



Université Gustave Eiffel  
**Laboratoire MAST-GPEM**

**Stage de Master II**  
**Mathématiques et applications**

Modélisations Analyses Numériques et Calculs Scientifiques (MACS)  
03 Avril - 29 Septembre 2023

---

*Application du Gradient Projecté Spectral à la résolution du modèle d'écoulement viscoplastique de Bingham*

---

Présenté et soutenu publiquement par :

**BOGNON Akouègnon Ghislain**

Sous la direction de

**Nicolas ROQUET**, Chargé de recherche dans le Laboratoire MAST-GPEM de l'Université Gustave Eiffel.

---

# DÉDICACES

---

À mes parents, mes frères, mes sœurs et à Norbert BAMBELOU;

Je dédie ce mémoire de Master II.

---

# REMERCIEMENTS

---

Je tiens tout d'abord à remercier Dieu Elchadaii le tout puissant pour le privilège de grâce qu'il nous a accordé en nous donnant la santé, la détermination, la force et l'intelligence.

Je remercie sincèrement du fond de cœur, toute personne de près ou de loin qui a apportée sa pierre à l'édifice pour la réussite de ce travail.

Je porte un remerciement spécial à l'égard de notre encadrant **Nicolas Roquet** qui a fait preuve d'une disponibilité agréable et dont les conseils nous ont aidés à réaliser ce travail.

Je ne vais jamais arrêter d'exprimer ma profonde gratitude à tous nos Professeurs et en particulier à ma responsable de formation Anaïs Crestetto pour la qualité des cours qu'ils ont eu à me donner ainsi que leurs conseils respectifs durant mon parcourt de MACS.

# Table des matières

DÉDICACES	i
REMERCIEMENTS	ii
INTRODUCTION	v
<b>1 MODÈLE PHYSIQUE ET FORMULATION VARIATIONNELLE</b>	<b>2</b>
1.1 Modèles de Bingham équivalents . . . . .	2
1.2 Formulation variationnelle . . . . .	7
1.3 Existence et unicité . . . . .	9
<b>2 GRADIENT PROJETÉ (GP) AVEC LA RÈGLE DE BARZILAI-BORWEIN ET GRIppo-LAMPORIELLO-LUCIDI</b>	<b>12</b>
2.1 Formulation - Minimisation par gradient projeté . . . . .	12
2.2 Calcul automatique d'un pas de descente variable . . . . .	13
2.2.1 Généralité sur Barzilai et Borwein (BB) . . . . .	13
2.2.2 Formulation Algorithmique . . . . .	17
<b>3 EXPÉRIMENTATIONS NUMÉRIQUES</b>	<b>20</b>
3.1 Approximation spatiale . . . . .	20
3.2 Algorithme en dimension finie . . . . .	22
3.3 Résultats numériques . . . . .	23
<b>4 CONCLUSION ET PERSPECTIVES</b>	<b>43</b>
.1 Annexe . . . . .	43
.2 Algorithme du gradient projeté localement pondéré par pénalisation avec BB et GLL (LPWPG) . . . . .	43
.2.1 Algorithme du gradient projeté localement pondéré (LWPG) . . . . .	43

## NOTATIONS GENERALES

$\Omega \subset \mathbb{R}^d$ , ouvert  $d \geq 2$

$$L^2(\Omega) = \left\{ u \text{ measurable sur } \Omega \text{ et } \int_{\Omega} |u|^2 dx < \infty \right\}$$

$$H_0^1(\Omega) = \left\{ u \in L^2(\Omega); \quad u|_{\partial(\Omega)} = 0 \text{ et } \partial_{x_i} u \in L^2(\Omega) \right\}$$

$\tau : \tau' = \sum_{i,j} \tau_{ij} \tau'_{ij}$  pour lequel on a la norme  $|||\tau||| = \sqrt{\tau : \tau'}$ ,  $\tau$  et  $\tau'$  dans  $\mathbb{R}^{d \times d}$

$$K = \left\{ \mu \in L^2(\Omega)^{d \times d}, |||\mu||| \leq 1 \text{ p.p sur } \Omega \right\}$$

$$V = \left\{ w \in H_0^1(\Omega)^d, \quad \operatorname{div} w = 0 \right\}$$

$$L_0^2(\Omega) = \left\{ u \in L^2(\Omega)^d, \quad \int_{\Omega} u = 0 \right\}$$

$|.|_{L^2}$  la norme  $L^2(\Omega)$  ainsi que pour les espaces  $L^2(\Omega)^d$  et  $L^2(\Omega)^{d \times d}$  ou  $|.|$  s'il n'y a pas d'ambiguité avec la valeur absolue

$(., .)$  pour produit scalaire sur  $L^2(\Omega)^{d \times d}$  ainsi que sur  $L^2(\Omega)$  et  $L^2(\Omega)^d$

$$((u, v)) = (D(u), D(v)) = \int_{\Omega} D(u) : D(v) dx, \quad \forall u, v \in V \text{ et } ||u|| = \sqrt{((u, v))}$$

# INTRODUCTION

L'Université Gustave Eiffel est une université nationale multi-site issue de la fusion récente de plusieurs établissements d'enseignement et de recherche. Le campus de Nantes de cette université, initialement Laboratoire Central des Ponts et Chaussées, est caractérisé par ses grands équipements permettant des expérimentations à échelle réelle dans le domaines du génie civil et plus récemment du développement durable. Ce campus regorge de chercheurs aux compétences très diverses dans son laboratoire GPEM (Granulats et Procédés d'Elaboration des Matériaux) qui vont des mathématiques appliquées au génie civil en passant par la physique, le génie des procédés ou encore la chimie.

Un point crucial reliant toutes les recherches du laboratoire concerne les pâtes granulaires. Ce sont des matériaux visco-plastiques, c'est-à-dire qu'ils se comportent comme un fluide ou un solide localement : plus ils sont soumis à une forte sollicitation à un endroit, plus ils s'écoulent de façon liquide au voisinage de cet endroit, comme de la purée. Il faut signaler que de nos jours, la modélisation des écoulements de pâtes granulaires offre de nombreuses applications en génie civil (bétons, enrobés bitumineux, terres pour la construction, boues d'épuration), environnementales (coulées de boue, avalanches), biologiques (fluides corporels tels que le sang). Le modèle de Bingham [Bingham, 1922] est un modèle viscoplastique et très pratique que l'on utilise généralement comme point de base, car il contient la difficulté fondamentale de la viscoplasticité : l'effet de seuil. Cette difficulté se traduit par l'existence possible de zones de l'écoulement dites rigides de mesure de Lebesgues non-négligeables dans lesquelles le taux de déformation (partie symétrique du gradient de vitesse) est nul.

En général, lorsqu'on a un ouvert borné  $\Omega$  inclus dans  $\mathbb{R}^d$ , le modèle de Bingham s'écrit sous la forme [Bingham, 1922] :

$$\begin{cases} |||\tau^*||| \leq \tau_0 \iff D^*(u^*) = 0 \\ |||\tau^*||| > \tau_0 \iff \tau^* = 2\mu D^*(u^*) + \tau_0 \frac{D^*(u^*)}{|||D^*(u^*)|||} \end{cases} \quad (1)$$

où

$u^*$  : champ de vitesse ,

$\tau_0$  : seuil de plasticité,

$\mu$  : la viscosité,

$D^*(u^*) = \frac{(\nabla u^* + (\nabla u^*)^T)}{2}$  : tenseur de taux de déformation

$\tau^*$  : le déviateur du tenseur de contrainte.

Plusieurs auteurs ont traité ce modèle dans les années récentes et c'est un sujet qui occupe la recherche en mathématiques appliquées depuis des dizaines d'années. On peut en particulier citer les travaux remarquables et pionniers de [Duvaut and Lions, 1972], [Trémolières et al., 2011] et récemment [Saramito and Wachs, 2017] qui dresse un état de l'art des méthodes numériques pour résoudre le système d'équations aux dérivées partielles qui exprime ce modèle d'écoulement.

La résolution mathématique des écoulements de fluide à seuil d'écoulement implique une optimisation convexe de problèmes non réguliers.

Les principales méthodes connues sont les suivantes :

- Lagrangien Augmenté ( **LA** développée dans les années 1980) dans [Fortin and Glowinski, 2000] a gagné en popularité. Elle a été la première à être largement utilisée qui correspond au cadre de l'algorithme de calcul de point de selle, en s'appuyant principalement sur un algorithme de type **UZAWA** pour la minimisation d'une fonctionnelle convexe non-différentiable sous une contrainte linéaire. Récemment, un algorithme amélioré nommé **FISTA**\* a été introduit dans [Teskatis et al., 2018] qui atteint le taux optimal en terme de convergence pour cette classe d'algorithme.
- la méthode de Newton dans [Saramito, 2016] et reformulation de la programmation par cône du second ordre dans [Bleyer, 2018] semblent surpasser tous les autres algorithmes. Toutes ces méthodes ont été mises en place dans le but d'améliorer la vitesse convergence des algorithmes.
- Gradient Projété (**UZAWA \ISTA**, 1976 par [Glowinski, 1976]) utilisé dans [Trémolières et al., 2011] pour trouver une partie non visqueuse appropriée du champ de contraintes, en tant que minimiseur d'une fonctionnelle quadratique convexe sous contrainte convexe ; Ces méthodes itératives de calcul de point de selle sont aussi trouvées dans [Trémolières et al., 2011], [Beck and Teboulle, 2009] et ses variantes récentes, et faciles à programmer mais dont la lenteur reste insatisfaisante en pratique malgré la populaire accélération récente initiée

dans [Beck and Teboulle, 2009] pour le traitement d'images à partir de travaux généraux de [Nesterov, 1983], puis appliquée à Bingham dans [Treskatis et al., 2018].

Le gradient projeté (GP) a été amélioré : dans [Birgin and Martínez, 2014], le Gradient projeté spectral (SPG pour *Spectral Projected Gradient*) est introduit puis sa généralisation *Scaled Gradient Projection* dans [Bonettini et al., 2008]. Notre motivation se base sur le fait que SPG a montré son efficacité supérieure à PG dans de nombreuses applications dans la littérature, mais n'a pas encore été testé pour Bingham. C'est ce que nous proposons dans ce travail.

En fait, c'est au bout d'une ancienne corde que nous en tissons une nouvelle. Nous prenons en effet la suite de travail de Benoît Sénard ancien étudiant de MACS à l'université de Nantes, de la manière suivante : il a considéré l'introduction de propriétés physiques dans le GP, tandis que nous introduisons une méthodes mathématique de calcul automatique du pas de descente  
Le travail est réalisé en quatre parties :

- Dans la première partie, nous présentons le modèle physique et la formulation variationnelle associée.
- Dans la deuxième partie, nous présentons une classe d'algorithme de Gradient Projeté, le Gradient Projeté Spectral, qui utilise la règle de *Barzilai-Borwein* [Barzilai and Borwein, 1988] avec *Grippo-Lemariello-Lucidi* [Grippo et al., 1986] en poursuivant ainsi le travail commencé dans le stage de Benoît Senard.
- Dans la troisième partie, nous exposons les expériences numériques en faisant un approximation spatiale par éléments finis pour présenter des résultats numériques.
- Nous faisons une conclusion de tout ce qui a été abordé et éventuellement nous évoquons nos perspectives possible.

# MODÈLE PHYSIQUE ET FORMULATION VARIATIONNELLE

---

## 1.1 Modèles de Bingham équivalents

Dans ce chapitre, nous formulons sous différentes formes l'écoulement stationnaire d'un matériau de Bingham confiné. Le modèle physique est d'abord écrit sous sa forme la plus usuelle, puis des expressions équivalentes connues dans la littérature mais moins usitées en mécanique sont décrites. Elles conduisent à une formulation adaptée à l'application de l'algorithme de gradient projeté. Le matériau est mis en mouvement avec forces extérieures et en absence de gravitation particulière et le fluide adhère aux parois.

Le modèle de Bingham (1) sans dimension se réécrit :

$$\begin{cases} |||\tau||| \leq Bi \iff D(u) = 0 \\ |||\tau||| > Bi \iff \tau = 2D(u) + Bi \frac{D(u)}{|||D(u)|||} \end{cases} \quad (1.1)$$

où

$u$  : champ de vitesse ;

$Bi$  : le nombre de Bingham, contient l'information sur la viscosité  $\mu$ ,  $\tau_0$  et  $Bi = \frac{\tau_0}{\mu_L U}$ ,  $U$  et  $L$  sont ici respectivement une vitesse et une longueur caractéristiques. Ce sont des grandeurs de référence qui caractérisent les ordres de grandeur du problème traité.

$D(u) = \frac{(\nabla u + (\nabla u)^T)}{2}$  : tenseur de taux de déformation ;

$\tau$  : le déviateur de tenseur de contrainte.

On complète le modèle (1.1) par les lois de conservations stationnaires :

$$\begin{cases} \rho(\nabla u).u - \operatorname{div} \tau - \nabla.p = f & \text{(conservation de quantité de mouvement)} \\ \operatorname{div} u = 0 & \text{(conservation de la masse)} \end{cases} \quad (1.2)$$

où

$(\nabla u) \cdot u$  est négligé dans la suite (*i.e* sans effet d'inertie).

En effet, cette écriture sans dimension se repose sur les grandeurs physiques mises en jeu.

De la relation (1) on a :

$$2\mu D^*(u^*) = \begin{cases} 0 & \text{sinon} \\ \frac{\tau^*}{|||\tau^*|||}(|||\tau^*||| - \tau_0) & \text{si } |||\tau^*||| > \tau_0 \end{cases}$$

et  $u = \frac{u^*}{U}$ ,  $x = \frac{x^*}{L}$  sans dimensions avec  $D = \frac{1}{L}D^*$  donc on obtient :

$$\begin{aligned} 2\mu \frac{U}{L} D(u) &= \begin{cases} 0 & \text{sinon} \\ \frac{\tau}{|||\tau|||}(|||\tau^*||| - \tau_0) & \text{si } |||\tau^*||| > \tau_0 \end{cases} \\ &\Updownarrow \\ 2D(u) &= \begin{cases} 0 & \text{sinon} \\ \frac{\tau}{|||\tau|||}(|||\tau||| - Bi) & \text{si } |||\tau||| > Bi \end{cases} \quad \text{en posant } Bi = \frac{\tau_0}{U} \text{ et } \tau = \frac{\tau^*}{\mu \frac{U}{L}} \end{aligned}$$

Et par conséquent, la relation (1.1) est bien équivalente à la relation (1).

Les équations de conservations dimensionnées qui sont considérées avec le modèle (1) s'écrivent :

$$\begin{cases} -\operatorname{div} \tau^* - \nabla \cdot p^* = f^* & (\text{conservation de quantité de mouvement}) \\ \operatorname{div} u^* = 0 & (\text{conservation de la masse}) \end{cases}$$

Puisque, la pression  $p^*$  et le second membre  $f^*$  sont homogènes à des contraintes, tout comme  $\tau^*$  alors elles se comportent comme ce dernier *i.e.*  $f = \frac{f^*}{U}$  et  $p = \frac{p^*}{U}$ . De plus  $u = \frac{u^*}{U}$ . Et par suite, on obtient aussi les relations (1.2).

**Définition 1.1.** On définit un champs de contrainte  $\lambda$  par la relation :

$$\tau = 2D(u) + \lambda Bi \tag{1.3}$$

La solution du problème qu'on propose consiste à appliquer une méthode en trouvant  $\lambda$  comme argument minimum d'une certaine fonctionnelle que nous allons découvrir très rapidement, et qui permet d'appliquer le gradient projeté.

- Pour l'accueil compréhensif des notions autour du modèle, commençons par quelques préliminaires de définitions, propriétés, remarques et lemmes intéressantes pour la suite du travail.

La Caractérisation de  $\lambda \in L^2(\Omega)^{d \times d}$  de la relation (1.3) pour  $u \in H^1(\Omega)^d$  que l'on propose est donnée par :

$$\left\{ \begin{array}{ll} \lambda = \frac{D(u)}{\| |D(u)| \|} & \text{si } D(u) \neq 0 \\ & \text{p.p. sur } \Omega \\ \| | \lambda | \| \leq 1 & \text{si } D(u) = 0 \end{array} \right. \quad (1.4)$$

**Définition 1.2.** On définit l'ensemble  $K = \left\{ \mu \in L^2(\Omega)^{d \times d}, \| | \mu | \| \leq 1 \text{ p.p sur } \Omega \right\}$  qui est convexe.

En effet,  $\forall \mu, \nu \in K$  et  $\alpha \in [0, 1]$  on a :

$$\begin{aligned} \| | \alpha \mu + (1 - \alpha) \nu | \| &\leq \| | \alpha \mu | \| + (1 - \alpha) \| | \nu | \| \\ &\leq \alpha + (1 - \alpha) \\ &\leq 1 \text{ car } \| | \mu | \| \leq 1 \text{ et } \| | \nu | \| \leq 1 \end{aligned}$$

**Lemme 1.1.** Nous pouvons observer l'équivalence entre la loi de comportement (1.1) et la définition de  $\lambda$  (1.3) :

$$\tau = 2D(u) + \lambda Bi \iff (1.1)$$

*Démonstration.* Supposons  $\tau = 2D(u) + \lambda Bi$

- Montrons que  $\| | \tau | \| \leq Bi \iff D(u) = 0$

Si  $\| | \tau | \| \leq Bi$

On a :  $\| | \tau | \| \leq Bi < Bi \| | \lambda | \| + 2 \| | D(u) | \|$

$$\text{donc } (1 - \| | \lambda | \|) \leq \frac{2}{Bi} \| | D(u) | \|$$

Supposons par l'absurde que  $D(u) \neq 0$ . Alors  $\| | \lambda | \| = 1$  (caractérisation de  $\lambda$ ). Il suit que  $0 \leq \frac{2}{Bi} \| | D(u) | \|$  et  $\| | D(u) | \| = 0$  (impossible).

Si  $D(u) = 0$

On a :  $\| | \tau | \| \leq Bi \| | \lambda | \| + 2 \| | D(u) | \|$  puis  $\| | D(u) | \| = 0$  et  $\| | \lambda | \| \leq 1$  (caractérisation de  $\lambda$ ). Donc  $\| | \tau | \| \leq Bi$ .

• Montrons que  $|||\tau||| > Bi \iff \tau = 2D(u) + Bi \frac{D(u)}{|||D(u)|||}$ .

si  $|||\tau||| > Bi$

On a :  $\tau = 2D(u) + Bi\lambda = 2D(u) + Bi \frac{D(u)}{|||D(u)|||}$  en identifiant  $\lambda = \frac{D(u)}{|||D(u)|||}$  avec  $D(u) \neq 0$ .

Si  $\tau = 2D(u) + Bi \frac{D(u)}{|||D(u)|||} = 2D(u) + Bi\lambda$  (caractérisation de  $\lambda$ ).

On suppose par l'absurde que  $|||\tau||| < Bi$

Alors  $|||D(u)||| = 0$  et  $|||\tau||| = Bi|||\lambda|||$ . Or  $|||\lambda||| = 1$ , donc  $|||\tau||| = Bi$  (impossible). Il suit que  $|||\tau||| > Bi$ .

Réiproquement, en supposant (1.1) avec la caractérisation de  $\lambda$  proposé, il est évident de constater que  $\tau = 2D(u) + \lambda Bi$ .  $\square$

**Remarque 1.1.** Nous verrons dans une proposition qu'en fait, pour tout  $\rho > 0$  on a :

$$\lambda = \begin{cases} \frac{\lambda + \rho D(u)}{|\lambda + \rho D(u)|} & \text{si } |\lambda + \rho D(u)| \geq 1 \\ \lambda + \rho D(u) & \text{sinon} \end{cases}$$

Toutes ces formulations ont un intérêt algorithmique. Nous nous limitons ici aux formulations qui nous sont utiles, mais d'autres écritures existent, qui sont utilisées dans la littérature pour pouvoir appliquer d'autre méthodes.

**Rappel :** (Caractérisation de la projection)

Soit  $X$  un espace de Hilbert avec son produit scalaire  $\langle \cdot, \cdot \rangle_X$ , et  $K$  une partie non-vide, convexe et fermée de  $X$ . Alors la projection orthogonale  $P_K$  vérifie, pour tout  $x \in X$ ,  $y \in K$

$$\langle x - P_K(x), y - P_K(x) \rangle_X \leq 0$$

**Rappel :**  $P_K$  la projection orthogonale sur  $K$  donnée par :

$$P_K \begin{cases} \mu & \text{si } |||\mu||| < 1 \\ \frac{\mu}{|||\mu|||} & \text{sinon} \end{cases}$$

**Remarque 1.2.** Le lemme suivant est un résultat classique qui joue un rôle essentiel dans l'écriture de classe d'algorithme pour Bingham. les assertions qui le constituent sont en général éparpillé dans différents ouvrages comme par exemple dans ([Glowinski, 2008]).

**Lemme 1.2.** Soit  $\lambda \in L^2(\Omega)^{d \times d}$  et  $u \in V$ . Alors nous avons les équivalences suivantes :

1.  $(\mu - \lambda, D(u)) \leq 0, \mu \in K$

2.  $\tau = 2D(u) + \lambda Bi$

3.  $\lambda = P_K(\lambda + \rho D(u)), \forall \rho > 0$

4.  $\lambda : D(u) = |||D(u)||| , |\lambda| \leq 1$

Démonstration.  $\odot 1. \implies 3.$

Supposons 1.

