Electronic Research Archive

http://www.aimspress.com/journal/era

ERA, 31(2): 1048–1064.
DOI: 10.3934/era.2023052
Received: 06 October 2022
Revised: 24 November 2022
Accepted: 30 November 2022
Published: 13 December 2022

Theory article

A numerical method for parabolic complementarity problem

Haiyan Song[1] **and Fei Sun**[2,*]

[1] School of Computer and Data Engineering, NingboTech University, Ningbo 315100, China
[2] School of Mathematics and Computational Science, Wuyi University, Jiangmen 529020, China

* **Correspondence:** Email: fsun@wyu.edu.cn.

Abstract: In this paper, we study the numerical solution of a parabolic complementarity problem which is a widely used model in many fields, such as option pricing, risk measures, etc. Using a power penalty method we represent the complementarity problem as a nonlinear parabolic partial differential equation (PDE). Then, we use the trapezoidal rule as the time discretization, for which we have to solve a nonlinear equation at each time step. We solve such a nonlinear equation by the fixed-point iteration and in this methodology solving a tridiagonal linear system is the major computation. We present an efficient backward substitution algorithm to handle this linear system. Numerical results are given to illustrate the advantage of the proposed algorithm (compared to the built-in command backslash in Matlab) in terms of CPU time.

Keywords: complementarity problem; discretization; trapezoidal rule; penalty method; risk measures

1. Introduction

We are interested in the following parabolic complementarity problem

$$\begin{cases} \frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(a(x)\frac{\partial u}{\partial x}\right) \le g(x,t), \\ u(x,t) - u^*(x,t) \le 0, \\ \left(\frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(a(x)\frac{\partial u}{\partial x}\right) - g(x,t)\right) \cdot (u(x,t) - u^*(x,t)) = 0, \end{cases} \tag{1.1}$$

where $x \in \Omega \subset \mathbb{R}$, $t \in (0,T)$ and $a(x) > 0$. The functions u^* and g are known and the initial and boundary conditions for the unknown solution u is

$$u(x,t) = 0 \text{ for } (x,t) \in \partial\Omega \times (0,T),\ u(x,0) = u_0(x) \text{ for } x \in \Omega. \tag{1.2}$$

This kind of dynamic problem arises in many real-world applications, such as engineering, stochastic control, mechanics, economics, and risk measures (see, e.g., [1–8]). In general, it is impossible

to obtain an analytical solution of (1.1) except for some special cases and hence we have to rely on numerical methods in practice.

There are many numerical methods for the problem (1.1) and were studied by various authors. A popular approach is to reformulate the problem as a parabolic PDE using the linear penalty term of the form $\mu [p - u^*]_+$, where $\mu \gg 1$ is a constant, p denotes the solution to the penalized equation and $[a]_+ := \max\{a, 0\}$. This method has been discussed in [2, 3, 9–12]. It has been proved in [2] that the solution of the penalized equation converges to the original one at a rate of $O\left(\mu^{-\frac{1}{2}}\right)$. In [10, 13, 14] power penalty methods have been used for solving constrained optimization problems by using an improved penalty term

$$\mu [p(x,t) - u^*(x,t)]_+^{\frac{1}{k}} = 0, \tag{1.3}$$

where $k \geq 1$ is a constant. (This includes the linear penalty $k = 1$ mentioned above as the special case.) The authors proved in [10] that p converges to the exact solution u in a proper Sobolev norm at a rate of $O\left(\mu^{-\frac{k}{2}}\right)$. However, the penalty term (1.3) has an unbounded derivative when $p - u^* \to 0^+$, and therefore it needs to be smoothed locally [10]. In [11], the authors generalized the penalty term in (1.3) to the form of

$$\mu \left([p(x,t) - u^*(x,t)]_+ + \epsilon\right)^{\frac{1}{k}} = 0, \tag{1.4}$$

where $1 \gg \epsilon > 0$ is a smoothing parameter. The authors proved that the penalized solution converges to that of the original one at a rate of $O\left(\left[\mu^{-k} + \epsilon\left(1 + \mu\epsilon^{\frac{1}{k}}\right)\right]^{1/2}\right)$.

The aim of this paper is to study the numerical solution of the following penalized equation

$$\begin{cases} \frac{\partial p}{\partial t} - \frac{\partial}{\partial x}\left(a(x)\frac{\partial p}{\partial x}\right) + \mu\left([p - u^*]_+ + \epsilon\right)^{\frac{1}{k}} = g + \mu\epsilon^{\frac{1}{k}}, & (x,t) \in \Omega \times (0,T), \\ p(x,t) = 0, & (x,t) \in \partial\Omega \times (0,T), \\ p(x,0) = u_0(x), & x \in \Omega. \end{cases} \tag{1.5}$$

In the following, we let $f = g + \mu\epsilon^{\frac{1}{k}}$. We will first discretize (1.5) in space by using the centered finite difference method [15]. This leads to large-scale nonlinear ordinary differential equations, for which we use the trapezoidal rule [16, 17] for time discretization. Such a time discretization avoids solving nonlinear equations, but at each time step, we have to handle a large-scale tridiagonal linear system, which is the major computation. By looking for the structure of the coefficient matrix we propose a backward substitution algorithm for such a linear system. Numerically, we find that this algorithm is more efficient than the widely used built-in command '`backslash`' in Matlab in terms of CPU time.

The rest of this paper is organized as follows. In Section 2 we recall some results concerning the unique solvability of (1.5) and convergence of p to u. In Section 3 we present the space and time discretization for (1.5). We present the backward substitution algorithm in Section 4 for solving the large-scale tridiagonal linear system at each time step. Numerical results are given in Section 5 and we conclude this paper in Section 6.

2. Some preliminary results

In this section, we recall some results for the penalized parabolic equation (1.5) at the continuous level. To this end, we introduce some notations used in the following. With $1 \leq s \leq \infty$, we let

$\mathbb{L}^s(\Omega) = \left\{v : \left(\int_\Omega |v(x)|^s \, \mathrm{d}x\right)^{\frac{1}{k}} < \infty\right\}$ denote the space of all s-power integrable functions on Ω. The inner product on $\mathbb{L}^2(\Omega)$ is denoted by $(\cdot,\cdot)_\Omega$. We use $\|\cdot\|_{\mathbb{L}^s(\Omega)}$ to denote the norm on $\mathbb{L}^s(\Omega)$ and $\mathbb{H}^s(\Omega)$ to denote the Sobolev space with norm $\|\cdot\|_{k,\Omega}$. Let $\mathbb{C}^s(\Omega)$ (respectively, $\mathbb{C}^s(\bar{\Omega})$) be the function set of which a function and its derivatives of up to order s are continuous on Ω (respectively, $\bar{\Omega}$). Let $\mathbb{H}_0^s(\Omega) = \{v \in \mathbb{H}^s(\Omega) : v(x) = 0, x \in \partial\Omega\}$. For any Hilbert space $\mathbb{H}(\Omega)$, we use $\mathbb{L}^s(0,T;\mathbb{H}(\Omega))$ to denote the space

$$\mathbb{L}^s(0,T;\mathbb{H}(\Omega)) = \{v(\cdot,t) : v(\cdot,t) \in \mathbb{H}(\Omega) \text{ a.e. in } (0,T); \|v(\cdot,t)\|_{\mathbb{H}} \in \mathbb{L}^s(0,T)\},$$

where $1 \le k \le \infty$ and $\|\cdot\|_{\mathbb{H}}$ denotes the natural norm on $\mathbb{H}(\Omega)$. The norm in this space is denoted by $\|\cdot\|_{\mathbb{L}^s(0,T;\mathbb{H})}$, i.e.,

$$\|v\|_{L^s(0,T;H(\Omega))} = \left(\int_0^T \|v(\cdot,t)\|_H^s \, \mathrm{d}t\right)^{\frac{1}{s}}.$$

We use $\mathbb{H}^{-1}(\Omega)$ to denote the dual space of $\mathbb{H}^1(\Omega)$ and use $\langle\cdot,\cdot\rangle$ to denote the duality pair between a Hilbert space and its dual space.

