

avec $\mathcal{D}_p = \sum_{i=1}^p \frac{u^{(i+1)}(t^n)}{i+1} k^i$. Cette correction fait intervenir les dérivées de la solution exacte $u^{(i+1)}(t^n)$ et permet de compenser l'erreur de troncature à l'ordre p . En pratique nous utilisons le schéma suivant :

$$\boxed{\frac{u^n - u^{n-1}}{k_1} + \mathcal{D}_p(u^n) = F(t^n, u^n)} , \quad (4.2)$$

$$\text{avec } \mathcal{D}_p(u^n) = \sum_{i=1}^p \frac{u^{(i+1),n}}{i+1} k^i .$$

L'enjeu est de déterminer le terme $u^{(i+1),n}$ qui correspond à l'approximation de la dérivée analytique à l'aide des valeurs numériques dont nous disposons.

4.1.1 Méthode DC2/BDF1

Construisons tout d'abord une méthode corrigée d'ordre 2 basée sur le schéma BDF1. Pour cela il nous faut déterminer l'erreur de troncature du BDF1, c'est-à-dire l'erreur d'approximation de la méthode numérique par rapport à la solution exacte. Nous avons déjà fait ce

travail dans la section 3.1, nous connaissons donc l'erreur de troncature de la méthode d'Euler implicite :

$$\mathcal{E}(t^n) = -u''(t^n)\frac{k_1}{2!} + u'''(t^n)\frac{k_1^2}{3!} + \dots \quad (4.3)$$

Pour compenser le premier terme de cette erreur, nous allons appliquer une correction $\mathcal{D}_1 = u''(t^n)\frac{k_1}{2!}$. À ce stade nous retrouvons le schéma semi-discret déjà vu dans l'équation (3.6) :

$$\frac{u_2^n - u_2^{n-1}}{k_1} + u''(t^n)\frac{k_1}{2!} = f(t^n, u_2^n). \quad (4.4)$$

Approximation de la dérivée seconde

Nous cherchons à déterminer l'erreur de troncature de la méthode BDF1 à l'ordre 2. Cependant nous avons vu que construire la dérivée à partir de la solution numérique peut mener à des schémas instables. Cette fois, l'idée est de construire la dérivée à partir d'un schéma BDF1 indépendant :

$$\begin{aligned} \frac{u_1^n - u_1^{n-1}}{k_1} &= f(t^n, u_1^n), \\ \frac{u_2^n - u_2^{n-1}}{k_1} + \mathcal{D}_1(u^n) &= f(t^n, u_2^n). \end{aligned} \quad (4.5)$$

Ce schéma est utilisé en parallèle du nouveau schéma corrigé, mais n'est pas affecté par la correction. Nous avons alors besoin d'un polynôme d'approximation de la solution indépendante du BDF1.

Pour le construire nous utilisons une table de différences divisées basée sur le membre de droite.

Tableau 4.1 Table des différences divisées basée sur $F(t, u_1)$

	t^n	F_1^n		
$k_1 = t^n - t^{n-1}$			$\delta_{11}^{F_1}$	
	t^{n-1}	F_1^{n-1}		$\delta_{12}^{F_1}$
$k_2 = t^{n-1} - t^{n-2}$			$\delta_{21}^{F_1}$	
	t^{n-2}	F_1^{n-2}		

Nous aurions pu utiliser une table de différences divisées basée sur la solution u pour en déduire un polynôme approximant la solution (comme nous l'avons vu pour la table 3.1). En dérivant deux fois ce polynôme, nous aurions obtenu une approximation de la dérivée seconde. Cependant en notant que $\frac{du}{dt} = F$, nous pouvons nous contenter d'une approximation de la dérivée première de F pour obtenir une approximation de $u''(t)$. Cette opération nous permet

de gagner une étape de dérivation et donc d'améliorer la précision de notre approximation, d'où le choix de travailler avec le second membre F :

$$F(t^n) \approx p(t) = F_1 + \delta_{11}^F(t - t^n),$$

et en le dérivant une première fois nous obtenons :

$$\frac{dF}{dt}(t^n) \approx \delta_{11}^F,$$

d'où

$$\frac{d^2u}{dt^2}(t^n) = \frac{dF}{dt}(t^n) \approx \delta_{11}^F.$$

La correction qui en résulte est donc :

$$\mathcal{D}_1 = \delta_{11}^F \frac{k_1}{2}.$$

Nous pouvons alors construire une méthode BDF1 corrigée :

$$\boxed{DC2/BDF1 \begin{cases} \frac{u_1^n - u_1^{n-1}}{k_1} & = f(t^n, u_1^n) \\ \frac{u_2^n - u_2^{n-1}}{k_1} + \delta_{11}^F \frac{k_1}{2} & = f(t^n, u_2^n) \end{cases}} \quad (4.6)$$

Cette méthode présente un ordre de convergence de 2 et hérite de la *A-stabilité* du BDF1. Bourgault et al. [23] ont en effet montré que dans le cas où la correction est basée sur une solution indépendante, les BDF corrigés gardaient leur caractère *A-stable*. Le léger inconvénient de cette méthode est qu'elle se base sur un BDF1 et que, lors de la résolution, les deux méthodes doivent être calculées en parallèle. Cette méthode requiert deux résolutions de système matriciel contre une seule pour le BDF2. Le coût de calcul est donc plus important. En revanche le coût de stockage est le même, avec deux solutions antérieures à garder en mémoire.

4.1.2 Méthode DC3/BDF1

Nous pouvons maintenant nous baser sur la méthode DC2/BDF1 pour augmenter l'ordre et créer un DC3/BDF1. Le développement est le même que pour le DC2/BDF1 cependant la correction permet de compenser les termes d'ordre 1 et 2 de l'erreur. Elle a donc la forme :

$$\mathcal{D}_2 = +u''(t^n) \frac{k_1}{2!} - u'''(t^n) \frac{(k_1)^2}{3!}.$$

Il faut, cette fois, obtenir les approximations des dérivées deuxième et troisième. Pour les mêmes raisons que précédemment, nous utilisons la table de différences divisées basée sur le second membre F (4.3). Nous cherchons une erreur d'ordre plus élevé, nous avons donc besoin de plus d'information et nous obtenons un polynôme d'approximation de F plus précis :

$$F(t^n) \approx F_1 + \delta_{11}^F(t - t^n) + \delta_{12}^F(t - t^n)(t - t^{n-1}).$$

Nous dérivons deux fois le polynôme pour obtenir des approximations des dérivées deuxième et troisième. En faisant cela, nous remarquons que l'approximation de la dérivée deuxième est plus précise et la correction sera donc meilleure que précédemment :

$$\frac{dF}{dt}(t) \approx \delta_{11}^F + \delta_{12}^F [(t - t^n) + (t - t^{n-1})],$$

$$\frac{d^2F}{dt^2}(t) \approx 2\delta_{12}^F.$$

Nous évaluons ensuite au temps courant t^n pour connaître la valeur de la correction à ce temps :

$$\mathcal{D}_3 = \frac{k_1}{2} (\delta_{11}^F + \delta_{12}^F(t^n - t^{n-1})) - \delta_{12}^F \frac{(k_1)^2}{3}. \quad (4.7)$$

Nous pouvons alors corriger la méthode DC2/BDF1 afin de gagner un ordre tout en conservant la *A-stabilité* :

$$DC3/BDF1 \begin{cases} \frac{u_1^n - u_1^{n-1}}{k_1} & = f(t^n, u_1^n) \\ \frac{u_2^n - u_2^{n-1}}{k_1} + \delta_{11}^{F_1} \frac{k_1}{2} & = f(t^n, u_2^n) \\ \frac{u_3^n - u_3^{n-1}}{k_1} + (\delta_{11}^{F_2} + \delta_{12}^{F_2} k_1) \frac{k_1}{2} - \delta_{12}^{F_2} \frac{(k_1)^2}{3} & = f(t^n, u_3^n) \end{cases} \quad (4.8)$$

Cette méthode d'ordre 3 présente l'avantage de la stabilité dans le cas d'équations raides, ce que ne fournit pas une méthode BDF3 classique qui est, certes, d'ordre 3, mais peut diverger selon certaines conditions sur le pas de temps. Un BDF3 est conditionnellement stable et n'est donc pas adapté à une utilisation en pas alternés par exemple. La méthode DC/BDF peut sembler lourde à mettre en place étant donné la cascade de méthodes qu'elle demande : BDF1, puis DC2/BDF1, pour arriver au DC3/BDF1. Mais il faut noter que ces méthodes sont individuellement simples à mettre en place, car elles demandent peu de calcul (contrairement aux méthodes de Runge-Kutta par exemple). De plus, le coût de calcul est en fait assez faible étant donné que cette cascade de systèmes utilise la même matrice pour le membre de gauche et il n'y a donc pas de calcul et d'assemblage de matrice supplémentaire.

Remarque 4.1.1. Notons que l'initialisation de la méthode DC3/BDF1 est un enjeu. En effet, au vu de la quantité d'information nécessaire pour construire la correction à l'ordre 3 (information sur 3 pas de temps), nous ne pouvons construire la solution pour le premier pas de temps. Nous étudierons dans la section 4.1.4 comment remédier à ce problème.

4.1.3 Méthodes DC/BDF1 d'ordres supérieurs

Nous avons vu comment construire une correction d'ordre 3, nous allons dans cette section présenter les résultats des corrections à l'ordre 4 et 5. Le but est toujours de partir d'un schéma d'Euler (BFD1) et de le corriger en annulant l'erreur de troncature.

DC4/BDF1

Dans le cas d'un DC4/BDF1 la correction prend la forme suivante :

$$\mathcal{D}_4 = +u''(t^n) \frac{k_1}{2!} - u'''(t^n) \frac{k_1^2}{3!} + u^{(4)}(t^n) \frac{k_1^3}{4!}.$$

Il faut donc maintenant approximer la dérivée quatrième de la solution. Nous utilisons toujours une table de différences divisées basée cette fois sur les pas de temps $t^n, t^{n-1}, t^{n-2}, t^{n-3}$ et dont les résidus correspondants sont issus du DC3/BDF1. En effet ces résidus doivent être d'ordre trois et sont donc obtenus à l'issue de la cascade DC/BDF1.

Tableau 4.2 Table des différences divisées basée sur le résidu à l'ordre 3 F_3

	t^n	F_3^n			
$k_1 = t^n - t^{n-1}$			$\delta_{11}^{F_3}$		
	t^{n-1}	F_3^{n-1}		$\delta_{12}^{F_3}$	
$k_2 = t^{n-1} - t^{n-2}$			$\delta_{21}^{F_3}$		$\delta_{13}^{F_3}$
	t^{n-2}	F_3^{n-2}		$\delta_{22}^{F_3}$	
$k_3 = t^{n-2} - t^{n-3}$			$\delta_{31}^{F_3}$		
	t^{n-3}	F_3^{n-3}			

Nous avons besoin des informations des trois pas de temps précédant le pas de temps courant. Cela permet d'avoir accès au coefficient $\delta_{13}^{F_3}$ qui intervient dans le calcul de la dérivée quatrième. De plus ce coefficient est également présent dans les dérivées d'ordres inférieurs,

il a donc pour effet d'affiner encore l'approximation de chacune des dérivées de la solution :

$$\begin{aligned}
u''(t^n) &\approx \delta_{11}^F + \delta_{12}^F(t^n - t^{n-1}) + \delta_{13}^F(t^n - t^{n-1})(t^n - t^{n-2}), \\
u'''(t^n) &\approx 2 \delta_{12}^F + 2 \delta_{13}^F [(t^n - t^{n-1}) + (t^n - t^{n-2})], \\
u^{(4)}(t^n) &\approx 6 \delta_{13}^F.
\end{aligned} \tag{4.9}$$

La correction est maintenant évaluée par trois dérivées, chacune étant plus précise que précédemment, nous pouvons donc facilement concevoir que la solution corrigée sera de meilleure qualité. Voici donc notre schéma DC4/BDF1 en utilisant les dérivées évaluées précédemment :

$$BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_1^n - u_1^{n-1}}{k_1} = f(t^n, u_1^n)} \end{array} \right. \tag{4.10}$$

$$DC2/BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_2^n - u_2^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2} = f(t^n, u_2^n)} \\ u''(t^n) \approx \delta_{11}^{F1} \end{array} \right. \tag{4.11}$$

$$DC3/BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_3^n - u_3^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2} - u'''(t^n) \frac{k_1^2}{3!} = f(t^n, u_3^n)} \\ u''(t^n) \approx \delta_{11}^F + \delta_{12}^{F2}(t^n - t^{n-1}) \\ u'''(t^n) \approx 2 \delta_{12}^{F2} \end{array} \right. \tag{4.12}$$

$$DC4/BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_4^n - u_4^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2} - u'''(t^n) \frac{k_1^2}{3!} + u^{(4)}(t^n) \frac{k_1^3}{4!} = f(t^n, u_4^n)} \\ u''(t^n) \approx \delta_{11}^{F3} + \delta_{12}^{F3}(t^n - t^{n-1}) + \delta_{13}^{F3}(t^n - t^{n-1})(t^n - t^{n-2}) \\ u'''(t^n) \approx 2 \delta_{12}^{F3} + 2 \delta_{13}^{F3} [(t^n - t^{n-1}) + (t^n - t^{n-2})] \\ u^{(4)}(t^n) \approx 6 \delta_{13}^{F3} \end{array} \right. \tag{4.13}$$

DC5/BDF1

Nous pouvons encore augmenter la précision de l'erreur de troncature à l'ordre 5 afin de construire une méthode DC5/BDF1 :

$$\mathcal{D}_5 = +u''(t^n) \frac{k_1}{2} - u'''(t^n) \frac{k_1^2}{3!} + u^{(4)}(t^n) \frac{k_1^3}{4!} - u^{(5)}(t^n) \frac{k_1^4}{5!}.$$

Il nous faut cette fois encore déterminer une dérivée supplémentaire à l'aide d'une table de différences divisées. Nous obtenons alors les dérivées suivantes :

$$\begin{aligned}
u''(t^n) &\approx \delta_{11}^F + \delta_{12}^F(t^n - t^{n-1}) \\
&\quad + \delta_{13}^F(t^n - t^{n-1})(t^n - t^{n-2}) + \delta_{14}^F(t^n - t^{n-1})(t^n - t^{n-2})(t^n - t^{n-3}) \\
u'''(t^n) &\approx 2 \delta_{12}^F + 2 \delta_{13}^F [(t^n - t^{n-1}) + (t^n - t^{n-2})] \\
&\quad + 2 \delta_{14}^F [(t^n - t^{n-1})(t^n - t^{n-2}) + (t^n - t^{n-1})(t^n - t^{n-3}) + (t^n - t^{n-2})(t^n - t^{n-3})] \\
u^{(4)}(t^n) &\approx 6 \delta_{13}^F + 6 \delta_{14}^F [(t^n - t^{n-1}) + (t^n - t^{n-2}) + (t^n - t^{n-3})] \\
u^{(5)}(t^n) &\approx 24 \delta_{14}^F
\end{aligned} \tag{4.14}$$

Nous pouvons évaluer la correction à partir des résultats du DC4/BDF1 et construire le DC5/BDF1.

$$BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_1^n - u_1^{n-1}}{k_1} = f(t^n, u_1^n)} \end{array} \right. \tag{4.15}$$

$$DC2/BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_2^n - u_2^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2} = f(t^n, u_2^n)} \\ u''(t^n) \approx \delta_{11}^{F1} \end{array} \right. \tag{4.16}$$

$$DC3/BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_3^n - u_3^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2} - u'''(t^n) \frac{k_1^2}{3!} = f(t^n, u_3^n)} \\ u''(t^n) \approx \delta_{11}^F + \delta_{12}^{F2}(t^n - t^{n-1}) \\ u'''(t^n) \approx \delta_{12}^{F2} \end{array} \right. \tag{4.17}$$

$$DC4/BDF1 \quad \left\{ \begin{array}{l} \boxed{\frac{u_4^n - u_4^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2} - u'''(t^n) \frac{k_1^2}{3!} + u^{(4)}(t^n) \frac{k_1^3}{4!} = f(t^n, u_4^n)} \\ u''(t^n) \approx \delta_{11}^{F3} + \delta_{12}^{F3}(t^n - t^{n-1}) + \delta_{13}^{F3}(t^n - t^{n-1})(t^n - t^{n-2}) \\ u'''(t^n) \approx 2 \delta_{12}^{F3} + 2 \delta_{13}^{F3} [(t^n - t^{n-1}) + (t^n - t^{n-2})] \\ u^{(4)}(t^n) \approx 6 \delta_{13}^{F3} \end{array} \right. \tag{4.18}$$

$$\left. \begin{array}{l}
\boxed{\frac{u_5^n - u_5^{n-1}}{k_1} + u''(t^n) \frac{k_1}{2} - u'''(t^n) \frac{k_1^2}{3!} + u^{(4)}(t^n) \frac{k_1^3}{4!} - u^{(5)}(t^n) \frac{k_1^4}{5!} = f(t^n, u_5^n)} \\
DC5/BDF1 \left\{ \begin{array}{l}
u''(t^n) \approx \delta_{11}^{F_4} + \delta_{12}^{F_4}(t^n - t^{n-1}) \\
\quad + \delta_{13}^{F_4}(t^n - t^{n-1})(t^n - t^{n-2}) + \delta_{14}^{F_4}(t^n - t^{n-1})(t^n - t^{n-2})(t^n - t^{n-3}) \\
u'''(t^n) \approx 2 \delta_{12}^{F_4} + 2 \delta_{13}^{F_4} [(t^n - t^{n-1}) + (t^n - t^{n-2})] \\
\quad + 2 \delta_{14}^{F_4} [(t^n - t^{n-1})(t^n - t^{n-2}) + (t^n - t^{n-1})(t^n - t^{n-3}) + (t^n - t^{n-2})(t^n - t^{n-3})] \\
u^{(4)}(t^n) \approx 6 \delta_{13}^{F_4} + 6 \delta_{14}^{F_4} [(t^n - t^{n-1}) + (t^n - t^{n-2}) + (t^n - t^{n-3})] \\
u^{(5)}(t^n) \approx 24 \delta_{14}^{F_4}
\end{array} \right.
\end{array} \right. \quad (4.19)$$

Remarque 4.1.2. Notons enfin que l'initialisation des DC4/BDF1 et DC5/BDF1 est également un enjeu. Par exemple, le démarrage d'un DC5/BDF1 requiert la connaissance des 4 premières solutions à l'ordre 5. Dans ce cas, nous appliquerons un procédé similaire à celui du schéma 4.1 en jouant avec des corrections évaluées à des pas de temps antérieurs au temps courant.

4.1.4 Démarrage de la méthode DC3/BDF1

Le principal problème de cette méthode concerne en fait son démarrage. Nous avons vu que, pour construire les corrections, nous utilisons une table de différences divisées se basant sur le membre de droite 4.3. Pour construire l'erreur à l'ordre 2, nous avons besoin des F_1^n et F_1^{n-1} soit les F_1^1 et F_1^0 dans le cas du démarrage de la méthode. Nous pouvons alors obtenir une valeur de la solution à t^1 par le DC2/BDF1. Pour la méthode DC3/BDF1, la correction nécessite la connaissance de F_2^n , F_2^{n-1} et F_2^{n-2} . La première valeur calculable par le DC3/BDF1 sera donc la solution au temps discrétisé t^2 . Nous ne connaissons pas de solution pour le premier pas de temps u_3^1 à l'ordre 3. Pour obtenir cette solution au temps t^1 nous devons utiliser un schéma différent tel que présenté sur la Figure 4.1.

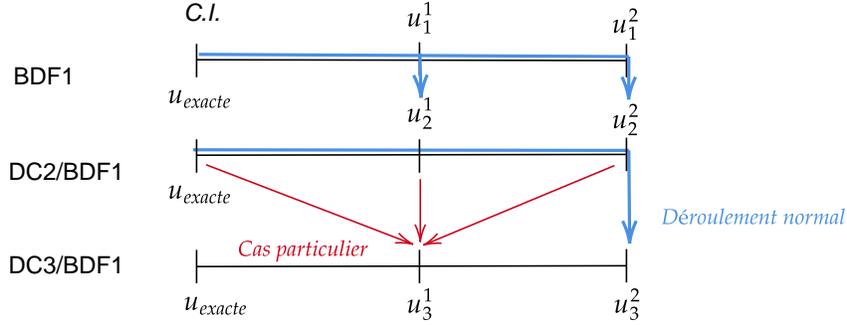


Figure 4.1 Principe de l'initialisation de la première valeur pour la méthode DC3/BDF1

Le démarrage consiste donc à se rendre au temps t^2 pour avoir suffisamment d'information et calculer u_3^1 et u_3^2 à l'ordre 3. Pour le calcul de u_3^1 , nous utilisons alors la même méthode que pour le DC3/BDF1 étudié à la section 4.1.2 : l'expression analytique de la correction \mathcal{D}_2 est inchangée. L'approximation des dérivées est également identique :

$$\frac{dF}{dt}(t) \approx \delta_{11}^F + \delta_{12}^F [(t - t^2) + (t - t^1)]$$

$$\frac{d^2F}{dt^2}(t) \approx 2\delta_{12}^F.$$

Mais, nous les évaluons cette fois au temps t^1 pour u^1 (contre t^2 pour obtenir u^2) et nous obtenons cette fois la correction suivante :

$$\mathcal{D}_2 = \left(\delta_{11}^F + \delta_{12}^F (t^1 - t^2) \right) \frac{k_2}{2!} - 2\delta_{12}^F \frac{(k_2)^2}{3!}.$$

Avec cette nouvelle correction centrée autour de t^1 , nous pouvons calculer la valeur numérique d'ordre trois u_3^1 . Puis, nous suivons le déroulement usuel pour les solutions suivantes u_3^n . Ainsi, nous avons construit une méthode ne nécessitant que la connaissance des conditions initiales pour démarrer et permettant de maintenir une erreur d'ordre constant sur l'ensemble du domaine temporel.

4.2 Méthodes DC/BDF2

Nous avons vu, dans la section précédente, comment augmenter l'ordre de convergence d'un schéma BDF1. Nous pouvons effectuer le même travail avec le schéma BDF2 (méthode de Gear). Nous rappelons que le BDF2 est *A-stable* [23] et nous allons chercher à maintenir cette caractéristique avec la méthode de corrections différées. Bourgault et al. [23] ont montré que

la correction ajoutée à un BDF2 *A-stable* permettait de conserver ce caractère .

Dans le cas du BDF2, nous introduisons le terme de correction \mathcal{D}_p (nous allons développer son expression analytique dans la section suivante) pour écrire le schéma BDF2 corrigé d'ordre p :

$$c_0^n u^n + c_1^n u^{n-1} + c_2^n u^{n-2} + \mathcal{D}_p = F(t^n, u^n). \quad (4.20)$$

Rappelons enfin que la correction sert à compenser les termes de l'erreur, nous avons donc $\mathcal{D}_p = -\mathcal{E}_p(t^n)$. Nous devons donc obtenir l'expression analytique de l'erreur de troncature du BDF2.

