arXiv:2602.05124v1 [math.NA] 4 Feb 2026

高维 Hamilton–Jacobi 方程的可扩展不动点框架*

YESOM PARK† 和 STANLEY OSHER‡

摘要。我们提出一种新颖的无网格、无梯度不动点方法，用于计算高维 Hamilton–Jacobi（HJ）方程的黏性解。借助 Hopf–Lax公式，我们通过 Picard 迭代反复求解相关变分问题，从而无需依赖网格、特征线或微分运算，即可高效计算解及其对应的控制。我们通过最高达 100维的数值实验，展示了该方法在实践中的有效性与可扩展性，实验涵盖控制问题和非光滑解。结果表明，所提出的方案精度高、效率出色，且计算时间基本不受维度影响，凸显了其适用于高维问题的优势。

关键词。Hamilton–Jacobi 方程，不动点迭代，高维问题，Hopf–Lax公式

MSC 分类号： 65N99, 65K10, 65N12, 93B40

1. 引言。Hamilton–Jacobi 偏微分方程（HJ方程）在数学、物理和工程的众多领域中发挥着核心作用，包括最优控制 [20, 48, 5], 力学 [19, 17], 以及动力系统的研究 [29, 47]。它们为描述受物理定律支配的系统提供了严谨的框架，并已应用于几何光学 [38, 36], 计算机视觉 [8, 26, 40], 机器人学 [33, 31, 4], 轨迹优化 [18, 44], 交通流建模 [27, 30], 和金融策略 [24, 7]等多个领域。这些应用中有许多自然会产生高维HJ方程，其中状态空间包含多个自由度或相互耦合的系统变量。这些问题固有的高维性给经典数值方法带来了严峻挑战。

传统的基于网格的方法，包括本质无振荡（ENO）和加权本质无振荡（WENO）格式 [42, 28, 6, 45], 半拉格朗日方法 [21, 12, 22],以及水平集方法 [41, 38, 39, 37, 2], ，在低维情形下能够达到较高精度，但严重依赖对状态空间进行离散化。随着维度增加，为获得精确近似所需的网格点数量呈指数级增长，导致计算代价高得令人难以承受。因此，对于许多高维控制问题，这些方法变得不可行，因为状态空间很容易超过十个维度。为应对这种维数灾难，研究者提出了其他方法，包括基于极大加代数的方法 [35, 1, 23]和基于 Hopf–Lax公式的方法 [16, 10, 11, 9], ，这些方法无需显式网格离散化，并通过在每个点求解优化问题来实现并行计算；不过，它们通常仅适用于特定类别的问题，而且往往需要满足凸性或其他结构性假设。

近期，基于神经网络的方法取得了进展 [49, 51, 46, 34, 32, 50] ，利用神经网络将解的信息存储在其参数中，从而提供可扩展的表示，能够部分缓解高维HJ方程所固有的维数挑战——

*提交给编辑的日期**资助：** S. Osher 和 Y. Park 获得了美国国防高级研究计划局 HR00112590074 和美国能源部 DE- SC0026262 的资助。S. Osher 还获得了美国国家科学基金会 2208272 的资助。

†加利福尼亚大学洛杉矶分校数学系（yeisom@math.ucla.edu）。
‡加利福尼亚大学洛杉矶分校数学系（sjo@math.ucla.edu）。

本手稿仅供审阅。

TABLE 1
Comparison of computational paradigms for solving Hamilton–Jacobi equations. The proposed fixed-point method performs 直接局部求值 *deep learning methods represent a* 全局近似 *requiring prior training, and WENO is an* 演化求解器 *advancing solutions causally in time.*

方法	计算类型	预计算	求值代价	可扩展性	收敛
基于网格的方法	演化型	网格生成	随网格规模呈指数	关于 d 呈指数级	有保证
深度学习方法	全局近似	训练开销大	低（前向网络传播）	关于 d 呈二次关系	No
不动点（我们的方法）	局部、无网格	None	与迭代次数呈线性关系	几乎与 d 无关	有保证

——在高维HJ方程中尤为突出。特别是，针对特定类别的HJ方程，开发专门的神经网络架构来表达变分公式或表示公式，已展现出良好的前景 [14, 15, 13]。此外，还有一种基于特征线的新方法，通过隐式公式高效计算黏性解 [43]。尽管取得了这些进展，这些方法通常缺乏收敛性和误差保证，或仅提供启发式保证，从而降低了所得解的可靠性。此外，随着问题维数增加，为反向传播而通过自动微分反复计算导数，会带来巨大的计算和内存开销。这些因素限制了神经网络求解器在超高维HJ方程中的实际可扩展性和效率；尤其是在真实应用中，精度与计算可行性都至关重要。

本文提出了一种新颖的不动点迭代框架，通过 Hopf–Lax公式求解高维 HJ方程。与传统的基于网格或神经网络的方法不同，我们的方法无网格、无因果性且无导数，因此尤其适用于状态维度较高的问题。通过将 Hopf–Lax公式重新表述为不动点问题，我们得到了一种简单的迭代方案，该方案收敛至唯一全局极小值点，从而无需空间离散化或反复求导，即可直接计算黏性解及相应的最优控制。重要的是，该算法实现简单，只需进行基本的迭代更新，因此非常便于实际应用。

在标准的凸性和 Lipschitz连续性假设下，我们还建立了严格的收敛性和误差界，适用于范围广泛的实际最优控制问题和高维 Hamilton–Jacobi问题。在这一框架下，该方法实现了前所未有的可扩展性，同时避免了网格、特征线追踪和导数计算。此外，在重叠特征线产生尖点、且不动点不再唯一的情况下，我们提出一种随机采样的多重初始化策略，以探索所有不动点，并可靠地识别出对应于黏性解的不动点。最高达 100维（包括非光滑问题和控制问题）的数值实验表明，该方法具有高精度、高效率，且计算代价几乎不受维度影响。

表1突出展示了我们的不动点方法、深度学习方法和经典基于网格的求解器之间的区别。与需要大量全局训练的深度学习方法，以及从 $t = 0$ 开始演化且具有因果性的基于网格的求解器不同，不动点方法支持在任意点 $(x,\ t)$进行直接局部求值，并保证收敛，且计算成本几乎与维度无关。总体而言，我们的框架弥补了现有经典数值格式和现代深度学习方法的不足：它将维度稳健的可扩展性 与可证明的精度相结合，为高维问题提供了理论可靠且实践高效的

本手稿仅供审阅。

用于高维HJ方程的方法论。

2. 方法。

2.1. 通过 Hopf–Lax 公式表示不动点。考虑哈密顿–雅可比（HJ）方程

$$\begin{cases} u_t + H(\nabla u) = 0, \\ u(x,0) = g(x), \end{cases} \tag{2.1}$$

其中 $H:\ \mathbb{R}^d \to \mathbb{R}$ 是凸（或凹）哈密顿量，$g:\ \mathbb{R}^d \to \mathbb{R}$ 是 Lipschitz 连续初始条件。众所周知，在适当的正则性假设下，方程 (2.1) 的唯一黏性解可通过 Hopf–Lax 公式表示：

$$u(x,t) = \inf_{y\in\mathbb{R}^n} \Big\{ t\, H^*\Big(\frac{x-y}{t}\Big) + g(y) \Big\}, \tag{2.2}$$

其中 H^* 表示 H 的勒让德变换。

虽然 Hopf–Lax 公式为该解提供了变分刻画，但要在实际中计算它，需要求出全局最小值点 y^*。对变分表达式进行形式微分后，我们得到关于该最小值点的如下不动点关系：

$$y = x - t\,\nabla H(\nabla g(y)), \tag{2.3}$$

这自然引出了用于近似 y^*的 Picard不动点迭代。具体而言，迭代过程

$$y^{(k+1)} = x - t\,\nabla H(\nabla g(y^{(k)})), \quad k = 0,1,2,\dots \tag{2.4}$$

