

Multigrid Methods for Image Registration Model Based on Optimal Mass Transport



Yangang Chen and Justin W. L. Wan

Abstract In this survey, we present fast, accurate and convergent numerical methods for solving non-rigid image registration based on optimal mass transport. To solve the model equation, we first transform the nonlinear PDEs into an HJB equation. We apply a mixed standard 7-point stencil and semi-Lagrangian wide stencil discretization, such that the numerical solution is guaranteed to converge to the viscosity solution of the Monge-Ampere equation. We design a numerical scheme that converges to the optimal transformation between the target and template images. Finally, we introduce fast multigrid methods for solving the discrete nonlinear system. In particular, we use a four-directional alternating line relaxation scheme as smoother, a coarsening strategy where wide stencil points are set as coarse grid points. Linear interpolation and injection are used in prolongation and restriction, respectively. Our numerical results show that the numerical solution yield good quality transformations for non-rigid image registration and the convergence rates of the proposed multigrid methods are mesh-independent.

Keywords Image registration · Optimal mass transport · Monge-Ampere equation · Multigrid · Monotone discretization scheme

1 Introduction

In many applications, one has to compare two images T (template) and R (reference) which display the same object, but the object inside the images is not spatially aligned,

Y. Chen

Department of Applied Mathematics, University of Waterloo, 200 University Avenue West, Waterloo, ON, Canada

e-mail: y493chen@uwaterloo.ca

J. W. L. Wan (✉)

David R. Cheriton School of Computer Science, University of Waterloo, 200 University Avenue West, Waterloo, ON, Canada

e-mail: justin.wan@uwaterloo.ca

© Springer Nature Singapore Pte Ltd. 2021

X.-C. Tai et al. (eds.), *Mathematical Methods in Image Processing and Inverse Problems*,

Springer Proceedings in Mathematics & Statistics 360,

https://doi.org/10.1007/978-981-16-2701-9_11

or the devices that record the two images are different. The image registration problem is to find a coordinate transformation ϕ which transforms the image T to another image T_ϕ , such that T_ϕ is similar and thus comparable to the image R .

One important application of image registration is to compare medical images of the same patient, such as CT (computed tomography) and MRI (magnetic resonance imaging) images of a damaged brain, which gives guidance for diagnosis and surgery [1, 24]. Image registration can also be used for image fusion [26]. Multiple images of the same object are taken, registered and then merged together, such that the integrated image provides more useful than the original ones. We refer readers to [2] for more discussion on applications.

Different approaches have been developed for image registration problems, including parametrized transformation [30, 44], landmark-based registration [36], elastic registration [8], fluid registration [14], diffusion registration [18], demon's registration [41], flow of diffeomorphism [16, 43], etc. A substantial discussion of existing methods can be found in [32, 40].

This paper surveys the three recent works of the authors [10–12] and it considers a non-rigid image registration method based on Monge-Kantorovich mass transport [9, 20, 22, 23, 33, 37]. Optimal mass transport problems appear in many applications and have been widely studied (see e.g. [13, 34, 38]). The use of optimal mass transport for image registration was first proposed in [22, 23]. This image registration model treats two images R and T as two mass densities. The goal is to find a mapping which transforms one mass density T to the other R with mass conservation. Such transformation is non-unique. By defining a transformation-dependent cost function and minimizing it, we can obtain a unique optimal transformation. This optimal transformation has desirable properties. For instance, it is usually diffeomorphic and does not introduce foldings and crossings.

The primary advantage of this image registration model is that, unlike many other non-rigid methods that are only applicable for images with small deformations, this model can be applied to images with large deformations. See Figs. 1 and 2 in [22] for an example of images with large deformations. Indeed, given *any* R and T , the transformed image T_ϕ under the mass transport formulation can be equal to R [33].

Numerical methods have been developed for solving the image registration model based on optimal mass transport. In [22, 23], the authors construct an initial mass-preserving mapping ϕ_0 by solving a nonlinear partial differential equation (PDE), and obtain a second mass-preserving mapping ϕ_s by solving another nonlinear PDE system, such that $\phi_0 \circ \phi_s$ is the optimal transformation. The entire process involves many intermediate steps. Also, in general, a nonlinear PDE (or PDE system) has multiple solutions. An immediate challenge is that the nonlinear PDE system in [22, 23] can give multiple transformations between R and T , which may not be the optimal transformation.

An alternative approach is to solve an equivalent nonlinear Monge-Ampère equation. The gradient of the unique globally convex solution corresponds to the optimal transformation between R and T [22, 27]. The convex solution itself is usually called a scalar potential that generates the optimal transformation. Some literature has investigated numerical schemes for the Monge-Ampère equation arising from the image

registration model [9, 20, 37]. However, for the approach in [20], the computational cost per pixel must increase to infinity as the image size increases [17]. The methods in [9, 37] are based on gradient descent, which is essentially equivalent to solving the Monge-Ampère equation using explicit pseudo-timestepping.

In this survey paper, we present a numerical approach for the image registration model based on optimal mass transport by solving the equivalent Monge-Ampère equation. In order to ensure that the numerical scheme yields the optimal transformation between R and T [20, 21], we will adopt a monotone finite difference discretization method based on our previous work [10], which can be proved that the resulting numerical solution converges to the viscosity solution [4] of the Monge-Ampère equation. We will also present efficient multigrid methods for solving the resulting nonlinear discretized system [11].

Standard multigrid methods turn out to have poor convergence. There are two major factors behind the poor convergence. One is that the PDE may become anisotropic along various directions. Standard pointwise smoothers fail to smooth the error along the weakly connected directions. The other factor is that the resulting matrix is non-symmetric, which is a well-known issue when applying multigrid. Algebraic multigrid (AMG) methods [35, 39] have been used as preconditioners. However, they are not efficient as stand-alone solvers since AMG methods assess geometric information indirectly through the strength of connections which is not effective for the monotone discretization.

To obtain a fast stand-alone multigrid solver for solving Monge-Ampère equations, we note that wide stencils introduce oscillations locally to the error, and such oscillations cannot be eliminated by smoothers, including the alternating line smoothers. However, the oscillations are restricted at the wide stencil points. One possible solution to capture the oscillations is to use a sophisticated interpolation, which can be complicated and expensive to set up. Instead, we use a non-standard coarsening strategy. Specifically, we set wide stencil points as coarse grid points. The purpose is to directly use the coarse grid points to capture the oscillations. As the wide stencils are mainly restricted to the singular points or singular lines, setting wide stencil points as coarse grid points does not significantly increase the number of the coarse grid points. In our numerical experiments, we illustrate that the proposed multigrid method has a mesh-independent convergence rate for various problems.

This paper is organized as follows. In Sect. 2, we describe the image registration model based on optimal mass transport. Section 3 describes a finite difference discretization for the Monge-Ampère equation arising from the mass transport image registration model. In Sect. 4, we present efficient multigrid methods to solve the discretized system. Numerical results in Sect. 5 show that our multigrid methods converge quickly with mesh-independent convergence rate. Image results are also provided to demonstrate the performance of the registration model. Section 6 concludes the paper.

2 Image Registration Model Based on Optimal Mass Transport

2.1 Image Registration

Given a template image T and a reference image R , the objective of the image registration problem is to align the two images. Mathematically, we consider the template (reference) image as a function defined on the domain Ω^T (Ω^R). For simplicity, we assume that $\Omega^T = \Omega^R = [0, 1] \times [0, 1]$. The image registration problem can be formulated as to find a coordinate transformation ϕ that minimizes the difference between ρ^{T_ϕ} and ρ^R , where ρ^T and ρ^R are the intensities of the template image and reference image, respectively, and ρ^{T_ϕ} is the intensity of the transformed image, T_ϕ . The intensities must be positive and bounded. The difference of the two images is usually measured by some function such as sum of squared differences

$$\mathcal{D}(\rho^{T_\phi}, \rho^R) \equiv \|\rho^{T_\phi} - \rho^R\|_{L_2(\Omega^R)}. \quad (1)$$

2.2 Optimal Mass Transport Model

Consider registering two images T and R . If we view them as two piles of soil with the densities ρ^T and ρ^R , then an image registration problem can be interpreted as a mass transport problem [22, 23, 33]. That is, we consider two piles of soil ρ^T and ρ^R with the same total mass:

$$\int_{\hat{\mathbf{x}} \in \Omega^T} \rho^T(\hat{\mathbf{x}}) d^2 \hat{\mathbf{x}} = \int_{\mathbf{x} \in \Omega^R} \rho^R(\mathbf{x}) d^2 \mathbf{x}. \quad (2)$$

The image registration problem becomes to find a coordinate transformation $\phi : \Omega^R \rightarrow \Omega^T$, or $\hat{\mathbf{x}} = \phi(\mathbf{x}) \in \mathbb{R}^2$, such that ρ^T is transformed to ρ^R while the total mass is conserved:

$$\int_{\mathbf{x} \in \Omega^R} \rho^T(\phi(\mathbf{x})) d^2 \phi(\mathbf{x}) = \int_{\mathbf{x} \in \Omega^R} \rho^R(\mathbf{x}) d^2 \mathbf{x}, \quad (3)$$

or equivalently,

$$\rho^T(\phi(\mathbf{x})) \det[D\phi(\mathbf{x})] = \rho^R(\mathbf{x}), \quad (4)$$

where $D\phi(\mathbf{x}) \in \mathbb{R}^{2 \times 2}$ is the Jacobian of the transformation $\phi(\mathbf{x})$.

