Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Comput. Methods Appl. Mech. Engrg. 192 (2003) 1451–1462
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Inexact Uzawa algorithms for variational inequalities of the second kind

Xiao-liang Cheng [a,1], Weimin Han [b,*,2]

[a] Department of Mathematics, Zhejiang University, Hangzhou 310028, PR China
[b] Department of Mathematics, University of Iowa, 14 MacLean Hall, Iowa City, IA 52242, USA

Received 20 March 2002; received in revised form 18 October 2002

Abstract

In this paper we discuss inexact Uzawa algorithms and inexact non-linear Uzawa algorithms to solve discretized variational inequalities of the second kind. We prove convergence results for the algorithms. Numerical examples are included to show the effectiveness of the algorithms.

Keywords: Uzawa algorithm; Inexact Uzawa algorithm; Variational inequality of the second kind

1. Introduction

Variational inequalities are an important family of non-linear boundary value or initial-boundary value problems with applications in mechanics, physical sciences, etc. A partial list of comprehensive references on the topic include [2,9,11,16,17,21,22,25,27].

In the literature (cf. e.g. [12]), variational inequalities are classified into two kinds. Variational inequalities of the first kind refer to those defined over convex subsets of function spaces, while variational inequalities of the second kind are those involving non-differentiable terms in the formulations. Let $\Omega \subset \mathbb{R}^d$ be a non-empty, open, bounded domain, and $V \subset H^1(\Omega)$ be a Sobolev space defined on Ω. Consider a general elliptic variational inequality of the second kind:

$$u \in V, \quad a(u, v-u) + j(v) - j(u) \geqslant l(v-u) \quad \forall v \in V. \tag{1.1}$$

Assume $a(\cdot,\cdot) : V \times V \to \mathbb{R}$ is bilinear, continuous and elliptic, $j(\cdot) : V \to \overline{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$ is proper, convex and lower semi-continuous, and $l(\cdot) : V \to \mathbb{R}$ is linear and continuous. Then (1.1) is uniquely solvable by a standard result [12,13].

* Corresponding author.
E-mail address: whan@math.uiowa.edu (W. Han).
[1] The work of this author was supported by National Natural Science Foundation (Grant no. 10001029) of China.
[2] The work of this author was supported by NSF under Grant DMS-0106781.

0045-7825/03/$ - see front matter
doi:10.1016/S0045-7825(02)00651-5

Denote V' the dual space of V. Recall the definition of sub-differential $\partial j(u) : w^* \in \partial j(u)$ is equivalent to

$$w^* \in V', \quad j(v) \geqslant j(u) + \langle w^*, v-u \rangle_{V'\times V} \quad \forall v \in V.$$

We can rewrite (1.1) as

$$j(v) \geqslant j(u) + l(v-u) - a(u, v-u) \quad \forall v \in V.$$

Note that $v \mapsto l(v) - a(u,v)$ defines an element in V'; we call it $w^* \in V'$. Then (1.1) is equivalent to the existence of u and w^* such that

$$u \in V, \quad w^* \in \partial j(u), \quad a(u,v) + \langle w^*, v \rangle_{V'\times V} = l(v) \quad \forall v \in V. \tag{1.2}$$

In many applications, the non-differentiable functional $j(v)$ is of the form

$$j(v) = \|\mathscr{L} v\|_{L^1(D)^{d_1}} \tag{1.3}$$

for some subset $D \subset \Omega$ or $D \subset \partial\Omega$. The operator $\mathscr{L}$ is linear and continuous from V to $L^2(D)^{d_1}$. The operator $\mathscr{L}$ will be mostly chosen as $\mathscr{L}v = gv$ and then $d_1 = 1$, or $\mathscr{L}v = g\nabla v$ and then $d_1 = d$, the function $g \geqslant 0$ being given in $L^\infty(D)^{d_1}$. Two particular cases for the functional j we will consider are

Case 1

$$j(v) = \int_{\Gamma_1} g|v|\,\mathrm{d}s,$$

where $\Gamma_1 \subset \partial\Omega$. For this case, $D = \Gamma_1$, $\mathscr{L}v = gv|_{\Gamma_1}$ for $v \in H^1(\Omega)$.

Case 2

$$j(v) = \int_\Omega g|\nabla v|\,\mathrm{d}x, \quad |\nabla v| = \left[\sum_{i=1}^d \left(\frac{\partial v}{\partial x_i}\right)^2\right]^{1/2}.$$

For this case, $D = \Omega$, $\mathscr{L}v = g\nabla v$ for $v \in H^1(\Omega)$.

In any case, we have a constant c_0 such that

$$\|\mathscr{L}v\|^2_{L^2(D)^{d_1}} \leqslant c_0 a(v,v) \quad \forall v \in V. \tag{1.4}$$

In the rest of the paper, we focus on the numerical solution of the variational inequality (1.1) with $j(\cdot)$ given by (1.3). Note that the functional $j(\cdot)$ is positively homogeneous:

$$j(\alpha v) = \alpha j(v) \quad \forall \alpha \geqslant 0, \ v \in V.$$

We see that the inequality

$$j(v) \geqslant j(u) + \langle w^*, v-u \rangle_{V'\times V} \quad \forall v \in V \tag{1.5}$$

is equivalent to

$$\langle w^*, v \rangle_{V'\times V} \leqslant j(v) \quad \forall v \in V, \tag{1.6}$$

$$\langle w^*, u \rangle_{V'\times V} = j(u). \tag{1.7}$$

A sketch of a proof of the equivalence is as follows. In (1.5) we take $v = 0$ and $2u$, and use the positive homogeneity of $j(\cdot)$ to obtain (1.7). Then (1.6) follows from (1.5) and (1.7). Conversely, subtracting (1.7) from (1.6), we obtain (1.5).

