# A NEWTON DIV-CURL LEAST-SQUARES FINITE ELEMENT METHOD FOR THE ELLIPTIC MONGE-AMPÈRE EQUATION

CHAD R. WESTPHAL*

**Abstract.** This paper develops a new finite element approach for the efficient approximation of classical solutions of the elliptic Monge-Ampère equation. We use an outer Newton-like linearization and a first-order system least-squares reformulation at the continuous level to define a sequence of first-order div-curl systems. For problems on convex domains with smooth and appropriately bounded data, this framework gives robust results: convergence of the nonlinear iteration in a small number of steps, and optimal finite element convergence rates with respect to the meshsize. Numerical results using standard piecewise quadratic or cubic elements for all unknowns illustrate convergence results.

**Key words.** Monge-Ampère, Fully nonlinear partial differential equations, Least-squares finite element methods

**AMS subject classifications.** 35J96, 65N30, 65N12

**1. Introduction and Background.** In this paper we develop a finite element method for the fully nonlinear elliptic Monge-Ampère equation given by

$$\begin{cases} det(D^2u) = f & \text{in } \Omega \\ u = g & \text{on } \partial\Omega, \end{cases} \tag{1.1}$$

where $\Omega$ is a Lipschitz smooth convex domain in $\mathbb{R}^2$ with boundary $\partial\Omega$, $D^2u$ is the Hessian matrix of the unknown $u$, and $det(D^2u) = u_{xx}u_{yy} - u_{xy}^2$ is its determinant. We assume that $f$ and $g$ are sufficiently smooth and that $f$ is positive almost everywhere in $\Omega$. We focus here on problems with smooth solutions, where generally $u \in H^2(\Omega)$ for reasons inherent to both the linearization approach and the underlying finite element discretization used. While the well-known regularity conditions of linear second-order elliptic problems do not apply here, there is a vibrant ongoing research on regularity for this and closely related problems; see, for example [25, 13, 8, 27].

While this remains a challenging numerical problem, the literature reflects many successful numerical approaches developed recently. The review article [16], gives a thorough summary of the relevant applications, numerical challenges, and history to-date of the work on this and other closely related fully nonlinear problems. Among the challenges, the two main features we focus on here is the treatment of the nonlinear iteration and the discretization method developed. Applications of problems relating to (1.1) include various topics in differential geometry, the prescribed Gauss curvature problem, and the optimal mass transport problem in the design of lenses and reflectors; see [2, 26] for example.

A considerable amount of work for this problem has been developed recently in the context of finite difference discretizations. In [24], Oberman considers the Monge-Ampère operator as a function of the eigenvalues of the Hessian and uses a wide finite difference scheme to approximate the eigenvalues, resulting in convergence to a viscosity solution independent of the regularity of the problem. The iteration requires an explicit time stepping scheme subject to a CFL condition. Taking a more traditional discretization approach, in [3], Benamou, Froese, and Oberman pose

*Department of Mathematics and Computer Science, Wabash College, Crawfordsville, IN 47933 (westphac@wabash.edu)

two finite difference approaches. The first directly uses a standard central difference scheme to discretize (1.1) as a system of quadratic equations, where a locally convex solution is found by the selection of the appropriate root at each step. Their second approach involves rearranging terms in (1.1) and linearizing the equation by freezing some of the terms from a previous iteration (a trick first noted in [14]). This is a type of Picard iteration and results in a simple sequence of Poisson solves discretized by finite differences. The two approaches are able to handle some problems with nonsmooth components, but may require a large number of iterations to converge. Additionally, in [1], Awanou considers regularization of nonsmooth data and standard finite difference discretizations proving convergence to viscosity solutions.

In the context of finite $x^2$ element methods, which can be more sensitive to regularity issues, many notable approaches have been proposed recently as well. In [15] for example, Dean and Glowinski discuss two finite element approaches. The first formulates the problem in a constrained optimization framework that leads to a saddle-point problem, which is solved by iterating between two discretized biharmonic problems. Their second formulation directly poses a least-squares solution as the minimizer of the residual of (1.1) in the $L^2$ norm and requires the introduction of a time discretization to which an operator splitting technique is applied. At each discretized time step the problem uses a mixed finite element formulation leading to iterating between solving a nonlinear system of algebraic equations and a discrete variational problem involving products of second order derivatives. In work following the spirit of this second approach, in [7], Caboussat, Glowinski, and Sorensen again begin by the idea of minimizing the residual of (1.1), leading to an relaxation type of iteration between two discrete minimization problems. Since the discretization involves second order derivatives of finite element functions, a special smoothing procedure is required to retain smoothness between iterations. In [6], Brenner, Gudi, Neilan, and Sung develop a $C^0$ penalty method that is able to achieve discretization convergence rates that are essentially optimal for finite element spaces up to degree 4. While the approach we develop in this paper shares some of the same overall motivating ideas to many of these methods, our approach handles the linearization via Newton's method completely at the continuous level as an outer iteration, then reformulates each subsequent linearized step as the least-squares solution of a div-curl system. We also note that our approach does not require any computation of edge terms and only uses simple conforming Lagrange finite element spaces.

The use of Newton's method in this context as an outer iteration is, of course, not new. It is well known that the linearized Monge-Ampère operator is a second-order elliptic operator with coefficients from the second derivatives of the previous iteration. In [22], Loeper and Rapetti prove convergence of the Newton iteration assuming periodic boundary conditions. In that paper, they illustrate the approach by discretizing each step with a simple second-order finite difference scheme on a uniform Cartesian grid. For their test cases they find that only a few Newton steps (approximately 10) are required to converge to the level of discretization error. Similarly, in [17], Froese and Oberman use an outer Newton linearization and an inner wide-stencil finite difference discretization. For smooth problems, they also find that fewer than 10 iterations is needed, while singular problems may require more. While in this paper we handle the inner iteration quite differently, we observe the same robust convergence in Newton's method as an outer iteration. Additionally, in [20], Lakkis and Pryer utilize Newton's method coupled with a Galerkin nonvariational finite element method.

There are many strong examples in the literature of using least squares finite

element methods based on a div-curl system. Additionally, examples of combining a Newton outer iteration with a well-formulated least squares discretization can be found in [12, 23]. The general framework for div-curl least squares functional minimization is established in [9, 10], and [18] provides a general overview of the least-squares finite element approach.

In section 2, we give details on the main numerical approach that we focus on in this paper. In section 3 we provide three numerical examples that illustrate the compelling features of the method, comparing results to other published works as available. All computations are done in FreeFem++ [19]. And finally, in section 4 we discuss extensions of the methodology presented here and give brief concluding thoughts on extensions of the basic idea presented here.

