

# A FAST PRIMAL-DUAL METHOD FOR THE OBSTACLE PROBLEM

DOMINIQUE ZOSSO, BRAXTON OSTING, MANDY (MENGQI) XIA,  
AND STANLEY J. OSHER

ABSTRACT. We solve the non-linearized and linearized obstacle problems efficiently using a primal-dual hybrid gradients method involving projection and/or  $L^1$  penalty. Since this method requires no matrix inversions or explicit identification of the contact set, we find that this method, on a variety of test problems, achieves the precision of previous methods with a speed up of 1-2 orders of magnitude. The derivation of this method is disciplined, relying on a saddle point formulation of the convex problem, and can be adapted to a wide range of other constrained convex optimization problems.

## 1. INTRODUCTION

The obstacle problem is a classic motivating example in the mathematical study of variational inequalities and free boundary problems with broad applications, *e.g.*, in fluid filtration in porous media, constrained heating, elasto-plasticity, optimal control, or financial mathematics [2, 7, 10, 22]. The classical formulation is motivated by the equilibrium position of a membrane (described by the graph of  $u: \Omega \subset \mathbb{R}^n \rightarrow \mathbb{R}$ ), with fixed values  $f$  imposed on the boundary  $\partial\Omega$  and constrained to lie (almost everywhere) above an obstacle function,  $\varphi: \Omega \rightarrow \mathbb{R}$ :

$$(1.1) \quad \min_{u \in H^1(\Omega)} \int_{\Omega} \sqrt{1 + |\nabla u|^2} \quad \text{s.t.} \quad u \geq \varphi \text{ a.e. in } \Omega \quad \text{and} \quad u = f \text{ on } \partial\Omega.$$

For small deflections, the above integral (1.1) is typically linearized into the Dirichlet integral:

$$(1.2) \quad \min_{u \in H^1(\Omega)} \int_{\Omega} |\nabla u|^2 \quad \text{s.t.} \quad u \geq \varphi \text{ a.e. in } \Omega \quad \text{and} \quad u = f \text{ on } \partial\Omega.$$

Within the Sobolev space  $H^1(\Omega)$ , the admissible functions for either obstacle problem, (1.1) or (1.2), define the closed convex set  $U = U(\Omega, \phi, f)$ :

$$(1.3) \quad U = \{u \in H^1(\Omega), u \geq \varphi \text{ a.e. in } \Omega, u|_{\partial\Omega} = f\}$$

Standard arguments from variational analysis can be used to show that a unique minimizer of (1.2) exists for sufficiently regular data  $(\Omega, \phi, f)$ ; see, for example,

---

*Date:* September 8, 2015.

2010 *Mathematics Subject Classification.* Primary: 35J20, 49Q05, 53C44, 65Dxx, 65K15, 65N06.

*Key words and phrases.* Obstacle problem, minimal surface, convex optimization, primal-dual hybrid gradients.

The authors gratefully acknowledge support from NSF DMS-1461138 and DMS-1118971, the W. M. Keck Foundation, UC Lab Fees grant, and ONR grant N00014-14-0444.

[10, 22, 2]. The non-coercivity of the nonlinear problem (1.1) can be handled using the methods in [7].

In practice, numerical methods are required to solve the obstacle problem. Lions and Mercier illustrated a splitting algorithm on the obstacle problem [16]. Finite element methods were used to solve the obstacle problem as a free boundary problem [15, 5, 11]. Iteration methods based on penalty and projected gradients are known approaches [6, 23]. In [12, 13], the obstacle constraint is incorporated into the objective using a Lagrange multiplier. Multigrid and multilevel methods have been employed to speed up computation: The obstacle problems described by variational inequalities can be solved using the iterative scheme proposed in [14], where at each step the system is solved by a multigrid algorithm; [28] discretizes the problem into a continuous piecewise linear finite element space and turn the problem into a quadratic programming problem with inequality constraints, which is then solved with multilevel projection (MP).

In [17], the authors propose a modified level set method to solve the obstacle problem and turn the problem into a nonsmooth minimization problem. Two methods are presented to solve the minimization problem: a proximal bundle method is employed for solving a general nonsmooth minimization problem; a gradient method is proposed to solve the regularized problems.

Most recently, in [26], the authors proposed an efficient numerical scheme for solving the (linearized) obstacle problem (and generalizations) based on a reformulation of the obstacle in terms of an  $L^1$ -like penalty on the variational problem, *in lieu* of the obstacle constraint:

$$(1.4) \quad \min_{u \in H^1(\Omega)} \int_{\Omega} |\nabla u|^2 + \mu(\varphi - u)_+ \quad \text{s.t.} \quad u|_{\partial\Omega} = f,$$

where  $(\cdot)_+ = \max(\cdot, 0)$ . It was shown that for sufficiently large but finite  $\mu$ , the minimizer of the unconstrained problem (1.4) is also the minimizer of the original, constrained problem (1.2).

The functionals and constraints in (1.1), (1.2), and (1.4) are all closed and convex, which makes the problems accessible to convex optimization techniques. There is a huge corpus of algorithms that apply to such convex optimization problems. Recently, primal-dual splitting algorithms have gained particular attention, most notably so in the context of TV and  $L^1$ -type problems in imaging; see, *e.g.*, [8, 9, 29, 30]. Zosso and Bustin have applied the theory to the particular case of the Beltrami functional [24], used as an interesting regularizer in imaging problems that interpolates between the classical  $H^1$  and TV regularizers, and which is a generalization of the surface area functional [31]. Zosso and Oosting use the method for efficient minimization of a minimal surface-based graph partitioning problem [32].

In this paper, we develop a primal-dual hybrid gradient method to efficiently solve constrained and  $L^1$ -penalty formulations of the linear and non-linear obstacle problems, (1.1) and (1.2). We comment that the original constrained problem is algorithmically related to the  $L^1$ -based problem for finite  $\mu$ , and strictly identical for the particular choice  $\mu = \infty$ . Our results achieve state-of-the-art precision in much shorter time; the speed up is 1-2 orders of magnitude with respect to the method in [26], and even larger compared to older methods [16, 17, 27]. This is demonstrated for a variety of one- and two-dimensional problems, including the extension to elasto-plastic problems with double obstacles and potential term.

It is important to note that our fast algorithm is derived from a primal-dual reformulation of the original convex optimization problems in a very disciplined way. The primal-dual hybrid gradients method employed here enjoys straightforward convergence guarantees. Moreover, in the particular case of the linear obstacle problem, the dual variable update is particularly simple, and the dual variable can be eliminated altogether, resulting in a single primal variable update scheme that greatly improves over standard explicit gradient descent in multiple ways. Namely, the resulting scheme is fast because previous iterates keep being taken into account, which has highly interesting interpretations in terms of “momentum method”, and can also be reformulated as a damped wave equation. We illustrate this nature on a very simple yet striking toy example. More general advantages of our method are the avoidance of matrix/operator inversion (all update steps are explicit), and the fact that the contact/coincidence set (the free boundary) is handled implicitly, with no explicit tracking being necessary.

*Outline:* The remainder of the paper is organized as follows: Section 2 gives a formulation of the discretized obstacle problem and its different flavors. Section 3 describes the proposed primal-dual algorithms. In Section 4, we compare the resulting primal-dual algorithms to some more classical ones. Section 5 presents the results of the primal-dual algorithm on a variety of numerical experiments. We conclude in Section 6 with a discussion.

## 2. DISCRETIZED OBSTACLE PROBLEM

In this section, we introduce a discretized obstacle problem and discuss some of its properties. To this end, we introduce the following notation.

Let  $X = \{x\}$ ,  $x \in \mathbb{R}^D$  denote the set of Cartesian grid points of the discretization of the  $D$ -dimensional domain  $\Omega \subset \mathbb{R}^D$ .

### 2.1. Inner products and norms.

**Definition 2.1.** For functions  $f, g: X \rightarrow \mathbb{R}$ , we define the  $L^2$  inner product

$$(2.1) \quad \langle f, g \rangle_{L^2(X)} := \sum_{x \in X} f(x)g(x)$$

and the induced  $L^2$ -norm

$$(2.2) \quad \|f\|_{L^2(X)} := \langle f, f \rangle_{L^2(X)}^{\frac{1}{2}} = \sqrt{\sum_{x \in X} f(x)^2}$$

**Definition 2.2.** Similarly, for vectors  $v, w: X \rightarrow \mathbb{R}^D$ , we define the  $L^2$  inner product

$$(2.3) \quad \langle v, w \rangle_{L^2(X)^D} := \sum_{x \in X} \sum_{d=1}^D v_d(x) \cdot w_d(x)$$

and obtain the induced  $L^2$ -norm

$$(2.4) \quad \|v\|_{L^2(X)^D} := \langle v, v \rangle_{L^2(X)^D}^{\frac{1}{2}} = \sqrt{\sum_{x \in X} \sum_{d=1}^D v_d(x)^2},$$

where  $v_d(x)$  denotes the  $d$ -th component of the vector at location  $x$ .