实际实现时，可以预设最大迭代次数和收敛容差。详细的算法实现总结于算法2.1中。这种不动点视角在 HJ方程的变分表述与高效计算实现之间架起了一座桥梁，使得即使在最优控制等高维场景中，也能进行可扩展的求值。

算法2.1 Hamilton–Jacobi 方程的不动点迭代

要求： $x \in \mathbb{R}^d$, $t > 0$, $g:\ \mathbb{R}^d \to \mathbb{R}$, $H:\ \mathbb{R}^d \to \mathbb{R}$, $K_{\max} \in \mathbb{N}$ （最大迭代次数 ），$\varepsilon > 0$ （容差）

确保： $y^* \in \mathbb{R}^d$，Hopf–Lax公式的近似最小值点

初始化 $y^{(0)} \leftarrow x$ **对于** $k = 0$ **到** $K_{\max} - 1$ **执行** $y^{(k+1)} \leftarrow x - t\,\nabla H(\nabla g(y^{(k)}))$ **如果** $\|y^{(k+1)} - y^{(k)}\|_2 < \varepsilon$ **则** **跳出** **结束如果** **结束循环** $y^* \leftarrow y^{(k+1)}$ **返回** y^*

本文仅供审阅。

控制论解释。 不动点关系 (2.3) 从最优控制的观点来看具有一种自然的解释。回想一下，Hamilton–Jacobi 方程可以等价地表示为

$$u_t + H(\nabla u) = u_t + \sup_q \{\nabla u^\top q - L(q)\} = 0,$$

其中， L 是 H的勒让德变换。该最优控制问题对应的值函数为

$$u(x,t) = \inf_{q(\cdot)} \left\{ \int_0^t L(q(s))\,\mathrm{d}s + g(y(0)) : y(t) = x,\ \dot{y}(s) = q(s) \right\}. \tag{2.5}$$

在标准的凸性和可微性假设下，通过不动点迭代得到的最小值点 y^* 对应于式（2.5）中的最优初始状态 $y(0)$，相关的最优控制为

$$q^* = \frac{x - y^*}{t}. \tag{2.6}$$

因此，所提出的不动点方案不仅能够高效计算黏性解 $u(x,\ t)$，还可以直接获得相应的最优控制，无需进行特征线追踪或基于网格的离散化。

优势。 与传统的基于网格的数值求解器（例如加权本质无振荡格式）或近期的基于深度学习的方法相比，所提出的不动点方法在保持高精度的同时，显著加快了求值速度。其主要优势包括：

- **无网格且可扩展：** 无需进行空间离散化或特征线追踪，因此该方法非常适用于高维问题。
- **无导数：** 该方法避免了显式计算导数，因此计算开销低且具有出色的可扩展性。
- **无因果性且高度可并行：** 不动点更新不依赖相邻网格点的因果关系，因此形成无因果性格式，可在所有空间位置上进行易于并行的计算，并显著缩短现代硬件上的实际运行时间。
- **易于实现：** 该算法由简单的迭代更新构成，因此易于实现，也便于实际应用。
- **在高维情形下速度极快：** 逐点不动点评估能够大幅提升计算速度，即使在极高维情形下也能保持高效率。
- **控制恢复：** 与哈密顿–雅可比解相关的最优控制可直接由不动点极小化解 (2.6) 得到。
- **适用范围广：** 该格式适用于广泛的一类凸哈密顿量和凸的、Lipschitz 可微的初始数据，并可扩展至某些非光滑情形。

2.2. 收敛性分析。 本节将为所提出的不动点迭代方案 (2.4) 建立严格的收敛性、误差和复杂度保证。我们推导极小化器误差和黏性解近似误差的定量界，并进一步分析达到指定精度所需的迭代复杂度。这些结果共同为所提出方法的效率、稳定性和精度提供了全面的理论基础。

本文仅供审阅。

我们首先给出确保不动点存在且唯一，以及迭代全局收敛的充分条件。本节中，$\|\cdot\|$ 表示 L^2 范数。

THEOREM 2.1（定点收敛）。令 $H:\mathbb{R}^n\to\mathbb{R}$ 且 $g:\mathbb{R}^n\to\mathbb{R}$ b e 连续可微函数，使得：

- H 是严格凸的，且其梯度为 L_H-*Lipschitz* 连续：

$$\|\nabla H(p_1)-\nabla H(p_2)\|\le L_H\|p_1-p_2\|,\quad \forall p_1,p_2\in\mathbb{R}^n.$$

- g 是凸的，且其梯度为 L_g-*Lipschitz* 连续：

$$\|\nabla g(y_1)-\nabla g(y_2)\|\le L_g\|y_1-y_2\|,\quad \forall y_1,y_2\in\mathbb{R}^n.$$

定义不动点算子

$$F(y):=x-t\,\nabla H(\nabla g(y)).$$

那么，对于任意 $x\in\mathbb{R}^n$ 以及满足条件的任意时间步长 $t>0$

$$t\,L_H\,L_g<1,$$

该迭代

$$y^{(k+1)}=F(y^{(k)}),\qquad k=0,1,2,\dots$$

收敛到 *Hopf–Lax* 公式 (2.2) 的唯一全局极小值点。

证明。我们旨在证明 $F(y)=x-t\nabla H(\nabla g(y))$ 是一个压缩映射。令 $y_1,y_2\in\mathbb{R}^n$。

于是：

$$\begin{aligned}\|F(y_1)-F(y_2)\|&=\|x-t\nabla H(\nabla g(y_1))-(x-t\nabla H(\nabla g(y_2)))\|\\&=t\,\|\nabla H(\nabla g(y_1))-\nabla H(\nabla g(y_2))\|\\&\le t\,L_H\,\|\nabla g(y_1)-\nabla g(y_2)\|\\&\le t\,L_H\,L_g\,\|y_1-y_2\|.\end{aligned}$$

第一个不等式由 ∇H 的 L_H-Lipschitz 连续性得出，第二个不等式则由 ∇g 的 L_g-Lipschitz 连续性得出。

定义 $\lambda:=t\cdot L_H\cdot L_g$。于是：

$$\|F(y_1)-F(y_2)\|\le\lambda\|y_1-y_2\|.$$

由于根据假设， F 是完备度量空间 $\mathbb{R}^n$上的压缩映射。根据巴拿赫不动点定理，存在唯一的不动点 $y^*\in\mathbb{R}^n$ ，使得

$$y^*=F(y^*)=x-t\nabla H(\nabla g(y^*)).$$

现在我们证明，这个不动点 y^* 是霍普夫–拉克斯函数的唯一极小值点

$$\phi(y):=tH^*\left(\frac{x-y}{t}\right)+g(y).$$

由于 H^* 是严格凸且可微函数H的凸共轭，Hopf-Lax函数是可微的。因此，ϕ 的临界点条件为：

$$0\in\nabla\phi(y)=-\nabla H^*\left(\frac{x-y}{t}\right)+\nabla g(y).$$

本手稿仅供审阅。

等价地,

$$\nabla g(y) = \nabla H^*\left(\frac{x-y}{t}\right) \quad \Longleftrightarrow \quad y = x - t\nabla H(\nabla g(y)),$$

其中我们利用了恒等式 $\nabla H^* = (\nabla H)^{-1}$（在严格凸性条件下）。因此，不动点方程的任何解都是 ϕ 的临界点，反之亦然。

由于 H^* 是严格凸的（因为 H 是严格凸的），且 g 是凸的，因此它们的和 $\phi(y)$ 是严格凸的。因此，ϕ 具有唯一临界点，该临界点必然是 ϕ 的唯一全局极小值点。