**2. Numerical Methods.** In this section we give details of the numerical algorithm for approximating solutions to (1.1). We assume throughout that $\Omega$ is a Lipschitz smooth convex domain in $\mathbb{R}^2$. We use standard Sobolev spaces $L^2(\Omega)^d$, $H^1(\Omega)^d$, $H(div)$ and $H(curl)$, where $d = 1, 2$, or $2 \times 2$, and generally omit the dimension when it is clear by context. For $V \in H^1(\Omega)^2$, we use the quantities

$$\begin{aligned} \nabla \cdot V &= \partial_x(V_1) + \partial_y(V_2), \\ \nabla \times V &= \partial_x(V_2) - \partial_y(V_1), \text{ and} \\ \nabla V &= \begin{pmatrix} \partial_x(V_1) & \partial_y(V_1) \\ \partial_x(V_2) & \partial_y(V_2) \end{pmatrix}. \end{aligned}$$

For $A, B \in L^2(\Omega)^{2\times 2}$, we use $A : B = \sum_{i,j=1}^{2} A_{ij}B_{ij}$ to denote the Frobenius product, and use

$$\nabla \cdot A = \begin{pmatrix} \partial_x(A_{11}) + \partial_y(A_{12}) \\ \partial_x(A_{21}) + \partial_y(A_{22}) \end{pmatrix},$$

where, if components of $A$ are defined element-wise, then components of $\nabla \cdot A$ are also taken element-wise.

As a motivating numerical approach, we review an idea presented in [14, 3], which provides motivation for the new approach. In particular, we focus on the linearization and discretization procedures as the two main components of the overall algorithms.

**Motivating Idea: (Picard/Galerkin)**. This method reformulates (1.1) to allow a simple outer Picard iteration to define a sequence of linear problems that can each be solved by a straightforward Galerkin finite element closure.

Combining $u_{xx}u_{yy} - u_{xy}^2 = f$ from (1.1) with the identity $(\Delta u)^2 = (u_{xx} + u_{yy})^2 = u_{xx}^2 + 2u_{xx}u_{yy} + u_{yy}^2$ results in $(\Delta u)^2 = u_{xx}^2 + u_{yy}^2 + 2u_{xy}^2 + 2f$. Solving for $\Delta u$ and using the positive square root to be consistent with convexity, the right-hand side can be *frozen* so that the equation is linearized:

$$\Delta u = \left(\tilde{u}_{xx}^2 + \tilde{u}_{yy}^2 + 2\tilde{u}_{xy}^2 + 2f\right)^{1/2},$$

where $\tilde{u}$ is a current approximation and $u$ is the new approximation. In [3] this is analyzed as a fixed-point iteration and shown to converge for problems with solutions in $H^2(\Omega)$. In practice, discretizing with finite differences and with finite elements have similar results overall for smooth problems. The number of nonlinear iterations depends on the smoothness of the solutions and finite element mesh redistribution can speed up convergence for less smooth examples. We focus here on the finite element approach.

To provide a frame of reference we implement this Picard/Galerkin approach in the following way. Let $\Omega^h$ be a quasiuniform triangulation of $\Omega$ with $n$ elements per side and meshsize parameter defined as $h = 1/n$. Let $\mathbf{V}^h$ represent the standard Lagrangian finite element spaces of order $p = 2$ or 3 (denoted as P2 and P3 respectively), equipped with Dirichlet boundary conditions. We also denote $\tilde{u}^h \in \mathbf{V}^h$ as the current approximation to $u$ and define

$$F(\tilde{u}^h) := \left((\tilde{u}^h_{xx})^2 + (\tilde{u}^h_{yy})^2 + 2(\tilde{u}^h_{xy})^2 + 2f\right)^{1/2}$$

on each element. Each iteration is the variational problem:

$$\text{Find } u^h \in \mathbf{V}^h \text{ such that } \langle \nabla u^h, \nabla v^h \rangle = \langle -F(\tilde{u}^h), v^h \rangle \quad \forall v^h \in \mathbf{V}^h. \tag{2.1}$$

It should be noted here that on each element if $\tilde{u}^h$ is continuous P2/P3 then the second derivative terms in $F$ are discontinuous and P0/P1. Thus, for variational problem 2.1, $F(\tilde{u}^h)$ is computed element-wise. Since this linearization results in a sequence of Poisson solves with data in $L^2(\Omega)$, the overall algorithm is described simply in algorithm 1.

**Algorithm 1** Picard/Galerkin Framework

(0) Initialize $\tilde{u}^h = 0$.
(1) Define $F(\tilde{u}^h)$ and solve problem (2.1) for $u^h$.
(2) $\tilde{u}^h \leftarrow u^h$.
(3) Test for convergence, repeat from (1) or stop.

Algorithm 1 has some attractive features, most notably its simplicity and divergence structure. Each step is a straightforward Poisson solve and the finite element framework provides a natural way to incorporate mesh refinement/redistribution. However, the required number of iterations in the outer linearization may be large. Additionally, since $F(\tilde{u}^h)$ is discontinuous, standard finite element theory indicates that discretization rates will likely be no better than $\mathcal{O}(h^2)$, even when the solution and domain are smooth (e.g., see [5]). Below, to address these concerns, we develop a new approach that combines a Newton linearization with a first-order system least-squares finite element discretization.

**Proposed Method: (Newton/Least Squares)**. Applying a Newton linearization to the first equation in (1.1) about current convex approximation $\tilde{u}$ and rearranging terms, we arrive at

$$\nabla \cdot (A(\tilde{u})\nabla u) = f + det(D^2\tilde{u}), \tag{2.2}$$

where $u$ is the new approximation and the coefficient matrix is the cofactor matrix of $D^2\tilde{u}$,

$$A(\tilde{u}) = \begin{pmatrix} \tilde{u}_{yy} & -\tilde{u}_{xy} \\ -\tilde{u}_{xy} & \tilde{u}_{xx} \end{pmatrix}.$$

We note here that when $\tilde{u}$ is convex $A(\tilde{u})$ is definite since $det(A(\tilde{u})) = \tilde{u}_{xx}\tilde{u}_{yy} - \tilde{u}^2_{xy} > 0$. As a continuous problem with sufficient smoothness assumptions, the Newton iteration would be relatively straightforward. In a standard finite element setting, however, there are immediate practical difficulties. Most notably, unless $\tilde{u}^h$ is represented as a $C^2$ function, then $A(\tilde{u}^h)$ will be discontinuous and the standard Galerkin

closure (using integration by parts) will introduce additional boundary terms on each element. Instead, we develop a div-curl least-squares variational problem that allows piecewise discontinuous coefficients and retains smoothness of the iterates by the use of a flux variable. Define $U = \nabla u$ and $\tilde{U} = \nabla \tilde{u}$, and note that we may write $\nabla \cdot (A(\tilde{u})\nabla u) = \nabla \cdot (\tilde{A}U)$ and $\nabla \times \tilde{U} = \partial_x(\tilde{U}_2) - \partial_y(\tilde{U}_1) = 0$, which uses $\tilde{A}$ taken symmetrically as

$$\tilde{A} = \begin{pmatrix} \partial_y \tilde{U}_2 & -\frac{1}{2}(\partial_y \tilde{U}_1 + \partial_x \tilde{U}_2) \\ -\frac{1}{2}(\partial_y \tilde{U}_1 + \partial_x \tilde{U}_2) & \partial_x \tilde{U}_1 \end{pmatrix}.$$