Under the transformation ϕ , define the intensity of the transformed image T_ϕ as

$$\rho^{T_\phi}(\mathbf{x}) \equiv \rho^T(\phi(\mathbf{x})) \det[D\phi(\mathbf{x})]. \quad (5)$$

Then the transformed template image is equal to the reference image:

$$\rho^{T\phi}(\mathbf{x}) = \rho^R(\mathbf{x}). \quad (6)$$

As a result, the mass transport model can transform any template image T to any reference image R [33].

The mass transport registration (5) is ill-posed. More specifically, there exist multiple transformations that move the soil ρ^T to ρ^R . Among all possible transformations, one of them requires the “least cost”, which is desirable. Following [5, 22, 23], we aim to find the optimal transformation $\phi^*(\mathbf{x})$ that minimizes the following cost function:

$$\phi^*(\mathbf{x}) \equiv \arg \min_{\phi(\mathbf{x})} \int_{\mathbb{R}^2} \|\mathbf{x} - \phi(\mathbf{x})\|^2 \rho^R(\mathbf{x}) d^2\mathbf{x}, \quad (7)$$

which is the weighted least squares displacement of the mass. In essence, (7) regularizes the mass transport registration and makes the transformation between ρ^T and ρ^R unique.

2.3 Monge-Ampère Equation

It has been proved in [27] that the optimal transformation that minimizes the cost function (7) can be written as

$$\phi^*(\mathbf{x}) = \nabla u(\mathbf{x}), \quad (8)$$

where $u \in C(\Omega^R)$ is a strictly convex scalar potential field, and its gradient ∇u generates the optimal transformation ϕ^* . Substituting (8) into (4), we have

$$\det[D^2u(\mathbf{x})] = \frac{\rho^R(\mathbf{x})}{\rho^T(\nabla u(\mathbf{x}))}, \quad (9)$$

$$u \text{ is strictly convex.} \quad (10)$$

Equations (9)–(10) is a **Monge-Ampère equation**.

Due to the nonlinearity, the equation (9) itself, without the convexity constraint (10), can have multiple solutions [6, 17]. However, the solution of (9) that satisfies the convexity constraint (10) is unique [20], which we will denote as u^* whenever we need to distinguish it from the other solutions. We emphasize that the convexity of u^* is equivalent to the optimality of the transformation $\phi^* = \nabla u^*$ [20, 22].

3 Finite Difference Discretization

In order to design a finite difference scheme that converges to the viscosity solution, we first convert the Monge-Ampère equation into an equivalent Hamilton-Jacobi-Bellman (HJB) equation. The equivalence of the two PDEs is first established in [28, 29]. Here we present the equivalent HJB equation as follows:

Theorem 1 *Let $u \in C^2(\Omega^R)$ be convex, and let $\rho^T \in C(\Omega^T)$ and $\rho^R \in C(\Omega^R)$ be two positive functions. Then the Monge-Ampère equation (9)–(10) is equivalent to the following HJB equation*

$$\hat{\mathcal{L}}_{c^*(\mathbf{x}), \theta^*(\mathbf{x})} u(\mathbf{x}) = 0, \tag{11}$$

$$\text{subject to } (c^*(\mathbf{x}), \theta^*(\mathbf{x})) \equiv \arg \max_{(c(\mathbf{x}), \theta(\mathbf{x})) \in \Gamma} \hat{\mathcal{L}}_{c(\mathbf{x}), \theta(\mathbf{x})} u(\mathbf{x}), \tag{12}$$

where the differential operator is

$$\begin{aligned} \hat{\mathcal{L}}_{c(\mathbf{x}), \theta(\mathbf{x})} u(\mathbf{x}) \equiv & -\sigma_{11}(c(\mathbf{x}), \theta(\mathbf{x}))u_{xx}(\mathbf{x}) - 2\sigma_{12}(c(\mathbf{x}), \theta(\mathbf{x}))u_{xy}(\mathbf{x}) \\ & -\sigma_{22}(c(\mathbf{x}), \theta(\mathbf{x}))u_{yy}(\mathbf{x}) + 2\sqrt{c(\mathbf{x})(1-c(\mathbf{x}))} \frac{\rho^R(\mathbf{x})}{\rho^T(\nabla u(\mathbf{x}))}, \end{aligned} \tag{13}$$

and $(c(\mathbf{x}), \theta(\mathbf{x}))$ is the pair of control at point \mathbf{x} , $\Gamma = [0, 1] \times [-\frac{\pi}{4}, \frac{\pi}{4}]$ is the set of admissible control. The coefficients are

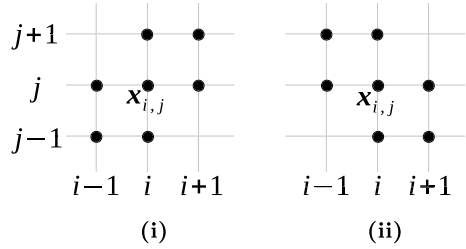
$$\begin{aligned} \sigma_{11}(c(\mathbf{x}), \theta(\mathbf{x})) &= \frac{1}{2}[1 - (1 - 2c(\mathbf{x})) \cos 2\theta(\mathbf{x})], \\ \sigma_{22}(c(\mathbf{x}), \theta(\mathbf{x})) &= \frac{1}{2}[1 + (1 - 2c(\mathbf{x})) \cos 2\theta(\mathbf{x})], \\ \sigma_{12}(c(\mathbf{x}), \theta(\mathbf{x})) &= \frac{1}{2}(1 - 2c(\mathbf{x})) \sin 2\theta(\mathbf{x}). \end{aligned} \tag{14}$$

Below, we will describe a monotone finite difference discretization scheme for the HJB equation (11)–(12).

3.1 Standard 7-Point Stencil Discretization

Consider discretizing the differential operator (13) at a grid point $\mathbf{x}_{i,j}$. We use the standard central differencing to approximate $u_{xx}(\mathbf{x}_{i,j})$ and $u_{yy}(\mathbf{x}_{i,j})$. Regarding the cross derivative $u_{xy}(\mathbf{x}_{i,j})$, it can be shown that the standard 7-point stencil discretization leads to a monotone scheme in the following two cases:

Fig. 1 (i) 7-point stencil of (16); (ii) 7-point stencil of (18)



- **Case 1.** When the coefficients (14) at a grid point $\mathbf{x}_{i,j}$ satisfy

$$\sigma_{11}(c_{i,j}, \theta_{i,j}) \geq |\sigma_{12}(c_{i,j}, \theta_{i,j})|, \sigma_{22}(c_{i,j}, \theta_{i,j}) \geq |\sigma_{12}(c_{i,j}, \theta_{i,j})|, \sigma_{12}(c_{i,j}, \theta_{i,j}) \geq 0, \quad (15)$$

we approximate $u_{xy}(\mathbf{x}_{i,j})$ using

$$\frac{1}{2}(D_x^+ D_y^+ + D_x^- D_y^-)u_{i,j} \equiv \frac{2u_{i,j} + u_{i+1,j+1} + u_{i-1,j-1} - u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1}}{2h^2}. \quad (16)$$

- **Case 2.** When the coefficients (14) at a grid point $\mathbf{x}_{i,j}$ satisfy

$$\sigma_{11}(c_{i,j}, \theta_{i,j}) \geq |\sigma_{12}(c_{i,j}, \theta_{i,j})|, \sigma_{22}(c_{i,j}, \theta_{i,j}) \geq |\sigma_{12}(c_{i,j}, \theta_{i,j})|, \sigma_{12}(c_{i,j}, \theta_{i,j}) \leq 0, \quad (17)$$

we approximate $u_{xy}(\mathbf{x}_{i,j})$ using

$$\frac{1}{2}(D_x^+ D_y^- + D_x^- D_y^+)u_{i,j} \equiv \frac{-2u_{i,j} - u_{i+1,j-1} - u_{i-1,j+1} + u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}}{2h^2}. \quad (18)$$

Figure 1 shows the stencil points of the 7-point stencil discretizations (16) and (18).