Soit  $\rho > 0$ . En multipliant par  $\rho$  alors  $(\mu - \lambda, \rho D(u)) \leq 0$

$\Leftrightarrow (\mu - \lambda, \lambda + \rho D(u) - \lambda) \leq 0$  (on ajoute et retranche  $\lambda$ )

$\Leftrightarrow \lambda = P_K(\lambda + \rho D(u))$  (caractérisation de la projection) i.e 3. est donc obtenu.

$\odot 3. \implies 2.$

Supposons 3. Montrons que  $\tau = 2D(u) + \lambda Bi$

On sait que par définition de  $\lambda$  :  $\tau = 2D(u) + \lambda Bi = 2D(u) + P_K(\lambda + \rho D(u))$

- si  $D(u) = 0$  Alors  $P_K(\lambda) = \lambda$ .

A-t-on  $|||\tau||| \leq Bi$  ?

On a  $\tau = 2 \times 0 + \lambda Bi$  et  $|||\tau||| = Bi|||\lambda|||$ . Puisque  $\lambda \in K$  alors  $|||\tau||| \leq Bi$ .

Par conséquent, on bien  $\tau = 2D(u) + \lambda Bi$

- si  $D(u) \neq 0$ . Nous utilisons la définition de  $\lambda$  donnée par :  $\tau = 2D(u) + \lambda Bi = 2D(u) + P_K(\lambda + \rho D(u))$ .

Montrons que  $P_K(\lambda + \rho D(u)) = \frac{D(u)}{|||D(u)|||}$ . Puisque  $|\lambda + \rho D(u)| > 1$  alors

$$P_K(\lambda + \rho D(u)) = \frac{\lambda + \rho D(u)}{|\lambda + \rho D(u)|} = \frac{\frac{D(u)}{|||D(u)|||} (1 + \rho |||D(u)|||)}{1 + \rho |||D(u)|||} = \frac{D(u)}{|||D(u)|||} = \lambda$$

Par suite,  $\tau = 2D(u) + \lambda Bi$ .

$\odot 2. \implies 4.$

Supposons 2. et montrons que  $\lambda : D(u) = |||D(u)||| , |\lambda| \leq 1$

Comme  $|\lambda| \leq 1$  alors  $D(u) = 0$ . Ainsi  $|||D(u)||| = 0$  et  $\lambda : D(u) = 0 = |||D(u)|||$ .

De plus on a bien  $|||\tau||| \leq Bi$  car  $|\lambda| \leq 1$  et  $D(u) = 0$

$\odot 4. \implies 1.$

Supposons 4.

Soit  $\mu \in K$ . Montrons que  $(\mu - \lambda, D(u)) \leq 0$

On a :

$$\begin{aligned}
 (\mu - \lambda) : D(u) &= \mu : D(u) - \lambda : D(u) \\
 &= \mu : D(u) - |||D(u)||| \\
 &\leq |\mu| \cdot |||D(u)||| - |||D(u)||| \text{ (Chauchy – Schwartz avec } \mu : D(u)) \\
 &\leq (|\mu| - 1) |||D(u)||| \\
 &\leq 0 \text{ car } \mu \in K
 \end{aligned}$$

□

**Proposition 1.1.** Soit  $\rho > 0$ , la caractérisation de  $\lambda$  de la relation (1.4) est équivalente à la relation :

$$\lambda = P_K(\lambda + \rho D(u)) \quad (1.5)$$

Démonstration. • Si  $D(u) = 0$ , l'équivalence est évidente :  $0 = P_K(0) \iff 0 \leq 1$

• Si  $D(u) \neq 0$ . Soit  $\rho > 0$  et supposons que  $\lambda = \frac{D(u)}{|||D(u)|||}$  i.e 1 est vérifié. On a :

$$|\lambda + \rho D(u)| = \left| \frac{D(u)}{|||D(u)|||} (1 + \rho |||D(u)|||) \right| = 1 + \rho |||D(u)||| > 1$$

et par définition de  $P_K$ , on a :

$$P_K(\lambda + \rho D(u)) = \frac{\lambda + \rho D(u)}{|\lambda + \rho D(u)|} = \frac{\frac{D(u)}{|||D(u)|||} (1 + \rho |||D(u)|||)}{1 + \rho |||D(u)|||} = \frac{D(u)}{|||D(u)|||} = \lambda$$

et donc la relation (1.6) est ainsi obtenue.

Dans le sens inverse, en supposant (1.6) i.e  $\lambda = P_K(\lambda + \rho D(u))$

On a nécessairement  $|\lambda + \rho D(u)| > 1$  sinon on aurait  $\rho = 0$  puisque  $D(u) \neq 0$ .

Par conséquent, en utilisant une fois encore la définition de  $P_K$ , on a  $\lambda = \frac{\lambda + \rho D(u)}{|\lambda + \rho D(u)|}$ . Et par factorisation on obtient :  $\lambda (|\lambda + \rho D(u)| - 1) = \rho D(u)$  et donc  $D(u) = \frac{\lambda (|\lambda + \rho D(u)| - 1)}{\rho}$ .

En passant à la norme on a :  $|||D(u)||| = \frac{|\lambda + \rho D(u)| - 1}{\rho}$  en remarquant que  $|\lambda| = 1$ .

D'où

$$\lambda = \frac{\lambda + \rho D(u)}{|\lambda + \rho D(u)|} \implies \lambda = \frac{D(u)}{|||D(u)|||}.$$

□

## 1.2 Formulation variationnelle

Pour écrire la formulation variationnelle que l'on propose pour Bingham, nous allons donner une idée qui permet de coupler cette formulation sous forme de définition liant  $(u, \lambda) \in V \times K$

On rappelle que la formule de Green valable pour le  $\nabla(\cdot)$  est aussi valable avec  $D(\cdot)$ , comme montré dans [Brezzi and Fortin, 2012] :

$$\int_{\Omega} \operatorname{div}(\xi) \cdot v \, dx = - \int_{\Omega} \xi : D(v) \, dx + \int_{\partial\Omega} (\xi \cdot n) \cdot v \, d\sigma$$

avec  $dx$  la mesure de Lebesgue,  $d\sigma$  la mesure de surface sur le bord,  $\xi$  un champs régulier de contraintes et  $v$  un champs de vitesse suffisamment réguliers.

D'une part, concernant la conservation de la quantité de mouvement :

- Dans un sens d'implication de l'écriture variationnelle  $\forall v \in V$ , en multipliant l'équation de quantité de mouvement (1.2) par  $v$  et en appliquant la formule de Green, on a :

$$\begin{aligned} - \int_{\partial\Omega} (\tau \cdot n) \cdot v \, d\sigma + \int_{\Omega} D(v) : \tau \, dx + \int_{\Omega} \operatorname{div}(v) p \, dx - \int_{\partial\Omega} p v \cdot n \, d\sigma &= \int_{\Omega} f \cdot v \, dx \\ \Rightarrow 2 \int_{\Omega} D(u) : D(v) \, dx + Bi \int_{\Omega} \lambda : D(v) \, dx &= \int_{\Omega} f \cdot v \, dx \end{aligned}$$

en utilisant la définition de :

$$\tau = 2D(u) + \lambda Bi,$$

avec  $v \in V = \left\{ w \in H_0^1(\Omega)^d, \quad \operatorname{div} w = 0 \right\}$  comme l'espace de fonction test, cela implique :

$$2(D(u), D(v)) + Bi(\lambda, D(v)) = (f, v), \quad \forall v \in V$$

- Dans le sens inverse de l'écriture variationnelle, on se repère à la condition «*inf-sup*» (dans [Brezzi and Fortin, 2012]). Cette condition assure la surjectivité de l'opérateur divergence i.e. :  $\exists p \in L_0^2(\Omega) = \left\{ u \in L^2(\Omega)^d, \quad \int_{\Omega} u = 0 \right\}$  tel que

$$\int_{\Omega} D(v) : \tau \, dx + \int_{\Omega} \operatorname{div}(v) p \, dx = \int_{\Omega} f \cdot v \, dx$$

D'autre part, le Lemme (1.2) montre que l'inéquation variationnelle  $\forall \mu \in K, (\mu - \lambda, D(u)) \leq 0$  est équivalente à la loi de comportement locale.

**Définition 1.3.** La vitesse  $u \in V$  et le champ de tension  $\lambda \in K$  sont liés dans le problème de Bingham par la formulation variationnelle :

$$\begin{cases} 2(D(u), D(v)) + Bi(\lambda, D(v)) = (f, v), \quad \forall v \in V \\ (\mu - \lambda, D(u)) \leq 0, \quad \mu \in K \end{cases} \quad (1.6)$$

- Par le théorème de Lax-Milgram, on peut remarquer que le problème variationnel : pour  $\mu \in L^2(\Omega)^{d \times d}$  donné, trouver  $u(\mu) \in V$  telle que :

$$2(D(u(\mu)), D(v)) + Bi(\mu, D(v)) = (f, v), \quad \forall v \in V$$

admet une unique solution puisque :

- $V$  est un espace de Hilbert dans ([Brezzi and Fortin, 2012]), ([Girault and Raviart, 1979]) et ([Brenner et al., 2008]),
- De plus,  $(v, w) \mapsto 2(D(v), D(w))$  est une forme bilinéaire, continue et coercive sur  $V$ ;
- $v \mapsto (f, v) - Bi(\mu, D(v))$  est une forme linéaire continue sur  $V$ .

Ainsi, nous pouvons introduire une fonctionnelle  $\phi : L^2(\Omega) \rightarrow \mathbb{R}$  définie par :

$$\phi(\mu) = \frac{1}{Bi} |D(u(\mu))|^2 \quad (1.7)$$

Dans la proposition suivante, nous abordons les propriétés de  $\phi$  qui permettent d'écrire la formulation variationnelle sous la forme d'un problème de minimisation.

**Proposition 1.2.** Soit  $\phi$  définie par (1.7). Alors les deux assertions suivantes sont équivalentes :

1.  $\lambda = \operatorname{armin}_{\mu \in K} \phi(\mu)$
2.  $(\mu - \lambda, D(u(\lambda))) \leq 0, \mu \in K$

*Démonstration.* 1.  $\Rightarrow$  2. Supposons que  $\lambda = \operatorname{armin}_{\mu \in K} \phi(\mu)$

donc  $\phi(\lambda) \leq \phi(\mu), \forall \mu \in K$

Soit  $\nu \in K, \alpha \in [0, 1]$ . On a  $K \ni (\alpha\nu + (1 - \alpha)\lambda) = \lambda + \alpha(\nu - \lambda)$

ainsi  $\phi(\lambda) \leq \phi(\alpha\nu + (1 - \alpha)\lambda) = \phi(\lambda + \alpha(\nu - \lambda))$

donc  $0 \leq \phi(\lambda + \alpha(\nu - \lambda)) - \phi(\lambda)$

Il suit que  $0 \leq \frac{1}{\alpha} (\phi(\lambda + \alpha(\nu - \lambda)) - \phi(\lambda))$

Et en faisant tendre  $\alpha$  vers 0 et en utilisant la formule du taux de variation, nous obtenons

$0 \leq (\nu - \lambda, \phi'(\lambda))$

Enfin par le Théorème 1.1 nous obtenons  $(\nu - \lambda, D(u(\lambda))) \leq 0$ .

1.  $\Leftarrow$  2. Supposons  $(\mu - \lambda, D(u(\lambda))) \leq 0, \mu \in K$  i.e  $0 \leq (\nu - \lambda, \phi'(\lambda))$  (Théorème (1.1))

Ainsi  $0 \leq \frac{1}{\alpha} (\phi(\lambda + \alpha(\mu - \lambda)) - \phi(\lambda))$  par la formule de taux de variation avec  $\alpha \rightarrow 0$

donc  $\phi(\lambda) \leq \phi(\lambda + \alpha(\mu - \lambda))$  et on prend  $\nu = \lambda + \alpha(\mu - \lambda) = (\alpha\mu + (1 - \alpha)\lambda)$  in  $K$  est quelconque car  $\mu$  est quelconque.

Il suit que  $\phi(\lambda) \leq \phi(\nu), \forall \nu \in K$  Autrement  $\lambda = \operatorname{armin}_{\mu \in K} \phi(\mu)$  □

**Remarque 1.3.** Nous avons écrit le problème de Bingham comme la minimisation d'une fonctionnelle convexe quadratique sur un convexe, dans le but d'appliquer des algorithmes connus pour ce contexte.

### 1.3 Existence et unicité

Le théorème suivant traite des propriétés de  $\phi$  qui garantissent l'existence d'une solution.

**Théorème 1.1.** Soit  $\phi$  la fonctionnelle de  $L^2(\Omega) \rightarrow \mathbb{R}$  définie par (1.7). Alors

1.  $\phi$  est convexe.
2.  $\phi$  est dérivable au sens de Fréchet sur  $L^2(\Omega)$ , de dérivée  $\phi'(\mu) = -D(u(\mu))$ .

Démonstration. 1. On sait que par définition  $u$  satisfait :

$$u(\alpha\lambda + (1-\alpha)\mu) = \alpha u(\lambda) + (1-\alpha)u(\mu), \forall \alpha \in [0;1], \lambda, \mu \in K$$

On a :

$$\begin{aligned} 2Bi\phi(\alpha\lambda + (1-\alpha)\mu) &= ||| \alpha D(u(\lambda)) + (1-\alpha)D(u(\mu)) |||^2 \\ &\leq (\alpha ||| D(u(\lambda)) ||| + (1-\alpha) ||| D(u(\mu)) |||)^2 \quad (\text{ingalit triangulaire}) \\ \phi(\alpha\lambda + (1-\alpha)\mu) &\leq \alpha^2\phi(\lambda) + (1-\alpha)^2\phi(\mu) + 2\alpha(1-\alpha)||| D(u(\mu)) ||| \times ||| D(u(\lambda)) ||| \\ \phi(\alpha\lambda + (1-\alpha)\mu) &\leq \alpha\phi(\lambda) + (1-\alpha)\phi(\mu) \end{aligned}$$

2. Soit  $\mu, \eta \in L^2(\Omega)$ .

Par un calcul d'identité remarquable, on a :

$$||| D(u(\mu+\eta)) |||^2 - ||| D(u(\mu)) |||^2 = ||| D(u(\mu+\eta)) - D(u(\mu)) |||^2 + 2(D(u(\mu+\eta)) - D(u(\mu)), D(u(\mu)))$$

En utilisant la ligne 1 de l'équation (1.1) on obtient :

$$(D(u(\mu+\eta)) - D(u(\mu)), D(v)) + Bi(\eta, D(v)) = 0, \forall v \in V \quad (1.8)$$

On prend  $v = (u(\mu))$  dans 1.8. Il vient donc que :

$$||| D(u(\mu+\eta)) |||^2 - ||| D(u(\mu)) |||^2 = ||| D(u(\mu+\eta)) - D(u(\mu)) |||^2 - 2Bi(\eta, D(u(\mu)))$$

D'autre part, on prend aussi  $v = u(\mu+\eta) - u(\mu)$  dans 1.8 et on a :

$$\begin{aligned} ||| D(u(\mu+\eta)) |||^2 - ||| D(u(\mu)) |||^2 &= -Bi(\eta, D(u(\mu+\eta)) - D(u(\mu))) \\ &\leq Bi|\eta|.||| D(u(\mu+\eta)) - D(u(\mu)) ||| \quad (\text{Cauchy-Schwarz}) \\ &\leq Bi^2 \frac{|\eta|^2}{2} + \frac{1}{2} ||| D(u(\mu+\eta)) - D(u(\mu)) |||^2 \quad (\text{Young}) \end{aligned}$$

Donc  $||| D(u(\mu+\eta)) - D(u(\mu)) |||^2 := O(|\eta|^2)$ .

Il suit que :  $\phi(\mu+\eta) = \phi(\mu) + (\eta, D(u(\mu))) + O(|\eta|^2)$

Par conséquent,  $\phi'(\mu) = -D(u(\mu))$

□

**Corollaire 1.1.** Il existe  $\lambda \in K$  solution de  $\min_{\mu \in K} \phi(\mu)$ .

La preuve repose sur le fait  $\phi$  est propre et qu'avec les propriétés du théorème 1.1, le théorème XXX de [?] s'applique.

Ainsi d'après le Théorème (1.1),  $(u(\lambda), \lambda) \in V \times K$  est donc solution du problème (1.3) de Bingham.

$u(\lambda)$  est unique mais  $\lambda$  ne l'est pas toujours (voir dans [Duvaut and Lions, 1972]).

**Remarque 1.4.** *L'unicité de  $u(\lambda)$  n'est pas l'application de Lax-Milgram. En effet, Lax-Milgram prouve que  $u(\mu)$  est unique pour un  $\mu$  donné. Considérons donc  $\lambda_1$  et  $\lambda_2$  deux minimiseurs de  $\phi$ , et  $u_1$  et  $u_2$  les vitesses respectivement associées. On a :*

$$((u_1 - u_2, v)) + Bi(\lambda_1 - \lambda_2, D(v)) = 0, \forall v \in V$$

en choisissant  $v = u_1 - u_2$ , il vient :

$$||u_1 - u_2||^2 + Bi(\lambda_1 - \lambda_2, D(u_1 - u_2)) = 0$$

Or, les inéquations variationnelles (1.3) permettent d'écrire, pour  $i = 1, 2$  :

$$(\mu - \lambda_i, D(u_i)) \leq 0$$

ce qui donne, en sommant les inéquations pour  $\mu = \lambda_1$  avec  $i = 2$ , et  $\mu = \lambda_2$  avec  $i = 1$  :

$$(\lambda_1 - \lambda_2, D(u_1 - u_2)) \leq 0$$

On obtient donc  $||u_1 - u_2|| \leq 0$ . Ceci prouve l'unicité de la vitesse  $u$ .

# GRADIENT PROJETÉ (GP) AVEC LA RÈGLE DE BARZILAI-BORWEIN ET GRIPPO-LAMPORIELLO-LUCIDI

## 2.1 Formulation - Minimisation par gradient projeté

La formulation variationnelle du modèle de Bingham donné dans 1.6 et reformulée sous forme de minimisation dans la proposition 1.2 permet d'écrire l'algorithme classique du Gradient Projété ([Trémolières et al., 2011]).

La méthode du gradient projeté que nous mettons en place ici calcule à chaque itération une nouvelle approximation d'un minimum de  $\phi$  en progressant d'un pas  $\rho$  le long de la direction opposée à celle de  $\phi'$  au point courant, puis en projetant sur  $K$  le point obtenu.

Soit  $\rho > 0$ ,  $\lambda^0 \in K$  et  $n \in \mathbb{N}$ . Par récurrence, la suite  $(u^{n+1}, \lambda^{n+1}) \in V \times K$  se calcule comme suit :

$$\begin{cases} 2(D(u^{n+1}), D(v)) + B_i(\lambda^n, D(v)) = (f, v), & \forall v \in V \\ \lambda^{n+1} = P_K(\lambda^n + \rho D(u^{n+1})) \end{cases} \quad (2.1)$$

On montre que cette suite converge vers  $(u, \bar{\lambda})$  qui est dans  $V \times K$  aussi donc solution du problème, avec une convergence faible pour  $\lambda^{n+1}$  et forte pour  $u^{n+1}$  (voir [Trémolières et al., 2011]). Pour établir un ordre de convergence  $|\phi(\lambda^n) - \phi(\lambda)| = (O)(1/k)$  (voir [Beck and Teboulle, 2009] pour la démonstration générale, avec une fonctionnelle  $\phi$  convexe différentiable), l'idée fondamentale se base sur le fait que la dérivée de la fonctionnelle  $\phi$  est *lipschitzienne*. Ce qui peut être établi dans le cas de Bingham :

**Proposition 2.1.**  $\phi'$  est Bi lipschitzienne et  $\phi$  localement lipschitzienne

En effet, d'après la ligne 1 de l'équation (1.1) on sait que :

$$(D(u(\lambda)) - D(u(\mu)), D(v)) = -B_i(\lambda - \mu, D(v))$$

D'autre part,

$$\begin{aligned} |||D(u(\lambda))|||^2 - |||D(u(\mu))|||^2 &= |||D(u(\lambda)) - D(u(\mu))|||^2 + 2(D(u(\lambda)) - D(u(\mu)), D(u(\mu))) \\ &= -Bi(\lambda - \mu, D(u(\lambda)) - D(u(\mu))) - 2Bi(\lambda - \mu, D(u(\mu))) \\ \text{car } |||D(u(\lambda)) - D(u(\mu))|||^2 &= -Bi(\lambda - \mu, D(u(\lambda)) - D(u(\mu))), \text{ pour } v = u(\lambda) - D(u(\mu)) \\ \text{et donc } |D(u(\lambda)) - D(u(\mu))| &\leq Bi|\lambda - \mu| \text{ (par Cauchy - Schwarz)} \\ \text{Il suis que } \phi' &\text{ est Bi lipschitzienne} \end{aligned}$$

De plus,

$$|||D(u(\lambda))|||^2 - |||D(u(\mu))|||^2 = -Bi(\lambda - \mu, D(u(\lambda)) + D(u(\mu)))$$

En réappliquant Cauchy-Schwarz, on obtient :

$$|||D(u(\lambda))|||^2 - |||D(u(\mu))|||^2 \leq Bi|\lambda - \mu|.|||D(u(\lambda)) + D(u(\mu))|||.$$

Et puisque  $(D(u(\lambda)) + D(u(\mu)), D(v)) = 2(f, v) + g(\lambda + \mu, D(v))$ ,  $\forall v \in V$ , il s'en suit que :

$$\begin{aligned} |||D(u(\lambda)) + D(u(\mu))||| &\leq 2|f| + Bi|\lambda + \mu| \\ &\leq 2(|f| + Bi), \text{ car } \lambda, \mu \in K \end{aligned}$$

Ainsi

$$\begin{aligned} \phi(\lambda) - \phi(\mu) &\leq \frac{1}{2Bi} \times 2Bi|\lambda - \mu|(|f| + Bi) \\ &\leq (|f| + Bi)|\lambda - \mu| \end{aligned}$$

## 2.2 Calcul automatique d'un pas de descente variable

Dans l'algorithme de gradient projeté (2.1), le pas de descente est constant. On peut toutefois ce demander si un pas variable  $\rho^n$  permettrait d'accélérer la convergence.