4.2.1 Erreur de troncature du BDF2

Nous écrivons tout d'abord l'écriture discrétisée du schéma BDF2 :

$$c_0^n u^n + c_1^n u^{n-1} + c_2^n u^{n-2} = F(t^n, u^n).$$

La correction de ce schéma sera de nouveau constituée de l'erreur de troncature, c'est à dire l'erreur d'approximation du BDF2 par rapport à la solution exacte. Dans le but de faire apparaître l'erreur de troncature $\mathcal{E}(t^n)$, nous remplaçons la solution discrète par la solution exacte évaluée au même pas de temps :

$$c_0^n u(t^n) + c_1^n u(t^{n-1}) + c_2^n u(t^{n-2}) - F(t^n, u(t^n)) = \mathcal{E}(t^n). \quad (4.21)$$

On obtient une expression de l'erreur de troncature en exprimant les solutions $u(t^{n-1})$ et $u(t^{n-2})$ en fonction de la solution $u(t^n)$ par un développement de Taylor :

$$\begin{aligned} u(t^{n-1}) &= u(t^n) - u'(t^n) k_1 + u''(t^n) \frac{k_1^2}{2} - u'''(t^n) \frac{k_1^3}{6} + u^{(4)}(t^n) \frac{k_1^4}{24} - u^{(5)} \frac{k_1^5}{120} + \dots \\ u(t^{n-2}) &= u(t^n) - u'(t^n) (k_1 + k_2) + u''(t^n) \frac{(k_1 + k_2)^2}{2} - u'''(t^n) \frac{(k_1 + k_2)^3}{6} \\ &\quad + u^{(4)}(t^n) \frac{(k_1 + k_2)^4}{24} - u^{(5)}(t^n) \frac{(k_1 + k_2)^5}{120} + \dots \end{aligned} \quad (4.22)$$

En injectant ces séries de Taylor dans l'équation (4.21), nous n'avons plus que des termes en $u(t^n)$:

$$\begin{aligned}
u(t^n) (c_0^n + c_1^n + c_2^n) - u'(t^n) (c_1^n k_1 + c_2^n (k_1 + k_2)) + \frac{u''(t^n)}{2} (c_1^n k_1^2 + c_2^n (k_1 + k_2)^2) \\
- \frac{u'''(t^n)}{6} (c_1^n k_1^3 + c_2^n (k_1 + k_2)^3) + \frac{u^{(4)}(t^n)}{24} (c_1^n k_1^4 + c_2^n (k_1 + k_2)^4) \\
- \frac{u^{(5)}(t^n)}{120} (c_1^n k_1^5 + c_2^n (k_1 + k_2)^5) + \dots - F(t^n, u(t^n)) = \mathcal{E}(t^n)
\end{aligned} \tag{4.23}$$

Pour faire apparaître les coefficient c_0^n , c_1^n et c_2^n , nous montrons les relations suivantes :

$$\begin{aligned}
c_0^n + c_1^n + c_2^n &= \left(\frac{1}{k_1} + \frac{1}{k_1 + k_2} \right) + \left(-\frac{1}{k_1} - \frac{1}{k_2} \right) + \left(\frac{k_1}{k_2} \frac{1}{k_1 + k_2} \right) = 0 \\
c_1^n k_1 + c_2^n (k_1 + k_2) &= \left(-\frac{1}{k_1} - \frac{1}{k_2} \right) k_1 + \left(\frac{k_1}{k_2} \frac{1}{k_1 + k_2} \right) (k_1 + k_2) = -1 \\
c_1^n k_1^2 + c_2^n (k_1 + k_2)^2 &= \left(-\frac{1}{k_1} - \frac{1}{k_2} \right) k_1^2 + \left(\frac{k_1}{k_2} \frac{1}{k_1 + k_2} \right) (k_1 + k_2)^2 = 0 \\
c_1^n k_1^3 + c_2^n (k_1 + k_2)^3 &= \left(-\frac{1}{k_1} - \frac{1}{k_2} \right) k_1^3 + \left(\frac{k_1}{k_2} \frac{1}{k_1 + k_2} \right) (k_1 + k_2)^3 = k_1 (k_1 + k_2) \\
c_1^n k_1^4 + c_2^n (k_1 + k_2)^4 &= \left(-\frac{1}{k_1} - \frac{1}{k_2} \right) k_1^4 + \left(\frac{k_1}{k_2} \frac{1}{k_1 + k_2} \right) (k_1 + k_2)^4 = k_1 (k_1 + k_2) (2k_1 + k_2) \\
c_1^n k_1^5 + c_2^n (k_1 + k_2)^5 &= \left(-\frac{1}{k_1} - \frac{1}{k_2} \right) k_1^5 + \left(\frac{k_1}{k_2} \frac{1}{k_1 + k_2} \right) (k_1 + k_2)^5 = k_1 (k_1 + k_2) (3k_1^2 + 3k_1 k_2 + k_2^2)
\end{aligned} \tag{4.24}$$

En se servant de ces relations, nous faisons apparaître l'équation différentielle exacte (comme nous l'avons fait dans l'équation (3.4)) :

$$u'(t^n) - F(t^n, u(t^n)) = 0. \tag{4.25}$$

Il ne nous reste donc que les termes correspondant à l'erreur de troncature :

$$\boxed{
\begin{aligned}
\mathcal{E}(t^n) &= -\frac{u'''(t^n)}{6} k_1 (k_1 + k_2) + \frac{u^{(4)}(t^n)}{24} k_1 (k_1 + k_2) (2k_1 + k_2) \\
&\quad - \frac{u^{(5)}(t^n)}{120} k_1 (k_1 + k_2) (3k_1^2 + 3k_1 k_2 + k_2^2) + \dots
\end{aligned}
} \tag{4.26}$$

Nous rappelons que l'expression de la correction que nous cherchons à établir permet de compenser certains termes de l'erreur. C'est la raison pour laquelle l'expression analytique de l'erreur sera la base de notre travail par la suite.

4.2.2 DC3/BDF2

Pour construire une méthode d'ordre 3 nous utilisons la correction D_3 basée sur l'erreur de troncature à l'ordre 3 :

$$\mathcal{D}_3 = \frac{u'''(t^n)}{6} k_1 (k_1 + k_2).$$

Cette correction fait apparaître la dérivée troisième. Encore une fois, tout l'enjeu réside dans la manière d'approcher cette dérivée. Comme nous l'avons vu avec le DC/BDF1, nous construisons la dérivée à partir d'une solution calculée indépendamment et non de la solution courante, car cela poserait des problèmes de stabilité (nous obtiendrions un BDF3 non *A-stable*). Nous utilisons une table de différences divisées basée sur le second membre $F^n = F(t^n, u^n)$ évalué à la solution du BDF2 u_2^n qui est donc d'ordre 2.

Tableau 4.3 Table de différences divisées basée sur $F(t, u_2)$

$k_1 = t^n - t^{n-1}$	t^n	F_2^n	$\delta_{11}^{F_2}$	
$k_2 = t^{n-1} - t^{n-2}$	t^{n-1}	F_2^{n-1}	$\delta_{21}^{F_2}$	$\delta_{12}^{F_2}$
	t^{n-2}	F_2^{n-2}		

Cette table nous permet d'obtenir un polynôme d'approximation de Newton sur le second membre :

$$F_2(t) = F_2^n + \delta_{11}^{F_2}(t - t^n) + \delta_{12}^{F_2}(t - t^n)(t - t^{n-1}). \quad (4.27)$$

Pour rappel, nous nous servons du fait que $\frac{du}{dt} = F$ pour gagner un ordre de dérivation. En dérivant deux fois F , nous obtenons une approximation de la dérivée troisième de la solution u que l'on évalue au temps t^n :

$$\frac{dF_2}{dt}(t^n) \approx \delta_{11}^{F_2} + \delta_{12}^{F_2}(t^n - t^{n-1}),$$

$$\frac{d^2F_2}{dt^2}(t^n) \approx 2\delta_{12}^{F_2}.$$

Sachant que $u'''(t^n) \approx 2\delta_{12}^{F_2}$, nous pouvons construire la correction appropriée au BDF2 :

$$\mathcal{D}_1 = \delta_{12}^{F_2} \frac{k_1 (k_1 + k_2)}{3}. \quad (4.28)$$

La construction de la méthode DC/BDF2 est la même que précédemment. Nous calculons tout d'abord une solution à l'ordre 2 issue du BDF2 u_2 . Puis nous utilisons ce résultat pour

calculer la correction obtenue (4.28). L'erreur de cette nouvelle méthode sera maintenant dominée par le terme $\frac{u^{(4)}(t^n)}{24} k_1 (k_1 + k_2) (2k_1 + k_2)$. Nous obtenons donc une solution corrigée à l'ordre 3 toujours *A-stable*.

$$BDF2 \left\{ \begin{array}{l} \sum_{i=0}^2 c_i^n u_2^{n-i} = f(t^n, u_2^n) \end{array} \right. \quad (4.29)$$

$$DC3/BDF2 \left\{ \begin{array}{l} \sum_{i=0}^2 c_i^n u_3^{n-i} + \frac{u'''(t^n)}{6} k_1 (k_1 + k_2) = f(t^n, u_3^n) \\ u'''(t^n) \approx 2\delta_{12}^{F_2} \end{array} \right. \quad (4.30)$$

Nous avons ainsi construit la méthode DC3/BDF2. En se basant sur l'erreur de troncature pour construire la correction nous avons pu augmenter l'ordre de convergence du BDF2. De plus nous nous sommes basés sur une solution parallèle pour calculer les dérivées intervenant dans le terme de correction. Cette structure en cascade permet l'indépendance du BDF2 vis-à-vis du DC3/BDF2 et maintient la *A-stabilité* recherchée.

4.2.3 DC4/BDF2 et DC5/BDF2

À l'instar du travail réalisé pour le BDF, nous pouvons encore augmenter l'ordre de notre méthode en ajoutant de la précision à la correction. Ce gain d'ordre s'effectue toujours en cascade, car la solution précédente sert à calculer les seconds membres nécessaires à l'évaluation de la correction. Le calcul de la correction à l'ordre quatre et cinq est similaire aux corrections précédentes en considérant cette fois plus de termes dans l'erreur de troncature. Sans expliciter les étapes des tables de différences divisées, nous pouvons écrire l'enchaînement des méthodes DC/BDF2.

$$BDF2 \left\{ \begin{array}{l} \sum_{i=0}^2 c_i^n u_2^{n-i} = f(t^n, u_2^n) \end{array} \right. \quad (4.31)$$

$$DC3/BDF2 \left\{ \begin{array}{l} \sum_{i=0}^2 c_i^n u_2^{n-i} + \frac{u'''(t^n)}{6} k_1 (k_1 + k_2) = f(t^n, u_3^n) \\ u'''(t^n) \approx 2\delta_{12}^{F_2} \end{array} \right. \quad (4.32)$$

$$\left. \begin{array}{l}
DC4/BDF2 \left\{ \begin{array}{l}
\boxed{\sum_{i=0}^2 c_i^n u_2^{n-i} + \frac{u'''(t^n)}{6} k_1 (k_1 + k_2) - \frac{u^{(4)}(t^n)}{24} k_1 (k_1 + k_2) (2k_1 + k_2)} \\
\boxed{= f(t^n, u_3^n)} \\
u'''(t^n) \approx 2 \delta_{12}^{F_3} + 2 \delta_{13}^{F_3} (2k_1 + k_2) \\
u^{(4)}(t^n) \approx 6 \delta_{13}^{F_3}
\end{array} \right.
\end{array} \right. \quad (4.33)$$

$$\left. \begin{array}{l}
DC5/BDF2 \left\{ \begin{array}{l}
\boxed{\sum_{i=0}^2 c_i^n u_2^{n-i} + \frac{u'''(t^n)}{6} k_1 (k_1 + k_2) - \frac{u^{(4)}(t^n)}{24} k_1 (k_1 + k_2) (2k_1 + k_2)} \\
\boxed{+ \frac{u^{(5)}(t^n)}{120} k_1 (k_1 + k_2) (3k_1^2 + 3k_1 k_2 + k_2^2) = f(t^n, u_4^n)} \\
u'''(t^n) \approx 2 \delta_{12}^{F_4} + 2 \delta_{13}^{F_4} (2k_1 + k_2) \\
\quad + 2 \delta_{14}^{F_4} (3k_1^2 + 4k_1 k_2 + 2k_3 k_1 + k_2^2 + k_3 k_2) \\
u^{(4)}(t^n) \approx 6 \delta_{13}^{F_4} + 6 \delta_{14}^{F_4} (3k_1 + 2k_2 + k_3) \\
u^{(5)}(t^n) \approx 24 \delta_{14}^{F_4}
\end{array} \right.
\end{array} \right. \quad (4.34)$$

4.3 Utilisation de DC/BDF1 pour démarrer DC/BDF2

Nous allons maintenant combiner les deux méthodes. Nous avons vu que la méthode DC/BDF1 n'avait besoin que d'une condition initiale pour démarrer. C'est à dire que la seule connaissance de la solution u^0 au temps t^0 permet de trouver les solutions suivantes en maintenant une erreur consistante. La méthode BDF2 nécessite les deux premières solutions pour calculer la solution suivante. En effet, pour rappel, voici le schéma du BDF2 :

$$c_0^n u_2^n + c_1^n u_2^{n-1} + c_2^n u_2^{n-2} = F(t^n, u_2^n).$$

Dans le cas d'un BDF2 (et par extension d'un DC/BDF2) la solution initiale u^0 est connue, en revanche u^1 est inconnue. L'idée est donc de d'utiliser le schéma DC/BDF1, le temps d'une itération, afin d'obtenir u^1 à l'ordre souhaité.

Par exemple, si nous voulons utiliser le schéma DC3/BDF2 sur un problème dont la solution analytique est inconnue, nous utilisons un DC3/BDF1 afin de trouver la solution u^1 à l'ordre 2 pour l'injecter dans le BDF2, et une solution u^1 à l'ordre 3 pour l'injecter dans le DC3/BDF2. Puis, nous pouvons enchaîner sur l'utilisation normale du DC3/BDF2 pour toutes les solutions suivantes. Nous représentons ce processus dans le schéma ci-dessous.

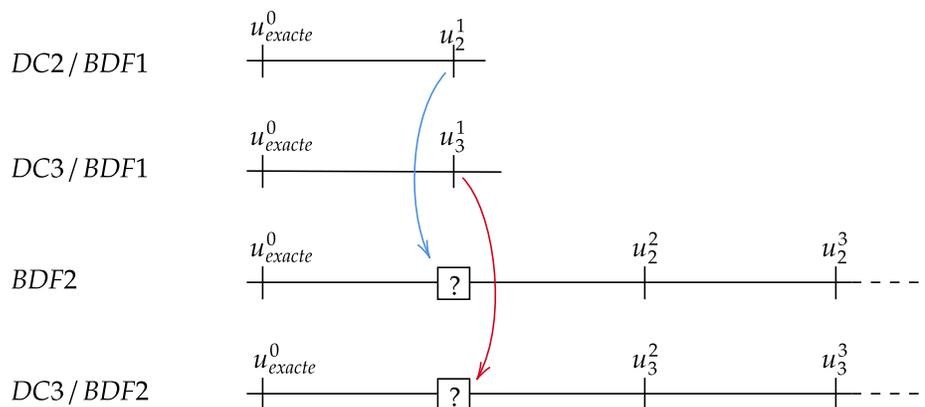


Figure 4.2 Utilisation de la méthode DC/BDF1 pour initialiser DC/BDF2

Dans le cas d'un schéma d'ordre plus élevé, DC5/BDF2 par exemple, nous ferions de même en initialisant les u^1, u^2 et u^3 à l'aide d'un DC5/BDF1 qui fournit des solutions d'ordre 5.

Conclusion

Nous avons vu comment il était possible d'utiliser l'erreur de troncature pour construire des corrections permettant d'augmenter l'ordre des schémas BDF1 et BDF2. Forts de la méthode DC5/BDF1, nous avons vu comment nous pouvions développer une méthode DC5/BDF2 initialisée grâce au DC5/BDF1. En faisant cela, nous obtenons une méthode d'ordre 5, *A-stable*, nécessitant seulement une condition initiale pour démarrer.

CHAPITRE 5 IMPOSITION DES CONDITIONS LIMITES ET PERTE D'ORDRE

Dans le chapitre précédent, nous avons vu comment étaient construites les méthodes BDF et DC/BDF. Nous allons désormais nous intéresser à l'imposition des conditions limites lors de nos simulations, car nous verrons que cela impacte les résultats de convergence de nos méthodes.

5.1 Perte d'ordre

Lors de nos simulations en dimensions 1 et 2, nous imposons des conditions limites de Dirichlet aux frontières de nos différents domaines. Or nous remarquons que cela peut avoir un impact sur les résultats de nos calculs pour des cas particuliers. Lors d'une simulation à **pas de temps alternés** et en imposant fortement des **conditions limites dépendantes du temps**, *i.e.* en imposant $u = u_D$ sur la partie de la frontière sujette à des conditions limites de Dirichlet, nous observons une perte d'ordre dans le cas d'un BDF corrigé deux fois ou plus (DC $n + 2$ /BDF n). Par exemple, un schéma BDF1 corrigé deux fois pour obtenir un DC3/BDF1 présentera un ordre de convergence de 3 en utilisant des pas de temps constants, mais plafonnera à 2 en utilisant des pas de temps alternés et de conditions limites dépendantes du temps. C'est également le cas pour une méthode BDF2 corrigée deux fois. Nous avons pour cela effectué des simulations numériques sur des EDP dont les résultats sont donnés dans les tableaux 7.8 et 7.9 du chapitre 7.

Pour contrer ce problème, nous avons donc étudié la distribution de l'erreur dans le temps et l'espace dans les cas problématiques. La distribution de l'erreur observée dans les figures 7.1 et 7.2 nous permet de faire l'hypothèse que la perte d'ordre est liée à l'imposition des conditions limites. Jusqu'ici, nos conditions limites ont été imposées fortement au moyen de condition de Dirichlet, c'est-à-dire en fixant directement la solution de la variable primaire aux frontières. Dans le cas d'un problème d'éléments finis, cela revient à imposer fortement les conditions limites. Nous allons maintenant nous intéresser à une méthode d'imposition faible des conditions limites.

5.2 Imposition faible des conditions limites et correction de Verwer

En ce qui concerne la méthode des éléments finis, imposer faiblement des conditions limites revient à écrire directement dans la formulation faible et donc dans nos matrices élémentaires la valeur de la solution au bord. En pratique, pour imposer cette valeur, nous utilisons un multiplicateur de Lagrange. Mais, nous n'allons pas simplement imposer la valeur de la solution au bord, nous allons nous inspirer de la méthode utilisée par Verwer pour résoudre des pertes d'ordre lors de discrétisation aux différences finis [26]. Cette méthode revient à imposer faiblement les solutions d'une EDO définie aux bords du domaine. Pour comprendre la mise en œuvre de cette méthode nous allons étudier le cas d'un problème de diffusion instationnaire.

Exemple d'un problème de diffusion instationnaire

Nous écrivons la forme faible d'un problème de diffusion instationnaire pour lequel nous imposons fortement les conditions limites. Nous définissons Ω le domaine d'étude et $\Gamma = \partial\Omega$ le bord de ce domaine. Ainsi, nous cherchons $u \in H^1(\Omega)$ et $\forall \hat{u} \in H^1_\Gamma(\Omega)$ tel que :

$$\int_{\Omega} \hat{u} \frac{\partial u}{\partial t} d\Omega + \int_{\Omega} k_D \nabla \hat{u} \cdot \nabla u d\Omega + \int_{\Omega} \hat{u} S d\Omega = 0 \quad (5.1)$$

avec k_D la conductivité thermique et S le terme source. Considérons désormais une variable sur le bord du domaine $v(x, t)$. Nous notons $u(x, t)$ la solution numérique sur le domaine et $\gamma(x, t)$ la solution exacte du problème. Au bord du domaine nous voulons imposer :

$$\frac{\partial v}{\partial t}(x, t) = \frac{\partial \gamma}{\partial t}(x, t) \quad \text{tel que } v(x, 0) = \gamma(x, 0). \quad (5.2)$$

Nous nous munissons alors d'un multiplicateur de Lagrange $\lambda(x, t)$ qui permet d'imposer v comme valeur de u au bord.

Cela se traduit par l'ajout d'une intégrale au bord du domaine :

$$\begin{cases} \int_{\Omega} \hat{u} \frac{\partial u}{\partial t} d\Omega + \int_{\Omega} k_D \nabla \hat{u} \cdot \nabla u d\Omega + \int_{\Omega} \hat{u} S d\Omega + \int_{\Gamma} \hat{u} \lambda d\Gamma = 0, \\ \int_{\Gamma} \lambda (u - v) d\Gamma = 0, \\ \int_{\Gamma} \hat{v} \left(\frac{\partial v}{\partial t} - \frac{\partial \gamma}{\partial t} \right) d\Gamma = 0. \end{cases} \quad (5.3)$$

Cette méthode permet d'éviter une perte d'ordre sur la variable primaire u et fait apparaître une variable secondaire λ qui, nous allons le voir, à un sens physique.

5.3 Interprétation du multiplicateur

La présence du multiplicateur implique des calculs supplémentaires, mais a un réel intérêt, car le multiplicateur a un sens physique. Dans notre cas, le multiplicateur correspond au flux de chaleur. Mais surtout, dans le cas d'une simulation des équations de Navier-Stokes, le multiplicateur correspond aux forces de tractions aux bords du domaines. Le multiplicateur correspond à la réaction à l'imposition des conditions limites essentielles imposées faiblement. Cela est très avantageux, car en temps normal le calcul des forces peut s'avérer complexe et fastidieux.

CHAPITRE 6 VERIFICATION POUR DES EDO

Nous avons construit des méthodes à corrections différées allant jusqu'à l'ordre 5. Nous allons effectuer des tests sur des EDO pour vérifier la théorie et l'implémentation des schémas DC. La théorie prévoit des taux de convergence de l'erreur temporelle, nous vérifierons qu'ils sont bien atteints en simulant des problèmes construits à l'aide de solutions manufacturées. Dans un premier temps, nous nous intéressons aux EDO et aux systèmes d'EDO, car ces problèmes sont dépendants d'une seule variable et ne font pas intervenir de discrétisation spatiale. Nous nous intéresserons par la suite à des situations d'EDP.

Considérons le problème de référence :

$$\boxed{\begin{cases} \frac{du}{dt} &= s(u, t), \\ u(0) &= u_0. \end{cases}} \quad (6.1)$$

6.0.1 Solutions manufacturées

Pour vérifier nos méthodes nous allons utiliser la méthode des solutions manufacturées [27]. Dans tous les tests suivants, nous connaissons une solution exacte $u(t)$: le principe des solutions manufacturées consiste à évaluer l'équation de référence (6.1) en la solution exacte. On obtient ainsi une expression du terme source $s(u, t)$. En construisant ainsi notre EDO nous obtiendrons une solution numérique u^n que nous pourrons comparer à la solution exacte.