As a result, the discretization of the differential operator (13) at $\mathbf{x}_{i,j}$ reads

$$\begin{aligned} \mathcal{L}_{i,j}(c_{i,j}, \theta_{i,j}; u_h) \equiv & -\sigma_{11}(c_{i,j}, \theta_{i,j})D_x^+ D_x^- u_{i,j} - \sigma_{12}(c_{i,j}, \theta_{i,j})(D_x^+ D_y^{\pm} + D_x^- D_y^{\mp})u_{i,j} \\ & - \sigma_{22}(c_{i,j}, \theta_{i,j})D_y^+ D_y^- u_{i,j} + 2\sqrt{c_{i,j}(1 - c_{i,j})}f_{i,j}. \end{aligned} \quad (19)$$

3.2 Semi-Lagrangian Wide Stencil Discretization

However, if neither of Conditions (15) and (17) is fulfilled at the grid point $\mathbf{x}_{i,j}$, then it is unclear how to directly discretize the cross derivative $u_{xy}(\mathbf{x}_{i,j})$ in (13) monotonically. Our approach is to consider a **semi-Lagrangian wide stencil discretization** [15, 31]. Figure 2 illustrates the discretization process. More specifically, we consider eliminating the cross derivative $u_{xy}(\mathbf{x}_{i,j})$ by a local coordinate transformation. Let $\{(\mathbf{e}_z)_{i,j}, (\mathbf{e}_w)_{i,j}\}$ be a local orthogonal basis obtained by a clockwise rotation

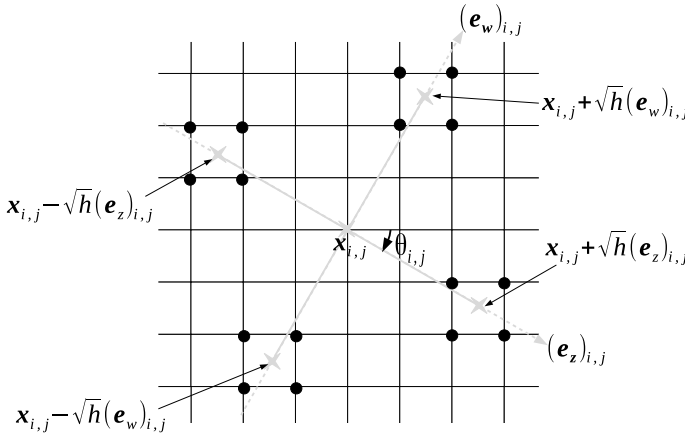


Fig. 2 Semi-Lagrangian wide stencil discretization at a grid point $\mathbf{x}_{i,j}$ inside the computational domain

of the standard axes $\{(\mathbf{e}_x)_{i,j}, (\mathbf{e}_y)_{i,j}\}$, as represented by the grey axes in Fig. 2. By straightforward algebra, one can show that if the rotation angle is

$$\frac{1}{2} \arctan \frac{2\sigma_{12}(c_{i,j}, \theta_{i,j})}{\sigma_{22}(c_{i,j}, \theta_{i,j}) - \sigma_{11}(c_{i,j}, \theta_{i,j})} = \theta_{i,j},$$

then the cross derivative vanishes under the basis $\{(\mathbf{e}_z)_{i,j}, (\mathbf{e}_w)_{i,j}\}$. As a result, (13) becomes

$$-c_{i,j} u_{zz}(\mathbf{x}_{i,j}) - (1 - c_{i,j}) u_{ww}(\mathbf{x}_{i,j}) + 2\sqrt{c_{i,j}(1 - c_{i,j})} f_{i,j} \quad (20)$$

Here $u_{zz}(\mathbf{x}_{i,j})$ and $u_{ww}(\mathbf{x}_{i,j})$ are the directional derivatives along the basis $(\mathbf{e}_z)_{i,j}$ and $(\mathbf{e}_w)_{i,j}$. We note that (20) still depends on $\theta_{i,j}$, as the basis $(\mathbf{e}_z)_{i,j}$ and $(\mathbf{e}_w)_{i,j}$ depend on $\theta_{i,j}$.

To discretize (20), one may consider applying the standard central differencing to $u_{zz}(\mathbf{x}_{i,j})$ and $u_{ww}(\mathbf{x}_{i,j})$. For instance, we approximate $u_{zz}(\mathbf{x}_{i,j})$ by

$$\frac{1}{h^2} [u(\mathbf{x}_{i,j} + h(\mathbf{e}_z)_{i,j}) - 2u_{i,j} + u(\mathbf{x}_{i,j} - h(\mathbf{e}_z)_{i,j})]. \quad (21)$$

However, since the stencil is rotated, the stencil points $\mathbf{x}_{i,j} \pm h(\mathbf{e}_z)_{i,j}$ may no longer coincide with any grid points. In such cases, we consider approximating $u(\mathbf{x}_{i,j} \pm h(\mathbf{e}_z)_{i,j})$ using bilinear interpolation from the neighboring grid points. However, a consequence of the bilinear interpolation is that the truncation error of (21) turns out to be $O(1)$, which is not consistent. In order to maintain consistency, we choose the

stencil length \sqrt{h} , which yields $O(h)$ truncation error. We note that the stencil length \sqrt{h} is greater than h , which gives rise to a “wide” stencil.

Under the stencil length \sqrt{h} , the new stencil points are $\mathbf{x}_{i,j} \pm \sqrt{h}(\mathbf{e}_z)_{i,j}$ and $\mathbf{x}_{i,j} \pm \sqrt{h}(\mathbf{e}_w)_{i,j}$, as represented by the grey stars in Fig. 2. The unknown values at these stencil points are approximated by the bilinear interpolation from their neighboring points, as represented by the black dots in Fig. 2. We denote these interpolated unknown values as $\mathcal{I}_h u|_{\mathbf{x}_{i,j} \pm \sqrt{h}(\mathbf{e}_z)_{i,j}}$ and $\mathcal{I}_h u|_{\mathbf{x}_{i,j} \pm \sqrt{h}(\mathbf{e}_w)_{i,j}}$. The finite difference discretizations for $u_{zz}(\mathbf{x}_{i,j})$ and $u_{ww}(\mathbf{x}_{i,j})$ are then given by

$$D_z^+ D_z^- u_{i,j} \equiv \frac{\mathcal{I}_h u|_{\mathbf{x}_{i,j} + \sqrt{h}(\mathbf{e}_z)_{i,j}} - 2u_{i,j} + \mathcal{I}_h u|_{\mathbf{x}_{i,j} - \sqrt{h}(\mathbf{e}_z)_{i,j}}}{h}, \quad (22)$$

$$D_w^+ D_w^- u_{i,j} \equiv \frac{\mathcal{I}_h u|_{\mathbf{x}_{i,j} + \sqrt{h}(\mathbf{e}_w)_{i,j}} - 2u_{i,j} + \mathcal{I}_h u|_{\mathbf{x}_{i,j} - \sqrt{h}(\mathbf{e}_w)_{i,j}}}{h}. \quad (23)$$

Finally, the discretization of the differential operator (13) at $\mathbf{x}_{i,j}$ reads

$$\mathcal{L}_{i,j}(c_{i,j}, \theta_{i,j}; u_h) \equiv -c_{i,j} D_z^+ D_z^- u_{i,j} - (1 - c_{i,j}) D_w^+ D_w^- u_{i,j} + 2\sqrt{c_{i,j}(1 - c_{i,j})} f_{i,j}. \quad (24)$$

We remark that here we have only discussed the scenario where $\mathbf{x}_{i,j}$ is well inside the computational domain. The scenario where $\mathbf{x}_{i,j}$ is near the boundary can be handled similarly.

3.3 Mixed Discretization

The advantage of the semi-Lagrangian wide stencil discretization (24) is that it is unconditionally monotone but it is only first order accurate. On the other hand, the standard 7-point stencil discretization is second order accurate. In order to combine the advantages of both discretization schemes, we will only apply the semi-Lagrangian wide stencil discretization at the grid points where neither (15) nor (17) is satisfied. Otherwise, the standard 7-point stencil discretization is applied. The resulting discretization method can be written as:

The significance of this mixed discretization is that monotonicity is strictly maintained at every grid point, and meanwhile, by using the standard 7-point stencil discretization as much as possible, the numerical scheme is as accurate as possible.

The mixed discretization scheme gives rise to a nonlinear discrete system which can be written in the following matrix form:

$$A_h(c_h^*, \theta_h^*) u_h = b_h(c_h^*, \theta_h^*), \quad (25)$$

$$\text{subject to } (c_h^*, \theta_h^*) \equiv \arg \max_{(c_h, \theta_h) \in \Gamma} \{A_h(c_h, \theta_h) u_h - b_h(c_h, \theta_h)\}, \quad (26)$$

where the matrix $A_h \in \mathbb{R}^{n_x n_y \times n_x n_y}$ and the vectors $u_h, c_h, \theta_h, b_h \in \mathbb{R}^{n_x n_y}$.