Dans ce but, la méthode du gradient projeté spectral (SPG) de Birgin, Martinez et Raydan en 2000-2003 pour la résolution de problèmes avec contraintes convexes, est née du mariage du schéma non monotone de Barzilai- Borwein (Raydan 1997) avec la méthode classique du gradient projeté (Bertsekas 1976, Goldstein 1964, Levitin et Polyak 1966) qui ont été utilisés dans les statistiques.

### 2.2.1 Généralité sur Barzilai et Borwein (BB)

Dans ([Birgin et al., 2000]) les auteurs ont relaté que Barzilai et Borwein (BB) en 1988 dans [Barzilai and Borwein, 1988] ont proposé une méthode pour la minimisation *sans contrainte* d'une

fonction différentiable  $\phi_N : \mathbb{R}^N \rightarrow \mathbb{R}$

On se donne  $x^0 \in \mathbb{R}^N$ , l'itération BB est donnée par :

$$x^{k+1} = x^k - \rho^k \nabla \phi_N(x^k) \quad (2.2)$$

où  $\rho$  est la longueur de pas,  $\rho^0$  est arbitraire et  $\forall k = 1, \dots,$

$$\rho^k = \frac{S_{k-1}^T S_{k-1}}{S_{k-1}^T y_{k-1}} \quad (2.3)$$

avec  $S_{k-1} = x^k - x^{k-1}$  et  $y_{k-1} = \nabla \phi_N(x^k) - \nabla \phi_N(x^{k-1})$

Lorsque  $\phi_N(x) = \frac{1}{2}x^T \mathcal{A}x + bx^T + c$  est la fonction quadratique et  $\mathcal{A}$  une matrice symétrique définie positive alors :

$$\rho^k = \frac{\nabla \phi_N(x^{k-1})^T \phi_N(x^{k-1})}{\phi_N(x^{k-1})^T \mathcal{A} \phi_N(x^{k-1})}$$

Les auteurs définissent le Spectral Projected Gradient (**SPG**) dans lequel on utilise le pas de descente Barzilai et Borwein (BB), en conjonction avec une généralisation de la règle d'Armijo, nommée règle **GLL** (pour Grippo, Lampariello et Lucidi). SPG est défini dans l'article fondateur : [Birgin et al., 2000] et **GLL** est définie dans : [Grippo et al., 1986].

Le terme Spectral provient du fait que le pas de descente BB est compris entre les valeurs propres extrêmes de la dérivée seconde de la fonctionnelle  $\phi$ . La règle d'Armijo garantit que la fonctionnelle à minimiser décroît au fil des itérations. Cependant, la littérature a progressivement constaté que la vitesse de convergence peut être significativement améliorée si on relaxe la monotonie en ne l'imposant que tous les M itérations, l'algorithme étant au contraire inefficace sinon. C'est le but de GLL en remplacement d'Armijo. Cela semble vrai notamment pour **SPG**.

**Remarque 2.1.** • L'efficacité de la méthode BB est liée à la relation entre les  $\rho_k$  et les valeurs propres de la matrice Hésienne de  $\phi_N$  et non à la diminution de la valeur de la fonction.

• **SPG** a montré son efficacité supérieure à PG dans ces nombreuses applications.

$\phi$  étant une fonctionnelle quadratique, elle peut s'écrire :

$$\phi(\mu) = \frac{1}{B} |F - Q\mu|^2$$

en faisant les identifications suivantes :

on définit  $A : V \rightarrow V'$  et  $B : V \rightarrow L^2(\Omega)^{d \times d}$  par des crochets de dualité

$$\begin{aligned} & \langle Av, w \rangle = (D(v), D(w)) \quad \forall v, w \in V \\ & (Bv, \mu) = (D(v), \mu) \quad \forall v \in V \quad \forall \mu \in L^2(\Omega)^{d \times d} \end{aligned}$$

et

$$Q = BiBA^{-1}B^*; F = BA^{-1}f$$

On a :

$$\phi'(\mu) = \frac{2}{Bi}Q^*(Q\mu - F); \phi''(\mu) = \frac{2}{Bi}Q^*Q$$

- l’Idée fondamentale de **BB** est de chercher  $\rho^k \in \mathbb{R}_+^*$  tel que  $\rho^k \phi'(\lambda^k)$  soit proche de  $\left\{\phi''(\lambda^k)\right\}^{-1} \phi'(\lambda^k)$ .

En faisant le calcul de  $\rho^k$  au lieu d’utiliser les expressions de Barzilai et Borwein en espérant trouver une expression simple et facile à calculer, on a :

$$\begin{aligned} \rho^k &= \operatorname{argmin}_{\rho} |\rho \lambda^k - (\phi'')^{-1} \lambda^k|^2 \\ &\Leftrightarrow (\lambda^k, \rho \lambda^k - (\phi'')^{-1} \lambda^k) = 0 \\ &\Leftrightarrow \rho^k = \frac{(\lambda^k, (\phi'')^{-1} \lambda^k)}{|\lambda^k|^2} \end{aligned}$$

Dans [Birgin et al., 2000] il est rappelé que la règle BB s’écrit classiquement, à partir de l’équation de la sécante :

$$\rho_{sec}^k := \frac{|\lambda^k - \lambda^{k-1}|^2}{(\lambda^k - \lambda^{k-1}, \phi'(\lambda^k) - \phi'(\lambda^{k-1}))} = \frac{|\lambda^k - \lambda^{k-1}|^2}{(\lambda^k - \lambda^{k-1}, \frac{1}{Bi}Q^*Q(\lambda^k - \lambda^{k-1}))}$$

Dans le cas où il n’y a pas de contrainte, l’itération de descente est :

$$\lambda^k - \lambda^{k-1} = -\rho^{k-1} \phi'(\lambda^{k-1})$$

et donc nous obtenons l’expression équivalente :

$$\rho_{quad}^k := \frac{||\phi'(\lambda^{k-1})||^2}{(\phi'(\lambda^{k-1}), \frac{1}{Bi}Q^*Q\phi'(\lambda^{k-1}))}$$

Pour Bingham, on a en outre les propriétés :

$$Q = Q^*; Q^2 = BiQ \text{ puisque } A = BB^*$$

Par suite nous avons les formes simplifiées :

**Lemme 2.1.** Soient les opérateurs  $A$ ,  $B$  et  $Q$  définis précédemment. Alors les expressions de  $\rho_{sec}$  et de  $\rho_{quad}$  sont données par :

$$\rho_{sec}^k := \frac{|\lambda^k - \lambda^{k-1}|^2}{(\lambda^k - \lambda^{k-1}, Q(\lambda^k - \lambda^{k-1}))} ; \rho_{quad}^k := \frac{|\phi'(\lambda^{k-1})|^2}{(\phi'(\lambda^{k-1}), Q\phi'(\lambda^{k-1}))}$$

- $\rho_{sec}$  est toujours positif d’après le Corollaire suivant :

**Corollaire 2.1.** Soient les opérateurs  $A$ ,  $B$  et  $Q$  définis précédemment. Alors :

$$\rho_{quad}^k := \frac{1}{Bi}$$

Démonstration. On sait que :

$\rho_{quad}^k := \frac{|\phi'(\lambda^{k-1})|^2}{(\phi'(\lambda^{k-1}), Q\phi'(\lambda^{k-1}))}$  et en remplaçant  $Q = Bi BA^{-1}B^*$  et  $A = B^*B$ , on obtient :

$$\begin{aligned}\rho_{quad}^k &:= \frac{|\phi'(\lambda^{k-1})|^2}{(\phi'(\lambda^{k-1}), Bi I\phi'(\lambda^{k-1}))} \\ &= \frac{|\phi'(\lambda^{k-1})|^2}{Bi |\phi'(\lambda^{k-1})|^2} \\ &= \frac{1}{Bi}\end{aligned}$$

□

Quelles sont donc les relations entre  $\rho_{sec}^k$  et  $\rho_{quad}^k$  ?

- A priori,  $\rho_{quad}$  utilise l'information de dérivabilité seconde, et possède donc un avantage.
- Comme la projection affaiblit cet avantage, il est peut-être préférable d'utiliser  $\rho_{sec}$  qui informe sur le convexe  $K$ .
- Pour prendre du recul sur l'algorithme **SPG** qui vient, lorsqu'on utilise  $\lambda$  plutôt que  $\tau$  et en respectant nos notations habituelles, nous présentons cet algorithme dit de gradient projeté qui est encore l'itération de (**BB**), et qui se caractérise par :

$$\lambda^{n+1} = \lambda^n - \rho G_\rho(\lambda^n) \quad (2.4)$$

où

$$G_\rho(\mu) = (-\mu + P_K(\mu - \rho\psi(\mu)))$$

est une direction de descente de et  $\rho$  un pas de descente,

avec

$$\psi(\mu) = \phi(\mu) + 1_K(\mu) \text{ et}$$

$$1_K(x) \left\{ \begin{array}{ll} 0 & \text{si } x \in K \\ +\infty & \text{sinon} \end{array} \right.$$

## 2.2.2 Formulation Algorithmique

**Lemme 2.2.**  $\mu \mapsto u(\mu)$  est convexe et vérifie de plus la propriété :

$$u(\alpha\mu + (1 - \alpha)\nu) = \alpha u(\mu) + (1 - \alpha)u(\nu), \forall \mu, \nu \in K, \alpha \in [0; 1]$$

*Démonstration.* En effet, pour tout  $\alpha, \beta$  dans  $[0, 1]$  on a d'une part :

$$2(D(u(\alpha\mu + \beta\nu)), D(v)) = (f, v) - Bi(\alpha\mu + \beta\nu, D(v)), \forall v \in V.$$

$$-Bi(\alpha\mu + \beta\nu, D(v)) = -Bi\alpha(\mu, D(v)) - Bi\beta(\nu, D(v))$$

$$= -\alpha \{(f, v) - 2(D(u(\mu)), D(v))\} - \beta \{(f, v) - 2(D(u(\nu)), D(v))\} \text{ Ainsi}$$

$$2(D(u(\alpha\mu + \beta\nu)), D(v)) = (f, v) - \alpha \{(f, v) - 2(D(u(\mu)), D(v))\}$$

$$- \beta \{(f, v) - 2(D(u(\nu)), D(v))\}$$

$$\iff 2(D(u(\alpha\mu + \beta\nu) - \alpha u(\mu) - \beta u(\nu)), D(v)) = (1 - \alpha - \beta)(f, v)$$

□

D'autre part, pour avoir une estimation de  $\phi(\lambda^n + \alpha G_\rho(\lambda^n))$  moins chère en terme du temps de calcul, on évalue  $\forall n \geq 1, \lambda^n + \alpha G_\rho(\lambda^n)$ . On a :

$$\lambda^n + \alpha G_\rho(\lambda^n) = \lambda^n + \alpha \left( P_K(\lambda^n + \rho^n D(u^{n+1})) - \lambda^n \right)$$

$$= (1 - \alpha)\lambda^n + \alpha P_K(\lambda^n + \rho^n D(u^{n+1}))$$

$$u(\lambda^n + \alpha G_\rho(\lambda^n)) = \alpha u((P_K(\lambda^n + \rho^n D(u^{n+1}))) + (1 - \alpha)u^{n+1})$$

$$2Bi\phi(\lambda^n + \alpha G_\rho(\lambda^n)) = |\alpha D(u(P_K(\lambda^n + \rho^n D(u^{n+1})))) + (1 - \alpha)D(u^{n+1})|^2$$

$$= \alpha^2 |D(u(P_K(\lambda^n + \rho^n D(u^{n+1}))))|^2 + 2\alpha(1 - \alpha) \left( D(u(P_K(\lambda^n + \rho^n D(u^{n+1})))), D(u^{n+1}) \right)$$

$$+ (1 - \alpha)^2 |D(u^{n+1})|^2$$

**Remarque 2.2.** Pour écrire l'algorithme dans un formalisme mathématique avec **BB** et **GLL**, il est aussi instructif d'essayer de poursuivre par une idée de simplification.

Pour cela, On pose  $\forall n \geq 1$  :

$$\bar{\lambda}^n = \lambda^n + \rho^n D(u^{n+1})$$

$$A^n = \frac{1}{Bi} |D(u(P_K(\bar{\lambda}^n)))|^2,$$

$$B^n = \frac{1}{Bi} (D(u(P_K(\bar{\lambda}^n))), D(u^{n+1})),$$

$$T^n = (D(u^{n+1}), G_\rho(\lambda^n))$$

$$\sigma_1 = 0.1, \sigma_2 = 0.9, \gamma = 10^{-4}, \varepsilon = 10^{-7} , \quad , \quad \begin{cases} \phi(\lambda^n) = \frac{1}{Bi} |D(u^{n+1})|^2 \\ \phi_{max} = \max \left\{ \phi(\lambda^{n-k}), 0 \leq k \leq \min \{n, M-1\} \right\} \end{cases} ,$$

$$\rho_{sec} = \frac{1}{Bi} \frac{|\lambda^n - \lambda^{n-1}|^2}{(\lambda^n - \lambda^{n-1}, D(u^{n+1}) - D(u^n))}, \rho_{min} = 10^{-30}, \rho_{max} = 10^{-30} \text{ et } G_\rho(\lambda^n) = P_K(\bar{\lambda}^n) - \lambda^n$$

**Proposition 2.2.** Soient  $n = 0$  et  $\lambda^0 \in K$  fixés.

Alors l'**algorithme BBPG avec GLL** s'écrit pour tout  $n \geq 1$  :

$$\left\{ \begin{array}{l} \textbf{BB} \quad \left\{ \begin{array}{l} 2(D(u^{n+1}), D(v)) + B_i(\lambda^n, D(v)) = (f, v), \quad \forall v \in V \\ G_\rho(\lambda^n) = P_K(\bar{\lambda}^n) - \lambda^n \\ \rho^n = \begin{cases} \rho_{max} & \text{si } ((\lambda^n - \lambda^{n-1}), D(u^{n+1}) - D(u^n)) \leq 0 \\ \max \{\rho_{min}, \min \{\rho_{sec}, \rho_{max}\}\} & \text{sinon} \end{cases} \end{array} \right. \\ \textbf{GLL} \quad \left\{ \begin{array}{l} \text{on cherche la suite } \alpha^n \text{ telle que :} \\ \left\{ \begin{array}{l} \alpha^{n,1} = 1 \\ \text{Si } (\alpha^{n,k})^2 A^n + 2\alpha^{n,k}(1-\alpha^{n,k})B^n + (1-\alpha^{n,k})^2 \phi_a(\lambda^n) \leq \phi_{max} - \gamma \alpha^{n,k} T^n \quad \text{Alors } \alpha^n = \alpha^{n,k} \\ \text{Sortie de la sous boucle} \\ \alpha^{n,k+1} = \begin{cases} \alpha_{temp} = \frac{1}{2}(\alpha^{n,k})^2 T^n & \text{si } \alpha_{temp} \in [\sigma_1, \sigma_2 \alpha^{n,k}] \\ \frac{\alpha^{n,k}}{2} & \text{sinon} \end{cases} \end{array} \right. \\ \lambda^{n+1} = \lambda^n + \alpha^n G_\rho(\lambda^n) \end{array} \right. \end{array} \right. \quad (2.5)$$

Soit  $\varepsilon > 0$  nous utilisons le Critère d'arrêt :  $(|z^n|_{L^2} + |\lambda^{n+1} : D(u^{n+1}) - |||D(u^{n+1})||||_{L^\infty}) < \varepsilon$  pour l'algorithme.

Cinq résidus qui doivent tendre vers zéro entrent en jeu dans la mise en place de notre algorithme :

- $|\lambda^{n+1} - \lambda^n|$
- $\|u^{n+r1} - u^n\|$
- Par la loi de comportement la relation (1.2) et la propriété  $\lambda : D(u) = |||D(u)|||$ , on a  $|\lambda^{n+1} : D(u^{n+1}) - |||D(u^{n+1})||||_{L^\infty} .$

- $|D(u^{n+1})| - |D(u^n)|$  qui mesure la convergence de la fonctionnelle  $\phi$ .
- Par l'écriture variationnelle  $(z^n, v) = 2(D(u^{n+1}), D(v)) + B_i(\lambda^{n+1}, D(v)) - (f, v)$  qui découle du problème de Bingham,  $\forall v \in H_0^1(\Omega)$ , où  $z^n = \tilde{A}u^{n+1} + Bi\tilde{B}^*\lambda^{n+1} - f$ ; ( $A$  correspond au Laplacien et  $B$  au taux de déformation). Et on utilise la norme  $|z^n|_{L^2}$  comme résidu.

En fait, si la somme de résidus :  $|z^n|_{L^2} + |\lambda^{n+1} : D(u^{n+1}) - |||D(u^{n+1})||||_{L^\infty}$  tend vers zéro, cela suffit pour avoir un critère d'arrêt pour notre algorithme.

Dans ([Grippo et al., 1986]), les conditions ont été données pour la mise en œuvre de la méthode BB pour la minimisation à l'aide d'une stratégie non monotone. Raydan a développé une méthode de convergence en 1997 en utilisant la stratégie Grippo-Lampariello-Lucidi (GLL), et la méthode BB donnée plus haut par la relation (2.3).

# EXPÉRIMENTATIONS NUMÉRIQUES

## 3.1 Approximation spatiale

Pour déterminer la solution de la formulation variationnelle discrète du problème modèle, nous considérons les espaces :

$$V_h = \left\{ w_h \in H_0^1(\Omega)^d, \operatorname{div} w_h = 0 \right\}$$
$$Q = L^2(\Omega)^{d \times d}.$$

$M_h$ , l'espace des fonctions constantes par cellules.

Nous exposons une approximation éléments finis sur un domaine annulaire  $\Omega \subset \mathbb{R}^2$  avec des rayons intérieur et extérieur donnés respectivement notés  $R_I$  et  $R_E$ ,  $R_E > R_I > 0$ .

Un maillage cartésien quadrangulaire polygonal régulier  $G_h$  définit un domaine discret  $\Omega_h$  et se rapproche de  $\Omega$ , avec des nœuds :

$$X_{i,0} = R_I e_i, X_{i+1,j} = X_{i,j} + \Delta_{r,j} e_i, X_{i,N_r} = R_E e_i$$
$$\text{avec } e_i = \frac{\tilde{e}}{\|\tilde{e}\|} \text{ et } \tilde{e} := (\cos(2\pi i/N_\theta), \sin(2\pi i/N_\theta))$$
$$\text{pour } 0 \leq i < N_\theta, \quad 0 \leq j \leq N_r$$

où la longueur  $\Delta_{r,j}$  des bords radiaux  $[X_{i,j}; X_{i,j+1}]$  (pour tout i) est incrémentalement réglée sur la longueur du bord angulaire  $[X_{i,j}; X_{i+1,j}]$ , tandis que  $N_\theta = 2^l$ , avec un  $6 \leq l \leq 9$  prescrit dans tous les résultats ultérieurs. La valeur de  $N_r$  est telle que  $\|X_{i,N_r-1}\| < R_E$  et  $\|X_{i,N_r} - X_{i,N_r-1}\| > \|X_{i,N_r} - X_{i+1,N_r}\|$  de manière à éviter les cellules de grille éventuellement plates.

Comme d'habitude dans les méthodes d'éléments finis,  $h$  désigne le plus grand diamètre des cellules quadrilatérales  $[X_{i,j} X_{i+1,j} X_{i+1,j+1} X_{i,j+1}]$ , et est utilisé pour indexer toutes les entités discrètes. Ici, la discréétisation du problème à l'aide d'une méthode d'éléments finis, consiste à définir des espaces fonctionnels de dimensions finies  $X_h$  et  $M_h$  proches respectivement des espaces  $V$  et  $Q$  de dimensions infinies. Cette méthode est utilisée pour calculer la solution approchée  $(u_h, \lambda_h)$  de  $(u, \lambda)$ .

Le domaine discret qui approche  $\Omega$  correspondant est  $\Omega_h$  et on note  $M_{h,0} = M_h \cap L_0^2(\Omega)$ .

On approche le champ  $u$  comme un vecteur continu de  $4Q_1$  par  $u_h$ , i.e  $u_h \in X_h$  qui est l'espace vectoriel de champ linéaire bidimensionnelle continu .

Le champ de pression  $p$  est approché par  $p_h$  sur  $Q_0$ .