By an integration by parts, it is clear that the variational problem corresponding to (1.5) is the following problem: *find* $p(t) \in \mathbb{H}_0^1(\Omega)$ *such that for all* $v \in \mathbb{H}_0^1(\Omega)$ *it holds*

$$\left(-\frac{\partial p(t)}{\partial t}, v\right) + (a(x)\partial_x p(t), \partial_x v) + \mu\left(\left([p(t) - u^*]_+ + \epsilon\right)^{\frac{1}{k}}, v\right) = (f, v), \tag{2.1}$$

with the initial condition $p(x,0) = u_0(x)$ in Ω and $f = g + \mu\epsilon^{\frac{1}{k}}$.

Theorem 1. *For* $a(x) > 0$*, problem* (2.1) *has a unique solution for fixed* $\epsilon \ge 0$ *and* $\mu \ge 1$.

Proof. Clearly, problem (2.1) is equivalent to

$$\left(-\frac{\partial p(t)}{\partial t}, v\right) + (a(x)\partial_x p(t), \partial_x v) + \mu\left(\Phi_\epsilon\left(p(t)\right), v\right) = \left(f - \mu(\phi(0) + \epsilon)^{\frac{1}{k}}, v\right),$$

where $\Phi_\epsilon(w) = (\phi(w) + \epsilon)^{\frac{1}{k}} - (\phi(0) + \epsilon)^{\frac{1}{k}}$. Clearly,

$$\Phi_\epsilon(0) = (\phi(0) + \epsilon)^{\frac{1}{k}} - (\phi(0) + \epsilon)^{\frac{1}{k}} = 0,$$

and $\Phi_\epsilon(w)$ is a monotonically increasing function of w because of the monotonicity of $([w - u^*]_+ + \epsilon)^{\frac{1}{k}}$ in w. This implies

$$\langle\Phi_\epsilon(v) - \Phi_\epsilon(w), v - w\rangle \ge 0, \ \forall v, w \in \mathbb{H}_0^1(\Omega). \tag{2.2a}$$

Since $a(x) > 0$, it holds

$$(a(x)\nabla v, \nabla v) \ge a_0(\nabla v, \nabla v) = a_0\|\nabla v\|_0 \ge C\|v\|_1, \tag{2.2b}$$

where we have used the Poincare-Friedrich inequality, i.e., $\|\nabla v\|_0 \ge C\|v\|_1 (\forall v \in \mathbb{H}_0^1(\Omega))$. By integration by parts, we therefore have

$$\begin{aligned} &\langle(-\partial_x(a(x)\partial_x v)) - (-\partial_x(a(x)\partial_x w)), v - w\rangle \\ &= \langle a(x)(\partial_x v - \partial_x w), \partial_x(v - w)\rangle \ge 0, \end{aligned} \tag{2.2c}$$

which holds for any $v, w \in \mathbb{H}_0^1(\Omega)$. From (2.2a) and (2.2b) we have

$$\langle \mu\Phi_\epsilon(v) - \partial_x(a(x)\partial_x v)) - (\mu\Phi_\epsilon(w) - \partial_x(a(x)\partial_x w))\,, v - w\rangle \geq 0. \tag{2.2d}$$

Similarly, by using (2.2b) it holds

$$\langle \mu\Phi_\epsilon(v) - \partial_x(a(x)\partial_x v)), v\rangle \geq C\|v\|_1^2 + \mu\,(\Phi_\epsilon(v), v) \geq C\|v\|_1^2, \tag{2.3}$$

because $(\Phi_\epsilon(v), v) = (\Phi_\epsilon(v) - \Phi_\epsilon(0), v - 0) \geq 0$.

From the definitions of Φ_ϵ we have

$$\begin{aligned}
(\Phi_\epsilon(v), w) &= \int_\Omega ([v - u^*]_+ + \epsilon)^{\frac{1}{k}}\, w\, \mathrm{d}x - \int_\Omega ([-u^*]_+ + \epsilon)^{\frac{1}{k}} w\, \mathrm{d}x \\
&\leq \left[\left(\int_\Omega ([v - u^*]_+ + \epsilon)^{2/k}\ \mathrm{d}x\right)^{1/2} + C_1(t)\right]\|w\|_0 \\
&\leq C\left[\left(\int_\Omega ([v - u^*]_+ + \epsilon)^{2}\ \mathrm{d}x\right)^{1/2k} + C_1(t)\right]\|w\|_0 \\
&= C\left(\|[v - u^*]_+ + \epsilon\|_0^{\frac{1}{k}} + C_1(t)\right)\|w\|_0\Big] \leq (C_2\|v\|_0 + C_1(t))\,\|w\|_0,
\end{aligned}$$

where we have used $\|v\|_0^{\frac{1}{k}} \leq \max\{1, \|v\|_0\}$ for $k \geq 1$. Here, $C_1 \in \mathbb{L}^2(0, T)$ is a generic positive function of t and $C_2 > 0$ a generic constant independent of ϵ and k. This gives

$$\langle \mu\Phi_\epsilon(v) - \partial_x(a(x)\partial_x v)), w\rangle \leq \|w\|_1\,(C_1(t) + C_2\|v\|_1)\,, \forall v, w \in \mathbb{H}_0^1(\Omega).$$

Dividing both sides of this inequality by $\|w\|_1$ and taking the supremum with respect to w we obtain

$$\|\mu\Phi_\epsilon(v) - \partial_x(a(x)\partial_x v))\|_{\mathbb{H}^{-1}(\Omega)} \leq C_1(t) + C_2\|v\|_1. \tag{2.4}$$

Now, by using (2.2d), (2.3) and (2.4) the unique existence of the solution of (2.1) is guaranteed by the theory established in [4, 18].

We now present the convergence of the approximate solution p of the penalized problem (1.5) to that of the original problem (1.1).

Theorem 2. *Let $a(x) > 0$ and $\frac{\partial u}{\partial t} \in \mathbb{L}^{k+1}(\Omega)$. Then there exists a constant $C > 0$, independent of u, p and μ such that*

$$\|u - p\|_{\mathbb{L}^\infty(0,T;\mathbb{L}^2(\Omega))} + \|u - p\|_{\mathbb{L}^2(0,T;\mathbb{H}_0^1(\Omega))} \leq C\left[\frac{1}{\mu^k} + \epsilon\left(\mu\epsilon^{\frac{1}{k}} + 1\right)\right]^{1/2}, \tag{2.5}$$

where μ, k and ϵ are the parameters used in (1.5).

Proof. In the asymptotic sense (i.e., $\epsilon \to 0$) it is clear that (1.5) is equivalent to

$$\frac{\partial p}{\partial t} - \frac{\partial}{\partial x}\left(a(x)\frac{\partial p}{\partial x}\right) + \mu\,([p - u^*]_+ + \epsilon)^{\frac{1}{k}} = g. \tag{2.6}$$

(Here we have used $f = g + \mu\epsilon^{\frac{1}{k}}$.) In [3, 11, 12] it was proved that the solution of (2.6) and the solution of the original problem (1.1) satisfy (2.5). Since

$$\mu\epsilon^{\frac{1}{k}} \leq \epsilon^{\frac{1}{2}}\left(\mu\epsilon^{\frac{1}{k}} + 1\right)^{\frac{1}{2}},$$

it is clear that the solution of (1.5) and the solution of the original problem (1.1) satisfy (2.5) as well.