Algorithm 1 Mixed discretization for the HJB equation (11)-(12)

```

1: for  $i = 1, \dots, n_x$  do
2:   for  $j = 1, \dots, n_y$  do
3:     if  $(c_{i,j}, \theta_{i,j})$  satisfies Conditions (15) or (17) then
4:       The discrete equation at  $(i, j)$ ,  $\mathcal{L}_{i,j}(c_{i,j}, \theta_{i,j}; u_h)$ , is given by the standard 7-point stencil discretization (19)
5:     else
6:       The discrete equation at  $(i, j)$ ,  $\mathcal{L}_{i,j}(c_{i,j}, \theta_{i,j}; u_h)$ , is given by the semi-Lagrangian wide stencil discretization (24)
7:     end if
8:   end for
9: end for

```

4 Multigrid Methods

We will apply multigrid methods for solving (25). We start with multigrid methods for the standard 7-point stencil discretization. More precisely, we consider the case where the standard 7-point stencil discretization can be applied on the entire computational domain and still results in a monotone scheme. We will leave the discussion of multigrid for more general mixed stencil discretization to Sect. 4.3.

4.1 Policy-MG Iteration

One family of multigrid methods for solving the discretized HJB equation (25) is based on a global Newton-like iteration for the nonlinear system, called policy iteration (or Howard's algorithm) [19, 25]. At each policy iteration, a linear multigrid solver is applied to solve the linearized system. The algorithm can be written as follows:

Start with an initial guess of the solution $u_h^{(0)}$.

For $k = 0, 1, \dots$ until convergence:

1. Solve for the optimal control pair $(a_h^{(k)}, \theta_h^{(k)})$ under the current solution $u_h^{(k)}$:

$$(a_{i,j}^{(k)}, \theta_{i,j}^{(k)}) = \arg \max_{(a_{i,j}, \theta_{i,j}) \in \Gamma_{i,j}} \left\{ A_h(a_h, \theta_h) u_h^{(k)} - b_h(a_h, \theta_h) \right\}_{i,j}, \quad (27)$$

for all $\mathbf{x}_{i,j} \in \Omega$. Here $\Gamma_{i,j} = [0, 1] \times [-\frac{\pi}{4}, \frac{\pi}{4}]$ is the control set at $\mathbf{x}_{i,j}$.

Meanwhile, obtain the residual

$$r_h^{(k)} = A_h(a_h^{(k)}, \theta_h^{(k)}) u_h^{(k)} - b_h(a_h^{(k)}, \theta_h^{(k)}). \quad (28)$$

2. If $\|r_h^{(k)}\| \leq \text{tolerance}$: break

Else, use the multigrid V-cycle to solve the following linear system for the solution $u_h^{(k+1)}$ under the current optimal control pair $(a_h^{(k)}, \theta_h^{(k)})$:

$$A_h(a_h^{(k)}, \theta_h^{(k)}) u_h^{(k+1)} = b_h(a_h^{(k)}, \theta_h^{(k)}) \Rightarrow u_h^{(k+1)}. \quad (29)$$

To summarize, in order to solve (25), the inner multigrid V-cycle iteration for linearized problems is nested in an outer policy iteration. For convenience, we refer this type of multigrid methods as “policy-MG iteration”.

The advantage of using this approach is that policy iteration is guaranteed to converge for any initial guess $u_h^{(0)}$, if HJB equation is monotonically discretized [3, 7]. Policy iteration consists of two sub-steps. The first sub-step is to solve the optimization problem at each grid point $\mathbf{x}_{i,j}$; see (27). Our recent work [10] discusses speeding up computation of the optimization problem in details. The second sub-step of the policy iteration is to solve the linear system under a given control pair; see (29). The second sub-step is our focus of developing multigrid methods.

4.2 MG for 7-Point Stencil

The components of the standard multigrid include pointwise smoother, full coarsening, full-weighting restriction, bilinear interpolation and coarse grid operator (i.e., Galerkin coarse grid operator or direct discretization). However, the standard multigrid leads to a poor convergence for the HJB equation. We need to adapt each multigrid component to the HJB equation in order to achieve fast convergence.

4.2.1 Nonlinear Smoother

First, we discuss smoothers. We observe that (11) may become anisotropic. For instance, if $c^* = \epsilon$ is a small constant close to 0 and $\theta^* = 0$, then (11) becomes

$$-\epsilon u_{xx} - (1 - \epsilon) u_{yy} + 2\sqrt{\epsilon(1 - \epsilon)} f = 0,$$

which is an anisotropic Poisson equation. It is well-known that when solving anisotropic equations, the standard pointwise smoothers do not smooth errors along the weakly connected axis, which causes poor convergence rates [42].

To address anisotropy, we consider using line smoothers. More specifically, instead of updating the unknowns point by point, we update strongly-connected grid points collectively. In general, the strongly-connected direction of the 7-point discretization can change alignment to either the x -axis, or the y -axis, or the diagonal axes, in different parts of the computational domain. In view of this, we apply

four-direction alternating Gauss-Seidel line smoother. Thus, the line smoother is applied four times: along the x -axis (left to right), the y -axis (top to bottom), the diagonal axis (top left to bottom right) and the transpose diagonal axis (top right to bottom left). We summarize the nonlinear smoother in Algorithm 2.

Algorithm 2 Nonlinear four-direction alternating Gauss-Seidel line smoother

- 1: **subroutine** $\bar{u}_h = \text{SMOOTH}(u_h)$
 - 2: **for** $i = 1, \dots, n_x$ **do**
 - 3: **for** $j = 1, \dots, n_y$ **do**
 - 4: Update the control: $(\bar{c}_{i,j}, \bar{\theta}_{i,j}) = \arg \max_{(c_{i,j}, \theta_{i,j}) \in \Gamma} \mathcal{L}_{i,j}(c_{i,j}, \theta_{i,j}; u_h)$.
 - 5: **end for**
 - 6: **end for**
 - 7: Apply the one-step four-direction alternating Gauss-Seidel line smoother to the linearized system $A_h(\bar{c}_h, \bar{\theta}_h) u_h = b_h(\bar{c}_h, \bar{\theta}_h)$, which updates the solution $u_h \rightarrow \bar{u}_h$.
-

4.2.2 Restriction and Interpolation

Once the error becomes smooth along the x , y and diagonal axes after using the four-direction alternating line smoother, the standard full-coarsening can be applied. In order to capture the directional feature of the 7-point discretization, we follow [42] and apply 7-point restriction operators to (19). Using the stencil notation introduced, the corresponding 7-point restriction operators are given by

$$R^{[1]} = \frac{1}{8} \begin{bmatrix} 0 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \quad R^{[2]} = \frac{1}{8} \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad (30)$$

respectively. The interpolation operator is the scaled transpose of the restriction operator:

$$P = 4R^T. \quad (31)$$

4.3 MG for Mixed Discretization

In this section, we will discuss multigrid methods for the more general mixed discretization, where the semi-Lagrangian wide stencil discretization is applied to part of the computational domain. We will propose global linearization multigrid methods instead of FAS methods. One reason is that mixed discretization with wide stencils is a more difficult problem than the pure standard 7-point stencil discretization. We would like to use the Petrov-Galerkin coarse grid operators, which is more robust in

terms of the accuracy of the error estimate but is incompatible with the nonlinearity of FAS. Another reason, which will be shown, is that the coarse grids of our proposed approach are no longer square grids, which poses difficulties in defining an FAS coarse grid problem using direct discretization.

4.3.1 Issues

To start with a simple scenario, we consider solving the mixed discretization of the following linearized HJB equation:

$$\begin{aligned} \frac{1}{2}u_{xx} + \frac{1}{2}u_{yy} &= \sqrt{f}, \text{ in } \Omega \setminus \{(0, 0)\}, \\ \frac{2 + \sqrt{2}}{4}u_{xx} + \frac{2 - \sqrt{2}}{4}u_{yy} + \frac{1}{\sqrt{2}}u_{xy} &= 0, \text{ at } (0, 0), \\ u &= g, \text{ on } \partial\Omega. \end{aligned} \tag{32}$$

In other words, we assume that the control is given as $(c^*, \theta^*) = (\frac{1}{2}, 0)$ on the entire computational domain Ω , where the standard 7-point stencil discretization is applied, except that the control is $(c^*, \theta^*) = (1, \frac{\pi}{8})$ at the origin (the center of Ω), where wide stencil discretization is applied. Figure 3(ii) shows the error after applying the four-direction alternating line smoother. In particular, the cross section of the smoothed error shows that a kink appears at the origin (0,0). In general, wherever the wide stencil discretization is applied at a grid point, a kink appears in a smoothed error. Unfortunately, such kinks cannot be eliminated by other types of smoothers either.

4.3.2 Coarsening Strategy

Despite kink(s), Fig. 3(ii) shows that, after smoothing, kink(s) are restricted to the wide stencil point(s), and the error at the other grid points (i.e., the standard 7-point stencil points) is still smooth. This motivates us to apply full-coarsening to the standard 7-point stencil points, and consider a special type of coarsening strategy at the wide stencil points.