Introduce the set

$$\Lambda = \{\boldsymbol{\mu} \in L^\infty(D)^{d_1} : \|\boldsymbol{\mu}\|_{L^\infty(D)^{d_1}} \leqslant 1\}$$

for the Lagrangian variable $\boldsymbol{\lambda}$, and the projection mapping $\mathscr{P}_\Lambda : L^\infty(D)^{d_1} \to \Lambda$ by the formula

$$\mathscr{P}_\Lambda \boldsymbol{\mu} = \frac{\boldsymbol{\mu}}{\sup(1, |\boldsymbol{\mu}|)}.$$

It is easy to show that for any vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ of the same dimension, we have

$$\left| \frac{\boldsymbol{a}}{\sup(1, |\boldsymbol{a}|)} - \frac{\boldsymbol{b}}{\sup(1, |\boldsymbol{b}|)} \right| \leqslant |\boldsymbol{a} - \boldsymbol{b}|.$$

Hence, $\mathscr{P}_\Lambda$ is non-expansive:

$$\|\mathscr{P}_\Lambda \boldsymbol{\mu}_1 - \mathscr{P}_\Lambda \boldsymbol{\mu}_2\|_{L^2(D)^{d_1}} \leqslant \|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_{L^2(D)^{d_1}} \quad \forall \boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in L^\infty(D)^{d_1}. \tag{1.8}$$

We can rewrite (1.2) as

$$u \in V, \quad a(u, v) + \langle \boldsymbol{\lambda}, \mathscr{L} v \rangle_{L^\infty(D)^{d_1} \times L^1(D)^{d_1}} = l(v) \quad \forall v \in V,$$
$$\boldsymbol{\lambda} \in \Lambda, \quad \langle \boldsymbol{\lambda}, \mathscr{L} u \rangle_{L^\infty(D)^{d_1} \times L^1(D)^{d_1}} = \|\mathscr{L} u\|_{L^1(D)^{d_1}},$$

or equivalently,

$$u \in V, \quad a(u, v) + (\boldsymbol{\lambda}, \mathscr{L} v)_{L^2(D)^{d_1}} = l(v) \quad \forall v \in V,$$
$$\boldsymbol{\lambda} \in \Lambda, \quad (\boldsymbol{\lambda}, \mathscr{L} u)_{L^2(D)^{d_1}} = \|\mathscr{L} u\|_{L^1(D)^{d_1}}.$$

We see that on the region where $\mathscr{L} u \neq \mathbf{0}$, we have $\boldsymbol{\lambda} = \mathscr{L} u / |\mathscr{L} u|$, whereas on the remaining region, $|\boldsymbol{\lambda}| \leqslant 1$. Thus, for any $\rho > 0$,

$$\boldsymbol{\lambda} = \mathscr{P}_\Lambda(\boldsymbol{\lambda} + \rho \mathscr{L} u). \tag{1.9}$$

Turning now to numerical approximations, let $V_h \subset V$ be a finite element space. We apply the finite element method to solve the variational inequality (1.1): Find $u_h \in V_h$ such that

$$a(u_h, v_h - u_h) + j(v_h) - j(u_h) \geqslant l(v_h - u_h) \quad \forall v_h \in V_h. \tag{1.10}$$

Again we can characterize the solution u_h of the discrete variational inequality (1.10) in the form of an equivalent non-linear system:

$$u_h \in V_h, \quad a(u_h, v_h) + (\boldsymbol{\lambda}_h, \mathscr{L} v_h)_{L^2(D)^{d_1}} = l(v_h) \quad \forall v_h \in V_h, \tag{1.11}$$

$$\boldsymbol{\lambda}_h \in \Lambda, \quad (\boldsymbol{\lambda}_h, \mathscr{L} u_h)_{L^2(D)^{d_1}} = \|\mathscr{L} u_h\|_{L^1(D)^{d_1}}. \tag{1.12}$$

Convergence and error estimates of the approximation (1.10) to (1.1) are discussed in several references, see e.g. [12,13,16]. In this paper, we focus our attention on iterative algorithms for the discretized variational inequality (1.10). In the literature, numerous iterative algorithms have been studied for discrete variational inequalities. In [12], relaxation and over-relaxation methods are analyzed for solving (1.10). These are extensions of the Gauss–Seidel and SOR methods to the iterative solution of discrete variational inequalities. There are some other extended relaxation methods as well, for example, the multigrid method [18,19,23,24] and projection method based on the domain decomposition [29].

The Uzawa algorithm is a predictor–corrector type method for solving non-linear problems with more than one variable. It was first studied by Uzawa and his co-workers in 1958 [1] in applying the gradient method to the minimization problem of the dual functional of the Stokes problem, which is a saddle point problem. The most attractive features of the Uzawa algorithm are its simplicity and robustness (see [7,12,14,15,26]). However, a well-known drawback of the Uzawa algorithm is the requirement of solving a linear system exactly for the predictor equation in each step of the iteration. For many applications, this requirement is costly. Recently, inexact Uzawa algorithms have attracted much attention in solving saddle point problems. In an inexact Uzawa algorithm, the predictor equation is solved inexactly. Usually, preconditioned one-step iteration method or multistep preconditioning conjugate gradient method is used in the predictor step. See [3–6,8,10,15,20] and the references therein for applications of inexact Uzawa algorithms in solving saddle point problems.

So far in the literature, there has been no discussion on inexact Uzawa-type algorithms for solving discrete variational inequalities. In [12], an exact Uzawa-type algorithm by duality method is studied for some variational inequalities. The problem (1.11) and (1.12) is solved by a predictor–corrector type Uzawa algorithm. Eq. (1.11) is used as a predictor for a given $\boldsymbol{\lambda}_h^n$, $n = 0, 1, 2, \ldots$,

$$a(u_h^{n+1}, v_h) = l(v_h) - (\boldsymbol{\lambda}_h^n, \mathscr{L} v_h)_{L^2(D)^{d_1}} \quad \forall v_h \in V_h, \tag{1.13}$$

and λ_h is updated by

$$\boldsymbol{\lambda}_h^{n+1} = \mathscr{P}_\Lambda(\boldsymbol{\lambda}_h^n + \rho \mathscr{L} u_h^{n+1}), \quad \rho > 0. \tag{1.14}$$

The main drawback of above Uzawa's algorithm is that it requires the solution of the linear system (1.11) at each iteration. To overcome this difficulty, it is natural to consider inexact Uzawa's algorithms as described above.