**Definition 2.3.** For a vector  $v: X \rightarrow \mathbb{R}^D$ , the magnitude  $|v|: X \rightarrow \mathbb{R}$  is defined as follows:

$$(2.5) \quad |v|(x) := \sqrt{\sum_{d=1}^D v_d(x)^2}, \quad x \in X.$$

The magnitude operator,  $|\cdot|: L^2(X)^D \rightarrow L^2(X)$ , thus maps vector-valued functions to scalar-valued functions. It is easy to see that  $\|v\|_{L^2(X)^D} = \||v|\|_{L^2(X)}$ .

**Definition 2.4.** Finally, for the obstacle penalty, we need the single sided  $L^1$ -penalty of a function  $f: X \rightarrow \mathbb{R}$ :

$$(2.6) \quad \|f\|_+ := \sum_{x \in X} \max(f(x), 0).$$

A corresponding negative-sided penalty can be obtained directly as

$$(2.7) \quad \|f\|_- := \|-f\|_+.$$

It is easy to verify that  $\|f\|_+, \|f\|_- \geq 0$ , and  $\|f\|_+ + \|f\|_- = \|f\|_1$ .

## 2.2. Finite difference operators.

**Definition 2.5** (Gradient). For functions  $f: X \rightarrow \mathbb{R}$ , we denote by  $\nabla_X: L^2(X) \rightarrow L^2(X)^D$  a suitable discretization of the continuous gradient by *finite differences*, subject to desired boundary constraints.

**Definition 2.6** (Divergence). Using the discrete gradient, for vectors  $v: X \rightarrow \mathbb{R}^D$  we define the discretized divergence  $\text{div}_X: L^2(X)^D \rightarrow L^2(X)$ , such that the two operators satisfy the usual negative adjoint property:

$$(2.8) \quad \langle \nabla_X f, v \rangle_{L^2(X)^D} = -\langle f, \text{div}_X v \rangle_{L^2(X)}, \quad f: X \rightarrow \mathbb{R}, \quad v: X \rightarrow \mathbb{R}^D.$$

**Definition 2.7** (Laplacian). For functions  $f: X \rightarrow \mathbb{R}$ , the composition of divergence and gradient yields the discrete Laplacian operator  $\Delta_X: L^2(X) \rightarrow L^2(X)$ :

$$(2.9) \quad \Delta_X f := (\text{div}_X \circ \nabla_X) f.$$

For example, we consider the standard forward differences stencil for the gradient; then the adjoint relation leads to backward differences in the divergence and the resulting discretized Laplacian corresponds to the commonly used  $2D + 1$  point centered differences stencil.

**2.3. Surface area and Dirichlet energy.** Based on the preceding definitions, we define the following two functionals:

**Definition 2.8** (Surface area). For a Monge surface, discretized by the graph of  $u: X \rightarrow \mathbb{R}$ , the discrete surface area,  $S: L^2(X) \rightarrow \mathbb{R}$ , is defined as

$$(2.10) \quad S[u] := \sum_{x \in X} \sqrt{1 + |\nabla_X u|(x)^2}$$

**Definition 2.9** (Dirichlet energy). Similarly the discrete Dirichlet energy,  $D: L^2(X) \rightarrow \mathbb{R}$ , is defined as

$$(2.11) \quad D[u] := \frac{1}{2} \|\nabla_X u\|_{L^2(X)^D}^2$$

For slowly varying functions  $u$ , *i.e.*, such that  $\forall x \in X: |\nabla_X u|(x) \ll 1$ , the Dirichlet energy is an approximation to the surface area,

$$S[u] \approx |X| + D[u],$$

where  $|X|$  denotes the number of sampling points in  $X$ , *i.e.*, the area of the discretized domain  $\Omega$ . Note that both the surface area and the Dirichlet energy functional are proper, closed, and convex.

**2.4. Convex conjugates.** We recall the definition of the convex conjugate (a.k.a. Legendre-Fenchel transform) of a function.

**Definition 2.10** (Legendre-Fenchel transform). The convex conjugate of a function  $f$  is the function  $f^*$  defined by

$$(2.12) \quad f^*(s) = \sup_u \langle s, u \rangle - f(u).$$

The biconjugate,  $f^{**} := (f^*)^*$ , is the largest closed convex function with  $f^{**} \leq f$ . As a result,  $f \equiv f^{**}$  iff  $f$  is closed convex (Fenchel-Moreau-Rockafellar Theorem) [21, Theorem 5].

Using Definitions 2.10 and 2.11, we can determine the respective convex conjugate functionals,  $D^*, S^*: L^2(X)^D \rightarrow \mathbb{R}$  of the Dirichlet energy and surface area as follows:

$$(2.13a) \quad \begin{aligned} D^*[p] &:= \sup_{u \in L^2(X)} \langle \nabla_X u, p \rangle_{L^2(X)^D} - D[u] \\ &= \frac{1}{2} \|p\|_{L^2(X)^D}^2, \end{aligned}$$

and

$$(2.13b) \quad \begin{aligned} S^*[p] &:= \sup_{u \in L^2(X)} \langle \nabla_X u, p \rangle_{L^2(X)^D} - S[u] \\ &= \begin{cases} - \sum_{x \in X} \sqrt{1 - |p|(x)^2} & \text{if } \forall x \in X: |p|(x) \leq 1, \\ \infty & \text{otherwise.} \end{cases} \end{aligned}$$

Thanks to the Fenchel-Moreau theorem we know, that the biconjugate functionals,  $S^{**}$  and  $D^{**}$ , are identical to the original functionals:

$$(2.14a) \quad S^{**}[u] := \sup_{\substack{p \in L^2(X)^D \\ |p|(x) \leq 1}} \langle \nabla_X u, p \rangle_{L^2(X)^D} - S^*[p] = S[u]$$

$$(2.14b) \quad D^{**}[u] := \sup_{p \in L^2(X)^D} \langle \nabla_X u, p \rangle_{L^2(X)^D} - D^*[p] = D[u].$$

**2.5. Discretized obstacle problem.** Having defined the discretized surface area and Dirichlet energy functionals, we next define the discretized optimization problem.

Given a discretization  $X$  and appropriate finite difference operators for gradient, divergence, and Laplacian, we declare a subset  $\partial X \subset X$  to be boundary points. Let  $\varphi: X \rightarrow \mathbb{R}$  be an obstacle, and  $f: \partial X \rightarrow \mathbb{R}$  the prescribed boundary values, compatible with the obstacle, *i.e.*,  $\forall x \in \partial X: \varphi(x) \leq f(x)$ . Consider further the set  $U_f$  of square integrable functions who satisfy the boundary constraints,

$$(2.15) \quad U_f := \{u \in L^2(X) \mid u(x) = f(x) \forall x \in \partial X\},$$

and the subset  $U_{f,\varphi}$  thereof additionally satisfying the obstacle constraint,

$$(2.16) \quad U_{f,\varphi} := \{u \in L^2(X) \mid u(x) = f(x) \ \forall x \in \partial X \text{ and } u(x) \geq \varphi(x) \ \forall x \in X\}.$$

Both these sets can be shown to be closed convex. We are interested in solving the following four (primal) problems.

*Constrained obstacle problem:*

$$(P1) \quad \min_{u \in U_{f,\varphi}} S[u]$$

*Constrained linearized obstacle problem:*

$$(P2) \quad \min_{u \in U_{f,\varphi}} D[u]$$

*$L^1$ -penalty obstacle problem:*

$$(P3) \quad \min_{u \in U_f} S[u] + \mu \|\varphi - u\|_+$$

*Linearized  $L^1$ -penalty obstacle problem:*

$$(P4) \quad \min_{u \in U_f} D[u] + \mu \|\varphi - u\|_+$$

Due to the convexity of functionals and admissible sets it follows from standard arguments that all of these problems admit unique solutions.

In [26, Theorem 3.1] it is shown that for any  $\mu \geq -\Delta\phi$ , the minimizers of the constrained and  $L^1$ -penalty linearized obstacle problems, the continuum version of (P2) and (P4), are the same. Below, in Theorem 2.2, we give the analogous result for the discrete problems, (P2) and (P4). The proof is largely the same, except for one small but interesting point. In the continuum, one has that for any  $f \in H^1$ ,  $\langle \nabla f_+, \nabla f_- \rangle = 0$ , where  $f_+ = \max(f, 0)$ ,  $f_- = \min(f, 0)$ , and  $f = f_+ + f_-$ . In our discrete setting we have the following result.