It thus follows that $\nabla \cdot \tilde{A} = \mathbf{0}$, which means that

$$\begin{aligned} \nabla \cdot (\tilde{A}U) &= \tilde{A} : \nabla U + (\nabla \cdot \tilde{A}) \cdot \nabla U = \tilde{A} : \nabla U \\ &= \partial_y(\tilde{U}_2)\partial_x(U_1) - \frac{1}{2}(\partial_y(\tilde{U}_1) + \partial_x(\tilde{U}_2))(\partial_y(U_1) + \partial_x(U_2)) + \partial_x(\tilde{U}_1)\partial_y(U_2). \end{aligned}$$

Thus, (2.2) may be replaced by the system

$$\begin{cases} \tilde{A} : \nabla U = \mathcal{F}(\tilde{U}), \\ \nabla \times U = 0, \\ U - \nabla u = \mathbf{0}, \end{cases}$$

where $\mathcal{F}(\tilde{U}) = f + det(\tilde{A}) = f + \partial_x(\tilde{U}_1)\partial_y(\tilde{U}_2) - \frac{1}{4}(\partial_y(\tilde{U}_1) + \partial_x(\tilde{U}_2))^2$. With sufficient smoothness, $\mathcal{F}(\tilde{U}) \in L^2(\Omega)$, but for problems with reduced regularity this inclusion isn't guaranteed. It is important to note here that this system does not require the computation of any second derivative values numerically, and that the effective diffusion matrix, which involves first derivatives of computed solutions, is on the outside of the derivative operator. This allows the method to avoid inheriting the difficulties associated with nonsmooth components that other discretizations may have in constructing the Hessian from $u^h$. We thus define the least squares functional

$$G(u, U; \mathcal{F}(\tilde{U})) = \|\tilde{A} : \nabla U - \mathcal{F}(\tilde{U})\|^2 + \|\nabla \times U\|^2 + \|U - \nabla u\|^2,$$

and the sets

$$\begin{aligned} \mathcal{V} &= \{v \in H^1(\Omega) \mid v = g \text{ on } \partial\Omega\}, \\ \mathcal{W} &= \{W \in L^2(\Omega)^2 \mid \tilde{A} : \nabla W \in L^2(\Omega), \nabla \times W \in L^2(\Omega), \hat{\tau} \cdot W = \hat{\tau} \cdot \nabla g \text{ on } \partial\Omega\}, \end{aligned}$$

where $\hat{\tau}$ is the counterclockwise unit tangent to the boundary of $\Omega$. In the continuous framework, when $\tilde{U}$ is sufficiently smooth, the solution space for each iterate is a subset of $H^1(\Omega) \times H(div) \cap H(curl)$ and $\tilde{A} \in L^2(\Omega)$. In practice, we take $\mathcal{V}^h$ and $\mathcal{W}^h$ from P2 or P3 in each component, equipped with the appropriate Dirichlet boundary conditions. The solution at each Newton step is thus the minimizer of $G$:

$$(u^h, U^h) = \underset{(v^h, V^h) \in \mathcal{V}^h \times \mathcal{W}^h}{argmin} G(v^h, V^h; \mathcal{F}(\tilde{U}^h)),$$

which is equivalent to the solution of the symmetric variational problem: Find $(u^h, U^h) \in \mathcal{V}^h \times \mathcal{W}^h$ such that

$$\mathcal{B}(u^h, U^h; v^h, V^h) = \ell(v^h, V^h) \quad \forall\ (v^h, V^h) \in \mathcal{V}^h \times \mathcal{W}^h, \tag{2.3}$$

where bilinear form $\mathcal{B}$ and functional $\ell$ are given by

$$\mathcal{B}(u, U; v, V) = \langle \tilde{A} : \nabla U, \tilde{A} : \nabla V \rangle + \langle \nabla \times U, \nabla \times V \rangle + \langle U - \nabla u, V - \nabla v \rangle,$$
$$\ell(v, V) = \langle \mathcal{F}(\tilde{U}), \tilde{A} : \nabla V \rangle.$$

While each step in the general nonlinear iteration involves the solution of a div-curl system, we note that linearizing about a zero initial guess is problematic since it would yield $\tilde{A} = \mathbf{0}$. We thus begin the method with an initial step from algorithm 1. This directly yields $\tilde{u}^h$, and we can then construct $\tilde{U}^h$ by computing $\nabla \tilde{u}^h$ and projecting onto $\mathcal{W}^h$. In the continuous setting, $\mathcal{F}(\tilde{U})$ here retains the same smoothness as $F(\tilde{u})$ in problem 2.1, including nonsmooth cases where the data may fail to be in $L^2(\Omega)$ globally. As in problem 2.1, in the discrete setting here, $\mathcal{F}(\tilde{U}^h)$ is computed elementwise in variational problem 2.3.

Algorithm 2 illustrates the overall Newton/LS iterative method.

**Algorithm 2** Newton/LS Framework

(0) Initialize $\tilde{u}^h = 0$.
(1) Compute $u^h$ with one step of Picard/Galerkin method (algorithm 1).
(2) Set $\tilde{u}^h \leftarrow u^h$, $\tilde{U}^h \leftarrow \nabla \tilde{u}^h|_{\mathcal{W}^h}$
(3) Solve problem (2.3) for $(u^h, U^h) \in \mathcal{V}^h \times \mathcal{W}^h$
(4) Set $\tilde{u}^h \leftarrow u^h$, $\tilde{U}^h \leftarrow \tilde{U}^h$
(5) Test for convergence, repeat from (3) or stop.

As noted above, convergence of Newton's method to the unique viscosity solution has been established in [22, 17] under reasonable assumptions on the original problem as long as the initial guess is sufficiently close to the exact solution. Similarly, for example, in [23], convergence of Newton's method is established in the context of a reformulated div-curl system, assuming sufficient regularity and good initial guesses. While this is for a different PDE, the basic structure is similar to the problem studied here. As for the convergence of each linearized problem, we note that problem (2.3) follows the structure of the div-curl least squares system studied in detail in [10]. The main notable difference is in the smoothness assumption required in establishing the equivalence of the homogenous least squares functional to the product $H^1$ norm of the error. Under the general framework in [10], the coefficient matrix, $A$, in the operator $\nabla \cdot (A \nabla u)$ is required to be $C^{1,1}$. In the discrete setting, since we construct $\tilde{A}^h$ from derivatives of $\tilde{U}^h$, which are in $H^1$-conforming spaces, we obviously have only piecewise smooth coefficients, giving $\tilde{A}^h \in L^2(\Omega)^{2\times 2}$. While this seems like a significant difference, we note that because we explicitly invoke $\nabla \cdot \tilde{A} = \mathbf{0}$, the terms involving derivatives of $\tilde{A}$ vanish, effectively rendering the high smoothness requirements on the coefficients unnecessary. The numerical results presented in the next section show that the Newton/LS framework retains fast convergence in the nonlinear iteration and optimal finite element convergence in the discretization as long as $u \in H^2(\Omega)$ and $f \in L^2(\Omega)$.