To motivate our coarsening strategy for wide stencils, we define a C -point as a fine grid point that is kept in its corresponding coarse grid; and an F -point otherwise. Let us first consider a one-dimensional cross section of a smoothed error; see Fig. 4(i). Black dots are C -points, while hollow dots are F -points. Assume that the standard full-coarsening assigns a wide stencil point (indicated by the red arrow) as an F -point. Let the black curves represent the underlying fine grid error. On the coarse grid, let its estimated error match the underlying fine grid error exactly, i.e., let the values of the black dots sit on the black curve. After linear interpolation of the coarse grid error, we obtain the interpolated error (grey curve) on the fine grid. Ideally, the interpolated error (grey curve) should match the underlying fine grid error (black

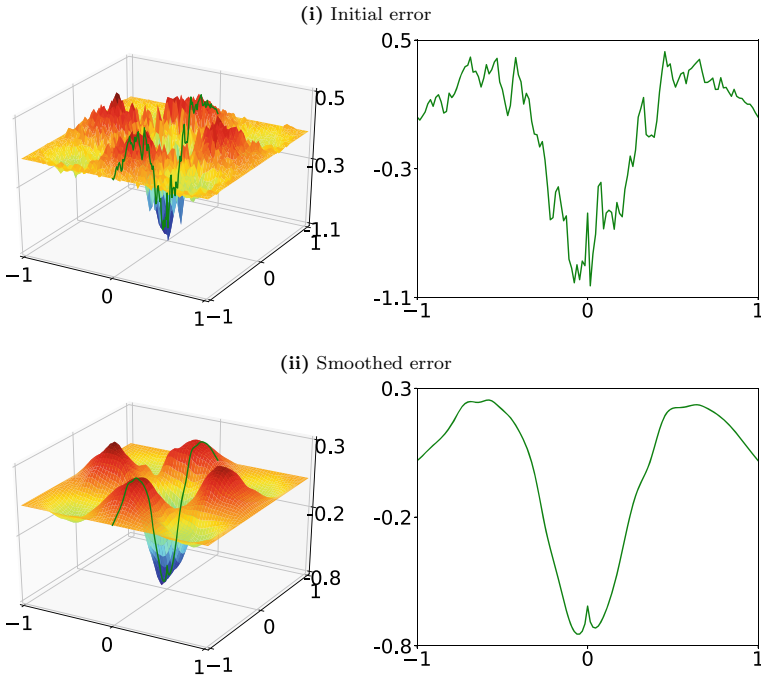


Fig. 3 The error after one step four-direction alternating Gauss-Seidel line smoothing. **(i)** Initial error and its cross section along the x -axis. **(ii)** Smoothed error and its cross section along the x -axis. A kink appears at the origin $(0,0)$

curve) as closely as possible. However, since the underlying fine grid error has a kink at the wide stencil point, the resulting interpolated error turns out to have a mismatch, as indicated by the red arrow. In other words, if the wide stencil point is an F -point, a linearly interpolated error will fail to capture the kink accurately.

Instead, our approach is simply setting the wide stencil F -point as a coarse grid point, i.e., a C -point; see Fig. 4(ii). As a result, interpolation at the wide stencil point is no longer needed. The error at the wide stencil point is simply copied from the coarse grid to the fine grid. This yields a more accurate fine grid estimated error, as indicated by the green arrow.

The above coarsening strategy can be extended to two dimensions. Figure 5 illustrates the coarsening process. On the fine grid, the black dots are selected as C -points, and the hollow dots are selected as F -points. Suppose wide stencils are applied to the three red dots. Then these three dots are all assigned as C -points. The resulting first coarse grid is a combination of a square grid that comes from geometric coarsening, and some additional coarse grid points that come from wide stencils. We can continue to coarsen the square sub-grid and meanwhile keep all the wide stencil points as C -points, which generates the second coarse grid. Such a coarsening strategy can be applied recursively until the coarsest level.

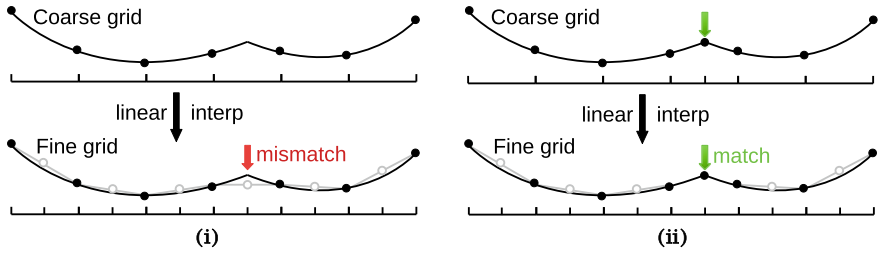


Fig. 4 Coarsening strategy at a wide stencil point. (i) Standard coarsening with linear interpolation at a wide stencil F -point (red arrow). (ii) Setting the wide stencil point as a coarse grid C -point (green arrow)

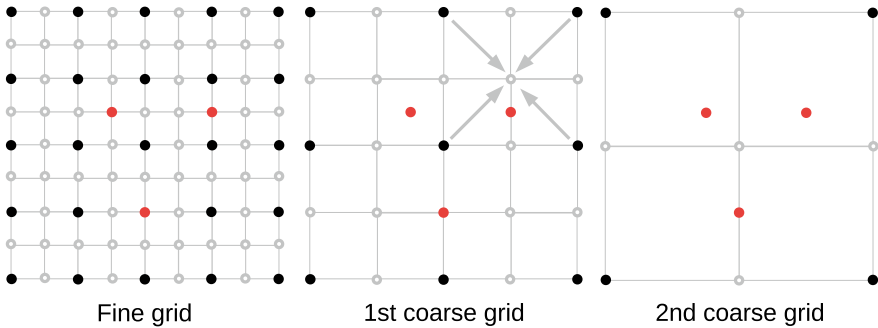


Fig. 5 Wide stencil grid points (red) are kept as C -points as the grid is coarsened from a fine grid to a coarse grid

One may argue that by setting all the wide stencil points as coarse grid points, the number of coarse grid points, and thus the computational complexity, will increase. However, it is observed in numerical simulations that wide stencils typically account for a negligible proportion of the total grid points in practical applications (such as image registration). Setting wide stencil points as coarse grid points would not result in a significant increase of the number of coarse grid points, and would still approximately maintain the square grid structure as the grid coarsens.

4.3.3 Interpolation

Under the proposed coarsening strategy, all the wide stencil points are excluded from the set of F -points. In other words, F -points must be the standard 7-point stencils. Hence, the 7-point interpolation, as described in Sect. 4.2.2, can be used for interpolating the errors at these F -points.

We note that the coarse grids are no longer square grids; see Fig. 5. However, each of these coarse grids can be seen as a combination of a square grid and some additional wide-stencil C -points. Then all the F -points can still be interpolated from

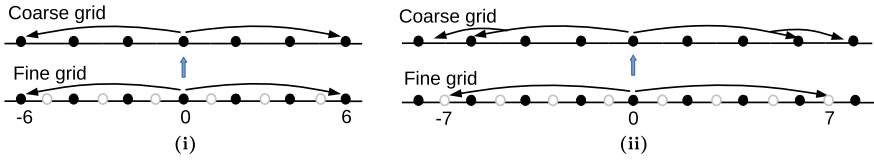


Fig. 6 Restriction for one-dimensional Poisson equation. (i) $h = \frac{1}{36}$ and $\sqrt{h} = 6h$. (ii) $h = \frac{1}{49}$ and $\sqrt{h} = 7h$

the C -points on the square grid. The arrows in Fig. 5 show how an F -point can be interpolated.

4.3.4 Restriction

In both the standard geometric and algebraic multigrid methods, restriction is simply the transpose of interpolation. However, it does not result in mesh-independent convergence rates for the non-symmetric matrices A_h arising from the mixed discretization. We will show such poor convergence in Sect. 5.2. Instead, we propose a restriction operator R that is different from the transpose of the interpolation P .

Our approach is simply to use **injection** on wide stencil points. To motivate the use of injection, let us simplify our problem and start with the one-dimensional Poisson equation

$$-u_{xx} = 0, \quad x \in [-0.5, 0.5]. \tag{33}$$

We apply the wide stencil discretization at $x = 0$ and the standard finite difference discretization on the rest of the computational domain. Figure 6 shows that under our coarsening strategy (which in this case is the same as the standard full coarsening), the fine grid points with even indices are C -points (black points), and the ones with odd indices are F -points (hollow points). The wide stencil point is $i = 0$. A naive choice of restriction at $i = 0$ would be the transpose of the linear interpolation, i.e., the standard full-weighting restriction:

$$r_0^H = \frac{1}{4}r_{-1} + \frac{1}{2}r_0 + \frac{1}{4}r_1, \tag{34}$$

where r_{-1}, r_0, r_1 are the fine grid residuals at $i = -1, 0, 1$, respectively, and r_0^H is the restricted residual at the coarse grid point. However, this leads to a poor coarse grid estimated error. In order to find a better restriction, we investigate two cases.