In this paper, we study inexact Uzawa and non-linear inexact Uzawa algorithms (also called multistep inexact Uzawa algorithm in the literature) for solving the problem (1.11) and (1.12). We prove convergence of the algorithms. We also show applications of the algorithms in solving some sample problems, including a simplified version of the friction problem in elasticity and the flow of a viscous plastic fluid in a pipe. Finally, some numerical results are reported to show the better performance of inexact Uzawa algorithms over the standard Uzawa algorithm.

2. Inexact Uzawa algorithms

For the sake of simplicity, we assume that Ω is a polyhedral domain of $\mathbb{R}^d$. Let $\{\mathscr{T}_h\}$ be a regular family of triangulations of $\overline{\Omega}$. As usual, h denotes the length of the largest diameter of the elements in the triangulation. We approximate V by the corresponding finite element subspaces $\{V_h\}$. Let $\{\phi_i\}_{i=1}^m$ be a set of basis functions of the space V_h. Denote A the stiffness matrix associated with the bilinear form $a(\cdot,\cdot)$,

$$a_{i,j} = a(\phi_i, \phi_j), \quad i, j = 1, 2, \ldots, m.$$

Let A_0 be a symmetric, positive definite $m \times m$ matrix, satisfying (cf. [5])

$$(1-\gamma)(A_0\boldsymbol{y}, \boldsymbol{y}) \leqslant (A\boldsymbol{y}, \boldsymbol{y}) \leqslant (A_0\boldsymbol{y}, \boldsymbol{y}) \quad \forall \boldsymbol{y} \in \mathbb{R}^m \tag{2.1}$$

for some constant $\gamma \in (0, 1)$. The matrix A_0 is chosen as a preconditioner of matrix A, so in particular, the equation $A_0\boldsymbol{x} = \boldsymbol{b}$ should be easier to solve than the equation $A\boldsymbol{x} = \boldsymbol{b}$. The matrix A_0 can be constructed from incomplete LU decomposition or be derived from multigrid method, multilevel method and domain decomposition method; see [3,4,28].

For the convenience of analysis, we denote

$$a_0(u_h, v_h) = (A_0\boldsymbol{u}, \boldsymbol{v})$$

for

$$u_h = \sum_{i=1}^m u_i\phi_i, \quad \boldsymbol{u} = (u_1, \ldots, u_m)^{\mathrm{T}},$$

$$v_h = \sum_{i=1}^m v_i\phi_i, \quad \boldsymbol{v} = (v_1, \ldots, v_m)^{\mathrm{T}}.$$

Then for $\omega = 1/(1-\gamma)$ we have

$$a(v_h, v_h) \leqslant a_0(v_h, v_h) \leqslant \omega a(v_h, v_h) \quad \forall v_h \in V_h. \tag{2.2}$$

Now we introduce an inexact Uzawa algorithm for the problem (1.11) and (1.12).

Algorithm 2.1. Let $\boldsymbol{\lambda}_h^0 \in \Lambda$ and $u_h^0 \in V_h$. For $n = 0, 1, 2, \ldots$, we compute $u_h^{n+1} \in V_h$ and $\boldsymbol{\lambda}_h^{n+1}$ inductively by

$$a_0(u_h^{n+1}, v_h) = a_0(u_h^n, v_h) - [a(u_h^n, v_h) + (\boldsymbol{\lambda}_h^n, \mathscr{L} v_h)_{L^2(D)^{d_1}} - l(v_h)] \quad \forall v_h \in V_h, \tag{2.3}$$

$$\boldsymbol{\lambda}_h^{n+1} = \mathscr{P}_\Lambda(\boldsymbol{\lambda}_h^n + \rho \mathscr{L} u_h^{n+1}). \tag{2.4}$$

Regarding the convergence of the algorithm, we have the following result.

Theorem 2.2. *Assume* (2.2). *Then for* $1 \leqslant \omega < 5$ *and* $0 < \rho < (5-\omega)/(2c_0)$, *Algorithm* 2.1 *converges*:

$$\lim_{n\to\infty} \|u_h^n - u_h\|_V = 0. \tag{2.5}$$

Proof. Denote the nth iteration error

$$e_h^n = u_h - u_h^n, \quad \boldsymbol{\epsilon}_h^n = \boldsymbol{\lambda}_h - \boldsymbol{\lambda}_h^n, \qquad n = 0, 1, 2, \ldots$$

Then from (1.11) and (2.3) we get the relation

$$(\boldsymbol{\epsilon}_h^n, \mathscr{L} e_h^{n+1})_{L^2(D)^{d_1}} = a_0(e_h^n - e_h^{n+1}, e_h^{n+1}) - a(e_h^n, e_h^{n+1}). \tag{2.6}$$

It follows from (1.12) that (the discrete analogue of (1.9))

$$\boldsymbol{\lambda}_h = \mathscr{P}_\Lambda(\boldsymbol{\lambda}_h + \rho \mathscr{L} u_h).$$

Thus

$$\boldsymbol{\epsilon}_h^{n+1} = \mathscr{P}_\Lambda(\boldsymbol{\lambda}_h + \rho \mathscr{L} u_h) - \mathscr{P}_\Lambda(\boldsymbol{\lambda}_h^n + \rho \mathscr{L} u_h^{n+1}).$$