因此，不动点 y^* 是 Hopf–Lax 泛函的唯一极小值点:

$$u(x,t) = \min_y \left\{ tH^*\left(\frac{x-y}{t}\right) + g(y) \right\} = \phi(y^*). \qquad \square$$

通过确立不动点迭代的收敛性，定理 2.1 为一种无梯度、无网格且无因果性的方法提供了理论基础，用于求解高维 Hamilton–Jacobi 方程的黏性解。

现在我们继续量化残差误差与最小值点及 Hopf–Lax 泛函值所得界之间的关系，从而为该方法的近似精度提供明确估计。

THEOREM 2.2（误差分析）。设 H 和 g 满足定理 2.1，$t > 0$，$L_F = tL_H L_g < 1$，$y^{(k+1)} =$的假设，且满足：。考虑迭代()，其中()(())，并将最小值点记为 。定义残差 $r^{(k)} := y^{(k+1)} - y^{(k)}$。则:

1. （***Hopf–Lax*** 最小值点误差）

$$\|y^{(k)} - y^*\| \le \frac{1}{1-L_F}\|r^{(k)}\|.$$

2. *(*解误差*)* 若 H^* 在相关定义域上是 L_{H^*}-*Lipschitz* 的，则

$$|u(x,t) - u^{(k)}(x,t)| \le \frac{L_{H^*}/t + L_g}{1-L_F}\,\|r^{(k)}\|,$$

其中 $u^{(k)}(x,t) := t\,H^*\left(\frac{x-y^{(k)}}{t}\right) + g(y^{(k)})$。

3. （解的梯度误差）

$$\|\nabla u(x,t) - \nabla u^{(k)}(x,t)\| \le L_g \|r^{(k)}\|.$$

证明。 **1. Hopf–Lax 最小值点误差界**。 根据定理 2.1，Hopf–Lax公式 $y^\star$ 的最小值点是 F的唯一不动点。由于 F 是压缩映射，压缩常数为 L_F，因此我们有

$$\|F(y) - F(y^*)\| \le L_F\|y - y^*\| \quad \forall y \in \mathbb{R}^n.$$

根据残差的定义,

$$r^{(k)} = y^{(k+1)} - y^{(k)} = F(y^{(k)}) - y^{(k)}.$$

然后,

$$\begin{aligned}
\|y^{(k)} - y^*\| &= \|y^{(k)} - F(y^*)\| \\
&= \|y^{(k)} - F(y^{(k)}) + F(y^{(k)}) - F(y^*)\| \\
&\le \|r^{(k)}\| + \|F(y^{(k)}) - F(y^*)\| \\
&\le \|r^{(k)}\| + L_F\|y^{(k)} - y^*\|.
\end{aligned}$$

本文仅供审阅。

重新整理可得

$$\|y^{(k)} - y^*\| \le \frac{1}{1 - L_F}\|r^{(k)}\|.$$

2. 解误差界。令

$$u(x,t) = tH^*\Big(\frac{x - y^*}{t}\Big) + g(y^*), \quad u^{(k)}(x,t) = tH^*\Big(\frac{x - y^{(k)}}{t}\Big) + g(y^{(k)}).$$

利用 H^* 和 g的 Lipschitz 连续性,

$$\begin{aligned}
|u(x,t) - u^{(k)}(x,t)| &\le t\left|H^*\Big(\frac{x - y^*}{t}\Big) - H^*\Big(\frac{x - y^{(k)}}{t}\Big)\right| + |g(y^*) - g(y^{(k)})| \\
&\le t\, L_{H^*}\frac{\|y^{(k)} - y^*\|}{t} + L_g\|y^{(k)} - y^*\| \\
&= (L_{H^*}/t + L_g)\|y^{(k)} - y^*\| \\
&\le \frac{L_{H^*}/t + L_g}{1 - L_F}\|r^{(k)}\|.
\end{aligned}$$

3. 解的梯度误差界。 对 Hopf–Lax公式关于 x 求导，得到

$$\nabla u(x,t) = \nabla H^*((x - y^*)/t).$$

结合隐式关系 (2.3)，可得

$$\nabla u(x,t) = \nabla H^*(\nabla H(\nabla g(y^*))) = \nabla g(y^*),$$

其中，最后一个等式源于这一事实： $\nabla H^* = \nabla H^{-1}$。因此，我们有

$$\begin{aligned}
\|\nabla u - \nabla u^{(k)}\| &= \Big\|\nabla g(y^{(k)}) - \nabla g(y^*)\Big\| \\
&\le L_g\|r^{(k)}\|.
\end{aligned}$$

证明至此完成。 □

定理 2.2 给出了残差范数与最小化误差及 Hopf–Lax近似误差之间关系的明确量化界。特别地，同时控制 Hopf–Lax最小化器 y^* 和解 u中的误差，等价于不仅控制黏性解本身的误差，还控制其梯度的误差。从控制理论的角度来看，这保证了误差不仅在值函数中得到控制，也在相关的最优控制中得到控制，从而为解和控制的精度提供了严格保证。这种等价性凸显了所提方法的一项关键优势。在实践中，该结果提供了严格且高效的终止准则：一旦残差 $\|r^{(k)}\|$低于容差 ε，就能保证解及其梯度（或者等价地，值函数及相关最优控制）中的相应误差保持在可控范围内。

现在，我们将刻画收敛速率以及达到预定误差阈值所需的迭代次数。

THEOREM 2.3（迭代复杂度）。在定理 *2.2* 的相同假设和记号下，对于任意初始猜测 $y^{(0)} \in \mathbb{R}^d$，达到残差范数 $\|r^{(k)}\| \le \varepsilon$ 所需的迭代次数满足

$$k \ge \frac{\log\Big(\frac{(1+L_F)\|y^{(0)} - y^*\|}{\varepsilon}\Big)}{-\log L_F}.$$

本稿仅供审阅。

特别地，不动点迭代以速率 L_F 线性收敛，因此迭代复杂度为

$$k = O\big(\log(1/\varepsilon)\big).$$

证明。 由于 F 是模数为 L_F 的压缩映射，因此有

$$\|y^{(k+1)} - y^*\| \le L_F \|y^{(k)} - y^*\|.$$

由归纳法可得，

$$\|y^{(k)} - y^*\| \le L_F^k \|y^{(0)} - y^*\|.$$

残差满足

$$r^{(k)} = y^{(k+1)} - y^{(k)} = (y^{(k+1)} - y^*) - (y^{(k)} - y^*),$$

因此

$$\|r^{(k)}\| \le \|y^{(k+1)} - y^*\| + \|y^{(k)} - y^*\| \le (1 + L_F) L_F^k \|y^{(0)} - y^*\|.$$

要实现 $\|r^{(k)}\| \le \varepsilon$，只需满足

$$(1 + L_F) L_F^k \|y^{(0)} - y^*\| \le \varepsilon.$$

求解 k ，得到

$$k \ge \frac{\log\Big(\frac{(1+L_F)\|y^{(0)} - y^*\|}{\varepsilon}\Big)}{-\log L_F}.$$

由于 –log $L_F > 0$ 且与 ε无关，我们有 $k = O(\log(1/\varepsilon))$，证明完毕。 □

结合定理 2.2 和 2.3 的结果，可以保证不动点迭代线性收敛，其收敛速率由压缩模 L_F 决定。尽管迭代次数的形式上界与问题维数 d 无关，但在实践中， L_F 可能随 d 增大，从而导致收敛速度适度减慢。尽管如此，这种与维度相关的影响远不如经典基于网格的格式或近期基于深度学习的方法中观察到的计算代价指数尺度增长严重，因此不动点方法对于高维哈密顿-雅可比问题具有很强的可扩展性。