**Lemma 2.1.** *For any  $f \in L^2(X)$ ,  $\langle \nabla_X f_+, \nabla_X f_- \rangle_{L^2(X)^D} \geq 0$ .*

*Proof.* Denoting the adjacent vertices of  $x \in X$  by  $N(x)$ , we compute

$$\begin{aligned} \langle \nabla_X f_+, \nabla_X f_- \rangle_{L^2(X)^D} &= -\langle \Delta_X f_+, f_- \rangle_{L^2(X)} \\ &= -\sum_{x \in X} (\Delta_X f_+)(x) f_-(x) \\ &= -\sum_{x \in X} \left( \sum_{y \in N(x)} f_+(y) - f_+(x) \right) f_-(x) \\ &= \sum_{x \in X} |f_-(x)| \sum_{y \in N(x)} |f_+(y)| \\ &\geq 0. \end{aligned}$$

□

**Theorem 2.2.** *Let  $u$  and  $u_\mu$  denote the minimizers of (P2) and (P4) respectively. For any  $\mu \geq \max\{-\Delta_X \phi(x) : x \in X\}$ , we have that  $u_\mu = u$ .*

*Proof.* Consider  $w := u_\mu + (\phi - v)_+$ . Abbreviate  $\langle \cdot, \cdot \rangle = \langle \cdot, \cdot \rangle_{L^2(X)^D}$  and  $\|\cdot\| = \|\cdot\|_{L^2(X)^D}$ . Evaluating the objective of (P4) at the admissible  $w \in U_{f,\phi}$ , we obtain

$$\begin{aligned} D[w] + \mu\|\phi - w\|_+ &= D[w] \\ &= D[u_\mu] + \langle \nabla_X u_\mu, \nabla_X(\phi - u_\mu)_+ \rangle + \frac{1}{2} \langle \nabla_X(\phi - u_\mu)_+, \nabla_X(\phi - u_\mu)_+ \rangle \\ &= D[u_\mu] + \langle -\Delta_X \phi, (\phi - u_\mu)_+ \rangle_{L^2(X)} \\ &\quad - \frac{1}{2} \|\nabla_X(\phi - u_\mu)_+\|^2 - \langle \nabla_X(\phi - u_\mu)_+, \nabla_X(\phi - u_\mu)_- \rangle \end{aligned}$$

Now assuming  $\mu \geq \max\{-\Delta_X \phi(x) : x \in X\}$  and using Lemma 2.1, we obtain

$$D[w] + \mu\|\phi - w\|_+ \leq D[u_\mu] + \mu\|\phi - u_\mu\|_+.$$

Since  $u_\mu$  is the unique minimizer of  $D[\cdot] + \mu\|\phi - \cdot\|_+$  over  $U_f$ , we conclude that  $w = u_\mu$  which implies that  $u_\mu \geq \phi$ , i.e.,  $u_\mu \in U_{f,\phi}$ . Additionally, we have that

$$D[u_\mu] = D[u_\mu] + \mu\|\phi - u_\mu\|_+ \leq D[u] + \mu\|\phi - u\|_+ = D[u].$$

Since  $u$  is the unique minimizer of  $D[\cdot]$  over  $U_{f,\phi}$  and  $u_\mu$  is admissible, we conclude that  $u = u_\mu$ , as desired.  $\square$

**2.6. Saddle point formulation.** Thanks to the identity of a closed convex functional with its biconjugate, we can substitute the functionals  $S[u]$  and  $D[u]$  by the definition of their respective biconjugates (2.14a) and (2.14b). We thus obtain the four primal-dual saddle point problems, respectively equivalent to the primal problems (P1)–(P4):

*Constrained obstacle problem:*

$$(PD1) \quad \min_{u \in U_{f,\phi}} \max_{\substack{p \in L^2(X)^D \\ |p|(x) \leq 1}} \langle \nabla_X u, p \rangle_{L^2(X)^D} - S^*[p]$$

*Constrained linearized obstacle problem:*

$$(PD2) \quad \min_{u \in U_{f,\phi}} \max_{p \in L^2(X)^D} \langle \nabla_X u, p \rangle_{L^2(X)^D} - D^*[p]$$

*$L^1$ -penalty obstacle problem:*

$$(PD3) \quad \min_{u \in U_f} \max_{\substack{p \in L^2(X)^D \\ |p|(x) \leq 1}} \langle \nabla_X u, p \rangle_{L^2(X)^D} - S^*[p] + \mu\|\phi - u\|_+$$

*Linearized  $L^1$ -penalty obstacle problem:*

$$(PD4) \quad \min_{u \in U_f} \max_{p \in L^2(X)^D} \langle \nabla_X u, p \rangle_{L^2(X)^D} - D^*[p] + \mu\|\phi - u\|_+$$

For discussing algorithms for the preceding problems (PD1)–(PD4), it is convenient to summarize these saddle point problems in the following form:

$$(2.17) \quad \min_{u \in U} \max_{p \in W^*} \langle \nabla_X u, p \rangle_{L^2(X)^D} - F^*[p] + G[u],$$

where we identify the admissible sets,  $U$  and  $W^*$ , and the functionals,  $F^*$  and  $G$ , as follows:

$$(2.18a) \quad U := \begin{cases} U_{f,\varphi} & \text{for (PD1), (PD2)} \\ U_f & \text{for (PD3), (PD4)} \end{cases}$$

$$(2.18b) \quad W^* := \begin{cases} L^2(X)^D \cap \{p \mid |p|(x) \leq 1\} & \text{for (PD1), (PD3)} \\ L^2(X)^D & \text{for (PD2), (PD4)} \end{cases}$$

$$(2.18c) \quad F^*[p] := \begin{cases} S^*[p] & \text{for (PD1), (PD3)} \\ D^*[p] & \text{for (PD2), (PD4)} \end{cases}$$

$$(2.18d) \quad G[u] := \begin{cases} 0 & \text{for (PD1), (PD2)} \\ \mu \|\varphi - u\|_+ & \text{for (PD3), (PD4)}. \end{cases}$$

### 3. ALGORITHMS

**3.1. Primal-dual hybrid gradients scheme.** To solve the general saddle point problem (2.17) efficiently, we propose an adaptation of the primal-dual hybrid gradients algorithm [8, 29, 30, 32]. For (2.17), the structure of the PDHG algorithm is as follows:

$$(3.1a) \quad p^{n+1} = \arg \min_{p \in W^*} -\langle \nabla_X \bar{u}^n, p \rangle_{L^2(X)^D} + F^*[p] + \frac{1}{2r_1} \|p - p^n\|_{L^2(X)^D}^2$$

$$(3.1b) \quad u^{n+1} = \arg \min_{u \in U} -\langle u, \operatorname{div}_X p^{n+1} \rangle_{L^2(X)} + G[u] + \frac{1}{2r_2} \|u - u^n\|_{L^2(X)}^2$$

$$(3.1c) \quad \bar{u}^{n+1} = 2u^{n+1} - u^n.$$

The first two steps are the proximal update of the dual and primal variable, respectively. The third, extra-gradient step is an overrelaxation of the primal update in order to overcome the stepsize shortening typical of first order methods; it is a prediction of the primal variable update used in the dual variable update. Note the appearance of  $\bar{u}$  instead of  $u$  in the dual variable update 3.1a.

In general,  $O(1/n)$  (where  $n$  is the number of iterations) convergence has been shown for fixed  $r_1, r_2$  satisfying

$$(3.2) \quad r_1 r_2 \|\nabla_X\|^2 \leq 1,$$

where  $\|\nabla_X\|$  is the operator norm/induced norm of the discretized gradient operator; equivalently,  $\|\nabla_X\|^2 = \|\Delta_X\|$  is the induced norm of the composition  $\Delta_X = \operatorname{div}_X \circ \nabla_X$ .

In the following subsections, we show how the respective minimization sub-problems (3.1a) and (3.1b) can be solved efficiently for each specific configuration in (2.18) corresponding to (PD1)–(PD4).

**3.2. Dual variable update.** We first address the update of the *dual* variable  $p$ . This step is unaffected by the choice of constrained or  $L^1$ -penalty formulation, but differs between the non-linear,  $S[u]$ , and linearized,  $D[u]$ , case.

3.2.1. *Linear cases, (PD2) and (PD4).* In the linearized cases (PD2) and (PD4), where  $F^* = D^*$ , the minimization problem of the dual variable update (3.1a) is given by

$$(3.3) \quad p^{n+1} = \arg \min_{p \in L^2(X)^D} -\langle \nabla_X \bar{u}^n, p \rangle_{L^2(X)^D} + \frac{1}{2} \|p\|_{L^2(X)^D}^2 + \frac{1}{2r_1} \|p - p^n\|_{L^2(X)^D}^2.$$

The minimizer of this quadratic function is immediately found as

$$(3.4) \quad p^{n+1} = \frac{p^n + r_1 \nabla_X \bar{u}^n}{1 + r_1}.$$

3.2.2. *Non-linear cases (PD1) and (PD3).* In the nonlinear cases (PD1) and (PD3), where  $F^* = S^*$ , by rationalizing the numerator of the dual, the minimization problem of the dual variable update (3.1a) can be rewritten as

$$(3.5) \quad p^{n+1} = \arg \min_{\substack{p \in L^2(X)^D \\ |p|(x) \leq 1}} -\langle \nabla_X \bar{u}^n, p \rangle_{L^2(X)^D} + \sum_{x \in X} \frac{|p|(x)^2 - 1}{\sqrt{1 - |p|(x)^2}} + \frac{1}{2r_1} \|p - p^n\|_{L^2(X)^D}^2.$$

We propose to solve this in an iteratively reweighted least squares (IRLS) approach as follows. We observe that the objective in (3.5) without the square-root in the denominator is quadratic in  $p$ . We fix the current estimate of  $p$  in the square-root to obtain a weighted least squares problem, this weight is then updated, and the process is repeated until convergence:

$$(3.6a) \quad \psi^0 = p^n$$

$$(3.6b) \quad \psi^{t+1} = \arg \min_{\substack{\psi \in L^2(X)^D \\ |\psi|(x) \leq 1}} -\langle \nabla_X \bar{u}^n, \psi \rangle_{L^2(X)^D} + \sum_{x \in X} \frac{|\psi|(x)^2 - 1}{\sqrt{1 - |\psi^t|(x)^2}} + \frac{1}{2r_1} \|\psi - p^n\|_{L^2(X)^D}^2$$

$$(3.6c) \quad p^{n+1} = \lim_{t \rightarrow \infty} \psi^t$$

Our focus is on the solution of the fixed-point minimization step (3.6b). Indeed, it can be shown (see [32]) that for small enough  $r_1$ , the solution to (3.6b) is given by

$$(3.7) \quad \psi^{t+1}(x) = \frac{\varepsilon^t(x)}{\varepsilon^t(x) + 2r_1} [p^n(x) + r_1 (\nabla_X \bar{u}^n)(x)], \quad x \in X,$$

where  $\varepsilon^t(x) := \sqrt{1 - |\psi^t|(x)^2}$ .