6.0.2 Norme de l'erreur

Pour pouvoir comparer la solution numérique et la solution exacte nous devons nous munir d'une norme permettant de caractériser cette erreur. La norme mesure le maximum de l'erreur parmi tous les pas de temps. Nous définissons un domaine Ω_t pour la variable t : $\Omega_t = [0, T]$: dans la suite de nos simulations $\Omega_t = [0, 1] \in \mathbb{R}$. Nous écrivons donc l'expression de la norme que nous utilisons :

$$\mathcal{E} = \|u(t^n) - u^n\|_{L^\infty(\Omega_t)} = \sup_{t \in [0, T]} |u(t) - u^n|. \quad (6.2)$$

Avec cette mesure de l'erreur, nous nous attendons à observer les taux de convergence suivants [23] :

Schéma temporel	Ordre de convergence de l'erreur temporelle
BDF_n	n
$DC_n/BDF1$	n
$DC_n/BDF2$	n

6.0.3 Initialisation des méthodes BDF

Nous avons vu que l'initialisation des méthodes BDF2 et DC/BDF2 était un enjeu pour le reste de la simulation. Étant donné que nous utilisons la méthodes des solutions manufacturées pour la vérification, nous connaissons une solution exacte à nos problèmes testés. Dans un premier temps, nous utiliserons donc ces solutions exactes pour l'initialisation des valeurs manquantes. Nous vérifierons l'initialisation développée dans la section 4.3 au moyen de cas test sur des EDP en une dimension dans la section 7.5.

6.0.4 Choix du pas de temps

Dans une optique d'adaptation du pas de temps, nous étudierons des cas où le pas de temps est constant $k_1/k_2 = 1$ et des simulations pour lesquelles le pas de temps est alterné deux à deux : $k_1/k_2 = 4$. Nous avons vu que dans cette situation, le rapport d'accroissement ω_n du pas de temps peut excéder $\sqrt{2} + 1$. Nous nous attendons à obtenir des taux de convergence semblables aux pas constants [23].

6.1 Pas de temps constant

Pour commencer, nous effectuons des tests sur une unique EDO afin de vérifier les taux de convergence des méthodes BDF1, DC2/BDF1, DC3/BDF1, DC3/BDF2, DC4/BDF2, DC5/BDF2 en initialisant avec la solution exacte. Nous testons pour les solutions listées dans le tableau 6.1. Nous avons fait ce choix de cas tests, car il comprend des solutions (u_1, u_2) dont le terme source dépend du temps, mais également des solutions dont le terme source dépend de la solution (u_3, u_4) .

Tableau 6.1 Solutions exactes et termes sources pour les tests DC/BDF appliqués à une EDO

Solution	$u_1(t) = t^2$	$u_2(t) = t^4$	$u_3(t) = e^t$	$u_4(t) = -\frac{1}{t+1}$
Terme source	$s_1(u, t) = 2t$	$s_2(u, t) = 4t^3$	$s_3(u, t) = u_3$	$s_4(u, t) = u_4^2$

Nous effectuons les simulations pour les méthodes DC/BDF1 et DC/BDF2. Nous présentons l'erreur commise en norme $L^\infty(\Omega_t)$ pour chaque méthode et nous en déduisons l'ordre de convergence correspondant.

Tableau 6.2 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF1** pour la solution $u_1(t) = t^2$ à **pas de temps constants**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	1,0000e-01	—	1,1102e-16	—	1,1102e-16	—	1,1102e-16	—	1,1102e-16	—
20	5,0000e-02	1,00	2,2204e-16	—	1,1102e-16	—	1,1102e-16	—	1,1102e-16	—
40	2,5000e-02	1,00	2,2204e-16	—	4,4409e-16	—	4,4409e-16	—	4,4409e-16	—
80	1,2500e-02	1,00	2,2204e-16	—	3,3307e-16	—	3,3307e-16	—	3,3307e-16	—
160	6,2500e-03	1,00	5,5511e-16	—	6,6613e-16	—	6,6613e-16	—	6,6613e-16	—

Tableau 6.3 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF2** pour la solution $u_1(t) = t^2$ à **pas de temps constants**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	0,0000e+00	—	0,0000e+00	—	0,0000e+00	—	0,0000e+00	—
20	6,6600e-16	—	6,6600e-16	—	6,6613e-16	—	6,6613e-16	—
40	4,4400e-16	—	4,4400e-16	—	4,4409e-16	—	4,4409e-16	—
80	3,5500e-15	—	3,5500e-15	—	3,5527e-15	—	3,5527e-15	—
160	4,0000e-15	—	4,0000e-15	—	3,9968e-15	—	3,9968e-15	—

Nous observons que la fonction est intégrée de manière exacte par toutes les méthodes exceptées BDF1, cette méthode ne pouvant intégrer que des polynômes de degré inférieur ou égal à 1. Ces résultats sont bien conformes à nos attentes.

Tableau 6.4 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF1** pour la solution $u_2(t) = t^4$ à **pas de temps constants**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	2,1000e-01	—	9,1000e-03	—	8,0000e-04	—	1,1102e-16	—	1,1102e-16	—
20	1,0250e-01	1,03	2,4437e-03	1,90	1,1250e-04	2,83	1,1102e-16	—	1,1102e-16	—
40	5,0625e-02	1,02	6,2148e-04	1,98	1,4844e-05	2,92	1,6653e-16	—	1,6653e-16	—
80	2,5156e-02	1,01	1,5603e-04	1,99	1,9043e-06	2,96	2,2204e-16	—	2,2204e-16	—
160	1,2539e-02	1,00	3,9048e-05	1,99	2,4109e-07	2,98	4,4409e-16	—	4,4409e-16	—

Tableau 6.5 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF2** pour la solution $u_2(t) = t^4$ à **pas de temps constants**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	3,9300e-02	—	1,5000e-03	—	1,1102e-16	—	1,1102e-16	—
20	1,0081e-02	1,96	2,1875e-04	2,78	8,8818e-16	—	8,8818e-16	—
40	2,5207e-03	2,00	2,9297e-05	2,90	1,2212e-15	—	1,2212e-15	—
80	6,2825e-04	2,00	3,7842e-06	2,95	5,3291e-15	—	5,3291e-15	—

Dans ce cas, seules les méthodes DC4/BDF et DC5/BDF permettent d'intégrer de manière exacte le monôme d'ordre 4. Les autres méthodes font apparaître un ordre de convergence conforme aux résultats attendus.

Tableau 6.6 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF1** pour la solution $u_3(t) = e^t$ à **pas de temps constants**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	8,7346e-02	—	1,1416e-03	—	6,0257e-05	—	3,7752e-06	—	2,0441e-07	—
20	4,3315e-02	1,01	3,2425e-04	1,81	8,2530e-06	2,87	2,5944e-07	3,86	8,2533e-09	4,63
40	2,1568e-02	1,01	8,5436e-05	1,92	1,0755e-06	2,94	1,6960e-08	3,94	2,8643e-10	4,85
80	1,0762e-02	1,00	2,1876e-05	1,97	1,3715e-07	2,97	1,0834e-09	3,97	9,3929e-12	4,93
160	5,3752e-03	1,00	5,5318e-06	1,98	1,7312e-08	2,99	6,8452e-11	3,98	3,0154e-13	4,96

Tableau 6.7 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF2** pour la solution $u_3(t) = e^t$ à **pas de temps constants**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	4,9360e-03	—	1,2115e-04	—	6,6049e-06	—	4,2791e-07	—
20	1,3397e-03	1,88	1,6594e-05	2,87	4,6111e-07	3,84	1,5262e-08	4,81
40	3,4684e-04	1,95	2,1578e-06	2,94	3,0273e-08	3,93	5,0643e-10	4,91
80	8,8124e-05	1,98	2,7476e-07	2,97	1,9367e-09	3,97	1,6296e-11	4,96
160	2,2204e-05	1,99	3,4653e-08	2,99	1,2244e-10	3,98	5,2802e-13	4,95

Tableau 6.8 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF1** pour la solution $u_4(t) = -\frac{1}{t+1}$ à **pas de temps constants**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	3,6045e-02	—	5,4949e-04	—	1,1235e-04	—	4,1511e-05	—	4,6563e-06	—
20	1,8386e-02	0,97	2,2178e-04	1,31	2,0505e-05	2,45	3,1378e-06	3,73	2,9721e-07	3,97
40	9,2839e-03	0,99	7,0821e-05	1,65	3,0691e-06	2,74	2,1647e-07	3,86	1,4202e-08	4,39
80	4,6647e-03	0,99	7,0821e-05	1,82	4,1920e-07	2,87	1,4230e-08	3,93	5,4761e-10	4,70
160	2,3381e-03	1,00	5,3437e-06	1,92	5,4761e-08	2,94	9,1239e-10	3,96	1,9009e-11	4,85

Tableau 6.9 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF2** pour la solution $u_4(t) = -\frac{1}{t+1}$ à **pas de temps constants**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	2,4089e-03	—	1,6946e-04	—	3,8845e-05	—	1,4306e-05	—
20	9,2798e-04	1,38	3,4693e-05	2,29	4,0087e-06	3,28	7,1157e-07	4,33
40	2,8974e-04	1,68	5,6061e-06	2,63	3,2587e-07	3,62	2,8445e-08	4,64
80	8,1165e-05	1,84	7,9978e-07	2,81	2,3327e-08	3,80	1,0100e-09	4,82
160	2,1496e-05	1,92	1,0692e-07	2,90	1,5621e-09	3,90	3,3686e-11	4,91

Les deux derniers problèmes sont non linéaires et permettent de vérifier que nos méthodes fonctionnent bien dans ce cas, les ordres de convergences de l'erreur de chaque méthodes correspondent aux ordres théoriques. De plus, aucune des fonctions ne sont intégrées exactement et on voit donc l'apparition des ordres de convergence de l'erreur propre à chacune des méthodes. À pas constants, les résultats sont conformes à nos attentes. Nous pouvons récapituler nos résultats dans les tableaux 6.10 et 6.11.

Tableau 6.10 Ordre de convergence de la méthode **DC/BDF1** à pas constants. "inté" signifie que la méthode intègre parfaitement l'équation.

Pas de temps constant	BDF1	DC2/BDF1	DC3/BDF1	DC4/BDF1	DC5/BDF1
$u_1 = t^2$	$\mathcal{O}(k^1)$	inté	inté	inté	inté
$u_2 = t^4$	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	inté	inté
$u_3 = e^t$	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$
$u_4 = -\frac{1}{t+1}$	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$

Tableau 6.11 Ordre de convergence de la méthode **DC/BDF2** à pas constants. "inté" signifie que la méthode intègre parfaitement l'équation.

Pas de temps constant	DC2/BDF2	DC3/BDF2	DC4/BDF2	DC5/BDF2
$u_1 = t^2$	inté	inté	inté	inté
$u_2 = t^4$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	inté	inté
$u_3 = e^t$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$
$u_4 = -\frac{1}{t+1}$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$

6.1.1 Exemples à pas de temps alternés

Nous pouvons effectuer ces mêmes calculs avec des pas de temps alternés $k_1/k_2 = 4$. Ces tests ont pour but de montrer que les méthodes développées pour une EDO sont robustes et que l'on peut les utiliser pour des méthodes à pas de temps adaptatifs. Pour ne pas alourdir la section, les résultats sont présentés uniquement pour la solution analytique u_3 . Nous aborderons plus d'exemple de ce type dans le chapitre suivant.

Tableau 6.12 Erreur en norme $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF1** pour la solution $u_3(t) = e^t$ à **pas de temps alternés**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	9,2912e-02	—	2,2680e-03	—	8,8554e-05	—	5,6624e-06	—	3,0914e-07	—
20	5,2345e-02	0,83	6,5346e-04	1,80	1,2966e-05	2,77	4,4395e-07	3,67	1,3762e-08	4,49
40	2,7680e-02	0,92	1,7461e-04	1,90	1,7460e-06	2,89	3,0750e-08	3,85	4,9786e-10	4,79
80	1,4221e-02	0,96	4,5085e-05	1,95	2,2629e-07	2,95	2,0188e-09	3,93	1,6648e-11	4,90
160	7,2064e-03	0,98	1,1452e-05	1,98	2,8795e-08	2,97	1,2926e-10	3,97	5,3602e-13	4,96

Tableau 6.13 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF2** pour la solution $u_3(t) = e^t$ à **pas de temps alternés**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	4,3479e-03	—	1,1797e-04	—	7,1137e-06	—	4,5201e-07	—
20	1,2712e-03	1,77	1,7559e-05	2,75	5,6190e-07	3,66	1,8261e-08	4,63
40	3,3854e-04	1,91	2,3549e-06	2,90	3,8516e-08	3,87	6,3260e-10	4,85
80	8,7104e-05	1,96	3,0393e-07	2,95	2,5098e-09	3,94	2,0707e-11	4,93
160	2,2077e-05	1,98	3,8577e-08	2,98	1,5998e-10	3,97	6,1462e-13	5,07

Nous constatons que ces résultats à pas alternés sont quasiment similaires à ceux obtenus pour le pas variable dans les tableaux 6.6 et 6.7. Les taux de convergence attendus sont de nouveaux atteints.

6.1.2 Système de deux EDO

Considérons maintenant un système de deux équations différentielles ordinaires couplées. Pour cela, nous allons utiliser le modèle du ressort amorti :

$$\boxed{\begin{cases} m \frac{d^2x}{dt^2} + c \frac{dx}{dt} + kx = 0, \\ x(0) = x_0, \\ \frac{dx}{dt}(0) = v_0. \end{cases}} \quad (6.3)$$

Dans ce problème $x(t)$ représente l'élongation du ressort par rapport à la position d'équilibre x_0 . La variable $v = \frac{dx}{dt}$ représente la vitesse de déformation et, v_0 la vitesse initiale. Nous prenons les paramètres de masse, d'amortissement et de raideur suivants : $m = 1 \text{ kg}$, $c = 3 \text{ N m}^{-1} \text{ s}$ et $k = 2 \text{ N m}^{-1}$. Avec ces paramètres le ressort est sous-amorti et nous connaissons la solution exacte :

$$x(t) = -\frac{x_0}{\tau_1 - \tau_2} e^{-t/\tau_1} - \frac{x_0}{\tau_2 - \tau_1} e^{-t/\tau_2},$$

avec

$$\begin{aligned} \circ \tau_1 &= \sqrt{\frac{m}{k}} \frac{1}{\xi - \sqrt{\xi^2 - 1}}, \\ \circ \tau_2 &= \sqrt{\frac{m}{k}} \frac{1}{\xi + \sqrt{\xi^2 - 1}}, \\ \circ \xi &= \frac{c}{2\sqrt{km}}. \end{aligned}$$

Nous écrivons alors ce problème comme un système de deux équations d'ordre 1 :

$$\begin{cases} \frac{dv}{dt} + \frac{c}{m}v + \frac{k}{m}x = 0, \\ \frac{dx}{dt} - v = 0, \\ x(0) = x_0, \\ v(0) = v_0. \end{cases} \quad \longrightarrow \quad \frac{d}{dt} \begin{pmatrix} v \\ x \end{pmatrix} = \begin{pmatrix} -\frac{c}{m} & -\frac{k}{m} \\ 1 & 0 \end{pmatrix} \begin{pmatrix} v \\ x \end{pmatrix} \quad (6.4)$$

Nous mesurons l'erreur $L^\infty(\Omega_t)$ commise sur l'approximation du déplacement x^n .

Tableau 6.14 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF1** pour la solution manufacturée du ressort à **pas constants**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	9,1574e-03	—	1,0256e-03	—	5,9996e-05	—	1,2937e-05	—	8,9125e-07	—
20	4,9399e-03	0,89	2,7768e-04	1,88	1,2547e-05	2,26	9,5719e-07	3,76	3,0723e-08	4,86
40	2,5874e-03	0,93	7,2372e-05	1,94	1,9473e-06	2,69	6,4921e-08	3,88	1,0741e-09	4,84
80	1,3260e-03	0,96	1,8483e-05	1,97	2,6979e-07	2,85	4,2265e-09	3,94	4,1960e-11	4,68
160	6,7102e-04	0,98	4,6708e-06	1,98	3,5419e-08	2,93	2,6958e-10	3,97	1,4546e-12	4,85

Tableau 6.15 Erreur $L^\infty(\Omega_t)$ et ordre de convergence de la méthode **DC/BDF2** pour la solution manufacturée du ressort à **pas constants**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	1,3746e-03	—	1,1909e-04	—	1,5801e-05	—	2,5141e-06	—
20	5,1960e-04	1,40	1,9550e-05	2,61	1,2663e-06	3,64	9,4890e-08	4,73
40	1,5685e-04	1,73	2,7508e-06	2,83	8,8194e-08	3,84	3,2231e-09	4,88
80	4,2963e-05	1,87	3,6328e-07	2,92	5,7997e-09	3,93	1,0477e-10	4,94
160	1,1235e-05	1,94	4,6640e-08	2,96	3,7167e-10	3,96	3,3366e-12	4,97

Pour les méthodes DC/BDF1 et DC/BDF2, nous observons les taux de convergence attendus. Nous obtenons également les taux attendus à pas de temps alternés comme le montrent les résultats en annexe A.1 et A.2.

De plus nous vérifions également que nos intégrateurs fonctionnent pour des situations raides. Pour cela nous changeons les paramètres du ressort pour obtenir un système plus sensible et donc plus raide.

6.1.3 Système de deux EDO raides

Nous allons tester le même problème dans une situation plus raide. La définition mathématique de la raideur est controversée. Nous pouvons tout de même citer les travaux de Curtiss et Hirschfelder [28] pour qui la raideur d'une équation se définit par le fait que les méthodes implicites et plus particulièrement les BDF sont plus performantes que les méthodes explicites sur ce type de problème. Dans la pratique nous pouvons définir un coefficient de raideur.

Définition 3. *Coefficient de raideur*

Considérons un système linéaire dont les valeurs propres minimales et maximales sont λ_{min} et λ_{max} . Le coefficient de raideur correspond au rapport des parties réelles de ces deux extrema :

$$\frac{|Re(\lambda_{max})|}{|Re(\lambda_{min})|}.$$

En reprenant l'équation du ressort (8.1) et en changeant les paramètres :

- $m = 1 \text{ kg}$,
- $c = 1001 \text{ N m}^{-1} \text{ s}$,
- $k = 1000 \text{ N m}^{-1}$.

Nous obtenons la solution exacte suivante :

$$x(t) = x_0 \left(-\frac{1}{999} \exp(-t) + \frac{1000}{999} \exp(-1000t) \right). \quad (6.5)$$

Cette fois le coefficient de raideur vaut 1000.

Tableau 6.16 Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système ressort pour une simulation à **pas de temps constants** avec un **coefficient de raideur de 1000**.

Nb de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	1,7682e-02	0,00	6,0211e-04	0,00	3,8043e-04	0,00	4,7553e-04	0,00	7,6488e-04	0,00
20	9,0191e-03	0,97	4,7920e-04	0,33	3,6145e-04	0,07	4,3471e-04	0,13	6,9090e-04	0,15
40	4,5557e-03	0,99	4,2665e-04	0,17	3,1369e-04	0,20	3,6368e-04	0,26	5,6492e-04	0,29
80	2,2896e-03	0,99	3,5527e-04	0,26	2,3341e-04	0,43	2,5470e-04	0,51	3,8122e-04	0,57
160	1,1478e-03	1,00	2,3586e-04	0,59	1,2365e-04	0,92	1,2557e-04	1,02	1,8103e-04	1,07
320	5,7464e-04	1,00	8,8567e-05	1,41	2,8542e-05	2,12	3,3209e-05	1,92	5,2199e-05	1,79
640	2,8751e-04	1,00	3,6778e-05	1,27	6,7709e-06	2,08	5,6986e-06	2,54	8,8666e-06	2,56
1280	1,4380e-04	1,00	1,2690e-05	1,54	1,1870e-06	2,51	1,1383e-06	2,32	6,5961e-07	3,75
2560	7,1912e-05	1,00	3,8212e-06	1,73	1,8762e-07	2,66	1,4717e-07	2,95	2,6022e-08	4,66
5120	3,5959e-05	1,00	1,0595e-06	1,85	8,0913e-08	1,21	1,2747e-08	3,53	9,9629e-10	4,71
10240	1,7980e-05	1,00	2,8072e-07	1,92	1,5497e-08	2,38	9,3001e-10	3,78	3,3664e-11	4,89
20480	8,9902e-06	1,00	7,2309e-08	1,96	2,3273e-09	2,74	6,2674e-11	3,89	1,0901e-12	4,95

Tableau 6.17 Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système ressort pour une simulation à **pas de temps constants** avec un **coefficient de raideur de 1000**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	8,9316e-04	0,00	2,2967e-05	0,00	1,3300e-06	0,00	4,6561e-05	0,00
20	2,6500e-04	1,75	3,4234e-06	2,75	9,7539e-08	3,77	4,5893e-05	0,02
40	7,1468e-05	1,89	4,5536e-07	2,91	6,4610e-09	3,92	4,4594e-05	0,04
80	1,8522e-05	1,95	5,8472e-08	2,96	4,1438e-10	3,96	4,2204e-05	0,08
160	4,7126e-06	1,97	7,4018e-09	2,98	1,2812e-07	-8,27	3,7651e-05	0,16
320	1,1884e-06	1,99	1,8227e-07	-4,62	1,8909e-06	-3,88	2,4150e-05	0,64
640	3,0916e-06	-1,38	1,6179e-06	-3,15	2,6005e-06	-0,46	6,1704e-06	1,97
1280	7,6726e-06	-1,31	1,8545e-06	-0,20	9,5309e-07	1,45	8,2503e-07	2,90
2560	6,0587e-06	0,34	7,0632e-07	1,39	1,6318e-07	2,55	5,2454e-08	3,98
5120	2,7083e-06	1,16	1,4965e-07	2,24	1,6720e-08	3,29	2,3991e-09	4,45
10240	8,9334e-07	1,60	2,3473e-08	2,67	1,3051e-09	3,68	8,9670e-11	4,74
20480	2,5580e-07	1,80	3,2480e-09	2,85	9,0041e-11	3,86	3,0314e-12	4,89

Ces résultats sont intéressants à analyser. Pour le schéma DC/BDF1, les taux de convergence

n'apparaissent qu'avec un nombre conséquent de pas de temps. L'équation étant très raide, les variations sont très brusques et requièrent donc un pas de temps très fin pour être capturé. En ce qui concerne le schéma DC/BDF2, nous pouvons voir apparaître rapidement les ordres de convergence attendus. Notre hypothèse est que la brusque variation est lissée et n'apparaît donc pas. Cependant, en raffinant suffisamment, nous sommes en mesure de capturer la variation. Et en raffinant encore nous pouvons voir apparaître les taux de convergence conformes à la méthode DC/BDF2. Pour ce problème très raide, nos intégrateurs aboutissent aux taux de convergence attendus.

Conclusion

Nous avons effectué des tests sur des EDO et un système d'EDO, dans toutes ces situations les schémas DC/BDF permettent d'atteindre les bons taux de convergence. Nous avons également simulé un problème raide pour observer le comportement de nos méthodes. Après un raffinement du pas de temps beaucoup plus important, les taux de convergence sont également atteints. Pour compléter cette étude nous avons effectué des tests sur un système de trois EDO, disponible en annexe B.