2.3. 具有多重初始化的不动点求解器。一般而言，Hopf–Lax公式的临界点不一定唯一。即使哈密顿量 H 和初始条件 g 均为光滑函数，随着时间推移，特征线相交也可能导致 Hamilton–Jacobi 方程的黏性解呈现非光滑行为，进而形成尖点。在这种情况下，Hopf–Lax能量泛函可能存在多个驻点；等价地，不动点方程 (2.3) 可能有多个解。

为稳健地处理这种情况，我们对不动点迭代采用多重初始化策略。具体步骤总结如下：

1. **多次随机初始化：** 生成 N 个相互独立的随机初始猜测值 $\{y_0^{(i)}\}_{i=1}^N$，每个值均从超立方体 $[-\alpha, \alpha]^d$中按照预先指定的 $\alpha > 0$进行均匀采样。

本文仅供审阅。

2. **不动点迭代：** 对于每个初始值 $y_0^{(i)}$ ，执行以下迭代更新

$$y_{k+1}^{(i)} = G(y_k^{(i)}, t, x),$$

直至收敛，即，

$$\|y_{k+1}^{(i)} - y_k^{(i)}\|_2 < \varepsilon,$$

或直至达到最大迭代次数。

3. **能量评估：** 对于每个收敛的解 $y^{(i)}$，计算 Hopf– Lax 能量

$$\mathcal{E}(y^{(i)}; x, t) = t\, H^*\left(\frac{x - y^{(i)}}{t}\right) + g\Big(y^{(i)}\Big),$$

其中 H^* 表示 H 的勒让德变换。

4. **最优定点选择：** 在收敛的不动点中，选择

$$y^* = \arg\min_i \mathcal{E}(y^{(i)}; x, t),$$

这对应于 Hopf–Lax公式 (2.2) 中的最小值点。

这种多重初始化方法使算法能够探索多个潜在的不动点，并确保选定的解对应于真实的黏性解，即使存在尖点等非光滑特征也是如此。

THEOREM 2.4（多重初始化的概率完备性）。设 $G\colon \mathbb{R}^d \to \mathbb{R}^d$ 为连续映射，并假设它有一个有限的不动点集$\{y_1^*, \ldots, y_M^*\}$，满足

$$y_i^* = G(y_i^*), \quad i = 1, \ldots, M.$$

假设满足以下条件：

- （局部压缩）对于每个 i，存在一个开邻域 $\mathcal{B}_i \subset \mathbb{R}^d$，包含 y_i^* ，使得

$$\|G(y) - G(z)\| \le \kappa_i \|y - z\|, \qquad \forall y, z \in \mathcal{B}_i,$$

对某个 $\kappa_i \in [0, 1)$成立。因此，在迭代 $y_{k+1} = G(y_k)$下，每个 $\mathcal{B}_i$ 都是一个吸引域。

- （非零测度）每个 $\mathcal{B}_i$ 都具有正的勒贝格测度，即 $\mu（\mathcal{B}_i）> 0$。
- （独立采样）初始猜测 $y（1）0$、…、$y（N）0$分别独立地从概率分布 ρ 中抽取，该概率分布相对于勒贝格测度绝对连续，且其支集为满足$\mu（\Omega）> 0$的紧集 Ω。

于是，以概率 *1*，当 $N \to \infty$ 时，不动点迭代

$$y_{k+1}^{(j)} = G(y_k^{(j)}), \qquad j = 1, \ldots, N,$$

收敛到所有不动点 $\{y_i^*\}_{i=1}^M$。

证明。对于每个 i，记单次随机初始值落入吸引域 $\mathcal{B}_i$ 的概率为

$$p_i = \int_{\mathcal{B}_i} \rho(y)\, dy.$$

根据假设， ρ 绝对连续，且 $\mu(\mathcal{B}_i) > 0$，因此 $p_i > 0$。

本文仅供审阅。

N 个相互独立的初始值均落在 $\mathcal{B}_i$ 内的概率为$(1-p_i)^N$。因此，至少有一个初始值位于 $\mathcal{B}_i$ 内的概率为

$$1-(1-p_i)^N.$$

由于吸引域的数量有限（$M<\infty$），所有吸引域都被覆盖的概率为

$$\mathbb{P}_N=\prod_{i=1}^{M}\left(1-(1-p_i)^N\right).$$

当 $N\to\infty$ 时，每一项都满足 $\lim_{N\to\infty}(1-(1-p_i)^N)=1$，因此根据有限乘积的连续性，

$$\lim_{N\to\infty}\mathbb{P}_N=1.$$

这证明，随着概率趋近于 1，每个吸引域 $\mathcal{B}_i$ 至少会被采样一次。

在每个 $\mathcal{B}_i$ 内，局部压缩性质保证，对于任意初始值$y_0^{(j)}\in\mathcal{B}_i$，迭代 $y_{k+1}^{(j)}=G(y_k^{(j)})$ 都会根据巴拿赫不动点定理线性收敛到相应的不动点 y_i^*。

最后，由于每个不动点 y_i^* 都是 Hopf–Lax 能量 $\mathcal{E}$的驻点，因此，在 $\{y_i^*\}_{i=1}^M$ 中使 $\mathcal{E}$ 最小的点对应于 Hopf–Lax 变分公式的全局最小值点。由于几乎必然能找到所有 $\{y_i^*\}$（当 $N\to\infty$ 时），所选最小值点 y^* 几乎必然等于最优解。 □

理论上，要保证收敛到所有不动点，需要无穷多个初始点。然而，正如第 3 节所示，在实践中，有限个初始猜测便已足够。

3. 实验结果。 本节通过各种数值实验评估所提出的不动点方法的精度和效率。我们展示该方法在高维问题中的可扩展性，将其性能与现有求解器在精度和计算效率方面进行比较，并验证其恢复最优控制以及近似非光滑黏性解的能力。

所有实验均在一台 Linux 服务器上进行，该服务器配备 Intel Core i7-6850K CPU（6 核、12 线程，主频 3.6 GHz）、62 GB RAM 和 NVIDIA TITAN V GPU（12 GB）。为确保与现有深度学习方法进行公平比较，WENO 求解器和所提出的不动点方法均使用 Python 3 实现。

3.1. 高维问题。

3.1.1. Burgers 方程。 我们考虑如下 Hamilton–Jacobi 设置：

$$H(p)=\frac{1}{2}\|p\|_2^2,\qquad g(x)=\frac{1}{2}\|x\|_2^2, \tag{3.1}$$

它在原点处有唯一的临界点。精确解为$u(x,\ t)=\frac{\|x\|_2^2}{2}{}_{1+t}()$。使用不动点迭代

$$y^{(k+1)}=x-t\nabla H(\nabla g(y^{(k)})).$$

作为比较，我们还考虑五阶 WENO 格式这一具有代表性的基于网格的方法，以及 ImplicitHJ [43]，这一基于深度学习的方法，用于

本文仅供审阅。

TABLE 2

Comparison of numerical and gradient errors computation time and memory usage for dif- , , „ ferent dimensions using WENO ImplicitHJ and Ours methods.