3. Space and time discretizations

We now present space and time discretizations for the penalized equation (1.5). To this end, we partition the computation domain $\Omega \times [0, T]$ by mesh sizes h and τ and denote an arbitrary grid on this domain by $(jh, n\tau)$, where $j = 0, 1, \ldots, J$ and $n = 0, 1, \ldots, N$. We first discretize the spatial derivative by the centered finite difference formula

$$\begin{aligned}\frac{\partial}{\partial x}\left(a(x)\frac{\partial p}{\partial x}\right)\Big|_{x=x_j} &= \frac{\frac{a_{j+1}p_{j+1}(t)-a_j p_j(t)}{h} - \frac{a_j p_j(t)-a_{j-1}p_{j-1}(t)}{h}}{h} + r_j \\ &= \frac{a_{j+1}p_{j+1}(t) - 2a_j p_j(t) + a_{j-1}p_{j-1}(t)}{h^2} + r_j,\end{aligned}$$

where $a_l = a(x_l)$, $p_l(t) \approx p(x_l, t)$ (with $l = j-1, j, j+1$), $r_j = O(h^2)$ is the truncation error and $j = 1, 2, \ldots, J-1$. For $j = 0$ and $j = J$ it holds $p_j(t) = 0$ according to the boundary condition in (1.5).

Dropping this truncation error in the above formulas gives the approximations as

$$\frac{\partial}{\partial x}\left(a(x)\frac{\partial p}{\partial x}\right)\Big|_{x=x_j} \approx \frac{a_{j+1}p_{j+1}(t) - 2a_j p_j(t) + a_{j-1}p_{j-1}(t)}{h^2}.$$

Substituting this into (1.5) gives the semi-discrete system as

$$\boldsymbol{p}'(t) + A\boldsymbol{p}(t) + \mu\left([\boldsymbol{p}(t) - \boldsymbol{u}^*(t)]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} = \boldsymbol{f}(t),\ t \in (0, T), \tag{3.1}$$

with initial value condition $\boldsymbol{p}(0) = \boldsymbol{u}_0$, where

$$\boldsymbol{p}(t) = \begin{bmatrix} p_1(t) \\ p_2(t) \\ \vdots \\ p_{J-1}(t) \end{bmatrix},\ \boldsymbol{u}^* = \begin{bmatrix} u^*(x_1, t) \\ u^*(x_2, t) \\ \vdots \\ u^*(x_{J-1}, t) \end{bmatrix},\ \boldsymbol{f}(t) = \begin{bmatrix} f(x_1, t) \\ f(x_2, t) \\ \vdots \\ f(x_{J-1}, t) \end{bmatrix},\ \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon \\ \epsilon \\ \vdots \\ \epsilon \end{bmatrix}. \tag{3.2a}$$

The Matrix A is a tridiagonal matrix

$$A = \frac{1}{h^2}\begin{bmatrix} 2a_1 & -a_2 & & & \\ -a_1 & 2a_2 & -a_3 & & \\ & \ddots & \ddots & \ddots & \\ & & -a_{J-3} & 2a_{J-2} & -a_{J-1} \\ & & & -a_{J-2} & 2a_{J-1} \end{bmatrix}. \tag{3.2b}$$

We now introduce time discretization for (3.1). To match the second-order accuracy of the space discretization, we need a second-order time discretization. To this end, we rewrite (3.1) as an integral equation in the interval $[t_n, t_{n+1}]$

$$\int_{t_n}^{t_{n+1}} \boldsymbol{p}'(s)ds = \int_{t_n}^{t_{n+1}} \left(\boldsymbol{f}(s) - A\boldsymbol{p}(s) - \mu\left([\boldsymbol{p}(s) - \boldsymbol{u}^*(s)]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}\right)ds,$$

i.e.,

$$\boldsymbol{p}(t_{n+1}) - \boldsymbol{p}(t_n) = \int_{t_n}^{t_{n+1}} \left(\boldsymbol{f}(s) - A\boldsymbol{p}(s) - \mu\left([\boldsymbol{p}(s) - \boldsymbol{u}^*(s)]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}\right)ds. \tag{3.3}$$

Let $\boldsymbol{f}(t) - A\boldsymbol{p}(t) - \mu\left([\boldsymbol{p}(t) - \boldsymbol{u}^*(t)]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} = (q_1(t), q_2(t), \ldots, q_{J-1}(t))^\top$. Then,

$$\int_{t_n}^{t_{n+1}} \left(\boldsymbol{f}(s) - A\boldsymbol{p}(s) - \mu\left([\boldsymbol{p}(s) - \boldsymbol{u}^*(s)]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}\right) ds = \begin{bmatrix} \int_{t_n}^{t_{n+1}} q_1(s)ds \\ \int_{t_n}^{t_{n+1}} q_2(s)ds \\ \vdots \\ \int_{t_n}^{t_{n+1}} q_{J-1}(s)ds \end{bmatrix}.$$

Let $q(t)$ be an arbitrary component of the vector function $(q_1(t), q_2(t), \ldots, q_{J-1}(t))^\top$. The integral $\int_{t_n}^{t_{n+1}} q(s)ds$ is the area of the trapezoid with a curved edge as shown in Figure 1 on the left. In general, it is impossible to get the precise value of such an area. To get an approximation of this area we can consider a regular trapezoid with the curved edge being replaced by a straight edge; see Figure 1 on the right. Then, we have

$$\int_{t_n}^{t_{n+1}} q(s)ds = \frac{p(t_n) + p(t_{n+1})}{2}(t_{n+1} - t_n) + \eta_n = \frac{p(t_n) + p(t_{n+1})}{2}\tau + \eta_n,$$

where $\eta_n = O(\tau^2)$ denotes the approximation error [15, Chapter 2].

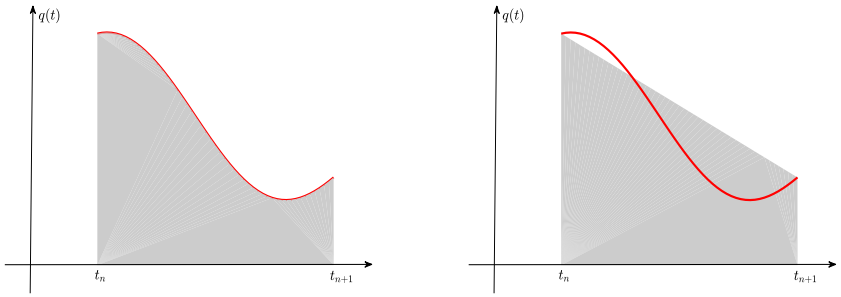

Figure 1. The trapezoidal rule lies in approximating the trapezoid with a curved edge by a regular one. The area of the regular trapezoid is $\frac{p(t_n)+p(t_{n+1})}{2}(t_{n+1} - t_n) = \frac{p(t_n)+p(t_{n+1})}{2}\tau$.

Dropping the approximation error and letting $\boldsymbol{p}_n \approx \boldsymbol{p}(t_n)$, we therefore obtain an approximation of the right hand-side of (3.3):

$$\begin{aligned} &\int_{t_n}^{t_{n+1}} \left(\boldsymbol{f}(s) - A\boldsymbol{p}(s) - \mu\left([\boldsymbol{p}(s) - \boldsymbol{u}^*(s)]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}\right) ds \\ &\approx \tau\tilde{\boldsymbol{f}}_n - \frac{\tau A}{2}(\boldsymbol{p}_n + \boldsymbol{p}_{n+1}) - \frac{\tau\mu}{2}\left(\left([\boldsymbol{p}_n - \boldsymbol{u}_n^*]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} + \left([\boldsymbol{p}_{n+1} - \boldsymbol{u}_{n+1}^*]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}\right), \end{aligned}$$

where $\tilde{\boldsymbol{f}}_n = \frac{f(t_n)+f(t_{n+1})}{2}$. Substituting this into (3.3) gives the full discretization of the penalized equation (1.5):

$$\boldsymbol{p}_{n+1} - \boldsymbol{p}_n = \tau\tilde{\boldsymbol{f}}_n - \frac{\tau A}{2}(\boldsymbol{p}_n + \boldsymbol{p}_{n+1}) - \frac{\tau\mu}{2}\left(\left([\boldsymbol{p}_n - \boldsymbol{u}_n^*]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} + \left([\boldsymbol{p}_{n+1} - \boldsymbol{u}_{n+1}^*]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}\right),$$

where $n = 0, 1, \ldots, N-1$. This is a nonlinear system due to the term

$$\left([\boldsymbol{p}_{n+1} - \boldsymbol{u}_{n+1}^*]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}.$$