Soit l'ensemble convexe  $K$ , on considère l'ensemble discret :

$$K_h = \left\{ \mu_h \in M_h^{2 \times 2}; |||\mu_h||| \leq 1 \text{ sur } \Omega_h \right\}$$

ainsi que  $P_h(\mu_h) = \frac{\mu_h}{\max(1, |||\mu_h|||)}$ ,  $\forall \mu_h \in M_h^{2 \times 2}$

Le problème discrétisé de Bingham correspondant peut alors s'écrire :

trouver  $(u_{h,0}, p_h, \lambda_h) \in X_{h,0} \times M_{h,0} \times K_h$  tel que

$$\begin{cases} 2(D(u_{h,0}), D(v_h)) + Bi(\lambda_h, D(v_h)) - (p_h, \operatorname{div} v_h) = (f_h, v_h), \forall v_h \in X_{h,0} \\ (\mu_h - \lambda_h, D(u_{h,0})) \leq 0, \mu_h \in K_h \end{cases}, \quad (3.1)$$

Puisque  $v_h \in V_h$  alors  $\forall \mu_h \in M_h^{2 \times 2}$ , la solution  $u_{h,0}(\mu_h) \in V_h$  de la formulation variationnelle discrète du problème modèle (ici problème de couette) recherchée est définie comme étant l'unique solution de :

$$2(D(u_{h,0}), D(v_h)) + Bi(\lambda_h, D(v_h)) = (f_h, v_h), \forall v_h \in V_h \quad (3.2)$$

Le problème de minimisation associé est donné par :

$$\lambda_h = \operatorname{argmin}_{\mu_h \in K_h} \phi_h(\mu_h) := \frac{1}{Bi} |D(u_{h,0}(\mu_h))|^2 \quad (3.3)$$

Ainsi par un raisonnement analogue comme décrit dans la section (2), il existe  $(u_{h,0}, \lambda_h) \in X_{h,0} \times M_{h,0}$  pour le problème (3.1) avec  $u_{h,0}$  unique et cette solution peut être calculer en utilisant une minimisation algorithmique pour  $\phi_h$ .

L'analyse de convergence des approximations par éléments finis du problème de Bingham est étudiée depuis les années 70. Une analyse générale faite dans les pionniers travaux [Glowinski, 1976] et [Glowinski, ], et plus loin dans [Han and Reddy, 1995] fournit une estimation sous-optimale  $\|u - u_h\| = O(\sqrt{h})$  avec des méthodes du premier ordre, et les recherches actuelles visent à prouver ces travaux afin de se conformer à des expériences numériques quasi-optimales. Pour cela, la plupart des auteurs se concentrent sur le problème de Mosolov, défini comme la courbure rotationnelle dans un tuyau droit à section constante sous un gradient de pression. Dans cet écoulement simplifié, la vitesse est supposée être un champ scalaire parallèle à l'axe du tuyau, tandis que la principale difficulté de non-linéarité persiste. A noter également que la limitation de régularité de la solution de Bingham justifie l'utilisation d'approximations d'ordre inférieur dans la littérature, dans la mesure où l'adaptation du maillage n'est pas utilisée (voir [Roquet et al., 2000]

et [Gustafsson and Lederer, 2022] pour une analyse d'ordre supérieur). Dans ce contexte, les résultats  $O(h)$  optimaux sont d'abord obtenus dans [Falk and Mercier, 1977], en utilisant une reformulation des modèles où de la vitesse est la variable principale approchée par un champ constant par morceaux. Une estimation optimale est prouvée dans [Carstensen et al., 2016] pour une méthode non linéaire par morceaux conforme avec une nouvelle méthode discrète à trois champs appropriée pour la formulation, et les résultats sont étendus aux écoulements 3D. Dans [Gustafsson and Lederer, 2022], la formulation (3.1) pour le problème de Mosolov est considéré, avec une approche abstraite conforme. Une condition de stabilité LBB associant vitesse et contrainte des espaces discrètes conduit alors à  $\|u - u_h\| + \|div(\lambda - \lambda_h)\|_{H^{-1}(\Omega)} = O(h)$  a priori estimer sous l'hypothèse de régularité  $u \in H^2(\Omega)$  et  $div \lambda \in L^2(\Omega)^2$ , où  $u$  et  $\lambda$  sont des composantes non nulles respectivement de  $u$  et  $\lambda$  dans la problème.

## 3.2 Algorithme en dimension finie

Dans cette section, nous abordons la question de  $div D(v)$  et de  $\Delta v$ .

Soit à considérer une fonctionnelle sur l'espace discret  $V_h$

En fait, lorsqu'on travaille dans un espace discret, on ne dit pas que  $div v_h = 0$  mais plutôt que  $(div v_h, q_h = 0), \forall q_h \in M_h$  où  $M_h$  est l'espace des fonctions constantes par cellules. Ce qui n'est pas équivalent à  $div v_h = 0$ . Il faut une inclusion d'espace d'abord. Or ( $div v_h$  ne coïncide pas avec  $M_h$ ). Par exemple si :

$$\begin{aligned}\phi_h(\mu_h) &= \frac{1}{Bi} |D(u_h(\mu_h))|^2 \text{ avec} \\ 2(D(u_h), D(v_h)) &= (f, v_h) - Bi(D(v_h), u_h) - (div p_h, div v_h) \quad \forall v_h \in M_h \\ \text{et on impose } (div v_h, q_h = 0), \forall q_h \in M_h.\end{aligned}$$

La conséquence est que  $\phi'_h$  ne sera pas la même que celle qu'on connaissait si on utilise un solveur de l'opérateur laplacien pour l'approximation de l'opérateur  $A$ . Un tel choix est en effet parfois très utile pour simplifier les coûts de programmations et de calcul.

En fait,

$$\phi'_h(\mu_h) = -D(u_h(\mu_h)) + \text{Un terme} \tag{3.4}$$

C'est un peu cher en prix. Mais puisque  $(div v_h, q_h = 0)$  par imposition, on néglige le second terme qui s'est rajouté. Certes, c'est pas anodin. Il a des éléments finis pour lesquels  $div v_h = 0$  mais pas tous. Ainsi dans un élément fini dont  $div v_h \neq 0$ , la dérivée de la fonctionnelle n'est pas

le même. Et quand on fait même l'itération de gradient projeté :  $\lambda^{n+1} = P_K(\lambda^n - \rho\phi'(\lambda^n))$ , on se pose la question du terme qui se rajoute à  $\phi'$ .

**Remarque 3.1.** En fait,  $2\operatorname{div} D(v) = \Delta v$  avec le problème de Stokes :  $2(D(v), D(w)) = (\nabla v, \nabla w) - (\operatorname{div} v, \operatorname{div} w)$ . Mais ça, c'est au niveau continu. Lorsqu'on discrétise, l'égalité n'est plus vérifiée. Il est facile de montrer que c'est pas vrai en 2D. On travail dans un espace à divergence nulle ( $V_h$ ). Mais quand on écrit l'algorithme de gradient projeté, on fait une minimisation d'une fonctionnelle et on garde  $D$  (le taux de déformation) partout dans la théorie de Bingham. Nous avons un solveur algébrique qui est rapide parce que quand on fait les éléments finis, il n'est plus vrai que  $\operatorname{div} D(v) = \Delta v$ .

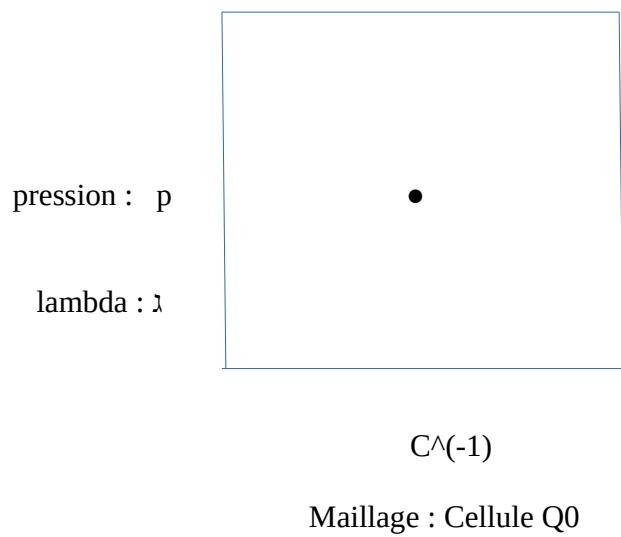
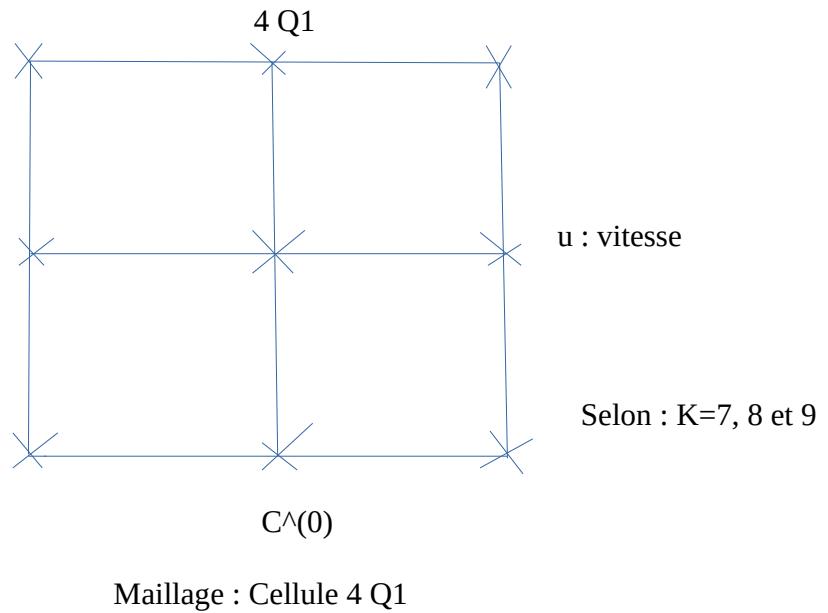
- **Une différence :**

On définit un certain  $\theta_h$  qui vérifie l'équation variationnelle dans  $M_h$ .  $\theta_h$  est la projection  $L^2$  de  $d_h(u_h)$  sur  $M_h$  :  $\theta_h = d_h^{-1}d_h(u_h)$  ie  $(\theta_h - d_h(u_h), q_h) = 0, \forall q_h \in M_h \iff \theta_h = P_K(d_h(u_h))$ .

Si on choisit  $u_h$  bilinéaire par cellule sur un maillage qui est raffiné deux fois, on calcule une quantité qui est constante par cellule. Si on dérive  $d_h(u_h)$ , on obtient un élément de  $M_h$ . Il faut donc calculer sa projection. Donc en gros, l'algorithme variationnellement et mine de rien ce n'est plus exactement la même chose.

### 3.3 Résultats numériques

Le Maillage de notre domaine annulaire  $\Omega \subset \mathbb{R}^2$  avec l'approximation par éléments finis considéré est se présente comme suit :



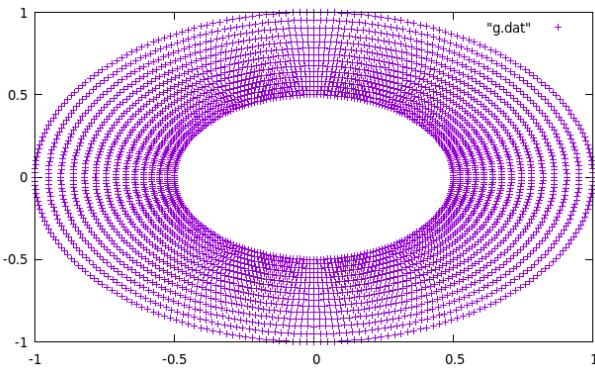
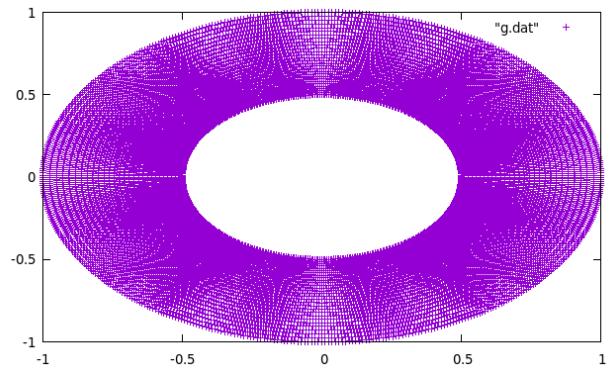
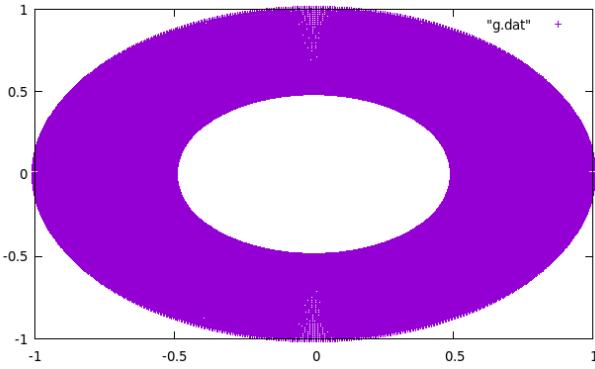
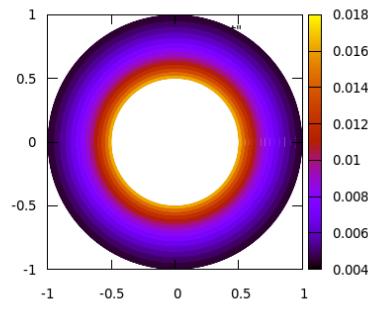
FIGURE 3.1 – Maillage pour  $k=7$ FIGURE 3.2 – Maillage pour  $k=8$ FIGURE 3.3 – Maillage pour  $k=9$ 

FIGURE 3.4 – Domaine annulaire approché

Dans cette section, nous testons l’algorithme **SPG** (**BBPG**). Nous simulons l’écoulement d’un fluide viscoplastique deux cylindres co-axiaux en rotation, de rayons (resp. intérieur et extérieur)  $R = r_1$  et  $r = 1$ , et de vitesses angulaires  $\omega_1$  et  $\omega = 1$ . Nous utilisons les conditions de Dirichlet homogène. Le programme de Couette (Fortran) utilisé a permis d’avoir un temps de calcul optimal. La force de volume  $f$  du second membre qui met le fluide en mouvement est bien choisie. C’est aussi dans ce programme qu’on fait le choix de  $Bi$ , du maillage, et la tolérance pour contrôler le résidu itéré. Il a été appliqué dans tous les tests effectués. Nous avons fixé un nombre maximal d’itérations  $N_{max} = 10000$ .

Nous présentons les similarités entre le **BBPG** et **UZAWA** avec différents cas tests. Il est représenté dans chaque image le résidu itéré en fonction du nombre d’itérations  $Nit$ . Dès lors que la tolérance est atteint ici  $tol = 10^{-10}$ , on arrête.

Afin de chercher le  $\rho_{optim}$  optimal, on prend  $\rho$  échelonné  $\rho_{max} = \frac{l}{Bi}$ ,  $l \in \{1;4\}$ . Les valeurs  $l = 1$  et  $l = 2$  apparaissent optima car demandent moins d’itérations avant d’atteindre  $tol$ .  $l = 3$  a tendance à demander un nombre important d’itérations avant d’atteindre le  $tol$  et ceci pour un

maillage grossier ou fin.  $l = 4$  demande le nombre maximal d'itération pour satisfaire le critère de la tolérance.  $l = 2$  est demande moins d'itération que  $l = 1$  alors on retient  $\rho_{optim} = \frac{2}{Bi}$ .

La valeur de  $\rho$  de **BB** optimal noté  $\rho_{optim}$  qui est  $\frac{2}{Bi}$  comparé celui de **UZAWA** qui est  $\frac{4}{Bi}$  selon le  $Bi$  choisi.

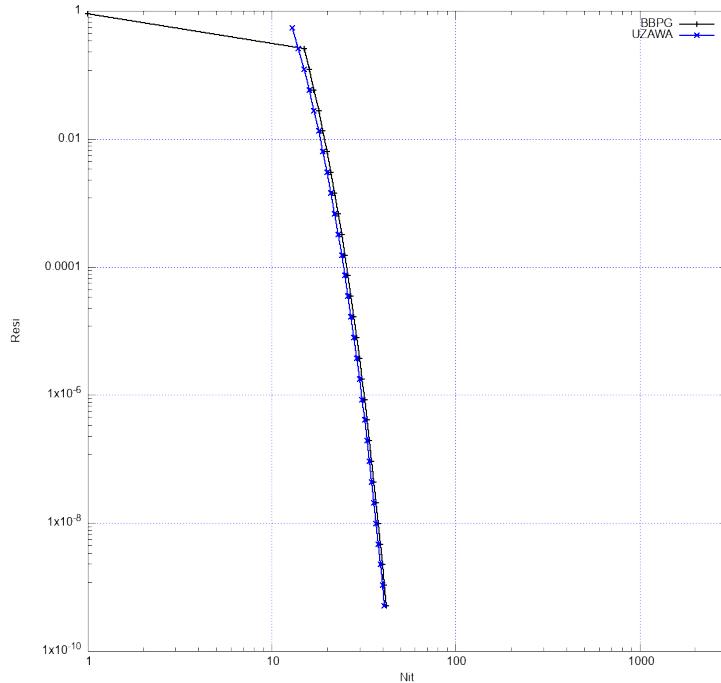


FIGURE 3.5 – BBPG vs UZAWA à  $Bi=10$  et  $k=7$

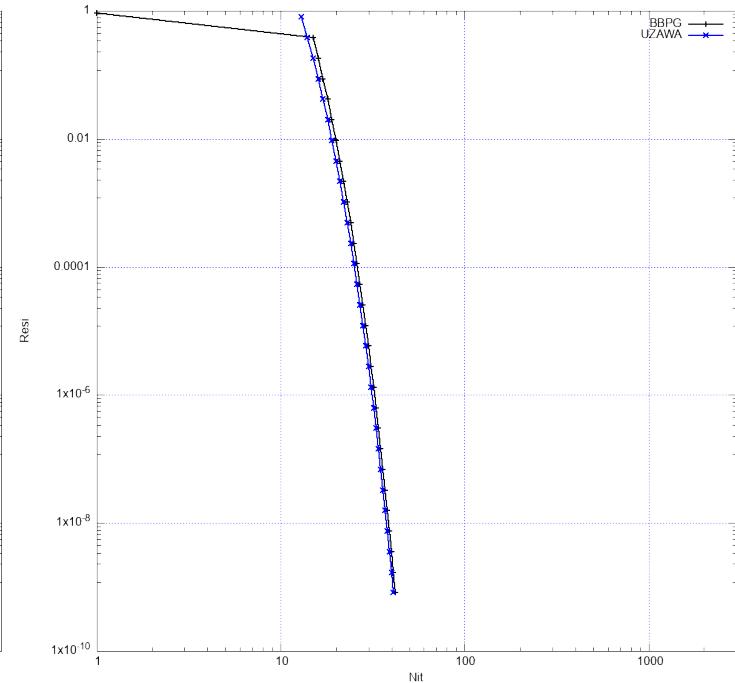


FIGURE 3.6 – BBPG vs UZAWA à  $Bi=10$  et  $k=8$

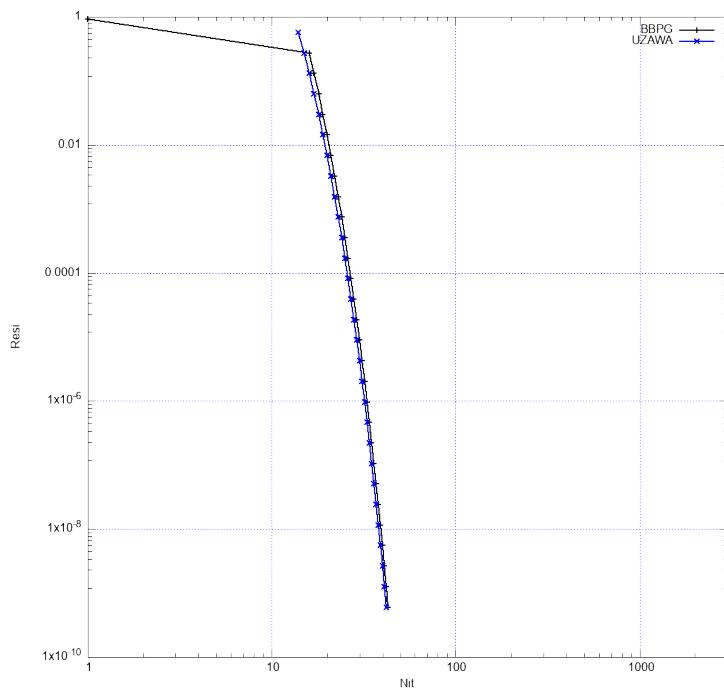


FIGURE 3.7 – BBPG vs UZAWA à  $Bi=10$  et  $k=9$

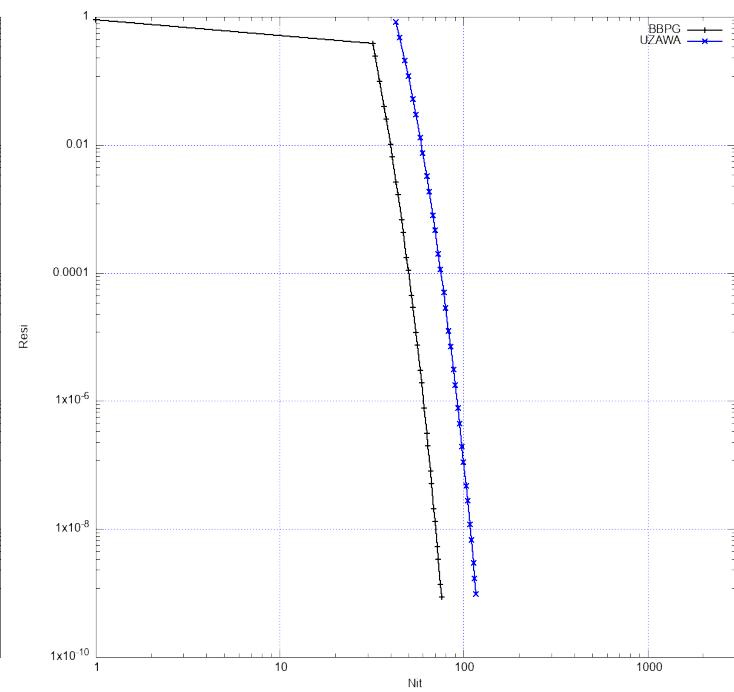


FIGURE 3.8 – BBPG vs UZAWA à  $Bi=50$  et  $k=7$

Dans les graphiques (3.5) à (3.22), nous présentons les résultats de l'algorithme **BBPG** comparé à **UZAWA**. **BBPG** présente un comportement à quelques différences près de **UZAWA** (un gain léger par rapport à **UZAWA**). Le résidu décroît au fur et à mesure que le nombre d'itérations et augmente. **BBPG** demande moins d'itérations avant d'atteindre la tolérance. Selon le maillage ( $k = 7, 8$  et  $9$ ) et la valeur de  $Bi$  prisent, **BBPG** prend une avance sur **UZAWA**. Dans tout les cas, au niveau des deux algorithmes, le résidu a tendance à descendre légèrement et progressivement au cours de l'itération jusqu'à de la valeur  $Nit_{max}$  où la tolérance est obtenue. Cependant, dans le cas où  $Bi = 10$  au niveau duquel les deux résidus atteignent le tol aux mêmes  $Nit_{max}$ . A partir de  $Bi = 50$ ,  $Nit_{max} \simeq Bi$  une fois que la valeur de tol est atteint. Tout ceci se réalise dans les mêmes conditions au niveau des deux algorithmes à l'exception des pics observés au niveau du **BBPG** qui a contribué au gain obtenu. Il paraît que  $Bi$  semble ne pas intervenir vraiment dans l'efficacité (convergence rapide de **BBPG**) du résidu mais plutôt le calcul automatique de  $\rho$  de **BB** et le maillage choisi. On pourrait donc de ce fait supposer que ces pics de **BBPG** sont proviennent du calcul automatique de  $\rho$ . Les deux algorithmes demandent pratiquement les mêmes nombre d'itération avant d'atteindre la tolérance souhaité. Quelque soit la valeur de  $Bi = 50, 100, 200 \dots$ ; le résidu prend son départ au voisinage du point  $(1,1)$  au niveau de **BBPG** alors que le départ d'**UZAWA** dépend du nombre de Bingham  $Bi$  choisi (en fait, *resi* prend