**3. Computational Results.** In this section we provide computational examples for test problems that illustrate the robust nature of algorithm 2. In particular, in the first example we provide an explicit comparison of algorithms 1 and 2, focusing on convergence of the nonlinear iteration as well as discretization convergence for quadratic (P2) and cubic (P3) elements. In the second example we focus on a problem with a nearly singular solution, which is studied in [15, 3]. The third example is also

used in several previous studies and is a case where the solution has an unbounded second derivative.

Together, these examples demonstrate the tradeoff between the relative advantages and disadvantages of algorithms 1 and 2. For smooth problems like the first example below, the Newton/LS method has robust convergence in the nonlinear iteration and optimal finite element discretization rates, while the Picard/Galerkin method requires more nonlinear iterations and has suboptimal discretization rates. As with many least-squares finite element methods based on first-order systems, the Newton/LS method is necessarily more sensitive to a loss of regularity. The second and third examples here demonstrate how far the robustness of the Newton/LS approach extends as smoothness is lost.

In each case, unless otherwise noted, we choose $\Omega^h$ as a quasi-uniform triangulation of $\Omega$, with meshsize parameter $h = 1/n$.

**Test Problem 1:** Let $\Omega = (-1, 1)^2$ and $f = (1 + x^2 + y^2)exp(x^2 + y^2)$, which yields the smooth and convex exact solution

$$u = exp\left(\frac{1}{2}(x^2 + y^2)\right).$$

Here, we use meshes with resolutions of $n = 32, 64, 128$, and 256 elements per side on $\Omega^h$. We consider the Picard/Galerkin approach from algorithm 1 and the Newton/LS approach from algorithm 2, where all unknowns are approximated with either P2 or P3 elements in each case. Figure 3.1 shows a contour plot of the solution components $u^h$, $U_1^h$, and $U_2^h$ at the $n = 32$ resolution.

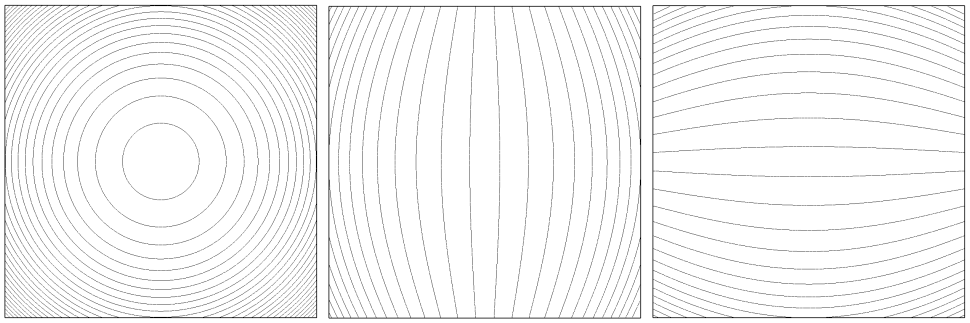

FIG. 3.1. *Solution plots for Test Problem 1: $u^h$ (left), $U_1^h$ (middle), and $U_2^h$ (right) at the $n = 32$ resolution.*

Since this is a test problem studied by other authors, we report both the $L^2$ error and the maximum norm error of the computed solutions for the Newton/LS approach. Results are shown in figures 3.2 and 3.3.

The observed convergence rate is computed by the approximations on the two finest levels. Optimal discretization convergence rates for this smooth function are those predicted by standard finite element interpolation estimates, which are $\mathcal{O}(h^3)$ for P2 and $\mathcal{O}(h^4)$ for P3. The Picard/Galerkin approach is limited to $\mathcal{O}(h^2)$ in both cases, as anticipated from the formulation discussed in section 2, which is consistent with the numerics presented in [3]. By directly approximating $U = \nabla u$, the Newton/LS approach enforces a higher level of smoothness in the iterates and is able to match the optimal convergence rate in both cases.

Picard/Galerkin

| 25 steps | $\|u-u^h\|$ | | $\|u-u^h\|_\infty$ | |
|---|---|---|---|---|
| $n$ | $P2$ | $P3$ | $P2$ | $P3$ |
| 32 | 7.11 e-04 | 9.46 e-05 | 5.96 e-04 | 7.76 e-05 |
| 64 | 1.76 e-04 | 2.36 e-05 | 1.47 e-04 | 1.95 e-05 |
| 128 | 4.18 e-05 | 5.85 e-06 | 3.54 e-05 | 4.85 e-06 |
| 256 | 1.05 e-05 | 1.48 e-06 | 8.88 e-06 | 1.23 e-06 |
| rate $\sim$ | 1.99 | 1.98 | 2.00 | 1.98 |
| optimal | 3 | 4 | 3 | 4 |

FIG. 3.2. *Convergence of the Picard/Galerkin method in the $L^2$ and $L^\infty$ norms using P2 and P3 elements for Test Problem 1.*

Newton/LS

| 8 steps | $\|u-u^h\|$ | | $\|u-u^h\|_\infty$ | |
|---|---|---|---|---|
| $n$ | $P2$ | $P3$ | $P2$ | $P3$ |
| 32 | 1.28 e-05 | 1.95 e-07 | 2.35 e-05 | 5.72 e-07 |
| 64 | 1.60 e-06 | 1.24 e-08 | 3.80 e-06 | 4.65 e-08 |
| 128 | 1.90 e-07 | 7.35 e-10 | 7.08 e-07 | 3.24 e-09 |
| 256 | 2.44 e-08 | 4.59 e-11 | 8.99 e-08 | 2.04 e-10 |
| rate $\sim$ | 2.96 | 4.00 | 2.98 | 3.99 |
| optimal | 3 | 4 | 3 | 4 |

FIG. 3.3. *Convergence of the Newton/LS method in the $L^2$ and $L^\infty$ norms using P2 and P3 elements for Test Problem 1.*

Additionally, the use of Newton's method as the outer iteration provides a much faster convergence with respect to the number of linearization steps required than the Picard linearization in the first method. For this problem we took 25 steps for the Picard/Galerkin method and 8 steps for the Newton/LS method. And while each step of the Newton/LS method is more expensive than a single step of the Picard/Galerkin method (with three times the overall number of degrees of freedom in each step), accuracy per computational cost still easily favors Newton/LS. Figure 3.4 shows a convergence comparison in the $L^2$ norm between the Picard/Galerkin and Newton/LS approaches. The advantage of higher order convergence becomes striking, where the error for Newton/LS is smaller for $n = 32$ than for the Picard/Galerkin method on a mesh with $n = 256$.