**3.3. Primal variable update.** We now consider the update problem (3.1b) of the primal variable. Here, the choice of non-linear versus linearized cost functional is irrelevant; instead, we distinguish the original constraint formulations (PD1) and (PD2) from the relaxed  $L^1$ -penalty formulations (PD3) and (PD4).

3.3.1. *Constrained formulation (PD1) and (PD2)*. The constrained primal variable iteration (3.1b) reads

$$(3.8) \quad u^{n+1} = \arg \min_{u \in U_{f,\varphi}} -\langle u, \operatorname{div}_X p^{n+1} \rangle_{L^2(X)} + \frac{1}{2r_2} \|u - u^n\|_{L^2(X)}^2.$$

Based on the projection theorem and variational inequalities [6], and in the tradition of the Arrow-Hurwitz-Uzawa algorithm [1] and the first recent proposition of the PDHG algorithm [29], this update is approximated by seeking a minimizer in the relaxed set  $L^2(X)$  and projecting onto the admissible set  $U_{f,\varphi}$ . This approximation to the iteration can be written

$$(3.9) \quad \begin{aligned} u^{n+1} &= P_{U_{f,\varphi}} \left[ \arg \min_{u \in L^2(X)} -\langle u, \operatorname{div}_X p^{n+1} \rangle_{L^2(X)} + \frac{1}{2r_2} \|u - u^n\|_{L^2(X)}^2 \right] \\ &= P_{U_{f,\varphi}}[u^*], \end{aligned}$$

with

$$(3.10) \quad u^* := u^n + r_2 \operatorname{div}_X p^{n+1}$$

and where the projection  $P_{U_{f,\varphi}} : L^2(X) \rightarrow U_{f,\varphi}$  is simply defined as

$$(3.11) \quad P_{U_{f,\varphi}}[u^*] := \arg \min_{u \in U_{f,\varphi}} \|u - u^*\|_{L^2(X)}.$$

Therefore, more explicitly for  $x \in X$ , the update of the primal variable in the constrained setting (PD1), (PD2), is:

$$(3.12) \quad u^{n+1}(x) := \begin{cases} f(x) & \text{if } x \in \partial X, \\ \varphi(x) & \text{if } x \notin \partial X \text{ and } u^*(x) < \varphi(x), \\ u^*(x) & \text{otherwise.} \end{cases}$$

3.3.2.  *$L^1$ -penalty formulations (PD3) and (PD4)*. The  $L^1$ -penalty method relaxes the admissible set to  $U_f$ , i.e.,  $L^2(X)$  subject to only the boundary values  $f$  on  $\partial X$ , and introduces the  $L^1$ -like penalty term  $\|\varphi - u\|_+$ , instead. The resulting instance of (3.1b) can be rewritten as

$$(3.13) \quad u^{n+1} = \arg \min_{u \in U_f} \mu \|\varphi - u\|_+ + \frac{1}{2r_2} \|u - (u^n + r_2 \operatorname{div}_X p^{n+1})\|_{L^2(X)}^2.$$

The minimizer can be easily found by making a change of variables,  $z = \varphi - u$ , leading to

$$(3.14) \quad \begin{aligned} z^* &= \arg \min_z r_2 \mu \|z\|_+ + \frac{1}{2} \|\varphi - (u^n + r_2 \operatorname{div}_X p^{n+1}) - z\|_{L^2(X)}^2 \\ &= \operatorname{shrink}_+(\varphi - (u^n + r_2 \operatorname{div}_X p^{n+1}), r_2 \mu), \end{aligned}$$

where  $\operatorname{shrink}_+$  denotes single-sided soft-thresholding (shrinkage) as follows:

$$(3.15) \quad \operatorname{shrink}_+(s, \tau) := \begin{cases} s & \text{if } s < 0, \\ 0 & \text{if } 0 \leq s \leq \tau, \\ s - \tau & \text{if } s > \tau. \end{cases}$$

Let again  $u^* = u^n + r_2 \operatorname{div}_X p^{n+1}$ . We re-substitute  $u^{n+1} = \varphi - z^*$  and respect the boundary conditions through projection  $P_{U_f}$ ,

$$(3.16) \quad u^{n+1} = P_{U_f}[\varphi - \operatorname{shrink}_+(\varphi - u^*, r_2 \mu)].$$

This leads to the following explicit primal update for the  $L^1$ -penalty based obstacle problems, (PD3), (PD4):

$$(3.17) \quad u^{n+1}(x) = \begin{cases} f(x) & \text{if } x \in \partial X, \\ u^*(x) + r_2\mu & \text{if } x \notin \partial X \text{ and } u^*(x) < \varphi(x) - r_2\mu, \\ \varphi(x) & \text{if } x \notin \partial X \text{ and } \varphi(x) - r_2\mu \leq u^*(x) \leq \varphi(x), \\ u^*(x) & \text{otherwise.} \end{cases}$$

*Remark 1.* Recall that in Theorem 2.2, we proved that the solutions to the constrained and  $L^1$ -penalized obstacle problems, (PD2) and (PD4), are equivalent for  $\mu \geq \max\{-\Delta_X \varphi(x) : x \in X\}$ ; see also [26]. In light of this, it is interesting to compare primal iterations (3.12) and (3.17) for the constrained and  $L^1$ -penalized obstacle problems. These two algorithms only disagree for points  $x \notin \partial X$  with  $u^*(x) < \varphi(x) - r_2\mu$ , *i.e.*, when the tentative membrane update  $u^*$  violates the obstacle constraint by an amount in excess of  $r_2\mu$ . This suggests that not only do the solutions of these two problems agree for sufficiently large  $\mu$ , but so do all of the iterates in the proposed primal-dual methods.

**3.4. Elimination of the dual variable in the linear case, (PD2) and (PD4).** In the updates of the primal variable, the dual variable only appears through its divergence,  $\operatorname{div}_X p^{n+1}$ ; see (3.10) being used in (3.12) and (3.17). However, in the linear case, (PD2) and (PD4), by applying the divergence to (3.4), it is easy to see that

$$(3.18) \quad \operatorname{div}_X p^{n+1} = \frac{\operatorname{div}_X p^n + r_1 \Delta_X \bar{u}^n}{1 + r_1}.$$

Assuming  $p^0(x) = 0$  for all  $x \in X$ , this recursion relation can be solved, giving

$$(3.19) \quad \operatorname{div}_X p^{n+1} = \sum_{\eta=0}^n \frac{r_1 \Delta_X \bar{u}^{n-\eta}}{(1 + r_1)^{\eta+1}}.$$

This allows rewriting the primal variable updates without the dual variable, effectively eliminating the dual variable from the scheme altogether. After elimination, the intermediate primal update (3.10) is now

$$(3.20) \quad u^* = u^n + r_1 r_2 \sum_{\eta=0}^n \frac{\Delta_X \bar{u}^{n-\eta}}{(1 + r_1)^{\eta+1}},$$

which is then projected/shrunk according to (3.12) or (3.17) as before.

**3.5. Complete algorithms.** The complete algorithms for the four different flavors of the obstacle problem (PD1)–(PD4) are summarized in Algorithms 1–2. These algorithms have extremely simple structure and require only low-complexity operations. In particular, no matrix inversions are required. These algorithms converge in  $O(1/n)$  (where  $n$  is the number of iterations) for  $r_1, r_2$  satisfying

$$r_1 r_2 \|\Delta_X\| \leq 1.$$

*Remark 2.* Although Algorithms 1 and 2 appear to be specifically engineered to serve a particular purpose, it is important to note that they have been derived in a disciplined manner: the re-formulation of the original, primal obstacle problem as a saddle-point problem and by implementing a primal-dual hybrid gradients algorithm with known convergence guarantees.