CHAPITRE 7 VERIFICATION POUR DES EDP EN 1D

Nous avons pu vérifier les méthodes DC/BDF1 et DC/BDF2 pour les EDO, nous présentons maintenant les résultats pour les EDP. Les problèmes d'EDP sont plus complexes, car ils requièrent une discrétisation spatiale. Nous utilisons pour cela la méthode des éléments finis. Nous effectuons des simulations sur l'équation de la chaleur en une dimension spatiale avec un terme source. Nous testons pour plusieurs solutions analytiques que nous construisons par la méthode des solutions manufacturées.

Considérons l'équation modèle sur le domaine temporel $\Omega_t = [0, 1] \in \mathbb{R}$ et sur le domaine spatial $\Omega_x = [0, 5] \in \mathbb{R}$. Nous notons $\Gamma = \partial\Omega_x$ le bord du domaine :

$$\boxed{\begin{cases} \frac{\partial T}{\partial t} - k\Delta T & = s(x, t), \\ T(x, 0) & = T_0, \\ T(x, t) & = T_\Gamma(x, t) \text{ pour } x \in \Gamma. \end{cases}} \quad (7.1)$$

Comme décrit au chapitre 6, nous utilisons la méthode des solutions manufacturées afin de vérifier nos résultats. Le principe est toujours de construire le terme source satisfaisant la solution exacte $T_{\text{exacte}}(x, t)$, mais cette fois l'équation étant différente, le terme source prend la forme :

$$s(x, t) = \frac{\partial T_{\text{exacte}}}{\partial t} - k \frac{\partial^2 T_{\text{exacte}}}{\partial^2 x} \quad \text{où } k = 0.1 \text{ W m K}^{-1}.$$

7.1 Erreurs de discrétisation

Dans cette situation, nous devons discrétiser temporellement et spatialement notre problème. Nous devons tout d'abord définir une norme. Spatialement nous mesurons l'erreur en norme L^2 :

$$\mathcal{E}_x = \|u(x, t^n) - u^n\|_{L^2(\Omega)} = \sqrt{\int_{\Omega_x} (u(x, t) - u^{n,h})^2 dx}. \quad (7.2)$$

L'erreur temporelle est toujours mesurée par le maximum des erreurs sur le domaine comme défini dans la section 6.0.2. Nous avons donc la norme suivante :

$$\mathcal{E} = \|u(t^n) - u^n\|_{L^2(\Omega_x), L^\infty(\Omega_t)} = \sup_{t \in [0, T]} \sqrt{\int_{\Omega_x} (u(x, t) - u^{n,h})^2 dx}. \quad (7.3)$$

L'erreur de discrétisation temporelle dépend de notre intégrateur et de la taille des pas de temps utilisée. Notre domaine spatial est découpé en éléments de taille caractéristique h . L'erreur de discrétisation spatiale dépend alors cette taille caractéristique et du type d'élément utilisé. L'erreur peut donc s'écrire [29] :

$$\mathcal{E} \sim \mathcal{O}(k^p, h^m). \quad (7.4)$$

où p correspond à l'ordre de convergence de la méthode temporelle et m l'ordre de convergence en espace.

Dans l'approximation élémentaire, nous utiliserons des polynômes de Lagrange de degrés m , notés P^m , pour lesquels l'erreur en norme L^2 satisfait [30] :

$$\mathcal{E}_x \sim \mathcal{O}(h^{m+1}). \quad (7.5)$$

Cependant nos tests portent sur des méthodes temporelles et nous cherchons donc à nous affranchir de cette erreur. Deux stratégies sont alors possibles. Nous pouvons tout d'abord utiliser une solution entièrement capturée par nos éléments, par exemple un polynôme en x^3 sera parfaitement capturé par une approximation polynômiale d'ordre 3 ou supérieur. En faisant cela, nous sommes certains de ne mesurer que des erreurs temporelles et nous nous assurons ainsi que les ordres observés ne dépendent que de nos méthodes temporelles. Dans le cas de solutions non polynômiales, une erreur d'interpolation subsiste. L'idée est cette fois de la faire décroître plus rapidement que l'erreur temporelle pour que les ordres de convergences reflètent l'effet de la méthode temporelle seulement. Cela consiste à appliquer un raffinement spatial plus rapide que le raffinement temporel.

Méthode de quadrature

Enfin, un dernier point concerne la méthode utilisée pour calculer les intégrales qui apparaissent dans les formulations faibles lors d'une discrétisation par éléments finis. Pour cela, nous utilisons une quadrature de Gauss-Legendre, qui permet d'évaluer de manière exacte des polynômes de degré n avec $2n - 1$ points. Nous utilisons également cette quadrature pour le calcul de l'erreur. Nous veillerons donc à choisir un nombre de points suffisant pour minimiser l'erreur d'intégration. Dans nos simulations, nous choisissons assez de points de quadratures pour intégrer exactement un polynôme de degré 10.

Conditions limites

Nous allons utiliser des conditions limites de Dirichlet en imposant directement la solution exacte aux bords. Nous allons différencier deux types de conditions limites de Dirichlet : les conditions limites dépendantes et indépendantes du temps. Nous discuterons dans la partie 7.4 des enjeux de ces conditions limites.

Initialisation de la méthode DC/BDF2

La question de l'initialisation se pose encore pour le solveur DC/BDF2. Pour le moment, nous choisissons d'utiliser la solution théorique pour évaluer les premières solutions. Cela n'est possible que dans le cas des solutions manufacturées et nous permet de nous concentrer sur les autres paramètres mis en jeu lors de nos simulations. Nous verrons dans les tests effectués à la section 7.5 que l'initialisation par le DC/BDF1 fonctionne également.

Solutions analytiques et termes sources

Les deux solutions analytiques choisies pour les test de cette section sont données dans le tableau 7.1, ainsi que le terme source associé calculé par la méthode des solutions manufacturées. Ces problèmes sont choisis, car la partie spatiale est parfaitement capturée par l'interpolation de Lagrange P^4 que nous allons utiliser, et font donc apparaître uniquement une erreur temporelle. Les deux problèmes diffèrent par leurs conditions limites, constantes aux bords ou dépendantes du temps.

Tableau 7.1 Solutions manufacturées dont les conditions limites dépendent ou non du temps

Solution exacte	Terme source
$u_1(x, t) = x(x^2 - 25)t^6$	$s(x, t) = 6x(x^2 - 25)t^5 - 6kxt^6$
$u_2(x, t) = x(x^2 - 25)t^6 + x^3t^6 + 1$	$s(x, t) = 6x(x^2 - 25)t^5 + 6x^3t^5 - 12kxt^6$

7.2 Tests à pas de temps constants

Tout d'abord, nous nous intéressons aux simulations à pas constants. C'est la première étape afin de valider nos schémas. Nous intégrons les EDP avec deux types de conditions limites de Dirichlet : dépendantes ou pas du temps sur les deux schémas DC/BDF1 et DC/BDF2.

7.2.1 Conditions limites indépendantes du temps

Tableau 7.2 Erreur et ordre de convergence de la méthode **DC/BDF1** à **pas de temps constants** pour la solution \mathbf{u}_1 donc à **condition limites indépendantes** du temps.

Nb de pas de temps	Nb d'éléments	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
		\mathcal{E}	p								
10	10	5,6660e-01	—	3,8499e-02	—	5,0635e-03	—	5,8651e-04	—	6,6853e-06	—
20	20	2,7528e-01	1,04	9,6721e-03	1,99	6,6823e-04	2,92	3,9929e-05	3,88	2,1954e-07	4,93
40	40	1,3562e-01	1,02	2,4237e-03	2,00	8,5456e-05	2,97	2,5971e-06	3,94	7,0200e-09	4,97
80	80	6,7304e-02	1,01	6,0663e-04	2,00	1,0794e-05	2,98	1,6548e-07	3,97	2,2183e-10	4,98
160	160	3,3525e-02	1,01	1,5174e-04	2,00	1,3559e-06	2,99	1,0441e-08	3,99	6,9706e-12	4,99

Tableau 7.3 Erreur et ordre de convergence de la méthode **DC/BDF2** à **pas de temps constants** pour la solution \mathbf{u}_1 donc à **condition limites indépendantes** du temps.

Nombre de pas de temps	Nombre d'éléments	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
		\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	8,7385e+00	—	7,8623e-01	—	1,1636e-01	—	1,3129e-02	—
20	20	2,0312e+00	2,11	9,6969e-02	3,02	7,7567e-03	3,91	4,7694e-04	4,78
40	40	4,9161e-01	2,05	1,2019e-02	3,01	4,9625e-04	3,97	1,5940e-05	4,90
80	80	1,2113e-01	2,02	1,4964e-03	3,01	3,1317e-05	3,99	5,1428e-07	4,95
160	160	3,0079e-02	2,01	1,8672e-04	3,00	1,9658e-06	3,99	1,6323e-08	4,98

7.2.2 Conditions limites dépendantes du temps

Tableau 7.4 Erreur et ordre de convergence de la méthode **DC/BDF1** à **pas de temps constants** pour la solution \mathbf{u}_2 donc à **condition limites dépendantes** du temps.

Nb de pas de temps	Nb d'éléments	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
		\mathcal{E}	p								
10	10	2,3782e+01	—	2,3976e+00	—	3,9388e-01	—	6,6299e-02	—	7,2409e-03	—
20	20	1,1589e+01	1,04	6,1171e-01	1,97	5,2522e-02	2,91	4,5283e-03	3,87	2,7696e-04	4,71
40	40	5,7169e+00	1,02	1,5450e-01	1,99	6,7748e-03	2,95	2,9516e-04	3,94	9,4321e-06	4,88
80	80	2,8388e+00	1,01	3,8821e-02	1,99	8,6010e-04	2,98	1,8829e-05	3,97	3,0678e-07	4,94
160	160	1,4144e+00	1,01	9,7299e-03	2,00	1,0835e-04	2,99	1,1888e-06	3,99	9,7737e-09	4,97

Tableau 7.5 Erreur et ordre de convergence de la méthode **DC/BDF2** à **pas de temps constant** pour la solution \mathbf{u}_2 donc à **condition limites dépendantes** du temps.

Nombre de pas de temps	Nombre d'éléments	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
		\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	8,3728e+00	—	7,6969e-01	—	1,0913e-01	—	1,1883e-02	—
20	20	1,9544e+00	2,10	9,3330e-02	3,04	7,1594e-03	3,93	4,2037e-04	4,82
40	40	4,7385e-01	2,04	1,1458e-02	3,03	4,5504e-04	3,98	1,3879e-05	4,92
80	80	1,1683e-01	2,02	1,4192e-03	3,01	2,8631e-05	3,99	4,4512e-07	4,96
160	160	2,9016e-02	2,01	1,7659e-04	3,01	1,7948e-06	4,00	1,4087e-08	4,98

Ces simulations permettent de montrer qu'à pas de temps constants, quelles que soient les conditions limites, nos schémas convergent avec les taux attendus.

7.3 Tests à pas de temps alternés

Nous simulons avec des pas de temps alternés de ratio $k_1/k_2 = 4$, afin de tester la robustesse des DC/BDF dans une situation d'adaptation du pas de temps.

7.3.1 Conditions limites indépendantes du temps

Tableau 7.6 Erreur et ordre de convergence de la méthode **DC/BDF1** à pas de temps alternés pour la solution \mathbf{u}_1 donc à condition limites indépendantes du temps.

Nb de pas de temps	Nb d'éléments	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
		\mathcal{E}	p								
10	10	1,7165e+01	—	2,2038e+00	—	3,7725e-01	—	8,5606e-02	—	1,0943e-02	—
20	20	1,1748e+01	0,55	7,6131e-01	1,53	6,3146e-02	2,58	7,0035e-03	3,61	4,7167e-04	4,54
40	40	6,7971e+00	0,79	2,2102e-01	1,78	9,0308e-03	2,81	4,9351e-04	3,83	1,7292e-05	4,77
80	80	3,6469e+00	0,90	5,9385e-02	1,90	1,2045e-03	2,91	3,2654e-05	3,92	7,5801e-07	4,51
160	160	1,8878e+00	0,95	1,5382e-02	1,95	1,5543e-04	2,95	2,0986e-06	3,96	6,0755e-08	3,64

Tableau 7.7 Erreur et ordre de convergence de la méthode **DC/BDF2** à pas de temps alternés pour la solution \mathbf{u}_1 donc à condition limites indépendantes du temps.

Nombre de pas de temps	Nombre d'éléments	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
		\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	7,1785e+00	—	7,5543e-01	—	1,2795e-01	—	1,3937e-02	—
20	20	1,8549e+00	1,95	1,0202e-01	2,89	9,6202e-03	3,73	5,7464e-04	4,60
40	40	4,7045e-01	1,98	1,3100e-02	2,96	6,3923e-04	3,91	2,0112e-05	4,84
80	80	1,1854e-01	1,99	1,6576e-03	2,98	4,0941e-05	3,96	6,6209e-07	4,92
160	160	2,9758e-02	1,99	2,0845e-04	2,99	2,5867e-06	3,98	2,1233e-08	4,96

À pas de temps alternés nous observons des changements, mais nos méthodes convergent aux taux attendus dans le cas où les conditions limites sont indépendantes du temps. Notons tout de même le résultat du DC5/BDF1 : **le schéma ne converge pas à l'ordre 5, nous n'avons pas d'explication sur ce phénomène.**

7.3.2 Conditions limites dépendantes du temps

Tableau 7.8 Erreur et ordre de convergence de la méthode **DC/BDF1** à pas de temps alternés pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps.

Nb de pas de temps	Nb d'éléments	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
		\mathcal{E}	p								
10	10	1,6206e+01	—	2,6466e+00	—	5,7453e-01	—	1,9855e-01	—	1,3110e-01	0,00
20	20	1,1308e+01	0,52	9,5508e-01	1,47	1,2249e-01	2,23	5,7176e-02	1,80	7,4398e-02	0,82
40	40	6,6054e+00	0,78	2,8298e-01	1,75	2,3209e-02	2,40	1,5515e-02	1,88	2,6865e-02	1,47
80	80	3,5610e+00	0,89	7,6788e-02	1,88	4,3802e-03	2,41	3,6981e-03	2,07	7,1548e-03	1,91
160	160	1,8478e+00	0,95	1,9988e-02	1,94	8,5717e-04	2,35	8,2739e-04	2,16	1,6741e-03	2,10

Tableau 7.9 Erreur et ordre de convergence de la méthode **DC/BDF2** à pas de temps alternés pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps.

Nombre de pas de temps	Nombre d'éléments	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
		\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	6,8678e+00	—	7,6999e-01	—	1,4810e-01	—	7,0941e-02	—
20	20	1,7828e+00	1,95	1,0352e-01	2,89	1,4623e-02	3,34	1,8070e-02	1,97
40	40	4,5319e-01	1,98	1,3348e-02	2,96	1,5754e-03	3,21	3,1414e-03	2,52
80	80	1,1429e-01	1,99	1,6924e-03	2,98	1,7142e-04	3,20	4,0900e-04	2,94
160	160	2,8702e-02	1,99	2,1302e-04	2,99	1,8413e-05	3,22	4,6525e-05	3,14

Cette fois, cependant, nous remarquons une perte d'ordre pour les méthodes DC4/BDF2, DC5/BDF2, ainsi que pour les méthodes DC3/BDF1, DC4/BDF1 et DC5/BDF1 lorsque les conditions limites dépendent du temps. Autrement dit une méthodes corrigée deux fois ou plus subit un phénomène de perte d'ordre. Pour comprendre la source de ces pertes d'ordre, nous traçons la différence entre la solution analytique et numérique sur chaque éléments du domaine et pour plusieurs pas de temps.

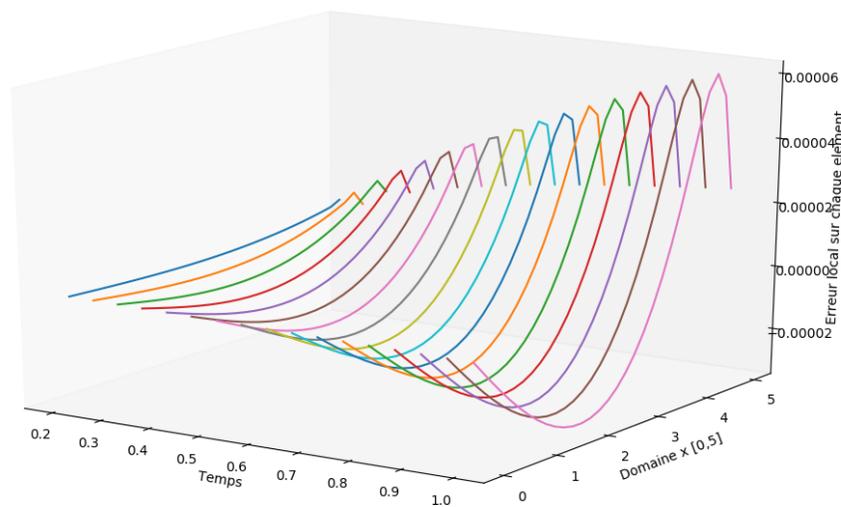


Figure 7.1 Répartition de l'erreur sur le domaine pour plusieurs de pas de temps constants

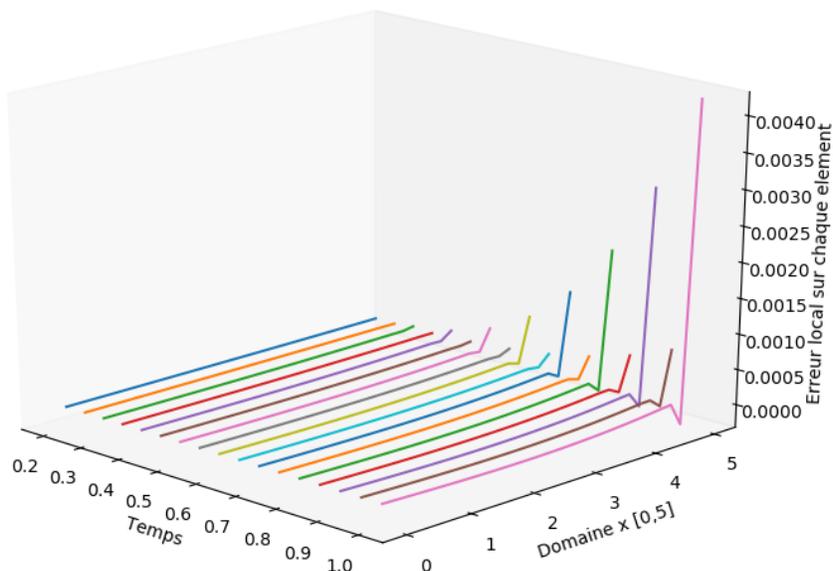


Figure 7.2 Répartition de l'erreur sur le domaine pour plusieurs de **pas de temps alternés**

Nous observons que les erreurs à pas constants sont lisses et réparties sur le domaine tandis que les erreurs à pas variables sont concentrées à la frontière où nous imposons les conditions limites qui dépendent du temps. Nous pouvons donc penser que ces erreurs sont directement liées à la manière d'imposer nos conditions limites. C'est la raison pour laquelle nous développons une imposition des conditions limites à l'aide de la correction de Verwer, comme nous l'avons vu dans le chapitre 5.

7.4 Correction de Verwer

Nous reprenons ici les simulations résumées dans les tableaux 7.5 et 7.9, en imposant cette fois les conditions limites selon la méthodologie de Verwer. Comme expliqué à la section 5.2 nous imposons la condition limite par une équation différentielle du premier ordre au bord. Cette contrainte est prise en compte dans la formulation faible de notre problème au moyen d'un multiplicateur de Lagrange λ . La résolution aboutit désormais à un résultat sur la variable primaire, soit la température T dans le cas d'un problème de diffusion thermique, mais également sur la variable secondaire λ . De plus, en observant la formulation faible faisant intervenir ce multiplicateur (5.3), nous avons donné un sens physique à ce dernier, le multiplicateur λ représente ici le flux de chaleur aux bords :

$$\lambda = -k \left. \frac{\partial T}{\partial n} \right|_{\text{bords}} .$$

Tableau 7.10 Erreur et ordre de convergence de la méthode **DC/BDF2** à pas de temps constants pour la solution manufacturée \mathbf{u}_2 donc à condition limites dépendantes du temps avec la méthode de Verwer.

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire T									
Nombre de pas de temps	Nombre d'elements	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	1,0385e+01	—	9,3093e-01	—	1,3858e-01	—	1,5708e-02	—
20	20	2,4125e+00	2,11	1,1511e-01	3,02	9,2608e-03	3,90	5,7331e-04	4,78
40	40	5,8376e-01	2,05	1,4288e-02	3,01	5,9313e-04	3,96	1,9206e-05	4,90
80	80	1,4383e-01	2,02	1,7803e-03	3,00	3,7450e-05	3,99	6,2035e-07	4,95
160	160	3,5713e-02	2,01	2,2223e-04	3,00	2,3514e-06	3,99	1,9701e-08	4,98
Variable secondaire λ									
10	10	1,3284e+00	0,00	1,2016e-01	0,00	1,7563e-02	0,00	1,9600e-03	0,00
20	20	3,0915e-01	2,10	1,4752e-02	3,03	1,1657e-03	3,91	7,0704e-05	4,79
40	40	7,4861e-02	2,05	1,8237e-03	3,02	7,4449e-05	3,97	2,3555e-06	4,91
80	80	1,8449e-02	2,02	2,2674e-04	3,01	4,6946e-06	3,99	7,5879e-08	4,96
160	160	4,5814e-03	2,01	2,8271e-05	3,00	2,9458e-07	3,99	2,4073e-09	4,98

Premièrement, nous pouvons voir que cette implémentation des conditions limites ne change pas les résultats à pas constants et les ordres sur le multiplicateur correspondent à ceux attendus.

Tableau 7.11 Erreur et ordre de convergence de la méthode **DC/BDF2** à pas de temps alternés pour la solution manufacturée \mathbf{u}_2 donc à condition limites dépendantes du temps avec la méthode de Verwer.

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire T									
Nombre de pas de temps	Nombre d'elements	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	8,5312e+00	0,00	8,9031e-01	0,00	1,5118e-01	0,00	1,6280e-02	0,00
20	20	2,2032e+00	1,95	1,2046e-01	2,89	1,1403e-02	3,73	6,7319e-04	4,60
40	40	5,5865e-01	1,98	1,5481e-02	2,96	7,5846e-04	3,91	2,3565e-05	4,84
80	80	1,4075e-01	1,99	1,9598e-03	2,98	4,8593e-05	3,96	7,7398e-07	4,93
160	160	3,5332e-02	1,99	2,4651e-04	2,99	3,0704e-06	3,98	2,4699e-08	4,97
Variable secondaire λ									
10	10	1,0903e+00	0,00	1,1679e-01	0,00	2,0028e-02	0,00	2,8955e-03	0,00
20	20	2,8215e-01	1,95	1,5782e-02	2,89	1,5790e-03	3,66	2,4062e-04	3,59
40	40	7,1615e-02	1,98	2,0297e-03	2,96	1,1159e-04	3,82	2,4876e-05	3,27
80	80	1,8051e-02	1,99	2,5723e-04	2,98	7,7093e-06	3,86	2,5269e-06	3,30
160	160	4,5321e-03	1,99	3,2406e-05	2,99	5,3457e-07	3,85	2,3987e-07	3,40

À pas alternés, la méthode permet de régler le problème de la perte d'ordre sur la variable primaire, on a en revanche une perte d'ordre qui apparaît sur le multiplicateur. Ainsi, cette

méthode a certes le défaut d'agrandir le système matriciel à résoudre, mais elle fournit également de l'information supplémentaire sur le problème simulé et permet de solutionner le problème de perte d'ordre lorsque les conditions limites du problème dépendaient du temps.