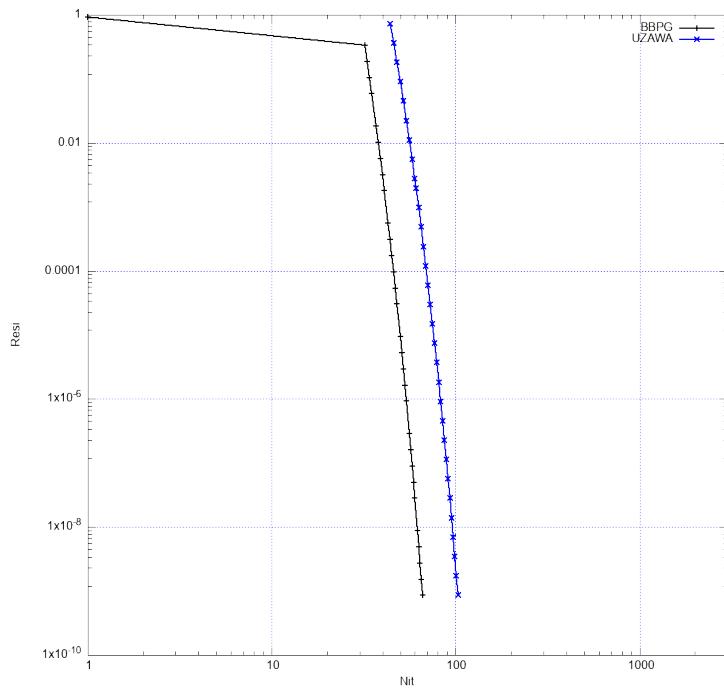


FIGURE 3.9 – BBPG vs UZAWA à  $Bi=50$  et  $k=8$

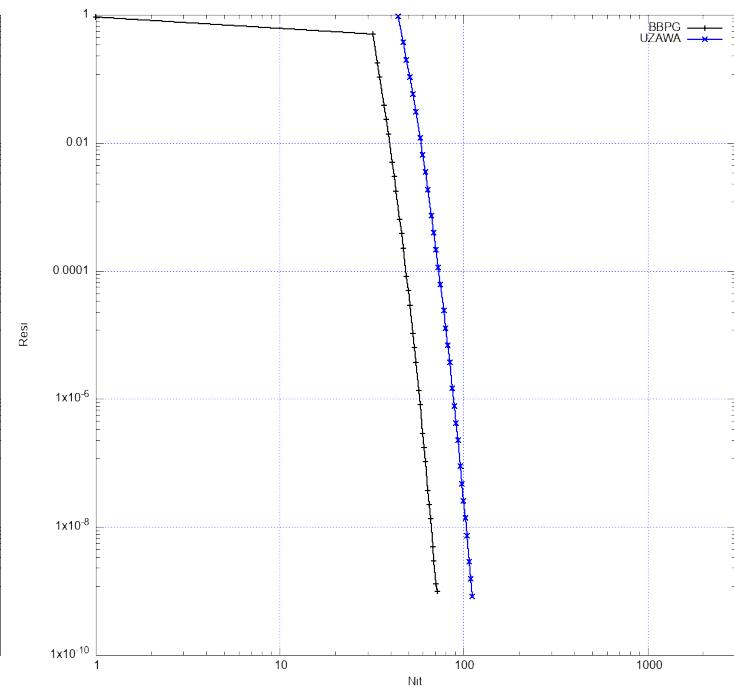


FIGURE 3.10 – BBPG vs UZAWA à  $Bi=50$  et  $k=9$

son départ de  $Nit_{max}(\text{BBPG}) + \beta$  selon le maillage choisi ). Cet écart  $\beta$  mesuré a permis de garder un rythme parallèle conforme entre les deux résidus itérés. Par contre le cas  $Bi = 10$  semble violé cette condition de rythme conforme. Ceci peut être relié aux pics observés au niveau de **BBPG**. Ce qui a permis la validation de l'algorithme **BBPG** du modèle numérique est l'ordre de convergence  $O(h)$  d'approximation par éléments finis qui est bien vérifié pour les deux erreurs calculées pour les grandeurs  $10^{-8}$  et  $10^{-10}$  qui est pratiquement le même que l'algorithme d'**UZAWA**.

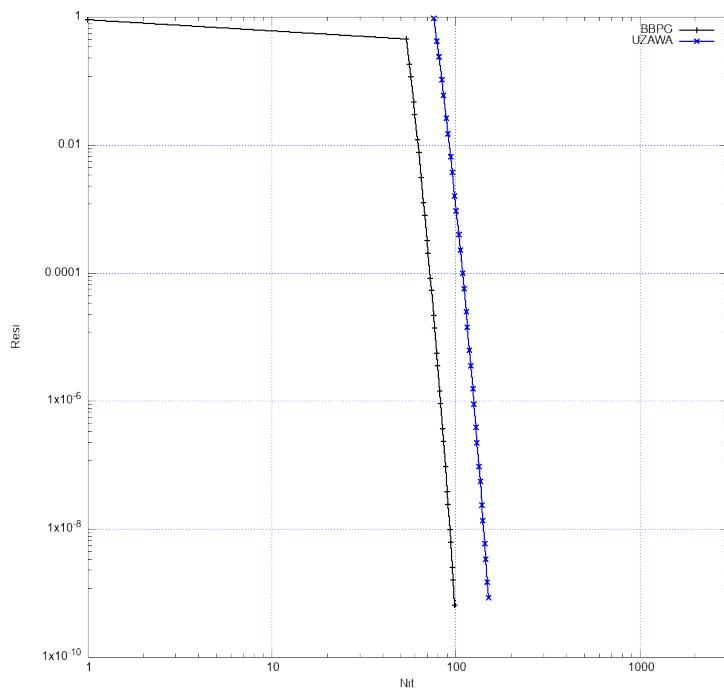


FIGURE 3.11 – BBPG vs UZAWA à  $Bi=100$  et  $k=7$

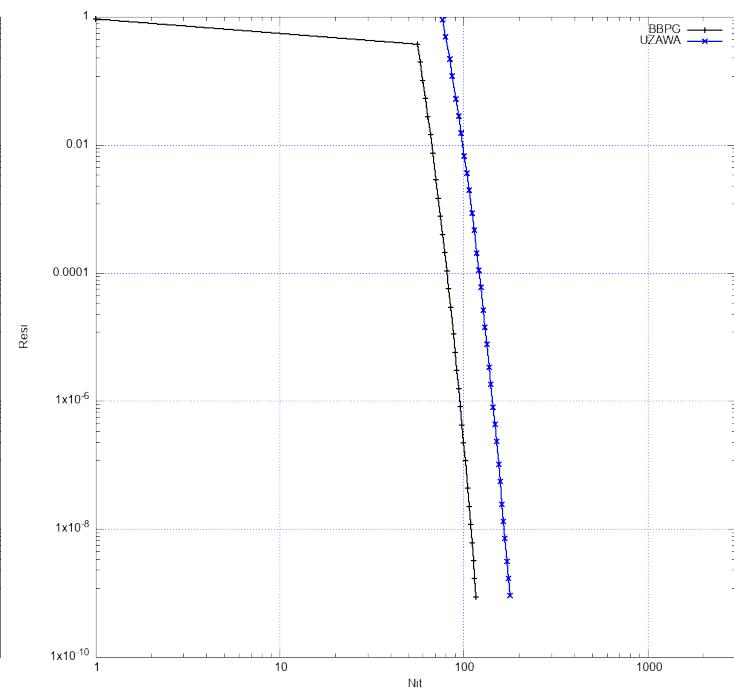


FIGURE 3.12 – BBPG vs UZAWA à  $Bi=100$  et  $k=8$

Il faut dire qu'avec le calcul de  $\rho_{sec}$  qui devrait être l'équivalent de  $\frac{2}{Bi}$  pour  $\rho_{quad}$  avec **UZAWA** a permis d'améliorer la situation puisqu'on gagne au moins 30% de performance sur l'accélération de la vitesse de convergence de l'algorithme **BBPG** comparé à **UZAWA**. Le choix du paramètre  $M$  pour la détermination de  $\alpha$  itéré pour **GLL** a aussi participé à cette remarquable amélioration observée car dès lors que ce fameux paramètre passe à 100, 200, 300, ... la situation n'est plus améliorée (le résidu itéré se stagne et s'explose quelque soit l'itération).

Pour bien voir comment la valeur  $\frac{2}{Bi}$  demeure l'optimale pour **BBPG**, nous avons de représenté pour différentes valeurs de  $\rho$  ( $\rho_1 = \frac{2}{Bi}$ ,  $\rho_2 = \frac{3}{Bi}$  et  $\rho_3 = \frac{4}{Bi}$ ) et pour  $Bi = 10, 50$  et  $100$  pour les trois maillages  $k = 7, 8$  et  $9$ .

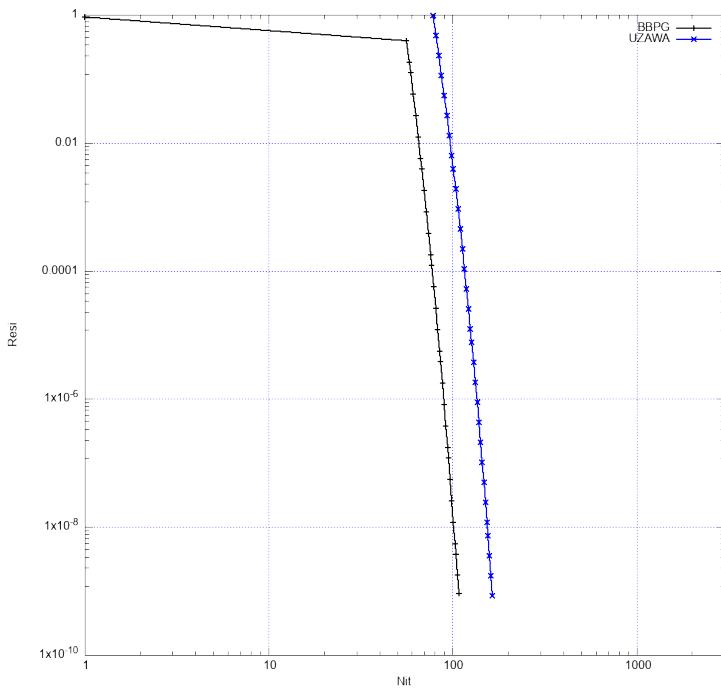


FIGURE 3.13 – BBPG vs UZAWA à  $Bi=100$  et  
 $k=9$

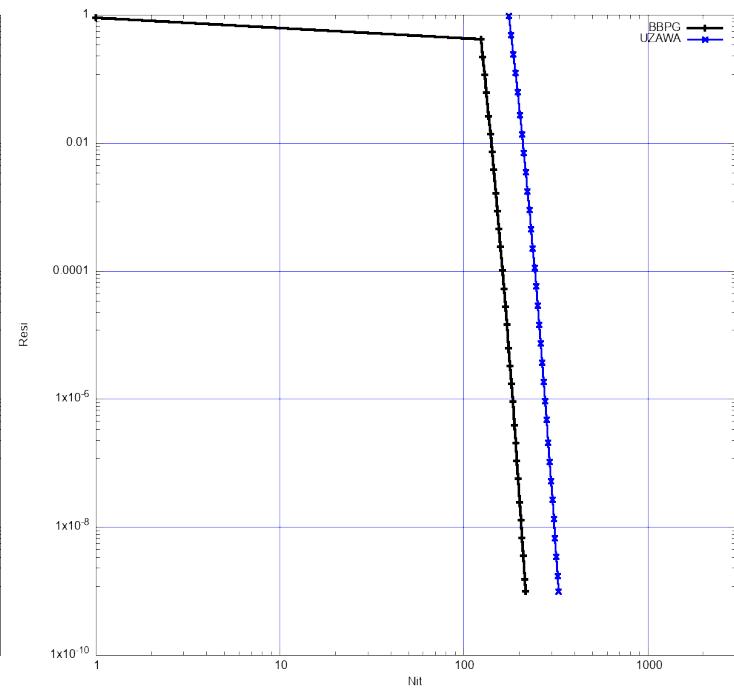


FIGURE 3.14 – BBPG vs UZAWA à  $Bi=250$  et  
 $k=7$

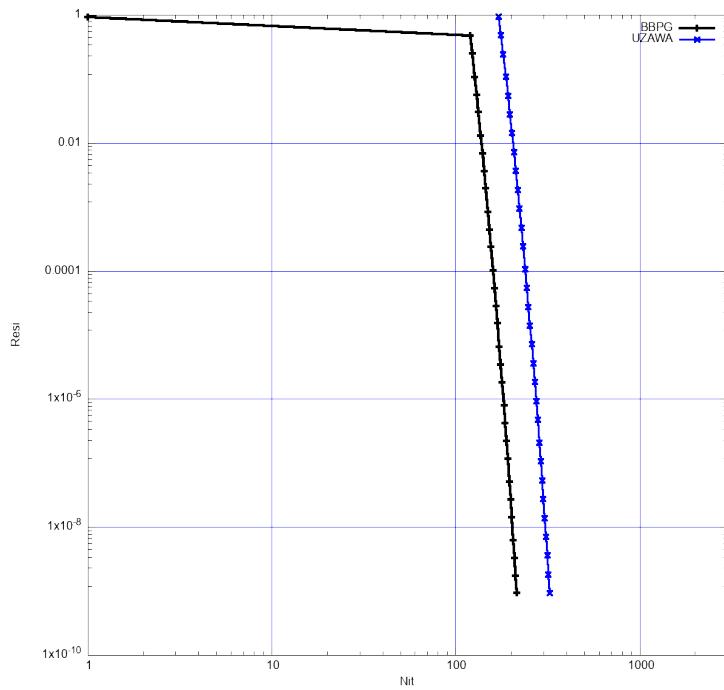


FIGURE 3.15 – BBPG vs UZAWA à  $Bi=250$  et  $k=8$

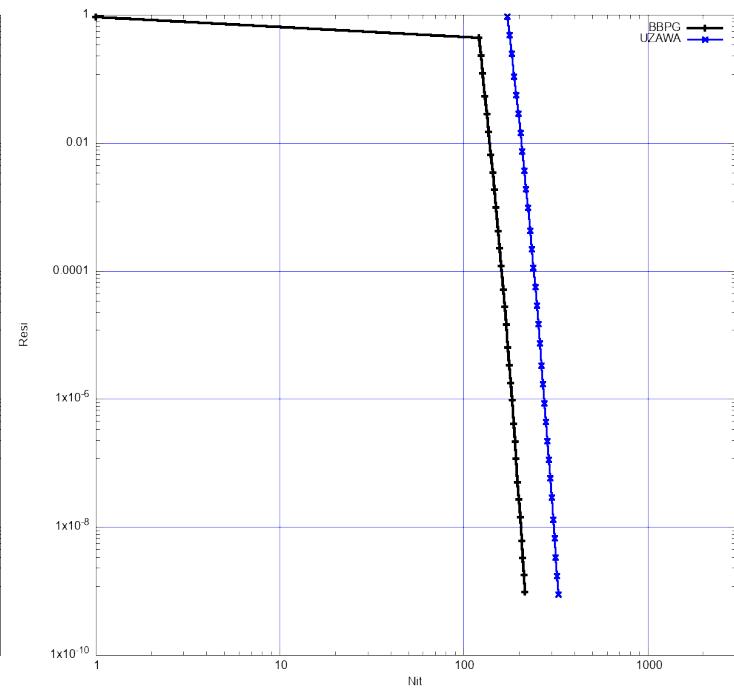


FIGURE 3.16 – BBPG vs UZAWA à  $Bi=250$  et  $k=9$

Le temps de calcul a aussi confirmé l'efficacité de SPG comparé à UZAWA. Comme le montre les quatre différents tableaux ci-dessous. Les deux premiers révèlent que pour  $Bi = 10$  (nombre de Bingham petit), on gagne déjà en temps de calcul et les deux derniers tableaux ont continué sur la même lancée pour  $Bi = 100$  (nombre de Bingham élevé).