From (1.8) and the relation (2.6), we obtain

$$\begin{aligned}\|\boldsymbol{\epsilon}_h^{n+1}\|^2_{L^2(D)^{d_1}} &\leqslant \|\boldsymbol{\epsilon}_h^n\|^2_{L^2(D)^{d_1}} + 2\rho(\boldsymbol{\epsilon}_h^n, \mathscr{L} e_h^{n+1})_{L^2(D)^{d_1}} + \rho^2 \|\mathscr{L} e_h^{n+1}\|^2_{L^2(D)^{d_1}} \\ &= \|\boldsymbol{\epsilon}_h^n\|^2_{L^2(D)^{d_1}} + 2\rho(a_0(e_h^n - e_h^{n+1}, e_h^{n+1}) - a(e_h^n, e_h^{n+1})) + \rho^2 \|\mathscr{L} e_h^{n+1}\|^2_{L^2(D)^{d_1}}.\end{aligned}$$

By the assumption (2.2) we have

$$\begin{aligned}a_0(e_h^n - e_h^{n+1}, e_h^{n+1}) - a(e_h^n, e_h^{n+1}) &= -a_0\big(\tfrac{1}{2}e_h^n - e_h^{n+1}, \tfrac{1}{2}e_h^n - e_h^{n+1}\big) + \tfrac{1}{4}a_0(e_h^n, e_h^n) - a(e_h^n, e_h^{n+1}) \\ &\leqslant -a\big(\tfrac{1}{2}e_h^n - e_h^{n+1}, \tfrac{1}{2}e_h^n - e_h^{n+1}\big) + \tfrac{1}{4}\omega a(e_h^n, e_h^n) - a(e_h^n, e_h^{n+1}) \\ &= \tfrac{1}{4}(\omega - 1) a(e_h^n, e_h^n) - a(e_h^{n+1}, e_h^{n+1}).\end{aligned}$$

Then

$$\|\boldsymbol{\epsilon}_h^{n+1}\|^2_{L^2(D)^{d_1}} \leqslant \|\boldsymbol{\epsilon}_h^n\|^2_{L^2(D)^{d_1}} + \tfrac{1}{2}(\omega - 1)\rho a(e_h^n, e_h^n) + (\rho^2 c_0 - 2\rho) a(e_h^{n+1}, e_h^{n+1}). \tag{2.7}$$

Denote

$$t_n = \|\boldsymbol{\epsilon}_h^n\|^2_{L^2(D)^{d_1}} + \tfrac{1}{2}\rho(\omega - 1) a(e_h^n, e_h^n). \tag{2.8}$$

Then we have from (2.7),

$$t_{n+1} \leqslant t_n + \big(c_0\rho^2 + \tfrac{1}{2}\rho(\omega - 5)\big) a(e_h^{n+1}, e_h^{n+1}). \tag{2.9}$$

When $1 \leqslant \omega < 5$ and $\rho < (5-\omega)/(2c_0)$, $\{t_n\}$ is a decreasing sequence and has a lower bound 0. Then the sequence $\{t_n\}$ has a limit and from its definition (2.8) and (2.9), we see that

$$\lim_{n\to\infty} a(e_h^n, e_h^n) = 0.$$

Therefore, we have the convergence (2.5). □

By Theorem 2.2, we see that the preconditioner A_0 should satisfy (2.2) with $1 \leqslant \omega < 5$. In practical computation, the multi-step preconditioning conjugate gradient method can be used to get an approximate solution of (1.13). For $\boldsymbol{\phi} \in \mathbb{R}^m$, denote $\boldsymbol{\psi}(\boldsymbol{\phi})$ an approximation to the solution $\boldsymbol{\xi}$ of

$$A\boldsymbol{\xi} = \boldsymbol{\phi}. \tag{2.10}$$

We assume that for some $0 \leqslant \delta < 1$, our approximation satisfies

$$\|\boldsymbol{\psi}(\boldsymbol{\phi}) - A^{-1}\boldsymbol{\phi}\|_A \leqslant \delta \|\boldsymbol{\phi}\|_{A^{-1}} \quad \forall \boldsymbol{\phi} \in \mathbb{R}^m. \tag{2.11}$$

Here, for a symmetric, positive definite matrix Q, we use the Q-norm:

$$\|\boldsymbol{\phi}\|_Q = (\boldsymbol{\phi}, Q\boldsymbol{\phi})^{1/2}.$$

As described in [5], the condition (2.11) is a reasonable assumption which is satisfied by the approximate inverse associated with the PCG algorithm. For example, let Q_A be a symmetric and positive definite $m \times m$ matrix and we apply l steps of the conjugate gradient algorithm preconditioned by Q_A to solve the linear equation (2.10) with a zero starting iterate. That is

$$\boldsymbol{\xi}_0 = \boldsymbol{0}, \quad \boldsymbol{r}_0 = A\boldsymbol{\xi}_0 - \boldsymbol{\phi}, \quad \boldsymbol{h}_0 = Q_A^{-1}\boldsymbol{r}_0, \quad \boldsymbol{d}_0 = -\boldsymbol{h}_0,$$

and for $k = 0, 1, \ldots, l-1$,

$$\begin{aligned}
&\tau_k = (\boldsymbol{r}_k, \boldsymbol{h}_k)/(\boldsymbol{d}_k, A\boldsymbol{d}_k), && \boldsymbol{\xi}_{k+1} = \boldsymbol{\xi}_k + \tau_k \boldsymbol{d}_k,\\
&\boldsymbol{r}_{k+1} = \boldsymbol{r}_k + \tau_k A\boldsymbol{d}_k, && \boldsymbol{h}_{k+1} = Q_A^{-1}\boldsymbol{r}_{k+1},\\
&\beta_k = (\boldsymbol{r}_{k+1}, \boldsymbol{h}_{k+1})/(\boldsymbol{r}_k, \boldsymbol{h}_k), && \boldsymbol{d}_{k+1} = -\boldsymbol{h}_{k+1} + \beta_k \boldsymbol{d}_k.
\end{aligned}$$