Nous utilisons également la correction de Verwer pour le méthode DC/BDF1. Dans ce cas nous observons que cette correction n'est efficace que pour le DC3/BDF1. Les discontinuités semblent avoir été lissées au vu de l'ordre de grandeur des erreurs, mais les taux de convergence optimaux des méthodes DC4/BDF1 et DC5/BDF1 ne sont pas atteints.

Tableau 7.12 Erreur et ordre de convergence de la méthode **DC/BDF1** à **pas de temps alternés** pour la solution \mathbf{u}_2 donc à **condition limites dépendantes** du temps avec la **méthode de Verwer**.

		BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
Variable primaire T											
Nombre de pas de temps	Nombre d'éléments	\mathcal{E}	p								
10	10	4,5838e+02	0,00	4,9800e+01	0,00	7,8062e+00	0,00	1,6145e+00	0,00	3,4932e-01	0,00
20	20	3,0733e+02	0,58	1,6642e+01	1,58	1,2087e+00	2,69	1,0053e-01	4,01	7,1749e-02	2,28
40	40	1,7603e+02	0,80	4,7464e+00	1,81	1,6294e-01	2,89	7,5826e-03	3,73	1,0543e-02	2,77
80	80	9,3977e+01	0,91	1,2631e+00	1,91	2,0723e-02	2,98	1,0374e-03	2,87	1,2906e-03	3,03
160	160	4,8527e+01	0,95	3,2546e-01	1,96	2,5732e-03	3,01	1,2658e-04	3,03	1,4598e-04	3,14
Variable secondaire λ											
10	10	5,4820e+00	0,00	1,3947e-01	0,00	3,5105e-01	0,00	7,1870e-02	0,00	1,3343e-02	0,00
20	20	3,3868e+00	0,69	1,3687e-01	0,03	7,2104e-02	2,28	6,7033e-03	3,42	5,2980e-04	4,65
40	40	1,8383e+00	0,88	5,2442e-02	1,38	1,1157e-02	2,69	4,9676e-04	3,75	1,8319e-05	4,85
80	80	9,5247e-01	0,95	1,5681e-02	1,74	1,5407e-03	2,86	3,3589e-05	3,89	6,0049e-07	4,93
160	160	4,8416e-01	0,98	4,2597e-03	1,88	2,0208e-04	2,93	2,1801e-06	3,95	1,9212e-08	4,97

En conclusion, nous pouvons récapituler les résultats de nos différentes méthodes.

Tableau 7.13 Taux de convergence des méthodes **DC/BDF1** et **DC/BDF2** pour l'équation de la chaleur en une dimension spatiale.

	BDF1	DC2/BDF1	DC3/BDF1	DC4/BDF1	DC5/BDF1
u_1 et $k_1/k_2 = 1$	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$
u_2 et $k_1/k_2 = 1$	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$
u_1 et $k_1/k_2 = 4$	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$
u_2 et $k_1/k_2 = 4$	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	Perte d'ordre	Perte d'ordre	Perte d'ordre
u_2 et $k_1/k_2 = 4$ \hookrightarrow Verwer	$\mathcal{O}(k^1)$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	Perte d'ordre	Perte d'ordre
	BDF2	DC3/BDF2	DC4/BDF2	DC5/BDF2	
u_1 et $k_1/k_2 = 1$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$	
u_2 et $k_1/k_2 = 1$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$	
u_1 et $k_1/k_2 = 4$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$	
u_2 et $k_1/k_2 = 4$	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	Perte d'ordre	Perte d'ordre	
u_2 et $k_1/k_2 = 4$ \hookrightarrow Verwer	$\mathcal{O}(k^2)$	$\mathcal{O}(k^3)$	$\mathcal{O}(k^4)$	$\mathcal{O}(k^5)$	

7.5 Initialisation du DC/BDF2 à l'aide du DC/BDF1

Nous avons évoqué dans la section 4.3 la nécessité d'utiliser le schéma DC/BDF1 pour initialiser la méthode DC/BDF2. Nous allons observer les résultats de cette combinaison. Pour cette situation, nous n'avons plus besoin de connaître la solution analytique, la condition initiale suffit. Nous allons reproduire les résultats obtenus dans les tableaux 7.10 et 7.11. Notre but est de montrer qu'utiliser les valeurs provenant du schéma DC/BDF1 permet bien de maintenir une erreur conforme avec le schéma DC/BDF2.

Tableau 7.14 Erreur et ordre de convergence de la méthode **DC/BDF2** à pas de temps constants pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps avec méthode de Verwer et avec initialisation DC/BDF1.

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire T									
Nombre de pas de temps	Nombre d'elements	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	6,9775e+00	—	6,9274e-01	—	5,8975e-01	—	8,4862e-02	—
20	20	1,9727e+00	1,82	9,8963e-02	2,81	7,0527e-03	3,87	8,6279e-04	4,75
40	40	5,2792e-01	1,90	1,3247e-02	2,90	6,3617e-04	3,93	2,26e-05	4,89
80	80	1,3679e-01	1,95	1,7144e-03	2,95	3,8912e-05	3,96	6,8712e-07	4,93
160	160	3,4830e-02	1,97	2,1809e-04	2,97	2,5618e-06	3,98	1,9431e-08	4,97
Variable secondaire λ									
10	10	8,9535e-01	—	8,9716e-02	—	1,5619e-02	—	1,5431e-03	—
20	20	2,5326e-01	1,82	1,2710e-02	2,82	1,5631e-03	3,84	7,5648e-05	4,80
40	40	6,7765e-02	1,90	1,6928e-03	2,91	7,6718e-05	3,95	2,4672e-06	4,98
80	80	1,7554e-02	1,95	2,1847e-04	2,95	4,6054e-06	3,98	7,4723e-08	4,95
160	160	4,4691e-03	1,97	2,7751e-05	2,98	2,8925e-07	3,99	2,3615e-09	4,97

Tableau 7.15 Erreur et ordre de convergence de la méthode **DC/BDF2** à pas de temps alternés pour la solution \mathbf{u}_2 donc à condition limites dépendantes du temps avec méthode de Verwer et avec initialisation DC/BDF1.

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire T									
Nombre de pas de temps	Nombre d'elements	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	10	7,9923e+00	—	8,7666e-01	—	1,4328e-01	—	1,4932e-02	—
20	20	2,1175e+00	1,92	1,1790e-01	2,89	1,0304e-02	3,75	6,3719e-04	4,63
40	40	5,4730e-01	1,95	1,5289e-02	2,95	8,1257e-04	3,92	1,9671e-05	4,86
80	80	1,3930e-01	1,97	1,9472e-03	2,97	4,6217e-05	3,97	7,5931e-07	4,93
160	160	3,5148e-02	1,99	2,4571e-04	2,99	3,1167e-06	3,98	2,4531e-08	4,96
Variable secondaire λ									
10	10	1,0265e+00	—	1,1645e-01	—	2,1097e-02	—	2,7980e-03	—
20	20	2,7203e-01	1,92	1,5615e-02	2,90	1,4410e-03	3,69	2,5931e-04	3,56
40	40	7,0278e-02	1,95	2,0205e-03	2,95	1,1831e-04	3,83	2,0671e-05	3,43
80	80	1,7880e-02	1,97	2,5699e-04	2,97	7,8207e-06	3,85	1,6812e-06	3,29
160	160	4,5105e-03	1,99	3,2406e-05	2,99	5,3917e-07	3,86	1,9374e-07	3,21

Ces tests montrent que l'initialisation du DC/BDF2 par le DC/BDF1 mène aux taux es-comptés. Ces résultats sont en ligne avec ceux obtenus avec l'initialisation avec la solution exacte. Par la suite, cela nous permettra d'utiliser par la suite le DC/BDF2 sans connaître de solution analytique.

Conclusion

Dans ce chapitre, nous avons vérifié la construction et l'implémentation de nos intégrateurs temporels sur des EDP en temps et en une dimension d'espace. Le problème d'intérêt est l'EDP linéaire de diffusion thermique. Nous avons également identifié un phénomène de perte d'ordre intervenant lors de l'utilisation de pas de temps alternés et de conditions limites dépendantes du temps. Cela nous a amené à imposer faiblement les conditions limites selon la méthode de Verwer. Dans le prochain chapitre, nous testons nos intégrateurs et la méthode de Verwer dans le cas bidimensionnel en espace.

CHAPITRE 8 VERIFICATION ET VALIDATION DES EDP EN 2D

Nous allons maintenant nous intéresser à des situations bidimensionnelles en espace. Nous avons vu que les enjeux concernant les simulations en une dimension portaient principalement sur l'imposition de conditions limites aux frontières du domaine de calcul, c'est-à-dire sur des points. Maintenant, les frontières sont des courbes, ce qui soulève des enjeux supplémentaires concernant, entre autres, la régularité de l'approximation du multiplicateur de Lagrange intervenant lors de l'imposition faible des conditions essentielles.

8.1 Équation de la chaleur en 2D

Considérons de nouveau l'équation modèle sur le domaine temporel $\Omega_t = [0, 1] \in \mathbb{R}$ et sur le domaine spatial $\Omega_x = [-1, 1] \times [-1, 1] \in \mathbb{R}^2$. Nous notons $\Gamma = \partial\Omega_x$ le bord du domaine :

$$\left\{ \begin{array}{l} \frac{\partial T}{\partial t} - k\Delta T = s(x, y, t), \\ T(x, y, 0) = T_0, \\ T(x, y, t) = T_\Gamma(x, y, t) \text{ pour } (x, y) \in \Gamma. \end{array} \right. \quad (8.1)$$

Pour cette étude, nous utilisons une solution polynomiale avec une conductivité thermique $k = 0.1 \text{ W m K}^{-1}$:

$$T(x, y, t) = t^8(x^2 + 1)y. \quad (8.2)$$

Le terme source que l'on applique vaut donc :

$$s(x, y, t) = 8t^7(x^2 + 1)y - 2kt^8y. \quad (8.3)$$

Paramètres de la simulation

Nous avons vu que la méthode de Verwer permettait de contrer la perte d'ordre lorsque les pas de temps sont variables. Nous allons donc effectuer toutes nos simulations avec cette méthode. Pour la suite dans simulations, nous utilisons les paramètres suivant :

- Interpolant de la variable primaire P^3 ,
- Interpolant de la variable secondaire P^3 . Nous faisons l'hypothèse que l'approximation de λ est continue, ce qui est une hypothèse forte concernant la régularité du multiplicateur,
- Interpolant géométrique P^1 ,

- Degré de quadrature : 15,
- Nombre de pas de temps allant de 10 à 80,
- Initialisation de DC/BDF2 avec la solution exacte,
- Imposition des conditions limites par la méthode de Verwer.

Maillage

Nous générons les maillages avec Gmsh en utilisant une triangulation de Delaunay. Nous obtenons un premier maillage de 40 éléments. Nous raffinons ensuite en divisant la taille des arêtes par 2 pour obtenir une suite de maillages imbriqués. Cela nous permet de garder un contrôle sur la taille caractéristique des éléments et donc de contrôler l'erreur induite par la discrétisation spatiale.

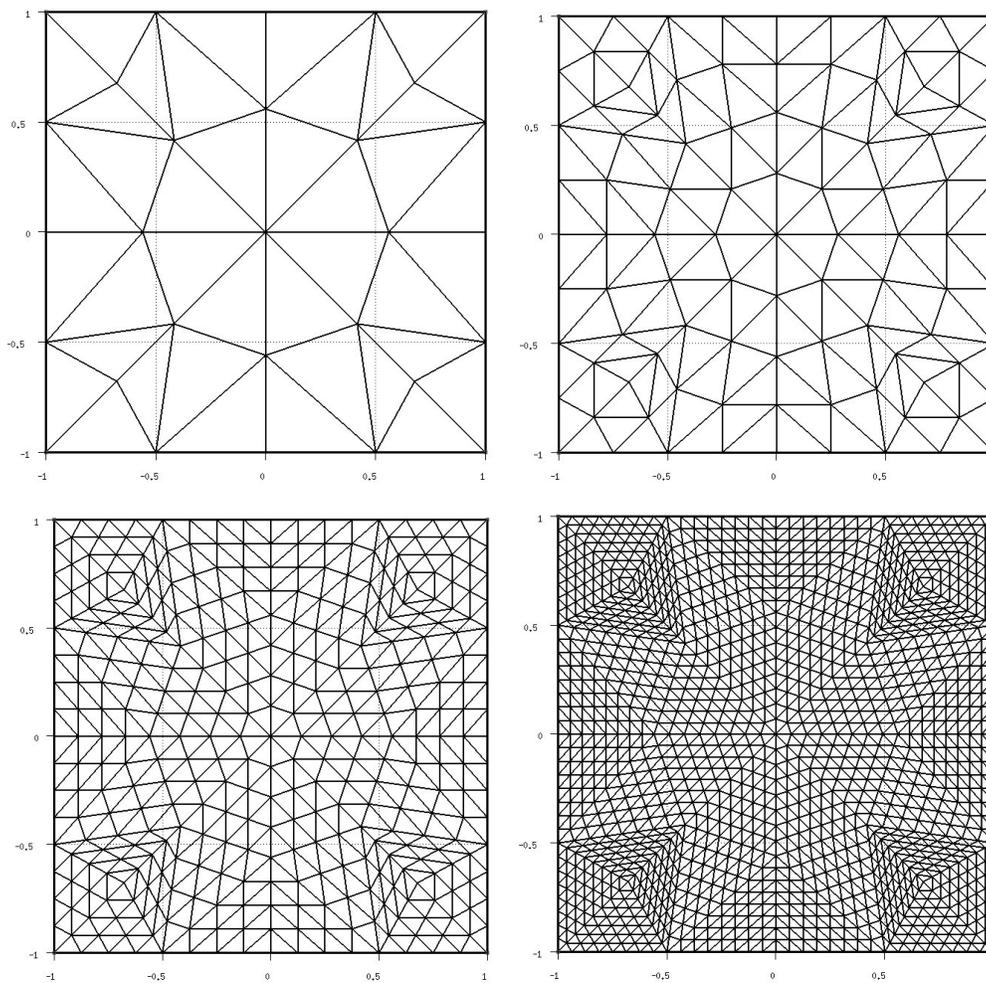


Figure 8.1 Maillages de la géométrie carré composés de 40, 160, 640 et 2560 éléments

Résultats

Dans un premier temps, nous effectuons une simulation à pas constants $k_1/k_2 = 1$. Nous mesurons avec la même norme $\|u(t^n) - u^n\|_{L^2(\Omega_x), L^\infty(\Omega_t)}$ les erreurs sur la variable primaire (la température T) et sur le multiplicateur λ .

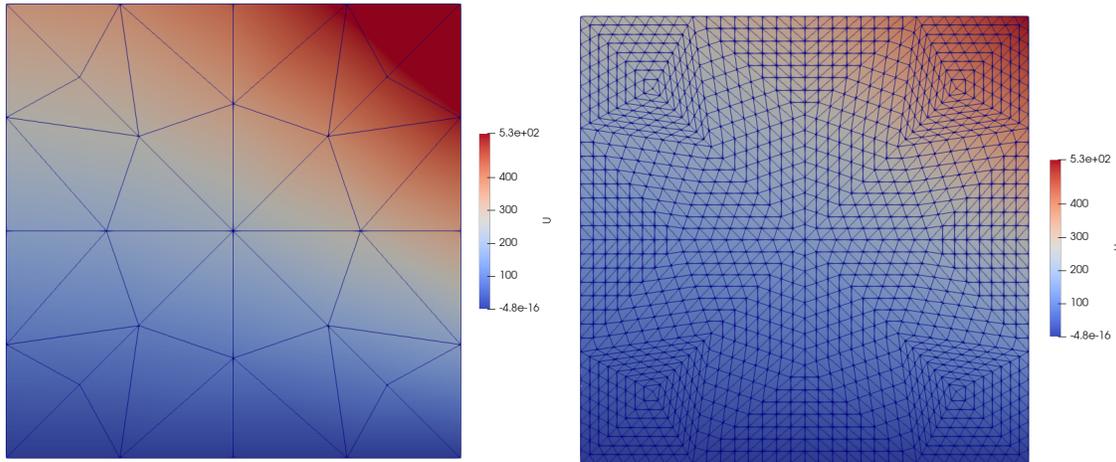


Figure 8.2 Champ de température pour la solution (8.2) dans le cas d'un maillage de 40 et de 2560 éléments

Tableau 8.1 Erreurs et ordres de convergence de la méthode DC/BDF2 à **pas de temps constant** par rapport à la solution (8.2) sur un carré. Mesure de l'erreur sur λ en norme L^2 .

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire T									
Nombre de pas de temps	Nombre d'éléments	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	40	4,8648e+01	—	6,2995e+00	—	1,4774e+00	—	3,3761e-01	—
20	160	1,0603e+01	2,20	7,4340e-01	3,08	9,6760e-02	3,93	1,2334e-02	4,77
40	640	2,5056e+00	2,08	9,1062e-02	3,03	6,1999e-03	3,96	4,0923e-04	4,91
80	2560	6,1160e-01	2,03	1,1304e-02	3,01	3,9306e-04	3,98	1,3159e-05	4,96
Variable secondaire λ									
10	40	7,5189e+00	—	3,5541e+00	—	3,4810e+00	—	3,4840e+00	—
20	160	2,2257e+00	1,76	1,6979e+00	1,07	1,6983e+00	1,04	1,6987e+00	1,04
40	640	1,0436e+00	1,09	9,9030e-01	0,78	9,9055e-01	0,78	9,9058e-01	0,78
80	2560	6,3922e-01	0,71	6,3492e-01	0,64	6,3495e-01	0,64	6,3495e-01	0,64

Tout d'abord, remarquons que l'erreur commise sur la température permet d'atteindre les ordres escomptés. En revanche, les résultats concernant le multiplicateur de Lagrange ne sont pas ceux attendus : les erreurs plafonnent et aucun taux de convergence ne dépasse 1. Nous pouvons expliquer ce comportement en observant l'allure du flux de chaleur au coin de la géométrie, figure 8.3.

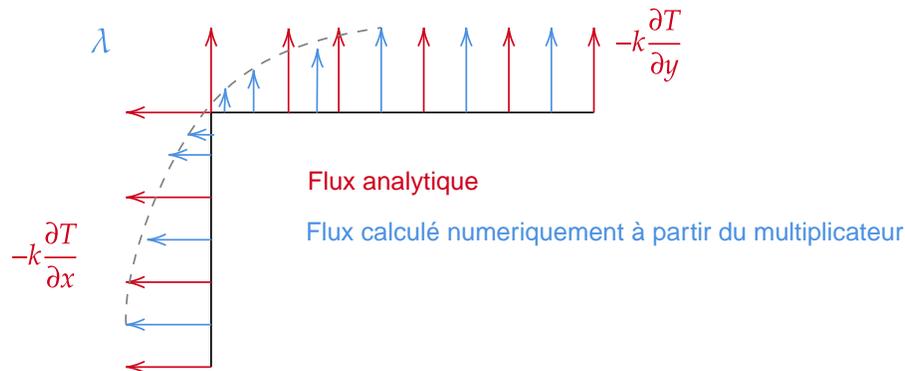


Figure 8.3 Flux numériques et théorique sur le coin d'un carré

Sur la géométrie considérée, la normale est discontinue au coin et le flux de chaleur est donc discontinu par construction. Le choix d'une approximation continue du multiplicateur ne peut donc pas représenter la solution analytique de ce problème. Néanmoins, l'erreur commise se limite à un nombre fini de discontinuité.

8.1.1 Exclusion des angles dans la géométrie

Nous vérifions ces hypothèses en calculant les erreurs des flux en excluant les coins, et en excluant ainsi les problèmes de discontinuités.

Tableau 8.2 Erreurs et ordres de convergence de la méthode DC/BDF2 à **pas de temps constants** par rapport à la solution (8.2) sur un carré. Mesure de l'erreur sur λ en norme L^2 en **excluant les coins**.

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire T									
Nombre de pas de temps	Nombre d'elements	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	180	3,6081e+01	—	5,5099e+00	—	1,5287e+00	—	3,5590e-01	—
20	720	9,2201e+00	1,97	7,4147e-01	2,89	1,1434e-01	3,74	1,4402e-02	4,63
40	2880	2,3431e+00	1,98	9,6132e-02	2,95	7,7011e-03	3,89	4,8658e-04	4,89
Variable secondaire λ									
10	180	7,6991e+00	—	1,4274e+00	—	7,9585e-01	—	7,6566e-01	—
20	720	1,9445e+00	1,99	1,6557e-01	3,11	5,7450e-02	3,79	5,3167e-02	3,85
40	2880	4,9577e-01	1,97	2,0894e-02	2,99	2,0261e-03	4,83	9,5172e-04	5,80

Nous constatons que si la solution est suffisamment régulière, l'approximation continue du multiplicateur converge vers la solution analytique de la solution manufacturée, même en

présence de discontinuité sans effets sur le comportement de la variable primaire.

8.1.2 Interprétation du multiplicateur

En général, pour des applications d'IFS, on s'intéresse plutôt à l'intégrale de la variable secondaire, plutôt qu'à son évaluation ponctuelle : par exemple la résultante des forces de contact sur la frontière d'un objet est l'intégrale du vecteur-contrainte. Dans notre exemple, nous calculons donc l'erreur sur l'intégrale du flux de chaleur.

Tableau 8.3 Erreurs et ordres de convergence de la méthode DC/BDF2 à **pas de temps constants** par rapport à la solution 8.2 sur un carré. Mesure de l'erreur de **l'intégrale de λ** .