Results of the proposed method for this problem compare favorably with results in the literature. For this same test problem, in [15], the $L^2$ errors are reported at the $\approx 10^{-3}$ level at a resolution of $n = 64$ and an approximate convergence rate of 2. In [7], the $L^2$ errors are also reduced at $\mathcal{O}(h^2)$ with a smallest value on the order of $10^{-4}$ at a resolution of approximately $n = 128$. In [24], results for two wide stencil schemes show maximum norm errors at the $10^{-4}$ level at a resolution of $n = 128$, and the convergence rates seem to be about order 1. Both methods examined in [3] have $\mathcal{O}(h^2)$ reduction in the maximum norm with a smallest value on the order of $10^{-5}$ at a resolution of about $n = 220$. And in [17], the reported results have a maximum norm error of $5 \times 10^{-7}$ at a resolution of $n = 361$ and a convergence rate of about order 2.

For a slightly different smooth test problem, Brenner et al. in [6] achieve higher order convergence at essentially optimal discretization rates using finite element spaces

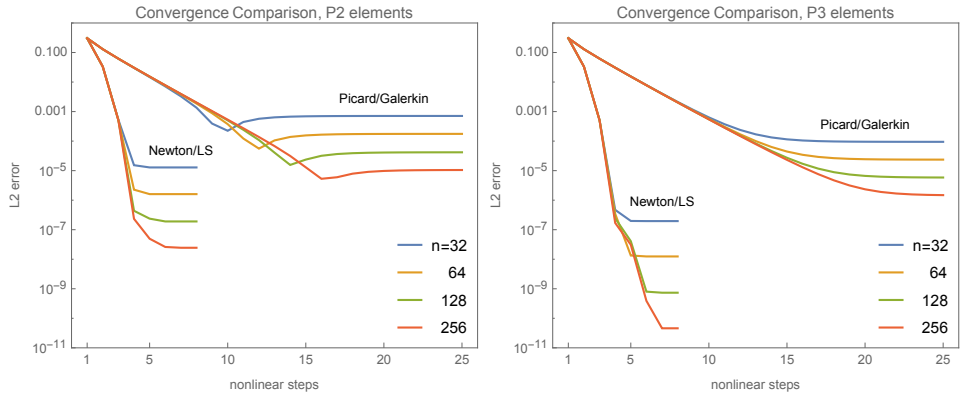


FIG. 3.4. *Convergence in the $L^2$ norm using P2 elements (left) and P3 elements (right) for Test Problem 1.*

of up to degree 4. The numerical performance of their method is comparable to the results of the proposed method here.

**Test Problem 2:** In this example we consider a problem with a nearly singular solution. Let $\Omega = (0,1)^2$ and $f = R^2/(R^2 - x^2 - y^2)^2$, which corresponds to the exact solution

$$u = -\sqrt{R^2 - x^2 - y^2},$$

for $R \geq \sqrt{2}$. The solution here is simply a section of a spherical surface of radius $R$, where $u \in H^2(\Omega)$ and $f \in L^2(\Omega)$ when $R > \sqrt{2}$.

We first consider the Picard/Galerkin approach using quasiuniform meshes with resolutions of $n = 16, 32, 64$, and 128 elements per side, where in each case we take 50 iterations. Figure 3.5 summarizes the $L^2$ error for P2 elements as $R$ approaches the singular limit. Similar to the first example, the $L^2$ error converges at approximately $\mathcal{O}(h^2)$. As in the previous example, using P3 elements gives only slightly smaller errors with essentially the same convergence rate. For a fixed resolution, errors grow as $R \to \sqrt{2}^+$.

| $\|u - u^h\|$ | Picard/Galerkin | | | |
|---|---|---|---|---|
| $n$ | $R = 2$ | $\sqrt{2} + 0.1$ | $\sqrt{2} + 0.01$ | $\sqrt{2} + 0.001$ |
| 16 | 1.31 e-05 | 8.57 e-05 | 4.56 e-04 | 2.07 e-03 |
| 32 | 3.28 e-06 | 2.04 e-05 | 6.91 e-05 | 4.82 e-04 |
| 64 | 8.55 e-07 | 5.93 e-06 | 1.67 e-05 | 1.07 e-04 |
| 128 | 2.01 e-07 | 1.33 e-06 | 3.43 e-06 | 1.97 e-05 |
| rate $\sim$ | 2.09 | 2.16 | 2.28 | 2.44 |

FIG. 3.5. *$L^2$ convergence for the Picard/Galerkin approach for Test Problem 2, taking 50 iterations and using P2 elements for $R \to \sqrt{2}^+$.*

We now consider the Newton/LS approach for this problem. We first example the relatively smooth cases $R = 2$ and $R = \sqrt{2} + 0.1$. Figure 3.6 shows the $L^2$ error at convergence after taking 10 Newton steps for each value of $R$ and using either $P2$ or $P3$ elements for all unknowns. As in the previous example, discretization convergence

rates are essentially optimal. We report up to a resolution of $n = 128$ here, noting that at higher resolutions with $P3$ elements we begin to see slight machine precision effects. Results for the same values of $R$ can be compared, for example, with those in [7], which show $L^2$ convergence at $\mathcal{O}(h^2)$.

| Newton/LS $\|u - u^h\|$ | $R = 2$ | | $R = \sqrt{2} + 0.1$ | |
|---|---|---|---|---|
| $n$ | $P2$ | $P3$ | $P2$ | $P3$ |
| 16 | 7.47 e-07 | 1.23 e-08 | 6.16 e-06 | 3.04 e-07 |
| 32 | 8.88 e-08 | 7.30 e-10 | 7.40 e-07 | 1.94 e-08 |
| 64 | 1.27 e-08 | 5.65 e-11 | 1.45 e-07 | 2.61 e-09 |
| 128 | 1.47 e-09 | 3.19 e-12 | 1.76 e-08 | 1.72 e-10 |
| rate $\sim$ | 3.11 | 4.15 | 3.04 | 3.92 |
| optimal | 3 | 4 | 3 | 4 |

FIG. 3.6. *L2 error at convergence using P2/P3 elements for $R = 2$ and $R = \sqrt{2} + 0.1$ for Test Problem 2.*

In addition, since the exact solution is not always known, we introduce the residual norm measure as

$$\mathcal{R}(U^h) = \|\partial_x(U_1^h)\partial_y(U_2^h) - \partial_y(U_1^h)\partial_x(U_2^h) - f\|,$$

where the norm here is computed as the sum of element-wise values. Figure 3.7 shows convergence results in this measure, which yield observed convergence rates approximately one order less than the $L^2$ measures in each case (similar to the $H^1$ norm of the error, e.g.).

| Newton/LS $\mathcal{R}(U^h)$ | $R = 2$ | | $R = \sqrt{2} + 0.1$ | |
|---|---|---|---|---|
| $n$ | $P2$ | $P3$ | $P2$ | $P3$ |
| 16 | 1.68 e-04 | 1.14 e-06 | 1.67 e-02 | 1.08 e-03 |
| 32 | 4.09 e-05 | 1.47 e-07 | 4.03 e-03 | 9.52 e-05 |
| 64 | 1.17 e-05 | 2.58 e-08 | 1.41 e-03 | 1.73 e-05 |
| 128 | 2.76 e-06 | 2.89 e-09 | 3.64 e-04 | 2.16 e-06 |
| rate $\sim$ | 2.08 | 3.16 | 1.95 | 3.00 |