$Bi=10$ différents maillages	Temps de calcul CPU	
SPG	K=7	$Average = 1.092 \cdot 10^{-2}$ et $Total = 4.588 \cdot 10^{-1}$
	K=8	$Average = 4.518 \cdot 10^{-2}$ et $Total = 1.898 \cdot 10^0$
SPG	K=9	$Average = 2.118 \cdot 10^{-1}$ et $Total = 9.107 \cdot 10^0$

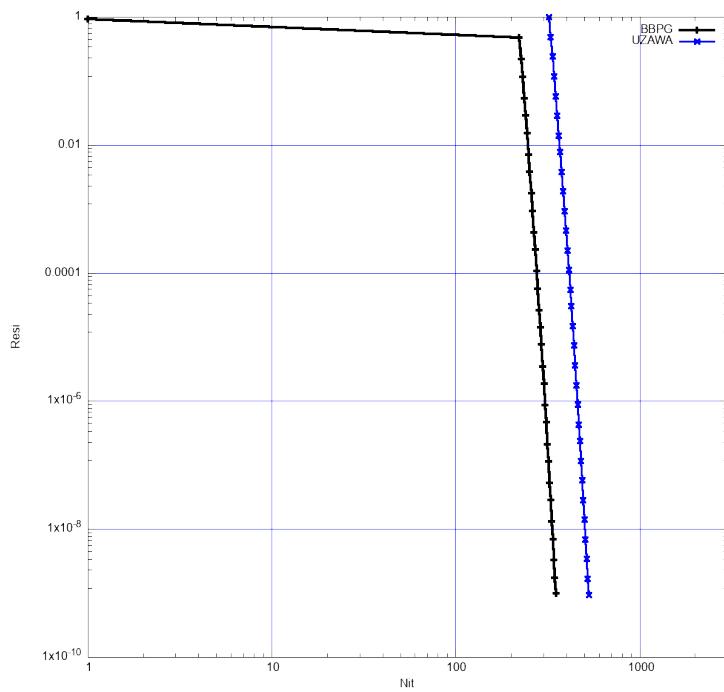


FIGURE 3.17 – BBPG vs UZAWA à  $Bi=500$  et  
 $k=7$

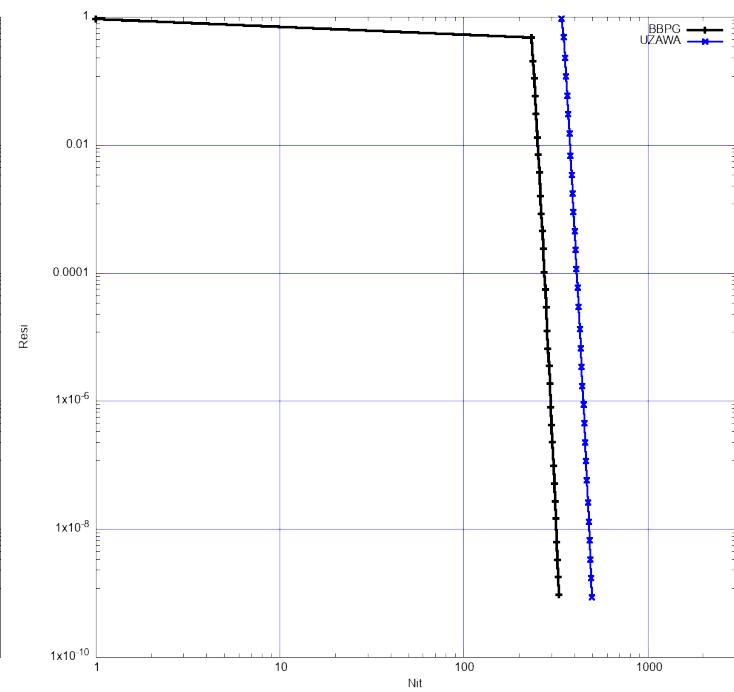


FIGURE 3.18 – BBPG vs UZAWA à  $Bi=500$  et  
 $k=8$

Bi=10 différents maillages		Temps de calcul CPU	
GP	K=7	<i>Average</i> = $1.345.10^{-2}$	et <i>Total</i> = $5.515.10^{-1}$
GP	K=8	<i>Average</i> = $5.970.10^{-2}$	et <i>Total</i> = $2.448.10^0$
GP	K=9	<i>Average</i> = $2.923.10^{-1}$	et <i>Total</i> = $1.128.10^1$

Bi=100 différents maillages		Temps de calcul CPU	
SPG	K=7	<i>Average</i> = $1.334.10^{-2}$	et <i>Total</i> = $2.014.10^0$
SPG	K=8	<i>Average</i> = $5.962.10^{-2}$	et <i>Total</i> = $1.061.10^1$
SPG	K=9	<i>Average</i> = $2.935.10^{-1}$	et <i>Total</i> = $4.843.10^1$

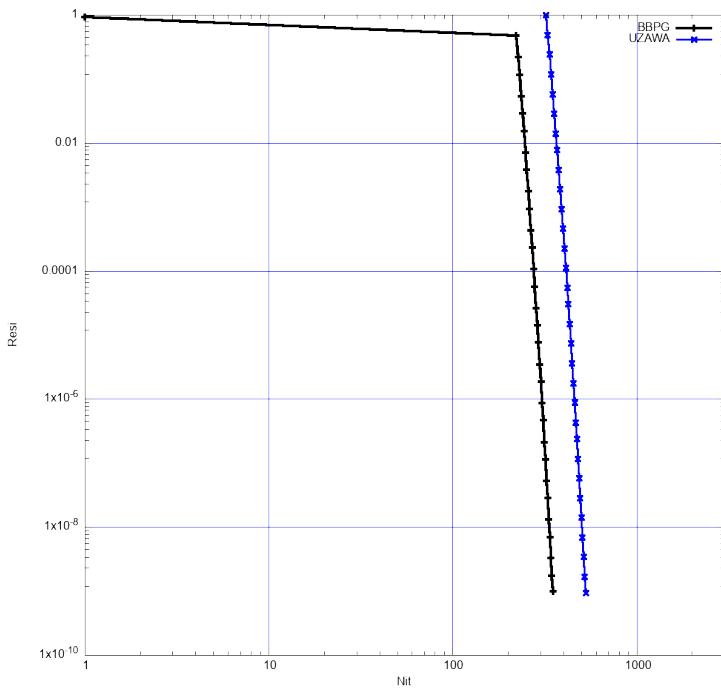


FIGURE 3.19 – BBPG vs UZAWA à  $Bi=500$  et  
 $k=9$

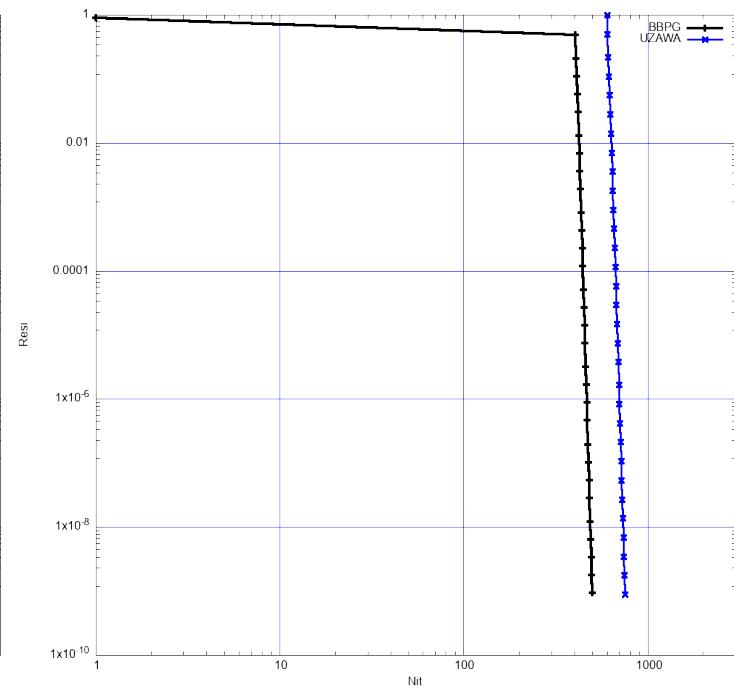


FIGURE 3.20 – BBPG vs UZAWA à  $Bi=750$  et  
 $k=7$

Bi=100 différents maillages	Temps de calcul CPU	
GP	K=7	$Average = 3.297.10^{-2}$ et $Total = 3.264.10^0$
GP	K=8	$Average = 1.470.10^{-1}$ et $Total = 1.719.10^1$
GP	K=9	$Average = 7.399.10^{-1}$ et $Total = 7.990.10^1$

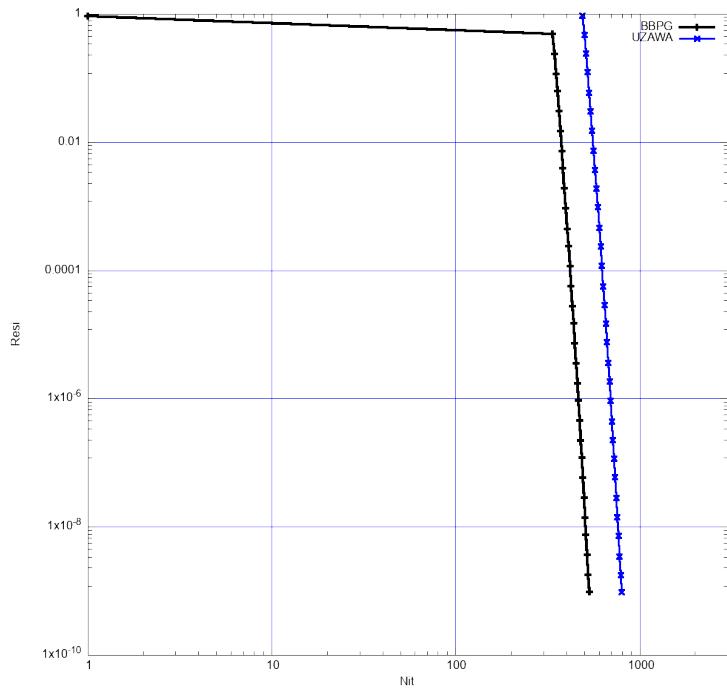


FIGURE 3.21 – BBPG vs UZAWA à  $Bi=750$  et  $k=8$

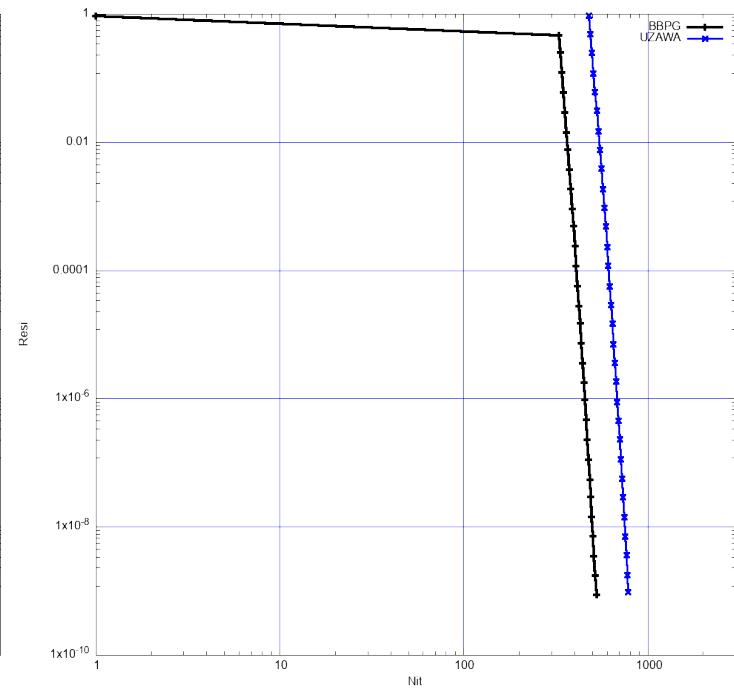


FIGURE 3.22 – BBPG vs UZAWA à  $Bi=750$  et  $k=9$

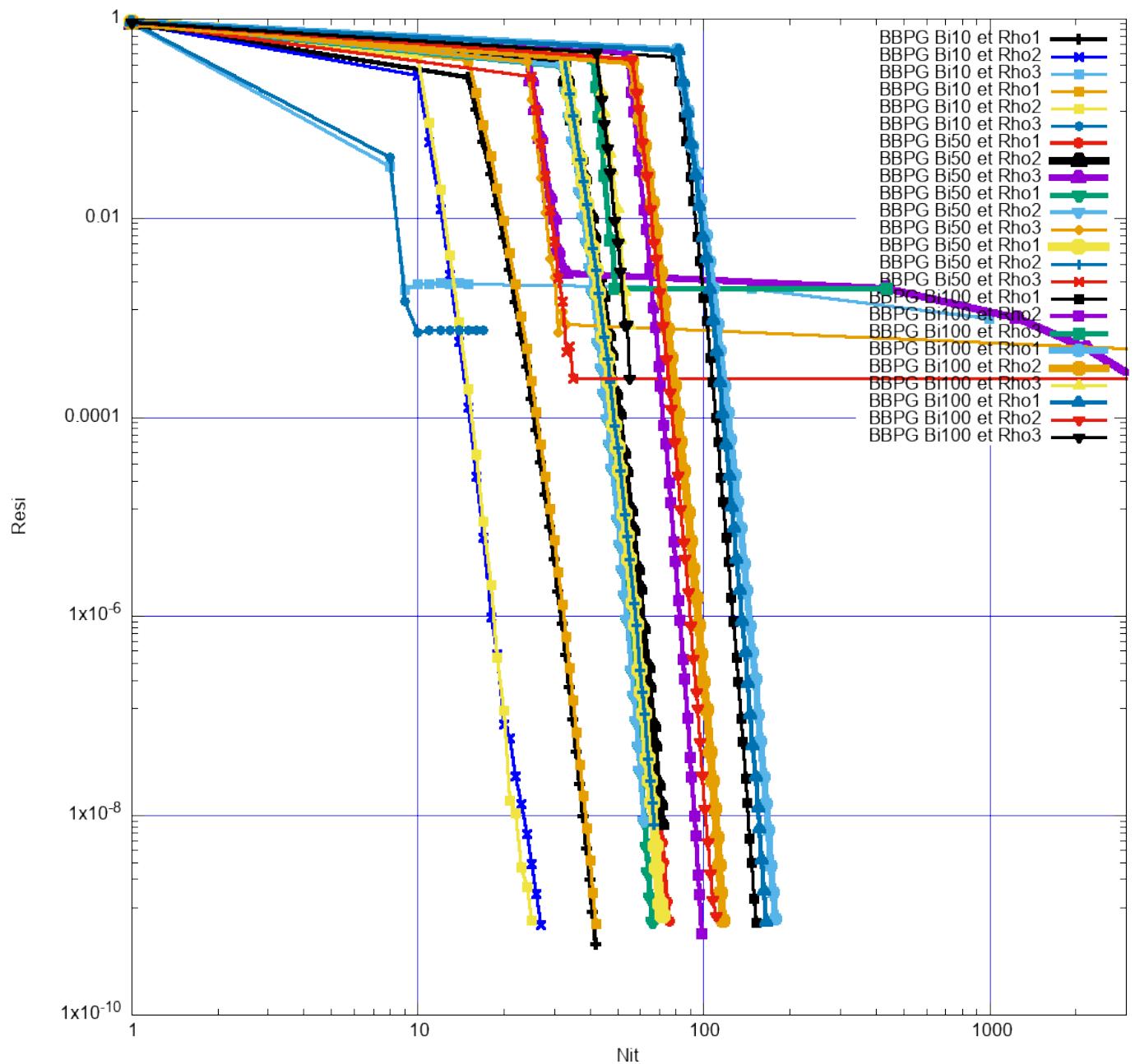


FIGURE 3.23 – BBPG et  $\rho_{echelon}$  à  $\text{Bi}=10, 50, 100$  puis  $k=7, 8$  et  $9$

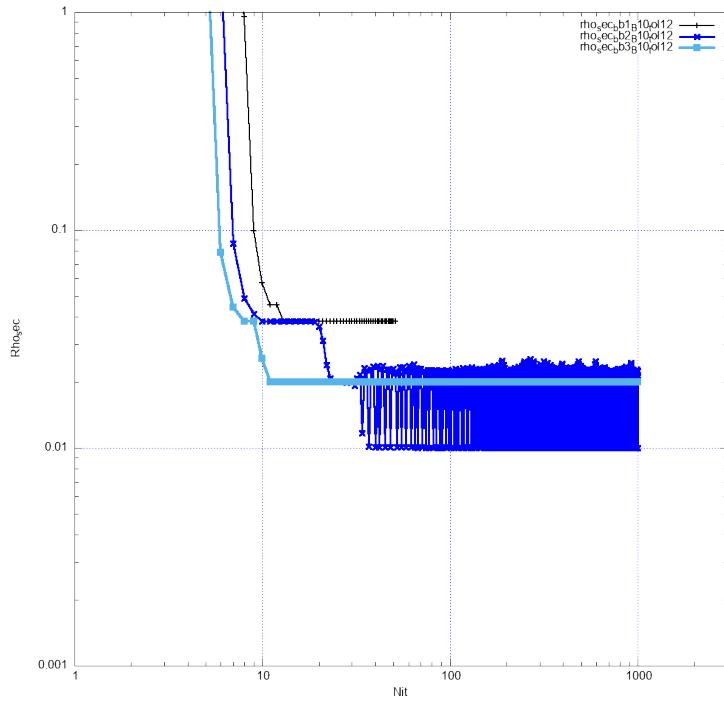


FIGURE 3.24 – **BBPG** –  $\rho_{sec}$  à  $Bi=10$  et  $tol = 10^{-12}$

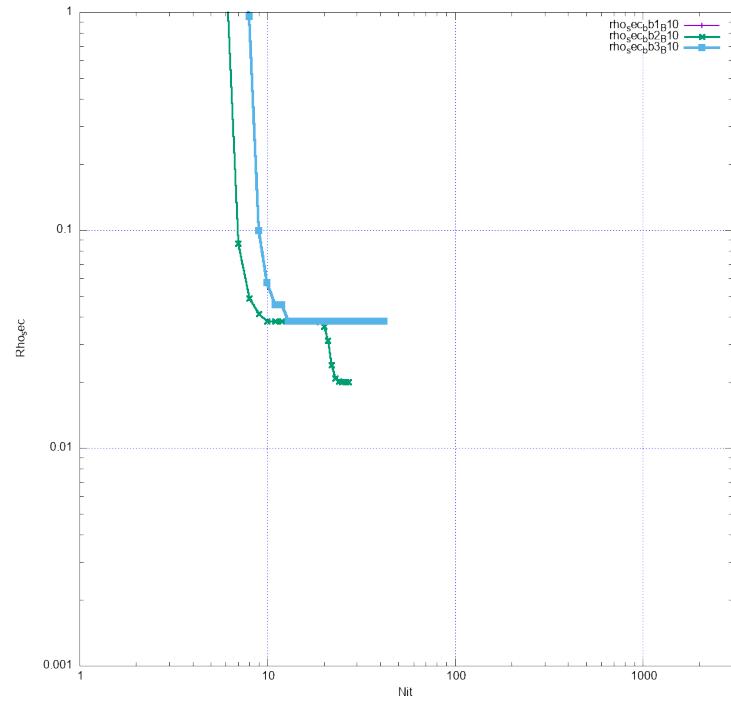


FIGURE 3.25 – **BBPG** –  $\rho_{sec}$  à  $Bi=10$  et  $tol = 10^{-9}$

Afin de mieux justifier la raison pour laquelle, l'idée de faire un **BBPG** pour Bingham est importante, nous avons tracé le comportement de  $\rho_{sec}$  en fonction de nombre d'itérations.

Nous avons fait des tests pour le choix de deux tolérances  $tol = 10^{-9}$  et  $tol = 10^{-12}$  pour une bonne précision.

Les graphes ( figures 3.20 à 3.20 ) présent les différents caractéristiques de crucial  $\rho_{sec}$  sur qui

repose le calcul de **Barzilai Borwein** et qui introduit **GLL**. Sur ces graphes, il représenter l'évolution de  $\rho_{sec}$  itéré en fonction du nombre d'itérations et on arrête avec la tolérance. Les figures du coté gauche sont tracées avec  $tol = 10^{-12}$  tandis que celles du droit avec  $tol = 10^{-9}$ .

Tout comme le résidu itéré,  $\rho_{sec}$  itéré décroît au cours du nombre d'itération. On constate que  $\rho_{sec}$  présente de petits paliers au fur et à mesure que Nit augmente et il a tendance à diminuer. Les tests ont été réalisés pour des valeurs de  $Bi = 10, 100, 250$  et  $750$  pour bien propager toutes les variantes liées à  $\rho_{sec}$ . Le constat général fait est que, plus  $Bi$  augmente, plus  $\rho_{sec}$  plus d'itérations quelque soit le maillage choisi.