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire T									
Nombre de pas de temps	Nombre d'éléments	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	40	4,8648e+01	—	6,2995e+00	—	1,4774e+00	—	3,3761e-01	—
20	160	1,0603e+01	2,20	7,4340e-01	3,08	9,6760e-02	3,93	1,2334e-02	4,77
40	640	2,5056e+00	2,08	9,1062e-02	3,03	6,1999e-03	3,96	4,0923e-04	4,91
80	2560	6,1160e-01	2,03	1,1304e-02	3,01	3,9306e-04	3,98	1,3159e-05	4,96
$\int_{\partial\Omega} \lambda d\Gamma$									
10	40	2,6261e+00	—	3,9583e-01	—	1,1516e-01	—	3,5476e-01	—
20	160	5,9377e-01	2,14	4,4194e-02	3,16	4,9397e-03	4,54	2,0095e-03	7,46
40	640	1,4207e-01	2,06	5,1574e-03	3,10	3,0683e-04	4,01	1,7522e-05	6,84
80	2560	3,4819e-02	2,03	6,2046e-04	3,06	1,9131e-05	4,00	5,5452e-07	4,98

En utilisant maintenant la géométrie complète, c'est-à-dire sans exclure les angles, nous obtenons les taux de convergence attendus sur l'intégrale du multiplicateur sur une surface fermée. Une piste de réflexion consiste à utiliser des approximations discontinues, donc des approximations non conformes de type Crouzeix–Raviart. Nous ne développerons pas cette solution car dans la pratique nous utilisons surtout l'intégrale du multiplicateur, qui prend surtout du sens dans le cas d'une application fluide-structure comme nous l'avons vu dans la section 5.3.

8.2 Écoulements isothermes en 2D

Nous effectuons maintenant des simulations pour un problème d'écoulement isotherme. Nous simulons désormais un phénomène où l'on tient compte du terme d'advection $(u \cdot \nabla)u$, ce qui peut être un facteur d'instabilité. Le problème linéarisé présente des valeurs propres complexes, elles peuvent donc se trouver dans la zone d'instabilité d'un schéma BDF3 par

exemple comme nous l'avons dans la section 3.3 et la figure 3.1. De plus, notre simulation comporte désormais plusieurs champs couplés.

Pour ce problème nous résolvons les équations de Navier-Stokes pour un fluide incompressible non soumis à la gravité dans un repère cartésien. Nous choisissons un domaine temporel $\Omega_t = [0, 1] \in \mathbb{R}$ et spatial $\Omega_x = [-1, 1] \times [-1, 1] \in \mathbb{R}^2$ et nous notons $\Gamma = \partial\Omega_x$:

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \\ \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right), \\ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right), \\ u(x, y, 0) = u_0, \\ v(x, y, 0) = v_0, \\ u(x, y, t) = u_\Gamma(x, y, t) \text{ sur } \Gamma, \\ v(x, y, t) = v_\Gamma(x, y, t) \text{ sur } \Gamma, \\ p(-1, -1, t) = 0 \text{ Pa.} \end{array} \right. \quad (8.4)$$

Nous choisissons comme solution analytique à ce problème :

$$\left\{ \begin{array}{l} u(x, y, t) = -y(1 - x^2)t^6, \\ v(x, y, t) = x(1 - y^2)t^6, \\ p(x, y, t) = 2 + x + y. \end{array} \right. \quad (8.5)$$

Pour cette solution, nous choisissons une masse volumique et une viscosité cinématique égales respectivement à 1 kg m^{-3} et $1 \text{ m}^2 \text{ s}^{-1}$. Nous utilisons des éléments de Taylor-Hood afin de respecter la condition inf-sup discrète [31]. Ces éléments ont des interpolants continus de degré P^m et P^{m-1} , en vitesse et en pression respectivement, et permettent d'assurer la stabilité de la simulation. En outre, nous utilisons la méthode de Newton-Raphson pour résoudre le système non linéaire (8.4) à chaque étape de temps.

8.2.1 Paramètres du solveur

Les résultats qui suivent sont obtenus avec le code d'éléments finis paramétré par :

- Interpolant P^3 pour les vitesses et P^2 pour la pression
- Degré de quadrature de Gausse-Legendre : 15
- Précision du solveur de Newton : 10^{-8}
- Nombre d'itérations maximum du solveur de Newton : 20

- Initialisation de DC/BDF2 avec la solution exacte
- Imposition des conditions limites par la méthode de Verwer, car nous avons vu que cela ne modifiait pas le comportement des DC à pas constant, mais que cela permettait en revanche de parer la perte d'ordre à pas variables.

8.2.2 Simulation à pas de temps constants

Dans un premier temps, nous allons observer comment nos méthodes temporelles se comportent avec des pas de temps constants. Nous pouvons tout d'abord visualiser le champ de vitesse pour vérifier que l'écoulement se comporte bien de la manière attendue.

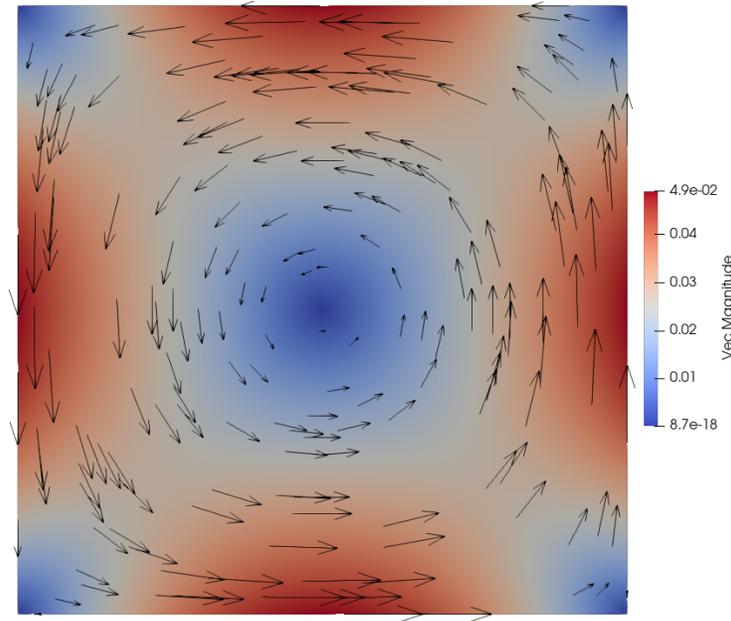


Figure 8.4 Champ de vitesses dans le cas du problème (8.5)

Le vecteur vitesse est nul au centre et tangent aux frontières, les forces normales du fluide sur la paroi sont donc nulles. Notons que nous mesurons cette fois les erreurs sur les champs de vitesse u et v . Notre mesure de la norme est donc vectorielle et s'écrit comme :

$$\mathcal{E} = \|\vec{u} - \vec{u}_h\|_{L^2(\Omega), L^\infty(T^0, T^1)} = \sup_{t \in [0, T]} \sqrt{\int_{\Omega_x} (\vec{u}(x, t) - \vec{u}^{n, h}) \cdot (\vec{u}(x, t) - \vec{u}^{n, h}) dx}. \quad (8.6)$$

Nous présentons ici les résultat pour les schémas DC/BDF1 et DC/BDF2.

Tableau 8.4 Erreurs et ordres de convergence de la méthode DC/BDF2 à **pas de temps constants** pour le problème (8.5) sur un carré. Mesure de l'erreur de **l'intégrale de λ_u** .

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variable primaire \bar{u}									
Nombre de pas de temps	Nombre d'elements	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	40	1,2379e-01	—	1,1115e-02	—	1,6369e-03	—	1,8399e-04	—
20	160	2,8784e-02	2,10	1,3701e-03	3,02	1,0916e-04	3,91	6,6966e-06	4,78
40	640	6,9676e-03	2,05	1,6978e-04	3,01	6,9857e-06	3,97	2,2407e-07	4,90
80	2560	1,7169e-03	2,02	2,1137e-05	3,01	4,4092e-07	3,99	7,2333e-09	4,95
$\int_{\partial\Omega} \lambda d\Gamma$									
10	40	3,5353e-15	—	5,1304e-16	—	2,9426e-16	—	1,8249e-15	—
20	160	1,2281e-15	—	4,3568e-16	—	2,7965e-16	—	5,5623e-16	—
40	640	8,8173e-15	—	1,0975e-15	—	5,4807e-15	—	3,3441e-16	—
80	2560	1,8129e-16	—	6,5211e-16	—	9,6278e-15	—	8,3461e-16	—

De nouveau, les taux de convergence correspondent à nos attentes. Nous n'observons pas de pertes d'ordre car les pas de temps sont constants. L'intégrale du multiplicateur, quant à elle, est nulle étant donné qu'elle correspond aux forces du fluide sur la paroi.

8.2.3 Simulation à pas de temps alternés

Nous réalisons les mêmes simulations avec des pas de temps alternés toujours avec le même ratio : $k_1/k_2 = 4$.

Tableau 8.5 Erreurs et ordres de convergence de la méthode DC/BDF2 à **pas de temps alternés** pour le problème 8.5 sur un carré. Mesure de l'erreur de **l'intégrale de λ_u** .

		BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
Variables primaires \bar{u}									
Nombre de pas de temps	Nombre d'elements	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	40	1,0144e-01	—	1,0758e-02	—	2,0050e-03	—	4,1489e-04	—
20	160	2,6250e-02	1,95	1,4712e-03	2,87	1,5902e-04	3,66	2,7712e-05	3,90
40	640	6,6630e-03	1,98	1,9047e-04	2,95	1,0940e-05	3,86	1,6087e-06	4,11
80	2560	1,6795e-03	1,99	2,4218e-05	2,98	7,1608e-07	3,93	8,8454e-08	4,18
Variable secondaire λ									
10	40	7,5538e-15	—	1,1175e-15	—	1,2799e-16	—	2,0544e-16	—
20	160	1,9479e-15	—	5,3937e-15	—	5,6357e-16	—	1,0022e-15	—
40	640	1,3240e-15	—	2,5859e-16	—	2,6183e-15	—	5,0128e-16	—
80	2560	9,3095e-16	—	1,2552e-15	—	1,2593e-16	—	2,5063e-15	—

À pas alternés, les méthodes fonctionnent également. Les taux de convergences des variables principales sont corrects même si nous remarquons que la méthode DC5/BDF2 met plus de

temps à converger vers le bon ordre. La correction de Verwer permet bien de ne pas subir de perte d'ordre sur les méthodes d'ordre élevés.

8.3 Allées de von Kármán

Pour compléter nos simulations en deux dimensions, nous nous intéressons aux tourbillons de von Kármán. Ce problème est bien connu dans la littérature. Cela va donc nous permettre de vérifier et valider nos méthodes, en comparant nos résultats numériques avec d'autres résultats expérimentaux ou numériques publiés.

8.3.1 Paramètres des simulations

Lors de ces simulations, nous cherchons à obtenir des allées de von Kármán. Ce phénomène tourbillonnaire ne peut être obtenu que dans des conditions particulières que nous allons détailler ici. Les allées de von Kármán peuvent apparaître sur tous types de géométrie peu profilées, le cas classique étant celui du cylindre. Dans notre situation en deux dimensions cela correspond donc à un cercle plongé dans un champ de vitesses, qui modélise un cylindre de longueur infinie. Nous allons prendre un cercle de diamètre 1 m centré en $(0, 0)$ et $\Omega_x = (-8, 28) \times (-8, 8) \in \mathbb{R}^2$ et $\Omega_t = (0, 60) \in \mathbb{R}$. Ce domaine est étendu, car nous ne voulons pas subir l'influence des frontières aux abords du cercle et il faut également permettre aux tourbillons de se développer dans le sillage du cercle. Le domaine de calcul est représenté sur la figure 8.5.

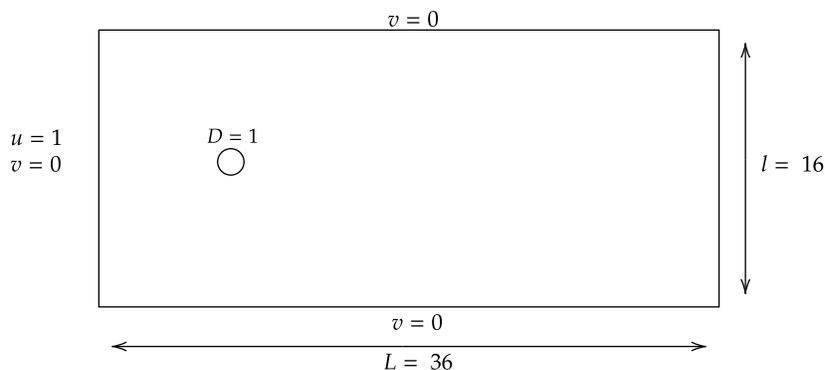


Figure 8.5 Géométrie utilisée pour générer des allées de von Kármán

Les conditions limites sont imposées telles que décrites à la figure 8.5. En particulier nous imposons également le champ de vitesses en entrée, avec une vitesse horizontale de 1 m s^{-1} . Les simulations sont réalisées à un nombre de **Reynolds de 150** [32] pour nous placer dans

la zone d'observation des tourbillons. Nous avons une masse volumique de 1 kg m^{-3} et cela nous impose une viscosité dynamique de $0.0033 \text{ m}^2 \text{ s}^{-1}$.

Pour faciliter l'apparition des tourbillons, nous imposons un écoulement cisailé pendant les dix premières secondes de la simulation, à l'aide du champ de vitesse suivant. Une fois les instabilités amorcées nous imposons $u = 1 \text{ m s}^{-1}$ en entrée. En entrée, nous imposons donc le champ de vitesses suivant :

$$\begin{aligned} u_{in} &= 0.125y - 0.5 + \min\left(\frac{t}{10}, 1\right)(-0.125y + 0.5) + 1 \\ v &= 0 \end{aligned} \quad (8.7)$$

Ce qui concrètement nous donne un champs de vitesse cisailé ayant l'allure suivante :

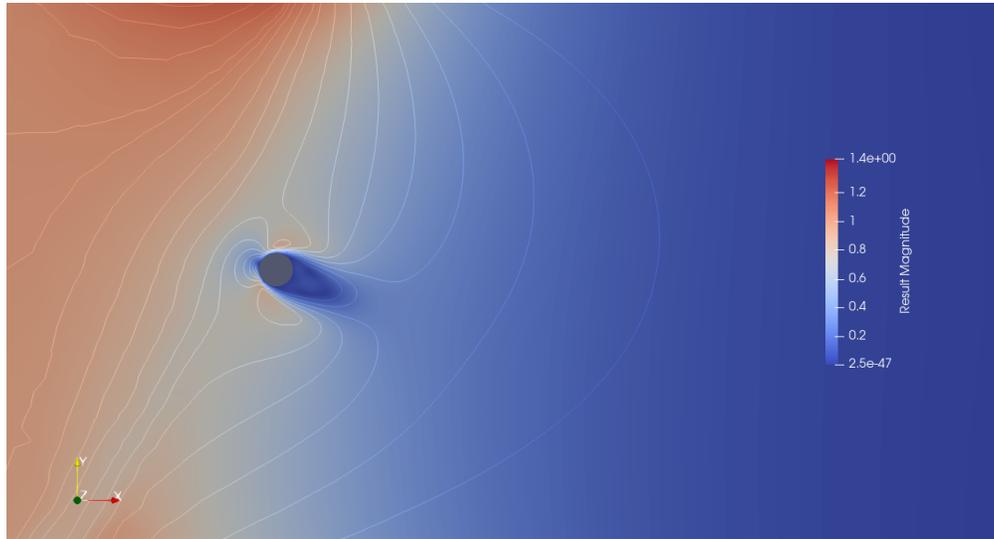


Figure 8.6 Champ de vitesse horizontale imposé initialement afin de provoquer des instabilités

Enfin, il nous faut aborder la question du maillage qui est primordiale pour cette simulation. Une fois que les allées de von Kármán sont amorcées, il faut un maillage suffisamment fin pour les capturer. Nous raffinons donc le maillage en aval du cylindre. De plus nous nous intéressons aux forces sur le cylindre, le maillage doit donc être raffiné dans cette zone (en orange). Nous obtenons un maillage de 93338 éléments sur lesquels nous utilisons des interpolants P^2 pour la vitesse et P^1 pour la pression.

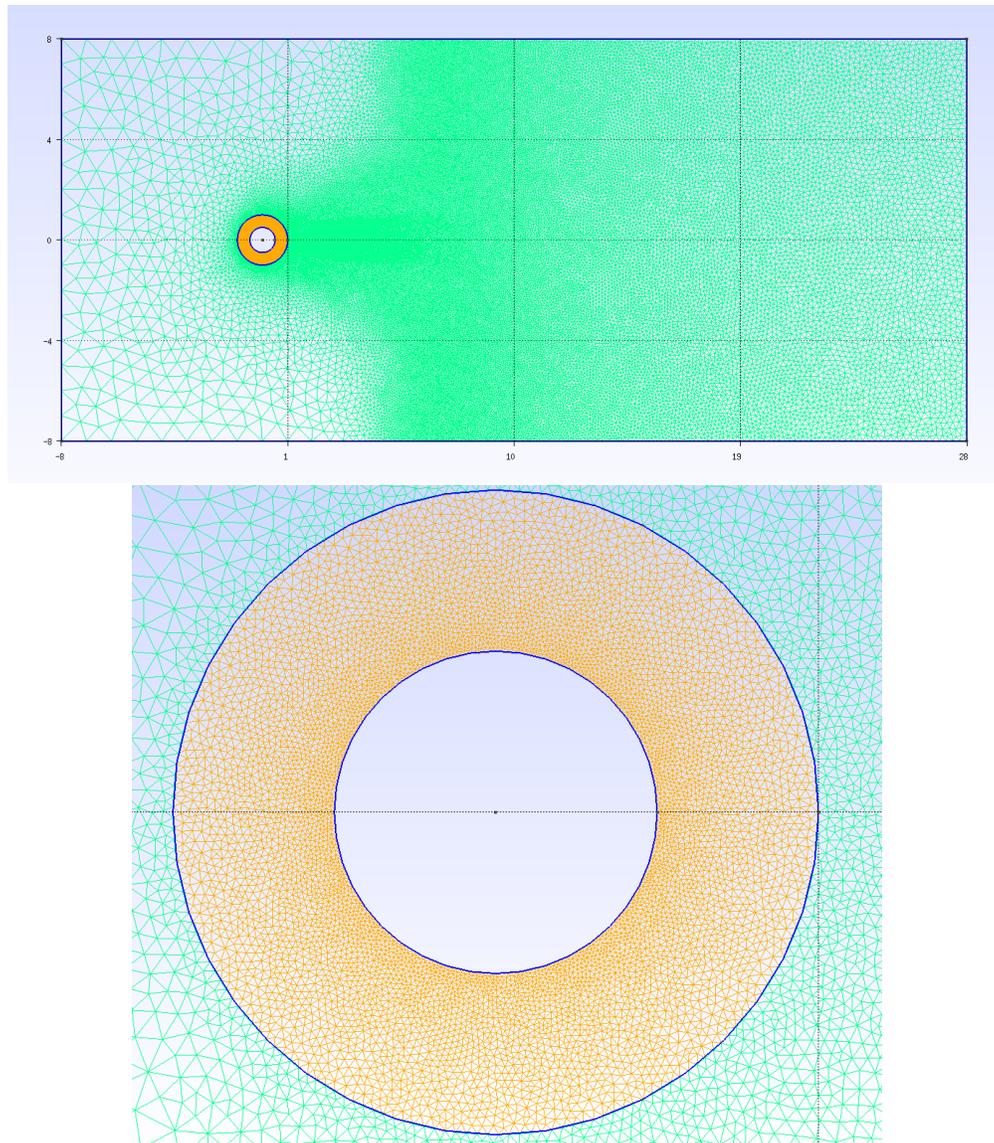


Figure 8.7 Maillage de 93338 éléments utilisé pour capturer les allées de von Kármán

Schéma d'intégration temporelle

Nous allons utiliser le schéma le plus précis dont nous disposons, c'est-à-dire le DC5/BDF2. Pour ce problème, ne connaissant pas la solution exacte de cet écoulement, nous ne pouvons pas utiliser la méthode des solutions manufacturées pour mesurer l'erreur de la solution. Ainsi nous utilisons l'intégrateur DC5/BDF1 pour initialiser les premières valeurs du DC5/BDF2 comme nous l'avons vu dans la section 4.3.

De plus nous imposons faiblement les conditions limites sur le cercle avec la correction de Verwer. Cela nous permet de mesurer avec grande précision les forces exercées sur le cylindre

par l'intégration du multiplicateur de Lagrange, comme nous l'avons évoqué à la section 5.3.

Enfin, précisons les paramètres avec lesquels nous avons réalisé ces simulations :

- Pas de temps 0.225, 0.0625, 0.03125 et 0.015625,
- Interpolant P^2 pour les vitesses et P^1 pour la pression,
- Degré de quadrature de Gausse-Legendre : 15,
- Précision du solveur de Newton : 10^{-8} ,
- Nombre d'itérations maximum du solveur de Newton : 20,
- Initialisation du DC/BDF2 avec le DC/BDF1,
- Imposition des conditions limites par la méthode de Verwer sur le cylindre.

Après que l'imposition de l'écoulement cisailé, d'une durée de 10 secondes, nous observons l'apparition de tourbillons se formant en aval du cylindre. Une fois amorcés, ces lâchés tourbillonnaires sont périodiques. Ces oscillations provoquent sur le cylindre une force de traînée, mais également une force de portance.

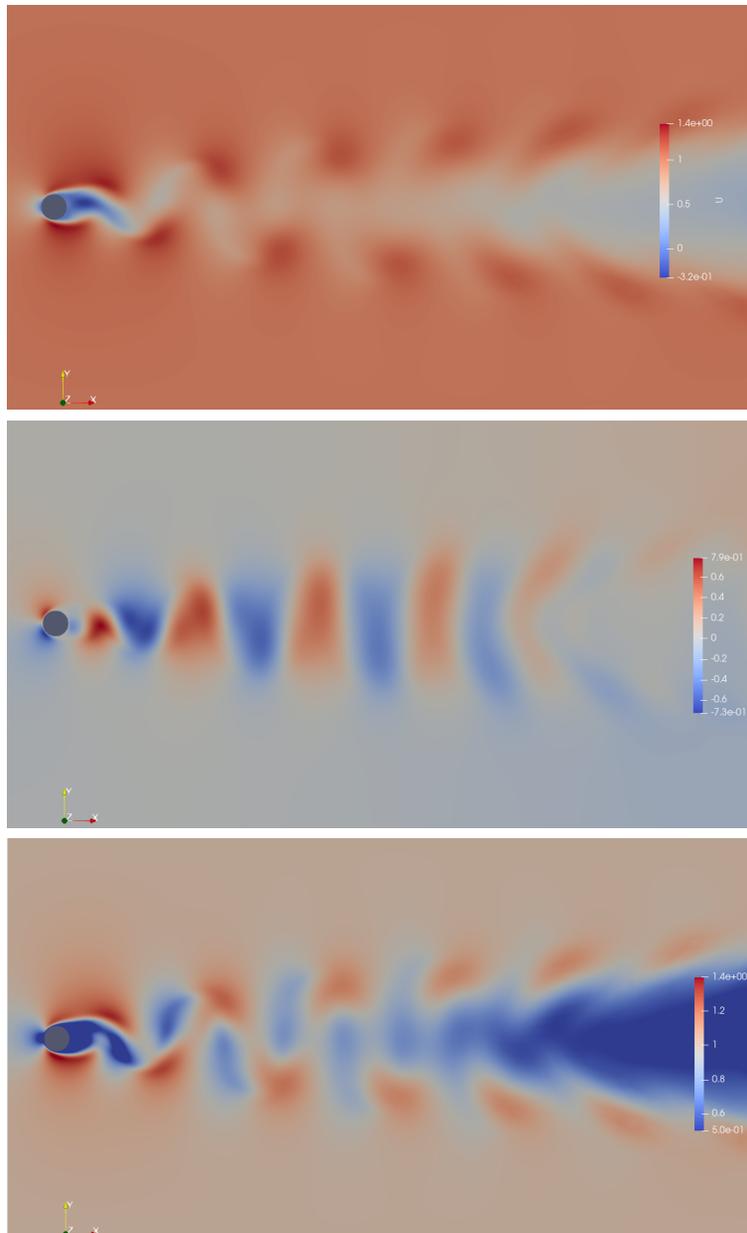


Figure 8.8 Vitesses horizontale u , verticale v et norme du vecteur vitesse $||\vec{u}||$ en aval du cercle pour un lâché de tourbillons de von Kármán..

