

Un algorithme de Gradient Conjugué Projeté préconditionné pour la résolution de problèmes unilatéraux

Nicolas Tardieu ^{a,*}, Fabien Youbissi ^b, Eric Chamberland ^b

^a LaMSID – UMR EDF/CNRS 2832, EDF-R&D, 1, avenue du Général-de-Gaulle, 92140 Clamart cedex, France

^b GIREF, Université Laval, Québec, Canada G1K7P4

Reçu le 15 janvier 2008 ; accepté après révision le 19 octobre 2008

Disponible sur Internet le 18 novembre 2008

Présenté par Huy Duong Bui

Résumé

L'algorithme du Gradient Conjugué Projeté (GCP) est issu du domaine de l'optimisation convexe où il est utilisé pour résoudre des problèmes de minimisation sous contraintes inégalités linéaires. Il se prête donc bien au cas du contact unilatéral. Appliqué à la forme duale du problème de contact et associé à un préconditionneur de Dirichlet, il exhibe un comportement très performant notamment quand on raffine le maillage. *Pour citer cet article* : N. Tardieu et al., C. R. Mecanique 336 (2008).

© 2008 Publié par Elsevier Masson SAS pour l'Académie des sciences.

Abstract

A preconditioned Projected Conjugate Gradient for the solution of unilateral problems. The Projected Conjugate Gradient method comes from the field of convex optimization, where it is used to solve minimization problems with linear inequality constraints. Thus, it is well suited for unilateral contact. Utilized on the dual form of the contact problem and associated to a Dirichlet preconditioner, it exhibits a performing behaviour, especially as the mesh size increases. *To cite this article*: N. Tardieu et al., C. R. Mecanique 336 (2008).

© 2008 Publié par Elsevier Masson SAS pour l'Académie des sciences.

Mots-clés : Contact ; Gradient Conjugué ; Préconditionneur

Keywords: Contact; Conjugate Gradient; Preconditioner

Abridged English version

Let us consider a contact problem on a part Γ_C of the boundary of an elastic body Ω , whose solution expresses as the minimization:

$$\min_{\mathbf{u} \cdot \mathbf{n} \geq \text{Gap}} \mathcal{J}(\mathbf{u}) \quad (1)$$

* Auteur correspondant.

Adresses e-mail : nicolas.tardieu@edf.fr (N. Tardieu), fyoubiss@giref.ulaval.ca (F. Youbissi), eric.chamberland@giref.ulaval.ca (E. Chamberland).

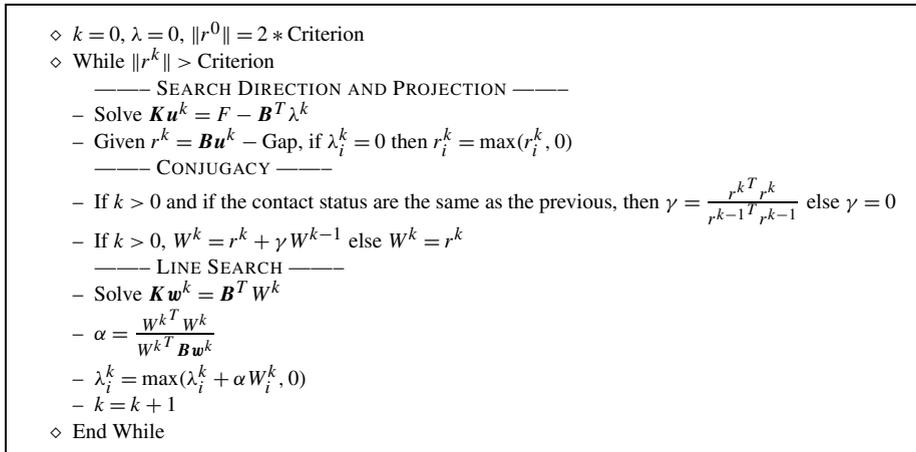


Fig. 1. Projected Conjugate Gradient.

where:

- \mathcal{J} is the potential energy function;
- \mathbf{u} is the displacement field;
- \mathbf{n} is the outward normal on Γ_C ;
- Gap is the gap function.