**Algorithm 1** Non-linear obstacle problem solver (PD1) and (PD3)**Input:**  $X, \partial X, f, \varphi, r_1, r_2$  such that  $r_1 r_2 \|\Delta_X\| \leq 1$ .Initialize  $u^0, \bar{u}^0, p^0 \leftarrow 0$ **repeat** $\psi^0 \leftarrow p^n$ **repeat** $\varepsilon^t \leftarrow \sqrt{1 - |\psi^t|^2}$  $\psi^{t+1} \leftarrow \frac{\varepsilon^t}{\varepsilon^t + 2r_1} [p^n + r_1 \nabla_X \bar{u}^n]$   $\triangleright$  (3.7)**until** convergence $p^{n+1} \leftarrow \psi^\infty$ **switch** Problem **do****case** (PD1) $u^{n+1} \leftarrow P_{U_{f,\varphi}}[u^n + r_2 \operatorname{div}_X p^{n+1}]$   $\triangleright$  (3.12)**end case****case** (PD3) $u^{n+1} \leftarrow P_{U_f}[\varphi - \operatorname{shrink}_+(\varphi - (u^n + r_2 \operatorname{div}_X p^{n+1}), r_2 \mu)]$   $\triangleright$  (3.17)**end case****end switch** $\bar{u}^{n+1} \leftarrow 2u^{n+1} - u^n$ **until** convergence:  $\|u^{n+1} - u^n\|_\infty < \epsilon$ **Output:** Membrane  $u$ .**Algorithm 2** Linear obstacle problem solver (PD2) and (PD4)**Input:**  $X, \partial X, f, \varphi, r_1, r_2$  such that  $r_1 r_2 \|\Delta_X\| \leq 1$ .Initialize  $u^0, \bar{u}^0, \Sigma^0 \leftarrow 0$ **repeat** $\Sigma^{n+1} \leftarrow \frac{\Sigma^n + \Delta_X \bar{u}^n}{(1+r_1)}$   $\triangleright$  Accumulate sum in (3.20)**switch** Problem **do****case** (PD2) $u^{n+1} \leftarrow P_{U_{f,\varphi}}[u^n + r_1 r_2 \Sigma^{n+1}]$   $\triangleright$  (3.12)**end case****case** (PD4) $u^{n+1} \leftarrow P_{U_f}[\varphi - \operatorname{shrink}_+(\varphi - (u^n + r_1 r_2 \Sigma^{n+1}), r_2 \mu)]$   $\triangleright$  (3.17)**end case****end switch** $\bar{u}^{n+1} \leftarrow 2u^{n+1} - u^n$ **until** convergence:  $\|u^{n+1} - u^n\|_\infty < \epsilon$ **Output:** Membrane  $u$ .

## 4. INTERPRETATION/COMPARISONS FOR PROPOSED ALGORITHM

In this section we compare the proposed primal-dual algorithm (summarized in Algorithms 1 and 2) to two time-stepping schemes arising in classical problems.

**4.1. Comparison to the forward Euler scheme for the discrete heat equation.** For a comparison, recall that the forward Euler time-stepping scheme for the discrete heat equation, can be written

$$(4.1a) \quad u^* = u^n + r_1 r_2 \Delta_X u^n$$

$$(4.1b) \quad = u^0 + r_1 r_2 \sum_{\eta=0}^n \Delta_X u^\eta,$$

where  $r_1 r_2$  is the time step. Thus, after elimination of the dual variable, the primal-dual method for the linear problem *seems* to perform something akin to an alternating sequence of explicit heat diffusion steps (4.1), projection/thresholding (3.12) or (3.17), and an extra gradient step (3.1c). Unfortunately, the explicit heat diffusion (4.1) is well-known to be severely limited by its stability criterion and therefore dreadfully slow. *Why would this be a reasonable thing to do, then?*

Indeed, the first term,  $\eta = 0$ , of the summation in (3.20) is an explicit heat update step, subject to the usual constraint on the time step, here  $r_1 r_2 \leq \rho(\Delta_X)^{-1}$ . However, thanks to the sum, previous iterates are taken into account (discounted by a compounding factor of  $\frac{1}{1+r_1}$ ), the impact of which is essential for the efficiency and convergence of the scheme. Consider the following two cases:

*Away from the solution*, subsequent heat updates point towards the same direction and the sum lets the history of updates interfere constructively, effectively greatly increasing the time step of the explicit heat update. Assuming  $\Delta_X \bar{u}^{n-\eta} \approx \Delta_X u^n$  is constant for recent iterations (small  $\eta$ ), and noting that  $\sum_{\eta=0}^n \frac{1}{(1+r_1)^{\eta+1}} = \frac{1}{r_1}$ , we can approximate (3.20) by

$$u^* \approx u^n + r_2 \Delta_X u^n.$$

The proposed scheme is thus heat diffusion with a  $1/r_1$ -fold larger effective time-step, see (4.1). This is more pronounced for  $r_1 \ll 1$ , reducing the discounting on previous time steps (*i.e.*, increasing the memory horizon of the summation).

*Close to the solution*, the membrane tends to oscillate about the optimum and subsequent heat steps have opposing signs. The summation leads to destructive interference of subsequent updates, dampening the oscillations and effectively stabilizing the explicit heat update. In this case, larger values of  $r_1$  reduce the memory horizon and allow the system to transition from constructive interference to a stabilizing, destructive interference, thereby reducing overshooting.

**4.2. Comparison to a time-stepping scheme for the damped wave equation.** Another interesting analysis of the resulting update scheme, in the linear case and away from obstacle and boundary value projections, emerges when we rewrite it as a three-point scheme in time.

**Lemma 4.1.** *The proposed update iterations of the primal variable (ignoring projections) satisfy the following three-point equality:*

$$(4.2) \quad (u^{n+1} - 2u^n + u^{n-1}) + r_1 (u^{n+1} - u^n) = r_1 r_2 (2\Delta_X u^n - \Delta_X u^{n-1}).$$

*Remark 3.* The left hand terms in (4.2) are immediately identified as finite difference approximations of  $u_{tt}^n \approx (u^{n+1} - 2u^n + u^{n-1})/h^2$  and  $u_t^n \approx (u^{n+1} - u^n)/h$ , and one clearly recognizes elements of a damped wave equation:

$$(4.3) \quad u_{tt}^n + \frac{r_1}{h} u_t^n = \frac{r_1 r_2}{h^2} (2\Delta_X u^n - \Delta_X u^{n-1})$$

*Proof.* By substituting the extragradient (3.1c) into (3.18), we get:

$$(1 + r_1) \operatorname{div}_X p^{n+1} = \operatorname{div}_X p^n + r_1 (2\Delta_X u^n - \Delta_X u^{n-1}).$$

From (3.10) and (3.12)/(3.17), but ignoring shrinkage and projections, we get

$$\operatorname{div}_X p^{n+1} = \frac{u^{n+1} - u^n}{r_2}.$$

Put together, this yields the primal update

$$(4.4) \quad u^{n+1} = u^n + \frac{u^n - u^{n-1}}{1 + r_1} + \frac{r_1 r_2}{1 + r_1} (2\Delta_X u^n - \Delta_X u^{n-1}).$$

from which (4.2) follows by simple term rearrangements.  $\square$

The three-point primal update scheme (4.4) highlights how the proposed scheme performs explicit heat diffusion (based on the forward-projected Laplacian, in the extragradient step), on top of which the most recent update is added. This scheme is related to the “*momentum method*”—because it mimicks the motion of a system  $u$  with a certain mass (and therefore inertia) [19]. This type of gradient descent acceleration method is classically used in gradient-descent back-propagation training of neural networks [20], and has been associated with a specific type of conjugate gradients [4]. Also, the idea of maintaining momentum by mixing in an extrapolation of most recent updates is the essence of Nesterov’s gradient acceleration scheme [18] and the core of the FISTA algorithm [3].

The evolution of a simple toy problem under both, standard Euler explicit heat updates (4.1) and the proposed primal-dual scheme (3.20) is illustrated in Figure 1. It is easy to see how the summation in (3.20) causes the peak to decay at a faster pace after the first iterate. As a result, the initial localized state splits into two components that propagate away from the center and decay, illustrating the damped wave nature of the iterations. The wave is reflected at the fixed boundaries, inverting the sign, in contrast to the explicit scheme which remains non-negative (according to the maximum principle). Evidently, the proposed scheme converges to the (trivial) flat solution  $u = 0$  much faster than the explicit heat update.

## 5. NUMERICAL EXPERIMENTS

In all following problems, we assume  $X$  to be a Cartesian sampling of either an interval in  $\mathbb{R}$ , or of a rectangular subset of  $\mathbb{R}^2$ . The differential operators are chosen as forward differences for the gradient, and, as a result, backward differences for the divergence and central differences for the Laplacian.