To handle such a nonlinear system we propose a *fixed-point iteration* as follows. We can use Newton's method to handle such a nonlinear problem [19], but in this case, we have to compute the Jacobian matrix of the whole nonlinear term. First, we rewrite this system as

$$\begin{aligned} &\boldsymbol{p}_{n+1} + \frac{\tau A}{2}\boldsymbol{p}_{n+1} = \frac{\tau\mu}{2}\left(\left[\boldsymbol{p}_{n+1} - \boldsymbol{u}^*_{n+1}\right]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} + \boldsymbol{b}_n, \\ &\boldsymbol{b}_n := \tau\tilde{\boldsymbol{f}}_n + \boldsymbol{p}_n - \frac{\tau A}{2}\boldsymbol{p}_n - \frac{\tau\mu}{2}\left([\boldsymbol{p}_n - \boldsymbol{u}^*_n]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}}. \end{aligned} \tag{3.4}$$

(The vector $\boldsymbol{b}_n$ is a known term.) Then, we solve $\boldsymbol{p}_{n+1}$ via the following iterations

$$\begin{aligned} &\boldsymbol{p}^{[l+1]}_{n+1} + \frac{\tau A}{2}\boldsymbol{p}^{[l+1]}_{n+1} = \frac{\tau\mu}{2}\left(\left[\boldsymbol{p}^{[l]}_{n+1} - \boldsymbol{u}^*_{n+1}\right]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} + \boldsymbol{b}_n,\ l = 0, 1, \ldots, l_{\max} - 1, \\ &\boldsymbol{p}_{n+1} = \boldsymbol{p}^{[l_{\max}]}_{n+1}, \end{aligned} \tag{3.5}$$

where l is the iteration index and $l_{\max}$ is specified by the prescribed tolerance.

The following is the convergence analysis for the above fixed-point iteration.

Theorem 3. *Let $a(x) > 0$ for $x \in \Omega$, $\epsilon > 0$ and $k > 1$. Then the fixed-point iteration* (3.5) *converges to the unique solution with a rate at least*

$$\rho = \frac{\tau\mu a_{\max} h^2}{[2a_{\max}h^2 + (2 - 2\cos(h\pi))\tau]k\epsilon^{\frac{k-1}{k}}}, \tag{3.6}$$

if $\rho < 1$, where $a_{\max} = \max_{x\in\Omega} a(x)$.

Proof. From (3.5) we have

$$\boldsymbol{p}^{[l+1]}_{n+1} = \frac{\tau\mu}{2}\left(I_x + \frac{\tau A}{2}\right)^{-1}\left(\left[\boldsymbol{p}^{[l]}_{n+1} - \boldsymbol{u}^*_{n+1}\right]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} + \tilde{\boldsymbol{b}}_n,$$

where $\tilde{\boldsymbol{b}}_n = (I_x + \frac{\tau A}{2})^{-1}\boldsymbol{b}_n$ and $I_x \in \mathbb{R}^{(J-1)\times(J-1)}$ is an identity matrix. Let

$$Y(\boldsymbol{x}) = \left(\left[\boldsymbol{x} - \boldsymbol{u}^*_{n+1}\right]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}},\ F(\boldsymbol{x}) = \frac{\tau\mu}{2}\left(I_x + \frac{\tau A}{2}\right)^{-1} Y(\boldsymbol{x}) + \tilde{\boldsymbol{b}}_n. \tag{3.7}$$

Then, it is sufficient to study the contraction property of the function $F(\boldsymbol{x})$, i.e.,

$$\|F(\boldsymbol{x}_1) - F(\boldsymbol{x}_2)\|_\infty \le \rho\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_\infty, \tag{3.8}$$

where ρ is the quantity that we need to look insight into. The analysis of ρ is given as follows. It holds

$$F(\boldsymbol{x}_1) - F(\boldsymbol{x}_2) = \frac{\tau\mu}{2}\left(I_x + \frac{\tau A}{2}\right)^{-1}[Y(\boldsymbol{x}_1) - Y(\boldsymbol{x}_2)],$$

which implies

$$\|F(\boldsymbol{x}_1) - F(\boldsymbol{x}_2)\|_\infty \le \frac{\tau\mu}{2}\left\|\left(I_x + \frac{\tau A}{2}\right)^{-1}\right\|_\infty \|Y(\boldsymbol{x}_1) - Y(\boldsymbol{x}_2)\|_\infty. \tag{3.9}$$

From the structure of the matrix A (cf. (3.2b)) we have

$$I_x + \frac{\tau A}{2} = I_x + \frac{\tau}{2}\Lambda D_a,$$

where

$$\Lambda = \frac{1}{h^2}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}, D_a = \begin{bmatrix} a_1 & & & & \\ & a_2 & & & \\ & & \ddots & & \\ & & & a_{J-2} & \\ & & & & a_{J-1} \end{bmatrix}.$$

Let $\lambda(\cdot)$ denote an arbitrary eigenvalue of the involved matrix. It it clear

$$\lambda\left(I_x + \frac{\tau A}{2}\right) = 1 + \frac{\tau}{2}\lambda(\Lambda D_a).$$

Since $a(x) > 0$ we know that D_a is an invertible diagonal matrix with positive diagonal elements. Hence, by a similarity transform it holds

$$\lambda(\Lambda D_a) = \lambda(D_a^{\frac{1}{2}}(\Lambda D_a)D_a^{-\frac{1}{2}}) = \lambda(D_a^{\frac{1}{2}}\Lambda D_a^{\frac{1}{2}}), \tag{3.10a}$$

where $D_a^{\pm\frac{1}{2}} = \text{diag}(a_1^{\pm\frac{1}{2}}, a_2^{\pm\frac{1}{2}}, \ldots, a_{J-1}^{\pm\frac{1}{2}})$. The matrix $D_a^{\frac{1}{2}}\Lambda D_a^{\frac{1}{2}}$ is a symmetric positive definite matrix and thus it holds (see, e.g., [20, Chapter 5])

$$\min_{z\in\mathbb{R}^{J-1}} \frac{z^\top D_a^{\frac{1}{2}}\Lambda D_a^{\frac{1}{2}} z}{z^\top z} \le \lambda(D_a^{\frac{1}{2}}\Lambda D_a^{\frac{1}{2}}) \le \max_{z\in\mathbb{R}^{J-1}} \frac{z^\top D_a^{\frac{1}{2}}\Lambda D_a^{\frac{1}{2}} z}{z^\top z}. \tag{3.10b}$$

By letting $\tilde{z} = D_a^{\frac{1}{2}} z$ we have

$$\frac{z^\top D_a^{\frac{1}{2}}\Lambda D_a^{\frac{1}{2}} z}{z^\top z} = \frac{\tilde{z}^\top \Lambda \tilde{z}}{\tilde{z}^\top D_a^{-1}\tilde{z}}. \tag{3.11}$$

Since Λ is a symmetric positive definite matrix, from [20, Chapter 5] we have

$$\lambda_{\min}(\Lambda) \le \tilde{z}^\top \Lambda \tilde{z} \le \lambda_{\max}(\Lambda),\ \forall \tilde{z} \in \mathbb{R}^{J-1}.$$

From [21, 22] we know that the eigenvalues of the tridiagonal matrix Λ are given by

$$\lambda_j(\Lambda) = \frac{2 - 2\cos\left(\frac{j\pi}{J}\right)}{h^2},\ j = 1, 2, \ldots, J-1.$$