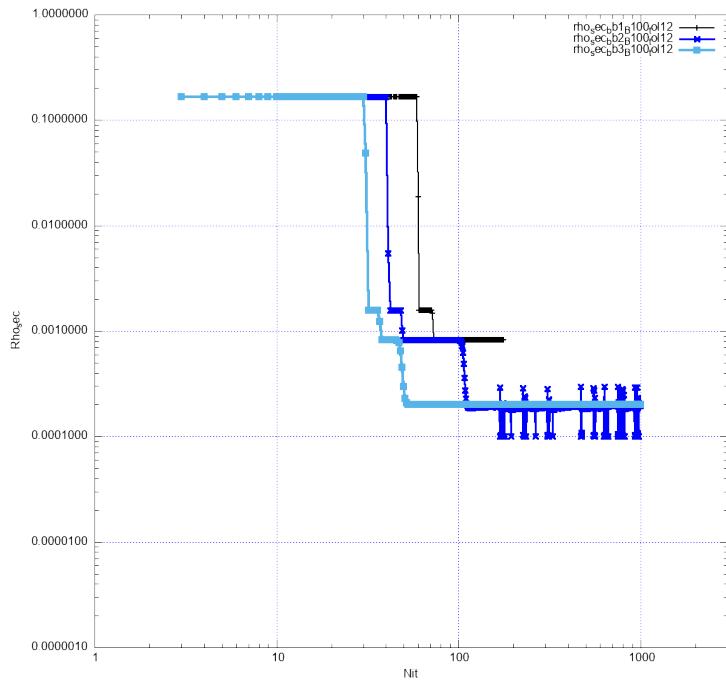


FIGURE 3.26 – *BBPG* –  $\rho_{sec}$  à  $Bi=100$  et  
 $tol = 10^{-12}$

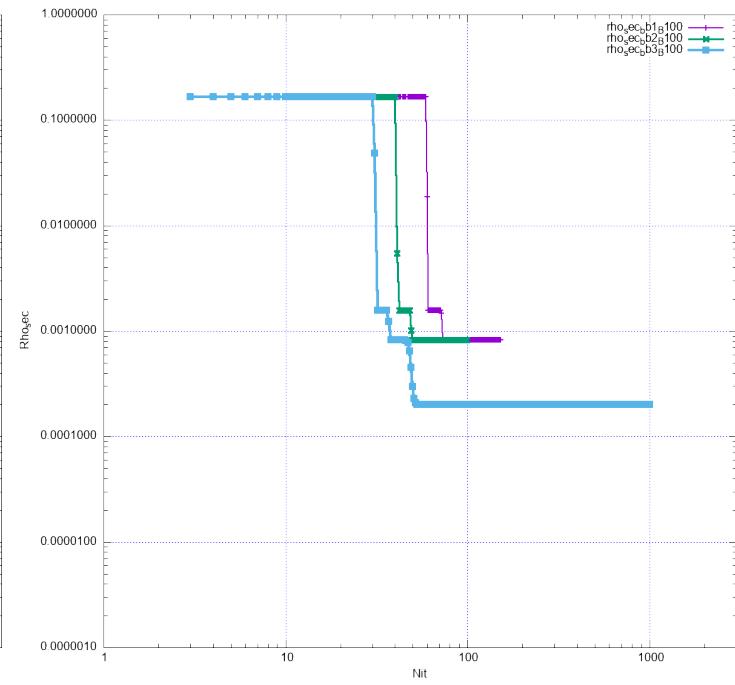


FIGURE 3.27 – *BBPG* –  $\rho_{sec}$  à  $Bi=100$  et  
 $tol = 10^{-9}$

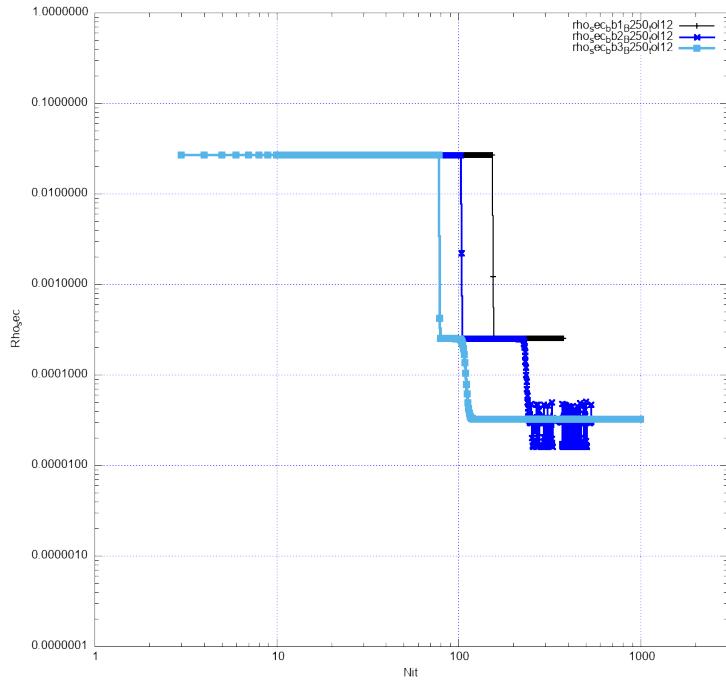


FIGURE 3.28 – *BBPG* –  $\rho_{sec}$  à  $Bi=250$  et  
 $tol = 10^{-12}$

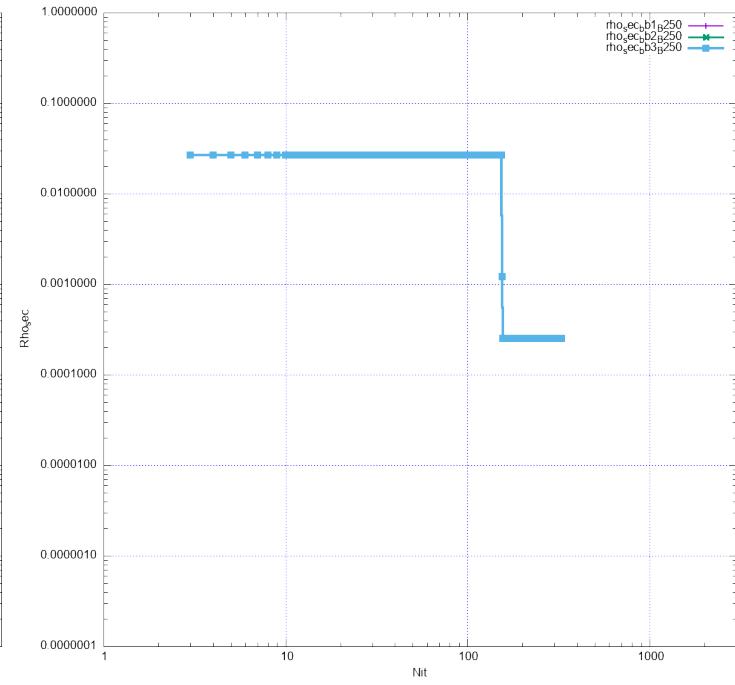


FIGURE 3.29 – *BBPG* –  $\rho_{sec}$  à  $Bi=250$  et  
 $tol = 10^{-9}$

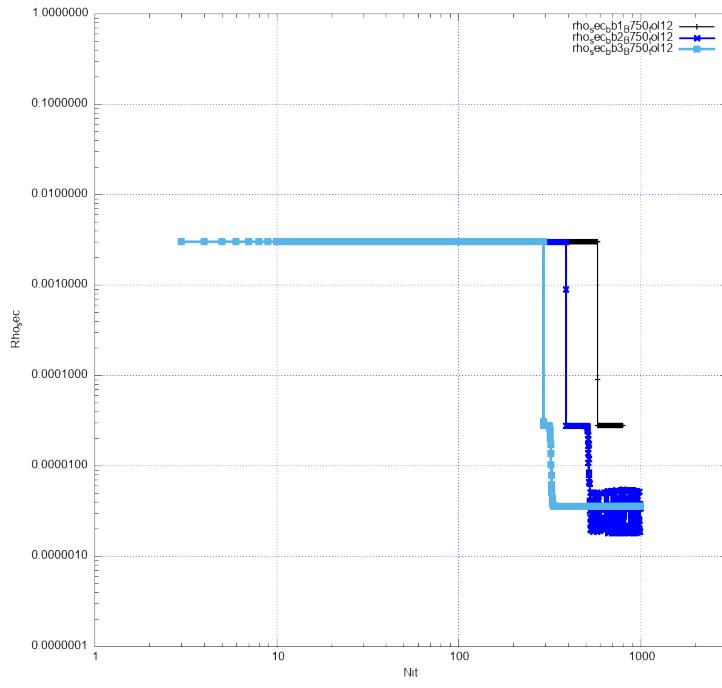


FIGURE 3.30 – **BBPG** –  $\rho_{sec}$  à  $Bi=750$  et  $tol = 10^{-9}$

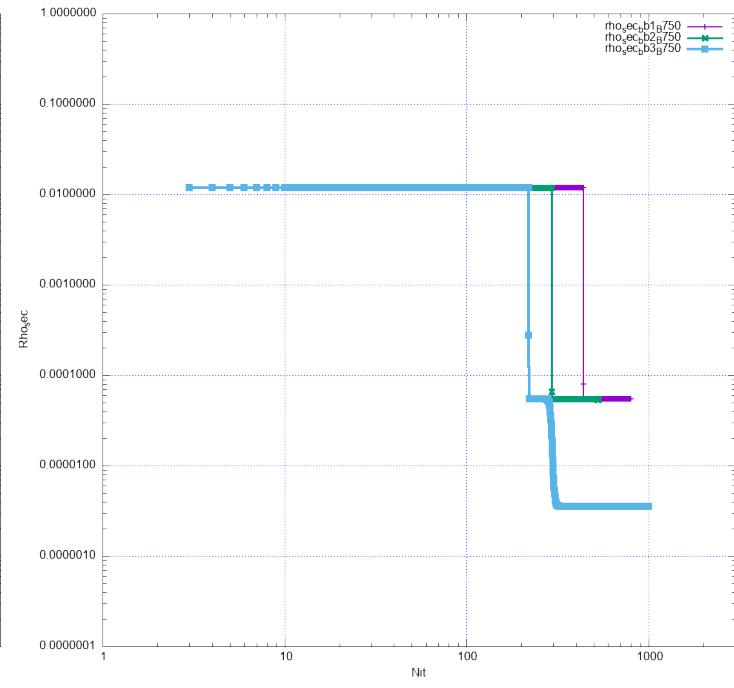


FIGURE 3.31 – **BBPG** –  $\rho_{sec}$  à  $Bi=750$  et  $tol = 10^{-9}$

Manifestement avec  $tol = 10^{-9}$ , pour  $Bi = 100$ ,  $\rho_{sec}$  demande 328; pour  $Bi = 250$ ,  $\rho_{sec}$  demande 152  $Bi = 750$ ,  $\rho_{sec}$  demande 797 itérations. Il s'est aussi avéré que les valeurs initiales  $\rho_{max} = \frac{2}{Bi}$ ,  $\frac{3}{Bi}$  et  $\frac{4}{Bi}$  ont participé à l'augmentation des itérations, donc au gain obtenu.  $\rho_{max} = \frac{4}{Bi}$  fait passer pour  $Bi = 100$ , 750 à 1000 itérations. Mais dans le cas de  $Bi = 250$ , les itérations ne dépassent pas 328 quelque soit  $\rho_{max}$ .

Pour  $tol = 10^{-12}$  on observe le même comportement de paliers avec  $\rho_{max} = \frac{2}{Bi}$  et  $\frac{4}{Bi}$  mais pour  $\rho_{max} = \frac{3}{Bi}$ , à la fin des paliers,  $\rho_{sec}$  présente des cisaillements avec un même rythme.

On ne peut pas comparer  $\rho_{sec}$  à une valeur constante d'**UZAWA**. Ces résultats montrent l'intérêt de faire **BBPG**. On gagne 30 pour 100 et on a ce gain parce qu'au cours de Nit, il faut passer d'un palier à un autre avec des courbes décroissantes de paliers. L'efficacité de **BBPG** comparé à **UZAWA** est ainsi basé sur ce fameux  $\rho_{sec}$  qui change la situation.

Les figures 3.28 et 3.29 présentent l'ensemble des  $\rho_{sec}$  représentés pour les tolérances respectivement  $tol = 10^{-9}$  et  $tol = 10^{-12}$

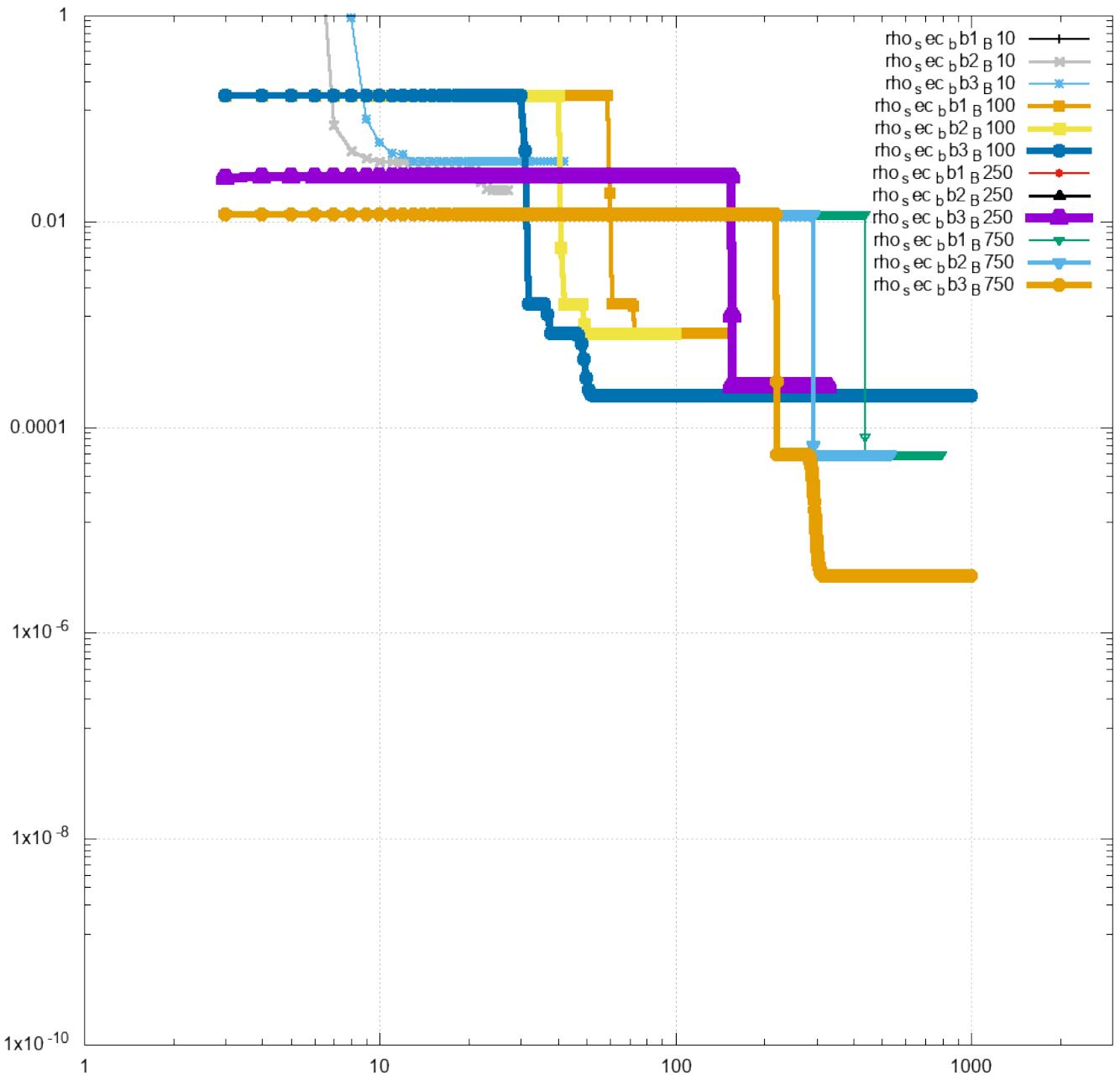


FIGURE 3.32 – BBPG et  $\rho_{echelon}$  à  $Bi=10, 50, 100$  puis  $k=7, 8, 9$  et  $tol = 10^{-9}$

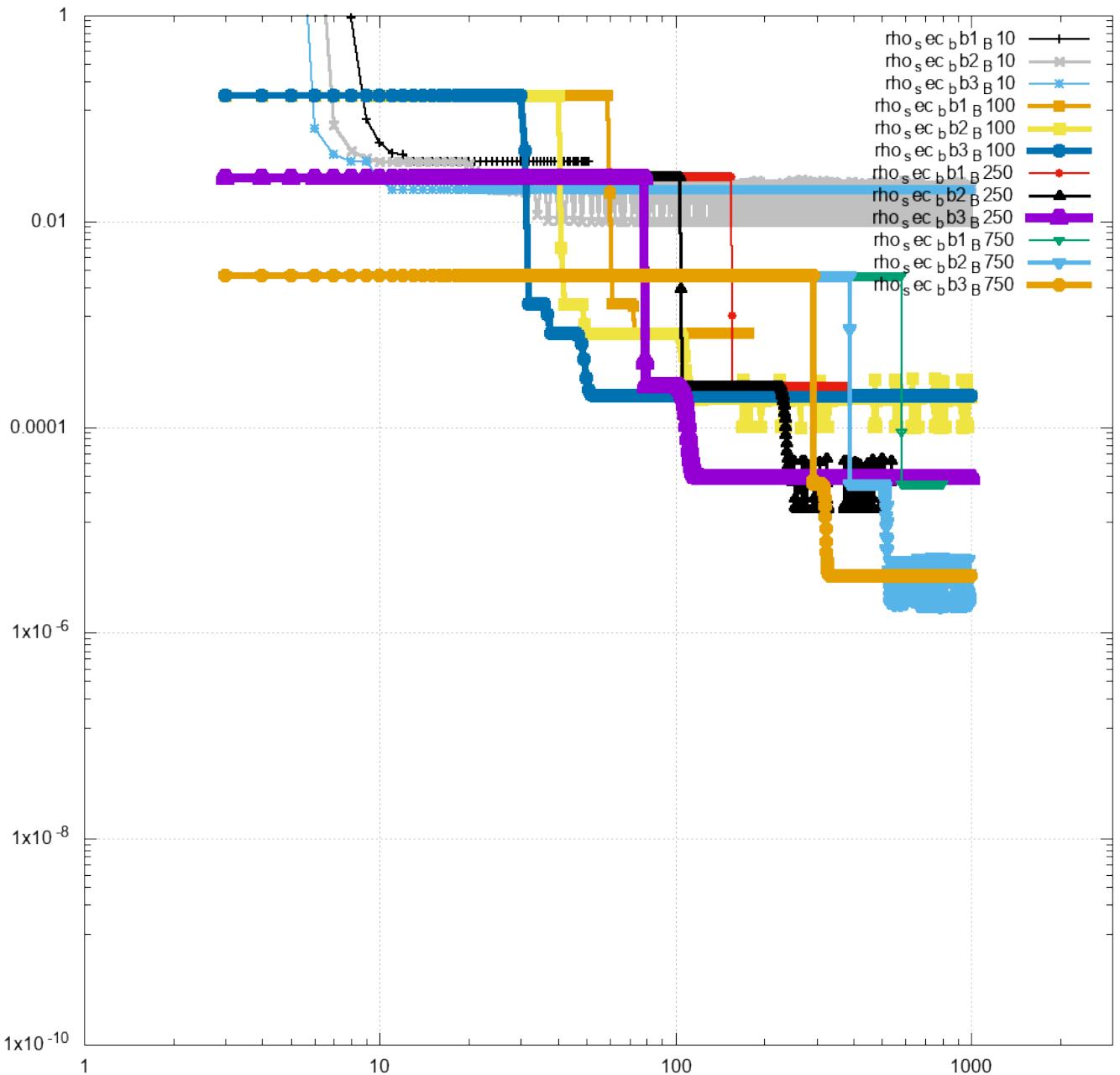
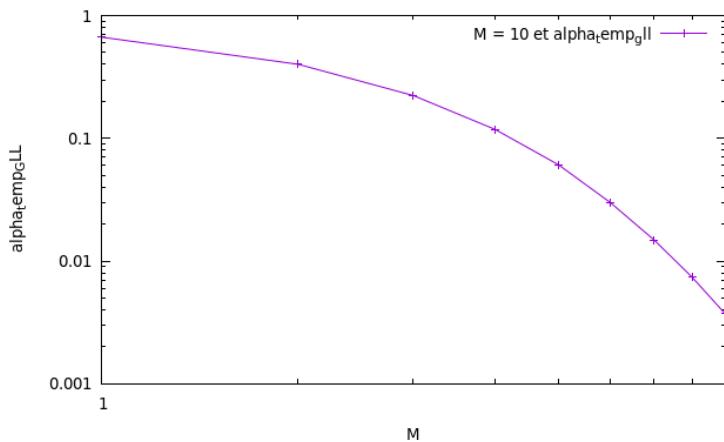
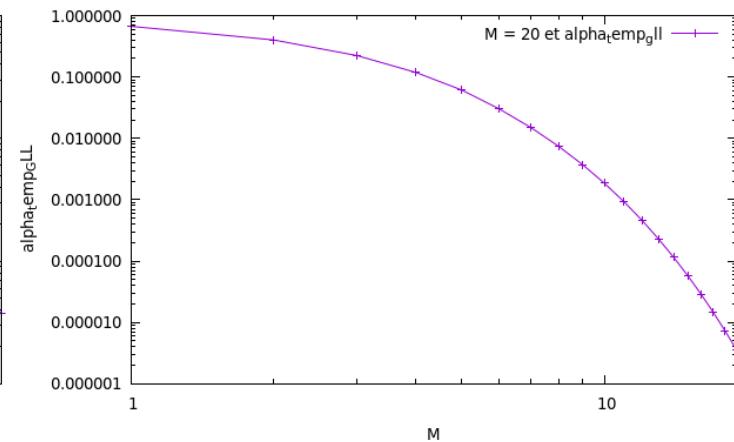


FIGURE 3.33 – BBPG et  $\rho_{echelon}$  à  $Bi=10, 50, 100$  puis  $k=7, 8, 9$  et  $tol = 10^{-12}$

Nous avons aussi représenté une caractéristique du pas de descente  $\alpha^k$  de **GLL** en fonction de l'évolution du paramètre  $M$ . En fait, c'est le  $\alpha_{temp}$  temporaire qui gouverne cette évolution de  $\alpha^k$ . Comme on l'espérait,  $\alpha^k$  de **GLL** garde au cours d'une certaines itérations ( jusqu'à  $\frac{M}{2}$  itération ), la valeur 1. Ce dernier décroît au cours des restes d'itérations et devient asymptotiquement nul. Ce qui est normal, car cette décroissance a permis à  $\alpha^k$  de garder la condition de relaxation. Il faut aussi noter qu'il est important de bien le choisir le paramètre  $M$  pour permettre à ce fameux  $\alpha$  de maintenir le rythme de décroissance. C'est d'ailleurs pour cette raison que les auteurs l'appellent paramètre de relaxation de la condition de monotonie. Comme le montre les figures (3.30) à (3.35), lorsque ce paramètre devient grand  $\alpha$  devient asymptotiquement nul, ce qui n'améliore plus la situation dans ce cas. Les auteurs ont de ce fait raison de pourvoir mettre en exergue ce paramètre  $M$ .

Le calcul de  $\alpha$  avec **GLL** révèle un gain optimal, car  $\alpha$  présente une monotonie comme espéré, et cela permet de valider l'implémentation l'algorithme au niveau **GLL**. Cet maintient de condition de monotonie de  $\alpha$  est due aux réglages des valeurs de  $\sigma_1, \sigma_2$  et  $\gamma$  proposées par **GLL** car une fois ses valeurs ne sont pas respectées (bien réglées),  $\alpha$  ne respecte plus la condition de monotonie. Cette condition de relation de la monotonie est avérée nécessaire pour avoir une convergence.

C'est par conséquent là, la nécessité d'accompagner le calcul automatique de  $\rho^k$  de **BB** avec le calcul de  $\alpha^k$  de **GLL**. Il s'est avérer aussi que pour  $M = 1, 2$  et  $3$   $\alpha^k$  ne respecte pas la condition de relaxation de la monotonie. Par contre pour  $4 \leq M \leq 50$  Cette condition de relaxation de la monotonie est respectée et tout baigne bien. Mais dès lors que  $M$  devient grand ( $M \geq 10^x$  avec  $x \geq 2$ ) la condition de relation de la monotonie est de nouveau violée.