Case 1: $h = \frac{1}{36}$ and $\sqrt{h} = 6h$. Figure 6(i) shows that on the fine grid, the stencil points of $i = 0$ fall onto $i = \pm 6$. In this case, the wide stencil discretization at $i = 0$ reads

$$\frac{-u_{-6} + 2u_0 - u_6}{(6h)^2} = 0. \tag{35}$$

The residual at $i = 0$ is then given by

$$r_0 = \frac{-e_{-6} + 2e_0 - e_6}{(6h)^2}. \quad (36)$$

We notice that $i = 0$, $i = -6$ and $i = 6$ are all C -points. Then a natural construction of the coarse grid problem at $i = 0$ is to discretized the Poisson equation using these three points, or more precisely,

$$\frac{-e_{-6}^H + 2e_0^H - e_6^H}{(6h)^2} = r_0^H, \quad (37)$$

where the left hand side is a discretization of the Poisson equation on the coarse grid with the stencil length $6h$, and the right hand side is the coarse grid residual r_0^H . Comparing (36) and (37), we can see that the restriction at $i = 0$ is a simple injection:

$$r_0^H \equiv r_0. \quad (38)$$

Case 2: $h = \frac{1}{49}$ and $\sqrt{h} = 7h$. Figure 6(ii) shows that on the fine grid, the stencil points of $i = 0$ fall onto $i = \pm 7$. Unlike the previous case, here the two points $i = \pm 7$ are both F -points. To discretize the Poisson equation on the coarse grid, we interpolate the errors at $i = 7$ and $i = -7$ from their neighboring C -points, which gives

$$\frac{-\frac{1}{2}(e_{-8}^H + e_{-6}^H) + 2e_0^H - \frac{1}{2}(e_6^H + e_8^H)}{(7h)^2} = r_0^H. \quad (39)$$

We want to find a restriction, i.e., to rewrite r_0^H as a linear combination of fine grid residuals, such that it matches the left hand side of (39). One scheme is to use the linear combination of the following fine grid residuals:

$$r_0 = \frac{-e_{-7} + 2e_0 - e_7}{(7h)^2}, \quad r_7 = \frac{-e_6 + 2e_7 - e_8}{h^2}, \quad r_{-7} = \frac{-e_{-6} + 2e_{-7} - e_{-8}}{h^2}. \quad (40)$$

If we combine r_0 , r_7 and r_{-7} as follows

$$r_0 + \frac{1}{98}r_7 + \frac{1}{98}r_{-7} = \frac{-\frac{1}{2}(e_{-8} + e_{-6}) + 2e_0 - \frac{1}{2}(e_6 + e_8)}{(7h)^2}, \quad (41)$$

then (41) matches the left hand side of (39) in the exact sense. Equation (41) defines a possible restriction, i.e.,

$$r_0^H \equiv r_0 + \frac{1}{98}r_7 + \frac{1}{98}r_{-7}. \quad (42)$$

We note that the restriction (41) makes use of the residuals r_7 and r_{-7} , which are the points that the wide stencil point $i = 0$ connects to. This is different from the standard full weighting restriction (34), which uses the neighboring points r_1 and r_{-1} . Since the coefficients of r_7 and r_{-7} are small, we simply drop them from (42) and yield again an injection:

$$r_0^H \equiv r_0. \tag{43}$$

More generally, given a wide stencil C -point $i \in C$ with a stencil length \sqrt{h} , the non-zero restriction weights occur at the set of the F -points that it connects to, denoted as $\{j \mid j \in F, A_{i,j} \neq 0\}$. We can show that the restriction weights are

$$w_{i,j} = -\frac{A_{i,j}}{A_{j,j}} = -\frac{-\frac{1}{(\sqrt{h})^2}}{\frac{2}{h^2}} = \frac{h}{2}. \tag{44}$$

When h is small, the restriction (44) can be left out. In other words, injection is sufficient for a good coarse grid problem.

We extend the proposed injection at wide stencil C -points from the one-dimensional Poisson equation to the two-dimensional HJB equation. Note that the resulting restriction operator R_h is no longer the transpose of the interpolation. Once the restriction operator is specified, we construct the coarse grid operator by

$$A_{2h} \equiv R_h A_h P_h. \tag{45}$$

Since $R_h \neq P_h^T$, it results in the Petrov-Galerkin coarse grid operator.

The benefits of injection at wide stencil C -points are two-fold. One is that the resulting restriction operator and Petrov-Galerkin operator (45) are significantly sparser than their counterparts if other types of restriction operators are used (such as AMG restriction). This reduces the computational complexity. The other benefit is that such restriction would lead to an accurate coarse grid error estimate and eventually a mesh-independent convergence rate (Fig. 7).

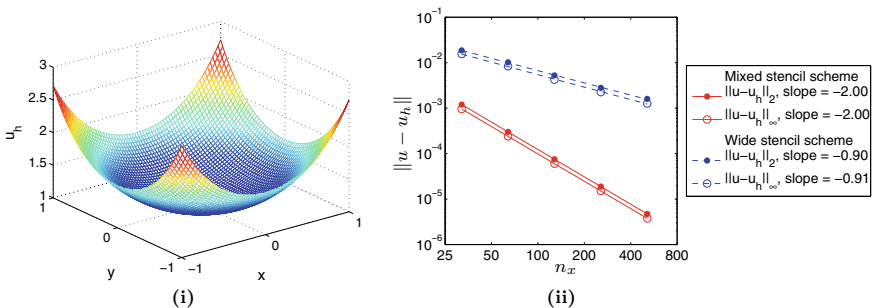


Fig. 7 Example 1: The exact solution is $u(x, y) = e^{\frac{1}{2}(x^2+y^2)}$. (i) Numerical solution. (ii) Norms of the errors $\|u - u_h\|$

5 Numerical Results

In this section, we demonstrate the mesh-independent convergence rates of the proposed multigrid methods for solving the discretized system (25)–(26). Details can be found in [10, 11].

5.1 Multigrid for Standard 7-Point Stencil Discretization

In Examples 1–2, the standard 7-point stencil discretization can be applied monotonically on the entire computational domain. We compare the performance of two families of multigrid methods - global linearization methods and full approximation scheme (FAS). For global linearization methods, the residual tolerances for the outer policy iteration and the inner multigrid V-cycle are 10^{-6} and 10^{-7} , respectively. The Gauss-Seidel smoother, the standard full coarsening and the 7-point restriction and interpolation are applied. The Petrov-Galerkin coarse grid operators are used to construct coarse grid problems. For FAS, the multigrid components are the same as the global linearization methods, except that we use the nonlinear version of the smoothers and direct discretization coarse grid operators.

Example 1 Consider solving the following equation:

$$\begin{aligned} u_{xx}u_{yy} - u_{xy}^2 &= f(x, y) = (1 + x^2 + y^2)e^{x^2+y^2}, & \text{in } \Omega, \\ u(x, y) &= g(x, y) = e^{\frac{1}{2}(x^2+y^2)}, & \text{on } \partial\Omega, \end{aligned}$$

where $\Omega = (-1, 1) \times (-1, 1)$. The exact solution $u(x, y) = e^{\frac{1}{2}(x^2+y^2)}$ is smooth.

This example is isotropic, so it suffices to apply the less expensive pointwise Gauss-Seidel smoother. First we show the convergence rates of the global linearization method; see the first and second columns of Table 1. To understand the reported numbers, we take the grid size of 32×32 as an example. The numbers “8, 7, 2” mean that it takes 3 policy iterations to converge to the solution of the nonlinear

Table 1 Convergence of the global linearization method and the FAS for Example 1

$n_x \times n_y$	Global linearization method		FAS
	Number of multigrid V-cycles within each policy iteration	Total number of multigrid V-cycles	Total number of multigrid V-cycles
32×32	8, 7, 2	17	8
64×64	9, 7, 3	19	8
128×128	9, 7, 3	19	9
256×256	9, 7, 3	19	9

problem, where the 1st policy iteration takes 8 V-cycles to converge to the solution of the linearized problem, the 2nd policy iteration takes 7 V-cycles, and the 3rd policy iteration takes 2 V-cycles. The table shows that the number of multigrid V-cycles within each policy iteration ranges from 2–9. The total number of multigrid V-cycles for solving the nonlinear problem is 17–19, independent of mesh size. As a side remark, we use the solution of the k -th policy iteration, $u_h^{(k)}$, as the initial guess of the multigrid V-cycles at the $(k + 1)$ -th policy iteration. Hence, as policy iteration converges, the initial guess of multigrid V-cycles becomes more and more precise, and the number of multigrid V-cycles within each policy iteration decreases.