Dim	方法	L^2 误差	L^∞ 误差	L^2 梯度误差	L^∞ 梯度误差	时间（秒）	内存（MB）
1D	WENO	1.84×10^{-10}	1.52×10^{-4}	3.06×10^{-5}	6.56×10^{-3}	0.34	0.05
	ImplicitHJ	4.88×10^{-10}	6.91×10^{-5}	1.70×10^{-6}	4.08×10^{-3}	156081.30	0.27
	Ours	2.43×10^{-17}	1.49×10^{-8}	6.37×10^{-15}	2.98×10^{-7}	0.034	0.001
2D	WENO	2.30×10^{-6}	8.62×10^{-3}	3.72×10^{-5}	1.58×10^{-1}	8.45	4.42
	ImplicitHJ	1.57×10^{-8}	3.40×10^{-4}	1.22×10^{-5}	8.95×10^{-3}	175223.30	0.34
	Ours	2.82×10^{-16}	5.96×10^{-8}	1.03×10^{-14}	2.98×10^{-7}	0.04	0.001
3D	WENO	7.22×10^{-3}	2.54×10^{-2}	5.46×10^{-3}	8.91×10^{-1}	1768.01	642.58
	ImplicitHJ	1.52×10^{-7}	1.06×10^{-3}	7.70×10^{-5}	2.46×10^{-2}	177784.40	0.42
	Ours	5.98×10^{-16}	1.19×10^{-7}	1.81×10^{-14}	2.98×10^{-7}	0.04	0.001
10D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	5.05×10^{-6}	6.62×10^{-3}	1.45×10^{-3}	3.83×10^{-2}	195997.48	0.95
	Ours	7.90×10^{-15}	2.38×10^{-7}	6.10×10^{-14}	3.58×10^{-7}	0.04	0.001
50D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	4.71×10^{-2}	3.67×10^{-1}	2.58×10^{-1}	2.53×10^{-1}	258987.28	4.01
	Ours	2.66×10^{-13}	1.91×10^{-6}	2.97×10^{-13}	3.58×10^{-7}	0.05	0.001
100D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	1.37×10^{-1}	4.91×10^{-1}	4.20×10^{-1}	5.46×10^{-1}	324249.12	7.82
	Ours	1.23×10^{-12}	3.81×10^{-6}	6.06×10^{-13}	3.58×10^{-7}	0.05	0.001

近似 HJ方程的黏性解。我们在 1、2、3、10、50 和 100维下评估性能。表 2 汇总了解及其梯度的 L^2 和 L^∞ 误差，以及计算时间和内存使用量。所有误差均在 128 个点上计算。

WENO 计算使用包含 128 个点的均匀网格，CFL 数为 0.2。ImplicitHJ 按照参考文献 [43], 中的原始配置进行训练，其计算时间对应于总训练时间。不动点方法最多迭代 1000 次，所有维度上的收敛容差均为 10^{-6}。为确保公平比较，所有方法均使用 Python 实现。

需要指出的是，这三种方法在设计原则和计算特性上存在根本差异。要进行完全公平的比较颇具挑战，尤其是在考察极高维问题时。在设置实验时，我们仔细确保以符合各方法预期用途的方式对其进行评估，同时保持不同维度下所报告指标的一致性。

表 2 中的结果表明，所提出的不动点方法在所有维度下均表现出色。值得注意的是，我们的方法始终能将解误差和梯度误差控制在接近机器精度的水平，而 WENO 和 ImplicitHJ 的误差则随着维度增加而增大。在三维及更高维度下，WENO 在计算上变得不可行，凸显了基于网格的方法所固有的维数灾难。ImplicitHJ 虽然具有可扩展性，但需要大量训练，并且在高维下会产生不可忽略的误差。

相比之下，不动点方法无网格、无因果性、无梯度且无需训练，因此能够天然扩展至超高维问题。该方法可直接计算解及其梯度，并在理论上保证精度，从而消除离散化或神经网络训练所带来的近似误差。此外，由于该方法不依赖稠密网格，也不依赖

本文仅供审稿使用。

随维度增加而表现不佳的迭代求解器，因此其计算时间和内存使用量在不同维度下几乎保持不变，这与定理 2.2 中确立的界一致。

这些结果凸显了不动点方法在实践中的优势：它能够为解及其梯度提供有理论依据的高精度近似，同时克服传统基于网格的方法和基于深度学习的方法的关键局限。与维度无关的效率和可证明的精度相结合，使我们的方法成为求解高维 Hamilton–Jacobi 方程的一种稳健且可扩展的替代方案。

值得注意的是，即使在 100 维的情况下，计算所得解的 L^2-误差也低至 2.06×10^{-14}，这表明所提出的不动点方案在高维场景中兼具高精度和可扩展性。

3.1.2. LQR 控制问题。我们考虑一种线性二次调节器（LQR）型最优控制问题，其中代价泛函关于控制和初始状态均为二次型：

$$u(x,t) = \min_{q(\cdot)} \left\{ \frac{1}{2} \int_0^t q(s)^\top R q(s)\, ds + \frac{1}{2} y(0)^\top Q y(0) \;\middle|\; \dot{y}(s) = q(s),\ y(t) = x \right\}, \tag{3.2}$$

其中， R 和 Q 是对称正定矩阵。相应的 Hamiltonian 具有各向异性二次型形式：

$$H(p) = \frac{1}{2} p^\top R^{-1} p.$$

该问题存在闭式解析解：

$$y^* = \left(I + tQR^{-1}\right)^{-1} x, \qquad u(x,t) = \frac{1}{2} x^\top \left(I + tQR^{-1}\right)^{-1} Q x. \tag{3.3}$$

所提出的不动点迭代

$$y^{(k+1)} = x - t\, y^{(k)} (R^{-1}Q)^\top$$

保证在以下条件下收敛：

$$t < \frac{1}{\|R^{-1}Q\|_2},$$

其中 $\|\cdot\|_2$ 表示谱范数。

对于给定的维度 d，我们使用随机高斯矩阵生成了 Q， $R \in \mathbb{R}^{d\times d}$ 个对称正定矩阵。具体而言，我们抽取了 A， $B \in \mathbb{R}^{d\times d}$ 个具有独立标准正态元素的矩阵，并构造

$$Q = s_1(A^\top A + I_d), \qquad R = s_2(B^\top B + I_d),$$

采用正缩放因子 s_1， $s_2 > 0$，确保 Q， $R \succ 0$。

我们在满足压缩条件 $t < 1/\|R^{-1}Q\|_2$ 的 100 个随机采样点 $(x,\ t)$ 上评估了不动点迭代，并记录了 L^2 和 L^∞误差以及计算时间。表 3 汇总了 Q 和 R 两组独立随机实现的结果。两组实验均表明，不动点迭代在不同维度下都具有较高精度和较低计算代价。

本文仅供审阅。

TABLE 3
Fixed-point results for the LQR control problem with two independent random realizations of Q and R. Errors are measured against the analytic solution.

Dim	第一随机集			第二随机集		
	L^2 误差	L^∞ 误差	时间（秒）	L^2 误差	L^∞ 误差	时间（秒）
1	6.70×10^{-17}	1.19×10^{-7}	0.0041	4.31×10^{-16}	5.96×10^{-8}	0.0045
10	3.26×10^{-12}	5.72×10^{-6}	0.0050	4.82×10^{-12}	7.63×10^{-6}	0.0052
50	2.21×10^{-9}	1.22×10^{-4}	0.0069	1.87×10^{-9}	1.22×10^{-4}	0.0080
100	3.84×10^{-8}	4.88×10^{-4}	0.0171	2.25×10^{-8}	3.66×10^{-4}	0.0104

TABLE 4
Fixed-point results for the cubical problem.