FIGURE 3.34 – BBPG et  $\alpha^k$  à  $M = 10$ FIGURE 3.35 – BBPG et  $\alpha^k$  à  $M = 20$

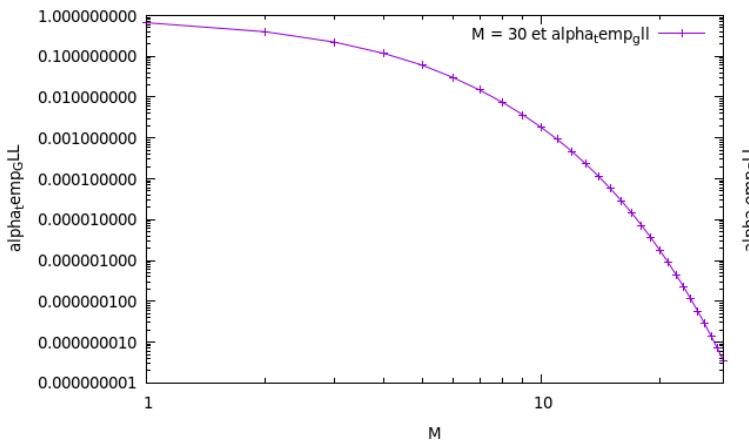


FIGURE 3.36 – BBPG et  $\alpha^k$  à  $M = 30$

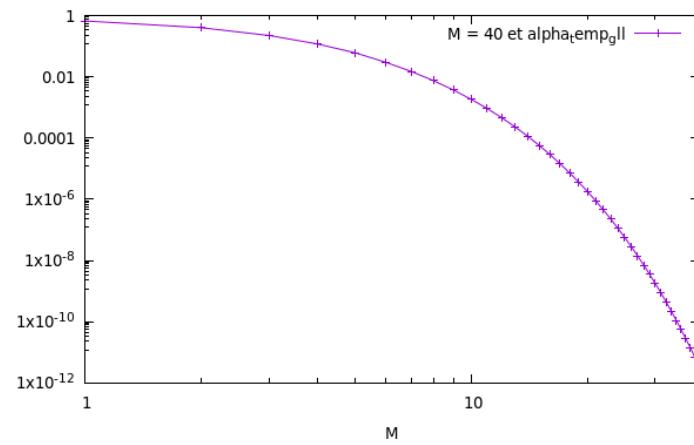


FIGURE 3.37 – BBPG et  $\alpha^k$  à  $M = 40$

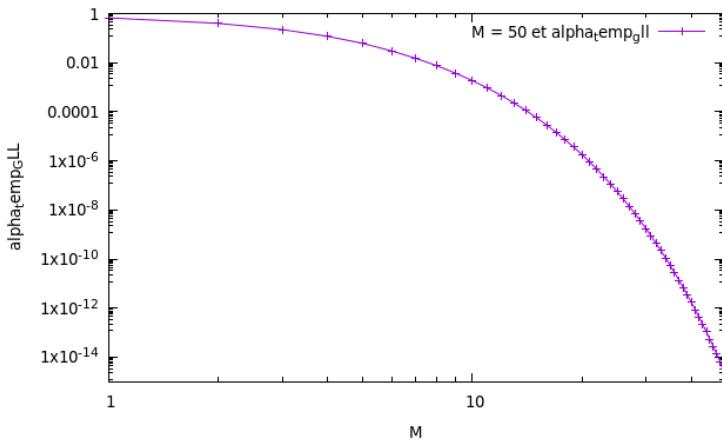


FIGURE 3.38 – BBPG et  $\alpha^k$  à  $M = 50$

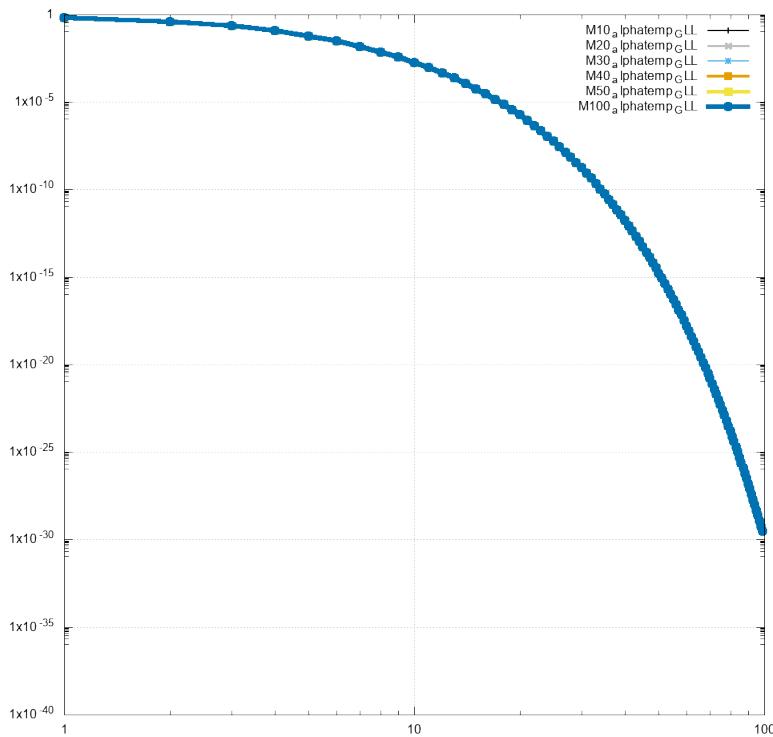


FIGURE 3.39 – BBPG et  $\alpha^k$  à  $M = 100$

# CONCLUSION ET PERSPECTIVES

Dans ce stage, nous avons mis en place une nouvelle approche algorithmique (SPG) plus performante que celle d'UZAWA/ISTA classique pour le modèle de Bingham . En effet, cette amélioration observée est due aux idées de Barzilai-Borwein (BB) et Grippo-Lempori-Lucidi qui n'a jamais été fait, à notre connaissance, par les auteurs qui travaillent sur le problème de Bingham. J'ai programmé l'algorithme pour un cas simple du problème de Couette. Il est aussi donné l'idée générale d'établissement de la convergence de pour l'algorithme.

Un gain d'amélioration de BBPG est remarquable (30%) a été obtenu comparé à UZAWA. Et c'est ce qui a permis de valider l'efficacité de BBPG.

Nous projetons de faire des tests pour le BBPG avec la pénalisation en limitant le  $\rho_{sec}$  de Barzilai Borwein dans la zone rigide avec le problème de Couette. Nous prévoyons également :

- d'une part, de faire des tests avec  $\rho_{R^k} / \rho_{F^k}$  de BBPG et le comparer à la version pénalisé à pas constant ( $\rho = cts$ ) d'UZAWA qu'a fait Benoît Senard,
- d'autre part, faire les mêmes testes mais cette fois ci avec le fameux  $\phi$  de la relation (3.4) avec le terme de surplus et regarder son influence.

## .1 Annexe

## .2 Algorithme du gradient projeté localement pondéré par pénalisation avec BB et GLL (LPWPG)

### .2.1 Algorithme du gradient projeté localement pondéré (LWPG)

- Benoît Sénard dans son travail, a expliqué la raison pour laquelle qu'au lieu d'utiliser  $D(u) = 0$ , on utilise plutôt  $|||2D(u) + Bi\lambda||| \leq Bi$  ce qui permet d'écrire une forme améliorée de l'algorithme précédent avec le calcul de  $\rho^n$  en tenant compte de la zone rigide.

L'algorithme qu'on utilise à cette étape dit du gradient projet localement pondéré (**LWPG**) :

$$\left\{ \begin{array}{l} 2(D(u^{n+1}), D(v)) + B_i(\lambda^n, D(v)) = (f, v), \quad \forall v \in V \\ \rho^n = \begin{cases} \rho_R & \text{si } |||D(u^{n+1}) + Bi\lambda^n||| < Bi \\ \rho_F & \text{sinon} \end{cases} \\ \lambda^{n+1} = P_K(\lambda^n + \rho^n D(u^{n+1})) \quad \forall \rho > 0 \end{array} \right. \quad (1)$$

Il s'est d'abord dans un premier temps, intéresser à des valeurs constantes du pas  $\rho$  dans les deux types de région (Rigide et Fluide) pour examiner les performances sur des cas mono-dimensionnels simples, tout en démontrant la convergence de l'algorithme de gradient projeté localement pondéré (LWPG). Dans un second temps, il a réalisé des tests numériques sur le problème de Couette avec l'algorithme LWPG ainsi que la variante de pénalisation (LPWPG : que nous présenterons plus tard) pour mettre en évidence les gains de performances qu'on espère obtenir par rapport à l'algorithme d'**UZAWA** classique.

A fin d'utiliser l'hypothèse de la zone rigide et de bien exposer l'algorithme en utilisant les règles **BB** et de **GLL** avec la pénalisation (**LPWPG**), on considère la formulation suivante :

$$\left\{ \begin{array}{l} \phi(\mu) = \frac{1}{Bi}|D(u(\mu))|^2, \quad \forall v \in V \\ R(\mu) = \{x \in \Omega; |2D(u(\mu)) + Bi\mu| \leq Bi\} \\ 2(D(u(\mu)), D(v)) = (f, v) - Bi(\mu, D(v)) \quad \forall v \in V \end{array} \right. \quad (2)$$

où  $R(\lambda)$  est l'ensemble des zones rigides .

On se lance dans l'idée intéressante de pouvoir exploiter cette condition  $D(u) = 0$  caractérisant les zones rigides.

Pour cela considère la formulation de Bingham avec pénalisation :

$$\left\{ \begin{array}{l} 2(D(u_a(\lambda)), D(v)) + a_R(D(u_a(\lambda)), D(v))_R + Bi(\lambda, D(v)) = (f, v), \quad \forall v \in V \\ (\mu - \lambda, D(u_a(\lambda))) \leq 0, \mu \in K \end{array} \right. ,$$

ce qui permet d'écrire l'algorithme pour la pénalisation **LPWPG** en utilisant le calcul de  $\rho$  avec **Barzilai Borwein** et celui de  $\alpha$  avec **GLL** noté **BBLPWPG** dans le code.

Nous avons utilisé les mêmes idées que précédemment, mais jusqu'ici, la fonctionnelle de minimisation change (avec la pénalisation) d'expression et aussi on tient compte de la limitation de  $\rho_R^n$  dans la zone rigide et puis du calcul de  $\rho_F^n$  avec  $\rho_{sec}$  dans la zone fluide.

On a :  $\phi_{a_R}(\lambda^n) = \frac{1}{2iBi}|u_{a_R}(\lambda^n)|_{L^2(\Omega)}^2 + a_R|u_{a_R}(\lambda^n)|_{L^2(\mathbb{R})}$

De plus, on fait les calculs de réductions avec l'expression de cette fonctionnelle de pénalisation.

On a :  $\forall n \geq 1$  :

$$\begin{aligned}\lambda^n + \alpha G_\rho(\lambda^n) &= \lambda^n + \alpha \left( P_K(\lambda^n + \rho^n D(u^{n+1})) - \lambda^n \right) \\ &= (1 - \alpha)\lambda^n + \alpha P_K(\lambda^n + \rho^n D(u^{n+1})) \\ u_{a_R}(\lambda^n + \alpha G_\rho(\lambda^n)) &= \alpha u_{a_R}((P_K(\bar{\lambda}^n)) + (1 - \alpha)u^{n+1}) \\ 2Bi\phi(\lambda^n + \alpha G_\rho(\lambda^n)) &= |\alpha u_{a_R}(P_K(\bar{\lambda}^n)) + (1 - \alpha)u^{n+1}|^2 + a_R|\alpha u_{a_R}(P_K(\bar{\lambda}^n)) + (1 - \alpha)u^{n+1}| \\ &= \alpha^2|u_{a_R}(P_K(\bar{\lambda}^n))|^2 + 2\alpha(1 - \alpha) \left( u_{a_R}(P_K(\bar{\lambda}^n)), D(u^{n+1}) \right) + (1 - \alpha)^2|u^{n+1}|^2 \\ &\quad + a_R|\alpha u_{a_R}(P_K(\bar{\lambda}^n)) + (1 - \alpha)u^{n+1}|\end{aligned}$$

On pose  $\forall n \geq 1$  :

$$\begin{aligned}\bar{\lambda}^n &= \lambda^n + \rho^n D(u^{n+1}) \\ A^n &= \frac{1}{Bi}|u_{a_R}(P_K(\bar{\lambda}^n))|^2, \\ B^n &= \frac{1}{Bi} \left( u_{a_R}(P_K(\bar{\lambda}^n)), u^{n+1} \right), \\ C^n &= a_R|\alpha u_{a_R}(P_K(\bar{\lambda}^n)) + (1 - \alpha)u^{n+1}|, \\ T^n &= (D(u^{n+1}), G_\rho(\lambda^n)) \\ \sigma_1 = 0.1, \sigma_2 = 0.9, \gamma = 10^{-4}, \varepsilon = 10^{-7} &, \begin{cases} \phi_{a_R}(\lambda^n) = \frac{1}{2iBi}|u_{a_R}(\lambda^n)|_{L^2(\Omega)}^2 + a_R|u_{a_R}(\lambda^n)|_{L^2(\mathbb{R})} \\ \phi_{max} = max \left\{ \phi(\lambda^{n-k}), 0 \leq k \leq min \{n, M-1\} \right\} \end{cases}, \\ \rho_{sec} = \frac{1}{Bi} \frac{|\lambda^n - \lambda^{n-1}|^2}{(\lambda^n - \lambda^{n-1}, D(u^{n+1}) - D(u^n))}, \rho_{min} = 10^{-30}, \rho_{max} = 10^{-30} &\text{ et } G_\rho(\lambda^n) = P_K(\bar{\lambda}^n) - \lambda^n\end{aligned}$$

$$\left\{
\begin{array}{l}
2(D(u^{n+1}), D(v)) + a_R(D(u^{n+1}), D(v))_R = (f, v) - Bi(\lambda^n, D(v)), \quad \forall v \in V \\
G_\rho(\lambda^n) = P_K(\bar{\lambda}^n) - \lambda^n \\
\textbf{BB} \quad \rho^n = \begin{cases} \rho_R^n = \rho_{max} & \text{si } ((\lambda^n - \lambda^{n-1}), D(u^{n+1}) - D(u^n)) \leq 0 \text{ et } |||2D(u^{n+1}) + Bi\lambda^n||| < Bi \\ \rho_F^n = \max \{\rho_{min}, \min \{\rho_{sec}, \rho_{max}\}\} & \text{sinon} \end{cases} \\
\text{on cherche la suite } \alpha^n \text{ telle que :} \\
\textbf{GLL} \quad \begin{cases} \alpha^{n,1} = 1 \\ \text{Si } (\alpha^{n,k})^2 A^n + 2\alpha^{n,k}(1 - \alpha^{n,k})B^n + (1 - \alpha^{n,k})^2 |u|^{n+1} + C^n \leq \phi_{max} - \gamma\alpha^{n,k}T^n \text{ Alors } \alpha^n = \alpha^{n,k} \\ \text{Sortie de la sous boucle} \\ \alpha^{n,k+1} = \begin{cases} \alpha_{temp} = \frac{1}{2}(\alpha^{n,k})^2 T^n & \text{si } \alpha_{temp} \in [\sigma_1, \sigma_2 \alpha^{n,k}] \\ \frac{\alpha^{n,k}}{2} & \text{sinon} \end{cases} \end{cases} \\
\lambda^{n+1} = \lambda^n + \alpha^n G_\rho(\lambda^n)
\end{array}
\right. \tag{3}$$

où  $a_R > 0$  est le coefficient de pénalisation qui a pour rôle de forcer la suite itérée  $D(u^{n+1})$  à converger le plus vite possible vers  $D(u)$  et  $R$

Soit  $\varepsilon > 0$  nous utilisons le Critère d'arrêt :  $(|z^n|_{L^2} + |\lambda^{n+1} : D(u^{n+1}) - |||D(u^{n+1})||||_{L^\infty}) < \varepsilon$  pour l'algorithme.

# Bibliographie

- [Barzilai and Borwein, 1988] Barzilai, J. and Borwein, J. M. (1988). Two-point step size gradient methods. *IMA journal of numerical analysis*, 8(1) :141–148.
- [Beck and Teboulle, 2009] Beck, A. and Teboulle, M. (2009). Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE transactions on image processing*, 18(11) :2419–2434.
- [Bingham, 1922] Bingham, E. C. (1922). *Fluidity and plasticity*, volume 2. McGraw-Hill.
- [Birgin and Martínez, 2014] Birgin, E. G. and Martínez, J. M. (2014). Practical augmented lagrangian methods for constrained optimization.
- [Birgin et al., 2000] Birgin, E. G., Martínez, J. M., and Raydan, M. (2000). Nonmonotone spectral projected gradient methods on convex sets. *SIAM Journal on Optimization*, 10(4) :1196–1211.
- [Bleyer, 2018] Bleyer, J. (2018). Advances in the simulation of viscoplastic fluid flows using interior-point methods. *Computer Methods in Applied Mechanics and Engineering*, 330 :368–394.
- [Bonettini et al., 2008] Bonettini, S., Zanella, R., and Zanni, L. (2008). A scaled gradient projection method for constrained image deblurring. *Inverse problems*, 25(1) :015002.
- [Brenner et al., 2008] Brenner, S. C., Scott, L. R., and Scott, L. R. (2008). *The mathematical theory of finite element methods*, volume 3. Springer.
- [Brezzi and Fortin, 2012] Brezzi, F. and Fortin, M. (2012). Mixed and hybrid finite element methods. *Journal of Non-Newtonian fluid mechanics*, 15.
- [Carstensen et al., 2016] Carstensen, C., Reddy, B. D., and Schedensack, M. (2016). A natural nonconforming fem for the bingham flow problem is quasi-optimal. *Numerische Mathematik*, 133(1) :37–66.
- [Duvaut and Lions, 1972] Duvaut, G. and Lions, J.-L. (1972). Transfert de chaleur dans un fluide de bingham dont la viscosité dépend de la température. *Journal of Functional Analysis*, 11(1) :93–110.

- [Falk and Mercier, 1977] Falk, R. S. and Mercier, B. (1977). Error estimates for elasto-plastic problems. *RAIRO. Analyse numérique*, 11(2) :135–144.
- [Fortin and Glowinski, 2000] Fortin, M. and Glowinski, R. (2000). *Augmented Lagrangian methods : applications to the numerical solution of boundary-value problems*. Elsevier.
- [Girault and Raviart, 1979] Girault, V. and Raviart, P.-A. (1979). An analysis of a mixed finite element method for the navier-stokes equations. *Numerische Mathematik*, 33 :235–271.
- [Glowinski, ] Glowinski, R. R., 1984," numerical methods for nonlinear variational problems.
- [Glowinski, 1976] Glowinski, R. (1976). Sur l'approximation d'une inéquation variationnelle elliptique de type bingham. *Revue française d'automatique, informatique, recherche opérationnelle. Analyse numérique*, 10(R3) :13–30.
- [Glowinski, 2008] Glowinski, R. (2008). *Lectures on numerical methods for non-linear variational problems*. Springer Science & Business Media.
- [Grippo et al., 1986] Grippo, L., Lampariello, F., and Lucidi, S. (1986). A nonmonotone line search technique for newton's method. *SIAM Journal on Numerical Analysis*, 23(4) :707–716.
- [Gustafsson and Lederer, 2022] Gustafsson, T. and Lederer, P. L. (2022). Mixed finite elements for bingham flow in a pipe. *Numerische Mathematik*, 152(4) :819–840.
- [Han and Reddy, 1995] Han, W. and Reddy, B. D. (1995). On the finite element method for mixed variational inequalities arising in elastoplasticity. *SIAM journal on numerical analysis*, 32(6) :1778–1807.
- [Nesterov, 1983] Nesterov, Y. E. (1983). A method of solving a convex programming problem with convergence rate  $O(\sqrt{k^2})$ . In *Doklady Akademii Nauk*, volume 269, pages 543–547. Russian Academy of Sciences.
- [Roquet et al., 2000] Roquet, N., Michel, R., and Saramito, P. (2000). Errors estimate for a viscoplastic fluid by using pk finite elements and adaptive meshes. *COMPTES RENDUS DE L'ACADEMIE DES SCIENCES SERIE I-MATHEMATIQUE*, 331(7) :563–568.
- [Saramito, 2016] Saramito, P. (2016). A damped newton algorithm for computing viscoplastic fluid flows. *Journal of Non-Newtonian fluid mechanics*, 238 :6–15.
- [Saramito and Wachs, 2017] Saramito, P. and Wachs, A. (2017). Progress in numerical simulation of yield stress fluid flows. *Rheologica Acta*, 56 :211–230.
- [Trémolières et al., 2011] Trémolières, R., Lions, J.-L., and Glowinski, R. (2011). *Numerical analysis of variational inequalities*. Elsevier.

[Treskatis et al., 2018] Treskatis, T., Roustaei, A., Frigaard, I., and Wachs, A. (2018). Practical guidelines for fast, efficient and robust simulations of yield-stress flows without regularisation : A study of accelerated proximal gradient and augmented lagrangian methods. *Journal of Non-Newtonian Fluid Mechanics*, 262 :149–164.