By introducing Lagrange multipliers λ , dualizing the finite element expression of (1), we obtain a dual discretized functional $\mathcal{H}(\lambda)$. Adding the indicator \mathcal{I}_{Δ^+} of the convex subset $\Delta^+ = \{\lambda \geq 0\}$ (for precise positivity definition, see [4]), we get the equivalent formulation:

$$\min_{\lambda} \mathcal{L}(\lambda) = \min_{\lambda} (\mathcal{H}(\lambda) + \mathcal{I}_{\Delta^+}(\lambda)) \tag{2}$$

Since this minimization problem is unconstrained, it can be solved using a Conjugate Gradient algorithm. One can notice that the functional \mathcal{L} is non-differentiable and thus implies the use of subgradients. Given the determination of the minimal norm subgradient in order to define a descent direction [5], the algorithm can be written as shown in Fig. 1.

The following remarks can be made on this algorithm:

- the algorithm manipulates primal fields since they are commonly available in finite element softwares but it minimizes the dual functional \mathcal{L} ;
- the line-search is unfeasible in the sense that the Lagrange multipliers λ can temporarily lie outside the admissible set but are then projected on the admissible set. A feasible line-search can easily be defined by limiting the value of the step ρ ;
- conjugacy must be carefully applied only if the contact status are unchanged from an iteration to another, otherwise it loses its mathematical meaning;
- the algorithm performs very well when using a direct solver for the linear systems. The matrix \mathbf{K} is then already factorized and only forward-backward substitutions are necessary for further resolutions.

In order to ease the convergence, the use of a dedicated preconditioner is mandatory [6]. The one we propose is based on the fact that the line-search phase implies an update equation $\lambda^k + \alpha W^k$. It must be noted that this two vectors belong to different functional spaces: $\lambda^k \in H^{-1/2}(\Gamma_C)$ while $W^k \in H^{1/2}(\Gamma_C)$. A Steklov–Poincaré operator is missing in order to transfer the search direction W^k from one set to the other. An auxiliary problem is proposed where \tilde{W}^k is the preconditioned of the search direction W^k :

$$\begin{cases} \mathbf{K} \mathbf{u} + \mathbf{B}_{\text{act}}^T \tilde{W}^k = \mathbf{0} \\ \mathbf{B}_{\text{act}} \mathbf{u} = W^k \end{cases} \tag{3}$$

where \mathbf{B}_{act} is the gap function limited to the active set of contact relations.

This problem (3) is solved using the previous method based on the minimization of the dual functional, which now implies gradients and not subgradients in order to use a Conjugate Gradient algorithm. The method uses once again the matrix \mathbf{K} which is already factorized.

With these different ingredients, the algorithm seems to exhibit an optimal behaviour as the mesh size decreases.

This method is similar to the FETI-C method [8], since it solves a series of Neumann problems preconditionned by a Dirichlet preconditionner. Several differences must nevertheless be noted. The present method is not build upon the FETI domain decomposition technique [9]; it arises from the use of sub-differential optimization and gives a natural definition of the projection operator. Furthermore, its simple algorithm is easily implemented in an existing finite element software.

1. Introduction

L'algorithme du Gradient Conjugué Projeté (GCP) est issu du domaine de l'optimisation convexe où il est utilisé pour résoudre des problèmes de minimisation sous contraintes inégalités linéaires. Des travaux précédents ont été réalisés portant sur l'utilisation de cet algorithme dans le cadre des problèmes de contact. La spécificité du présent travail réside dans l'application de cet algorithme à des cas de solides déformables (par opposition à [1]), dans l'utilisation de la formulation duale du problème de contact (par opposition à [2] et [3]) et surtout dans la définition d'un preconditionneur basé sur des problématiques mathématiques résolues par des considérations mécaniques.

On considère un problème de contact sur une partie Γ_C de la frontière d'un solide élastique Ω , dont la résolution s'exprime sous la minimisation suivante :

$$\min_{\mathbf{u} \cdot \mathbf{n} \geq \text{Gap}} \mathcal{J}(\mathbf{u}) \quad (4)$$

où :

- \mathcal{J} est la fonctionnelle d'énergie potentielle ;
- \mathbf{u} est le champ de déplacement ;
- \mathbf{n} est la normale extérieure sur Γ_C ;
- Gap est la fonction écart.

Il est classique d'introduire un champ de multiplicateurs de Lagrange λ de manière à relaxer la contrainte sur le champ de déplacements. Après discrétisation par éléments finis, des manipulations algébriques permettent d'exprimer cette résolution sous sa forme duale :

$$\min_{\lambda \geq 0} \mathcal{H}(\lambda) \quad (5)$$

où :

- $\mathcal{H}(\lambda) = \frac{1}{2} \lambda^T \mathbf{B} \mathbf{K}^{-1} \mathbf{B}^T \lambda + \lambda^T (\text{Gap} - \mathbf{B} \mathbf{K}^{-1} \mathbf{F}) + \frac{1}{2} \mathbf{F}^T \mathbf{K}^{-1} \mathbf{F}$;
- \mathbf{K} est la matrice de rigidité du système ;
- \mathbf{B} est l'opérateur de jeu sur Γ_C ;
- \mathbf{F} est le second membre qui contient les chargements.