We implemented the algorithms in MATLAB<sup>®</sup><sup>1</sup>, and ran our experiments on standard computing equipment (2012 laptop with 2.80 GHz Intel<sup>®</sup> Core<sup>™</sup> i7, with 4 GB RAM). For comparison, we also ran the same experiments on a 2015 high-end machine (20-core 3.0 GHz Intel<sup>®</sup> Xeon<sup>®</sup>, with 128 GB of RAM)—results were

<sup>1</sup>Code available at <http://www.math.ucla.edu/~zosso/code.html>

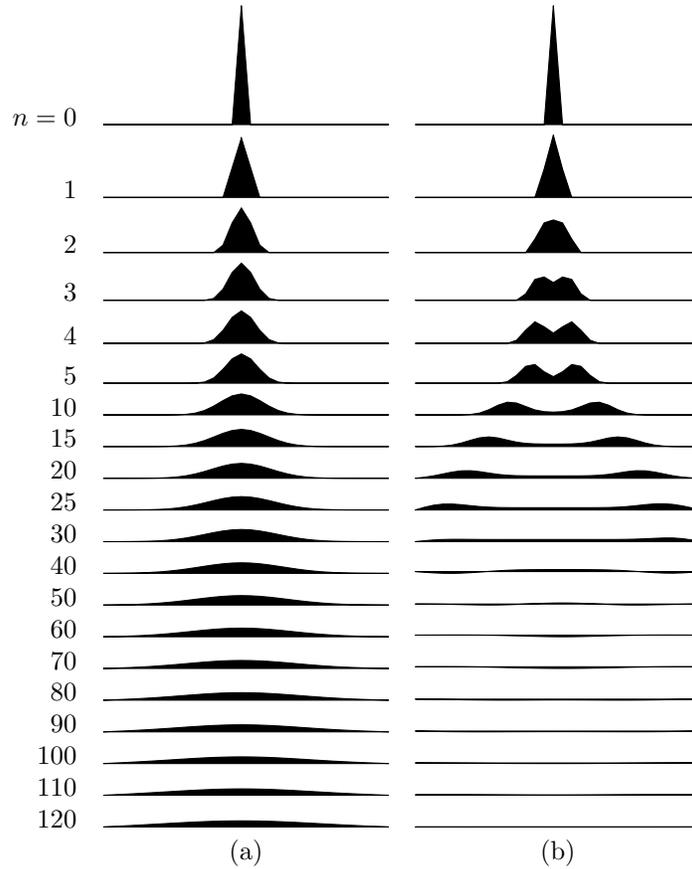


FIGURE 1. Toy example problem: obstacle-free linearized minimal surface problem (PD2) on 32-point 1D domain discretization,  $f = 0$  boundary conditions, and single peak initialization at the center. Evolution of the membrane over iterations  $n$ . (a) standard Euler explicit heat update (4.1). (b) our primal-dual scheme (3.20). Black area indicates error with respect to (trivial) solution.

identical and the observed running times not dramatically different primarily because the code is implemented in serial and cannot immediately benefit from the extra cores and memory.

**5.1. One-dimensional obstacles.** We consider two obstacles on  $[0, 1]$ , previously considered in [23, 26]:

$$(5.1) \quad \varphi_1(x) := \begin{cases} 100x^2 & \text{for } 0 \leq x \leq 0.25 \\ 100x(1-x) & \text{for } 0.25 \leq x \leq 0.5 \\ \varphi_1(1-x) & \text{for } 0.5 \leq x \leq 1, \end{cases}$$

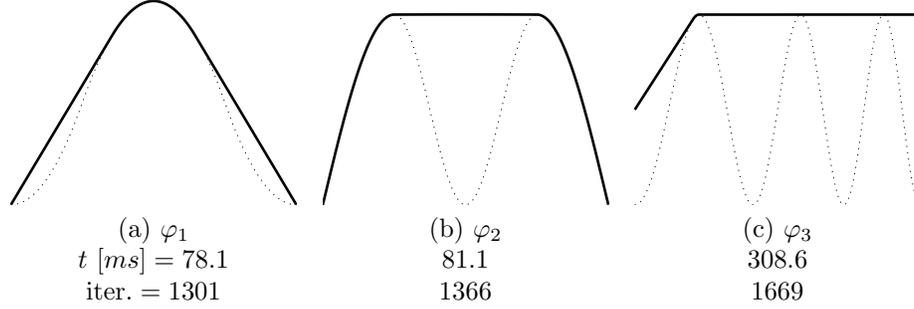


FIGURE 2. 1D obstacles  $\varphi_1$  through  $\varphi_3$  (dotted) and respective solutions to the obstacle problem (solid). (a)–(b) Linear obstacle problem.  $N = 256$ ,  $\mu = 5$ , boundary conditions  $u(0) = u(1) = 0$ . (c) Non-linear obstacle problem.  $N = 512$ ,  $\mu = \infty$ ,  $u(0) = 5$  and  $u(1) = 10$ . For all three:  $r_1 = 0.01$ ,  $r_2 = 25$ ,  $\epsilon = 10^{-5}$ .

and

$$(5.2) \quad \varphi_2(x) := \begin{cases} 10 \sin(2\pi x) & \text{for } 0 \leq x \leq 0.25 \\ 5 \cos(\pi(4x - 1)) + 5 & \text{for } 0.25 \leq x \leq 0.5 \\ \varphi_1(1 - x) & \text{for } 0.5 \leq x \leq 1. \end{cases}$$

A third obstacle (from [26]) is given by

$$(5.3) \quad \varphi_3(x) := 10 \sin^2(\pi(x + 1)^2), \quad x \in [0, 1].$$

All three problems and the obtained solutions are illustrated in Figure 2. In the 1D case, there is of course no difference between the solutions of the linear and the non-linear versions: the solutions are identical to the obstacle on the contact set, and straight lines away from it.

**5.2. Two-dimensional obstacles.** We now look at example problems in 2D (from [26]). First, consider the square domain,  $\varphi_4: \Omega \rightarrow \mathbb{R}$ , with  $\Omega = [0, 1]^2$ , and the obstacle

$$(5.4) \quad \varphi_4(x, y) := \begin{cases} 5 & \text{for } |x - 0.6| + |y - 0.6| < 0.04 \\ 4.5 & \text{for } (x - 0.6)^2 + (y - 0.25)^2 < 0.001 \\ 4.5 & \text{for } y = 0.57 \text{ and } 0.075 < x < 0.13 \\ 0 & \text{otherwise.} \end{cases}$$

We solve the linear and non-linear obstacle problem (with 0 boundary values imposed) at different resolutions, as illustrated in Figure 3. Parameter values are as follows:  $\mu = \infty$ ,  $r_1 = 0.01$ ,  $r_2 = 12.5$ ,  $\epsilon = 10^{-5}$ .

In Table 1 we can compare to numerical experiments of previous algorithm implementations, in particular the  $L^1$ -penalty method [26], and the Lions-Mercier splitting scheme [16], equally reported in [26]. These comparisons hold only in a loose sense, because the implementation and computing infrastructure is undisclosed for [26]. Nevertheless, our method seems to outperform by 1-2 orders of magnitude.

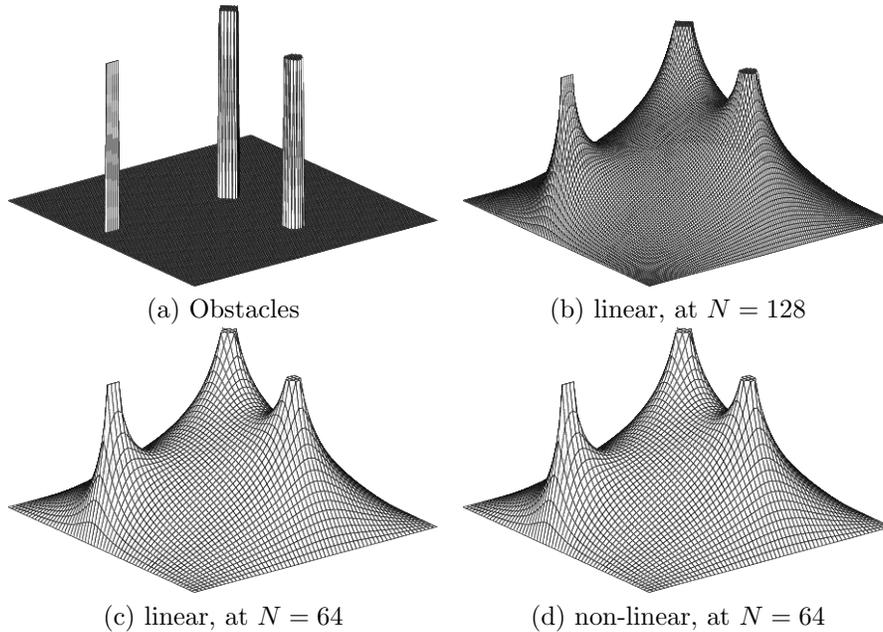


FIGURE 3. Solution to the linear and nonlinear problems (PD1) and (PD2) for a 2D obstacle,  $\varphi_4(x, y)$  in (5.4). (a) The obstacles at  $128 \times 128$ . (b) The linear obstacle problem solution computed at medium resolution  $128 \times 128$ . (c)  $64 \times 64$  low-resolution solution of the linear problem. (d) The low-resolution non-linear obstacle problem solution.

TABLE 1. Running times for 2D obstacle problem  $\varphi_4$ , at different resolution. Values of [26], [16] provided for very approximate comparison only: Figures for [16] were reported in [26], so presumably run on the same, unspecified platform/computing infrastructure.