Then $\boldsymbol{\psi}(\boldsymbol{\phi}) = \boldsymbol{\xi}_l$ satisfies the inequality

$$\|\boldsymbol{\xi}_l - A^{-1}\boldsymbol{\phi}\|_A \leqslant \delta \|\boldsymbol{\phi}\|_{A^{-1}},$$

where

$$\delta = \delta_l \leqslant \frac{1}{\cosh(l \cosh^{-1}\eta)}, \qquad \eta = \frac{\kappa(Q_A^{-1}A) + 1}{\kappa(Q_A^{-1}A) - 1},$$

or

$$\delta = \delta_l \leqslant 2\left(\frac{(\kappa(Q_A^{-1}A))^{1/2} - 1}{(\kappa(Q_A^{-1}A))^{1/2} + 1}\right)^l$$

with $\kappa(Q_A^{-1}A)$ being the condition number of Q_A^{-1}. Since δ_l tends to zero as l tends to infinity, we can make δ_l as small as we wish by taking a suitably large number of PCG iterations.

Now we discuss the following non-linear inexact Uzawa algorithm:

Algorithm 2.3. Let $\boldsymbol{\lambda}_h^0 \in \Lambda$ and $u_h^0 \in V_h$. For $n = 0, 1, 2, \ldots$, we compute $u_h^{n+1} \in V_h$ and $\boldsymbol{\lambda}_h^{n+1} \in \Lambda$ as follows:

$$\begin{cases} \boldsymbol{u}^{n+1} = \boldsymbol{u}^n - \dfrac{1}{1+\delta}\boldsymbol{\psi}(\boldsymbol{s}^n),\\ \boldsymbol{\lambda}_h^{n+1} = \mathscr{P}_\Lambda(\boldsymbol{\lambda}_h^n + \rho \mathscr{L} u_h^{n+1}), \end{cases} \tag{2.12}$$

where $\boldsymbol{u}^n = (u_1^n, \ldots, u_m^n)^{\mathrm{T}}$, $u_h^n = \sum_{j=1}^m u_j^n \phi_j \in V_h$ and the vector $\boldsymbol{s}^n$ has the components

$$s_j^n = a(u_h^n, \phi_j) + (\boldsymbol{\lambda}_h^n, \mathscr{L}\phi_j)_{L^2(D)^{d_1}} - l(\phi_j), \quad j = 1, 2, \ldots, m.$$

For a convergence analysis of Algorithm 2.3, we will rewrite it as an equivalent linear inexact Uzawa algorithm. We need some preparatory results.

Lemma 2.4. [3, p. 661] *Let H be a symmetric, positive definite $m \times m$ matrix and let $\boldsymbol{d}, \hat{\boldsymbol{d}} \in \mathbb{R}^m$ satisfy*

$$\|\boldsymbol{d} - \hat{\boldsymbol{d}}\|_H \leqslant \delta \|\boldsymbol{d}\|_H$$

with $0 \leqslant \delta < 1$. Then there exists a symmetric, positive definite matrix $\widehat{H}$ such that $\widehat{H}\hat{\boldsymbol{d}} = H\boldsymbol{d}$ and

$$\|I - \widehat{H}^{-1/2} H \widehat{H}^{-1/2}\| \leqslant \delta, \tag{2.13}$$

where $\|\cdot\|$ denotes the matrix spectral norm.

We express (2.13) in an equivalent form, directly useful for our analysis.

Lemma 2.5. *Assume H and $\widehat{H}$ are symmetric, positive definite $m \times m$ matrices. Then* (2.13) *is equivalent to the inequalities*

$$(1-\delta)(\widehat{H}\boldsymbol{w}, \boldsymbol{w}) \leqslant (H\boldsymbol{w}, \boldsymbol{w}) \leqslant (1+\delta)(\widehat{H}\boldsymbol{w}, \boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^m.$$

Proof. As H and $\widehat{H}$ are symmetric, positive definite $m \times m$ matrices, $I - \widehat{H}^{-1/2} H \widehat{H}^{-1/2}$ is symmetric. Thus, (2.13) is equivalent to the inequalities

$$-\delta(\boldsymbol{w}, \boldsymbol{w}) \leqslant ((I - \widehat{H}^{-1/2} H \widehat{H}^{-1/2})\boldsymbol{w}, \boldsymbol{w}) \leqslant \delta(\boldsymbol{w}, \boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^m,$$

or

$$(1-\delta)(\boldsymbol{w}, \boldsymbol{w}) \leqslant (\widehat{H}^{-1/2} H \widehat{H}^{-1/2}\boldsymbol{w}, \boldsymbol{w}) \leqslant (1+\delta)(\boldsymbol{w}, \boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^m.$$

Replacing $\boldsymbol{w}$ by $\widehat{H}^{1/2}\boldsymbol{w}$ we obtain the result. □

Applying Lemmas 2.4 and 2.5 to (2.11), we conclude that there exists a symmetric, positive definite matrix $\hat{A}$ such that ($\boldsymbol{d} = A^{-1}\boldsymbol{\phi}$, $H = A$, $\hat{\boldsymbol{d}} = \boldsymbol{\psi}(\boldsymbol{\phi})$, $\widehat{H} = \hat{A}$)

$$\hat{A}\boldsymbol{\psi}(\boldsymbol{\phi}) = \boldsymbol{\phi},$$

and

$$(1-\delta)(\hat{A}\boldsymbol{w}, \boldsymbol{w}) \leqslant (A\boldsymbol{w}, \boldsymbol{w}) \leqslant (1+\delta)(\hat{A}\boldsymbol{w}, \boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^m.$$

Define a preconditioner of the matrix A by scaling the matrix $\hat{A}$,

$$\widetilde{Q}_A = (1+\delta)\hat{A}.$$

Then

$$(1-\delta')(\widetilde{Q}_A\boldsymbol{w}, \boldsymbol{w}) \leqslant (A\boldsymbol{w}, \boldsymbol{w}) \leqslant (\widetilde{Q}_A\boldsymbol{w}, \boldsymbol{w}) \quad \forall \boldsymbol{w} \in \mathbb{R}^m,$$

with $\delta' = 2\delta/(1+\delta)$. So the first equation of (2.12) is the same as