We compare the global linearization method with the FAS iteration. The last column of Table 1 shows that the total number of the FAS iterations is 8–9 and is independent of mesh size. We note that for both the global linearization method and the FAS iteration, the computational cost per multigrid iteration is approximately the same. Hence, the FAS iteration is less expensive and converges faster.

Example 2 We consider the following equation:

$$\begin{aligned} u_{xx}u_{yy} - u_{xy}^2 &= f(x, y) = 1 + 24(x + y)^2, & \text{in } \Omega, \\ u(x, y) &= g(x, y) = \frac{1}{2}(x^2 + y^2) + (x + y)^4, & \text{on } \partial\Omega. \end{aligned}$$

The exact solution is $u(x, y) = \frac{1}{2}(x^2 + y^2) + (x + y)^4$.

Table 2 reports the convergence of the global linearization method using alternating line smoother and pointwise smoother. The multigrid V-cycle with the alternating line smoother converges at 20–32 iterations in total, which is approximately independent of mesh size. Conversely, the multigrid V-cycle with a pointwise smoother converges with more than 70 iterations, and the number of iterations is more than doubled as n_x increases from 32 to 256. This is because the example is anisotropic, and a pointwise smoother is not efficient in smoothing errors along weakly connected directions.

Similar to Example 1, we also compare the total numbers of multigrid V-cycles given by the global linearization method with the numbers given by the FAS. The

Table 2 Convergence of the global linearization method for Example 2 using alternating line smoother and pointwise smoother

$n_x \times n_y$	MG with alternating line smoother		MG with pointwise smoother
	Number of multigrid V-cycles within each policy iteration	Total number of multigrid V-cycles	Total number of multigrid V-cycles
32×32	5,5,5,3,2	20	73
64×64	5,6,6,4,2,1	24	94
128×128	6,6,7,5,3,1	28	129
256×256	7,7,7,6,3,1	32	161

Table 3 Total number of multigrid V-cycles of the global linearization method and the FAS for Example 2 using the alternating line smoother

$n_x \times n_y$	Global linearization method	FAS
32×32	20	5
64×64	24	6
128×128	28	6
256×256	32	6

alternating line smoother is used. Table 3 shows that the global linearization method converges in 20–32 iterations, whereas the FAS converges in 5–6 iterations, which is significantly faster.

5.2 Multigrid for Mixed Discretization

In this section, we illustrate the multigrid convergence rates for the mixed discretization. Thus, we apply four-direction alternating line smoother. At standard 7-point stencil points, we apply the standard full coarsening and the 7-point restriction and interpolation. At wide stencil points, we set them as coarse grid points, and use injection as the restriction. The Petrov-Galerkin coarse grid operators are used for constructing coarse grid problems.

Example 3 We consider solving the linearized HJB equation (32), where f and g are the same as in Example 1. Consider applying the wide stencil at the origin and the standard 5-point stencil discretization everywhere else. We compare the performance of our multigrid method (Scheme I), the standard multigrid with four-direction alternating line smoother (Scheme II), and the standard multigrid with pointwise Gauss-Seidel smoother (Scheme III). For this example, the only difference between Schemes I and II is that injection is applied at the wide stencil point for Scheme I, while full-weighting restriction is applied at the same point for Scheme II. Table 4 shows that Scheme III has poor convergence. Scheme II converges in less than 20 iterations, but the convergence rate grows as n_x increases. Scheme I converges in 5–6 iterations, and the convergence rate is independent of mesh size.

Figure 8 explains the convergence observed in Table 4 by examining the evolution of errors during one two-grid cycle. Only the cross sections along the x-axis are plotted. Start with the same initial error (green lines) for both our and the standard schemes. The pre-smoothed error (blue lines) is smooth everywhere, except that a kink appears at the wide stencil point $x = 0$. Figure 8(i) uses our algorithm, where injection is applied at the wide stencil point $x = 0$. The resulting coarse grid problem yields an accurate coarse grid estimated error, i.e., the red line matches the blue line well. Such accurate coarse grid estimate eliminates the error effectively, and yields

Table 4 Convergence of linear multigrid V-cycles for Example 3

$n_x \times n_y$	Scheme I: Our MG	Scheme II: Standard MG with alternating line smoother	Scheme III: Standard MG with pointwise smoother
32×32	5	7	23
64×64	5	9	46
128×128	6	12	198
256×256	6	17	more than 200

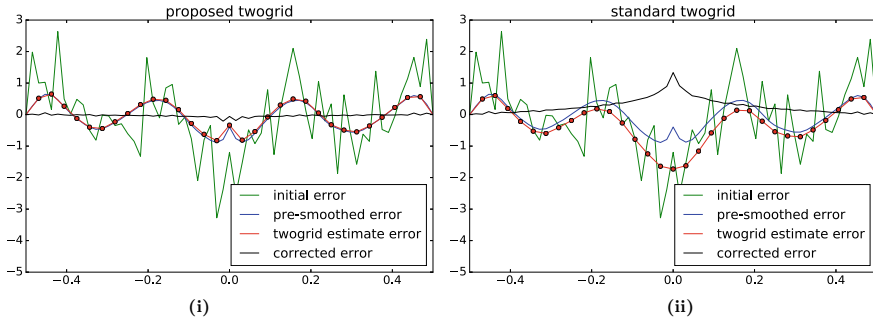


Fig. 8 Cross sections of errors along the x-axis. **(i)** Our algorithm, where injection is used at the wide stencil point $x = 0$. **(ii)** Standard algorithm, where full-weighting restriction is used

a small post-corrected error (black line). Conversely, under the same smoother, if the standard full-weighting is used at the wide stencil, then Fig. 8(ii) shows that the coarse grid estimated error (red line) is no longer a good approximation of the pre-smoothed error (blue line).

Example 4 We use the global linearization method to solve the Monge-Ampère equation as in Example 1, where

$$f(x, y) = \max \left(1 - \frac{0.15}{\sqrt{x^2 + y^2}}, 0 \right), \quad g(x, y) = \frac{1}{2}(\sqrt{x^2 + y^2} - 0.15)^2$$

on $\Omega = (-0.5, 0.5) \times (-0.5, 0.5)$. The viscosity solution is given by $u(x, y) = \frac{1}{2} \max(\sqrt{x^2 + y^2} - 0.15, 0)^2$. This is a C^1 function where the solution is not smooth at the ring $x^2 + y^2 = 0.15^2$. Semi-Lagrangian wide stencils are applied near the ring (Fig. 9).

Table 5 reports the convergence of the global linearization method. The number of outer policy iterations increases from 5 to 10 as n_x increases from 32 to 256. Such increase of outer iteration is related to nonlinearity and the singularity on the ring.

To compare the number of multigrid V-cycles across different mesh sizes fairly, we compute the average number of multigrid V-cycles per policy iteration. Table 5

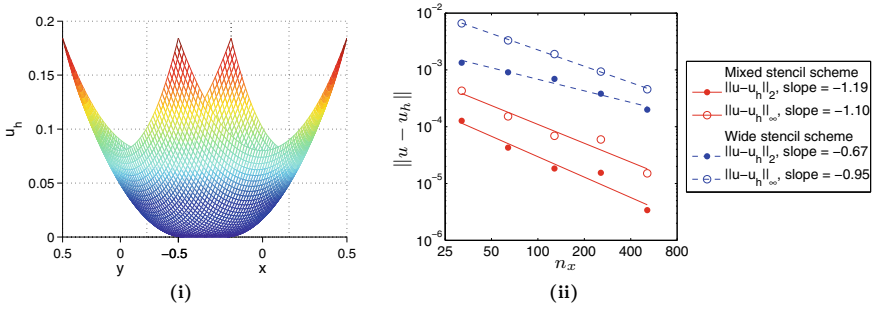


Fig. 9 Example 3: The exact solution is $\frac{1}{2} \max(\sqrt{x^2 + y^2} - 0.15, 0)^2$. (i) Numerical solution. (ii) Norms of the error $\|u - u_h\|$

Table 5 Convergence of the global linearization multigrid method for Example 4

$n_x \times n_y$	Number of multigrid V-cycles within each policy iteration	Average number of multigrid V-cycles per policy iteration
32×32	4,5,3,2,1	3.0
64×64	4,6,3,2,1	3.2
128×128	5,6,4,3,3,2	3.8
256×256	6,6,6,6,5,4,3,3,2,1	4.2

shows that the average V-cycle count is approximately a constant ranging from 3.0 to 4.2 as n_x increases from 32 to 256. Hence, the inner multigrid V-cycle for solving linearized systems is nearly mesh-independent.