En se servant du multiplicateur de Lagrange sur le bord du cercle, nous pouvons obtenir les forces de traînée et de portance. L'intégrale du multiplicateur associé aux champs de vitesse horizontale nous permet d'obtenir la force de traînée F_D . La force de portance F_L est, quant à elle, obtenue grâce au multiplicateur associé à v , la vitesse verticale. Afin de pouvoir comparer ces valeurs avec la littérature, nous travaillons avec les coefficients de traînée et

portance qui sont définis respectivement par :

$$C_D(t) = \frac{2F_D}{\rho U^2 D} \quad (8.8)$$

$$C_L(t) = \frac{2F_L}{\rho U^2 D} \quad (8.9)$$

À l'issue de cette simulation, nous disposons de l'évolution des forces de traînée et de portance en fonction du temps.

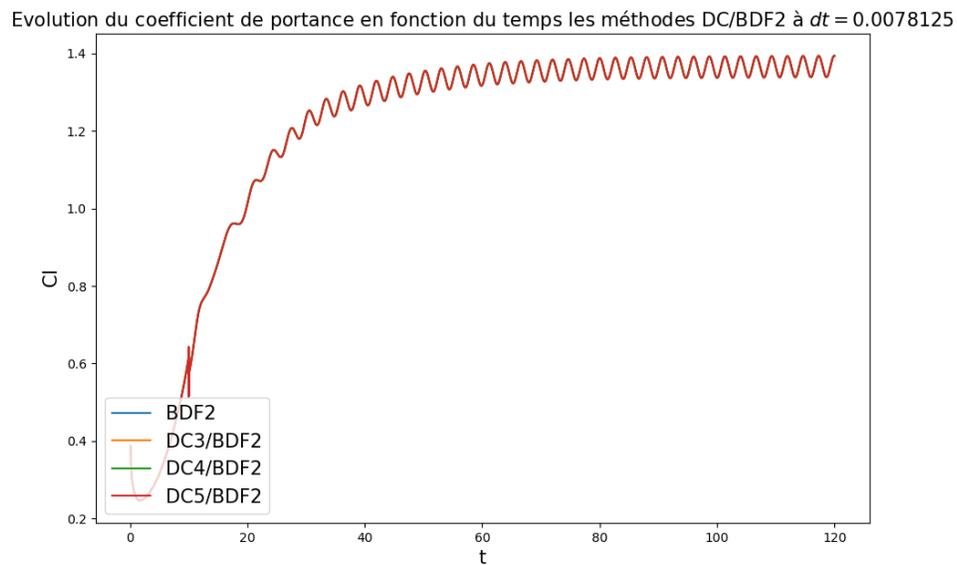


Figure 8.9 Évolution du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. **Simulation avec un pas de temps de 0.03125.**

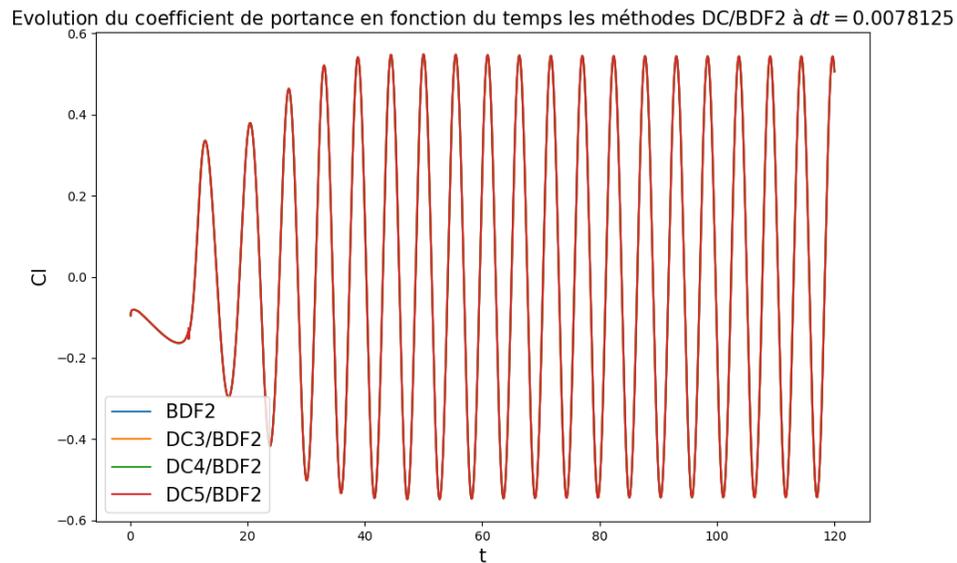


Figure 8.10 Évolution du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. **Simulation avec un pas de temps de 0.03125.**

Nous observons bien la période transitoire, durant laquelle les allées commencent à se former. Ensuite, nous observons que les coefficients de traînée et de portance varient périodiquement.

8.3.2 Étude de raffinement temporelle

Pour vérifier que nos méthodes convergent, nous effectuons plusieurs simulations avec différents pas de temps : 0.225, 0.0625, 0.03125, 0.015625. Nous ne nous intéressons qu'à la période établie, c'est-à-dire pour $t \in [90, 120]$. Nous traçons alors l'évolution du coefficient de portance en fonction du coefficient de traînée, c'est une manière classique de représenter des résultats d'allée de von Kármán.

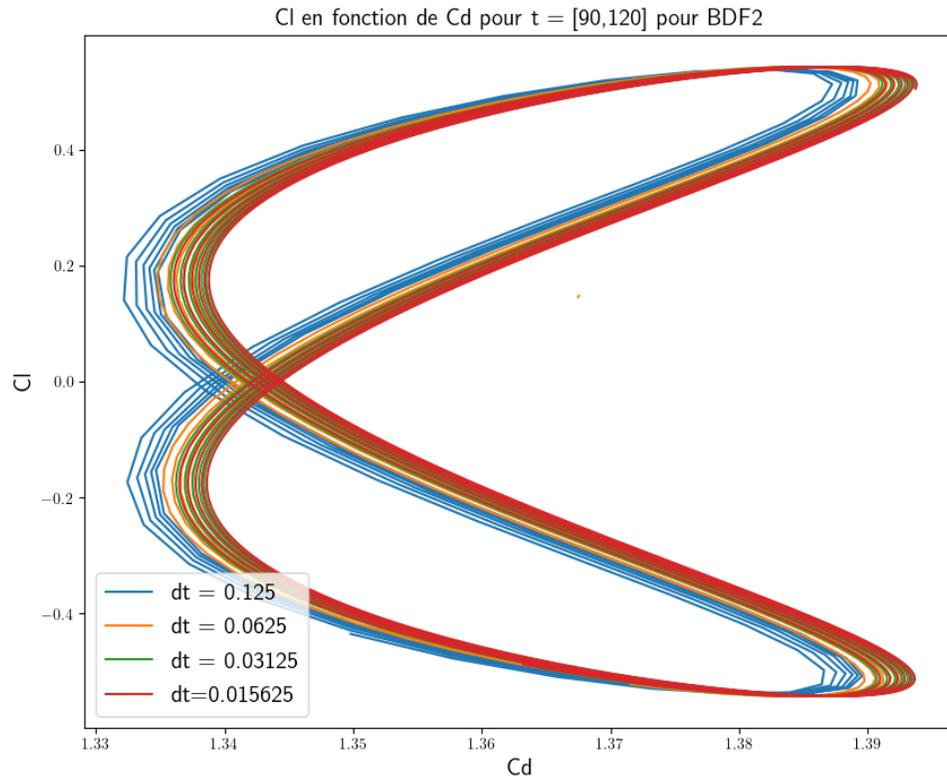


Figure 8.11 Évolution du coefficient de traînée en fonction du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Méthode BDF2 pour plusieurs pas de temps.

Dans le cas du BDF2, nous observons que diminuer le pas de temps a pour effet de faire converger les coefficients vers une même courbe. Nous pouvons donc considérer que la méthode BDF2 converge bien.

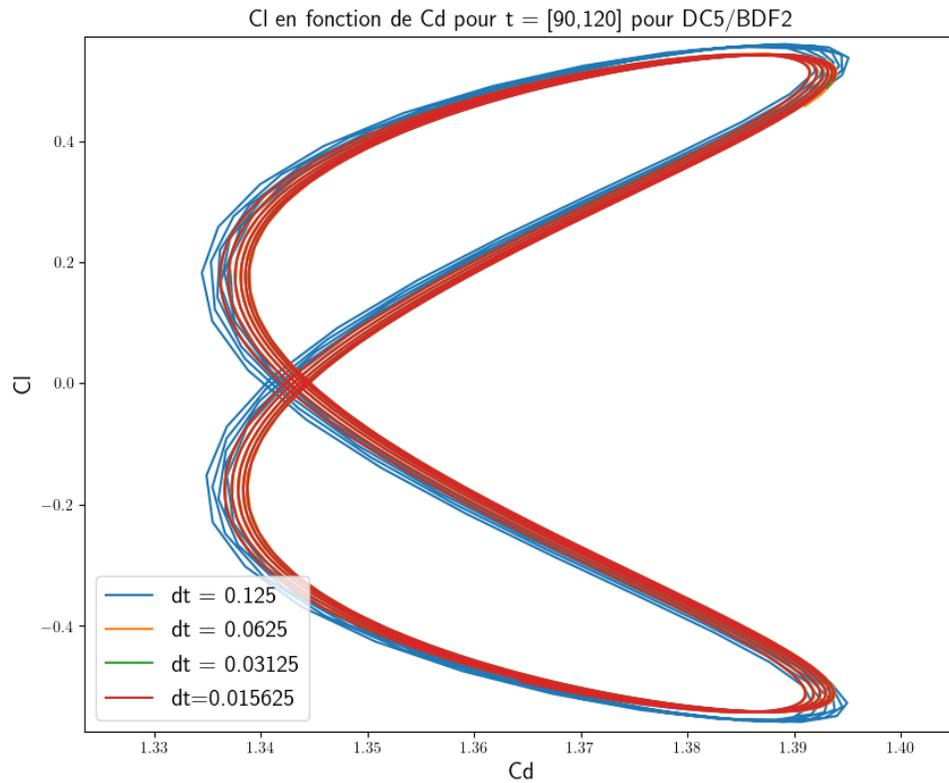


Figure 8.12 Évolution du coefficient de traînée en fonction du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Méthode DC5/BDF2 pour plusieurs pas de temps.

De même pour la méthode DC5/BDF2, les courbes tendent à se superposer lorsque le pas de temps diminue. Ce phénomène, plus rapide que pour le BDF2, nous montre que la méthode DC5/BDF2 converge. Il en va de même pour les méthodes intermédiaires DC4/BDF2 et DC3/BDF2.

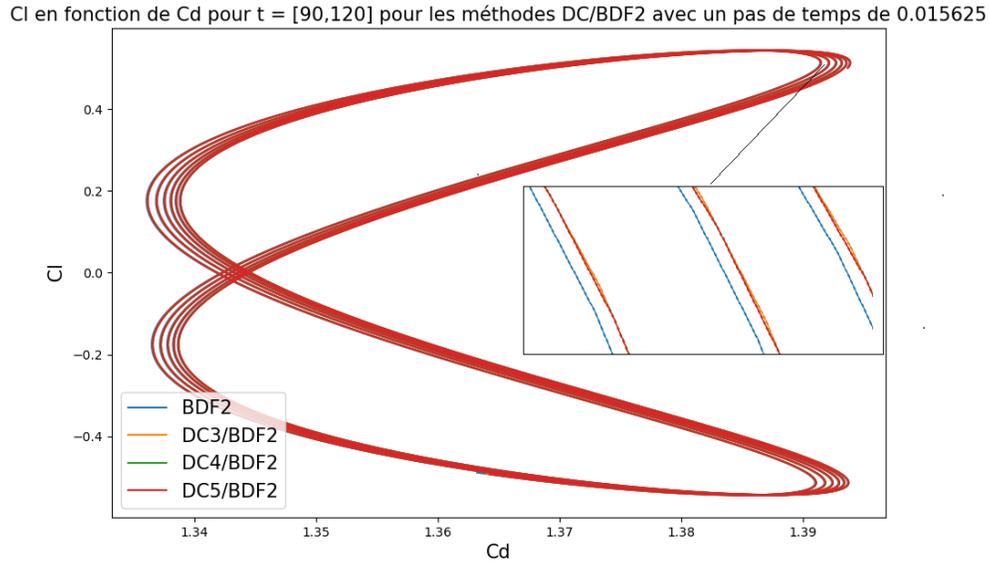


Figure 8.13 Évolution du coefficient de portance pour un lâché de tourbillon de von Kármán d'un cylindre. Simulation pour un pas de temps de 0.03125 pour les différentes méthodes DC/BDF2.

Dans la Figure 8.13, nous observons que les solutions BDF2 à DC5/BDF2 sont superposées à la précision de l'intégration temporelle.

Sachant que nos méthodes convergent, nous tentons donc de faire apparaître les ordres convergences des erreurs propres à chaque méthode. Cependant nous ne disposons pas de solution analytique pour cette situation. L'idée est d'utiliser la solution d'ordre supérieur comme solution de référence. Ainsi nous calculons l'erreur d'approximation sur le multiplicateur dans le cas du BDF2 comme :

$$\mathcal{E}_{BDF2} = \max(|C^{BDF2} - C^{DC3}|)_{\Omega_t}. \quad (8.10)$$

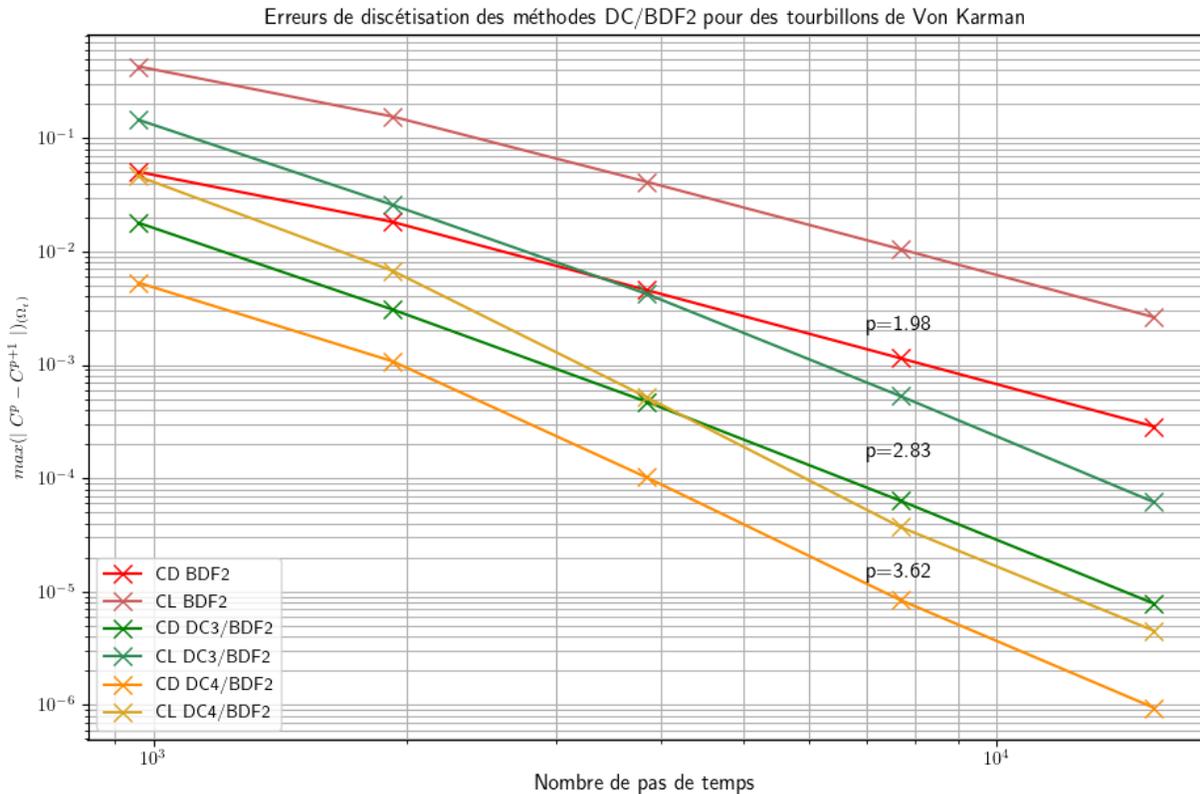


Figure 8.14 Erreurs d'approximation sur le multiplicateur en fonction du nombre de pas de temps pour les méthodes DC/BDF2.

Nous voyons que les erreurs de discrétisation de la méthode converge bien avec un ordre deux. En revanche les méthode DC3/BDF2 et DC4/BDF2 ne présente pas un ordre de convergence conforme à nos attentes. Pour expliquer ces différences, nous pouvons avancer plusieurs hypothèses : le maillage n'est pas raffiné lors du raffinement en pas de temps, une erreur de discrétisation spatiale résiduelle peut donc apparaître. De plus ce maillage, discrétisé par des éléments P^2/P^1 , peut surement être amélioré.

8.3.3 Résultats

Si nous ne nous intéressons qu'à la partie établie, c'est-à-dire une fois que les tourbillons sont bien formés (à partir de $t = 90$), nous pouvons obtenir la valeur moyenne des coefficients : $C_D = 1.3657$ avec une demi-amplitude de variation de $\Delta C_D = 0.028$, $C_L = 0$ avec une demi-amplitude de $\Delta C_L = 0.054$.

Pour mieux appréhender ces valeurs, nous pouvons replacer ces résultats par rapport à ceux obtenus dans la littérature dans la Table 8.6.

Tableau 8.6 Données comparatives pour les allées de von Kármán

Références	$C_D \pm \Delta C_D$	$\pm \Delta C_L$
Lecoite et Piquet [32]	1.29 ± 0.04	± 0.60
Zhang et Dalton [33]	1.25 ± 0.03	± 0.54
Franke et al. [34]	1.31	± 0.65
Nos résultats	1.3657 ± 0.028	± 0.54

Conclusion

Nous avons présenté dans cette section un exemple d'application à un problème d'interaction Fluide-Structure. Les simulations ont été réalisées à l'aide des méthodes DC/BDF2 et DC/BDF1 pour l'initialisation du calcul. Nous avons eu recours à l'imposition faible des conditions limites sur le cylindre pour obtenir un calcul précis des forces exercées sur celui-ci.

L'ensemble de ces résultats préliminaires sont encourageants et semble indiquer que nos prédictions sont similaires à celles publiées dans la littérature.

CHAPITRE 9 CONCLUSION

Nous avons étudié des méthodes d'intégrations temporelles inconditionnellement stables, de degrés élevés – supérieurs à l'ordre 2 –, basées sur des méthodes BDF d'ordre 1 et 2. Nous avons étudié numériquement ces méthodes pour le calcul de systèmes d'équations différentielles ordinaires (EDO) et d'équations aux dérivées partielles (EDP) – à l'aide d'un code d'éléments finis conçu pour leur mise en œuvre. Nous pouvons tirer les conclusions sur l'efficacité de ces méthodes par rapport à leur résultats théoriques, mais aussi établir les limites et les améliorations qui peuvent être apportées.

9.1 Synthèse

Les méthodes d'intégration en temps de type BDF sont très utilisées en éléments finis. En revanche, au-delà de l'ordre 2, ces méthodes sont conditionnellement stables et des stratégies (heuristiques) pour contrôler simultanément l'ordre et le pas de temps doivent être mises en œuvre pour contrer cette perte de stabilité. Il est cependant plus simple de ne contrôler que le pas de temps, si on dispose d'une méthode dont la précision et la stabilité sont connues a priori.

Dans ce contexte, les méthodes à corrections différées ont été proposées pour construire des méthodes *A-stable* d'un ordre arbitraire. Nous avons vu que nous pouvions augmenter l'ordre de convergence du schéma en le corrigeant de manière telle à annuler les premiers termes de l'erreur de troncature. Cependant, cela doit être fait à partir d'une solution calculée indépendamment afin de ne pas compromettre la stabilité. Dans le cadre de ce mémoire, nous avons donc construit deux méthodes à corrections différées qui utilisent soit la solution issue du BDF1 ou celle du BDF2, car elles sont inconditionnellement stables. Ainsi, nous avons théoriquement construit deux classes de schémas jusqu'à l'ordre 5 nommées DC/BDF1 et DC/BDF2.

Nous avons tout d'abord utilisé numériquement ces méthodes à corrections différées pour calculer des solutions d'EDO et de système d'EDO à matrice masse indépendante du temps. Ces essais numériques ont montré que les taux de convergence, numérique et théorique, coïncident pour les essais à pas de temps constants ou à pas de temps alternés, et cela pour une vaste gamme de solutions manufacturées d'équations différentielles et d'équations diffé-

rentielles raides. Mentionnons que dans le cadre de la résolution des équations aux dérivées partielles, les conditions limites essentielles peuvent être modélisées faiblement à l'aide de systèmes d'EDO. Cette nouvelle approche a été utile pour étendre la portée de cette étude lorsque les conditions essentielles changent en fonction du temps.

Nous avons ensuite testé nos méthodes pour le calcul d'équations aux dérivées partielles par la méthode des éléments finis. Pour ce faire, nous avons considéré le calcul de l'équation de la chaleur en une dimension. Dans ce cas, les résultats sont cohérents à pas de temps constants, c'est-à-dire que les taux de convergence numériques coïncident aux taux de convergence théoriques pour toutes les méthodes jusqu'à l'ordre 5.

Toutefois, nous avons remarqué qu'une perte d'ordre intervenait lors d'une simulation si les conditions limites essentielles des variables primaires – condition de Dirichlet – dépendent du temps lors des tests à pas de temps alternés. Pour contrer ce phénomène, nous avons développé une méthode originale pour imposer faiblement les conditions limites au moyen d'un multiplicateur de Lagrange. Ce faisant, nous obtenons également la réaction à l'imposition de la condition essentielle. Nous pouvons donc extraire des informations utiles comme le flux de chaleur ou les forces résultantes sur un solide immergé dans un fluide (modélisé par les équations de Navier-Stokes).