Since $J = \frac{1}{h}$, it holds

$$\lambda_{\min}(\Lambda) = \frac{2 - 2\cos(h\pi)}{h^2},\ \lambda_{\max}(\Lambda) = \frac{2 + 2\cos(h\pi)}{h^2}.$$

This gives

$$\frac{2 - 2\cos(h\pi)}{h^2} \le \tilde{z}^\top \Lambda \tilde{z} \le \frac{2 + 2\cos(h\pi)}{h^2},\ \forall \tilde{z} \in \mathbb{R}^{J-1}. \tag{3.12a}$$

Let $a_{\max} = \max_{x\in\Omega} a(x)$ and $a_{\min} = \min_{x\in\Omega} a(x)$. Then, it holds

$$a_{\max}^{-1}\tilde{z}^\top\tilde{z} \le \tilde{z}^\top D_a^{-1}\tilde{z} \le a_{\min}^{-1}\tilde{z}^\top\tilde{z},\ \forall \tilde{z} \in \mathbb{R}^{J-1}. \tag{3.12b}$$

Substituting (3.12a) and (3.12b) into (3.11) gives

$$\frac{2-2\cos(h\pi)}{h^2 a_{\max}} = \frac{\lambda_{\min}(\Lambda)}{a_{\max}} \le \frac{\boldsymbol{z}^\top D_a^{\frac{1}{2}} \Lambda D_a^{\frac{1}{2}} \boldsymbol{z}}{\boldsymbol{z}^\top \boldsymbol{z}} \le \frac{\lambda_{\max}(\Lambda)}{a_{\min}} = \frac{2+2\cos(h\pi)}{h^2 a_{\min}}.$$

This together with (3.10a) and (3.10b) gives

$$\lambda\left(I_x + \frac{\tau A}{2}\right) \in \left[1 + \frac{(2-2\cos(h\pi))\tau}{2a_{\max}h^2}, 1 + \frac{(2+2\cos(h\pi))\tau}{2a_{\min}h^2}\right]. \tag{3.13}$$

This indicates that $I_x + \frac{\tau A}{2}$ is an invertible matrix, since $\lambda_{\min}\left(I_x + \frac{\tau A}{2}\right) > 0$.

We next estimate of the infinity-norm of the inverse matrix $\left(I_x + \frac{\tau A}{2}\right)^{-1}$. By the definition of the infinity-norm of a square matrix, it holds

$$\left\|\left(I_x + \frac{\tau A}{2}\right)^{-1}\right\|_\infty = \max_{\boldsymbol{z}\in\mathbb{R}^{J-1}, \|\boldsymbol{z}\|_\infty = 1} \left\|\left(I_x + \frac{\tau A}{2}\right)^{-1} \boldsymbol{z}\right\|_\infty.$$

Since $\lambda\left(I_x + \frac{\tau A}{2}\right)$ is invertible, the eigenvectors of this matrix consist of a complete basis of the space $\mathbb{R}^{J-1}$. Then, without loss of generality, we assume that $\boldsymbol{z}$ is an arbitrary normalized eigenvector (with corresponding eigenvalue λ), i.e., $\left(I_x + \frac{\tau A}{2}\right)\boldsymbol{z} = \lambda \boldsymbol{z}$. Hence,

$$\left(I_x + \frac{\tau A}{2}\right)^{-1} \boldsymbol{z} = \lambda^{-1}\boldsymbol{z}.$$

This implies (cf. (3.13)).

$$\left\|\left(I_x + \frac{\tau A}{2}\right)^{-1} \boldsymbol{z}\right\|_\infty = |\lambda| \in \left[\frac{2a_{\min}h^2}{2a_{\min}h^2 + (2+2\cos(h\pi))\tau}, \frac{2a_{\max}h^2}{2a_{\max}h^2 + (2-2\cos(h\pi))\tau}\right].$$

In summary, we have

$$\left\|\left(I_x + \frac{\tau A}{2}\right)^{-1}\right\|_\infty \in \left[\frac{2a_{\min}h^2}{2a_{\min}h^2 + (2+2\cos(h\pi))\tau}, \frac{2a_{\max}h^2}{2a_{\max}h^2 + (2-2\cos(h\pi))\tau}\right]. \tag{3.14}$$

We next explore the relationship between $\|Y(\boldsymbol{x}_1) - Y(\boldsymbol{x}_2)\|_\infty$ and $\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_\infty$ with $Y(\boldsymbol{x})$ being the function defined by (3.7). Since

$$Y(\boldsymbol{x}) = \left(([x_1 - u_1^*]_+ + \epsilon)^{\frac{1}{k}}, ([x_2 - u_2^*]_+ + \epsilon)^{\frac{1}{k}}, \ldots, ([x_{J-1} - u_{J-1}^*]_+ + \epsilon)^{\frac{1}{k}}\right)^\top,$$

it is sufficient to study the contraction property of $y(x) := ([x]_+ + \epsilon)^{\frac{1}{k}}$ with $x \in \mathbb{R}$. We claim

$$|y(x_1) - y(x_2)| \le \frac{1}{k\epsilon^{\frac{k-1}{k}}}|x_1 - x_2|. \tag{3.15}$$

We consider three cases as shown in Figure 2. For the case $x_1 \le 0$ and $x_2 \le 0$, it is clear that (3.15) holds. For $x_1 \le 0$ and $x_2 > 0$ (i.e., the middle case in Figure 2), it holds

$$|y(x_1) - y(x_2)| = y(x_2) - y(0) = y'(\xi)x_2 = \frac{x_2}{k(\xi+\epsilon)^{\frac{k-1}{k}}} \le \frac{x_2 - x_1}{k\epsilon^{\frac{k-1}{k}}} = \frac{|x_2 - x_1|}{k\epsilon^{\frac{k-1}{k}}},$$

where $\xi \in (0, x_2)$. It remains to consider $x_2 > x_1 > 0$ (cf. Figure 2 on the right). We have

$$|y(x_1) - y(x_2)| = y(x_2) - y(x_1) = y'(\xi)(x_2 - x_1) = \frac{x_2 - x_1}{k(\xi + \epsilon)^{\frac{k-1}{k}}} \le \frac{x_2 - x_1}{k\epsilon^{\frac{k-1}{k}}} = \frac{|x_2 - x_1|}{k\epsilon^{\frac{k-1}{k}}}.$$

In summary, for all the three cases shown in Figure 2 the estimate (3.15) holds. Hence,

$$\|Y(\boldsymbol{x}_1) - Y(\boldsymbol{x}_2)\|_\infty \le \frac{1}{k\epsilon^{\frac{k-1}{k}}}\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_\infty. \tag{3.16}$$

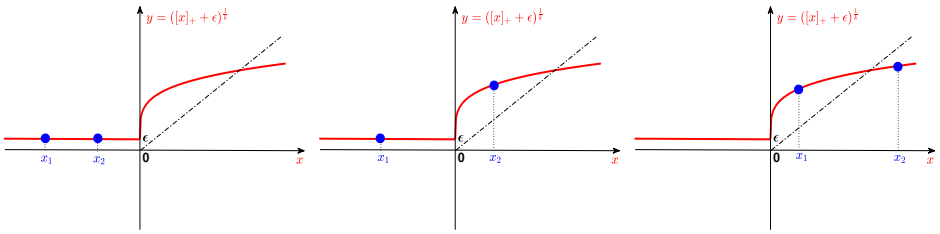

Figure 2. The three cases for analyzing the contraction property of $([x]_+ + \epsilon)^{\frac{1}{k}}$.

Now, substituting (3.14) and (3.16) into (3.9) gives

$$\|F(\boldsymbol{x}_1) - F(\boldsymbol{x}_2)\|_\infty \le \frac{\tau\mu a_{\max}h^2}{[2a_{\max}h^2 + (2 - 2\cos(h\pi))\tau]k\epsilon^{\frac{k-1}{k}}}\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_\infty. \tag{3.17}$$

This proves the desired result as stated by Theorem 3.