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n - \widetilde{Q}_A^{-1}(\boldsymbol{s}^n),$$

and with $\delta' = 2\delta/(1+\delta)$ and

$$(1-\delta')(\widetilde{Q}_A \boldsymbol{v}, \boldsymbol{v}) \leqslant (A\boldsymbol{v}, \boldsymbol{v}) \leqslant (\widetilde{Q}_A \boldsymbol{v}, \boldsymbol{v}).$$

Define the symmetric bilinear form

$$\tilde{b}(u_h, v_h) = (\widetilde{Q}_A \boldsymbol{u}, \boldsymbol{v})$$

for

$$u_h = \sum_{i=1}^{m} u_i \phi_i, \quad \boldsymbol{u} = (u_1, \ldots, u_m)^{\mathrm{T}},$$
$$v_h = \sum_{i=1}^{m} v_i \phi_i, \quad \boldsymbol{v} = (v_1, \ldots, v_m)^{\mathrm{T}}.$$

Then we have the inequalities

$$a(v_h, v_h) \leqslant \tilde{b}(v_h, v_h) \leqslant \omega' a(v_h, v_h) \quad \forall v_h \in V_h$$

with

$$\omega' = \frac{1+\delta}{1-\delta}.$$

Thus the non-linear Uzawa algorithm is equivalent to the following linear inexact Uzawa algorithm:

Algorithm 2.6. Let $\boldsymbol{\lambda}_h^0 \in \Lambda$ and $u_h^0 \in V_h$. For $n = 0, 1, 2, \ldots$, we compute $u_h^{n+1} \in V_h$ and $\boldsymbol{\lambda}_h^{n+1}$ by

$$\tilde{b}(u_h^{n+1}, v_h) = \tilde{b}(u_h^n, v_h) - [a(u_h^n, v_h) + (\boldsymbol{\lambda}_h^n, \mathscr{L} v_h)_{L^2(D)^{d_1}} - l(v_h)] \quad \forall v_h \in V_h,$$
$$\boldsymbol{\lambda}_h^{n+1} = \mathscr{P}_\Lambda(\boldsymbol{\lambda}_h^n + \rho \mathscr{L} u_h^{n+1}).$$

Applying Theorem 2.2, we obtain

Theorem 2.7. *Assume* (2.11). *Then if* $0 \leqslant \delta < 2/3$ *and* $0 < \rho < (2-3\delta)/((1-\delta)c_0)$, *we have the convergence for Algorithm* 2.3:

$$\lim_{n\to\infty} \|u_h^n - u_h\|_V = 0.$$

Remark 2.8. The condition $\delta < 2/3$ is satisfied if a sufficient number of PCG iterations are applied.

We now apply the above results to a flow problem.

Example 2.9. Let $\Omega \subset \mathbb{R}^2$ be a bounded domain and let μ, g be given positive parameters. Over the space $V = H_0^1(\Omega)$, define

$$a(u, v) = \mu \int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}x,$$
$$j(v) = g \int_\Omega |\nabla v| \, \mathrm{d}x, \quad |\nabla v| = \sqrt{(\partial_{x_1} v)^2 + (\partial_{x_2} v)^2},$$
$$l(v) = \int_\Omega f v \, \mathrm{d}x.$$

We consider the corresponding variational inequality: Find $u \in V$ such that

$$a(u, v-u) + j(v) - j(u) \geqslant l(v-u) \quad \forall v \in V. \tag{2.14}$$

If $l(v) = C\int_\Omega v\,\mathrm{d}x$, then (2.14) models the laminar stationary flow of a Bingham fluid in a cylindrical pipe of cross section Ω. The constant C is the linear decay of pressure and μ, g are the viscosity and plasticity yield of the fluid [12].

The finite element method for the variational inequality (2.14) is: Find $u_h \in V_h$ such that

$$a(u_h, v_h - u_h) + j(v_h) - j(u_h) \geqslant l(v_h - u_h) \quad \forall v_h \in V_h. \tag{2.15}$$

It is easy to see that (1.4) holds with $c_0 = g^2/\mu$. So we can apply Theorems 2.2 and 2.7 to conclude convergence of the linear and non-linear inexact Uzawa algorithms for solving the discrete variational inequality (2.15).

3. Discretization with numerical integration

Notice that the $j(\cdot)$ term is non-differentiable. In finite element computations, it is convenient to replace this term by $j_h(\cdot)$ obtained through a numerical integration procedure. In this section, we take a simplified friction problem as a model and show that the convergence theory for inexact Uzawa algorithms developed in the previous section can be extended to this case.

Let Ω be a bounded polygonal domain of $\mathbb{R}^2$. Decompose the boundary $\partial\Omega$ into two subsets $\overline{\Gamma_0}$ and $\overline{\Gamma_1}$ with Γ_0 and Γ_1 relatively open in $\partial\Omega$. Over the space

$$V = H^1_{\Gamma_0}(\Omega) = \{v \in H^1(\Omega) : v = 0 \text{ a.e. on } \Gamma_0\},$$

we define

$$a(u,v) = \int_\Omega (\nabla u \cdot \nabla v + uv)\,\mathrm{d}x,$$

$$j(v) = g\int_{\Gamma_1} |v|\,\mathrm{d}s, \quad g > 0,$$

$$l(v) = \int_\Omega fv\,\mathrm{d}x.$$

The corresponding problem (1.1) is a simplified version of the friction problem in elasticity (cf. [9,12]). We assume $\{\mathscr{T}_h\}$ is a regular family of triangulations of $\overline{\Omega}$ such that if $T \in \mathscr{T}_h$ has a non-empty intersection with Γ_0, then $T \cap \overline{\Gamma_0}$ is an entire side of T. Define the corresponding linear element spaces:

$$V_h = \{v_h \in C^0(\overline{\Omega}) : v_h|_T \in P_1(T)\ \forall T \in \mathscr{T}_h, v_h = 0 \text{ on } \Gamma_0\}.$$