L'intérêt d'utiliser cette expression est que l'opérateur obtenu présente un meilleur conditionnement que son pendant primal. Comme l'algorithme envisagé est de nature itérative, cet élément prend toute son importance. Le problème (5) peut être réécrit en supprimant la contrainte explicite de positivité sur λ en ajoutant à la fonctionnelle l'indicatrice \mathcal{I}_{Δ^+} du convexe $\Delta^+ = \{\lambda \geq 0\}$. Pour la définition précise de la positivité du multiplicateur, nous renvoyons à [4].

$$\min_{\lambda} \mathcal{L}(\lambda) = \min_{\lambda} (\mathcal{H}(\lambda) + \mathcal{I}_{\Delta^+}(\lambda)) \quad (6)$$

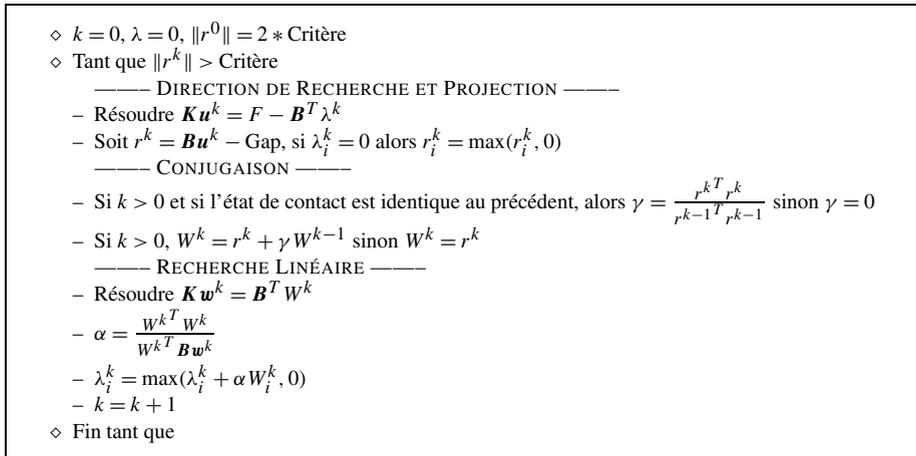


Fig. 2. Algorithme de GCP.

2. Algorithme de résolution

Comme (6) est un problème de minimisation sans contraintes, on peut le résoudre par l'algorithme du GCP en prenant garde à la non-différentiabilité de la fonctionnelle qui nécessite la manipulation de sous-gradients. En outre, de manière à obtenir une direction de descente efficace, on cherche à déterminer l'élément de norme minimale du sous-gradient $\partial \mathcal{L}$ [5]. On obtient alors :

$$\operatorname{argmin}_{j \in \partial \mathcal{L}} \|j\| = \begin{cases} \max(\mathbf{B}u - \text{Gap}, 0) & \text{si } \lambda = 0 \\ \mathbf{B}u - \text{Gap} & \text{sinon} \end{cases} \quad (7)$$

On peut utiliser ce résultat pour écrire un algorithme de descente très classique où l'introduction de la conjugaison des directions de recherche accélère sensiblement la convergence. On propose en Fig. 2 l'algorithme en question dont l'écriture est basée sur la formulation primale du problème de contact. On insiste sur le fait que cet algorithme réalise bien la minimisation de la fonctionnelle $\mathcal{L}(\lambda)$ mais il manipule les grandeurs primales qui sont communément accessibles dans un code de calcul par éléments finis. A la lecture de cet algorithme, on peut remarquer que :

- la phase de recherche linéaire est non-admissible au sens où la mise à jour du multiplicateur de Lagrange λ peut le faire sortir du domaine admissible ce qui est corrigé par une phase de projection. Il est très facile de définir une recherche linéaire admissible en limitant la valeur du pas d'avancement ρ ;
- il faut prendre garde à appliquer la conjugaison uniquement si l'ensemble des contraintes actives ne change pas d'une itération à l'autre sans quoi la conjugaison perd sa pertinence mathématique ;
- l'algorithme trouve sa performance dans l'utilisation d'un solveur direct pour la résolution des systèmes linéaires. La matrice \mathbf{K} est alors déjà factorisée et chaque résolution se limite donc à une descente-remontée.