Dim	L^2 误差	L^∞ 误差	时间（秒）
1	4.53×10^{-9}	8.81×10^{-8}	0.0409
10	2.92×10^{-8}	1.25×10^{-7}	0.0458
50	1.73×10^{-7}	4.80×10^{-7}	0.0476
100	4.14×10^{-7}	1.15×10^{-6}	0.0561

3.1.3. 三次哈密顿问题。我们考虑具有非线性哈密顿量和初始条件的哈密顿-雅可比方程

$$H(p) = \frac{1}{3}\|p\|^3, \qquad g(x) = \|x\|^3,$$

其对应的勒让德变换为

$$H^*(q) = \frac{2}{3}\|q\|^{3/2}.$$

我们针对不同的维度 d 进行了最小化计算，并计算了相对于解析解的 L^2和最大误差。结果汇总于表 4。

结果表明，霍普夫–拉克斯最小化在不同维度下均能达到较高精度，随着维度增加，误差和计算时间仅略有上升。

3.2. 非光滑解。

3.2.1. 绝对值二次型。 我们考虑具有非光滑初始条件的 Burgers 方程

$$g(x) = x|x|,$$

这可以等价地表述为以下最优控制问题：

$$u(x,t) = \inf_{q(\cdot)} \left\{ \frac{1}{2}\int_0^t q(s)^2\, ds + g(y(0)) \;\middle|\; y(t) = x,\ \dot{y}(s) = q(s),\ 0 \le s \le t \right\}. \tag{3.4}$$

使用不动点迭代预测的解如图 1 所示。随着时间推移，可以观察到解中尖点的形成，这表明所提出的方法能够准确捕捉由非光滑初始数据产生的此类尖锐特征。

本手稿仅供审阅。

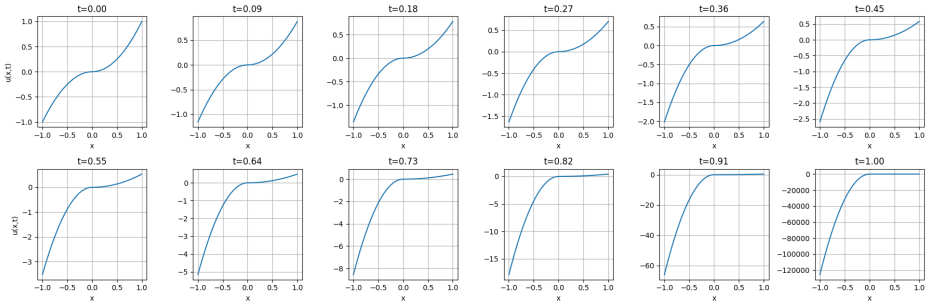

FIG. 1。

Solution of Burgers' equation with non-smooth initial condition g x = x|x| computed ( ) , via fixed-point iteration. Sharp gradients (kinks) develop as time evolves.

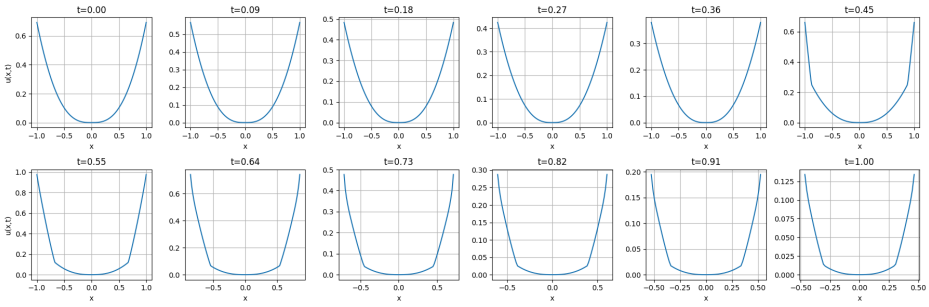

FIG. 2。 *Solution of Burgers' equation with non-smooth initial condition g (x) = x²log (1 + |x|) , computed via fixed-point iteration. Sharp gradients (kinks) develop as time evolves.*

3.2.2. 对数形式。我们考虑具有非光滑初始条件的 Burgers 方程

$$g(x) = x^2 \log(2 + |x|).$$

使用不动点迭代预测的解如图 2 所示。随着时间的推移，可以观察到解中尖点的形成，这表明所提出的方法能够准确捕捉由非光滑初始数据产生的此类尖锐特征。

3.3. 多重初始化。本节研究多重初始化方案在处理由相交特征线产生的尖点方面的有效性。

3.3.1. 稳态折点。考虑二次哈密顿量 $H(p) = \frac{1}{2}\|p\|_2^2$ 以及初始函数 g (x) $= -\|x\|_1$。精确解为

$$u(x,t) = -\|x\|_1 - \frac{dt}{2},$$

其中，稳态折点出现在 $x = 0$。这是由于特征线在区域 $-t < x < t$ 内重叠，导致该区域内的不动点具有非唯一性。在本实验中，我们考察第 2.3 节提出的多重初始化策略能否有效地从这些可能解中识别出与黏性解对应的不动点。

我们采用多重初始化的方法所得结果与 WENO 和 ImplicitHJ 的对比汇总于表 5 中。在实验中，初始值点的数量设为 {v1}维。

本文仅供审阅使用。

TABLE 5

Comparison of numerical and gradient errors computation time and memory usage for dif- , , „ ferent dimensions using WENO ImplicitHJ and Ours methods on the steady kink example.

Dim	方法	L^2 误差	L^∞ 误差	L^2 梯度误差	L^∞ 梯度误差	时间（秒）	内存（MB）
1D	WENO	1.51×10^{-7}	1.59×10^{-3}	2.17×10^{-4}	2.44×10^{-3}	0.35	0.07
	ImplicitHJ	8.59×10^{-6}	1.72×10^{-5}	2.01×10^{-5}	3.15×10^{-5}	210241.37	52.91
	Ours	1.54×10^{-14}	3.43×10^{-6}	2.15×10^{-13}	1.08×10^{-6}	25.97	3.86
2D	WENO	3.64×10^{-5}	1.02×10^{-2}	7.77×10^{-3}	7.45×10^{-1}	4.50	4.42
	ImplicitHJ	1.10×10^{-4}	2.05×10^{-4}	3.15×10^{-4}	5.10×10^{-4}	168770.98	52.99
	Ours	2.82×10^{-13}	1.13×10^{-5}	4.69×10^{-12}	5.47×10^{-5}	1.2	0.38
3D	WENO	5.38×10^{-5}	1.20×10^{-2}	9.21×10^{-2}	7.65×10^{-1}	1728.87	642.54
	ImplicitHJ	1.15×10^{-4}	2.05×10^{-4}	3.20×10^{-4}	5.15×10^{-4}	177594.47	53.06
	Ours	6.36×10^{-11}	2.41×10^{-4}	1.56×10^{-9}	4.60×10^{-4}	0.32	0.11
10D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	1.23×10^{-3}	2.45×10^{-3}	3.56×10^{-3}	6.32×10^{-3}	195088.43	53.60
	Ours	2.82×10^{-8}	1.13×10^{-3}	4.69×10^{-7}	8.41×10^{-3}	1.20	0.38
50D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	5.12×10^{-1}	1.02×10^{0}	1.08×10^{0}	1.05×10^{0}	242050.79	56.70
	Ours	1.72×10^{-5}	2.90×10^{-2}	6.25×10^{-3}	1.00×10^{0}	18.25	3.86
100D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	3.32×10^{-1}	6.65×10^{-1}	1.13×10^{0}	1.10×10^{0}	307335.52	62.61
	Ours	7.54×10^{-4}	3.43×10^{-2}	2.15×10^{-1}	1.00×10^{0}	25.97	3.86

从结果可以看出，多重初始化方法能够有效捕捉多个不动点，并准确逼近具有尖点的解。此外，与 WENO 相比，该方法展现出更好的可扩展性，同时比 ImplicitHJ 更准确、更稳定。尽管多重初始化会增加计算代价，我们的方法仍然比这两种基线方法都快。

3.3.2. 非定常尖点。现在考虑一个表现出非定常尖点的问题。设哈密顿量和初始函数分别为 $H(p) = \frac{1}{2}\|p\|_2^2$ 初始函数$g(\mathbf{x}) = \sum_{i=1}^{d} g_0(x_i)$，其中 $g_0(x) = \begin{cases} x, & \text{若 } 0 \\ \end{cases}$，

若 $x \geq 0.$ 精确解为

$$u(x,t) = \sum_{x_i < t/2} (x_i - t/2),$$

该解呈现出随时间移动的尖点，位于 $x = t/2$。在此问题中，特征线在区域 $0 < x < t$内重叠，导致不动点具有非唯一性。我们研究多重初始化方案能否有效捕捉此类移动尖点。

与 WENO 和 ImplicitHJ 相比，多重初始化方案的结果汇总于表6。与稳态折点情形类似，多重初始化不动点方法能够有效捕捉呈现非定常尖点的解。该方法展现出更强的可扩展性，且比 WENO 和 ImplicitHJ 更准确、更稳定。尽管采用多重初始化会增加计算成本，但我们的方法仍比这两种基准方法更快。

4. 结论。我们提出了一种新颖的不动点迭代框架，通过 Hopf–Lax公式求解高维 Hamilton–Jacobi 方程。该方法将变分问题重新表述为不动点方程，

本文仅供审稿使用。

TABLE 6

Comparison of numerical and gradient errors computation time and memory usage for dif- , , „ ferent dimensions using WENO ImplicitHJ and Ours methods on the non-steady kink example.