FIG. 3.7. *Residual measure at convergence using P2/P3 elements for $R = 2$ and $R = \sqrt{2} + 0.1$ for Test Problem 2.*

For the Newton/LS formulation, the singular limit of this problem represents a challenge since $\nabla u$ becomes unbounded as $R \to \sqrt{2}^+$. For $R = \sqrt{2}$, the problem loses smoothness with $u \notin H^2(\Omega)$, $U = \nabla u \notin H^1(\Omega)$ and $f \notin L^2(\Omega)$. Since the proposed method here directly approximates $U$, it will not converge when $R = \sqrt{2}$. To demonstrate performance of the proposed method toward the singular limit, we introduce an adaptive mesh refinement routine based on the locally defined least-squares functional. Let $\mathcal{T}$ represent the set of elements in $\Omega^h$ and define

$$G_\tau = \|\tilde{A}^h : \nabla U^h - \mathcal{F}(\tilde{U}^h)\|_\tau^2 + \|\nabla \times U^h\|_\tau^2 + \|U^h - \nabla u^h\|_\tau^2,$$

as the least squares functional evaluated on element $\tau$. The global functional norm is thus denoted as $G^{1/2} = \left(\sum_{\tau \in \mathcal{T}} G_\tau\right)^{1/2}$. We use the basic framework in algorithm 2,

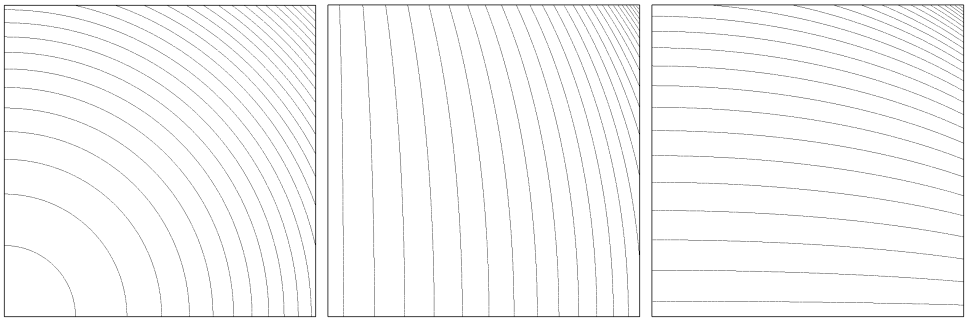

FIG. 3.8. *Solution plots for Test Problem 2: $u^h$ (left), $U_1^h$ (middle), and $U_2^h$ (right) for $R = \sqrt{2} + 0.1$ at the $n = 32$ resolution.*

starting from a quasiuniform mesh using $P2$ elements for each unknown. After 5 Newton steps on the initial mesh, the mesh is refined between each of the next four Newton steps. Refinement is done by marking the elements on which $G_\tau$ is largest, targeting at least 5% of elements to each be divided into 4 elements. Elements adjacent to those marked are bisected to avoid hanging nodes. All current approximations on the coarse mesh are interpolated to the refined mesh to define the next linearized problem, (2.3), to be solved. For more details on this nested iteration approach for least-squares methods, see [12, 23].

Figure 3.9 illustrates the results of the adaptive mesh routine, showing $G_\tau$ on $\Omega^h$ and the resulting refined mesh at four refinement levels for the test problem with $R = \sqrt{2} + 0.01$. As expected, the refinement targets the corner where $\nabla u$ is large.

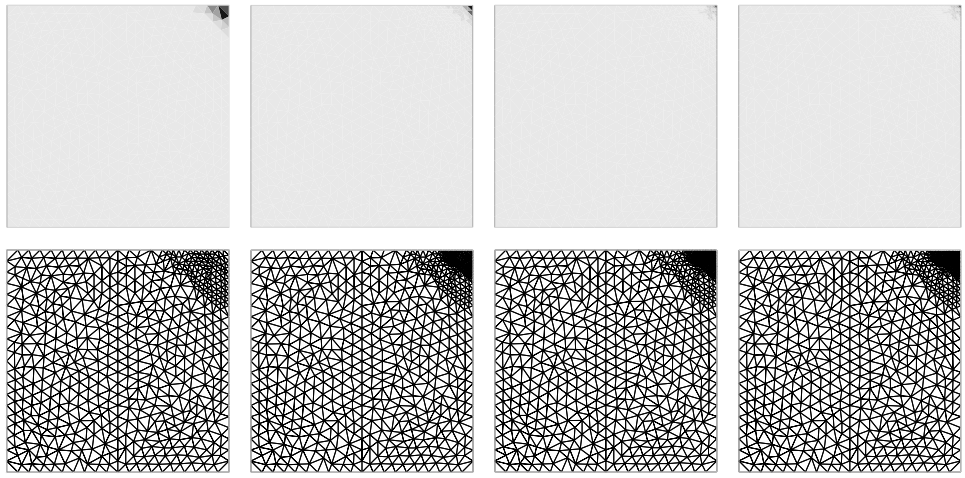

FIG. 3.9. *Test Problem 2 with $R = \sqrt{2} + 0.01$ solved using $P2$ elements and an adaptive mesh routine. Top row: locally evaluated least squares functional values used for four levels of refinement (larger values are colored darker). Bottom row: the mesh after each refinement step.*

Figure 3.10 shows numerical convergence results for $R = \sqrt{2} + \{0.1, 0.01, 0.001\}$. The first row shows values of the $L^2$ error in $u$ and the least squares functional norm after the first 5 Newton steps on the initial quasiuniform mesh, and the subsequent rows show the next 4 Newton steps on adaptively refined meshes. $N_\tau$ gives the

number of elements in the mesh used. As $R$ approaches the singular limit of $\sqrt{2}$ the method continues to converge, but with significantly larger functional values, which is consistent with the increase in the $H^1$ norm of $U$. At the finest resolution here the smallest elements have size 1/320.