3. Préconditionneur

Comme il est usuel dans le cas de l'utilisation de méthodes itératives, il est souhaitable de preconditionner le problème afin d'en accélérer la résolution [6]. Le preconditionneur proposé est basé sur des remarques sur les espaces fonctionnels auxquels appartiennent les différents objets manipulés dans l'algorithme. Durant la phase de recherche linéaire, le multiplicateur de Lagrange λ est mis à jour par la formule $\lambda^k + \alpha W^k$. Or, ces deux vecteurs n'appartiennent pas aux mêmes espaces fonctionnels : $\lambda^k \in H^{-1/2}(\Gamma_C)$ tandis que $W^k \in H^{1/2}(\Gamma_C)$. Il manque ici un opérateur de Steklov–Poincaré pour transférer la direction de recherche W^k d'un espace à l'autre.

En s'inspirant de la technique de relèvement développée dans [7], on propose de résoudre le problème annexe suivant où \tilde{W}^k est le preconditionné de la direction de descente W^k du GCP :

$$\begin{cases} \mathbf{K}u + \mathbf{B}_{\text{act}}^T \tilde{W}^k = \mathbf{0} \\ \mathbf{B}_{\text{act}} u = W^k \end{cases} \quad (8)$$

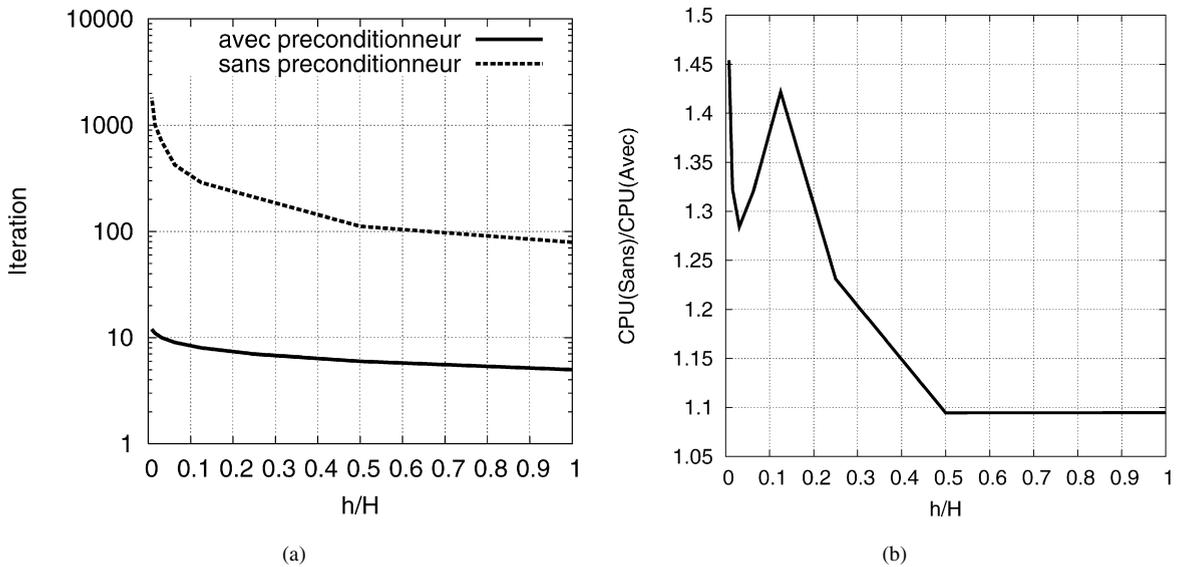


Fig. 3. Comparaison des algorithmes avec et sans préconditionneur.

où \mathbf{B}_{act} est l'opérateur de jeu limité aux liaisons actives de l'itération courante.

Ce problème linéaire est résolu en suivant la même démarche présentée précédemment en exploitant sa forme duale. On manipule alors des gradients et non des sous-gradients pour alimenter un algorithme de Gradient Conjugué. On remarque que les résolutions de systèmes linéaires impliquent la matrice \mathbf{K} qui est déjà factorisée.