Size	Our method		Tran et al. [26]		Lions/Mercier [16]	
	$t$ [s]	#iter.	$t$ [s]	#iter.	$t$ [s]	#iter.
$64 \times 64$	0.39	1416	20.8	315	55.1	3508
$128 \times 128$	1.91	1562	57.7	337	193.7	3997
$256 \times 256$	10.6	1707	222.3	469	1158.3	4383

TABLE 2. Running times for spherical 2D problem  $\varphi_5$ , at different resolution. Values of [17] provided for very approximate comparison only: [17] was implemented with Fortran 77 and run on an HP9000/J5600 workstation ( $2 \times 552$  MHz PA8600 CPU).

Size	Our method		Majava/Tai [17]	
	$t$ [s]	#iter.	$t$ [s]	#iter.
$64 \times 64$	0.55	1827	1862	$1.3 \cdot 10^6$
$128 \times 128$	3.4	2607	51064	$9.6 \cdot 10^6$
$256 \times 256$	18.0	2855	n/a	n/a

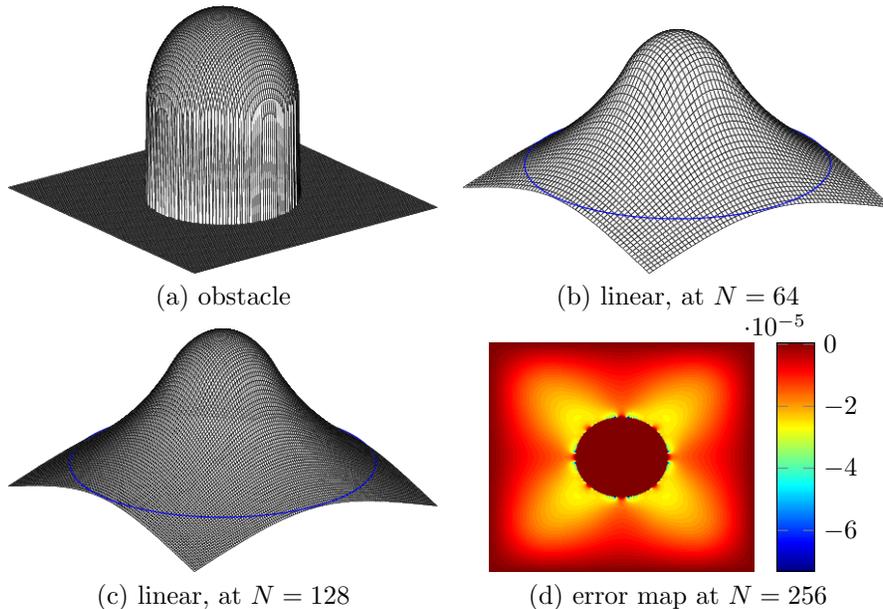


FIGURE 4. Radially symmetric 2D obstacle problem. (a) Obstacle  $\varphi_5(x, y)$ . (b)–(c) Low- and medium-resolution solution and circular zero-level set (blue). (d) Error map of the high-resolution solution  $256 \times 256$ . The error is concentrated as peaks near the free boundary, mostly due to spatial discretization, and propagating from there into the free membrane parts.

Next, we consider the square domain  $\Omega = [-2, 2]^2$ , and the radially-symmetric obstacle

$$(5.5) \quad \varphi_5(r) := \begin{cases} \sqrt{1-r^2} & \text{for } r \leq 1, \\ -1 & \text{otherwise.} \end{cases}$$

This obstacle has been used in [17, 26], and can be traced back to at least [25], albeit with a typo in its definition. Assuming  $f = 0$  on a circular boundary of radius  $R = 2^2$ , the linear obstacle problem admits the following radially-symmetric analytical solution:

$$(5.6) \quad u_5^*(r) = \begin{cases} \sqrt{1-r^2} & \text{for } r \leq r^*, \\ -(r^*)^2 \ln(r/R) / \sqrt{1-(r^*)^2} & \text{otherwise,} \end{cases}$$

where  $r^* = 0.6979651482\dots$  satisfies  $(r^*)^2(1 - \ln(r^*/R)) = 1$ .

Using the values of this analytical solution as prescribed boundary values on the boundaries of the square domain, we solve the linear obstacle problem. We set the parameters as follows:  $\mu = 0.1$ ,  $r_1 = 0.008$ ,  $r_2 = 15.625$ ,  $\epsilon = 10^{-7}$ . Results are shown in Figure 4. The pointwise error is of the order of  $10^{-5}$  and concentrated at the circular free boundary, mainly as a result of discretization. This type of error is comparable to the results reported in [17] and [26].

<sup>2</sup>None of these sources explicitly specifies the boundary values other than claiming  $f = 0$  on “some” (undisclosed) location, clearly different from the square border  $\partial\Omega$ .

For this obstacle problem, we can compare to numerical experiments of previous algorithm implementations, see Table 2. In particular, we compare the runtime and number of iterations with the levelset gradient-method [17]. Even assuming that the 2004 computing times of [17] may be discounted by Moore's law (by half each 18 months, about a factor of 160-200 overall), we note that our method clearly outperforms by 1-2 orders of magnitude.

**5.3. Generalization to a double obstacle with forcing.** A more general form of the obstacle problem deals with two obstacles, bounding the membrane from below ( $\varphi$ ) and above ( $\psi$ ), and the inclusion of a force term  $v$  acting vertically on the membrane:

$$(5.7) \quad \min_u \int_{\Omega} |\nabla u|^2 - uv \quad \text{s.t.} \quad \psi \geq u \geq \varphi, \quad u|_{\partial\Omega} = f.$$

It is cumbersome but straightforward to extend the proposed algorithms to this case: the double obstacle results in a second shrinkage term (albeit with opposite signs), and the force  $v$  becomes an extra term in the primal variable update,  $u^* = u^n + r_2 \operatorname{div}_X p^{n+1} + v$ . The exact details are left as an exercise to the reader.

As an illustration of the performance of this slightly extended algorithm, we consider an elastic-plastic torsion problem described in [27]. Let  $\Omega = [0, 1]^2$  and  $X$  a Cartesian sampling thereof. The obstacles are constructed as

$$(5.8a) \quad \varphi_6(x, y) := -\operatorname{dist}(x, \partial\Omega), \quad \text{and}$$

$$(5.8b) \quad \psi_6(x, y) := 0.2.$$

The boundary values are  $u = 0$  on  $\partial\Omega$ . The force  $v: \Omega \rightarrow \mathbb{R}$  is defined as

$$(5.9) \quad v(x, y) := \begin{cases} 300, & \text{if } (x, y) \in S := \{(x, y) : |x - y| \leq 0.1 \wedge x \leq 0.3\}, \\ -70e^y g(x), & \text{if } (x, y) \notin S \wedge x \leq 1 - y, \\ 15e^y g(x), & \text{if } (x, y) \notin S \wedge x > 1 - y, \end{cases}$$

where we further have

$$(5.10) \quad g(x) := \begin{cases} 6x, & \text{if } 0 \leq x \leq 1/6, \\ 2(1 - 3x), & \text{if } 1/6 < x \leq 1/3, \\ 6(x - 1/3), & \text{if } 1/3 < x \leq 1/2, \\ 2(1 - 3(x - 1/3)), & \text{if } 1/2 < x \leq 2/3, \\ 6(x - 2/3), & \text{if } 2/3 < x \leq 5/6, \\ 2(1 - 3(x - 2/3)), & \text{if } 5/6 < x \leq 1. \end{cases}$$

The algorithm on a coarse grid  $64 \times 64$  (as in [27]), with  $\mu = 0.1$ ,  $r_1 = 0.008$ ,  $r_2 = 15.625$ , and for a tolerance  $\epsilon = 10^{-6}$ , converges in 1104 iterations, requiring only 0.45 seconds. The algorithm presented in [27] is reported to perform in 916.8-1275.1 seconds, which is more than three orders of magnitude slower. The computed minimum-energy membrane and the associated contact set (coincidence set) are shown in Figure 5, and compare very well to the results in [27].