Dim	方法	L^2 误差	L^∞ 误差	L^2 梯度误差	L^∞ 梯度误差	时间（秒）	内存（MB）
1D	WENO	1.46×10^{-7}	2.02×10^{-3}	2.19×10^{-3}	2.40×10^{-1}	0.38	0.05
	ImplicitHJ	5.16×10^{-4}	1.33×10^{-1}	5.57×10^{-2}	9.57×10^{-1}	158959.15	0.27
	Ours	1.51×10^{-16}	1.39×10^{-7}	7.81×10^{-13}	2.10×10^{-4}	0.003	0.002
2D	WENO	6.84×10^{-5}	1.68×10^{-2}	2.25×10^{-3}	4.59×10^{-1}	9.072	4.43
	ImplicitHJ	1.01×10^{-4}	1.04×10^{-1}	1.62×10^{-2}	9.13×10^{-1}	169483.23	0.34
	Ours	3.17×10^{-15}	6.37×10^{-6}	7.81×10^{-10}	7.31×10^{-3}	0.21	0.08
3D	WENO	4.16×10^{-5}	1.76×10^{-1}	2.03×10^{0}	5.66×10^{0}	1734.16	642.54
	ImplicitHJ	2.70×10^{-3}	4.08×10^{-1}	6.89×10^{-2}	1.08×10^{0}	185356.57	0.42
	Ours	6.36×10^{-15}	2.41×10^{-5}	1.56×10^{-2}	8.27×10^{-1}	0.32	0.12
10D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	1.01×10^{-4}	3.02×10^{-2}	3.14×10^{-2}	6.32×10^{-1}	192581.22	0.95
	Ours	2.82×10^{-9}	1.13×10^{-3}	4.69×10^{-2}	1.00×10^{0}	1.20	0.38
50D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	2.01×10^{-4}	5.24×10^{-2}	1.08×10^{-1}	5.48×10^{-1}	256352.28	4.01
	Ours	1.72×10^{-6}	3.90×10^{-3}	6.25×10^{-2}	1.00×10^{0}	18.25	3.86
100D	WENO	N/A	N/A	N/A	N/A	N/A	N/A
	ImplicitHJ	3.32×10^{-1}	1.57×10^{0}	1.13×10^{0}	1.57×10^{0}	321914.77	7.82
	Ours	1.54×10^{-4}	3.43×10^{-2}	2.15×10^{-1}	1.00×10^{0}	25.97	3.86

无需进行空间离散化、特征线追踪或反复求导，即可直接以无网格方式求得黏性解及相关的最优控制。我们的理论分析表明，在标准的凸性和 Lipschitz连续性假设下，该方法能够收敛并满足误差界，从而确保精度和稳定性。尽管凸性要求在一定程度上限制了该方法可直接应用的问题类别，但在最高达 100 维的问题上进行的数值实验表明，该方法精度高、计算代价低，且可扩展性几乎不受维度影响；无论效率还是可靠性，均优于经典的基于网格的求解器和近期的深度学习方法。总体而言，该不动点框架弥补了现有经典数值格式和现代深度学习方法的不足，为高维 Hamilton–Jacobi问题提供了一种可扩展、精度有理论保证且计算效率高的方法，可用于最优控制、机器人学、金融及其他涉及高维偏微分方程的领域。

未来研究仍有若干方向。将该方法扩展至非凸或非 Lipschitz 哈密顿量，可拓宽其在更广泛问题中的适用性。加入扩散项，将该方法扩展至二阶哈密顿–雅可比方程，有望使其应用于随机控制、平均场博弈及相关领域。此外，加入自适应步长控制、动量、锚定、安德森加速或拟牛顿更新，可能进一步提升收敛速度和稳健性，尤其是在超高维场景中。最后，将不动点方法与隐式神经网络架构相结合 [3, 25] ，或可为求解反问题及其他复杂高维任务开辟新途径，从而进一步扩大该方法的实际影响。

参考文献

本文仅供审阅。

[1] M. Akian, S. Gaubert, and A. Lakhoua, *The max-plus finite element method for solving deterministic optimal control problems: basic properties and convergence analysis*, SIAM 控制与优化杂志, 47 (2008), pp. 817–848. [2] M. Ansari, A. Khajepour, and E. Esmailzadeh, *Application of level set method to optimal vibration control of plate structures* 332 2013 687 , 《声与振动杂志》, ( ), pp. – 700. [3] S. Bai, J. Z. Kolter, and V. Koltun, *Deep equilibrium models*, 神经信息处理系统进展, 32 (2019). [4] S. Bansal, A. Bajcsy, E. Ratner, A. D. Dragan, and C. J. Tomlin, *A hamilton-jacobi reachability-based framework for predicting and analyzing human motion for safe planning*, 载于 2020年IEEE国际机器人与自动化会议 (ICRA), IEEE, 2020, pp. 7149–7155. [5] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, *Hamilton-jacobi reachability: A briefoverview and recent advances*, 载于 2017年第56届IEEE决策与控制年会 (CDC), IEEE, 2017, pp. 2242–2253. [6] S. Bryson and D. Levy, *High-order central weno schemes for multidimensional hamilton-jacobi equations* 41 2003 1339 1369 , SIAM 数值分析杂志, ( ), pp. – . [7] Y. Cao and N. Wan, *Optimal proportional reinsurance and investment based on hamilton–jacobi–bellman equation* 45 2009 157 162 , 《保险：数学与经济学》, ( ), pp. – . [8] V. Caselles, R. Kimmel, and G. Sapiro, *Geodesic active contours*, 《国际计算机视觉杂志》, 22 (1997), pp. 61–79. [9] P. Chen, J. Darbon, and T. Meng, *Hopf-type representation formulas and efficient algorithms for certain high-dimensional optimal control problems*, 《计算机与应用数学》, 161 (2024), pp. 90–120. [10] Y. T. Chow, J. Darbon, S. Osher, and W. Yin, *Algorithm for overcoming the curse of dimensionality for time-dependent non-convex hamilton–jacobi equations arising from optimal control and differential games problems*, 《科学计算杂志》, 73 (2017), pp. 617–643. [11] Y. T. Chow, J. Darbon, S. Osher, and W. Yin, *Algorithm for overcoming the curse of dimensionality for state-dependent hamilton-jacobi equations*, 《计算物理学杂志》, 387 (2019), pp. 376–409. [12] E. Cristiani and M. Falcone, *Fast semi-lagrangian schemes for the eikonal equation and applications* 45 2007 1979 2011 , SIAM 数值分析杂志, ( ), pp. – . [13] J. Darbon, P. M. Dower, and T. Meng, *Neural network architectures using min-plus algebra for solving certain high-dimensional optimal control problems and hamilton–jacobi pdes* , 《控制、信号与系统数学》, 35 (2023), pp. 1–44. [14] J. Darbon, G. P. Langlois, and T. Meng, *Overcoming the curse of dimensionality for some hamilton–jacobi partial differential equations via neural network architectures* , 《数学科学研究》, 7 (2020), p. 20. [15] J. Darbon and T. Meng, *On some neural network architectures that can represent viscosity solutions of certain high dimensional hamilton–jacobi partial differential equations* , 《计算物理学杂志》, 425 (2021), p. 109907. [16] J. Darbon and S. Osher, *Algorithms for overcoming the curse of dimensionality for certain hamilton–jacobi equations arising in control theory and elsewhere* , 《数学科学研究》, 3 (2016), p. 19. [17] M. de León, J. C. Marrero, and D. M. de Diego, *Linear almost poisson structures and hamilton-jacobi equation. applications to nonholonomic mechanics* , arXiv 预印本 arXiv:0801.4358, (2008). [18] D. Delahaye, S. Puechmorel, P. Tsiotras, and E. Féron, *Mathematical models for aircrafttrajectory design: A survey*, 载于《空中交通管理与系统：第三届ENRI空中交通管理/通信导航监视国际研讨会（EIWAC2013）论文选集》, 施普林格, 2014, pp. 205–247. [19] H. H. Denman and L. H, Buch, *Solution of the hamilton-jacobi equation for certain dissipative classical mechanical systems* 14 1973 326 329 , 《数学物理学杂志》, ( ), pp. – . [20] L. C. Evans and P. E. Souganidis, *Differential games and representation formulas for solutions of hamilton-jacobi-isaacs equations*, 《印第安纳大学数学杂志》, 33 (1984), pp. 773–797. [21] M. Falcone and R. Ferretti, *Semi-lagrangian schemes for hamilton–jacobi equations, discrete representation formulae and godunov methods*, 《计算物理学杂志》, 175 (2002), pp. 559–575. [22] M. Falcone and R. Ferretti, *Semi-Lagrangian approximation schemes for linear and HamiltonJacobi equations* 2013 , SIAM, .