4. Backward substitution algorithm

In the implementation of (3.5) we have to solve a linear system

$$\left(I + \frac{\tau A}{2}\right)\boldsymbol{p}_{n+1}^{[l+1]} = \boldsymbol{b}_n^{[l]},\ \boldsymbol{b}_n^{[l]} := \frac{\tau\mu}{2}\left(\left[\boldsymbol{p}_{n+1}^{[l]} - \boldsymbol{u}_{n+1}^*\right]_+ + \boldsymbol{\epsilon}\right)^{\frac{1}{k}} + \boldsymbol{b}_n, \tag{4.1}$$

for each iteration index $l \ge 0$. This would be the major computation for solving the penalized equation (1.5) and in this section, we propose an algorithm to handle this problem. (There are also many other efficient algorithms can be used to solve such a banded linear system, such as the hybrid algorithm in [23].)

To clearly describe our idea, we let

$$\begin{aligned}
&\boldsymbol{p}_{n+1}^{[l+1]} = (y_1, y_2, \ldots, y_{J-1})^\top,\ \boldsymbol{b}_n^{[l]} = (b_1, b_2, \ldots, b_{J-1})^\top,\ \gamma = \frac{\tau}{2h^2},\\
&\tilde{a}_1 = \frac{\gamma a_2}{1 + 2\gamma a_1},\ \tilde{b}_1 = \frac{b_1}{1 + 2\gamma a_1},\\
&\tilde{a}_j = \frac{\gamma a_{j+1}}{1 + 2\gamma a_j - \gamma a_{j-1}\tilde{a}_{j-1}},\ \tilde{b}_j = \frac{b_j + \gamma a_{j-1}\tilde{b}_{j-1}}{1 + 2\gamma a_j - \gamma a_{j-1}\tilde{a}_{j-1}},\ j = 2, 3, \ldots, J-2.
\end{aligned} \tag{4.2}$$

Then, it holds

$$\begin{cases}(1+2\gamma a_1)y_1-\gamma a_2y_2=b_1,\\ -\gamma a_{j-1}+(1+2\gamma a_j)y_j-\gamma a_{j+1}y_{j+1}=b_j,\ j=2,3,\ldots,J-2,\\ -\gamma a_{J-2}y_{J-2}+(1+2\gamma a_{J-1})y_{J-1}=b_{J-1}.\end{cases} \tag{4.3}$$

From the first equation we have

$$y_1=\frac{\gamma a_2}{1+2\gamma a_1}y_2+\frac{b_1}{1+2\gamma a_1}=\tilde{a}_1y_2+\tilde{b}_1. \tag{4.4a}$$

Substituting this into the middle equation in (4.3) with $j=2$ gives

$$-\gamma a_1(\tilde{a}_1y_2+\tilde{b}_1)+(1+2\gamma a_2)y_2-\gamma a_3y_3=b_2,$$

i.e.,

$$y_2=\frac{\gamma a_3}{1+2\gamma a_2-\gamma a_1\tilde{a}_1}y_3+\frac{b_2+\gamma a_1\tilde{b}_1}{1+2\gamma a_2-\gamma a_1\tilde{a}_1}=\tilde{a}_2y_3+\tilde{b}_2.$$

In general we have

$$\begin{aligned}y_j&=-\gamma a_{j-1}y_{j-1}+(1+2\gamma a_j)y_j-\gamma a_{j+1}y_{j+1}\\&=-\gamma a_{j-1}(\tilde{a}_{j-1}y_j+\tilde{b}_{j-1})+(1+2\gamma a_j)y_j-\gamma a_{j+1}y_{j+1},\end{aligned}$$

i.e.,

$$y_j=\frac{\gamma a_{j+1}}{1+2\gamma a_j-\gamma a_{j-1}\tilde{a}_{j-1}}y_{j+1}+\frac{b_j+\gamma a_{j-1}\tilde{b}_{j-1}}{1+2\gamma a_j-\gamma a_{j-1}\tilde{a}_{j-1}}=\tilde{a}_jy_{j+1}+\tilde{b}_j, \tag{4.4b}$$

where $j=2,3,\ldots,J-2$. In particular for $j=J-1$ we have $y_{J-2}=\tilde{a}_{J-2}y_{J-1}+\tilde{b}_{J-2}$. This together with the last equation in (4.3) gives

$$\begin{cases}y_{J-2}=\tilde{a}_{J-2}y_{J-1}+\tilde{b}_{J-2},\\ -\gamma a_{J-2}y_{J-2}+(1+2\gamma a_{J-1})y_{J-1}=b_{J-1}.\end{cases}$$

Substituting the first equation into the second one gives

$$y_{J-1}=\frac{b_{J-1}+\gamma a_{J-2}\tilde{b}_{J-2}}{1+2\gamma a_{J-1}-\gamma a_{J-2}\tilde{a}_{J-2}}. \tag{4.4c}$$

With y_{J-1}, we can get $y_{J-2},y_{J-3},\ldots,y_1$ via a backward substitution according to (4.4a) and (4.4b). In summary, we can solve the linear system (4.1) as follows.

Algorithm 1 Backward Substitution Algorithm.

Initialization: Define $\{\tilde{a}_j,\tilde{b}_j\}_{j=1}^{J-2}$ according to (4.2).

Step 1 Compute y_{J-1} according to (4.4c).

Step 2 Compute $\{y_{J-2},y_{J-3},\ldots,y_2,y_1\}$ according to (4.4b) and (4.4a).

Step 3 Let $\boldsymbol{p}_{n+1}^{[l+1]}=(y_1,y_2,\ldots,y_{J-1})^\top$.

5. Numerical results

In this section, we present numerical results to study the proposed fixed-point iteration (3.5) and the backward substitution algorithm in Section 4. We use the following data

$$\begin{aligned} &\Omega = (0,1),\ t \in (0,T) \text{ with } T = 4,\ a(x) = \min\left\{0.8, \max\left\{0.2, \sin\left(\frac{\pi x}{2}\right)\right\}\right\}, \\ &u^* = \sin(\sqrt{tx}|0.5 - x|\pi),\ g = 2(1-x)\sin(2\pi t(T-t)x). \end{aligned} \tag{5.1}$$

For the penalty parameters, we use $\epsilon = 1e-4$, $k = 4$ and several values of μ. All numerical results are implemented by Matlab R2017a installed in a desk computer with Mac OS and 2.7 GHz Intel Core i5. The tolerance for the fixed-point iteration (3.5) is set to $\texttt{tol} = \frac{\min\{h^2,\tau^2\}}{10}$, which is sufficient to match the discretization errors.

Let $h = \tau = \frac{1}{100}$. Then, in Figure 3 we plot the profile of $p(x,t)$ and $u^*(x,t)$ for two values of μ: $\mu = 3$ (top row) and $\mu = 15$ (bottom row). We see that the parameter μ has a remarkable influence on the solution p. In particular, for small μ (e.g., $\mu = 3$) it does not hold $p \le u^*$ for all $(x,t) \in \Omega \times (0,T)$ as we can see in the top-right subfigure. For large μ, as we can see in the bottom-right subfigure, it holds $p \le u^*$ uniformly.