Newton/LS

| $R = \sqrt{2} + 0.1$ | | | $R = \sqrt{2} + 0.01$ | | | $R = \sqrt{2} + 0.001$ | | |
|---|---|---|---|---|---|---|---|---|
| $N_\tau$ | $\|u - u^h\|$ | $G^{1/2}$ | $N_\tau$ | $\|u - u^h\|$ | $G^{1/2}$ | $N_\tau$ | $\|u - u^h\|$ | $G^{1/2}$ |
| 944 | 3.13e-06 | 1.08e-02 | 944 | 1.94e-04 | 1.90e-00 | 944 | 1.02e-02 | 3.01e+01 |
| 1088 | 1.64e-06 | 3.14e-03 | 1089 | 8.98e-05 | 1.40e-00 | 1106 | 6.99e-03 | 3.14e+01 |
| 1252 | 1.35e-06 | 1.69e-03 | 1254 | 1.79e-05 | 4.84e-01 | 1276 | 4.27e-03 | 2.69e+01 |
| 1443 | 1.13e-06 | 1.35e-03 | 1446 | 4.05e-06 | 1.32e-01 | 1468 | 2.52e-03 | 1.33e+01 |
| 1663 | 1.02e-06 | 1.15e-03 | 1663 | 3.39e-06 | 3.54e-02 | 1689 | 8.98e-04 | 5.25e-00 |

FIG. 3.10. *Convergence of the $L^2$ and least squares functional in the adaptive refinement routine in the singular limit of Test Problem 2.*

**Test Problem 3:** As a final example we consider the case with $f = 1$ on $\Omega = (-1, 1)^2$ and $u = 1$ on $\partial\Omega$. On this domain it can be seen that having $u$ constant along the polygonal boundary is inconsistent with the solution satisfying $u_{xx}u_{yy} - u_{xy}^2 = 1$ since along each boundary segment either $u_{xx}$ or $u_{yy}$ is zero. This inconsistency induces a nonsmooth component to $u$ along $\partial\Omega$. This illustrates a problem where, even though $f$ is smooth and positive, a loss of regularity occurs since the domain is not strictly convex and smooth. Since we do not have a known classical solution for this problem, we instead monitor convergence in the residual norm as in the previous example as well as the value of $u^h$ in the center of the domain.

In implementing the Newton/LS approach here we follow the same structure as before, except that, because of the inconsistency of the problem at the boundaries, we enforce Dirichlet conditions on $u^h$ only and do not impose any boundary conditions on the flux variable $U^h$. This allows the method to have, for example, $\partial_x(U_1)$ near zero and $\partial_y(U_2)$ large along the bottom boundary at $y = 0$. Enforcing $\partial_x(U_1) = 0$ strongly there would force $\partial_y(U_2)$ to be unbounded. We also note that since convergence rates should be limited by the smoothness of the solution we only report results using P2 elements for all unknowns. A contour plot of the solution components at a resolution of $n = 32$ is shown in figure 3.11.

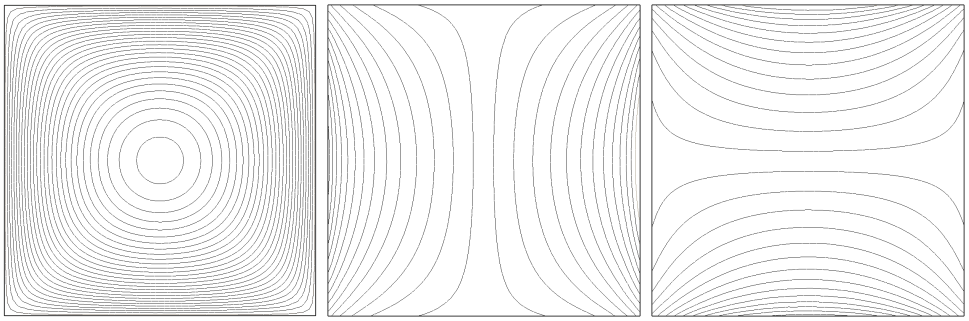

FIG. 3.11. *Solution plots for Test Problem 3: $u^h$ (left), $U_1^h$ (middle), and $U_2^h$ (right) at the $n = 32$ resolution.*

Numerical results for test problem 3 are given in figure 3.12, using quasiuniform meshes with resolutions of $n = 32, 64, 128, 256$, and 512 elements per side. In addition to numerical results for the Newton/LS method, we give values for the Picard/Galerkin approach described in algorithm 1. For each resolution we take 100 iterations for the Picard/Galerkin approach and 20 iterations for the Newton/LS method. For the Newton/LS method the residual norm converges at a rate of approximately 0.44, which is consistent with the reduced smoothness of this problem, likely indicating a solution $u \in H^1(\Omega) \backslash H^2(\Omega)$. (We note that a comparable discrete residual measure for the Picard/Galerkin approach isn't feasible since it would require computation of second derivatives of finite element functions.)

Since no exact solution exists for this problem, to provide a comparison to other published methods we also report the value of the computed solution at the center of the domain, $u^h(0,0)$ for both approaches (see figure 3.12). In [24], Oberman develops finite difference methods with 9 and 17-point stencils are developed and applied to the same test problem. Under the highest resolution reported, each stencil yields an approximation of 0.3131 for the minimum of $u$. Similarly, in [3], Benamou, Froese, and Oberman develop two finite difference approaches which are applied to the same test problem. At the highest resolution reported, their methods yield a value of 0.2621. For comparison, they also implement the wide stencil methods of [24] and report values in the range 0.2695 to 0.3074 at the highest resolution. Our approximations seem to be converging to values similar to previously published results, with the Picard/Galerkin method converging from below and the Newton/LS method converging from above.

Additionally, for this problem, the number of required iterations of the finite difference implementations in [3] are reported to grow proportional to $n^2$, where, for example, at $n = 141$ the iteration count is on the order of $10^5$ for each method studied. While the Newton/LS method for this problem certainly has slower convergence than for the smooth test problems, it seems to still be relatively robust in the number of required nonlinear iterations.

| | Newton/LS | | P/G |
|---|---|---|---|
| $n$ | $\mathcal{R}(U^h)$ | $u^h(0,0)$ | $u^h(0,0)$ |
| 32 | 6.98 e-02 | 0.3327 | 0.2510 |
| 64 | 5.29 e-02 | 0.3174 | 0.2557 |
| 128 | 3.83 e-02 | 0.3041 | 0.2580 |
| 256 | 2.75 e-02 | 0.2932 | 0.2588 |
| 512 | 2.03 e-02 | 0.2848 | 0.2592 |
| rate $\sim$ | 0.44 | – | – |

FIG. 3.12. *Residual norm and $u^h(0,0)$ at convergence for Test Problem 3, using 20 iterations for Newton/LS and 100 iterations for the Picard/Galerkin method.*

As a final illustration, we include plots of the locally evaluated least squares functional in figure 3.13 for the $n = 32$, 64, and 128 resolutions. As in the previous test problem this indicates that the error is concentrated in the corners where $U = \nabla u$ is large.