Les fondements mathématiques de cette méthode sont similaires à ceux de la méthode FETI-C [8], dans la mesure où toutes deux résolvent une série de problèmes de Neumann préconditionnés par un problème de Dirichlet. Des différences notables entre ces deux approches peuvent néanmoins être relevées :

- Construction formelle : la méthode FETI-C est obtenue par extension de la méthode de décomposition de domaines FETI [9] à des relations unilatérales entre sous-domaines. Pour ce faire, une algorithmie particulière dotée d'une projection adéquate du résidu d'interface est définie par les auteurs. L'introduction de la présente méthode ne se fait pas dans le cadre de la décomposition de domaine ; on aborde le problème de contact comme un problème de minimisation sous contrainte. Par la suite, l'utilisation d'outils de l'optimisation non-différentiable permettent de construire l'algorithme de manière naturelle et surtout de donner une justification originale à la définition de la projection du résidu (sous-gradient de norme minimale) ;
- Implémentation numérique : la mise en œuvre numérique de la méthode FETI-C se fait dans le cadre de celle de la méthode FETI ; elle nécessite de fait l'ensemble des outils et structures de données de cette dernière. La mise en œuvre numérique de la présente méthode se distingue par le fait qu'elle se libère du cadre de la décomposition de domaine et devient donc beaucoup plus simple à la fois dans son écriture et dans sa programmation. Ces éléments rendent cette méthode aisément intégrable dans un logiciel de simulation existant.

4. Illustration

On propose d'illustrer le comportement de l'algorithme proposé sur le cas d'un problème de Hertz (écrasement d'une sphère sur un plan rigide) modélisé en axisymétrie. On compare le nombre d'itérations ainsi que le rapport des temps CPU requis pour obtenir la solution avec et sans préconditionneur quand on raffine le maillage. On utilise pour cela un maillage à base d'éléments triangulaires linéaires que l'on subdivise en découpant leurs arêtes en 2. On passe ainsi de 10^3 à 2×10^6 degrés de liberté. Dans les Figs. 3 (a) et (b), H représente la taille initiale des éléments tandis que h est leur taille courante. On remarque sur cette figure que :

- l'utilisation du préconditionneur réduit toujours le nombre d'itérations requis pour atteindre la convergence ;

- l’augmentation du nombre d’itérations avec la taille du problème quand on utilise le préconditionneur est très limitée : on passe de 5 à 12 itérations tandis que sans préconditionneur, on passe de 79 à 1823 itérations, ce qui autorise à supposer son optimalité ;
- en terme de temps CPU, l’utilisation du préconditionneur est toujours intéressante et cet intérêt croît avec la taille du modèle.

5. Conclusions

Dans ce papier a été présenté un algorithme de Gradient Conjugué Projeté Préconditionné basé sur la formulation duale d’un problème de contact unilatéral. Associé à un préconditionneur de nature Dirichlet, il exhibe des performances intéressantes quand la taille du maillage augmente. Une autre qualité majeure de cet algorithme est le très faible surcoût mémoire et disque du traitement du contact (pas de matrice tangente ou de factorisation supplémentaire). Si le problème sans contact « passe » sur une machine donnée, il est acquis qu’il en sera de même pour le problème avec contact. Cette constatation a été vérifiée lors de l’implantation de cet algorithme et de son préconditionneur dans le logiciel de simulation *Code_Aster* [10], où des simulations industrielles impliquant plusieurs millions de degrés de liberté en présence de contact ont pu être menées à bien.

Références

- [1] M. Renouf, P. Alart, Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials, *Comput. Methods Appl. Mech. Engrg.* 194 (2005) 2019–2041.
- [2] H.-O. May, The conjugate gradient method for unilateral problems, *Comput. & Structures* 22 (1986) 595–598.
- [3] G. Dilintas, P. Laurent-Gengoux, D. Trystram, A conjugate projected gradient method with preconditioning for unilateral contact problems, *Comput. & Structures* 29 (1988) 675–680.
- [4] P. Hild, Numerical implementation of two nonconforming finite element methods for unilateral contact, *Comput. Methods Appl. Mech. Engrg.* 184 (2000) 99–123.
- [5] M.S. Bazaraa, H.D. Sherali, M. Shetty, *Nonlinear Programming – Theory and Algorithms*, John Wiley and Sons, 1993.
- [6] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, 2005.
- [7] M. Fortin, N. Tardieu, E. Chamberland, Un algorithme de sous-gradient pour le traitement du contact frottant, in : *Congrès de Mécanique de Giens*, 2005.
- [8] D. Duraisseix, C. Farhat, A numerically scalable domain decomposition method for the solution to frictionless contact problems, *Int. J. Numer. Meth. Engrg.* 50 (2001) 2643–2666.
- [9] C. Farhat, F.-X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, *Int. J. Numer. Meth. Engrg.* 32 (1991) 1205–1227.
- [10] Site web de *Code_Aster*, www.code-aster.org.