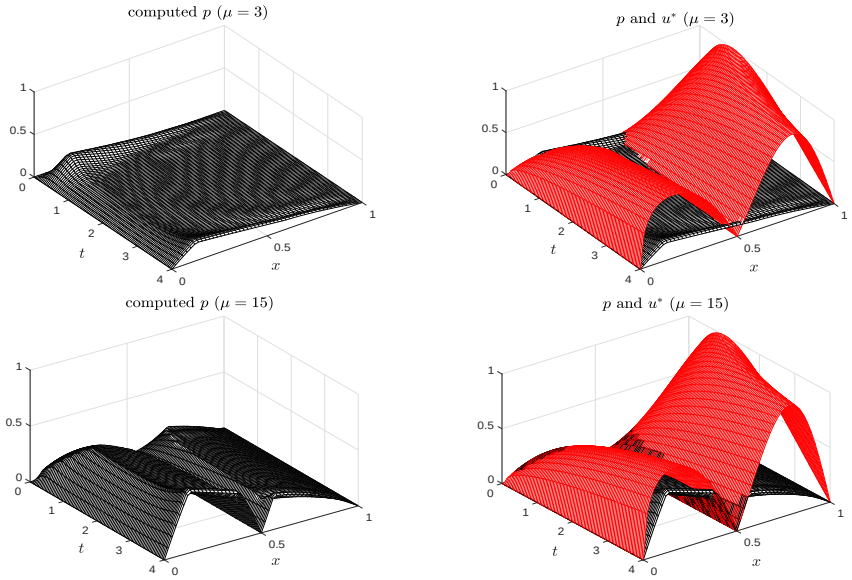

Figure 3. With the data given by (5.1), the function u^* and the computed solution p. Left: the computed p; Right: putting the computed solution p and the constraint u^* in a single panel. Top: $\mu = 3$ (and in this case it does not hold $p \le u^*$ for all $(x,t) \in \Omega \times (0,T)$); Bottom: $\mu = 15$ and it holds $p \le u^*$ uniformly.

In Figure 4 we show the local details of the constraint u^* and the computed solution p at two different time points: $t = 2$ (left) and $t = 4$ (right). Here we consider three values of penalty parameter μ: $\mu = 3, 7, 15$. The results in Figure 4 clearly indicate that as μ grows the condition $p \le u^*$ holds uniformly on the space and time domains. This observation confirms the conclusion in [3, 10–12].

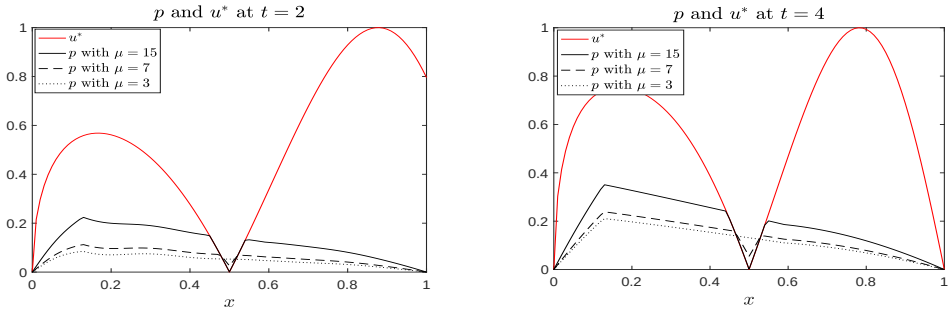

Figure 4. Local details of the constraint u^* and the computed solution p at two different time points: $t = 2$ (left) and $t = 4$ (right).

In Figure 5 we show the measured iteration number of the fixed-point iteration (3.5) for three values of the penalty parameter μ. (Such an iteration number is the quantity $L_{\max}$ such that the infinity norm of the residual arrives at the prescribed tolerance $\texttt{tol}=\frac{\min\{h^2,\tau^2\}}{10}$.) We see that as μ grows the required iteration number of the fixed-point iteration (3.5) increases as well. This is an undesirable phenomenon and needs further study.

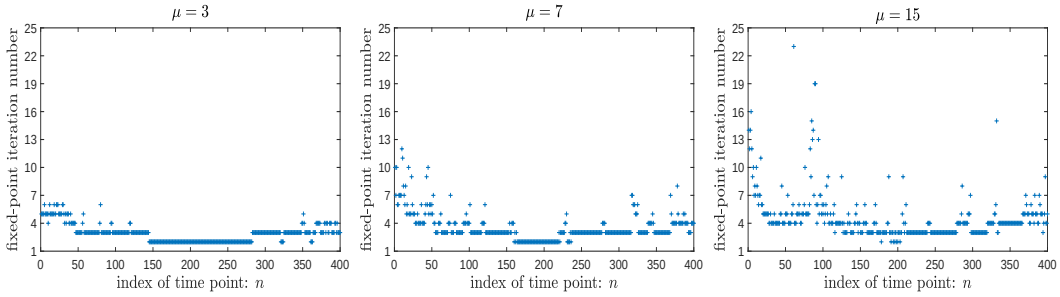

Figure 5. Iteration number of the fixed-point iteration (3.5) for three values of the penalty parameter μ.

Let

$$\mathrm{It}_{\mathrm{fp}}^{\max} = \max_{n=1,2,\ldots,N} \mathrm{It}_{\mathrm{fp}}^{n},$$

where the subscript 'fp' denotes fixed-point iteration and $\mathrm{It}_{\mathrm{fp}}^{n}$ is the iteration number for the n-th time point. The quantity $\mathrm{It}_{\mathrm{fp}}^{\max}$ is the maximal iteration number over all the N time points. In Figure 6 we show $\mathrm{It}_{\mathrm{fp}}^{\max}$ as we refine the discretization mesh sizes h (and τ) from 2^{-6} to 2^{-11}. Clearly, the results in Figure 6 indicates that the convergence rate of the fixed-point iteration algorithm is robust with respect to the mesh sizes. Such robustness is a very important property when high accuracy (i.e., h and τ should be small) is pursued.

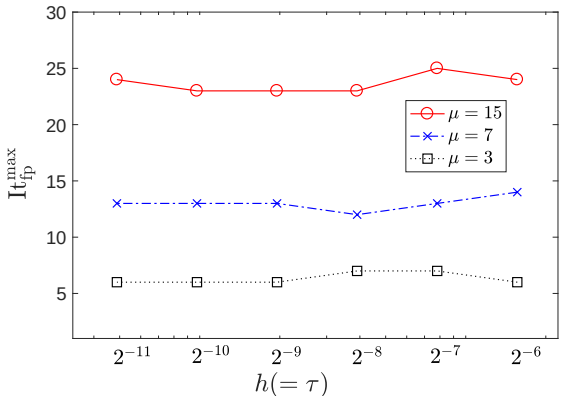

Figure 6. Maximal iteration number of the fixed-point iteration algorithm (3.5) when we refine $h(=\tau)$ from 2^{-6} to 2^{-11}. The iteration number is robust in terms of the mesh sizes.

We next study the backward substitution algorithm proposed in Section 4. We will compare it with the built-in command `backslash` in Matlab. Let p be the solution obtained by using the backward substitution algorithm proposed in Section 4 and p_{inv} is the solution obtained by using the `backslash` command in Matlab. These two linear solvers are used to handle the same linear system in (3.5) for each fixed-point iteration. In Figure 7 we show the error between p and p_{inv} on the space and time domains. We see that both linear solvers leads to the same numerical solution if we neglect the roundoff error due to floating point operations. This implies that the backward substitution algorithm indeed produces a reliable numerical solution.

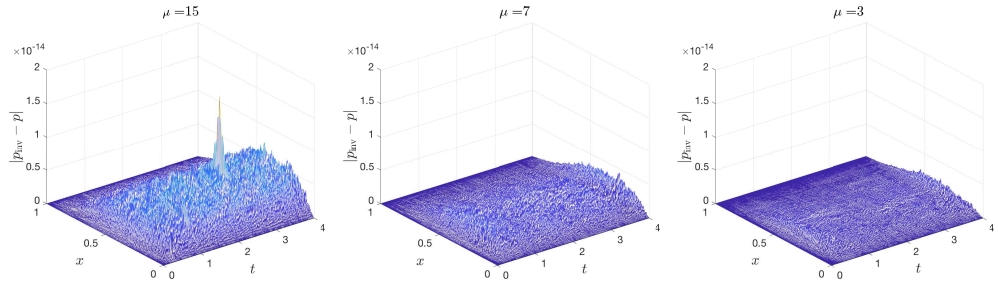

Figure 7. The error between p and p_{inv} for different values of the penalty parameter μ.