**4. Extensions and Concluding Remarks.** In this paper we have proposed a new finite element method, based on least-squares minimization principles and Newton's method, for classical smooth solutions of the elliptic Monge-Ampère equation. In essence, by directly controlling the flux variable $U = \nabla u$, the method here is able to capitalize on the power of higher order finite element spaces. As a result, for smooth

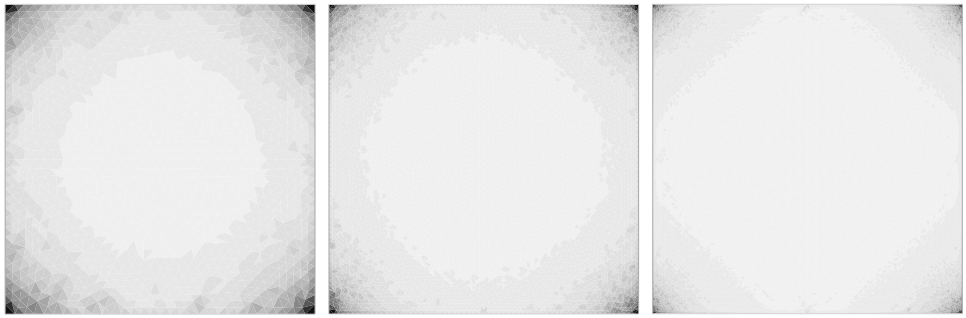

FIG. 3.13. *Local least squares functional values at convergence for Test Problem 3 for the* $n = 32$*,* $64$*, and* $128$ *resolutions.*

problems, our approach is able to achieve optimal convergence rates using standard conforming Lagrange finite element spaces.

While not explored in this paper, the approach here inherits a range of attractive features inherent in the least-squares finite element framework. For example, the linear systems produced at each step are symmetric and positive definite, and are generally solved efficiently by multilevel iterative methods. And while the focus here is primarily on smooth solutions, directly approximating $U = \nabla u$ makes this approach necessarily more sensitive to reduced regularity than many other approaches. In the least-squares context this has been successfully addressed by weighted-norm methods (see [21, 11, 4], for example).

## REFERENCES


[1] G. Awanou. On standard finite difference discretizations of the elliptic Monge–Ampère equation. *Journal of Scientific Computing*, 69(2):892–904, 2016.

[2] R. Beltman, J.H.M. Boonkkamp, and W. Ijzerman. A least-squares method for the inverse reflector problem in arbitrary orthogonal coordinates. *Journal of Computational Physics*, 367:347 – 373, 2018.

[3] J.-D. Benamou, B. Froese, and A. Oberman. Two numerical methods for the elliptic Monge-Ampère equation. *M2AN Math. Model. Numer. Anal.*, 44:737–758, 2010.

[4] S. Bidwell, M. Hassell, and C.R. Westphal. A weighted least squares finite element method for elliptic problems with degenerate and singular coefficients. *Math. Comp.*, 82:672–688, 2013.

[5] D. Braess. *Finite Elements: Theory, Fast Solvers and Applications in Solid Mechanics*. Cambridge, 2001.

[6] S. C. Brenner, T. Gudi, M. Neilan, and L.-y. Sung. $C^0$ penalty methods for the fully nonlinear Monge-Ampère equation. *Math. Comp.*, 80:1979–1995, 2011.

[7] A. Caboussat, R. Glowinski, and D.C. Sorensen. A least-squares method for the numerical solution of the Dirichlet problem for the elliptic Monge-Ampère equation in dimension two. *ESAIM: COCV*, 19:780–810, 07 2013.

[8] L. A. Caffarelli. Some regularity properties of solutions of Monge-Ampère equation. *Communications on Pure and Applied Mathematics*, 44(8?9):965–969, 1991.

[9] Z. Cai, R. Lazarov, T.A. Manteuffel, and S.F. McCormick. First-order system least squares for second-order partial differential equations: part i. *SIAM J. Numer. Anal.*, 31(6):1785–1799, 1994.

[10] Z. Cai, T.A. Manteuffel, and S.F. McCormick. First-order system least squares for second-order partial differential equations: part ii. *SIAM J. Numer. Anal.*, 34(2):425–454, 1997.

[11] Z. Cai and C.R. Westphal. A weighted H(div) least-squares method for second-order elliptic problems. *SIAM J. Numer. Anal.*, 46(3):1640–1651, 2008.

[12] A.L. Codd, T.A. Manteuffel, and S.F. McCormick. Multilevel first-order system least squares for nonlinear elliptic partial differential equations. *SIAM J. Numer. Anal.*, 41(6):2197–2209,

2003.
[13] G. De Philippis and A. Figalli. Sobolev regularity for Monge-Ampère type equations. *SIAM Journal on Mathematical Analysis*, 45(3):1812–1824, 2013.
[14] E. J. Dean and R. Glowinski. Numerical solution of the two-dimensional elliptic Monge-Ampère equation with Dirichlet boundary conditions: An augmented Lagrangian approach. *C. R. Math. Acad. Sci. Paris*, 336:779?784, 2003.
[15] E.J. Dean and R. Glowinski. Numerical methods for fully nonlinear elliptic equations of the Monge-Ampère type. *Comput. Methods Appl. Mech. Engrg.*, 195:1344–1386, 2006.
[16] X. Feng, R. Glowinski, and M. Neilan. Recent developments in numerical methods for fully nonlinear second order partial differential equations. *SIAM Review*, 55(2):205–267, 2013.
[17] B. Froese and A. Oberman. Convergent finite difference solvers for viscosity solutions of the elliptic Monge-Ampère equation in dimensions two and higher. *SIAM Journal on Numerical Analysis*, 49(4):1692–1714, 2011.
[18] M.D. Gunzburger and P.B. Bochev. *Least-Squares Finite Element Methods*. Springer, 2009.
[19] F. Hecht. New development in FreeFem++. *J. Numer. Math.*, 20(3-4):251–265, 2012.
[20] O. Lakkis. and T. Pryer. A finite element method for nonlinear elliptic problems. *SIAM Journal on Scientific Computing*, 35(4):A2025–A2045, 2013.
[21] E. Lee, T.A. Manteuffel, and C.R. Westphal. Weighted-norm first-order system least squares (FOSLS) for problems with corner singularities. *SIAM J. Numer. Anal.*, 44(5):1974–1996, 2006.
[22] G. Loeper and F. Rapetti. Numerical solution of the Monge-Ampère equation by a Newton's algorithm. *C.R. Acad. Sci. Paris*, 340:319–324, 2005.
[23] T.A. Manteuffel, S.F. McCormick, J.G. Schmidt, and C.R. Westphal. First-order system least squares (FOSLS) for geometrically nonlinear elasticity. *SIAM J. Numer. Anal.*, 44(5):2057–2081, 2006.
[24] A. Oberman. Wide stencil finite difference schemes for the elliptic Monge-Ampère equation and functions of the eigenvalues of the Hessian. *Disc. and Contin. Dyn. Syst., Series B*, 10(1):221–238, 2008.
[25] G. De Philippis and A. Figalli. The Monge-Ampère equation and its link to optimal transportation. *Bull. Amer. Math. Soc.*, 51(4):527–580, 2014.
[26] C.R. Prins, R. Beltman, J.H.M. Boonkkamp, W. Ijzerman, and T. W. Tukker. A least-squares method for optimal transport using the Monge-Ampère equation. 37:B937–B961, 01 2015.
[27] X.-J. Wang. Some counterexamples to the regularity of Monge-Ampère equations. *Proceedings of the American Mathematical Society*, 123(3):841–845, 1995.