本文仅供审阅。

[23] W. H. Fleming and W. M. McEneaney, *A max-plus-based algorithm for a hamilton–jacobi–bellman equation of nonlinear filtering*, 《SIAM 控制与优化杂志》, 38（2000）, 第 683–710 页。
[24] P. A. Forsyth and G. Labahn, *Numerical methods for controlled hamilton-jacobi-bellman pdes in finance* 11 2007 1 , 《计算金融杂志》, （）, 第 页。[25] S. W. Fung, H. Heaton, Q. Li, D. McKenzie, S. Osher, and W. Yin, *Jfb: Jacobian-free backpropagation for implicit networks*, 载于《AAAI 人工智能大会论文集》, 第 36 卷, 2022 年, 第 6648–6656 页。
[26] G. Gilboa and S. Osher, *Nonlocal operators with applications to image processing*, 《多尺度建模与仿真》, 7（2009）, 第 1005–1028 页。[27] C. Imbert, R. Monneau, and H. Zidani, *A hamilton-jacobi approach to junction problemsand application to traffic flows*, 《ESAIM：控制、优化与变分法》, 19（2013）, 第 129–166 页。[28] G.-S. Jiang and D. Peng, *Weighted eno schemes for hamilton–jacobi equations*, 《SIAM 科学计算杂志》, 21（2000）, 第 2126–2143 页。[29] K. Khanin and A. Sobolevski, *Particle dynamics inside shocks in hamilton–jacobi equations*, 《英国皇家学会哲学汇刊 A 辑：数学、物理与工程科学》, 368（2010）, 第 1579–1593 页。[30] J. A. Laval and L. Leclercq, *The hamilton–jacobi partial differential equation and the threerepresentations of traffic flow*, 《交通研究 B 辑：方法论》, 52（2013）, 第 17–30 页。[31] F. L. Lewis, D. M. Dawson, and C. T. Abdallah, *Robot manipulator control: theory and practice* 2003 , CRC 出版社, 。
[32] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar , *Fourier neural operator for parametric partial differential equations*, arXiv 预印本 arXiv:2010.08895, （2020）。
[33] F. Lin and R. D. Brandt, *An optimal control approach to robust control of robot manipulators*, 《IEEE 机器人与自动化汇刊》, 14（1998）, 第 69–77 页。[34] L. Lu, P. Jin, and G. E. Karniadakis, *Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators*, arXiv 预印本 arXiv:1910.03193, （2019）。[35] W. M. McEneaney, *Max-plus methods for nonlinear control and estimation*, 第 2 卷, 施普林格科学与商业媒体, 2006 年。[36] J. C. Miñano, P. Benítez, and A. Santamaría, *Hamilton-jacobi equation in momentum space* 14 2006 9083 9092 , 《光学快报》, （）, 第 – 页。[37] I. Mitchell, A. Bayen, and C. J. Tomlin, *Computing reachable sets for continuous dynamic games using level set methods* 2004 , 1 月提交, （）。
[38] S. Osher, *A level set formulation for the solution of the dirichlet problem for hamilton–jacobi equations* 24 1993 1145 1152 , 《SIAM 数学分析杂志》, （）, 第 – 页。[39] S. Osher, R. Fedkiw, and K. Piechor, *Level set methods and dynamic implicit surfaces*, 《应用力学评论》, 57（2004）, 第 B15–B15 页。
[40] S. Osher and N. Paragios, *Geometric level set methods in imaging, vision, and graphics*, 施普林格科学与商业媒体, 2007 年。[41] S. Osher and J. A. Sethian, *Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations* 79 1988 12 , 《计算物理学杂志》, （）, 第 – 49 页。[42] S. Osher and C.-W. Shu, *High-order essentially nonoscillatory schemes for hamilton–jacobi equations* 28 1991 907 922 , 《SIAM 数值分析杂志》, （）, 第 – 页。[43] Y. Park and S. Osher, *Neural implicit solution formula for efficiently solving hamilton-jacobi equations* 2025 2501 19351, 2501 19351 , （）, https://arxiv.org/abs/ 。https://arxiv.org/abs/ 。[44] C. Parzani and S. Puechmorel, *On a hamilton-jacobi-bellman approach for coordinated opti-mal aircraft trajectories planning*, 《最优控制应用与方法》, 39（2018）, 第 933–948 页。[45] J. Qiu and C.-W. Shu, *Hermite weno schemes for hamilton–jacobi equations*, 《计算物理学杂志》, 204（2005）, 第 82–99 页。
[46] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations* 378 2019 686 707 , 《计算物理学杂志》, （）, 第 – 页。[47] S. Rajeev, *A hamilton–jacobi formalism for thermodynamics*, 《物理学年鉴》, 323（2008）, 第 2265–2285 页。[48] A. Sideris and J. E. Bobrow, *An efficient sequential linear quadratic algorithm for solving nonlinear optimal control problems*, 载于《2005 年美国控制会议论文集》,

本文仅供审阅。

论文集，2005 年，IEEE，2005 年，第 2275–2280 页。[49] J. Sirignano and K. Spiliopoulos, *Dgm: A deep learning algorithm for solving partial differential equations* 375 2018 1339 1364 ，《计算物理学杂志》，（），第 – 页。[50] L. Yang, S. Liu, T. Meng, and S. J. Osher, *In-context operator learning with data prompts for differential equation problems*，《美国国家科学院院刊》，120 (2023)，第 e2310142120 页。[51] B. Yu et al., *The deep ritz method: a deep learning-based numerical algorithm for solving variational problems* 6 2018 1 12 ，《数学与统计学通信》，（），第 – 页。

本文仅供审稿使用。