Using the two linear solvers for implementing the fixed-point iteration (3.5), we show in Figure 8 the measured CPU time for solving the penalized equation (1.5). Clearly, the computation using the backward substitution algorithm needs much less CPU time and this indicates a great advantage for using it in practice.

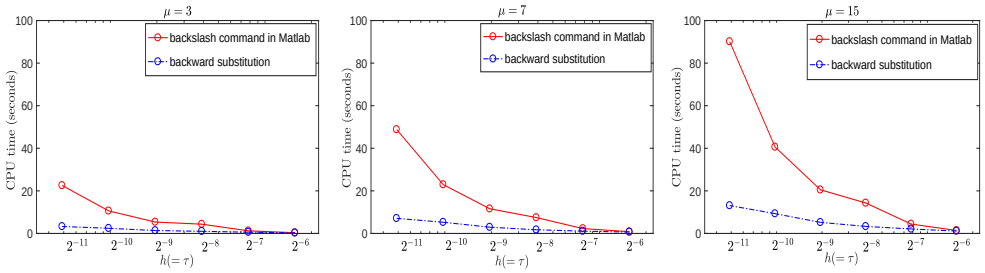

Figure 8. Measured CPU time for computing the solution of the penalized equation (1.5) by using two linear solvers, the `backslash` command in Matlab and the backward substitution algorithm proposed in Section 4. From left to right: $\mu = 3, 7, 15$. Here, $h = \tau = \frac{1}{100}$.

6. Conclusions

We have made a numerical study of the parabolic complementarity problems. We first represent the problem via a penalty method following the idea in [2, 3, 9–12] and then we discretize the penalized equation via a Crank-Nicolson method consisting of a centered finite difference formula for the space derivative and a trapezoidal rule for time discretization. In each time step, we have to solve a nonlinear system due to the penalty term and we proposed a fixed-point iteration to handle such a nonlinear system. The major computation of the fixed-point iteration is to solve a tridiagonal linear problem, for which we proposed a backward substitution algorithm, which is a direct linear solver (i.e., it is not iterative). Extensive numerical results are given, which indicate how the discretization mesh sizes and the penalty parameters affect the convergence rate of the fixed-point iteration. In terms of CPU time, the numerical results also indicate an obvious advantage of the backward substitution algorithm compared to the built-in command `backslash` in Matlab. This would be a very important advantage to deal with large-scale nonlinear/linear systems, such as the one arising from discretizing the backward stochastic differential equation (BSDE) [6, 8]. The research of BSDE in risk measure is a hot topic at present, which provides a new direction for us to further study the relation between numerical method for parabolic complementarity problem and risk measure. In the forthcoming study, we will further discuss how to use numerical method proposed in this paper to handle risk measure.

Acknowledgements

The authors are very grateful to the anonymous referees for their careful reading of a preliminary version of the manuscript and their valuable suggestions, which greatly improved the quality of this paper. This work is supported by the Guangdong Basic and Applied Basic Research Foundation (2020A1515110671), the Jiangmen Basic and Applied Basic Research Foundation (2021030100070004859).

Conflict of interest

The authors declare that there is no conflicts of interest.

References

1. L. Angermann, S. Wang, Convergence of a fitted finite volume method for European and American option valuation, *Numer. Math.*, **106** (2007), 1–40. https://doi.org/10.1007/s00211-006-0057-7
2. A. Bensoussan, J. L. Lions, *Applications of Variational Inequalities in Stochastic Control*, North-Holland Amsterdam, New York, Oxford, 1978.
3. T. B. Gyulov, M. N. Koleva, Penalty method for indifference pricing of American option in a liquidity switching market, *Appl. Numer. Math.*, **172** (2022), 525–545. https://doi.org/10.1016/j.apnum.2021.11.002
4. J. Haslinger, M. Miettinen, *Finite Element Method for Hemivariational Inequalities*, Kluwer Academic Publisher, 1999. https://doi.org/10.1007/978-1-4757-5233-5
5. L. Scurria, D. Fauconnier, P. Jiranek, T. Tamarozzi, A Galerkin/hyper-reduction technique to reduce steady-state elastohydrodynamic line contact problems, *Comput. Methods Appl. Mech. Eng.*, **386** (2021), 114132. https://doi.org/10.1016/j.cma.2021.114132
6. W. D. Zhao, Numerical methods for forward backward stochastic differential equations, *Math. Numer. Sin.*, **37** (2015), 337–373. https://doi.org/10.12286/jssx.2015.4.337
7. L. Jiang, Convexity, translation invariance and subadditivity for g-expectations and related risk measures, *Ann. Appl. Probab.*, **18** (2008), 245–258. https://doi.org/10.1214/105051607000000294
8. I. Penner, A. Reveillac, Risk measures for processes and BSDEs, *Finance Stochastics*, **19** (2015), 23–66. https://doi.org/10.1007/s00780-014-0243-x
9. R. Glowinski, *Numerical Methods for Nonlinear Variational Problems*, Springer-Verlag, 1984. https://doi.org/10.1007/978-3-662-12613-4
10. S. Wang, X. Q. Yang, K. L. Teo, Power penalty method for a linear complementarity problem arising from American option valuation, *J. Optim. Theory Appl.*, **129** (2006), 227–254. https://doi.org/10.1007/s10957-006-9062-3
11. S. Wang, C. S. Huang, A power penalty method for solving a nonlinear parabolic complementarity problem, *Nonlinear Anal. Theory Methods Appl.*, **69** (2008), 1125–1137. https://doi.org/10.1016/j.na.2007.06.014
12. M. Chen, C. Huang, A power penalty method for a class of linearly constrained variational inequality, *J. Ind. Manage. Optim.*, **14** (2018), 1381–1396. https://doi.org/10.3934/jimo.2018012
13. A. M. Rubinov, X. Q. Yang, *Lagrange-type Functions in Constrained Non-convex Optimization*, Kluwer Academic Publishers, Dordrecht, Holland, 2003.
14. X. Q. Yang, X. X. Huang, Nonlinear lagrangian approach to constrained optimization problems, *SIAM J. Optim.*, **11** (2001), 1119–1144. https://doi.org/10.1137/S1052623400371806
15. M. H. Holmes, *Introduction to Mumerical Methods in Differential Equations*, Springer New York, 2009.
16. O. T. Hanna, New explicit and implicit "improved Euler" methods for the integration of ordinary differential equations, *Comput. Chem. Eng.*, **12** (1988), 1083–1086. https://doi.org/10.1016/0098-1354(88)87030-3

17. R. Santos, L. Alves, A comparative analysis of explicit, IMEX and implicit strong stability preserving Runge-Kutta schemes, *Appl. Numer. Math.*, **159** (2021), 204–220. https://doi.org/10.1016/j.apnum.2020.09.007
18. E. Zeidler, *Nonlinear Functional Analysis and Its Applications II/B: Nonlinear Monotone Operators*, Springer-Verlag, New York, 1990.
19. Y. L. Zhao, P. Y. Zhu, X. M. Gu, X. L. Zhao, H. Y. Jian, A preconditioning technique for all-at-once system from the nonlinear tempered fractional diffusion equation, *J. Sci. Comput.*, **83** (2020), 10. https://doi.org/10.1007/s10915-020-01193-1
20. G. H. Golub, C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 2012.
21. C. M. da Fonseca, On the eigenvalues of some tridiagonal matrices, *J. Comput. Appl. Math.*, **200** (2007), 283–286. https://doi.org/10.1016/j.cam.2005.08.047
22. A. R. Willms, Analytic results for the eigenvalues of certain tridiagonal matrices, *SIAM J. Matrix Anal. Appl.*, **30** (2008), 639–656. https://doi.org/10.1137/070695411
23. W. Luo, X. M. Gu, B. Carpentieri, A hybrid triangulation method for banded linear systems, *Math. Comput. Simul.*, **194** (2022), 97–108. https://doi.org/10.1016/j.matcom.2021.11.012

AIMS Press