L'étude numérique de la mise en œuvre des méthodes de corrections différées a, par la suite, été étendue aux simulations en deux dimensions. Les tests ont été réalisés à l'aide de solutions manufacturées pour l'équation de la chaleur et les équations de Navier-Stokes pour confirmer nos résultats. Notons ici un niveau de difficulté accru lié à l'approximation du multiplicateur, utilisé précédemment, pour l'imposition faible des conditions limites essentielles dépendantes du temps. De plus, les essais portent essentiellement sur l'intégration temporelle basée sur la méthode DC/BDF2. Ici, la méthode DC/BDF1 n'est utilisée que pour initialiser adéquatement les premières étapes de temps du schéma de base BDF2. Finalement, nous avons présenté des résultats comparatifs obtenus avec le schéma DC/BDF2 avec la simulation des allées de von Kármán provenant de la littérature. Nos simulations sont demeurées stables jusqu'à l'ordre 5, sans avoir à contrôler le pas de temps pour mitiger l'émergence d'instabilités numériques; notons que l'intégration temporelle à l'aide du schéma BDF3 est instable pour les allées de von Kármán.

9.2 Limites de nos travaux

Dans ce mémoire, nous avons construit et étudié des méthodes d'intégration temporelle, à corrections différées, en supposant que les matrices (masse) des systèmes différentiels étaient indépendantes du temps et de la solution. Ces deux hypothèses sont très contraignantes et limitent, pour l'instant, la portée des contributions de ce mémoire à la résolution de problèmes multiphysiques plus complexes, tels que la simulation de l'interaction fluide-structure de corps déformables ou en mouvement dans un fluide.

Pour cette raison, il serait également opportun de revoir l'initialisation des méthodes DC, puisque la mise en œuvre actuelle nécessite la construction de la solution et de sa dérivée pour démarrer le calcul. Cette approche est certainement viable pour la solution des équations aux dérivées ordinaires, mais doit être reconsidérée pour la simulation des équations aux dérivées partielles, comme l'équation de la chaleur et les équations de Navier-Stokes.

Pour contrer la problématique de la perte d'ordre observée lors de l'imposition de conditions limites essentielles dépendantes du temps, celles-ci furent imposées faiblement à l'aide d'un multiplicateur de Lagrange. Cette nouvelle approche substitue à l'expression analytique de la condition essentielle celle de la solution d'un système d'EDO équivalent. Elle est donc une généralisation de l'approche de Verwer aux formulations par éléments finis des équations de conservation.

En théorie, l'espace V du multiplicateur correspond à l'espace dual de la trace de la solution de l'équation aux dérivées partielles, c'est-à-dire $H^1(\Omega)$ pour les EDP elliptiques d'ordre 2. Cet espace V est donc très irrégulier. En première approximation, nous avons choisi de construire une approximation polynômiale et continue du multiplicateur, sachant fort bien que l'approximation du multiplicateur doit être judicieusement choisie pour être compatible avec l'approximation de la variable primaire. Par exemple, pour l'équation de la chaleur, le multiplicateur de Lagrange correspond au flux de chaleur à la surface du domaine de calcul. Ainsi, on peut facilement concevoir une solution analytique de l'équation de la chaleur telle que le flux de chaleur est discontinu aux faces adjacentes d'un domaine de calcul rectangulaire.

Cette approximation continue nous permet d'obtenir des valeurs précises de l'intégrale du flux de chaleur – ou de toutes autres quantités telles que les forces exercées sur corps solide immergé –, de manière équivalente à la méthode dite des réactions. Malheureusement, elle

ne permet pas de mesurer de manière simple et rigoureuse le taux de convergence du multiplicateur à cause de la nature intrinsèquement discontinue de celui-ci. Ainsi, les taux de convergence temporelle du multiplicateur sont parfois difficiles à mesurer et à atteindre après quelques étapes de corrections différées pour atteindre des approximations d'ordre élevées ; en particulier, pour la méthode DC/BDF1.

9.3 Améliorations futures et recommandations

En réponse aux problématiques précédentes, nous proposons les améliorations suivantes :

1. Modifier la formulation des méthodes DC/BDF1 et DC/BDF2 pour la résolution de systèmes différentiels tels que les matrices (masse) sont non linéaires en fonction du temps et de la solution.

Une analyse préliminaire montre qu'une modification judicieuse consisterait à construire une l'approximation de la dérivée temporelle de la variable primaire, sans modifier la construction du résidu du système algébrique – comme dans la mise en œuvre décrite dans ce mémoire.

2. Modifier des méthodes DC pour exclure la construction de la dérivée temporelle de la variable primaire du processus d'initialisation (au temps initial).

Pour s'affranchir du calcul de la dérivée temporelle au temps initial, nous devons modifier la construction du polynôme d'interpolation de Newton des dérivées premières pour ne pas utiliser de solutions antérieures à la solution courante faisant intervenir la solution initiale. Ces modifications complexifieront certainement le processus de calcul et devront être vérifiées pour ne pas compromettre la précision des étapes subséquentes.

3. Étudier et mettre en œuvre des approximations non conformes des équations de la chaleur et des équations de Navier-Stokes, pour faciliter l'imposition faible des conditions essentielles des variables primaires.

Cette approche devrait permettre l'approximation discontinue des multiplicateurs et améliorer la qualité (locale) de l'approximation des flux de chaleur (ou des efforts). Le choix d'une norme appropriée (et sa méthode de construction) pour mesurer la qualité de cette approximation doit être étudiée avec soin avant de poursuivre dans cette voie. Nous avons brièvement exploré la possibilité de travailler avec des éléments non conformes. Nous avons effectué des tests avec des éléments linéaires pour la solution de l'équation de la chaleur qui sont prometteurs.

4. Développer et tester un algorithme pour adapter le pas de temps pour le pour contrôler l'erreur d'intégration temporelle des méthodes DC.

Dans le cadre de ce mémoire, les méthodes DC ont passé avec succès le test des pas de temps alternés. Rappelons que ce test a été élaboré par Gear pour étudier la résilience des méthodes d'intégration à la variation de pas de temps successifs. Des méthodes ne réussissant pas ce test ne peuvent être retenues dans le contexte de l'adaptation du pas de temps.

RÉFÉRENCES

- [1] P. Birken, K. J. Quint, S. Hartmann et A. Meister, “A time-adaptive fluid-structure interaction method for thermal coupling,” *Computing and visualization in science*, vol. 13, n^o. 7, p. 331–340, 2010.
- [2] M. Mayr, W. Wall et M. Gee, “Adaptive time stepping for fluid-structure interaction solvers,” *Finite Elements in Analysis and Design*, vol. 141, p. 55–69, 2018. [En ligne]. Disponible : <https://www.sciencedirect.com/science/article/pii/S0168874X17305000>
- [3] V. John et J. Rang, “Adaptive time step control for the incompressible navier–stokes equations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 199, n^o. 9-12, p. 514–524, 2010.
- [4] L. R. Petzold, “Description of dassl : a differential/algebraic system solver,” Sandia National Labs., Livermore, CA (USA), Rapport technique, 1982.
- [5] C. W. Gear et L. R. Petzold, “Differential/algebraic systems and matrix pencils,” dans *Matrix Pencils*. Springer, 1983, p. 75–89.
- [6] S. Skelboe, “The control of order and steplength for backward differentiation methods,” *BIT Numerical Mathematics*, vol. 17, n^o. 1, p. 91–107, 1977.
- [7] L. Fox, “Some improvements in the use of relaxation methods for the solution of ordinary and partial differential equations,” *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, vol. 190, n^o. 1020, p. 31–59, 1947.
- [8] L. Fox et E. Goodwin, “Some new methods for the numerical integration of ordinary differential equations,” dans *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 45, n^o. 3. Cambridge University Press, 1949, p. 373–388.
- [9] L. Fox, *Numerical solution of ordinary and partial differential equations : based on a summer school held in Oxford, August-September 1961*. Elsevier, 2014.
- [10] V. Pereyra, *The difference correction method for non-linear two-point boundary value problems*. Stanford University, 1965.
- [11] W. B. Gragg et H. J. Stetter, “Generalized multistep predictor-corrector methods,” *Journal of the ACM (JACM)*, vol. 11, n^o. 2, p. 188–209, 1964.
- [12] A. Dutt, L. Greengard et V. Rokhlin, “Spectral deferred correction methods for ordinary differential equations,” *BIT Numerical Mathematics*, vol. 40, n^o. 2, p. 241–266, 2000.

- [13] M. L. Minion, “Semi-implicit spectral deferred correction methods for ordinary differential equations,” *Communications in Mathematical Sciences*, vol. 1, n^o. 3, p. 471–500, 2003.
- [14] R. Speck, D. Ruprecht, M. Emmett, M. Minion, M. Bolten et R. Krause, “A multi-level spectral deferred correction method,” *BIT Numerical Mathematics*, vol. 55, n^o. 3, p. 843–867, 2015.
- [15] W. Kress et B. Gustafsson, “Deferred correction methods for initial boundary value problems,” *Journal of scientific computing*, vol. 17, n^o. 1, p. 241–251, 2002.
- [16] B. Gustafsson et W. Kress, “Deferred correction methods for initial value problems,” *BIT Numerical Mathematics*, vol. 41, n^o. 5, p. 986–995, 2001.
- [17] S. Geng, “The deferred correction procedure for linear multistep formulas,” *Journal of Computational Mathematics*, p. 41–49, 1985.
- [18] J.-L. Guermond et P. Minev, “High-order time stepping for the incompressible navier–stokes equations,” *SIAM Journal on Scientific Computing*, vol. 37, n^o. 6, p. A2656–A2681, 2015.
- [19] A. Fortin, *Analyse numérique pour ingénieurs*. Presses inter Polytechnique, 2008.
- [20] G. Wanner et E. Hairer, *Solving ordinary differential equations II*. Springer Berlin Heidelberg, 1996, vol. 375.
- [21] A. Hay, S. Etienne, D. Pelletier et A. Garon, “hp-adaptive time integration based on the bdf for viscous flows,” *Journal of Computational Physics*, vol. 291, p. 151–176, 2015.
- [22] C. W. Gear, “The numerical integration of ordinary differential equations of various orders.” [En ligne]. Disponible : <https://www.osti.gov/biblio/4534813>
- [23] Y. Bourgault et A. Garon, “Variable-step deferred correction methods based on backward differentiation formulae for ordinary differential equations,” *BIT Numerical Mathematics*, p. 1–34, 2022.
- [24] G. Dahlquist, “Convergence and stability in the numerical integration of ordinary differential equations,” *Mathematica Scandinavica*, p. 33–53, 1956.
- [25] M. Crouzeix et A. L. Mignot, *Analyse numérique des équations différentielles*. Masson, 1984, vol. 1.
- [26] W. Hundsdorfer et J. G. Verwer, *Numerical solution of time-dependent advection-diffusion-reaction equations*. Springer Science & Business Media, 2007, vol. 33.
- [27] P. J. Roache, “Code verification by the method of manufactured solutions,” *J. Fluids Eng.*, vol. 124, n^o. 1, p. 4–10, 2002.

- [28] C. F. Curtiss et J. O. Hirschfelder, “Integration of stiff equations,” *Proceedings of the National Academy of Sciences*, vol. 38, n°. 3, p. 235–243, 1952.
- [29] A. Ern et J.-L. Guermond, *Eléments finis : théorie, applications, mise en oeuvre*. Springer Science & Business Media, 2002, vol. 36.
- [30] P. G. Ciarlet, *The finite element method for elliptic problems*. SIAM, 2002.
- [31] A. Fortin et A. Garon, “Les éléments finis : de la théorie à la pratique,” *Université Laval*, 2011.
- [32] Y. Lecointe et J. Piquet, “Flow structure in the wake of an oscillating cylinder,” 1989.
- [33] J. Zhang et C. Dalton, “Interaction of a steady approach flow and a circular cylinder undergoing forced oscillation,” 1997.
- [34] R. Franke, W. Rodi et B. Schönung, “Numerical calculation of laminar vortex-shedding flow past cylinders,” *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 35, p. 237–257, 1990.

ANNEXE A RÉSULTATS COMPLÉMENTAIRES POUR DES SYSTÈMES MATRICIELS EN OD À PAS ALTERNÉS

Nous présentons ici les résultats sur le problème du ressort simulé sous forme de deux équations couplées. Ces résultats viennent compléter les tableaux 6.14 et 6.15 en utilisant cette fois un pas de temps alterné.

Tableau A.1 Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système ressort pour une simulation à **pas de temps alternés**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	9,9090e-03	—	1,2124e-03	—	2,7970e-04	—	4,1707e-05	—	7,0096e-06	—
20	5,8913e-03	0,75	4,0646e-04	1,58	4,5991e-05	2,60	3,8233e-06	3,45	4,5475e-07	3,95
40	3,3360e-03	0,82	1,1610e-04	1,81	6,6901e-06	2,78	2,9166e-07	3,71	2,0177e-08	4,49
80	1,7538e-03	0,93	3,0913e-05	1,91	9,1145e-07	2,88	2,0287e-08	3,85	7,4229e-10	4,76
160	9,0001e-04	0,96	7,9691e-06	1,96	1,1863e-07	2,94	1,3375e-09	3,92	2,5199e-11	4,88

Tableau A.2 Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système ressort pour une simulation à **pas de temps alternés**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	1,2470e-03	—	7,9075e-05	—	1,6659e-05	—	1,3402e-06	—
20	4,9677e-04	1,33	1,2907e-05	2,62	1,4790e-06	3,49	4,7397e-08	4,82
40	1,5391e-04	1,69	1,7982e-06	2,84	1,0889e-07	3,76	1,5367e-09	4,95
80	4,2572e-05	1,85	2,3518e-07	2,93	7,3963e-09	3,88	4,8425e-11	4,99
160	1,1182e-05	1,93	3,0028e-08	2,97	4,8171e-10	3,94	1,5232e-12	4,99

Les taux de convergence sont bien atteints lorsque les pas de temps sont alternés.

ANNEXE B RÉSULTATS COMPLÉMENTAIRES SUR UN SYSTÈME DE TROIS ÉQUATIONS

Nous avons effectué des tests sur de EDO et un système d'EDO, dans toutes ces situations les schémas DC/BDF permettent d'atteindre les bons taux de convergence. Nous avons également simulé un problème très raide pour observer le comportement de nos méthodes. Après un raffinement du pas de temps beaucoup plus important, les taux de convergence sont également atteints. Pour compléter cette étude nous avons effectuer des tests sur un système de trois EDO couplées.

Considérons maintenant le problème constitué de 3 équations à trois inconnues.

$$\frac{dX}{dt} + AX = 0 \quad \text{ou} \quad A = \begin{pmatrix} 1 & -1 & -1 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \quad (\text{B.1})$$

Avec comme conditions initiales :

$$X(t = 0) = \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix}$$

Et la solutions exacte est la suivante :

$$X(t) = e^{-t} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \cos(t) \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + \sin(t) \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} \quad (\text{B.2})$$

Tableau B.1 Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système matriciel à **pas constants**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	4,8457e-02	—	2,2477e-03	—	1,0293e-04	—	9,4505e-06	—	3,2157e-07	—
20	2,4820e-02	0,97	5,4395e-04	2,05	1,5419e-05	2,74	6,3744e-07	3,89	6,1718e-09	5,70
40	1,2546e-02	0,98	1,3308e-04	2,03	2,0924e-06	2,88	4,0800e-08	3,97	1,7642e-10	5,13
80	6,3056e-03	0,99	3,2867e-05	2,02	2,7203e-07	2,94	2,5713e-09	3,99	6,1029e-12	4,85
160	3,1607e-03	1,00	8,1637e-06	2,01	3,4666e-08	2,97	1,6123e-10	4,00	1,9962e-13	4,93

Tableau B.2 Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système matriciel à **pas constants**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	8,3102e-04	—	9,9719e-05	—	1,8832e-06	—	2,2221e-07	—
20	3,0869e-04	1,43	1,5628e-05	2,67	1,5036e-07	3,65	8,5885e-09	4,69
40	9,0665e-05	1,77	2,1266e-06	2,88	1,0765e-08	3,80	2,9350e-10	4,87
80	2,4404e-05	1,89	2,7566e-07	2,95	7,2063e-10	3,90	9,5344e-12	4,94
160	6,3206e-06	1,95	3,5039e-08	2,98	4,6594e-11	3,95	3,0131e-13	4,98

Tableau B.3 Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système matriciel pour une simulation à **pas de temps alternés**.

Nombre de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	5,3435e-02	—	2,7682e-03	—	3,3441e-04	—	5,3968e-05	—	1,6145e-06	—
20	3,0497e-02	0,81	8,2096e-04	1,75	5,0924e-05	2,72	4,1005e-06	3,72	9,4662e-08	4,09
40	1,6230e-02	0,91	2,2174e-04	1,89	6,9827e-06	2,87	2,7961e-07	3,87	3,7368e-09	4,66
80	8,3650e-03	0,96	5,7512e-05	1,95	9,1383e-07	2,93	1,8202e-08	3,94	1,3015e-10	4,84
160	4,2456e-03	0,98	1,4638e-05	1,97	1,1676e-07	2,97	1,1602e-09	3,97	4,2915e-12	4,92

Tableau B.4 Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système matriciel pour une simulation à **pas de temps alternés**.

Nombre de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	7,3819e-04	—	6,0222e-05	—	2,3208e-06	—	1,0288e-07	—
20	3,0004e-04	1,30	9,9040e-06	2,60	1,9828e-07	3,55	3,9723e-09	4,69
40	8,9386e-05	1,75	1,3594e-06	2,87	1,4269e-08	3,80	1,3394e-10	4,89
80	2,4227e-05	1,88	1,7647e-07	2,95	9,5331e-10	3,90	4,3234e-12	4,95
160	6,2971e-06	1,94	2,2433e-08	2,98	6,1590e-11	3,95	1,7986e-13	4,59

Dans cet exemple, on voit clairement apparaître les ordres de convergences de différentes méthodes. Cela vient vérifier la convergence de nos méthodes DC/BDF dans le cas d'EDO. De même les simulations à pas variables convergent également aux taux optimaux.

Reprenons le problème de trois EDO écrit sous forme matriciel. Cette fois nous prenons un coefficient de raideur de 100 contre 1 précédemment. Pour cela nous faisons apparaître la

pulsation $\omega = 100$.

$$\frac{dX}{dt} + AX = 0 \quad \text{ou} \quad A = \begin{pmatrix} 1 & -1 & -1 \\ 0 & 0 & -\omega \\ 0 & \omega & 0 \end{pmatrix} \quad (\text{B.3})$$

Avec comme conditions initiales :

$$X(t = 0) = \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix}$$

Et la solutions exacte est la suivante :

$$X(t) = e^{-t} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + \cos(\omega t) \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} + \sin(\omega t) \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} \quad (\text{B.4})$$

Tableau B.5 Erreur et ordre de convergence de la méthode DC/BDF1 pour la solution manufacturée du système de 3 EDO pour une simulation à **pas de temps constants** avec un **coefficient de raideur de 100**.

Nb de pas de temps	BDF1		DC2/BDF1		DC3/BDF1		DC4/BDF1		DC5/BDF1	
	\mathcal{E}	p								
10	1,4963e+00	0,00	1,4517e+00	0,00	1,4071e+00	0,00	1,4088e+00	0,00	1,3420e+00	0,00
20	1,4044e+00	0,09	1,4134e+00	0,04	1,4134e+00	-0,01	1,6268e+00	-0,21	2,3319e+00	-0,80
40	1,4090e+00	0,00	1,5220e+00	-0,11	1,7098e+00	-0,27	1,8696e+00	-0,20	1,6461e+00	0,50
80	1,4163e+00	-0,01	1,4854e+00	0,04	1,4491e+00	0,24	1,5064e+00	0,31	1,4666e+00	0,17
160	1,4152e+00	0,00	1,4142e+00	0,07	1,4142e+00	0,04	1,4142e+00	0,09	1,4142e+00	0,05
320	1,4146e+00	0,00	1,4142e+00	0,00	1,4144e+00	0,00	1,4145e+00	0,00	1,4144e+00	0,00
640	1,4134e+00	0,00	1,4077e+00	0,01	1,3896e+00	0,03	1,3407e+00	0,08	1,2446e+00	0,18
1280	1,3834e+00	0,03	1,2658e+00	0,15	1,0406e+00	0,42	7,5305e-01	0,83	4,7810e-01	1,38
2560	1,2064e+00	0,20	8,0752e-01	0,65	4,2539e-01	1,29	1,8156e-01	2,05	6,4655e-02	2,89
5120	8,7198e-01	0,47	3,5220e-01	1,20	1,0312e-01	2,04	2,3508e-02	2,95	4,3794e-03	3,88
10240	5,3857e-01	0,70	1,1884e-01	1,57	1,8232e-02	2,50	2,1391e-03	3,46	2,0325e-04	4,43
20480	3,0141e-01	0,84	3,4692e-02	1,78	2,7202e-03	2,74	1,6175e-04	3,73	7,7561e-06	4,71

Ici encore nos méthodes fonctionnent. La méthode DC/BDF1 met du temps à s'établir tandis que le DC/BDF2 semble d'abord fonctionner rapidement avant d'atteindre un palier. Mais après avoir finalement pu capturer les rapides variations le schéma DC/BDF2 fonctionne également.

Tableau B.6 Erreur et ordre de convergence de la méthode DC/BDF2 pour la solution manufacturée du système de 3 EDO pour une simulation à **pas de temps constants** avec un **coefficient de raideur de 100**.

Nb de pas de temps	BDF2		DC3/BDF2		DC4/BDF2		DC5/BDF2	
	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p	\mathcal{E}	p
10	1,4038e+00	0,00	1,3033e+00	0,00	3,0231e+00	0,00	6,9060e+00	0,00
20	1,7854e+00	-0,35	2,4936e+00	-0,94	3,1464e+00	-0,06	3,6823e+00	0,91
40	1,6543e+00	0,11	2,1910e+00	0,19	3,3997e+00	-0,11	5,1654e+00	-0,49
80	1,8619e+00	-0,17	2,2688e+00	-0,05	3,4337e+00	-0,01	5,0367e+00	0,04
160	1,9356e+00	-0,06	2,9103e+00	-0,36	4,8249e+00	-0,49	8,3283e+00	-0,73
320	2,1641e+00	-0,16	3,1421e+00	-0,11	3,1422e+00	0,62	2,2002e+00	1,92
640	1,0379e+00	1,06	4,3220e-01	2,86	1,1526e-01	4,77	2,5320e-02	6,44
1280	2,8268e-01	1,88	3,2892e-02	3,72	2,6879e-03	5,42	1,6166e-04	7,29
2560	7,1514e-02	1,98	2,4399e-03	3,75	6,3648e-05	5,40	1,4245e-06	6,83
5120	1,7920e-02	2,00	1,9616e-04	3,64	2,0025e-06	4,99	2,2180e-08	6,01
10240	4,4828e-03	2,00	1,7658e-05	3,47	8,5416e-08	4,55	5,0979e-10	5,44
20480	1,1209e-03	2,00	1,7773e-06	3,31	4,4458e-09	4,26	1,3894e-11	5,20
40960	2,8026e-04	2,00	1,9527e-07	3,19	2,5547e-10	4,12	3,6282e-13	5,26