6 Conclusion

This paper presents a numerical scheme for solving the mass transport registration model. In particular, we introduce a mixed standard 7-point stencil and wide stencil finite difference discretization. Furthermore, we present multigrid methods for solving the mixed discretization of the Monge-Ampère equation. We investigate two scenarios. One scenario is when the standard 7-point stencil discretization is applied on the entire computational domain. FAS gives the optimal mesh-independent convergence. The other scenario is the general mixed discretization. Global linearization method is used. We set all wide stencil points as coarse grid points and propose injection of residuals at wide stencil points. The resulting multigrid methods converge at mesh-independent rates.

References

1. J.B. Antoine Maintz, M.A. Viergever, A survey of medical image registration. *Medical Image Anal.* **2**(1), 1–36 (1998)
2. A. Ardehsir Goshtasby, *2-D and 3-D Image Registration: For Medical, Remote Sensing, and Industrial Applications* (Wiley, 2005)
3. P. Azimzadeh, P.A. Forsyth, Weakly Chained Matrices, Policy Iteration, and Impulse Control. *SIAM J. Numer. Anal.* **54**(3), 1341–1364 (2016)
4. G. Barles, P.E. Souganidis, Convergence of approximation schemes for fully nonlinear second order equations. *Asymptotic Anal.* **4**(3), 271–283 (1991)
5. J.-D. Benamou, Y. Brenier, K. Guittet, The Monge-Kantorovitch mass transfer and its computational fluid mechanics formulation. *Int. J. Numer. Methods Fluids* **40**(1–2), 21–30 (2002). ICFD Conference on Numerical Methods for Fluid Dynamics (Oxford, 2001)
6. J.-D. Benamou, B.D. Froese, A.M. Oberman, Two numerical methods for the elliptic Monge-Ampère equation. *M2AN Math. Model. Numer. Anal.* **44**(4), 737–758 (2010)
7. Olivier Bokanowski, Stefania Maroso, and Hasnaa Zidani. Some convergence results for Howard’s algorithm. *SIAM J. Numer. Anal.*, 47(4):3001–3026, 2009
8. C. Broit, *Optimal Registration of Deformed Images* (1981)
9. Rick Chartrand, Brendt Wohlberg, Kevin R. Vixie, and Erik M. Bollt. A gradient descent solution to the Monge-Kantorovich problem. *Appl. Math. Sci. (Ruse)*, 3(21–24):1071–1080, 2009
10. Y. Chen, J.W.L. Wan, J. Lin, Monotone mixed finite difference scheme for Monge–Ampère equation. *J. Sci. Comput.* **76**, 1839–1867 (2018)
11. Y. Chen, J.W.L. Wan, Multigrid methods for convergent mixed finite difference scheme for Monge-Ampère equation. *Comput. Visual. Sci.*, 1–15 (2017)
12. Y. Chen, J.W.L. Wan, Numerical method for image registration model based on optimal mass transport. *Inverse Probl. Imaging* **12**(2), 401–432 (2018)
13. S.N. Chow, W. Li, H. Zhou, A discrete Schrödinger equation via optimal transport on graphs. *Journal of Functional Analysis* **276**, 2440–2469 (2019)
14. G.E. Christensen, *Deformable shape models for anatomy*. Ph.D. thesis, Washington University Saint Louis, Mississippi (1994)
15. Kristian Debrabant and Espen R. Jakobsen. Semi-Lagrangian schemes for linear and fully non-linear diffusion equations. *Math. Comp.*, 82(283):1433–1462, 2013
16. P. Dupuis, U. Grenander, M.I. Miller, Variational problems on flows of diffeomorphisms for image matching. *Quart. Appl. Mathe.*, 587–600 (1998)
17. Xiaobing Feng, Roland Glowinski, and Michael Neilan. Recent developments in numerical methods for fully nonlinear second order partial differential equations. *SIAM Rev.*, 55(2):205–267, 2013
18. B. Fischer, J. Modersitzki, Fast inversion of matrices arising in image processing. *Numerical Algorithms* **22**(1), 1–11 (1999)
19. P.A. Forsyth, G. Labahn, Numerical methods for controlled Hamilton-Jacobi-Bellman PDEs in finance. *Journal of Computational Finance* **11**(2), 1 (2007)
20. Brittany D. Froese. A numerical method for the elliptic Monge-Ampère equation with transport boundary conditions. *SIAM J. Sci. Comput.*, 34(3):A1432–A1459, 2012
21. Brittany D. Froese and Adam M. Oberman. Convergent finite difference solvers for viscosity solutions of the elliptic Monge-Ampère equation in dimensions two and higher. *SIAM J. Numer. Anal.*, 49(4):1692–1714, 2011
22. S. Haker, A. Tannenbaum, Optimal mass transport and image registration, in *Variational and Level Set Methods in Computer Vision, 2001. Proceedings. IEEE Workshop on*, pp. 29–36. IEEE (2001)
23. S. Haker, L. Zhu, A. Tannenbaum, S. Angenent, Optimal mass transport for registration and warping. *International Journal of computer vision* **60**(3), 225–240 (2004)
24. D.L.G. Hill, P.G. Batchelor, M. Holden, D.J. Hawkes, Medical image registration. *Phys. Medicine Biol.* **46**(3), R1 (2001)

25. R.A. Howard, *Dynamic Programming and Markov Processes*. The Technology Press of M.I.T., Cambridge, Mass (Wiley, New York-London, 1960)
26. M. Irani, S. Peleg, Improving resolution by image registration. *CVGIP: Graph. Models Image Process.* **53**(3), 231–239 (1991)
27. M. Knott, C.S. Smith, On the optimal mapping of distributions. *J. Optim. Theory Appl.* **43**(1), 39–49 (1984)
28. N.V. Krylov, The control of the solution of a stochastic integral equation. *Teor. Veroyatnost. i Primenen.* **17**, 111–128 (1972)
29. P.-L. Lions, Hamilton-Jacobi-Bellman equations and the optimal control of stochastic systems, in *Proceedings of the International Congress of Mathematicians, Vol. 1, 2 (Warsaw, 1983)*, pp. 1403–1417. PWN, Warsaw (1984)
30. Lisa Gottesfeld Brown, A survey of image registration techniques. *ACM Comput. Surv. (CSUR)* **24**(4), 325–376 (1992)
31. K. Ma, P.A. Forsyth, An unconditionally monotone numerical scheme for the two factor uncertain volatility model. Preprint (2014)
32. J. Modersitzki, *Numerical Methods for Image Registration*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York, 2004. Oxford Science Publications
33. Oleg Museyko, Michael Stiglmayr, Kathrin Klamroth, and Günter Leugering. On the application of the Monge-Kantorovich problem to image registration. *SIAM J. Imaging Sci.*, **2**(4):1068–1097, 2009
34. G. Peyré, M. Cuturi, Computational optimal transport. *Foundations and Trends in Machine Learning* **51**(1), 1–44 (2019)
35. C. Reisinger, J.R. Arto, Boundary treatment and multigrid preconditioning for semi-Lagrangian schemes applied to Hamilton-Jacobi-Bellman equations. arXiv preprint [arXiv:1605.04821](https://arxiv.org/abs/1605.04821) (2016)
36. K. Rohr, *Landmark-Based Image Analysis: Using Geometric and Intensity Models*, vol. 21 (Springer Science & Business Media, 2001)
37. Louis-Philippe Saumier, Martial Agueh, and Boualem Khouider. An efficient numerical algorithm for the L^2 optimal transport problem with periodic densities. *IMA J. Appl. Math.*, **80**(1):135–157, 2015
38. M. Schmitz, M. Heitz, N. Bonneel, F.M. Ngole Mboula, D. Coeurjolly, M. Cuturi, G. Peyré, J-L. Starck, Wasserstein dictionary learning: Optimal transport-based unsupervised non-linear dictionary learning. *SIAM J. Imaging Sci.* **11**(1), 643–678 (2018)
39. Benjamin Seibold. Performance of algebraic multigrid methods for non-symmetric matrices arising in particle methods. *Numer. Linear Algebra Appl.*, **17**(2–3):433–451, 2010
40. A. Sotiras, C. Davatzikos, N. Paragios, Deformable medical image registration: A survey. *IEEE transactions on medical imaging* **32**(7), 1153–1190 (2013)
41. J.-P. Thirion, Image matching as a diffusion process: an analogy with maxwell’s demons. *Medical image analysis* **2**(3), 243–260 (1998)
42. U. Trottenberg, C.W. Oosterlee, A. Schüller., *textitMultigrid*. Academic Press, Inc., San Diego, CA, 2001. With contributions by A. Brandt, P. Oswald and K. Stüben
43. A. Trounev, Diffeomorphisms groups and pattern matching in image analysis. *International Journal of Computer Vision* **28**(3), 213–221 (1998)
44. P. Viola, W.M. Wells III, Alignment by maximization of mutual information. *Int. J. Comput. Vis.* **24**(2), 137–154 (1997)