## 6. CONCLUSIONS AND OUTLOOK

In this paper we have proposed a primal-dual hybrid gradients based method for the efficient solution of the obstacle problem. We consider both the non-linear

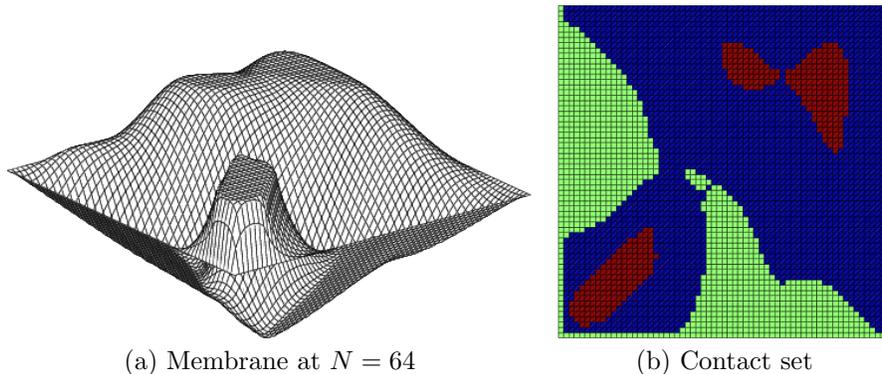


FIGURE 5. Double obstacle with force: elasto-plastic torsion problem. (a) Computed optimal membrane  $u$ . (b) Computed contact set: free membrane (blue), contact with upper obstacle (red), and contact with lower obstacle (green). Compare with results in [27].

and linearized versions of the discrete obstacle problem, as well as both the original, constrained formulation and the  $L^1$ -penalty relaxation. All these problems are convex minimization problems, and we start by reformulating them as primal-dual problems, based on the Legendre-Fenchel transform (convex conjugate) of the surface area and the Dirichlet energy, respectively. The resulting saddle-point problem is solved by the primal-dual hybrid gradients method, which consists of three iterative steps: the dual and primal variable proximal updates, and an extra-gradient step (overrelaxation) of the primal variable. The proximal updates can be solved efficiently, and in the linear case even particularly so. Indeed, the linear case allows eliminating the dual variable altogether, resulting in primal-variable update and extragradient scheme, only. We demonstrate on various 1D and 2D problems that the resulting scheme outperforms current state-of-the-art methods by 1-2 orders of magnitude.

In addition to being efficient, the proposed algorithm also benefits from a highly interesting physical interpretation, as discussed in Sections 3.4 and 4.2. Firstly, the elimination of the dual variable results in a single variable scheme that is strongly reminiscent of gradient descent, except for the fact that previous iterates remain involved. As a result, there is build-up of a certain “momentum” that accelerates the updates beyond the limits of the usual CFL step-size criteria, see Section 3.4. Indeed, away from the solution the previous iterates interfere constructively resulting in a greatly increased apparent time step; whereas closer to the solution the membrane has a tendency to oscillate, and the iterates cancel out leading to increased stability. Similarly, the scheme can be brought into a form that is highly reminiscent of a damped wave equation (4.2). This nature is also particularly illustrated by the toy-example considered in Figure 1, where the momentum splits an initial central state into two separate parts, who then propagate away laterally and eventually dissipate.

A general advantage of the proposed primal-dual based method for the obstacle problem is the absence of matrix/operator inversions, since the proximal updates only contain explicit operations involving adjoints, at most. Also, in analogy to [23,

26] but in contrast to e.g., [12, 13], the proposed method does not need to track the contact set explicitly; indeed, the method handles the interface between coincidence set and harmonic membrane as a free boundary, intrinsically. Finally, it is to note that multigrid and multilevel methods that have been employed to speed up computation in different approaches such as [28] may be applicable to the proposed method as well, expected to result in a further speed up of the computations.

We remark that the properties of the proposed algorithm are not specifically engineered, but emerge from a very disciplined derivation from convex optimization, see Remark 2. More applications of this method, including elliptic PDE problems, will be reported in [33].

#### ACKNOWLEDGMENTS

The authors thank W. Feldman, I. Kim, and G. Tran for helpful discussions.

#### REFERENCES

- [1] Kenneth J Arrow, Leonid Hurwicz, and Hirofumi Uzawa. *Studies in linear and non-linear programming*. Cambridge Univ. Press, 1958.
- [2] Hedy Attouch, Giuseppe Buttazzo, and Gérard Michaille. *Variational analysis in Sobolev and BV spaces: applications to PDEs and optimization*. SIAM, second edition, 2014.
- [3] Amir Beck and Marc Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–34, November 2009.
- [4] Amit Bhaya and Eugenius Kaszkurewicz. Steepest descent with momentum for quadratic functions is a version of the conjugate gradient method. *Neural networks : the official journal of the International Neural Network Society*, 17(1):65–71, January 2004.
- [5] Dietrich Braess, Carsten Carstensen, and Ronald H. W. Hoppe. Convergence analysis of a conforming adaptive finite element method for an obstacle problem. *Numerische Mathematik*, 107(3):455–471, July 2007.
- [6] Haïm Brezis and Moïse Sibony. Méthodes d’approximation et d’itération pour les opérateurs monotones. *Archive for Rational Mechanics and Analysis*, 28(1):59–82, 1968.
- [7] Luis A. Caffarelli. The obstacle problem revisited. *The Journal of Fourier Analysis and Applications*, 4(4-5):383–402, July 1998.
- [8] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, December 2011.
- [9] Ernie Esser, Xiaoqun Zhang, and Tony F. Chan. A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIAM Journal on Imaging Sciences*, 3(4):1015–1046, January 2010.
- [10] Avner Friedmann. *Variational principles and free-boundary problems*. Wiley, New York, 1982.
- [11] Roland Glowinski. *Numerical methods for nonlinear variational problems*. Springer Science & Business Media, 2013.
- [12] Michael Hintermüller, Kazufumi Ito, and Karl Kunisch. The primal-dual active set strategy as a semismooth newton method. *SIAM Journal on Optimization*, 13(3):865–888, January 2002.
- [13] Michael Hintermüller, Victor A. Kovtunencko, and Karl Kunisch. Obstacle problems with cohesion: a hemivariational inequality approach and its efficient numerical solution. *SIAM Journal on Optimization*, 21(2):491–516, April 2011.
- [14] Ronald H. W. Hoppe. Multigrid algorithms for variational inequalities. *SIAM Journal on Numerical Analysis*, 24(5):1046–1065, October 1987.
- [15] Claes Johnson. Adaptive finite element methods for the obstacle problem. *Mathematical Models and Methods in Applied Sciences*, 02(04):483–487, December 1992.
- [16] Pierre-Louis Lions and Bertrand Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM J. Numer. Anal.*, 16(6):964–979, 1979.

- [17] Kirsi Majava and Xue-Cheng Tai. A level set method for solving free boundary problems associated with obstacles. *International Journal of Numerical Analysis and Modeling*, 1(2):157–171, 2004.
- [18] Yurii Nesterov. A method of solving a convex programming problem with convergence rate  $O(1/k^2)$ . *Soviet Mathematics Doklady*, 27(2):372–376, 1983.
- [19] Boris T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, January 1964.
- [20] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1):145–151, January 1999.
- [21] Ralph Tyrrell Rockafellar. *Conjugate Duality and Optimization*. Society for Industrial and Applied Mathematics, January 1974.
- [22] José-Francisco Rodrigues. *Obstacle problems in mathematical physics*. Elsevier Science Publishers, Amsterdam, 1987.
- [23] Reinhard Scholz. Numerical solution of the obstacle problem by the penalty method. *Computing*, 32(4):297–306, December 1984.
- [24] Nir Sochen, Ron Kimmel, and Ravi Malladi. A general framework for low level vision. *IEEE Transactions on Image Processing*, 7(3):310–318, 1998.
- [25] Xue-Cheng Tai. Rate of convergence for some constraint decomposition methods for nonlinear variational inequalities. *Numerische Mathematik*, 93(4):755–786, February 2003.
- [26] Giang Tran, Hayden Schaeffer, William M. Feldman, and Stanley J. Osher. An  $L^1$  penalty method for general obstacle problems. *SIAM Journal on Applied Mathematics*, 75(4):1424–1444, July 2015.
- [27] Fei Wang and Xiao-Liang Cheng. An algorithm for solving the double obstacle problems. *Applied Mathematics and Computation*, 201(1-2):221–228, July 2008.
- [28] Yongmin Zhang. Multilevel projection algorithm for solving obstacle problems. *Computers & Mathematics with Applications*, 41(12):1505–1513, June 2001.
- [29] Mingqiang Zhu and Tony Chan. An efficient primal-dual hybrid gradient algorithm for total variation image restoration. Technical report, UCLA CAM Report 08-34, 2008.
- [30] Mingqiang Zhu, Stephen J. Wright, and Tony F. Chan. Duality-based algorithms for total-variation-regularized image restoration. *Computational Optimization and Applications*, 47(3):377–400, December 2010.
- [31] Dominique Zosso and Aurelien Bustin. A primal-dual projected gradient algorithm for efficient Beltrami regularization. *submitted to: Computer Vision and Image Understanding – Special Issue on Discrete and Continuous Geometry in Computer Vision*, page 17, 2014.
- [32] Dominique Zosso and Braxton Osting. A minimal surface criterion for graph partitioning. *submitted to: AIMS J. Inverse Problems and Imaging*, page 30, 2015.
- [33] Dominique Zosso, Braxton Osting, and Stanley J. Osher. Elliptic PDE and primal-dual optimization methods. *in preparation*, 2015.

(D. Zosso, M. Xia, S. J. Osher) DEPARTMENT OF MATHEMATICS, UNIVERSITY OF CALIFORNIA, LOS ANGELES, LOS ANGELES, CA 90095-1555, USA

*E-mail address*, Corresponding author: `zosso@math.ucla.edu`

*E-mail address*: `xiamengqi2012@ucla.edu`

*E-mail address*: `sjo@math.ucla.edu`

(B. Osting) DEPARTMENT OF MATHEMATICS, UNIVERSITY OF UTAH, SALT LAKE CITY, UT 84112, USA

*E-mail address*: `osting@math.utah.edu`