We use the composite trapezoidal rule to evaluate j:

$$j_h(v_h) = g\sum_{e\in E_b} \frac{h_e}{2}(|v_h(\boldsymbol{x}_e^1)| + |v_h(\boldsymbol{x}_e^2)|),$$

where E_b is the set of all edges of the elements on Γ_1, $\boldsymbol{x}_e^1$, $\boldsymbol{x}_e^2$ are the two endpoints of an edge e. The finite element solution is then defined by

$$u_h \in V_h, \quad a(u_h, v_h) + j_h(v_h) - j_h(u_h) \geqslant l(v_h - u_h) \quad \forall v_h \in V_h. \tag{3.1}$$

Let $M_h = \{w_h : |w_h(\boldsymbol{x}_f)| \leqslant 1\ \forall \boldsymbol{x}_f \in N_b\}$, N_b being the set of all the nodes on Γ_1, and define a discrete inner product

$$(\lambda_h, v_h)_h = g\sum_{e\in E_b} \frac{h_e}{2}(\lambda_h(\boldsymbol{x}_e^1)v_h(\boldsymbol{x}_e^1) + \lambda_h(\boldsymbol{x}_e^2)v_h(\boldsymbol{x}_e^2)).$$

Then (3.1) is characterized by $u_h \in V_h$ and $\lambda_h \in M_h$ such that

$$a(u_h, v_h) + (\lambda_h, v_h)_h = l(v_h) \quad \forall v_h \in V_h,$$
$$\lambda_h(\boldsymbol{x}_f)u_h(\boldsymbol{x}_f) = |u_h(\boldsymbol{x}_f)| \quad \forall \boldsymbol{x}_f \in N_b.$$

Let $\lambda_h^0 \in \Lambda$ and $u_h^0 \in V_h$ be given. Denote the projection operator on $\mathbb{R}$ to $[-1, 1]$:

$$\mathscr{P}_0 t = \frac{t}{\sup(1, |t|)}, \quad t \in \mathbb{R}.$$

For $n = 0, 1, 2, \ldots$, we can then apply the following variant of Algorithm 2.1:

$$u_h^{n+1} \in V_h, \quad a_0(u_h^{n+1}, v_h) = a_0(u_h^n, v_h) - \left[a(u_h^n, v_h) + (\lambda_h^n, \mathscr{L}v_h)_h - l(v_h)\right] \quad \forall v_h \in V_h, \tag{3.2}$$

$$\lambda_h^{n+1}(\boldsymbol{x}_f) = \mathscr{P}_0(\lambda_h^n(\boldsymbol{x}_f) + \rho u_h^{n+1}(\boldsymbol{x}_f)) \quad \forall \boldsymbol{x}_f \in N_b, \tag{3.3}$$

or the following variant of Algorithm 2.3:

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n - \frac{1}{1+\delta}\boldsymbol{\psi}(\boldsymbol{s}^n), \tag{3.4}$$

$$\lambda_h^{n+1}(\boldsymbol{x}_f) = \mathscr{P}_0(\lambda_h^n(\boldsymbol{x}_f) + \rho u_h^{n+1}(\boldsymbol{x}_f)) \quad \forall \boldsymbol{x}_f \in N_b, \tag{3.5}$$

where $\boldsymbol{u}^n = (u_1^n, \ldots, u_m^n)^{\mathrm{T}}$, $u_h^n = \sum_{j=1}^m u_j^n \phi_j \in V_h$ and the vector $\boldsymbol{s}^n$ has the components

$$s_j^n = a(u_h^n, \phi_j) + (\boldsymbol{\lambda}_h^n, \mathscr{L}\phi_j)_h - l(\phi_j), \quad j = 1, 2, \ldots, m.$$

Convergence of these inexact Uzawa algorithms does not follow directly from Theorems 2.2 and 2.7. However, we can present the following arguments.

The following trace inequality is well-known:

$$\int_{\Gamma_1} v^2 \, \mathrm{d}s \leqslant \hat{c}_0 \int_\Omega (|\nabla v|^2 + v^2) \, \mathrm{d}x \quad \forall v \in H^1(\Omega),$$

where $\hat{c}_0 = 1/\lambda_{\min}$ and $\lambda_{\min} > 0$ is the smallest eigenvalue of the problem

$$-\Delta u + u = 0 \quad \text{in } \Omega,$$
$$u = 0 \quad \text{on } \Gamma_0,$$
$$\frac{\partial u}{\partial n} = \lambda u \quad \text{on } \Gamma_1.$$

We have

$$(\mathscr{L}v_h, \mathscr{L}v_h)_h = g\sum_{e \in E_b} \frac{h_e}{2}(v_h(\boldsymbol{x}_e^1)^2 + v_h(\boldsymbol{x}_e^2)^2) \leqslant \frac{3g}{2}\int_\Omega v_h(s)^2 \, \mathrm{d}s \leqslant \frac{3g\hat{c}_0}{2} a(v_h, v_h).$$

Thus we can modify the proofs of Theorems 2.2 and 2.7 in a straightforward way (replacing $(\cdot,\cdot)_{L^2(D)^{d_1}}$ and $\|\cdot\|^2_{L^2(D)^{d_1}}$ by $(\cdot,\cdot)_h$ in the proofs) to conclude convergence of the linear and non-linear inexact Uzawa algorithms (3.2), (3.3) and (3.4), (3.5), under the conditions stated in Theorems 2.2 and 2.7 with $c_0 = 3g\hat{c}_0/2$.

4. Numerical experiments

In this section we present some numerical results for solving the discrete system (2.15). We choose $\Omega = (0, 1) \times (0, 1)$ be the unit square and $\mu = g = 1$, $l(v) = \int_\Omega v \, \mathrm{d}x$. Let $\mathscr{T}_h$ be the uniform triangulation of

Table 1
CPU time (s)

	$h = 1/20$	$h = 1/30$	$h = 1/40$
CPU time I	64.95	506.59	3120.30
CPU time II	54.63	360.73	1580.50

Table 2
Inexact Uzawa algorithm, $h = 1/20$

ρ	1.5	1.5	1.5	1.5	1.0	1.0	1.0	1.0
PCG inner iterations	25	20	15	10	25	20	15	10
Outer iterations	67	67	68	66	87	102	84	77
CPU time	56.95	54.99	52.73	49.30	69.04	74.70	61.26	54.81

Ω, dividing Ω into $2(n+1)^2$ triangles. Denote $h = 1/(n+1)$. We compare the numerical results for the Uzawa algorithm (1.13) and (1.14) (denoted by I) and the inexact non-linear Uzawa algorithm (2.12) (denoted by II), with 20 steps of PCG method for Ψ and $\delta = 0.5$ in (2.12). The stop criterion is $\|u_h^n - u_h^{n-1}\|_0 \leqslant 10^{-5}$. Let $\rho = 1.5$.

It is evident from Table 1 that as the size of the discrete system gets larger, it is more efficient to use the inexact Uzawa algorithm.

For the inexact Uzawa algorithm, we have done some experiments for the effect of the number of PCG inner iterations and the value of the parameter ρ on the performance of the method. The results are summarized in Table 2.

References

[1] K. Arrow, L. Hurwicz, H. Uzawa, Studies in Linear and Nonlinear Programming, Stanford University Press, Stanford, 1958.
[2] C. Baiocchi, A. Capelo, Variational and Quasivariational Inequalities: Applications to Free-Boundary Problems, John Wiley, Chichester, 1984.
[3] R. Bank, B. Welfert, H. Yserentant, A class of iterative methods for solving saddle point problems, Numer. Math. 56 (1990) 645–666.
[4] J. Bramble, J. Pasciak, A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems, Math. Comput. 50 (1988) 1–18.
[5] J.H. Bramble, J.E. Pasciak, A.T. Vassilev, Analysis of the inexact Uzawa algorithm for saddle point problems, SIAM J. Numer. Anal. 34 (1997) 1072–1092.
[6] J.H. Bramble, J.E. Pasciak, A.T. Vassilev, Uzawa type algorithms for nonsymmetric saddle point problems, Math. Comput. 69 (1999) 667–689.
[7] F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Methods, Springer-Verlag, New York, 1991.
[8] X.L. Cheng, On the nonlinear inexact Uzawa algorithm for saddle point problems, SIAM J. Numer. Anal. 37 (2000) 1930–1934.
[9] G. Duvaut, J.-L. Lions, Inequalities in Mechanics and Physics, Springer-Verlag, Berlin, 1976.
[10] H. Elman, G. Golub, Inexact and preconditioned Uzawa algorithms for saddle point problems, SIAM J. Numer. Anal. 31 (1994) 1645–1661.
[11] A. Friedman, Variational Principles and Free-boundary Problems, John Wiley, New York, 1982.
[12] R. Glowinski, Numerical Methods for Nonlinear Variational Problems, Springer-Verlag, New York, 1984.
[13] R. Glowinski, J.-L. Lions, R. Trémolières, Numerical Analysis of Variational Inequalities, North-Holland, Amsterdam, 1981.
[14] R. Glowinski, P.L. Tallec, An Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics, SIAM, Philadelphia, 1989.
[15] V. Girault, P.A. Raviart, Finite Element Methods for Navier–Stokes Equations, Springer-Verlag, Berlin, 1986.
[16] W. Han, B.D. Reddy, Plasticity: Mathematical Theory and Numerical Analysis, Springer-Verlag, New York, 1999.
[17] W. Han, M. Sofonea, Quasistatic Contact Problems in Viscoelasticity and Viscoplasticity, American Mathematical Society and International Press, Providence, RI, 2002.

[18] R.H.W. Hoppe, Multigrid algorithms for variational inequalities, SIAM J. Numer. Anal. 24 (1987) 1046–1065.
[19] R.H.W. Hoppe, R. Kornhuber, Adaptive multilevel methods for obstacle problems, SIAM J. Numer. Anal. 31 (1994) 301–323.
[20] Q. Hu, J. Zou, An iterative method with variable relaxation parameters for saddle-point problems, SIAM J. Matrix Anal. Appl. 23 (2001) 317–338.
[21] N. Kikuchi, J.T. Oden, Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Element Methods, SIAM, Philadelphia, 1988.
[22] D. Kinderlehrer, G. Stampacchia, An Introduction to Variational Inequalities and their Applications, Academic Press, New York, 1980.
[23] R. Kornhuber, Monotone multigrid methods for elliptic variational inequalities I, Numer. Math. 69 (1994) 167–184.
[24] R. Kornhuber, Monotone multigrid methods for elliptic variational inequalities II, Numer. Math. 72 (1996) 481–499.
[25] P.D. Panagiotopoulos, Inequality Problems in Mechanics and Applications, Birkhäuser, Boston, 1985.
[26] W. Queck, The convergence factor of preconditioned algorithms of the Arrow–Hurwicz type, SIAM J. Numer. Anal. 26 (1989) 1016–1030.
[27] J.F. Rodrigues, Obstacle Problems in Mathematical Physics, North-Holland, Amsterdam, 1987.
[28] T. Rusten, R. Winther, A preconditioned iterative method for saddle point problems, SIAM J. Matrix Anal. Appl. 13 (1992) 887–904.
[29] J. Schöberl, Efficient contact solvers based on domain decomposition techniques, Comp. Math. Appl. 42 (2001) 1217–1228.